

AUTOMATIZACIJA MREŽNIH UREĐAJA

AUTOMATING NETWORK DEVICES

Barlović Filip¹, Dubravko Žigman², Valter Perinović²

¹Student TVZ-a diplomirao 2019.

²Tehničko Veleučilište u Zagrebu, Vrbik 8, Zagreb, Hrvatska

SAŽETAK

Mrežni sustavi današnjice sve su složeniji zbog velikog broja elektroničkih uređaja između kojih je potrebno uspostaviti komunikaciju. Veoma je teško kvalitetno upravljati složenim sustavom mrežnih uređaja te se iz tog razloga implementiraju sustavi automatizacije. Automatizacija mrežnih uređaja sljedeći je logični korak prema bržem i učinkovitijem načinu upravljanja. Potpuna automatizacija mrežnog sustava moguća je isključivo koristeći moderne mrežne uređaje koji podržavaju potrebna programska sučelja. Mali broj sustava podržava potpunu automatizaciju sustava te se za većinu slučajeva koristi jednostavnije rješenje koje podržava sve generacije mrežnih uređaja. Ovaj rad prikazuje automatizaciju Cisco mrežnih uređaja koristeći programski jezik Python. Rad sadrži tri primjera koja prikazuju neke od mogućih načina automatizacije uređaja kroz programske skripte.

Ključne riječi: mrežni uređaji, automatizacija mrežnih uređaja, Python, Cisco

ABSTRACT

The complexity of modern network systems has increased dramatically due to the increase in number of network devices in an individual system. Managing a complex network manually is very difficult and because of it, implementation of network automatization has become necessary. Network automation is the next logical step towards a simpler and more efficient network management. Automating the entire network system is only possible using modern network devices that support use of application programming interfaces.

Only a small number of network systems can be fully automated, most systems can use a simpler automation method that support every network device generation. This paper demonstrates automatization of Cisco network devices using Python programming language. The paper presents three examples of network device automation that show automation possibilities.

Keywords: network devices, automating network devices, Python, Cisco

1. UVOD

1. INTRODUCTION

Mrežna tehnologija jedna je od najvažnijih otkrića zbog mogućnosti povezivanja uređaja na globalnoj razini i unaprjeđenja načina razmjene komunikacije. Broj mrežnih uređaja eksponencijalno raste što otežava upravljanje i nadzor uređaja u sustavu. Izričito je teško kvalitetno upravljati pojedinim uređajem u mrežnim sustavima sa stotinama ili tisućama mrežnih uređaja. Ako nastane problem na mreži, potrebno je prvo locirati problem te onda riješiti uzrok problema što nije jednostavno u velikim mrežama. Današnji ubrzani svijet zahtjeva dinamične sustave koji automatski reagiraju na kvar te ispravljaju problem kako bi novčani gubitak bio što manji.

Tehnologija mrežne automatizacije predstavlja brzo i dinamično mijenjanje postavki mrežnih uređaja te uklanjanje nastalih kvarova. Automatizacija se odrađuje kroz sustave potpune automatizacije ili programske skripte. Sustavi potpune automatizacije zahtijevaju središnji uređaj koji kontrolira sve mrežne uređaje te izravno upravlja sustavom i većinom njegovih funkcija.

Za ovu vrstu automatizacije potrebni su mrežni uređaji koji podržavaju programska sučelja preko kojih se vrši komunikacija sa središnjim uređajem. Tehnologija potpune automatizacije mrežnog sustava pojavila se nedavno što znači da iznimno mali broj sustava posjeduju moderne mrežne uređaje koje podržavaju programska sučelja. Za veliku većinu mrežnih sustava automatizacija je moguća isključivo putem razvojnih okolina ili programskih skripti. Navedene tehnologije automatiziraju uređaje putem upravljanja na daljinu koristeći SSH (engl. Secure Shell) ili telnet protokol. [1]

Na početku rada prikazat će se i opisati dvije razvojne okoline. U ostatku rada prikazana su tri praktična primjera automatizacije koristeći programske skripte jezika Python. Python je zbog svoje jednostavnosti i mogućnosti iznimno popularan programski jezik u području automatizacije. Veliki broj javno dostupnih modula olakšava zadatke koje je potrebno odraditi.

2. USPOREDBA RAZVOJNIH OKOLINA

2. *COMPARING DEVELOPMENT ENVIRONMENTS*

Veliki broj korisnika razumije prednosti mrežne automatizacije ali ne posjeduju znanja potrebna za samostalno odrađivanje automatizacije. Razvojne okoline na jednostavan način omogućavaju automatizaciju putem grafičkih sučelja. U nastavku se uspoređuju dvije popularne razvojne okoline, SolarWinds i APIC-EM (engl. Application Policy Infrastructure Controller Enterprise Module).

SolarWinds je razvojna okolina s velikim rasponom mogućnosti i podrškom za većinu mrežnih uređaja. SolarWinds je podijeljen na zasebne programe gdje korisnik odabire potrebne usluge. Programi su specijalizirani za nadgledanje sljedećih dijelova sustava: mrežne aktivnosti, poslužitelja i programa, baze podataka te sigurnosti. Osim nadgledanja sustava moguće je automatizirati sljedeće usluge: ažuriranje ugrađenih programa (engl. firmware), trenutna izvješća o stanju mreže, definiranje pravila kojih se uređaji moraju pridržavati te generiranje

skripti u slučaju potrebnih izmjena, dopuštanje izmjene konfiguracije određenim korisnicima i još mnogo drugih. SolarWinds je odlična razvojna okolina za nadgledanje i uvođenje osnovne razine automatizacije u sustav koji sadrži mrežne uređaje različitih proizvođača. Međutim, održavanje sustave s mrežnim uređajima različitih proizvođača povećava mogućnost greške ili kvara u sustavu. Iz tog razloga se češće koristi isti proizvođač mrežne opreme u cijelom sustavu, a SolarWinds ne pruža dovoljnu dubinu mogućnosti kao razvojna okolina proizvođača mrežne opreme. Za mrežne sustave u kojima se upotrebljavaju isključivo Cisco mrežni uređaji iz tog razloga koriste APIC-EM[2].

APIC-EM je razvojna okolina namijenjena Cisco mrežnim uređajima. Automatizira postavljanje i upravljanje fizičkim te virtualnim mrežnim servisima raznovrsnih platformi. Pruža automatsku konfiguraciju novih uređaja bez korisničke akcije, prati put paketa po mreži te kroz grafičko sučelje prikazuje lokaciju i razlog mogućih problema. Pojednostavljuje postavljanje QoS (engl. Quality of Service) pravila koja pružaju bolje uvijete određenim programima ili određenom tipu podatka na mreži. Grafičko sučelje pruža jednostavan odabir prioriteta određenog programa ili tipa podataka i prevodi odabir korisnika u naredbe koje se primjenjuju na potrebnim mrežnim uređajima. APIC-EM također pruža usluge upravljanja IWAN (engl. Intelligent Wide Area Network) vezama, izradu i licenciranje virtualnih mrežnih uređaja ili servisa te sigurnosne preporuke proizvođača[3].

3. PRIMJERI AUTOMATIZACIJE MREŽNIH UREĐAJA

3. *EXAMPLES OF NETWORK DEVICE AUTOMATION*

Navedene razvojne okoline su dobar korak u početak automatizacije mrežnih uređaja, ali nisu potpuno prilagodljivi korisničkim potrebama. Korisnicima sa znanjem programiranja na raspolaganju je potpuno prilagodljiv način automatizacije mreže koristeći programske skripte. Ovaj dio rada prikazuje tri primjera automatizacije korištenjem Python programskog jezika.

3.1. AUTOMATIZACIJA PREKLOPNIKA POMOĆU MICROSOFT EXCELA

3.1. AUTOMATING A SWITCH USING MICROSOFT EXCEL

Potreba za mijenjanjem konfiguracije preklopnika događa se na dnevnoj bazi zbog dodavanja novih računala ili promjene unutar mrežnog sustava. Zbog stalnih promjena konfiguracije važno je imati ažurnu dokumentaciju mrežnog sustava. Najjednostavniji način praćenja trenutnog stanja važnih informacija je putem Microsoft Excel programa. Ovaj primjer prikazuje promjenu konfiguracije mrežnog uređaja ovisno o korisnički stvorenoj Microsoft Excel tablici. Slika 1. prikazuje informacije sadržane u tablici.

	A	B	C	D
1	S2			
2	IP adresa	192.168.21.2		
3				
4	Sučelje	Ethernet 0/0	Ethernet 0/2	
5	VLAN	101		
6	TIP	Trunk	Access	
7	STATUS	UP	DOWN	
8	OPIS	Veza prema R2	Opis sučelja E0/2	
9				
10	Sučelje	Ethernet 0/1	Ethernet 0/3	
11	VLAN	102	101	
12	TIP	Access	Access	
13	STATUS	DOWN	DOWN	
14	OPIS	Opis sučelja E0/1	Opis sučelja E0/3	
15				
16	VLAN broj	101	102	
17	Ime	Opis za VLAN 101	Opis za VLAN 102	

Slika 1 Primjer Excel tablice

Figure 1 Excel table example

Nakon upisa potpune putanje Excel datoteke od strane korisnika, program provjerava dostupnost uređaja slanjem ping poruka na definiranu IP adresu. Nakon uspješnog utvrđivanja dostupnosti uređaja program zapisuje informacije iz tablice u liste varijabli. Vrijednost se uspoređuju s dopuštenim vrijednostima te se pretvaraju u naredbe i dodaju u varijablu liste. Nakon prolaska kroz sve vrijednosti tablice, programska skripta pokreće spajanje na uređaj putem SSH protokola uz unaprijed određene parametre spajanja. Korištenjem petlje prolazi se kroz spremljene naredbe te se na kraju sprema trenutna konfiguracija i prekida SSH sesija. Slika 2. prikazuje primjer izvršavanja programa.

3.2. PRIKUPLJANJE INFORMACIJA O MREŽI

3.2. *GATHERING NETWORK INFORMATION*

Kvalitetna dokumentacija sadrži ažurirane podatke o svakom mrežnom uređaju sustava, javno je dostupna unutar organizacija te šteti vrijeme koje je inače utrošeno na otkrivanje poznatih činjenica. Ovaj primjer prikazuje jednostavan način prikupljanja informacija o uređaju na mreži koristeći CDP (engl. Cisco Discovery Protocol) i nekolicinu naredbi za prikazivanje informacija o uređaju. Programska skripta spaja se na mrežne uređaje prolaskom kroz unaprijed određenu listu IP adresa. Prikupljanje informacije odrađuje se analizom odgovora na poslano show naredbe. Korištene naredbe su: show ip interface brief, show description, show version i show cdp entry *. Svaka naredba sadrži određene informacije o trenutnom stanju uređaja. Odvajanjem dobivenih informacije zasebnih naredbi u pojedinačne redove te analizom tih redova korištenjem regularnih izraza dobivaju se informacije prikazane na slici 3.

3.3. ODRŽAVANJE PRISTUPNIH LISTA

3.3. *MAINTAINING ACCESS LISTS*

Jedan od načina zaštite mrežnog sustava od neželjenog prometa je uporaba pristupnih lista putem kojeg se provjerava primljeni ili poslani paketi. Paketi se uspoređuju s listom pravila koja definiraju određenu radnju ako se paket podudara s određenim pravilom. Poredak pravila u listi je važan i potrebno je pažljivo poredati pravila kako bi se postigao traženi rezultat. Pravila pristupne liste najčešće se drže u tekstualnoj datoteci gdje je jednostavno dodati novo pravilo ili promijeniti poredak postojećih pravila.

Ažuriranje postojeće pristupne liste odrađuje se u dva koraka: prvi korak je brisanje postojeće liste, a drugi je primjena novih pravila iz tekstualne datoteke. Zadnji primjer prikazuje automatiziranje navedenog procesa. Slika 4. prikazuje tekstualnu datoteku s pravilima pristupne liste.

```

>>> exec(open("C:/Users/Filip/PycharmProjects/Switch/Primjer_1.py").read())
Unesite punu putanju do Excel datoteke:>> D:\S2.xlsx

Pinging 192.168.21.2 with 32 bytes of data:
Reply from 192.168.21.2: bytes=32 time=9ms TTL=253
Reply from 192.168.21.2: bytes=32 time=7ms TTL=253
Reply from 192.168.21.2: bytes=32 time=14ms TTL=253
Reply from 192.168.21.2: bytes=32 time=15ms TTL=253

Ping statistics for 192.168.21.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 7ms, Maximum = 15ms, Average = 11ms
config term
Enter configuration commands, one per line.  End with CNTL/Z.
S2(config)#vlan 101
S2(config-vlan)#name Opis za VLAN 101
S2(config-vlan)#vlan 102
S2(config-vlan)#name Opis za VLAN 102
S2(config-vlan)#interface Ethernet 0/0
S2(config-if)#no switchport mode access vlan
S2(config-if)#switchport trunk encapsulation dot1q
S2(config-if)#switchport mode trunk
S2(config-if)#no shut
S2(config-if)#description Veza prema R1
S2(config-if)#interface Ethernet 0/1
S2(config-if)#switchport access vlan 102
S2(config-if)#switchport mode access
S2(config-if)#shut
S2(config-if)#description Opis sucelja E0/1
S2(config-if)#interface Ethernet 0/2
S2(config-if)#switchport access vlan 101
S2(config-if)#switchport mode access
S2(config-if)#shut
S2(config-if)#description Opis sucelja E0/2
S2(config-if)#interface Ethernet 0/3
S2(config-if)#switchport access vlan 101
S2(config-if)#switchport mode access
S2(config-if)#shut
S2(config-if)#description Opis sucelja E0/3
S2(config-if)#end
S2#
Building configuration...
Compressed configuration from 2859 bytes to 1491 bytes[OK]
>>>

```

Provjera
dostupnosti
uređaja

Stvaranje
definiranih
VLAN-ova

Konfiguracija
definiranih sučelja
od ethernet nula
do ethernet tri

Spremanje trenutne
konfiguracije

Slika 2 Izvršavanje prvog primjera

Figure 2 Execution of first example

```

Pinging 192.168.19.2 with 32 bytes of data:
Reply from 192.168.19.2: bytes=32 time=16ms TTL=253
Reply from 192.168.19.2: bytes=32 time=17ms TTL=253
Reply from 192.168.19.2: bytes=32 time=14ms TTL=253
Reply from 192.168.19.2: bytes=32 time=12ms TTL=253

Ping statistics for 192.168.19.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 12ms, Maximum = 17ms, Average = 14ms
Ime uređaja: R3
Lozinka uređaja: lozinka
Sucelje:FastEthernet0/0, IP adresa:192.168.22.1, Status linije: up, Status protokola: up, opis sucelja je: Veza_prema_S3
Sucelje:Serial0/0, IP adresa:192.168.19.2, Status linije: up, Status protokola: up, opis sucelja je: Veza_prema_R2
Sucelje:FastEthernet0/1, IP adresa:unassigned, Status linije: administratively down, Status protokola: down, opis sucelja je: nema opisa
Uređaj S3.tvz.com spojen je sučeljem Ethernet0/0(192.168.22.2) na lokalno sučelje FastEthernet0/0
Uređaj R2.tvz.com spojen je sučeljem Serial1/1(192.168.19.1) na lokalno sučelje Serial0/0
Verzija hardvera: C3725
Verzija softvera: ADVENTERPRISEK9-M
Verzija operativnog sustava: 12.4(15)

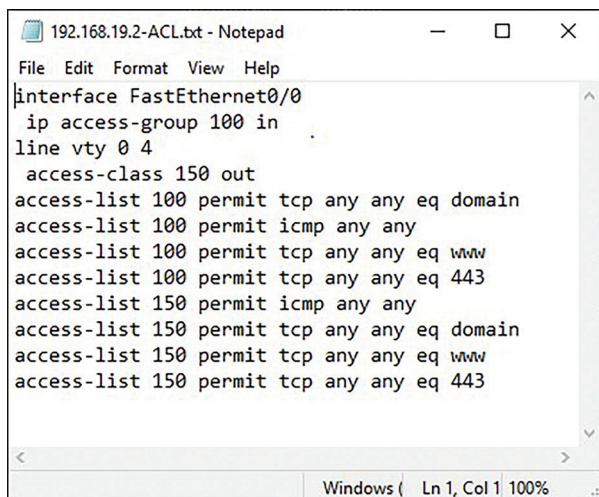
```

Provjera dostupnosti
uređaja

Informacije o uređaju

Slika 3 Izvršavanje drugog primjera

Figure 3 Execution of second example



```

192.168.19.2-ACL.txt - Notepad
File Edit Format View Help
interface FastEthernet0/0
 ip access-group 100 in
 line vty 0 4
  access-class 150 out
access-list 100 permit tcp any any eq domain
access-list 100 permit icmp any any
access-list 100 permit tcp any any eq www
access-list 100 permit tcp any any eq 443
access-list 150 permit icmp any any
access-list 150 permit tcp any any eq domain
access-list 150 permit tcp any any eq www
access-list 150 permit tcp any any eq 443

```

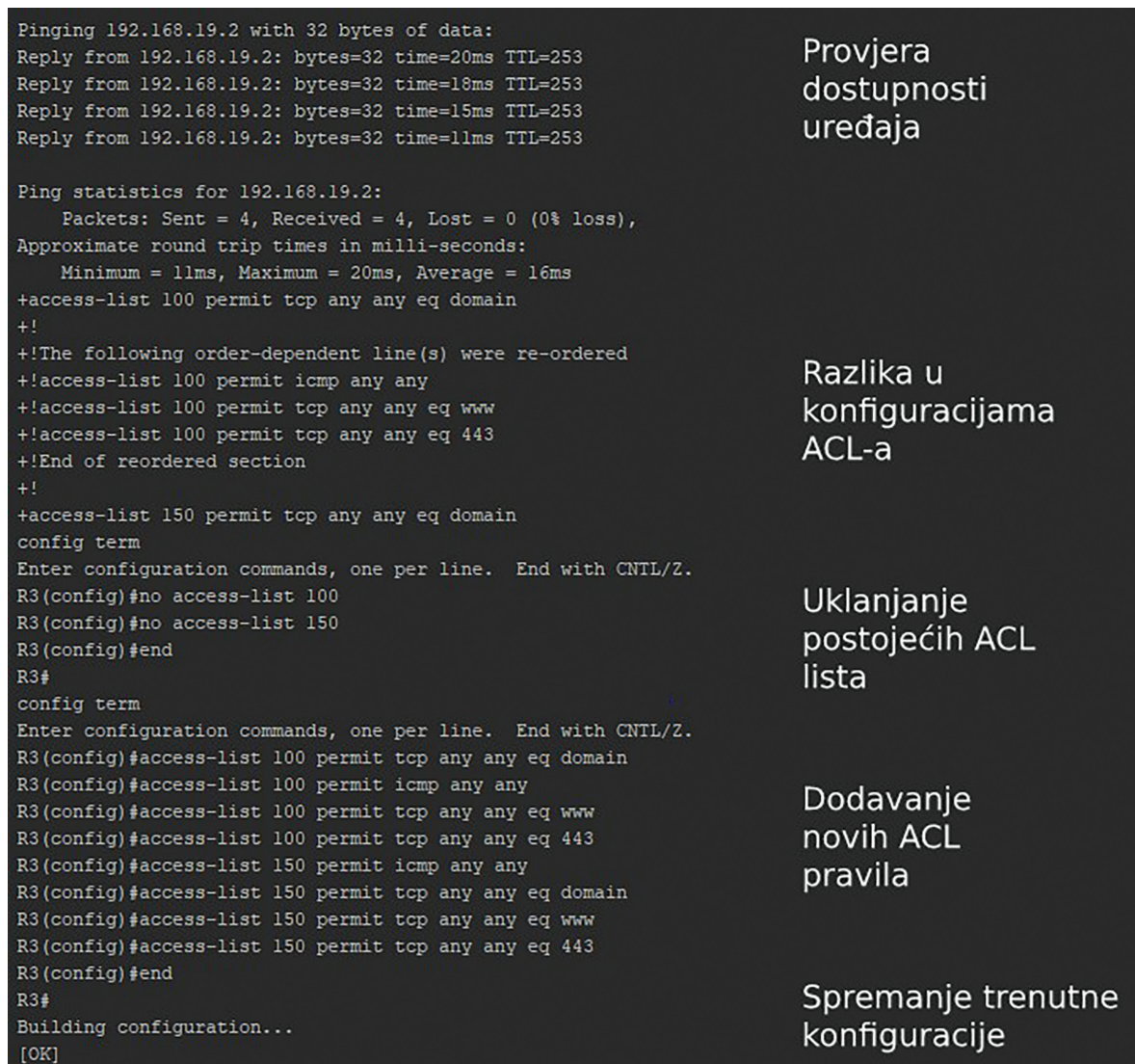
Slika 4 Pravila pristupne liste

Figure 4 Access list rules

Mrežni uređaj u primjeru sadrži pravila pristupne liste prikazana na slici 4., ali je prvo pravilo liste 100 i zadnje pravilo pristupne liste 150 uklonjeno.

Programska skripta provjerava dostupnost uređaja te nakon SSH spajanja uspoređuje pristupne liste uređaja i tekstualne datoteke. Ako razlika postoji ona će biti ispisana i broj ili ime liste će biti spremljeno u varijablu. Zapisane liste su zatim uklonjene te se zatim nova lista dodaje na uređaj i sprema se trenutna konfiguracija.

Slika 5. prikazuje izvršavanje programa.



```

Pinging 192.168.19.2 with 32 bytes of data:
Reply from 192.168.19.2: bytes=32 time=20ms TTL=253
Reply from 192.168.19.2: bytes=32 time=18ms TTL=253
Reply from 192.168.19.2: bytes=32 time=15ms TTL=253
Reply from 192.168.19.2: bytes=32 time=11ms TTL=253

Ping statistics for 192.168.19.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 11ms, Maximum = 20ms, Average = 16ms
+access-list 100 permit tcp any any eq domain
+!
+!The following order-dependent line(s) were re-ordered
+!access-list 100 permit icmp any any
+!access-list 100 permit tcp any any eq www
+!access-list 100 permit tcp any any eq 443
+!End of reordered section
+!
+access-list 150 permit tcp any any eq domain
config term
Enter configuration commands, one per line. End with CNTL/Z.
R3(config)#no access-list 100
R3(config)#no access-list 150
R3(config)#end
R3#
config term
Enter configuration commands, one per line. End with CNTL/Z.
R3(config)#access-list 100 permit tcp any any eq domain
R3(config)#access-list 100 permit icmp any any
R3(config)#access-list 100 permit tcp any any eq www
R3(config)#access-list 100 permit tcp any any eq 443
R3(config)#access-list 150 permit icmp any any
R3(config)#access-list 150 permit tcp any any eq domain
R3(config)#access-list 150 permit tcp any any eq www
R3(config)#access-list 150 permit tcp any any eq 443
R3(config)#end
R3#
Building configuration...
[OK]

```

Provjera
dostupnosti
uređaja

Razlika u
konfiguracijama
ACL-a

Uklanjanje
postojećih ACL
lista

Dodavanje
novih ACL
pravila

Spremanje trenutne
konfiguracije

Slika 5 Izvršavanje trećeg primjera

Figure 5 Execution of third example

4. ZAKLJUČAK

4. CONCLUSION

Razvoj mrežnih tehnologija sa sobom donosi i rast broja uređaja u mrežnom sustavu što otežava upravljanje nad uređajima. Tehnologije automatizacije mreže pokušavaju riješiti taj problem pomoću brzih i efikasnih programskih akcija. Konfiguracija uređaja jednostavnija je zbog unaprijed definiranih radnji koje reagiraju na promjenu i tako smanjuju vrijeme čekanja korisnika na promjenu stanja mreže.

Automatizacija se odrađuje kroz sustave automatizaciji ili kroz programske skripte. Sustav automatizacije je složeno rješenje koje zahtjeva moderne mrežne uređaje kojima je moguće upravljati koristeći programska sučelja. U radu su prikazana tri praktična primjera automatizacija starijih mrežnih uređaja koristeći programske skripte i popularni programski jezik Python. Primjeri prikazuju konfiguraciju preklopnika korištenjem Microsoft Excel tablice, prikupljanje informacija o mreži i održavanje pristupnih lista.

Mrežna automatizacija je nova tehnologija koja obećava velika poboljšanja u načinu rada mreže te manje troškove održavanja.

Za potpunu iskoristivost automatizacije potrebni su noviji mrežni uređaji što znači da će prijelaz na potpunu automatizaciju mreže biti spor. U međuvremenu postoje razvojne okoline poput SolarWinds i APIC-EM koje svim korisnicima omogućavaju ograničenu ali jednostavnu automatizaciju starijih mrežnih uređaja.

5. REFERENCE

5. REFERENCES

- [1.] Odom, W., Hogg, S., CCNA Routing and Switching ICND2 200-105 Official Cert Guide, Cisco Press, 2017
- [2.] Adato, L., Monitoring 101 A primer to the philosophy, theory, and fundamental concepts involved in systems monitoring (1st edition), SolarWinds, 2015
- [3.] Lammle, T., CCNA Routing and Switching Complete Study Guide (2nd edition), John Wiley & Sons, 2016

AUTORI · AUTHORS

● **Dubravko Žigman** - nepromjenjena biografija nalazi se u časopisu Polytechnic & Design Vol. 2, No. 1, 2014.

Korespondencija · Correspondence
dzigman@tvz.hr

● **Valter Perinović** - nepromjenjena biografija nalazi se u časopisu Polytechnic & Design Vol. 2, No. 2, 2014.

Korespondencija · Correspondence
valter.perinovic@tvz.hr