Category: Rev

Challenge: MyZoo

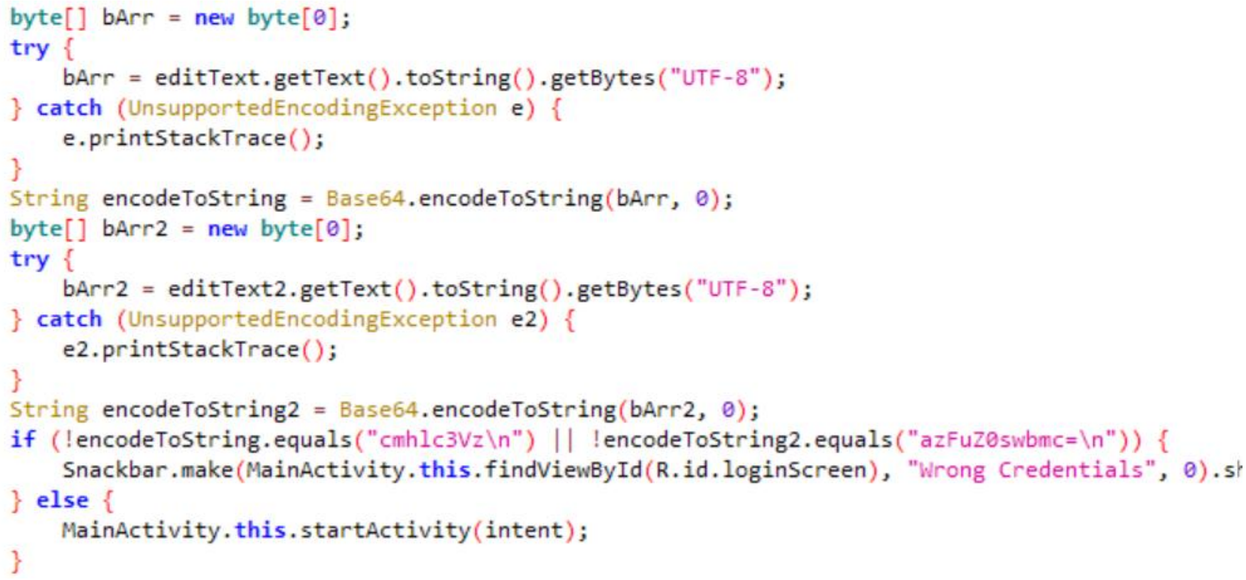
Writeup:

This is an android reverse challenge. As usual, we are opening it with jadx-gui ,installing it to the phone/ emulator and setting up proxy server.

Step 1 (finding login credentials):

We are lucky because the code is not obfuscated.

When I browse to MainActivity class in jadx I see the following code



```
byte[] bArr = new byte[0];
try {
    bArr = editText.getText().toString().getBytes("UTF-8");
} catch (UnsupportedEncodingException e) {
    e.printStackTrace();
}
String encodeToString = Base64.encodeToString(bArr, 0);
byte[] bArr2 = new byte[0];
try {
    bArr2 = editText2.getText().toString().getBytes("UTF-8");
} catch (UnsupportedEncodingException e2) {
    e2.printStackTrace();
}
String encodeToString2 = Base64.encodeToString(bArr2, 0);
if (!encodeToString.equals("cmhlc3Vz\n") || !encodeToString2.equals("azFuZ0swbmc=\n")) {
    Snackbar.make(MainActivity.this.findViewById(R.id.loginScreen), "Wrong Credentials", 0).sh
} else {
    MainActivity.this.startActivity(intent);
}
```

The login credential check does not be made in server-side. That is why the credentials are hardcoded. It is easy to understand that credentials are Base64 encoded.

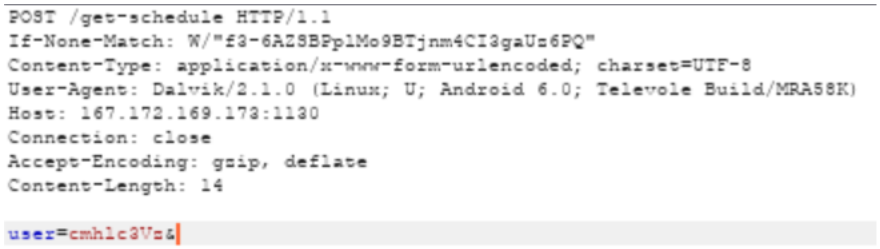
Just decoding the *cmhlc3Vz* and *azFuZ0swbmc=*

username: rhesus

password: k1ngK0ng

Step 2:

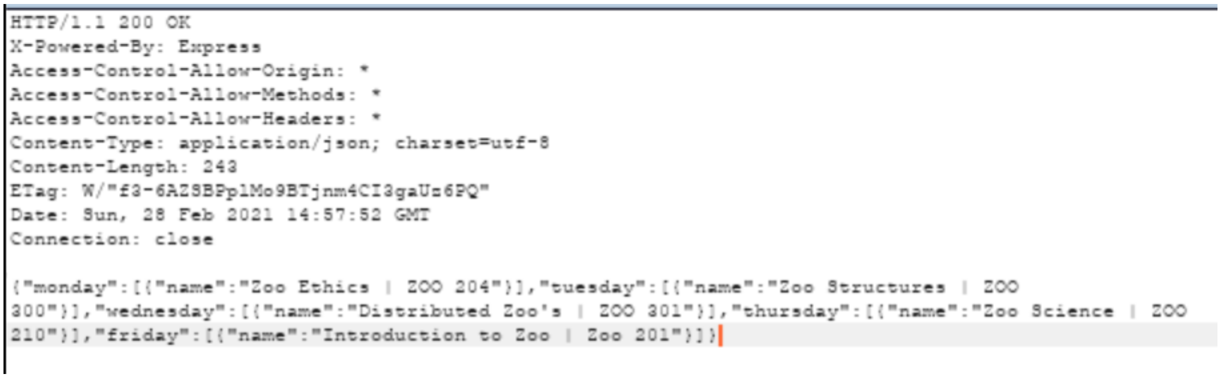
We already setup burp. When I logged in a request sent to *167.172.169.173:1130/get-schedule/*



```
POST /get-schedule HTTP/1.1
If-None-Match: W/"f3-6AZSBPplMo9BTjnm4CI3gaUz6PQ"
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
User-Agent: Dalvik/2.1.0 (Linux; U; Android 6.0; Televole Build/MRA58K)
Host: 167.172.169.173:1130
Connection: close
Accept-Encoding: gzip, deflate
Content-Length: 14

user=cmhlc3Vz&
```

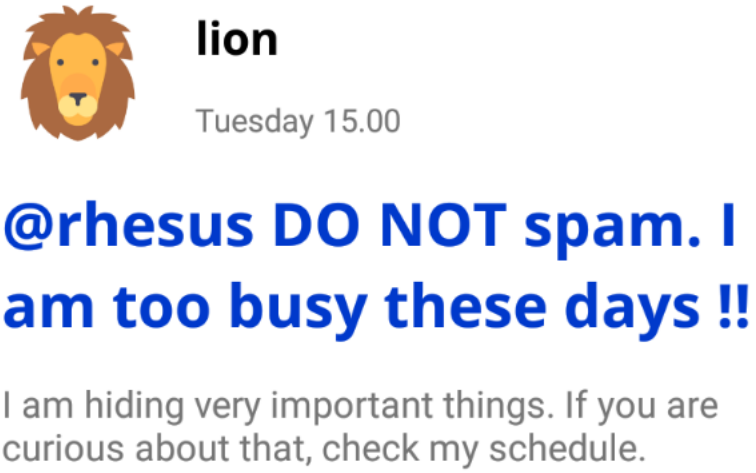
And that request returned :



```
HTTP/1.1 200 OK
X-Powered-By: Express
Access-Control-Allow-Origin: *
Access-Control-Allow-Methods: *
Access-Control-Allow-Headers: *
Content-Type: application/json; charset=utf-8
Content-Length: 243
ETag: W/"f3-6AZSBPplMo9BTjnm4CI3gaUz6PQ"
Date: Sun, 28 Feb 2021 14:57:52 GMT
Connection: close

{"monday":[{"name":"Zoo Ethics | ZOO 204"}],"tuesday":[{"name":"Zoo Structures | ZOO
300"}],"wednesday":[{"name":"Distributed Zoo's | ZOO 301"}],"thursday":[{"name":"Zoo Science | ZOO
210"}],"friday":[{"name":"Introduction to Zoo | Zoo 201"}]}
```

I noted that down and started to investigate in app. In KarpuzNet lion says:



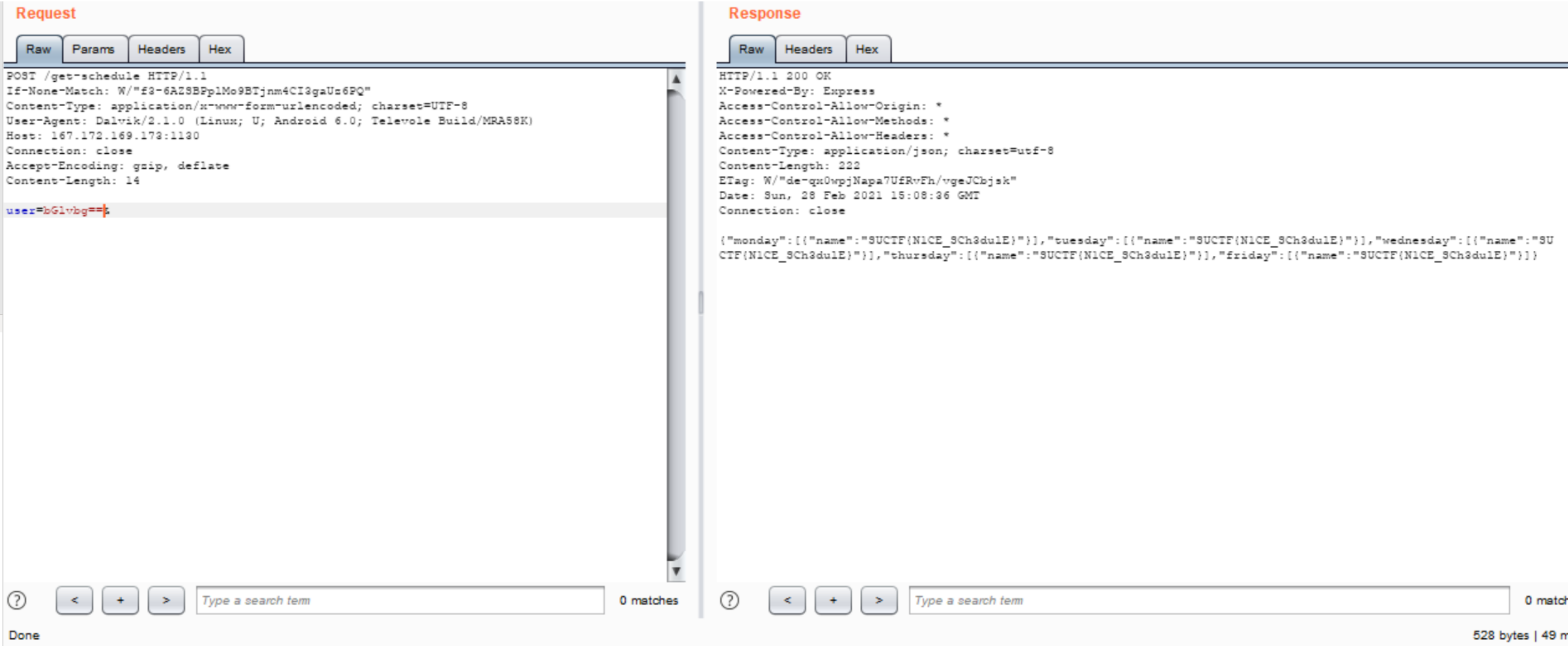
lion

Tuesday 15.00

@rhesus DO NOT spam. I am too busy these days !!

I am hiding very important things. If you are curious about that, check my schedule.

It is obvious that we should dump lion’s schedule. I replicate the get-schedule request by setting *user=bGlvbg==* because *cmhlc3Vz* is the Base64 encoded version of rhesus which is the logged in user, *bGlvbg==* is the Base64 encoded version of lion.



```
POST /get-schedule HTTP/1.1
If-None-Match: W/"f3-6AZSBPplMo9BTjnm4CI3gaUz6PQ"
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
User-Agent: Dalvik/2.1.0 (Linux; U; Android 6.0; Televole Build/MRA58K)
Host: 167.172.169.173:1130
Connection: close
Accept-Encoding: gzip, deflate
Content-Length: 14

user=bGlvbg==%
```

```
HTTP/1.1 200 OK
X-Powered-By: Express
Access-Control-Allow-Origin: *
Access-Control-Allow-Methods: *
Access-Control-Allow-Headers: *
Content-Type: application/json; charset=utf-8
Content-Length: 222
ETag: W/"de-qx0wpjNapa7UfRvFh/vgeJCbjsk"
Date: Sun, 28 Feb 2021 15:08:36 GMT
Connection: close

{"monday":[{"name":"SUCTF{N1CE_SCh3dulE}"}],"tuesday":[{"name":"SUCTF{N1CE_SCh3dulE}"}],"wednesday":[{"name":"SU
CTF{N1CE_SCh3dulE}"}],"thursday":[{"name":"SUCTF{N1CE_SCh3dulE}"}],"friday":[{"name":"SUCTF{N1CE_SCh3dulE}"}]}
```

It returned the flag. SUCTF{N1CE_SCh3dulE}