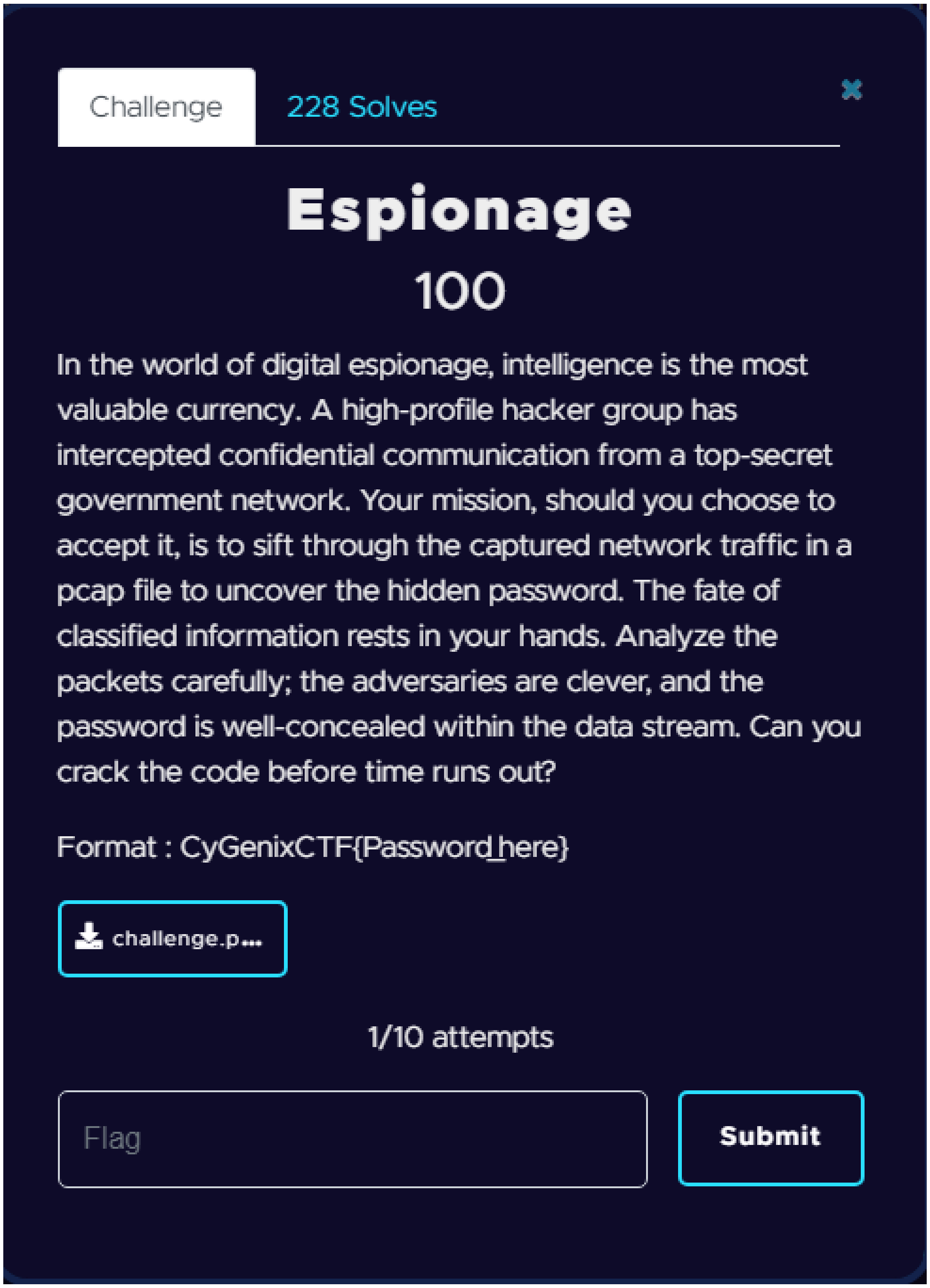
Challenge Description:



Espionage

100

In the world of digital espionage, intelligence is the most valuable currency. A high-profile hacker group has intercepted confidential communication from a top-secret government network. Your mission, should you choose to accept it, is to sift through the captured network traffic in a pcap file to uncover the hidden password. The fate of classified information rests in your hands. Analyze the packets carefully; the adversaries are clever, and the password is well-concealed within the data stream. Can you crack the code before time runs out?

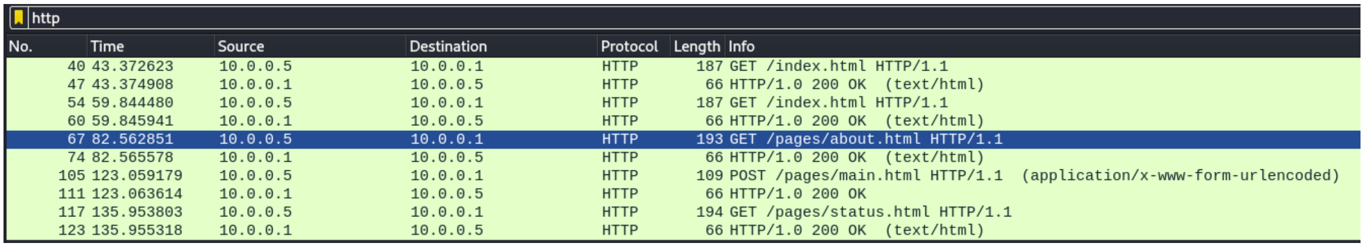
Format : CyGenixCTF{Password_here}

1/10 attempts

This one was quite simple we start with a pcap file so guess its time open up wireshark.

After some looking around I found some http request those are always nice so lets take a look. We can set a filter in wirehark to only display http request.

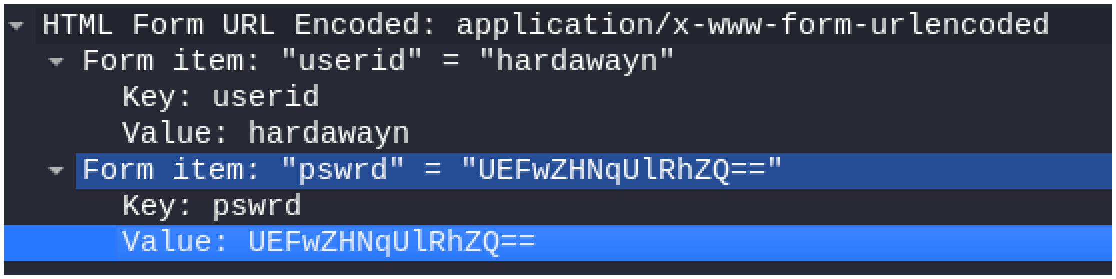
This is the result of applying the filter



http

No.	Time	Source	Destination	Protocol	Length	Info
40	43.372623	10.0.0.5	10.0.0.1	HTTP	187	GET /index.html HTTP/1.1
47	43.374908	10.0.0.1	10.0.0.5	HTTP	66	HTTP/1.0 200 OK (text/html)
54	59.844480	10.0.0.5	10.0.0.1	HTTP	187	GET /index.html HTTP/1.1
60	59.845941	10.0.0.1	10.0.0.5	HTTP	66	HTTP/1.0 200 OK (text/html)
67	82.562851	10.0.0.5	10.0.0.1	HTTP	193	GET /pages/about.html HTTP/1.1
74	82.565578	10.0.0.1	10.0.0.5	HTTP	66	HTTP/1.0 200 OK (text/html)
105	123.059179	10.0.0.5	10.0.0.1	HTTP	109	POST /pages/main.html HTTP/1.1 (application/x-www-form-urlencoded)
111	123.063614	10.0.0.1	10.0.0.5	HTTP	66	HTTP/1.0 200 OK
117	135.953803	10.0.0.5	10.0.0.1	HTTP	194	GET /pages/status.html HTTP/1.1
123	135.955318	10.0.0.1	10.0.0.5	HTTP	66	HTTP/1.0 200 OK (text/html)

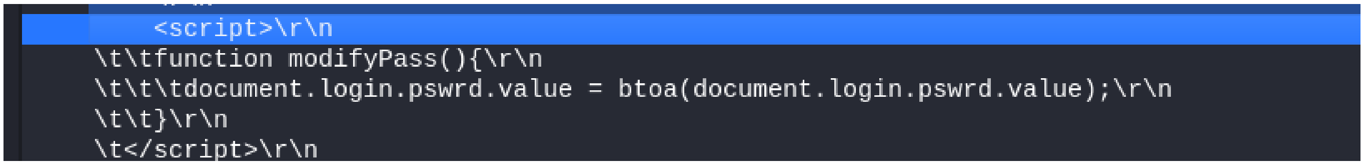
After some looking around in each of the captures I found a username and password inside a post request sent to /pages/main.html



```
▾ HTML Form URL Encoded: application/x-www-form-urlencoded
  ▾ Form item: "userid" = "hardawayn"
      Key: userid
      Value: hardawayn
  ▾ Form item: "pswrd" = "UEFwZHNqUlRhZQ=="
      Key: pswrd
      Value: UEFwZHNqUlRhZQ==
```

The password looks like it might be encoded in some sort of format lets look through the other request to see if we can find the function that is used to encode the password.

After some looking I found in the first get response to /index.html there is a javascript function called modify pass.

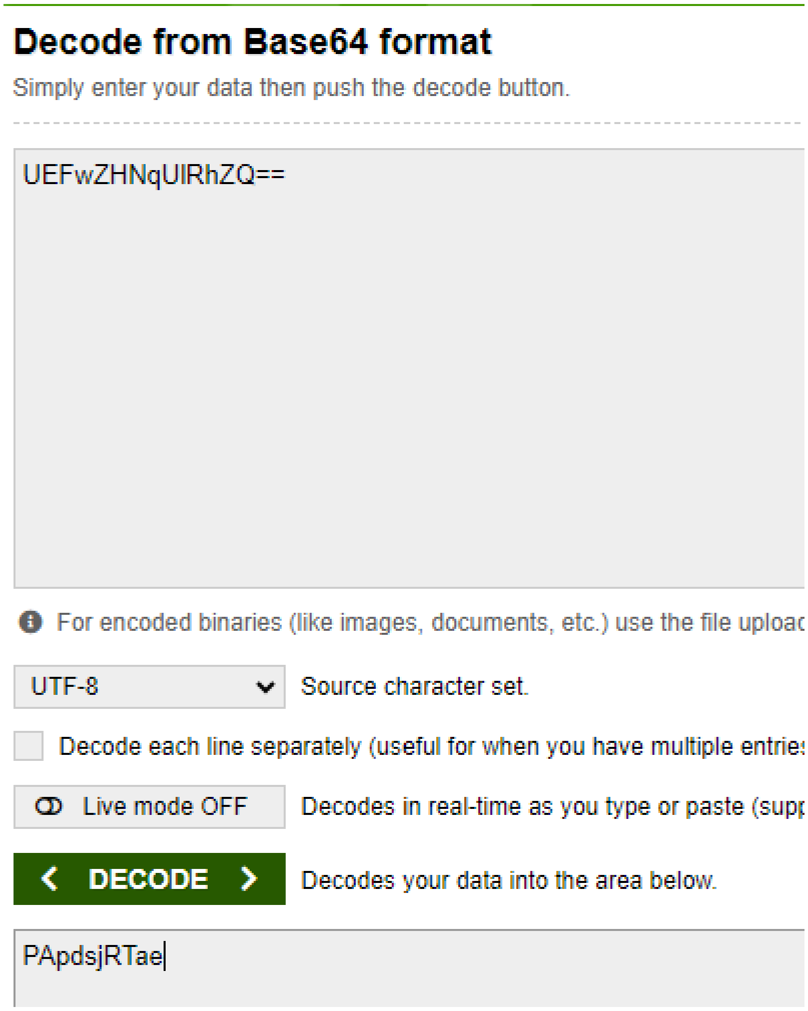


```
    <script>\r\n
\t\tfunction modifyPass(){\r\n
\t\t\tdocument.login.pswrd.value = btoa(document.login.pswrd.value);\r\n
\t\t}\r\n
\t</script>\r\n
```

Now we can be sure the password is encoded in some sort of format. Lets lookup what btoa does and if we can reverse it.

Found this link that describes it btoa()
It just uses base64 to encode the passwd

base64 can be reversed lets find a website that can do it for us.



Decode from Base64 format

Simply enter your data then push the decode button.

```
UEFwZHNqUlRhZQ==
```

For encoded binaries (like images, documents, etc.) use the file uploac

UTF-8 Source character set.

□ Decode each line separately (useful for when you have multiple entrie

Decodes in real-time as you type or paste (supp

Decodes your data into the area below.

```
PApdsjRTae
```

There we go we got the flag.

CyGenixCTF{PApdsjRTae}