Proyecto final - Auditoría app mobile

App Android: DragonBallAndroidAvanzado

Keepcoding Cybersecurity Full Stack 4ª Edición

Red Team:
Francisco Javier García Varela
Arturo Rafael Perez López
Danylo Shved Guevska

White Team:
Carlos Cilleruelo

Índice

Introducción

La presente auditoría tiene como objetivo encontrar las vulnerabilidades de la aplicación Dragonball Android Avanzado.

El documento se divide en Análisis Estático, referido al análisis de código y configuración previo a ejecución, Análisis Dinámico, las pruebas realizadas durante el funcionamiento de la aplicación, Tráfico, donde se analizan las comunicaciones de la App, Pruebas de fuerza bruta, donde se explican los ataques de este tipo contra la API, y por último Vulnerabilidades encontradas.
A lo largo de las pruebas hemos encontrado tanto el repositorio de Github como la redirección de la página personal del creador.

Repositorios de la App

Datos extraídos de código, decompilado y captura de tráfico.
https://github.com/tecosabri/DragonBallAndroidAvanzado

Acceso por página particular del creador.
https://lotuslebanon.com/?_=%2Ftecosabri%2FDragonBallAndroidAvanzado%23%2BXqIkUpI%2B2DvlfC%2FmP8PSlSq

Servidor de la App

También las herramientas nos muestran que el servidor es de Amazon con la ip 34.241.118.168

Url servidor (amazon vps)
https://ec2-34-241-118-168.eu-west-1.compute.amazonaws.com/

Repositorio de la API revelado por las herramientas.
https://github.com/joselbe1976/kcAppsStore

Análisis Estático

MOBSF

Utilizamos la herramienta MOBSF para realizar un análisis estático de la APK.

Análisis de Manifest

MANIFEST ANALYSIS

Search:

NO	ISSUE	SEVERITY	DESCRIPTION	OPTIONS
1	App can be installed on a vulnerable Android version [minSdk=21]	warning	This application can be installed on an older version of android that has multiple unfixed vulnerabilities. Support an Android version > 8, API 26 to receive reasonable security updates.	
2	Debug Enabled For App [android:debuggable=true]	high	Debugging was enabled on the app which makes it easier for reverse engineers to hook a debugger to it. This allows dumping a stack trace and accessing debugging helper classes.	
3	Application Data can be Backed up [android:allowBackup=true]	warning	This flag allows anyone to backup your application data via adb. It allows users who have enabled USB debugging to copy application data off of the device.	

Showing 1 to 3 of 3 entries

Previous 1 Next

Análisis de Certificados

CERTIFICATE ANALYSIS

Search:

TITLE	SEVERITY	DESCRIPTION
Application signed with debug certificate	high	Application signed with a debug certificate. Production application must not be shipped with a debug certificate.
Application vulnerable to Janus Vulnerability	warning	Application is signed with v1 signature scheme, making it vulnerable to Janus vulnerability on Android 5.0-8.0, if signed only with v1 signature scheme. Applications running on Android 5.0-7.0 signed with v1, and v2/v3 scheme is also vulnerable.
Certificate algorithm might be vulnerable to hash collision	warning	Application is signed with SHA1withRSA. SHA1 hash algorithm is known to have collision issues. The manifest file indicates SHA256withRSA is in use.
Signed Application	info	Application is signed with a code signing certificate

Showing 1 to 4 of 4 entries

Previous 1 Next

Analisis de Codigo

CODE ANALYSIS

Search:

NO	ISSUE	SEVERITY	STANDARDS	FILES	OPTIONS
1	The App logs information. Sensitive information should never be logged.	info	**CWE:** CWE-532: Insertion of Sensitive Information into Log File **OWASP MASVS:** MSTG-STORAGE-3	com/isabri/dragonballandroidavanzado/ui/detail/DetailViewModel$getHeroes$1.java com/isabri/dragonballandroidavanzado/ui/login/LoginActivity$observeLoadingState$1.java junit/runner/BaseTestRunner.java junit/runner/Version.java junit/textui/TestRunner.java	
2	The App uses an insecure Random Number Generator.	warning	**CWE:** CWE-330: Use of Insufficiently Random Values **OWASP Top 10:** M5: Insufficient Cryptography **OWASP MASVS:** MSTG-CRYPTO-6	org/junit/runner/manipulation/Ordering.java	
3	App can write to App Directory. Sensitive Information should be encrypted.	info	**CWE:** CWE-276: Incorrect Default Permissions **OWASP MASVS:** MSTG-STORAGE-14	com/isabri/dragonballandroidavanzado/di/RemoteModule.java	
4	This App uses SSL certificate pinning to detect or prevent MITM attacks in secure communication channel.	secure	**OWASP MASVS:** MSTG-NETWORK-4	com/isabri/dragonballandroidavanzado/di/RemoteModule.java	
5	App creates temp file. Sensitive information should never be written into a temp file.	warning	**CWE:** CWE-276: Incorrect Default Permissions **OWASP Top 10:** M2: Insecure Data Storage **OWASP MASVS:** MSTG-STORAGE-2	coil/decode/SourceImageSource.java org/junit/rules/TemporaryFolder.java	
6	Files may contain hardcoded sensitive information like usernames, passwords, keys etc.	warning	**CWE:** CWE-312: Cleartext Storage of Sensitive Information **OWASP Top 10:** M9: Reverse Engineering **OWASP MASVS:** MSTG-STORAGE-14	coil/memory/MemoryCache.java coil/memory/MemoryCacheService.java coil/request/Parameters.java com/isabri/dragonballandroidavanzado/BuildConfig.java	

Showing 1 to 6 of 6 entries

Previous 1 Next

Tras este primer análisis podemos observar varios fallos críticos como tener habilitado el debug de la app y emplear certificados no seguros.

apktool

Se instala apktool para analizar el contenido del APK.
`sudo apt install apktool`

Una vez instalado utilizamos el siguiente comando para descompilar el APK.
`apktool d DragonBall.apk`

Una vez descompilado ya podemos ver el contenido del archivo AndroidManifest.xml.
También podemos acceder a los archivos XML que emplea la app y podemos revisar los diferentes recursos gráficos de la app.

AndroidManifest.xml

Tras un primer análisis automatizado procedemos a revisar el archivo Manifest.xml en búsqueda de diversos factores que nos permitan la explotación de la app.

En este archivo vemos el nombre que tiene el paquete una vez instalado:
com.isabri.dragonballandroidavanzado.App

Posteriormente verificamos que la aplicación está en modo debug, ya que el atributo “debuggable” está en “true”. Confirmamos también que tiene el atributo “allowBackup” en “true”, lo que permite realizar copias de seguridad. Esto puede ser un riesgo en caso de que se almacenen contraseñas o información confidencial. Como veremos más adelante, esto nos permite obtener el archivo que se crea cuando se inicia sesión donde se almacena el token que usa la app para validar el usuario.

```
14     <application android:allowBackup="true" android:appComponentFactory="androidx.core.app.CoreComponentFactory"
   android:dataExtractionRules="@xml/data_extraction_rules" android:debuggable="true"
   android:enableOnBackInvokedCallback="true" android:fullBackupContent="@xml/backup_rules" android:icon="@mipmap/
   ic_launcher" android:label="@string/app_name" android:name="com.isabri.dragonballandroidavanzado.App"
   android:roundIcon="@mipmap/ic_launcher_round" android:supportsRtl="true" android:theme="@style/
   Theme.DragonBallAndroidAvanzado">
```

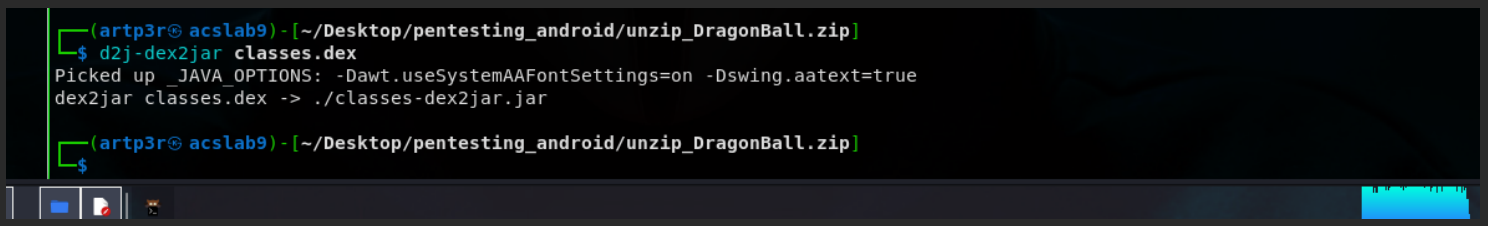
También encontramos 2 actividades exportadas, lo que nos permite invocarlas desde adb y ver que resultados devuelven.

```
15         <activity android:exported="true" android:name="com.isabri.dragonballandroidavanzado.ui.login.LoginActivity">
16             <intent-filter>
17                 <action android:name="android.intent.action.MAIN" />
18                 <category android:name="android.intent.category.LAUNCHER" />
19             </intent-filter>
20             <meta-data android:name="android.app.lib_name" android:value="" />
21         </activity>
22         <activity android:exported="true" android:name="com.isabri.dragonballandroidavanzado.ui.HeroesListActivity" android:theme="@style/
   Theme.DragonBallAndroidAvanzado.NoActionBar">
23             <meta-data android:name="android.app.lib_name" android:value="" />
24         </activity>
25         <meta-data android:name="com.google.android.geo.API_KEY" android:value="AIzaSyAbS67KgMeqOdH__Zg2VdF1YXv32QE2ODY" />
26         <uses-library android:name="org.apache.http.legacy" android:required="false" />
```

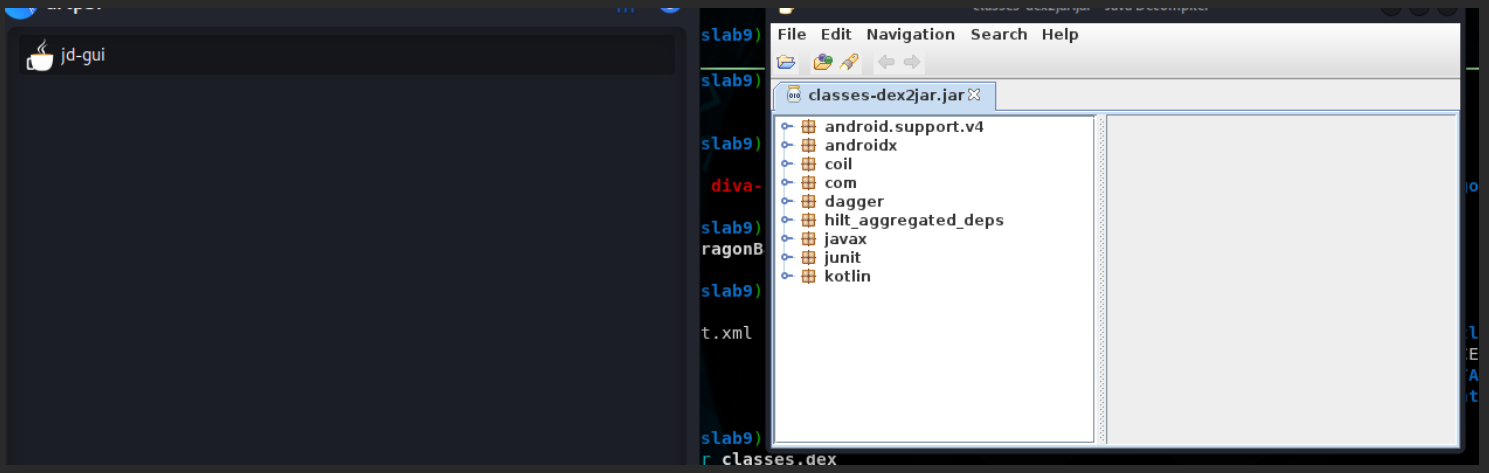
Java Decompiler

Además de apktool podemos utilizar el APK como si se tratase de un archivo comprimido. Para ello lo que haremos será cambiar la extensión APK por la extensión ZIP y descomprimir el fichero DragonBall.zip.
Para realizar esta acción usaremos el siguiente comando: `unzip DragonBall.zip` `unzip_DragonBall.zip`.

En la carpeta descomprimida podemos encontrar los archivos DEX, los cuales necesitamos convertir en archivos JAR para poder trabajar con ellos y revisarlos. Para ello emplearemos la herramienta dex2jar para transformar los archivos empleando el comando: `d2j-dex2jar classes.dex`.



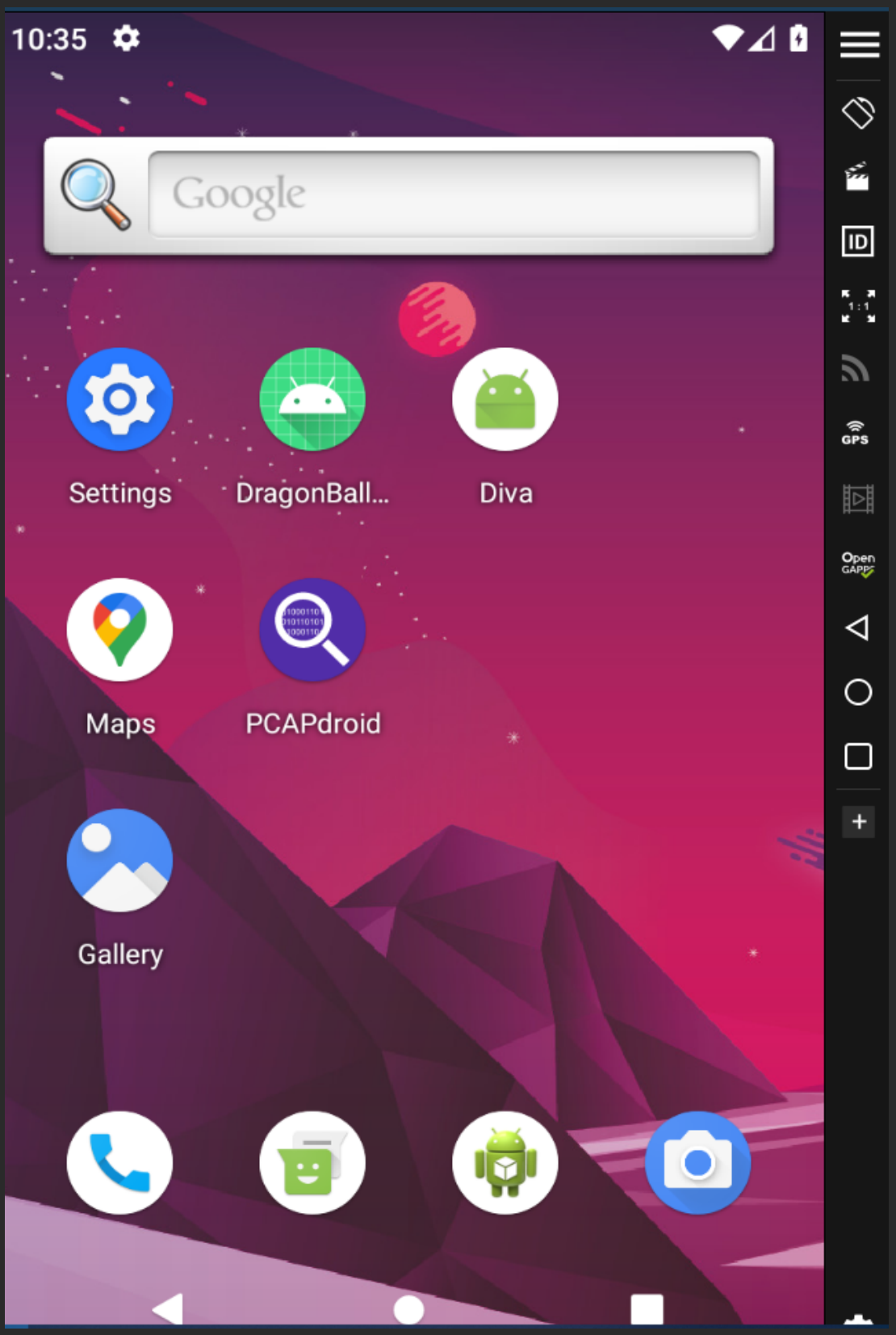
Una vez tenemos creados los archivos JAR usaremos la herramienta jd-gui para poder abrir estos ficheros y revisar todo su contenido. El objetivo de revisar estos archivos es localizar ciertos strings que usa la app.



Análisis Dinámico

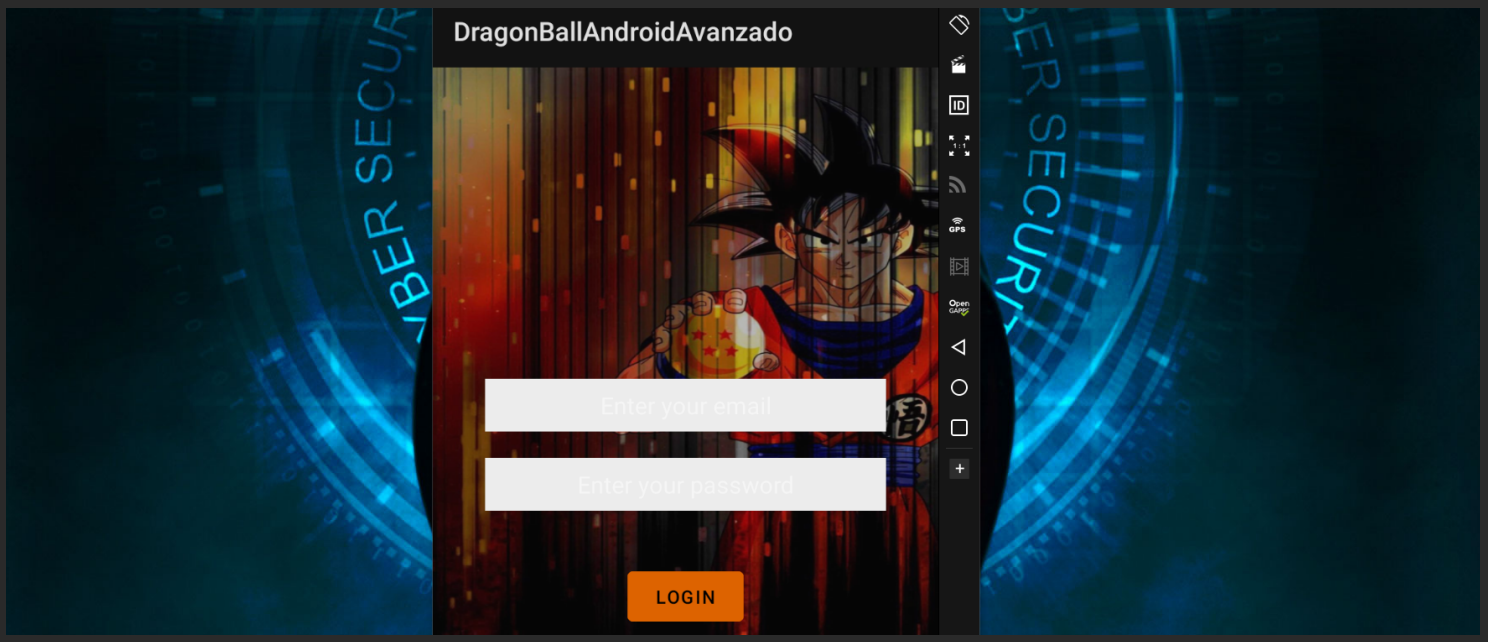
Genymotion

Genymotion es una aplicación de emulación de dispositivos móviles para Windows basada en Virtualbox. Dispone de cientos de dispositivos como plantilla para instalar Android. Para testear utilizaremos una máquina virtual de android Vbox/Genymotion en su versión más genérica. Es útil tanto como emulador para el análisis dinámico como de cara a conectarlo a BurpSuite o a Postman para analizar el tráfico.



Android Genérico

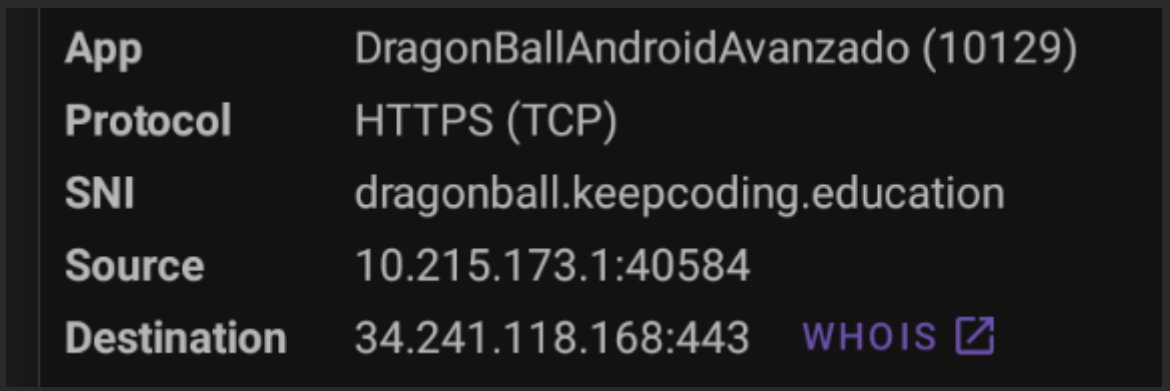
Creamos un móvil genérico Android e instalamos Open Gapps. Ya tiene instalado Google Maps por defecto, y esos son todos los requisitos para hacer funcionar la app que estamos auditando.



Desde el White Team hemos recibido la siguiente información para la auditoría:
Credencial de acceso con permisos de usuario.
bejl@keepcoding.es
123456

PCAPdroid

De forma preliminar …
Entre los procesos iniciales de la auditoría se instala PCAPdroid desde el Market de Android para sniffear tráfico.



Vemos por primera vez la api que utiliza:
https://dragonball.keepcoding.education/
10.215.173.1:40584
34.241.118.168:443
WHOIS: https://search.arin.net/rdap/?query=34.241.118.168

App: Unknown (-1)
Protocol: TCP

Source: 192.168.1.111:46120
Destination: 10.215.173.2:853

Status: Active
Traffic: 0 B received — 360 B sent
Packets: 0 received — 6 sent
Payload: 0 B
Duration: > 1 m
First seen: 02/19/23 23:14:45.847
Last seen: 02/19/23 23:15:47.937

App: Unknown (-1)
Protocol: DNS (UDP)
Query: i.annihil.us
Source: 10.215.173.1:12273
Destination: 10.215.173.2:53
Status: Closed
Traffic: 107 B received — 58 B sent
Packets: 1 received — 1 sent
Payload: 109 B
Duration: < 1 s
First seen: 02/19/23 23:15:08.710
Last seen: 02/19/23 23:15:08.735

App: Unknown (-1)
Protocol: DNS (UDP)
Query: dragonball.keepcoding.education
Source: 10.215.173.1:10311
Destination: 10.215.173.2:53
Status: Closed
Traffic: 93 B received — 77 B sent
Packets: 1 received — 1 sent
Payload: 114 B
Duration: < 1 s
First seen: 02/19/23 23:15:19.531
Last seen: 02/19/23 23:15:19.803

App: DragonBallAndroidAvanzado (10129)
Protocol: HTTPS (TCP)
SNI: dragonball.keepcoding.education
Source: 10.215.173.1:44804
Destination: 34.241.118.168:443
Status: Active
Traffic: 6.3 KB received — 2.8 KB sent
Packets: 14 received — 14 sent
Payload: 8.0 KB
Duration: < 1 s
First seen: 02/19/23 23:15:35.255
Last seen: 02/19/23 23:15:35.773

Mediante las capturas de tráfico podemos ver de forma preliminar a dónde realiza conexiones la app. Principalmente a servicios de Google para utilizar los mapas, y a páginas privadas de donde extrae las imágenes de los héroes (**i.annihil.us** , **cdn.alfabetajuega.com** y **wallpaperaccess.com**). La más destacable de las conexiones, y donde presumimos que se encuentra el objetivo es en **dragonball.keepcoding.education** .

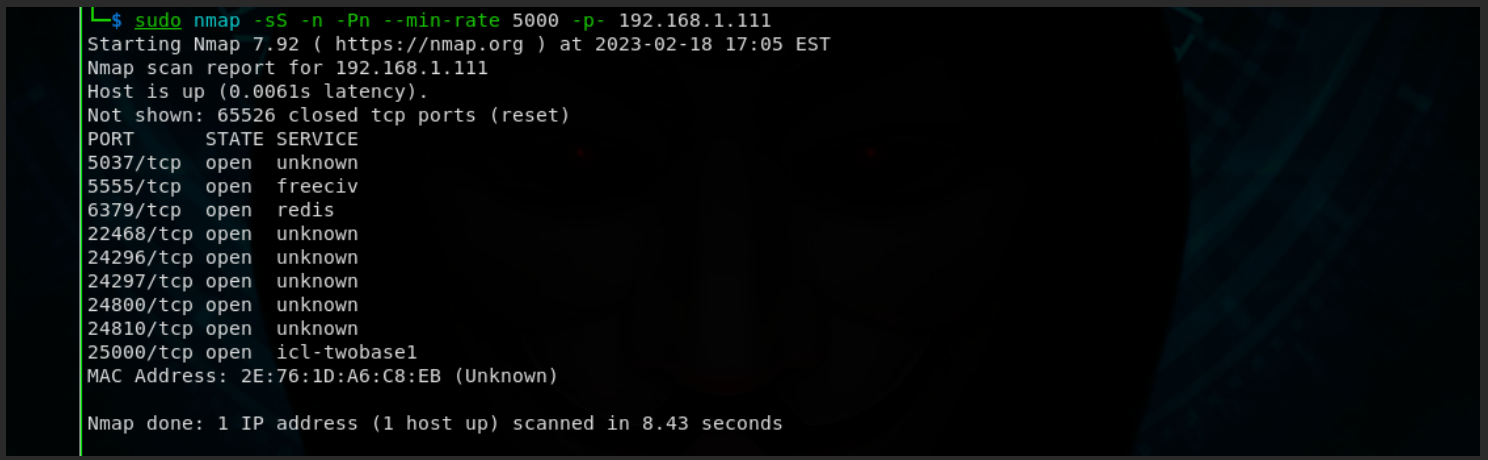
Kali Linux

Kali es nuestro principal sistema operativo para las pruebas, estando virtualizado con Virtualbox 7. Será en él donde descarguemos e instalemos las herramientas que no son exclusivas de Windows.

nmap (contra máquina Android Genérico)

```
sudo nmap -sS -n -Pn --min-rate 5000 -p- 192.168.1.111
```

Se realiza un scan de puertos tipo TCP SYN contra el android virtualizado para observar qué puertos tiene abiertos. Desactivamos resolución dns y descubrimiento de máquinas.



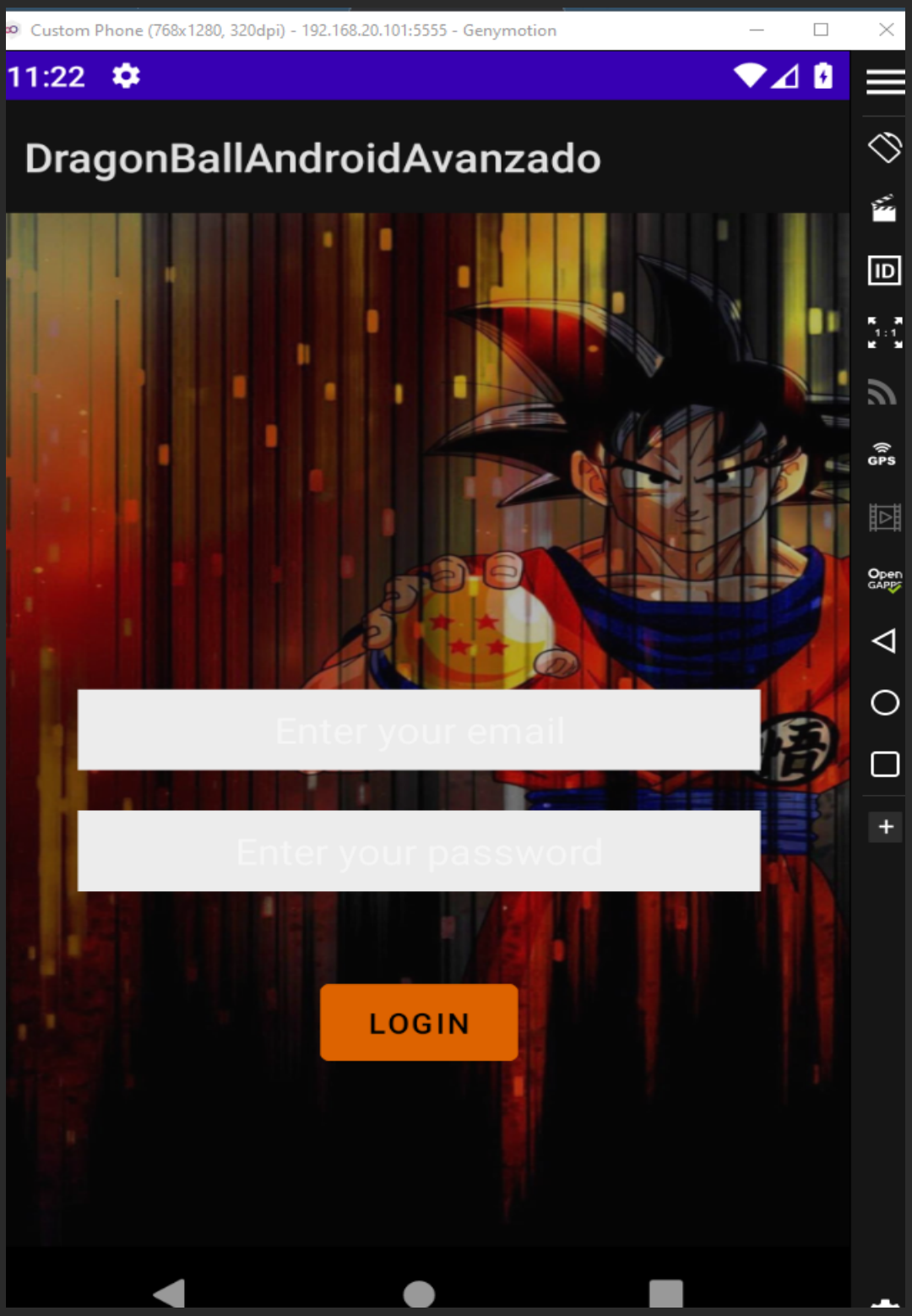
5555/tcp open freeciv es el más relevante para nosotros, debido a que es el que utilizaremos para conectarnos al android desde el kali mediante adb.

6379/tcp redis es viene de Remote Dictionary Server es una base de datos.

adb

```
adb connect 192.168.1.111
adb install DragonBall.apk
```

Mediante la herramienta adb conectamos al dispositivo android emulado y le instalamos el paquete facilitado por el white team.



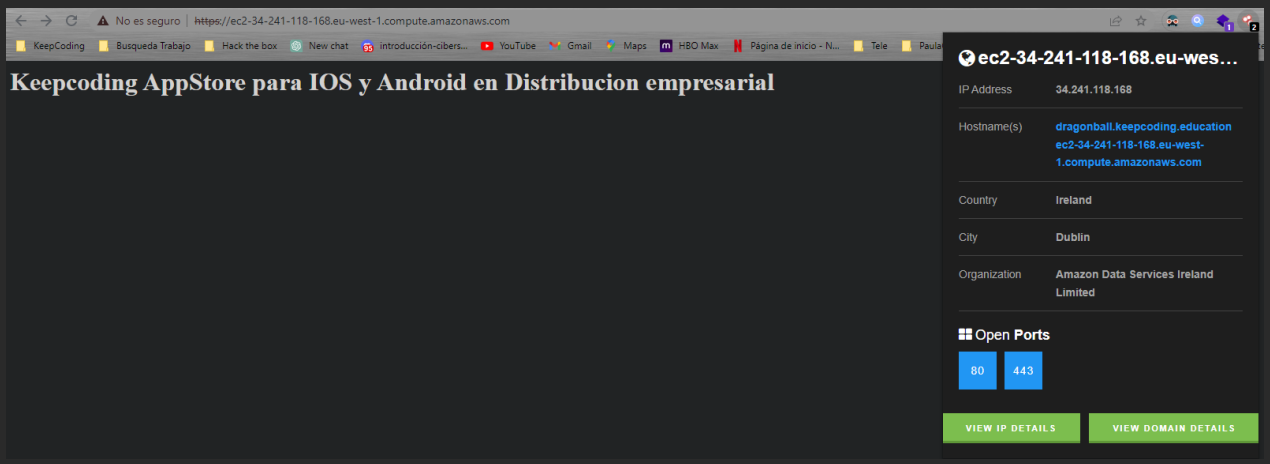
Tráfico

Keepcoding

https://dragonball.keepcoding.education/
En esta sección documentamos todas las pruebas relativas a la url excepto las de la API.

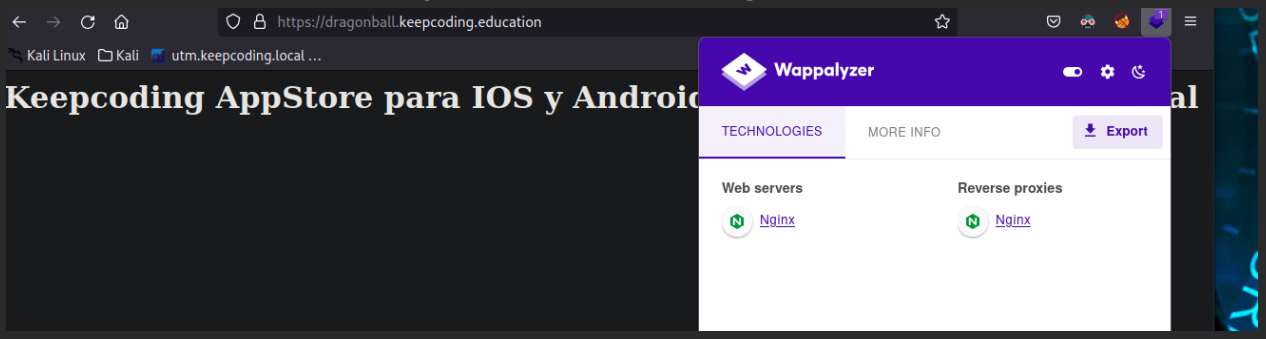
Shodan

Mediante Shodan realizamos un reconocimiento preliminar. Nos indica la dirección del servidor de AWS, la IP, geolocalización y puertos abiertos. Nada fuera de lo común.



Firefox / Wappalyzer

La primera comprobación al obtener una URL es visualizarla en el navegador y reconocer las tecnologías utilizadas en esa dirección. La página solo muestra un mensaje genérico que hemos buscado mediante Google Dorks sin resultados. La tecnología de servidor web es **nginx** aunque queda oculta su versión.



whatweb

Esta herramienta es similar a Wappalyzer y la utilizamos contra el servidor de la app.

```
└─$ whatweb https://dragonball.keepcoding.education/
https://dragonball.keepcoding.education/ [200 OK] Country[UNITED STATES][US], HTTPServer[nginx], IP[34.241.118.168], nginx
```

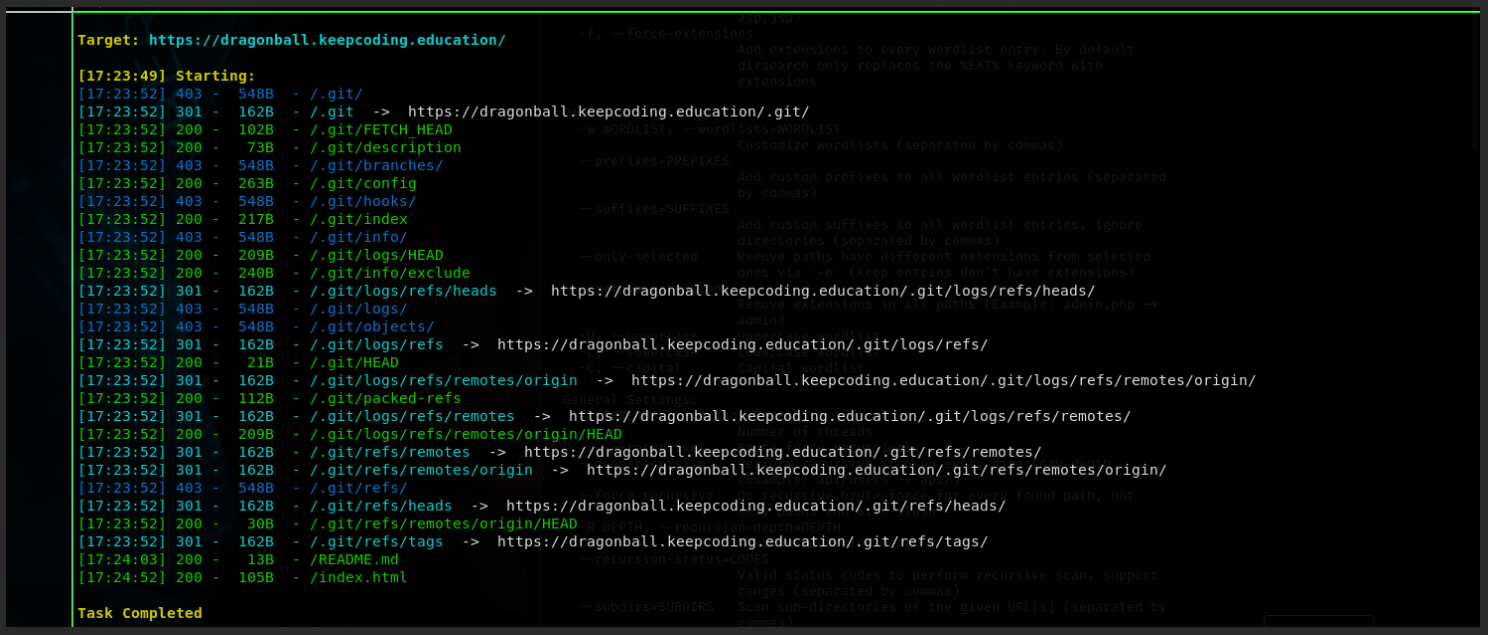
```
whatweb https://dragonball.keepcoding.education/
https://dragonball.keepcoding.education/ [200 OK] Country[UNITED STATES][US], HTTPServer[nginx], IP[34.241.118.168], nginx
```

dirsearch

Utilizamos con éxito dirsearch para descubrir directorios. Con ello exponemos una vulnerabilidad explicada en "Vulnerabilidad Improper data management"

Instalación de la herramienta en repositorio Kali.

```
sudo apt install dirsearch
dirsearch -u https://dragonball.keepcoding.education/
```




```
Target: https://dragonball.keepcoding.education/
[17:23:49] Starting:
[17:23:52] 403 -  548B  - /.git/
[17:23:52] 301 -  162B  - /.git  ->  https://dragonball.keepcoding.education/.git/
[17:23:52] 200 -  102B  - /.git/FETCH_HEAD
[17:23:52] 200 -   73B  - /.git/description
[17:23:52] 403 -  548B  - /.git/branches/
[17:23:52] 200 -  263B  - /.git/config
[17:23:52] 403 -  548B  - /.git/hooks/
[17:23:52] 200 -  217B  - /.git/index
[17:23:52] 403 -  548B  - /.git/info/
[17:23:52] 200 -  209B  - /.git/logs/HEAD
[17:23:52] 200 -  240B  - /.git/info/exclude
[17:23:52] 301 -  162B  - /.git/logs/refs/heads  ->  https://dragonball.keepcoding.education/.git/logs/refs/heads/
[17:23:52] 403 -  548B  - /.git/logs/
[17:23:52] 403 -  548B  - /.git/objects/
[17:23:52] 301 -  162B  - /.git/logs/refs  ->  https://dragonball.keepcoding.education/.git/logs/refs/
[17:23:52] 200 -   21B  - /.git/HEAD
[17:23:52]         301        -                  162B                 -        /.git/logs/refs/remotes/origin                 ->
https://dragonball.keepcoding.education/.git/logs/refs/remotes/origin/
[17:23:52] 200 -  112B  - /.git/packed-refs
[17:23:52] 301 -  162B  - /.git/logs/refs/remotes  ->  https://dragonball.keepcoding.education/.git/logs/refs/remotes/
[17:23:52] 200 -  209B  - /.git/logs/refs/remotes/origin/HEAD
[17:23:52] 301 -  162B  - /.git/refs/remotes  ->  https://dragonball.keepcoding.education/.git/refs/remotes/
[17:23:52] 301 -  162B  - /.git/refs/remotes/origin  ->  https://dragonball.keepcoding.education/.git/refs/remotes/origin/
[17:23:52] 403 -  548B  - /.git/refs/
[17:23:52] 301 -  162B  - /.git/refs/heads  ->  https://dragonball.keepcoding.education/.git/refs/heads/
[17:23:52] 200 -   30B  - /.git/refs/remotes/origin/HEAD
[17:23:52] 301 -  162B  - /.git/refs/tags  ->  https://dragonball.keepcoding.education/.git/refs/tags/
[17:24:03] 200 -   13B  - /README.md
[17:24:52] 200 -  105B  - /index.html
```


nmap TCP SYN port scan

Realizamos un primer scan de puertos TCP SYN contra el servidor de la app. En los primeros intentos no se detectan todos los puertos.

```
Not shown: 65532 filtered tcp ports (no-response)
PORT    STATE  SERVICE REASON
22/tcp  closed ssh     reset ttl 46
80/tcp  open   http    syn-ack ttl 47
443/tcp open   https   syn-ack ttl 46
```

```
sudo nmap -sS -Pn --min-rate 1000 -p- 34.241.118.168 -oA keepcoding -vvv
Starting Nmap 7.92 ( https://nmap.org ) at 2023-02-20 08:37 EST
Initiating Parallel DNS resolution of 1 host. at 08:37
Completed Parallel DNS resolution of 1 host. at 08:37, 0.41s elapsed
DNS resolution of 1 IPs took 0.41s. Mode: Async [#: 2, OK: 1, NX: 0, DR: 0, SF: 0, TR: 1, CN: 0]
Initiating SYN Stealth Scan at 08:37
Scanning ec2-34-241-118-168.eu-west-1.compute.amazonaws.com (34.241.118.168) [65535 ports]
Discovered open port 443/tcp on 34.241.118.168
Discovered open port 80/tcp on 34.241.118.168
SYN Stealth Scan Timing: About 23.62% done; ETC: 08:39 (0:01:40 remaining)
SYN Stealth Scan Timing: About 49.82% done; ETC: 08:39 (0:01:01 remaining)
Increasing send delay for 34.241.118.168 from 0 to 5 due to 20 out of 66 dropped probes since last increase.
SYN Stealth Scan Timing: About 72.69% done; ETC: 08:39 (0:00:34 remaining)
Increasing send delay for 34.241.118.168 from 5 to 10 due to 11 out of 11 dropped probes since last increase.
Increasing send delay for 34.241.118.168 from 10 to 20 due to 11 out of 15 dropped probes since last increase.
Completed SYN Stealth Scan at 08:39, 126.60s elapsed (65535 total ports)
Nmap scan report for ec2-34-241-118-168.eu-west-1.compute.amazonaws.com (34.241.118.168)
Host is up, received user-set (0.046s latency).
Scanned at 2023-02-20 08:37:17 EST for 127s
Not shown: 65532 filtered tcp ports (no-response)
PORT    STATE  SERVICE REASON
22/tcp  closed ssh     reset ttl 46
80/tcp  open   http    syn-ack ttl 47
443/tcp open   https   syn-ack ttl 46

Read data files from: /usr/bin/../share/nmap
Nmap done: 1 IP address (1 host up) scanned in 127.49 seconds
           Raw packets sent: 131164 (5.771MB) | Rcvd: 56 (2.424KB)
```

Lanzamos de nuevo el nmap y vemos nuevos puertos abiertos:

```
sudo nmap -sS -n -Pn --min-rate 5000 -p- 34.241.118.168
```

```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -sS -n -Pn --min-rate 5000 -p- 34.241.118.168
[sudo] password for kali:
Starting Nmap 7.93 ( https://nmap.org ) at 2023-02-23 16:18 EST
Nmap scan report for 34.241.118.168
Host is up (0.053s latency).
Not shown: 65529 filtered tcp ports (no-response)
PORT     STATE  SERVICE
22/tcp   closed ssh
53/tcp   open   domain
80/tcp   open   http
443/tcp  open   https
2854/tcp open   infomover
5432/tcp closed postgresql

Nmap done: 1 IP address (1 host up) scanned in 26.51 seconds
```

Tras ver los puertos que localiza tratamos de descubrir las versiones:

```
sudo nmap -sS -sV -n -Pn --min-rate 5000 -p 22,53,80,443,2854,5432
34.241.118.168
```

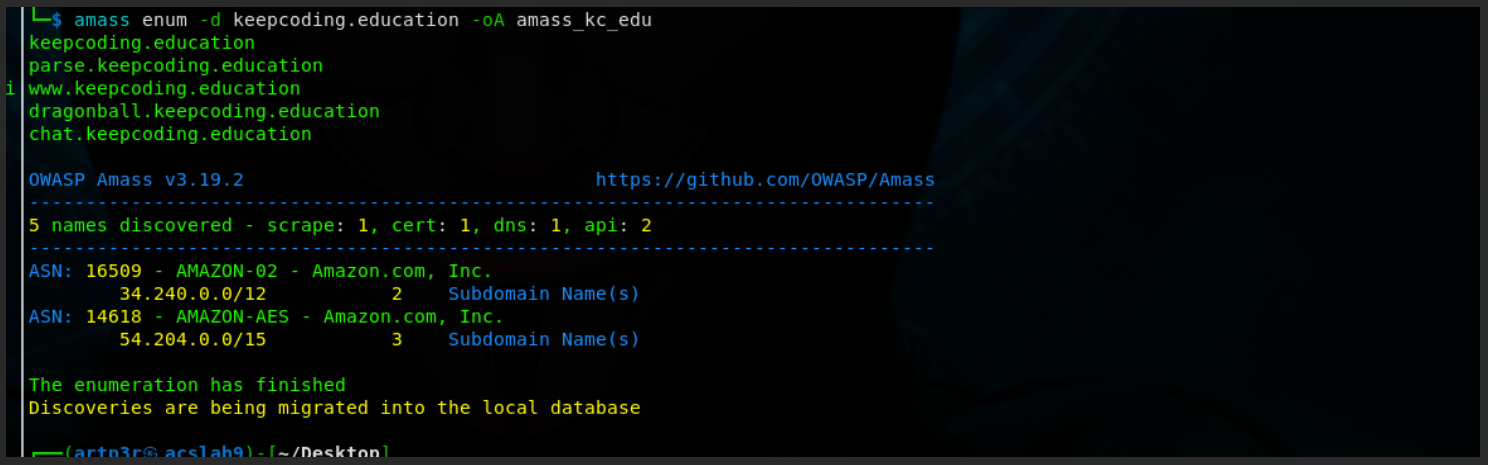
```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -sS -sV -n -Pn --min-rate 5000 -p 22,53,80,443,2854,5432 34.241.118.168
Starting Nmap 7.93 ( https://nmap.org ) at 2023-02-23 16:19 EST
Nmap scan report for 34.241.118.168
Host is up (0.023s latency).
ists/dirbuster/directory-list-2.3-small.txt
PORT     STATE  SERVICE    VERSION
22/tcp   closed ssh
53/tcp   open   domain     dnsmasq 2.78
80/tcp   open   http       nginx
443/tcp  open   ssl/http   nginx
2854/tcp open   ssh        OpenSSH 8.2p1 Ubuntu 4ubuntu0.5 (Ubuntu Linux; protocol 2.0)
5432/tcp closed postgresql
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 14.02 seconds
```

amass

Realizamos un reconocimiento de amplio espectro con Amass. De aquí sacamos todos los subdominios de **keepcoding.education** , abajo marcados en amarillo.

amass enum -d keepcoding.education -oA amass_kc_edu



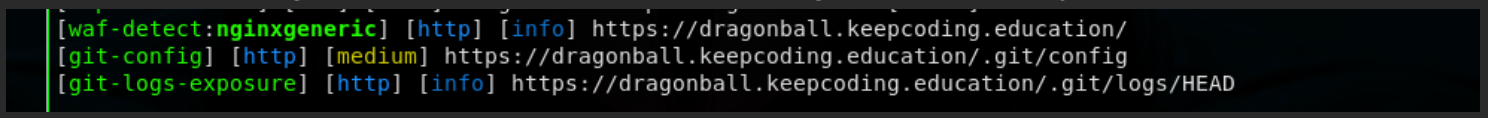
keepcoding.education
parse.keepcoding.education
www.keepcoding.education
dragonball.keepcoding.education
chat.keepcoding.education

```
OWASP Amass v3.19.2                                  https://github.com/OWASP/Amass
--------------------------------------------------------------------------------
5 names discovered - scrape: 1, cert: 1, dns: 1, api: 2
--------------------------------------------------------------------------------
ASN: 16509 - AMAZON-02 - Amazon.com, Inc.
      34.240.0.0/12     2    Subdomain Name(s)
ASN: 14618 - AMAZON-AES - Amazon.com, Inc.
      54.204.0.0/15     3    Subdomain Name(s)

The enumeration has finished
Discoveries are being migrated into the local database
```

nuclei

Con nuclei hacemos un fingerprint, ya subiendo el nivel y el grado de ruido. No da resultados especialmente interesantes pero si algo más de información que podemos seguir revisando (abajo marcado en amarillo).




```
nuclei -u https://dragonball.keepcoding.education/

                     __     _
   ____  __  _______/ /__  (_)
  / __ \/ / / / ___/ / _ \/ /
 / / / / /_/ / /__/ /  __/ /
/_/ /_/\__,_/\___/_/\___/_/   v2.8.9

                projectdiscovery.io

[INF] Using Nuclei Engine 2.8.9 (latest)
[INF] Using Nuclei Templates 9.3.7 (latest)
[INF] Templates added in last update: 58
[INF] Templates loaded for scan: 4927
[INF] Targets loaded for scan: 1
[INF] Templates clustered: 980 (Reduced 901 Requests)
[INF] Using Interactsh Server: oast.online
[tech-detect:nginx] [http] [info] https://dragonball.keepcoding.education/
[ssl-dns-names] [ssl] [info] dragonball.keepcoding.education [dragonball.keepcoding.education]
[ssl-issuer] [ssl] [info] dragonball.keepcoding.education [Let's Encrypt]
[tls-version] [ssl] [info] dragonball.keepcoding.education [tls13]
[http-missing-security-headers:x-content-type-options]                  [http]                  [info]
https://dragonball.keepcoding.education/
[http-missing-security-headers:clear-site-data] [http] [info] https://dragonball.keepcoding.education/
[http-missing-security-headers:cross-origin-embedder-policy]               [http]               [info]
https://dragonball.keepcoding.education/
[http-missing-security-headers:access-control-allow-origin]               [http]               [info]
https://dragonball.keepcoding.education/
[http-missing-security-headers:access-control-allow-credentials]             [http]             [info]
https://dragonball.keepcoding.education/
[http-missing-security-headers:access-control-expose-headers]              [http]              [info]
https://dragonball.keepcoding.education/
[http-missing-security-headers:access-control-allow-headers]               [http]              [info]
https://dragonball.keepcoding.education/
[http-missing-security-headers:strict-transport-security]                 [http]                 [info]
https://dragonball.keepcoding.education/
[http-missing-security-headers:permissions-policy]                      [http]                  [info]
https://dragonball.keepcoding.education/
[http-missing-security-headers:access-control-max-age]                  [http]                  [info]
https://dragonball.keepcoding.education/
[http-missing-security-headers:content-security-policy]                  [http]                 [info]
https://dragonball.keepcoding.education/
[http-missing-security-headers:x-frame-options] [http] [info] https://dragonball.keepcoding.education/
[http-missing-security-headers:x-permitted-cross-domain-policies]             [http]             [info]
https://dragonball.keepcoding.education/
[http-missing-security-headers:referrer-policy] [http] [info] https://dragonball.keepcoding.education/
[http-missing-security-headers:cross-origin-opener-policy]                [http]                [info]
https://dragonball.keepcoding.education/
[http-missing-security-headers:cross-origin-resource-policy]              [http]                [info]
https://dragonball.keepcoding.education/
[http-missing-security-headers:access-control-allow-methods]              [http]                [info]
https://dragonball.keepcoding.education/
[deprecated-tls] [ssl] [info] dragonball.keepcoding.education [tls12]
[deprecated-tls] [ssl] [info] dragonball.keepcoding.education [tls12]
[deprecated-tls] [ssl] [info] dragonball.keepcoding.education [tls12]
[waf-detect:nginxgeneric] [http] [info] https://dragonball.keepcoding.education/
```

```
[git-config] [http] [medium] https://dragonball.keepcoding.education/.git/config
[git-logs-exposure] [http] [info] https://dragonball.keepcoding.education/.git/logs/HEAD
```

nmap - scan de scripts básicos y versiones de servicios

Profundizamos con nmap en el scan de puertos detectados, buscando más información de los servicios y sus versiones.

```
nmap -sCV -p22,80,443 34.241.118.168
```

```
└─$ nmap -sCV -p22,80,443 34.241.118.168
Starting Nmap 7.92 ( https://nmap.org ) at 2023-02-21 18:24 EST
Nmap scan report for ec2-34-241-118-168.eu-west-1.compute.amazonaws.com (34.241.118.168)
Host is up (0.045s latency).

PORT    STATE  SERVICE  VERSION
22/tcp  closed ssh
80/tcp  open   http     nginx
|_http-title: 404 Not Found
443/tcp open   ssl/http nginx
| ssl-cert: Subject: commonName=dragonball.keepcoding.education
| Subject Alternative Name: DNS:dragonball.keepcoding.education
| Not valid before: 2023-01-19T07:15:09
|_Not valid after:  2023-04-19T07:15:08
|_http-title: Site doesn't have a title (text/html).
| http-git:
|   34.241.118.168:443/.git/
|     Git repository found!
|     Repository description: Unnamed repository; edit this file 'description' to name the...
|     Remotes:
|_      https://github.com/joselbe1976/kcAppsStore
|_ssl-date: TLS randomness does not represent time

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 17.70 seconds
```

Encontramos un **repositorio de Git**. Lo podemos descargar con una herramienta para revisarlo en local.

```
2854/tcp open     ssh        OpenSSH 8.2p1 Ubuntu 4ubuntu0.5 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 fe:66:a4:e5:bb:8b:cb:78:76:0c:2a:63:6c:a2:87:d9 (RSA)
|   256 fe:63:f3:10:ae:04:08:d6:86:c9:e6:d2:d4:c0:0e:47 (ECDSA)
|_  256 37:db:45:3a:c2:64:34:92:b6:a7:02:d5:dd:48:c8:11 (ED25519)
5432/tcp closed   postgresql
```

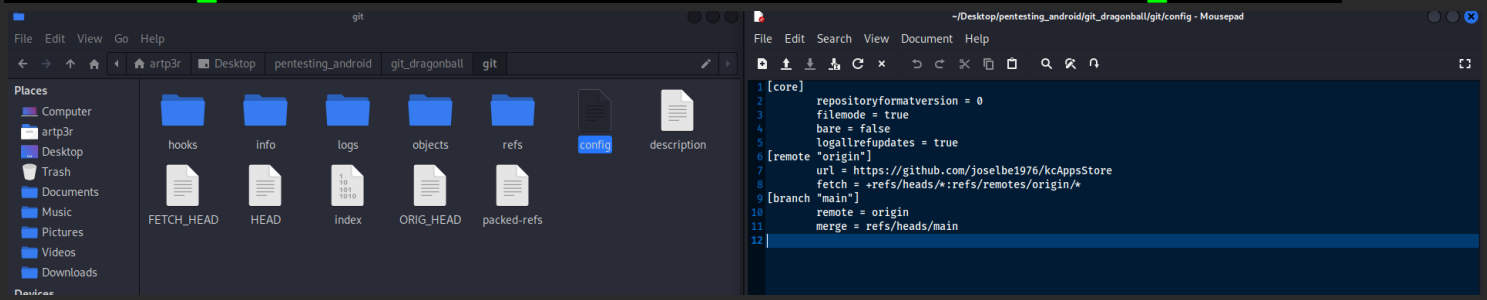
También extraemos las key de ssh y siempre se muestra en el scan el puerto de PostgreSQL que aparece cerrado.

git-dumper

Esta herramienta se utiliza para descargar un repositorio completo de Git desde un servidor remoto utilizando el protocolo HTTP o SSH.

https://github.com/arthaud/git-dumper

```
python3 git_dumper.py https://dragonball.keepcoding.education git_dragonball
```



Con git-dumper descargamos el repositorio git de la web y encontramos algún dato interesante. El fichero config entre otros contiene datos de este repositorio.

```
[core]
	repositoryformatversion = 0
	filemode = true
	bare = false
	logallrefupdates = true
[remote "origin"]
	url = https://github.com/joselbe1976/kcAppsStore
	fetch = +refs/heads/*:refs/remotes/origin/*
[branch "main"]
	remote = origin
	merge = refs/heads/main
```

---

gobuster

Usamos la herramienta gobuster para intentar descubrir más subdominios no listados en DNS, usando una lista estándar de **SecLists.**

```
gobuster vhost -u https://dragonball.keepcoding.education -w /usr/share/wordlists/amass/subdomains-top1mil-20000.txt --append-domain
```

```
┌──(root㉿kali)-[/home/kali]
└─# gobuster vhost -u https://dragonball.keepcoding.education -w /usr/share/wordlists/amass/subdomains-top1mil-20000.txt --append-domain
===============================================================
Gobuster v3.4
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:          https://dragonball.keepcoding.education
[+] Method:       GET
[+] Threads:      10
[+] Wordlist:     /usr/share/wordlists/amass/subdomains-top1mil-20000.txt
[+] User Agent:   gobuster/3.4
[+] Timeout:      10s
[+] Append Domain: true
===============================================================
2023/02/23 16:27:09 Starting gobuster in VHOST enumeration mode
===============================================================
Found: m.dragonball.keepcoding.education Status: 400 [Size: 150]
Found: ns2.cl.bellsouth.net.dragonball.keepcoding.education Status: 400 [Size: 150]
Found: ns1.viviotech.net.dragonball.keepcoding.education Status: 400 [Size: 150]
Found: ns2.viviotech.net.dragonball.keepcoding.education Status: 400 [Size: 150]
Found: ns3.cl.bellsouth.net.dragonball.keepcoding.education Status: 400 [Size: 150]
Found: ferrari.fortwayne.com.dragonball.keepcoding.education Status: 400 [Size: 150]
Found: jordan.fortwayne.com.dragonball.keepcoding.education Status: 400 [Size: 150]
Found: quatro.oweb.com.dragonball.keepcoding.education Status: 400 [Size: 150]
Found: c.ns.emailvision.net.dragonball.keepcoding.education Status: 400 [Size: 150]
Found: a.ns.emailvision.net.dragonball.keepcoding.education Status: 400 [Size: 150]
Found: b.ns.emailvision.net.dragonball.keepcoding.education Status: 400 [Size: 150]
Found: d.ns.emailvision.net.dragonball.keepcoding.education Status: 400 [Size: 150]
Found: mail.dragonball.keepcoding.education Status: 400 [Size: 150]
Found: ns2.simpleviewinc.com.dragonball.keepcoding.education Status: 400 [Size: 150]
Found: ns1.simpleviewinc.com.dragonball.keepcoding.education Status: 400 [Size: 150]
Found: ns-1.open.ro.dragonball.keepcoding.education Status: 400 [Size: 150]
Found: ns-3.open.ro.dragonball.keepcoding.education Status: 400 [Size: 150]
Found: ns-2.open.ro.dragonball.keepcoding.education Status: 400 [Size: 150]
Found: ns1.twtelecom.net.dragonball.keepcoding.education Status: 400 [Size: 150]
Found: ns2.twtelecom.net.dragonball.keepcoding.education Status: 400 [Size: 150]
Found: dns1.freshegg.net.dragonball.keepcoding.education Status: 400 [Size: 150]
Found: pns.dtag.de.dragonball.keepcoding.education Status: 400 [Size: 150]
Found: easydns2.dualtec.com.br.dragonball.keepcoding.education Status: 400 [Size: 150]
Found: easydns1.dualtec.com.br.dragonball.keepcoding.education Status: 400 [Size: 150]
Progress: 20000 / 20001 (100.00%)
===============================================================
2023/02/23 16:29:23 Finished
===============================================================
```

Como se observa en la imagen anterior, se detectan varios posibles destinos, aunque todos ellos con el mismo resultado, siendo este error 400. Esto puede ser debido a que la dirección dragonball.keepcoding.education resuelve en una ip de AWS, en la cual seguramente se hayan alojado previamente diferentes dominios y subdominios no relacionados con nuestro objetivo.

API

De las pruebas que hemos ido haciendo con Burp y herramientas de captura y manipulación de tráfico, hemos extraído las peticiones para las diferentes funcionalidades de la app que se comunican mediante una API. Esta API atiende a las siguientes direcciones:

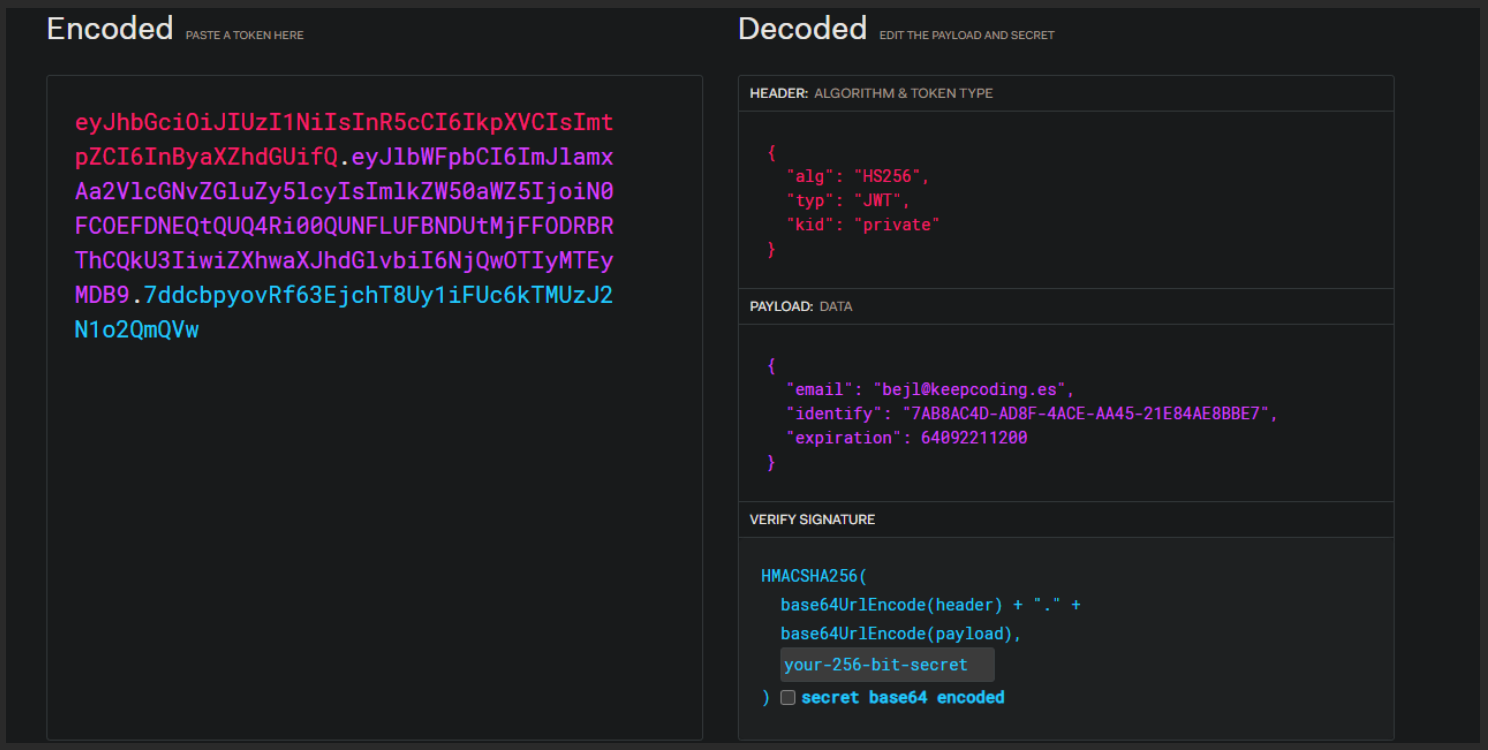
/api/auth/login

Esta dirección se utiliza para la autenticación de usuario, recibiendo una cabecera *Authorization* con el contenido *Basic* y las credenciales en formato base64. En la respuesta se devuelve en el *body* un JWT que la app utilizará en el resto de comunicaciones con la cabecera *Authorization: Bearer <JWT>*.

Petición: (solo cabeceras)
POST /api/auth/login HTTP/1.1
Host: dragonball.keepcoding.education
Content-Type: Application/Json
Authorization: Basic YmVqbEBrZWVwY29kaW5nLmVzOjEyMzQ1Ng==
Content-Length: 0
Accept-Encoding: gzip, deflate
User-Agent: okhttp/5.0.0-alpha.3
Connection: close

Debajo deben ir dos líneas en blanco, una como separación de las cabeceras y otra para indicar que no hay cuerpo ya que la API solo necesita el dato contenido en Authorization.

El JSON Web Token contiene lo siguiente:



HEADER:ALGORITHM & TOKEN TYPE
{
"alg": "HS256",
"typ": "JWT",
"kid": "private"
}
PAYLOAD:DATA
{
"email": "bejl@keepcoding.es",
"identify": "7AB8AC4D-AD8F-4ACE-AA45-21E84AE8BBE7",

```
 "expiration": 64092211200
}
VERIFY SIGNATURE
```

```
HMACSHA256(
  base64UrlEncode(header) + "." +
  base64UrlEncode(payload),
```

```
) secret base64 encoded
```

La respuesta incluye en su cuerpo el JWT que utilizará la app para autenticar cada petición mediante la cabecera *Authorization*. Con ella la app por defecto llama a la api de nuevo, en la siguiente dirección.

/api/heros/all

Esta dirección se utiliza para extraer datos de la base de datos de heroes, previa autenticación mediante el JWT recibido en la dirección arriba indicada, aunque hemos observado en todas las herramientas que la app intenta autenticar estas peticiones de forma duplicada de ambas formas, mediante las credenciales básicas (sin obtener resultado) y mediante el JWT generado y almacenado de forma local que da pie a varias vulnerabilidades (ver en el apartado de vulnerabilidades).

El cuerpo de las peticiones debe ser en formato JSON y debe incluir solo el dato *name* para buscar en la base de datos remota todos los héroes que contengan en su nombre la cadena indicada por el anterior dato. Y en la respuesta incluye el resto de datos.

Se ha intentado inyectar sql en diversos formatos sin éxito.

Aquí el comportamiento detectado, de enviar todas las peticiones duplicadas a la API. La primera de ellas utiliza la autorización básica y recibe un error 401 (Unauthorized), y la segunda, con autorización Bearer, ya devuelve la respuesta correcta. Esto más que una vulnerabilidad es un fallo de diseño.

4	https://dragonball.keepcoding.e...	POST	/api/heros/all	✓	401	205	JSON
5	https://dragonball.keepcoding.e...	POST	/api/heros/all	✓	200	13862	JSON
28	https://dragonball.keepcoding.e...	POST	/api/heros/all	✓	401	205	JSON
29	https://dragonball.keepcoding.e...	POST	/api/heros/all	✓	200	1064	JSON
31	https://dragonball.keepcoding.e...	POST	/api/heros/locations	✓	401	205	JSON
33	https://dragonball.keepcoding.e...	POST	/api/heros/locations	✓	200	1907	JSON
41	https://play.googleapis.com	POST	/play/log?format=raw&proto_v2=true	✓	200	558	text
50	https://dragonball.keepcoding.e...	POST	/api/heros/all	✓	401	205	JSON
51	https://dragonball.keepcoding.e...	POST	/api/heros/all	✓	200	953	JSON
52	https://dragonball.keepcoding.e...	POST	/api/heros/locations	✓	401	205	JSON
53	https://dragonball.keepcoding.e...	POST	/api/heros/locations	✓	200	360	JSON
59	https://play.googleapis.com	POST	/play/log?format=raw&proto_v2=true	✓	200	558	text

Authorization: Basic <credencial base64>

Request

Pretty Raw Hex \n Select extension...

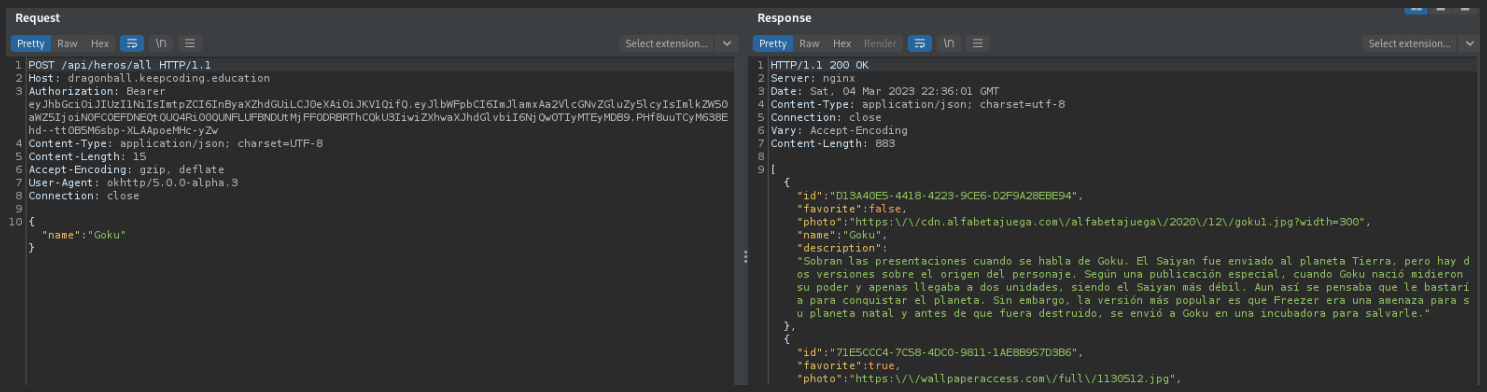
```
1 POST /api/heros/all HTTP/1.1
2 Host: dragonball.keepcoding.education
3 Authorization: Basic YmVqbEBrZWVwY29kaW5nLmVzOjEyMzQ1Ng==
4 Content-Type: application/json; charset=UTF-8
5 Content-Length: 15
6 Accept-Encoding: gzip, deflate
7 User-Agent: okhttp/5.0.0-alpha.3
8 Connection: close
9
10 {
   "name":"Goku"
  }
```

Response

Pretty Raw Hex Render \n

```
1 HTTP/1.1 401 Unauthorized
2 Server: nginx
3 Date: Sat, 04 Mar 2023 22:36:01 GMT
4 Content-Type: application/json; charset=utf-8
5 Content-Length: 38
6 Connection: close
7
8 {
   "reason":"Unauthorized",
   "error":true
  }
```

Authorization: Bearer <JWT>



/api/heros/locations

Esta dirección se utiliza para obtener las localizaciones donde se ha visto al héroe la última vez. Al igual que la anterior solo exige un dato en formato JSON, la ID del héroe en la base de datos, y en la respuesta se devuelve la información de localización en el mismo formato. Estos datos son utilizados por la app para conectar con la API de Google Maps que permite emplazar las coordenadas en el mapa de Google.

Petición:

```
Authorization: Bearer
eyJhbGciOiJIUzI1NiIsImtpZCI6InByaXZhdGUiLCJ0eXAiOiJKV1QifQ.eyJlbWFpbCI6ImJlamxAa2VlcGNvZGluZy5lcyIsImlkZW50aWZ5IjoiN0FCOEFDNEQtQUQ4Ri00QUNFLUFBNDUtMjFFODRBRThCQkU3IiwiZXhwaXJhdGlvbiI6NjQwOTIyMTEyMDB9.PHf8uuTCyM638Ehd--tt0B5M6sbp-XLAApoeMHc-yZw
Content-Type: application/json; charset=UTF-8
Content-Length: 45
Accept-Encoding: gzip, deflate
User-Agent: okhttp/5.0.0-alpha.3
Connection: close

{
  "id":"EA0D9204-9894-4A86-B7F1-92DDBBC8BD23"
}
```

Respuesta:

```
iTCyM638E
5 Connection: close
6 Vary: Accept-Encoding
7 Content-Length: 179
8
9 [
    {
      "id":"640B6466-28A8-4D07-889F-DE66CF4CBA12",
      "hero":{
        "id":"EA0D9204-9894-4A86-B7F1-92DDBBC8BD23"
      },
      "latitud":"42.645815",
      "longitud":"18.058942",
      "dateShow":"2022-09-26T00:00:00Z"
    }
]
```

/api/data/herolike

Esta dirección se utiliza para dar like o favorito al héroe. Es la menos útil y a la vez menos vulnerable de las funciones de la API, ya que solo envía un parámetro y se confirma en la respuesta con un código 201.

Request

Pretty | Raw | Hex | \n | Select extension...

```
1  POST /api/data/herolike HTTP/1.1
2  Host: dragonball.keepcoding.education
3  Authorization: Bearer
   eyJhbGciOiJIUzI1NiIsImtpZCI6InByaXZhdGUiLCJ0eXAiOiJKV1QifQ.ey
   JlbWFpbCI6ImJlamxAa2VlcGNvZGluZy5lcyIsImlkZW50aWZ5IjoiN0FC0EF
   DNEQtQUQ4Ri00QUNFLUFBNDUtMjFFODRBRThCQkU3IiwiZXhwaXJhdGlvbiI6
   NjQwOTIyMTEyMDB9.PHf8uuTCyM638Ehd--tt0B5M6sbp-XLAApoeMHc-yZw
4  Content-Type: application/json; charset=UTF-8
5  Content-Length: 47
6  Accept-Encoding: gzip, deflate
7  User-Agent: okhttp/5.0.0-alpha.3
8  Connection: close
9
10 {
     "hero":"14BB8E98-6586-4EA7-B4D7-35D6A63F5AA3"
   }
```

Response

Pretty | Raw | Hex | Render | \n

```
1  HTTP/1.1 201 Created
2  Server: nginx
3  Date: Tue, 07 Mar 2023 19:54:03 GMT
4  Content-Length: 0
5  Connection: close
6
7
```

Pruebas de fuerza bruta

Fuzzing de credenciales

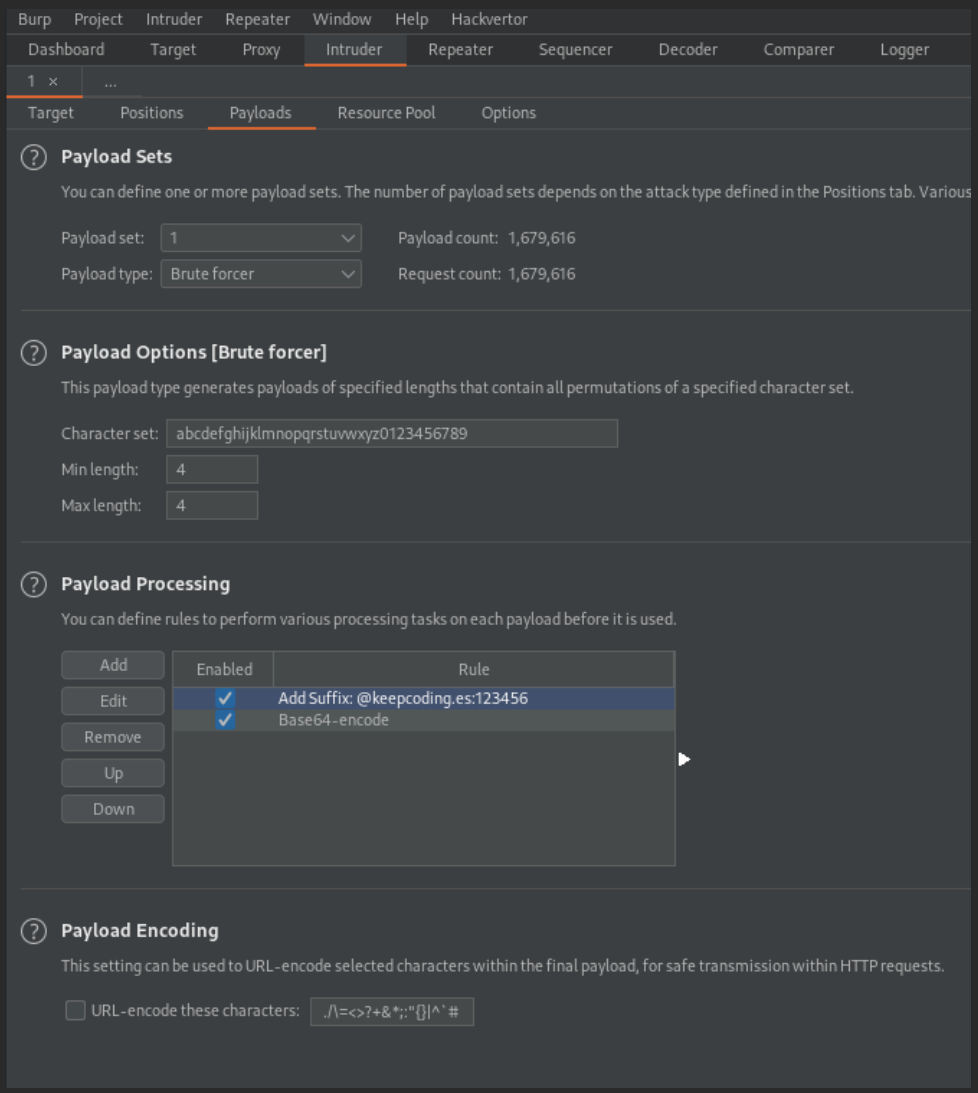
Extraemos la petición de login y la llevamos al intruder de BurpSuite.

Intruder - BurpSuite

POST /api/auth/login HTTP/1.1
Host: dragonball.keepcoding.education
Content-Type: Application/Json
Authorization: Basic §YmVqbEBrZWVwY29kaW5nLmVzOjEyMzQ1Ng==§
Content-Length: 0
Accept-Encoding: gzip, deflate
User-Agent: okhttp/5.0.0-alpha.3
Connection: close

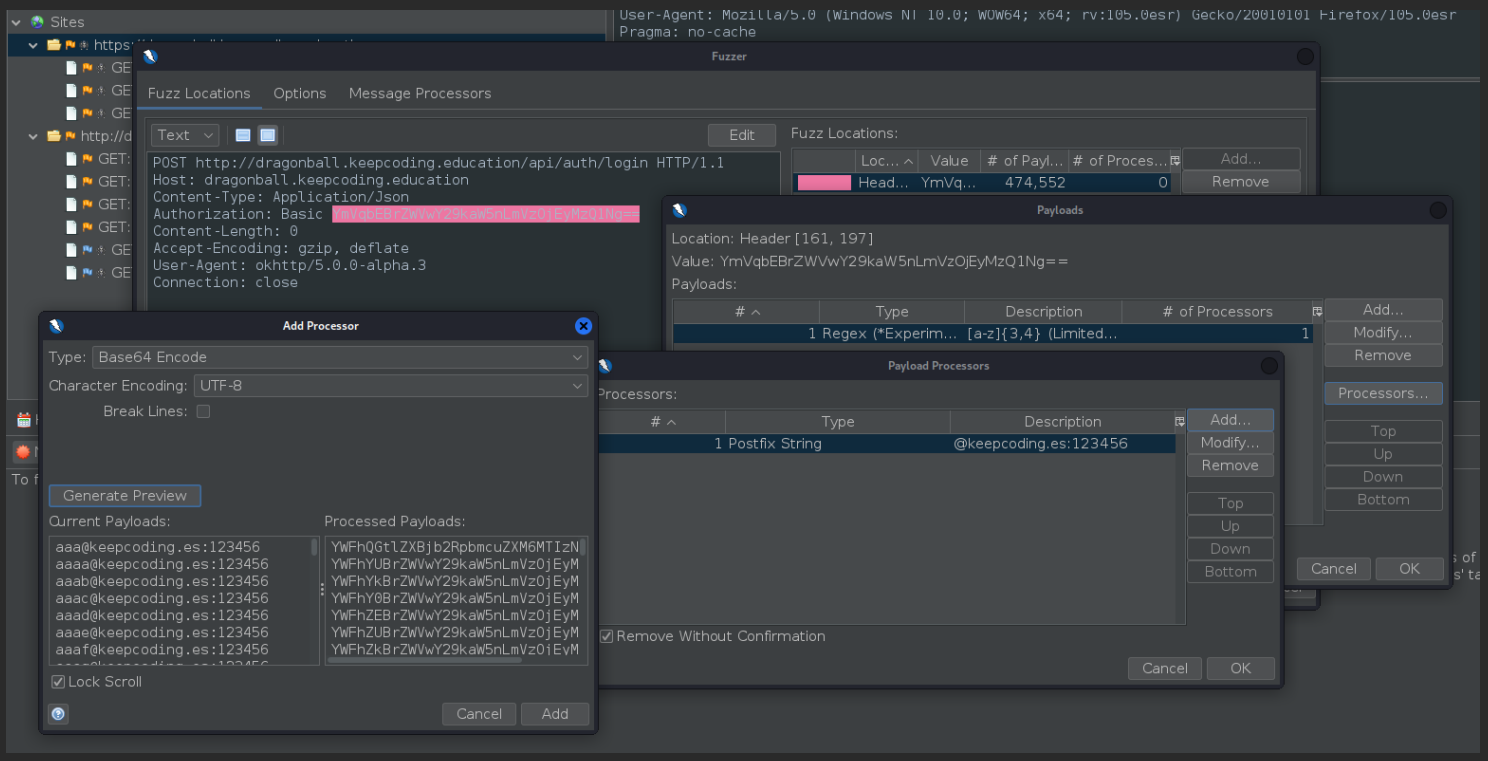
Payload: Bruteforcer 3-4 caracteres abcdefghijlmnopqrstuvz permite reducir las peticiones a 292008.
Añadiendo sufijo @keepcoding.es:1234546 y encodeando en base64.

De todas formas va bastante lento por ser la versión Community de Burp. Lanza unas 400 peticiones por hora pero se va ralentizando a medida que avanza, quizá por el WAF.



Dado que Burp va bastante lento probamos también con el fuzzer de ZAP, el proxy de ataque de OWASP.

Fuzzer - Zed Attack Proxy



Petición fuzzeada:

```
POST https://dragonball.keepcoding.education/api/auth/login HTTP/1.1
Host: dragonball.keepcoding.education
Content-Type: Application/Json
Authorization: Basic <FUZZ>
Content-Length: 0
Accept-Encoding: gzip, deflate
User-Agent: okhttp/5.0.0-alpha.3
Connection: close
```

Payload:
Regex: [a-z]{4} (aproximadamente 550000 combinaciones)
Postfix: @keepcoding.es:123456 (en suposición de que exista otra cuenta de keepcoding con esa contraseña)
Encode: base64

Sin resultados.

Ataques de Inyección

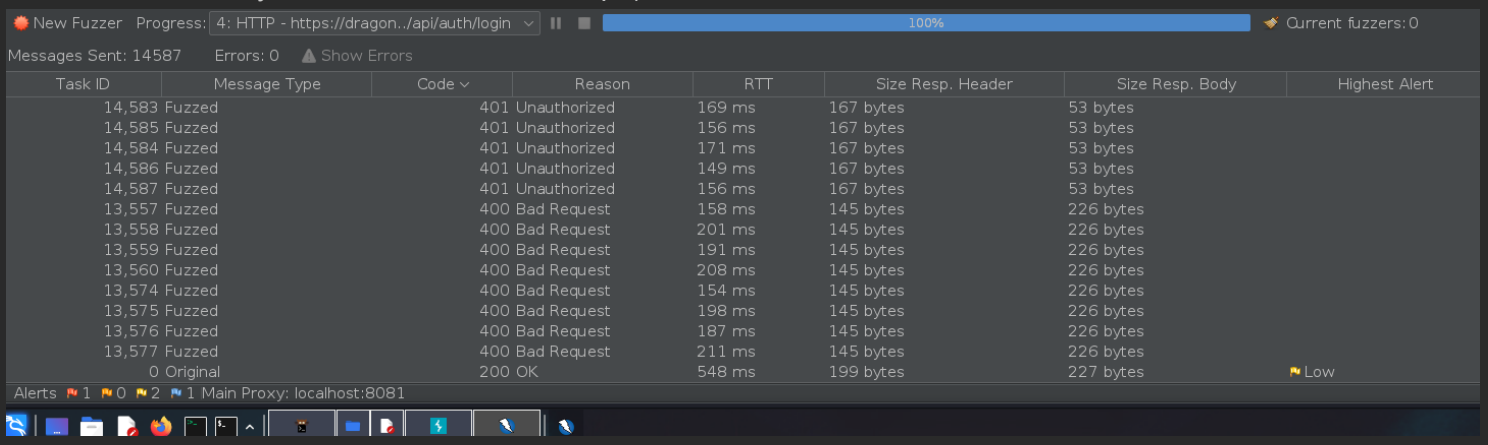
Burp Intruder / Repeater

Se modelan las posibles peticiones con Repeater.
Se intentan fuzzear diversas inyecciones SQL con procesado de encoding base64 y dándole forma de correo, e incluso escapando caracteres especiales, sin éxito. Se utilizan ficheros de diccionario incluídos en la herramientas de Kali:

/usr/share/wordlists/wfuzz/Injections/SQL.txt
/usr/share/wordlists/wfuzz/vulns/sql_inj.txt
/usr/share/wordlists/wfuzz/Injections/All_attack.txt

ZAP Fuzzer / Requester

También se utilizan todos los ficheros de diccionario de ataques incluídos en el Fuzzer de Zed Attack Proxy, unos 14500 ataques diferentes con combinaciones de diferentes procesados (base64 encode, postfix con forma de correo, comillas y otros caracteres de escape).



Los códigos 400 Bad Request proceden de peticiones con payload para buffer overflow, que afectan a la propia petición, al impedir la recepción de las cabeceras posteriores a Authorization.

Vulnerabilidades Encontradas

Almacenamiento de datos inseguro

Vemos en jd-gui que la aplicación guarda credenciales en el archivo NAME.xml. Usando la app en el emulador genymotion, desde el kali nos conectamos a al android con `adb connect 192.168.1.111`. Después sacamos la shell con `adb shell`. Sabemos que todas las aplicaciones están en el directorio /data/data así que para localizar nuestra aplicación usamos ls `/data/data | grep drag` lo que nos da

```
vbox86p:/ # ls /data/data | grep drag
com.isabri.dragonballandroidavanzado
vbox86p:/ #
```

Dentro, haciendo un cat Name.xml encontramos almacenados el usuario y contraseña en base64. Además de un token interesante.

```
vbox86p:/data/data/com.isabri.dragonballandroidavanzado/shared_prefs # ls -l
total 24
-rw-rw---- 1 u0_a130 u0_a130 138 2023-02-22 01:55 MapviewInitializerPreferences.xml
-rw-rw---- 1 u0_a130 u0_a130 430 2023-02-22 01:55 NAME.xml
-rw-rw---- 1 u0_a130 u0_a130 213 2023-02-22 01:55 com.google.maps.api.android.lib6.drd.PREFERENCES_FILE.xml
at NAME.xml
<?xml version='1.0' encoding='utf-8' standalone='yes' ?>
<map>
    <string name="CREDENTIAL">Basic YmVqbEBrZWVwY29kaW5nLmVzOjEyMzQ1Ng==</string>
    <string name="TOKEN">eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCIsImtpZCI6InByaXZhdGUifQ.eyJleHBpcmF0aW9uIjo2NDA5MjIxMTIwMCwiaWRlbnRpZnkiOiI3QUI4QUM0RC1BRDhGLTRBQ0UtQUE0NS0yMUU4NEFFOEJCRTciLCJlbWFpbCI6ImJlamxAa2VlcGNvZGluZy5lcyJ9.18bwBNb2ShveuqbGUm2AmYYdRpyJlqS46Uw-K54cFfA</string>
</map>
vbox86p:/data/data/com.isabri.dragonballandroidavanzado/shared_prefs # ^C
130|vbox86p:/data/data/com.isabri.dragonballandroidavanzado/shared_prefs #
```

Mitigación

Esta información idealmente debería de ir cifrada.

Insecure Logging

Con `adb shell ps | grep dragon` vemos el id de la aplicación:

```
┌──(kali㉿kali)-[~]
└─$ adb shell ps | grep dragon
u0_a130       3271   405 12793952 158204 ep_poll   73a9b0a3c90a S com.isabri.dragonballandroidavanzado

┌──(kali㉿kali)-[~]
└─$ ss
```

Con `adb logcat | grep 3271` mientras hacemos login en nuestro emulador vemos que hace un post en la pagina https://dragonball.keepcoding.education/api/auth/login. Además subrayado lleva el usuario y la contraseña en base64:

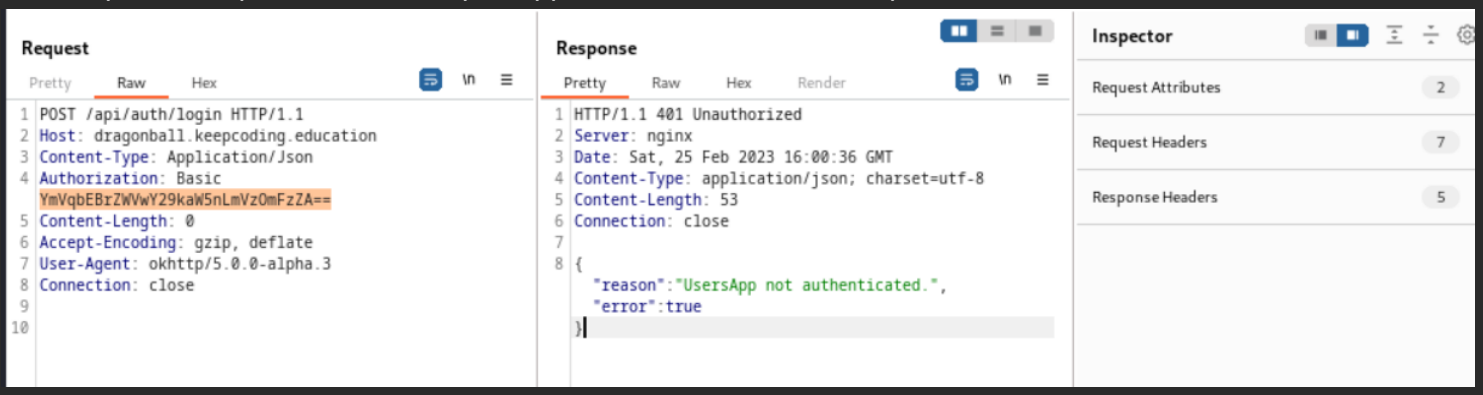
```
└─$ adb logcat | grep 3271
02-22 01:30:16.453   670   700 I ActivityManager: Start proc 3271:com.isabri.dragonballandroidavanzado/u0a130 for pre-top-activity {com.isabri.dragonballandroidavanzado/com.isabri.dragonballandroidavanzado.ui.login.LoginActivity}
02-22 01:39:39.124  3271  3542 I okhttp.OkHttpClient: ⟶ POST https://dragonball.keepcoding.education/api/auth/login
02-22 01:39:39.124  3271  3542 I okhttp.OkHttpClient: Content-Length: 0
02-22 01:39:39.124  3271  3542 I okhttp.OkHttpClient: Content-Type: Application/Json
02-22 01:39:39.124  3271  3542 I okhttp.OkHttpClient: Authorization: Basic YmVqbEBrZWVwY29kaW5nLmVzOjIxMzE4MjA2MTc=
02-22 01:39:39.124  3271  3542 I okhttp.OkHttpClient: ⟶ END POST (0-byte body)
02-22 01:39:40.572  3271  3542 I okhttp.OkHttpClient: ⟵ HTTP FAILED: java.net.ProtocolException: Too many follow-up requests: 21
02-22 01:39:40.573  3271  3271 D MyLog   : Error while retrieving the token
```

Mitigación

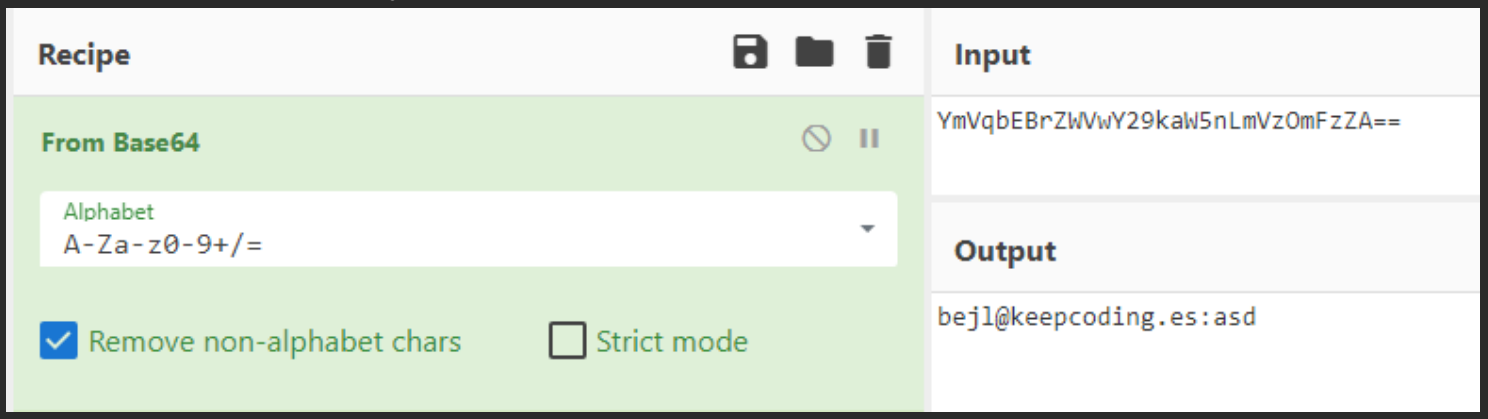
Esta información no debería de ir en texto plano, debería estar cifrada.

Fuerza Bruta con Burp Suite

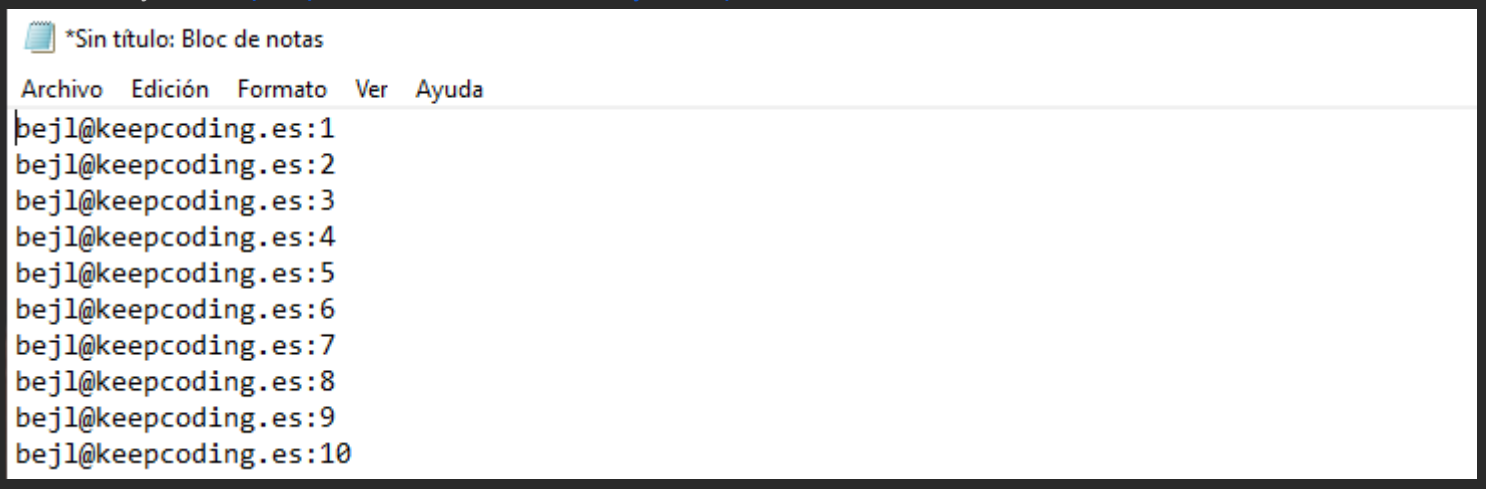
Vemos que en las peticiones de burp la app manda un base64 a la api.



En este base64 va el usuario y la contraseña que metemos en la aplicación.



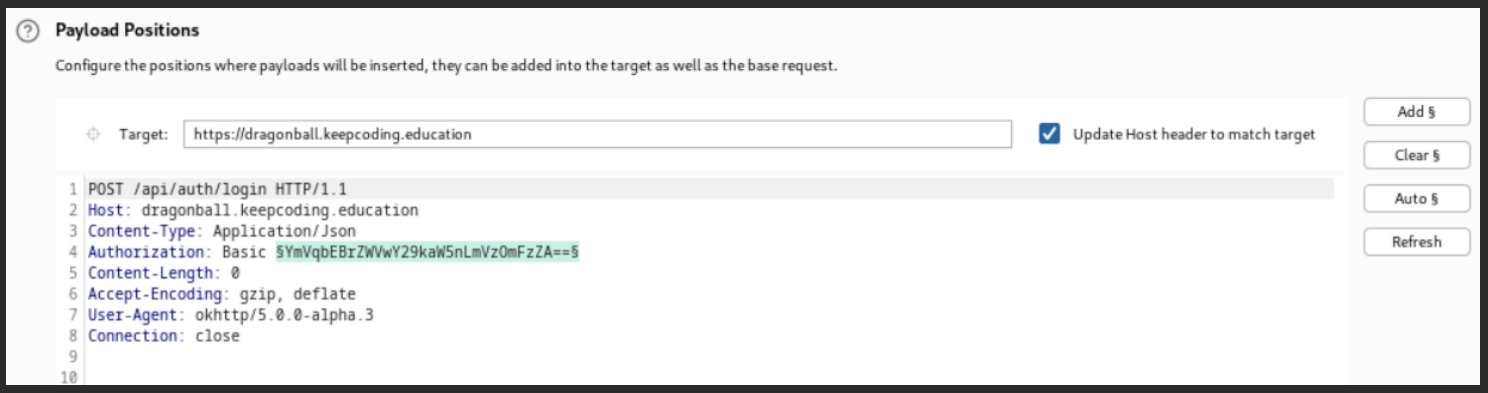
Con excel y con https://pinetools.com/es/buscar-y-reemplazar creamos una lista:



Con https://www.base64encode.org/ convertimos linea a linea a base 64 en vez de el archivo entero:

```
YmVqbEBrZWVwY29kaW5nLmVzOjE=
YmVqbEBrZWVwY29kaW5nLmVzOjI=
YmVqbEBrZWVwY29kaW5nLmVzOjM=
YmVqbEBrZWVwY29kaW5nLmVzOjQ=
YmVqbEBrZWVwY29kaW5nLmVzOjU=
YmVqbEBrZWVwY29kaW5nLmVzOjY=
YmVqbEBrZWVwY29kaW5nLmVzOjc=
YmVqbEBrZWVwY29kaW5nLmVzOjg=
YmVqbEBrZWVwY29kaW5nLmVzOjk=
```

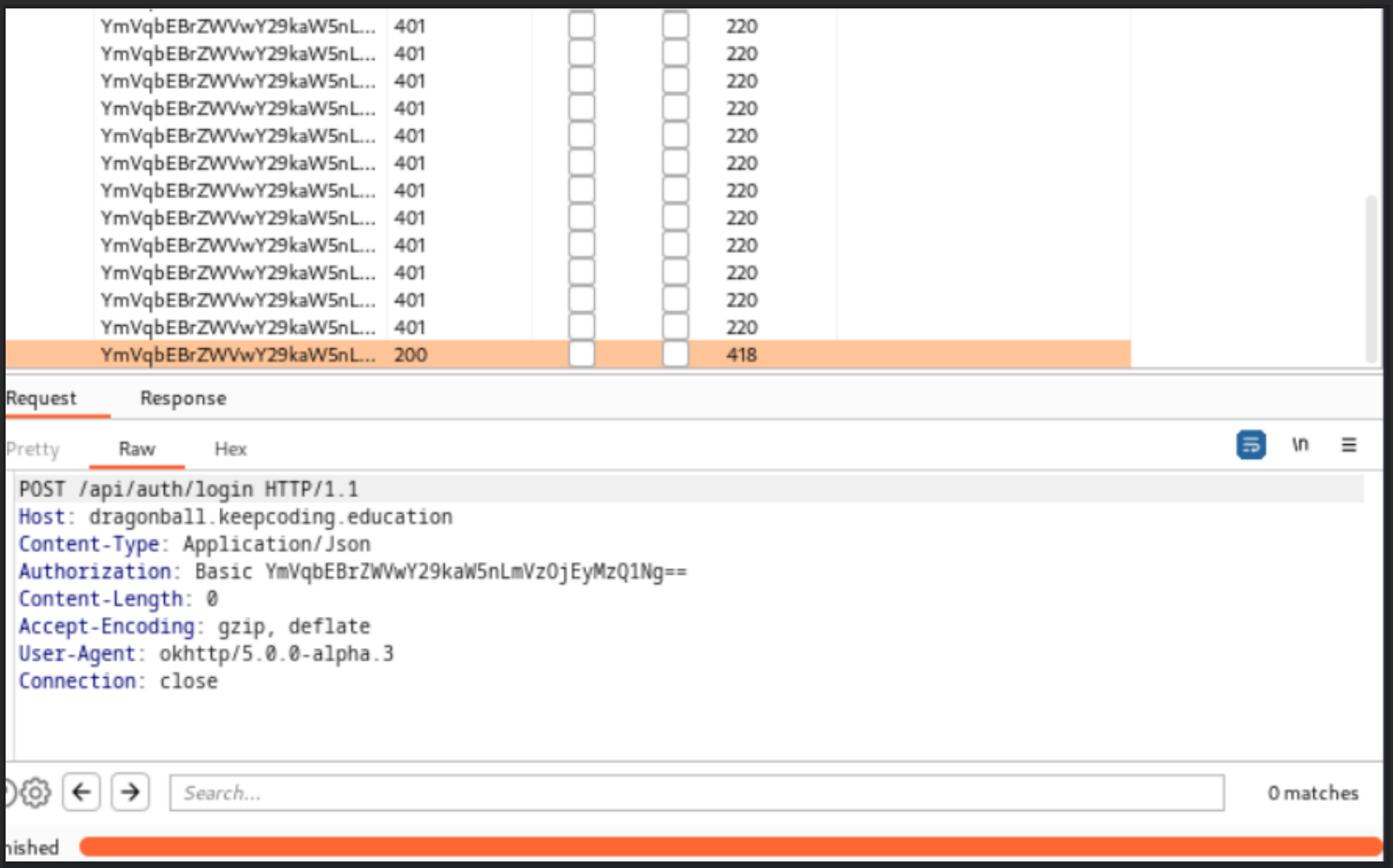
La petición del login la mandamos a intruder desde burpsuite, seleccionamos el base64 y le añadimos add



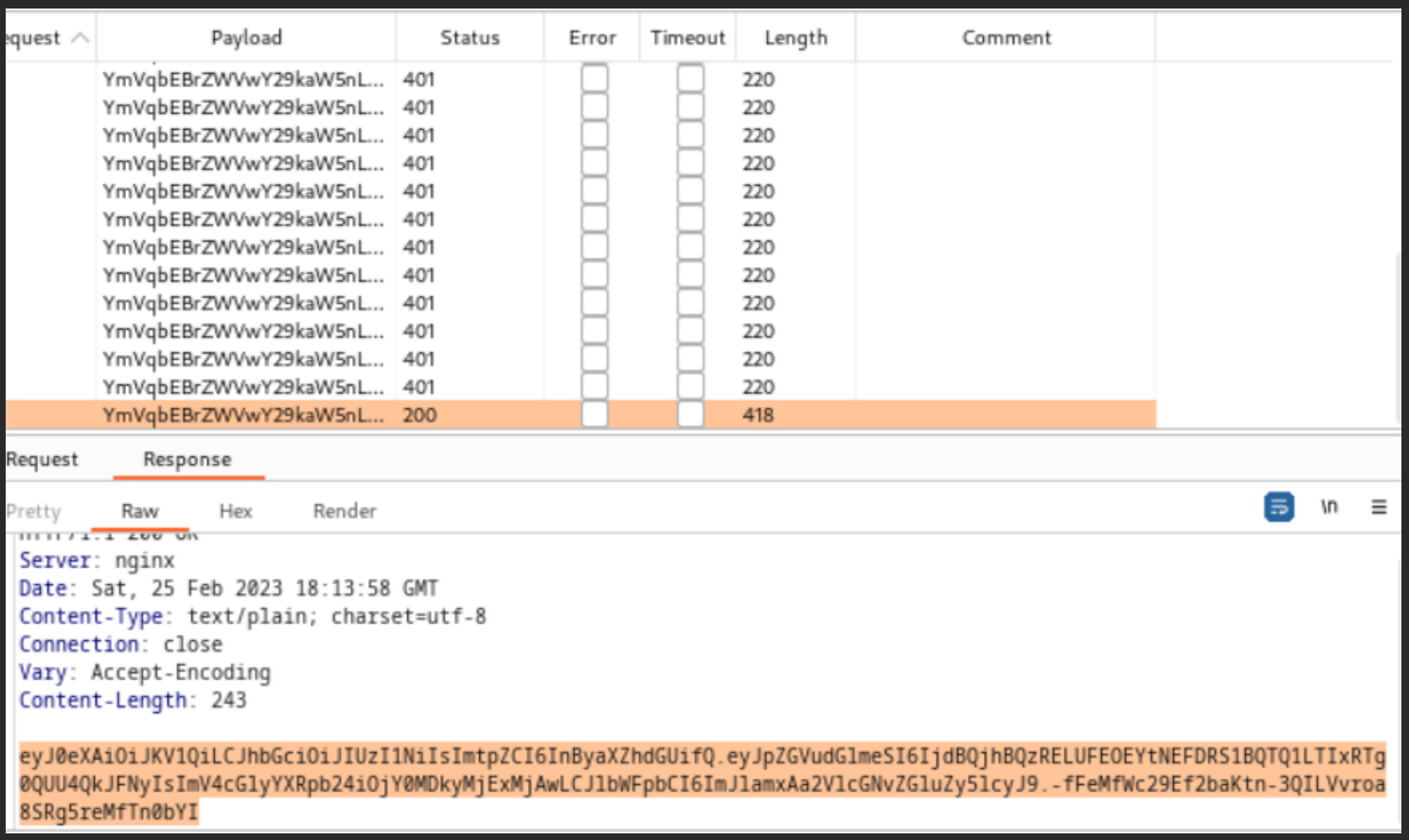
En payloads cargamos los códigos base64

En un ejercicio en el que solo tuviésemos el usuario y no la contraseña se emplearía un diccionario adaptado a la aplicación.

Una vez encontrada la contraseña nos da un `HTTP 200 OK`:



Y nos devuelve un token como respuesta:



Decodificado seria : {
"identify": "7AB8AC4D-AD8F-4ACE-AA45-21E84AE8BBE7",

```
    "expiration": 64092211200,
    "email": "bejl@keepcoding.es"
}
```

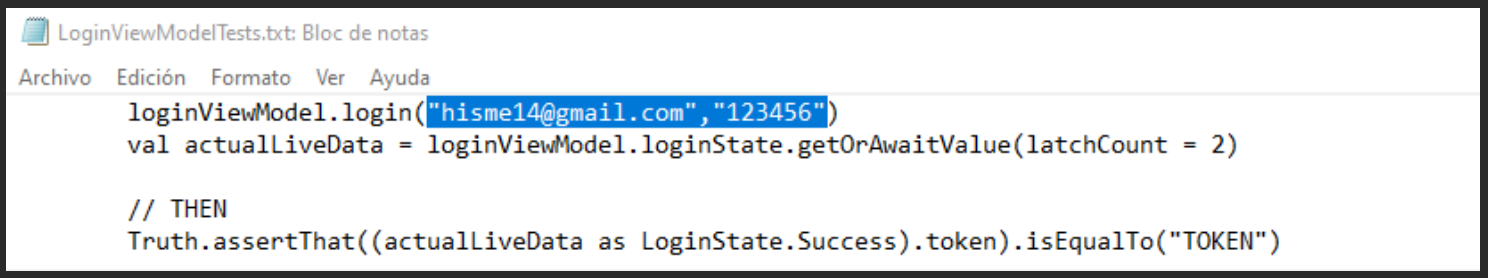

Queda probada la vulnerabilidad por fuerza bruta. En este caso la aplicación no debería de permitirte probar usuarios y contraseñas de manera infinita.

Mitigación

Lo ideal es que después de varios intentos tengas que esperar periodos de tiempo, incrementados por cada intento de login fallido.

Credenciales en texto plano en el código fuente.

Una vez visto con logcat que el nombre de la aplicación es com.isabri.dragonbalandroidavanzado, mediante osint encontramos el github del propietario de la aplicación y el codigo de esta: En la dirección https://github.com/tecosabri/DragonBallAndroidAvanzado en DragonBallAndroidAvanzado-main\app\src\test\java\com\isabri\dragonballandroidavanzado\ui\login encontramos "hisme14@gmail.com","123456"

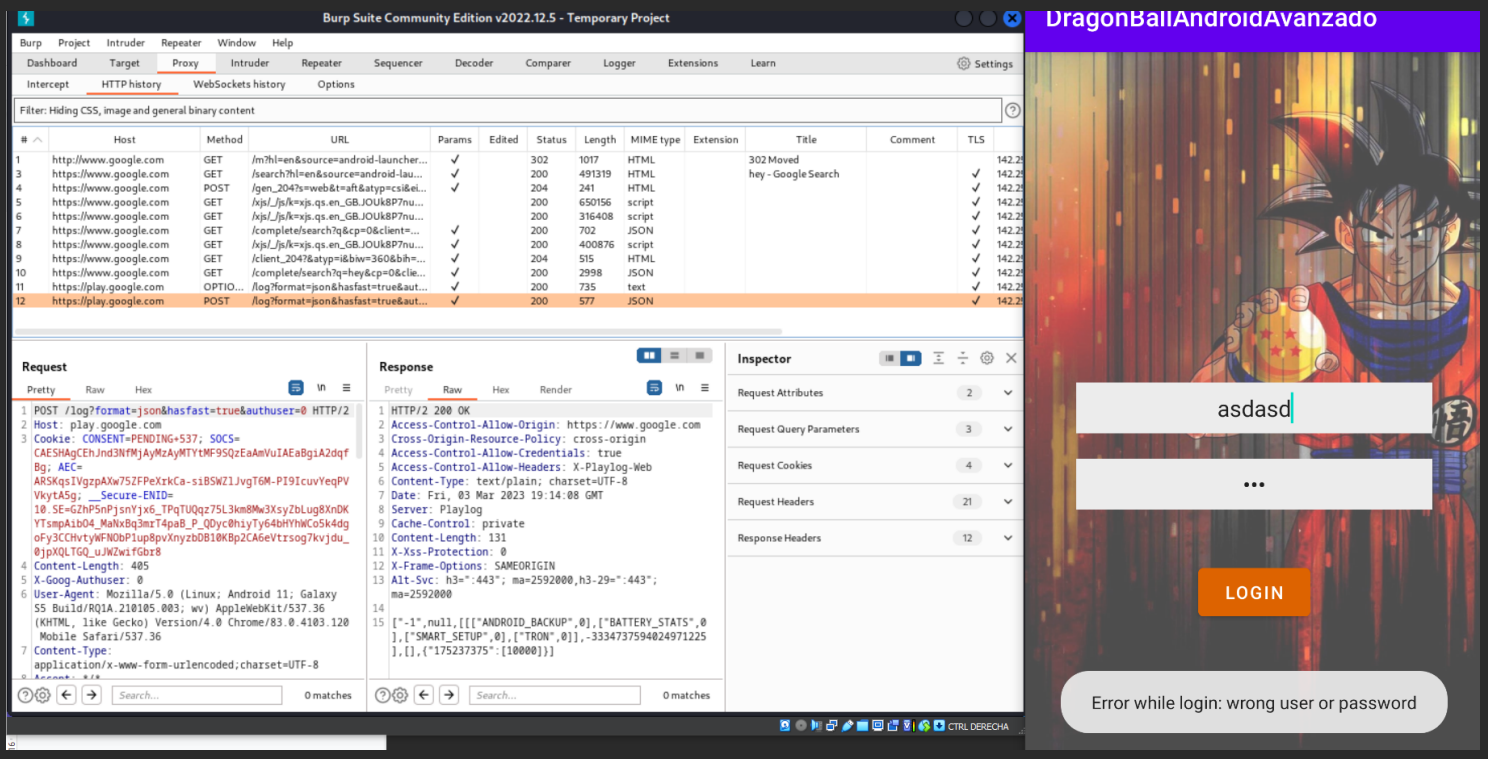


Mitigación

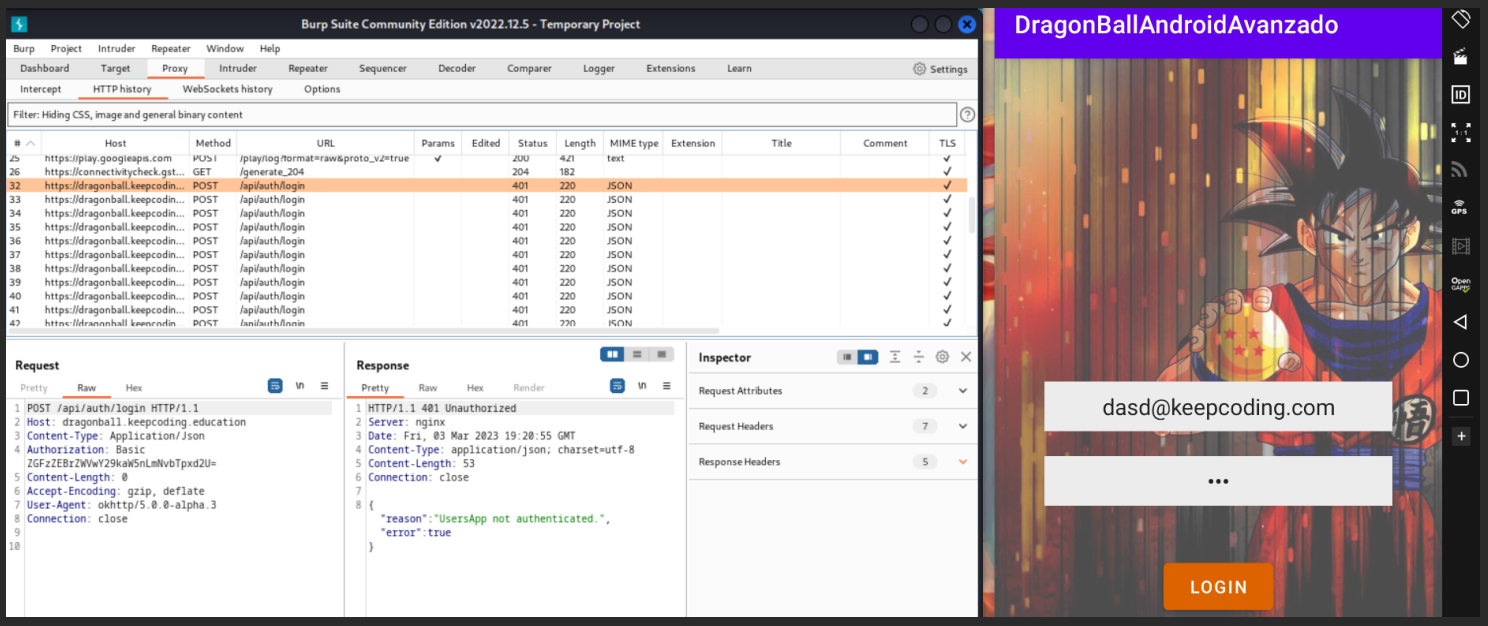
Idealmente desde una perspectiva de ciberseguridad no deberían de estar las credenciales en el código. Entendemos que es una aplicación que no está en producción por lo que es más normal.

Insecure Login 2.1; Perspectiva Token y Credential.

Con el fin de entender el funcionamiento de la app, en vez de poner exclusivamente la vulnerabilidad en un pantallazo prefiero poner tres ejemplos de causa efecto entendiendo causa lo que ponemos en el login y efecto en su respuesta.
De entrada si pones un usuario y contraseña erróneo responde con “invalid user” + “error while login: wrong user or password”. Esto lo hace sin mandar ningún paquete a la api. Esto es porque la aplicación verifica primero contra sí misma el formato del usuario.

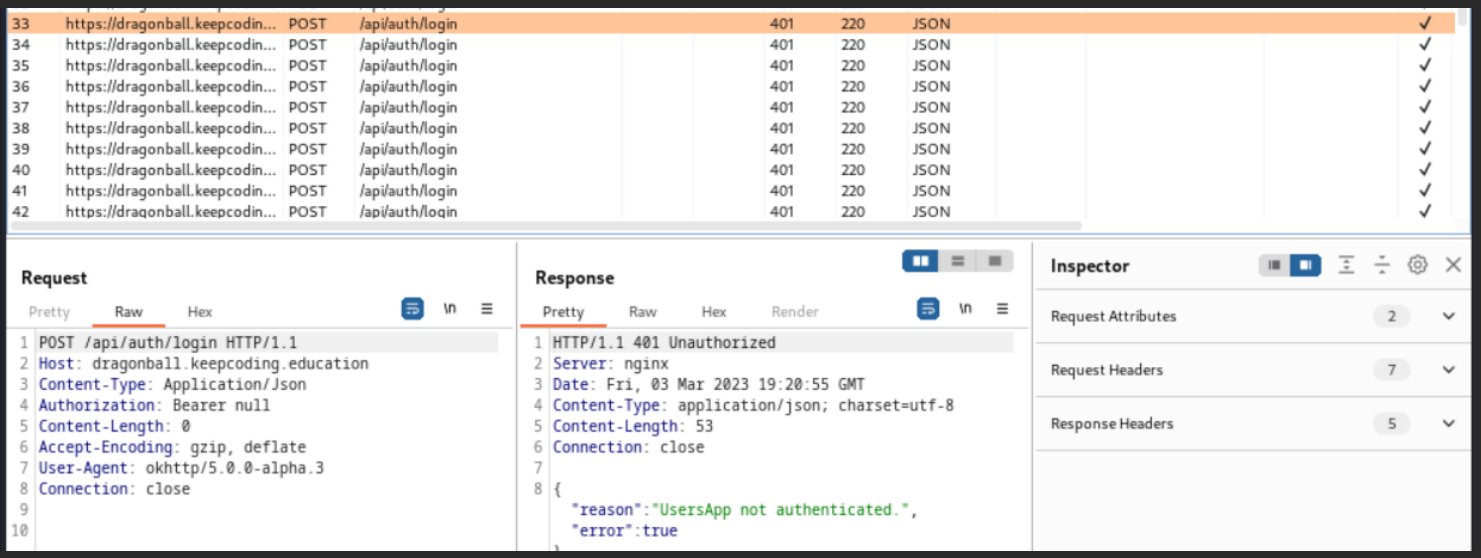


Si por el contrario ponernos como usuario <texto>@keepcoding.com, al verificar contra sí misma como un formato de usuario válido, no muestra “invalid user”,manda una primera petición a la api con el usuario y contraseña en base64, muestra “error while login: wrong user or password”.



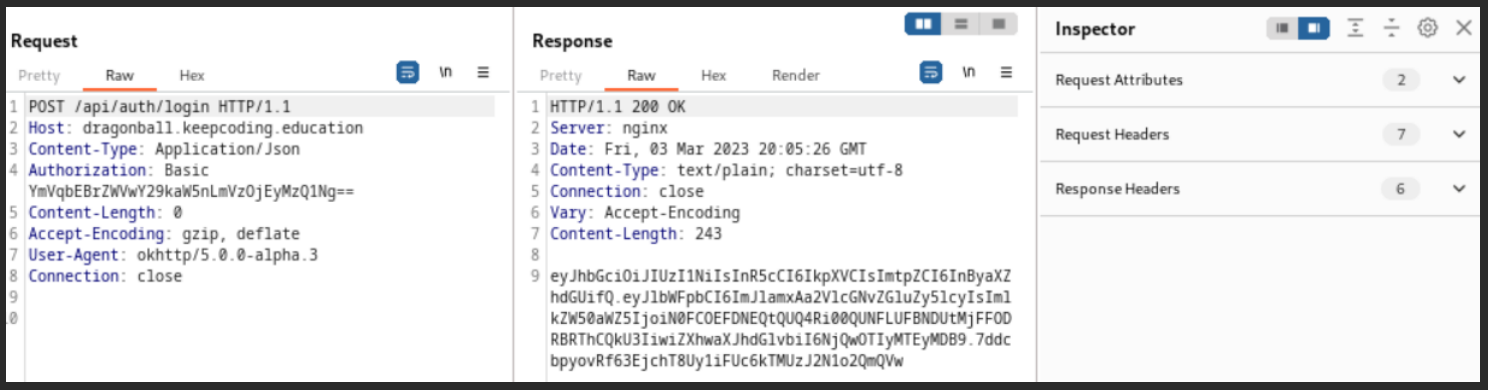
El criterio para aceptar el usuario antes de mandarlo a la api es “<texto>@<texto>.<texto>”. Las variables de texto da igual de que se compongan.

Además manda una secuencia de peticiones “vacías” sin el base64 cambiando el tipo de autorización a “Bearer null” en vez de basic:

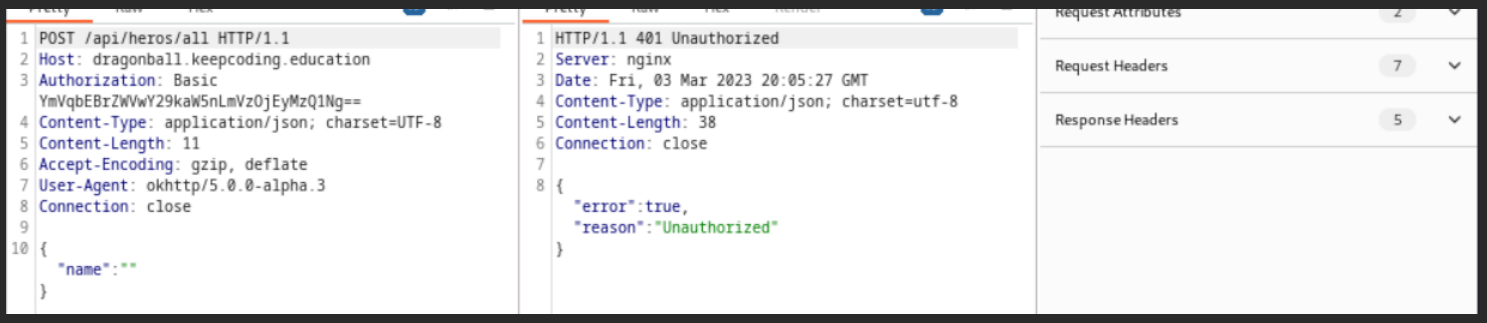


(De la 33 a la 42 es la misma petición)

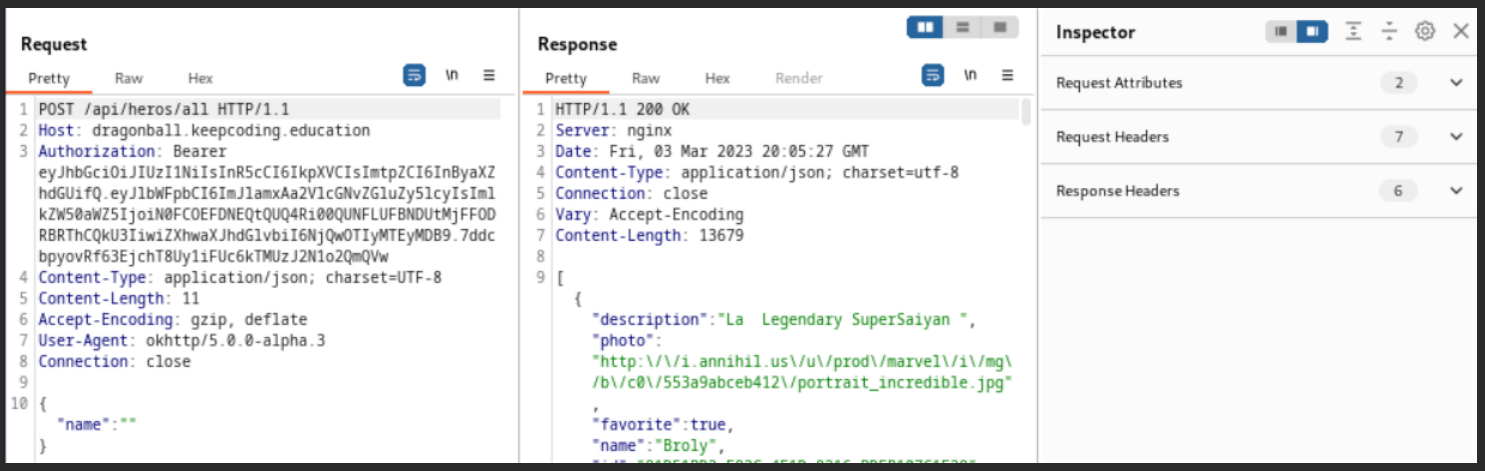
Seguidamente tenemos una petición de un usuario y contraseñas correcta: bejl@keepcoding.es:123456
Hacemos petición y nos devuelve token:



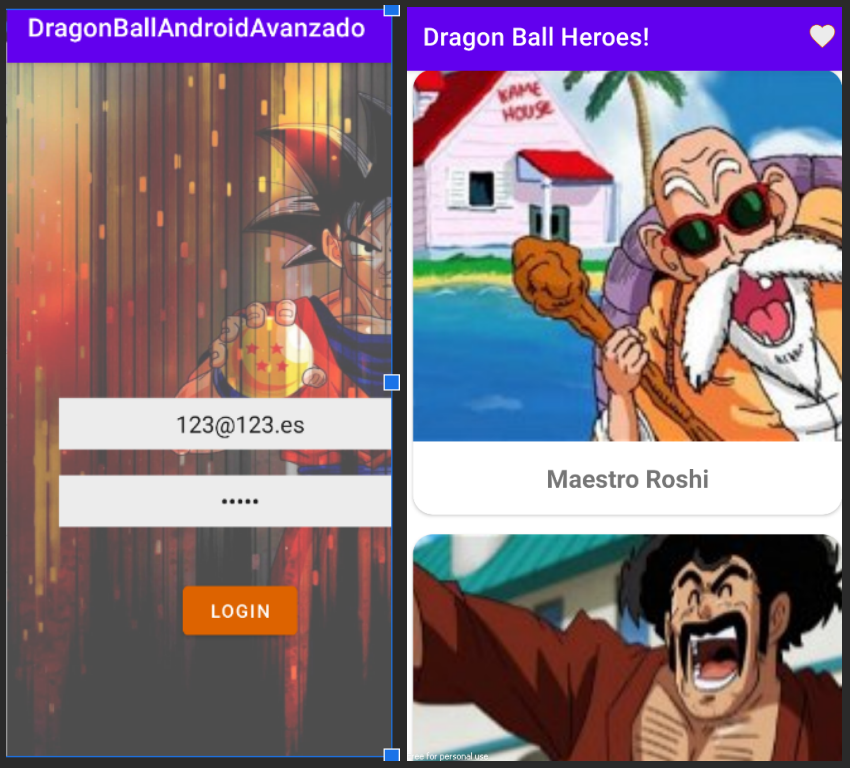
Hace otra petición para ver los héroes con el base64:



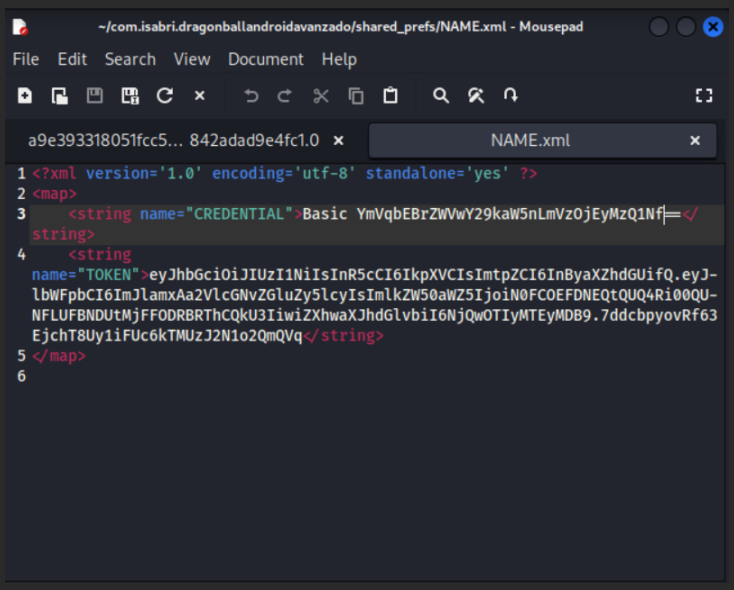
Recibe un código 401 y hace la misma petición con el token; esta vez entra en api/heros/all y recibe todos los datos de los héroes. De modo guarda ambas cosas en el dispositivo, tanto el token como las fotografías y datos de los héroes.



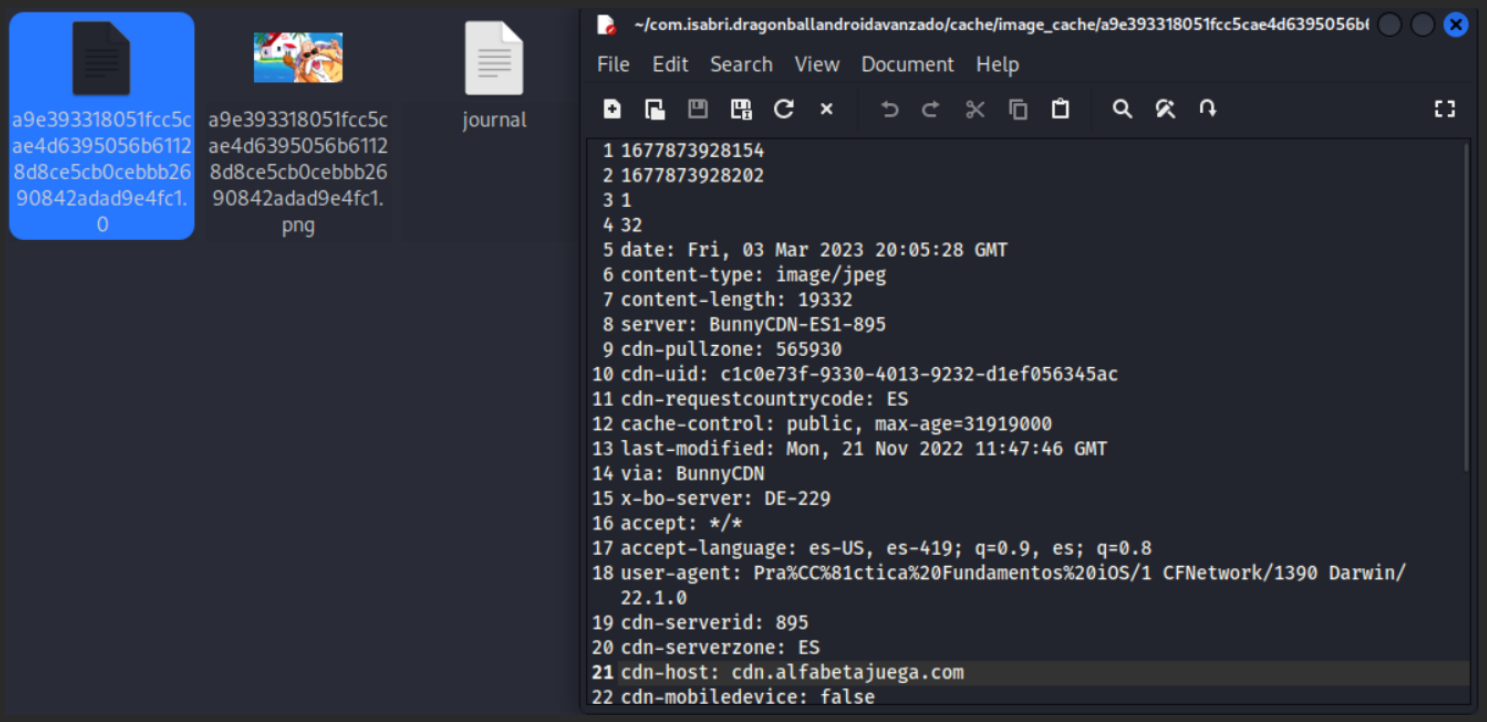
Aquí viene la vulnerabilidad. Una vez tienes el token da igual que contraseña pongas que te deja entrar si o si. La app tiene todos los datos de los héroes, y el token. Y aunque pongas usuario y contraseña errónea y desactives el adaptador de red, permite acceder igualmente, porque no mira el user y la pass y no manda petición a la api.



Esta información la guarda (repito, esto es una vez ya nos hemos logueado correctamente y se ha guardado el token y el base64 y después hemos salido de la aplicación) en /home/kali/com.isabri.dragonballandroidavanzado/shared_prefs/NAME.xml



Mientras que la información de las fotos esta en /cache/image_cache/



Mitigación

Idealmente, la app siempre debería de verificar el usuario y la contraseña contra la api independientemente de si te has logueado antes o no.

Insecure Login 2.2

Además de que no es necesario poner el usuario contraseña correcta para entrar, tampoco hace falta poner el token o la credencial correcta. Las podemos modificar a nuestra voluntad. Con tal de que exista el archivo NAME.xml con datos de token y credencial, entra en la aplicación.

Esto es porque una vez que hemos entrado y salido, ya está descargada la información de /api/heros/all. Y no vuelve a mandar petición a dicha api para verificar el token y las credenciales.

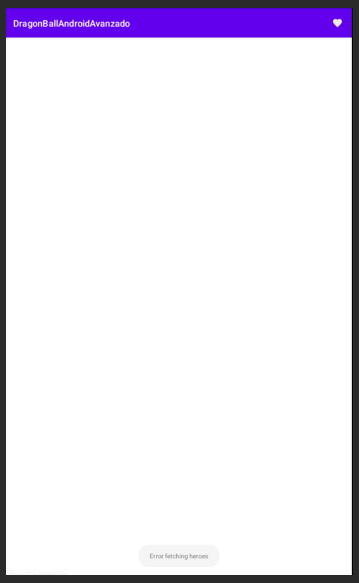
Mitigación:

Realizar verificaciones de entrada en el lado cliente y servidor.

Insecure Login 2.3 - Bypass login

```
adb shell am start -n com.isabri.dragonballandroidavanzado/com.isabri.dragonballandroidavanzado.ui.HeroesListActivity
```

Por último con el comando anterior comprobamos que podemos lanzar directamente la pantalla de la app. En caso de habernos autenticado en alguna ocasión previa en la app se nos abre nuevamente sin solicitar autenticación, pero en caso de no habernos autenticado nunca vemos que devuelve una pantalla en blanco, ya que la app solo nos sirve una información obtenida de una página a la que le pasa nuestro token, como ese token no existe la página que nos devuelve la app se queda en blanco.



Mitigación

La aplicación debería de mandar la petición con el token a la api para verificarlo antes de entrar.

Improper data management:

Como hemos visto en el apartado de tráfico usando el comando:
`dirsearch -u https://dragonball.keepcoding.education/`
Encontramos varios archivos a los que no deberíamos tener acceso:
https://dragonball.keepcoding.education/.git/info/exclude
https://dragonball.keepcoding.education/.git/config
https://dragonball.keepcoding.education/.git/index
y un hash en:
https://dragonball.keepcoding.education/.git/refs/heads/main

Mitigación

En la configuración de la api debería de haber la opción de dejar estos directorios como “forbbiden”