Penetration testing izvještaj – Tim 9

SEP 2023

# Tim

* Miloš Popović, R2-31/2022
* Veljko Tošić, R2-4/2022
* Ivan Mršulja, R2-30/2022
* Marko Bjelica, R2-10/2022

# Korišćeni alati

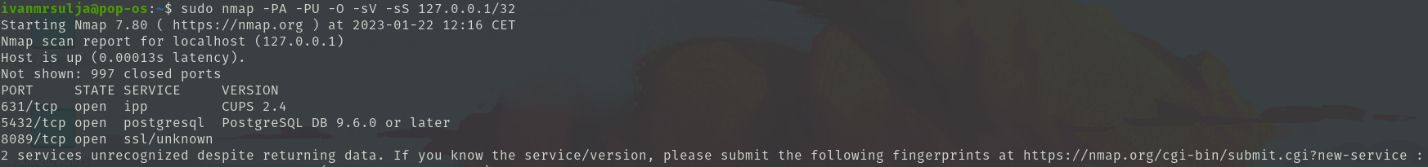
Za realizaciju dodatnog zadatka (*penetration testing*) odlučili smo se na korišćenje 2 alata:

1. *Nmap*
2. *Nikto*

Kao bonus, „konsultovali“ smo i *Github Dependabot*-a kako bi dobili obavještenja o eventualnim ranjivostima u zavisnostima koje naš projekat koristi.

# Nmap

Prilikom svakog *penetration testing*-a prvo je potrebno izvršiti skeniranje mreže, u cilju detekcije svih dostupnih servisa, sakupljanja informacija o njima i planiranju sledećeg koraka. *Payment Service Provider*(PSP u nastavku) servis je pokrenut lokalno, sa svim svojim neophodnim zavisnostima i pokrenuto je skeniranje lokalne mreže.

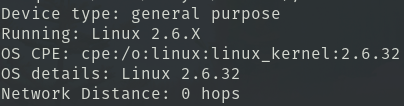


Prvo je bitno napomenuti da je opsjeg skeniranih adresa smanjen na samo jednu kako ne bi gubili vrijeme na filtriranje gomile ispisa koji ne znači ništa, u praksi bi se pokrenulo na nekom realnijem opsjegu npr. 127.0.0.0/24.

Iz ispisa alata, možemo primijetiti dobro poznate servise kao što su *PostgreSQL* baza i *IPP*, kao i neki nepoznati *SSL* servis. Naravno, u pitanju je *backend* naše aplikacije, ali kako je sertifikat ručno generisan, alat ne može sa sigurnošću znati o kome se radi. Takođe, možemo vidjeti i to, da se alat pokušao obratiti i ka dva nepoznata servisa na mreži za koje je izgenerisao sledeće *fingerprint*-e.



U pitanju su *frontend* i *backend* servis naše aplikacije, respektivno. Bitno je napomeuti da *TLS/SSL* nije bio podešen za *frontend* u trenutnku testiranja, međutim, to nije predstavljao problem kako bi se testiranje sprovelo uspješno. Vidimo da je alat pokušao otvoriti *TCP* konekciju sa *frontend* serverom, gdje je dobio *Method Not Supported* kao odgovor (očekivano), dok mu je *backend* server javio da je potrebno prvo odraditi potpunu autetifikaciju kako bi pristup resursima bio moguć (takođe očekivano).

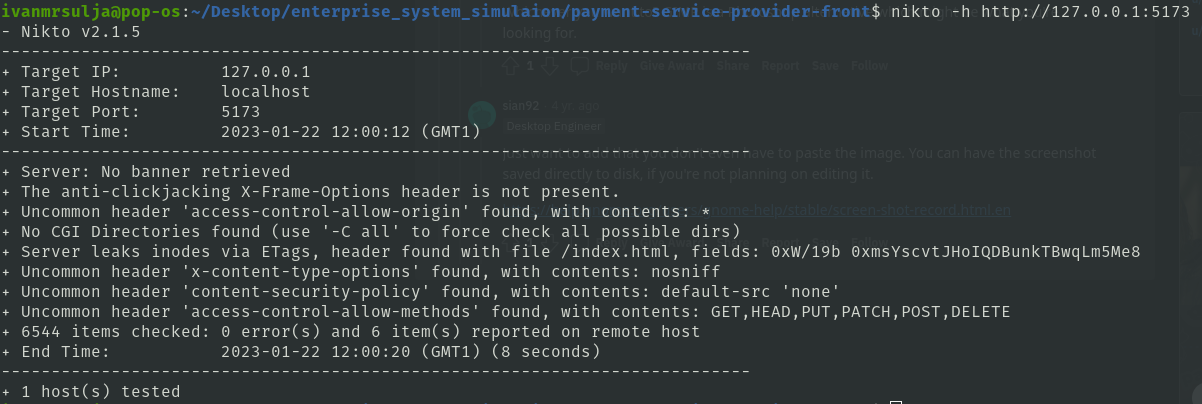


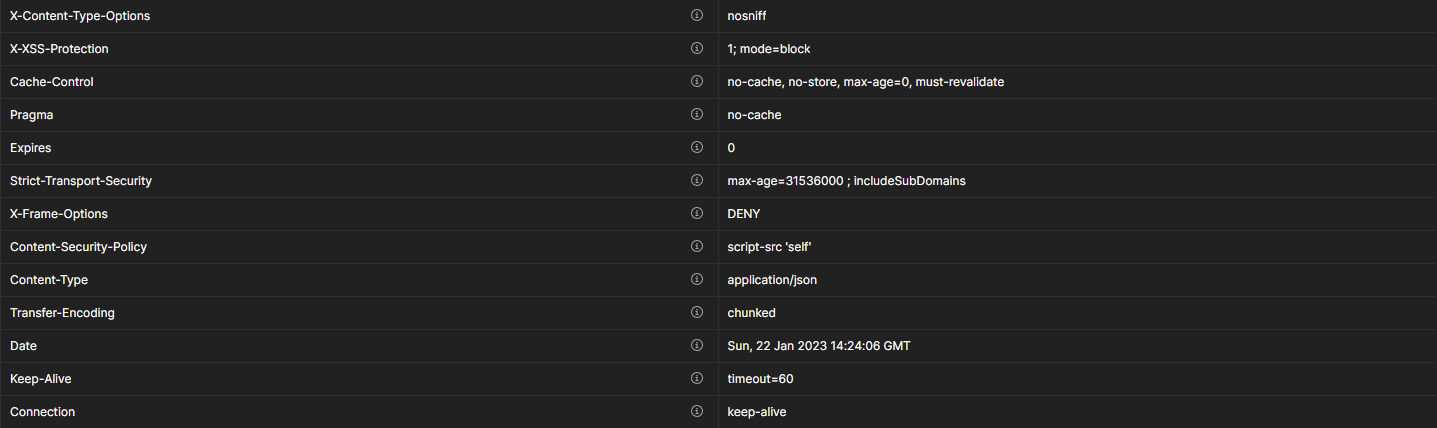
Takođe, alat je pronašao i neke detalje vezane za *host* mašinu na kojoj su pokrenute instance ovih servisa, gdje vidimo da je riječ o *up-to-date* verziji *Linux*-a na lokalnoj mreži (*network distance* je 0).

Sa prikupljenim podacima možemo krenuti dalje u ispitivanje rajivosti samih servisa.

# Nikto

Za ispitivanje ranjivosti aplikacije korišćen je *Nikto* alat. Pokrenuto je skeniranje *frontend* dijela aplikacije s obzirom na to da je *backend*-u nemoguće pristupiti direktno zbog podešenog CORS-a.

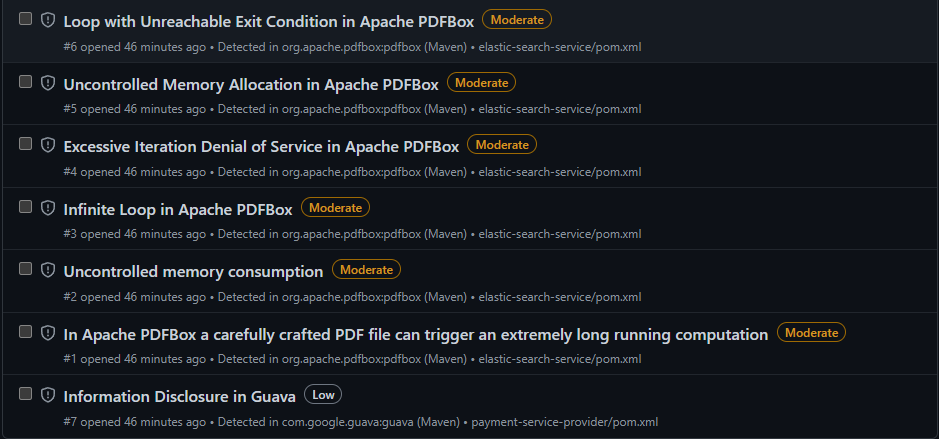


Prije nego iskomentarišemo rezultate skeniranja, bitno je napomenuti da nam je *vite* server pukao prilikom testiranja gdje je sajt postao neupotrebljiv dok se nije odradio *refresh* stranice. Razlog za ovo je *debug* režim rada, koji je bacao grešku prilikom pokušaja pristupa putanji koja nije definisana u *vue router*-u. Malom ispravkom u *vite-config.js* fajlu da se koristi *build* režim rada, nestala je i greška te je testiranje nastavljeno. Nedostatak *X-Frame-Options header*-a nas je zbunio, s obzirom da je eksplicitno podešen na *backend*-u. 

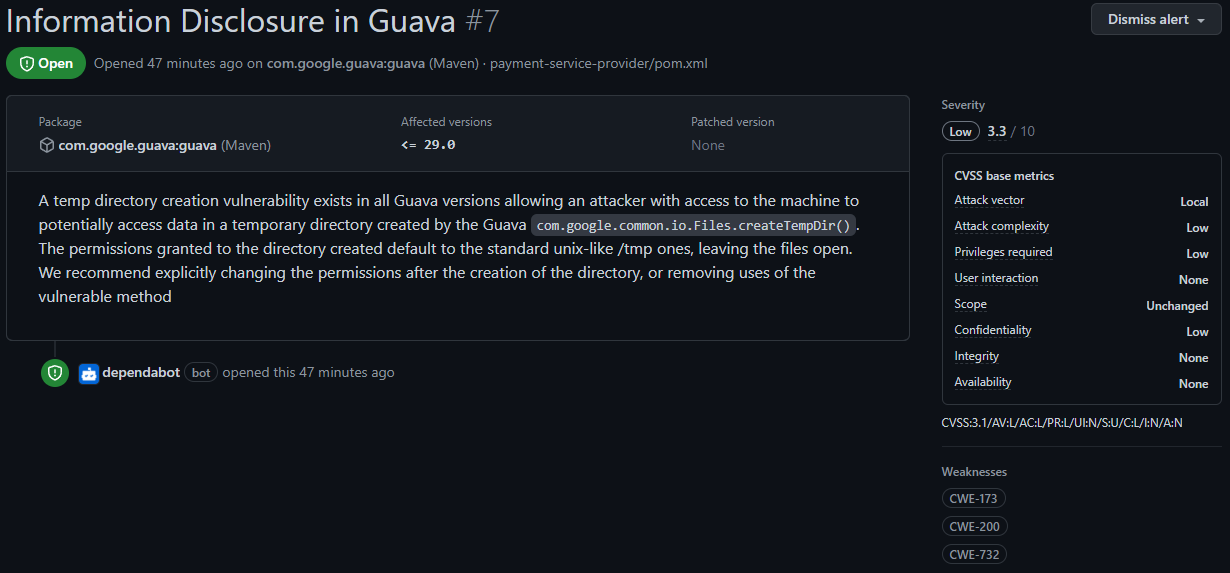
Onde gdje nije podešen, to je sam *vite* server, gdje nam realno nije ni potreban jer se dobavljaju samo *index.html* sa uglifikovanim i minifikovanim .*js bundle*-ovima, tako da je ovo okarakterisano kao *false positive*. Slična je i situacija sa *access-control-allow-origin*, koji iako je a *backend*-u strogo podešen da može primati zahtjeve samo od onih servisa od kojih mora, logično je da na frontendu bude “\*” jer njemu treba da svi mogu da mu pristupe. Prijavljeni *server leak*-ovi preko *ETag*-ova su, srećom, takođe *false positive*, jer se *ETag*-ovi nalaze u pojedinim sličicama koje koristimo na *frotend*-u i ne sadrže nikakve osjetljive informacije. Posljednja 3 prijavljena *uncommon header*-a ne predstavljaju grešku već samo daju određene detalje implementacije. Štaviše, ovo je i potvrda da je konfiguracija na *backend*-u dobro odrađena s obzirom da su vrijedosti ovih *header*-a ručno postavljane u konfiguraciji *Spring Security*-a.

# Github Dependabot

*Github dependbot* prijavio je puno ranjivosti:



Prvih 6 vezano je za *pdfbox* biblioteku i njenu lošu implementaciju, koja čak nije ni korišćena za realizaciju funkcionalosti *PSP* servisa, već *ES* servisa za potpuno drugi predmet. Sve u svemu, *update*-ovanjem verzije biblioteke, otklonjeno je i prvih 6 problema. Realan problem bio je *Information Disclosure* problem u *Guava* biblioteci u *PSP* servisu, pogotovo jer se ona koristi kao tranzitivna zavisnost u *bitcoinj* biblioteci.



Za izvršenje ovog nnapada potreban lokalni pristup serveru, te je jedini način zaštite manuelno postavljanje prava pristupa na kreirani *temp* fajl. Ništa kritično, ali zanimljivo da se navede.