SVEUČILIŠTE U SPLITU

FAKULTET ELEKTROTEHNIKE, STROJARSTVA I BRODOGRADNJE

Seminarski rad

EVIL TWIN

Računalna forenzika

Dino Sudić

Split, lipanj 2020.

# SADRŽAJ

**1. UVOD** 3

**2. ESP8266**  4

**3. CAPTIVE PORTAL**  6

**4. DEAUTENTIKACIJA**  7

**5. PRAKTIČNA REALIZACIJA**  9

**5.1. Evil twin** 9

**5.2. Deauther** 11

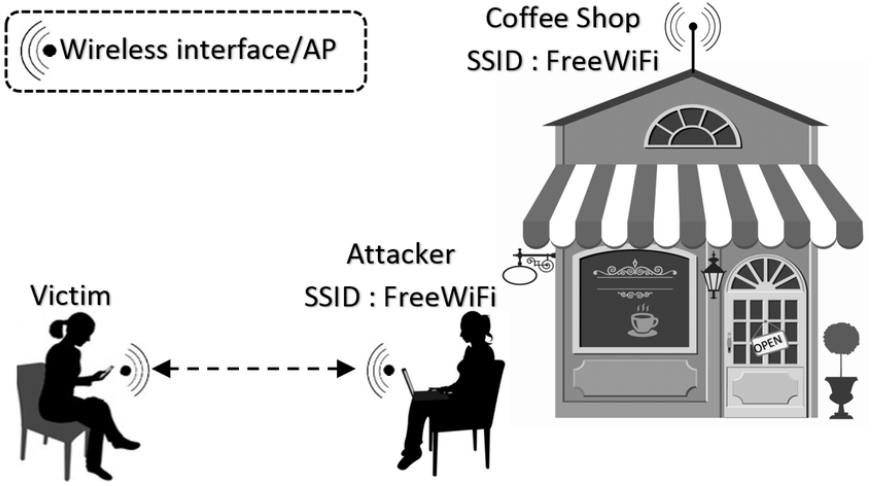
**6. SPRJEČAVANJE NAPADA**  13

**LITERATURA**  14

1. **Uvod**

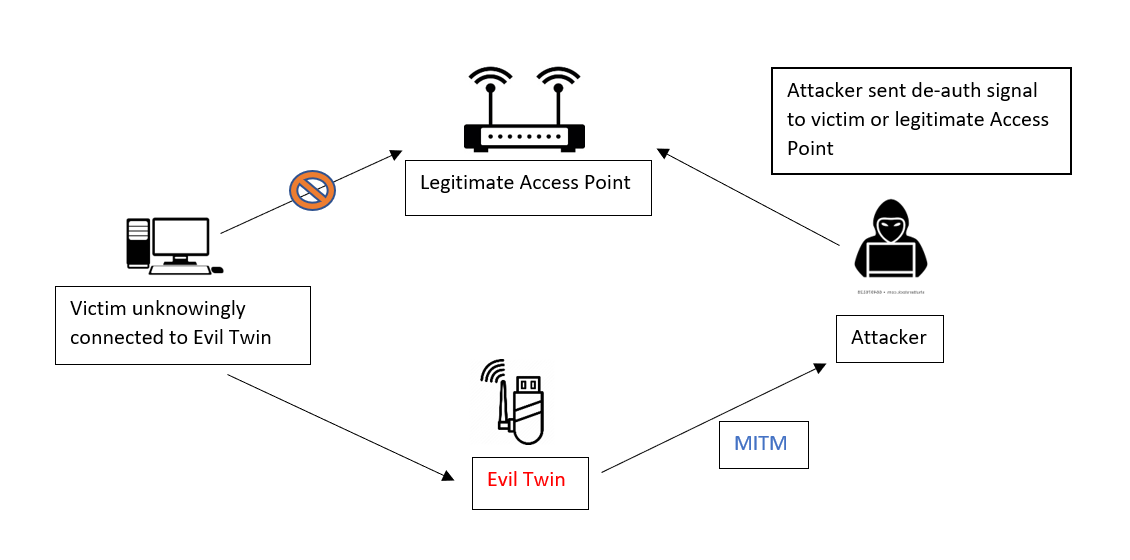
Današnje bežične mreže posjeduju mnoge ranjivosti koje napadač može iskoristiti za prisluškivanje i krađu osjetljivih podataka, a da korisnik za to niti ne zna.

Jedan od takvih napada je „evil twin“ napad kod kojeg napadač kreira lažnu pristupnu točku istog imena kao i ciljana legitimna WiFi mreža, a zatim čeka da se korisnik spoji na njegovu lažnu pristupnu točku. Kada se korisnik spoji, ideja je proslijediti ga na „captive portal“ u kojem se od njega traži da unese nekakve informacije koje bi napadaču mogle biti od koristi (npr. šifra za pristup WiFi mreži koju se napada, ukoliko je ona zaštićena), a koje se zatim direktno šalju napadaču.



Kako bi natjerali korisnika da se spoji na lažnu mrežu, možemo ga izbaciti sa legitimne mreže i onemogućiti mu da se ponovno spoji, ali o tome detaljnije malo kasnije. Na taj način će korisnik odustati od daljnjeg pokušavanja spajanja te će se, nadamo se, spojiti na lažnu mrežu.

Ako je signal lažne pristupne točke jači od signala legitimne pristupne točke tada će se uređaj kojim se korisnik spaja automatski prebaciti na drugu (lažnu) mrežu istog imena pa je tako prvi dio napada uspješno obavljen.



U ovom projektu je pokazano kako se jednostavno može realizirati „evil twin“ napad uz pomoć vrlo jeftinog ESP8266 mikrokontrolera.

1. **ESP8266**

Napad će biti realiziran koristeći NodeMCU IoT razvojnu pločicu koja se temelji na ESP8266 WiFi SoC-u (*System on a Chip*) s integriranim 32-bitnim mikrokontrolerom.



**Značajke:**

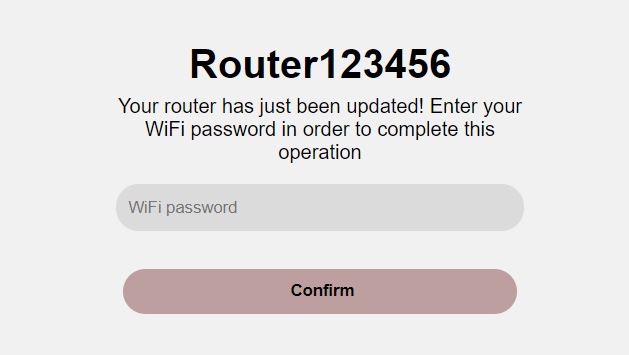
* Tensilica 32-bit RISC CPU Xtensa LX106 @ 80MHz
* Radni napon: 3.3V
* Ulazni napon; 7-12V
* Digitalni U/I pinovi: 16
* Analogni ulazi: 1
* UART, SPI, I2C, I2S
* PWM
* 10-bitni ADC
* Flash memorija: 4MB
* SRAM: 64kB
* Ugrađen USB – TTL bridge (CP2102)
* IEEE 802.11 b/g/n Wi-Fi
* Buđenje i transmitiranje za < 2ms
* Sigurnost: WEP / WPA-PSK / WPA-PSK
* Mrežni protokoli: IPv4, TCP / UDP / FTP / HTTP
* Potrošnja do 80mA tijekom transmisije, 20µA u sleep modu

Najvažnije značajke ovog modula potrebne za realizaciju „evil twin“ napada su mogućnost konfiguracije u pristupnu točku na koju se korisnici spajaju te mogućnost slanja proizvoljnih paketa, iako ograničenih veličina, ali dovoljnih za deautentikaciju korisnika sa ciljane mreže.

1. **Captive portal**

Jedan od najčešćih tipova „evil twin“ napada je onaj u kojem se koristi captive portal. Captive portal je web stranica kojoj se pristupa putem web preglednika, a koja se prikazuje novopovezanim korisnicima WiFi ili žičane mreže prije nego što im se odobri širi pristup mrežnim resursima.

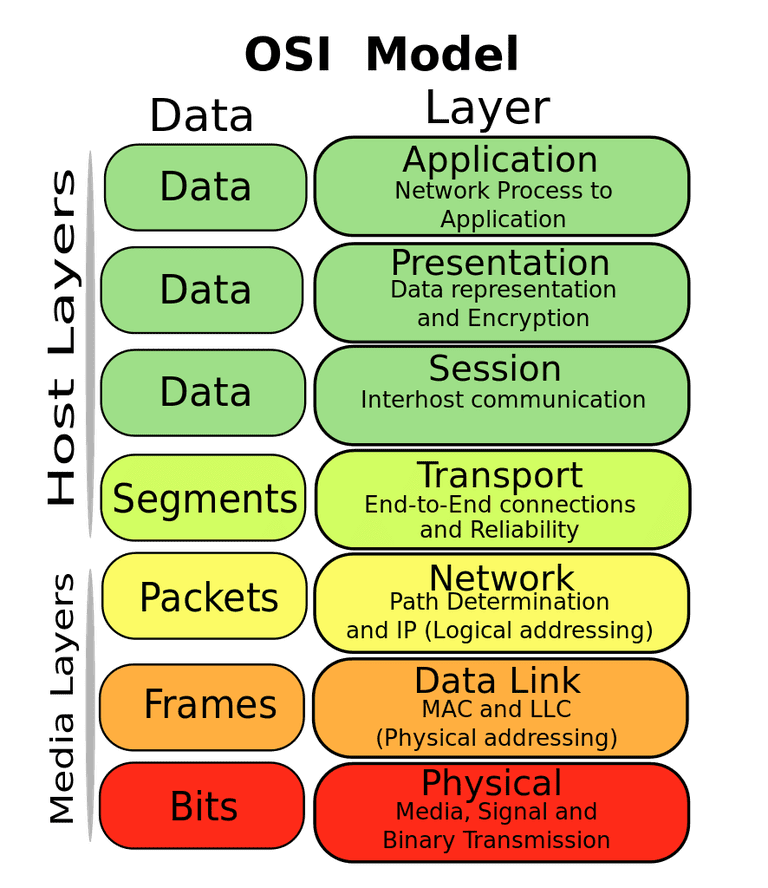
To je nešto na što su korisnici naviknuti viđati nakon spajanja na neke mreže pa ćemo upravo to iskoristiti u našu korist kako bi dobili šifru od legitimne mreže prikazujući captive portal koji izgleda kao da je potrebno ažuriranje routera.



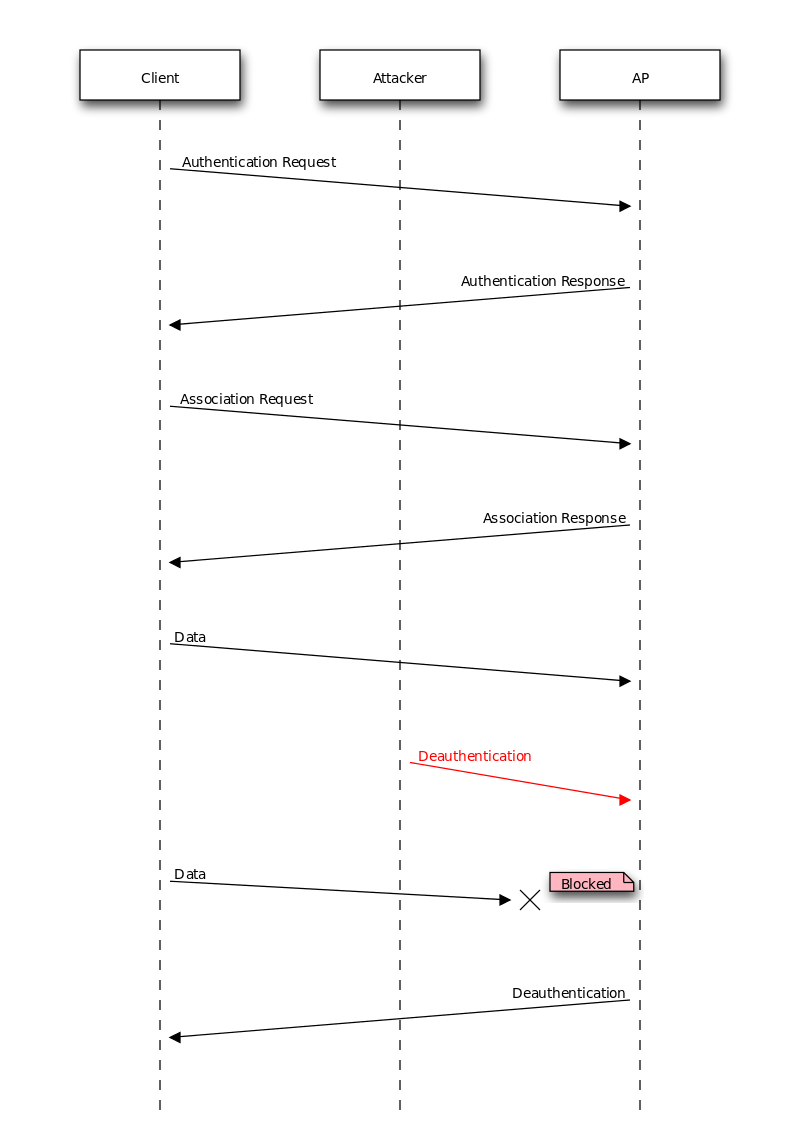
U ovom koraku praktički možemo što god nam padne na pamet prikazivati u captive portalu, npr. možemo lažirati web stranicu za prijavu na Facebook ili email gdje će se od spojenih korisnika tražiti unos email adrese i šifre.

1. **Deautentikacija**

IEEE 802.11 standard sastoji se od tri glavna tipa okvira, to su okviri za upravljanje, kontrolni okviri i podatkovni okviri, a koriste se u MAC ([*Medium access control*](https://en.wikipedia.org/wiki/Medium_access_control)) sloju koji je dio podatkovne razine OSI modela.



Najvažniji okvir za realizaciju ovog napada je upravljači okvir za deautentikaciju kojeg pristupna točka šalje korisniku kako bi prekinula vezu s njim.



Ovi okviri predstavljaju samo obavijest, a ne zahtjev korisniku jer se ne očekuje nikakva potvrda od korisnika nakon primanja deautentikacijskog okvira te ga primatelj ne može odbiti.

Napadač iskorištava ranjivost deautentikacijskih okvira koji nisu enkriptirani pa samo poznavajući MAC adresu žrtve kreira lažni okvir koji šalje pristupnoj točki, a pristupna točka misleći da je deautentikacijski okvir stigao od žrtve, prekida vezu s njom.

Kod evil twin napada kontinuirano šaljemo deautentikacijske okvire pristupnoj točci sa MAC adresom žrtve kako bi joj onemogućili ponovno spajanje na legitimnu mrežu, a budući da mi emitiramo lažnu mrežu sa istim SSID-jem, tako će se žrtvin uređaj automatski spojiti na lažnu mrežu.

Važno je napomenuti da ovakvo slanje deautentikacijskih paketa predstavlja DoS (*Denial of Service*) napad!

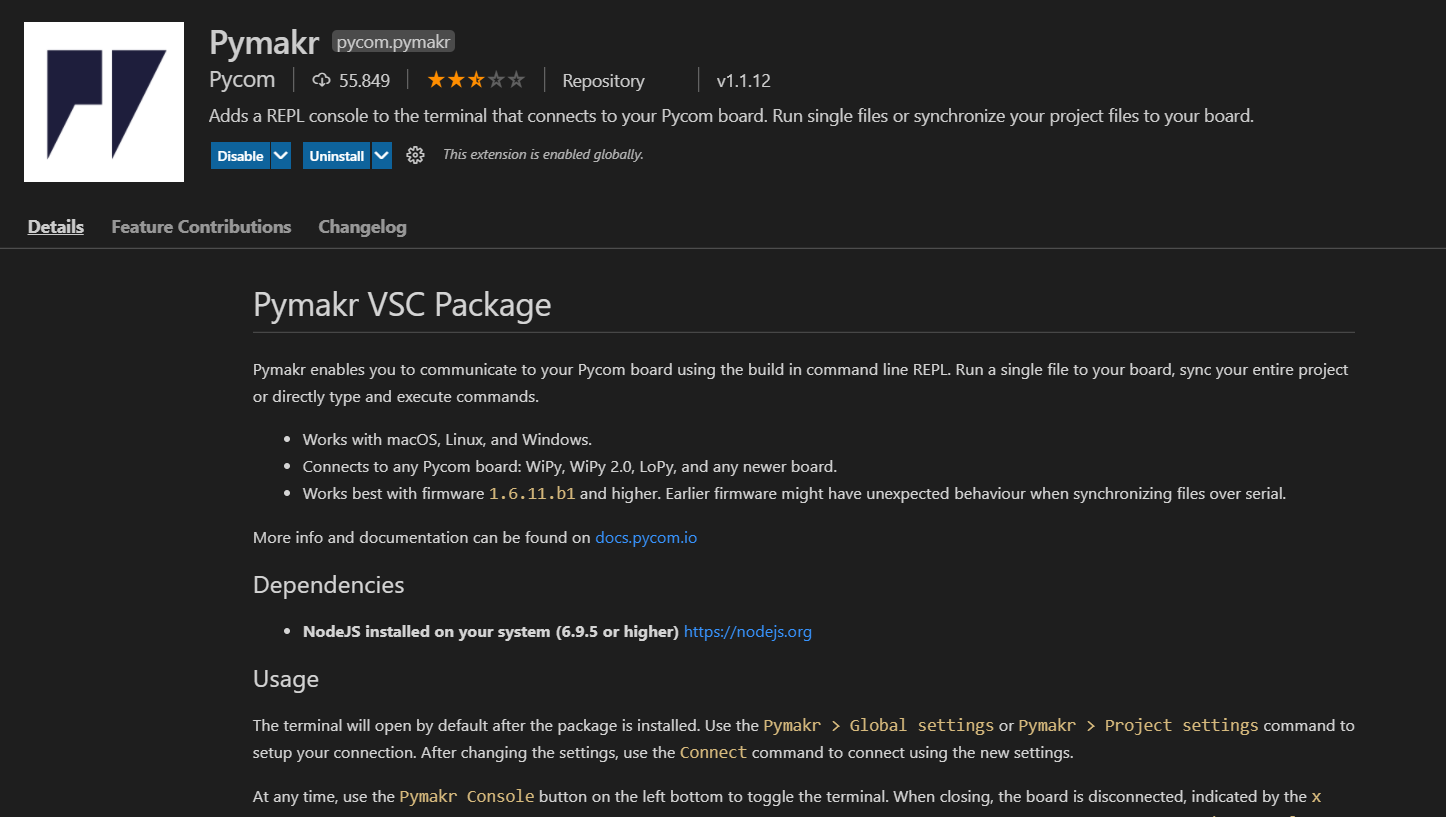
IEEE 802.11w predstavlja dopunu 802.11 standardu kojim se povećava sigurnost upravljačkih okvira enkriptiranjem samih okvira. Prilikom primanja ovakvih upravljačkih okvira, primatelj provjerava integritet istih pa postoji mogućnost da ih samo ignorira čime se znatno otežava deautentikacijski napad. Da bi komunikacija bila uspješna, obje strane moraju podržavati ovaj standard.

1. **Praktična realizacija**

S obzirom da ESP8266 ima samo jednu ugrađenu antenu, nije moguće da je istovremeno konfiguriran kao pristupna točka i da šalje deautentikacijske okvire pa će se koristiti dva ESP8266 mikrokontrolera, jedan za realizaciju evil twin napada (lažna pristupna točka) i jedan za deautentikaciju korisnika sa legitimne mreže.

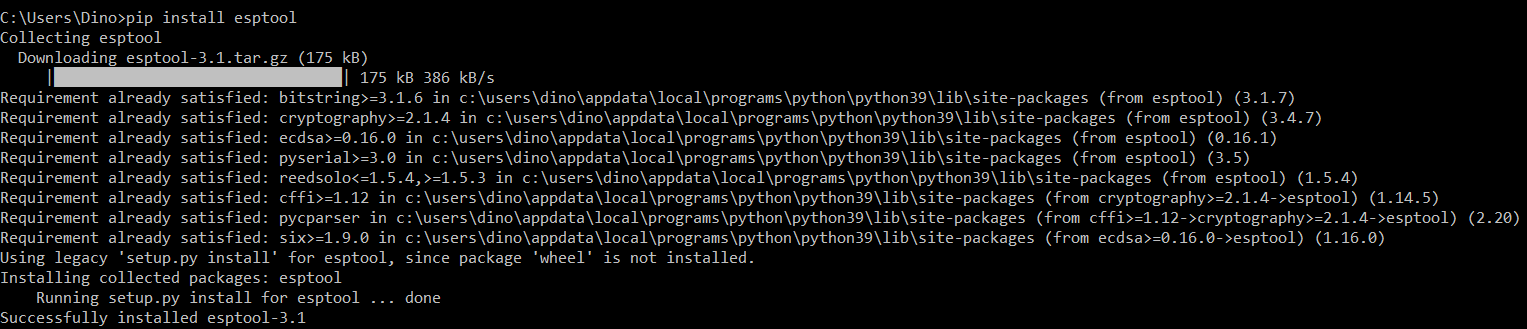
* 1. **Evil twin**

Ovaj ESP8266 mikrokontroler programiran je u Micropythonu koristeći Visual Studio Code sa Pymakr proširenjem.

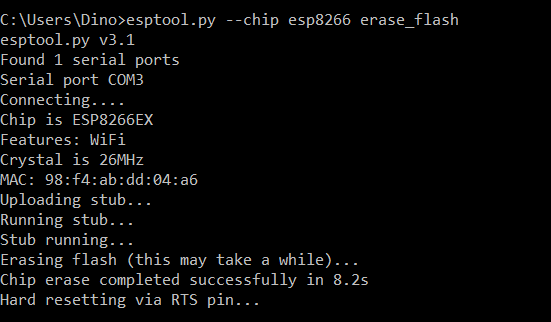


Kako bi ESP8266 podržavao Micropython, potrebno je prethodno u flash memoriju učitati novi firmware koji je naveden na GitHub stranici projekta [13].

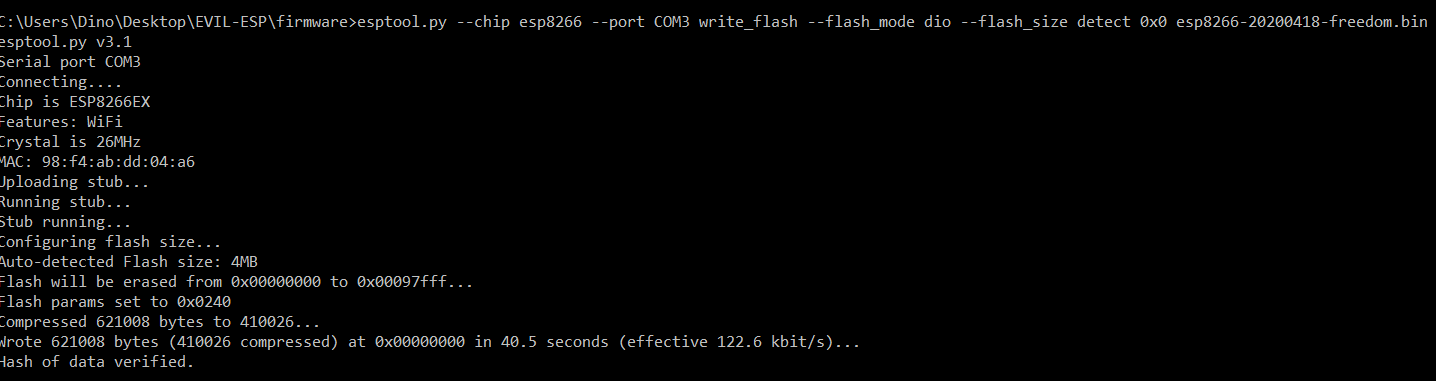
Za učitavanje firmware-a korišten je alat ***esptool***kojeg je moguće instalirati iz terminala naredbom ***pip install esptool*** uz uvjet da računalo ima instaliran *Python.*



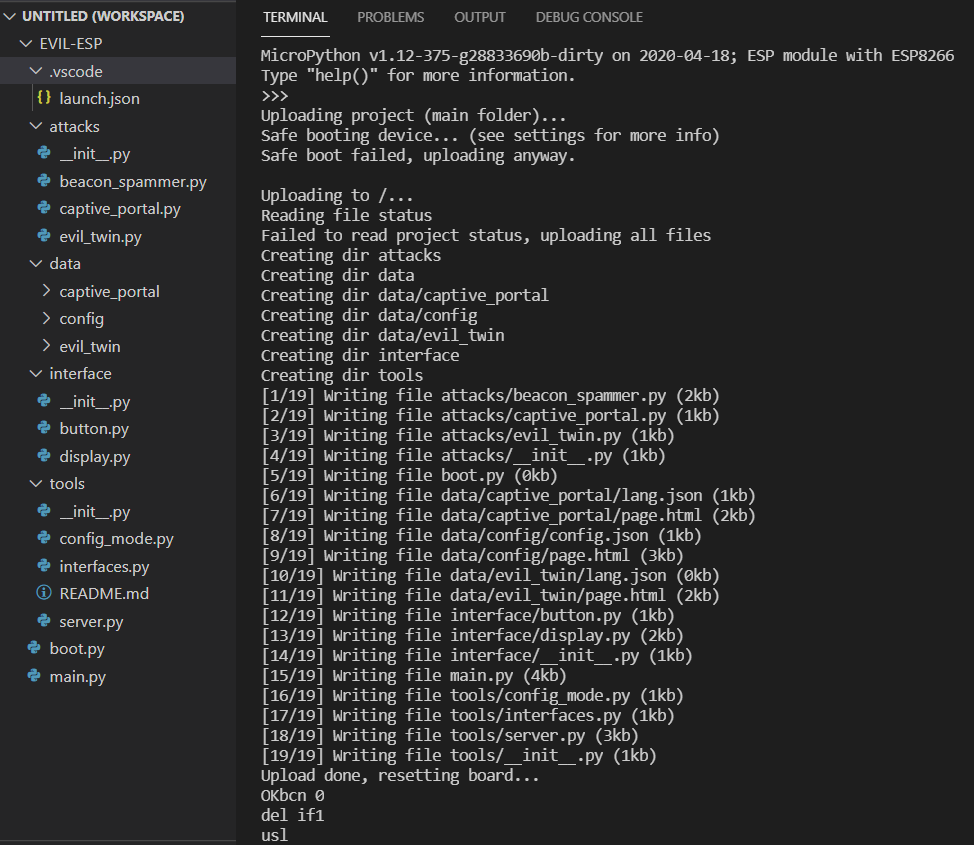
Prije učitavanja novog firmware-a potrebno je izbrisati flash memoriju naredbom ***esptool.py --chip esp8266 erase\_flash***



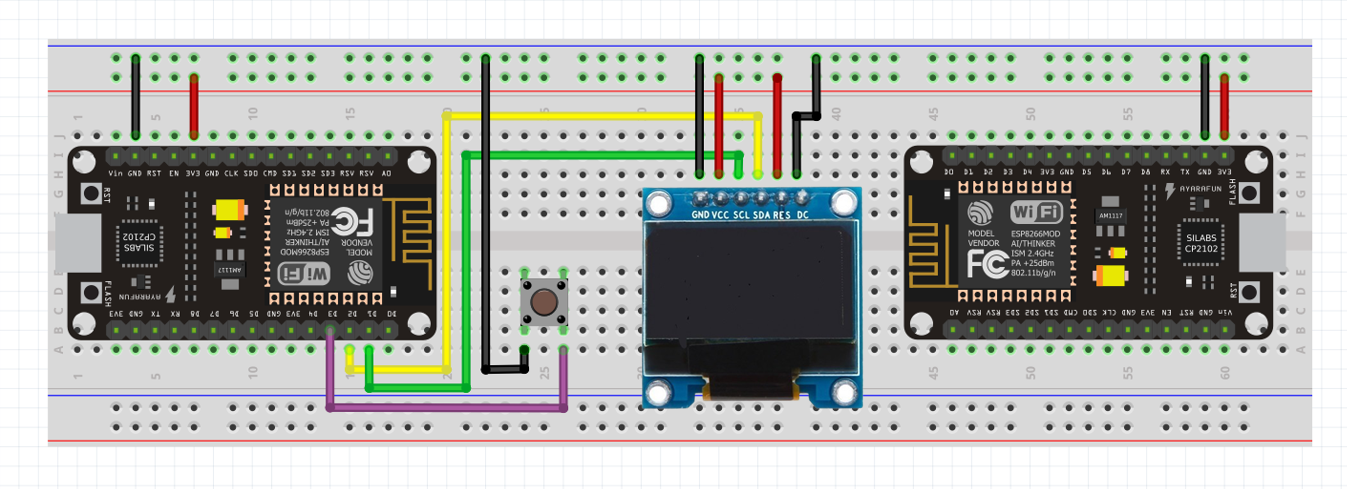
Zatim se pozicioniramo na lokaciju gdje se nalazi novi firmware te ga učitamo u flash memoriju naredbom ***esptool.py --chip esp8266 --port COM3 write\_flash --flash\_mode dio --flash\_size detect 0x0 esp8266-20200418-freedom.bin***



Konačno, otvorimo cijeli projekt u Visual Studiu te ga uploadamo na ESP8266.



Kako bi uređaj bio prenosiv i neovisan o računalu dodat ćemo OLED displej i jednu tipku za navigaciju. Shema spajanja na protip pločici je prikazana ispod:

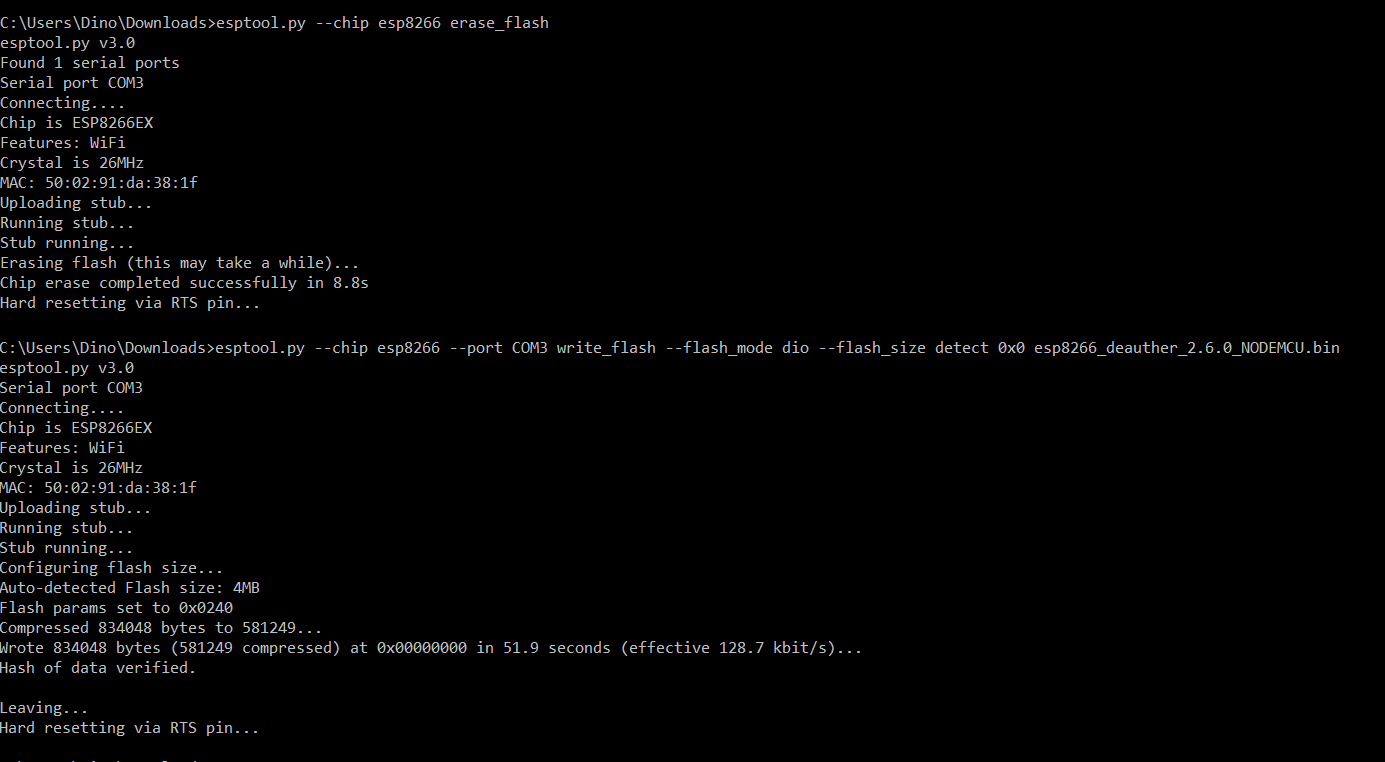


* 1. **Deauther**

Novije verzije ESP8266 mikrokontrolera dolaze sa ograničenim funkcijama za slanje proizvoljnih paketa upravo da bi se izbjegla zlouporaba istih.

Međutim, na internetu je moguće naći prilagođeni firmware sa svim funkcionalnostima koje će nam omogućiti slanje deautentikacijskih okvira.

Pa zato i u drugom ESP8266 mikrokontroleru najprije brišemo flash memoriju, a zatim učitavamo novi firmware sa GitHub stranice projekta [14].

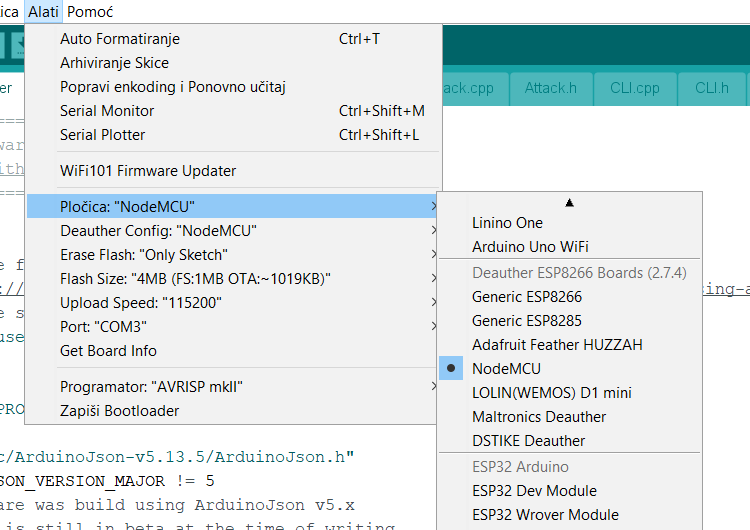


Za ovaj slučaj ESP8266 ćemo programirati koristeći Arduino razvojno okruženje. Razvijene su posebne pločice za ovaj projekt koje je potrebno instalirati i odabrati odgovarajuću prije samog učitavanja programa na pločicu.

Za instaliranje idemo unutar Arduino okruženja pod ***File > Preferences*** i pod ***Additional Boards Manager URLs*** dodamo: ***https://raw.githubusercontent.com/SpacehuhnTech/arduino/main/package\_spacehuhn\_index.json***

Zatim pod ***Tools > Bord > Board Manager*** upišemo *deauther* i instaliramo ***Deauther ESP8266 Boards.***

Na kraju odaberemo ispravnu ESP8266 pločicu i port te učitamo program.



Kada je program učitan možemo se spojiti na WiFi mrežu koju ovaj ESP emitira i u Internet preglednik upisati 192.168.4.1. čime ulazimo u web sučelje za odabir željenih uređaja koje želimo izbaciti s mreže.

1. **Sprječavanje napada**

Veliki dio korisnika nije upoznat s ovakvim i sličnim napadima te će se u većini slučajeva uvijek spojiti na bilo koju otvorenu WiFi mrežu.

U istraživanju provedenom u Americi nad preko 1000 ispitanika više od 60% korisnika smatra da su njihovi podaci sigurni pri korištenju javnog interneta, a pri tom misle da su internet stranice koje posjećuju odgovorne za zaštitu njihovih podataka.

Kolike posljedice može imati napad pokazuje i činjenica da neke od najčešćih aktivnosti pri korištenju javnog interneta uključuju prijave u osobne e-mail račune, prijave na društvene mreže, pristup bankovnim računima i raznim financijskim podacima...

Najjednostavniji i najbrži način zaštite od ovakvog napada je izbjegavanje spajanja na otvorene i javne WiFi mreže.

Za zaštitu prilikom korištenja javnih mreža preporuča se korištenje VPN-a koja šifrira sav promet te povećava sigurnost i privatnost.

Općenito, ako je već potrebno služiti se javnom WiFi mrežom, budite svjesni rizika i ne unosite previše osobnih informacija.

*“Think about the cost of being connected all the time. Nothing is free”* - David Lee, Norton Security

Daljni razvoj samog projekta bi bio u smjeru hvatanja *handshake*-a koji bi služio za provjeru ispravnosti unesene WiFi šifre u captive portalu. Ako je zaporka točna, prekini vezu s korisnikom nakon čega se on ponovno spaja na legitimnu mrežu kao da se ništa nije dogodilo, a ako nije, zatraži ponovni unos šifre.

zatim realizacija MITM napada.

**LITERATURA**

1. <https://www.researchgate.net/figure/Illustration-of-an-Evil-Twin-Attack-The-attacker-can-successfully-lure-a-victim-into_fig5_321122614>
2. <https://www.thecybersploit.com/2019/12/Hacking-WPA2-Wi-Fi-password-using-Evil-Twin-Attack.html>
3. <https://www.cnbc.com/2016/06/28/most-people-unaware-of-the-risks-of-using-public-wi-fi.html>
4. <https://en.wikipedia.org/wiki/Captive_portal>
5. <https://en.m.wikipedia.org/wiki/Evil_twin_(wireless_networks)>
6. <https://null-byte.wonderhowto.com/how-to/hack-wi-fi-stealing-wi-fi-passwords-with-evil-twin-attack-0183880/>
7. <https://en.wikipedia.org/wiki/Medium_access_control>
8. <https://en.wikipedia.org/wiki/Data_link_layer>
9. <https://commons.wikimedia.org/wiki/File:Osi-model-jb.svg>
10. <https://en.wikipedia.org/wiki/IEEE_802.11>
11. <https://en.wikipedia.org/wiki/IEEE_802.11w-2009>
12. <https://en.wikipedia.org/wiki/Wi-Fi_deauthentication_attack>
13. <https://github.com/tomellericcardo/EVIL-ESP>
14. <https://github.com/SpacehuhnTech/esp8266_deauther>
15. <https://lh5.googleusercontent.com/xfRKaTzwCHxrOEMwwz463jSzHiZIHgjT-YAwbABaVQ-qM95Bq3M86Kw8O1NfTrrHsw8I7EY86CfmzWsH5kAvtUscy-HHYsU3ERvDAK4cId1TPfjZGgxTHGivp-_REasfuQY1-reV>
16. <https://randomnerdtutorials.com/micropython-esp32-esp8266-vs-code-pymakr/>
17. <https://randomnerdtutorials.com/flashing-micropython-firmware-esptool-py-esp32-esp8266/>
18. <https://github.com/espressif/esptool>