

Uncommon and Advance Techniques for Account Takeover Attacks

Ayoub Safa @sandh0t

About me

- **Engineering Degree in Computer Science and Networking**
- **Pen Tester with 10 years (OSCP, OSCE, GXPN)**
- **Senior Security Consultant @ MDSec**
- **Bug Bounty Hunter @ HackerOne**
- **Twitter: @sandh0t**

Disclaimers

- Please don't break the law
- Play Nice, Be Ethical
- My opinions are my own

What is ATO?

Account Takeover (ATO) is an Attack where an unauthorized individual gains control over a user's account,

enabling them to they can impersonate the legitimate user, manipulate account settings, perform fraudulent activities, access sensitive information

Why ATO?

- Critical/High Security Impact
- Evading Firewall Detection
- Multiple Ways of ATO
- Handsomely Rewarded

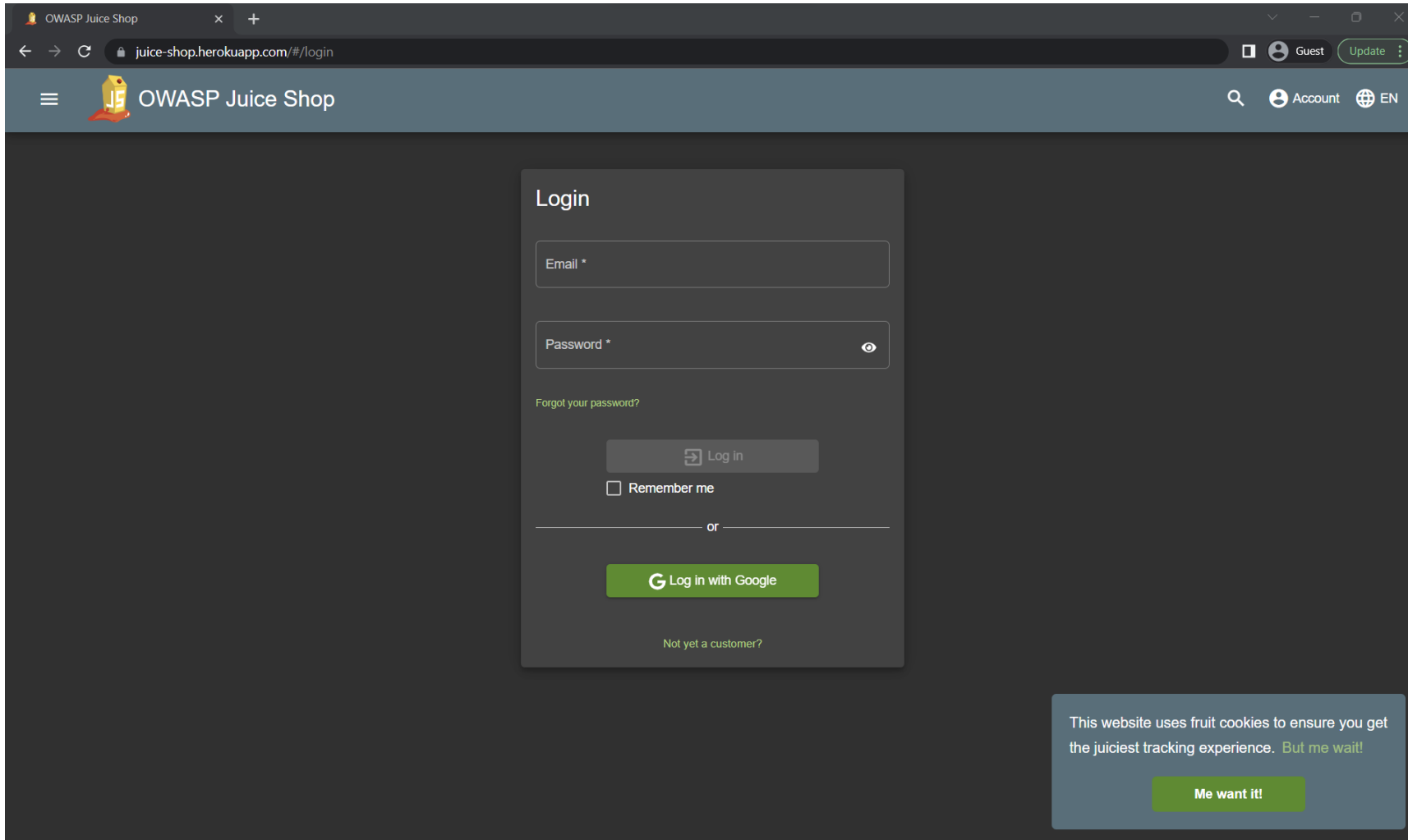


APPLICATION SECURITY

Facebook Pays Out \$40,000 for Account Takeover Exploit Chain

Social media giant Facebook on Thursday announced a new payout guideline to help vulnerability hunters better understand its bounty decisions related to given bugs.

Entry Points for ATO



The screenshot shows the OWASP Juice Shop login page in a web browser. The browser's address bar displays the URL `juice-shop.herokuapp.com/#/login`. The page header includes the OWASP Juice Shop logo and navigation links for 'Account' and 'EN'. The main content area features a 'Login' form with the following elements:

- Email ***: A text input field for the user's email address.
- Password ***: A password input field with a toggle icon for visibility.
- Forgot your password?**: A link to the password recovery page.
- Log in**: A button with a lock icon.
- Remember me**: A checkbox for remembering the user's login details.
- or**: A separator line with the word 'or' in the center.
- Log in with Google**: A green button with the Google logo for social login.
- Not yet a customer?**: A link to the registration page.

In the bottom right corner, there is a cookie consent banner with the text: 'This website uses fruit cookies to ensure you get the juiciest tracking experience. But me wait!' and a green button labeled 'Me want it!'.

Entry Points for ATO

The image shows a screenshot of the OWASP Juice Shop login page. The page has a dark theme with a central login form. The browser's address bar shows the URL `juice-shop.herokuapp.com/#/login`. The page header includes the OWASP Juice Shop logo and navigation links for 'Account' and 'EN'. The login form contains fields for 'Email *' and 'Password *', a 'Forgot your password?' link, a 'Log in' button, a 'Remember me' checkbox, and a 'Log in with Google' button. Below the login form is a link for 'Not yet a customer?'. A cookie consent banner is visible at the bottom right.

Callouts with red arrows point to the following entry points for Account Takeover (ATO):

- Password Reset Takeover**: Points to the 'Forgot your password?' link.
- Login Takeover**: Points to the 'Log in' button.
- Session Takeover**: Points to the 'Remember me' checkbox.
- Registration Takeover**: Points to the 'Not yet a customer?' link.
- Oauth Takeover**: Points to the 'Log in with Google' button.

Cookie consent banner text: "This website uses fruit cookies to ensure you get the juiciest tracking experience. But me wait! Me want it!"

Entry Points for ATO

The image shows a screenshot of the OWASP Juice Shop login page. The page has a dark theme with a central login form. The browser's address bar shows the URL `juice-shop.herokuapp.com/#/login`. The page header includes the OWASP Juice Shop logo and navigation links for 'Account' and 'EN'. The login form contains fields for 'Email *' and 'Password *', a 'Forgot your password?' link, a 'Log in' button, a 'Remember me' checkbox, and a 'Log in with Google' button. A 'Not yet a customer?' link is at the bottom of the form. A cookie consent banner is at the bottom right of the page.

Three red arrows point from external labels to specific elements on the page:

- Password Reset Takeover** (yellow box) points to the 'Forgot your password?' link.
- Login Takeover** (orange box) points to the 'Log in' button.
- Session Takeover** (grey box) points to the 'Remember me' checkbox.

At the bottom right, a cookie consent banner reads: 'This website uses fruit cookies to ensure you get the juiciest tracking experience. But me wait!' with a 'Me want it!' button.

Session Takeover

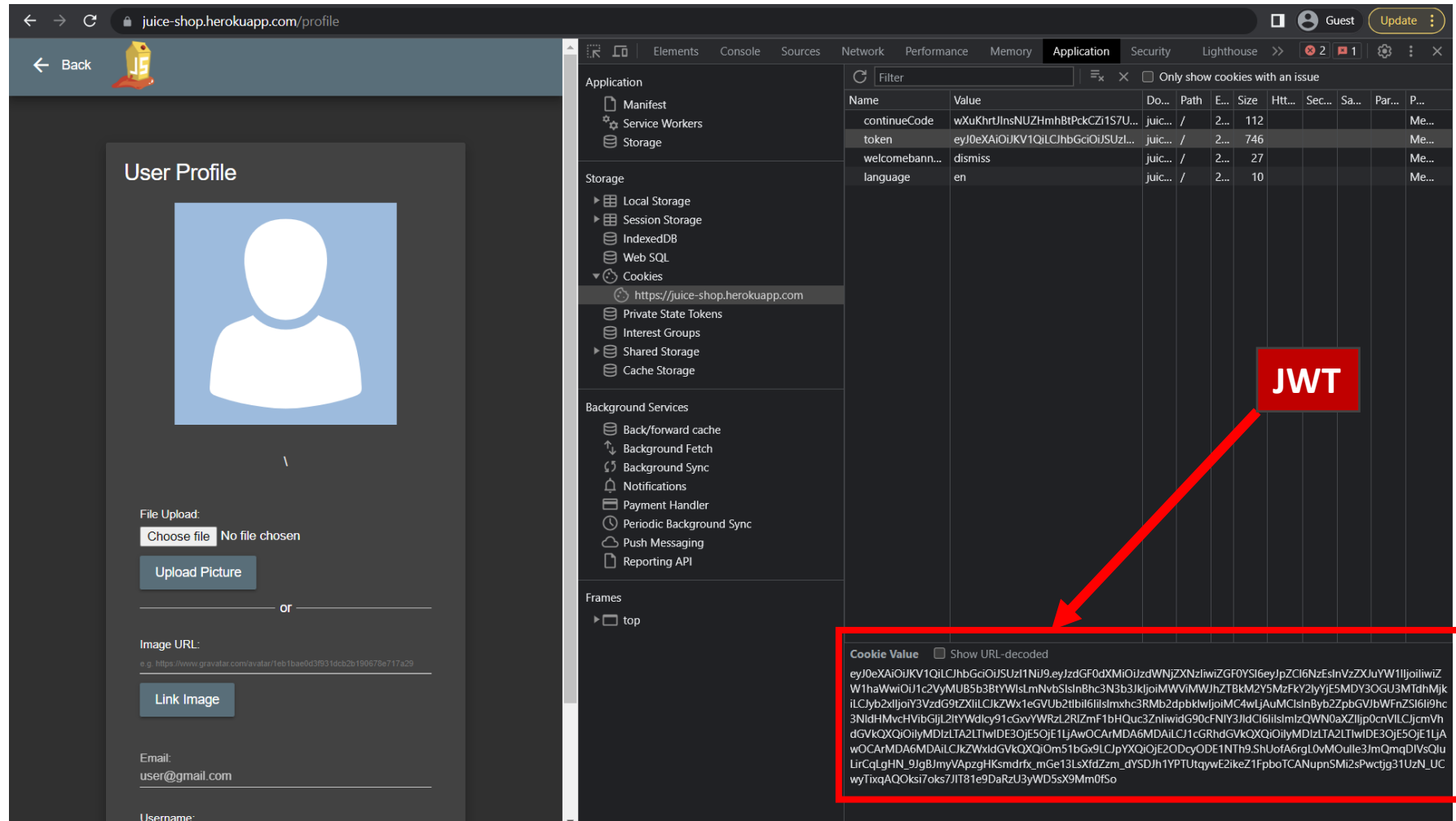
Session Takeover

The screenshot shows a web browser at `juice-shop.herokuapp.com/profile`. The page displays a "User Profile" section with a placeholder image, a "File Upload" section with a "Choose file" button and "No file chosen" text, and an "Image URL" section with a "Link Image" button. The browser's DevTools Application tab is open, showing a list of cookies. A red arrow points from the "token" cookie in the list to a red box labeled "Cookie".

Name	Value	Domain	Path	Expires	Size	HttpOnly	Secure	SameSite	Partitioned	Persistent
continueCode	wXuKhrUjnsNUZHmh8tPckCZ1S7U...	juic...	/	2...	112					Me...
token	eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI...	juic...	/	2...	746					Me...
welcome	dismiss	juic...	/	2...	27					Me...
language		juic...	/	2...	10					Me...

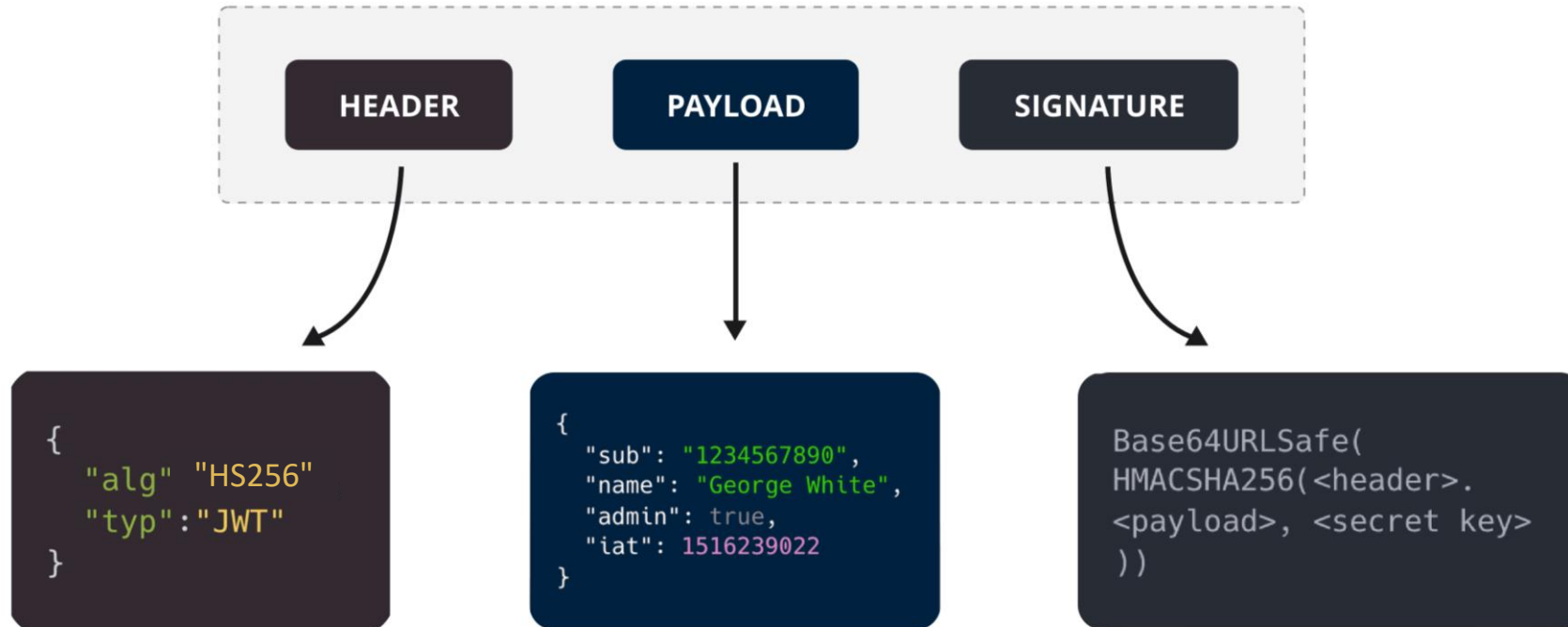
Cookie

Session Takeover : JWT (JSON Web Tokens)



Session Takeover : JWT Structure

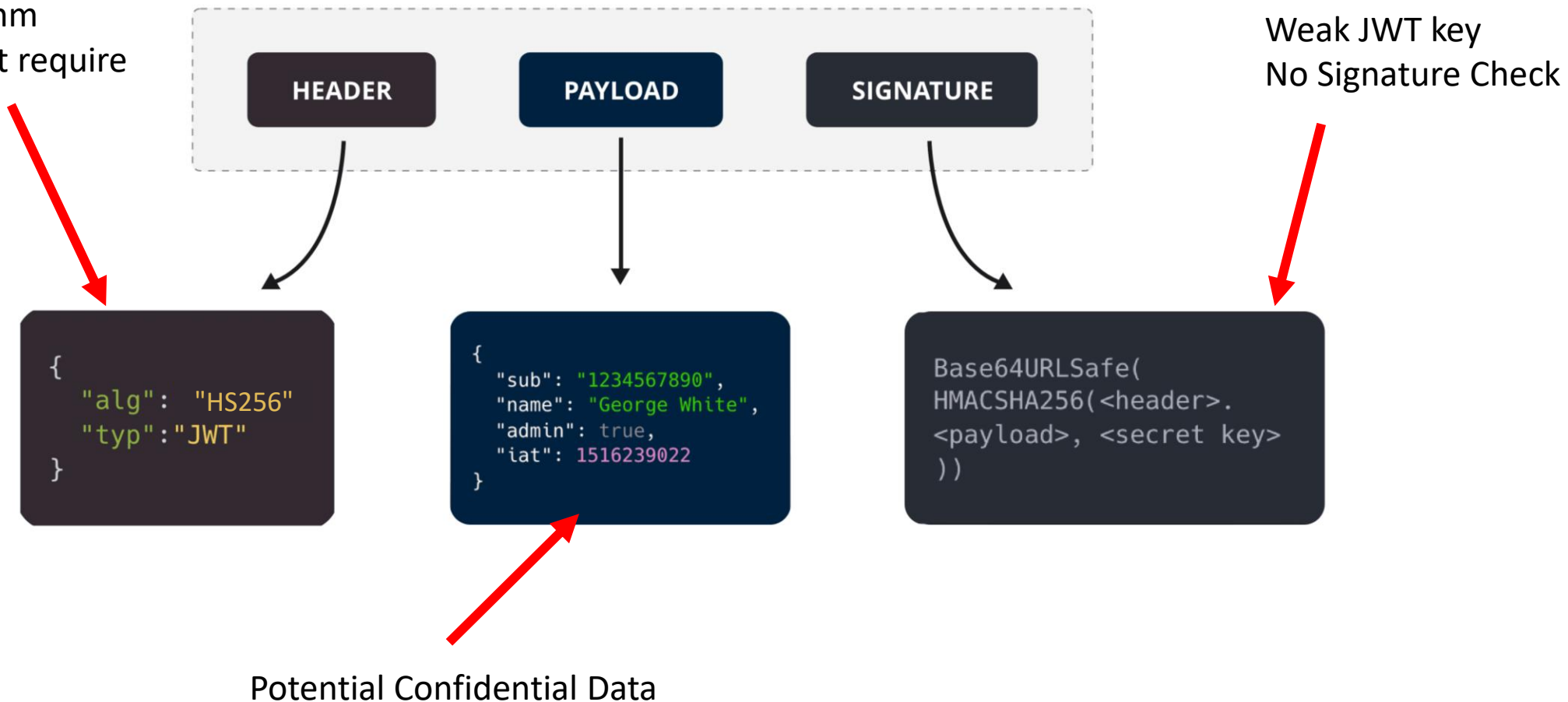
Structure of a JSON Web Token (JWT)



Session Takeover : JWT Misconfiguration

Structure of a JSON Web Token (JWT)

None Algorithm
which doesn't require
Signature



Session Takeover : Json Web Token Attacker

The screenshot displays the Burp Suite interface, specifically the HTTP history and the JSON Web Token (JWT) attack configuration.

HTTP History: The top panel shows a list of HTTP requests. The selected request is a GET request to `/rest/user/whoami` with a status code of 200. The response contains a JWT token.

JSON Web Tokens: The bottom panel shows the configuration for the JWT attack. The target is `https://juice-shop.herokuapp.com`. The request is a GET request to `/rest/user/whoami`. The response is a JSON object containing a JWT token.

Request: The request is a GET request to `/rest/user/whoami` with a status code of 200. The response is a JSON object containing a JWT token.

Response: The response is a JSON object containing a JWT token.

Attack Configuration: The attack configuration is set to "Do not automatically modify signature". The "Alg" dropdown is set to "None". The "Secret" field is empty.

Session Takeover : JWT Uncommon Vulnerability

The JWT from The Main Web Application

SendCancel<>

Target: https://api[REDACTED] [Pencil Icon] HTTP/2 ?

Request

PrettyRawHex\n☰

1 GET /users/details HTTP/2
2 Host: api.[REDACTED]
3 Cookie: _gcl_au=1.1.1856608971.1631473630; _gid=GA1.2.2095338827.1631473630; _fbp=
fb.1.1631473630665.984309487; _gat_UA-80231189-8=1; _gat_gtag_UA-80231189_8=1;
_ga_HNSX08WGK7=G81.1.1631473629.1.1.1631474569.0; _ga=GA1.1.915960157.1631473630;
_vca_current=
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6MTMzMzODQsImNyZWVzZWVpbGl6IjIwMjE0MDktMTIyYTY6M
[REDACTED]
[REDACTED]
4 Sec-Ch-Ua: "Chromium";v="92", " Not A;Brand";v="99", "Google Chrome";v="92"
5 Sec-Ch-Ua-Mobile: ?0
6 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/92.0.4515.131 Safari/537.36
7 Content-Type: application/json
8 Accept: */*
9 [REDACTED]
10 Sec-Fetch-Site: same-site
11 Sec-Fetch-Mode: cors
12 Sec-Fetch-Dest: empty
13 [REDACTED]
14 Accept-Encoding: gzip, deflate
15 Accept-Language: en-US,en;q=0.9
16
17

Response

PrettyRawHexRender\n☰

12 Surrogate-Control: no-store
13 Cache-Control: no-store, no-cache, must-revalidate, proxy-revalidate
14 Pragma: no-cache
15 Expires: 0
16 Content-Security-Policy: default-src 'self'; style-src 'self'
17 Access-Control-Allow-Origin: https://www.[REDACTED]
18 Vary: Origin
19 Access-Control-Allow-Credentials: true
20 Set-Cookie: _vca_current=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6MTMzMzODQsImNyZWVzZWVpbGl6IjIwMjE0MDktMTIyYTY6M
Domain=[REDACTED]; Path=/; Expires=Mon, 13 Sep 2021 19:23:16 GMT; HttpOnly; Secure
21 Etag: W/"1cd-7/97BjkQ/lkGNrCDKhO9aHEMsU"
22
23 {
"status":201,
"message":"User Data",
"data":{
"id":13384,
"email":"sand[REDACTED]@gmail.com",
"mobile":"undefined",
"firstName":"Bya",
"lastName":"Syb",
"createdOn":"2021-09-12 16:33:01.616",
"lastUpdated":"2021-09-12T16:33:02.000Z",
"active":{
"type":"Buffer",
"data":[
1
]
},
[REDACTED]
"productType":"","
"issuerBid":"","
}

Session Takeover : JWT Uncommon Vulnerability

The JWT from The Main Web Application

eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6MTAzMDcsImIhdCI6MTY0NjIxOTU0NiwiZXhwIjoxNjgwNDMzOTQ2fQ.C6g3y48Q8ZFvEIOTwZ5NckObGXY5aX-Xn-7w-G3

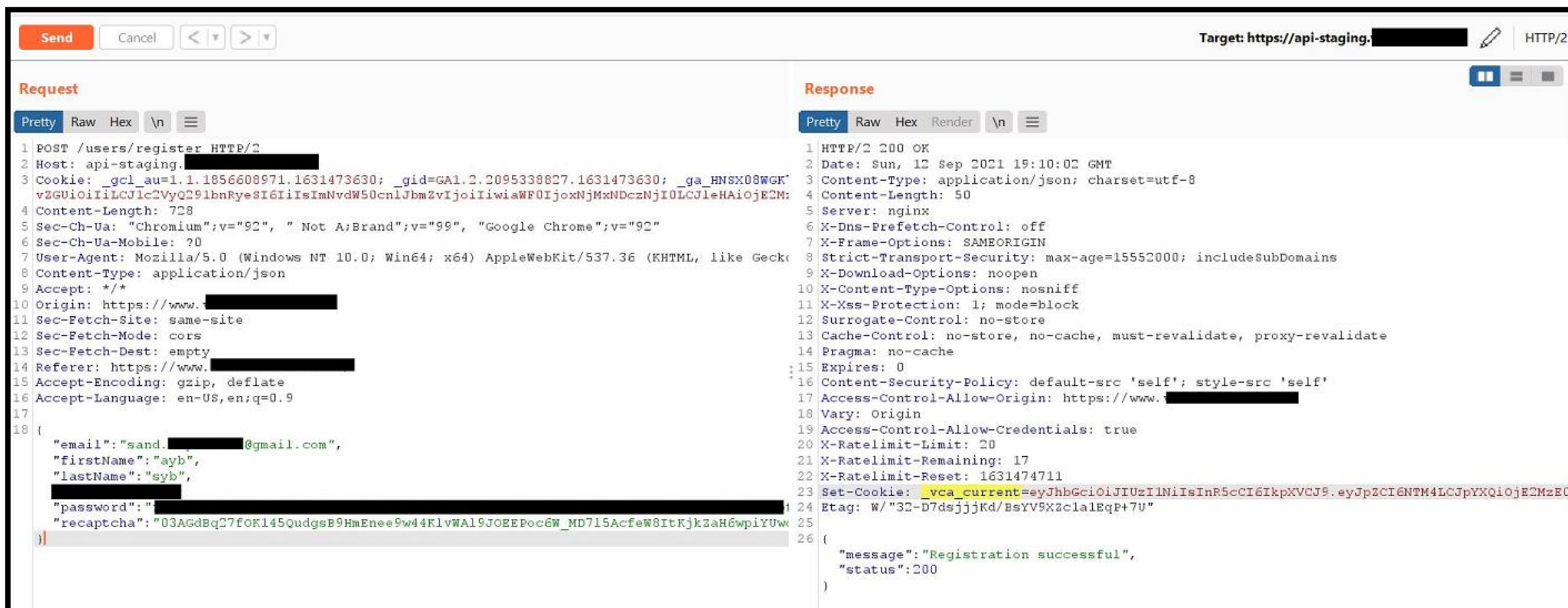
```
{  
  "alg": "HS256",  
  "typ": "JWT"  
}
```

```
{  
  "id": 13384,  
  "iat": 1646219546,  
  "exp": 1680433946  
}
```

```
HMACSHA256(Base64(header).Base64(payload),secret)  
=  
C6g3y48Q8ZFvEIOTwZ5NckObGXY5aX-Xn-7w-G3
```


Session Takeover : JWT Uncommon Vulnerability

The JWT from The Staging Web Application



Session Handling Takeover : JWT Uncommon Vulnerability

The JWT from The Staging Web Application

eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6NTYyLCJpYXQiOiE2NDYyMTk1NDYsImV4cCI6MTY4ME5E.Zsd2ny48Q8ZFvEIOTwZ5NckObGXY5aCSy-Br-h7

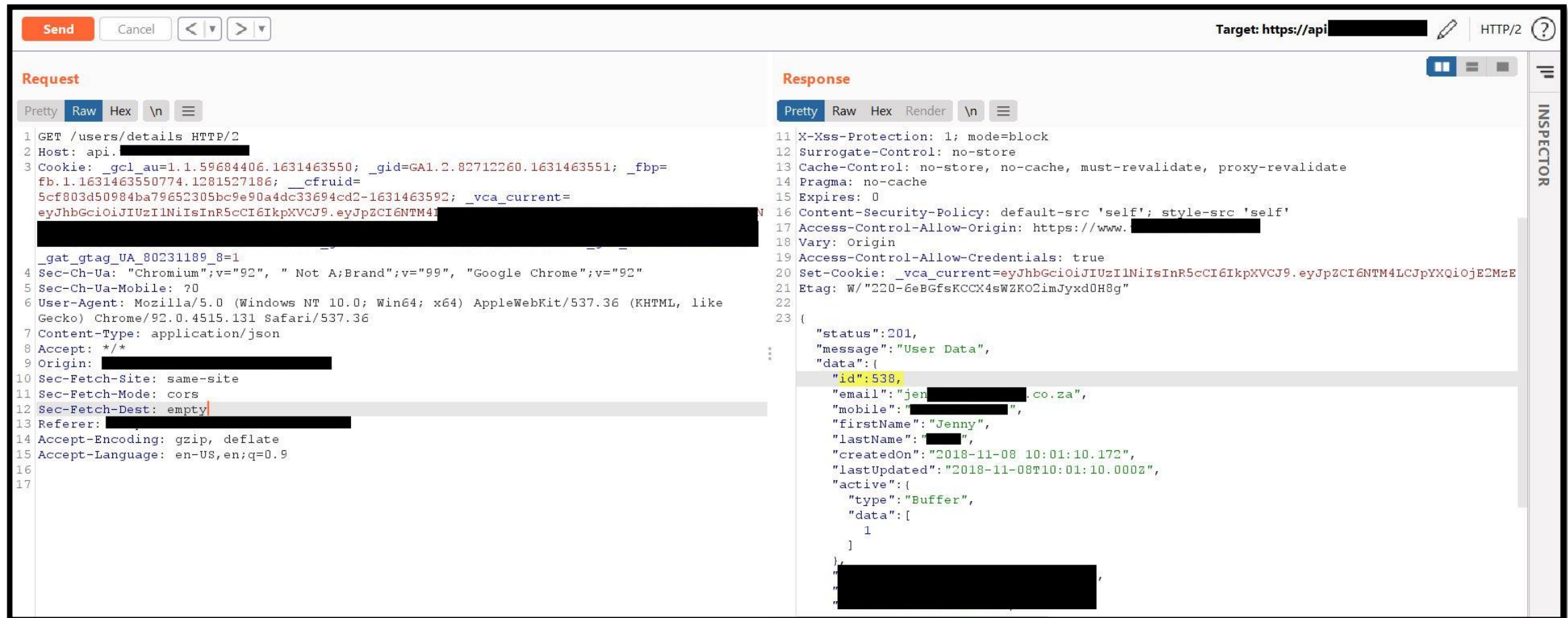
```
{  
  "alg": "HS256",  
  "typ": "JWT"  
}
```

```
{  
  "id": 538,  
  "iat": 1646219546,  
  "exp": 1680433946  
}
```

```
HMACSHA256(Base64(header).Base64(payload),secret)  
=  
Zsd2ny48Q8ZFvEIOTwZ5NckObGXYCSy-Xn-Nr-h7
```

Hmm, This look Interesting



JWT Reuse Attack



Session Takeover : JWT Uncommon Vulnerability

JWT Reuse Attack

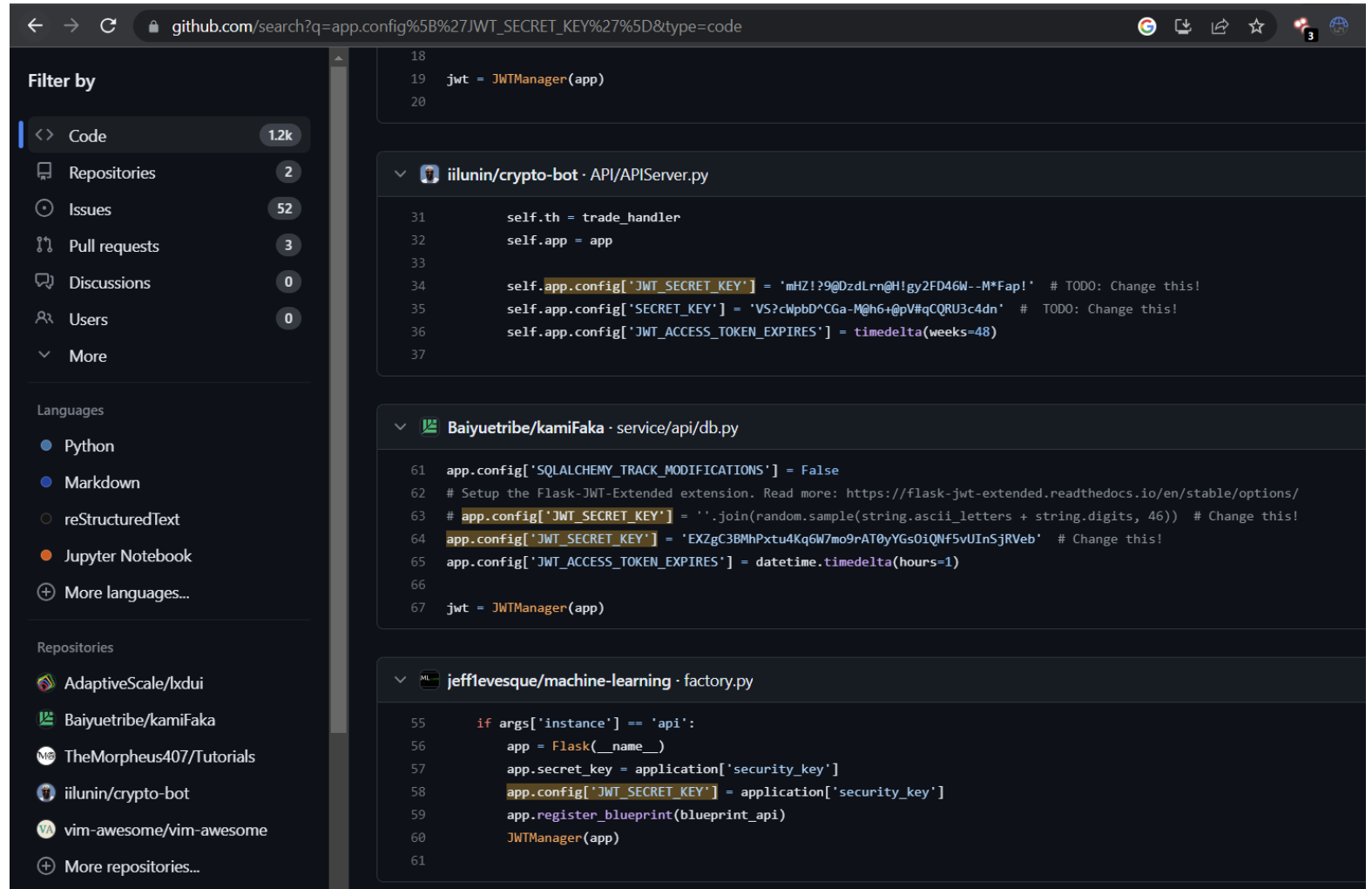
13387 – 538 = 12849 Accounts got hacked

Severity	 High (7.4)
Asset: Dom...	www. 
Weakness	Improper Access Control - Generic
Bounty	\$3,000
Time spent	<i>None</i>

Session Takeover : JWT Uncommon Vulnerability

JWT Reuse Attack

- Staging Environments
- Similar Web Application
- GitHub is your friend !!!



The screenshot shows a GitHub search results page for the query `app.config['JWT_SECRET_KEY']`. The left sidebar contains filters for 'Filter by' (Code: 1.2k, Repositories: 2, Issues: 52, Pull requests: 3, Discussions: 0, Users: 0) and 'Languages' (Python, Markdown, reStructuredText, Jupyter Notebook, More languages...). The main content area displays three search results:

- iilunin/crypto-bot · API/APIServer.py**
Lines 31-37 show the configuration of the JWT secret key and access token expiration.

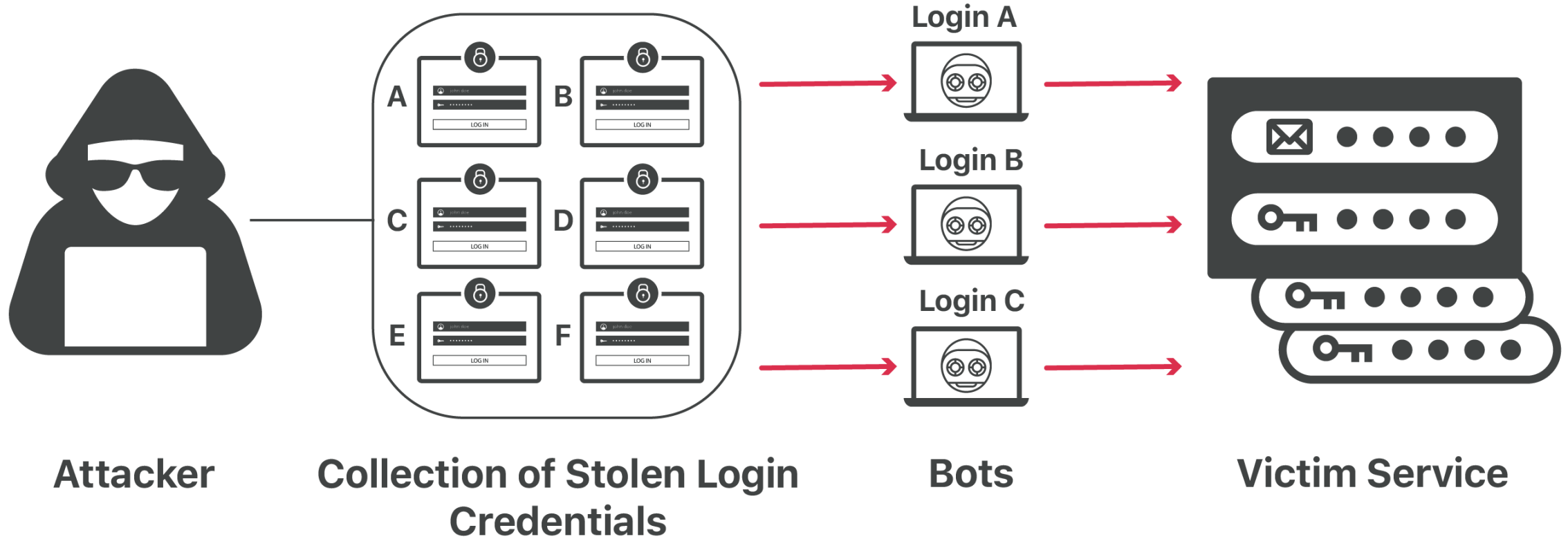
```
18
19  jwt = JWTManager(app)
20
31  self.th = trade_handler
32  self.app = app
33
34  self.app.config['JWT_SECRET_KEY'] = 'mHZ!p9@DzdLrn@H!gy2FD46W--M*Fap!' # TODO: Change this!
35  self.app.config['SECRET_KEY'] = 'VS?chpbD^CGa-M@h6+@pV#qCQRU3c4dn' # TODO: Change this!
36  self.app.config['JWT_ACCESS_TOKEN_EXPIRES'] = timedelta(weeks=48)
37
```
- Baiyuetribe/kamiFaka · service/api/db.py**
Lines 61-67 show the configuration of the JWT secret key and access token expiration.

```
61  app.config['SQLALCHEMY_TRACK_MODIFICATIONS'] = False
62  # Setup the Flask-JWT-Extended extension. Read more: https://flask-jwt-extended.readthedocs.io/en/stable/options/
63  # app.config['JWT_SECRET_KEY'] = ''.join(random.sample(string.ascii_letters + string.digits, 46)) # Change this!
64  app.config['JWT_SECRET_KEY'] = 'EX2gC3BMhPxTu4Kq6W7mo9rAT0yYGs0iQnf5vUInSjRVeb' # Change this!
65  app.config['JWT_ACCESS_TOKEN_EXPIRES'] = datetime.timedelta(hours=1)
66
67  jwt = JWTManager(app)
```
- jefflevesque/machine-learning · factory.py**
Lines 55-61 show the configuration of the JWT secret key and access token expiration.

```
55  if args['instance'] == 'api':
56      app = Flask(__name__)
57      app.secret_key = application['security_key']
58      app.config['JWT_SECRET_KEY'] = application['security_key']
59      app.register_blueprint(blueprint_api)
60      JWTManager(app)
61
```

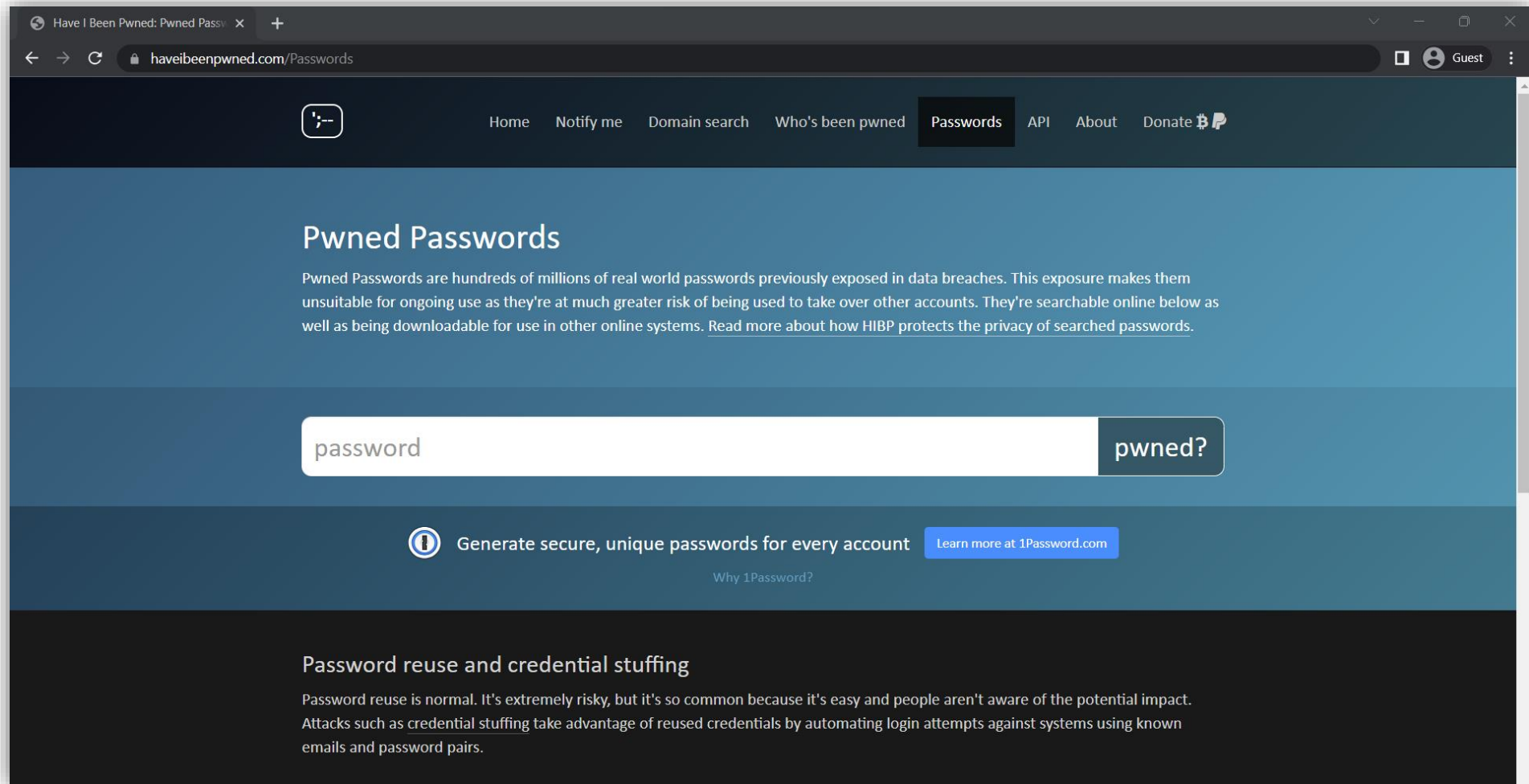
Login Takeover

Login Takeover: Credential Stuffing



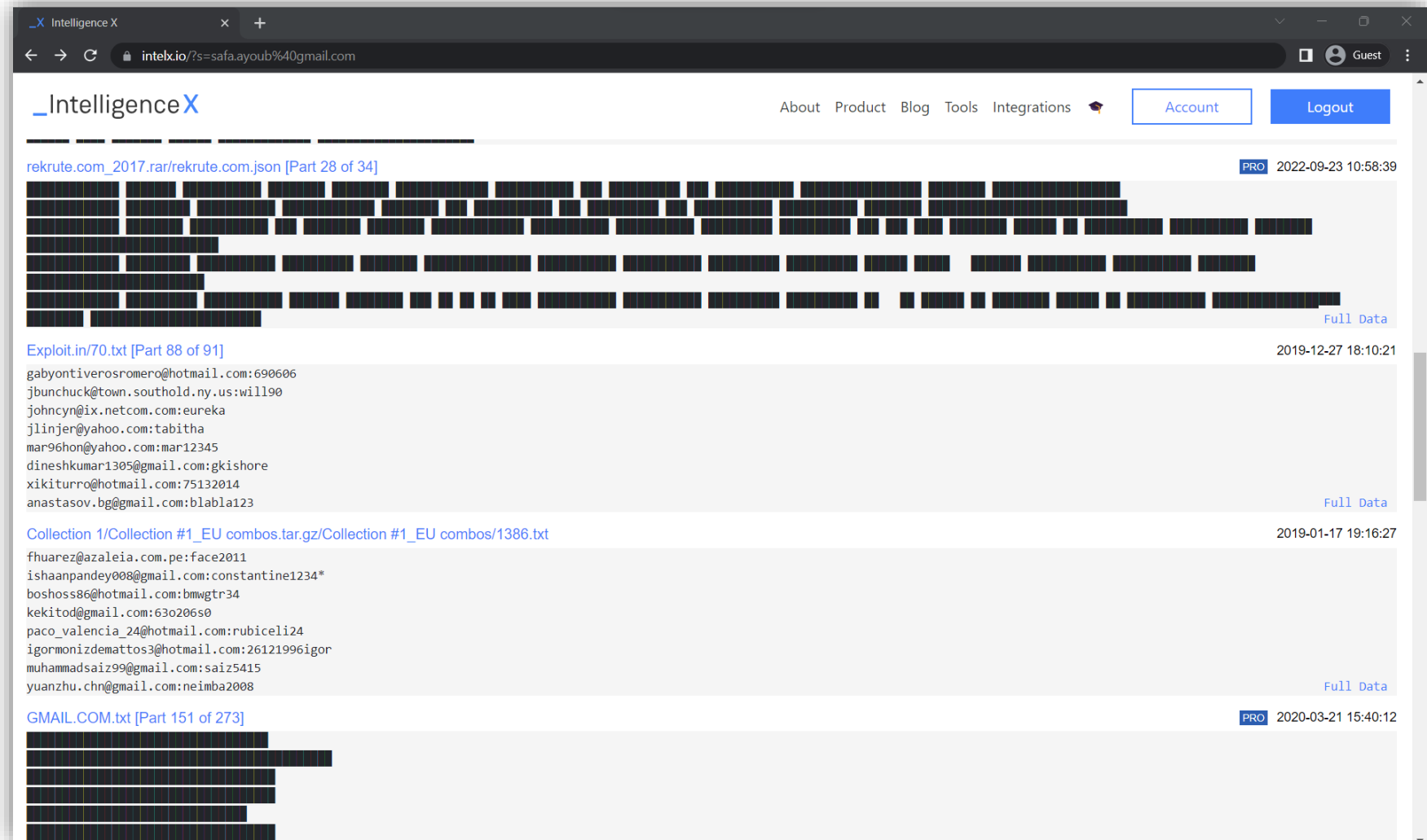
Login Takeover: Credential Stuffing

<https://haveibeenpwned.com/>



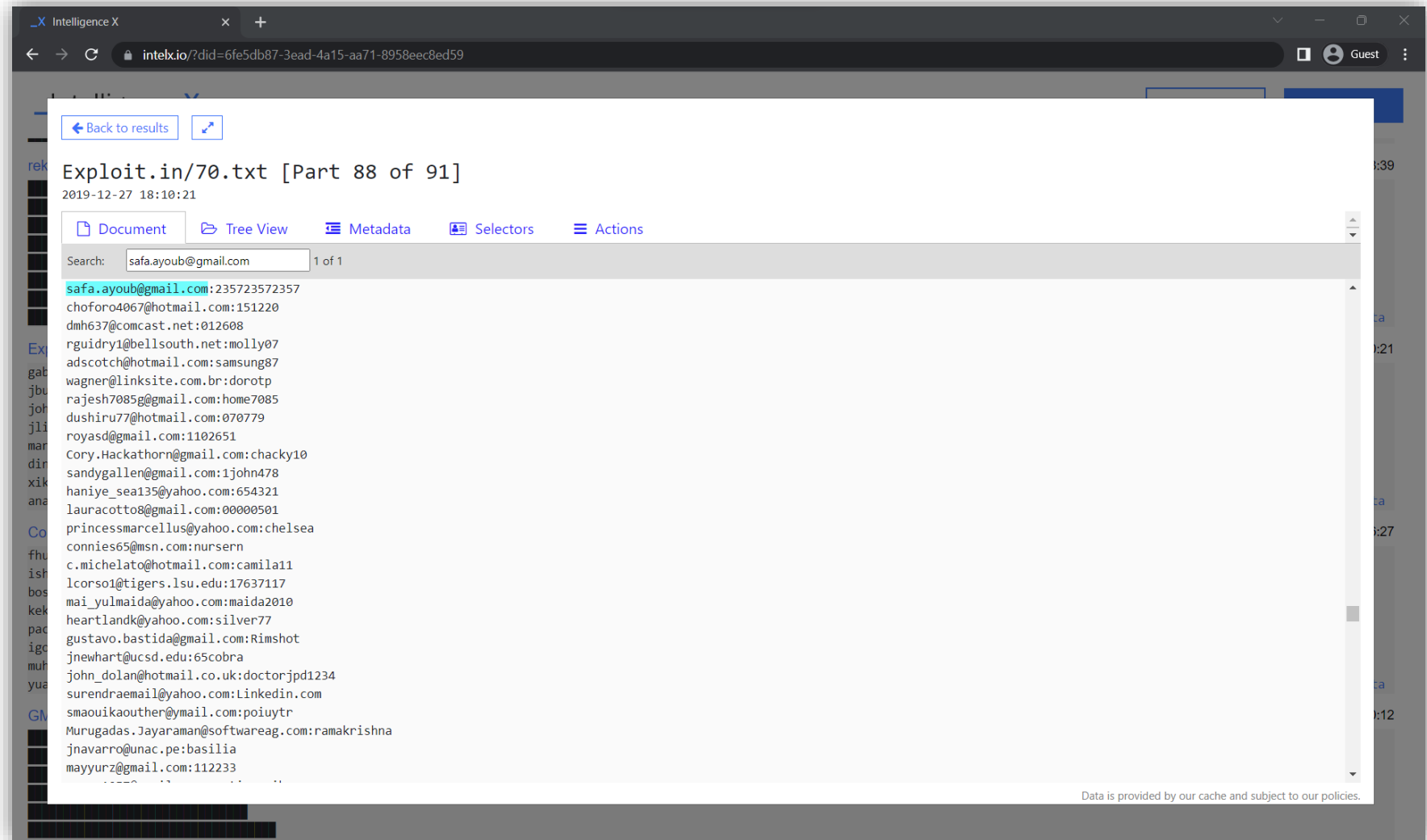
Login Takeover: Credential Stuffing

<https://intelx.io/>



Login Takeover: Credential Stuffing

<https://intelx.io/>



Login Takeover: Credential Stuffing

GitHub

Filter by

Code

34.6k

Repositories

384

Issues

341k

Pull requests

9k

Discussions

731

Users

17

More

Languages

SQL

TSQL

PLpgSQL

PLSQL

Text

More languages...

Repositories

...ing-PostgreSQL-11-Third-Edition

WWBN/AVideo

bbalet/jorani

...c-Management-System-ASP.NET

knadh/listmonk

More repositories...

34.6k files (243 ms)

mohsinenur/Ecommerce-Website-Using-Python-Flask · database/menshut.sql

39

`"password" varchar(100) NOT NULL,`

49

`_RT INTO `admin` (`id`, `firstName`, `lastName`, `email`, `mobile`, `address`, `password`, `ty`

50

`(4, 'Nur', 'Mohsin', 'mohsin@gmail.com', '01677876551', 'Dhaka', '5rounds=535000$W0A0Mdg0K2J`

227

`"password" varchar(100) NOT NULL,`

239

`INSERT INTO `users` (`id`, `name`, `email`, `username`, `password`, `mobile`, `reg_time`, `onl`

240

`(12, 'Mukul', 'mukul@gmail.com', 'mukul', '5rounds=535000$6PJhbzF1fJbcQbza$FbrPa3qk1RJ5MSff`

241

`(9, 'Nur Mohsin', 'mohsin@gmail.com', 'mohsin', '5rounds=535000$EnLkwqfGwGcWk1RL$g9PbYw/TVXS`

242

`(14, 'Nur Mohsin', 'khan@gmail.com', 'khan', '5rounds=535000$wLKTQexvPQHueUsK$aFrFUXBHjrrAH6`

Show 1 more match

winston-dsouza/ecommerce-website · ecommerce.sql

76

`-- Dumping data for table `users``

77

`--`

78

79

`INSERT INTO `users` (`id`, `email_id`, `first_name`, `last_name`, `phone`, `registration_time``

80

`(65, 'sharew5m123@gmail.com', 'reys', 'rudt', 0, '2019-03-18 13:46:33', 'e4f194cba29960e12d8b8`

81

`(66, 'sgah234@gmail.com', 'werty', 'erty', 0, '2019-03-18 13:55:46', 'e10adc3949ba59abbe56e057`

82

`(67, 'sham1234@gmail.com', 'Sham', 'das', 0, '2019-03-19 07:37:46', 'e10adc3949ba59abbe56e057f`

Show 1 more match

xiusin/pinecms · resources/pinecms.sql

870

``account` varchar(40) CHARACTER SET utf8 COLLATE utf8_general_ci DEFAULT NULL COMMENT '账号'`

871

`"password" varchar(32) CHARACTER SET utf8 COLLATE utf8_general_ci DEFAULT NULL COMMENT '密码'`

872

`"avatar" varchar(100) CHARACTER SET utf8 COLLATE utf8_general_ci DEFAULT NULL COMMENT '头像'`

1191

`INSERT INTO `pinecms_setting` VALUES (29, 'UPLOAD_MAX_SIZE', '20', '存储配置', '2', '上传大小(ME`

1192

`INSERT INTO `pinecms_setting` VALUES (30, 'EMAIL_ADMIN_EMAIL', 'xiusin.chen@gmail.com', '邮箱设`

Code

b56f6a4

Go to file

bulk_stu.php

bulk_stu_det.php

check.php

combine_sub.php

contact_us.php

dashboard.php

database.sql

date_range_vise.php

date_vise.php

dist_sub.php

div_vise.php

divide_batch.php

dropdown.php

extra_holiday.php

fac_update.php

footer.php

footer_table.php

header_admin.php

header_at.php

header_fac.php

MetiendMano / FromOthers / Online-Attendance-System-PHP-MySQL / database.sql

Code

Blame

1999 lines (1840 loc) · 70.5 KB

1050

`(9, 'MU', 'Milan Undavia', 1, 'milanundaviaMU@gmail.com', 0, 'milanundaviaMU', 2),`

1051

`(10, 'PK', 'Pooja Khatri', 2, 'poojakhatriPK@gmail.com', 0, 'poojakhatriPK', 2),`

1052

`(11, 'AP', 'Akanksha Patel', 2, 'akanshapatelAP@gmail.com', 0, 'akanshapatelAP', 2),`

1053

`(12, 'KC', 'Kinjal Choksi', 1, 'kinjalchoksiKC@gmail.com', 0, 'kinjalchoksiKC', 2),`

1054

`(13, 'UB', 'Uday Bhatt', 2, 'udaybhattUB@gmail.com', 0, 'udaybhattUB', 2),`

1055

`(14, 'AYB', 'Amrita Y. Bardiya', 1, 'amritaybardiyaAYB@gmail.com', 0, 'amritaybardiyaAYB', 2),`

1056

`(15, 'RR', 'Dr. R. Radha', 2, 'rradharR@gmail.com', 0, 'rradharR', 1),`

1057

`(16, 'AG', 'Dr. Anjali Gokhru', 1, 'anjalogokhruAG@gmail.com', 0, 'anjalogokhruAG', 2),`

1058

`(17, 'IJ', 'Ingita Jain', 1, 'ingitajainIJ@gmail.com', 0, 'ingitajainIJ', 2),`

1059

`(18, 'RG', 'Dr. Rachna Gandhi', 1, 'rachnagandhiRG@gmail.com', 0, 'rachnagandhiRG', 2),`

1060

`(19, 'SF', 'Suman Fulwani', 1, 'sumanfulwaniSF@gmail.com', 0, 'sumanfulwaniSF', 2),`

1061

`(20, 'VS', 'Vishva Shah', 1, 'vishvashahVS@gmail.com', 0, 'vishvashahVS', 2),`

1062

`(21, 'VN', 'Vishali Nindroda', 2, 'vaishalinindrodaVN@gmail.com', 0, 'vaishalinindrodaVN', 2),`

1063

`(22, 'AA', 'Anita Ahuja', 1, 'anitaahujaAA@gmail.com', 0, 'anitaahujaAA', 2),`

1064

`(23, 'PM', 'Priyanka Mehta', 1, 'priyankamehtaPM@gmail.com', 0, 'priyankamehtaPM', 2),`

1065

`(24, 'SA', 'DR. Shamina Ansari', 1, 'shaminaansariSA@gmail.com', 0, 'shaminaansariSA', 2),`

1066

`(25, 'RS', 'Richa Seth', 1, 'richasethRS@gmail.com', 0, 'richasethRS', 2),`

1067

`(26, 'HP', 'Dr. Hiral Parikh', 1, 'hiralparikhHP@gmail.com', 0, 'hiralparikhHP', 2),`

1068

`(27, 'AMG', 'Akanxa M. Galande', 1, 'akanxamhgalandeAMG@gmail.com', 0, 'akanxamhgalandeAMG', 2),`

1069

`(28, 'IS', 'Ishita Sakariya', 1, 'ishitasakariyaIS@gmail.com', 0, 'ishitasakariyaIS', 2),`

1070

`(29, 'AV', 'Ankita Vaidya', 1, 'ankitavaidyaAV@gmail.com', 0, 'ankitavaidyaAV', 2),`

1071

`(30, 'AB', 'Asha Brahmshatriya', 2, 'ashabrahmshatriyaAB@gmail.com', 0, 'ashabrahmshatriyaAB', 2),`

1072

`(31, 'NC', 'Nirul Chaudhary', 1, 'nirulchaudharyNC@gmail.com', 0, 'nirulchaudharyNC', 2),`

1073

`(32, 'NG', 'Dr. Neelkamal Gogna', 1, 'neelkamalgognaNG@gmail.com', 0, 'neelkamalgognaNG', 2),`

1074

`(33, 'KP', 'Kalyani Patel', 0, 'kalyanipatelKP@gmail.com', 0, 'kalyanipatelKP', 2),`

1075

`(34, 'NG', 'Nandita Goswami', 2, 'nanditagoswamiNG@gmail.com', 0, 'nanditagoswamiNG', 2),`

1076

`(35, 'SC', 'Sonali Chakraborty', 0, 'sonalichokrabortySC@gmail.com', 0, 'sonalichokrabortySC', 2),`

1077

`(36, 'SS', 'Shaltee Shah', 0, 'shalleeshahSS@gmail.com', 0, 'shalleeshahSS', 2),`

1078

`(37, 'PD', 'Palak Dabhi', 0, 'palakdabhiPD@gmail.com', 0, 'palakdabhiPD', 2),`

1079

`(38, 'VS', 'Vidhi Sutariya', 0, 'vidhisutariyaVS@gmail.com', 0, 'vidhisutariyaVS', 2),`

1080

`(39, 'DB', 'Dipti Bhatt', 0, 'diptibhattDB@gmail.com', 0, 'diptibhattDB', 2),`

1081

`(40, 'PA', 'Priyanka Anorra', 0, 'priyankaanorraPA@gmail.com', 0, 'priyankaanorraPA', 2),`

1082

`(41, 'RS', 'Rujuta Shah', 0, 'rujutashahRS@gmail.com', 0, 'rujutashahRS', 2),`

1083

`(42, 'JP', 'Jaimini Patel', 0, 'jaiminipatelJP@gmail.com', 0, 'jaiminipatelJP', 2),`

1084

`(43, 'ND', 'Namita Doshi', 0, 'namitadoshiND@gmail.com', 0, 'namitadoshiND', 2),`

1085

`(44, 'AK', 'Amit Kalyani', 0, 'amitkalyaniAK@gmail.com', 0, 'amitkalyaniAK', 2),`

1086

`(45, 'JR', 'Jigar Raval', 0, 'jigararavallJR@gmail.com', 0, 'jigararavallJR', 2),`

1087

`(46, 'EK', 'Ekta Kikiani', 0, 'ektakikianiEK@gmail.com', 0, 'ektakikianiEK', 2),`

Documentation · Share feedback

Login Takeover: Credential Stuffing

#789950

Credentials disclosure from a Public GitHub repository allowing access to your Sharepoint Instance via ForeFront TMG

Participants



State

● Resolved (Closed)

Reported to



Managed

Severity

Critical (10.0)

Asset: Wil...

.com

Weakness

Cleartext Storage of Sensitive Information

Bounty

\$4,000

Time spent

None

Visibility

Private

Password Reset Takeover

Password Reset Takeover : GUID / UUID

<https://target.com/password/reset?token=3fcf5140-47ca-11ec-9755-c75cdea7a1c7>

Password Reset Takeover : GUID / UUID

Did you know that there are different types of GUIDs?

Nil GUID – Version 0

00000000-0000-0000-0000-000000000000

DCE Security GUID – Version 2

b165e8c6-5e9a-21ea-9e00-0242ac130003

Time-based GUID – Version 1

e6e3422c-c82d-11ed-8761-3ff799965458

Name-based GUID - Version 3 and 5

18f99f82-61f7-3530-8d8a-8fdf2cd0cae0
b21b95a4-56c3-51de-8828-1bb7bd249fd2

Randomly Generated GUID - Version 4

0d706e07-75b5-4553-8abd-6c3d52fdbf70

Password Reset Takeover : GUID / UUID

Did you know that there are different types of GUIDs?

Nil/null GUID – **Version 0**

00000000-0000-0000-0000-000000000000

DCE Security GUID – **Version 2**

b165e8c6-5e9a-21ea-9e00-0242ac130003

Time-based GUID – **Version 1**

e6e3422c-c82d-11ed-8761-3ff799965458

Name-based GUID - **Version 3 and 5**

18f99f82-61f7-3530-8d8a-8fdf2cd0cae0
b21b95a4-56c3-51de-8828-1bb7bd249fd2

Randomly Generated GUID - **Version 4**

0d706e07-75b5-4553-8abd-6c3d52fdbf70

Password Reset Takeover : Guessable GUID / UUID

Time-based GUID – Version 1

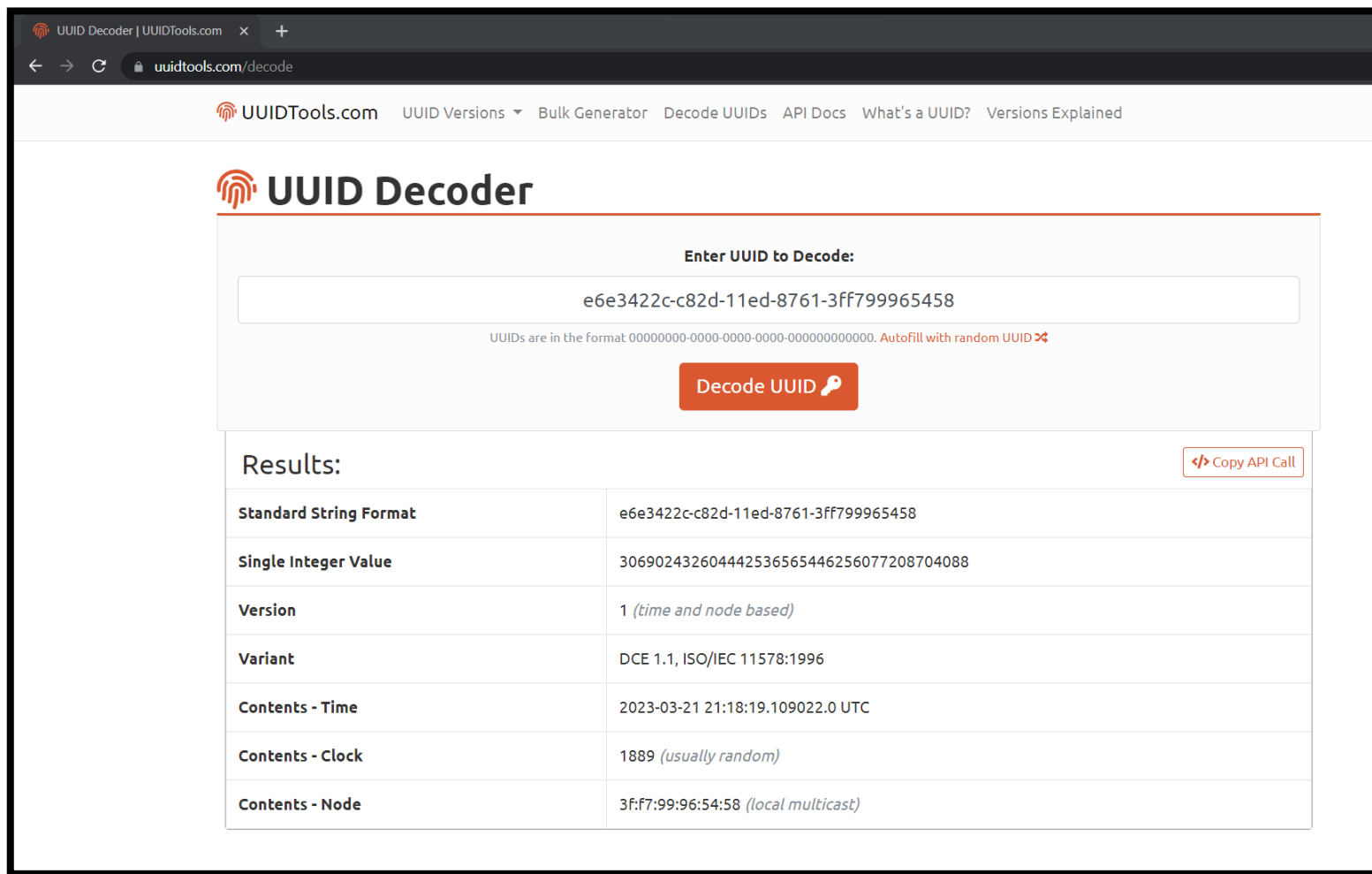
e6e3422c-c82d-11ed-8761-3ff799965458

The diagram illustrates the structure of a Time-based GUID (Version 1). The GUID is shown as 'e6e3422c-c82d-11ed-8761-3ff799965458'. Colored arrows point from specific parts of the GUID to their corresponding components: a red arrow from the '1' in '11ed' to the 'Version 1' label; a green arrow from 'e6e3422c' to the 'Timestamp' label; a blue arrow from 'c82d' to the 'Timestamp' label; a purple arrow from '11ed' to the 'Timestamp' label; and a gold arrow from '3ff799965458' to the 'MAC Address' label.

Timestamp-1edc82de6e3422c

MAC Address: 3f:f7:99:96:54:58

Password Reset Takeover : Guessable GUID / UUID



The screenshot shows the UUID Decoder interface on the website uuidtools.com. The browser address bar shows the URL `uuidtools.com/decode`. The page header includes navigation links: [UUID Versions](#), [Bulk Generator](#), [Decode UUIDs](#), [API Docs](#), [What's a UUID?](#), and [Versions Explained](#).

The main heading is **UUID Decoder**. Below it, there is a section titled **Enter UUID to Decode:** containing a text input field with the value `e6e3422c-c82d-11ed-8761-3ff799965458`. Below the input field, a note states: "UUIDs are in the format 00000000-0000-0000-0000-000000000000. [Autofill with random UUID](#)". A red button labeled **Decode UUID** with a key icon is positioned below the input field.

The **Results:** section displays the decoded information in a table. A [Copy API Call](#) button is located to the right of the table. The table contains the following data:

Standard String Format	e6e3422c-c82d-11ed-8761-3ff799965458
Single Integer Value	306902432604442536565446256077208704088
Version	1 <i>(time and node based)</i>
Variant	DCE 1.1, ISO/IEC 11578:1996
Contents - Time	2023-03-21 21:18:19.109022.0 UTC
Contents - Clock	1889 <i>(usually random)</i>
Contents - Node	3f:f7:99:96:54:58 <i>(local multicast)</i>

Password Reset Takeover : Guessable GUID / UUID

```
wman@DESKTOP-6TQ5L4U:/opt/guidtool$ guidtool -t '2023-03-22 01:30:00' e6e3422c-c82d-11ed-8761-3ff799965458
0f1fc580-c851-11ed-8761-3ff799965458
0f1fec90-c851-11ed-8761-3ff799965458
0f2013a0-c851-11ed-8761-3ff799965458
0f203ab0-c851-11ed-8761-3ff799965458
0f2061c0-c851-11ed-8761-3ff799965458
0f2088d0-c851-11ed-8761-3ff799965458
0f20afe0-c851-11ed-8761-3ff799965458
0f20d6f0-c851-11ed-8761-3ff799965458
0f20fe00-c851-11ed-8761-3ff799965458
0f212510-c851-11ed-8761-3ff799965458
0f214c20-c851-11ed-8761-3ff799965458
0f217330-c851-11ed-8761-3ff799965458
0f219a40-c851-11ed-8761-3ff799965458
0f21c150-c851-11ed-8761-3ff799965458
0f21e860-c851-11ed-8761-3ff799965458
0f220f70-c851-11ed-8761-3ff799965458
0f223680-c851-11ed-8761-3ff799965458
0f225d90-c851-11ed-8761-3ff799965458
0f2284a0-c851-11ed-8761-3ff799965458
0f22abb0-c851-11ed-8761-3ff799965458
0f22d2c0-c851-11ed-8761-3ff799965458
0f22f9d0-c851-11ed-8761-3ff799965458
0f2320e0-c851-11ed-8761-3ff799965458
0f2347f0-c851-11ed-8761-3ff799965458
0f236f00-c851-11ed-8761-3ff799965458
0f239610-c851-11ed-8761-3ff799965458
0f23bd20-c851-11ed-8761-3ff799965458
```

Reference: <https://github.com/intruder-io/guidtool>

Password Reset Takeover : Guessable GUID / UUID

[Reset your password for](#) [REDACTED]

[\[REDACTED\]/account/set-password/d144a080-5a07-11ed-9ea4-](#)
[\[REDACTED\]](#)

Thank you!

Reach out on Twitter [@sandh0t](https://twitter.com/sandh0t)
Or <https://ayoubsafa.com>