Penetration testing aplikacije "SecureIT"

Dušan Lazić SW 04/2019
Milan Ajder SW 31/2019
Anđela Mišković SW 33/2019

Za potrebe pentestinga projekta "SecureIT" iz predmeta "Bezbednost u sistemima elektronskog poslovanja" primenjene su određeni alati i tehnike kako bi se identifikovali eventualni bezbednosni propusti. U nastavku su prikazani rezultati analize i testiranja.

Gobuster — skeniranje sadržaja

Pomoću **gobuster** alata (alternativa dirbusteru pisan u golang-u) skenirali smo putanje nad dve aplikacije:

- frontend aplikaciju — localhost:4200
- API — localhost:8001

Frontend (Angular)

```
$ gobuster -w /usr/share/dirb/wordlists/big.txt -u http://localhost:4200/`
```

```
===============================================================
Gobuster v2.0.1              OJ Reeves (@TheColonial)
===============================================================
[+] Mode         : dir
[+] Url/Domain   : http://localhost:4200/
[+] Threads      : 10
[+] Wordlist     : /usr/share/dirb/wordlists/big.txt
[+] Status codes : 200,204,301,302,307,403
[+] Timeout      : 10s
===============================================================
2023/06/18 22:28:12 Starting gobuster
===============================================================
/favicon.ico (Status: 200)
/main (Status: 200)
/runtime (Status: 200)
/styles (Status: 200)
/vendor (Status: 200)
===============================================================
2023/06/18 22:28:24 Finished
===============================================================
```

Nakon posećivanja pronađenih stranica, utvrđeno je da svaka od njih vraća frontend aplikaciju koja renderuje 404 stranicu, iako je status 200. To je očekivano s obzirom da se radi o frontend aplikaciji. Nije pronađen ni jedan osetljiv podatak ili resurs na navedenim putanjama.

API (Spring Boot)

```
$ gobuster -w /usr/share/dirb/wordlists/big.txt -u http://localhost:8001/
```

```
===============================================================
Gobuster v2.0.1              OJ Reeves (@TheColonial)
===============================================================
[+] Mode         : dir
[+] Url/Domain   : http://localhost:8001/
[+] Threads      : 10
[+] Wordlist     : /usr/share/dirb/wordlists/big.txt
[+] Status codes : 200,204,301,302,307,403
[+] Timeout      : 10s
===============================================================
2023/06/18 22:32:36 Starting gobuster
===============================================================
/health (Status: 200)
/logout (Status: 302)
===============================================================
2023/06/18 22:32:48 Finished
===============================================================
```

Nakon analize API-ja, identifikovana su samo dva dostupna endpointa:

- `/health` (Status: 200): Ovaj endpoint služi samo za proveru da li server radi. Vraća JSON odgovor sa statusom 200 i porukom "It works."
- `/logout` (Status: 302): Ovaj endpoint se koristi za odjavljivanje korisnika. Prilikom poziva, cookie (access token) se uklanja.

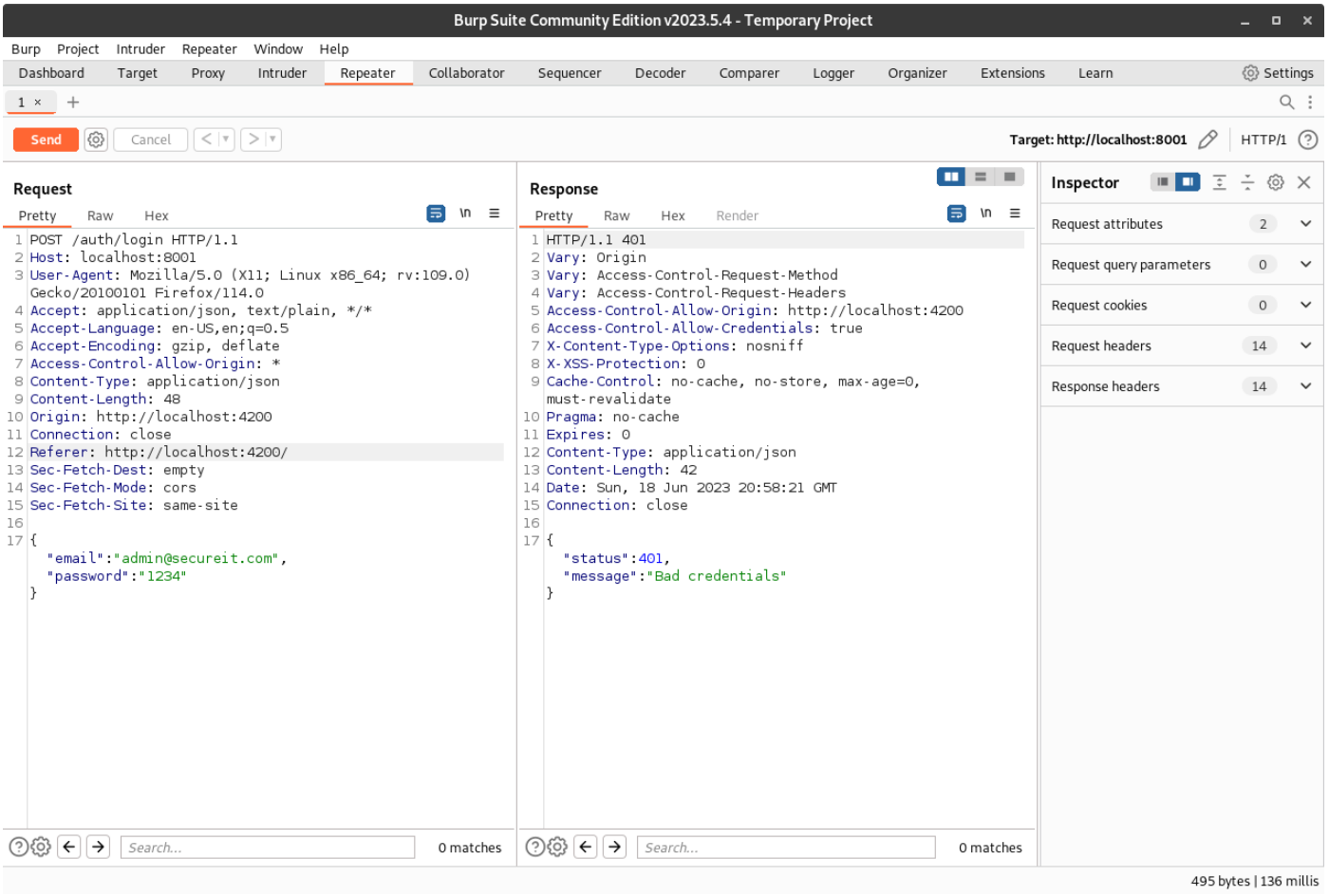
Na osnovu sprovednog skeniranja sadržaja, nije pronađen ni jedan bezbednosni propust niti bilo koji od endpointa pruža mogućnost za dalju istragu.

BurpSuite — Brute-force login strane

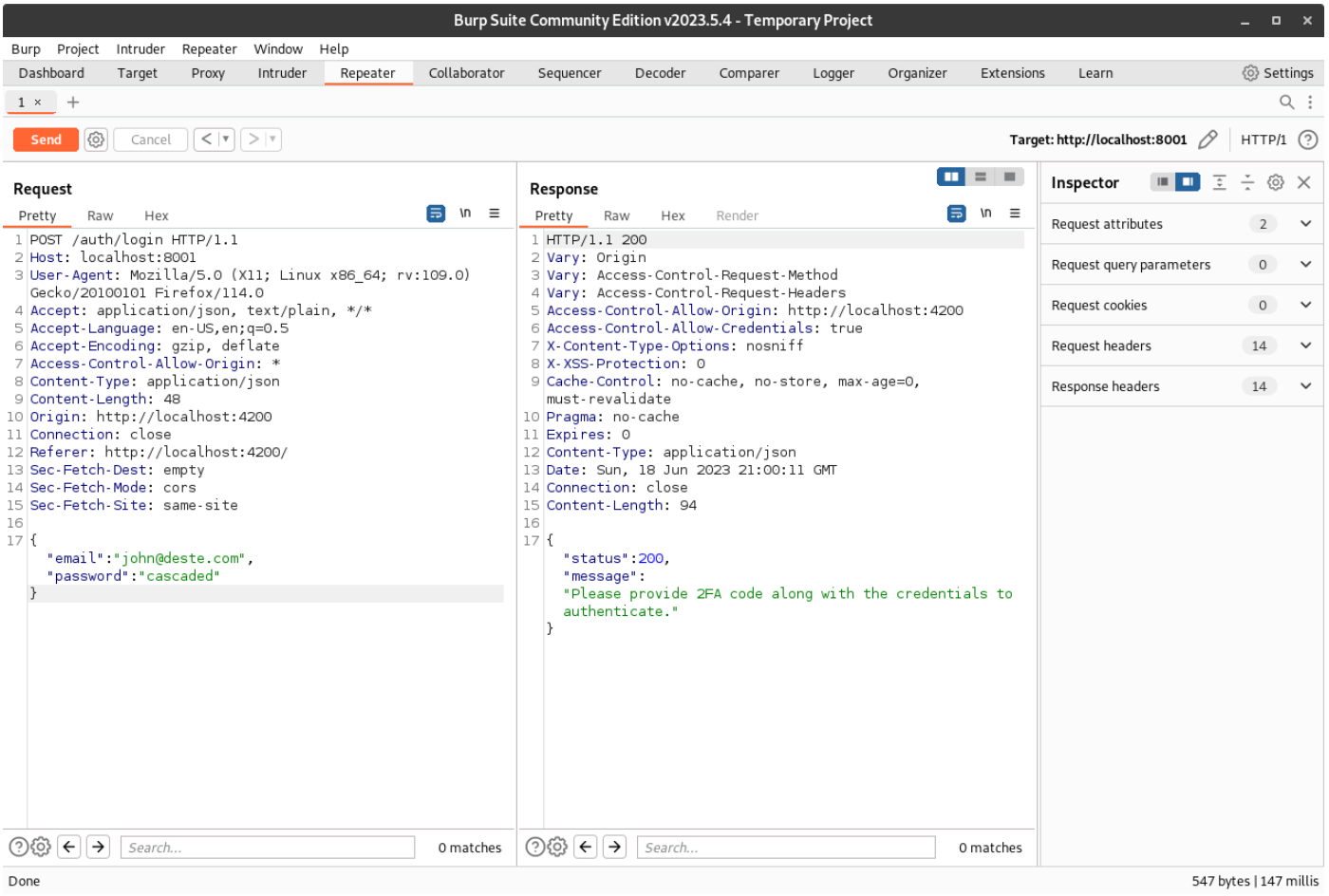
Korišćen je alat Burp Suite kako bi se izvršio brute force napad na login stranu aplikacije. Konfigurisali smo FoxyProxy ekstenziju u Firefoxu kako bi Burp Suite mogao da funkcioniše kao proxy za web browser.

Repeater — analiza odgovora

Najpre smo presreli zahtev za prijavljivanje na sistem.

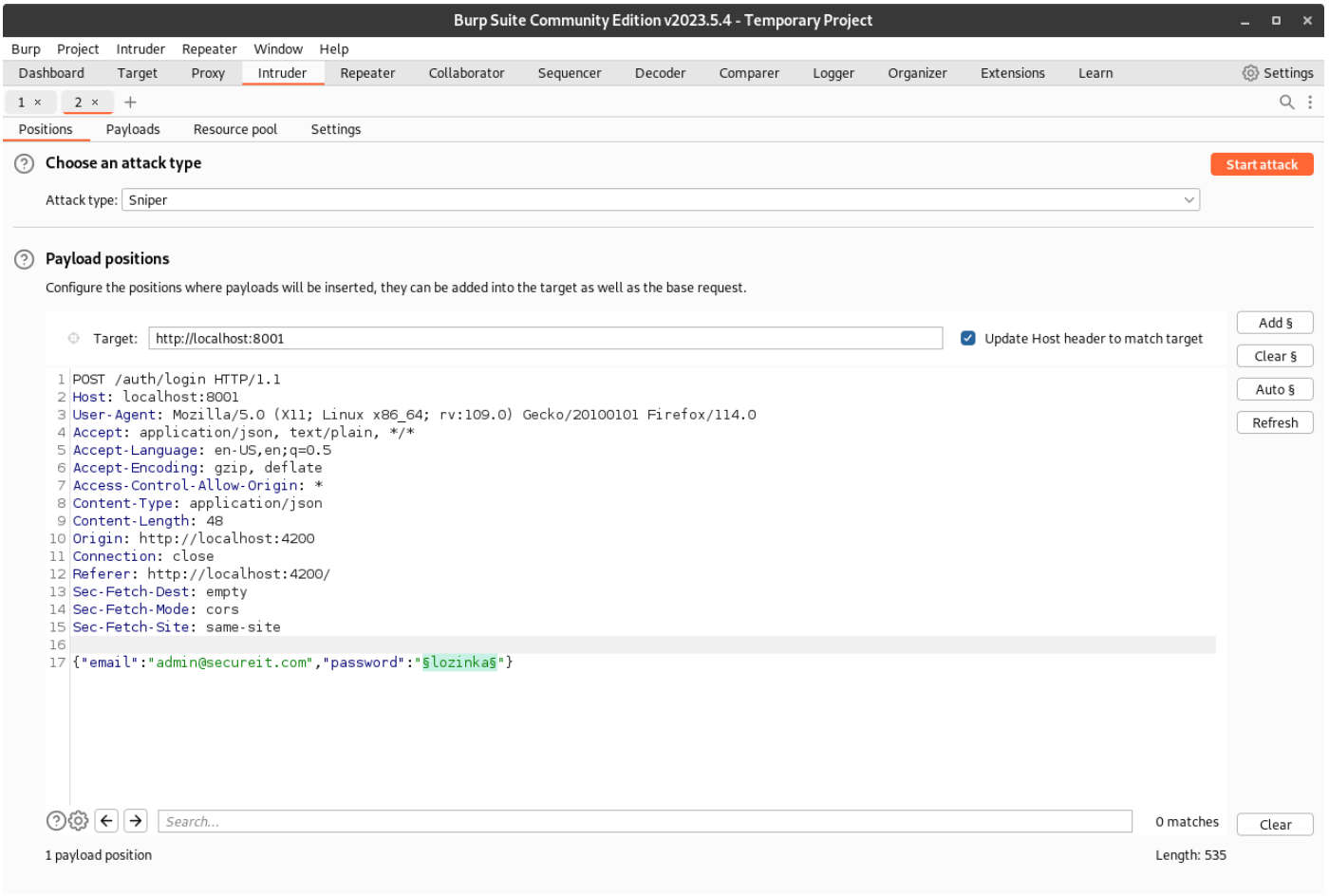


Pokušali smo prijavljivanje sa tačnom i netačnom lozinkom na hipotetičkom nalogu kojem napadač već ima pristup. Analizom odgovora servera, primećeno je da se razlikuju u statusu. Unos tačne lozinke vraća status 200, dok je za netačnu status 401.

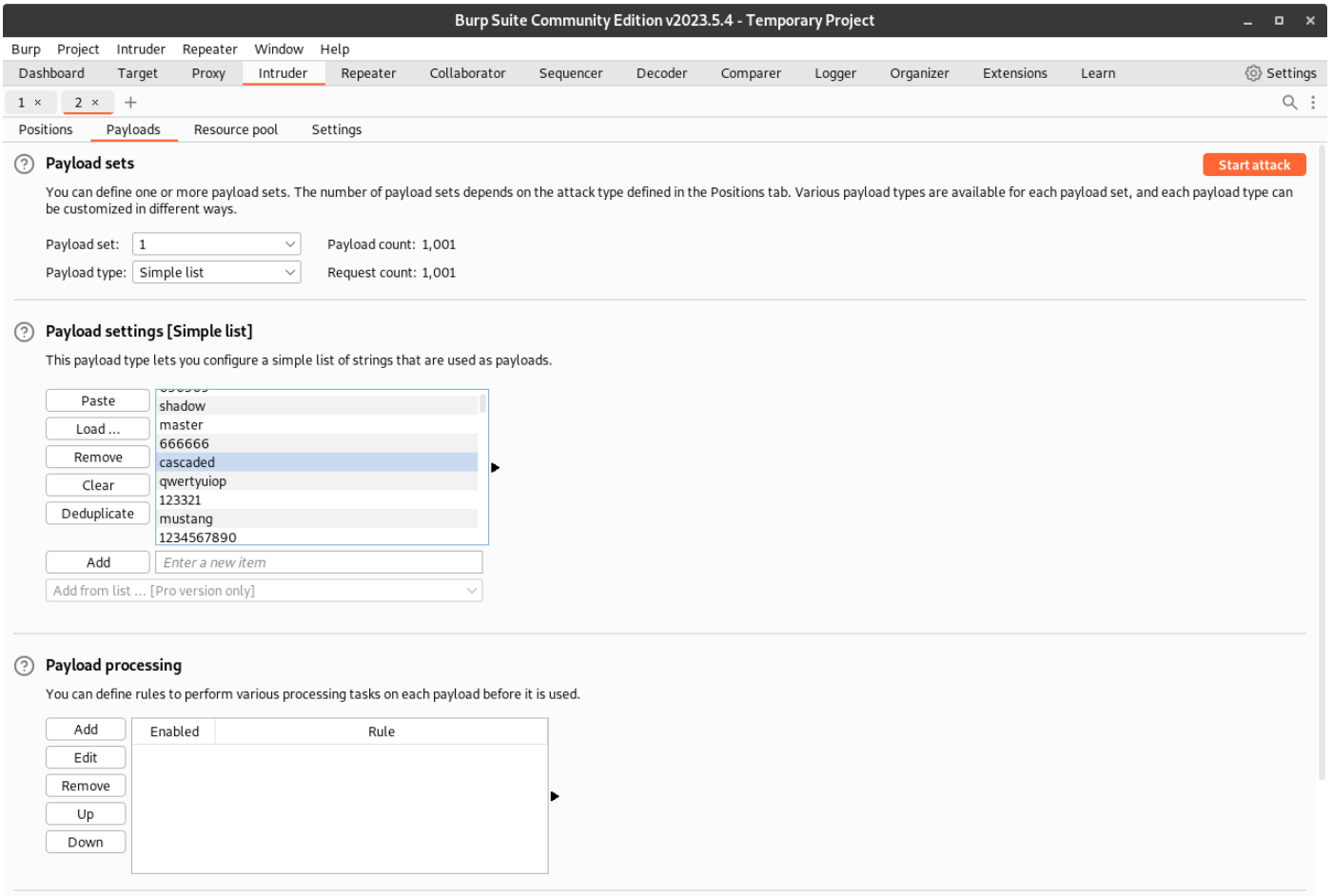


Intruder — brute-force napad

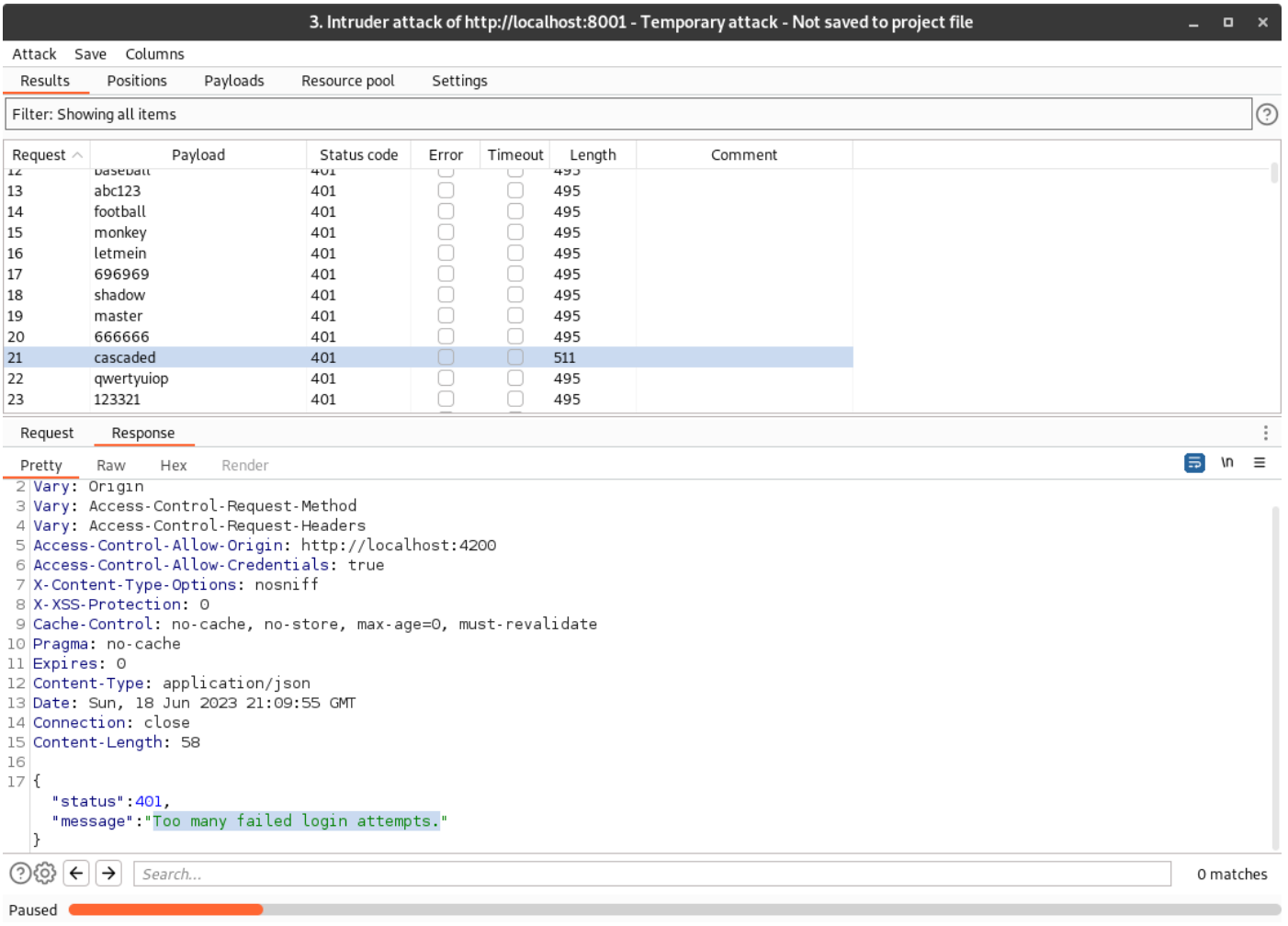
Nakon toga, zahtev smo prosledili u Intruder alat. Označili smo deo zahteva koji predstavlja lozinku, a kao payload korišćena je lista od 1000 čestih lozinki, uz dodatak tačne lozinke radi simuliranja napada koji ima šansu da uspe.



Prava lozinka do koje napadač pokušava da dođe je "cascaded", i kao što smo rekli nalazi se u našoj listi.



Rezultat napada bio je da je za svaku lozinku, uključujući i tačnu, vraćen error 401. Međutim, kod tačne lozinke primećeno je da se dužina odgovora razlikuje od uobičajene dužine odgovora za sve ostale lozinke (511 bajtova u odnosu na uobičajenih 495 bajtova).



Odgovor servera za bilo koju lozinku je:

```
{"status":401,"message":"Bad credentials"}
```

Dok je za tačnu lozinku:

```
{"status":401,"message":"Too many failed login attempts."}
```

Iako nije uspelo prijavljivanje na tuđi nalog (još uvek), ovaj napad je omogućio otkrivanje lozinke za tog korisnika. Kroz odgovor servera za tačnu lozinku, vidljiva je poruka koja objašnjava razlog zaključavanja naloga. U logovima je takođe zabeleženo da je nalog zaključan, uz naveden isti razlog:

```
2023-06-18T23:09:51.661101196+02:00  WARNING [AUTHENTICATION] : Account of user
admin@secureit.com has been locked until 2023-06-18T21:29:51.660462643Z because 'Too many failed
login attempts.'.
```

Napadaču bi bilo dovoljno da sačeka da zaključavanje naloga istekne, i da se zatim uloguje koristeći otkrivenu lozinku.

Nakon malo čekanja, uspešno smo se ulogovali.

```
Referer: http://localhost:4200/
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: same-site

{
  "email":"admin@secureit.com",
  "password":"cascaded"
}
```

```
Content-Type: application/json
Date: Sun, 18 Jun 2023 21:29:27 GMT
Connection: close
Content-Length: 94

{
  "status":200,
  "message":
  "Please provide 2FA code along with the credentials to
  authenticate."
}
```

Zaključak

Ovim pentestingom zaključili smo da je moguće zaobići implementiran mehanizam zaštite od brute-force napada. U našem slučaju, mehanizam za zaključavanje naloga otkriva poruku o razlogu zaključavanja korisnicima koji unesu validne kredencijale. To dovodi do toga da je poruka različita za tačnu i netačnu lozinku, što izdvaja tačnu lozinku od netačnih.

Jedno od mogućih rešenja bi bilo eliminisati ovu razliku u odgovorima, što se može postići na tri načina:

- Uvek vraćati razlog zaključavanja — Bilo bi dozvoljeno svakome da otkrije razlog zaključavanja naloga drugih korisnika, što možda otkriva previše informacija. ❌
- Nikada ne vraćati razlog zaključavanja — Korisnici ne mogu znati da im je nalog zaključan. ❌
- Vraćati razloge zaključavanja, osim u slučaju brute-force napada — Ukoliko je nalog zaključan samo iz ovog razloga, ne vraćati razlog zaključavanja već "Bad credentials" kao i za svaku drugu lozinku. ✔️

Takođe se može i razmotriti neki drugi mehanizam, koji bi umesto zaključavanja naloga blokirao IP adresu napadača.