SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 6351

Kontrola ulaza korištenjem beskontaktnih kartica

Filip Ptiček

Zagreb, lipanj 2019.

Umjesto ove stranice umetnite izvornik Vašeg rada.
Da bi ste uklonili ovu stranicu obrišite naredbu `\izvornik`.

SADRŽAJ

1. Uvod **1**

2. NFC tehnologija **2**

2.1. Povijest 2

2.2. Tehnološki standard 3

2.3. Sigurnosni problemi NFC tehnologije 5

2.3.1. Prisluškivanje 5

2.3.2. Korupcija i manipulacija podataka 5

2.3.3. Presretanje 5

2.3.4. Krađa 5

2.4. Jedinstveni identifikacijski broj (UID) 6

3. Sustav za kontrolu ulaza korištenjem beskontaktnih kartica **7**

3.1. Motivacija i opis problema 7

3.2. Korišteni razvojni alati i uređaji 8

3.2.1. Operacijski sustav GNU/Linux 8

3.2.2. Ugradbeno računalo Raspberry Pi 8

3.2.3. NFC čitač Gemini 2000 Orbit IP 9

3.2.4. Programski jezik Python 3 10

3.2.5. Baza podataka TinyDB 10

3.3. Arhitektura i dizajn sustava 10

3.3.1. Struktura programske podrške 11

3.3.2. Baza podataka 11

3.3.3. Autentifikacija 12

3.3.4. Integrirano rješenje za kontrolu ulaza 17

4. Zaključak **19**

5. Literatura **20**

1. Uvod

U današnjem svijetu pokušavamo povezati sve više stvari, uređaja i pomagala s tehnologijom. Na taj način se pokušava olakšati korištenje i mogućnost automatizacije pomoću jednog centralnog mjesta. Takva rješenja nam omogućuju korištenje jednog uređaja za upotrebu u plaćanju, identifikaciji te mnogim drugim stvarima. Neki od centralnih mjesta su mobilni telefoni te beskontaktne kartice koje se nalaze u džepu većine današnjeg stanovništva.

Kontrola ulaza je jedan od poslova koji se od antičkih civilizacija prepuštalo da obavlja čovjek. Vrata koja su koristila mehanizme ključa i brave nisu dopuštale odstupanje od te norme. Tek pojavom kartica s magnetskom trakom došlo je do promjena. Takve kartice dopuštale su da se svakoj osobi dodijeli jedinstveni identifikator. Pomoću čitača kartica koje su sadržavale spremljene identifikatore moglo se dopustiti ulaz samo određenim osobama i ujedno voditi evidencija pristupa. Jedna od mana ovakve tehnologije je što korisnik treba karticu dovesti u direktan kontakt s čitačem te mogućnost jednostavnog repliciranja informacija spremljenih na njima.

Tehnologije kao što su radio-frekvencijska identifikacija (RFID) te NFC (eng. *Near field communication*), dozvoljavaju udaljenu komunikaciju između čitača i kartice ili oznake. U današnje vrijeme zamijenile su upotrebu kartica s magnetskom trakom zbog veće sigurnosti i u slučaju RFID-a mogućnosti za praćenjem položaja kartice ili oznake korištenjem komunikacije velikih frekvencija.

Danas se najčešće za kontrolu ulaza koristi NFC tehnologija zbog raširenosti u mobilnim telefonima i u slučaju studentske akademske zajednice Republike Hrvatske u njihovim akademskim iskaznicama, što ne zahtijeva uvođenje posebnih oznaka kao u slučaju RFID-a.

U sljedećem poglavlju obrađuje se pojedinosti NFC tehnologije, a nakon toga sustav za autentifikaciju na temelju beskontaktnih kartica.

2. NFC tehnologija

NFC (eng. *Near field communication*) je vrsta beskontaktne komunikacije između uređaja poput mobilnih telefona i beskontaktnih kartica. Beskontaktna komunikacija dozvoljava komunikaciju na male udaljenosti bez potrebe da uređaji dolaze u neposredni doticaj.

NFC dozvoljava uređaju, koji služi kao čitač, tj. ispitivač, proizvodi aktivno radio frekvenciju što mu dozvoljava da komunicira s ostalim NFC kompatibilnim uređajima ili karticama. Pasivni uređaji poput beskontaktnih kartica i oznaka, imaju spremljenu informaciju te tu informaciju razmjenjuju s čitačima, ali ne dozvoljavaju čitanje informacija drugih uređaja. Kod komunikacije dva aktivna uređaja postoji obostrana komunikacija slanja i primanja informacija.

Integracija beskontaktnih tehnologija u kreditne, putničke i kuponske kartice dozvoljava korisnicima da obavljaju plaćanja, ukrcavanje na javni prijevoz i razmjenu informacija preko jednostavnog približavanja kartica. Također postoji mogućnost integracije više usluga preko mobilnih telefona te tako eliminirati korištenje različitih kartica za različite usluge.

U današnje vrijeme sve više poduzeća implementira beskontaktne tehnologije u svoje usluge, uključujući implementacije virtualnih kartica koje dozvoljavaju plaćanjem na kartičnim terminalima pomoću mobilnih telefona. [1]

2.1. Povijest

NFC je nastao na temeljima radio-frekvencijske identifikacije (RFID). NFC je podset RFID-a s kraćim dometom zbog sigurnosnih razloga.

2004. godine Nokia, Sony i Philips zajedno su formirali NFC Forum. Njihov zajednički cilj je bio promoviranje sigurnosti, jednostavnosti korištenja i popularnosti NFC tehnologije. Zaduženje Foruma je održavanje standarda koji dozvoljava da tehnologija može funkcionirati između sva uređaja. Ako netko želi proizvesti NFC uređaj potrebno je slijediti standarde postavljene od strane NFC Foruma. To osigurava da ko-

Slika 2.1: Upotreba NFC-a [16]

risnik s bilo kojim NFC uređajem može komunicirati s nekim drugim NFC uređajem.

Iako je NFC Forum formiran 2004. godine, prve specifikacije za NFC oznake su se pojavile 2006. godine. NFC oznake su mali objekti koji sadrže informacije koje NFC čitaći mogu pročitati. Informacije na oznakama se u većini slučajeva mogu samo čitati, ali postoje i oznake u koje se mogu upisati nove ili promijeniti stare informacije.

Prvi mobilni telefon koji je sadržavao NFC sposobnosti je bila Nokia 6131 NFC predstavljena 2006. godine. Sazrijevanjem tehnologije dolazile su i nove specifikacije koje su osim podržavanja plaćanja dozvoljavale i razmjenu slika, videa i ostalih informacija. Danas je NFC tehnologija dostupna na većini mobilnih uređaja, od telefona, pametnih satova do integracije u automobile. [2]

2.2. Tehnološki standard

Kada se razvijaju NFC uređaji potrebno je pratiti NFC standarde. Standardi postoje da sve vrste NFC uređaja mogu međusobno komunicirati kako oni dizajnirani u prošlosti tako i u budućnosti. Postoje dvije vrste glavnih specifikacija za NFC tehnologiju:

- ISO/IEC 14443 koja definira identifikacijske kartice koje spremaju informacije, poput NFC oznaka
- ISO/IEC 18000-3 koja definira radio-frekvencijsku identifikacijsku komunikacijsko NFC uređaja

ISO/IEC 18000-3 je internacionalni standard za sve uređaje koji komuniciraju bežično na frekvenciji 13.56MHz. Uređaji trebaju biti na minimalnoj udaljenosti od 4cm

prije nego dolazi razmjene informacija. Ovi standardi opisuju način na koji čitač i NFC oznaka s koje se čita trebaju komunicirati međusobno.

Čitač šalje signal oznaci. Ako su uređaji dovoljno blizu jedan drugom oznaka postaje napajana preko signala čitača. To dozvoljava da oznaka bude malena i bez potrebe da sadrži bateriju ili svoj vlastiti izvor napajanja.

Ta dva uređaja kreiraju visoko frekvencijsko magnetsko polje između zavojnica uređaja. Jednom kada je polje uspostavljeno, dolazi do konekcije i razmjene informacija između čitača i oznake. Čitač šalje prvu poruku oznaci kako bi doznao vrstu komunikacije koju oznaka koristi, tip A ili tip B. Kada oznaka odgovori čitač šalje prvu naredbu koja mora odgovarati specifikaciji.

Oznaka nakon primitka naredbe provjerava je li ona ispravna. Ako nije, oznaka ne odgovara. U protivnom odgovara s zahtijevanom informacijom. Kod nekih komunikacija poput kartičnih plaćanja dolazi do uspostave sigurnog komunikacijskog kanala i sve informacije poslane su enkriptirane.

NFC oznake rade na principu da u jednom trenutku mogu samo primati ili slati informacije, dok čitači mogu primati informacije i slati naredbe istovremeno. Naredbe se šalju s čitača preko modulacijskog faznog podrhtavanja (eng. *phase jitter modulation*) (PJM) kako bi modificirao okružujuće polje i poslao signal. Oznaka odgovara koristeći induktivnost kako bi poslala naboj preko svojih zavojnica.

Prateći ove specifikacije osiguravamo da svi NFC uređaji mogu međusobno komunicirati. [3]

Slika 2.2: Unutrašnjost NFC oznake [12]

2.3. Sigurnosni problemi NFC tehnologije

Korisnici NFC tehnologije se naravno pitaju koji su sigurnosni rizici pogotovo oni koji ju koriste za kartična plaćanja. Jesu li njihove informacije sigurne i otporne na krađu? U nastavku su neki od sigurnosnih problema koji se mogu pojaviti i kako NFC tehnologija ih sprječava.

2.3.1. Prisluškivanje

Prisluškivanje se dešava kada osoba prati komunikaciju između čitača i oznaka. Nije potrebno pratiti svaki signal da bi se prikupila privatna informacija. Postoje dvije metode kako spriječiti prisluškivanje:

- udaljenost na kojoj NFC radi. Kako NFC radi na maloj udaljenosti prisluškivanje se može desiti na jako malom području.
- Sigurni kanali. Kada je uspostavljen sigurni kanal, sve informacije koje se razmjenjuju su enkriptirane i samo ovlašteni uređaji ih mogu dekriptirati. Potrebno je samo provjeriti koriste li čitaći sigurne kanale.

2.3.2. Korupcija i manipulacija podataka

Korupcija i manipulacija se dešavaju kada osoba manipulira informacijama koje se šalju čitaču ili posreduje informacijama tako da budu iskvarene i beskorisne kada dođu do čitača. Kako bi se spriječila korupcija i manipulacija koriste se sigurni kanali. Neki uređaji mogu prepoznati takve napade i spriječiti ih prije nego što se dogode.

2.3.3. Presretanje

Presretanje je slično manipulaciji podataka. Kod presretanja postoji posrednik koji čita, mijenja sve informacije koje se razmjenjuju između čitača i oznake. Ovakva vrsta napada je složena za izvesti i rjeđe se dešava. Kako bi se spriječio jedan uređaj mora djelovati aktivno, a drugi pasivno. To znači da jedan šalje, a drugi prima informacije umjesto da oba šalju i primaju informacije.

2.3.4. Krađa

Ako dođe do krađe mobilnog telefona ili kartice kradljivac može lako oponašati osobu te tako dobiti pristup plaćanjima i ulazima. Zato je potrebno upotrijebiti dodatne mjere

sigurnosti kao postavljanje sigurnosnih lozinka na svoje mobilne uređaje ili u slučaju kartica i oznaka prijaviti krađu svim administratorima sustava kod kojih je ta kartica ili oznaka zabilježena. [4]

2.4. Jedinstveni identifikacijski broj (UID)

Svaka NFC oznaka ima svoj globalno jedinstven, dobiven od strane proizvođača identifikacijski broj koji se može samo čitati (eng. *Unique identifier*) (UID). U većini slučajeva UID je 7 bajta duljine i prikazuje se u heksadekadskom obliku.. UID nije moguće mijenjati ili brisati. Spremljen je u posebnom dijelu memorije NFC oznake koja ne dozvoljava da se bitovi mijenjaju.

UID nije potpuno slučajan. Prvi bajt predstavlja proizvođača oznake, a ostatak je dodijeljen ovisno o proizvođaču. [13]

3. Sustav za kontrolu ulaza korištenjem beskontaktnih kartica

3.1. Motivacija i opis problema

Sustavi za kontrolu ulaza korištenjem beskontaktnih kartica danas su sve više prisutniji u današnjem društvu. Motivacija ovog rada je bila razviti vlastiti sustav koji je moguće lako integrirati i održavati. Sustav se može upotrebiti na vlastitim kućnim vratima, vratima prostorija i raznim područjima na kojima se treba provesti kontrola pristupa. Sustav zamjenjuje potrebu za nošenjem dodatnih ključeva što zamjenjuje beskontaktnim karticama i dozvoljava provedbu evidencije pristupa.

Sustav za kontrolu ulaza treba se sastojati od nekoliko dijelova:

- Softverskog dijela za autentifikaciju
- Hardverskog dijela za očitavanje NFC kartica
- NFC kartice
- Mehaničke brave za otvaranje ulaza

Potrebno je pronaći adekvatni uređaj na kojem bi se izvršavala autentifikacija te uređaj za čitanje kartica. Oba uređaja trebaju biti kompaktni, laki za montiranje i neinvazivni. Proces kojim bi sustav trebao raditi je:

- Korisnik prilaže NFC karticu u bliski domet čitača
- Čitač čita identifikacijsku informaciju s NFC kartice
- Čitač šalje informaciju uređaju za autentifikaciju
- Uređaj pomoću programa koji se izvodi na njemu radi autentifikaciju korisnika
- Program zapisuje evidenciju o čitanju
- Uređaj vraća čitaču informaciju ima li korisnik autorizaciju za otvaranje ulaza

– Čitač ovisno o odgovoru uređaja otvara mehaničku bravu ili signalizira korisniku da nema pristup

3.2. Korišteni razvojni alati i uređaji

3.2.1. Operacijski sustav GNU/Linux

GNU/Linux je operacijski sustav temeljen na Linux jezgri i GNU programskoj potpori te je temeljen na principima otvorenog koda. Jezgra je nastala 1991. godine od strane Linusa Torvaldsa. Razvijana je po uzoru na UNIX operacijski sustav.[14]

GNU je sloj iznad jezgre koji se sastoji od skupa programskih paketa koji omogućuju da zadnji aplikacijski sloj cijelog operacijskog sustava funkcionira. [15]

GNU/Linux danas pokreće većinu poslužiteljskih računala, mobilnih telefona, ugrađenih sustava i sve više osobnih računala. Raznovrsnost i rasprostranjenost ovog operacijskog sustava nam omogućuje da razvijamo aplikacije koje će se izvršavati na što više uređaja.

Operacijski sustav dolaze putem različitih distribucija. Neke od popularnih su: Debian, Ubuntu, SUSE, Red Hat Enterprise Linux te onih namijenjenih za slabija i manja računala poput Raspbiana temeljenog na Debianu.

Za razvoj ovog rada korištena je distribucija Ubuntu zbog svoje dobre programske i korisničke podrške. Za konačnu implementaciju i izvršavanje se koristi Raspbian koji se pokreće na Rapberry Pi-u.

3.2.2. Ugradbeno računalo Raspberry Pi

Raspberry Pi je serija računala malih dimenzija, velikih performansi i niske cijene. Ova računala se koriste zbog svojeg dizajna u razvoju, edukaciji i kao poslužitelji za male projekte.

U ovom radu korišten je Raspberry Pi 2 Model B [5], a njegove karakteristike su:

– 900MHz četverojezgreni ARM Cortex-A7 procesor

– 1GB RAM

– 100BASE Ethernet

– 4 USB priključka

– HDMI priključak

– Micro SD utor za karticu

Razlog odabira ovog računala je njegova mala veličina, jednostavnost korištenja i mala potrošnja električne energije. Na računalu se izvršava GNU/Linux distribucija naziva Raspbian.

3.2.3. NFC čitač Gemini 2000 Orbit IP

Gemini 2000 Orbit IP [6] je čitač NFC oznaka i uređaja. Čitač je napajan preko Ethernet-a (eng. *Power over Ethernet*) (PoE) standard IEEE 802.3af-2003 te se može koristiti PoE mrežni preklopnik ili aktivni 48V ubrizgač. Čitač također koristi Ethernet priključak za komunikaciju s web poslužiteljem.

Orbit IP radi kao samostalan web klijent te komunicira s web poslužiteljem tako da šalje HTTP zahtjeve i tada čeka povrati odgovor također u obliku HTTP zahtjeva.

Čitač podržava ISO 14443 Tip A i B oznake.

Čitač je odabran za ovaj rad zbog jednostavnog razvoja, jer se koristi jednostavna HTTP komunikacija s jednostavnim naredbama, te mogućnosti da se rade složeniji sustavi s više čitača jer se spaja u mrežu.

Kod implementacije rada napravljena je mreža samo između Raspberry Pi-a i čitača te tako cijeli sustav ostaje izoliran.

Slika 3.1: Gemini 2000 Orbit IP [11]

3.2.4. Programski jezik Python 3

Python [7] je interpretativni programski jezik nastao od strane Guido van Rossuma 1991. godine. Najčešće je zbog svoje jednostavnosti i proširivosti različitim bibliotekama popularan u različitim područjima razvoja. Neka od područja su:

- Skriptiranje - kao interpretativni jezik koristi se za podešavanje sustava te pisanje jednostavnih skripti za različite namjene
- Matematika - zbog velikog izbora matematičkih biblioteka Python je jezik koji se često koristi za statistiku i strojno učenje
- Razvoj web aplikacija - poslužiteljski dio aplikacija sve više je razvijan u Pythonu. Interpretativni način rada iako nije najbrži dozvoljava brži razvoj i testiranje aplikacija. Također zbog mnogo razvojnih okvira koji nude brzi razvoj i proširivost

Za razvoj ovog rada odabran je Python zbog svoje rasprostranjenosti, količine biblioteka i neovisnosti o sustavu na kojem se izvršava. Python također karakterizira i čitljivost koda te kao takav je savršen za razvoj manjih projekata.

3.2.5. Baza podataka TinyDB

TinyDB [8] je mala baza podataka orijentirana na dokumentima. Podaci se spremaju u jednu datoteku koja je zapisana u JSON formatu. Zbog malog broja informacija koje se moraju spremati u bazu, nema potrebe za bazama podataka koje imaju više mogućnosti, brže su, ali i zauzimaju više memorije kod izvršavanja.

3.3. Arhitektura i dizajn sustava

U ovome dijelu rada objašnjena je cijela struktura sustava za kontrolu ulaza korištenjem beskontaktnih kartica. Sustav se sastoji od:

- sustava za autentifikaciju
- Orbit IP čitača

Sustav je zamišljen kao jednostavan sustav za kontrolu ulaza koji omogućuje autentifikaciju preko beskontaktnih kartica koje sadrže NFC tehnologiju. Naravno postoji mogućnost autentifikacije i mobilnim telefonima koji podržavaju NFC.

3.3.1. Struktura programske podrške

Programska podrška se sastoji od skripte za pokretanje aplikacije za kontrolu ulaza, aplikacije za kontrolu ulaza, aplikacije za unošenje identifikatora u bazu podataka i baze podataka.

```
NFC_Handler
├── database_input.py
├── db.json
├── nfc_handler.py
└── startup.sh
```

Slika 3.2: Struktura programske podrške

3.3.2. Baza podataka

Baza podataka je jednostavna datotečna baza u JSON formatu. Nalazi se u datoteci **db.json**.

JSON je tekstualni format koji je jezično neovisan, ali koristi konvencije koje su poznate većini programera.

JSON podaci se spremaju u obliku {'ime' : 'vrijedost'} koje predstavljaju jedan objekt. Objekti su međusobno odijeljeni zarezima.

Kod baze podataka ovog rada postoji glavni objekt **_default** koji predstavlja korijen te sadrži objekte nizane slijednim brojevima. Svaki broj sadrži objekt imena **uid** (eng. *Unique ID*) i vrijednost koja predstavlja identifikacijski broj NFC oznake. Identifikacijski broj oznake sadrži vrijednosti zapisane u **heksadecimalnom** formatu, a veličina identifikatora ovisi o proizvođaču, ali je najčešće veličine 7 bajta.

```
1 {"_default": {
2   "1": {"uid": "AEF056"},
3   "2": {"uid": "123456"},
4   "3": {"uid": "555555"},
5   "4": {"uid": "31053B71"},
6   "5": {"uid": "123456"}
7 }}
```

Listing 3.1: Primjer izgleda datoteke db.json

3.3.3. Autentifikacija

Autentifikacija korisnika se provodi preko aplikacije **nfc_handler.py**.

Tijek izvršavanja programa je sljedeći:

- Kreiranje HTTP servera
- Učitavanje baze podataka
- Čekanje na zahtjev čitača
- Korisnik prilaže NFC karticu
- Čitač pošalje zahtjev
- Aplikacija parsira dobiven zahtjev
- Aplikacija provjerava u bazi podataka
    - Postoji UID, vrati čitaču da otvori vrata
    - Ne postoji UID, vrati čitaču da ne otvara vrata
    - Nema UID u poslanom zahtjevu, pošalji prazan odgovor
- Vrati se na čekanje zahtjeva

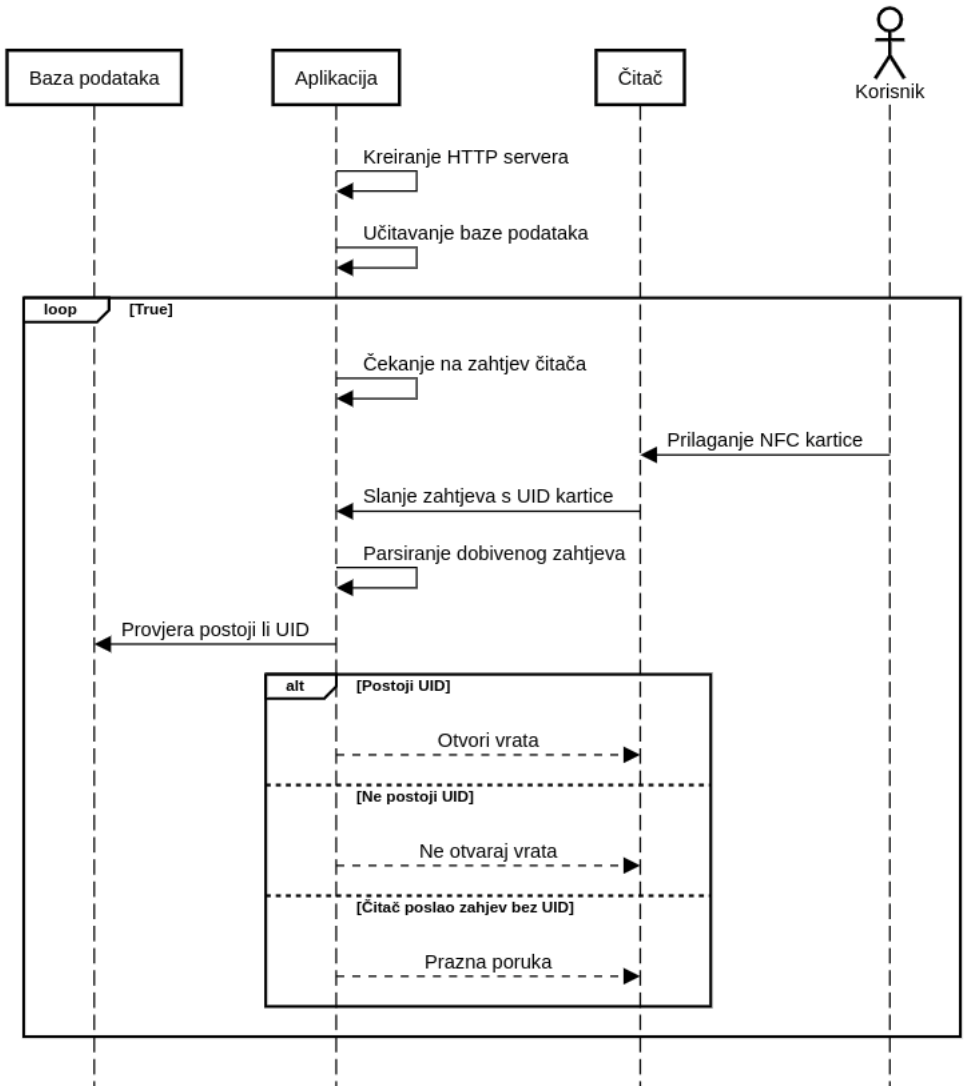
Inicijalizacija aplikacije

Aplikacija se inicijalizira uključivanjem iz biblioteka:

- **http.server** klase:
    - HTTPServer
    - BaseHTTPRequestHandler
- **tinydb** klase
    - TinyDB
    - Query
- **datetime**, klasu datetime
- **socket**

Postavlja **HOST_NAME** i **PORT_NUMBER**. Te varijable su postavljene na IP adresu '192.168.7.191' i vrata 80 jer čitač zadano šalje svoje zahtjeve na tu adresu i vrata.

Dolazi do inicijalizacije web poslužitelja i dohvaćanje baze podataka. Nakon toga aplikacija čeka na zahtjeve od čitača.



Slika 3.3: Sekvencijski dijagram sustava

Parsiranje zahtjeva

Kada aplikacija dobije zahtjev od čitača, čitač sve informacije šalje putem URL-a. Zato je potrebno parsirati URL i pronaći UID. Za to služi funkcija **handle_nfc**.

UID se pronalazi prvo traženjem njegove duljine u bajtima, a zatim se traži identifikacijski broj.

Funkcija vraća ili duljinu UID-a i UID ili vrijednosti False, False ovisno o uspješnosti pronalaska istih u zahtjevu.

Provjera baze podataka

Nakon parsiranja zahtjeva dolazimo do provjere postoji li dobiveni UID u bazi podataka. Koristimo funkciju **handle_nfc**.

```
#!/usr/bin/python3
from http.server import HTTPServer, BaseHTTPRequestHandler
from tinydb import TinyDB, Query
from datetime import datetime
import socket

HOST_NAME = '192.168.7.191'
PORT_NUMBER = 80

try:
    server = HTTPServer((HOST_NAME, PORT_NUMBER), Server)
    print ("%s Started httpserver on port %d" % (str(datetime.now()), PORT_NUMBER))
    db = TinyDB('db.json')

    server.serve_forever()

except KeyboardInterrupt:
    print ('^C received, shutting down the web server')
    server.socket.close()

```

Slika 3.4: Inicijalizacija aplikacije

```
def get_uid(url):
    if "ulen" in url:
        uid_len = int(url[url.find("ulen") + 5])
        uid_start = int(url.find("uid")) + 4
        uid = url[uid_start:(uid_start + uid_len * 2)]
        return (uid_len, uid)
    else:
        return (False, False)

```

Slika 3.5: Parsiranje zahtjeva

Provjera se izvršava preko instancirane baze **db** i njene funkcije **search**.

Ako postoji UID u bazi zabilježavamo vrijeme, otvaranje vrata i UID te vraćamo **'Open'**.

Ako ne postoji UID u bazi zabilježavamo vrijeme, neautoriziran pristup i UID te vraćamo **'Close'**.

Ako u zahtjevu nije postojao UID, tj. čitač je poslao svoj dijagnostički zahtjev, vraćamo **'PING'**.

```
def handle_nfc(url):
    uid_len, uid = get_uid(url)
    print ('Uid_len:', uid_len)
    print ('Uid:', uid)
    if uid_len is not False:
        Uid = Query()
        if db.search(Uid.uid == uid):
            print ("%s Door opening. UID: %s" % (str(datetime.now()), uid))
            return 'Open'
        else:
            print("%s Unauthorized access. UID: %s" % (str(datetime.now()), uid))
            return 'Close'
    else:
        return 'PING'

```

Slika 3.6: Provjera baze podataka

Obrada zahtjeva

Klasa Server čeka HTTP zahtjev od strane čitača. Kada čitač pošalje zahtjev klasa šalje URL zahtjeva prethodno opisanoj funkciji **handle_nfc**.

Kada funkcija vrati povratnu informaciju ovisno o njoj šalje čitaču odgovor.

Nakon slanja odgovora aplikacija se vraća u stanje čekanja zahtjeva.

```
HTTP/1.0 200 OK
Server: BaseHTTP/0.6 Python/3.6.7
Date: Mon, 10 Jun 2019 14:11:13 GMT
Content-type: text/html

<ORBIT>
GRNT=05
UI=820432

</ORBIT>^C
```

Slika 3.7: Odgovor kod uspješne autentifikacije

```
HTTP/1.0 200 OK
Server: BaseHTTP/0.6 Python/3.6.7
Date: Mon, 10 Jun 2019 14:10:07 GMT
Content-type: text/html

<ORBIT>
DENY=05
UI=A00332

</ORBIT>^C
```

Slika 3.8: Odgovor kod neuspješne autentifikacije

```
HTTP/1.0 200 OK
Server: BaseHTTP/0.6 Python/3.6.7
Date: Mon, 10 Jun 2019 14:12:39 GMT
Content-type: text/html

<ORBIT>RLY=1
UI=000000

</ORBIT>
```

Slika 3.9: Odgovor kod dijagnostičkog zahtjeva

```
11 class Server(BaseHTTPRequestHandler):
12     def do_GET(self):
13         command = handle_nfc(self.path)
14 
15         if command is 'Open':
16             self.send_response(200)
17             self.send_header('Content-type','text/html')
18             self.end_headers()
19             self.wfile.write("<ORBIT>\nGRNT=05\nUI=820432\n\n</ORBIT>".encode("utf-8"))
20 
21         elif command is 'Close':
22             self.send_response(200)
23             self.send_header('Content-type','text/html')
24             self.end_headers()
25             self.wfile.write("<ORBIT>\nDENY=05\nUI=A00332\n\n</ORBIT>".encode("utf-8"))
26 
27         elif command is 'PING':
28             self.send_response(200)
29             self.send_header('Content-type','text/html')
30             self.end_headers()
31             self.wfile.write("<ORBIT>RLY=1\nUI=000000\n\n</ORBIT>".encode("utf-8"))
32 
33         return
34 
```

Slika 3.10: Obrada zahtjeva

3.3.4. Integrirano rješenje za kontrolu ulaza

Čitač i uređaj na kojoj se pokreće poslužiteljska aplikacija, Raspberry Pi Model 2 B, komuniciraju putem Ethernet priključka.

Čitač je i napajan preko Ethernet priključka te je za njegov rad potreban PoE mrežni preklopnik ili PoE ubrizgivač. Kod implementacije rada koristio se Mikrotik Gigabit PoE ubrizgivač u koji se dovodi upredena parica i 48V istosmjerni pretvornik.

Slika 3.11: Mirkotik Gigabit PoE ubrizgivač [9]

Čitač i Raspberry Pi povezani su međusobno bez upotrebe usmjeritelja. Zato je potrebno na Raspberry Pi-u uspostaviti mrežu. Mreža se uspostavlja preko skripte **startup.sh**.

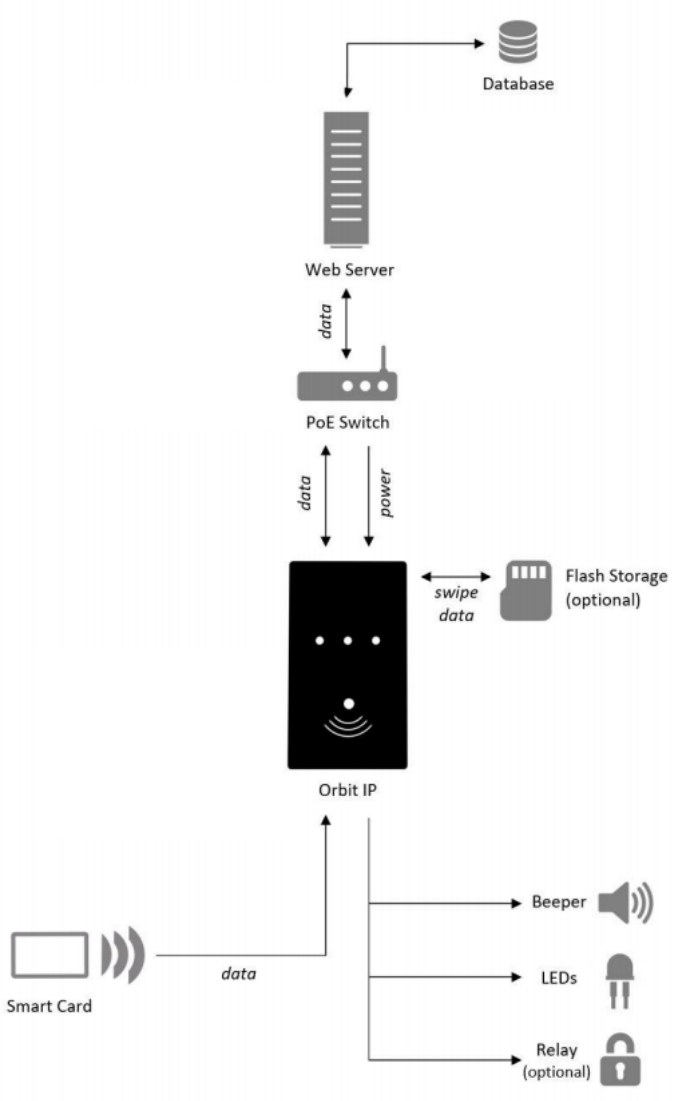
Skripta također služi za instalaciju baze podataka **TinyDB** i pokreće poslužiteljsku aplikaciju **nfc_handler.py**.

```
1 #!/bin/bash
2
3 pip3 install tinydb
4
5 sudo ip ad add 192.168.7.191/24 dev enp2s0
6
7 sudo python3 nfc_handler.py
```

Slika 3.12: Bash skripta startup.sh

Zadnja dio sustava je elektronička brava koja otvara bravu kada dobije napon od 12V. Napon dobiva iz čitača koji putem releja kod odgovora od poslužiteljske aplikacije daje napon kroz žice.

Slika 3.13: Elektronička brava [10]



Slika 3.14: Shema sustava

4. Zaključak

Korištenje novih tehnologija sve je izraženije u našem društvu. Tako i jedna od stvari koja se može unaprijediti je i kontrola ulaza. Čovječanstvo je kontrolu ulaza od antičkog vremena provodilo jednostavnim ključem i bravom.

Dolaskom tehnologija radio frekvencijske identifikacije moguće je zamijeniti takav star i primitivan sustav jednostavnijim sustavom sa strane korisnika. Korisnik više nema potrebu nositi ključeve za svaki ulaz pojedinačno nego može pomoću jedne centralne kartice ili mobilnog uređaja dobiti pristup ulazu. Takvi uređaji nalaze se u džepu većine osoba te nema potrebe za izdavanjem novih.

Ovakav sustav je jednostavno implementirati postavljanjem čitača, računala za poslužitelj i elektroničke brave. Također moguće je implementacija puno većih sustav jer se ovakav tip čitača spaja na mrežu te je moguće putem jednog poslužitelja napraviti sustav u kojem se može kontrolirati ulaz cijele zgrade.

Jedan od nedostataka ovakvog sustava je da u slučaju krađe NFC oznake ili mobilnog uređaja kradljivac ima pristup svim ulazima kao i prvobitni korisnik. Takvom nedostatku se primijeniti još dodatna autentifikacija lozinkom, što ovaj sustav ne podržava.

Konačan zaključak ovog rada je da je ovakva vrsta kontrole ulaza optimalno rješenje za mjesta gdje puno korisnika mora imati pristup, ali i gdje treba provesti evidenciju dolaska i odlaska. Vrlo je jednostavan za implementaciju i smanjuje problem kasnijeg održavanja, a dozvoljava proširenje samog sustava.

5. Literatura

[1] What is NFC? Near Field Communication Explained, Mar 2017. URL http://nearfieldcommunication.org/about-nfc.html. [Online; accessed 1. Jun. 2019].

[2] History of Near Field Communication - NearFieldCommunication.org, Mar 2017. URL http://nearfieldcommunication.org/history-nfc.html. [Online; accessed 1. Jun. 2019].

[3] Near Field Communication Technology Standards – NearFieldCommunication.org, Mar 2017. URL http://nearfieldcommunication.org/technology.html. [Online; accessed 2. Jun. 2019].

[4] Security Concerns with NFC Technology - NearFieldCommunication.org, Mar 2017. URL http://nearfieldcommunication.org/nfc-security.html. [Online; accessed 2. Jun. 2019].

[5] Buy a Raspberry Pi 2 Model B – Raspberry Pi, Jun 2019. URL https://www.raspberrypi.org/products/raspberry-pi-2-model-b. [Online; accessed 4. Jun. 2019].

[6] Orbit IP: PoE Contactless NFC Card Reader - Gemini 2000, Jun 2019. URL http://www.gemini2k.com/orbit-ip-poe-nfc-smart-card-reader. [Online; accessed 5. Jun. 2019].

[7] Welcome to Python.org, Jun 2019. URL https://www.python.org. [Online; accessed 5. Jun. 2019].

[8] Welcome to TinyDB! — TinyDB 3.13.0 documentation, May 2019. URL https://tinydb.readthedocs.io/en/latest. [Online; accessed 5. Jun. 2019].

[9] MikroTik, Jun 2019. URL https://mikrotik.com/product/RBGPOE. [Online; accessed 7. Jun. 2019].

[10] Jun 2019. URL http://www.ellabo.hr/pd/elektro-prihvatnik-8-12v-deblokada-senzor/ET1719015/g/205/l/HR#tab-longDesc. [Online; accessed 7. Jun. 2019].

[11] Orbit IP: PoE NFC Reader - Gemini 2000, Jun 2019. URL http://www.gemini2k.com/shop/nfc-devices/orbit-ip-poe-nfc-card-reader. [Online; accessed 5. Jun. 2019].

[12] NFC Marketing | SmsNation Mobile, Jun 2019. URL http://smsnation.co.rw/nfc-marketing. [Online; accessed 2. Jun. 2019].

[13] NFC UID - GoToTags Help, Jun 2019. URL https://help.gototags.com/article/nfc-uid. [Online; accessed 4. Jun. 2019].

[14] What is Linux?, Jun 2019. URL https://www.linux.com/what-is-linux. [Online; accessed 4. Jun. 2019].

[15] gnu.org, Jun 2019. URL https://www.gnu.org. [Online; accessed 4. Jun. 2019].

[16] Blue Bite. The Promising Future of NFC. *Medium*, May 2018. URL https://medium.com/blue-bite/the-promising-future-of-nfc-d172f6905b7b. [Online; accessed 1. Jun. 2019].

Kontrola ulaza korištenjem beskontaktnih kartica

Sažetak

U ovom radu bavi se problematikom implementacije sustava za kontrolu ulaza korištenjem beskontaktnih kartica. Objašnjeni su korišteni alati za razvoj te svi uređaji koji se koriste u implementaciji sustava. Objašnjena je NFC tehnologija, njena povijest, njen standard donesen od strane NFC Foruma, sigurnosni problemi koje rješava i način identifikacije. Prikazana je arhitektura cijelog sustava od baze podataka, poslužiteljske aplikacije, način funkcioniranja čitača i funkcioniranje cijelog sustava zajedno.

Ključne riječi: Kontrola ulaza, Beskontaktne kartice, Autentifikacija, NFC, Python, Čitač kartica

Access control using contactless cards

Abstract

This paper deals with the issue of implementing an access control system using contactless cards. The development tools used and all devices used in the implementation of the system are explained. The near-field communication technology, its history, its standard developed by the NFC Forum, the security issues it solves and the way the NFC tags are identified are explained. The architecture of the entire system is presented from the database, the server application, how the reader functions, and how the whole system functions together.

Keywords: Access control, Contactless cards, Near-field communication, NFC, Python, Card reader