Sigurnost računala ugrađenih u vozila

Poslončec, Saša

SVEUČILIŠTE U ZAGREBU

FAKULTET ORGANIZACIJE I INFORMATIKE

V A R A Ž D I N

Saša Poslončec

SIGURNOST RAČUNALA UGRAĐENIH U VOZILA

DIPLOMSKI RAD

Varaždin, 2020.

SVEUČILIŠTE U ZAGREBU

FAKULTET ORGANIZACIJE I INFORMATIKE

V A R A Ž D I N

Saša Poslončec

Matični broj: 44467/15–R

Studij: Baze podataka i baze znanja

SIGURNOST RAČUNALA UGRAĐENIH U VOZILA

DIPLOMSKI RAD

Mentor/Mentorica:

Doc. dr. sc. Nikola Ivković

Varaždin, travanj 2020.

Saša Poslončec

Izjava o izvornosti

Izjavljujem da je moj diplomski rad izvorni rezultat mojeg rada te da se u izradi istoga nisam koristio drugim izvorima osim onima koji su u njemu navedeni. Za izradu rada su korištene etički prikladne i prihvatljive metode i tehnike rada.

Autor potvrdio prihvaćanjem odredbi u sustavu FOI-radovi

Sažetak

Problem sigurnosti u automobilskoj industriji je prisutan od kad su se u vozila počeli ugrađivati elektronički dijelovi. Svi elektronički dijelovi su podložni nekoj vrsti napada, pa tako i dijelovi ugrađeni u vozila.

Cilj ovog rada bio je analizirati sigurnost automobila, te kroz praktičan primjer pokušati iskoristiti otkrivenu ranjivost. Većim se dijelom bazirao na softverskom dijelu problema (CAN protokol), iako je bitno naglasiti da postoji i hardverska komponenta istog, no ukoliko napadač s malicioznim namjerama dođe u posjed bilo koje hardverske komponente (to uključuje i samo vozilo), može se smatrati da je automobil u najmanju ruku ugrožen.

U uvodnom dijelu objašnjen je problem sigurnosti, kao i povijesni razvoj sigurnosnih mehanizama. Sljedeća je cjelina posvećena metodama, uređajima i alatima korištenima pri izradi ovog rada. Bez detaljnog poznavanja komunikacijskih protokola korištenih u vozilima, ovaj rad ne bi bio moguć, pa se tako treće poglavlje fokusira upravo na njih. Potom slijedi analiza sigurnosti vozila, u ovom slučaju VW Golf IV, gdje su identificirane potencijalne prijetnje, napravljen njihov model, te izvršena procjena rizika za svaku od njih. Posljednja se cjelina bavi praktičnom primjenom, odnosno pokušajem iskorištavanja jedne od prijetnji pomoću razvijenog programskog rješenja, nakon čega slijedi osvrt na temu, te zaključak rada.

Ključne riječi: CAN protokol, CAN okvir, CAN sabirnica, ECU, OBDII, DREAD, SOCKETCAN

Sadržaj

Sadržaj iii
1. Uvod 1
1.1. Povijest sigurnosti vozila 1
1.1.1. Brave i ključevi 2
1.2. Alarmni sustavi 3
1.3. Imobilizatori 4
1.4. Praćenje vozila 6
1.5. Fizičke mjere osiguranja nisu dovoljne 6
2. Metode i tehnike rada 8
3. Komunikacijski protokoli 9
3.1. ISO 9141-2 9
3.2. LIN protokol 10
3.3. FlexRay 10
3.4. CAN protokol 11
3.4.1. Mrežni model 12
3.4.2. Standard CAN, Extended CAN i vrste paketa 12
3.4.2.1. Standard CAN 13
3.4.2.2. Extended CAN 14
4. Analiza sigurnosti vozila (VW GOLF VI 2011) 15
4.1. Modeliranje prijetnji 15
4.1.1. Razina 0 17
4.1.2. Razina 1 17
4.1.3. Identifikacija prijetnji 18
4.1.4. Procjena rizika (DREAD) 20
5. Realizacija napada 23
5.1. CANSniffer 25
5.2. Analiza paketa 30
5.3. Iskorištavanje ranjivosti pomoću CANSniffer-a 35
6. Zaključak 37
Popis literature 38
Popis slika 40

1. Uvod

Ovaj rad se bavi sigurnošću automobila, no često će se susretati riječ vozilo, koje će u trenutnom kontekstu označavati automobil. Bitno je napomenuti da se metode korištene u njemu mogu primijeniti i na druge vrste vozila.

Moderna vozila krcata su elektronikom i elektroničkim uređajima. Nerijetko se viđa da jedno vozilo ima čak i nekoliko desetaka različitih računala. Ta računala funkcioniraju na način da primaju podatke od raznih senzora, te na taj način zaključuju kakvi su uvjeti u kojima to vozilo funkcionira. Na temelju dobivenih podataka, vozilo prilagođava svoj rad zbog nekoliko razloga. U kasnijim dijelovima će se na primjerima prikazati neki od tih razloga, no za sad će samo biti spomenuti:

- Osiguravanje sigurnosti svih korisnika vozila
- Sprječavanje neželjenih posljedica za vozilo
- Ugođaj vožnje za korisnika

Sigurnost bilo kojeg uređaja, pa tako i automobila je nešto na što bi se trebalo misliti od samog početka procesa planiranja. Kad se radi o fizičkoj sigurnosti vozila ili sigurnosti putnika, to možda i jest tako, no kad se radi o računalima ugrađenima u vozila i njihovim sustavima, to na žalost nije slučaj. Svi protokoli koji su se do sad koristili u vozilima za komunikaciju među uređajima, nisu bili fokusirani na sigurnost, već na brzinu prijenosa informacija ili još bolje na mogućnost istovremenog slanja mnogo informacija na mnogo mjesta. Upravo zbog toga su ti protokoli, a samim time i računala koja ih koriste, ranjivi i podložni napadima.

1.1. Povijest sigurnosti vozila

Kada se u kontekstu automobilske industrije priča o sigurnosti, u većini slučajeva se misli na zaštitu od krađe vozila. Zaštita od krađe je naravno jedan od segmenata sigurnosti samog vozila, no maliciozna namjera ne mora nužno značiti otuđivanje vozila. Ona može označavati namjeru stvaranja kvara, promjenu parametara, praćenje tog vozila i dr.

U ovom dijelu ću se baviti upravo tom fizičkom komponentom sigurnosti, jer ukoliko napadač uspije provaliti u vozilo, on dobiva neograničeni pristup svakoj komponenti istog, te može sa tim vozilom raditi što god želi. Upravo iz tog razloga proizvođači pokušavaju ograničiti pristup vozilu tako da samo osobe koje su ovlaštene (vlasnici, osobe od povjerenja) mogu pristupiti istom.

1.1.1. Brave i ključevi

U početku su samo veoma bogati mogli priuštiti automobil za osobnu upotrebu. Samim time što su bili bogati, mogli su si priuštiti vozača koji je cijelo vrijeme bio uz vozilo te ga čuvao. Kako je vrijeme prolazilo, a automobili postajali sve pristupačniji općoj populaciji koja si nije mogla priuštiti svog vozača, bilo je potrebno pronaći način da vozilo bude relativno sigurno i kad nitko nije uz njega. Upravo zbog toga su se proizvođači odlučili posvetiti sigurnosti.

Prva mjera sigurnosti koju su proizvođači odlučili implementirati bile su brave s odgovarajućim setovima ključeva. U početku su te brave bile ugrađivane na vozačkim i suvozačkim vratima, te pretincu za ruke. Nekoliko godina kasnije su slične brave počeli ugrađivati i za pokretanje motora (prije električnog načina, motor se pokretao ručno izvan vozila). [1]

Na slici ispod mogu se vidjeti primjeri ključeva koji se trenutno koriste u automobilskoj industriji.

Slika 1: Vrste ključeva

Svi ključevi prikazani na slici iznad, služe za pokretanje motora, te svaki od njih ima mehanički metalni dio (sjekirica ključa). Prvi s lijeva je najjednostavniji ključ (Zastava 750, 1980). Pri proizvodnji ovakvog ključa, on se uparuje sa specifičnom bravom tako da ni jedan drugi ključ istu ne može aktivirati. Na žalost ovaj pristup ima veliki nedostatak. Na nekim starijim automobilima bilo je dosta koristiti komad metalne žice kako bi se brava aktivirala, te samim time napadaču omogućila priliku da ukrade automobil.

Sljedeći je ključ malo napredniji (VW Golf 6, 2011). Ovaj ključ kao i svaki sljedeći ima funkcionalnost bežičnog otključavanja i zaključavanja pomoću radio valova uz pritisak na odgovarajuću tipku. Uz mehanički dio koji mora odgovarati bravi, ključ sadrži i elektronički dio. Taj dio je RFID čip (eng. *transponder*) kojeg imobilizator (uređaj koji će biti kasnije opisan) čita, te odlučuje treba li ključu dopustiti uključivanje motora.

Treći i četvrti ključ su nešto drugačiji (Renault Laguna 2003, 2007). To su zapravo pametne kartice, no postoji jedna velika razlika između ta dva ključa. Ključ sa zelenom drškom na sjekirici mora se obavezno staviti u utor za karticu u automobilu (prorez na vrhu omogućava zaključavanje kartice kako se tokom vožnje ne bi mogla izvaditi iz utora), dok ključ sa crnom drškom na sjekirici ima podršku za „Keyless GO“ tehnologiju. To znači da ključ ne mora biti u nikakvom utoru, već je dovoljno da bude blizu uređaja koji taj ključ očitava.

Sljedeći ključ (VW Passat 2012) je varijacija drugog ključa s lijeve strane, no postoje dvije bitne razlike. Dok se za drugi ključ koristi mehanička brava, za ovaj se koristi digitalna. Također ovaj ključ ima dodatnu funkcionalnost pod nazivom „Keyless Entry“. To je sustav kada nije potrebno pritisnuti ni jednu tipku za otključavanje i zaključavanje vozila, već vozilo samo detektira kada je ključ dovoljno blizu ili daleko te na temelju toga otključava i zaključava vozilo.

Posljednji ključ je potpuno drugačiji od ostalih (Mercedes-Benz 2002 – nadalje). On koristi tehnologiju infracrvenih valova kako bi komunicirao sa bravom.

Još je jednu metodu osiguranja vozila bitno napomenuti kada je riječ o ključevima i bravama, a to je zaključavanje upravljača vozila. Naime ukoliko se upravljač pokušava okretati bez ključa ili bez odgovarajućeg ključa, upravljač se zaglavi u jednom položaju te ostaje tako dok se ne upotrijebi pravi ključ odnosno u nekim slučajevima kartica.

1.2. Alarmni sustavi

Prema izvoru [1], alarmni sustavi su izumljeni od strane neimenovanog zatvorenika iz Denvera 1913. godine. Taj alarm se ručno aktivirao kada bi netko pokušao pokrenuti motor. Jest da je to bio rudimentaran pokušaj osiguravanja automobila, ali je dao ideju proizvođačima da pokušaju implementirati konkretne mjere u vozila.

Danas postoje dvije vrste alarmnih sustava koje se ugrađuju u vozila. Prva vrsta je OEM (eng. *Original equipment manufacturer*). To je alarmni sustav kojeg proizvođač ugrađuje u tvornici. Odlika takvog alarmnog sustava je da je odlično integriran u automobil, te povezan sa dosta njegovih funkcija. Druga vrsta je eng. *Aftermarket*. Takva vrsta je dobra ako se želi povećati stupanj sigurnosti vozila, no nije toliko dobro integrirana u vozilo, što može značiti da takav sustav nije toliko pouzdan kao OEM.

Na slici ispod se može vidjeti primjer aftermarket alarmnog sustava.

Slika 2: Aftermarket alarmni sustav

Bez obzira na to koje su vrste, alarmni sustavi funkcioniraju na isti ili sličan način. Većina njih komunicira sa senzorima pomoću kojih može otkriti želi li napadač ukrasti vozilo. Ukoliko alarmni sustav detektira maliciozni napad, postoji više opcija koje može poduzeti. Najčešće se upale truba i žmigavci, a kod malo naprednijih sustava može se desiti da se određene funkcije automobila onemoguće kako bi se spriječio napad.

1.3. Imobilizatori

Europska Unija je 1998. godine donijela regulativu kojom su imobilizatori postali obavezna oprema u automobilima koji se prodaju na području Europske Unije. Izvor [2] navodi kako je upravo ta regulativa u Nizozemskoj donijela ukupno smanjenje krađi automobila za 70%, dok je u Engleskoj ta ista regulativa donijela smanjenje od otprilike 80%.

Imobilizator je uređaj koji gasi motor ukoliko nije korišten odgovarajući ključ za njegovo pokretanje. Može raditi na dva načina. Prvi je da imobilizator nakon pokretanja isključi motor, dok mu drugi način totalno onemogućava pokretanje. Da bi imobilizator mogao obavljati svoju funkciju moraju postojati tri komponente koje zajedno surađuju, a to su imobilizator, brava i ključ, odnosno *transponder* u ključu. Kada se ključ stavi u bravu, zavojnica oko brave čita kod ključa sa *transpondera*, te taj podatak šalje u imobilizator. Ukoliko kod ključa ne odgovara

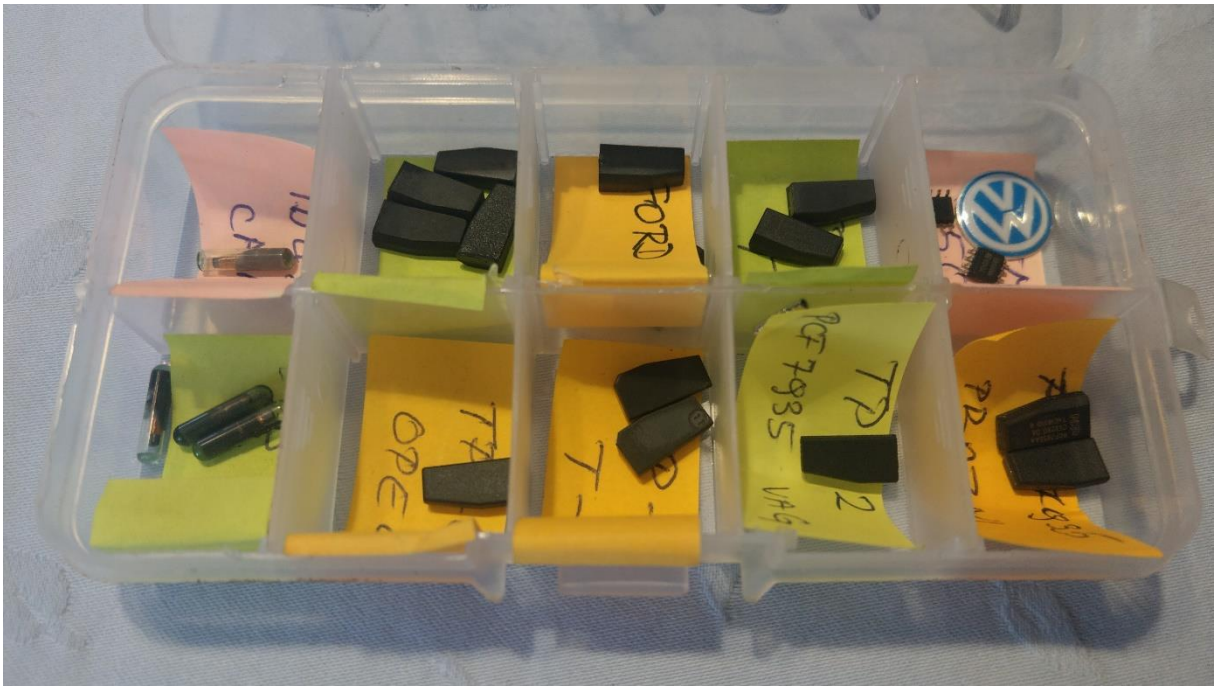
onom koji je zapisan u imobilizatoru, imobilizator šalje računalu motora signal da se ne koristi odgovarajući ključ, nakon čega računalo motora isti isključuje.

Slika 3: Imobilizator (zasebni, Golf 3)

Slika 4: Imobilizator (integrirani, gore: Audi, dolje: VW)

Slike iznad prikazuju dvije izvedbe imobilizatora. Na slici 3 su prikazani imobilizatori u odvojenim uređajima, dok se na slici 4 prikazuju nadzorne ploče automobila (eng. *Dashboard*) u koje je već integriran imobilizator. Kao što je već napomenuto, da bi imobilizator mogao obavljati svoju funkciju, brava mora pročitati kod zapisan u transponderu koji se nalazi u ključu.

Postoje svega tri vrste transpondera koje se koriste: stakleni, keramički i čip. Na slici ispod se mogu vidjeti sve te vrste.



Slika 5: Transponderi

1.4. Praćenje vozila

Noviji modeli automobila imaju ugrađen sustav za praćenje vozila. Takav sustav pomaže locirati vozilo u slučaju da je pomaknuto ili ukradeno. U tu svrhu se koriste ili GPS (eng. *Global Positioning System*) ili GLONASS (eng. *Global Navigation Satellite System*). Prema izvoru [1], postoje dvije vrste praćenja, a to su pasivno i aktivno praćenje. Pasivni sustavi funkcioniraju na način da se uključuje na neki poticaj. Taj poticaj može biti kretanje vozila ili neki drugi. Sustav povremeno sprema lokaciju, smjer i brzinu, te u određenim intervalima šalje te podatke u neki centralni sustav ili samom korisniku. Kada se vozilo vrati na predefiniranu lokaciju sustav se isključuje. Aktivni sustavi u stvarnom vremenu šalju te podatke tako da korisnik ili neki centralni sustav mogu u svakom trenutku znati gdje je vozilo te u kojem se smjeru kreće.

1.5. Fizičke mjere osiguranja nisu dovoljne

Ukoliko je maliciozni napadač neiskusan, gore navedene mjere ga mogu obeshrabriti od napada ili barem malo usporiti, no ukoliko je napadač iskusan i dovoljno tehnički potkovan,

te mjere mu nisu neka prijetnja. Naime sve se te mjere na veoma jednostavan način mogu zaobići.

Fizički pristup automobilu se može na jako jednostavan način postići razbijanjem bilo kojeg stakla, no ako napadač želi biti malo diskretniji i elegantniji, postoje drugi načini. Sigurnost baziranu samo na mehaničkom ključu se može lako zaobići običnim odvijačem i to tako da se prisili okretanje brave za pokretanje motora. Automobili koji imaju centralno daljinsko zaključavanje, no nekim slučajem nemaju ugrađen imobilizator malo su izazovniji, ali i dalje nije neki preveliki problem. Centralno daljinsko zaključavanje u većini slučajeva funkcionira na način da odašilje radio valove. Postoje uređaji koji mogu na daljinu presresti frekvenciju tih radio valova. Na taj način napadač dolazi u posjed frekvencije koja služi za otključavanje vozila te si lako može omogućiti fizički pristup. Kao što je već navedeno, pojava imobilizatora je malo otežala otuđivanje vozila zbog nemogućnosti paljenja motora, no to je sigurnosna mjera koja se aktivira tek nakon što napadač ostvari fizički pristup vozilu. Ona se u većini slučajeva može zaobići na veoma jednostavne načine. Na slici 3 su bili prikazani zasebni imobilizatori. Oni u sebi već imaju onemogućen kod ključa, što znači da napadač može zamijeniti taj uređaj te tako onemogućiti sigurnosnu mjeru. Ukoliko je imobilizator integriran u nadzornu ploču, proces je malo kompliciraniji, ali i dalje izvediv. Praćenje automobila može napadaču stvoriti probleme, pogotovo zato što je obično takav sustav spojen na sve glavne sustave u automobilu, no takav se sustav može zavarati slanjem modificiranih podataka tako da centralni sustav ili korisnik imaju pogrešne informacije o lokaciji.

2. Metode i tehnike rada

Kao što je u prijašnjem poglavlju napomenuto, u radu se radi o sigurnosti automobila, odnosno o sigurnosti računala ugrađenih u vozila. Kod analize sigurnosti bilo čega pa tako i automobila, potrebno je koristiti razne metode i razne alate. U ovom poglavlju će biti opisane metode i alati korišteni u ovom radu.

Kako bi se procijenile ranjivosti sustava ugrađenih u vozila u ovom će se radu koristiti DREAD metoda za procjenu rizika. Inicijalno je bilo zamišljeno da će se koristiti još i CVSS metoda, no nakon opsežnog istraživanja o njoj, zaključeno je da to ne bi donijelo mnogo koristi. Naime CVSS je veoma kompleksan i za razumjeti, a i za implementirati. Za razliku od DREAD-a, CVSS koristi skalu ocjenjivanja od 1 do 10, što dovodi do problema jer je teško načiniti razliku recimo između ocjena 5 i 6.

DREAD (eng. *Damage*, *Reproducibility*, *Exploitability*, *Affected users*, *Discoverability*), je metoda za procjenu rizika koja se oslanja na pet kriterija. Velika prednost ove DREAD-a je to što nijedan od kriterija nije usko vezan uz drugi. Upravo to je razlog što je ova metoda toliko često korištena pri procjeni rizika. Svaka otkrivena ranjivost se evaluira prema gore spomenutih pet kriterija, te se dodjeljuje ocjena od 1 do 3. Ukoliko je zbroj ocjena svih kriterija između 5 i 7, smatra se da je rizik malen, dok se zbroj ocjena između 12 i 15 smatra visokim rizikom. Najveća prednost ove metode procjene rizika je to što je jednostavna za implementaciju i procjenu. Najveći nedostatak ove metode je to što za neke primjene i neke profesionalce ona nije dovoljno detaljna. [3]

Kako bi se na praktičnom primjeru mogle iskoristiti neke od otkrivenih ranjivosti bilo je potrebno nabaviti alat koji će omogućiti komunikaciju između automobila i računala. Taj alat se zove „USB-CAN Plus“ od tvrtke VSCAN. To je uređaj za serijsku komunikaciju s automobilom. Radi na način da ga računalo registrira kao uređaj za mrežnu komunikaciju nakon čega se može koristiti kao običan mrežni uređaj poput WLAN kartice. Više informacija oko ovog uređaja se može doznati na web adresi: *http://www.vscom.de/usb-can-plus.html*.

Da bi gore navedeni uređaj mogao komunicirati s računalom, potrebno je imati odgovarajući upravljački program (eng. *Driver*). Upravljački su programi za uređaje koji komuniciraju CAN protokolom kod linux operacijskih sustava u većini slučajeva već ugrađeni u samu jezgru OS-a. Tako se *slcan,* upravljački program potreban za funkcioniranje spomenutog uređaja, nalazi u jezgri linux OS-a već od verzije 2.6.38.

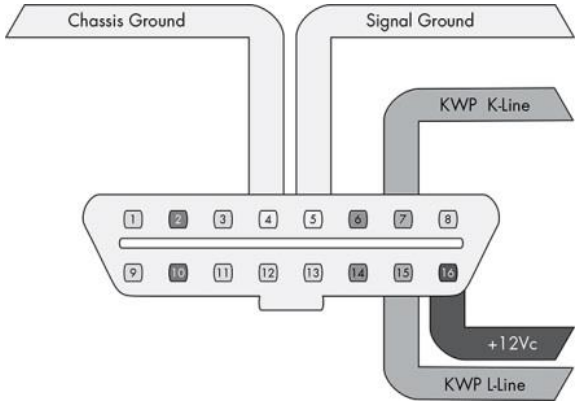
Uz sve gore navedene alate i metode, za grafičko sučelje programa i obradu podataka, korištena je *PyQt5* programska biblioteka u kombinaciji sa *Python* programskim jezikom.

3. Komunikacijski protokoli

Nekada su prijevozna sredstva imala ništa ili veoma malo elektronike, eventualno su imali električni sustav paljenja, te električna svjetla. Vremena su se kao i prijevozna sredstva drastično promijenila. Danas je sve električno odnosno elektroničko. U vozila se ugrađuje sve veći broj senzora, elektromotora i raznih uređaja koji moraju biti u mogućnosti međusobno komunicirati. Tu u igru ulaze komunikacijski protokoli. Bez njih bi ta komunikacija bila gotovo nemoguća. Upravo su saznanja o istima temelj za praktični primjer opisan u kasnijem dijelu rada.

3.1. ISO 9141-2

Ovaj protokol je varijacija KWP-a (eng. *Keyword Protocol*) koji se od 2003 godine ugrađuje u automobile Američke proizvodnje. ISO 9141-2 koji se ponekad još naziva i K-linija, najviše se ugrađivao u Europske automobile. K-linija koristi pin 7, te opcionalno pin 15 OBD-II konektora. Na slici ispod se može vidjeti raspored žica na spomenutom konektoru.



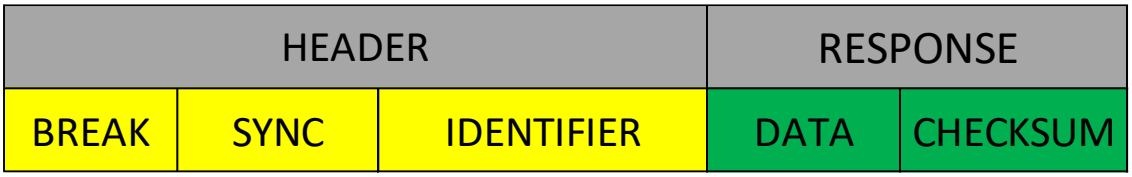
Slika 6: ISO 9141-2 pinout

Za ODB-II je bitno napomenuti da se plus i minus uvijek nalaze na istim pinovima. +12V je uvijek pin 16, dok je GND uvijek pin 4. Ovaj protokol za komunikaciju koristi UART (eng. *Universal Asynchronous receiver-transmitter*) protokol za prijenos podataka. UART protokol koristi startni bit, a može koristiti paritetni bit te stop bit. Najvažnija stvar vezana uz ovaj protokol je ta da svi paketi koji se šalju mrežom imaju svoje polazište i svoje odredište. To znači da uređaj koji šalje paket točno zna kojem uređaju šalje, te zapisuje svoju i adresu primatelja u paket. [1]

3.2. LIN protokol

Ovaj protokol se spominje samo zato što se u zadnje vrijeme vrlo često javljaju oglasi za posao koji zahtijevaju poznavanje istog. Također isti je slučaj i sa FlexRay protokolom o kojem će više govora biti u sljedećem poglavlju.

LIN funkcionira na *Master – Slave* principu. To znači da postoji jedan *Master* čvor u LIN mreži koji odašilje ostalim *Slave* čvorovima. LIN može podržati do 16 *Slave* čvorova. *Slave* čvorovi mogu vratiti odgovor *Master* čvoru, no nisu zamišljeni za takav način rada. S obzirom da je LIN napravljen kako bi komplementirao CAN protokolu, često se u vozilima može desiti da je spojen na CAN sabirnicu. Nadalje LIN funkcionira pri brzini od 20 Kbps na 12V. LIN protokol za komunikaciju koristi pakete, a njihov raspored se može vidjeti na slici ispod. [1]



Slika 7: LIN paket

Header se sastoji od tri polja. *Break* služi kako bi se svi čvorovi na mreži mogli pripremiti na zaprimanje ostalih podataka iz *Headera.* U *Sync* polju se nalazi standardizirana poruka heksadecimalnog zapisa *0x55* prema kojoj se *Slave* čvorovi na mreži prilagođavaju brzinu prijenosa. *Identifier* polje služi kako bi *Slave* čvorovi mogli odrediti koji od čvorova je za tu poruku odgovoran, te koji čvor mora na nju odgovoriti. *Response* je odgovor *Slave* čvora na poruku koju je *Master* čvor poslao. [4]

3.3. FlexRay

FlexRay je zamišljen kako bi mogao upravljati vremenski osjetljivim operacijama, a to mu omogućuje brzina od 10 Mbps na kojoj ovaj protokol funkcionira. Kao i *CAN* protokol, *FlexRay* koristi par uvijenih parica za komunikaciju između uređaja. Može koristiti topologiju sabirnice, te kao i CAN protokol mora na krajevima imati terminirajući otpor određenog iznosa. Uz topologiju sabirnice, ovaj protokol može koristiti i topologiju zvijezde u kojoj mora postojati centralni čvor koji komunicira sa svim ostalim čvorovima u mreži. Na taj način se može dobiti veća fleksibilnost mreže. Kao i LIN, ovaj se protokol spominje samo zato što se u zadnje vrijeme često javljaju oglasi za posao koji zahtijevaju njegovo poznavanje.

FlexRay se implementira na način da se uređajima prilikom proizvodnje odredi adresa, te da se cijeloj mreži odredi topologija kako bi mogli međusobno komunicirati. Takav način je veoma isplativ prilikom proizvodnje pošto je mnogo jednostavniji za implementaciju, a to

također znači da je testiranje veoma komplicirano pošto testni uređaj ne zna na koji je način mreža konfigurirana. [1]

Ovdje se neće navoditi daljnji detalji o ovom protokolu iz razloga što još uvijek nije u širokoj upotrebi. Prema trenutnim saznanjima ovaj se protokol ugrađuje samo u automobile vrlo visoke klase, što znači da je u velikoj manjini naspram CAN protokola koji će biti opisan u narednom dijelu.

3.4. CAN protokol

Svi do sad spomenuti protokoli ili polagano odlaze u povijest ili se pokušavaju probiti u automobilskoj industriji. CAN protokol je ostavljen za kraj jer se trenutno daleko najviše koristi u svim industrijama koje se bave proizvodnjom prijevoznih sredstava.

80-ih godina prošlog stoljeća u automobile se počelo ugrađivati sve više i više elektroničkih komponenti kao što su ECU, ABS, instrument-ploča (eng. *instrument cluster)*, razni senzori, te mnoge druge. Da bi te komponente radile kako treba, moraju nekako moći i komunicirati. Tu u igru ulazi CAN protokol.

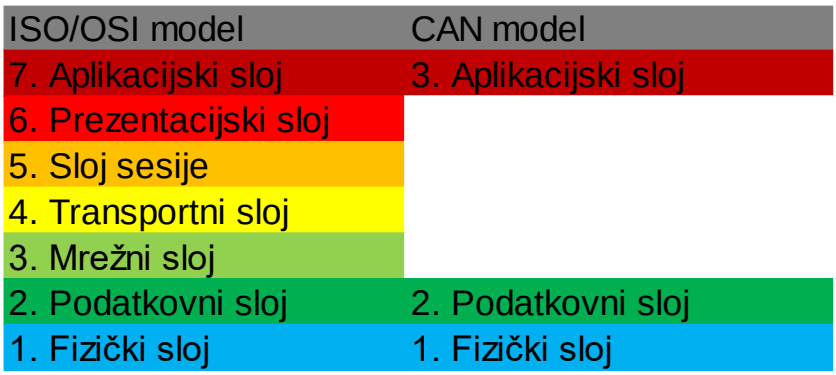
Uzmimo za primjer automobil koji ima običan motor sa unutarnjim izgaranjem. Recimo da je potrebno ići brže. To znači da bi u tom slučaju trebalo dodati gas. Onog trena kada se stisne papučica gasa, senzor registrira u kojem je papučica položaju, te tu informaciju šalje u ECU. Također ECU konstantno dobiva podatke od još nekoliko senzora kao što su senzor radilice (mjeri broj okretaja motora u minuti), senzor protoka zraka (mjeri protok zraka kroz motor), te senzor koji mjeri količinu ispušnih plinova. Postoji još senzora koji šalju podatke u ECU, ali za potrebe ovog primjera i ovo je dovoljno. Na temelju dobivenih podataka ECU određuje koji omjer goriva, zraka i ispušnih plinova treba motoru kako bi proizveo željenu snagu, a samim time i brzinu.

Primjer iznad je moguće implementirati i bez ikakve elektronike, a i sa bilo kojim drugim mrežnim protokolom, no CAN to uvelike olakšava. Upravo zbog toga je BOSCH i kreirao ovaj protokol. Naime CAN je *multi – master* protokol koji uvijek šalje pakete na sve čvorove u mreži maksimalne brzine od 1 mega bita u sekundi. Pa što sve ovo znači? U prošlosti su se PC računala bazirala na *Master – Slave* konfiguraciji, što je značilo da je jedan uređaj na sabirnici glavni (*master*) i ima prioritet nad ostalima. Kad master odašilje podatke, slave čeka. CAN funkcionira kao multi master, što znači da svi čvorovi u mreži mogu slati podatke istovremeno. U slučaju kada dva ili više čvorova pokušaju poslati poruke istovremeno, u nedestruktivnoj arbitraži pobjedi čvor koji šalje poruku s najvišim prioritetom (najmanji identifikator poruke), a ostali prekinu slanje i pokušaju poslati svoju poruku ponovno kada sabirnica postane slobodna [13, 14]. Isto tako, dok se drugim mrežnim tehnologijama mogu prenositi veliki paketi i velike količine podataka, CAN je kreiran na način da se može slati veliki broj malih paketa koji mogu

označavati bilo što od temperature vozila, položaja papučice gasa i još mnoge druge informacije. [5]

3.4.1. Mrežni model

Kao i gotovo svi mrežni protokoli, CAN je definiran pomoću ISO/OSI modela, no postoje neke razlike.



Slika 8: OSI i CAN

Na slici iznad se vide razlike i sličnosti između OSI i CAN modela mreže. Slojevi pod brojevima 1 i 2 su još poznati i kao niži slojevi OSI modela, dok se ostali smatraju višim slojevima. U CAN protokolu su niži slojevi standardizirani, dok više slojeve implementatori mogu izvesti i optimizirati kako im najviše odgovara.

Fizički sloj se brine o kreiranju bitova podataka koji se žele poslati od uređaja do uređaja i to na način da regulira razliku potencijala između *CAN H* i *CAN L* linija. Sastoji se od hardvera koji šalje i prima te bitove (*CAN TRANSCEIVER*) .

Podatkovni sloj zaprima bitove kreirane od fizičkog sloja, pakira ih u pakete, te ih pokušava poslati bez pojave pogreške. Nakon slanja paketa čeka odgovor od čvora koji je zaprimio taj paket.

Aplikacijski sloj se koristi kako bi se moglo pristupiti CAN mreži te upravljati njome. Tu spada recimo *SocketCAN* i drugi. [6]

3.4.2. Standard CAN, Extended CAN i vrste paketa

Kad god se spominje CAN protokol, odnosno paketi koji se odašilju preko CAN sabirnice, u većini se slučajeva misli na *Classic CAN. Classic CAN* može odašiljati do maksimalno 8 bajtova podataka u jednom paketu. Postoji još i *CAN FD (flexible data-rate)* koji može u jednom paketu prenijeti i do 64 bajta podataka, no BOSCH je ovaj tip protokola počeo razvijati tek 2011 godine, te još uvijek nije u toliko širokoj upotrebi. Upravo zbog toga ću se u ovom dijelu fokusirati na *Classic CAN.* Kod *Classic CAN*-a postoje dvije moguće

implementacije. Ona raširenija ima polje za identifikatore veliko 11 bitova, dok ona malo novija implementacija ima to isto polje veličine 29 bitova. [7]

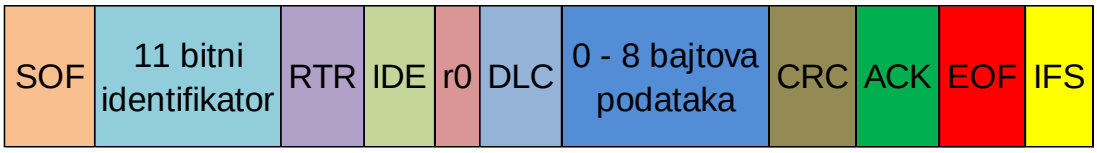
Komunikacija CAN protokolom se može svesti na jednu relativno veliku skraćenicu, a to je CSMA/CD+AMP:

- CSMA (*carrier-sense, multiple-access)* – svaki čvor u mreži mora čekati određeni broj perioda prije no što može slati bilo kakve pakete.
- CD+AMP (*collision detection + arbitration on message priority)* – kolizije paketa se rješavaju pomoću identifikatora. Što je identifikator manji, to je većeg prioriteta.

U nastavku su prikazane razlike između standardnog i proširenog CAN paketa, no prvo treba neke stvari razjasniti. Na početku valja spomenuti da kad se radi o CAN sabirnici, postoje dva moguća stanja, a to su dominantno i recesivno, što omogućuje nedestruktivnu arbitražu i brzo potvrđivanje okvira. [13, 14] U tipičnoj CAN mreži koriste se samo dvije linije, a to su *CAN H(High)* i *CAN L(Low)*. Obje linije imaju nominalni napon od 2.5V. Kada se odašilju podaci preko CAN sabirnice, napon na ovim linijama varira. Tako se *CAN H* može nalaziti na 2.5V ili 3.5V, dok se *CAN L* može nalaziti na 2.5V i 1.5V. Ukoliko jedna od ovih linija promijeni svoj napon, mijenja se i druga. Promjena potencijala između ovih linija označava logičku nulu ili logičku jedinicu. U CAN protokolu sabirnica može biti u dominantnom ili recesivnom stanju. Tu do izražaja dolazi sve što je do sada pisano o naponima. Ukoliko je razlika potencijala između dvije CAN linije 0V, tada se radi o recesivnom bitu (logičko 1), a ukoliko je razlika potencijala 2V, tada se radi o dominantnom bitu (logičko 0).

3.4.2.1. Standard CAN

Slika ispod označava standardni CAN paket koji se najčešće koristi u automobilskoj industriji. Ispod slike su objašnjena sva polja unutar jednog standardnog paketa.



Slika 9: Standard CAN paket

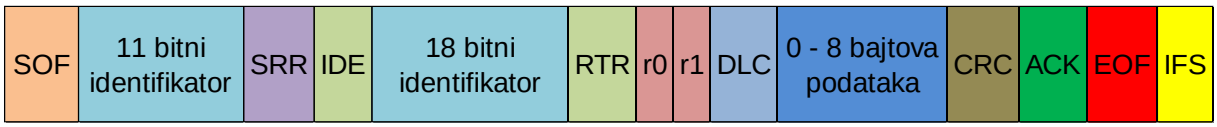
- *SOF* – dominantan bit koji označava početak paketa. Također sinkronizira sve čvorove na mreži
- Identifikator – u standardnom CAN protokolu ovaj identifikator može biti duljine 11 bitova i označava prioritet paketa. Što je binarna vrijednost manja, to je prioritet veći.
- *RTR* (*Remote Transmission Request)* – dominantan u slučaju da je potreban podatak od nekog čvora u mreži. Svi čvorovi na mreži primaju ovaj paket. Prema identifikatoru

se određuje na koga se odnosi. Kad neki čvor i odgovori, svi ostali primaju odgovor te taj isti može iskoristiti bilo koji drugi čvor.

- *IDE* – dominantan bit ovdje znači da se radi o standardnom paketu (kasnije će biti jasnije čemu ovaj bit služi)
- r0 – bit koji trenutno ne služi ničemu, već je rezerviran za buduće promjene i implementacije
- *DLC* – polje veličine 4 bita koje govori o tome koliko bajtova podataka se prenosi
- *Data* (podaci) – do 64 bita podataka koji se prenose u paketu
- *CRC* – 16 bitno polje koje sadrži kontrolni zbroj prenesenih podataka i služi za detekciju grešaka. Sastoji se od 15 bitova kontrolnog zbroja, te jednog bita koji sliži kao *delimiter*
- *ACK* – polje koje se sastoji od dva bita. Jedan kao potvrdu zaprimanja, a drugi kao *delimiter*. Kada čvor zaprimi ispravan paket, u bit za potvrdu zaprimanja zapiše dominantnu vrijednost kako bi čvor koji šalje znao da je poruka uspješno primljena i ispravna. Kada se to ne bi desilo, čvor koji je poslao originalnu poruku bi istu poslao ponovo.
- *EOF* – sedam bitno polje koje označava kraj paketa
- *IFS* – sedam bitno polje koje označava vrijeme potrebno kontroleru da bi premjestio ispravan paket u svoj međuspremnik [5]

3.4.2.2. Extended CAN

Uz standardni CAN paket postoji i prošireni. Potreba za ovom vrstom paketa se pojavila kako se sve više i više uređaja spajalo na CAN sabirnicu. Naime standardni paket može imati svega 2048 zasebnih identifikatora , dok prošireni može podržati do 537 milijuna identifikatora. Na taj način se može u mrežu uključiti mnogo više uređaja. Na slici ispod može se vidjeti format proširenog CAN paketa kao i razlike u odnosu na standardni paket.



Slika 10: Prošireni CAN paket

Prošireni paket ima samo četiri bitne razlike, a to su:

- *SRR* (*Substitute Remote Request*)– mijenja RTR bit na tome mjestu
- *IDE* – ukoliko je u ovom polju recesivan bit, radi se o proširenom CAN paketu
- 18 bitni identifikator – nastavak na 11 bitni identifikator
- r1 – služi istoj svrsi kao i r0 [5]

4. Analiza sigurnosti vozila (VW GOLF VI 2011)

Kada se neki sustav razvija, veoma je poželjno napraviti detaljnu analizu sigurnosti istog kako bi se otkrile prijetnje i rizici, te kako bi se pravovremeno uspjelo te prijetnje i rizike spriječiti.

Potencijalni napadači također za određeni sustav rade analizu sigurnosti, no iz potpuno suprotnih razloga. Dok je kod *developera* cilj spriječiti prijetnje i smanjiti rizike, napadaču je cilj naći vektor napada kod kojeg uz minimalan napor može doći do maksimalnih rezultata.

Kod analize sigurnosti postoje dva glavna koraka, a to su modeliranje prijetnji i procjena rizika. Oba koraka idu ruku pod ruku i jedan bez drugog nemaju smisla. S obzirom da su informacije dobivene u ovom poglavlju biti veoma bitne za poglavlje 5, ovu je analiza napravljena na temelju praktičnog primjera. Vozilo na kojem se analiza temelji je VW Golf 2.0 TDI iz 2011. godine, u daljnjem tekstu meta.

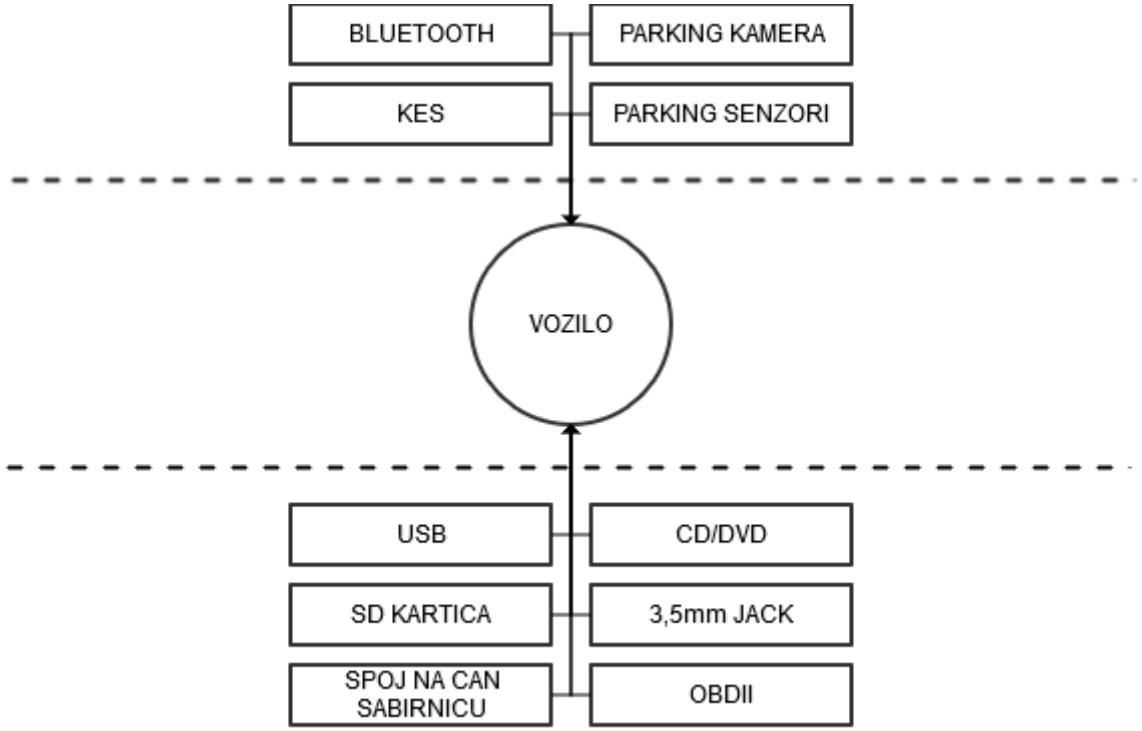
4.1. Modeliranje prijetnji

Kod modeliranja prijetnji za pojedini sustav, potrebno je staviti se u poziciju potencijalnog malicioznog napadača i pokušati odrediti što je moguće više vektora napada na sustav. U početnom stadiju kreiranja modela prijetnji, proučava se arhitektura vozila, te se dokumentiraju svi načini na koje signal može ući u vozilo. Nakon toga se kreiraju dijagrami koji na zoran način prikazuju kako moduli unutar vozila međusobno komuniciraju. Ta komunikacija je veoma bitna kako bi kasnije mogli procijeniti rizik od pojedinog napada. Dijagrami su poredani po razinama, te je svaka razina detaljnija od prethodne. U ovom slučaju ići će se do druge razine dijagrama, s obzirom da je u kasnijem dijelu fokus na samo jednoj prijetnji. Ulazi signala koji su identificirani na meti su sljedeći:

- KES (*Keyless Entry System)*
- Parking senzori
- Parking kamera
- Bluetooth (u radiju/navigaciji)
- Bluetooth (originalni modul)
- OBDII port
- Direktno spajanje na CAN sabirnicu (preko žica)
- Radio/Navigacija
- USB
- 3.5 mm audio konektor
- CD/DVD
- SD kartica

4.1.1.Razina 0

Na ovoj razini se meta gleda sa ptičje perspektive. Pokušava se nacrtati dijagram koji prikazuje kojim kanalima signal može ući u vozilo. Ti kanali se mogu nalaziti unutar ili izvan vozila, no najzornije će to prikazati dijagram ispod.

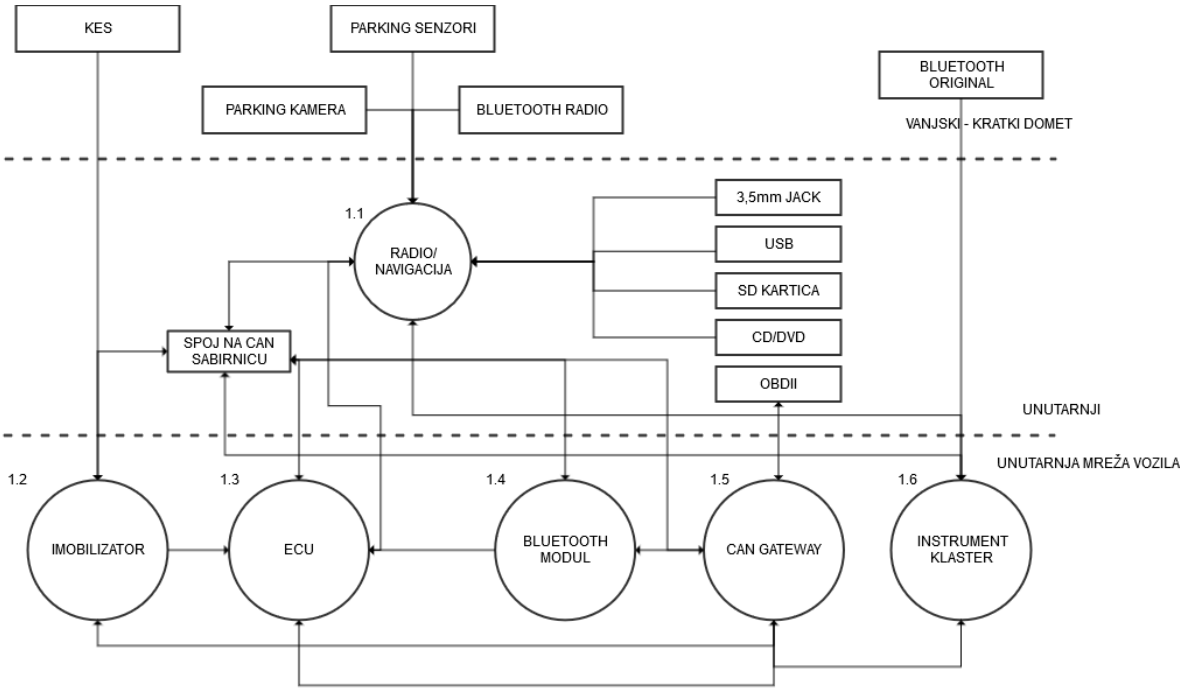


Slika 11: Razina 0 modela prijetnje

Pravokutnici na dijagramu se mogu zamisliti kao prijetnje, odnosno načine na koje napadač može pokušati naštetiti vozilu. To su ulazne točke podataka u vozilo, a kako bi ti podaci došli do vozila, moraju prijeći isprekidane crte. Te crte označavaju unutarnju i vanjsku zonu prijetnji. Unutarnja zona sadrži sve ulazne točke za koje napadač mora imati fizički pristup vozilu (unutrašnjost, pristup žicama itd.), dok kod vanjske zone napadač ne mora biti u neposrednoj blizini ili unutrašnjosti vozila. Krug u centru dijagrama reprezentira složeni proces koji se može razdijeliti na manje složene.

4.1.2.Razina 1

Cilj crtanja ovih dijagrama je doći do relativno niske razine apstrakcije kako bi kod analize rizika mogli procijeniti kojim se prijetnjama moramo posvetiti, a koje možemo ostaviti jer ne predstavljaju veliki rizik. Na sljedećem dijagramu se može vidjeti razrada procesa 1.0 sa prethodnog dijagrama.



Slika 12: Razina 1

Razrada na daljnje pod procese je veoma bitna kod planiranja i razvoja vozila, odnosno sustava koji će se kad tad ugraditi u vozilo, kako bi se u detalje utvrdile sigurnosne prijetnje i rizici. Tako bi se svaki proces označen sa X.X mogao dalje razlomiti na još manje cjeline, no za potrebe ovog rada mislim da je i ovo dovoljno.

4.1.3. Identifikacija prijetnji

Prilikom identifikacije prijetnji, bitno je popisati što je više moguće prijetnji. Također je bitno navesti sve prijetnje bez obzira što postoji mogućnost da su u kasnijim razradama te iste možda neutralizirane.

Prvo se pokušavaju identificirati prijetnje s pogledom na nultu razinu. Potrebno je imati na umu da prijetnje koje su ovdje navedene, moraju biti relativno općenite, s obzirom da na nultoj razini još ni jedan proces nije razrađen, odnosno nije razlomljen na primatelje i pošiljatelje signala. [8]

Gledajući nultu razinu, maliciozni napadač bi mogao:

- Prisluškivati razgovore putnika
- Preuzeti kontrolu nad vozilom
- Onemogućiti vozilo
- Sabotirati vozilo (sigurnosni sustavi, kočioni sustavi, ...)
- Pratiti vozilo
- Podmetnuti maliciozni kod

Ovo su samo neke od prijetnji nulte razine. U praksi bi se ovime trebao baviti cijeli tim ljudi zaduženih za sigurnost. Nakon generalnih prijetnji nulte razine, prebacujemo fokus na prijetnje prve razine. Kod prve razine potrebno je zagrebati malo dublje ispod površine, odnosno prebaciti fokus na sve načine na koje maliciozni napadač može pristupiti bilo kojem sustavu vozila. To znači da će se ovdje pokušati identificirati prijetnje vezane uz moguće točke upada koje su na slici 12 prikazane kao pravokutnici. Izuzetak od toga je RADIO/NAVIGACIJA kojeg se ujedno može smatrati složenim procesom kao i točkom upada.

Maliciozni napadač bi mogao iskoristiti KES na način da:

- Presretne signal kod otključavanja/zaključavanja
- Klonira ključ/daljinski (potencijalni alati: *zedd bull, galetto*, ...)
- Ometa regularan signal
- Probije algoritam za generiranje ključa za otključavanje/zaključavanje
- Pošalje modificirani signal kako bi imobilizator ušao u neočekivano stanje

Maliciozni napadač bi mogao iskoristiti parking kameru na način da:

- Umetne maliciozni kod preko slike/videa
- Se spoji direktno na žice i pristupi radiju/navigaciji
- Uništi kameru

Maliciozni napadač bi mogao iskoristiti parking senzore na način da:

- Se spoji direktno na žice i pristupi radiju/navigaciji
- Uništi senzore
- Modificira signal tako da vozilo misli da nije blizu objekta

Maliciozni napadač bi mogao iskoristiti bluetooth na način da:

- Se ubaci između vozila i korisnika
- Prisluškuje korisnike vozila
- Dođe do osjetljivih informacija o korisniku
- Preuzme kontrolu nad vozilom
- Promijeni parametre vozila
- Promijeni parametre radija/navigacije

S obzirom da su 3,5mm JACK, USB, SD KARTICA, te CD/DVD relativno srodni, prijetnje koje dolaze od njih su također slične.
Maliciozni napadač bi mogao iskoristiti ove ulaze na način da:

- Umetne maliciozni kod u radio/navigaciju
- Instalira modificiranu nadogradnju sustava radija/navigacije
- Ostvari kratki spoj i time uništi radio/navigaciju
- Dođe do osjetljivih informacija (kretnje korisnika itd.)
- Zaobiđe sigurnost i ostvari pristup do interne CAN mreže
- Korisniku prikazuje pogrešne/modificirane podatke
- Podmetne maliciozni uređaj sa kojim može sam sebi slati osjetljive podatke

Maliciozni napadač bi mogao iskoristiti CAN mrežu (ODBII, spoj na žicu)

- Se spoji direktno na bilo koju CAN sabirnicu (Multimedija, Motor, Udobnost, ...)
- Očita dijagnostičke podatke vozila
- Podmetne maliciozni kod koji omogućava udaljenu komunikaciju s vozilom
- Preuzme kontrolu nad određenim sustavom u vozilu ili cijelim vozilom
- Promijeni parametre vozila
- Ošteti određene komponente u vozilu
- Ostvari kratki spoj i onemogući komunikaciju
- Zamijeni originalnu komponentu sa modificiranom
- Preuzme kontrolu nad vozilom usred vožnje

Sad kada je identificirana većina mogućih prijetnji, potrebno je te prijetnje ocijeniti kako bi saznali koje od njih su najopasnije. U tome nam može pomoći nekoliko metoda za procjenu rizika promatranih prijetnji. U automobilskoj industriji se veoma često koriste metode ISO26262 ASIL, te MIL-STD-882E, no u ovom su nam scenariju one gotovo beskorisne, s obzirom da se koriste za ocjenjivanje fizičke sigurnosti vozila, te putnika istog. Upravo iz tog razloga je odlučeno da će se koristiti DREAD metoda. [8]

4.1.4. Procjena rizika (DREAD)

Kao što je već napomenuto, za procjenu rizika će se koristiti metodu DREAD. Ova se metoda inače koristi za procjenu rizika prijetnji web aplikacija, no može se primijeniti i kod vozila. DREAD nije ništa drugo no akronim sastavljen od prvih slova svake kategorije koje se ocjenjuju. Te kategorije su:

- Potencijalna šteta (***D****amage potential*) – kolika je potencijalna šteta ukoliko napadač uspije iskoristiti ranjivost
- Mogućnost ponavljanja (***R****eproducibility*) – Koliko je jednostavno/složeno ponovo proizvesti napad
- Iskoristivost (***E****xploitability*) – Koliko je jednostavno iskoristiti pojedinu ranjivost

- Zahvaćeni korisnici (***A**ffected users*) – Koliko korisnika može biti zahvaćeno iskorištavanjem pojedine ranjivosti
- Mogućnost otkrivanja (***D**iscoverability*) – Koliko je jednostavno/složeno otkriti pojedinu ranjivost

DREAD se koristi na način da se prijetnje ocjenjuju na skali od 1 do 3 za svaku kategoriju. Na kraju se za svaku prijetnju zbrajaju ocjene, te se dobiva ukupna ocjena između 5 i 15. Ukoliko je ukupna ocjena između 5 i 7, smatra se da je ranjivost niskog rizika. Ranjivosti s ocjenama između 8 i 11 su srednjeg rizika, dok su one s ocjenama između 12 i 15 visokog rizika. Tablica ispod malo zornije prikazuje način vrednovanja pojedine kategorije.[9]

KATEGORIJA	RIZIK		
	VISOK(3)	SREDNJI(2)	NIZAK(1)
D	Potpuna kontrola nad vozilom	Pristup osjetljivim informacijama	Pristup trivijalnim informacijama
R	Napad je uvijek moguće realizirati	Napad je moguće realizirati samo u specifičnim uvjetima	Čak i uz detaljne informacije o ranjivosti, napad je veoma teško realizirati
E	Napadač bi mogao biti laik	Napadač bi trebao biti vješt	Napadač bi trebao biti veoma vješt sa specifičnim setom znanja i vještina
A	Zahvaća sve podsustave vozila	Zahvaća neke sustave vozila	Zahvaća nekritične sustave vozila preko kojih ne može počiniti veliku štetu
D	Ranjivost je jednostavno otkriti	Napadač bi trebao biti veoma kreativan kako bi otkrio ranjivost	Ranjivost je toliko dobro sakrivena da je vrlo mala šansa da bude otkrivena

Slika 13: Ocjenjivanje prema DREAD metodi

Sad kada je napravljen popis mogućih prijetnji i DREAD metoda je malo više poznata, ocjenjivanje pojedine prijetnje može započeti. Cilj ovog rada je analizirati sigurnost vozila, te pokušati iskoristiti neku od ranjivosti. U sljedećem poglavlju je iskorištena jedna od ranjivosti CAN sabirnice. Iz tog razloga će se ocijeniti samo prijetnje vezane uz CAN. Ocjenjivanje svih ostalih prijetnji bi se odradilo na potpuno identičan način.

CAN PRIJETNJE	D	R	E	A	D	UKUPNO	RIZIK
Direktan spoj na sabirnicu	3	3	2	3	3	14	VISOK
Očitavanje dijagnostičkih podataka	2	3	2	2	3	12	VISOK
Podmetanje malicioznog koda	3	3	1	2	1	10	SREDNJI
Preuzimanje kontrole	3	1	1	3	1	9	SREDNJI
Promjena parametara vozila	3	3	2	2	2	12	VISOK
Šteta na komponentama	3	2	2	2	2	11	SREDNJI
Nemogućnost komunikacije kratkim spojem	2	3	2	2	3	12	VISOK
Zamjena komponente modificiranom	3	2	1	3	2	11	SREDNJI
Kontrola vozila usred vožnje	3	1	1	3	1	9	SREDNJI

Slika 14: Analiza rizika prijetnji CAN sabirnice

Kao što se iz tablice iznad može vidjeti, uz pomoć DREAD metode, direktan spoj na CAN sabirnicu je procijenjen kao najopasnija prijetnja od svih identificiranih. Zato će se u sljedećem dijelu pokušati iskoristiti ta ranjivost. Naravno da sve ostale prijetnje nisu bezazlene, no ovaj rad bi bio malo prevelik ukoliko bi se pokušale iskoristiti sve ove ranjivosti, tim više što su neke od njih previše kompleksne.

5. Realizacija napada

U ranijemu dijelu ovog rada je već natuknuto da će se pokušati iskoristiti ranjivost vozila na način da se direktno spoji na jednu od CAN sabirnica unutar istog. Vozilo koje će poslužiti kao primjer služi za svakodnevnu upotrebu, stoga je bio potreban veliki oprez kako se ne bi oštetilo. Radi se o automobilu Volkswagen Golf 6 iz 2011 godine.

Tokom faze planiranja ovog rada, utvrđena je potreba za nekim sučeljem preko kojeg bi se mogla ostvariti komunikacija s vozilom. U tom je trenu na raspolaganju bilo nekoliko njih koji su bili potencijalni kandidati. Ross-Tech VCDS i ELM327 su bili samo od nekih, no ubrzo je primijećeno da oni baš i neće odgovarati. Naime većina tih sučelja su dijagnostička, te nisu u stanju ostvariti *full-duplex* vezu. To znači da u danom trenutku takvo sučelje može biti samo pošiljatelj ili samo primatelj. Jedan od glavnih uvjeta za odabir sučelja je bila *out-of-the-box* podrška od strane linux operacijskog sustava, te jednostavnost inicijalne konfiguracije putem *python* programskog jezika. Na kraju je odabrano SLCAN sučelje pod nazivom Vscom USB-CAN Plus. Razloga za odabir baš tog sučelja je bilo mnogo, a neki od glavnih su:

- Industrijski dizajn i robusnost
- Spajanje na računalo putem USB-a
- Cijena
- Podrška za CAN mrežu do 1Mbit/s
- Vizualni pokazatelj postoje li greške prilikom prijenosa paketa
- Podrška za standardni SLCAN *driver* (upravljački program) koji se nalazi u jezgri linux operacijskog sustava
- Podrška u *python-can* biblioteci *Python* programskog jezika

Sučelje je bilo odabrano i sve je teklo po planu, no naravno pojavio se novi problem. Tvrtka koja proizvodi to sučelje posluje samo sa drugim tvrtkama (B2B). Na svu sreću prodaja i dostava su nakon nekog vremena uspješno dogovoreni, te je sučelje uskoro stiglo. Još i prije dolaska sučelja započeto je opsežno istraživanje vezano uz najbolju moguću iskoristivost. Prvi pokušaji ulaska u cijelu tematiku CAN protokola su se sastojali od doslovno dva agenta (pošiljatelj i primatelj) pokrenutih u linux terminalu i povezanih sa sučeljem. Nakon nekoliko neuspjelih pokušaja, i malih promjena kod poziva sučelja, stigli su prvi pozitivni rezultati. Paketi su se uredno slali i zaprimali preko sučelja, te je sve izgledalo uredno, no pravi test je tek slijedio. Plan je bio povezati se s vozilom preko OBDII dijagnostičkog porta, no pojavila se prepreka. Svi raspoloživi OBDII kablovi su s druge strane imali USB priključak, dok se na sučelju nalazio RS232. Tu su postojale svega dvije opcije. Naručivanje novog kabla koji bi prouzročio čekanje i daljnje kašnjenje ili izrada vlastitog kabla. Odlučeno je da je da će se

izraditi novi kabel, no sa jednim dodatkom. Taj dodatak su tri žice sa krokodil štipaljkama koje se mogu direktno spojiti na CAN sabirnicu (slika ispod). Upravo se taj dodatak kasnije pokazao kao ključan.

Slika 15: OBDII kabel za direktno spajanje na CAN sabirnicu

Tokom tih inicijalnih testiranja, na raspolaganju su bila dva automobila: VW Golf 2000, te Hyundai Accent 2006. S obzirom da je elektronika VW-a puno više dokumentirana (VW je bio taj koji je u linux jezgru ugradio podršku za CAN protokol), odlučeno je da će se na njemu izvršiti inicijalno testiranje. Ubrzo nakon početka testiranja, to se pokazalo kao pogrešna odluka. Test je proveden, no rezultati su bili razočaravajući. Naime VW 2000-te godine još uvijek nije u potpunosti integrirao CAN protokol u svoja vozila, već samo za neke specifične dijelove do kojih nije baš jednostavno doći, stoga je pažnja preusmjerena na Hyundai. Drugi krug testiranja, ovaj put sa novijim vozilom, je prošao u redu, nakon čega je mogao započeti razvoj aplikacije za iskorištavanje ranjivosti.

Aplikacija je razvijena primarno za linux operacijske sustave, a pisana je u programskom jeziku python verzije 3. O ovom programskom jeziku nema potrebe pisati previše toga, jer svatko tko ima imalo doticaja sa informatikom zna za njega i zna čemu služi. Ono što je manje poznato i što zaslužuje malo više pažnje su dodatni moduli koji su ovdje korišteni.

- *PyQt5* – modul koji sadrži skup poveznica (*eng. Bindings)* za Qt, koji je ustvari C++ programska biblioteka za razvoj modernih mobilnih i desktop aplikacija [10]
- *python-can* – modul koji pythonu donosi podršku za CAN. Omogućava povezivanje različitih sučelja, te olakšava komunikaciju s CAN mrežom [11]

- *pySerial* modul koji omogućuje i olakšava otkrivanje i komunikaciju sa serijskim portovima kao što je USB [12]

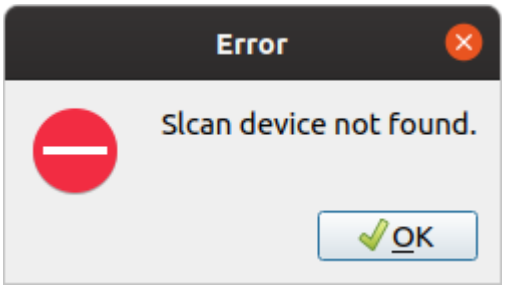
Tokom izrade i testiranja aplikacije, nabavljen je novi automobil koji za komunikaciju između velike većine svojih podsustava koristi CAN, a to je upravo VW Golf 6 iz uvoda ovog poglavlja.

5.1. CANSniffer

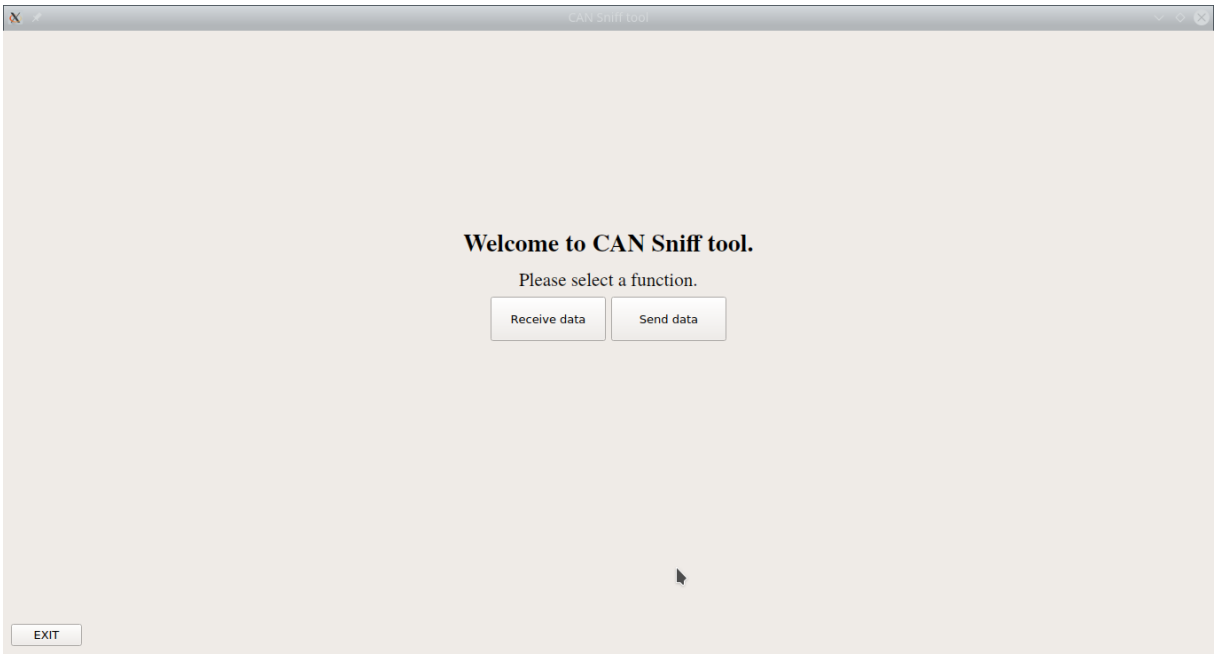
Za aplikaciju je odabran naziv CANSniffer. Možda on i nije baš idealan, s obzirom da se pomoću nje može i primati, a i slati pakete, no što je, tu je. U narednom dijelu je malo zornije prikazano na koji način aplikacija funkcionira, a na kraju rada je kompletan programski kod aplikacije u slučaju da je netko želi koristiti, nadograditi ili promijeniti.

Ranije je već rečeno da je aplikacija rađena isključivo za linux operacijske sustave, a sve oznake unutar nje su pisane engleskim jezikom kako bi bila pristupačnija širem spektru korisnika. Python-can koristi UDP mrežni protokol kako bi komunicirao sa vozilo, a s obzirom da se sama aplikacija pokreće u linux okruženju, to znači da ju je potrebno pokrenuti u administratorskom načinu (*sudo*).

Nakon pokretanja programa naredbom *sudo python3 putanja_do_programa/main.py* (umjesto *putanja_do_programa/main.py* može pisati *main.py* samo u slučaju prethodnog pozicioniranja u mapu projekta gdje se nalazi datoteka *main.py*), izvršava se provjera je li u USB *port* priključen odgovarajući uređaj pomoću *pySerial* modula. U ovom slučaju se radi o prethodno navedenom *USB-CAN Plus* uređaju. Ukoliko nijedan takav nije pronađen, na ekran se izbacuje poruka sa slike 16, dok se u protivnom otvara početni ekran aplikacije. On je prikazan na slici 17.



Slika 16: Slcan uređaj nije pronađen



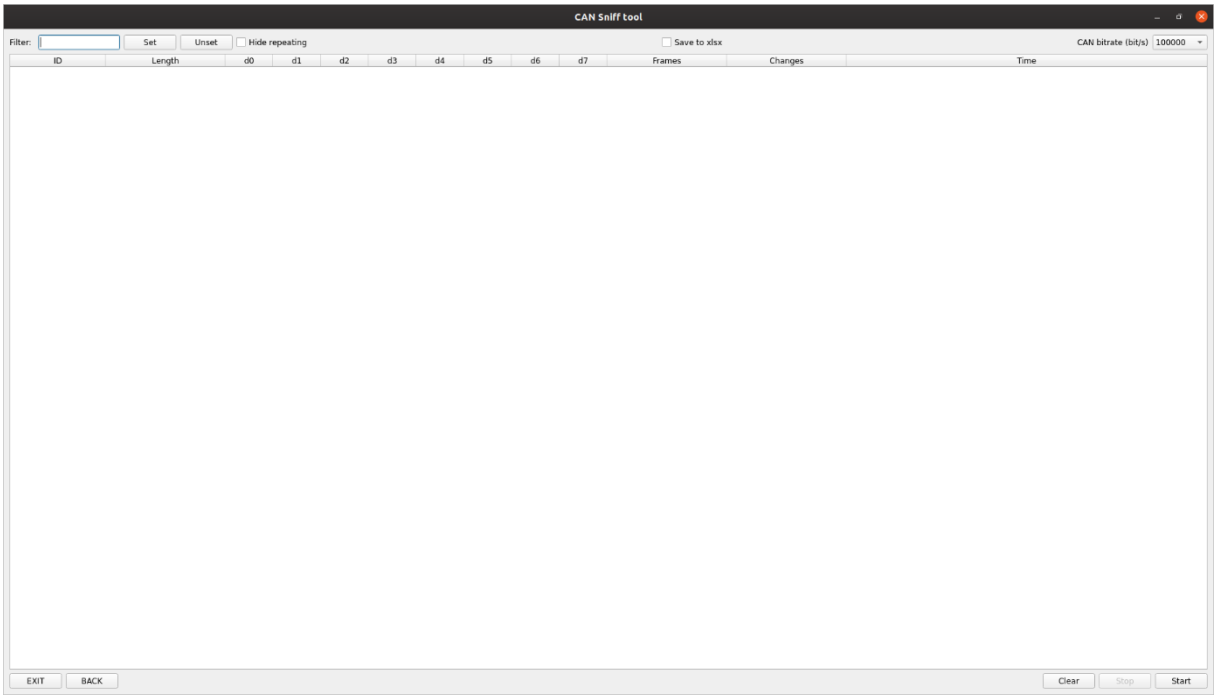
Slika 17: Početni zaslon aplikacije

Već je spomenuto da se za grafički dio aplikacije koristi PyQt modul kako bi se osigurala neka osnovna razina korisničkog iskustva. Upravo zbog toga se prije prikazivanja početnog zaslona, u datoteci *MainWindow.py,* inicijalizira „okvir“ kojeg koriste svi ekrani aplikacije, pa tako i ovaj početni. To znači da se „okvir“ proglasi *QStackecWidget*-om na unutar kojeg se ekrani otvaraju, odnosno mijenjaju. Zamjena ekrana se obavlja na način da se u „okvir“ doda novi (sljedeći), dok se onaj stari (prethodni) obriše.

Svaki od ekrana sadrži određene *PyQt* elemente (*widgets*) kako bi se omogućila manipulacija akcijama ili omogućio pregled. Elementi korišteni u ovoj aplikaciji su:

- Oznaka (*Label*) [*QLabel*] – Slovčana oznaka.
- Potvrdni okvir (Checkbox) [QCheckBox] – Polje u koje se klikom stavlja kvačica, te može biti u jednom od dva stanja. Označeno ili neoznačeno.
- Unosno polje (Input) [QLineEdit] – Jedno linijsko polje za unos teksta.
- Padajući izbornik (Combo Box) [QComboBox] – Padajući izbornik koji dopušta odabir samo jedne vrijednosti.
- Gumb (Button) [QButton] – Običan gumb.
- Tablica (Table) [QTableWidget] – Tablica sa retcima i stupcima koja može prikazivati razne vrijednosti.
- Lista (List) [QListWidget] – Lista vrijednosti. U trenutnom programu je omogućen odabir samo jedne vrijednosti.
- Tekstualno polje (Text Area) [QPlainTextEdit] – Više linijsko polje za unos teksta.

Aplikacija je predviđena za dva načina rada, a to su zaprimanje i slanje CAN paketa. Klikom na gumb *Receive data*, otvara se sučelje za zaprimanje CAN paketa koje se može vidjeti na sljedećoj slici.



Slika 18: Prazno sučelje za zaprimanje paketa

Glavni dio sučelja je tablica sa 13 kolona koje redom označavaju:

- *ID* – Identifikator prioriteta paketa. Može se smatrati i identifikatorom pošiljatelja
- *Length* – Veličina CAN paketa u bajtovima (0 – 8)
- d0 – d7 – Individualni bajtovi podataka
- *Frames* – Broj zaprimljenih paketa
- *Changes* – Koliko puta se desila promjena u paketima
- *Time* – Vrijeme kad je zaprimljen posljednji paket pod određenim *ID*-em, iskazan u obliku Unix vremena

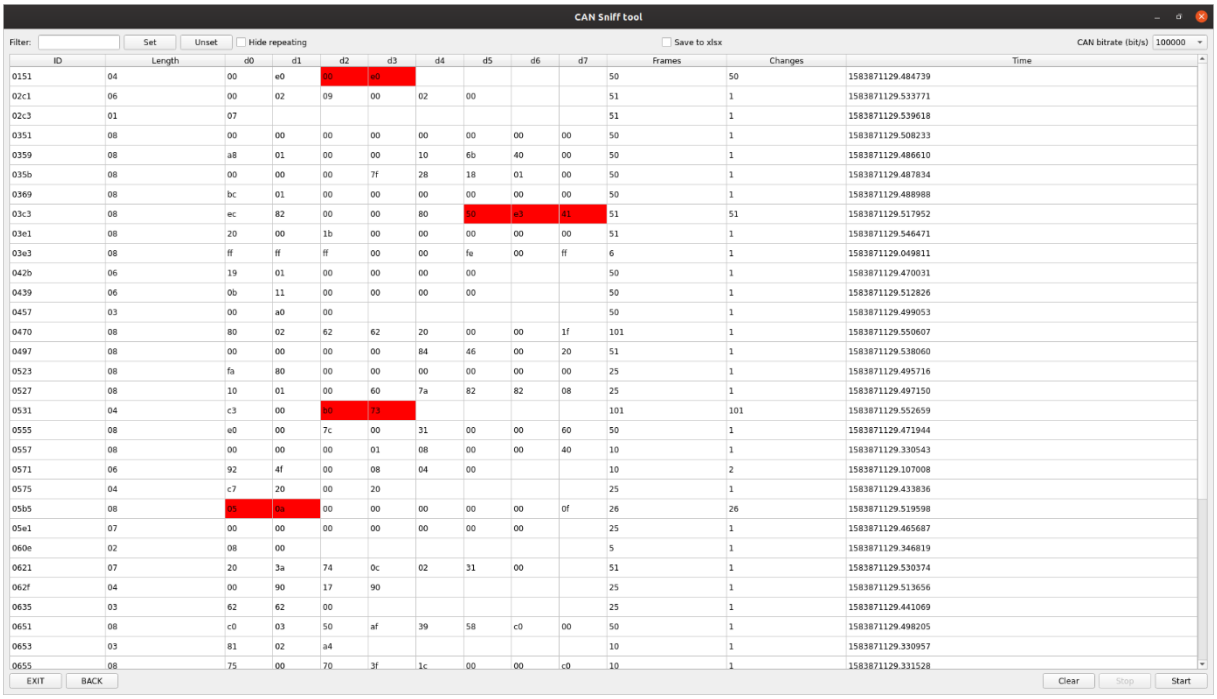
U redu iznad tablice se redom mogu naći akcije:

- *Filter* – Služi za filtriranje paketa unosom *ID*-a. Ukoliko se želi prikazati više paketa, potrebno je *ID*-eve odvojiti zarezom
- *Set* – Primjenjuje filter
- *Unset* – Poništava filter
- *Hide repeating* – ukoliko je označeno, sakriva pakete koji su se ponovili 10 ili više puta
- *Save to xlsx* – ukoliko je označeno, svi paketi koji pristignu se upisuju u xlsx datoteku. Svako ponovno pokretanje procesa zaprimanja kreira novu datoteku.
- *CAN bitrate* – U vozilima postoje različite CAN sabirnice koje funkcioniraju na različitim brzinama. Ovaj padajući izbornik služi za postavljanje te brzine

Redak ispod tablice sadrži samo gumbe koji izvode određene akcije:

- *Exit* – Izlaz iz aplikacije
- *Back* – povratak na početni zaslon
- *Clear* – Briše sadržaj tablice, ali samo ukoliko zaprimanje paketa nije u tijeku
- *Stop* – Zaustavlja proces zaprimanja paketa ukoliko zaprimanje nije u tijeku
- *Start* – Započinje proces zaprimanja paketa. Onemogućen ukoliko je zaprimanje traje

Na slici ispod se može vidjeti izgled ekrana nakon što se pokrene proces zaprimanja paketa.

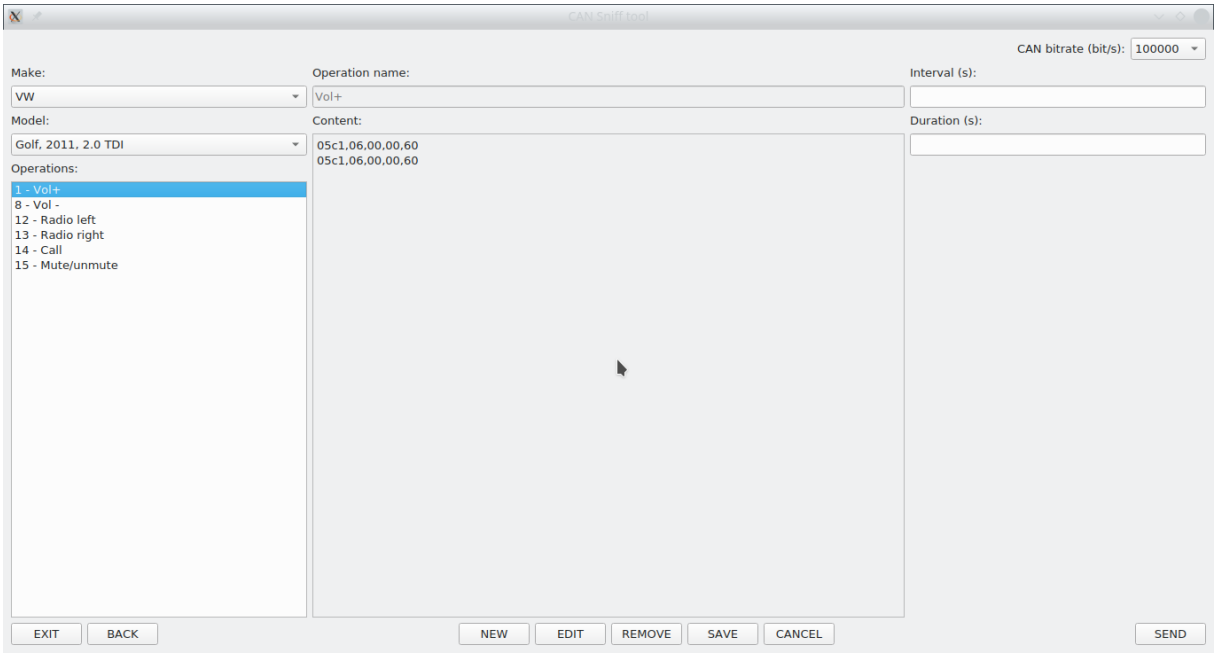


Slika 19: Popunjeno sučelje za zaprimanje

Svaki paket koji prolazi CAN sabirnicom ulazi u aplikaciju, te se po redu dospijeća zapisuju na ekran sa slike 19. U tablici se oni prikazuju poredani od najnižeg identifikatora prema najvišem pomoću algoritma za sortiranje. Ukoliko se radi o prvom pristiglom paketu, isti se zapisuje u prvi redak tablice. Za svaki sljedeći se uzima identifikator (*ID*), te se po redu uspoređuje sa identifikatorima iz tablice. Ukoliko je identifikator pristiglog paketa manji od prvog iz tablice, zapisuje se na prvo mjesto, a ostali iz tablice se pomiču za jedno mjesto dole. U protivnom se traži prvi veći identifikator od pristiglog. Ukoliko se takav pronađe, pristigli paket se zapiše u redak prije njega, ili na posljednje mjesto u slučaju da takav nije pronađen. Postoji mogućnost da identifikator pristiglog paketa odgovara nekom identifikatoru iz tablice. U tom slučaju se provjeravaju kolone d0 – d7. Ako se *data* podatak paketa na mjestu „x“ razlikuje od podatka zapisanog u dx koloni tablice za zadani paket, stanje u tablici se osvježava sa podatkom iz paketa, te se ta ćelija zacrveni kako bi se označila novonastala promjena. Istovremeno se osvježavaju i podaci o duljini poruke, broju pristiglih paketa za pojedini identifikator, broju

promjena, te vremenu zaprimanja paketa. Crvena boja ćelije se miče ako se podatak iz paketa i onaj iz ćelije ne razlikuju kod sljedećeg paketa s istim identifikatorom.

Aplikacija također može raditi u načinu rada za slanje paketa. Tom načinu rada se može pristupiti klikom na gumb *Send data* sa slike 17. Nakon klika na spomenuti gumb, korisniku se otvara sučelje sa slike ispod.



Slika 20: Sučelje za slanje

Kao i svi ostali ekrani, ovaj je sačinjen od određenih elemenata sučelja. Ti elementi su:

- *CAN bitrate* – postavljanje brzine CAN sabirnice
- *Make* – Odabir marke vozila
- *Model* – Odabir model vozila
- *Operations* – Spremljene operacije koje su ustvari poruke spremne za slanje
- *Operation content* – Naziv spremljene operacije
- *Content* – Sadržaj operacije koja se može sastojati od više individualnih paketa.
- *Interval* – Interval slanja sadržaja operacije u sekundama
- *Duration* – Trajanje odašiljanja paketa
- *Exit* – Izlaz iz aplikacije
- *Back* – povratak na početni zaslon
- *New* – Nova operacija
- *Edit* – Ažuriranje postojeće operacije
- *Remove* – Brisanje postojeće operacije

- *Save* – Spremanje nove ili ažuriranje postojeće operacije
- *Cancel* – Otkazivanje kreiranja nove ili ažuriranja stare operacije
- *Send* – Slanje paketa

Ovaj ekran korisniku služi za kreiranje, ažuriranje, brisanje i spremanje poruka (poruka je sačinjena od jednog ili više paketa) koje bi eventualno kasnije htio poslati prema CAN sabirnici. Kod inicijalnog otvaranja ekrana, gotovo sve opcije na njemu su onemogućene. Kako bi se iste omogućile, potrebno je odabrati marku i model vozila. Popunjavanjem marke i modela, na ekranu se prikazuju sve spremljene poruke za određeni model ukoliko ih ima. Ukoliko ne postoji ni jedna poruka, lista je prazna, te se omogućava kreiranje nove. Odabirom određene poruke, u centralnom dijelu ekrana se prikazuje njen sadržaj koji se može po potrebi editirati. Postoji li potreba za uklanjanjem određene poruke, to se može učiniti označavanjem iste, te klikom na gumb *Remove.*

Prilikom klika na gumb *Send*, provjerava se je li u polja *Interval* i *Duration* upisana kakva vrijednost. Ukoliko je bilo koje od tih polja popunjeno, oni se uzimaju u obzir prilikom slanja. *Interval* služi kao odgoda slanja sljedeće poruke za onoliki period koji je u polje upisan. S druge strane, *Duration* je vremensko ograničenje koje govori koliko dugo se moraju poruke slati. Nakon provjere je li sve u redu sa porukom, kreira se veza prema CAN sabirnici pomoću *Bus* klase *python-can* modula, nakon čega započinje slanje poruke. Ovisno o sadržaju poruke, na vozilu se može primijetiti neka promjena. Kada bi se za primjer poslala poruka označena na slici (*Vol+*), u vozilu bi na radiju/navigaciji primijetili da se glasnoća zvuka povećala.

5.2. Analiza paketa

CANSniffer aplikacija je veoma dobra ukoliko se reverznim inženjeringom želi otkriti kako određeni paket djeluje na vozilo, no nije naročito dobro za neke malo dublje analize. Upravo zato je u aplikaciju implementirana funkcionalnost zapisivanja svih dolaznih paketa u xlsx datoteku kako bi se pomoću Excel alata mogle izvršavati daljnje analize.

Za potrebe ovog rada, napravljene su dvije xlsx datoteke. Prva je kreirana snimajući sve pristigle pakete u vremenskom periodu od otprilike 60 sekundi, te je snimila nešto više od 16000 paketa. Snimanje je izvršeno sa pokrenutim motorom vozila i bez ikakve ljudske interakcije s vozilom koja bi mogla utjecati na rezultate, što znači da bi se u datoteci trebali nalaziti samo oni paketi koje vozilo razmjenjuje u neutralnom načinu rada. Upravo iz tog razloga ta datoteka za potrebe ovog rada nije

toliko zanimljiva za analizu, no ukoliko je potrebno, može je se pronaći u prilozima pod nazivom *CANSniffer-03-09-2020-21-55-31_idle_original.*

Druga datoteka je mnogo zanimljivija. Snimljena je tijekom maksimalne moguće ljudske interakcije s vozilom kako bi se identificiralo čim više paketa sa različitim identifikatorima. Snimanje datoteke je trajalo nekoliko minuta, te je zabilježeno nešto više od 50000 paketa. Originalna datoteka snimanja, kao i datoteka u kojoj su sve analize, mogu se također naći u prilozima pod nazivima: *CANSniffer-03-12-2020-21-10-01_all_original,* te *CANSniffer-03-12-2020-21-10-01_all_analiza.*

Na sljedećoj se slici može naći rezultat analize druge datoteke.

ARB ID	BROJ ANALIZIRANIH PAKETA	PROSJEČNO VRIJEME IZMEĐU PAKETA (s)	MINIMALNO VRIJEME IZMEĐU PAKETA (s)	MAKSIMALNO VRIJEME IZMEĐU PAKETA (s)	DULJINA PODATAKA U BAJTOVIMA	NAMJENA
0151	1804	0.100639	0.012820	0.452670	4	
0291	50	0.090568	0.010480	1.260740	5	Unutarnja tipka za otključavanje/zaključavanje
02c1	1842	0.098516	0.026500	0.465550	6	
02c3	1804	0.100594	0.032640	0.467020	2	Ručice za brisače i svjetla/žmigavce
0351	1802	0.100712	0.034620	0.458030	8	
0359	1804	0.100636	0.043120	0.453500	8	
035b	1810	0.100257	0.003990	0.452640	8	
0369	1802	0.100709	0.044130	0.478490	8	
03c3	1804	0.100642	0.022780	0.454700	8	
03e1	1814	0.100105	0.013170	0.452320	8	
03e3	225	0.806648	0.510360	1.065030	8	
042b	1804	0.100636	0.012660	0.451850	6	
0439	1802	0.100713	0.029330	0.445200	6	
0457	1803	0.100703	0.027400	0.446760	4	
0470	3607	0.050331	0.005640	0.420800	8	
0497	1818	0.099865	0.008610	0.455840	8	
0523	900	0.201611	0.091060	0.546610	8	
0527	900	0.201611	0.094530	0.540140	8	
0531	3647	0.049779	0.003850	0.415600	4	
054b	1638	0.104958	0.039850	6.970550	8	Parking senzori
0555	1804	0.100647	0.023740	0.452870	8	
0557	360	0.503229	0.286250	0.728740	8	
0571	361	0.503161	0.293320	0.866250	6	
0575	907	0.200084	0.024630	0.591700	4	
05b5	902	0.201400	0.104450	0.520100	8	
05c1	209	0.109994	0.012470	1.882530	4	Naredbe na upravljaču
05e1	904	0.200858	0.098250	0.548710	7	
060e	180	1.006441	0.392020	1.405420	2	
0621	1802	0.100712	0.031660	0.453180	7	
062f	911	0.199395	0.033770	0.517050	4	
0635	942	0.192723	0.011850	0.547030	3	
0651	1803	0.100656	0.012770	0.457630	8	
0653	360	0.503232	0.286170	0.727650	3	
0655	360	0.503237	0.285960	0.727620	8	
0658	1804	0.100644	0.017450	0.450410	8	
065d	180	1.006124	0.409340	1.358260	8	
065f	902	0.201195	0.083330	0.590950	8	
0661	179	1.012409	0.630560	1.476540	8	
066c	266	0.680130	0.020250	1.408820	3,4,5,6,7,8	
067a	171	1.053851	0.075620	8.128860	2,4	
067c	40	4.081941	0.344260	20.715890	2	
067d	234	0.773457	0.016660	1.393230	3,4,5,8	
06c9	39	4.291528	0.380200	21.176720	2	
06cb	3	14.961020	0.102550	29.819490	4	
06da	3284	0.054855	0.009350	2.212200	3,4,5,6,8	
06db	314	0.578399	0.018060	1.362490	3,4,5,6,8	

Slika 21: Rezultati snimanja paketa

S obzirom da CAN protokol funkcionira na način da paketi sa manjim identifikatorom imaju veći prioritet, cilj ove analize je bio istražiti vremenske parametre paketa, moguće duljine podataka za pojedine identifikatore, te ukoliko je moguće, pokušati pronaći njihovu namjenu. Tako se u prvoj koloni tablice može naći identifikator paketa. Što je on manji, prioritet mu je veći. U sljedećoj koloni je ukupan broj paketa po pojedinom identifikatoru. Naredne tri kolone se odnose na vremenske parametre paketa. Prva je prosjek vremena između dolaska dva paketa sa istim identifikatorom. Naravno, što je broj paketa veći, to je i informacija u ovoj koloni pouzdanija. Minimalno vrijeme između paketa označava najmanji vremenski period između dva uzastopna paketa sa istim identifikatorom, dok maksimalno vrijeme označava suprotno od toga. Duljina podatka paketa može varirati od 0 do 8 bajtova, te se u šestoj koloni može naći podatak o tome kojih duljina se sve šalju paketi za određene identifikatore. Za neke identifikatore je tokom rada i analize utvrđeno čemu oni služe, odnosno čime upravljaju, te se ta informacija može naći u posljednjoj koloni.

Iz tablice se može izvući nekoliko veoma zanimljivih zaključaka. Snimanjem je utvrđeno 46 jedinstvenih identifikatora. Za dobar dio njih se može primijetiti kako je u vremenskom periodu u kojem je provedeno snimanje, zaprimljen veoma sličan broj paketa za neke od njih.

- *0470, 0531* – oko 3600 paketa
- *0151, 02c3, 0351, 0359, 035b, 0369, 03c3, 03e1, 042b, 0439, 0457, 0497, 0555, 0621, 0651, 0658* – oko 1800 paketa
- *0523, 0527, 0575, 05b5, 05e1, 062f, 0635, 065f* – oko 900 paketa
- *0557, 0571, 0653, 0655* – oko 360 paketa
- *060e, 065d* – oko 180 paketa

Iz prethodnog nabrajanja su namjerno izostavljeni svi identifikatori kojima je utvrđena namjena, jer se oni odašilju/zaprimaju samo u slučaju da je ta funkcionalnost uključena. Preostali identifikatori nemaju neki regularan ponavljajući broj paketa, te ih nije bilo moguće svrstati u ni jednu kategoriju. Pretpostavka je da se radi o kontrolnim ili o paketima koji upravljaju nekim sekundarnim funkcijama vozila koje ne zahtijevaju mnogo informacija u jedinici vremena. Za iznad nabrojene pakete je zaključeno kako se radi ili o kontrolnim paketima, ili o paketima koji su apsolutno nužni za normalan rad vozila i njegovih komponenti kao što je recimo motor.

Prije ove analize mišljenje je bilo da se paketi sa različitim identifikatorima šalju u relativno sličnom vremenskom periodu, no to baš i nije tako. Naime sa slike je vidljivo da se paketi sa manjim identifikatorima generalno šalju učestalije od onih sa višim identifikatorima (*kolona prosječno vrijeme između paketa)*. Gotovo cijela prva polovica paketa s najnižim identifikatorima se šalje svakih 100 milisekunda, dok je kod druge polovice paketa prosječno vrijeme slanja od 100 milisekunda, pa do gotovo 15 sekundi, no broj paketa za neke od identifikatora je jednostavno pre malen kako bi rezultati bili reprezentativni. Postoji još nekoliko objašnjenja za toliki vremenski skok između minimalnog i maksimalnog vremena između dva

paketa, a to su potencijalne greške pri prijenosu paketa unutar same *CAN* sabirnice, ali i greška u obradi istih u aplikaciji. Naime aplikacija je napravljena na način da preskoči paket ukoliko dođe do greške pri obradi istog, te nastavi sa obradom sljedećeg pristiglog.

Još jedan zanimljiv zaključak se može izvući iz kolone 6. Kod većine paketa, poruka je uvijek konstantne duljine, no to nije slučaj kod paketa sa identifikatorima *066c, 067a, 067d, 06da,* te *06db.* Kod tih paketa, poruke mogu imati podatke varijabilne duljine, no zanimljivo je da ni jedan paket nije imao manju duljinu poruke od 2 bajta.

Naredne tablice su sve istog formata, ali s dobrim razlogom. Svaka od njih je posvećena jednom od identifikatora kojem je otkrivena namjena, te prikazuje koji se podatak šalje u određenom *data* polju kako bi se izvršila predodređena akcija. Značenja kolona i redaka su sljedeća:

- Stupac *Arbitration ID* – identifikator paketa koji označava njegov prioritet
- Stupci *D0, .., D7* – podatkovni skup paketa. Svaka od ovih kolona označava bajt podatka na toj poziciji
- Stupac *Namjena* – označava akciju koja se izvrši ukoliko kroz *CAN* sabirnicu dođe paket sa identifikatorom i podacima iz tog retka
- Redak *Dodatni opis* – odnosi se samo na stupce podatkovnog skupa. Prikazuje mijenja li se podatak na toj poziciji paketa

Bitno je napomenuti da su u svakoj tablici plavo označeni varijabilni dijelovi podatkovnog skupa paketa, odnosno oni dijelovi koji su zaduženi za obavljanje predodređene akcije.

ARBITRATION ID	D0	D1	D2	D3	D4	D5	D6	D7	NAMJENA
0291	0a	55	04	00	00	-	-	-	UNUTARNJA TIPKA ZA ZAKLJUČAVANJE
0291	0a	aa	02	01	01	-	-	-	UNUTARNJA TIPKA ZA OTKLJUČAVANJE
DODATNI OPIS	-	PROMJENJIVO	PROMJENJIVO	-	-	-	-	-	-

Slika 22: ARB ID - 0291, paketi unutarnje tipke zaključavanja/otključavanja

Tablica sa slike 22 prikazuje format paketa koji se šalje *CAN* sabirnicom ukoliko se pritisne prekidač za otključavanje ili zaključavanje na vozačevim vratima. Vidljivo je kako se paket sastoji od 5 bajtova podataka, od čega su promjenjivi samo bajtovi na pozicijama *D1 i D2.* Format ovih paketa je relativno jednostavan s obzirom da se radi o obavljanju samo dvije akcije, ali je savršen kao uvodni primjer u ovakav prikaz podataka.

ARBITRATION ID	D0	D1	D2	D3	D4	D5	D6	D7	NAMJENA
02c1	01	xx	xx	00	02	00	-	-	LIJEVI ŽMIGAVAC
02c1	02	xx	xx	00	02	00	-	-	DESNI ŽMIGAVAC
02c1	04	xx	xx	00	02	00	-	-	BLICANJE
02c1	08	xx	xx	00	02	00	-	-	DUGA SVJETLA
02c1	xx	00	xx	00	02	00	-	-	BEZ OPERACIJE (IDLE)
02c1	xx	01	xx	00	02	00	-	-	BRISAČI 1. BRZINA
02c1	xx	02	xx	00	02	00	-	-	BRISAČI 2. BRZINA
02c1	xx	04	xx	00	02	00	-	-	BRISAČI 3. BRZINA
02c1	xx	08	xx	00	02	00	-	-	BRISAČI 4. BRZINA
02c1	xx	10	xx	00	02	00	-	-	BRISANJE VJETROBRANSKOG STAKLA NAKON PRSKANJA
02c1	xx	11	xx	00	02	00	-	-	BRISANJE VJETROBRANSKOG STAKLA NAKON PRSKANJA 1. BRZINA BRISANJA
02c1	xx	12	xx	00	02	00	-	-	BRISANJE VJETROBRANSKOG STAKLA NAKON PRSKANJA 2. BRZINA BRISANJA
02c1	xx	14	xx	00	02	00	-	-	BRISANJE VJETROBRANSKOG STAKLA NAKON PRSKANJA 3. BRZINA BRISANJA
02c1	xx	18	xx	00	02	00	-	-	BRISANJE VJETROBRANSKOG STAKLA NAKON PRSKANJA 4. BRZINA BRISANJA
02c1	xx	30	xx	00	02	00	-	-	PRSKANJE VJETROBRANSKOG STAKLA
02c1	xx	31	xx	00	02	00	-	-	PRSKANJE VJETROBRANSKOG STAKLA 1. BRZINA BRISANJA
02c1	xx	32	xx	00	02	00	-	-	PRSKANJE VJETROBRANSKOG STAKLA 2. BRZINA BRISANJA
02c1	xx	34	xx	00	02	00	-	-	PRSKANJE VJETROBRANSKOG STAKLA 3. BRZINA BRISANJA
02c1	xx	38	xx	00	02	00	-	-	PRSKANJE VJETROBRANSKOG STAKLA 4. BRZINA BRISANJA
02c1	xx	40	xx	00	02	00	-	-	BRISANJE STRAŽNJEG STAKLA
02c1	xx	41	xx	00	02	00	-	-	BRISANJE STRAŽNJEG STAKLA, PREDNJI BRISAČI 1. BRZINA
02c1	xx	42	xx	00	02	00	-	-	BRISANJE STRAŽNJEG STAKLA, PREDNJI BRISAČI 2. BRZINA
02c1	xx	44	xx	00	02	00	-	-	BRISANJE STRAŽNJEG STAKLA, PREDNJI BRISAČI 3. BRZINA
02c1	xx	48	xx	00	02	00	-	-	BRISANJE STRAŽNJEG STAKLA, PREDNJI BRISAČI 4. BRZINA
02c1	xx	80	xx	00	02	00	-	-	PRSKANJE STRAŽNJEG STAKLA
02c1	xx	81	xx	00	02	00	-	-	PRSKANJE STRAŽNJEG STAKLA, PREDNJI BRISAČI 1. BRZINA
02c1	xx	82	xx	00	02	00	-	-	PRSKANJE STRAŽNJEG STAKLA, PREDNJI BRISAČI 2. BRZINA
02c1	xx	84	xx	00	02	00	-	-	PRSKANJE STRAŽNJEG STAKLA, PREDNJI BRISAČI 3. BRZINA
02c1	xx	88	xx	00	02	00	-	-	PRSKANJE STRAŽNJEG STAKLA, PREDNJI BRISAČI 4. BRZINA
02c1	xx	xx	01	00	02	00	-	-	PREKIDAČ BRZINE PREDNJIH BRISAČA, 1. BRZINA
02c1	xx	xx	05	00	02	00	-	-	PREKIDAČ BRZINE PREDNJIH BRISAČA, 2. BRZINA
02c1	xx	xx	09	00	02	00	-	-	PREKIDAČ BRZINE PREDNJIH BRISAČA, 3. BRZINA
02c1	xx	xx	0d	00	02	00	-	-	PREKIDAČ BRZINE PREDNJIH BRISAČA, 4. BRZINA
DODATNI OPIS	PROMJENJIVO	PROMJENJIVO	PROMJENJIVO	-	-	-	-	-	-

Slika 23: ARB ID - 02c1, paketi ručica brisača i svjetla/žmigavaca

Slika 23 je u odnosnu na onu prethodnu mnogo kompliciranija. Paketi sa identifikatorom *02c1* su zaduženi za upravljanje akcijama koje se izvode pomicanjem ručica iza volana (svjetla, žmigavci, brisači). Iz slike je vidljivo kako su paketi konstantne duljine 6 bajta, a mijenjaju se samo bajtovi na pozicijama *D0, D1 i D2.* Pozicija *D1* je rezervirana za lijevu ručicu (žmigavci i svjetla), dok pozicije *D1 i D2* služe za desnu ručicu (brisači). Pomnijim promatranjem bajtova na pozicijama *D1 i D2* može se zaključiti kako veća vrijednost na toj poziciji odgovara i većoj brzini rada brisača, dok se kombinacija brisanja i prskanja dobije zbrajanjem bajtova zaduženih za pojedinu akciju. Na primjer, prva brzina brisača ima vrijednost bajta *01* na poziciji *D1.* Prskanje vjetrobranskog stakla ima vrijednost *30* na istoj poziciji. Ukoliko se želi postići efekt prskanja i brisanja pri prvoj brzini, na poziciju *D1* se upisuje zbroj prethodna dva bajta (*01 + 30 = 31*).

ARBITRATION ID	D0	D1	D2	D3	D4	D5	D6	D7	NAMJENA
0541	ff - 00	xx	xx	xx	xx	xx	xx	xx	PREDNJI LIJEVI SENZOR
0541	xx	ff - 00	xx	xx	xx	xx	xx	xx	PREDNJI DESNI SENZOR
0541	xx	xx	ff - 00	xx	xx	xx	xx	xx	STRAŽNJI LIJEVI SENZOR
0541	xx	xx	xx	ff - 00	xx	xx	xx	xx	ZADNJI CENTRALNI DESNI SENZOR
0541	xx	xx	xx	xx	ff - 00	xx	xx	xx	PREDNJI CENTRALNI LIJEVI SENZOR
0541	xx	xx	xx	xx	xx	ff - 00	xx	xx	PREDNJI CENTRALNI DESNI SENZOR
0541	xx	xx	xx	xx	xx	xx	ff - 00	xx	STRAŽNJI CENTRALNI LIJEVI SENZOR
0541	xx	xx	xx	xx	xx	xx	xx	ff - 00	STRAŽNJI DESNI SENZOR
DODATNI OPIS	PROMJENJIVO	PROMJENJIVO	PROMJENJIVO	PROMJENJIVO	PROMJENJIVO	PROMJENJIVO	PROMJENJIVO	PROMJENJIVO	-

Slika 24: ARB ID - 054b, paketi parking senzora

Na slici 24 vidljivi su paketi koji se šalju ukoliko su uključeni senzori za mjerenje udaljenosti. Uvijek se šalju paketi sa istim identifikatorom (*0541*), te konstantne duljine paketa od 8 bajta. Zanimljivo je da svaka *Dx* pozicija odgovara jednom od senzora na vozilu (ukupno ih je 8). Detaljnijim proučavanjem pristiglih paketa pod ovim identifikatorom može se zaključiti kako se vrijednost bajta mijenja ovisno o udaljenosti. Naime što je vozilo kod određenog senzora bliže nekom objektu, vrijednost bajta na njegovoj poziciji je manja. Ukoliko se recimo prednji lijevi senzor (*D0*) počne približavati nekom objektu, inicijalna vrijednost na toj poziciji mu je *ff*. Daljnjim približavanjem se ta vrijednost smanjuje dok ne dođe do *00*, što znači da je senzor ili udario u objekt, ili da mu je veoma blizu.

ARBITRATION ID	D0	D1	D2	D3	D4	D5	D6	D7	NAMJENA
05c1	00	00	00	60	-	-	-	-	KONTROLNI PAKET
05c1	02	00	00	60	-	-	-	-	LIJEVA STRANA UPRAVLJAČA, TIPKA DESNO
05c1	03	00	00	60	-	-	-	-	LIJEVA STRANA UPRAVLJAČA, TIPKA LIJEVO
05c1	06	00	00	60	-	-	-	-	LIJEVA STRANA UPRAVLJAČA, TIPKA POJAČAVANJE
05c1	07	00	00	60	-	-	-	-	LIJEVA STRANA UPRAVLJAČA, TIPKA SMANJIVANJE
05c1	09	00	00	60	-	-	-	-	DESNA STRANA UPRAVLJAČA, TIPKA LIJEVO
05c1	0a	00	00	60	-	-	-	-	DESNA STRANA UPRAVLJAČA, TIPKA DESNO
05c1	1a	00	00	60	-	-	-	-	LIJEVA STRANA UPRAVLJAČA, TIPKA ZA BLUETOOTH
05c1	22	00	00	60	-	-	-	-	DESNA STRANA UPRAVLJAČA, TIPKA GORE
05c1	23	00	00	60	-	-	-	-	DESNA STRANA UPRAVLJAČA, TIPKA DOLE
05c1	28	00	00	60	-	-	-	-	DESNA STRANA UPRAVLJAČA, TIPKA OK
05c1	29	00	00	60	-	-	-	-	DESNA STRANA UPRAVLJAČA, TIPKA ZA POVRATAK
05c1	2a	00	00	60	-	-	-	-	LIJEVA STRANA UPRAVLJAČA, TIPKA ZA UŠUTKAVANJE (MUTE)
DODATNI OPIS	PROMJENJIVO	-	-	-	-	-	-	-	-

Slika 25: ARB ID - 05c1, paketi tipki upravljača

Posljednji primjer su paketi zaduženi za upravljanje tipkama na upravljaču. Uvijek se šalje paket duljine 4 bajta, no samo je bajt na poziciji *D0* varijabilan. I ovdje je uočena jedna zanimljivost. Pritiskom bilo koje tipke na upravljaču, uvijek se šalje 6 paketa. Prvi paket uvijek odgovara svojoj namjeni, dok su naredni paketi kontrolni. Recimo neka se pritisne tipka za pojačavanje, prvo se šalje paket koji na poziciji *D0* ima vrijednost *06*, a zatim slijedi 5 paketa koji na toj poziciji imaju vrijednost *00*. Ukoliko se želi pomoću aplikacije poslati naredba za pojačavanjem, kontrolne pakete nije potrebno slati.

5.3. Iskorištavanje ranjivosti pomoću CANSniffer-a

Iskorištavanje ranjivosti bilo kojeg sustava nije baš trivijalan zadatak. Već je razvoj same aplikacije iz prethodnog poglavlja bio zahtjevan pothvat koji je za preduvjet, među ostalim, imao dobro znanje nekog od programskih jezika (u ovom slučaju *Python*), znanje o mrežnim protokolima (prvenstveno *UDP*), te veoma duboko poznavanje CAN protokola, no izrada aplikacije je samo prvi korak. Kako je za ovaj rad odlučeno da će se ranjivost iskoristiti na način da se direktno spoji na CAN sabirnicu, bilo je potrebno locirati jednu od njih u vozilu. Postoje žice CAN sabirnice kojima se može pristupiti s vanjske strane, no s obzirom da se radi

o testnom vozilu, korištena je sabirnica za multimediju koja se nalazi ispod suvozačevog sjedala i spojena je na *bluetooth* modul.

CAN sabirnicu je inače relativno jednostavno identificirati. Samo je potrebno naći dvije uvrnute žice nalik na telefonske parice i spojiti se na njih preko zelene i žute žice kabela prikazanog na slici 15. Kako bi slanje i primanje paketa funkcioniralo, također je potrebno spojiti crnu žicu bilo kuda na šasiju jer se radi o minusu napajanja. Naravno zelenu i žutu žicu je potrebno pravilno spojiti. Parica sabirnice uvijek ima jednu žicu s crtom i jednu bez nje. Na žalost nije moguće znati koja je koja, no to se lako može provjeriti na dva načina. Prvi je multi metrom gdje se provjerava napon između dvije žice, dok se drugi zasniva na metodi pokušaja i promašaja. Kako postoje samo dvije moguće kombinacije, ovaj drugi je često i jednostavniji.

Cijeli cilj iskorištavanja ove ranjivosti je otkriti paket (pakete) zadužen za obavljanje neke akcije, te pokušati rekreirati tu akciju pomoću aplikacije. Tako je zaključeno da su akcije pojačavanja i smanjivanja glasnoće zvuka radija najjednostavnije za identificirati i rekreirati. Analizom zaprimanja paketa, otkriveno je da se recimo za pojačavanje glasnoće uvijek pošalje 6 paketa veličine 4 bajta i to sa ID-em *05c1*. Prvog sačinjavaju bajtovi *06 00 00 60*, dok je ostalih 5 oblika *00 00 00 60*. Također je otkriveno da samo prvi paket služi za pojačavanje zvuka, dok su ostali paketi kontrolni. Primjer takvog paketa se može vidjeti na slici 19 u retku pod prethodno navedenim ID-em.

Sa novootkrivenim znanjem je zatim potrebno otići na sučelje za slanje i kreirati novu operaciju. Primjer takve operacije je prikazan na slici 20, no na toj se slici nalaze samo dva paketa. Izvršavanjem novokreirane operacije je primijećeno da se glasnoća povisila dva puta za redom. To je bio dokaz da su paketi koji slijede nakon prvog zapravo kontrolni, ali još važnije, ovo je bio dokaz da je ranjivost uspješno iskorištena, te da je sigurnost vozila ugrožena.

U prethodnom se tekstu mogao vidjeti jednostavan primjer iskorištavanja navedene prijetnje, no mogućnosti su beskonačne. Sve akcije koje se u vozilu odvijaju, a potaknute su paketima koji prolaze CAN sabirnicom, mogu se identificirati i rekreirati, pa tako i recimo one za pokretanje ili kontrolu rada motora.

Kao što je i već ranije navedeno, aplikacije se može pronaći na posljednjoj stranici ovog rada.

6. Zaključak

Kako vozila dodavanjem novih funkcionalnosti postaju tehnički i elektronički sve složenija, tako raste i broj sigurnosnih rizika. U vrijeme isključivo mehaničkih vozila, postojalo je svega nekoliko mogućih prijetnji od kojih je najgora bila krađa istog. Danas je situacija drastično drugačija. Napretkom tehnologije i dodavanjem novih funkcionalnosti, vozila su postala više funkcionalna, no veći broj funkcionalnosti donosi i veći broj sigurnosnih prijetnji koje bi netko mogao iskoristiti. Trenutno više nije pitanje može li se iskoristiti ranjivost nekog vozila, već na koje sve načine. Upravo je cilj ovog rada bio identificirati sigurnosne rizike, ocijeniti koliko prijetnju predstavljaju, te na posljetku pokušati iskoristiti neku od njih.

U početku je nešto više riječi bilo o povijesnim pokušajima zaštite vozila, prvenstveno od krađe, koji su dobrim dijelom bili neuspješni iz razloga jer su lopovi uvijek bili barem dva koraka ispred onih koji su te sigurnosne prepreke dizajnirali. Nastavljeno je sa istraživanjem prošlih, sadašnjih i budućih komunikacijskih protokola koji se ugrađuju u vozila. Otkriveno je da većina tih protokola nije dizajnirana sa sigurnošću u prvom planu, već se sigurnost žrtvuje kako bi se osigurala velika brzina razmjene informacija, te maksimalna moguća interoperabilnost između komponenti. Jedan od tih protokola je i CAN, no postoje neki kao što su *FlexRay* ili *Automotive Ethernet*, koji bi ga mogli zamijeniti i usput pokriti neke od sigurnosnih propusta.

Analiziranjem različitih prijetnji utvrđeno je kako su daleko najopasnije one koje se tiču CAN sabirnice. Ukoliko napadač bilo kojim kanalom uspije ostvariti komunikaciju sa internom CAN mrežom vozila, može počiniti ogromnu štetu. U primjeru iz prethodnog poglavlja bi napadač trebao imati fizički pristup vozilu kako bi mogao slati maliciozne pakete, no kad bi se koristilo sučelje sa recimo bluetooth-om ili još gore, neko sučelje koje pomoću SIM kartice može slati podatke na veoma velike udaljenosti, posljedice bi mogle biti katastrofalne. Primjer iskorištavanja ranjivosti koji je u ovom radu prikazan je zapravo banalan, no spojem na neku kritičniju sabirnicu bi se moglo učiniti mnogo više.

2020 je godina. Svijet je sve bliže i bliže autonomnoj vožnji i komunikaciji među vozilima. Ako trenutno napadač može preuzeti kontrolu nad jednim vozilom, lako je zamisliti što bi sve mogao učiniti kad bi istovremeno mogao preuzeti kontrolu nad više njih. Zbog tog, a i mnogih drugih razloga, krajnje je vrijeme da proizvođači napokon počnu posvećivati više pažnje sigurnosti. U protivnom nam se svima loše piše.

Popis literature

[1] „The Evolution of Car Security - Eagle Ridge GM“ (20.07.2017), Vehicle Security [Na internetu]. Dostupno: https://www.eagleridgegm.com/evolution-car-security/ [pristupano 02.08.2018]

[2] Jan C. van Ours, Ben Vollaard, „The engine immobilizer: A non-starter for car thieves“, 04.07.2011. [Na internetu]. Dostupno: https://carsafe.org/letters/Automobile_Safety_Foundation_-_Attachment.pdf [pristupano 02.08.2018]

[3] Joe Stanganelli, „Which Threat Risk Model Is Right for Your Organization?“, 19.09.2018. [Na internetu]. Dostupno: https://www.esecurityplanet.com/network-security/which-threat-risk-model-is-right-for-your-organization.html [pristupano 03.08.2018]

[4] Bilal, „LIN Communication Protocol introduction, working and applications“, 3. Mjesec 2018. [Na internetu]. Dostupno: http://microcontrollerslab.com/lin-communication-protocol-working/ [pristupano 15.08.2018]

[5] Corrigan, „Introduction to the Controller Area Network (CAN)“, 5. Mjesec 2016. [Na internetu]. Dostupno: https://www.ti.com/lit/an/sloa101b/sloa101b.pdf [pristupano 20.03.2019]

[6] Parikh, „CAN Protocol – Understanding the Controller Area Network Protocol“, [Na internetu]. Dostupno: https://www.engineersgarage.com/article/what-is-controller-area-network [pristupano 20.03.2019]

[7] CAN FD – The basic idea, [Na internetu]. Dostupno: https://www.can-cia.org/jp/can-knowledge/can/can-fd/ [pristupano 22.03.2019]

[8] Craig Smith, „The Car Hacker's Handbook“, 2016

[9] Modeliranje sigurnosnih prijetnji (Threat modeling), [Na internetu]. Dostupno: https://www.cis.hr/files/dokumenti/CIS-DOC-2012-05-049.pdf [pristupano 06.01.2020]

[10] PyQt5, [Na internetu]. Dostupno: https://pypi.org/project/PyQt5/ [pristupano 08.01.2020]

[11] python-can, [Na internetu]. Dostupno: https://python-can.readthedocs.io/en/master/ [pristupano 08.01.2020]

[12] pySerial, [Na internetu]. Dostupno: https://pythonhosted.org/pyserial/ [pristupano 08.01.2020]

[13] Nikola Ivković, Dario Kresic, Kai-Steffen Hielscher, Reinhard German. "Verifying Worst Case Delays in Controller Area Network", Lecture Notes in Computer Science, 7201 (2012), 91-105

[14] Nikola Ivković, Luka Milić, Mario Konecki. "A Timed Automata Model for Systems with Gateway-Connected Controller Area Networks", 2018 IEEE 3rd International Conference on Communication and Information Systems (ICCIS), Singapore: Institute of Electrical and Electronics Engineers (IEEE), 2018. str. 97-101

Popis slika

Slika 1: Vrste ključeva 2
Slika 2: Aftermarket alarmni sustav 4
Slika 3: Imobilizator (zasebni, Golf 3) 5
Slika 4: Imobilizator (integrirani, gore: Audi, dolje: VW) 5
Slika 5: Transponderi 6
Slika 6: ISO 9141-2 pinout 9
Slika 7: LIN paket 10
Slika 8: OSI i CAN 12
Slika 9: Standard CAN paket 13
Slika 10: Prošireni CAN paket 14
Slika 11: Razina 0 modela prijetnje 17
Slika 12: Razina 1 18
Slika 13: Ocjenjivanje prema DREAD metodi 21
Slika 14: Analiza rizika prijetnji CAN sabirnice 22
Slika 15: OBDII kabel za direktno spajanje na CAN sabirnicu 24
Slika 16: Slcan uređaj nije pronađen 25
Slika 17: Početni zaslon aplikacije 26
Slika 18: Prazno sučelje za zaprimanje paketa 27
Slika 19: Popunjeno sučelje za zaprimanje 28
Slika 20: Sučelje za slanje 29
Slika 21: Rezultati snimanja paketa 31
Slika 22: ARB ID - 0291, paketi unutarnje tipke zaključavanja/otključavanja 33
Slika 23: ARB ID - 02c1, paketi ručica brisača i svjetla/žmigavaca 34
Slika 24: ARB ID - 054b, paketi parking senzora 34
Slika 25: ARB ID - 05c1, paketi tipki upravljača 35