

Seminarska naloga pri predmetu računalništvo

# RIPtide - Univerzalno orodje za konfiguracijo omrežij

Mentor: Marko Kastelic, prof.

Avtor: Jurij Fortuna , G 3. a

Ljubljana, 22. maj 2023

## **Povzetek**

V tej seminarski nalogi je opisana programska oprema RIPTide. Je odprtokodna in omogoča komurkoli, da s pomočjo RIPTide APIja ustvari podporo za svojo omrežno opremo ali uporablja že napisano.

**Ključne besede:** omrežna oprema, API, Java, vtičniki, odprtokodna programska oprema

## **Abstract**

This seminar task describes the RIPTide software. It is open-source and allows anyone to create support for their network equipment or use what has already been written, using the RIPTide API.

**Keywords:** network equipment, API, Java, plugins, open source software

# Kazalo

<b>1</b>	<b>Uvod</b>	<b>4</b>
<b>2</b>	<b>Uporabljene tehnologije</b>	<b>5</b>
<b>3</b>	<b>Ospredni del</b>	<b>6</b>
3.1	Konfiguracija . . . . .	6
3.1.1	Shranjevanje podatkov . . . . .	6
3.1.2	Barve vmesnika . . . . .	7
3.2	Upravljanje poverilnic in naprav . . . . .	7
3.3	Nalaganje Shedov . . . . .	7
<b>4</b>	<b>Aplikacijski programski vmesnik</b>	<b>8</b>
4.1	Dostop do API-ja . . . . .	8
4.2	Struktura API-ja . . . . .	8
4.3	Vgrajena orodja . . . . .	8
4.4	Komunikacija z napravami . . . . .	8
4.5	Čarovniki . . . . .	9
<b>5</b>	<b>Shedi</b>	<b>10</b>
5.1	Projektna struktura . . . . .	10
5.2	Metapodatki . . . . .	10
5.3	. . . . .	10
<b>6</b>	<b>Eksperimentalne funkcije</b>	<b>11</b>
6.1	Topološki pogled omrežja . . . . .	11

# 1 Uvod

Ideja za razvoj RIPTide-a se je pojavila, ko sem bil med konfiguracijo domačega omrežja prisiljen uporabljati tri različne programske opreme, različnih proizvajalcev. Med sabo so se nemalo razlikovale in so bile po večini nestabilne. Zato sem se odločil razviti generično programsko opremo za konfiguracijo omrežij. Deluje na principu “vtičnikov” (t.i. Shedov) za posamezne kose omrežne opreme.

The screenshot shows the Telekom Slovenije configuration interface. At the top, there's a navigation bar with icons for STATUS, WI-FI, OMREŽJE, APLIKACIJE, and SISTEM. Below this is a breadcrumb trail: Home > STATUS > Informacije. On the left, there's a sidebar menu with options like Stanje omrežja, Aktivni odjemalci, Dnevnik klicev, Usmerjevalna tabela, IPv6, Podatki o USB, 3G/LTE Information, and 5G Information. The main content area is titled 'Informacije' and contains two sections: 'Splošno' (General) and 'Status interneta' (Internet Status). The 'Splošno' section lists various hardware and software details, while the 'Status interneta' section shows connection status and network parameters.

Serijska številka naprave:	0000000000
Serijska številka GPON:	0000000000000000
MAC-naslov naprave:	00:00:00:00:00:00
Verzija programa:	1.1.1318
Model:	Innbox G78
Proizvajalec:	Iskrate
Sprejemna moč optičnega signala:	-14 dBm
Trajanje delovanja:	11 dni 2 ur 59 minut 18 sekund

Vmesnik:	ppp0.1 (Internet)
Status omrežja:	Nenastavljeno
Trajanje povezanosti:	0 dni 0 ur 0 minut 0 sekund
Trajanje povezanosti vmesnika:	11 dni 2 ur 58 minut 22 sekund
Števec vzpostavitev linije:	1
Naslov MAC:	00:00:00:00:00:00
Naslov IP:	
Omrežna maska:	255.255.255.255
Prehod:	
Primarni strežnik DNS:	

Slika 1: Primer konfiguracijske strani za usmerjevalnike Telekoma Slovenije

## **2 Uporabljene tehnologije**

Tako Shedi, kot tudi RIPTide so napisani v Javi. Java omogoča dinamično nalaganje modulov ob zagonu navideznega stroja, tudi iz zapakiranih JAR datotek. Sam RIPTide je zgrajen iz dveh glavnih delov: osrednega dela in aplikacijskega programskega vmesnika (v nadaljevanju API). Poleg tega velja omeniti Shede, ki jih lahko razvije kdorkoli z uporabo RIPTide API-ja.

## 3 Ospredni del

Ospredni del je odgovoren za konfiguracijo RIPTide-a, rokovanje z uporabnikovimi napravami ter pripadajočimi poverilnicami in za nalaganje Shedov.

### 3.1 Konfiguracija

Uporabnik lahko svoje naprave shrani, in si s tem prihrani čas za morebitno kasnejšo konfiguracijo. Poleg tega RIPTide omogoča spremembo barve vmesnika po uporabnikovi želji.

#### 3.1.1 Shranjevanje podatkov

Okoljske spremenljivke, naprave in poverilnice so zapisane v objektu imenovanem `Workspace`.

```
@Data
public class Workspace implements Serializable {
    private Theme theme;
    private ArrayList<Credential> credentials;
    private ArrayList<Connection> connections;

    public Workspace() {
        theme = Theme.PRIMER_DARK;
        credentials = new ArrayList<>();
        connections = new ArrayList<>();
    }

    public Workspace(Theme theme, ArrayList<Credential>
        credentials, ArrayList<Connection> connections) {
        this.theme = theme;
        this.credentials = credentials;
        this.connections = connections;
    }
}
```

Ko uporabnik nastavitve shrani je objekt serializiran in zapisan v datoteko na uporabnikov sistem. Datotek je lahko več, kar omogoča prilagoditev vmesnika za različna okolja (npr. posebej za domačo in poslovno rabo). `Workspace` datoteke

so shranjene v JSON formatu, kar uporabnikom omogoča enostavno izmenjavo ali prenos.

### **3.1.2 Barve vmesnika**

Uporabnik lahko vmesnik prilagodi na štiri priložene teme. V prihodnosti bo mogoče nalaganje barv iz CSS datoteke po meri.

## **3.2 Upravljanje poverilnic in naprav**

Za konfiguracijo večine omrežnih naprav je potrebna avtorizacija. RIPTide uporabnikom omogoča varno shranjevanje poverilnic na njihovem sistemskem keyringu. Poverilnice so ob povezavi na napravo prek API-ja podani Shedu, kar pomeni, da se razvijalci teh ne rabijo ukvarjati z varnim hranjenjem poverilnic. Več o roko-  
vanju z napravami v razdelku 4.4.

## **3.3 Nalaganje Shedov**

Nalaganje shedov poteka v treh korakih, lociranje Sheda, branje metapodatkov in vzpostavljanje povezave z napravo. Na uporabnikovem sistemu se nahajajo v obliki JAR datotek v dveh mapah, `~/.config/RIPTide/sheds` na NIX sistemih in `%UserProfile%\RIPTide\sheds` na Windows sistemih.

Ob izbiri naprave, RIPTide najprej prebere njen t.i. model path, ki izgleda približno tako: `telekomslovenije.innboxg78.ts`. Niz je sestavljen iz ID-ja Sheda, ID-ja modela naprave in ID-ja različice modela. Vsi ID-ji so edinstveni, kar pomeni, da lahko program najde Shed, v njem poišče model naprave, njegovo različico in vzpostavi povezavo z napravo prek API-ja.

## 4 Aplikacijski programski vmesnik

API omogoča uporabnikom, da ustvarijo lastno podporo za omrežno opremo, ki je programska oprema še ne podpira. To pomeni, da lahko skrbniki omrežij enostavno integrirajo nove naprave v svoje omrežje, ne da bi morali čakati, da prodajalec izda uradno podporo.

### 4.1 Dostop do API-ja

Dostop do APIja je mogoč preko SDK knjižnice. Nahaja se v RIPTide Maven repozitoriju. Ker so Javanski upravitelji paketov med seboj bolj kot ne kompatibilni, je tudi postopek dodajanja knjižnice med njimi zelo podoben. Več o projektni strukturi pa v razdelku 5.1.

### 4.2 Struktura API-ja

Vsi pomembni vmesniki in razredi za pisanje Shedov se nahajajo v paketu `org.riptide.sdk.sheds`. Paket je deljen na podpakeete za uporabniški vmesnik, tipe naprav, čarovnike in avtentikacijo.

### 4.3 Vgrajena orodja

API vsebuje uporabna orodja za izdelovanje statusnih strani in formularjev, ki se nahajajo v paketu `org.riptide.sdk.sheds.ui`.

### 4.4 Komunikacija z napravami

Osredni del s Shedi komunicira preko vmesnikov. Za primer vzemimo Telekomov usmerjevalnik. Program iz metapodatkov razbere, da naprava tipa router. To pomeni, da mora njen gonilni razred implementirati vmesnik `Router` in posledično `Device`, saj ga `Router` razširja.

```
public interface Device {  
    void initialize(String address, Credential credential);  
    TreeMap<String, Tab[]> getPages();  
}
```



```
public interface Router extends Device {  
    PATWizard patWizard();  
}
```

Ob izbiri naprave se ustvari nov objekt, ki je odgovoren za komunikacijo z napravo (postopek je opisan v razdelku 3.3). Za tem se kliče metoda `initialize`, ki shedu preda omrežni naslov naprave in poverilnice.

## 4.5 Čarovniki

## **5 Shedi**

V prihodnosti načrtujem omogočiti razširitev tudi drugih funkcionalnosti. Zaradi tega sem se za poimenovanje vtičnikov naprav odločil uporabiti izraz šhedi”namesto običajnega izraza ”plugin”.

### **5.1 Projektna struktura**

### **5.2 Metapodatki**

### **5.3**

## **6 Eksperimentalne funkcije**

RIPtide je opremljen z nekaj novimi eksperimentalnimi funkcijami, ki pa jih zaradi pomanjkanja časa nisem uspel povsem dokončati in stestirati. Te omogočajo naprednim uporabnikom nove možnosti avtomacije in pregleda nad omrežjem.

### **6.1 Topološki pogled omrežja**