ზ.ცირამუა ვ.ოთხოზორია შ.სვანიშვილი

ქსელური კავშირები და WAN ტექნოლოგიები
(ქსელის ადმინისტრირება 2)

თბილისი 2015

შესავალი

წინამდებარე სახელმძღვანელო შედგენილია პროფესიული სტუდენტებისათვის, საგანმანათლებლო პროგრამის „კომპიუტერული ქსელის ადმინისტრირება“ სწავლებისათვის და მოიცავს ამ პროგრამით გათვალისწინებულ II ეტაპზე სწავლებად ძირითად მოდულებს.

სახელმძღვანელოს მიზანია დაეხმაროს სტუდენტს დაეუფლოს კომპიუტერული ქსელის დაგეგმვა-გამართვის პრინციპებს და შეძლოს მისი ეფექტური გამოყენება პროფესიულ საქმიანობაში. სახელმძღვანელოში მოცემული საკითხები საინტერესო და აქტუალური იქნება ქსელის ადმინისტრატორებისათვის და დაეხმარება მათ მცირე და საშუალო ქსელების ადმინისტრირებაში. სახელმძღვანელოში მოცემული ინსტრუმენტების გამოყენება შესაძლებელია, როგორც სწავლების, ასევე დასაქმებისა და ყოველდღიური საქმიანობის პირობებში.

სახელმძღვანელოში აღწერილია მიმდინარე პერიოდში აქტუალური და ფართოდ გამოყენებადი პროგრამულ-აპარატურული უზრუნველყოფის ელემენტები და სერვისები.

სახელმძღვანელო დაყოფილია 6 ნაწილად (თავი). ყოველი თავი შეესაბამება კონკრეტული მოდულის დასახელებას, შესაბამისი ქვეთავები კი ეხმაურება მოდულის ჩარჩოთი განსაზღვრულ სწავლის შედეგებს. თითოეულ ნაწილში თემატურ ტექსტურ-ილუსტრირებულ მასალასთან ერთად წარმოდგენილია სავარჯიშოები, სახელმძღვანელოს ყოველი თავის ბოლოს შეფასების მიდგომებიდან გამომდინარე დართული აქვს შემაჯამებელი სამუშაო, პრაქტიკული დავალება-სავარჯიშო ან/და ტესტის ნიმუში.

რეცენზენტები:

მიხეილ სამხარაძე - საგანმანათლებლო პროგრამის შემმუშავებელი ჯგუფის წევრი, ექსპერტი

ვლადიმერ ადამია - სტუ-ს კომპიუტერული ქსელის ადმინისტრატორი

სარჩევი

1. მესამე დონის მარშრუტიზაციის პროტოკოლები

1.1. EIGRP პროტოკოლი

ძირითადი მახასიათებლები:

- მარშრუტის ღირებულების განსაზღვრა სხვადასხვა მეტრიკის საფუძველზე
- მაქსიმალური გადასვლების რიცხვი - 224
- RIP-გან განსხვავებით EIGRP პროტოკოლი არ შემოიფარგლება მხოლოდ საკუთარი მარშრუტიზაციის ცხრილით. ამ პროტოკოლისათვის იქმნება 2 ძირითადი მონაცემთა ბაზის ცხრილი: მეზობელი მარშრუტიზატორების ცხრილები და ტოპოლოგიის ცხრილი

Characteristics and Features	IGRP	EIGRP
Metric	Both use a composite metric consisting of bandwidth and delay. Reliability and load can also be included in the metric calculation.	
Updates Forwarded to Address	255.255.255.255	224.0.0.10
Supports VLSM	✗	✓
Supports CIDR	✗	✓
Supports Summarization	✗	✓
Supports Authentication	✗	✓

პრაქტიკული დავალება

ქვემოთ მოცემული მარშრუტიზაციის ცხრილების მიხედვით შექმენით ქსელის ფიზიკური და ლოგიკური მოდელი

მარშრუტიზატორი 1

EQE1#sh ip route

172.16.0.0/24 is subnetted, 6 subnets

D 172.16.252.0 [90/2681856] via 172.16.250.2, 00:18:54, Ethernet0/0

C 172.16.250.0 is directly connected, Ethernet0/0

C 172.16.251.0 is directly connected, Ethernet0/1

D 172.16.50.0 [90/2195456] via 172.16.250.2, 00:18:54, Ethernet0/0

C 172.16.1.0 is directly connected, Loopback0

D 172.16.100.0 [90/2707456] via 172.16.250.2, 00:18:54, Ethernet0/0

C 192.168.1.0/24 is directly connected, Loopback1

მარშრუტიზატორი 2

EQE2#sh ip route

172.16.0.0/24 is subnetted, 6 subnets

C 172.16.252.0 is directly connected, Serial0

D 172.16.250.0 [90/2681856] via 172.16.252.1, 00:21:10, Serial0

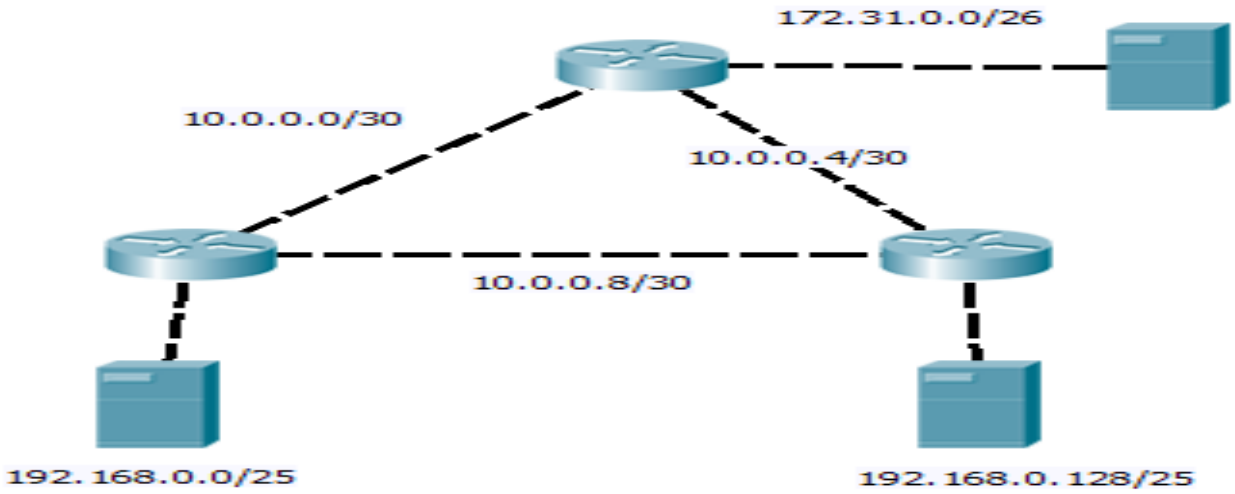
C 172.16.251.0 is directly connected, Serial1

D 172.16.50.0 [90/2195456] via 172.16.252.1, 00:21:10, Serial0

D 172.16.1.0 [90/2707456] via 172.16.252.1, 00:15:36, Serial0

C 172.16.100.0 is directly connected, Ethernet0

პრაქტიკული დავალება



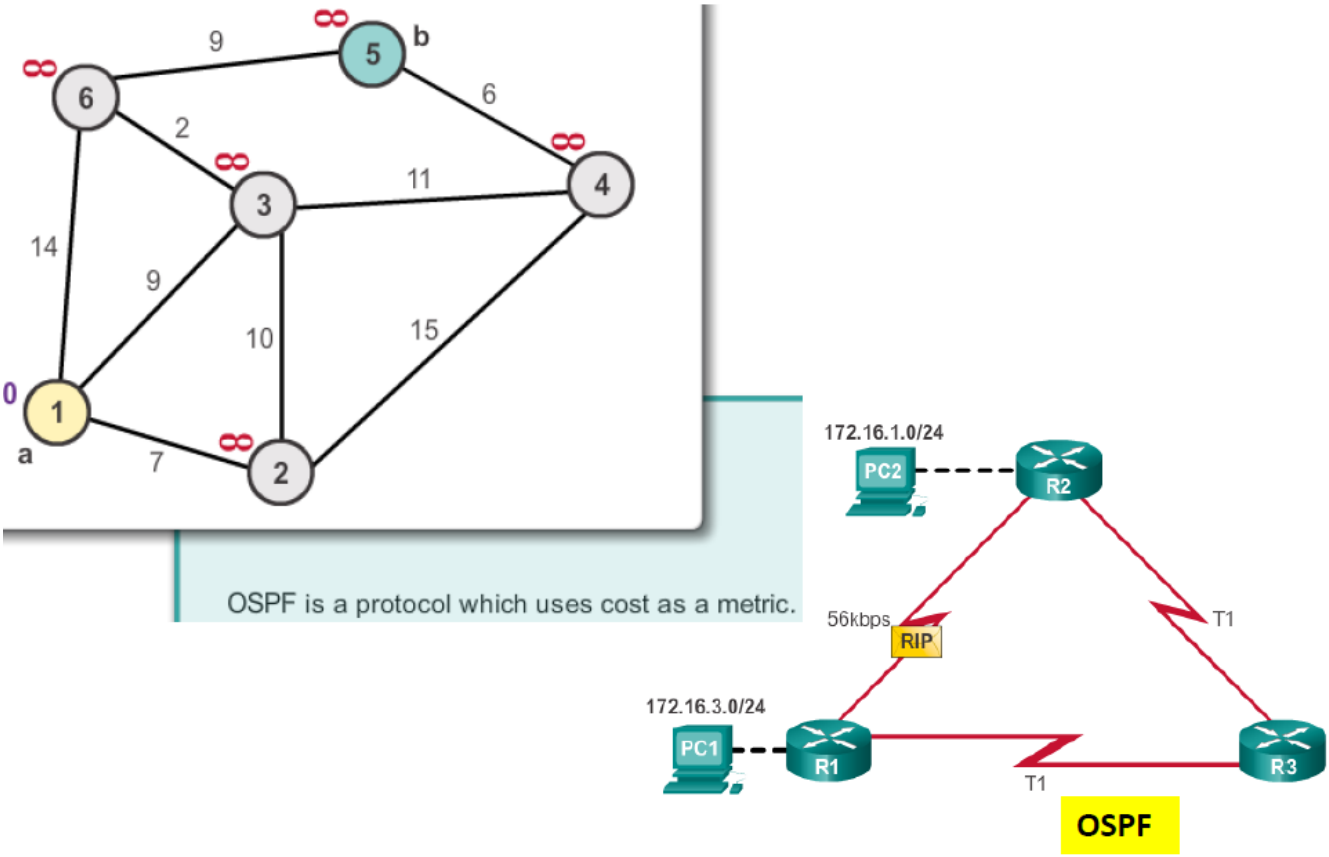
1. **შექმენით მოცემულის შესაბამისი ლოკალური ქსელი და ინტერფეისები დაამისამართეთ შესაბამისად**
2. **მარშრუტიზატორებზე გააქტიურეთ OSPF პროტოკოლი**
    a. **დანიშნეთ Router ID**
    b. **Refernce Bandwith განსაზღვრეთ 1000**
    c. **ლოკალური ინტერფეისებს მიანიჭეთ Passive სტატუსი**

1.2. OSPF პროტოკოლი

სხვადასხვა პროტოკოლები იყენებენ განსხვავებულ მეტრიკას დანიშნულების მისამართამდე მარშრუტის განსაზღვრისას

OSPF(Open Shortest Path First) პროტოკოლი ირჩევს მარშრუტს გამტარუნარიანობაზე (bandwidth) დაყრდნობით

მარშრუტიზაციის პროტოკოლების მეტრიკა



6.5.1. OSPF კონფიგურირების ძირითადი ბრძანებები

უშუალოდ მიერთებული ქსელის მითითება

- ✓ **network „ქსელის მისამართი“ wildcard-mask area „არეალის ნომერი“**

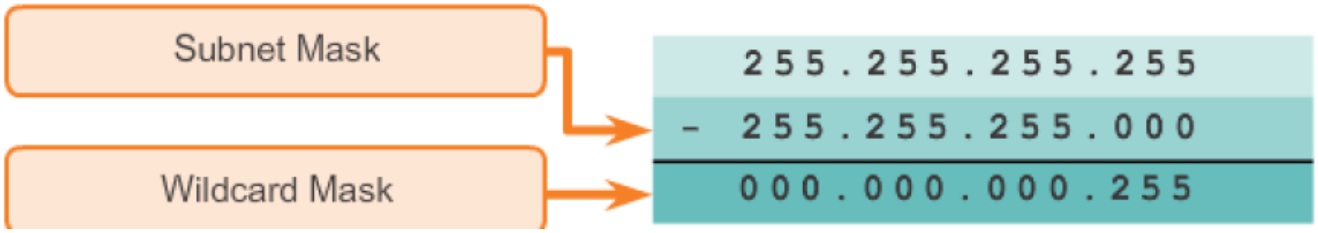
  მაგ.: network 192.168.16.0 0.0.0.255 area 0

- 192.168.16.0 - უშუალოდ მიერთებული ქსელია
- 0.0.0.255 - შებრუნებული ქვექსელის ნიღაბი
- 0 – OSPF area

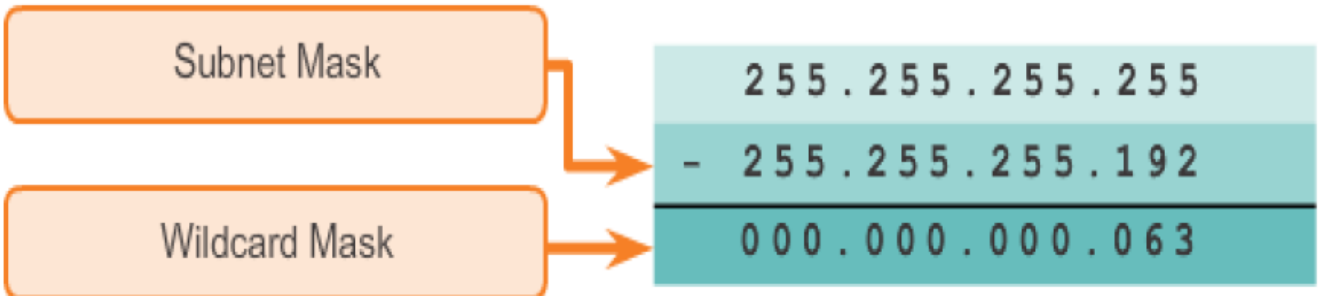
```
R1(config)# router ospf 10
R1(config-router)# network 172.16.1.0 0.0.0.255 area 0
R1(config-router)# network 172.16.3.0 0.0.0.3 area 0
R1(config-router)# network 192.168.10.4 0.0.0.3 area 0
R1(config-router)#
```

შებრუნებული ქვექსელის ნიღაბის (Wildcard mask)-ის გამოთვლა

მაგ.: გამოვითვალოთ 255.255.255.0 ანუ /24 – ის Wildcard mask

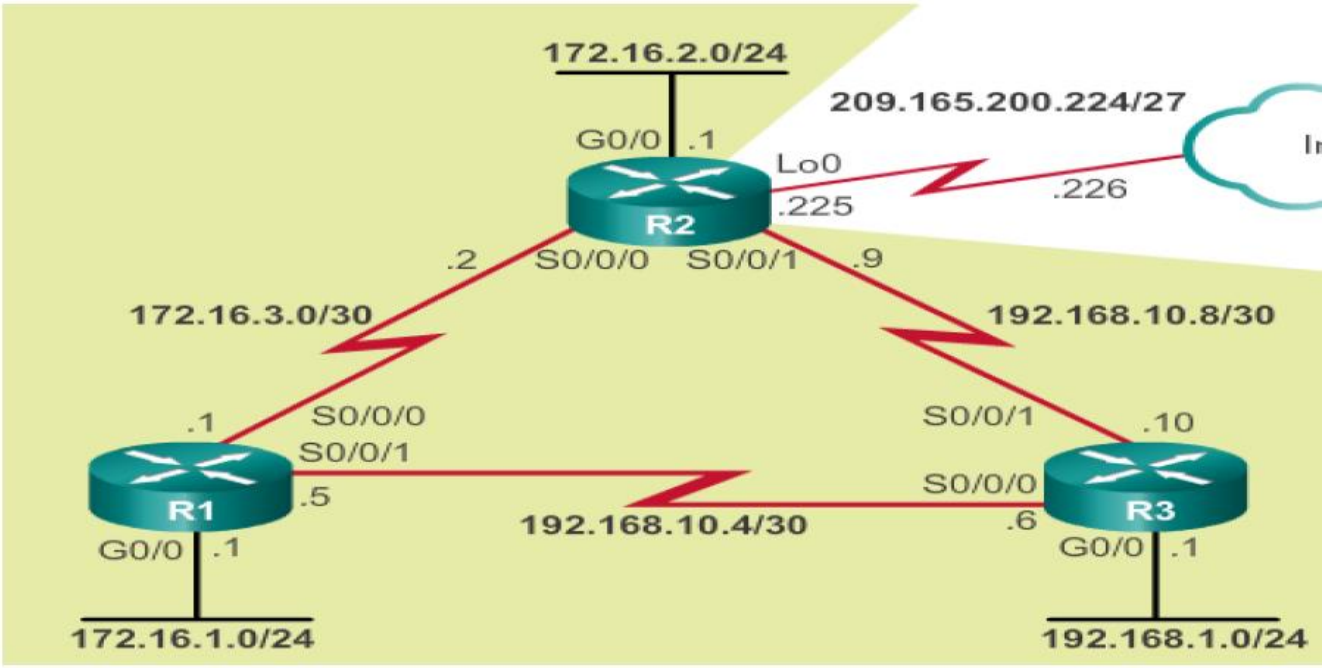


მაგ.: გამოვითვალოთ 255.255.255.192 ანუ /26 – ის Wildcard mask



Passive-interface- ის დანიშვნა

```
R1(config)# router ospf 10
R1(config-router)# passive-interface GigabitEthernet 0/0
R1(config-router)# end
R1#
```



OSPF Cost (ღირებულების) განსაზღვრა

Cost = reference bandwidth / interface bandwidth

Cost = 100,000,000 bps / interface bandwidth in bps

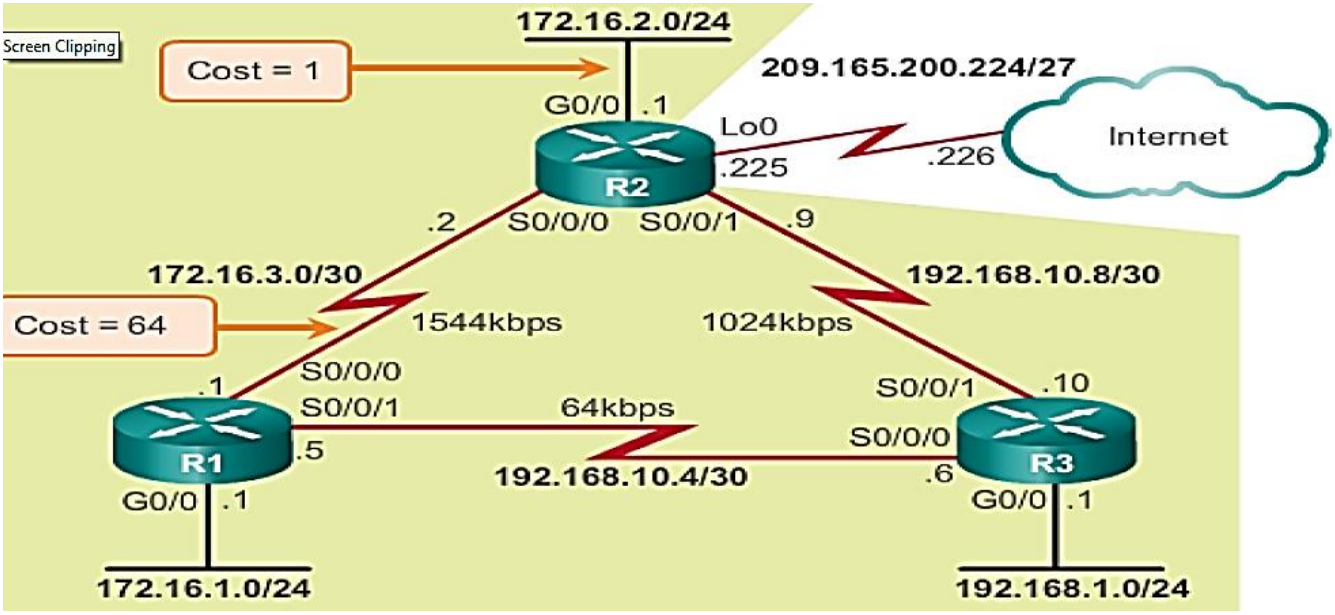
Interface Type	Reference Bandwidth in bps		Default Bandwidth in bps	Cost
10 Gigabit Ethernet 10 Gbps	100,000,000	÷	10,000,000,000	1
Gigabit Ethernet 1 Gbps	100,000,000	÷	1,000,000,000	1
Fast Ethernet 100 Mbps	100,000,000	÷	100,000,000	1
Ethernet 10 Mbps	100,000,000	÷	10,000,000	10
Serial 1.544 Mbps	100,000,000	÷	1,544,000	64
Serial 128 kbps	100,000,000	÷	128,000	781
Serial 64 kbps	100,000,000	÷	64,000	1562

მაგ.: სურათზე გამოსახულის მიხედვით განვსაზღვროთ R1-დან R2-ის 172.16.2.0/24 ქსელზე ჯამური ღირებულება

Serial link from R1 to R2 cost = 64

Gigabit Ethernet link on R2 cost = 1

Total cost to reach 172.16.2.0/24 = 65



auto-cost reference-bandwidth Mb/s

Default პარამეტრებით მარშრუტიზატორებზე „reference bandwidth“ მაჩვენებელი 100 Mb/s-ია, ამიტომაც Cost მაჩვენებელი FastEthernet; GigabitEthernet და 10GigabitEthernet ინტერფეისებისთვის იქნება იდენტური.

10 Gigabit Ethernet 10 Gbps	100,000,000	÷	10,000,000,000	1
Gigabit Ethernet 1 Gbps	100,000,000	÷	1,000,000,000	1
Fast Ethernet 100 Mbps	100,000,000	÷	100,000,000	1

ამიტომაც უპრიანია GigabitEthernet და 10GigabitEthernet ინტერფეისების შემთხვევაში დავნიშნოთ ქვემოთ მოცემულის შესაბამისი „reference bandwidth“ მაჩვენებელი

Gigabit Ethernet - auto-cost reference-bandwidth 1000

10 Gigabit Ethernet - auto-cost reference-bandwidth 10000

Cost მაჩვენებლის განსაზღვრის ალტერნატილი გზები :

ინტერფეისზე შევცვალოთ Bandwidth მაჩვენებელი

```
R1(config)# int s0/0/1
R1(config-if)# bandwidth 64
R1(config-if)# end
R1#
*Mar 27 10:10:07.735: %SYS-5-CONFIG_I: Configured from console by c
R1#
R1# show interfaces serial 0/0/1 | include BW
  MTU 1500 bytes, BW 64 Kbit/sec, DLY 20000 usec,
R1#
R1# show ip ospf interface serial 0/0/1 | include Cost:
  Process ID 10, Router ID 1.1.1.1, Network Type
  POINT_TO_POINT, Cost: 15625
R1#
```

ხელით შევცვალოთ(დავადგინოთ) ip ospf cost ღირებულება

```
R1(config)# int s0/0/1
R1(config-if)# no bandwidth 64
R1(config-if)# ip ospf cost 15625
R1(config-if)# end
R1#
R1# show interface serial 0/0/1 | include BW
    MTU 1500 bytes, BW 1544 Kbit/sec, DLY 20000 usec,
R1#
R1# show ip ospf interface serial 0/0/1 | include Cost:
    Process ID 10, Router ID 1.1.1.1, Network Type POINT_TO_POINT,
    Cost: 15625
R1#
```

OSPF შემოწმება

```
R1# show ip protocols
*** IP Routing is NSF aware ***

Routing Protocol is "ospf 10"
  Outgoing update filter list for all interfaces is not
  set
  Incoming update filter list for all interfaces is not
  set
  Router ID 1.1.1.1
  Number of areas in this router is 1. 1 normal 0 stub 0
  nssa
  Maximum path: 4
  Routing for Networks:
    172.16.1.0 0.0.0.255 area 0
    172.16.3.0 0.0.0.3 area 0
    192.168.10.4 0.0.0.3 area 0
  Routing Information Sources:
    Gateway         Distance      Last Update
    2.2.2.2              110      00:17:18
    3.3.3.3              110      00:14:49
  Distance: (default is 110)
```

```
Screen Clipping how ip ospf neighbor

Neighbor ID  Pri  State   Dead Time Address        Interface
3.3.3.3      0    FULL/-  00:00:37  192.168.10.6   Serial0/0/1
2.2.2.2      0    FULL/-  00:00:30  172.16.3.2     Serial0/0/0
R1#
```

```
R1# show ip ospf interface brief
Interface  PID  Area  IP Address/Mask  Cost  State  Nbrs F/C
Se0/0/1    10   0     192.168.10.5/30  15625 P2P    1/1
Se0/0/0    10   0     172.16.3.1/30    647   P2P    1/1
Gi0/0      10   0     172.16.1.1/24    1     DR     0/0
R1#
```

პრაქტიკული სამუშაო

OSPF პროტოკოლის კონფიგურირება (სამუშაო სრულდება Packet Tracer-ში)

სამუშაოს შესაბამისი ფაილების გადმოსაწერად გააქტიურეთ შემდეგი ბმული:

- http://1drv.ms/1puiS5P
- http://1drv.ms/1puj1Ge
- http://1drv.ms/1jWt5HB

6.5.2. OSPFv3 კონფიგურირება

```
R1(config)# ipv6 unicast-routing
```

```
R1(config)# interface GigabitEthernet 0/0
R1(config-if)# ipv6 address fe80::1 link-local
R1(config-if)# exit
R1(config)# interface Serial0/0/0
R1(config-if)# ipv6 address fe80::1 link-local
R1(config-if)# exit
R1(config)# interface Serial0/0/1
R1(config-if)# ipv6 address fe80::1 link-local
R1(config-if)#
```

```
R1(config)# ipv6 router ospf 10
R1(config-rtr)#
*Mar 29 11:21:53.739: %OSPFv3-4-NORTRID: Process OSPFv3-1-
IPv6 could not pick a router-id, please configure manually
R1(config-rtr)#
R1(config-rtr)# router-id 1.1.1.1
R1(config-rtr)#
R1(config-rtr)# auto-cost reference-bandwidth 1000
% OSPFv3-1-IPv6: Reference bandwidth is changed. Please
ensure reference bandwidth is consistent across all routers
R1(config-rtr)#
R1(config-rtr)# end
R1#
R1# show ipv6 protocols
IPv6 Routing Protocol is "connected"
IPv6 Routing Protocol is "ND"
IPv6 Routing Protocol is "ospf 10"
  Router ID 1.1.1.1
  Number of areas: 0 normal, 0 stub, 0 nssa
  Redistribution:
    None
R1#
```

```
R1(config)# ipv6 unicast-routing
```

```
R1(config)# interface GigabitEthernet 0/0
R1(config-if)# ipv6 ospf 10 area 0
R1(config-if)#
R1(config-if)# interface Serial0/0/0
R1(config-if)# ipv6 ospf 10 area 0
R1(config-if)#
R1(config-if)# interface Serial0/0/1
R1(config-if)# ipv6 ospf 10 area 0
R1(config-if)#
R1(config-if)# end
R1#
R1# show ipv6 ospf interfaces brief
Interface   PID   Area      Intf ID  Cost    State  Nbrs F/C
Se0/0/1     10      0         7      15625   P2P    0/0
Se0/0/0     10      0         6      647     P2P    0/0
Gi0/0       10      0         3       1      WAIT   0/0
R1#
```

OSPFv3 შემოწმება

```
R1# show ipv6 ospf neighbor

OSPFv3 Router with ID (1.1.1.1) (Process ID 10)

Neighbor ID  Pri  State      Dead Time  Interface ID Interface
3.3.3.3        0  FULL/  -   00:00:39   6            Serial0/0/1
2.2.2.2        0  FULL/  -   00:00:36   6            Serial0/0/0
R1#
```

```
R1# show ipv6 protocols
IPv6 Routing Protocol is "connected"
IPv6 Routing Protocol is "ND"
IPv6 Routing Protocol is "ospf 10"
  Router ID 1.1.1.1
  Number of areas: 1 normal, 0 stub, 0 nssa
  Interfaces (Area 0):
    Serial0/0/1
    Serial0/0/0
    GigabitEthernet0/0
  Redistribution:
    None
R1#
```

```
R1# show ipv6 ospf interface brief
Interface    PID   Area            Intf ID    Cost  State Nbrs F/C
Se0/0/1      10    0               7          15625 P2P   1/1
Se0/0/0      10    0               6          647   P2P   1/1
Gi0/0        10    0               3          1     DR    0/0
R1#
```

```
R1# show ipv6 route ospf
IPv6 Routing Table - default - 10 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user
Static route
       B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
       I2 - ISIS L2, IA - ISIS interarea, IS - ISIS
summary, D - EIGRP
       EX - EIGRP external, ND - ND Default, NDp - ND
Prefix, DCE - Destination
       NDr - Redirect, O - OSPF Intra, OI - OSPF Inter,
OE1 - OSPF ext 1
       OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF
NSSA ext 2
O    2001:DB8:CAFE:2::/64 [110/657]
     via FE80::2, Serial0/0/0
O    2001:DB8:CAFE:3::/64 [110/1304]
     via FE80::2, Serial0/0/0
O    2001:DB8:CAFE:A002::/64 [110/1294]
     via FE80::2, Serial0/0/0
R1#
```

პრაქტიკული სამუშაო

OSPFv3 პროტოკოლის კონფიგურირება (სამუშაო სრულდება Packet Tracer-ში)

სამუშაოს შესაბამისი ფაილების გადმოსაწერად გააქტიურეთ შემდეგი ბმულები:

- http://1drv.ms/1nrWMUr
- http://1drv.ms/1nrWTiZ
- http://1drv.ms/1nrWZXC

1.3. RIP, EIGRP,OSPF შორის მარშრუტების გაცვლა და ოპტიმიზირება.

RIP (Routing Information Protocol) პროტოკოლი

RIP1

- განახლებები იგზავნება 255.255.255.255 მისამართზე ყოველ 30 წამში
- მარშრუტი განისაზღვრება გადასასვლელების(Hop) რაოდენობით
- მაქსიმალური გადასასვლელების რაოდენობაა 15

RIP2

- განახლებები იგზავნება 224.0.0.9 მისამართზე
- აქვს უკლასო მარშრუტიზაციის VLSM და CIDR მხარდაჭერა
- აქვს ჯამური მარშუტიზაციის მხარდაჭერა
- უსაფრთხოების თვალსაზრისით აქვს აუტენტიფიკაციის მექანიზმის მხარდაჭერა

Characteristics and Features	RIPv1	RIPv2
Metric	Both use hop count as a simple metric. The maximum number of hops is 15.	
Updates Forwarded to Address	255.255.255.255	224.0.0.9
Supports VLSM	✗	✓
Supports CIDR	✗	✓
Supports Summarization	✗	✓
Supports Authentication	✗	✓

IPv6 მხარდაჭერის პროტოკოლია RIPNG, მაქსიმალური 15 გადასასვლელითა და administrative distance 120-ით

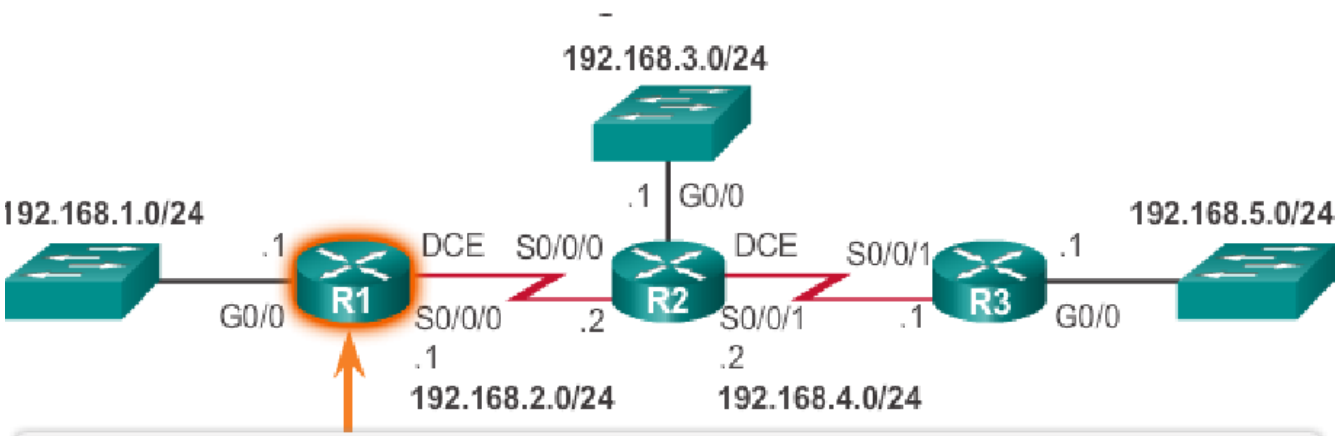
1.3.1. *RIP კონფიგურირება*

ძირითადი მახასიათებლები:

- RIP იყენებს გადასასვლელების რიცხვს, როგორც მეტრიკას მარშრუტის არჩევისას
- 15-ზე მეტი გადასასვლელის შემთხვევაში განსაზღვრავს როგორც მიუღწეველ მარშრუტს
- აგზავნის მარშრუტიზაციის ცხრილის მონაცემებს ყოველ 30 წამში ;

პროტოკოლის გამართვა

- Router(config)#router rip
- Router (config-router)#version 2
- Router(config-router)#network [ქსელის მისამართი]




```
R1(config)# router rip
R1(config-router)# network 192.168.1.0
R1(config-router)# network 192.168.2.0
R1(config-router)#
```

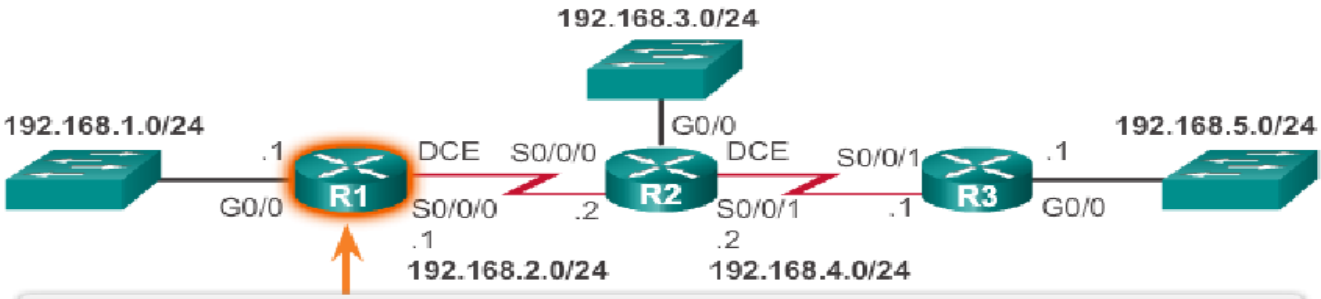
პარამეტრების შემოწმება

```
R1# show ip protocols
*** IP Routing is NSF aware ***

Routing Protocol is "rip"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Sending updates every 30 seconds, next due in 16 seconds
  Invalid after 180 seconds, hold down 180, flushed after 240
  Redistributing: rip

  Default version control: send version 1, receive any version
    Interface              Send  Recv  Triggered RIP  Key-chain
    GigabitEthernet0/0     1     1 2
    Serial0/0/0            1     1 2

  Automatic network summarization is in effect
  Maximum path: 4
  Routing for Networks:
    192.168.1.0
    192.168.2.0

  Routing Information Sources:
    Gateway         Distance      Last Update
    192.168.2.2           120      00:00:15
  Distance: (default is 120)
```

მარშრუტიზაციის ცხრილში RIP პროტოკოლით მიღებული ჩანაწერი R სიმბოლოთი აღინიშნება

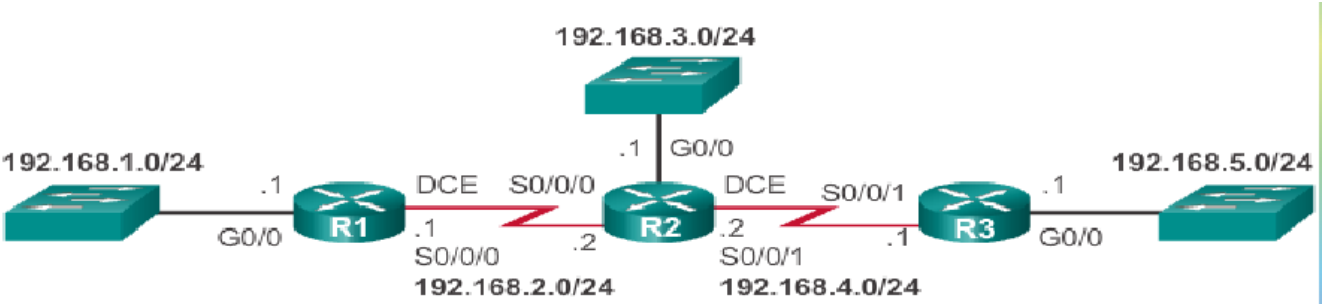
```
R1# show ip route | begin Gateway
Gateway of last resort is not set

      192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.1.0/24 is directly connected, GigabitEthernet0/0
L        192.168.1.1/32 is directly connected, GigabitEthernet0/0
      192.168.2.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.2.0/24 is directly connected, Serial0/0/0
L        192.168.2.1/32 is directly connected, Serial0/0/0
R     192.168.3.0/24 [120/1] via 192.168.2.2, 00:00:24, Serial0/0/0
R     192.168.4.0/24 [120/1] via 192.168.2.2, 00:00:24, Serial0/0/0
R     192.168.5.0/24 [120/2] via 192.168.2.2, 00:00:24, Serial0/0/0
R1#
```

RIPv2-ის გააქტიურება




```
R1(config)# router rip
R1(config-router)# version 2
R1(config-router)# ^Z
R1#
R1# show ip protocols | section Default
  Default version control: send version 2, receive version 2
    Interface              Send  Recv  Triggered RIP  Key-chain
    GigabitEthernet0/0     2     2
    Serial0/0/0            2     2
R1#
```

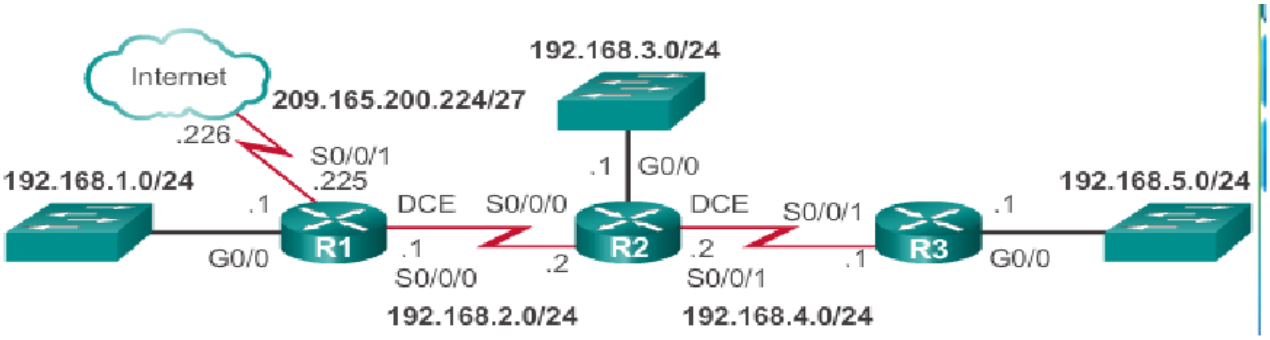
Passive ინტერფეისის დანიშვნა



მოცემულ სურათზე R1 მარშრუტიზატორის G0/0 ინტერფეისზე არ არის პროტოკოლებით ფორმირებული განახლებების დაგზავნის აუცილებლობა, რამეთუ მოცემულ ინტერფეისზე მიერთებულია LAN ქსელი, ამიტომაც მას უნდა მიენიჭოს Passive სტატუსი

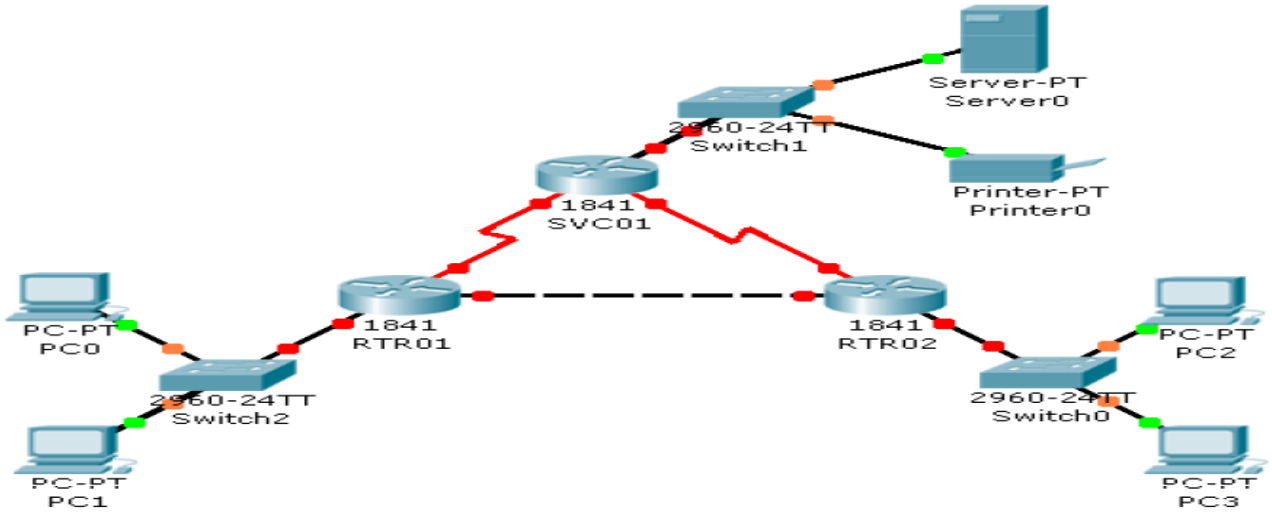
```
R1(config)# router rip
R1(config-router)# passive-interface g0/0
R1(config-router)# end
```

Default მარშრუტის გავრცელება(შეხამება) RIP პროტოკოლთან




```
R1(config)# ip route 0.0.0.0 0.0.0.0 S0/0/1 209.165.200.226
R1(config)# router rip
R1(config-router)# default-information originate
R1(config-router)# ^Z
R1#
*Mar 10 23:33:51.801: %SYS-5-CONFIG_I: Configured from console by
console
R1# show ip route | begin Gateway
Gateway of last resort is 209.165.200.226 to network 0.0.0.0

S*    0.0.0.0/0 [1/0] via 209.165.200.226, Serial0/0/1
```

პრაქტიკული სავარჯიშო



1. შექმენით სურათზე მოცემულის შესაბამისი ქსელის ფიზიკური მოდელი
2. ლოგიკური მისამართები შეარჩიეთ თქვენი სურვილისამებრ
3. მარშრუტიზატორებში გააქტიურეთ RIPv2 პროტოკოლი
4. შეამოწმეთ კავშირი სხვადასხვა ქსელის კვანძებს შორის

პრაქტიკული სამუშაო

RIPv2 პროტოკოლის კონფიგურირება (სამუშაო სრულდება Packet Tracer-ში)

სამუშაოს შესაბამისი ფაილის გადმოსაწერად გააქტიურეთ შემდეგი ბმული:

http://1drv.ms/1mUCEKm

პრაქტიკული დავალება

ქვემოთ მოცემული მარშრუტიზაციის ცხრილების მიხედვით შექმენით ქსელის ფიზიკური და ლოგიკური მოდელი

თქვენი გვარი1#show ip route

10.0.0.0/24 is subnetted, 1 subnets

C 10.0.0.0 is directly connected, Loopback0

172.16.0.0/28 is subnetted, 2 subnets

C 172.16.0.0 is directly connected, FastEthernet0/1

C 172.16.0.16 is directly connected, FastEthernet0/0

R 192.168.0.0/24 [120/1] via 172.16.0.2, 00:00:03, FastEthernet0/1

R 192.168.10.0/24 [120/1] via 172.16.0.18, 00:00:20, FastEthernet0/0

თქვენი გვარი2#show ip route

R 10.0.0.0/8 [120/1] via 172.16.0.17, 00:00:09, FastEthernet0/0

172.16.0.0/28 is subnetted, 2 subnets

R 172.16.0.0 [120/1] via 172.16.0.17, 00:00:09, FastEthernet0/0

C 172.16.0.16 is directly connected, FastEthernet0/0

R 192.168.0.0/24 [120/2] via 172.16.0.17, 00:00:09, FastEthernet0/0

C 192.168.10.0/24 is directly connected, Loopback0

თქვენი გვარი3#show ip route

R 10.0.0.0/8 [120/1] via 172.16.0.1, 00:00:26, FastEthernet0/0

172.16.0.0/28 is subnetted, 2 subnets

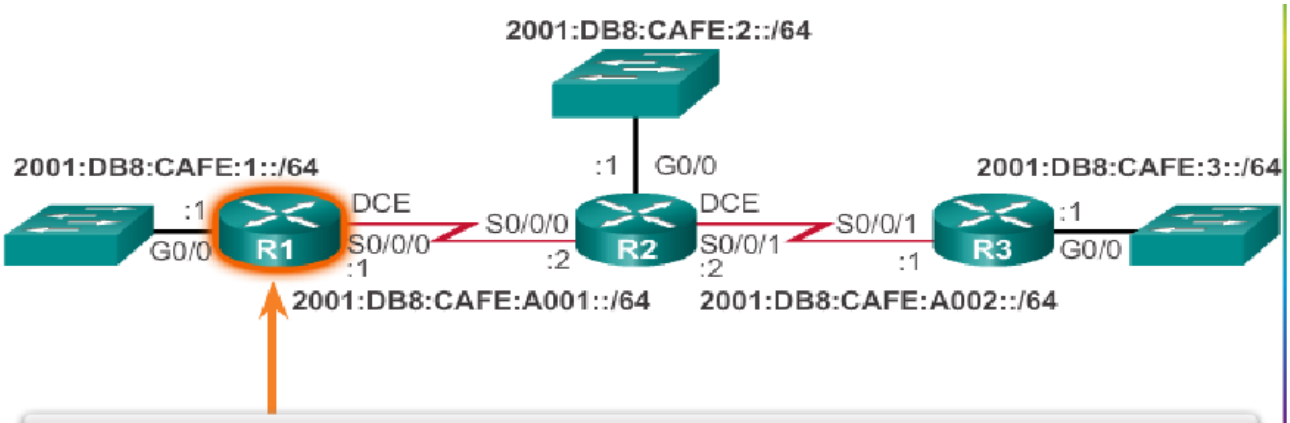
C 172.16.0.0 is directly connected, FastEthernet0/0

R 172.16.0.16 [120/1] via 172.16.0.1, 00:00:26, FastEthernet0/0

C 192.168.0.0/24 is directly connected, Loopback0

R 192.168.10.0/24 [120/2] via 172.16.0.1, 00:00:26, FastEthernet0/0

RIPng პროტოკოლის კონფიგურირება




```
R1(config)# ipv6 unicast-routing
R1(config)#
R1(config)# interface gigabitethernet 0/0
R1(config-if)# ipv6 rip RIP-AS enable
R1(config-if)# exit
R1(config)#
R1(config)# interface serial 0/0/0
R1(config-if)# ipv6 rip RIP-AS enable
R1(config-if)# no shutdown
R1(config-if)#
```

RIPng პროტოკოლის შემოწმება

```
R1# show ipv6 protocols
IPv6 Routing Protocol is "connected"
IPv6 Routing Protocol is "ND"
IPv6 Routing Protocol is "rip RIP-AS"
  Interfaces:
    Serial0/0/0
    GigabitEthernet0/0
```

```
R1# show ipv6 route
C    2001:DB8:CAFE:1::/64 [0/0]
     via GigabitEthernet0/0, directly connected
L    2001:DB8:CAFE:1::1/128 [0/0]
     via GigabitEthernet0/0, receive
R    2001:DB8:CAFE:2::/64 [120/2]
     via FE80::FE99:47FF:FE71:78A0, Serial0/0/0
R    2001:DB8:CAFE:3::/64 [120/3]
     via FE80::FE99:47FF:FE71:78A0, Serial0/0/0
C    2001:DB8:CAFE:A001::/64 [0/0]
     via Serial0/0/0, directly connected
L    2001:DB8:CAFE:A001::1/128 [0/0]
     via Serial0/0/0, receive
R    2001:DB8:CAFE:A002::/64 [120/2]
     via FE80::FE99:47FF:FE71:78A0, Serial0/0/0
L    FF00::/8 [0/0]
     via Null0, receive
```

```
R1# show ipv6 route rip
R    2001:DB8:CAFE:2::/64 [120/2]
     via FE80::FE99:47FF:FE71:78A0, Serial0/0/0
R    2001:DB8:CAFE:3::/64 [120/3]
     via FE80::FE99:47FF:FE71:78A0, Serial0/0/0
R    2001:DB8:CAFE:A002::/64 [120/2]
     via FE80::FE99:47FF:FE71:78A0, Serial0/0/0
```

RIPng პროტოკოლის კონფიგურირება (სამუშაო სრულდება Packet Tracer-ში)

სამუშაოს შესაბამისი ფაილების გადმოსაწერად გააქტიურეთ შემდეგი ბმული:

- http://1drv.ms/1mUEtXD
- http://1drv.ms/1mXzgKy

1.4. მესამე დონის მარშრუტიზაციის პროტოკოლების (RIP, EIGRP,OSPF) IPv6 კონფიგურაცია.

მარშრუტიზატორი იღებს გადაწყვეტილებას მარშრუტიზაციაზე, მის ცხრილში არსებული ინფორმაციის მიხედვით.

მარშრუტები შესაძლებელია დაინიშნოს ადმინისტრატორის მიერ სტატიკურად ან გამოეყოს მას დინამიურად სხვა მარშუტიზატორის მეშვეობით ან მარშრუტიზაციის პროგრამული პროტოკოლით.

მარშრუტი დგინდება 4 ძირითადი კომპონენტისაგან:

• მიმღების მისამართი

• ქვექსელის ნიღაბი

• კარიბჭის (Gateway) მისამართი ან ინტერფეისის სახელი

• მარშრუტის "ღირებულება" ან მეტრიკა

მარშრუტიზაციის პროტოკოლები

საკუთარი ინტერფეისისა და სხვა მარშრუტიზატორებისაგან მომავალი ინფორმაციის დინამიური მართვისათვის გამოიყენება მარშრუტიზაციის პროტოკოლები

დინამიური მარშრუტიზაცია ცვლის მარშრუტების სტატიკური დანიშვნის შრომატევად საქმიანობას, ქსელური ადმინისტრატორის ჩარევის გარეშე

მარშრუტიზაციის ალგორითმი

ინფორმაციის ანალიზი, ორი ძირითადი კრიტერიუმის საფუძველზე:

- მანძილი - რამდენად დაშორებულია მანძილი მოცემული მარშრუტიზატორიდან
- ვექტორი - რა მიმართულებითაა მიზანშეწონილი მოცემულ ქსელში პაკეტების გადამისამართება?

მანძილი მარშრუტში წარმოჩინდება ღირებულებით ან მეტრიკით, რომელიც შეიძლება ახასიათებს ერთ-ერთს შემდეგი პარამეტრებიდან:

- გადასვლების რიცხვი
- ადმინისტრაციული ირიბი ხარჯები
- გამტარობა

- გადაცემის სიჩქარე
- შეფერხების ალბათობა
- საიმედოობა
- პირდაპირი მარშრუტი

მარშრუტიზატორის ჩართვისთანავე სამუშაო რეჟიმში ერთვება მისი გამართული ინტერფეისები და მარშრუტიზაციის ცხრილში ინახება მასთან უშუალოდ მიერთებული ლოკალური ქსელის მისამართები, პირდაპირი მარშრუტების სახით.

Cisco-ს მარშრუტიზატორებში ამგვარი მარშრუტები აღინიშნება პრეფიქსით C. ისინი ავტომატურად ახლდება ხელახალი გამართვისას ან მარშრუტის გამორთვისას

სტატიკური მარშრუტი

ქსელის ადმინისტრატორს შეუძლია ხელით გამართოს სტატიკური მარშრუტი კონკრეტულ ქსელში. სტატიკური მარშრუტი არ იცვლება, მანამ სანამ ადმინისტრატორი არ შეცვლის მას.

მარშრუტიზაციის ცხრილში ეს მარშრუტი აღინიშნება პრეფიქსით S

“Default” მარშრუტი

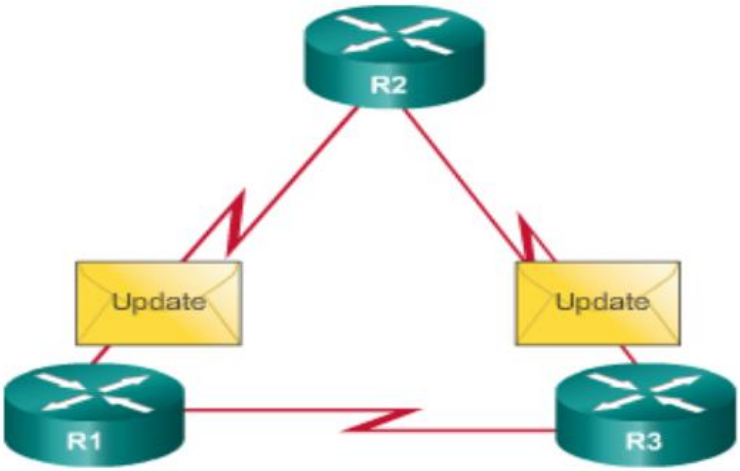
ქსელებისთვის, რომელთა გზაც არ არის ასახული მარშრუტიზაციის ცხრილში, გამოიყენება კარიბჭის მისამართი Default მითითებული . ჩვეულებრივ ეს არის შემდეგი მარშრუტიზატორი ISP-ისკენ გზაზე, თუ ქსელში მხოლოდ ერთი მარშრუტიზატორია, ავტომატურად ხდება მისი ამორჩევა

მარშრუტიზაციის ცხრილში ეს მარშრუტი აღინიშნება პრეფიქსით S*

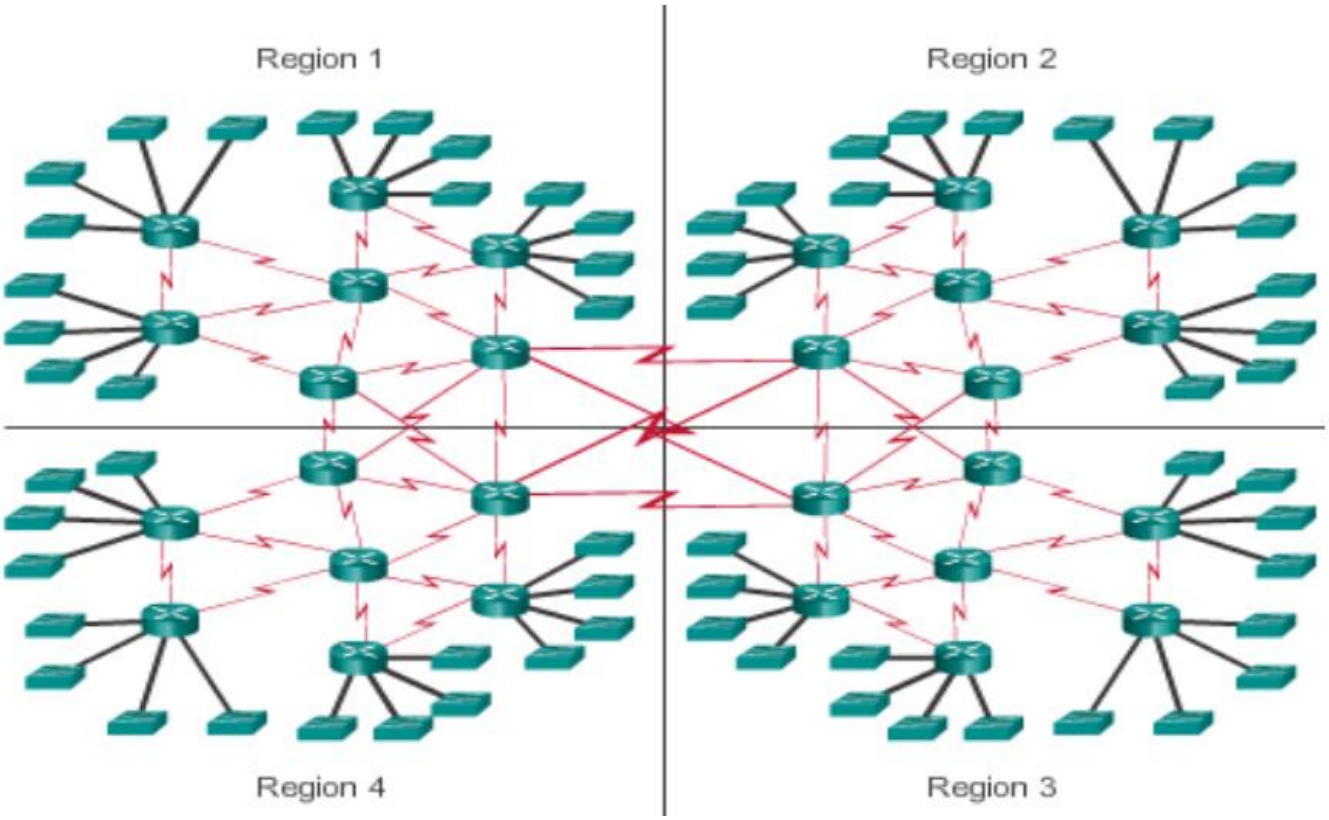
დინამიური (დინამიურად განახლებადი) მარშრუტები

დინამიური მარშრუტები ავტომატურად იქმნება და ახლდება მარშრუტიზაციის პროტოკოლების მიერ. ეს მარშრუტები მარშრუტიზაციის ცხრილში გამოისახება წინსართით რომელიც ასახავს მარშრუტის შემქმნელ პროტოკოლის ტიპს. მაგ.: R აღნიშნავს მარშრუტის ინფორმაციის RIP პროტოკოლს.

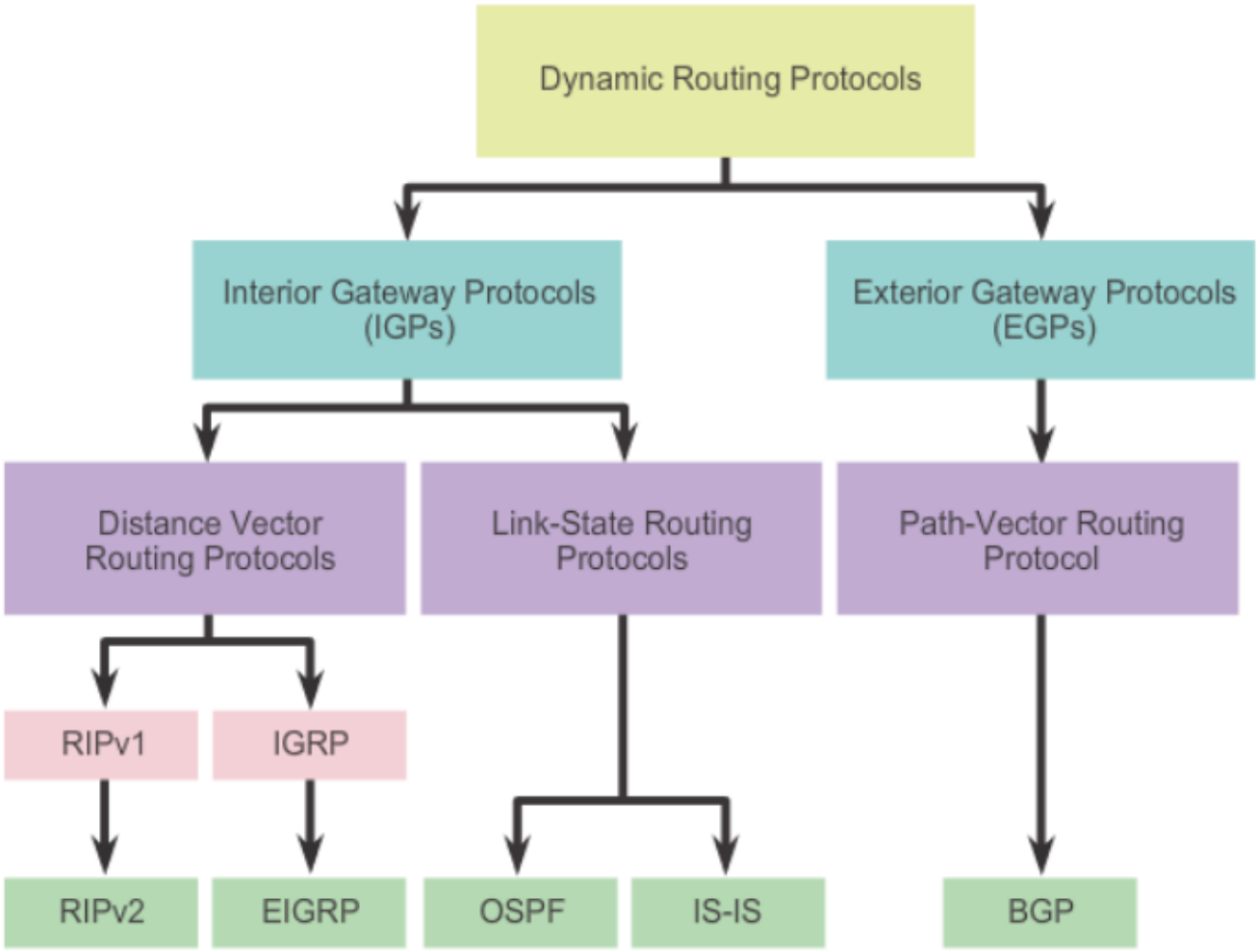
მარშრუტიზაციის პროტოკოლების მეშვეობით მარშრუტიზატორები უზიარებენ ერთმანეთს განახლებებს და დინამიურად ქმნიან და ანახლებენ მარშრუტიზაციის ცხრილებს



დინამიური მარშრუტიზაცია საუკეთესო(შესაბამისობითი) არჩევანია დიდი მასშტაბის ქსელების შემთხვევაში

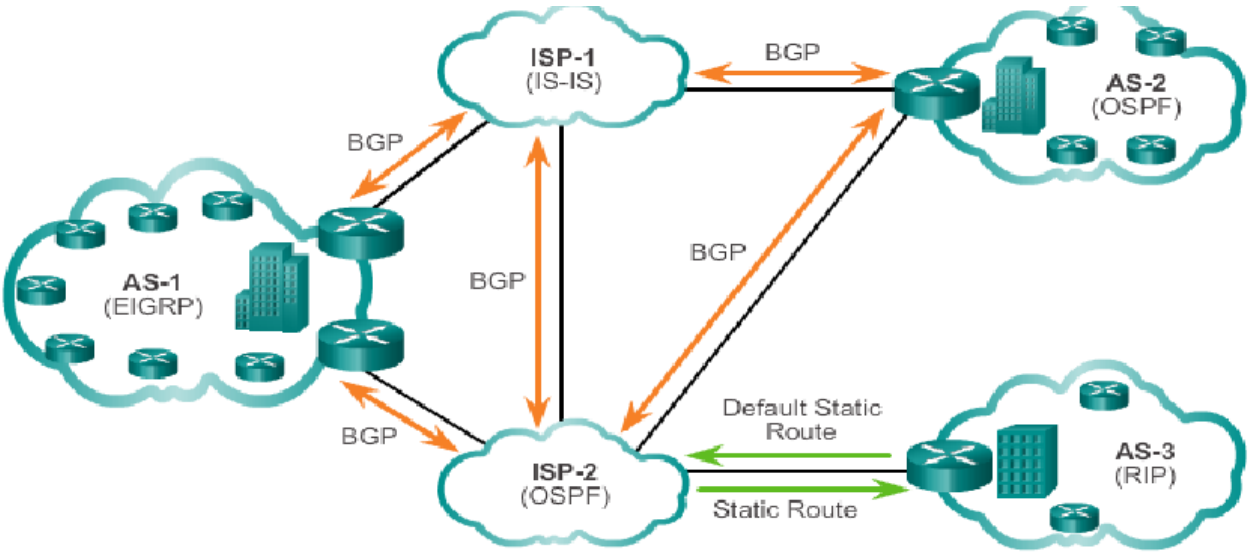


მარშრუტიზაციის პროტოკოლების ტიპები და კლასიფიკაცია



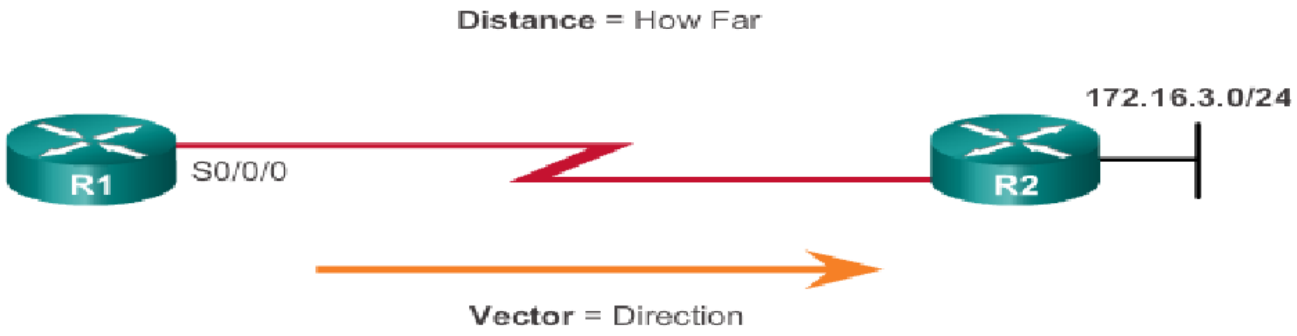
AS(ავტონომიური სისტემა) – ერთი ადმინისტრირების ქვეშ მოქცეული მარშრუტიზატორების ერთობლიობა.

ინტერნეტი დაფუძნებულია ავტონომიური სისტემების AS კონცეპციაზე, ამიტომაც რეალიზებულია პროტოკოლების 2 ჯგუფი: **Interior Gateway Protocols (IGP)** - გამოიყენება ერთი ავტონომიური სისტემის შიგნით მარშუტიზაციისათვის, **Exterior Gateway Protocols (EGP)** - გამოიყენება ავტონომიურ სისტემებს შორის მარშუტიზაციისათვის



Distance vector(IPv4 IGPs: RIPv1 , RIPv2, IGRP, EIGRP) პროტოკოლები ეფუძნება შემდეგ 2 მახასიათებელს:

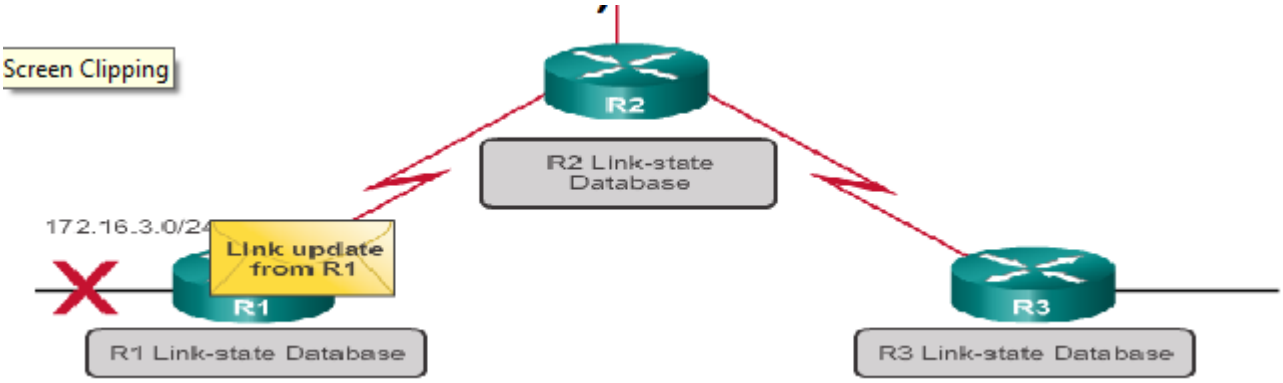
- **Distance** - განსაზღვრავს დანიშნულების ქსელამდე სიშორეს და დაფუძნებულია მეტრიკაზე, როგორიცაა: გადასასვლელების(Hop) რაოდენობა, ღირებულება, გამტარუნარიანობა, დაყოვნება და სხვ.
- **Vector** - განსაზღვრავს შემდეგი გადასასვლელი მარშრუტიზატორის ან გასასვლელის ინტერფეისს



Link-State IPv4 IGPs OSPF; IS-IS - მოცემული პროტოკოლებით კონფიგურირებულ მარშრუტიზატორებს შეუძლიათ შეიქმნან სრული წარმოდგენა ქსელის ტოპოლოგიაზე, ყველა სხვა მარშრუტიზატორიდან ინფორმაციის მიღების გზით.

მოცემული პროტოკოლები განსაკუთრებით ეფექტურია დიდი ზომის იერარქიულ ქსელებში, მაშინ როდესაც ქსელების სწრაფ კონვერგენციას აქვს გადამწყვეტი მნიშვნელობა

- RIP პროტოკოლებით კონფიგურირებული მარშრუტიზატორები პერიოდულად აგზავნიან განახლებებს მეზობელ როუტერებზე, მაშინ როდესაც Link-State განახლებები იგზავნება ქსელის ტოპოლოგიაში ცვლილების მოხდენის შემდეგ.



მარშრუტიზაციის პროტოკოლები შეგვიძლია შევადაროთ შემდეგი მახასიათებლების მიხედვით:

Speed of Convergence - კონვერგენციის სიჩქარე განსაზღვრავს თუ რამდენად სწრაფად გაუზიარებენ მარშრუტიზატორები ქსელში მომხდარ ცვლილებებს

Scalability(მასშტაბურობა) - რამდენად მასშტაბური შეიძლება იყოს ქსელი

Classful or Classless (Use of VLSM) - პროტოკოლები მარშრუტიზაციისას ქვექსელის ნიღაბის მხარდაჭერით (Classful) და მის გარეშე (Classless)

Resource Usage - რესურსების გამოყენება ეხმიანება მარშრუტიზაციის პროტოკოლების მოთხოვნებს, მათ შორის - მეხსიერების(RAM) ზომა, პროცესორის სისწრაფე, გამტარუნარიანობა...

Implementation and Maintenance - რეალიზაცია და მომსახურება აღწერს ქსელის ადმინისტრატორის ცოდნის დონეს , რათა მან შეძლოს მარშრუტიზაციის მოცემული პროტოკოლების საფუძველზე, ქსელის სათანადო მართვა

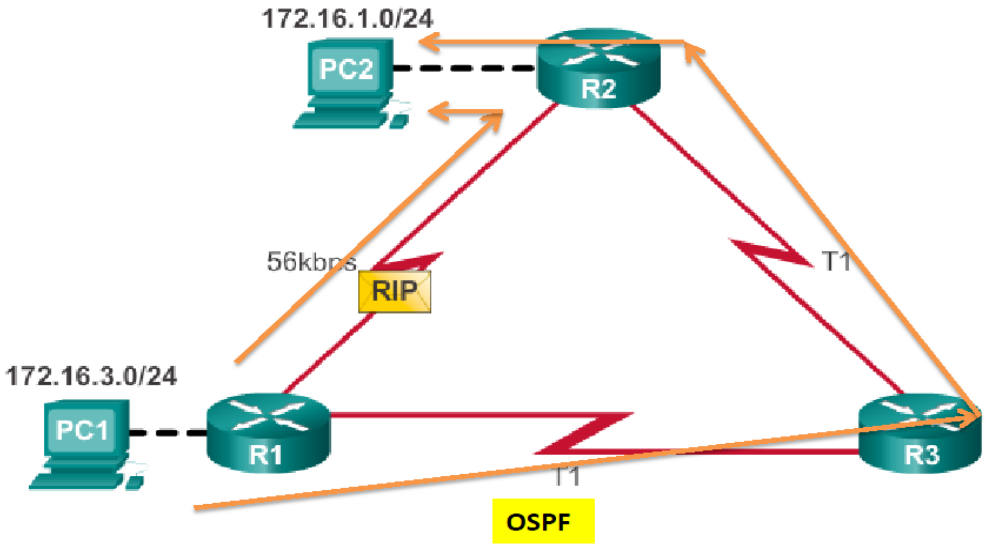
მარშრუტიზაციის პროტოკოლების შედარება

	Distance Vector				Link State	
	RIPv1	RIPv2	IGRP	EIGRP	OSPF	IS-IS
Speed Convergence	Slow	Slow	Slow	Fast	Fast	Fast
Scalability - Size of Network	Small	Small	Small	Large	Large	Large
Use of VLSM	No	Yes	No	Yes	Yes	Yes
Resource Usage	Low	Low	Low	Medium	High	High
Implementation and Maintenance	Simple	Simple	Simple	Complex	Complex	Complex

მარშრუტიზაციის პროტოკოლების მეტრიკა

სხვადასხვა პროტოკოლები იყენებენ განსხვავებულ მეტრიკას დანიშნულების მისამართამდე მარშრუტის განსაზღვრისას

- მაგ.: RIP პროტოკოლი განსაზღვრავს მარშრუტს გადასასვლელების(Hop) რაოდენობის მიხედვით, მაშინ როდესაც OSPF პროტოკოლი ირჩევს მარშრუტს გამტარუნარიანობაზე (bandwidth) დაყრდნობით



1.5. სტატიკური მარშრუტიზაციის საბაზისო კონფიგურაცია

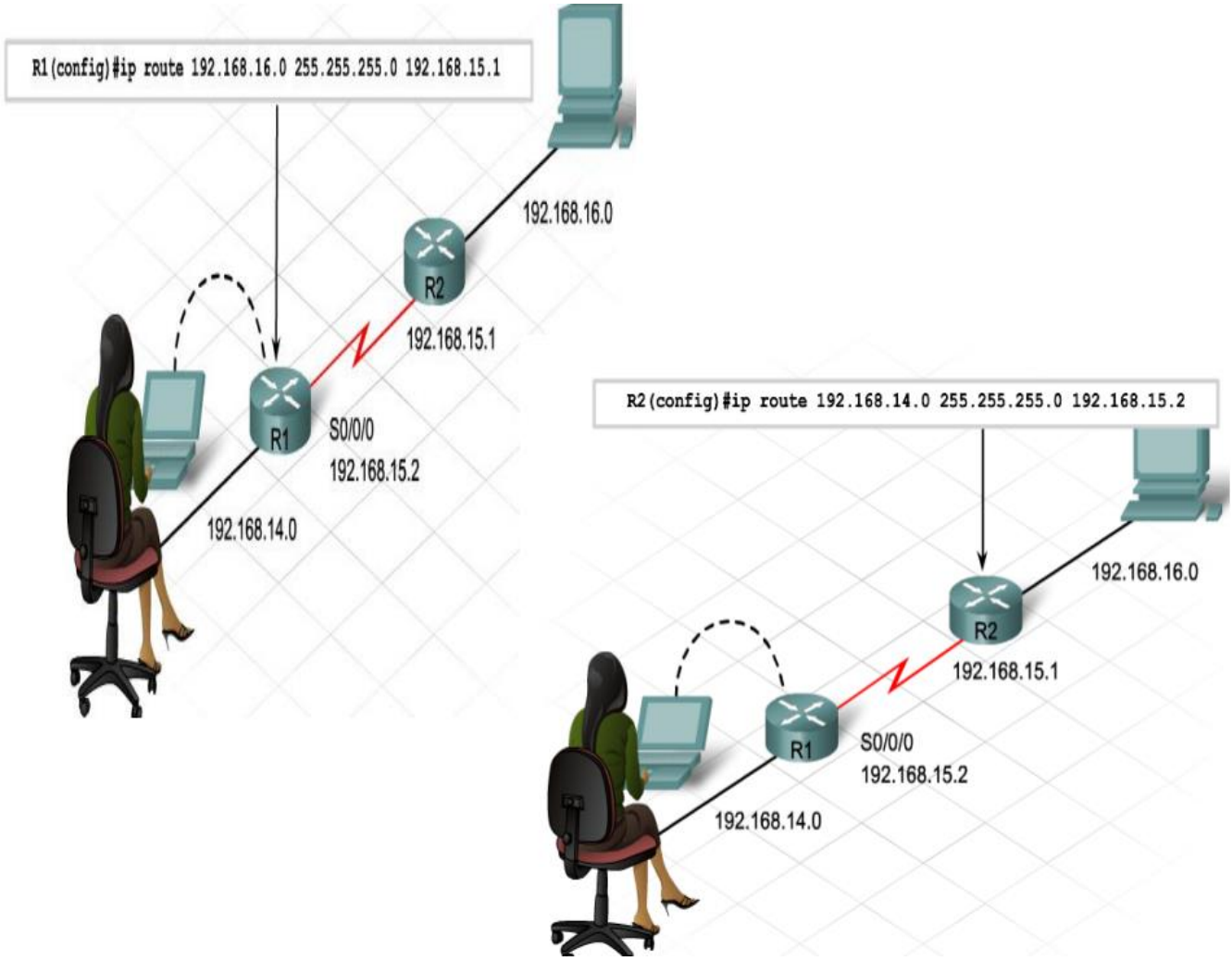
პრაქტიკული სავარჯიშო

დაინიშნოს 192.168.16.0/24 ქსელის კვანძებისთვის 192.168.15.1 მისამართის მქონე ინტერფეისი მარშრუტიზატორი, როგორც გამავალი ინტერფეისი

```
Router1>enable
Router1#config terminal
Router1(config)#ip route 192.168.16.0 255.255.255.0
192.168.15.1
```

ან

```
Router1(config)#ip route 192.168.16.0 255.255.255.0
S0/0/0
```


R1(config)#ip route 192.168.16.0 255.255.255.0 192.168.15.1
192.168.16.0
R2
192.168.15.1
R1
S0/0/0
192.168.15.2
192.168.14.0
R2(config)#ip route 192.168.14.0 255.255.255.0 192.168.15.2
192.168.16.0
R2
192.168.15.1
R1
S0/0/0
192.168.15.2
192.168.14.0

პრაქტიკული დავალება

ქვემოთ მოცემული მარშრუტიზაციის ცხრილების მიხედვით შექმენით ქსელის ფიზიკური და ლოგიკური მოდელი

Router 1

Router#show ip route

Gateway of last resort is 192.168.0.2 to network 0.0.0.0

10.0.0.0/25 is subnetted, 2 subnets

C 10.0.0.0 is directly connected, FastEthernet0/0

C 10.0.0.128 is directly connected, FastEthernet0/1

192.168.0.0/30 is subnetted, 1 subnets

C 192.168.0.0 is directly connected, Serial0/0/0

S* 0.0.0.0/0 [1/0] via 192.168.0.2

Router 2

Router#show ip route

Gateway of last resort is not set

10.0.0.0/25 is subnetted, 2 subnets

S 10.0.0.0 [1/0] via 192.168.0.1

S 10.0.0.128 [1/0] via 192.168.0.1

172.16.0.0/25 is subnetted, 2 subnets

S 172.16.0.0 [1/0] via 192.168.0.6

S 172.16.0.128 [1/0] via 192.168.0.6

192.168.0.0/30 is subnetted, 2 subnets

C 192.168.0.0 is directly connected, Serial0/0/0

C 192.168.0.4 is directly connected, Serial0/0/1

Router 3

Router#SHOW IP ROute

Gateway of last resort is 192.168.0.5 to network 0.0.0.0

172.16.0.0/25 is subnetted, 2 subnets

C 172.16.0.0 is directly connected, FastEthernet0/0

C 172.16.0.128 is directly connected, FastEthernet0/1

192.168.0.0/30 is subnetted, 1 subnets

C 192.168.0.4 is directly connected, Serial0/0/1

S* 0.0.0.0/0 [1/0] via 192.168.0.5

პროცესზე დაკვირვება

შემფასებელი აკვირდება შესაფასებელ პირის მუშაობას პროფესიული სტანდარტით(პროგრამით / მოდულით) განსაზღვრული ამოცანების შესრულების პროცესში . დაკვირვებახორციელდება სამუშაო ადგილზე ან სამუშაო პირობებში, თუმცა დასაშვებია ჩატარდესკომპიუტერებით აღჭურვილლაბორატორიაში,სადაც შესაფასებელი პირი პრაქტიკულ საქმიანობას ეწევა. შემფასებელმა წინასწარ უნდა დაგეგმოს დაკვირვების პროცესი, იმის დასადგენად, თუ რამდენად სწორად იყენებს შესაფასებელი პირი ცოდნას, უნარებსა და ყველა რესურსს შედეგის მისაღწევად.შეფასება დადებითია იმ შემთხვევაში თუ სტუდენტი ყველა კითხვას სწორად გასცემს პასუხს.

სტუდენტს მიეცა პრაქტიკული სავარჯიშო - **მესამე დონის პროტოკოლების გამოყენებით საბაზისო მარშუტიზაციის განხორციელება ყველა პროტოკოლის გამოყენებით**

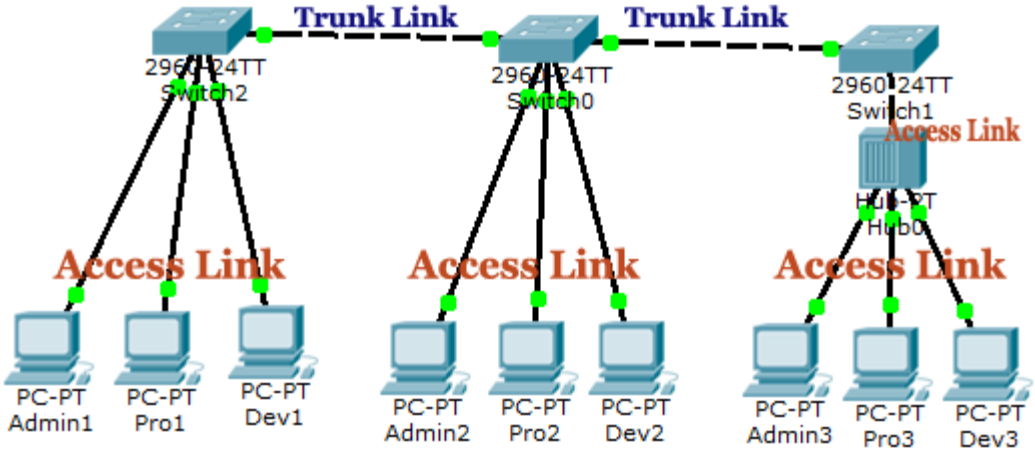
სწავლის შედეგი	დასახელება	შეფასება	
		კი	არა
სტატიკური მარშუტიზაციის საბაზისო კონფიგურაცია	სწორად დაკონფიგა სტანდარტული მარშუტი(Default Route)		
	დროულად დაადგინა და სწორად აღმოფხვრა სტატიკური მარშუტიზაციის პრობლემები.		
მარშუტიზაციის პროტოკოლი RIP -ის საბაზისო კონფიგურაცია	სწორად მოახდინა მარშუტების განაწილება RIP მარშუტიზაციის პროტოკოლის საშუალებით		
	დროულად დაადგინა და სწორად აღმოფხვრა მარშუტიზაციის პროცესში წარმოქმნილ პრობლემები		
მარშუტიზაციის პროტოკოლი EIGRP -ის საბაზისო კონფიგურაცია	სწორად მოახდინა მარშუტების განაწილება EIGRP მარშუტიზაციის პროტოკოლის საშუალებით		
	დროულად დაადგინა და სწორად აღმოფხვრა მარშუტიზაციის პროცესში წარმოქმნილ პრობლემები		
მარშუტიზაციის პროტოკოლი OSPF -ის საბაზისო კონფიგურაცია	სწორად მოახდინა მარშუტების განაწილება OSPF მარშუტიზაციის პროტოკოლის საშუალებით		
	სწორად მოახდინა მარშუტების განაწილება OSPF მარშუტიზაციის პროტოკოლის საშუალებით		

2. მეორე დონის პროტოკოლები

2.1. Trunk პროტოკოლი

ტელეკომუნიკაციაში Trunking - წარმოადგენს მეთოდს, რომელიც უზრუნველყოფს მრავალი კლიენტის წვდომას ქსელში ხაზების გაზიარებით, მათი ინდივიდუალური გამოყენების ნაცვლად.

Trunk – წარმოადგენს ორ წერტილს შორის კავშირის არხს, თითოეული წერტილი წარმოადგენს კომუტაციის ცენტრს ან კვანძს.

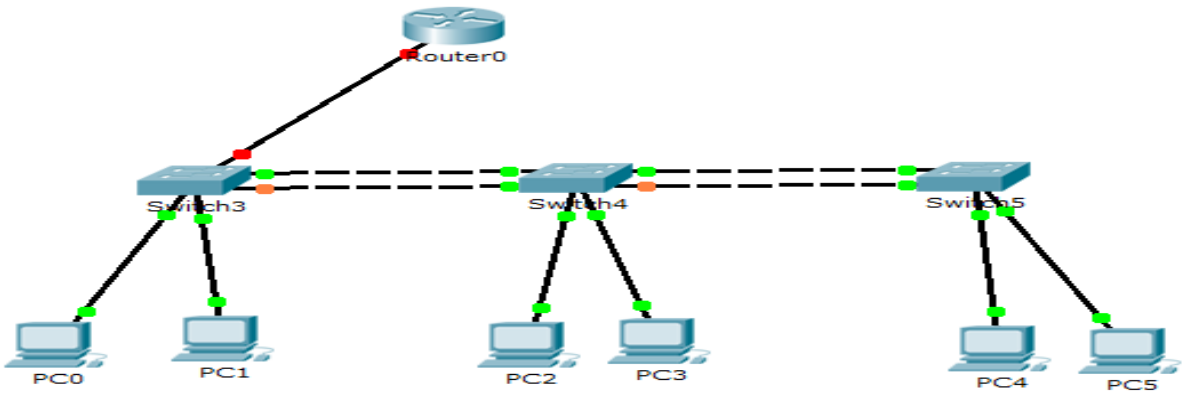


სურ.2.1. 1

განვიხილოთ VLAN ქსელის ფორმირება VTP (VLAN Trunking Protocol) და DTP (Dynamic Trunking Protocol) პროტოკოლების გამოყენებით

პრაქტიკული სავარჯიშო

ქვემოთ მოცემულის შესაბამისად(იხ. სურ.2.1.2) შექმენით ქსელის მოდელი და მოახდინეთ VLAN კონფიგურირება



სურ.2.1. 2

PCs Configuration

Device	IP Address	Subnet Mask	Gateway	VLAN	Connected With
PC0	10.0.0.2	255.0.0.0	10.0.0.1	VLAN 10	Office 1 Switch on F0/1
PC1	20.0.0.2	255.0.0.0	20.0.0.1	VLAN 20	Office 1 Switch on F0/2
PC2	10.0.0.3	255.0.0.0	10.0.0.1	VLAN 10	Office 2 Switch on F0/1
PC3	20.0.0.3	255.0.0.0	20.0.0.1	VLAN 20	Office 2 Switch on F0/2
PC4	10.0.0.4	255.0.0.0	10.0.0.1	VLAN 10	Office 3 Switch on F0/1
PC5	20.0.0.4	255.0.0.0	20.0.0.1	VLAN 20	Office 3 Switch on F0/2

Office 1 Switch Configuration

Port	Connected To	VLAN	Link	Status
F0/1	With PC0	VLAN 10	Access	OK
F0/2	With PC1	VLAN 20	Access	OK
Gig1/1	With Router	VLAN 10,20	Trunk	OK
Gig 1/2	With Switch2	VLAN 10,20	Trunk	OK
F0/24	Witch Switch2	VLAN 10,20	Trunk	STP - Blocked

Office 3 Switch Configuration

Port	Connected To	VLAN	Link	Status
F0/1	With PC0	VLAN 10	Access	OK
F0/2	With PC1	VLAN 20	Access	OK
Gig 1/1	With Switch2	VLAN 10,20	Trunk	OK
F0/24	Witch Switch1	VLAN 10,20	Trunk	STP - Blocked

სურ.2.1. 3

Office 2 Switch Configuration

Port	Connected To	VLAN	Link	Status
F0/1	With PC0	VLAN 10	Access	OK
F0/2	With PC1	VLAN 20	Access	OK
Gig 1/2	With Switch1	VLAN 10,20	Trunk	OK
Gig 1/1	With Switch3	VLAN 10,20	Trunk	OK
F0/24	Witch Switch1	VLAN 10,20	Trunk	STP - Blocked
F0/23	Witch Switch3	VLAN 10,20	Trunk	STP - Blocked

Router Configuration

Port	Connected To	VLAN	Link	Status
Fa0/0	with Office 1 Switch Gig 1/2	VLAN 10, 20	Trunk	Ok

VLAN Configuration

VLAN Number	VLAN Name	Gateway IP	PCs
10	Sales	10.0.0.1	PC0,PC2,PC4
20	Management	20.0.0.1	PC1,PC3,PC5

სურ.2.1. 4

VTP Server-ის კონფიგურირება

- კომუტატორს შეურჩიეთ hostname S1
- შეურჩიეთ domain name example
- შეურჩიეთ პაროლი student.

```
Switch>enable
Switch#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#hostname S1
S1(config)#vtp mode server
Device mode already VTP SERVER.
S1(config)#vtp domain example
Changing VTP domain name from NULL to example
S1(config)#vtp password vinita
Setting device VLAN database password to STUDENT
```

VTP Client-ის კონფიგურირება

```
Switch>enable
Switch#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#hostname S2
S2(config)#vtp mode client
Setting device to VTP CLIENT mode.
S2(config)#vtp domain example
Changing VTP domain name from NULL to example
S2(config)#vtp password vinita
Setting device VLAN database password to STUDENT
```

```
S2(config)#


Switch>enable
Switch#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#hostname S3
S3(config)#vtp mode client
Setting device to VTP CLIENT mode.
S3(config)#vtp domain example
Changing VTP domain name from NULL to example
S3(config)#vtp password vinita
Setting device VLAN database password to vinita
S3(config)#
```

კომუტატორებს შორის DTP კონფიგურირება

ქვემოთ ცხრილში მოცემულია კომუტატორებზე trunk ინტერფეისები (ასახულია სურ.2.1.3 და სურ.2.1.4)

Switch	Interfaces
Office 1	Gig1/1, Gig1/2, F0/24
Office 2	Gig1/1, Gig1/2, F0/23, F0/24
Office 3	Gig1/1, Gig1/2

Office 1 Switch

```
S1(config)#interface fastEthernet 0/24
S1(config-if)#switchport mode trunk
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/24,
changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/24,
changed state to up
S1(config-if)#exit
S1(config)#interface gigabitEthernet 1/1
S1(config-if)#switchport mode trunk
S1(config-if)#exit
S1(config)#interface gigabitEthernet 1/2
S1(config-if)#switchport mode trunk
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet1/2,
changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet1/2,
changed state to up
```

```
S1(config-if)#exit
S1(config)#
```

Office 2 Switch

```
S2(config)#interface gigabitEthernet 1/1
S2(config-if)#switchport mode trunk
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet1/1,
changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet1/1,
changed state to up
S2(config-if)#exit
S2(config)#interface gigabitEthernet 1/2
S2(config-if)#switchport mode trunk
S2(config-if)#exit
S2(config)#interface fastEthernet 0/23
S2(config-if)#switchport mode trunk
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/23,
changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/23,
changed state to up
S2(config-if)#exit
S2(config)#interface fastEthernet 0/24
S2(config-if)#switchport mode trunk
S2(config-if)#exit
```

Office 3 Switch

```
S3(config)#interface fastEthernet 0/24
S3(config-if)#switchport mode trunk
S3(config-if)#exit
S3(config)#interface gigabitEthernet 1/1
S3(config-if)#switchport mode trunk
S3(config-if)#exit
```

VLAN კონფიგურირება

Office 1 Switch

```
S1(config)#vlan 10
S1(config-vlan)#exit
S1(config)#vlan 20
S1(config-vlan)#exit
S1(config)#

S1(config)#interface fastEthernet 0/1
S1(config-if)#switchport access vlan 10
S1(config-if)#interface fastEthernet 0/2
S1(config-if)#switchport access vlan 20
```

Office 2 Switch

```
S2(config)#interface fastEthernet 0/1
```

```
S2(config-if)#switchport access vlan 10
S2(config-if)#interface fastEthernet 0/2
S2(config-if)#switchport access vlan 20
```

Office 3 Switch

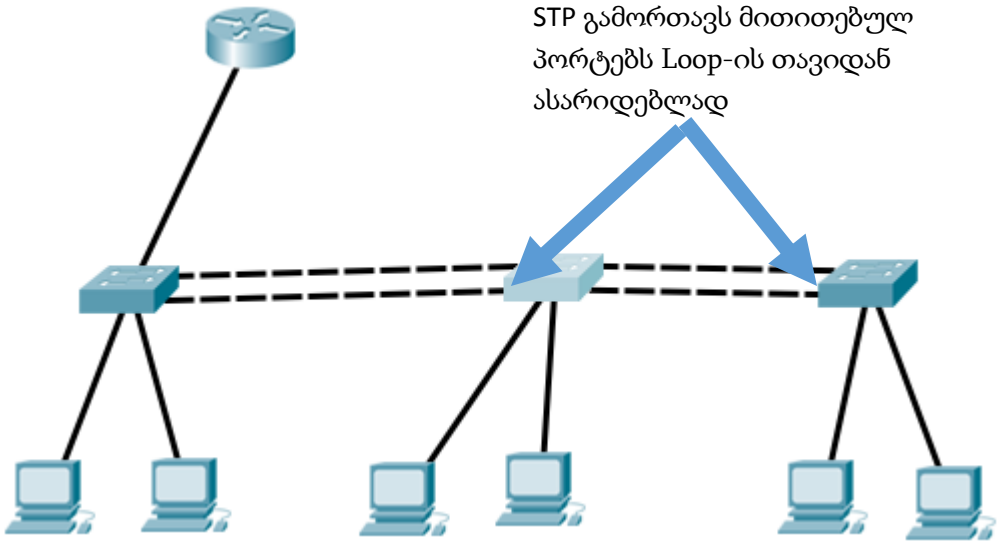
```
S3(config)#interface fastEthernet 0/1
S3(config-if)#switchport access vlan 10
S3(config-if)#interface fastEthernet 0/2
S3(config-if)#switchport access vlan 20
```

მარშრუტიზატორის კონფიგურირება

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface fastEthernet 0/0
Router(config-if)#no ip address
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#interface fastEthernet 0/0.10
Router(config-subif)#encapsulation dot1Q 10
Router(config-subif)#ip address 10.0.0.1 255.0.0.0
Router(config-subif)#exit
Router(config)#interface fastEthernet 0/0.20
Router(config-subif)#encapsulation dot1Q 20
Router(config-subif)#ip address 20.0.0.1 255.0.0.0
Router(config-subif)#exit
```

2.2. STP პროტოკოლი

Spanning Tree Protocol (STP) მე-2 დონის პროტოკოლის სპეციფიკაციაა IEEE 802.1D. STP-ს ძირითადი მიზანია ქსელში არსებული ჭარბი მარშრუტების შემთხვევაში „Loop“-ების აცილება.



სურ.2.2. 1

ცხრილში მოცემულია STP კონფიგურაციის ნაგულისხმევი პარამეტრები

Feature	Default Setting
Enable state	Enabled on VLAN 1. Up to 64 spanning-tree instances can be enabled.
Switch priority	32768.
Spanning-tree port priority (configurable on a per-interface basis—used on interfaces configured as Layer 2 access ports)	128.
Spanning-tree port cost (configurable on a per-interface basis—used on interfaces configured as Layer 2 access ports)	1000 Mbps: 4. 100 Mbps: 19. 10 Mbps: 100.
Spanning-tree VLAN port priority (configurable on a per-VLAN basis—used on interfaces configured as Layer 2 trunk ports)	128.
Spanning-tree VLAN port cost (configurable on a per-VLAN basis—used on interfaces configured as Layer 2 trunk ports)	1000 Mbps: 4. 100 Mbps: 19. 10 Mbps: 100.
Hello time	2 seconds.
Forward-delay time	15 seconds.
Maximum-aging time	20 seconds.

Root Switch კონფიგურირება

ნაბიჯი 1. გლობალური კონფიგურაციის რეჟიმში გადასვლა **configure Terminal**

ნაბიჯი 2. ბრძანება **spanning-tree vlan vlan-id root primary** [diameter net-diameter [hello-time seconds]]

Secondary Root Switch კონფიგურირება

ნაბიჯი 1. გლობალური კონფიგურაციის რეჟიმში გადასვლა **configure Terminal**

ნაბიჯი 2. ბრძანება **spanning-tree vlan vlan-id root secondary** [diameter net-diameter [hello-time seconds]]

პორტების პრიორიტეტების კონფიგურირება

ნაბიჯი 1. გლობალური კონფიგურაციის რეჟიმში გადასვლა **configure Terminal**

ნაბიჯი 2. ბრძანება **interface** interface-id

ნაბიჯი 3. ბრძანება **spanning-tree port-priority** priority (0-დან 255; უპირატესია დაბალი რიცხვითი ჩანაწერი, ნაგულისხმევად მინიჭებული პრიორიტეტია 128)

ნაბიჯი 4. ბრძანება **spanning-tree vlan** vlan-id **port-priority** priority

მარშრუტის ღირებულების(Path Cost) კონფიგურირება

ნაბიჯი 1. გლობალური კონფიგურაციის რეჟიმში გადასვლა **configure Terminal**

ნაბიჯი 2. ბრძანება **interface** interface-id

ნაბიჯი 3. ბრძანება **spanning-tree cost** cost (1-დან 2000000000-მდე; ნაგულისხმევად განისაზღვრება ინტერფეისის სიჩქარით)

ნაბიჯი 4. ბრძანება **spanning-tree vlan** vlan-id **cost** cost

VLAN-ის კომუტატორების პრიორიტეტულობის კონფიგურირება

ნაბიჯი 1. გლობალური კონფიგურაციის რეჟიმში გადასვლა **configure Terminal**

ნაბიჯი 2. ბრძანება **spanning-tree vlan** vlan-id **priority** priority ((1-დან 61440-მდე; ნაგულისხმევად არის 32768, უმცირესი რიცხვი შეირჩევა Root Switch-სთვის)

Hello Time კონფიგურირება

ნაბიჯი 1. გლობალური კონფიგურაციის რეჟიმში გადასვლა **configure Terminal**

ნაბიჯი 2. ბრძანება **spanning-tree vlan** vlan-id **hello-time** seconds (1-დან 10-მდე; ნაგულისხმევად არის 2)

VLAN-ისთვის Forwarding-Delay Time კონფიგურირება

ნაბიჯი 1. გლობალური კონფიგურაციის რეჟიმში გადასვლა **configure Terminal**

ნაბიჯი 2. ბრძანება **spanning-tree vlan** vlan-id **forward-time** seconds (4-დან 30-მდე; ნაგულისხმევად არის 15)

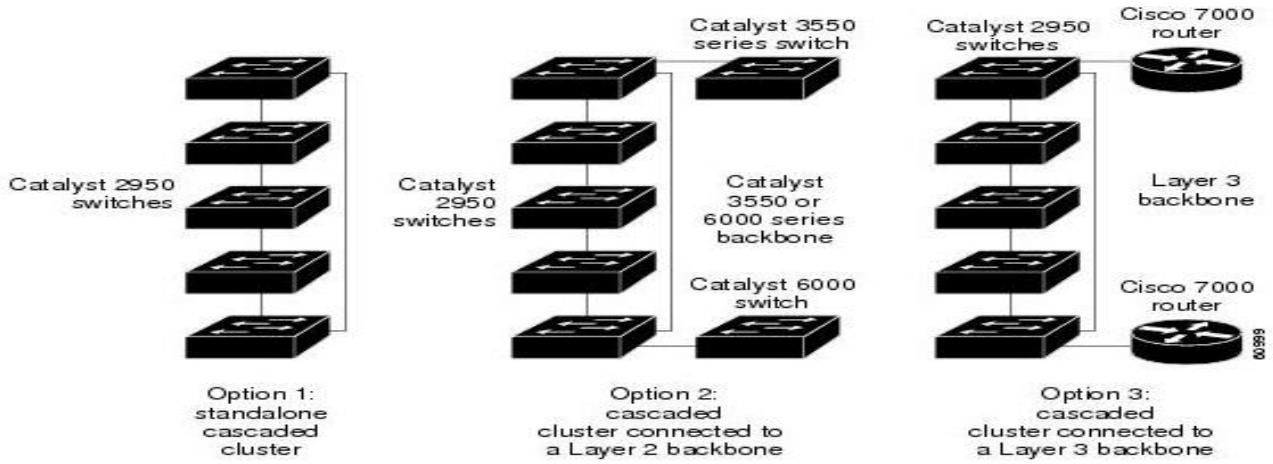
VLAN-ისთვის Maximum-Aging Time კონფიგურირება

ნაბიჯი 1. გლობალური კონფიგურაციის რეჟიმში გადასვლა **configure Terminal**

ნაბიჯი 2. ბრძანება **spanning-tree vlan** vlan-id **max-age seconds** seconds (6-დან 40-მდე; ნაგულისხმევად არის 20)

ცხრილში ნაჩვენებია ნაგულისხევი და რეკომენდირებული პარამეტრები, სურ.2.2.2 ასახული კომუტატორების განლაგებისთვის

STP Parameter	STP Default	Acceptable for Option 1	Acceptable for Option 2	Acceptable for Option 3
Hello Time	2	1	1	1
Max Age	20	6	10	6
Forwarding Delay	15	4	7	4

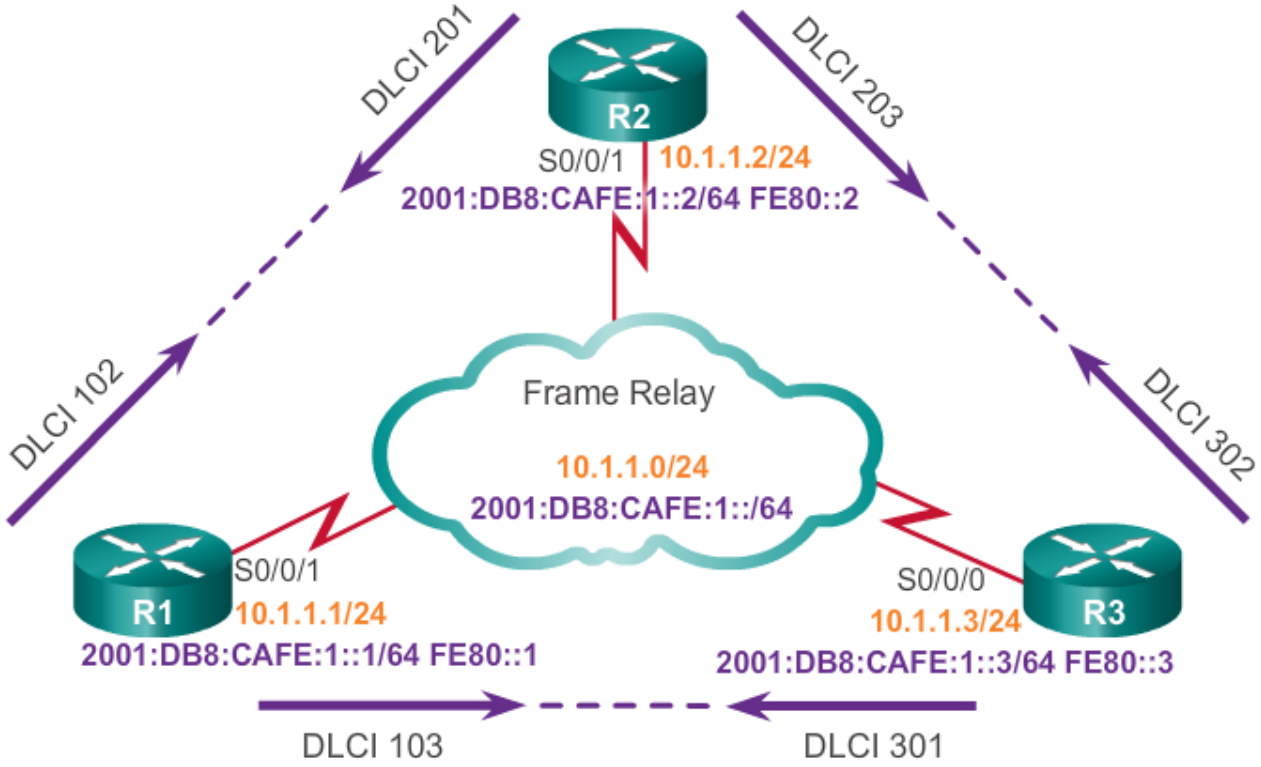


სურ.2.2. 2

2.3. Frame Relay პროტოკოლი

Frame Relay - მარალეფექტური WAN პროტოკოლია, რომელიც მუშაობს OSI მოდელის ფიზიკურ და არხულ დონეზე. მიუხედავად იმისა, რომ ახალმა სერვისებმა, როგორიცაა broadband და metro Ethernet შეამცირეს მისი გამოყენების არეალი, Frame Relay მაინც ინარჩუნებს აქტუალობას მსოფლიოს სხვადასხვა წერტილში.

Frame Relay კონფიგურირება



სურ.2.3. 1

ნაბიჯი 1. ინტერფეისების დამისამართება

```
R1(config)# interface Serial0/0/1
R1(config-if)# bandwidth 64
R1(config-if)# ip address 10.1.1.1 255.255.255.0
R1(config-if)# ipv6 address 2001:db8:cafe:1::1/64
R1(config-if)# ipv6 address fe80::1 link-local
R1(config-if)# encapsulation frame-relay
```

```
R2(config)# interface Serial0/0/1
R2(config-if)# bandwidth 64
R2(config-if)# ip address 10.1.1.2 255.255.255.0
R2(config-if)# ipv6 address 2001:db8:cafe:1::2/64
R2(config-if)# ipv6 address fe80::2 link-local
R2(config-if)# encapsulation frame-relay
```

```
R1# show interfaces serial 0/0/1
Serial0/0/1 is up, line protocol is up
  Hardware is GT96K Serial
  Internet address is 10.1.1.1/24
  MTU 1500 bytes, BW 64 Kbit/sec, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation FRAME-RELAY, loopback not set
  Keepalive set (10 sec)
  CRC checking enabled
  LMI enq sent  481, LMI stat recvd 483, LMI upd recvd 0, DTE LMI up
  LMI enq recvd 0, LMI stat sent  0, LMI upd sent  0
  LMI DLCI 1023  LMI type is CISCO  frame relay DTE
  FR SVC disabled, LAPF state down
  Broadcast queue 0/64, broadcasts sent/dropped 1/0, interface
```

ნაბიჯი 2. ინკაფსულაციის კონფიგურირება

სათანადო ინტერფეისზე Frame–relay ინკაფსულაციის დანიშვნის ბრძანებაა: **encapsulation frame-relay** [cisco | ietf]

- Cisco - ინკაფსულაციის ტიპი წარმოადგენს ნაგულისხმევ Frame-relay ინკაფსულაციას, გამოიყენეთ ეს ტიპი თუ ვუკავშირდებით ასევე ცისკოს მარშრუტიზატორს
- Ietf - ინკაფსულაციის ტიპი შეესაბამება RFC 1490 და RFC 2427 სპეციფიკაციას, გამოიყენეთ ეს ტიპი თუ დასაკავშირებელი მოწყობილობა არ არის ცისკოს მარშრუტიზატორი

ნაბიჯი 3. გამტარუნარიანობის დანიშვნა

მაგ.: **bandwidth 64**

ნაბიჯი 4. LMI ტიპის დანიშვნა

LMI ტიპებია: cisco, ANSI Annex D, და Q933-A Annex A, ნაგულისხმევი ტიპი გახლავთ cisco

Frame-relay კონფიგურაციის მაგალითი იხილეთ სურათზე.:

```
R1(config)# interface Serial0/0/1
R1(config-if)# bandwidth 64
R1(config-if)# ip address 10.1.1.1 255.255.255.0
R1(config-if)# ipv6 address 2001:db8:cafe:1::1/64
R1(config-if)# ipv6 address fe80::1 link-local
R1(config-if)# encapsulation frame-relay
```

```
R2(config)# interface Serial0/0/1
R2(config-if)# bandwidth 64
R2(config-if)# ip address 10.1.1.2 255.255.255.0
R2(config-if)# ipv6 address 2001:db8:cafe:1::2/64
R2(config-if)# ipv6 address fe80::2 link-local
R2(config-if)# encapsulation frame-relay
```

ნაბიჯი 5. სტატიკური Frame-relay რუქის კონფიგურირება

```
frame-relay map protocol protocol-address dlci [broadcast]
```

```
R1(config)# interface Serial0/0/1
R1(config-if)# bandwidth 64
R1(config-if)# ip address 10.1.1.1 255.255.255.0
R1(config-if)# ipv6 address 2001:db8:cafe:1::1/64
R1(config-if)# ipv6 address fe80::1 link-local
R1(config-if)# encapsulation frame-relay
R1(config-if)# frame-relay map ip 10.1.1.2 102 broadcast
R1(config-if)# frame-relay map ipv6 2001:DB8:CAFE:1::2 102
R1(config-if)# frame-relay map ipv6 FE80::2 102 broadcast
```

```
R2(config)# interface Serial0/0/1
R2(config-if)# bandwidth 64
R2(config-if)# ip address 10.1.1.2 255.255.255.0
R2(config-if)# ipv6 address 2001:db8:cafe:1::2/64
R2(config-if)# ipv6 address fe80::2 link-local
R2(config-if)# encapsulation frame-relay
R2(config-if)# frame-relay map ip 10.1.1.1 201 broadcast
R2(config-if)# frame-relay map ipv6 2001:DB8:CAFE:1::1 201
R2(config-if)# frame-relay map ipv6 FE80::1 201 broadcast
```

Frame-relay რუქის შემოწმება

```
R1# show frame-relay map
Serial0/0/1 (up): ipv6 2001:DB8:CAFE:1::2 dlci 102(0x66,0x1860),
                  static, CISCO, status defined, active
Serial0/0/1 (up): ipv6 FE80::2 dlci 102(0x66,0x1860), static,
                  broadcast, CISCO, status defined, active
Serial0/0/1 (up): ip 10.1.1.2 dlci 102(0x66,0x1860), static,
                  broadcast, CISCO, status defined, active
R1#
```

```
R2# show frame-relay map
Serial0/0/1 (up): ipv6 2001:DB8:CAFE:1::1 dlci 201(0xC9,0x3090),
                  static, CISCO, status defined, active
Serial0/0/1 (up): ipv6 FE80::1 dlci 201(0xC9,0x3090), static,
                  broadcast, CISCO, status defined, active
Serial0/0/1 (up): ip 10.1.1.1 dlci 201(0xC9,0x3090), static,
                  broadcast, CISCO, status defined, active
R2#
```

Frame-relay კონფიგურირების ვიდეო ბმულები

ნაწილი 1 - https://www.youtube.com/watch?v=hgYTS1BmHo0

ნაწილი 2 - https://www.youtube.com/watch?v=mjjWp5RjMX0

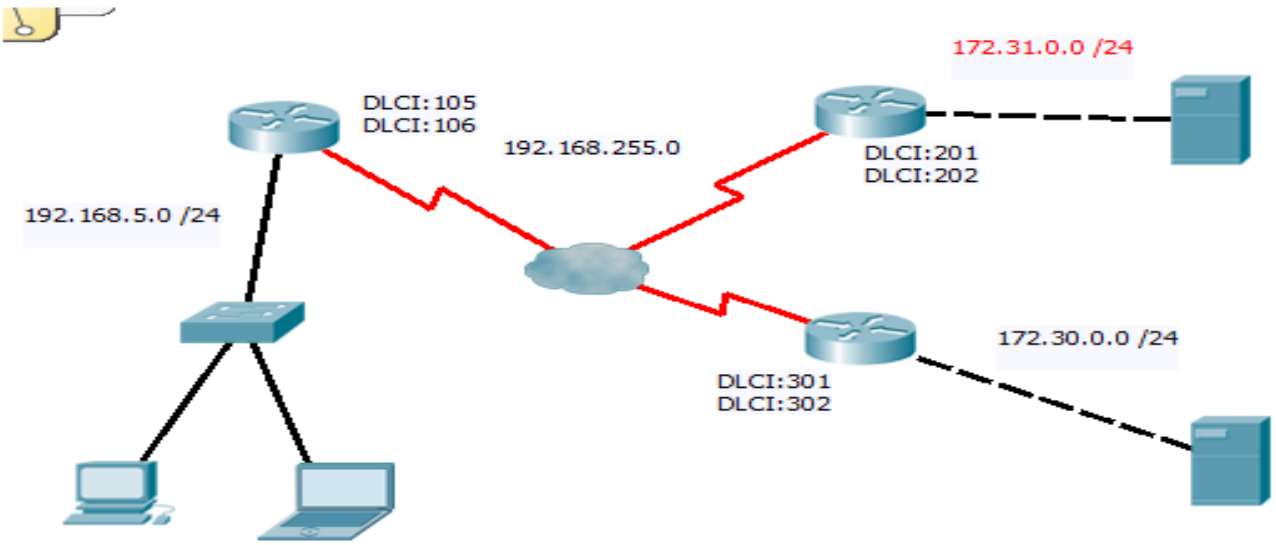
ნაწილი 3 - https://www.youtube.com/watch?v=luEQzT__dAk

ნაწილი 4 - https://www.youtube.com/watch?v=JpUveApljiw

ნაწილი 5 - https://www.youtube.com/watch?v=QPDjqXfsYc0

ნაწილი 6 - https://www.youtube.com/watch?v=IfUr1W7L_Mw

პრაქტიკული სავარჯიშო



- დაამისამართეთ მოწყობილობები მითითებულის შესაბამისად
- როუტერებს შეურჩიეთ ქსელური სახელი: თქვენი სახელი 1 -თქვენი სახელი 2 - თქვენი სახელი 3; ბრძანებათა ველის და პრივილეგირებული რეჟიმის პაროლი: თქვენი გვარი
- როუტერებში გააქტიურეთ RIP პროტოკოლი
- Cloud-ის შესაბამის Serial პორტებზე დაამატეთ მითითებული DLCI მნიშვნელობები სათანადო როუტერებისკენ
- Cloud-ის შესაბამის Frame Relay Conection კონფიგურაციაში მიუთითეთ(დაამატეთ) პორტებისა და Sublink გადასვლების რიგითობა
- როუტერების შესაბამის ინტერფეისებზე გააქტიურეთ Frame Relay ინკაფსულაცია, განსაზღვრეთ გამტარობა, Frame Relay map

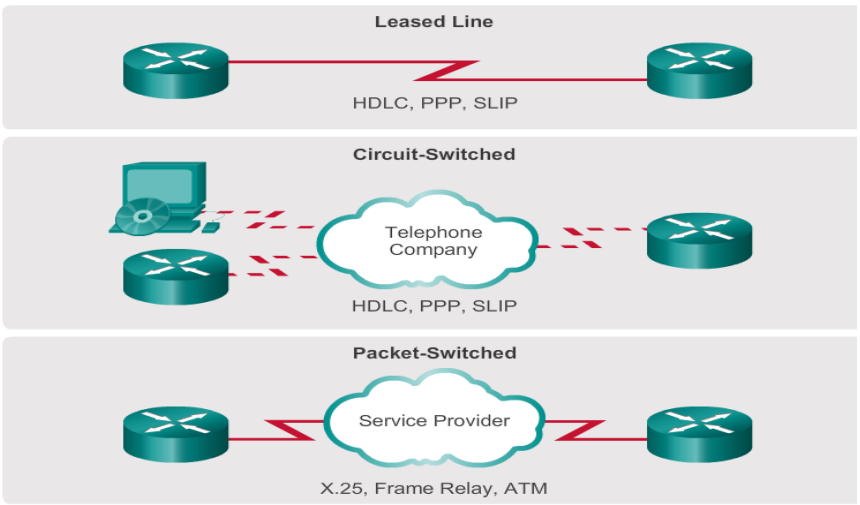
2.4. PPP პროტოკოლი

WAN კავშირის ერთ-ერთ ყველაზე გავრცელებულ ტიპს წარმოადგენს point-to-point კავშირი, რომელსაც ასევე უწოდებენ მიმდევრობით(Serial) ან გამოყოფილი ხაზის კავშირს.(იხ. სურ)

სურ.2.4. 1

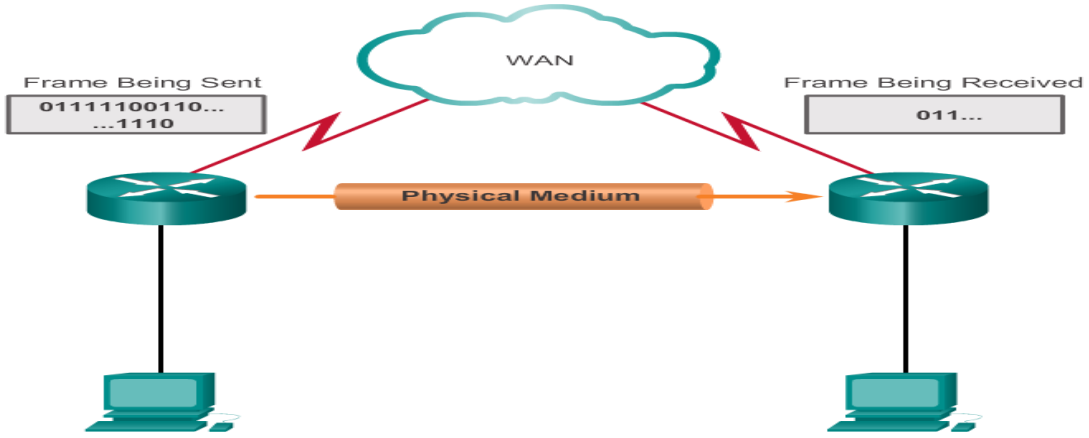
PPP პროტოკოლის ალტერნატივა შეიძლება იყოს HDLC პროტოკოლი, ამ ორი პროტოკოლის შედარებისას გამოარჩევენ შემდეგ უპირატესობებს:

- კონფიგურაციის სიმარტივე
- უსაფრთხოების პარამეტრები
- გამტარუნარიანობა და შეკუმშვა
- გამტარუნარიანობის კონსოლიდაცია
- არადაპატენტებული ქსელური მოწყობილობების მიმართ ადაპტირება



სურ.2.4. 2

სურ.2.4.3 ნაჩვენებია მიმდევრობითი(Serial) კავშირის მარტივი მაგალითი.

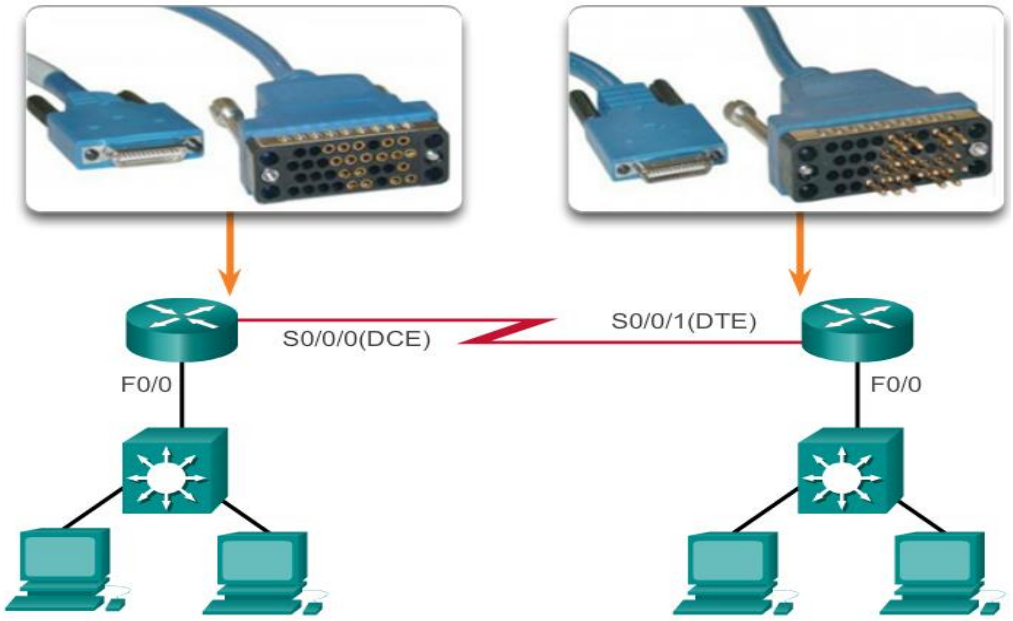


სურ.2.4. 3

მიმდევრობითი კავშირის ერთი მხარე არის DTE მოწყობილობა და მეორე მხარე DCE. ორ DCE მოწყობილობას შორის კავშირი კი წარმოადგენს პროვაიდერის ქსელს:

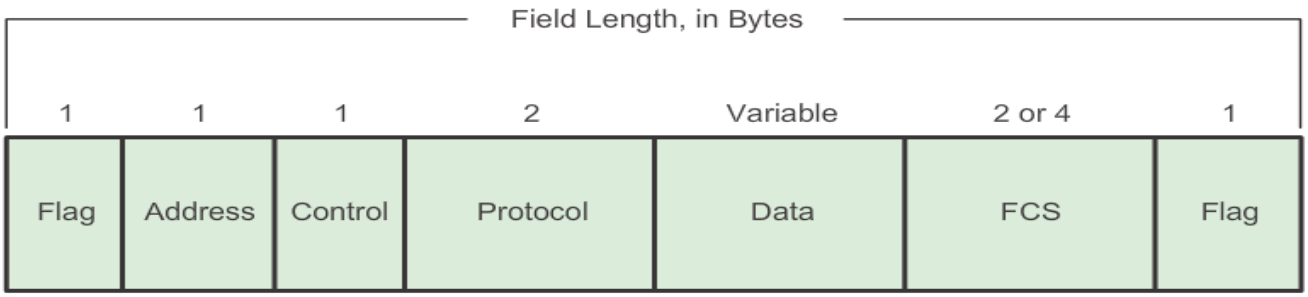
სურ.2.4. 4

Electronics Industry Association (EIA) და ITU-T წარმოადგენენ იმ ორგანიზაციებს, რომლებიც შეიმუშავებენ სტანდარტებს DCE-DTE კავშირის უზრუნველსაყოფად. სურ.2.4.5 -ზე ნაჩვენებია ამგვარი კავშირის მაგალითი, შესაბამისი კაბელით.



სურ.2.4. 5

PPP პროტოკოლის ფრეიმის ველები და მათი სიგრძე ბიტებში მოცემულია სურ.2.4.6-ზე.



სურ.2.4. 6

PPP კონფიგურაცია

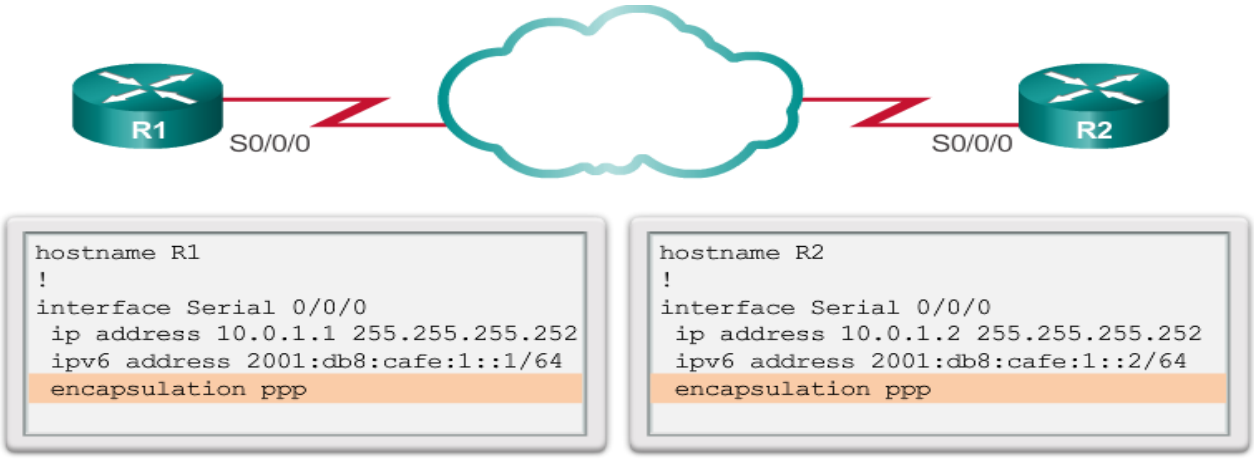
ცისკოს როუტერებში serial ინტერფეისებზე ნაგულისხმევი ინკაფსულაციის მეთოდი გახლავთ HDLC. serial 0/0/0 ინტერფეისზე PPP ინკაფსულაციის ნებართვის მაგალითია:

R3# configure terminal

R3(config)# interface serial 0/0/0

R3(config-if)# encapsulation ppp

როგორც სურ.2.4.7-ზე ჩანს, R1 და R2 როუტერები კონფიგურირებულია როგორც IPv4 ასევე IPv6 მისამართებით Serial ინტერფეისზე და გააქტიურებულია PPP:



სურ.2.4. 7

Point-to-Point პროგრამული შეკუმშვა სერიალ ინტერფეისებზე შესაძლებელია დაკონფიგურირდეს PPP ინკაფსულაციის ნებართვის შემდგომ.:

R3(config)# interface serial 0/0/0

R3(config-if)# encapsulation ppp

R3(config-if)# compress [predictor | stac]


```
hostname R1
!
interface Serial 0/0/0
 ip address 10.0.1.1 255.255.255.252
 ipv6 address 2001:db8:cafe:1::1/64
 encapsulation ppp
 compress predictor
```

```
hostname R2
!
interface Serial 0/0/0
 ip address 10.0.1.2 255.255.255.252
 ipv6 address 2001:db8:cafe:1::2/64
 encapsulation ppp
 compress predictor
```

```
Router(config if)# compress [predictor | stac]
```

Keyword	Description
predictor	(Optional) Specifies that a predictor compression algorithm will be used.
stac	(Optional) Specifies that a Stacker (LZS) compression algorithm will be used.

სურ.2.4. 8

დამატებით ხარისხობრივ ფაზას წარმოადგენს ლინკის ხარისხობრივი მონიტორინგი LQM.

ბრძანება - **PPP QUALITY "პროცენტული რიცხვი"** - ადგენს კავშირის ხარისხობრივ მაჩვენებელს, წინააღმდეგ შემთხვევაში კავშირი შეწყდება.

ქვემოთ მოცემულია R1 მარშრუტიზატორის S0/0/1 ინტერფეისზე PPP კონფიგურაციის მაგალითი, შეკუმშვითა და LQM პარამეტრის განსაზღვრით:

```
R1(config)# interface S0/0/1
R1(config-if)# ip address 10.0.1.5 255.255.255.252
R1(config-if)# ipv6 address 2001:db8:cafe:3::1/64
R1(config-if)# encapsulation ppp
R1(config-if)# compress predictor
R1(config-if)# ppp quality 90
```

PPP ინკაფსულაციის კონფიგურაციის შემოწმებისთვის გამოიყენება show interfaces serial ბრძანება.:

```
R2# show interfaces serial 0/0/0
Serial0/0/0 is up, line protocol is up
  Hardware is GT96K Serial
  Internet address is 10.0.1.2/30
  MTU 1500 bytes, BW 1544 Kbit/sec, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation PPP, LCP Open
  Open: IPCP, IPV6CP, CCP, CDPCP, loopback not set
  Keepalive set (10 sec)
  CRC checking enabled
  Last input 00:00:02, output 00:00:02, output hang never
  Last clearing of "show interface" counters 01:29:06
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output
drops: 0
  Queueing strategy: weighted fair
  Output queue: 0/1000/64/0 (size/max total/threshold/drops)
     Conversations  0/1/256 (active/max active/max total)
     Reserved Conversations 0/0 (allocated/max allocated)
     Available Bandwidth 1158 kilobits/sec
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     1944 packets input, 67803 bytes, 0 no buffer
     Received 0 broadcasts (0 IP multicasts)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0
```

```
abort
     1934 packets output, 67718 bytes, 0 underruns
     0 output errors, 0 collisions, 5 interface resets
     1 unknown protocol drops
     0 output buffer failures, 0 output buffers swapped out
     8 carrier transitions
     DCD=up  DSR=up  DTR=up  RTS=up  CTS=up

R2#
```

2.5. DHCP პროტოკოლი

მარშრუტიზატორის ძირითადი კონფიგურირება

ბრძანებათა ველის რეჟიმები

სამომხმარებლო რეჟიმი

– Router >

პრივილეგირებული რეჟიმი

– Router #

პრივილეგირებულ რეჟიმში შესვლა

– Router > enable

– Router #

პრივილეგირებული რეჟიმიდან გამოსვლა

– Router # exit (ან desible)

– Router >

გლობალური კონფიგურაციის რეჟიმი

– Router > enable

– Router # config t

– Router(config)#

ინტერფეისის კონფიგურირების რეჟიმი

– Router > enable

– Router # config t

– Router(config)#interface ტიპი ნომერი (მაგ.:interface FastEthernet 0/0)

– Router(config-if)#

მოწყობილობისთვის სახელის მინიჭება

– Router > enable

– Router # config t

– Router(config)#Hostname სახელი (მაგ.: Hostname CiscoRouter)

– CiscoRouter(config)#

პრივილეგირებული რეჟიმის პაროლით დაცვა

• არაშიფრირებული პაროლი

– Router > enable

– Router # config t

– Router(config)#enable password პაროლი

• დაშიფრული პაროლი

– Router > enable

– Router # config t

– Router(config)#enable secret პაროლი

მრძანებათა კონსოლის დაცვა

– Router > enable

– Router # config t

– Route(config)# line console 0

– Router(config)# password პაროლი

– Router(config)# login

ვირტუალური ტერმინალის პორტების დაცვა

– Router > enable

– Router # config t

– Route(config)# line vty 0 4

– Router(config)# password პაროლი

– Router(config)# login

პაროლების დაშიფრვა

– Router > enable

– Router # config t

– Route(config)# service password encryption

მიმდინარე პარამეტრების საწყის-ჩამტვირთავ პარამეტრებად შენახვა

– Router > enable

– Router# copy runn start

ინტერფეისის კონფიგურირება

ქვემოთ მოცემულია Fastethernet და Serial ინტერფეისებზე IP და ქვექსელის ნიღაბის მისამართის და ინტერფეისის გააქტიურების მაგალითი

```
Router(config)#interface fastethernet 0/0
Router(config-if)#description connection to Admin LAN
Router(config-if)#ip address 192.168.2.1 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#interface serial 0/0/0
Router(config-if)#description connection to Router2
Router(config-if)#ip address 192.168.1.125 255.255.255.0
Router(config-if)#clock rate 64000
Router(config-if)#no shutdown
```

DHCP სერვისის კონფიგურირება

ნაბიჯი I (DHCP მისამართების სივრცის შექმნა)

– Router > enable

– Router # config t

– Route(config)#ip dhcp pool სახელი

ნაბიჯი II (ქსელის და ქვექსელის ნიღაბის მისამართების მითითება)

– Router > enable

– Router # config t

– Router(config)#ip dhcp pool სახელი

– Router(dhcp-config)# network (მაგ.:) 192.168.1.0 255.255.255.0

ნაბიჯი III (IP მისამართების გამორიცხვა-დარეზერვება)

– Router > enable

– Router # config t

– Route(config)#ip dhcp excluded-address (მაგ.:192.168.1.1 192.168.1.49)

ნაბიჯი IV (DNS სერვერის დამისამართება)

– Router > enable

– Router # config t

– Router(config)#ip dhcp pool სახელი

– Router(dhcp-config)# network 192.168.1.0 255.255.255.0

– Router(dhcp-config)#dns-server (მაგ.:)192.168.1.10

ნაბიჯი V (Gateway დამისამართება)

– Router > enable

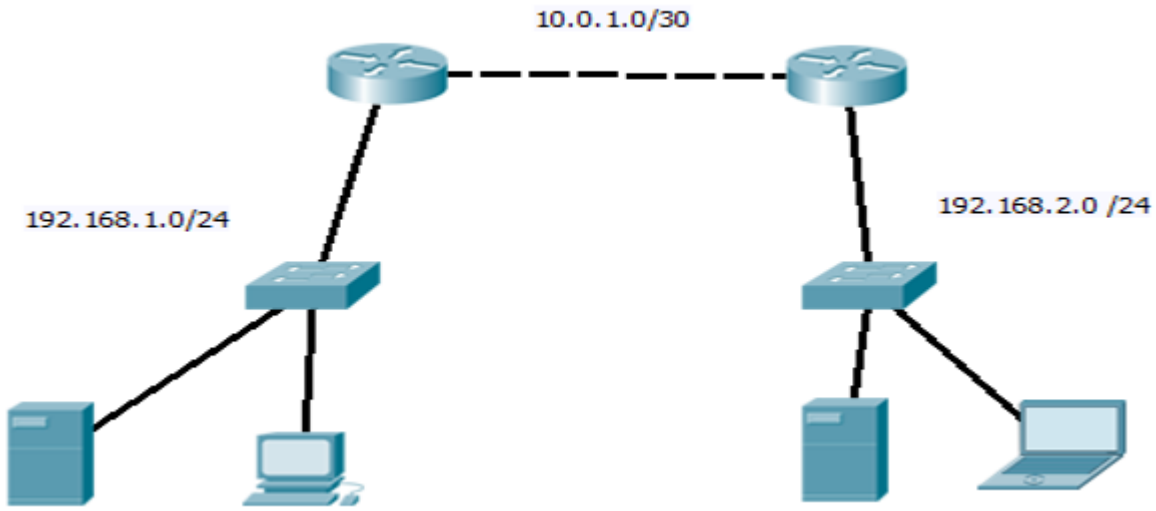
– Router # config t

– Router(config)#ip dhcp pool სახელი

– Router(dhcp-config)# network 192.168.1.0 255.255.255.0

– Router(dhcp-config)#default-router (მაგ.:) 192.168.1.1

პრაქტიკული სავარჯიშო



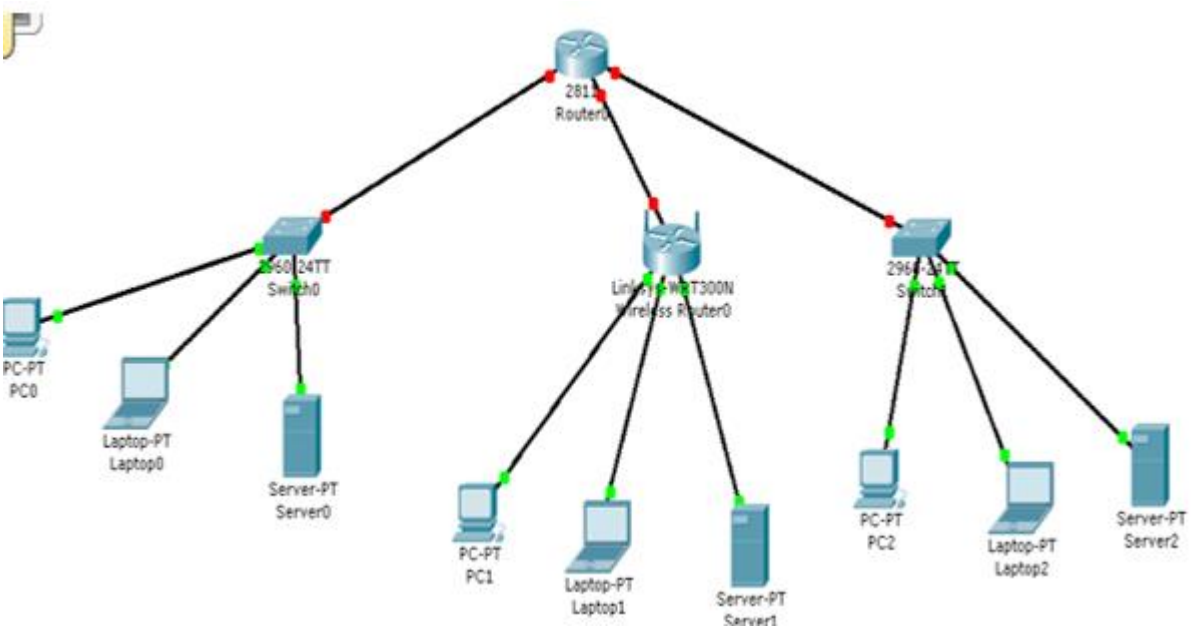
1. შექმენით ქსელის ფიზიკური და ლოგიკური მოდელი

2. როუტერების შეურჩიეთ ქსელური სახელი - თქვენი სახელი და გვარი

3. როუტერებზე გააქტიურეთ(შექმენით) ბრძანებათა ველის(Line Console) და პრივილეგირებული რეჟიმის(Enable) პაროლები - თქვენი გვარი

4. მე-2 როუტერზე გააქტიურეთ DHCP კონფიგურაცია, რომელიც დაარიგებს შესაბამისი ქსელის ჰოსტებზე IP, Gateway და DNS მისამართებს

პრაქტიკული სავარჯიშო



1. შექმენით მოცემულის შესაბამისი ქსელის მოდელი
2. მარცხენა ნაწილში განლაგებულ ქსელის მოდელში, ჩართეთ 157.207.114.78 /27 ქსელის II ქვექსელის საწყისი კვანძები (Host)
3. მე-2 ქსელში უკაბელო კავშირის მოწყობილობას შეურჩიეთ 192.168.250.0 /26 ქსელის პირველი ჰოსტის მისამართი, რომელიც თავის მხრივ DHCP პროტოკოლით დაურიგებს IP მისამართებს მასთან მიერთებულ 3 კვანძს (Host)
4. მე-3 ქსელში 136.25.139.110 /19 მისამართის მქონე სერვერი DHCP უტილიტას მეშვეობით ავტომატურად მიანიჭებს შესაბამის ქსელის მისამართებს იმავე ქსელში ჩართულ 2 კვანძს (Host)
5. მოცემული ქსელები დააკავშირეთ ერთმანეთთან მარშრუტიზატორის (Router) მეშვეობით (თუ საჭირო იქნება ამისთვის მარშრუტიზატორს დაუმატეთ ეზერნეტ პორტი)
6. ყველა კვანძს Gateway მისამართად გაუწერეთ შესაბამისი ქსელის ბოლო კვანძის(Host) მისამართი, შესაბამისი მისამართი გაუწერეთ მარშრუტიზატორის სათანადო ინტერფეისებს...

7. დააკონფიგურირეთ მარშრუტიზატორი, კერძოდ შეურჩიეთ სახელი: თქვენი სახელი
8. პრივილეგირებული რეჟიმი დაიცავით პაროლით (პაროლი- თქვენი სახელი_TEST1)
9. ბრძანებათა ველის ინტერფეისი დაიცავით პაროლით (პაროლი-თქვენი სახელი_TEST1)
10. მიმდინარე კონფიგურირების პარამეტრები შეინახეთ როგორც საწყისი კონფიგურირების პარამეტრი

3. ქსელური აპარატურის და ტექნოლოგიების უსაფრთხოება

3.1. ქსელის უსაფრთხოებასთან დაკავშირებული საფრთხეები.

ჩვენი დროის ერთ-ერთ დამახასიათებელ თავისებურებად შეიძლება ჩაითვალოს ის, რომ ინფორმაცია ხდება უფრო მნიშვნელოვანი რესურსი, ვიდრე მატერიალური ან ენერგეტიკული რესურსები. რესურსები ზოგადად განმარტებულია, როგორც საზოგადოების მფლობელობაში არსებული ეკონომიკური პოტენციალის ელემენტები, რომლებიც შესაძლებელია გამოყენებულ იქნას სამეურნეო მოღვაწეობაში კონკრეტული მიზნების მისაღწევად. თანამედროვე საზოგადოებისათვის ჩვეულებრივი გახდა ისეთი კატეგორიები, როგორიცაა მატერიალური, ფინანსური, შრომითი, ბუნებრივი რესურსები, მათი დანიშნულება ყველასათვის ნათელია. ასევე ჩვეულებრივი გახდა ცნება ”ინფორმაციული რესურსი”, რომელიც ლიტერატურაში განმარტებულია ასე: ინფორმაციული რესურსები წარმოადგენს ცალკეულ დოკუმენტებს და დოკუმენტების მასივებს, ინფორმაციულ სისტემაში (ბიბლიოთეკებში, არქივებში, მონაცემთა საცავებში, მონაცემთა ბანკებში და სხვა) განთავსებულ დოკუმენტებსა და დოკუმენტების მასივებს. ინფორმაციული რესურსები წარმოადგენენ რომელიმე ორგანოს ან ორგანიზაციის საკუთრებას, საჭიროებენ აღრიცხვასა და დაცვას, ვინაიდან ინფორმაციის გამოყენება შესაძლებელია არა მარტო საქონლის წარმოებისა და მომსახურებისათვის, არამედ შეიძლება მისი გადაქცევა ფულად სახსრებად, მისი ვინმეზე მიყიდვის ან, რაც ყველაზე სავალალოა, განადგურებით.

მეწარმისათვის საკუთარი ინფორმაცია წარმოადგენს მნიშვნელოვან ღირებულებას, ვინაიდან ხშირად ინფორმაციის მიღება (წარმოქმნა) შრომატევადი და ძვირადღირებული პროცესია.

ინფორმაციულ რესურსებს განსაკუთრებული როლი ენიჭება საბაზრო ეკონომიკის პირობებში. საბაზრო ეკონომიკის უმთავრესი ფაქტორია კონკურენცია. კონკურენციის პირობებში კი უპირატეს მდგომარეობაშია ის, ვინც ინფორმაციას ფლობს. კონკურენტულ ბრძოლაში ფართოდაა გავრცელებული კონფიდენციალური ინფორმაციის მოპოვების სხვადასხვა კანონიერი თუ უკანონო ხერხები. ამ პირობებში უკანონო მოპოვებისაგან ინფორმაციის დაცვას ენიჭება უაღრესად დიდი მნიშვნელობა.

ინფორმაციული უსაფრთხოება - საკმაოდ ტევადი და მრავალმხრივი პრობლემაა, რომელიც მოიცავს არა მარტო ინფორმაციის დაცვის აუცილებლობის განმარტებას, არამედ როგორ მოხდეს დაცვა, რისგან მოხდეს დაცვა, როდის მოხდეს დაცვა, რით მოხდეს დაცვა და როგორი უნდა იყოს ეს დაცვა.

ინფორმაციული უსაფრთხოების პრობლემის არსი. ინფორმაციული უსაფრთხოების ცნება

ტერმინი ”ინფორმაცია” სხვადასხვა მეცნიერებაში სხვადასხვაგვარადაა განმარტებული. მაგალითად, ფილოსოფიაში ინფორმაცია განმარტებულია, როგორც მატერიალური ობიექტებისა და პროცესების თვისება, შეინახონ ან წარმოშვან გარკვეული მდგომარეობები, რომლებიც სხვადასხვა საგნობრივ-ენერგეტიკული ფორმით შეიძლება გადაეცეს ერთი ობიქტიდან მეორეს. კიბერნეტიკაში განსაზღვრულია ინფორმაცია, როგორც განუსაზღვრელობის აღმოფხვრის ზომა. ჩვენ მომავალში, ამ სასწავლო კურსის ფარგლებში ინფორმაციას ვუწოდებთ ყველაფერ იმას, რაც შეიძლება აისახოს სასრული ალფაბეტის სიმბოლოების მიერ (მაგალითად ორობითი). მართლაც თანამედროვე გამოთვლითი ტექნიკის ბაზისური არქიტექტურიდან გამომდინარე, ყველაფერი რისი გადამუშავებაც ხდება ამ ტექნიკით, ორობით ფორმაშია წარმოდგენილი.

ინფორმაციული ტექნოლოგიების განვითარებამ და მისმა ადამიანის საქმიანობის ყველა სფეროში შეღწევამ გამოიწვია ინფორმაციული უსაფრთხოების პრობლემის აქტუალობა. ამასთან ყოველწლიურად ეს პრობლემა სულ უფრო რთულდება.

ინფორმაციის დამუშავების ტექნოლოგიები მუდმივად განიცდის სრულყოფას, მასთან ერთად სულ უფრო იცვლება და იხვეწება ინფორმაციული უსაფრთხოების პრაქტიკული მეთოდები. ინფორმაციის დაცვის უნივერსალური მეთოდები, როგორც ცნობილია არ არსებობს, რეალური სისტემისათვის უსაფრთხოების მექანიზმების აგება მეტწილად დამოკიდებულია სისტემის ინდივიდუალურ თავისებურებებზე, რომლის აღწერაც ძნელად ექვემდებარება ფორმალიზაციას. ამიტომ ხშირად ინფორმაციულ უსაფრთხოებას განიხილავენ, როგორც არაფორმალური რეკომენდაციების ერთობლიობას, ამა თუ იმ ტიპის ინფორმაციის დაცვის სისტემის ასაგებად. დაცვის სისტემის აგების პრაქტიკულ ნაბიჯებსა და ხერხებს საფუძვლად უდევს ისეთი ზოგადი კანონზომიერებები, რომლებიც საერთოა და

არაა დამოკიდებული კონკრეტულ ტექნიკურ რეალიზაციაზე. ასეთ ზოგად კანონზომიერებებსა და პრინციპებს შეისწავლის მეცნიერება, რომელსაც ინფორმაციული უსაფრთხოება ეწოდება.

ტერმინები "უსაფრთხოება" და "დაცვა" ხშირად ერევათ ერთმანეთში. ხშირად სასარგებლოა საზღვრის გავლება, ერთის მხრივ, იმ საერთო პრობლემებს შორის, რომლებიც დაკავშირებულია გარანტიებთან, რომ არ მოხდეს ინფორმაციის წაკითხვა და მოდიფიცირება არაავტორიზებული პირების მიერ და მეორე მხრივ, იმ სპეციფიკურ მექანიზმებს შორის, რომლებიც გამოიყენება უსაფრთხოების უზრუნველსაყოფად. ტერმინს "უსაფრთხოება" - გამოვიყენებთ საერთო პრობლემის აღსანიშნავად, ხოლო ტერმინს "დაცვა" - იმ სპეციფიკური მექანიზმების აღსაწერად, რომლებიც გამოიყენება კომპიუტერულ სისტემებში ინფორმაციული უსაფრთხოების უზრუნველსაყოფად. თუმცა, ამ ორ ტერმინს შორის საზღვარი მკვეთრი არ არის. სასწავლო კურსში ჯერ გავეცნობით უსაფრთხოების საკითხებს, რათა გავიგოთ პრობლემის ბუნება, ხოლო შემდეგ განვიხილავთ დაცვის მექანიზმებს და უსაფრთხოების უზრუნველყოფის მოდელებს.

უსაფრთხოების პრობლემა მრავალმხრივია. მისი შედარებით მნიშვნელოვანი ასპექტებია საფრთხეები, ბოროტმოქმედთა ბუნება და მონაცემთა შემთხვევითი დაკარგვა.

ჩვენი განხილვის ძირითად საგანს წარმოადგენს ინფორმაციული სისტემები. ინფორმაციული სისტემების ქვეშ ვგულისხმობთ შემდეგი ობიექტების ერთობლიობას:

1. გამოთვლითი ტექნიკა;
2. პროგრამული უზრუნველყოფა;
3. კავშირის არხები;
4. მონაცემთა საცავი;
5. პერსონალი და მომხმარებლები.

ინფორმაციულ სისტემას, უსაფრთხოების თვალსაზრისით, გადასაწყვეტი აქვს შემდეგი ძირითადი ამოცანები:

- ინფორმაციის კონფიდენციალობის უზრუნველყოფა;
- ინფორმაციის მთლიანობის უზრუნველყოფა;
- ინფორმაციის სარწმუნოობის უზრუნველყოფა;

- ინფორმაციისთან ხელმისაწვდომობის (მიმართვის, წვდომის) უზრუნველყოფა;
- ელექტრონული სახით წარმოდგენილი ინფორმაციის იურიდიული მნიშვნელობის უზრუნველყოფა;
- კლიენტის მოქმედებების კონფიდენციალობის უზრუნველყოფა.

ინფორმაციის კონფიდენციალობა მდგომარეობს იმაში, რომ საიდუმლო მონაცემები საიდუმლოდ უნდა დარჩეს. კერძოდ, თუ რაიმე მონაცემების მფლობელმა გადაწყვიტა, რომ ეს მონაცემები ცნობილი იყოს მხოლოდ გარკვეულ პირთათვის, მაშინ სისტემაში გარანტირებული უნდა იყოს, რომ ამ ინფორმაციის მიღებას სხვა პირები ვერ შეძლებენ. მონაცემების მფლობელს უნდა შეეძლოს ამ მონაცემებით სარგებლობის უფლების მქონე პირების სიის მითითება, ხოლო სისტემა უნდა უზრუნველყოფდეს ამ მოთხოვნებს.

მონაცემების მთლიანობაში იგულისხმება ინფორმაციის ან პროგრამული უზრუნველყოფის თვისება, შეინარჩუნოს თავისი სტრუქტურა და შინაარსი გადაცემის და შენახვის პროცესში ანუ არაავტორიზებულ მომხმარებლებს არ უნდა ჰქონდეთ მონაცემების მოდიფიცირების შესაძლებლობა მისი მფლობელის ნებართვის გარეშე. მონაცემთა მოდიფიცირება, ამ კონკრეტულ შემთხვევაში ნიშნავს არა მარტო მონაცემთა შეცვლას, არამედ მათ ამოღებას ან ყალბი მონაცემების დამატებას. თუ სისტემას არ შეუძლია იმის გარანტირება, რომ მასში შენახული მონაცემები უცვლელი დარჩება, ვიდრე მფლობელი არ გადაწყვეტს მათ შეცვლას, ასეთი სისტემა უსარგებლოა.

ინფორმაციის სარწმუნოობა მას მკაცრად მიაკუთვნებს ობიექტს, რომელიც მის წყაროს წარმოადგენს, ან იმ ობიექტს, რომლისგანაც ეს ინფორმაციაა მიღებული.

სისტემის ხელმისაწვდომობა ნიშნავს ინფორმაციასთან ოპერატიულად მიმართვის შესაძლებლობას ანუ ინფორმაციული რესურსის უნარს იყოს მისაწვდომი საბოლოო მომხმარებლისათვის მისი მოთხოვნის შესაბამისად. არავის არ უნდა შეეძლოს სისტემის მწყობრიდან გამოყვანა. მომსახურებაზე უარის თქმის ტიპის შეტევები სულ უფრო გავრცელებული ხდება. მაგალითად, თუ კომპიუტერი წარმოადგენს ინტერნეტის სერვერს, მაშინ იგი შეიძლება გადაიტვირთოს შემოსული მოთხოვნების მძლავრი ნაკადით. ამ დროს მისი მთელი პროცესორული სიმძლავრე და შესაბამისად დრო დასჭირდება შემოსული მოთხოვნების შესწავლას. მაგალითად თუ ვებ გვერდის წაკითხვის მოთხოვნის

დამუშავებას სჭირდება 100 მკწმ, მაშინ ნებისმიერი მომხმარებელი, რომელსაც წამში შეუძლია 10 000 შეკითხვის გამოგზავნა, შეძლებს სერვერის ლიკვიდირებას. ”მომსახურებაზე უარის თქმის” ტიპის შეტევების მოგერიება საკმაოდ რთულ ამოცანას წარმოადგენს.

ინფორმაციის იურიდიული მნიშვნელობა ნიშნავს, რომ დოკუმენტს იურიდიული ძალა გააჩნია. ამ მიზნით სუბიექტები, რომლებისთვისაც მნიშვნელოვანია გადაცემული გზავნილის იურიდიული მნიშვნელობა, თანხმდებიან ინფორმაციის იმ განსაკუთრებული ატრიბუტების საყოველთაო აღიარებაზე, რომლებიც გამოხატავენ მის იურიდიულ მნიშვნელობას. გზავნილების იურიდიული მნიშვნელობა განსაკუთრებით აქტუალურია ელექტრონული გადახდის სისტემებში, სადაც ხდება ფულის ელექტრონული გადარიცხვის ოპერაციები. დოკუმენტების იურიდიული მნიშვნელობის განმსაზღვრავი ატრიბუტები ცალსახად უნდა ადასტურებდნენ, რომ დოკუმენტი კონკრეტული პირის მიერაა გამოგზავნილი.

ოპერაციების კონფიდენციალობა ნიშნავს, რომ მომხმარებელს საშუალება აქვს აწარმოოს ოპერაციები ისე, რომ არავის შეეძლოს მისი თვალთვალი. მსგავსი მოთხოვნის აქტუალობა ცხადი გახდა ელექტრონული ფულის წარმოშობასთან ერთად. ელექტრონული ანგარიშსწორების სისტემასთან მიმართვისას მომხმარებელი აწვდის მას გარკვეულ საიდენტიფიკაციო ინფორმაციას. ამ სისტემის ფართო გავრცელებასთან ერთად შესაძლოა გაჩნდეს ანგარიშსწორების ოპერაციების კონტროლირების და მომხმარებლებზე ტოტალური თვალთვალის საშიშროება სახელმწიფო სტრუქტურების ან სხვა დაინტერესებული პირების მხრიდან.

უსაფრთხოების უზრუნველყოფის ძირითადი მეთოდებია:

1. სერვისული მეთოდები
   1.1. იდენტიფიკაცია და აუთენტიკაცია;
   1.2. წვდომების (მიმართვების) დანაწილება;
   1.3. პროტოკოლირება და აუდიტი;
   1.4. კრიპტოგრაფია.
2. საინჟინრო-ტექნიკური მეთოდები

2.1. კავშირის არხების დაცვა;

2.2. კავშირის არხების ეკრანირება.

3. საკანონმდებლო და ორგანიზაციული მეთოდები

3.1. პასუხისმგებლობა;

3.2. სახელმწიფო საიდუმლოებებთან მუშაობა;

3.3. საავტორო უფლებების დაცვა;

3.4. ლიცენზირება, სერთიფიკაცია.

4. ორგანიზაციული მეთოდები

4.1. პერსონალის მართვა;

4.2. შიდა კონტროლი.

5. თეორიული მეთოდები

5.1. პროცესების ფორმალიზაცია;

5.2. ადეკვატურობა და კორექტულობა

ინფორმაციული სისტემების საფრთხეები

ინფორმაციული სისტემის უსაფრთხოების განსაზღვრისას გამოვიყენეთ საფრთხის ცნება. მიღებულია საფრთხე ეწოდოს პოტენციურად შესაძლო ისეთ მოქმედებებს, მდგომარეობებს, პროცესებს ან მოვლენებს, რომლებმაც შეიძლება ზიანი მიაყენოს ვინმეს ინტერესებს. შესაბამისად, ინფორმაციული სისტემის საფრთხე - ესაა ისეთი ზემოქმედებების რეალიზაცია ინფორმაციულ სისტემაში დასამუშავებელ ინფორმაციაზე, რომელიც იწვევს კონფიდენციალობის, მთლიანობის ან წვდომის დარღვევას, აგრეთვე ინფორმაციული სისტემის კომპონენტებზე ისეთ ზემოქმედებას, რომელმაც შეიძლება გამოიწვიოს მათი დაკარგვა, განადგურება ან ფუნქციონირების შეფერხება.

საფრთხეების კლასიფიკაცია შეიძლება მოვახდინოთ სხვადასხვა ნიშანთვისებებით. მათ შორის ყველაზე გავრცელებულია:

1. წარმოშობის მიხედვით გამოყოფენ **ბუნებრივ** და **ხელოვნურ** საფრთხეებს. ბუნებრივად მიიჩნევა საფრთხე, რომელიც წარმოიშობა ობიექტური ფიზიკური პროცესების ან სტიქიური ბუნებრივი მოვლენების ზემოქმედებების გამო, რომლებიც ადამიანის ნებაზე

არაა დამოკიდებული. ხელოვნური საფრთხეები წარმოიშობა ადამიანური ფაქტორის ზემოქმედების გამო. ბუნებრივი საფრთხეების მაგალითია ხანძარი, წყალდიდობა, მიწისძვრა და ა.შ.

2. განზრახულობის მიხედვით განიხილავენ **შემთხვევით** და **წინასწარგანზრახულ** საფრთხეებს. შემთხვევითი საფრთხეები გამოწვეულია პერსონალის გულგრილობით ან არაწინასწარგანსაზღვრული შეცდომებით. წინასწარგანზრახული საფრთხეები ჩვეულებრივ წარმოიშობა ბოროტმოქმედის წინასწარ განსაზღვრული მოქმედებებით. შემთხვევითი საფრთხის მაგალითია არაწინასწარგანსაზღვრული არასწორი ინფორმაციის შეტანა, მოწყობილობის შეუგნებლად დაზიანება. წინასწარგან-საზღვრული საფრთხის მაგალითია - ბოროტმოქმედის მიღებული წესის დარღვევით შეღწევა დაცულ ტერიტორიაზე.
3. წყაროს მიხედვით გამოყოფენ:
   - საფრთხეს, რომლის წყაროს წარმოადგენს **ბუნებრივი** გარემო. მაგალითად ხანძარი, წყალდიდობა და სხვა სტიქიური უბედურებები;
   - საფრთხეს, რომლის წყაროს წარმოადგენს **ადამიანი.** მაგალითად აგენტების შეგზავნა პერსონალის რიგებში კონკურენტული ფირმის მიერ;
   - საფრთხეს, რომლის წყაროს წარმოადგენს **სანქცირებული პროგრამულ-აპარატურული** საშუალებები. მაგალითად სისტემური უტილიტების არაკომპე-ტენტური გამოყენება;
   - საფრთხეს, რომლის წყაროს წარმოადგენს **არასანქცირებული პროგრამულ-აპარატურული** საშუალებები. მაგალითად სისტემაში კეილოგერების ჩანერგვა.
4. წყაროს განლაგების მიხედვით გამოყოფენ:
   - საფრთხეებს, რომლის წყაროები განლაგებულია **კონტროლირებული ზონის გარეთ**. მაგალითად ელექტრომაგნიტური გამოსხივების ან არხით გადაცემული ინფორმა-ციის მიყურადება. დისტანციური ფოტო ან ვიდეო გადაღებები. აკუსტიკური ინფორმაციის მიყურადება მიმართული მიკროფონებით;

- საფრთხეებს, რომლის წყაროები განლაგებულია **კონტროლირებული ზონის ფარგლებში**. მაგალითად, კონფიდენციალური ინფორმაციის მატარებლების მოპარვა ან მიყურადების მოწყობილობების არასანქცირებული გამოყენება.

5. ზემოქმედების მიხედვით განიხილავენ **პასიურ** და **აქტიურ საფრთხეებს.** პასიური საფრთხის დროს არ ხდება ინფორმაციული სისტემის სტრუქტურისა და შემადგენლობის რაიმე ცვლილება. აქტიური საფრთხეების რეალიზებით კი პირიქით, ხდება სისტემის სტრუქტურის დარღვევა.
6. ინფორმაციული სისტემის რესურსებთან წვდომის მიხედვით გამოყოფენ:
   - საფრთხეებს, რომლებიც იყენებენ **სტანდარტულ წვდომას**. მაგალითად, პაროლის არასანქცირებული მიღება მოსყიდვის, შანტაჟის, მუქარის ან ფიზიკური ძალადობის გზით კანონიერი მფლობელისაგან.
   - **არასტანდარტული წვდომის** გამოყენების საფრთხეებს. მაგალითად, დაცვის საშუალებების არადეკლარირებული საშუალებით სარგებლობა.

სხვადასხვა ინფორმაციული სისტემისათვის საფრთხეების ჩამოთვლა წარმოადგენს საშიშროებების ანალიზის უმნიშვნელოვანეს ეტაპს, მაგალითად სისტემის უსაფრთხოების აუდიტის დროს და ქმნის საფუძველს რისკების ანალიზის ჩასატარებლად.

ბოროტმოქმედების ტიპები

შეიძლება ზოგიერთმა ჩათვალოს, რომ ადამიანთა უმრავლესობა იცავს კანონს, ამიტომ აზრი არ აქვს უსაფრთხოებაზე ზრუნვას. სამწუხაროდ ყველა ადამიანი არ არის პატიოსანი და ცდილობს სხვებს მიაყენოს უსიამოვნება (მაგალითად, პირადი კომერციული სარგებლის მიღების მიზნით). ასეთ ადამიანებს ვუწოდოთ ბოროტმოქმედი ან მტერი. ბოროტმოქ-მედები იყოფიან ორ ტიპად - პასიური, რომლებიც ცდილობენ წაიკითხონ აკრძალული ინფორმაცია და აქტიურები, რომლებიც უკანონოდ ცდილობენ მონაცემთა შეცვლას. ბოროტმოქმედებისგან თავდაცვის სისტემის შემუშავებისას საჭიროა კარგად იქნას გააზ-რებული მათი მოქმედების მეთოდები. ბოროტმოქმედთა ყველაზე მეტად გავრცელებული კატეგორიებია:

1. შემთხვევითი ცნობისმოყვარე მომხმარებლები, რომლებიც არ იყენებენ სპეციალურ ტექნიკურ საშუალებებს. მრავალ ადამიანს აქვს საერთო ფაილების სერვერთან შეერთებული კომპიუტერი და თუ არ ექნა დაყენებული სპეციალური დაცვა, უბრალო ცნობისმოყვარეობის გამო მრავალი ადამიანი დაიწყებს სხვისი ელექტრონული ფოსტის და ფაილების წაკითხვას. მაგალითად, UNIX-ის სისტემაში ახლად შექმნილი ფაილები ფარულად ხელმისაწვდომია ყველა მსურველისთვის.
2. ორგანიზაციების ჯაშუშობით დაკავებული წევრები, სტუდენტები, სისტემური პროგრამისტები, ოპერატორები და სხვა ტექნიკური პერსონალი ხშირად პირად გამოწვევად მიიჩნევენ ლოკალური კომპიუტერული სისტემის უსაფრთხოების სისტემის გატეხვას. როგორც წესი, მათ აქვთ მაღალი კვალიფიკაცია და მზად არიან დახარჯონ დიდი დრო დასახული მიზნის მისაღწევად.
3. ის, ვინც ცდილობს პირად გამდიდრებას. ბანკში მომუშავე ზოგიერთი პროგრამისტი ცდილობდა მოეპარა ბანკის ფული. გამოყენებული სქემები იცვლებოდა დაპროგრამებაში თანხების დამრგვალების ხერხების ცვლილების მიხედვით, შანტაჟის მიზნით, (”გადამიხადეთ, თორემ გავანადგურებ ბანკის მთელ ინფორმაციას”).
4. კომერციული და სამხედრო ჯაშუშობით დაკავებული პირები. ჯაშუშობა წარმოადგენს სერიოზულ და კარგად დაფინანსებულ მეთოდს კონკურენტის ან სხვა სახელმწიფოს მხრიდან, პროგრამის, კომერციული საიდუმლოს, მნიშვნელოვანი იდეის და ტექნოლოგიის, მიკროსქემების, ბიზნეს გეგმების მოპარვის მიზნით. ხშირად ასეთ დროს მიმართავენ კავშირგაბმულობის სისტემასთან მიერთების ან ანტენების დაყენების მეთოდს. ანტენები მიმართულია კომპიუტერისკენ, მისი ელექტრომაგნიტური გამოსხივების ”დაჭერის” მიზნით.

ცხადია, რომ სამხედრო საიდუმლოების მოპარვისგან თავდაცვა განსხვავდება სტუდენტებისგან თავის დასაცავად მიღებული ზომებისაგან, რომლებიც ცდილობენ საიტებზე განათავსონ სახალისო შეტყობინებები. საიდუმლოებების დაცვის ღონისძიებები დამოკიდებულია სავარაუდო მოწინააღმდეგეზე. კიდევ ერთ საფრთხეს წარმოადგენს ბოლო დროს გაჩენილი ვირუსი. იგი ზოგადად წარმოადგენს პროგრამას, რომელიც ახდენს საკუთარი თავის რეპლიკაციას (როგორც წესი) და გარკვეულ ზიანს აყენებს სისტემას.

გარკვეული გაგებით ვირუსის ავტორიც ითვლება ბოროტმოქმედად, რომელსაც აქვს მაღალი კვალიფიკაცია. მთავარი განსხვავება ჩვეულებრივ ბოროტმოქმედსა და ვირუსს შორის მდგომარეობს იმაში, რომ პირველი მათგანი ცდილობს სისტემის გატეხვას მისთვის ზიანის მიყენების მიზნით, მაშინ როცა ვირუსი არის პროგრამა, რომელიც შედგენილია ასეთი ადამიანის მიერ და გაშვებულია სისტემაში ზიანის (ზარალის) მიყენების მიზნით. ბოროტმოქმედები ცდილობენ გარკვეული სისტემის გატეხვას (რომელიმე ორგანიზაციის ან ბანკის) რათა მოიპარონ ან გაანადგურონ გარკვეული მონაცემები, მაშინ როცა ვირუსი ჩვეულებრივ მოქმედებს მიმართულების გარეშე. ამრიგად, ბოროტმოქმედი შეიძლება შევადაროთ დაქირავებულ მკვლელს (”ქილერს”), ხოლო ვირუსი ტერორისტს, რომელიც ცდილობს დიდი რაოდენობით ადამიანთა დახოცვას და არა კონკრეტულად ვინმეს მოკვლას.

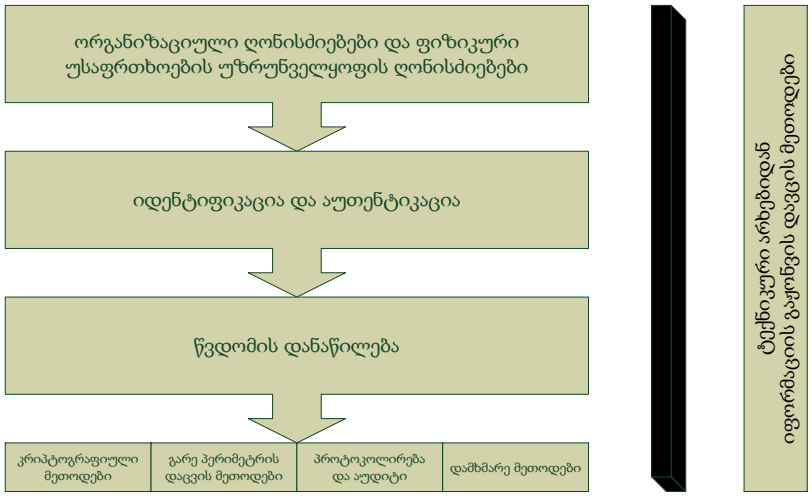
დაცვის სისტემების მოდელები

ინფორმაციულ სისტემებში კონფიდენციალობის დარღვევის საფრთხეებისაგან დაცვის სისტემის ასაგებად გამოიყენება კომპლექსური მეთოდები. ტრადიციული ეშელონირებული დაცვის სქემა მოტანილია სურ. 3.1.-ზე.

პირველადი დაცვა ხორციელდება ორგანიზაციული ღონისძიებებით და ინფორმაციულ სისტემასთან ფიზიკური წვდომის კონტროლის მექანიზმებით. შემდგომში, ლოგიკური წვდომის კონტროლის ეტაპზე, დაცვა ხორციელდება ქსელური უსაფრთხოების სხვადასხვა სერვისის გამოყენებით. ყველა ეტაპზე პარალელურად უნდა იყოს განხორციელებული (განლაგებული) ინფორმაციის დაცვის საინჟინრო-ტექნიკური კომპლექსის საშუალებები, რომლებიც აღკვეთავენ ტექნიკური არხებიდან ინფორმაციის გაჟონვას.

ორგანიზაციული და ფიზიკური უსაფრთხოების უზრუნველყოფის ზოგადი ღონისძიებებია:

- ინფორმაციულ სისტემასთან წვდომის კონტროლისა და ფიზიკური ურთიერთობის სისტემის განლაგება;
- დაცვისა და ფიზიკური უსაფრთხოების სამსახურის ორგანიზება;



სურ. 3.1. ინფორმაციის კონფიდენციალობის დარღვევის საფრთხისაგან დაცვის სისტემის სტრუქტურა

- რეგლამენტების, სამსახურებრივი ინსტრუქციებისა და სხვა მარეგულირებელი დოკუმენტების დამუშავება და დანერგვა;
- კონფიდენციალური ინფორმაციის მატარებელ ობიექტებთან მუშაობის წესები და რეგლამენტები.

ინფორმაციული სისტემის ლოგიკის გათვალიწინების გარეშეც ეს ღონისძიებები წარმოადგენენ უაღრესად ეფექტურ მექანიზმებს ნებისმიერი რეალური სისტემის უსაფრთხოების უზრუნველსაყოფად.

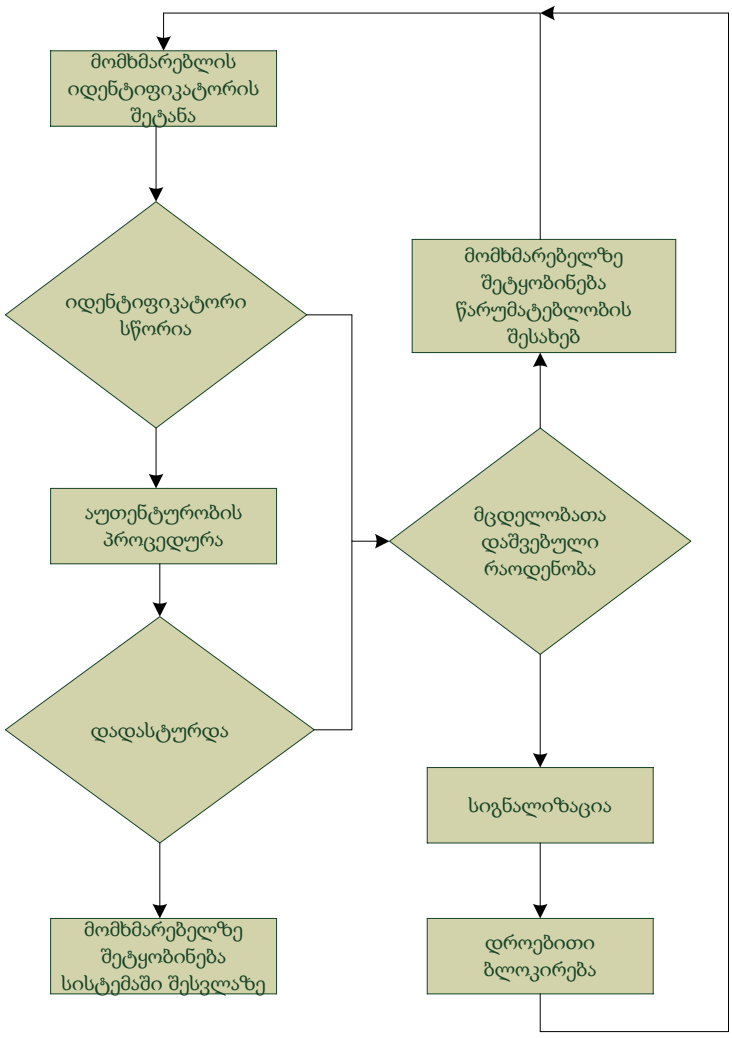
იდენტიფიკაცია და აუთენტიკაცია

იდენტიფიკაცია - ესაა სუბიექტებზე უნიკალური იდენტიფიკატორების მიკუთვნება (მინიჭება) და მათი შედარება შესაძლო იდენტიფიკატორების ვარიანტებთან. აუთენტიკაცია - ესაა შემოწმება სუბიექტის მიერ წარმოდგენილი იდენტიფიკატორის კუთვნილებისა მის წარმომდგენ სუბიექტთან და მისი უტყუარობის დადასტურება (სურ. 3.2.). ამგვარად, იდენტიფიკაციის ამოცანაა - გასცეს პასუხი კითხვას ”ვინაა ეს?”, ხოლო აუთენტიკაციის - ”მართლა ისაა”.

დღეისათვის გამოყენებული აუთენტიკაციის მეთოდების სიმრავლე შეიძლება დაიყოს 4 მსხვილ ჯგუფად:

1. მეთოდები, რომლებიც დამოკიდებულია რაიმე საიდუმლო ინფორმაციის ფლობასთან (ცოდნასთან). ასეთი მეთოდების კლასიკურ მეთოდს წარმოადგენს პაროლური სისტემა. ამ დროს მომხმარებლის აუთენტურობის დასადგენად გამოიყენება პაროლი - სიმბოლოების რაიმე მიმდევრობა. ეს მეთოდი აუთენტურობის დადგენის ყველაზე გავრცელებული მეთოდია;
2. მეთოდები, დაფუძნებული უნიკალური საგნის გამოყენებაზე. ასეთ საგნებად შეიძლება გამოყენებული იქნას სმარტ-ბარათები, ელექტრონული გასაღებები და სხვა;
3. მეთოდები დაფუძნებული ადამიანის ბიომეტრიულ მონაცემებზე. პრაქტიკაში ხშირად გამოიყენება ერთი ან რამდენიმე ქვემოთ ჩამოთვლილი ბიომეტრიული მახასიათებლები:
   - თითის ანაბეჭდები;
   - თვალის გუგის ან ბადურის ნახატები;
   - ხელის თბური სურათი;
   - სახის ფოტო ან თბური სურათი;
   - ხელმოწერა (კალიგრაფია);
   - ხმა.

   ყველაზე გავრცელებულია თითის ანაბეჭდების, თვალის გუგის ან ბადურის სურათების სკანერები.
4. მეთოდები, მომხმარებელთან ასოცირებული ინფორმაციის გათვალისწინებით. ასეთ ინფორმაციად შეიძლება გამოდგეს მომხმარებლის GPS კოორდინატები. რა თქმა უნდა, მარტო ეს პარამეტრი შეიძლება საკმარისი არ აღმოჩნდეს, მაგრამ სხვა პარამეტრებთან ერთად შეიძლება კარგი მექანიზმი აიგოს.



სურ. 3.2. იდენტიფიკაციისა და აუთენტიკაციის ზოგადი სქემა

ფართოდაა გავრცელებული ზემოთ ჩამოთვლილი მექანიზმებიდან რამდენიმეს ერთობლივად გამოყენების პრაქტიკა, ასეთ შემთხვევაში ლაპარაკია მრავალფაქტორულ აუთენტიკაციაზე.

პაროლური სისტემების აუთენტურობის თავისებურებები

პაროლური სისტემის ყველაზე მეტად გავრცელება გამოწვეულია:

- რეალიზაციის შედარებით სიმარტივე. ასეთი მექანიზმის რეალიზაცია, როგორც წესი, არ საჭიროებს დამატებითი მოწყობილობების გამოყენებას.

- ტრადიციულობა. ასეთი დაცვის მექანიზმს მიჩვეულია ინფორმაციული სისტემის მომხმარებლების უმრავლესობა და არ იწვევს რაიმე ფსიქოლოგიურ გაღიზიანებას, თვალის გუგის სკანერებისგან განსხვავებით.

აღსანიშნავია პაროლიანი სისტემების პარადოქსი, რაც ართულებს მის ეფექტურ რეალიზაციას და გამოყენებას: სანდო პაროლები რთულად გამოსაყენებელია ადამიანისთვის. მართლაც სანდოობა მიიღწევა პაროლის გართულებით. მაგრამ რაც უფრო რთულია პაროლი, მით უფრო რთულია მისი დამახსოვრება. ამიტომ მომხმარებელი ხშირად იძულებულია ჩაიწეროს ეს რთული პაროლი, რაც იწვევს დამატებით წყაროს (შესაძლებლობას) პაროლის დისკრედიტაციისათვის.

პაროლურ სისტემებში ბოროტმოქმედის მიერ პაროლის ხელში ჩაგდების ძირითადი საფრთხეებია:

1. ადამიანური სისუსტეების ფაქტორის გამოყენება. მაგალითად: თვალთვალი, მოსმენა, შანტაჟი, მუქარა, სხვისი პირადი ინფორმაციის გამოყენება და სხვა.
2. მორგების გზა. ამ შემთხვევაში გამოიყენება შემდეგი მეთოდები:
    - სრული გადარჩევა. ამ მეთოდით შესაძლებელია ნებისმიერი სირთულის პაროლის მიგნება, ამიტომ სანდო პაროლის გამოსაცნობად, ასეთი შეტევისათვის საჭირო დრო მნიშვნელოვნად უნდა აჭარბებდეს ბოროტმოქმედისათვის დასაშვები დროითი რესურსების შესაძლებლობას;
    - ლექსიკონით მორგება. პრაქტიკაში გამოყენებული პაროლების დიდი უმრავლესობა წარმოადგენენ აზრობრივ სიტყვებსა და გამოსახულებებს. არსებობენ გავრცელებული პაროლების ლექსიკონები, რომლის გამოყენებითაც შეიძლება საჭირო აღარ გახდეს სრული გადარჩევა;
    - მორგება მომხმარებლის შესახებ ინფორმაციის გამოყენებით. ასეთი ინტელექტუალური მეთოდი ეფუძნება იმ ფაქტს, რომ თუ ინფორმაციული უსაფრთხოების პოლიტიკა უშვებს პაროლების თავისუფლად შექმნას მომხმარებლის მიერ, უმრავლეს შემთხვევაში პაროლად ირჩევენ მომხმარებლის პერსონალურ ინფორმაციას. ასეთ ინფორმაციად ირჩევენ დაწყებულს სიდედრის დაბადების

დღიდან, დამთავრებული საყვარელი ძაღლის სახელით. თუ მომხმარებლის შესახებ ინფორმაცია ბოროტმოქმედისათვის ცნობილია, მან შეიძლება პაროლი მოარგოს.

3. პაროლური სისტემების რეალიზაციის სისუსტეების გამოყენებით. რეალიზაციის ასეთ სისუსტეებად ითვლება ქსელური სერვისების რეალიზაციის არასანდოობა და სისუსტეები, ან პროგრამული უზრუნველყოფის ან აპარატურის არადეკლარირებული შესაძლებლობები.

პაროლური სისტემების აგებისას უნდა მოხდეს ინფორმაციული სისტემის სპეციფიკის გათვალისწინება და ჩატარებული რისკების ანალიზის შედეგები. გასათვალისწინებელია შემდეგი პრაქტიკული რეკომენდაციები:

- პაროლის მინიმალური სიგრძის მოთხოვნა;
- პაროლის ალფაბეტის სიმძლავრის გაზრდა;
- ლექსიკონის მიხედვით პაროლის გასინჯვა და განთესვა;
- პაროლის მოქმედების მაქსიმალური დროის მოთხოვნა;
- პაროლის მოქმედების მინიმალური დროის მოთხოვნა;
- პაროლების ისტორიის ჟურნალის მიხედვით განთესვა;
- პაროლის შეტანის მცდელობათა რაოდენობის განსაზღვრა (შეზღუდვა);
- პაროლის შეცვლის აუცილებლობა პირველი შესვლის შემდეგ;
- არასწორი პაროლის შეტანის შემთხვევაში დაყოვნება;

მომხმარებლის მიერ პაროლის შერჩევისა და პაროლის ავტომატურად გენერირების აკრძალვა.

გავრცელებული საფრთხეები, თანამედროვე თავდასხმების კლასიფიკაცია, მათთან ბრძოლის მეთოდები

როგორც ზემოთ აღვნიშნეთ, საფრთხე ეწოდება პოტენციურად შესაძლო ისეთ მოქმედებებს, მდგომარეობებს, პროცესებს ან მოვლენებს, რომლებმაც შეიძლება ზიანი მიაყენოს ვინმეს ინტერესებს. ინფორმაციული სისტემის საფრთხე - ესაა ისეთი ზემოქმედებების რეალიზაცია ინფორმაციულ სისტემაში დასამუშავებელ მონაცემებზე, რომელიც იწვევს კონფიდენციალობის, მთლიანობის ან წვდომის დარღვევას.

შესაძლო საფრთხის გამოვლენა შეიძლება სხვადასხვაგვარი იყოს:

- ორგანიზაციის საქმიანი რეპუტაციის მორალური და მატერიალური ზიანი;
- ცალკეული პიროვნებების მორალური, მატერიალური ან ფიზიკური ზარალი, რომელიც დაკავშირებულია მისი პერსონალური ინფორმაციის გამხელასთან;
- მატერიალური (ფინანსური) ზარალი გამოწვეული კონფიდენციალური (დაცული) ინფორმაციის გამჟღავნებასთან;
- მატერიალური (ფინანსური) ზარალი, დაკავშირებული დაცული ინფორმაციული რესურსების აღდგენის აუცილებლობასთან;
- მატერიალური ზარალი (დანაკარგები), გამოწვეული იმ შეუსრულებელი ვალდებულებების გამო, რაც აღებული იქნა მესამე მხარის მიმართ;
- მორალური და მატერიალური ზარალი გამოწვეული საწარმოს მუშაობის დეზორგანიზაციის გამო.

სხვადასხვა ქვეყნის კანონმდებლობით, საწარმოს ინფორმაციული უსაფრთხოების სფეროში დანაშაულად ითვლება:

- ინფორმაციის მოპარვა;
- კომპიუტერული ინფორმაციის კოპირება;
- ინფორმაციის განადგურება (ინფორმაციაზე ისეთი ზემოქმედება, როდესაც ის ფიზიკურად აღარ იარსებებს ან მისი მიზნობრივი გამოყენება შეუძლებელი გახდება);
- კომპიუტერული ინფორმაციის განადგურება (კომპიუტერის მეხსიერებიდან წაშლა);
- ინფორმაციის დაზიანება;
- კომპიუტერული ინფორმაციის მოდიფიკაცია - ნებისმიერი ცვლილების შეტანა, გარდა იმ ცვლილებებისა რაც საჭიროა პროგრამების ან მონაცემთა ბაზების ადაპტაციისათვის;
- კომპიუტერული ინფორმაციის ბლოკირება - მომხმარებლის მიერ ინფორმაციაზე წვდომის ხელოვნური დაბრკოლება;
- ინფორმაციის არასანქცირებული განადგურება, ბლოკირება, მოდიფიცირება, კოპირება;
- გაყალბება (ყალბი ინფორმაციის თავსმოხვევა და გავრცელება);

მნიშვნელოვანია ორგანიზაციების მართვაში ინტეგრირებული ინფორმაციულ სისტემებში ყველაზე მეტად გავრცელებული საფრთხეების ანალიზი. ორგანიზაციის ლოკალურ ქსელსა და ორგანიზაციული მართვის ინფორმაციულ სისტემას პოტენციურად

საფრთხე შეიძლება შეუქმნას ნებისმიერმა პირმა, ობიექტმა ან ხდომილებამ. საფრთხის იდენტიფიკაცია გულისხმობს საფრთხის ზემოქმედებებისა და განხორციელებული შედეგების განხილვას. ჩვეულებრივ, საფრთხის ზემოქმედება იწვევს ინფორმაციის გამჟღავნებას, მოდიფიკაციას, განადგურებას ან მომსახურებაზე უარს.

შესაძლო საფრთხეების და დაცვის სუსტი წერტილების ცოდნა საჭიროა, რათა შეირჩეს უსაფრთხოების დაცვის ყველაზე უფრო ეკონომიური საშუალებები.

ყველაზე უფრო ხშირი და საშიში, მიყენებული ზარალის ზომის მხრივ, ესაა არა საფრთხეები, არამედ მომხმარებლების, ოპერატორების, სისტემური ადმინისტრატორების და სისტემის მომსახურე სხვა პერსონალის უნებლიე **შეცდომები.** ზოგჯერ ასეთი შეცდომები წარმოადგენენ საფრთხეებს (არასწორად შეტანილი მონაცემები, შეცდომები პროგრამებში), ხოლო ხანდახან ესაა ადამიანური სისუსტეები, რომლითაც შეიძლება ისარგებლოს ბოროტმოქმედებმა. ასეთია მაგალითად, ადმინისტრირების შეცდომები. დანაკარგების დაახლოებით 65% უნებლიე შეცდომების შედეგია. საერთოდ ხანძარი და წყალდიდობა არაფერია იმასთან შედარებით, რა ზარალსაც იწვევს ბევრი თანამშრომლის უვიცობა და უპასუხისმგებლობა. უნებლიე შეცდომებთან ბრძოლის ყველაზე რადიკალური მეთოდია - მაქსიმალური ავტომატიზება და შესრულებული მოქმედებების სისწორის მკაცრი კონტროლი.

ზარალის მიხედვით მეორე ადგილზე დგას ინფორმაციის **ქურდობები და გაყალბებები**. ძალიან საშიშია ე.წ. ”დაჩაგრული თანამშრომლები”, რომლებიც ცდილობენ მიაყენონ ზარალი ”მჩაგვრელ” ორგანიზაციას. ამ მიზნით მათ შეუძლიათ:

- დააზიანონ მოწყობილობები;
- სისტემაში ”ჩააშენონ” ლოგიკური ბომბი;
- შეიტანონ არასწორი ინფორმაცია;
- წაშალონ ან შეცვალონ ინფორმაცია.

გარემო პირობებიდან გამომდინარე საფრთხეები შეიძლება ძალიან მრავალფეროვანი იყოს. პირველ რიგში შეიძლება აღინიშნოს ინფრასტრუქტურის დარღვევა - ელექტრომომარაგების დარღვევა, კავშირის დროებითი დაკარგვა, სამოქალაქო

უწესრიგობები და სხვა. ხანძარზე, წყალზე და ინფორმაციული სისტემების ანალოგიურ ”მტრებზე” მოდის ზარალის 13%.

საგულისხმოა, რომ ყოველი ინტერნეტ-სერვერი, დღეში რამოდენიმეჯერ განიცდის შეღწევის მცდელობებს. ზოგჯერ ასეთი მცდელობები ”წარმატებულია”. ხშირად ისინი დაკავშირებულია ჯაშუშობასთან. აქედან ცხადია რამდენად მნიშვნელოვანი და სერიოზულია ინფორმაციული უსაფრთხოება.

საკმაოდ აქტუალური და მნიშვნელოვანია პროგრამული **ვირუსების** საკითხი.

ქვემოთ ჩამოთვლილია გავრცელებული **ტექნიკური საფრთხეები და მიზეზები**, რაც მათ იწვევს:

- **ლოკალურ გამოთვლით ქსელთან ან ინფორმაციულ სისტემასთან არაავტორიზებული წვდომა** - მიიღწევა არაავტორიზებული პიროვნებისგან ქსელთან ან ინფორმაციულ სისტემასთან წვდომის მიღებით;
- **ლოკალურ ქსელთან უნებართვო ან შეუსაბამო წვდომა** - ხდება ქსელის რესურსებზე ავტორიზებული ან არაავტორიზებული პირის მიერ წვდომის განხორციელება, არაავტორიზებული საშუალებით;
- **მონაცემების გამჟღავნება** - ხდება როცა ინფორმაციასთან წვდომას ან მის წაკითხვას ახორციელებს პირი არაავტორიზებული გზით;
- **მონაცემებისა და პროგრამების არაავტორიზებული მოდიფიკაცია** - შესაძლებელია ადამიანის მიერ ინფორმაციის ან ლოკალური ქსელის პროგრამული უზრუნველყოფის მოდიფიკაცია, წაშლა ან განადგურება არავტორიზებული ან შემთხვევითი გზით;
- **ლოკალური ქსელის ტრაფიკის გამჟღავნება** - ხდება ლოკალური ქსელით ინფორმაციის გადაცემის პროცესში მონაცემებზე წვდომა ან წაკითხვა ადამიანის მიერ არაავტორიზებული გზით და მისი შემდგომი გამჟღავნება;
- **ლოკალური ქსელის ტრაფიკის გაყალბება** - ესაა მისი გამოყენება არაავტორიზებული გზით, როდესაც ხდება ინფორმაციის გაგზავნა თითქოსდა კანონიერი გამგზავნის მიერ, არადა სინამდვილეში ასე არაა;
- **ლოკალური ქსელის უწესივრობა** - ესაა საფრთხის განხორციელების შედეგი, რომელიც არ იძლევა საშუალებას ლოკალური ქსელის რესურსების წვდომას.

ინფორმაციის დაცვის სამსახურები - ესაა დაცვის მექანიზმებისა და ორგანიზაციული ღონისძიებების მთელი ერთობლიობა, რომლებიც საჭიროა ლოკალური ქსელებისა და ინფორმაციული სისტემების საფრთხეებისაგან თავდასაცავად. მაგალითად აუთენტიფიცირების და იდენტიფიცირების სამსახურები იცავს ქსელსა და ინფორმაციულ სისტემას არაავტორიზებული შეღწევისაგან, ვინაიდან თითოეული მომხმარებლისგან ითხოვს თავისი თავის აუცილებელ იდენტიფიცირებას და თავისი იდენტიფიკატორის ჭეშმარიტების თანხმობას.

ინფორმაციული სისტემისა და ინტერნეტის ინტეგრირების საფრთხეები

თანამედროვე ორგანიზაციები აფართოებენ თავიანთ ინტრანეტს, სტატიკური შინაარსის ინფორმაციით, რომლებსაც გადასცემენ ინტერნეტის არხებით, გადადიან სხვადასხვა თვითმომსახურე სამომხმარებლო პროგრამებზე და ელექტრონული ბიზნესის რგოლების ორგანიზებაზე. Web-ის საშუალებით ინფორმაციის წვდომა ხდება სულ უფრო სრული და მოქნილი. Web-ის გამოყენების ამ გაფართოებამ გამოიწვია Web-სივრცის დაცვისადმი მიდგომების გადახედვა. ორგანიზაციამ უნდა გაუწიოს მართვა არამარტო იმათ, ვისაც მის კორპორატიულ Web-თან აქვს წვდომა, არამედ ყოველი კონკრეტული მომხმარებლის თითოეულ რესურსს. ინტრანეტისა და ექსტრანეტის უკეთ გამოყენებისგან მიღებული უპირატესობებით (მოგებებით) ორგანიზაციამ უნდა მართოს Web-იდან წვდომადი ყველა ინფორმაცია. მომხმარებლებს უნდა მისცეს უფლება გამოიყენონ მხოლოდ ის ინფორმაცია, რომლებიც მისთვის აუცილებელია და არა უფრო მეტი. ყველა შესაძლო ინფორმაციულ საშუალებაზე წვდომის უფლებამ, ორგანიზაცია შეიძლება დააყენოს შიდა საფრთხეების და თავდასხმების საშიშროებების წინაშე.

ინფორმაციული სისტემის ინტერნეტთან ურთიერთობისას, ორგანიზაციისათვის ძირითად ინფორმაციულ საფრთხეებს წარმოადგენს:

- ინფორმაციულ სისტემაზე ინტერნეტიდან არასანქცირებული გარე ზემოქმედებები, მის რესურსებზე წვდომისათვის და/ან მისი მუშა მდგომარეობის დარღვევისათვის;

- ინტერნეტთან კავშირის არხების, ქსელის სატელეკომუნიკაციო მოწყობილობების და ქსელთაშორისი ეკრანების აპარატურული და პროგრამული უზრუნველყოფის მტყუნებები;
- ორგანიზაციის თანამშრომლების, წინასწარ განზრახვის გარეშე (შეცდომით, შემთხვევით) განხორციელებული მოქმედებები, რომლებიც იწვევს რესურსებისა და დროის არასაწარმოო ხარჯვას, შეზღუდული გავრცელების (კონფიდენციალური) ინფორმაციის ინტერნეტით გამჟღავნებას;
- ორგანიზაციის თანამშრომლების მიერ, წინა პუნქტში ჩამოთვლილი საფრთხეების ჩადენა წინასწარგანზრახულად;
- ორგანიზაციის იმ თანამშრომლების მიერ, რომლებიც პასუხისმგებელი არიან სისტემების, ქსელების და სამომხმარებლო პროგრამების დაყენებაზე, ადმინისტრირებაზე, მხარდაჭერაზე, ჩადენილი შემთხვევითი ან წინასწარგანზრახული ქმედებები, რომლებიც იწვევენ ინფორმაციის გაჟონვას ან ინტერნეტთან კავშირის მოშლას;

შეიძლება დაისვას კითხვა: თუ ინტერნეტი ასეთ საფრთხეებს წარმოშობს, საჭიროა მისი გამოყენება? რა თქმა უნდა საჭიროა, ვინაიდან ინტერნეტი ორგანიზაციას საშუალებას აძლევს გამოიყენოს მისი მნიშვნელოვანი ინფორმაციულ-ანალიტიკური შესაძლებლობები და ინტერნეტში განთავსებული უსაზღვრო ინფორმაციული რესურსები. მაგრამ ინფორმაციული უსაფრთხოების კუთხით დაცული უნდა იყოს გარკვეული მოთხოვნები. ინტერნეტის ქსელთან მიერთება მკაცრად უნდა კონტროლდებოდეს სპეციალური ინფორმაციული სამსახურის მიერ, რომელიც პასუხისმგებელი უნდა იყოს ასეთ ნებართვებზე. ინტერნეტის ქსელთან დაკავშირების აპარატურული და პროგრამული უზრუნველყოფის ყველა კომპონენტი გამოყენებული უნდა იქნას მხოლოდ სამსა-ხურეობრივი მიზნით.

ინფორმაციული სისტემების საფრთხეების ანალიზი და რისკების შეფასება

ინტერნეტის პოპულარობის კოლოსალურ ზრდასთან ერთად, წარმოიშობა პერსონალური ინფორმაციის, კრიტიკულად მნიშვნელოვანი კორპორატიული რესურსების,

სახელმწიფო საიდუმლოებების და ა.შ. უპრეცედენტო საფრთხეები. ყოველდღიურად ჰაკერები საფრთხის წინაშე აყენებენ ამ რესურსებს, ცდილობენ რა დაამყარონ მათთან წვდომა სპეციალური შეტევების გზით. ეს შეტევები, სულ უფრო ფართო მასშტაბის და საშიში ხდება, რასაც ხელს უწყობს შემდეგი ორი ფაქტორი:

- **ინტერნეტის საყოველთაო გავრცელება.** დღესათვის ამ ქსელში ჩართულია მილიონობით მოწყობილობა. ამიტომ ჰაკერების მიერ დაუცველ მოწყობილობებში შეღწევის ალბათობა მუდმივად იზრდება. გარდა ამისა, ინტერნეტში შეიძლება მოიძებნოს ჰაკერებისათვის გამოსადეგი სხვადასხვა ინფორმაცია, შეტევის განხორციელების ინსტრუქციები, ზიანის გამომწვევი პროგრამის კოდები და მათი გამოყენების ხერხები;
- ადვილი მოსახმარისი **ოპერაციული სისტემებისა და მათი დამუშავების საშუალებების საყოველთაო გავრცელება**. თუ ადრე ჰაკერს მაღალ დონეზე უნდა ცოდნოდა პროგრამირების მეთოდები, ახლა, შეტევების განხორციელებისათვის, უკვე გამზადებული საჭირო ინსტრუმენტალური საშუალებების გამოყენებით, ადვილად შეუძლია ზიანი მიაყენოს სხვადასხვა საიტს, მხოლოდ IP მისამართის გამოყენებითაც კი.

ქსელური შეტევების კლასიფიკაცია

ქსელური შეტევები იმდენად მრავალფეროვანია, რამდენადაც მრავალფეროვანია ის სისტემები, რომლის მიმართაც ხორციელდება შეტევები. ზოგიერთი შეტევის მექანიზმი საკმაოდ რთულია, ხოლო ზოგიერთი შეიძლება განახორციელოს ჩვეულებრივმა ოპერატორმა, რომელმაც შეიძლება არც იცოდეს რა შედეგი მოყვება მის ქმედებას. შეტევების ტიპების შესაფასებლად საჭიროა იმ შეზღუდვების ცოდნა, რაც თავიდანვე ახასიათებდა TCP/IP პროტოკოლს. როგორც ცნობილია, ინტერნეტი იქმნებოდა სახელმწიფო ორგანიზაციებსა და უნივერსიტეტების ურთიერთდასაკავშირებლად. მაშინ ქსელის შემქმნელები ალბათ ვერც წარმოიდგენდნენ ქსელის ასე ფართოდ გავრცელებას. ინტერნეტ-პროტოკოლის (IP) ადრეულ ვერსიებში არ იყო ჩადებული უსაფრთხოების მოთხოვნები, ამიტომ ეს პროტოკოლები თავიდანვე შეიცავდნენ საფრთხეებს.

პაკეტების სნიფერი

პაკეტების სნიფერი (”დაყნოსვა”) წარმოადგენს გამოყენებით პროგრამას, რომელიც იყენებს promiscuous mode რეჟიმში მომუშავე ქსელურ ადაპტერს (ამ რეჟიმში ფიზიკური არხებიდან მიღებული ყველა პაკეტი ქსელური ადაპტერის მიერ ეგზავნება გამოყენებით პროგრამას). სნიფერი აწარმოებს ყველა პაკეტის გამოჭერას, რომლებიც გადაიცემა რომელიმე განსაზღვრულ დომენში. სნიფერი თანამედროვე ქსელში შეიძლება მუშაობდეს სრულიად კანონიერ საფუძველზე. ისინი გამოიყენება ტრაფიკის ანალიზისა და დაზიანების დიაგნოსტიკისათვის. ზოგიერთი ქსელური გამოყენებითი პროგრამა მონაცემებს გადასცემს ტექსტურ ფორმატში (telnet, FTP, SMTP, POP3 და სხვა), ამიტომ სნიფერით შეიძლება გავიგოთ სასარგებლო, ხოლო ხანდახან კონფიდენციალური ინფორმაცია (მაგ. მომხმარებლების სახელები, პაროლები).

სახელებისა და პაროლების გამოჭერა წარმოადგენს დიდ საშიშროებას, ვინაიდან მომხმარებლები ხშირად ერთი და იგივე სახელსა და პაროლს იყენებენ სხვადასხვა პროგრამებისათვის. ბევრი მომხმარებელი საერთოდ ერთ პაროლს იყენებს ყველა რესურსისა და პროგრამის წვდომისათვის. თუ პროგრამა მუშაობს კლიენტ-სერვერის რეჟიმში, ხოლო ქსელში გადაცემული აუთენტურობის მონაცემების წაკითხვა შესაძლებელია ტექსტურ ფორმატში, ამ ინფორმაციის გამოყენებით დიდი ალბათობით შესაძლებელია კორპორატიულ და სხვა რესურსებზე წვდომა. ჰაკერები ხშირად შეტევისათვის იყენებენ ასეთ ადამიანურ სისუსტეებს, ვინაიდან იციან, რომ ადამიანები ძირითადად ერთი და იგივე პაროლებს იყენებენ სხვადასხვა რესურსებზე წვდომისათვის. როდესაც ჰაკერი გაიგებს ამა თუ იმ პაროლს, მას შეუძლია სისტემურ დონეზე განახორციელოს წვდომა მნიშვნელოვან ინფორმაციაზე ან მომხმარებლის რესურსზე. პაკეტების სნიფინგის საშიშროების შემცირების გზებია:

აუთენტიკაციით - ეს პაკეტების სნიფინგისგან დაცვის მძლავრი საშუალებაა, ვინაიდან მისი იგნორირება ან შემოვლა ძნელია. ასეთი აუთენტიკაციის მაგალითია ერთჯერადი პაროლები (OTP – One-Time Passwords). OTP – ესაა ორფაქტორიანი ტექნოლოგია, რომელიც დაფუძნებულია იმ შესაბამისობაზე რაც თქვენ გაქვთ და რაც თქვენ იცით. ორფაქტორიანი აუთენტიკაციის ტიპიური მაგალითია ბანკომატი, რომელიც ამოიცნობს მომხმარებელს

ჯერ-ერთი მისი პლასტიკური ბარათით, ხოლო მეორე - მის მიერ შეტანილი PIN-კოდით. ქსელში - ”ბარათი”-ის (token) როლში იგულისხმება აპარატურული ან პროგრამული საშუალება, რომელიც შემთხვევითად გამოიმუშავებს უნიკალურ ერთჯერად პაროლს. თუ ჰაკერი სნიფერის საშუალებით ამ პაროლს გაიგებს, ეს ინფორმაცია მისთვის გამოუსადეგარი იქნება, ვინაიდან ამ მომენტისთვის ეს პაროლი ითვლება უკვე გამოყენებულად და იგი ხელ-მეორედ აღარ გამოიყენება. ეს მეთოდი ეფექტურია მხოლოდ პაროლების გამოსაჭერად. სნიფერები, რომლებიც სხვა ინფორმაციას გამოიჭერენ, სამწუხაროდ თავის ეფექტურობას არ კარგავენ.

კომუტირებული ინფრასტრუქტურა - ქსელურ გარემოში სნიფინგთან ბრძოლის ამ გზით მნიშვნელოვნად მცირდება საფრთხის სიმწვავე. მაგალითად, თუ მთელ ორგანიზაციაში გამოყენებულია კომუტირებადი Ethernet-ი, მაშინ ჰაკერს შეუძლია წვდომის დამყარება მხოლოდ იმ ტრაფიკთან, რომელიც მიეწოდება მხოლოდ იმ პორტს, რომელთანაც თვითონაა მიერთებული. სხვა ტრაფიკი მისთვის მიუწვდომელი იქნება.

ანტი-სნიფერი - სნიფინგთან ბრძოლის ეს გზა მდგომარეობს, ისეთი აპარატურული და პროგრამული საშუალებების გამოყენებაში, რომელიც გამოიცნობს ქსელში მომუშავე სნიფერებს. ამ საშუალებებს საფრთხის სრული ლიკვიდაცია არ შეუძლიათ, მაგრამ, როგორც კიდევ ბევრი სხვა ქსელური უსაფრთხოების საშუალება, ჩართულია დაცვის ზოგად სისტემაში. ე.წ. ანტი-სნიფერები ზომავენ ჰოსტების რეაგირების დროს და განსაზღვრავენ ხომ არ უწევთ ჰოსტებს ”ზედმეტი” ტრაფიკის დამუშავება.

კრიპტოგრაფია - ყველაზე ეფექტური ხერხია პაკეტების სნიფინგისას, რომელიც მართალია არ კრძალავს და არ ამოიცნობს სნიფერის მუშაობას, მაგრამ მას გამოუსადეგარს ხდის. თუ კავშირის არხი კრიპტოგრაფიულად დაცულია, ეს იმას ნიშნავს, რომ ჰაკერი გამოიჭერს არა შეტყობინებას, არამედ დაშიფრულ შეტყობინებას (ანუ მისთვის გაუგებარ ბიტების მიმდევრობას).

IP-სპუფინგი

IP-სპუფინგი (შეერთების იმიტაცია ან მოტყუება) ხდება, როდესაც ჰაკერი იმყოფება კორპორაციის შიგნით ან მის გარეთ თავს ასაღებს როგორც სანქცირებული მომხმარებელი. ამის განხორციელება ორი გზით შეიძლება: ჰაკერმა შეიძლება გამოიყენოს IP-მისმართი, რომელიც სანქცირებულია IP-მისამართების საზღვრების დიაპაზონში ან გამოიყენოს ავტორიზებული გარე მისამართი, რომელსაც უფლება აქვს მიმართოს ქსელის გარკვეულ რესურსებს. IP-სპუფინგით ხშირად იწყება სხვა სახის შეტევები.

ჩვეულებრივ IP-სპუფინგის დროს, მონაცემთა ნაკადში, რომელიც გადაიცემა კლიენტსა და სერვერს ან ერთრანგიანი ქსელის მოწყობილობების კავშირის არხებს შორის, ჩაისმება ყალბი ინფორმაცია ან ზიანის მომტანი ბრძანებები. ორმხრივი კავშირის დასამყარებლად ჰაკერმა, ტრაფიკის ყალბ IP-მისამართზე გადასაგზავნად, უნდა შეცვალოს მარშრუტიზაციის ყველა ცხრილი. ზოგჯერ ჰაკერები არც კი ცდილობენ პროგრამის პასუხის მიღებას. თუ მთავარი ამოცანაა სისტემისგან მნიშვნელოვანი ფაილის მიღება, მაშინ პროგრამაზე პასუხებს მნიშვნელობა არა აქვთ.

თუ ჰაკერი შეძლებს მარშრუტიზაციის ცხრილების შეცვლას და ტრაფიკის ყალბ IP-მისამართზე მიმართვას (გაგზავნას), მაშინ ჰაკერი მიიღებს ყველა პაკეტს და შეძლებს უპასუხოს მათ, თითქოს ის იყოს სანქცირებული მომხმარებელი.

სპუფინგის საფრთხე შეიძლება შემცირდეს (მაგრამ არ აღმოიფხვრას) შემდეგი ღონისძიებებით:

- **წვდომის კონტროლი** - IP-სპუფინგისგან თავის დაღწევის ყველაზე მარტივი გზაა წვდომის მართვის სწორი ორგანიზაცია. IP-სპუფინგის ეფექტურობის შესამცირებლად წვდომის კონტროლი ისე უნდა დარეგულირდეს, რომ მოიკვეთოს გარე ქსელიდან შემომავალი ისეთი საწყისი მისამართებიდან მომავალი ტრაფიკი, რომელიც წესით შიდა ქსელიდან უნდა მოდიოდეს. თუ გარე ქსელების თუნდაც ზოგიერთი მისამართები სანქცირებულია, ეს მეთოდი არაეფექტურია.
- RFC 2827 ფილტრაცია - ამ მეთოდით შეიძლება აღიკვეთოს სხვისი ქსელების სპუფინგის მცდელობები თქვენი ქსელის მომხმარებლების მიერ (და ამით გახდეთ წესიერი "ქსელური მოქალაქე"). ამისათვის საჭიროა დაიბლოკოს ყველა ის გამავალი ტრაფიკი,

რომლის მისამართი არაა თქვენი ქსელის ერთ-ერთი IP-მისამართი. ასეთი ფილტრაციას ეძახიან "RFC 2827" ფილტრაციას და იგი შეიძლება განახორციელოს პროვაიდერმაც. სამწუხაროდ, თუ ყველა პროვაიდერი არ დანერგავს ასეთ ფილტრებს, მისი ეფექტურობა დაბალი იქნება.

აუთენტიკაციის დამატებითი მეთოდების შემოტანა - IP-სპუფინგის წინააღმდეგ საბრძოლველად ეფექტური მეთოდი ისეთივეა, როგორიც პაკეტების IP-სნიფინგის შემთხვევაში: საჭიროა შეტევის აბსოლუტურად არაეფექტურად გადაქცევა. IP-სპუფინგი შეიძლება ფუნქციონირებდეს მხოლოდ იმ პირობით, რომ აუთენტიკაცია ხდება IP-მისამართების ბაზაზე. ამიტომ აუთენტიკაციის დამატებითი მეთოდების გამოყენება ამ ტიპის შეტევებს გამოუსადეგარს ხდის. დამატებითი აუთენტიკაციის საუკეთესო გზაა **კრიპტოგრაფია.** თუ ეს შეუძლებელია, მაშინ კარგ შედეგებს იძლევა **ერთჯერადი პაროლებით ორფაქტორიანი აუთენტიკაცია**.

უარი მომსახურებაზე (Denial of Service - DoS)

DoS - წარმოადგენს ჰაკერული შეტევის ყველაზე ცნობილ ფორმას. ძალიან რთულია ამ ტიპის შეტევის წინააღმდეგ 100%-იანი დაცვა. DoS შეტევის ორგანიზებისა და განხორციელებისათვის ყველაზე ნაკლები ცოდნა და უნარია საჭირო. მიუხედავად ამისა, სწორედ რეალიზაციის ეს სიმარტივე და გამოწვეული უდიდესი ზიანის შესაძლებლობა იწვევს ადმინისტრატორების მხრიდან DoS-შეტევების მიმართ დიდ ყურადღებას. DoS-შეტევების გავრცელებული ნაირსახეობებია:

- TCP SYN Flood,
- Ping of Death,
- Trible Flood Network (TFN) Tribe Flood Network 2000 (TFN2K),
- Trinco,
- Stacheldracht,
- Trinity.

DoS-შეტევები განსხვავდება სხვა შეტევებისგან - ისინი არაა მიმართული თქვენი ქსელისადმი წვდომაზე ან ამ ქსელიდან რაიმე ინფორმაციის მიღებაზე. DoS-შეტევა ქსელს

ხდის ჩვეულებრივი მოხმარებისათვის მიუწვდომელს, რაც გამოწვეულია ქსელზე განხორციელებული აგრესიის გამო ქსელის, ოპერაციული სისტემის ან გამოყენებითი პროგრამის ფუნქციონირების დასაშვები პარამეტრების გადაჭარბებით.

ზოგიერთი სერვერული პროგრამის გამოყენებისას (როგორიცაა Web-სერვერი ან FTP-სერვერი) DoS-შეტევამ შეიძლება გამოიწვიოს, ამ პროგრამებისათვს განკუთვნილი ყველა შეერთების დაკავება და ჩვეულებრივი მომხმარებლის მომსახურება შეუძლებელი აღმოჩნდება. DoS-შეტევების დროს შეიძლება გამოყენებული იქნას ჩვეულებრივი TCP ან ICMP (Internet Control Message Protocol) პროტოკოლები. DoS-შეტევების დიდი უმრავლესობა ხორციელდება პროგრამული შეცდომების ან სისტემის უსაფრთხოებაში "ღრეჩოების"- გამო. ზოგიერთი შეტევა ქსელის წარმადობას ნულამდე ამცირებს, გადაავსებს მას უსარგებლო და არასასურველი პაკეტებით. ამ შეტევებისგან თავდაცვა ძნელად განსახორციელებელია, ვინაიდან აუცილებელია პროვაიდერთან კოორდინირება. თუ ტრაფიკი, რომელიც ქსელს გადაავსებს, პროვაიდერთან არ გადაიკეტა, ქსელში შესასვლელთან ეს უკვე აღარ მოხერხდება, ვინაიდან ქსელის გამტარებლობა უკვე გადავსებული იქნება. როდესაც ასეთი შეტევა ხორციელდება ერთდროულად მრავალი მოწყობილობდან, მას ეძახიან "განაწილებულს" (DDoS – Distributed DoS).

DoS ტიპის შეტევები შეიძლება შევამციროთ შემდეგი 3 ხერხით:

- **ანტისპუფინგის ფუნქციების გამოყენებით.** მარშრუტიზატორებში და ქსელთაშორის ეკრანებში ანტისპუფინგური ფუნქციის სწორი კონფიგურირება გულისხმობს როგორც მინიმუმ RFC 2827 ფილტრაციის გამოყენებას. ამ შემთხვევაში, ჰაკერი ვერ შეძლებს დამალოს მისი ნამდვილი პიროვნება და შეტევის განხორციელება გაუჭირდება.
- **ანტი-DoS ფუნქციების რეალიზაცია.** ანტი-DoS ფუნქციების სწორი კონფიგურირება მარშრუტიზატორებსა და ქსელთაშორის ეკრანებში შეზღუდავს შეტევის ეფექტურობას. ეს ფუნქციები ხშირად ზღუდავენ ნებისმიერ მომენტში ნახევრადგახსნილი არხების რიცხვს.
- **ტრაფიკის მოცულობის შეზღუდვა** (Traffic rate limiting). ორგანიზაციას შეუძლია მოსთხოვოს პროვაიდერს შეუზღუდოს ტრაფიკის მოცულობა. ამ ტიპის ფილტრაცია შესაძლებელს გახდის შეზღუდოთ არაკრიტიკული ტრაფიკის მოცულობა, რომელიც

თქვენს ორგანიზაციაში გადის. ტიპიურ მაგალითს წარმოადგენს ICMP ტრაფიკის მოცულობის შეზღუდვა, რომელიც გამოიყენება მხოლოდ დიაგნოსტიკისათვის. DoS-ის შეტევები კი ხშირად ხორციელდება სწორედ ICMP-ს პაკეტების გამოყენებით.

პაროლებზე შეტევა

ჰაკერებს შეუძლიათ განახორციელონ პაროლებზე შეტევა მთელი რიგი მეთოდების გამოყენებით, ისეთი როგორიცაა მარტივი გადარჩევა (brute force attack), ტროას ცხენი, IP სპუფინგი და პაკეტების სნიფინგი. თუმცა მომხმარებლის სახელისა და პაროლის მიღება შესაძლებელია IP სპუფინგისა და პაკეტების სნიფინგის გამოყენებით, ჰაკერები ხშირად ცდილობენ მოარგონ პაროლი და სახელი, მიმართვის მრავალრიცხოვანი მცდელობით. ამას მარტივ გადარჩევას (brute force attack) უწოდებენ.

ხშირად მსგავსი შეტევისათვის გამოიყენება სპეციალური პროგრამა, რომელიც ცდილობს მიიღოს საერთო მოხმარების რესურსზე წვდომა (მაგ. სერვერზე). თუ ჰაკერი ამას შეძლებს, მაშინ ის გახდება თითქოსდა ჩვეულებრივი მომხმარებელი, რომელმაც პაროლი სწორად შეიტანა. თუ ამ მომხმარებელს აქვს მნიშვნელოვანი წვდომის პრივილეგია, ჰაკერმა შეიძლება შეიქმნას, მიმართვის უფლება მომავალში, რომელსაც გამოიყენებს მომხმარებლის მიერ თავისი პაროლისა და სახელის შეცვლის შემთხვევაშიც კი.

კიდევ ერთი პრობლემა წარმოიქმნება, როდესაც მომხმარებლები იყენებენ ერთი და იგივე (თუნდაც ძალიან კარგს) პაროლს კორპორაციულ, პერსონალურ და ინტერნეტ სისტემებთან წვდომისას. რადგან პაროლის მდგრადობა განსაზღვრულია ყველაზე სუსტი ჰოსტის მდგრადობით, ჰაკერი, რომელმაც გაიგო პაროლი ამ ჰოსტის გზით, მიიღებს წვდომას ყველა დანარჩენ რესურსთან, სადაც გამოიყენება იგივე პაროლი.

პაროლური შეტევის თავიდან აცილება შესაძლებელია თუ არ გამოვიყენებთ ტექსტური ფორმის პაროლებს. ერთჯერადი პაროლები და/ან კრიპტოგრაფიული აუთენტიკაცია მინიმუმამდე ამცირებს მსგავს შეტევებს, მაგრამ სამწუხაროდ ბევრ სისტემას, პროგრამასა და ჰოსტს არა აქვს მსგავსი აუთენტიკაციის მეთოდების მხარდაჭერა.

პაროლების გამოყენებისას უნდა შეირჩეს ისეთი პაროლები, რომელთა გამოცნობა რთული იქნება. პაროლის მინიმალური სიგრძე არ უნდა იყოს 8 სიმბოლოზე ნაკლები.

პაროლები უნდა შეიცავდნენ სიმბოლოების მაღალ და დაბალ რეგისტრებს, ციფრებს, სპეციალურ სიმბოლოებს. კარგი პაროლები რთული შესარჩევი და დასამახსოვრებელია, რაც აიძულებს მომხმარებლებს ჩაიწეროს ისინი ფურცელზე. ეს რომ არ მოხდეს მომხმარებლებს და ადმინისტრატორებს შეუძლიათ გამოიყენონ ტექნოლოგიური სიახლეები, მაგ. არსებობს გამოყენებითი პროგრამები, რომლებიც შიფრავს პაროლების სიას, და რომელიც შეიძლება შევინახოთ ჯიბის კომპიუტერში. ამ შემთხვევაში მომხმარებელს უწევს ერთი რთული პაროლის დამახსოვრება, ხოლო დანარჩენი პაროლები საიმედოდ იქნება შენახული პროგრამის მიერ.

Man-in-the-Middle ტიპის შეტევები

Man-in-the-Middle (შუა-კაცის) ტიპის შეტევისთვის ჰაკერს სჭირდება ქსელით გადაცემულ პაკეტებთან წვდომა. მსგავსი წვდომა ყველა პაკეტთან, რომელიც გადაიცემა პროვაიდერის მიერ ნებისმიერ სხვა ქსელში, შეუძლია, მაგ. მიიღოს პროვაიდერის თანამშრომელმა. მსგავსი შეტევისათვის ხშირად იყენებენ პაკეტების სნიფერებს, სატრანსპორტო პროტოკოლებს და მარშრუტიზაციის პროტოკოლებს. შეტევა ხორციელდება:

- ინფორმაციის მოპარვის მიზნით;
- მიმდინარე სესიის გამოსაჭერად და ცალკეულ ქსელურ რესურსებთან წვდომისათვის;
- ტრაფიკის ანალიზისათვის, რათა მიიღოს ინფორმაცია ქსელზე და მის მომხმარებლებზე;
- DoS შეტევისათვის;
- ტრაფიკის გაყალბებისათვის და არასანქცირებული ინფორმაციის შესატანად ქსელურ სესიაში.

Man-in-the-Middle შეტევის წინააღმდეგ ეფექტურად ბრძოლა შესაძლებელია მხოლოდ კრიპტოგრაფიული მეთოდების გამოყენებით. თუ ჰაკერი დაიჭერს დაშიფრული სესიის ინფორმაციას, მას ეკრანზე გამოუვა არა დაჭერილი ინფორმაცია არამედ უაზრო სიმბოლოების მიმდევრობა. აღსანიშნავია, რომ თუ ჰაკერი მიიღებს ინფორმაციას კრიპტოგრაფიულ სესიაზე (მაგ. სესიის გასაღები), მაშინ მას შეუძლია განახორციელოს Man-in-the-Middle შეტევა დაშიფრულ გარემოშიც კი.

შეტევები გამოყენებით დონეზე

შეტევები გამოყენებით დონეზე ხორციელდება სხვადასხვა გზით. მათ შორის ყველაზე გავრცელებული არის სერვერული პროგრამული უზრუნველყოფის კარგად ცნობილი ნაკლოვანებების გამოყენება (Sendmail, Ftp, HTTP). ამ ნაკლოვანებების გამოყენებით ჰაკერებს შეუძლიათ მიიღონ წვდომა კომპიუტერზე, იმ მომხმარებლის სახელით, რომელიც მუშაობს ამ პროგრამასთან. გამოყენებითი პროგრამების დონეზე განხორციელებული შეტევები ფართოდ შუქდება ტექნიკურ ლიტერატურაში და სხვა პუბლიკაციებში, ვინაიდან ადმინისტრატორებს საშუალება მიეცეთ აღმოფხვრან პრობლემა საკორექციო მოდულებით (”პატჩებით”). სამწუხაროდ ამ ინფორმაციის ნახვა ჰაკერებსაც შეუძლიათ, რაც საშუალებას აძლევთ განავითარონ თავიანთი ხერხები.

ასეთი შეტევების მთავარი პრობლემა მდგომარეობს იმაში, რომ ჰაკერები ხშირად იყენებენ ისეთ პორტებს, რომლებისთვის ნებადართულია ინფორმაციის გაცვლა ქსელთაშორისი ეკრანების მიერ. მაგალითად, ჰაკერები შეტევისას ხშირად იყენებენ TCP-ს 80 პორტს, ვებ-სერვერის ცნობილ სისუსტეს. ვინაიდან Web-სერვერმა მომხმარებელს უნდა მიაწოდოს ვებ-გვერდი, ქსელთაშორისმა ეკრანმა უნდა მისცეს ამ პორტზე წვდომის უფლება. ეკრანის მხრიდან შეტევა განიხილება, როგორც მე-80 პორტთან სტანდარტული ტრაფიკი.

- ყველაზე მთავარი ამ ტიპის შეტევის დროს ესაა კარგი ადმინისტრირება, საფრთხეების შესამცირებლად საჭიროა:
- Log-ფაილების სისტემატური დათვალიერება და ანალიზი, შესაძლებელია ამ მიზნით სპეციალური პროგრამული უზრუნველყოფის გამოყენებაც;
- ოპერაციული სისტემების და გამოყენებითი პროგრამების უახლესი ვერსიების, მათი საკორექციო მოდულების გამოყენება;
- სისტემური ადმინისტრირების გარდა შეტევების ამოცნობის სისტემების გამოყენება.

პორტების გადამისამართება

პორტების გადამისამართება თავისთავად წარმოადგენს ბოროტმოქმედულ დივერსიას, როდესაც გატეხილი ჰოსტი გამოიყენება ქსელთაშორისი ეკრანის გავლით ტრაფიკის გადაცემისთვის, რომელიც სხვა შემთხვევაში აირეკლებოდა ბრანდმაუერის მიერ.

სპამი

სპამი (ინგლ. Spam, Bulk ან Junk) არის ელექტრონული წერილის ტიპი, რომელიც იგზავნება პიროვნების ან კომპანიის მიერ, მიმღების დაუკითხავად და სურვილის გარეშე. მსგავსი წერილები უმეტესწილად სარეკლამო ხასიათისაა. პიროვნებას, რომელიც მსგავს წერილებს გზავნის ეწოდება სპამერი. მათი ძირითადი მიზანია თავიანთი პროდუქტის პოპულარიზაცია.

<u>როგორ ხვდება ჩვენი იმეილი სპამერის ხელში</u> - სპამერისთვის ყველაზე რთულია იმ მისამართების ხელში ჩაგდება, რომელთაც უნდა გაუგზავნოს ესა თუ ის მეილი. რადგან აქ ლაპარაკია არა ერთ და ორ, არამედ ათასობით მეილზე. ამისთვის სპამერები რამოდენიმე ხერხს მიმართავენ:

- ცნობილი კომპანიის მომხმარებლების ბაზის გატეხვა.
- კომპანიის თანამშრომლისგან არაოფიციალური გზით მომხმარებლის მონაცემების შეძენა.
- სხვადასხვა ყველასათვის ნაცნობი და საზოგადო საიტებიდან მონაცემების ამოღება სპეციალური პროგრამებით. (მაგ: Harvester).
- კომპიუტერიდან რომელიც დაინფიცირებულია ტროიანით შესაძლებელია მოიპოვო ყველა მეილი, რომელზეც მოხდა მიმოწერა დაინფიცირებული კომპიუტერით.
- სპამერებს აქვთ გავრცელებული სახელების ბიბლიოთეკა, რომელსაც სხვადასხვა კომბინაციებით იყენებენ და იგებენ უამრავ მეილს.

<u>რა არის სპამი?</u> - სპამი არის ელექტრონული წერილის ტიპი, რომელიც მასობრივად და ანონიმურად იგზავნება მიმღების ელექტრონული ფოსტის მისამართზე, მის დაუკითხავად

და სურვილის გარეშე. მსგავსი წერილები უმეტესწილად სარეკლამო ხასიათისაა და განკუთვნილია რაიმე მომსახურების ან საქონლის რეკლამირებისათვის (პოტენციის ასამაღლებელი აბების რეკლამა, ფიქტიური კომპანიებისგან საფონდო ბირჟებზე მომხიბვლელი გარიგებების შესახებ მოწვევები და უამრავი სხვა დამაინტრიგებელი წინადადება).

პიროვნებას, რომელიც მსგავს წერილებს აგზავნის ეწოდება სპამერი. გარდა რეკლამისა ბოროტმოქმედები სპამს იყენებენ ფიშინგური შეტევების განსახორციელებლად ან მავნე პროგრამების გასავრცელებლად. მიუხედავად იმისა, რომ ინტერნეტ მომხმარებლების უმეტესობამ იცის სპამის შესახებ და უარყოფითად აფასებს მას, არის ასევე ბევრი ისეთი პიროვნება ვინც ხდება სპამის მსხვერპლი. მრავალ ქვეყანაში სპამირება ისჯება კანონით.

ყოველდღიურად მსოფლიოში მილიარდობით ელექტრონული წერილი იგზავნება და სტატისტიკურად ელექტრონული წერილების ტრაფიკის 80%-ზე მეტი არის სპამი! შესაბამისად ნორმალური წერილების რაოდენობა არის 20%-ზე ნაკლები.

<u>სპამის სახეობები</u> - ყველაზე გავრცელებული სპამის სახეობებია:

- ლეგალური პროდუქციის რეკლამა, რომლის ღირებულებაც დაბალია.
- არალეგალური პროდუქციის რეკლამა, რომელიც აკრძალულია კანონით.
- კონკურენტი პროდუქციის ანტირეკლამა.
- ფიშინგი, რომელიც ხდება სპამ მეილის მიღების შემდეგ.

<u>რა არის სპამ ბოტი?</u> - სპამ ბოტი არის სპეციალური პროგრამა, რომელიც აგროვებს ელ-ფოსტის მისამართებს ინტერნეტში. სპამ ბოტი ავტომატურად ათვალიერებს სხვადასხვა ვებ გვერდებს, ფორუმებს, ბლოგებს და ტექსტში ეძებს ელ-ფოსტის მისამართს, შემდგომ ნაპოვნი ელ-ფოსტის მისამართები ხვდება სპამერების წერილების მასობრივი დაგზავნის სიაში, რის საშუალებითაც ისინი აგზავნიან სპამს.

თუ თქვენ ღია ტექსტით რომელიმე ვებ გვერდზე გაქვთ განთავსებული ან დაფიქსირებული თქვენი ელ-ფოსტის მისამართი, დაწრმუნებული იყავით რომ სპამ-ბოტი იპოვის მას, რის შემდგომაც მოგივათ აუარებელი სპამ წერილები, ამიტომ არ გამოაქვეყნოთ თქვენი ელ-ფოსტის მისამართი ვებ-გვერდზე ან ფორუმზე.

ვინაიდან სპამ ბოტი არის მავნე პროგრამა, მას აქვს ასევე სპამის დაგზავნის ფუნქციაც. მან შეიძლება თქვენი კომპიუტერის საშუალებით გააზვნოს ათასობით სპამ წერილი სხვადასხვა ელექტრონულ მისამართზე. მომხმარებლისთის სპამ ბოტის არსებობა მის კომპიუტერში არის ძალზედ სახიფათო, რადგან მისი კომპიუტერის IP მისამართიდან გაგზავნილი სპამი ადრე თუ გვიან მოხვდება რომელიმე შავ სიაში (სპამის ბლოკირების სიები, რომელიც საშუალებას აძლევს ადმინისტრატორებს დაბლოკონ წერილები იმ სისტემებიდან, რომლებსაც აქვთ სპამის დაგზავნის ისტორია), რის შემდგომ მას შეექმენა პრობლემა გააგზავნოს ჩვეულებილივი წერილები, ვინადან თუ მისი IP მისამართი დაფისქსირებულაი რომელიმე შავ სიაში, მიმღების სერვერი ავტომატურად დაბლოკავს მისგან მოსულ ნებისმიერ წერილს. ეს ძალზედ მნიშვენელოვანია ორგანიზაციებისათვოს რომელთაც აქვთ თავიანთი მეილ-სერვერი, რადგან თუნდაც ერთმა მომხმარებელმა შეიძლება შეუქმნას პრობლემა ყველა სხვა მომხმარებელს, რომელიც იყენებს სამსახურებრივ ელ-ფოსტას.

თქვენ შეგიძლიათ შეამოწმოთ თქვენი IP მისამართი შავ სიებში არსებობის შესახებ:

შეამოწმეთ: http://www.mxtoolbox.com/blacklists.aspx და http://www.dnsbl.info

OK - ნიშნავს რომ თქვენი IP მისამართი არის წესრიგში.

Listed - ნიშნავს რომ თქვენი IP მისამართი შეტანილია შესაბამის შავ სიაში.

ბრძოლა სპამთან - აშკარაა, რომ სპამს მოაქვს სერიოზული ეკონომიკური სარგებელი სპამის შემკვეთებისათვის. ამდენად ყველაზე საიმედო გზაა, რომ სპამით მოსულ

რეკლამაზე თქვა უარი და არ შეიძინო სპამით რეკლამირებული ნივთი. სპამისაგან 100% დაცვა არ არსებობს და მთლიანად სპამ წერილებისგან თავის არიდება რთულია, ვინაიდან სპამერებიც არ ჩამორჩებიან დაცვის სისტემებს და სულ ახალ მეთოდებს იგონებენ, რათა მათმა წერილებმა მიაღწიოს ადრესატამდე. ამისათვის თვითონ მომხმარებელმაც უნდა დაიცვას რამოდენიმე წესი რათა მისი ელქტრონული ფოსტის მისამართზე არათუ

მომრავლდეს არამედ შემცირდეს სპამის რაოდენობა. გთხოვთ გაითვალისწინოთ შემდეგი რეკომენდაციები:

- **არ უპასუხოთ სპამ წერილებს** - ამით თქვენ ადასტურებთ რომ თქვენი ელ-ფოსტის მისამართი არის აქტიური და ამის შემდეგ უფრო მეტი არასასურველი წერილი მოგივათ.
- **არ გახსნათ სპამ წერილში მითითებული არცერთი ლინკი (ბმული) რა შინაარსისაც არ უნდა იყოს ის**, ლინკზე დაჭერით შესაძლოა გადახვიდეთ სახიფათო ვებ-გვერდზე.
- **არ გახსნათ სპამ წერილში თანდართული ფაილი (attachment)**, რადგან შეიძლება შეიცავდეს ვირუსს ან მავნე პროგრამას.
- **არ გამოაქვეყნოთ თქვენი ელ-ფოსტის მისამართი ვებ-გვერდზე ან ფორუმზე**, რადგან სპამ ბოტები ათვალიერებენ ვებ გვერდებს და ელ-ფოსტის მისამართის პოვნისას ავტომატურად შეაქვთ სპამ სიაში.
- **შექმენით დამატებითი ელ-ფოსტის მისამართი** - თუ თქვენ ხშირად რეგისტრირდებით სხვადასხვა ვებ გვერდზე, ონლაინ სერვისებზე ან რაიმეს ყიდულობთ ინტერნეტის მეშვეობით. ამისათვის შექმენით სხვა ელ-ფოსტის მისამართი(ები), ეს მოგცემთ საშუალებას თქვენს ძირითად მისამართზე ნაკლები ”ნაგავი” მოვიდეს.
- **არ გადაამისამართოთ უცნობი წერილები** - თუ თქვენ უცნობისაგან მოგივიდათ წერილი, სადაც გთხოვენ გაავრცელოთ რაიმე ინფორმაცია და გადაუგზავნოთ თქვენს მეგობრებს, არ გააგზავნოთ რადგან ამ გზით სპამერს შეუძლია უფრო მეტი ელ-ფოსტის მისამართის გაგება.
- **გამოიყენეთ თქვენი ელ-ფოსტის პროგრამის ფილტრი** (Outlook, Thunderbird, The Bat, Live Mail), თქვენი სურვილის მიხედვით შეგიძლია შექმნათ წესები (rule) სადაც მიუთითებთ რის მიხედვით (From, Subject, Text), დაიბლოკოს არასასურველი წერილები ან გადაამისამართოთ სხვა ყუთში.
- **გამოიყენეთ ანტივირუსი** - ბევრ თანამედროვე ანტივირუსულ პროგრამებს გააჩნია ანტი-სპამ ფუქნცია.

ფიშინგი

ფიშინგი (ინგლისურად fishing - თევზაობა) — ინტერნეტ თაღლითობის დანაშაულებრივი ფორმა, რომლის მიზანია თაღლითური გზით მომხმარებელს გამოსძალოს პირადი საიდენტიფიკაციო მონაცემები, მაგალითად პაროლი, საკრედიტო ბარათის ან საბანკო ანაგარიშის ნომერი და სხვა კონფიდენციალური ინფორმაცია.

ფიშინგისას შენიღბული ინტერნეტ კომუნიკაციის საშუალებით ხდება მომხმარებლის შესახებ ისეთი ინფორმაციის მოპოვება, როგორიცაა მომხმარებლის სახელი, პაროლი, საკრედიტო ბარათის ან საბანკო ანგარიშის ნომერი. ეს მიიღწევა შემდეგი მეთოდებით: მასიური ელექტრონული წერილების დაგზავნით (წერილის ავტორებად იყენებენ ცნობილ ორგანიზაციებს და ბრენდებს), ასევე პირადული შეტყობინებებით სადაც იყენებენ ბანკის სახელს, მეილ სერვერების გამოყენებით და სოციალური ქსელების საშუალებებით. წერილში ხშირად არის ვებ გვერდის ბმული, რომლის ვიზუალური მხარე არ განსხვავდება ნამდვილისგან. შესაბამისად გაყალბებულ ვებ გვერდზე შეტანილი ინფორმაცია: მომხმარებლის სახელი, პაროლი, საკრედიტო ბარათის ან საბანკო ანგარიშის ნომერი ავტომატურად ხვდება ეგრედ წოდებული "ფიშერი"-ს ხელში.

<u>ფიშინგის ისტორია</u> - პირველი სტატია ფიშინგზე და ფიშინგის განხორციელების მეთოდებზე 1996 წელს დეტალურად იქნა აღწერილი ხაკერულ ჟურნალში "2600: The Hacker Quarterly".

<u>ახალი საფრთხეები</u> - დღევანდელ დღეს ფიშინგი ცდება უბრალოდ ინტერნეტ-თაღლითობას, ყალბი ვებ-გვერდების არსებობა გახდა თაღლითობის მრავალი მიმართულებიდან ერთ-ერთი ყველაზე აქტუალური და საგანგაშო. წერილი რომელიც თითქოს ბანკიდან არის გამოგზავნილი შეიძლება მომხმარებელს მოუწოდებდეს მითითებულ ნომერზე სავალდებულოდ დაკავშირებისაკენ, რათა მოგვარებული იქნას მის საბანკო ანგარიშზე არსებული პრობლემა. ამ მეთოდს ეწოდება ვიშინგი "ViShing" (ხმოვანი ფიშინგი). აღნიშნულ ნომერზე დარეკვის შემდეგ მომხმარებელი ხვდება ავტო მოპასუხესთან, რომელიც სთხოვს ინფორმაციას სხვადასხვა პირად მონაცემებზე, მაგალითად პინ-კოდი, ანგარიშის ნომრები, პაროლები და ა.შ. ასევე "ვიშერი"-ბი თავადაც რეკავენ მსხვერპლთან და არწმუნებენ მათ, რომ ისინი ოფიცილური ორგანიზაციიდან

არიან, რისთვისაც იყენებენ ყალბ სატელეფონო ნომრებს. საბოლოოდ კი მოიპოვებენ მომხმარებლის პირად ინფორმაციას.

აგრეთვე ძალიან პოპულარული ხდება SMS-ფიშინგი, ცნობილი როგორც სმიშინგი "SMiShing". თაღლითები თავის მსხვერპლს უგზავნიან SMS შეტყობინებას სადაც მოთავსებულია ფიშინგ საიტის ბმული. ბმულზე შესვლისთანავე მომხმარებელი ხდება ფიშერის მსხვერპლი და კარგავს პირად ინფორმაციას. SMS შეტყობინება

აგრეთვე შეიძლება მოუწოდებდეს მომხმარებელს მითითებულ ნომერზე დაუყონებლივ დაკავშირებას, რათა მისი პრობლემა იქნას გადაჭრილი.

ფიშინგ წერილი - ფიშინგ წერილი არის ელექტრონული ფოსტის მისამართზე მოსული ყალბი წერილი, რომელიც თითქოს გამოგზავნილია ბანკის, სერვის პროვაიდერის ან სხვა ცნობილი ბრენდ ორგანიზაციის მიერ, რომელთანაც მომხმარებელს აქვს შეხება, სადაც რაიმე მიზეზით სთხოვენ მას გაანახლოს თავისი პირადი მონაცემები მითითებული ბმულის საშუალებით. დასახელებული მიზეზი შეიძლება იყოს სხვადასხვაგვარი, მაგალითად ასეთი ყალბი წერილი შეიძლება იტყობინებოდეს, რომ მომხმარებლის ანგარიში იქნება შეჩერებული თუ არ მოხდა მისი პაროლის ან საკრედიტო ბარათის ინფორმაციის განახლება მოცემულ ვებ გევრდზე.

ფიშინგ წერილების დამახასიათებელი ნიშანია მაღალი ხარისხის გაყალბება. მაგალითად მიმღები იღებს წერილებს ბანკის ან სერვის პროვადერის ლოგოთი , რომელიც არის ორიგინალის ზუსტი ასლი. მომხმარებელი, რომელიც ვერ ხვდება ტყუილს გადადის არა ოფიციალურ ვებ გვერდზე არამედ ყალბზე, რომელიც ასევე ვიზუალურად ძალიან ჰგავს ცნობილი კომპანიის ვებ გვერდს მხოლოდ მისამართია განსხვავებული და ინფორმაციის შეგროვების შემდეგ შეიძლება მოხდეს მომხმარებლის გადამისამართება რეალურ ვებ-გვერდზე.

ფიშინგ ვებ გვერდი - როდესაც მომხმარებელი გადადის ყალბ ვებ გვერდზე და მითითებულ ველებში შეჰყავს თავისი მომხმარებლის სახელი, პაროლი და საბანკო რეკვიზიტები თაღლითებისთვის ხელმისაწვდომი ხდება მისი ელექტრონული ფოსტა ან თუნდაც ინტერნეტ ბანკინგის მონაცემები. აღსანიშნავია ისიც, რომ ყველა "ფიშერი" თავად

არ იყენებს მოპოვებულ ინფორმაციას, ისინი ყიდიან სხვა პირებზე რომლებსაც აქვთ კარგად შემუშავებული გეგმა თუ როგორ უნდა მოიპარონ ფული სხვისი ანგარიშებიდან.

როგორც წესი ფიშინგ ვებ გვერდი ფუნქციონირებს მხოლოდ რამდენიმე დღე (საშუალოდ 5-10), რადგან ანტი-ფიშინგ ფილტრები დროულად პოულობენ ახალ საშიშროებებს და ამის გამო ფიშერებს უწევთ სულ ახალ-ახალი ვებ გვერდების რეგისტრაცია ან არსებული სხვა ვებ გვერდების გატეხვა და მალულად ფიშერული საიტის განთავსება, მაგრამ ვებ გვერდების ვიზუალური მხარე მაინც უცვლელი რჩება და ძალიან ჰგავს კომპანიის რელურ ვებ-გვერდს.

როგორ ხორციელდება ფიშინგი - ფიშინგის ყველაზე ხშირ სამიზნეს წარმოადგენენ ბანკები, საფინანსო ორგანიზაციები, ელექტრონული აუქციონები და ინტერნეტ მაღაზიები. ვინაიდან თაღლითებს სურთ მოიპოვონ ინფორმაცია, რომელიც იძლევა ფულთან წვდომის საშუალებას. აგრეთვე პოპულარულია ელექტრნული ფოსტის მონაცემების მოპარვა, რათა შემდგომში გამოიყენონ სპამის და ვირუსების გასავრცელებლად.

ფიშერების კიდევ ერთი ხრიკი არის ბმულის URL მისამართები, რომელიც ძალიან ჰგავს ნამდვილი ვებ გევრდის სახელს და კარგი დაკვირვების გარეშე მომხმარებელმა შეიძლება ვერ შეამჩნიოს ასოების ცდომილება. ასევე ყალბი ვებ გვერდი შეიძლება იწყებოდეს IP მისამართით და გრძელდებოდეს ორგანიზაციის სახელით, ან მითითებული იყოს რაიმე სხვა ვებ გევრდი, რომელზეც მიბმულია სხვა ვებ გვერდის მისამართი.

პირადი მონაცემების მოპარვა არ არის ერთადერთი საფრთხე რომელიც მომხმარებელს შეიძლება შეემთხვეს. ფიშინგური ვებ გვერდი შეიძლება ასევე შეიცავდეს მავნე ან შპიონურ პროგრამას, ასე რომ თუ თქვენ არ გაქვთ არანაირი ანგარიში, რომელითაც შეიძლება თაღლითები დააინტერესდნენ, ეს იმას არ ნიშნავს რომ თქვენ ხართ უსაფრთხოდ, რადგან ფიშინგური შეტევების წარმატება დაფუძნებულია მომხმარებელთა გაუთვითცნობიერებაზე ან უყურადღებობაზე.

ფიშერები ხშირად იყენებენ გამოსახულებას ტექსტის ნაცვლად, რის შემდეგაც ანტიფიშინგის ფილტრებს ურთულდებათ მათი აღმოჩენა. მაგრამ სპეციალისტებმა შეიმუშავეს მეთოდი რომელიც მეილში მოსულ გამოსახულებას ადარებს ანტიფიშინგის ბაზას და ამის შემდეგ ბლოკავს მას თუ მეილში აღმოაჩენს ფიშინგის ელემენტებს.

ფიშინგთან ბრძოლა - რამდენიმე წლის წინ ფიშინგთან საბრძოლველად შეიქმნა ანტი-ფიშინგის სამუშაო ჯგუფი (Anti-Phishing Working Group - APWG), სადაც გაერთიანებულები არიან ფიშინგის სამიზნე კომპანიები, ასევე უსაფრხოების პროგრამული უზრუნველყოფის მწარმოებლები და კიბერ დანაშაულთან მებრძოლი კომპანიები, ჯამში 2500 კომპანიაზე მეტი. APWG-ს წევრები ატყობინებენ ერთმანეთს ახალი ფიშერული შეტევების შესახებ და ერთად ზრუნავენ ამ საკითხთან დაკავშირებით საზოგადოების განათლებაზე.

მავნე პროგრამების სახეობები ვირუსები და "ტროას ცხენი"-ს სახელით ცნობილი პროგრამები

პროგრამებს Malware, Spyware და Riskware კატეგორიაში მიიჩნევენ როგორც განსაკუთრებით დიდი საფრთხის გამომწვევად. ამ კატეგორიაში არსებული პროგრამები სხვა საფრთხეებთან ერთად შეიძლება შეიცავდეს ვირუსების, ვორმების (Worm) და ტროიანების (Trojan) სხვადასხვა სახეობებს. ამ საზიანო პროგრამების ეფექტი შესაძლოა მოიცავდეს მომხმარებლის კონფიდენციალური ინფორმაციის მიღებას, კომპიუტერის უკანონო მიზნებისათვის გამოყენებას ან მის მწყობრიდან გამოყვანას.

მალვეარი (Malware)

მალვეარი არის მავნე პროგრამა, რომელიც თავდამსხმელების მიერ გამოიყენება კომპიუტერის ფუნქციონირებისათვის ხელის შესაშლელად და პირად ინფორმაციასთან წვდომის მოსაპოვებლად. ის შეიძლება იყოს კოდი, სკრიპტი ან სხვა პროგრამა. მალვეარი არის ტერმინი, რომელიც გულისხმობს ინტრავირუსულ პროგრამას.

მალვეარი მოიცავს კომპიუტერულ ვირუსებს, რენსომვეარს, ვორმებს, ტროიანებს, რუთკიტებს, კეილოგერებს, სპაივეარს, ედვეარს და საზიანიო BHO-ს, აგრეთვე სხვა საზიანო პროგრამებს. მალვეარის ძირითადი საფრთხეები არის ვორმები ან ტროიანები და არა ვირუსები. მალვეარი არ არის იგივე რაც დეფექტიური პროგრამა, რომელიც არის პროგრამა, რომელსაც აქვს ლეგიტიმური მიზანი, მაგრამ შეიცავს საზიანო ბაგებს, რომლებიც არ იქნა შესწორებული ბაზარზე გამოშვებამდე. ზოგიეთრი მალვეარი შეიძლება მოდიოდეს კომპანიის ოფიციალური ვებ გვერდიდან. ამის მაგალითია პროგრამა, რომელსაც კომპანიები იყენებენ მარკეტინგული სტატისტიკის მოსაპოვებლად და მომხმარებელთა კვლევის დროს უსაფრთხო მონაცემებს იღებენ მომხმარებლის კომპიუტერიდან.

მალვეარის არსებობამ საჭირო გახადა ისეთი დამცავი პროგრამების შექმნა, როგორიცაა ანტივირუსები, ანტიმალვეარები და ფაიერვოლი. თითოეული ეს პროგრამა აქტიურად არის გამოყენებული კერძო მომხმარებლების მიერ მათი კომპიუტერების დასაცავად და ნებადაურთავი წვდომისაგან თავის ასარიდებლად.

კომპიუტერული ვირუსი

კომპიუტერული ვირუსი არის კომპიუტერული პროგრამა, რომელსაც აქვს გამრავლების და ერთი კომპიუტერიდან მეორეზე გავრცელების უნარი. ტერმინი ვირუსი არის ფართოდ გავრცელებული, თუმცა ხშირად იგი არასწორად გამოიყენება და მალვეარს არასწორად უწოდებენ ვირუსს.

მალვეარი მოიცავს კომპიუტერულ ვირუსებს, კომპიუტერულ ვორმებს, რენსომვეარს, ტროიანებს. ისეთი მალვეარი, როგორიცაა ტროიანები და ვორმები ხშირად ეშლებათ და ვირუსებად მოიხსენიებენ. მათ შორის კი ტექნიკური განსხვავებაა: ვორმს შეუძლია გამოიყენოს საფრთხეები და ავტომატურად გამრავლდეს კომპიუტერში და სხვა ქსელში არსებულ კომპიუტერებშიც. ხოლო ტროიანი არის პროგრამა, რომელიც თითქოს უსაფრთხოა, მაგრამ მალავს საზიანო ფუნქციებს. ვორმებსაც და ტროიანებსაც შეუძლიათ ზიანი მიაყენონ კომპიუტერული სისტემის მონაცემებს ან კომპიუტერული სისტემის მუშაობას.

კომპიუტერული ვირუსი არის საზიანო პროგრამა, რომელიც მალულად აღწევს კომპიუტერულ სისტემებში, პროგრამებში ან ფაილებში და შეუძლია თავისი თავის კოპირება და ამგვარად კომპიუტერის დაინფიცირება. ასეთი პროგრამები ხშირად კომპიუტერს რაიმე სახის ზიანს აყენებენ, მათ კომპიუტერის პროგრამულ კოდში საზიანო ცვლილებები შეაქვთ, რომელთა შედეგად ზიანდება კომპიუტერი ან მომხმარებლის მონაცემები.

ადრე ვირუსებს წერდნენ გართობის და ინტერესის მიზნით, მაგრამ დღეს მათ უკვე მიზნობრივი საზიანო ფუნქციები ახასიათებთ, როგორიცაა მაგალითად კომპიუტერის დაზიანება ან მონაცემების მოპარვა, მომხმარებლის კომპიუტერის კონტროლი. ხშირად ტერმინით, კომპიუტერული ვირუსი, ყველა ტიპის მავნე პროგრამებს აღნიშნავენ. არადა ასეთი პროგრამები სხვადასხვა ფუნქციების მატარებელია და შესაბამისად, სხვადასხვა კატეგორიებად იყოფა, რომლებსაც ქვემოთ განვიხილავთ.

კომპიუტერულ სისტემაში ზოგიერთი ვირუსის არსებობა ადვილად შესამჩნევია მომხმარებლისათვის, მაგრამ ბევრი ვირუსი არ აჩენს არანაირ სიმპტომს და მომხმარებელს ეჭვიც არ უჩნდება, რომ კომპიუტერი დავირუსებულია. ზოგიერთი ვირუსი გამრავლების გარდა არაფერს არ აკეთებს.

ვიდრე ვირუსი მოახერხებს კომპიუტერის დააინფიცირებს პირველ რიგში ის უნდა მოხვდეს კომპიუტერში. არსებობს ვირუსის გავრცელების სხვადასხვა საშუალებები:

- მოძრავი მედია საშუალებები - ინფორმაციის მატარებელი მოწყობილობები (კომპაქტ-დისკები CD, USB მოწყობილობები)
- ინტერნეტი - ელექტრონული-ფოსტა, საზიანო კოდის შემცველი ვებ-გვერდები, ინტერნეტიდან გადმოწერილი პროგრამები ან ფაილები, მესენჯერ პრგრამები IM.
- ქსელი - ლოკალურ ქსელში საზიარო ფოლდერები Shared Folder, საზოგადო ქსელები Public Networks.

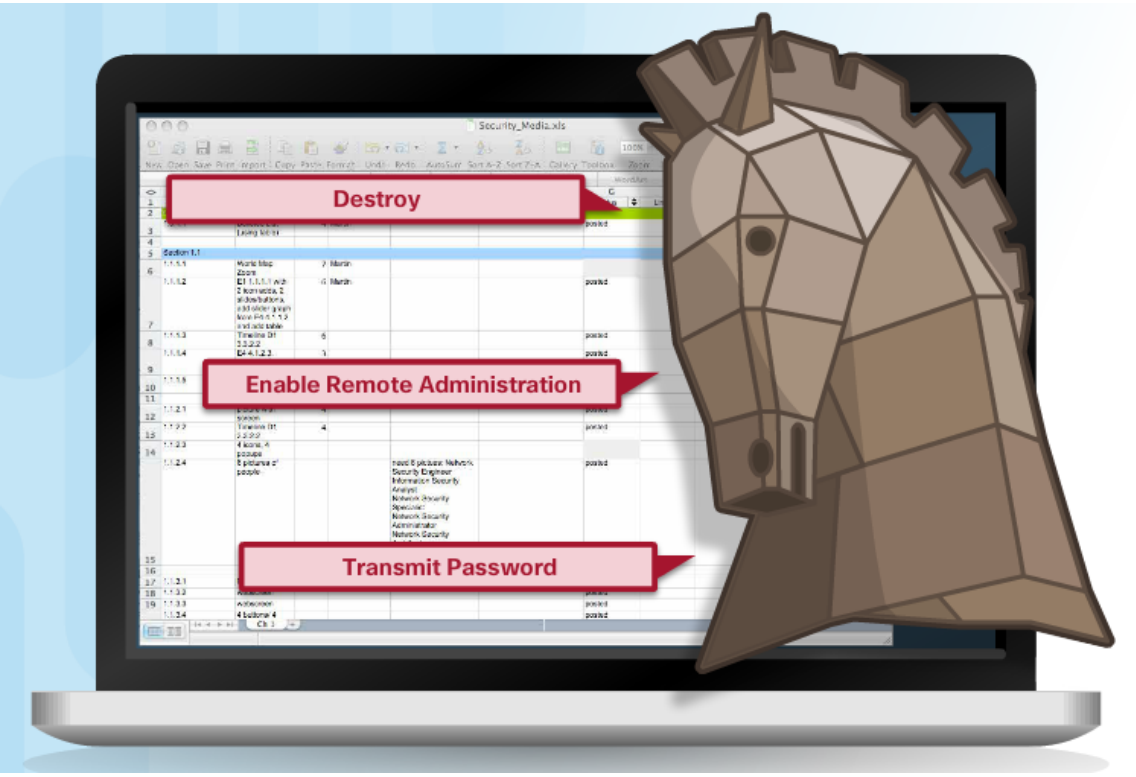
პიროვნებამ, რომელმაც დაწერა ვირუსი უნდა მოძებნოს გზა კომპიუტერში შემოსაღწევად. მან უნდა მოახერხოს რაიმე საიდუმლო ხერხით შემოუშვას ვირუსი თქვენს სისტემაში ან თქვენ მიგიტყუოთ და მოახერხოს რომ თქვენ თვითონ შეუშვათ ვირუსი თქვენს სისტემაში.

არსებობს მრავალი გზა, რითაც თავდამსხმელს შეუძლია ამის გაკეთება, მაგალითად:

- იპოვნოს დაუცველი კომპიუტერული სისტემა და შემოგიგზავნოთ ვირუსი.
- დაცულ კომპიუტერულ სისტემაში იპოვნოს ხარვეზი (vulnerability) და გამოიყენოს ეს ვირუსის შესაღწევად.
- მოატყუოს მომხმარებელი და დააჯეროს მას, რომ საზიანო ფაილი მისთვის სასურველი ფაილია.

ტროიანი (Trojan Horse)

ტროიან ჰორსი, იგივე ტროიანი არის მალვეარი, რომელსაც აქვს უნარი შეასრულოს სასურველი ფუნქცია და გაამარტივოს მომხმარებლის კომპიუტერულ მონაცემებზე წვდომა. ტროიანის მიზანი არაა სხვა ფაილებში შეღწევა ვირუსის მსგავსად. ტროიან ჰორსებს შეუძლიათ მოიპარონ ინფორმაცია ან ავნონ კომპიუტერულ სისტემას. ტროიანებმა შეიძლება გამოიყენონ დრაივ დაუნლოადები ან დააინსტალირონ თავი ონლაინ თამაშების ან ინტერნეტ აპლიკაციების მეშვეობით იმისათვის, რომ მიაღწიონ სასურველ კომპიუტერამდე.



სურ. 3.3 - ტროიან ჰორსის სახეები

ტერმინი ტროიან ჰორსი არის წარმოქმნილი ბერძნული მითოლოგიიდან ტროას ცხენის ლეგენდის შესახებ. ტროიანი თავს აჩვენებს მომხმარებელს თითქოს იგი არის სრულიად სანდო და უვნებელი. აგრეთვე, მომხმარებელს რაღაც ფორმით სთავაზობს საჩუქრებს, რათა მსხვერპლს მოტყუებით დააინსტალირებინოს თავი კომპიუტერულ სისტემაში.

Trojan-Spy - ტროიან შპიონი, მომხმარებლის კომპიუტერში მალულად აინსტალირებს პროგრამებს ისეთებს, როგორიცაა მაგალითად keyloggers, რის საშუალებითაც მესამე პირს შეუძლია მომხმარებლის მიერ კლავიატურაზე აკრეფილი ინფორმაცია წაიკითხოს.

Trojan-PSW- იპარავს პაროლებს და სხვა მნიშვნელოვან ინფორმაციას. მას აგრეთვე შეუძლია სხვა საზიანო პროგრამების დაყენებაც.

Trojan-Downloader - ინტენეტის საშუალებით ფარულად იწერს საზიანო ფაილებს მოშორებული სერვერიდან და შემდგომ ავტომატურად აინსტალირებს მომხმარებლის კომპიუტერზე.

Trojan-Dropper - შეიცავს ერთ ან რამდენიმე საზიანო პროგრამას, რომელსაც ის ფარულად აინსტალირებს და გამოიყენებს მომხმარებლის კომპიუტერზე.

Trojan-Proxy - საშუალებას აძლევს მომხმარებლის კომპიუტერის მეშვეობით არავტორიზებულმა პირებმა ანონიმურად ისარგებლონ ინტერნეტით.

Trojan-Dialer - მომხმარებლის კომპიუტერს სატელეფონო ხაზის მეშვეობით აკავშირებს ინტერნეტ ქსელთან. მას აგრეთვე შეუძლია მომხმარებელი გადაამისამართოს არასასურველ ვებ გვერდებზე.

რუთკიტი (Rootkit)

რუთკიტი არის პროგრამა რომელიც ცდილობს თავისი არსებობის დამალვას კომპიუტერის უსაფრთხოების პროგრამებისგან თავის აცილებით. კომპიუტერში შეღწევის შემდეგ საშუალებას აძლევს დისტანციურ მომხმარებელს საიდუმლოდ აკონტროლოს კომპიუტერის ოპერაციული სისტემა.

რუთკიტის აღმოჩენა არის საკმაოდ რთული, რადგან მას აქვს უნარი გაუმკლავდეს იმ პროგრამას, რომელიც მის აღმოსაჩენად უნდა იქნას გამოყენებული.

კომპიუტერული სისტემის რუთკიტისგან გაწმენდა საკმაოდ რთული საქმეა, ხშირად საჭირო ხდება ხელახალი ინსტალაცია, ვინაიდან მხოლოდ ამ გზით ხდება შესაძლებელი მისგან თავის დაღწევა.

ბექდორი (Backdoor)

ბექდორი არის ჯაშუშური პროგრამა რომელიც გამოიყენება არა მხოლოდ ინფორმაციის მიტაცებისთვის, არამედ კომპიუტერის უკანონოდ სამართავადაც. აქვს ცალკე არსებული ადმინისტრაციული შესაძლებლობა, რომელიც თავს არიდებს სტანდარტულ უსაფრთხოების მექანიზმებს კომპიუტერული პროგრამების, კომპიუტერის ან ქსელის მალულად სამართავად.

ვორმი (Worm)

კომპიუტერული ჭია არის საზიანო პროგრამა, რომელიც იყენებს კომპიუტერს და ქსელის შესაძლებლობებს, რათა ავტომატურად გავრცელდეს სხვა კომპიუტერებზე. ვირუსისგან განსხვავებით ის არ საჭიროებს, კომპიუტერში არსებულ რომელიმე პროგრამაზე იყოს დამოკიდებული. კომპიუტერული ჭია თითქმის ყოველთვის იწვევს მინიმალურ ზიანს ქსელში, თუნდაც შეუძლია მოიხმაროს მომხმარებლის ინტერნეტ სიჩქარე, ხოლო ვირუსები თითქმის ყოველთვის აზიანებს ან ცვლის ფაილებს და მონაცემებს კომპიუტერში.

ქსელური-ჭია (Net-Worm) - მრავლდება თავისი თავის სრულიად დამოუკიდებელი კოპიების ქსელში გავრცელებით.

P2P-ჭია (P2P-Worm) - ვრცელდება P2P პროგრამების და ქსელის (Emule, KaZaa, Imesh, Torrent) მეშვეობით, ძირითადად მაცდუნებელი ფაილის სახით.

ელ-ფოსტის ჭია (Email-Worm) - ვრცელდება ელექტრონული ფოსტის საშუალებით, ძირითადად მიბმული ფაილის (attachment) სახით.

IRC-ჭია (IRC-Worm) - ვრცელდება ინტერნეტ ჩატის ქსელის მეშვეობით.

IM-ჭია (IM-Worm) - ვრცელდება სწრაფი შეტყობინებების პროგრამების და ქსელის (IM, ICQ, Skype, Yahoo და MSN Messenger) მეშვეობით.

Bluetooth-ჭია (Bluetooth-Worm) ვრცელდება ბლუთუსის მოწყობილობების მეშვეობით.

უსაფრთხოების პოლიტიკა

სისტემის უსაფრთხოების პოლიტიკა წააგვას სახელმწიფოს საგარეო პოლიტიკას: ის განსაზღვრავს მიზნებსა და ამოცანებს, როდესაც სახელმწიფოს ადანაშაულებენ საგარეო პოლიტიკის არათანმიმდევრულობაში, ეს ხდება იმიტომ, რომ მის მოქმედებაში არ არის ლოგიკა და არ არის საერთო სტრატეგია. ზუსტად ასევე ხდება უსაფრთხოების პოლიტიკის არ არსებობის დროს, ციფრულ სისტემაში უკუქმედების ზომები იქნება მოუწესრიგებელი. პოლიტიკა – ეს არის ხერხი უზრუნველყოთ საერთო ურთიერთკავშირი.

კარგი პოლიტიკა ფორმირდება, როგორც პასუხი საფრთხეზე. თუ საფრთხე არ არსებობს მაშინ არ არის არც პოლიტიკაც: ყველას შეუძლია აკეთოს ყველაფერი. ამერიკის შეერთებულ შტატები საჭიროებს საგარეო პოლიტიკას, თუ მხედველობაში მივიღებთ საფრთხეებს, რომელიც ემუქრება სხვა სახელმწიფოებიდან. მაგრამ შტატი პენსილვანია არ საჭიროებს არანაირ საგარეო პოლიტიკას, იმიტომ რომ დანარჩენი შტატები არ წარმოადგენენ საფრთხეს მისთვის. ასევეა უსაფრთხოების პოლიტიკაშიც - ის აუცილებელია, ამიტომ საფრთხეების მოდელირება არ მთავრდება არასდროს ცარიელი ფურცლით. უსაფრთხოების პოლიტიკა განსაზღვრავს ჩარჩოებს, რომელშიც ხორციელდება უკუქმედების ზომების შერჩევა და რეალიზაცია.

არ არის საჭირო იმის მტკიცება, რომ ყოველ ორგანიზაციას სჭირდება თავისი კომპიუტერული ქსელისთვის უსაფრთხოების პოლიტიკა. პოლიტიკამ უნდა მოხაზოს პასუხისმგებლობის საზღვრები, განსაზღვროს თუ რა არის უსაფრთხოების პოლიტიკის საფუძველი და თუ რატომ არის ის მაინცდამინც საფუძველი. უკანასკნელი აღნიშვნა ძალიან მნიშვნელოვანია, რადგანაც შემთხვევითი პოლიტიკა ”ჩამოშვებული ზემოდან” განმარტებების გარეშე, საბოლოოდ მაინც იქნება იგნორირებული, რადგან უფრო სავარაუდოა, რომ თანამშრომლები გაყვებიან გასაგებ, მოკლე, ლოგიკურ და თანმიმდევრულ პოლიტიკას.

უსაფრთხოების პოლიტიკა – არის ის რასაც თქვენ განსაზღვრავთ, რა უკუქმედების ზომებს მიმართავთ. გჭირდებათ თუ არა ბრანდმაუერი? როგორ უნდა დააკონფიგურიროთ, საკმარისია თუ არა სისტემიში წვდომისას მარტო პაროლის გამოყენება, შესაძლებელია თუ არა მომხმარებლებს მიეცეთ ნება თავიანთი ბრაუზერიდან ვიდეოს ყურება.

ნებისმიერ შემთხვევაში უსაფრთხოების პოლიტიკა პირველ რიგში უნდა პასუხობდეს კითხვებზე ”რატომ”? და არა ”როგორ”.

”როგორ” ეს არის კონტრზომის ტაქტიკა. რთულია შეარჩიო სწორი პოლიტიკა, მაგრამ უფრო რთულია განსაზღვრო უკუქმედების ზომების კომპლექსი, რომლებიც მის რეალიზებას განაპირობებენ.

ქსელების უსაფრთხოების უზრუნველსაყოფად საჭიროა გამუდმებული მუშაობა და ყურადღება. ამ მუშაობაში იგულისხმება რომ წინასწარ უნდა იქნას შესწავლილი ბოროტმოქმედების მიერ ყველა შესაძლო ქმედება, დაცვითი მექანიზმების გზების ძიება და მომხმარებლების პერმანენტული განათლება. თუ მაინც მოხდა სისტემაში შემოჭრა უსაფრთხოების ადმინისტრატორმა უნდა შეძლოს დაცვითი სისტემის ხარვეზის აღმოჩენა, ხარვეზის მიზეზი და შემოჭრის გზა.

უსაფრთხოების სისტემის პოლიტიკის შედგენისას ადმინისტრატორმა პირველ რიგში უნდა ჩაატაროს რესურსების ინვენტარიზაცია, რომლის დაცვაც არის დაგეგმილი; იდენტიფიცირება უნდა გაუკეთდეს ყველა მომხმარებელს რომლებიც მუშაობენ აღნიშნულ რესურსებთან. უნდა გაკეთდეს ანალიზი თუ რომელი რესურსთან რა საფრთხე შეიძლება იყოს მოსალოდნელი. ყველა ამ ინფორმაციის ფლობის შემდეგ შესაძლებელია აიგოს უსაფრთხოების პოლიტიკა, რომელიც აუცილებელი გახდება ყველა მომხმარებლისათვის.

უსაფრთხოების პოლიტიკის აგება – ეს არ არის ჩვეულებრივი წესი, ის ბევრისთვის გაუგებარია. ის უნდა იყოს წარმოდგენილი სერიოზული დოკუმენტის სახით. იმისათვის რომ გამუდმებით შევახსენოთ მომხმარებლებს უსაფრთხოების წესები, დოკუმენტის ასლები უნდა იქნეს დარიგებული ყველა ოფისში, რათა ეს წესების თვალწინ ედოს ყველა თანამშრომელს.

კარგი უსაფრთხოების პოლიტიკის აგება ითვალისწინებს რამოდენიმე ელემენტს ზოგიერთი მათგანი მოცემულია ქვემოთ:

1. **რისკის შეფასება**. უნდა ვიცოდეთ თუ რას ვიცავთ დავისგან. ქსელში უნდა გამოიკვეთოს ღირებულებები დაპრობლემების შესაძლო წარმომქმნელები;

2. **პასუხისმგებლობა.** აუცილებელია მიეთითოს პასუხისმგებლები, რომლებიც ამა თუ იმ გზით პასუხს აგებენ უსაფრთხოებაზე, დაწყებული აღრიცხვითი ჩანაწერების გაკეთებიდან დამთავრებული დარღვევების გამოკვლევით;
3. **ქსელური რესურსები გამოყენების წესები.** პოლიტიკაში პირდაპირ უნდა იყოს მითითებული. რომ მომხმარებლებს არა აქვთ უფლება: გამოიყენონ ინფორმაცია არადანიშნულებით, გამოიყენონ ქსელი პირადი სარგებლობისთვის, აგრეთვე გამიზნულად მიაყენონ ზიანი ქსელს ან იქ განთავსებულ ინფორმაციას;
4. **იურიდიული ასპექტები.** აუცილებელია კონსულტირება იურისტთან, რადგან უნდა გაირკვეს ყველა კითხვა, რომელსაც შეიძლება კავშირი ქონდეს ქსელში შენახულ ან წარმოქმნილ ინფორმაციასთან და დართული უნდა იქნეს დოკუმენტებში, რომელიც ეხება უსაფრთხოების უზრუნველყოფას;
5. **სისტემის აღდგენის პროცედურები.** მითითებული უნდა იყოს, თუ რა უნდა იქნას გაკეთებული სისტემის დარღვევის შემთხვევაში და რა მოქმედებები უნდა ჩატარდეს იმათ მიმართ, ვინც გახდა მიზეზი ამის გამოიწვევისა. თუ საკმაოდ დიდხანს ვიმუშავებთ საფრთხეების მოდელირებაზე, გასაგები გახდება რომ ცნებას ”უსაფრთხოების სისტემა” აქვს განსხვავებული მნიშვნელობა იმისდა მიხედვით თუ რა სიტუაციასთან გვაქვს საქმე.

რამოდენიმე მაგალითი:

- კომპიუტერები, გამოყენებული საქმიან სფეროში უნდა იყვნენ დაცული ჰაკერებისგან, ქურდებისგან და სამრეწველო კონკურენტებისაგან. სამხედრო კომპიუტერები უნდა იყვნენ საიმედოდ დაცული იგივე საფრთხეებისაგან და აგრეთვე მტრული სამხედრო ძალების შეღწევისგან. ზოგიერთი კომპიუტერები ემსახურებიან სატელეფონო ქსელებს და ისინიც უნდა იყვნენ დაცული სამხედრო მოწინააღმდეგებისაგან.
- მრავალი საქალაქო სატრანსპორტო სისტემები უცხოეთში იყენებენ გამშვებ ბარათებს ნაღდი ფულის მაგივრად, მსგავსად ამისა გამოიყენება სატელეფონო ბარათებიც და სხვა. მსგავსი სისტემები აუცილებელია იყვნენ დაზღვეულები გაყალბებებისაგან. რა

თქმა უნდა ეს არ არის პრობლემა როდესაც ყალბის დამზადება გაცილებით ძვირი ჯდება.

- პროგრამები რომლებიც იცავენ ელ-ფოსტას უნდა უზრუნველყონ კორესპოდენციის დაცვა ყველა ნებისმიერი პლისტრაციისა და ცვლილების მცდელობისაგან. რასაკვირველია მრავალ შემთხვევაში პროგრამული საშუალებებით შეუძლებელია უსაფრთხო გავხადოთ სისტემა იმ მრავალი მანიპულაციებისგან, როგორიცაა: ტროას ცხენი კომპიუტერში, ვიდეოკამერა, და სხვა.

ხრიკი მდგომარეობს იმაში, რომ შევქმნათ სისტემა, რეალური საფრთხეების გათვალისწინებით და არ გამოვიყენოთ უსაფრთხოების ტექნოლოგიები ყველა რიგრიგობით, იმის იმედით რომ ამით რამე მაინც გამოგვივა. ამისათვის აუცილებელია შევიმუშაოთ უსაფრთხოების პოლიტიკა, რომელიც დამყარებული უნდა იყოს საფრთხეების ანალიზზე და შემდგომ შევქმნათ დაცვის მექანიზმები. რომლებიც რეალიზებას გაუკეთებენ ამ პოლიტიკას და წინ აღუდგებიან საფრთხეებს.

ტესტები თვითშემოწმებისათვის:

1. ჩამოთვლილთაგან რომელია უსაფრთხოების სერვისი:

ა. ჩანაწერების კეშირება

ბ. კონფლიქტების რეგულირება

გ. შიფრაცია

დ. იდენტიფიკაცია და აუთენტიფიკაცია

ე. მთლიანობის კონტროლი

2. ავტორიზებული ადრესატის მიერ იმ ფაქტის დადგენას, რომ მიღებული შეტყობინება ნამდვილად გამოგზავნილია ავტორიზებული მომხმარებლის მიერ, ეწოდება:

ა. აუთენტიფიკაცია

ბ. ინფორმაციის მთლიანობის კონტროლი

გ. იდენტიფიკაცია

დ. ავტორიზაცია

3. რა არის იდენტიფიკაცია?

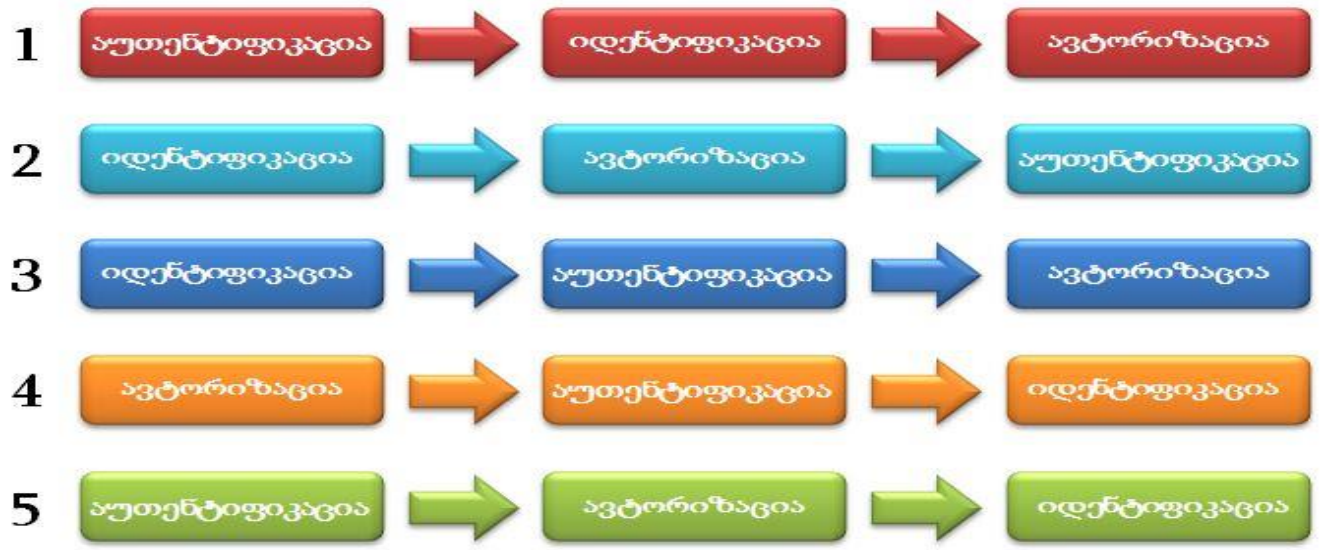
ა. სუბიექტებზე უნიკალური იდენტიფიკატორების მინიჭება და მათი შედარება შესაძლო იდენტიფიკატორების ვარიანტებთან

ბ. სუბიექტის მიერ შესრულებული ოპერაციების შემოწმება

გ. სუბიექტის მიერ შესრულებული ოპერაციების აღრიცხვა

დ. ესაა შემოწმება სუბიექტის მიერ წარმოდგენილი იდენტიფიკატორის კუთვნილებისა მის წარმომდგენ სუბიექტთან.

4. სისტემაში მომხმარებლის რეგისტრაცია მოიცავს სამ ურთიერთდაკავშირებულ, თანმიმდევრულად შესასრულებელ პროცედურას (ჩამოთვლილთაგან აირჩიეთ სწორი თანმიმდევრობა):



ა. 1

ბ. 5

გ. 4

დ. 2

ე. 3

5. რა არის აუთენტიკაცია?

ა. შემოწმება სუბიექტის მიერ წარმოდგენილი იდენტიფიკატორის კუთვნილებისა მის წარმომდგენ სუბიექტთან და მისი უტყუარობის დადასტურება

ბ. ესაა სუბიექტებზე უნიკალური იდენტიფიკატორების მინიჭება და მათი შედარება შესაძლო იდენტიფიკატორების ვარიანტებთან

გ. სუბიექტის მიერ შესრულებული ოპერაციების შემოწმება

დ. სუბიექტის მიერ შესრულებული ოპერაციების აღრიცხვა

6. რას შეისწავლის ინფორმაციული უსაფრთხოება?

ა. ქონების დაცვის მექანიზმებს

ბ. მატერიალურ-ტექნიკური რესურსების დაცვას

გ. ინფორმაციის კონფიდენციალობის, მთლიანობის და ხელმისაწვდომობის შენარჩუნებას და დაცვას

დ. ფინანსური რესურსების დაცვის მექანიზმებს

7. ინფორმაციული სისტემა, ინფორმაციული უსაფრთხოების თვალსაზრისით წყვეტს შემდეგ ძირითად ამოცანებს:

ა. ელექტრონული სახით წარმოდგენილი ინფორმაციის იურიდიულ მნიშვნელობას და კლიენტის ქმედებების კონფიდენციალობას

ბ. ინფორმაციის კონფიდენციალობა, მთლიანობა, სარწმუნოობა

გ. ინფორმაციასთან ხელმისაწვდომობა

დ. ყველა პასუხი სწორია

8. ჩამოთვლილთაგან რომელი სამი ცნება მიეკუთვნება დაცული ინფორმაციის თვისებებს?

ა. კონფიდენციალურობა

ბ. მოხერხებულობა

გ. მთლიანობა

დ. ადეკვატურობა

ე. ხელმისაწვდომობა

9. ყველაზე ხშირ, საშიშ და ზარალის მიმყენებელ საფრთხედ მიიჩნევა:

ა. პერსონალის უნებლიე შეცდომები

ბ. ქურდობები და გაყალბებები

გ. ტექნიკური საფრთხეები და ვირუსები

დ. გარემო პირობები

10. ინფორმაციული სისტემისათვის საფრთხე ეს არის:

ა. ზემოქმედებების რეალიზება მონაცემებზე, რომელიც იწვევს კონფიდენციალობის დარღვევას

ბ. ზემოქმედებების რეალიზება მონაცემებზე, რომელიც იწვევს მთლიანობის დარღვევას

გ. ზემოქმედებების რეალიზება მონაცემებზე, რომელიც იწვევს წვდომის დარღვევას

დ. ყველა პასუხი სწორია

11. ქვემოთ ჩამოთვლილთაგან რომელი მიეკუთვნება ტექნიკურ საფრთხეებს და მიზეზებს:

ა. აუთენტიკაცია, ანტი-სნიფერი, კრიპტოგრაფია

ბ. ლოკალური ქსელის ტრაფიკის გამჟღავნება-გაყალბება, ლოკალური ქსელის უწესრიგობა

გ. ინფორმაციულ სისტემასთან უნებართვო ან შეუსაბამო წვდომა

დ. მონაცემებისა და პროგრამების არაავტორიზებული მოდიფიკაცია, მონაცემების გამჟღავნება

12. ქვემოთ ჩამოთვლილი ტროიანული პროგრამებიდან რომელი გამოიყენება დაშორებული მართვისათვის:

ა. Backdoor

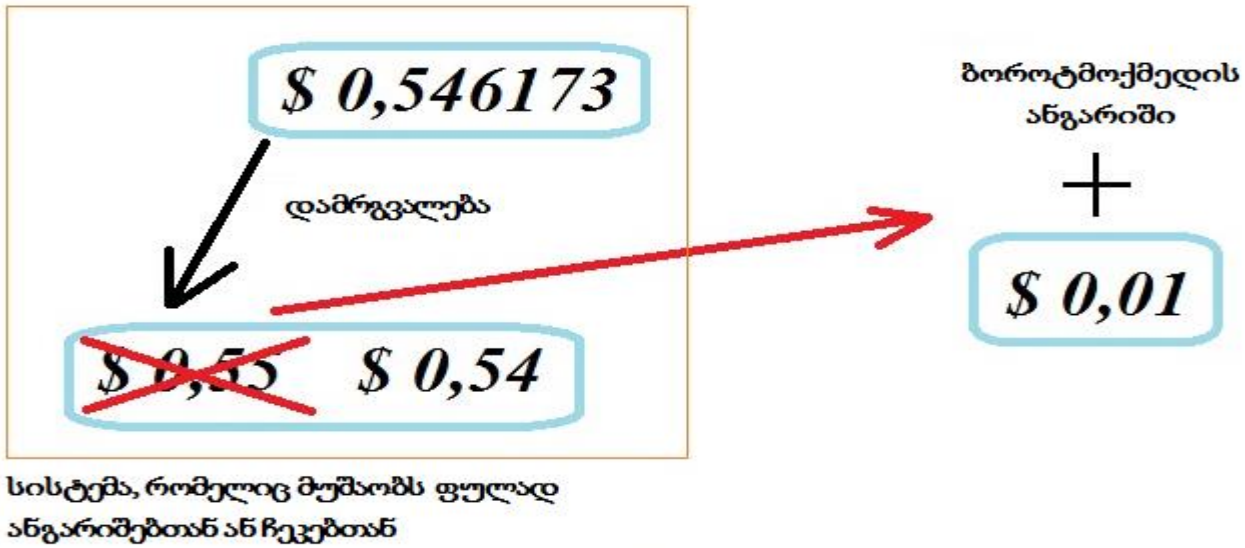
ბ. Trojan-Dropper

გ. Trojan-Clicker

დ. Trojan-Downloader

ე. Trojan-PSW

13. რა ჰქვია სურათზე გამოსახულ შეტევის ტიპს:



ა. SQL-ინექცია

ბ. Salami Attack

გ. Man-in-the-middle (შუა კაცის შეტევა)

დ. traffic Redirection (ტრაფიკის გადამისამართება)

ე. Buffer overflow (ბუფერის გადავსება)

14. ჩამოთვლილთაგან რომელი ტროიანული პროგრამა გამოიყენება პაროლების მოსაპარად:

ა. Trojan-Proxy

ბ. Trojan-Clicker

გ. Trojan-Downloader

დ. Trojan-PSW

ე. Trojan-Dropper

15. ქვემოთ ჩამოთვლილთაგან, რომელი ტროიანული პროგრამა გამოიყენება ინტერნეტ-რესურსებთან არასანქცირებული მიმართვების ორგანიზებისათვის:

ა. Trojan-Dropper

ბ. Trojan-Downloader

გ. Trojan-Clicker

დ. Trojan-PSW

ე. Trojan-Spy

16. . . . . პროგრამები ფარულად ახორციელებენ ანონიმურ წვდომას სხვადასხვა ინტერნეტ-რესურსებთან და როგორც წესი გამოიყენებიან სპამის დასაგზავნად. (ჩასვით გამოტოვებული სიტყვები):

ა. Trojan-Proxy

ბ. Trojan-Downloader

გ. Trojan-PSW

დ. Trojan-Dropper

ე. Trojan-Spy

17. ქვემოთ ჩამოთვლილთაგან, რომელი ტროიანული პროგრამები გამოიყენება შეტევის შესახებ გაფრთხილებისთვის:

ა. Trojan-Proxy

ბ. Trojan-PSW

გ. Trojan-Notifier

დ. Trojan-Spy

ე. Trojan-Dropper

18. განსაზღვრეთ შეტევის ტიპი, რეალიზაციის მოცემული მექანიზმით: „დიდი რაოდენობით ქსელური პაკეტების გადაგზავნა კონკრეტული ჰოსტისათვის":

ა. დაუცველი ადგილების მოძებნა

ბ. მომსახურებაზე უარის თქმა

გ. ქსელის ტოპოლოგიის ანალიზი

დ. პაროლებზე შეტევა

19. პაკეტების „სნიფერი" ეს არის:

ა. გამოყენებითი პროგრამა, რომელსაც იყენებენ ტრაფიკის ანალიზისა და დაზიანების დიაგნოსტიკისათვის

ბ. გამოყენებითი პროგრამა, რომელიც ახორციელებს ყველა პაკეტის გამოჭერას, რომლებიც გადაიცემა რომელიმე განსაზღვრულ დომენში

გ. გამოყენებითი პროგრამა, რომელსაც მწყობრიდან გამოყავს სისტემა

დ. პატარა პროგრამა, რომელიც ამოწმებს სისტემაში არსებულ ყველა პაროლს და უგზავნის მესამე პირს.

20. შეტევები გამოყენებით დონეზე არის:

ა. შეტევები, რომელსაც ჰაკერი ახორციელებს კორპორაციის შიგნით ან გარეთ, როგორც სანქცირებული მომხმარებელი

ბ. გამოყენებითი პროგრამა, რომელიც ახორციელებს ყველა პაკეტის გამოჭერას

გ. პორტების გადამისამართება

დ. შეტევები, რომელიც ხორციელდება სერვერის პროგრამული უზრუნველყოფის კარგად ცნობილი ნაკლოვანებების გამოყენებით

21. მიუთითეთ გამოტოვებული ეტაპების სწორი თანმიმდევრობა შეტევის შესრულებისას

<u>დანერგვა</u> - მსხვერპლის სისტემაში შესაღწევად სხვადასხვა დაუცველი ადგილების გამოყენება;

<u>შეტევის რეალიზაცია</u> - ინფორმაციის მოდიფიცირება, გადაადგილება, წაშლა, სისტემური რესურსების ბოროტად გამოყენება;

<u>უფლებამოსილების გადამეტება</u> - მომხმარებლის უფლების დონის გაზრდა, აკრძალულ რესურსებთან წვდომისათვის.

ა. 1 - გ; 2 - ბ; 3 - ა

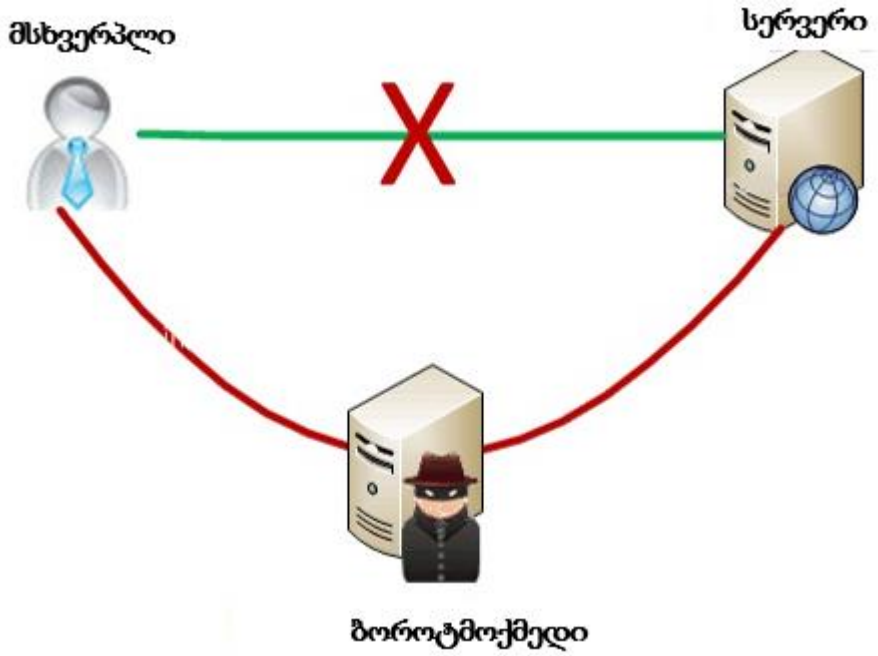
ბ. 1 - ა; 2 - ბ; 3 - გ

გ. 1 - ბ; 2 - ა; 3 - გ

დ. 1 - ბ; 2 - გ; 3 - ა

ე. 1 - ა; 2 - გ; 3 - ბ

22. შეტევის რომელი ტიპია წარმოდგენილი სურათზე:



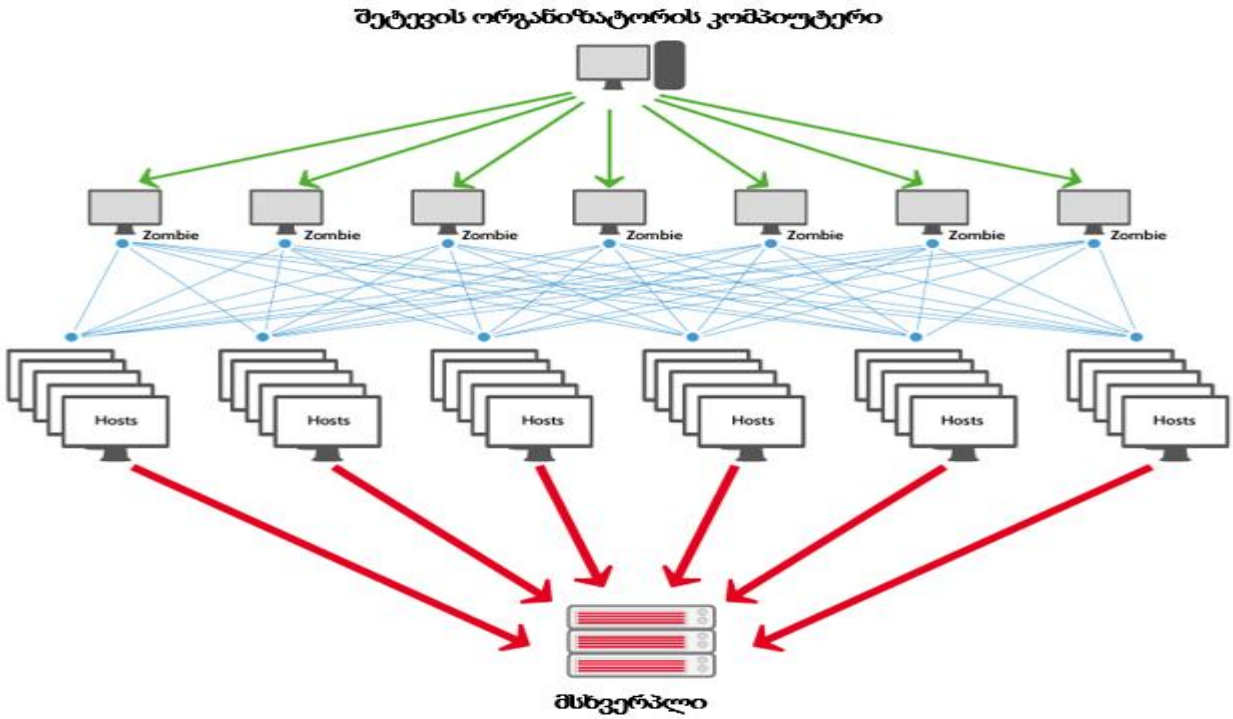
ა. IP-Spoofing

ბ. Man-in-the-Middle

გ. Buffer Overflow

დ. DDos

23. შეტევის რომელი ტიპია წარმოდგენილი სურათზე:



ა. Man-in-the-middle

ბ. IP-Spoofing

გ. Buffer Overflow

დ. DDos

24. ქვემოთ ჩამოთვლილთაგან შეტევის რომელი ტიპი გამოიყენება ვებ-გვერდის ან მონაცემთა ბაზის უკანონო ცვლილებისათვის:

ა. დარღვევა (Decomposition)

ბ. ინექცია (Injecting)

გ. სახეცვლილება (Defacement)

დ. კოდის რეორგანიზაცია (Refactoring)

ე. კომპილაცია (Compiling)

25. ბოროტმოქმედმა განახორციელა შეტევა, რომლის დროსაც კომპანიის მონაცემთა ბაზის სერვერზე გაგზავნილ იქნა მოთხოვნების დიდი რაოდენობა. შეტევის შედეგად სერვერმა შეწყვიტა ავტორიზებული მომხმარებლების მომსახურება. ქვემოთ ჩამოთვლილთაგან, რომელი ტიპის შეტევა განახორციელა ბოროტმოქმედმა:

ა. IP-Spoofing

ბ. Buffer Overflow

გ. Denial of Service

დ. Replay Attack

26. ქვემოთ ჩამოთვლილთაგან, რომელია „Brute force attack"-ის სწორი განმარტება:

ა. ეს არის სპამის ერთ-ერთი ტიპი

ბ. ეს არის ვირუსი, რომელიც შეტევას ახორციელებს კომპიუტერის მყარ დისკზე

გ. ეს არის ხაკერული შეტევის ტიპი, რომლის დროსაც შეტყობინება გადაიცემა ვირუსთან ერთად

დ. ეს არის ბოროტმოქმედის მცდელობა, პაროლების შერჩევის (მორგების) საფუძველზე გატეხოს სისტემა

ე. ეს არის პროგრამა, რომელიც სისტემის დაცვის გვერდის ავლით ხსნის წვდომას კომპიუტერის რესურსებთან

27. Spyware და Adware საზიანო პროგრამების დანიშნულებაა:

ა. დაინსტალირდეს კომპიუტერში და დაიწყოს რაიმე პროდუქტის რეკლამირება ყველა საშუალებით

ბ. ფარულად დააკვირდეს მომხმარებლის ქცევას. გაუგზავნოს მესამე პირს ისეთი კონფიდენციალური ინფორმაცია, როგორებიცაა პაროლები, საკრედიტო ბარათების ნომრები

გ. საიდუმლოდ აკონტროლოს ოპერაციული სისტემა, უსაფრთხოების პროგრამებისაგან თავის აცილების გზით

დ. დაასკანიროს კომპიუტერში არსებული ფაილები და მისი ასლი გაუგზავნოს მესამე პირს

28. Trackware პროგრამის დანიშნულებაა:

ა. შეაგროვოს ნებისმიერი კონფიდენციალური ინფორმაცია (პაროლები, საკრედიტო ბარათების ნომრები) და გაუგზავნოს მესამე პირს

ბ. საიდუმლოდ აკონტროლოს ოპერაციული სისტემა, უსაფრთხოების პროგრამებისაგან თავის აცილების გზით

გ. დაინსტალირდეს სისტემაში და გაუკეთოს რეკლამა რაიმე პროდუქციას

დ. დააკოპიროს და გაუგზავნოს ფაილების ასლი მესამე პირს

29. ჩამოთვლილთაგან რომელი ვირუსული პროგრამა აკეთებს თავისი თავის კოპირებას, ვრცელდება ქსელში და ცვლის ან აზიანებს ფაილებს და მონაცემებს კომპუტერში.

ა. Spyware

გ. Trojan horse

ბ. Worm

დ. Rootkit

30. ჩამოთვლილთაგან რომელი ახასიათებს ანტივირუსულ პროგრამას:

ა. ბაზის განახლების მხარდაჭერა

ბ. ევრისტიკული ანალიზი

გ. ჩაშენებული Firewall-ი

დ. ვინდოუსის განახლება

31. ტროიანული პროგრამა, რომელიც გამოიყენება საზიანო პროგრამების მიწოდებისათვის:

ა. Trojan-Dropper

ბ. Trojan-Clicker

გ. Trojan-Proxy

დ. Trojan-PSW

ე. Trojan-Downloader

32. ჩამოთვლილთაგან რომელი ტროიანული პროგრამაა, საზიანო პროგრამების ინსტალატორი:

ა. Trojan-Clicker

დ. Trojan-Proxy

ბ. Trojan-Dropper

ე. Trojan-Downloader

გ. Trojan-PSW

33. ქვემოთ ჩამოთვლილთაგან რომელია ტროიანულ-ჯაშუშური პროგრამა:

ა. Trojan-PSW

დ. Trojan-Spy

ბ. Trojan-Proxy

ე. Trojan-Dropper

გ. Trojan-Downloader

34. ჩამოთვლილთაგან რომელი ტიპის ვირუსები არსებობს (მონიშნეთ ყველა სწორი პასუხი):

ა. Script virus (სკრიპტ ვირუსი)

ბ. Macro virus (მაკრო ვირუსი)

გ. Bootable virus (ჩამტვირთავი ვირუსი)

დ. File infector (ფაილური ვირუსი)

ე. Random Virus (შემთხვევითი ვირუსი)

ვ. Network virus (ქსელური ვირუსი)

35. . . . . უზრუნველყოფს ქსელში გაზიარებული რესურსების არასანქცირებული გამოყენებისაგან დაცვას. ჩასვით გამოტოვებული სიტყვ(ები)ა:

ა. უარის თქმის შეუძლებლობა

ბ. აუთენტიფიკაცია

გ. მონაცემთა მთლიანობა

დ. მონაცემთა კონფიდენციალურობა

ე. წვდომის კონტროლი

36. უსაფრთხოების მექანიზმების ადმინისტრატორის მოვალეობებში შედის:

ა. მონაცემთა ბაზაში არასანქცირებული წვდომის განხორციელება (SQL-Injection)

ბ. წვდომის მართვის ადმინისტრირება

გ. მარშრუტიზაციისა და ნოტარიზაციის მართვა

დ. გასაღებების და შიფრაციის მართვა

ე. სისტემაში დაუცველი ადგილების მოძებნა და ამ ინფორმაციის გადაგზავნა იმ ბოროტმოქმედისათვის, რომელიც აპირებს მოცემულ სისტემაზე თავდასხმას

37. მათი ძირითადი დანიშნულებაა ისეთი საკომუნიკაციო მახასიათებლების დამოწმება, როგორიცაა მთლიანობა, დრო, გამგზავნის და მიმღების პიროვნება. დამოწმება ხდება სანდო მესამე პირის მიერ, რომელიც ფლობს საკმარის ინფორმაციას. უსაფრთხოების რომელ მექანიზმზეა საუბარი, მოცემულ განმარტებაში:

ა. ნოტარიზაციის მექანიზმები

ბ. შიფრაცია

გ. ელექტრონული ხელმოწერა

დ. წვდომის მართვის მექანიზმები

ე. მარშრუტიზაციის მართვის მექანიზმები

3.2. ACL(Access Control Lists)-ების კონფიგურირება

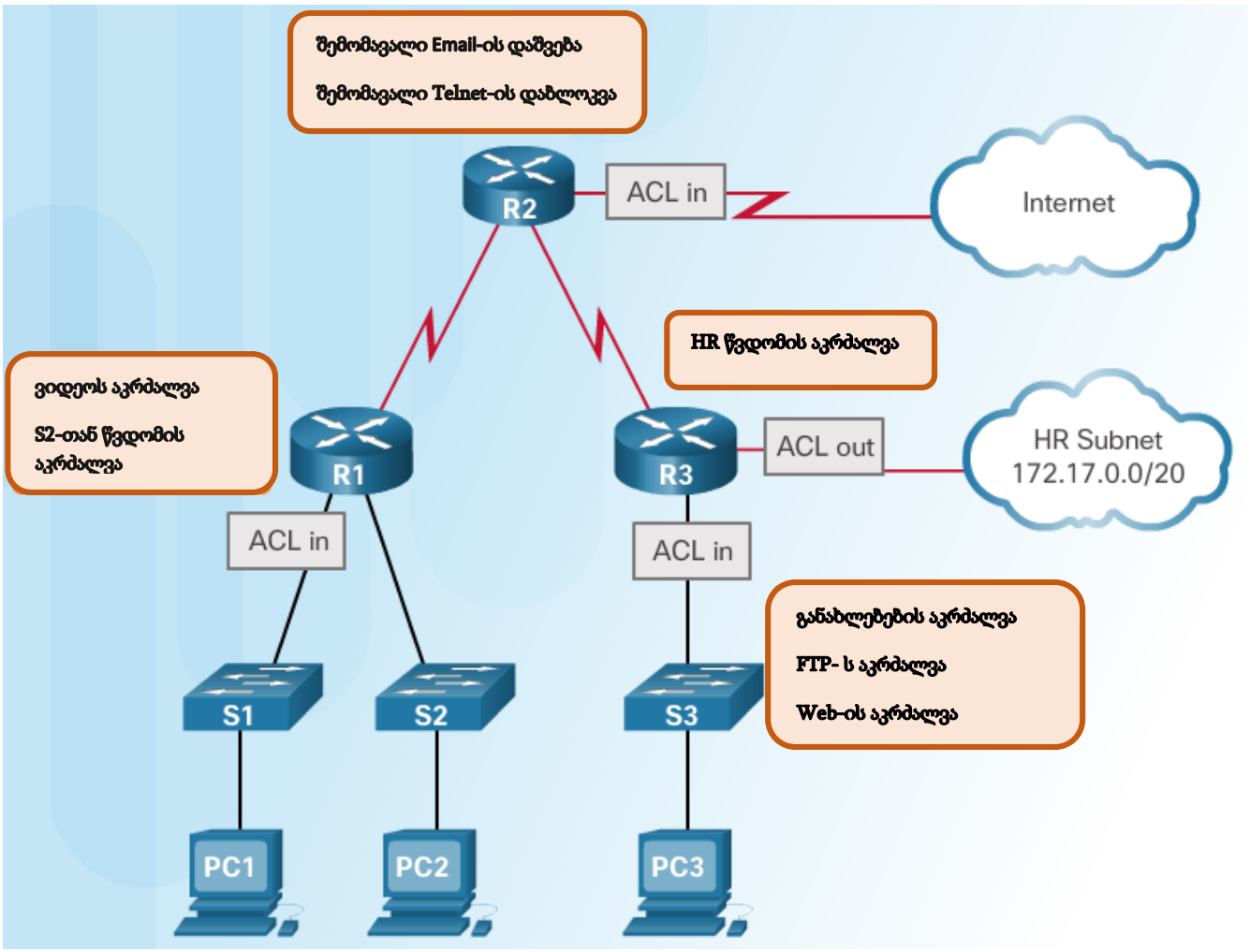
ქსელები რაც უფრო განაგრძობს ზრდას, მით უფრო ხშირად ხდება მისი გამოყენება კონფიდენციალური ინფორმაციის გადასაცემად და შესანახად. ამის გამო გაიზარდა უსაფრთხოების ძლიერ ტექნოლოგიებზე მოთხოვნა, რამაც გამოიწვია ფაიერვოლის გამოგონება. ტერმინი ფაიერვოლი თავდაპირველად დაკავშირებული იყო ცეცხლგამძლე კედელთან, რომელიც როგორც წესი დამზადებულია ქვისგან ან მეტალისგან, რაც ხელს უშლის ცეცხლის ალის გავრცელებას დაკავშირებულ სტრუქტურებს შორის. მსოფლიო ქსელში ფაიერვოლები მიჯნავენ დაცულ სივრცეებს დაუცველი სივრცეებისაგან, რაც უკრძალავს არაავტორიზებულ მომხმარებლებს დაცულ ქსელის რესურსებთან წვდომას.

თავდაპირველად, ბაზისური წვდომის კონტროლის სიები (**ACLs**), მათ შორის სტანდარტული, გაფართოებული, დანომრილი და სახელდებული, იყო ერთადერთი საშუალება, რომელიც უზრუნველყოფდა ფაიერვოლ-დაცვა. სხვა ფაიერვოლ ტექნოლოგიები ბაზარზე გამოჩნდა 90-იანი წლების მიწურულს. ქსელთაშორისი ფაიერვოლები იყენებენ ცხრილებს სესიების სრული ციკლის რეალური დროის მდგომარეობის თვალყურის დევნებისთვის. ქსელთაშორისი ფაიერვოლები ყურადღებას აქცევენ ქსელური ტრაფიკის სესიაზე ორიენტირებულ ბუნებას. პირველი ფაიერვოლები იყენებდნენ „**TCP** საყოველთაოდ აღიარებულ“ პარამეტრს წვდომის კონტროლის სიებისთვის (**ACLs**).

დღესდღეობით არსებობს უამრავი ტიპის ფაიერვოლი, მაგალითად, პაკეტების ფილტრის, მდგომარეობის თვალყურის დევნების, გამოყენებითი დონის გასასვლელის (**Gateway**), პროქსი, მისამართების თარგმნის, ჰოსტზე დაფუძნებული, გამჭვირვალე (**Transparent**) და ჰიბრიდული ფაიერვოლები. თანამედროვე ქსელის დიზაინი აუცილებლად უნდა შეიცავდეს სწორად განლაგებულ ერთი ან რამოდენიმე ფაიერვოლს იმ რესურსების დასაცავად, რომლებიც უნდა იყოს დაცული სანამ დავუშვებთ უსაფრთხო წვდომას ხელმისაწვდომ რესურსებზე.

3.2.1. წვდომის კონტროლის სიების (ACLs) მიმოხილვა

წვდომის კონტროლის სიები ფართოდ გამოიყენება კომპიუტერულ ქსელებში და ქსელურ უსაფრთხოებაში ქსელური შეტევების შესამცირებლად და ქსელური ტრაფიკის კონტროლისათვის. ადმინისტრატორებს შეუძლიათ **ACL**-ების გამოყენება ქსელურ მოწყობილობებზე ტრაფიკის განსაზღვრისა და კონტროლისათვის, რათა დაკმაყოფილებულ იქნას უსაფრთხოების მოთხოვნების ნაკრები, ისე როგორც ნაჩვენებია 3.2.1 სურათზე. **ACL**-ები შეიძლება განსაზღვრულ იქნას ღია სისტემების ურთიერთქმედების (**OSI**) მოდელის მე-2, მე-3, მე-4 და მე-7 დონეებზე.



სურ. 3.2.1. წვდომის კონტროლის სია (ACL)

ისტორიულად, **ACL**-ის ტიპი შეიძლება იდენტიფიცირებულ იქნას ციფრებით, ისე როგორც ნაჩვენებია 3.2.2 სურათზე. მაგალითად, დანომრილი წვდომის კონტროლის სიები

200-299 დიაპაზონში, გამოყენებულ იქნა ტრაფიკის კონტროლისათვის **Ethernet**-ის ტიპის შესაბამისად. 700-799-ით დანომრილი **ACL**-ები მიუთითებს, რომ ტრაფიკი კლასიფიცირებულია და იმართება **MAC** მისამართების საფუძველზე.

პროტოკოლი	დიაპაზონი
IP	1-99, 1300-1999
Extended IP	100-199, 2000-2699
Ethernet type code	200-299
DECnet and Extended DECnet	300-399
XNS	400-499
Extended XNS	500-599
AppleTalk	600-699
Ethernet address	700-799
IPX	800-899
Extended IPX	900-999
IPX SAP	1000-1099
Extended transparent bridging	1100-1199

სურ. 3.2.2. წვდომის კონტროლის სიების (ACLs) ტიპები

დღეს ტრაფიკის კლასიფიცირებისას, ყველაზე გავრცელებული ტიპის **ACL**-ები იყენებენ **IPv4** და **IPv6** მისამართებს და გადაცემის კონტროლის პროტოკოლისა (**TCP**) და მომხმარებლის **Datagram** პროტოკოლის (**UDP**) პორტის ნომრებს. სტანდარტული ან გაფართოებული **IPv4** წვდომის კონტროლის სიები შეიძლება იყოს დანომრილი და სახელდებული. **IPv6 ACL**-ები უნდა იყენებდნენ სახელს.

3.2.2. დანომრილი და სახელდებული წვდომის კონტროლის სიების კონფიგურაცია

ACL არის დაშვების და აკრძალვის მდგომარეობების მიმდევრობითი სია, რომელიც ცნობილია როგორც წვდომის კონტროლის ჩანაწერები (**ACEs**). **ACEs**-ს ასევე ხშირად ეძახიან **ACL** მდგომარეობებს. **ACEs** შეიძლება შექმნილ იქნას ტრაფიკის ფილტრაციისთვის, ზოგიერთი კრიტერიუმის საფუძველზე, როგორიცაა: გამგზავნის მისამართი, ადრესატის მისამართი, პროტოკოლი და პორტის ნომრები.

სტანდარტული წვდომის კონტროლის სიები ემთხვევა პაკეტებს, თუ შევისწავლით ამ პაკეტის **IP** თავსართის გამგზავნის წყაროს **IP** მისამართის ველს. მოცემული **ACL**-ები გამოიყენება პაკეტების ფილტრაციისთვის მხოლოდ მესამე დონის წყაროს ინფორმაციის საფუძველზე. დანომრილი სტანდარტული წვდომის კონტროლის სიის კონფიგურაციისთვის გამოიყენეთ 3.2.3 ცხრილში ნაჩვენები ბრძანებების სინტაქსი.

access-list {acl-#} {**permit**	**deny**	**remark**} *source-addr* [*source-wildcard*] [**log**]
სტანდარტული ACL-ები IP პაკეტებს ფილტრავს მხოლოდ გამგზავნის მისამართით

პარამეტრი	**აღწერა**
acl-#	ეს არის ათობითი რიცხვი 1-დან 99-მდე, ან 1300-დან 1999-მდე.
deny	კრძალავს წვდომას თუ პირობებს შეესაბამება
permit	უშვებს წვდომას თუ პირობებს შეესაბამება
remark	**IP** დაშვების სიაში ამატებს ჩანაწერების შესახებ შენიშვნებს, რათა ადვილი იყოს სიის გაგება და შემოწმება.
source-addr	ქსელის ან ჰოსტების რაოდენობა, რომლებსაც უნდა გაეგზავნოს პაკეტი. არსებობს source-addr-ის მითითების ორი გზა: • გამოიყენეთ 32 ბიტიანი სიდიდე ოთხ ნაწილად დაყოფილ, წერტილებით გამოყოფილ ათობითი რიცხვის ფორმატში. • გამოიყენეთ **any** საკვანძო სიტყვა, როგორც აბრევიატურა **source** და **source-wildcard**-სთვის 0.0.0.0 255.255.255.255-ზე.
source-wildcard	(არჩევითი) 32-ბიტიანი მიმაგრებული ნიღაბი, რომელიც გამოყენებულია წყაროსთვის. დასვით ერთიანი ბიტების პოზიციაში თუ გსურთ გამოტოვება.
log	(არჩევითი) იძახებს საინფორმაციო ლოგირების შეტყობინებას იმ პაკეტის შესახებ, რომელიც ემთხვევა იმ ჩანაწერს, რომელიც უნდა გაიგზავნოს კონსოლზე. (შეტყობინებების დონე, რომელიც ლოგირებულია კონსოლზე იმართება **logging console** ბრძანებით.)

	შეტყობინება შეიცავს **ACL** რიცხვს, როდის იქნა პაკეტი დაშვებული ან აკრძალული, გამგზავნის მისამართი და პაკეტების რაოდენობა. შეტყობინება შექმნილია პირველი პაკეტისთვის, რომელიც ემთხვევა და მომდევნო ხუთი წუთის ინტერვალის შემდეგ, რომელიც შეიცავს დაშვებული ან აკრძალული პაკეტების რაოდენობას წინა ხუთი წუთის ინტერვალში.

ცხრილი 3.2.3 - სტანდარტული დანომრილი ACL სინტაქსი

გაფართოებული წვდომის კონტროლის სიები შეესაბამება პაკეტებს მესამე და მეოთხე დონეების წყაროს და ადრესატის ინფორმაციის საფუძველზე. მეოთხე დონე შეიძლება შეიცავდეს **TCP** და **UDP** პორტის ინფორმაციას. გაფართოებული **ACL**-ები სტანდარტულ **ACL**-ებთან შედარებით იძლევა მეტ მოქნილობას და ქსელთან წვდომის კონტროლს. დანომრილი გაფართოებული წვდომის კონტროლის სიის კონფიგურაციისთვის გამოიყენეთ 3.2.4 ცხრილში მოცემული ბრძანებათა სინტაქსი.

access-list *acl-#* {**permit** | **deny** | **remark**} *protocol source-addr* {*source-wildcard*} *dest-addr* [*dest-wildcard*] [*operator port*] [**established**]

პარამეტრი	აღწერა
acl-#	ახდენს წვდომის სიის იდენტიფიკაციას 100-დან 199-მდე (გაფართოებული **IP ACL**-სთვის) და 2000-დან 2699-მდე რიცხვების გამოყენებით (გაშლილი **IP ACL**-ებისთვის).
deny	კრძალავს წვდომას თუ პირობებს შეესაბამება
permit	უშვებს წვდომას თუ პირობებს შეესაბამება
remark	გამოიყენება შენიშვნის ან კომენტარის შესატანად
protocol	ინტერნეტ პროტოკოლის სახელი ან რიცხვი. გავრცელებული საკვანძო სიტყვებია: **icmp**, **ip**, **tcp** ან **udp**. ნებისმიერი ინტერნეტ

	პროტოკოლის (**ICMP**, **TCP** და პროტოკოლების ჩათვლით) შესადარებლად გამოიყენეთ **ip** საკვანძო სიტყვა.
source-addr	ქსელის ან ჰოსტის რიცხვი საიდანაც პაკეტი იგზავნება
source- wildcard	მიმაგრებული ბიტები, რომლებიც გამოიყენება წყაროსთვის
destination-addr	ქსელის ან ჰოსტის რიცხვი ვისთვისაც პაკეტი იგზავნება
destination-wildcard	მიმაგრებული ბიტები, რომლებიც გამოიყენება ადრესატისთვის
operator	(არჩევითი) ადარებს გამგზავნის და ადრესატის პორტებს. შესაძლო ოპერანდები შეიცავს: **lt** (less than - ნაკლებია ვიდრე), **gt** (grether than - მეტია ვიდრე), **eq** (equal - უდრის), **neq** (not equal - არ უდრის) და **range** (დიაპაზონი).
port	(არჩევითი) **TCP** ან **UDP** პორტის ათობითი რიცხვი ან სახელი
established	(არჩევითი) მხოლოდ **TCP** პროტოკოლისათვის: მიუთითებს შექმნილ კავშირს.

3.2.4 ცხრილი - გაფართოებული დანომრილი ACL სინტაქსი

რიცხვების ნაცვლად **ACL**-ის კონფიგურაციისთვის შეიძლება გამოყენებულ იქნას სახელიც. სახელდებული წვდომის კონტროლის სიები უნდა იყოს მითითებული როგორც სტანდარტული ან გაფართოებული. სახელდებული სტანდარტული ან გაფართოებული **ACL**-ის კონფიგურაციისთვის გამოიყენეთ 3.2.5 ცხრილში მოცემული ბრძანებათა სინტაქსი.

ACL-ის სახელი

Router (config)# **ip access-list** [**standard** | **extended**] *name_of_ACL*

წვდომის კონტროლის ჩანაწერების (ACEs) კონფიგურაცია

სტანდარტული ACE სინტაქსი

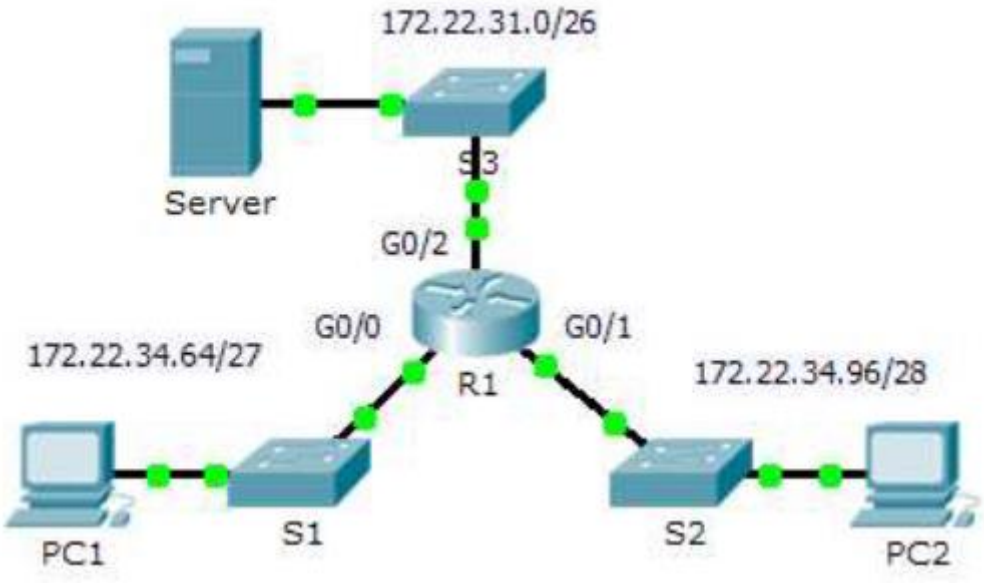
Router (config-std-nacl)# {**permit** | **deny** | **remark**} {source [source-wildcard] | **any**}

გაფართოებული ACE სინტაქსი

Router (config-ext-nacl)# {**permit** | **deny** | **remark**} protocol source-addr [source-wildcard] dest-address [dest-wildcard] [operator port]

გაფართოებული წვდომის კონტროლის სიების (ACLs) კონფიგურაცია

ტოპოლოგია



მისამართების ცხრილი

მოწყობილობა	ინტერფეისი	IP მისამართი	ქვექსელის ნიღაბი	ნაგულისხმევი გასასვლელი
R1	G0/0	172.22.34.65	255.255.255.224	არ აქვს
	G0/1	172.22.34.97	255.255.255.240	არ აქვს
	G0/2	172.22.34.1	255.255.255.192	არ აქვს
სერვერი	ქსელის ადაპტერი	172.22.34.62	255.255.255.192	172.22.34.1
PC1	ქსელის ადაპტერი	172.22.34.66	255.255.255.224	172.22.34.65
PC2	ქსელის ადაპტერი	172.22.34.98	255.255.255.240	172.22.34.97

შესასრულებელი დავალებები:

ნაწილი №1: გაფართოებული დანომრილი ACL-ების კონფიგურაცია, გამოყენება და შემოწმება

ნაწილი №2: გაფართოებული სახელდებული ACL-ების კონფიგურაცია, გამოყენება და შემოწმება

ზოგადი ინფორმაცია/ სცენარი

ორ თანამშრომელს სჭირდება სერვერის მიერ უზრუნველყოფილ სერვისებთან წვდომა. **PC1**-ს სჭირდება მხოლოდ **FTP** წვდომა, ხოლო **PC2**-ს - მხოლოდ ვებთან წვდომა. ორივე კომპიუტერს შეუძლია სერვერის დაპინგვა, მაგრამ ერთმანეთს ვერ პინგავენ.

ნაწილი №1: გაფართოებული დანომრილი ACL-ების კონფიგურაცია, გამოყენება და შემოწმება

პირველი ეტაპი: ACL-ის კონფიგურაცია FTP-ს და ICMP-ის დასაშვებად.

ა. **R1**-ის გლობალური კონფიგურაციის რეჟიმში შეიყვანეთ ქვემოთ მოცემული ბრძანება, გაფართოებული წვდომის სიისთვის პირველი მოქმედი რიცხვის დასადგენად.

R1 (config) # **access-list ?**

<1-99> IP standard access list (IP სტანდარტული წვდომის სია)

<100-199> IP extended access list (IP გაფართოებული წვდომის სია)

ბ. ბრძანებაში დაამატეთ რიცხვი **100**, ხოლო შემდეგ დაუწერეთ კითხვის ნიშანი.

R1 (config)# **access-list 100 ?**

deny Specify packets to reject (მიეთითება პაკეტების უარყოფისათვის)

permit Specify packets to forward (მიეთითება პაკეტების გადაგზავნის დასაშვებად)

remark Access list entry comment (წვდომის სიის მნიშვნელობის კომენტარი)

გ. FTP ტრაფიკის დაშვებისთვის შეიყვანეთ **permit** ბრძანება და შემდეგ კითხვის ნიშანი.

R1 (config) # **access-list 100 permit ?**

ahp Authentication Header Protocol (აუთენტიფიკაციის თავსართის პროტოკოლი)

eigrp Cisco's EIGRP routing protocol (Cisco-ს EIGRP მარშრუტიზაციის პროტოკოლი)

esp Encapsulation Security Payload (უსაფრთხოების დატვირთვის ენკაპსულაცია)

gre Cisco's GRE tunneling (Cisco-ს GRE ტუნელირება)

icmp Internet Control Message Protocol შეტყობინებების მართვის ინტერნეტ პროტოკოლი)

ip Any Internet Protocol (ნებისმიერი ინტერნეტის პროტოკოლი)

ospf OSPF routing protocol (OSPF მარშრუტიზაციის პროტოკოლი)

tcp Transmission Control Protocol (გადაცემის მართვის პროტოკოლი)

udp User Datagram Protocol (მომხმარებლის დატაგრამის* პროტოკოლი)

* **დატაგრამა** არის გადაცემის ბაზისური ერთეული, რომელიც დაკავშირებულია პაკეტებად-კომუტირებად ქსელთან

დ. მოცემული წვდომის კონტროლის სია უშვებს FTP და ICMP-ის. ICMP არის სიაში, FTP-არა, რადგან FTP იყენებს TCP პროტოკოლს. ამიტომ შეიყვანეთ **tcp** ბრძანება, ACL დახმარების დახვეწის ხელშეწყობისათვის.

R1 (config) # **access-list 100 permit tcp ?**

A. B. C. D Source address (გამგზავნის მისამართი)

any Any source host (ნებისმიერი გამგზავნი ჰოსტი)

host A single source host (ერთადერთი წყარო ჰოსტი)

ე. მიაქციეთ ყურადღება, რომ ჩვენ შეგვიძლია ფილტრის ჩართვა მხოლოდ **PC1**-სთვის, **host** საკვანძო სიტყვის გამოყენებით ან შეგვიძლია დავუშვათ ნებისმიერი - **any** ჰოსტი. ამ შემთხვევაში დაშვებულია ნებისმიერი მოწყობილობა, რომელიც მიეკუთვნება 172.22.34.64/27 ქსელს. შეიყვანეთ ქსელის მისამართი და მიუწერეთ კითხვის ნიშანიც.

R1 (config) # **access-list 100 permit tcp 172.22.34.64 ?**

A. B. C. D Source wildcard bits

3. გამოთვალეთ wildcard ნიღაბი, ქვექსელის ნიღაბის საპირისპირო ორობითი რიცხვის განსაზღვრისთვით.

11111111.11111111.11111111.11100000 = 255.255.255.224

00000000.00000000.00000000.00011111 = 0.0.0.31

ზ. შეიყვანეთ wildcard ნიღაბი კითხვის ნიშანთან ერთად.

R1 (config) # **access-list 100 permit tcp 172.22.34.64 0.0.0.31 ?**

A. B. C. D Destination address (ადრესატის მისამართი)

any Any destination host (ნებისმიერი ადრესატი ჰოსტი)

eq Match only packets on a given port number (ემთხვევა მხოლოდ მითითებული პორტის ნომრის მქონე პაკეტებს)

gt Match only packets with a greater port number (ემთხვევა მხოლოდ მაღალი პორტის ნომრის მქონე პაკეტებს)

host A single destination host (მხოლოდ ერთი დანიშნულების ჰოსტი)

lt Match only packets with a lower port number (ემთხვევა მხოლოდ დაბალი პორტის ნომრის მქონე პაკეტებს)

neq Match only packets not on a given port number (ემთხვევა მხოლოდ არა მითითებული პორტის ნომრის მქონე პაკეტებს)

range Match only packets in the range of port numbers (ემთხვევა მხოლოდ პაკეტებს, რომლებიც არიან პორტის ნომრების დიაპაზონში)

თ. ადრესატის მისამართის კონფიგურაცია. მოცემულ დავალებაში ჩვენ ვფილტრავთ ტრაფიკს ერთი დანიშნულების ადგილისთვის, როგორიცაა სერვერი. შეიყვანეთ **host** საკვანძო სიტყვა სერვერის IP მისამართის შემდეგ.

R1 (config) # **access-list 100 permit tcp 172.22.34.64 0.0.0.31 host 172.22.34.62 ?**

dscp	Match packets with given dscp value (შეესაბამება dscp მნიშვნელობის მქონე პაკეტებს)
eq	Match only packets on a given port number (ემთხვევა მხოლოდ მითითებული პორტის ნომრის მქონე პაკეტებს)
established	დადგენილი
gt	Match only packets with a greater port number (ემთხვევა მხოლოდ მაღალი პორტის ნომრის მქონე პაკეტებს)
lt	Match only packets with a lower port number (ემთხვევა მხოლოდ დაბალი პორტის ნომრის მქონე პაკეტებს)
neq	Match only packets not on a given port number (ემთხვევა მხოლოდ არა მითითებული პორტის ნომრის მქონე პაკეტებს)
precedence	Match packets with given precedence value (ემთხვევა პაკეტებს, რომლებსაც აქვთ მაღალი პრიორიტეტის მნიშვნელობა)
range	Match only packets in the range of port numbers (ემთხვევა მხოლოდ პაკეტებს, რომლებიც არიან პორტის ნომრების დიაპაზონში)

<cr>

ი. მიაქციეთ ყურადღება, რომ ერთ-ერთი პარამეტრი არის **<cr>** (გადაგზავნის დაბრუნება). სხვა სიტყვებით რომ ვთქვათ თქვენ შეგიძლიათ დააჭიროთ **Enter** ღილაკს და მოქმედება დაუშვებს ყველა TCP ტრაფიკს. თუმცა ჩვენ ვუშვებთ მხოლოდ FTP ტრაფიკს; ამიტომ შეიყვანეთ **eq** საკვანძო სიტყვა და მიუწერეთ კითხვის ნიშანი

ხელმისაწვდომი პარამეტრების გამოსატანად. შემდეგ შეიყვანეთ **ftp** და დააჭირეთ **Enter** ღილაკს.

R1 (config) # **access-list 100 permit tcp 172.22.34.64 0.0.0.31 host 172.22.34.62 eq ?**

<0 – 65535> port number (პორტის ნომერი)

ftp File Transfer Protocol (21) (ფაილების გადაცემის პროტოკოლი)

pop3 Post Office Protocol v3 (110) (საფოსტო პროტოკოლი)

smtp Simple Mail Transport Protocol (25) (ელექტრონული შეტყობინებების მარტივი გადაცემის პროტოკოლი)

telnet Telnet (23)

www Word Wide Web (HTTP, 80) (მსოფლიო ქსელი)

R1 (config) # **access-list 100 permit tcp 172.22.34.64 0.0.0.31 host 172.22.34.62. eq ftp**

კ. მეორე წვდომის სიის შექმნა ICMP-ის ტრაფიკის დაშვებისათვის **PC1**-დან **სერვერამდე.** აღსანიშნავია, რომ წვდომის სიის ნომერი რჩება იგივე და არ არის საჭირო რაიმე განსაკუთრებული ICMP ტრაფიკის ტიპის მითითება.

R1 (config) # **access-list 100 permit icmp 172.22.34.64 0.0.0.31 host 172.22.34.62**

ლ. ნაგულისხმევად ყველა სხვა ტრაფიკი არის დაბლოკილი.

მეორე ეტაპი: ACL-ის გამოყენება სწორ ინტერფეისზე, ტრაფიკის ფილტრაციისთვის

R1 მარშრუტიზატორის გადმოსახედიდან ტრაფიკი, რომელზეც ACL 100 გამოიყენება არის შემომავალი ქსელიდან, რომელიც დაკავშირებულია GigabitEthernet 0/0 ინტერფეისთან. შედით ინტერფეისის კონფიგურაციის რეჟიმში და გამოიყენეთ ACL-ი.

R1 (config) # **interface gigabitEthernet 0/0**

R1 (config-if) # **ip access-group 100 in**

მესამე ეტაპი: ACL-ის მოქმედების შემოწმება

ა. **PC1**-დან დაპინგეთ **სერვერი.** თუ პინგი წარუმატებლად დამთავრდა, გაგრძელებამდე შეამოწმეთ IP მისამართები.

ბ. განახორციელეთ FTP კავშირი **PC1**-დან **სერვერზე.** მომხმარებლის სახელი და პაროლი არის **cisco**.

PC> **ftp 172.22.34.62**

გ. სერვერის FTP სერვისიდან გამოსვლა

ftp> **quit**

დ. **PC1**-დან დაპინგეთ **PC2**. ადრესატი ჰოსტი უნდა იყოს მიუწვდომელი, რადგან ტრაფიკი არ არის დაშვებული.

ნაწილი №2: გაფართოებული სახელდებული წვდომის კონტროლის სიის (ACL) კონფიგურაცია, გამოყენება და შემოწმება

პირველი ეტაპი: ACL-ის კონფიგურაცია HTTP წვდომისა და ICMP-ის დასაშვებად.

ა. სახელდებული წვდომის კონტროლის სიები (ACLs) იწყება **ip** საკვანძო სიტყვით. **R1** მარშრუტიზატორის გლობალური კონფიგურაციის რეჟიმში შეიყვანეთ ქვემოთ მოცემული ბრძანება, კითხვის ნიშანთან ერთად.

R1 (config) # **ip access-list ?**

extended Extended Access List (გაფართოებული წვდომის სია)

standard Standard Access List (სტანდარტული წვდომის სია)

ბ. თქვენ შეგიძლიათ სახელდებული სტანდარტული და გაფართოებული წვდომის კონტროლის სიების კონფიგურაცია. მოცემული წვდომის სია ფილტრავს როგორც წყაროს ისე ადრესატის IP მისამართებს; თუმცა ის უნდა იყოს გაფართოებული. შეიყვანეთ **HTTP_ONLY**, როგორც დასახელება.

R1 (config) # **ip access-list extended HTTP_ONLY**

გ. ბრძანებათა ველის ცვლილებები. თქვენ ახლა იმყოფებით გაფართოებული სახელდებული წვდომის კონტროლის რეჟიმში. **PC2** LAN-ზე ყველა მოწყობილობას სჭირდება TCP დაშვება. შეიყვანეთ ქსელის მისამართი კითხვის ნიშანთან ერთად.

R1 (config-ext-nacl) **permit tcp 172.22.34.96 ?**

A. B. C. D Source wildcard bits

დ. Wildcard-ის გამოთვლის ალტერნატიული გზაა 255.255.255.255-ს გამოვაკლოთ ქვექსელის ნიღაბი.

255.255.255.255

-

255.255.255.240

- - - - - - - - - - - - - - -

= 0. 0. 0. 15

R1 (config-ext-nacl) # **permit tcp 172.22.34.96 0.0.0.15 ?**

ე. დაასრულეთ მდგომარეობა სერვერის მისამართის მითითებით, ისე როგორც ნაწილ №1-ში და გაფილტრეთ **www** ტრაფიკი.

R1 (config-ext-nacl) # **permit tcp 172.22.34.96 0.0.0.15 host 172.22.34.62 eq www**

ვ. მეორე წვდომის კონტროლის სიის შექმნა ICMP (ping და ა.შ) ტრაფიკის დასაშვებად **PC2**-დან **სერვერამდე.** შენიშვნა: ბრძანებათა ველი რჩება იგივე და არ არის საჭირო ICMP ტრაფიკის კონკრეტული ტიპის მითითება.

R1 (config-ext-nacl) # **permit icmp 172.22.34.96 0.0.0.15 host 172.22.34.62**

ზ. ნაგულისხმევად ყველა დანარჩენი ტრაფიკი დაბლოკილია. გამოდით გაფართოებული სახელდებული წვდომის კონტროლის სიის კონფიგურაციის რეჟიმიდან.

მეორე ეტაპი: ACL-ის გამოყენება სწორ ინტერფეისზე, ტრაფიკის ფილტრაციისთვის.

R1 მარშრუტიზატორის გადმოსახედიდან ტრაფიკი, რომელზეც წვდომის სია **HTTP_ONLY** გამოიყენება არის შემომავალი ქსელიდან, რომელიც დაკავშირებულია

GigabitEthernet 0/1 ინტერფეისთან. შედით ინტერფეისის კონფიგურაციის რეჟიმში და გამოიყენეთ ACL-ი.

R1 (config) # **interface gigabitEthernet 0/1**

R1 (config-if) # **ip access-group HTTP_ONLY in**

მესამე ეტაპი: ACL-ის მოქმედების შემოწმება

ა. **PC2**-დან დაპინგეთ **სერვერი.** პინგი უნდა დასრულდეს წარმატებულად. თუ პინგი წარუმატებლად დამთავრდა, გაგრძელებამდე შეამოწმეთ IP მისამართები.

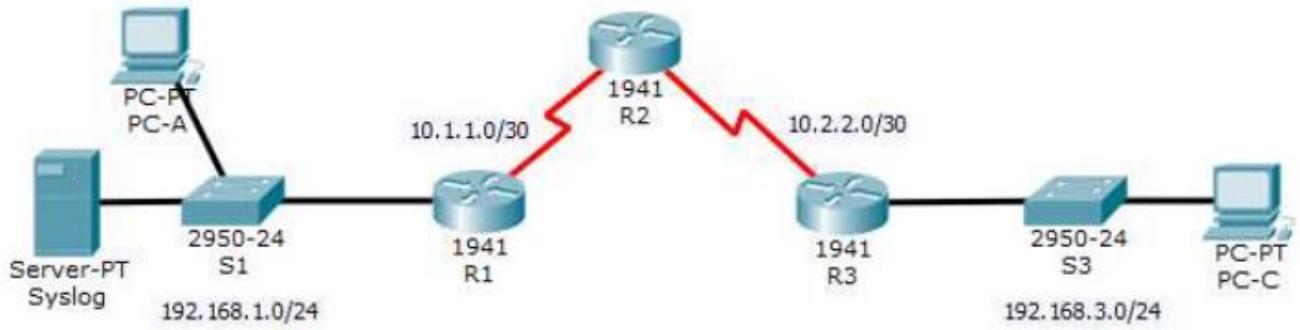
ბ. განახორციელეთ FTP კავშირი **PC2**-დან **სერვერზე.** კავშირი უნდა ჩავარდეს.

გ. **PC2**-ზე გახსენით ვებ ბრაუზერი და შეიყვანეთ **სერვერის** IP მისამართი, როგორც URL. კავშირი უნდა დასრულდეს წარმატებულად.

3.3. ქსელური შეტევებისგან თავდაცვისთვის საჭირო თანამედროვე მეთოდების გარჩევა

IOS-ის შეღწევის პრევენციის სისტემის (IPS) კონფიგურაცია ბრძანებათა ინტერფეისის (CLI) გამოყენებით

ტოპოლოგია



მისამართების ცხრილი:

მოწყობილობა	ინტერფეისი	IP მისამართი	ქვექსელის ნიღაბი	ნაგულისხმევი გასასვლელი	Switch Port
R1	G0/1	192.168.1.1	255.255.255.0	არ აქვს	S1 F0/1
	S0/0/0	10.1.1.1	255.255.255.252	არ აქვს	არ აქვს
R2	S0/0/0 (DCE)	10.1.1.2	255.255.255.252	არ აქვს	არ აქვს
	S0/0/1 (DCE)	10.2.2.2	255.255.255.252	არ აქვს	არ აქვს
R3	G0/1	192.168.3.1	255.255.255.0	არ აქვს	S3 F0/1
	S0/0/0	10.2.2.1	255.255.255.252	არ აქვს	არ აქვს
Syslog	ქსელის ადაპტერი	192.168.1.50	255.255.255.0	192.168.1.1	S1 F0/2
PC-A	ქსელის ადაპტერი	192.168.1.2	255.255.255.0	192.168.1.1	S1 F0/3
PC-C	ქსელის ადაპტერი	192.168.3.2	255.255.255.0	192.168.3.1	S3 F0/2

შესასრულებელი ამოცანები:

- IOS IPS-ის ჩართვა
- ლოგირების კონფიგურაცია
- IPS ხელწერის შეცვლა
- IPS-ის შემოწმება

ზოგადი ინფორმაცია / სცენარი

თქვენი დავალებაა IPS-ის ჩართვა R1 მარშრუტიზატორზე, 192.168.1.0 ქსელში შემავალი ტრაფიკის სკანირებისათვის.

Syslog სერვერი გამოიყენება IPS შეტყობინებების ლოგირებისთვის. თქვენ უნდა დააკონფიგუროთ მარშრუტიზატორი Syslog სერვერის იდენტიფიცირებისათვის, რათა მიიღოთ ლოგირების შეტყობინებები. სწორი დროისა და თარიღის ჩვენება Syslog შეტყობინებებში არის სასიცოცხლო მნიშვნელობის, როდესაც ხდება syslog-ით ქსელის მონიტორინგი. მომართეთ საათი და დააკონფიგურეთ დროითი ნიშნულის სერვისი, მარშრუტიზატორებზე ლოგირებისთვის. ბოლოს ჩართეთ IPS განგაშის სიგნალის წარმოებისთვის და ICMP ექო საპასუხო შიდა პაკეტების შეწყვეტისთვის.

სერვერი და პერსონალური კომპიუტერები წინასწარ არის დაკონფიგურებული. ასევე წინასწარაა დაკონფიგურებული მარშრუტიზატორები, ქვემოთ მოცემული პარამეტრებით:

- Enable პაროლი: **ciscoenpa55**
- კონსოლის პაროლი: **ciscoconpa55**
- SSH მომხმარებლის სახელი და პაროლი: **SSHadmin/ ciscosshpa55**
- OSPF 101

ნაწილი №1: IOS IPS-ის ჩართვა

შენიშვნა: Packet Tracer-ში, მარშრუტიზატორებს უკვე აქვთ იმპორტირებული და მოქმედებაში მყოფი ხელწერის ფაილები. მათ აქვთ ნაგულისხმევი xml ფაილები ფლეშ

მეხსიერებაში. ამიტომ არ არის აუცილებელი ღია შიფრაციის გასაღების კონფიგურაცია და ხელწერის ფაილების ხელით იმპორტის შესრულება.

პირველი ეტაპი: უსაფრთხოების ტექნოლოგიების პაკეტის (Security Technology Package) ჩართვა

ა. R1 მარშრუტიზატორზე გაუშვით show version ბრძანება ტექნოლოგიების პაკეტის ლიცენზიის ინფორმაციის სანახავად.

ბ. თუ უსაფრთხოების ტექნოლოგიების პაკეტი არ არის ჩართულიმ გამოიყენეთ ქვემოთ მოცემული ბრძანება პაკეტის ჩასართავად.

R1 (config)# **license boot module c1900 technology-package securityk9**

გ. დაეთანხმეთ საბოლოო მომხმარებლის სალიცენზიო შეთანხმებას.

დ. შეინახეთ გაშვებული კონფიგურაცია და ხელახლა ჩატვირთეთ მარშრუტიზატორი, უსაფრთოების ლიცენზიის ჩასართავად.

ე. **show version** ბრძანების გამოყენებით შეამოწმეთ ჩართულია თუ არა უსაფრთხოების ტექნოლოგიების პაკეტი.

მეორე ეტაპი: ქსელური შეერთების შემოწმება

ა. **PC-C**-დან დაპინგეთ **PC-A**. Ping-ი უნდა დასრულდეს წარმატებით.

ბ. **PC-A**-დან დაპინგეთ **PC-C**. Ping-ი უნდა დასრულდეს წარმატებით.

მესამე ეტაპი: IOS IPS საკონფიგურაციო კატალოგის შექმნა ფლეშ მეხსიერებაში.

R1 მარშრუტიზატორზე **mkdir** ბრძანების გამოყენებით ფლეშში შექმენით კატალოგი. კატალოგს სახელი მიანიჭეთ **ipsdir**.

მეოთხე ეტაპი: IPS ხელწერის შენახვის ადგილის კონფიგურაცია.

R1 მარშრუტიზატორზე დააკონფიგურეთ მესამე ეტაპზე შექმნილი კატალოგი IPS ხელწერის შენახვის ადგილად.

მეხუთე ეტაპი: IPS წესის შექმნა.

R1 მარშრუტიზატორზე გლობალური კონფიგურაციის რეჟიმში შექმენით IPS წესის სახელი **ip ips name *name*** ბრძანების გამოყენებით. IPS წესის სახელია **iosips**.

მეექვსე ეტაპი: ლოგირების ჩართვა

IOS IPS მხარს უჭერს syslog ფუნქციას, რათა გაგზავნილ იქნას ღონისძიების გაფრთხილება. Syslog გაფრთხილება ჩართულია ნაგულისხმევად. თუ ლოგირების კონსოლი ჩართულია, IPS აჩვენებს syslog შეტყობინებებს.

ა. თუ არ არის ჩართული syslog-ი, მოახდინეთ მისი ჩართვა.

ბ. აუცილებლობის შემთხვევაში გამოიყენეთ clock set ბრძანება პრივილეგირებულ EXEC რეჟიმში, საათის ხელახლა მოსამართად.

გ. show run ბრძანების გამოყენებით მარშრუტიზატორზე შეამოწმეთ ჩართულია თუ არა დროითი ნიშნული სერვისი ლოგირებისთვის. თუ არ არის ჩართული, ჩართეთ დროითი ნიშნულის სერვისი.

დ. გაგზავნეთ ლოგ შეტყობინებები syslog სერვერზე, 192.168.1.50 მისამართზე.

მეშვიდე ეტაპი: IOS IPS-ის კონფიგურაცია ხელწერის კატეგორიების გამოყენებისათვის.

მოაცილეთ ყველა ხელწერის კატეგორია retired true ბრძანებით (ყველა ხელწერა, ხელწერის გამოშვების მიხედვით). დააბრუნეთ IOS_IPS Basic კატეგორია retired false ბრძანების გამოყენებით.

მერვე ეტაპი: IPS წესის გამოყენება ინტერფეისზე.

გამოიყენეთ IPS წესი ინტერფეისზე ip ips name *direction* ბრძანებით, ინტერფეისის კონფიგურაციის რეჟიმში. გამოიყენეთ გამავალი წესი R1 მარშრუტიზატორის G0/1 ინტერფეისზე. IPS-ის ჩართვის შემდეგ, რამდენიმე ლოგ შეტყობინება გაიგზავნება კონსოლის ხაზზე, რომელიც მიანიშნებს რომ მიმდინარეობს IPS ძრავის ინიციალიზაცია.

შენიშვნა: in მიმართულება მიანიშნებს, რომ IPS ამოწმებს მხოლოდ ინტერფეისზე შემომავალ ტრაფიკს. ანალოგიურად, **out** ნიშნავს, რომ IPS ამოწმებს მხოლოდ ინტერფეისიდან გამავალ ტრაფიკს.

ნაწილი №2: ხელწერის შეცვლა

პირველი ეტაპი: ხელწერის ღონისძიების მოქმედების შეცვლა

დააბრუნეთ ექო მოთხოვნის ხელწერა (ხელწერა 2004, subsig ID 0), ჩართეთ ის და შეცვალეთ ხელწერის მოქმედება, რათა მოხდეს გამაფრთხილებელი სიგნალის გაშვება და შეწყვეტა.

მეორე ეტაპი: show ბრძანებების გამოყენება IPS-ის შესამოწმებლად.

გამოიყენეთ **show ip ips all** ბრძანება, IPS კონფიგურაციის საბოლოო მდგომარეობის სანახავად.

რომელ ინტერფეისებზე და რა მიმართულებითაა გამოყენებული **iosips** წესი?

_____

მესამე ეტაპი: შეამოწმეთ მუშაობს თუ არა IPS სწორად.

ა. **PC-C**-დან სცადეთ **PC-A**-ს დაპინგვა. არის პინგი წარმატებული? ახსენით

_____

ბ. **PC-A**-დან სცადეთ **PC-C**-ს დაპინგვა. არის პინგი წარმატებული? ახსენით

_____

მეოთხე ეტაპი: Syslog შეტყობინებების დათვალიერება.

ა. დააჭირეთ **Syslog** სერვერს.

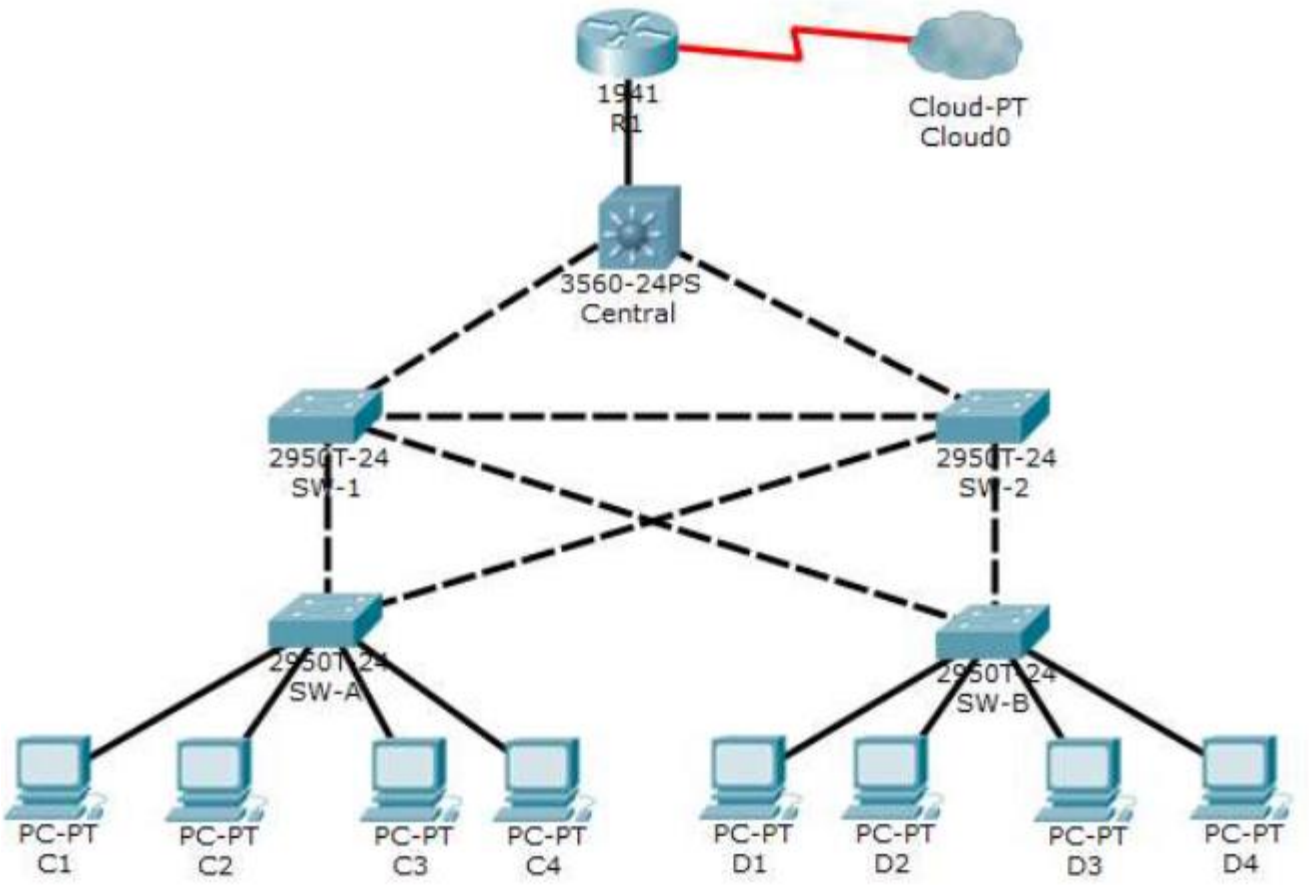
ბ. აირჩიეთ **Services** ჩანართი.

გ. მარცხენა ნავიგაციის მენიუში ლოგ ფაილების სანახავად აირჩიეთ **SYSLOG**.

3.4. ლოკალური ქსელური უსაფრთხოების უზრუნველყოფა

მეორე დონის უსაფრთხოება

ტოპოლოგია



შესასრულებელი ამოცანები:

- ცენტრალური კომუტატორის დანიშვნა, როგორც ძირეული ხიდი (root bridge)
- spanning-tree პარამეტრების დაცვა, STP manipulation შეტევების პრევენციისთვის.
- პორტების უსაფრთხოების ჩართვა CAM table overflow შეტევების პრევენციისთვის.

ზოგადი ინფორმაცია/ სცენარი

ქსელში განხორციელდა რამდენიმე შეტევა, ამიტომ ქსელის ადმინისტრატორმა მოგცათ დავალება, რომ დააკონფიგუროთ მეორე დონის უსაფრთხოება.

ოპტიმალური წარმადობისა და უსაფრთხოებისთვის, ადმინისტრატორს სურს დარწმუნდეს, რომ ძირეული ხიდი არის 3560 ცენტრალური კომუტატორი. Spanning-tree manipulation შეტევების პრევენციისთვის ადმინისტრატორს სურს დარწმუნდეს იმაში, რომ STP პარამეტრები არის დაცული. CAM table overflow შეტევების პრევენციისთვის ადმინისტრატორმა გადაწყვიტა პორტის უსაფრთხოების კონფიგურაცია, რათა შეზღუდოს ის MAC მისამართების რაოდენობა, რომელიც თითოეულმა კომუტატორის პორტმა შეიძლება ისწავლოს. თუ MAC მისამართების რაოდენობა გადააჭარბებს ლიმიტს, ადმინისტრატორს სურს რომ პორტი გაითიშოს.

ყველა კომუტატორი არის წინასწარ კონფიგურირებული, შემდეგნაირად:

- Enable პაროლი: **ciscoenpa55**
- კონსოლის პაროლი: **ciscoconpa55**
- SSH მომხმარებლის სახელი და პაროლი: **SSHadmin / ciscosshpa55**

ნაწილი №1: ძირეული ხიდის (Root Bridge) კონფიგურაცია.

პირველი ეტაპი: მიმდინარე ძირეული ხიდის დადგენა.

Central-დან გაუშვით **show spanning-tree** ბრძანება მიმდინარე ძირეული ხიდის დასადგენად, გამოყენებული პორტებისა და მათი სტატუსების სანახავად.

რომელი კომუტატორია მიმდინარე ძირეული ხიდი (Root Bridge)?

_____

მიმდინარე ძირეული ხიდის საფუძველზე, რა არის Spanning-tree-ს შედეგი? (დახაზეთ spanning-tree ტოპოლოგია).

მეორე ეტაპი: Central-ის დანიშვნა წამყვან ძირეულ ხიდად.

გამოიყენეთ s**panning-tree vlan 1 root primary** ბრძანება და განსაზღვრეთ **Central**-ი, როგორც ძირეული ხიდი (Root Bridge).

მესამე ეტაპი: SW-1-ის დანიშვნა მეორე ძირეულ ხიდად.

დანიშნეთ **SW-1**, როგორც მეორე ძირეული ხიდი, **spanning-tree vlan 1 root secondary** ბრძანების გამოყენებით.

მეოთხე ეტაპი: spanning-tree კონფიგურაციის შემოწმება.

გაუშვით **show spanning-tree** ბრძანება, იმის შესამოწმებლად არის თუ არა **Central**-ი ძირეული ხიდი (Root Bridge).

Central# **show spanning-tree**

VLAN0001

Spanning tree enabled protocol ieee

Root ID Priority 24577

Address 00D0.D31C.634C

This bridge is the root

Hello Time 2 sec Max age 20 sec Forward Delay 15 sec

რომელი კომუტატორის მიმდინარე ძირეული ხიდი (root bridge)?

_____

ახალი ძირეული ხიდის საფუძველზე, რა არის Spanning-tree-ს შედეგი? (დახაზეთ spanning-tree ტოპოლოგია).

ნაწილი №2: STP შეტევებისაგან დაცვა

დაიცავით STP პარამეტრები STP manipulation შეტევების პრევენციისათვის.

პირველი ეტაპი: PortFast-ის ჩართვა ყველა წვდომის პორტებზე.

PortFast არის კონფიგურებული დაშვების პორტებზე, რომელიც დაკავშირებულია ერთ სამუშაო სადგურთან ან სერვერთან, რათა მათ ჰქონდეთ საშუალება გახდნენ აქტიურები უფრო სწრაფად. **SW-A** და **SW-B**-ს დაკავშირებულ წვდომის პორტზე გამოიყენეთ **spanning-tree portfast** ბრძანება.

მეორე ეტაპი: BPDU დაცვის ჩართვა ყველა წვდომის პორტზე.

BPDU დაცვა არის ფუნქცია, რომელსაც შეუძლია დახმარება თაღლითი (rogue) კომუტატორების და წვდომის პორტებზე სპუფინგის პრევენციისათვის. ჩართეთ BPDU დაცვა **SW-A** და **SW-B** წვდომის პორტებზე.

შენიშვნა: Spanning-tree BPDU დაცვა შეიძლება იქნას ჩართული თითოეულ პორტზე ინდივიდუალურად **spanning-three bpduguard enable** ბრძანებით ინტერფეისის კონფიგურაციის რეჟიმში ან **spanning-tree portfast bpduguard default** ბრძანებით გლობალური კონფიგურაციის რეჟიმში.

მესამე ეტაპი: root დაცვის ჩართვა.

Root დაცვა შეიძლება იქნას ჩართული კომუტატორის ყველა პორტზე, რომლებიც არ არიან root პორტები. ის კარგადაა მორგებული პორტებზე, რომელიც უერთდება სხვა არა root კომუტატორებს. გამოიყენეთ **show spannign-three** ბრძანება თოთოეულ კომუტატორზე root პორტის ადგილის დასადგენად.

SW-1 კომუტატორზე ჩართეთ root დაცვა F0/23 და F0/24 პორტებზე. **SW-2** კომუტატორზე ასევე ჩართეთ root დაცვა F0/23 და F0/24 პორტებზე.

ნაწილი №3: პორტის უსაფრთხოების კონფიგურაცია და გამოუყენებელი პორტების გათიშვა.

პირველი ეტაპი: ბაზისური პორტის უსაფრთხოების ჩართვა ყველა პორტზე, რომელზეც დაკავშირებულია ჰოსტი მოწყობილობები.

ეს პროცედურა შეიძლება შესრულებულ იქნას **SW-A** და **SW-B**-ს ყველა წვდომის პორტზე. დააყენეთ შესწავლილი MAC მისამართების მაქსიმუმი რაოდენობა **2**-ზე, დაუშვით MAC მისამართების დინამიურად შესწავლა და მომართეთ დარღვევის დროს პორტის გათიშვა - **shutdown**.

შენიშვნა: პორტის უსაფრთხოების ჩასართავად, კომუტატორის პორტი უნდა იყოს კონფიგურირებული, როგორც წვდომის პორტი.

რატომ არ არის პორტის უსაფრთხოება ჩართული იმ პორტებზე, რომლებიც დაკავშირებულია სხვა კომუტატორ მოწყობილობებთან?

_____

_____

_____

მეორე ეტაპი: პორტის უსაფრთხოების შემოწმება.

ა. **SW-A** კომუტატორზე გაუშვით **show port-security interface f0/1** ბრძანება, რათა დავწმუნდეთ რომ პორტის უსაფრთხოება არის კონფიგურირებული.

SW-A# **show port-security interface f0/1**

Port Security	: Enabled
Port Status	: Secure-up
Violation Mode	: Shutdown
Aging Time	: 0 mins
Aging Type	: Absolute
SecureStatic Address Aging	: Disabled
Maximum Mac Addresses	: 2
Total Mac Addresses	: 0
Configured MAC Addresses	: 0
Sticky MAC Addresses	: 0
Last Source Address:Vlan	: 0000.0000.0000 : 0
Security Violation Count	: 0

ბ. **C1**-დან დაპინგეთ **C2** და გაუშვით **show port-security interface f0/1** ბრძანება ისევ, იმის შესამოწმებლად, რომ კომუტატორმა შეისწავლა **C1**-ის MAC მისამართი.

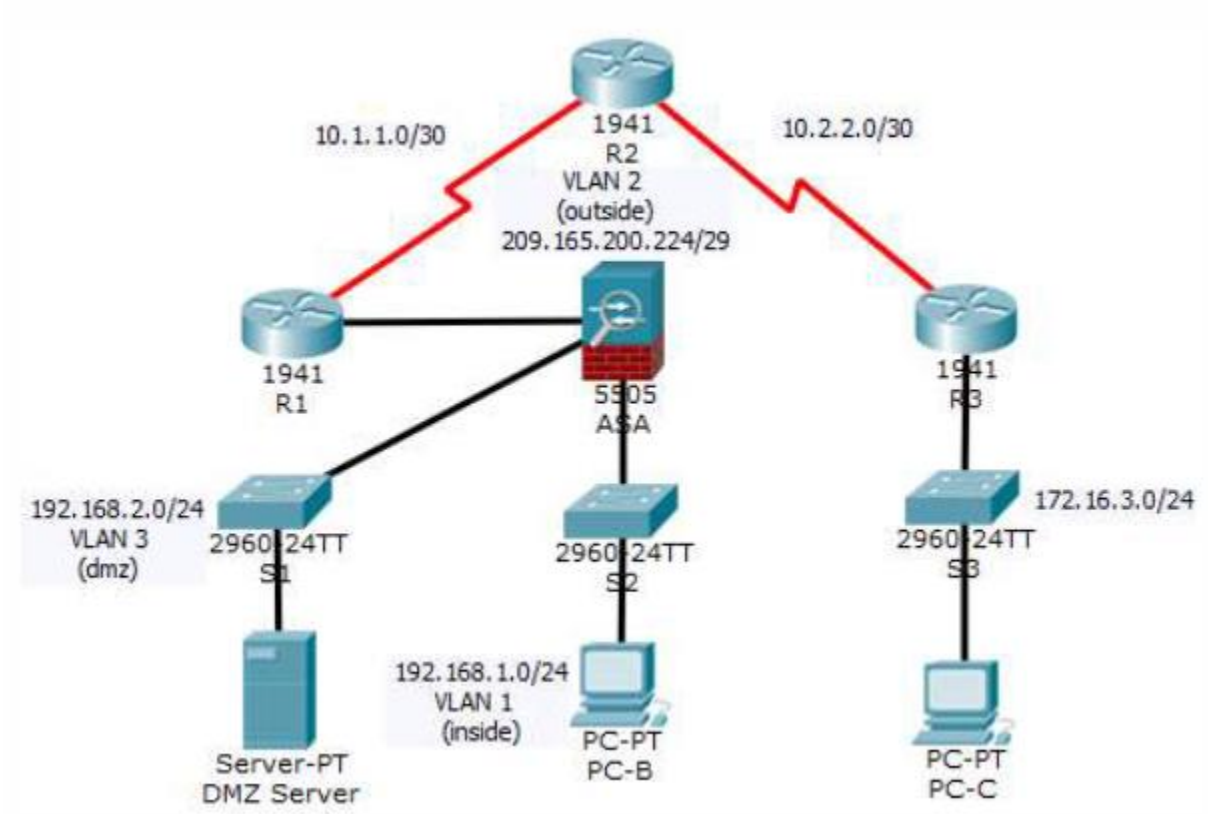
მესამე ეტაპი: გათიშეთ გამოუყენებელი პორტები.

გათიშეთ ყველა ის პორტი, რომელიც ამჯერად არ გამოიყენება.

3.5. Cisco ASA ფაიერვოლის კონფიგურირება

Cisco ASA-ს ბაზისური პარამეტრების და ფაიერვოლის კონფიგურაცია CLI-ის გამოყენებით

ტოპოლოგია



IP მისამართების ცხრილი:

მოწყობილობა	ინტერფეისი	IP მისამართი	ქვექსელის ნიღაბი	ნაგულისხმევი გასასვლელი
R1	G0/0	209.165.200.225	255.255.255.248	არ აქვს
	S0/0/0 (DCE)	10.1.1.1	255.255.255.252	არ აქვს
R2	S0/0/0	10.1.1.2	255.255.255.252	არ აქვს
	S0/0/0/1 (DCE)	10.2.2.2	255.255.255.252	არ აქვს
R3	G0/1	172.16.3.1	255.255.255.0	არ აქვს
	S0/0/1	10.2.2.1	255.255.255.252	არ აქვს

ASA	VLAN 1 (E0/1)	192.168.1.1	255.255.255.0	არ აქვს
ASA	VLAN 2 (E0/0)	209.165.200.226	255.255.255.248	არ აქვს
ASA	VLAN 3 (E0/2)	192.168.2.1	255.255.255.0	არ აქვს
DMZ Server	ქსელის ადაპტერი	192.168.2.3	255.255.255.0	192.168.2.1
PC-B	ქსელის ადაპტერი	192.168.1.3	255.255.255.0	192.168.1.1
PC-C	ქსელის ადაპტერი	172.16.3.3	255.255.255.0	172.16.3.1

შესასრულებელი სამუშაო:

- კავშირის შემოწმება და ASA-ს შესწავლა
- ASA-ს ბაზისური პარამეტრებისა და ინტერფეისის უსაფრთხოების დონეების კონფიგურაცია CLI-ს გამოყენებით
- მარშრუტიზაციის, მისამართების თარგმნის და პოლიტიკების შემოწმება CLI-ის გამოყენებით
- DHCP-ის, AAA-ს და SSH-ის კონფიგურაცია
- DMZ-ს, სტატიკური NAT-ის და ACL-ების კონფიგურაცია

სცენარი

თქვენი კომპანიის ერთი ადგილი დაკავშირებულია ინტერნეტ პროვაიდერთან (ISP). R1 წარმოადგენს CPE მოწყობილობას, რომელიც იმართება პროვაიდერის მიერ. R2 წარმოადგენს შუალედურ ინტერნეტის მარშრუტიზატორს. R3 არის პროვაიდერი, რომელიც დაკავშირებულია ადმინისტრატორთან ქსელის მართვის კომპანიიდან, რომელიც დაქირავებულია თქვენი ქსელის სამართავად. ASA არის უკიდურესი CPE უსაფრთხოების მოწყობილობა, რომელიც აკავშირებს შიდა კორპორატიულ ქსელს და DMZ-ს ინტერნეტ პროვაიდერთან, სანამ უზრუნველყოფს NAT და DHCP სერვისებს შიდა ჰოსტებისათვის. ASA იქნება კონფიგურირებული მართვისთვის ადმინისტრატორის მიერ შიდა ქსელზე და დაშორებული ადმინისტრატორის მიერ. მესამე დონის VLAN ინტერფეისები

უზრუნველყოფენ წვდომას ამ დავალებაში შექმნილ სამ სივრცესთან: შიდა, გარე და DMZ. პროვაიდერმა დანიშნა ღია IP მისამართის სივრცე 209.165.200.224/29, რომელიც გამოყენებული იქნება მისამართების თარგმნისთვის ASA-ზე.

ყველა მარშრუტიზატორი და კომუტატორი არის წინასწარ კონფიგურირებული შემდეგნაირად:

- Enable პაროლი: **ciscoenpa55**
- კონსოლის პაროლი: **ciscoconpa55**
- ადმინისტრატორის მომხმარებლის სახელი და პაროოლი: **admin/adminpa55**

ნაწილი №1: კავშირის შემოწმება და ASA-ს შესწავლა

პირველი ეტაპი: კავშირის შემოწმება

ASA ჯერ არ არის კონფიგურირებული. თუმცა ყველა მარშრუტიზატორი, პერსონალური კომპიუტერი და DMZ სერვერი კონფიგურირებულია. შეამოწმეთ, რომ PC-C-ს შეუძლია მარშრუტიზატორის ნებისმიერი ინტერფეისის დაპინგვა. PC-C-ს არ შეუძლია ASA-ს, PC-B-ს და DMZ სერვერის დაპინგვა.

მეორე ეტაპი: ASA-ს ვერსიის, ინტერფეისების და ლიცენზიის დადგენა.

გამოიყენეთ **show version** ბრძანება ASA მოწყობილობის სხვადასხვა ასპექტების დასადგენად.

მესამე ეტაპი: ფაილური სისტემისა და ფლეშ მეხსიერების შემცველობის დადგენა

ა. შედით პრივილეგირებულ EXEC რეჟიმში. პაროლი არ არის მომართული. დააჭირეთ **Enter** ღილაკს პაროლის მოთხოვნისას.

ბ. გამოიყენეთ **show file system** ბრძანება ASA-ს ფაილური სისტემის საჩვენებლად და დაადგინეთ რომელ ინდექსებს უჭერს მხარს.

გ. ფლეშ მეხსიერების შემცველობის საჩვენებლად გამოიყენეთ **show flash:** ან **show disk0:** ბრძანება.

ნაწილი №2: ASA-ს პარამეტრების და ინტერფეისის უსაფრთხოების კონფიგურაცია CLI-ს გამოყენებით

რჩევა: ბევრი ASA CLI ბრძანება არის მსგავსი, თუ არ არის იგივე, მაინც იყენებენ Cisco IOS CLI-ს. დამატებით, კონფიგურაციის რეჟიმებსა და ქვერეჟიმებს შორის გადასვლა არის იგივე.

პირველი ეტაპი: სახელისა და დომეინის სახელის კონფიგურაცია.

ა. მომართეთ ASA-ს სახელი და დააყენეთ **CCNAS-ASA**.

ბ. დომეინის სახელად დააყენეთ **ccnasecurity.com.**

მეორე ეტაპი: enable რეჟიმის პაროლის კონფიგურაცია

გამოიყენეთ **enable password** ბრძანება პრივილეგირებული EXEC რეჟიმის პაროლის შესაცვლელად **ciscoenpas55**-ით.

მესამე ეტაპი: თარიღისა და დროის დაყენება.

გამოიყენეთ **clock set** ბრძანება თარიღისა და დროის ხელით მომართვისთვის

მეოთხე ეტაპი: შიდა და გარე ინტერფეისების კონფიგურაცია.

ამჯერად თქვენ უნდა მომართოთ მხოლოდ VLAN 1 (შიდა) და VLAN 2 (გარე) ინტერფეისები. VLAN 3 (DMZ)-ს კონფიგურაცია მოხდება ამ დავალების ნაწილ №5-ში.

ა. დააკონფიგურეთ ლოგიკური VLAN 1 ინტერფეისი შიდა ქსელისთვის (192.168.1.0/24) და დააყენეთ უსაფრთხოების დონე მაღალ პარამეტრ - 100-ზე.

CCNAS-ASA (config) # **interface vlan 1**

CCNAS-ASA (config-if) # **nameif inside**

CCNAS-ASA (config-if) # **ip address 192.168.1.1 255.255.255.0**

CCNAS-ASA (config-if) # **security-level 100**

ბ. ლოგიკური VLAN 2 ინტერფეისის შექმნა გარე ქსელისთვის (209.165.200.224/29), უსაფრთხოების დონის მომართვა ყველაზე დაბალ პარამეტრ 0-ზე და VLAN 2 ინტერფეისის ჩართვა.

CCNAS-ASA (config-if) # **interface vlan 2**

CCNAS-ASA (config-if) # **nameif outside**

CCNAS-ASA (config-if) # **ip address 209.165.200.226 255.255.255.248**

CCNAS-ASA (config-if) # **security-level 0**

გ. გამოიყენეთ ქვემოთ მოცემული ვერიფიკაციის ბრძანებები, თქვენი კონფიგურაციის შესამოწმებლად:

1) ASA-ს ყველა ინტერფეისის მდგომარეობის სანახავად გამოიყენეთ **show interface ip brief** ბრძანება. **შენიშვნა:** მოცემული ბრძანება განსხვავდება IOS-ის **show ip interface brief**-სგან. თუ ნებისმიერი ფიზიკური თუ ლოგიკური წინასწარ კონფიგურირებული ინტერფეისი არ არის ჩართულ მდგომარეობაში, გაგრძელებამდე აუცილებელია პრობლემის მოგვარება.

   რჩევა: ASA **show** ბრძანებების უმრავლესობა, მათ შორის **ping**, **copy**, და სხვა, შეიძლება გაშვებულ იქნას ნებისმიერი კონფიგურაციის რეჟიმიდან სწრაფად, **do** ბრძანების გარეშე.

2) გამოიყენეთ **show ip address** ბრძანება მესამე დონის VLAN ინტერფეისებზე ინფორმაციის გამოსატანად.

3) გამოიყენეთ **show switch vlan** ბრძანება, ASA-ზე შიდა და გარე კონფიგურირებული VLAN-ებისა და მინიჭებული პორტების სანახავად.

მეხუთე ეტაპი: ASA-ს კავშირის შემოწმება

ა. თქვენ უნდა შეძლოთ ASA-ს შიდა ინტერფეისის მისამართის (192.168.1.1) დაპინგვა PC-B-დან. თუ პინგი არ დასრულდა წარმატებით, მოაგვარეთ კონფიგურაციის პრობლემა.

ბ. PC-B-დან დაპინგეთ VLAN 2-ის (გარე) ინტერფეისი 209.165.200.226 IP მისამართზე. თქვენ ვერ უნდა შეძლოთ მოცემული მისამართის დაპინგვა.

ნაწილი №3: მარშრუტიზაციის, მისამართების თარგმნის კონფიგურაცია და პოლიტიკების შემოწმება CLI-ს გამოყენებით.

პირველი ეტაპი: სტატიკური ნაგულისხმევი მარშრუტის კონფიგურაცია ASA-სთვის.

დააკონფიგურეთ ნაგულისხმევი სტატიკური მარშრუტი ASA-ზე, რათა დაუკავშირდეს გარე ქსელს.

ა. შექმენით „ოთხმაგი ნული“ ნაგულისხმევი მარშრუტი **route** ბრძანების გამოყენებით, მიანიჭეთ ის ASA-ს გარე ინტერფეისს და მიუთითეთ R1 G0/0 IP მისამართი (209.165.200.225) როგორც ბოლო მხარის გასასვლელი.

CCNAS-ASA (config) # **route outside 0.0.0.0 0.0.0.0 209.165.200.225**

ბ. გაუშვით **show route** ბრძანება იმის შესამოწმებლად არის თუ არა ნაგულისხმევი სტატიკური მარშრუტი ASA-ს მარშრუტიზაციის ცხრილში.

გ. შეამოწმეთ, რომ ASA-ს შეუძლია R1 S0/0/0 10.1.1.1 IP მისამართის დაპინგვა. თუ პინგი წარუმატებლად დასრულდა, აუცილებლად მოაგვარეთ პრობლემა.

მეორე ეტაპი: PAT-ის გამოყენებით მისამართების თარგმნისა და ქსელის ობიექტების კონფიგურაცია.

ა. შექმენით ქსელის ობიექტი **inside-net** და მიანიჭეთ ატრიბუტები მას **subnet** და **nat** ბრძანებების გამოყენებით.

CCNAS-ASA (config) # **object network inside-net**

CCNAS-ASA (config-network-object) # **subnet 192.168.1.0 255.255.255.0**

CCNAS-ASA (config-network-object) # **nat (inside, outside) dynamic interface**

CCNAS-ASA (config-network-object) # **end**

ბ. ASA ყოფს კონფიგურაციას ობიექტის ნაწილში, რომელიც განსაზღვრავს ქსელს თარგმნისთვის და მოქმედ **nat** ბრძანების პარამეტრებს. ისინი ჩნდებიან გაშვებული კონფიგურაციის ორ განსახვავებულ ადგილას. დაათვალიერეთ NAT ობიექტის კონფიგურაცია **show run** ბრძანების გამოყენებით.

გ. PC-B-დან სცადეთ დაპინგოთ R1 G0/0 ინტერფეისი 209.165.200.225 მისამართზე. პინგი უნდა ჩავარდეს.

დ. გაუშვით **show nat** ბრძანება ASA-ზე თარგმნილი და სათარგმნი ჰიტების (hits) სანახავად. მიაქციეთ ყურადღება, რომ PC-B-დან პინგები ოთხი ითარგმნა, ხოლო ოთხი - არა. გამავალი პინგები (ექოები) ითარგმნა და გაიგზავნა ადრესატთან. დაბრუნებილი ექო მოთხოვნები დაბლოკილ იქნა ფაიერვოლის პოლიტიკის მიერ. თქვენ უნდა დააკონფიგუროთ ნაგულისხმევი შემოწმების პოლიტიკა (Inspection Policy), რათა დაშვებულ იქნას ICMP ამ დავალების მესამე ნაწილში.

მესამე ეტაპი: ნაგულისხმევი MPF აპლიკაციის შემოწმების გლობალური სერვისის პოლიტიკის შეცვლა.

გამოყენებითი დონის შემოწმებისა და სხვა დამატებითი ოპციებისთვის ASA-ზე ხელმისაწვდომია MPF-ი.

Packet Tracer-ის ASA მოწყობილობას არ აქვს ნაგულისხმევად განთავსებული MPF პოლიტიკების რუკა. როგორც მოდიფიკაცია ჩვენ შეგვიძლია შევქმნათ ნაგულისხმევი პოლიტიკის რუკა, რომელიც შეასრულებს შიგნიდან გარეთ გამავალი ტრაფიკის შემოწმებას. როდესაც სწორადაა კონფიგურირებული მხოლოდ შიდა ქსელიდან გასულ ტრაფიკს შეუძლია უკან დაბრუნდეს გარე ინტერფეისებიდან. თქვენ უნდა დაამატოთ ICMP-ი შემოწმების სიას.

ა. შექმენით კლასს რუკა (class-map), პოლიტიკის რუკა (policy-map) და სერვის-პოლიტიკა (service-policy). დაამატეთ ICMP ტრაფიკის შემოწმება პოლიტიკების რუკის სიაში ქვემოთ მოცემული ბრძანებების გამოყენებით.

CCNAS-ASA (config) # **class-map inspection_default**

CCNAS-ASA (config-cmap) # **match default-inspection-traffic**

CCNAS-ASA (config-cmap) #**exit**

CCNAS-ASA (config) # **policy-map global_policy**

CCNAS-ASA (config-pmap) # **class inspection_default**

CCNAS-ASA (config-pmap-c) # **inspect icmp**

CCNA-ASA (config-pmap-c) # **exit**

CCNAS-ASA (config) # **service-policy global_policy global**

ბ. PC-B-დან სცადეთ R1 G0/0 ინტერფეისის დაპინგვა 209.165.200.225 IP მისამართზე. პინგი ახლა უკვე უნდა დასრულდეს წარმატებულად, რადგან ICMP ტრაფიკი ახლა უკვე შემოწმებულია და ტრაფიკის ლეგიტიმურად დაბრუნება დაშვებულია. თუ პინგი ჩავარდა, მოაგვარეთ თქვენი კონფიგურაციის პრობლემა.

ნაწილი №4: DHCP-ის, AAA-ს და SSH-ის კონფიგურაცია

პირველი ეტაპი: ASA-ს როგორც DHCP სერვერის კონფიგურაცია

ა. DHCP pool-ის კონფიგურაცია და მისი ჩართვა ASA-ს შიდა ინტერფეისზე.

CCNAS-ASA (config) # **dhcpd address 192.168.1.5-192.168.1.36 inside**

ბ. (არჩევითი) მიუთითეთ DNS სერვერის IP მისამართი, რომელიც მიეცემა კლიენტებს.

CCNAS-ASA (config) # **dhcpd dns 209.165.201.2 interface inside**

გ. ჩართეთ DHCP ASA-ში რათა მოუსმინოს DHCP კლიენტების მოთხოვნებს ჩართულ ინტერფეისზე (შიდა).

CCNAS-ASA (config) # **dhcpd enable inside**

დ. შეცვალეთ PC-B-ს სტატიკური IP მისამართი DHCP კლიენტით და შეამოწმეთ მიიღებს თუ არა IP დამისამართების ინფორმაციას. ნებისმიერი პრობლემის არსებობის შემთხვევაში აუცილებელია პრობლემის მოგვარება.

მეორე ეტაპი: AAA-ს კონფიგურაცია ლოკალური მონაცემთა ბაზის აუთენტიფიკაციისთვის გამოსაყენებლად.

ა. ლოკალური მომხმარებლის სახელად განსაზღვრეთ **admin**, **username** ბრძანების შეყვანით. მიანიჭეთ **adminpa55** პაროლი.

CCNAS-ASA (config) # **username admin password adminpa55**

ბ. დააკონფიგურეთ AAA ASA-ს ლოკალური მონაცემთა ბაზის გამოყენებლად SSH მომხმარებლის აუთენტიფიკაციისთვის.

CCNAS-ASA (config) # **aaa authentication ssh console LOCAL**

მესამე ეტაპი: დაშორებული წვდომის მართვის კონფიგურაცია ASA-ზე.

ASA შეიძლება დაკონფიგურებული ისე, რომ მიიღოს კავშირი ერთი ჰოსტიდან ან ჰოსტების დიაპაზონიდან, შიდა ან გარე ქსელზე. მოცემულ ეტაპზე გარე ქსელის ჰოსტებს შეუძლიათ გამოიყენონ მხოლოდ SSH ASA-სთან დაკავშირებისთვის. SSH სესიები შეიძლება გამოყენებულ იქნას შიდა ქსელიდან ASA-სთან წვდომისთვის.

ა. შექმენით RSA წყვილი გასაღები, რომელიც მოითხოვება SSH კავშირების მხარდაჭერისთვის. რადგან ASA მოწყობილობას უკვე აქვს განთავსებული RSA გასაღებები, შეიყვანეთ **no** მათი შეცვლის მოთხოვნისას.

CCNAS-ASA (config) # **crypto key generate rsa modulus 1024**

WARNING: You have a RSA keypair already defined named <Default-RSA-Key>.

Do you really want to replace them? [yes/no]: **no**

ERROR: Failed to create ne RSA keys named <Default-RSA-Key>

ბ. დააკონფიგურეთ ASA რათა დაუშვას SSH შეერთებები ნებისმიერი ჰოსტიდან შიდა ქსელზე (192.168.1.0/24) და დაშორებული მართვის ჰოსტიდან ფილიალის ოფისში (172.16.3.3) გარე ქსელზე. მომართეთ SSH ლოდინის დრო 10 წუთზე (ნაგულისხმევი დრო არის 5 წუთი).

CCNAS-ASA (config) # **ssh 192.168.1.0 255.255.255.0 inside**

CCNAS-ASA (config) # **ssh 172.16.3.3 255.255.255.255 outside**

CCNAS-ASA (config) # **ssh timeout 10**

გ. შექმენით SSH სესია PC-C-დან ASA (209.165.200.226)-ზე. წარუმატებლობის შემთხვევაში მოაგვარეთ პრობლემა.

PC> **ssh -1 admin 209.165.200.226**

დ. შექმენით SSH სესია PC-B-დან ASA (192.168.1.1)-ზე. წარუმატებლობის შემთხვევაში მოაგვარეთ პრობლემა.

PC> **ssh -1 admin 192.168.1.1**

ნაწილი №5: DMZ-ს, სტატიკური NAT-ის და ACL-ების კონფიგურაცია

R1 G0/0 და ASA გარე ინტერფეისი უკვე იყენებენ 209.165.200.225-ს და .226-ს შესაბამისად. თქვენ უნდა გამოიყენოთ გარე მისამართი 209.165.200.227 და სტატიკური NAT-ი, მისამართების თარგმნის წვდომის უზრუნველყოფისთვის სერვერზე.

პირველი ეტაპი: DMZ VLAN 3 ინტერფეისის კონფიგურაცია ASA-ზე.

ა. დააკონფიგურეთ DMZ VLAN 3, რომელიც არის იქ სადაც საერთო წვდომის ვებ სერვერი დადგება. მიანიჭეთ მას IP მისამართი 192.168.2.1/24, სახელი **dmz** და განუსაზღვრეთ უსაფრთხოების დონე 70-ზე. რადგან სერვერს არ სჭირდება შიდა მომხმარებლებთან კავშირის ინიციალიზაცია, გათიშეთ VLAN 1-თან გადაგზავნა.

CCNAS-ASA (config) # **interface vlan 3**

CCNAS-ASA (config-if) # **ip address 192.168.2.1 255.255.255.0**

CCNAS-ASA (config-if) # **no forward interface vlan 1**

CCNAS-ASA (config-if) # **nameif dmz**

INFO: Security level for “dmz” set to 0 by default.

CCNAS-ASA (config-if) # **security-level 70**

ბ. მიაკუთვნეთ ASA-ს E0/2 ფიზიკური ინტერფეისი DMZ VLAN 3-ს და ჩართეთ ინტერფეისი.

CCNAS-ASA (config-if) # **interface Ethernet0/2**

CCNAS-ASA (config-if) # **switchport access vlan 3**

გ. გამოიყენეთ ქვემოთ მოცემული ვერიფიკაციის ბრძანებები თქვენი კონფიგურაციის შესამოწმებლად:

1) გამოიყენეთ **show interface ip brief** ბრძანება ASA-ს ყველა ინტერფეისის მდგომარეობის საჩვენებლად.

2) გამოიყენეთ **show ip address** ბრძანება მესამე დონის VLAN ინტერფეისებზე ინფორმაციის გამოსატანად.

3) გამოიყენეთ **show switch vlan** ბრძანება კონფიგურირებული შიდა და გარე VLAN-ების და მინიჭებული პორტების სანახავად ASA-ზე.

მეორე ეტაპი: DMZ სერვერის სტატიკური NAT-ის კონფიგურაცია ქსელის ობიექტის გამოყენებით.

დააკონფიგურეთ ქსელის ობიექტი **dmz-server** სახელით და მიანიჭეთ მას სტატიკური IP მისამართი DMZ სერვერზე (192.168.2.3). სანამ იმყოფებით ობიექტის განსაზღვრის რეჟიმში, გამოიყენეთ **nat** ბრძანება იმის მისათითებლად, რომ მოცემული ობიექტი გამოყენებულია DMZ მისამართის სათარგმნად გარე მისამართზე სტატიკური NAT-ის გამოყენებით და მიუთითეთ საერთო (Public) თარგმნილი მისამართი 209.165.200.227.

CCNAS-ASA (config) # **object network dmz-server**

CCNAS-ASA (config-network-object) # **host 192.168.2.3**

CCNAS-ASA (config-network-object) # **nat (dmz, outside) static 209.165.200.227**

CCNAS-ASA (config-network-object) # **exit**

მესამე ეტაპი: ACL-ის კონფიგურაცია ინტერნეტიდან DMZ სერვერთან წვდომის დაშვებისათვის.

დააკონფიგურეთ სახელდებული წვდომის სია **OUTSIDE-DMZ**, რომელიც დაუშვებს TCP პროტოკოლს მე-80 პორტზე, ნებისმიერი გარე ჰოსტიდან შიდა DMZ სერვერის IP მისამართთან. გამოიყენეთ წვდომის სია ASA-ს გარე ინტერფეისზე „IN" მიმართულებისთვის.

CCNAS-ASA (config) # **access-list OUTSIDE-DMZ permit icmp any host 192.168.2.3**

CCNAS-ASA (config) # **access-list OUTSIDE-DMZ permit tcp any host 192.168.2.3 eq 80**

CCNAS-ASA (config) # **access-group OUTSIDE-DMZ in interface outside**

შენიშვნა: IOS ACL-ებისაგან განსხვავებით, ASA ACL დაშვების მდგომარეობამ უნდა დაუშვას წვდომა შიდა კერძო DMZ მისამართთან. გარე ჰოსტები სერვერთან წვდომას ახორციელებენ თავიანთი სტატიკური NAT მისამართებით, ASA თარგმნის მას შიდა ჰოსტის IP მისამართად და შემდეგ იყენებს ACL-ს.

პრაქტიკული სავარჯიშო

1. შეასრულეთ გაფართოებული დანომრილი ACL-ების კონფიგურაცია და შემოწმება;
2. შეასრულეთ გაფართოებული სახელდებული ACL-ების კონფიგურაცია და შემოწმება;
3. მოახდინეთ IOS-ის შეღწევის პრევენციის სისტემის (IPS) კონფიგურაცია ბრძანებათა ინტერფეისის (CLI) გამოყენებით;
4. დანიშნეთ ცენტრალური კომუტატორის ძირეული ხიდად (root bridge);
5. დაიცავით spanning-tree პარამეტრები STP manipulation შეტევების პრევენციისთვის;
6. ჩართეთ პორტების უსაფრთხოება CAM table overflow შეტევების პრევენციისთვის;
7. განახორციელეთ ASA-ს ბაზისური პარამეტრებისა და ინტერფეისის უსაფრთხოების დონეების კონფიგურაცია CLI-ს გამოყენებით;
8. შეასრულეთ მარშრუტიზაციის, მისამართების თარგმნის და პოლიტიკების შემოწმება CLI-ის გამოყენებით;
9. შეასრულეთ DHCP-ის, AAA-ს და SSH-ის კონფიგურაცია;
10. განახორციელეთ DMZ-ს, სტატიკური NAT-ის და ACL-ების კონფიგურაცია

ცოდნის შეფასება

სტუდენტებს მიეცემათ პრაქტიკული დავალება

- შეასრულონ ACL-ის კონფიგურაცია FTP-ს და ICMP-ის დასაშვებად, მოახდინონ ACL-ის გამოყენება სწორ ინტერფეისზე, ტრაფიკის ფილტრაციისთვის, შეასრულონ ACL-ის კონფიგურაცია HTTP წვდომისა და ICMP-ის დასაშვებად, განახორციელონ ACL-ის მოქმედების შემოწმება;
- შეასრულონ IOS IPS-ის ჩართვა, ლოგირების კონფიგურაცია, IPS ხელწერის შეცვლა, IPS-ის შემოწმება;
- მოახდინონ ძირეული ხიდის (Root Bridge) კონფიგურაცია, spanning-tree კონფიგურაციის შემოწმება, STP შეტევებისაგან დაცვა, PortFast-ის ჩართვა ყველა

წვდომის პორტებზე, BPDU დაცვის ჩართვა ყველა წვდომის პორტზე, root დაცვის ჩართვა, პორტის უსაფრთხოების კონფიგურაცია და გამოუყენებელი პორტების გათიშვა, ბაზისური პორტის უსაფრთხოების ჩართვა ყველა პორტზე, რომელზეც დაკავშირებულია ჰოსტი მოწყობილობები, პორტის უსაფრთხოების შემოწმება;

- განახორციელონ კავშირის შემოწმება და ASA-ს შესწავლა, ASA-ს პარამეტრების და ინტერფეისის უსაფრთხოების კონფიგურაცია CLI-ს გამოყენებით, მარშრუტიზაციის, მისამართების თარგმნის კონფიგურაცია და პოლიტიკების შემოწმება CLI-ს გამოყენებით, სტატიკური ნაგულისხმევი მარშრუტის კონფიგურაცია ASA-სთვის, PAT-ის გამოყენებით მისამართების თარგმნისა და ქსელის ობიექტების კონფიგურაცია, ნაგულისხმევი MPF აპლიკაციის შემოწმების გლობალური სერვისის პოლიტიკის შეცვლა;
- შეასრულონ DHCP-ის, AAA-ს და SSH-ის კონფიგურაცია, ASA-ს როგორც DHCP სერვერის კონფიგურაცია, AAA-ს კონფიგურაცია ლოკალური მონაცემთა ბაზის აუთენტიფიკაციისთვის გამოსაყენებლად, დაშორებული წვდომის მართვის კონფიგურაცია ASA-ზე, DMZ-ს, სტატიკური NAT-ის და ACL-ების კონფიგურაცია, DMZ სერვერის სტატიკური NAT-ის კონფიგურაცია ქსელის ობიექტის გამოყენებით, ACL-ის კონფიგურაცია ინტერნეტიდან DMZ სერვერთან წვდომის დაშვებისათვის.

შემფასებელი აკვირდება შესაფასებელ პირის მუშაობას პროფესიული სტანდარტით (პროგრამით / მოდულით) განსაზღვრული ამოცანების შესრულების პროცესში. დაკვირვება ხორციელდება კომპიუტერებით აღჭურვილ ლაბორატორიაში, სადაც შესაფასებელი პირი პრაქტიკულ საქმიანობას ეწევა. შემფასებელმა წინასწარ უნდა დაგეგმოს დაკვირვების პროცესი, იმის დასადგენად, თუ რამდენად სწორად იყენებს შესაფასებელი პირი ცოდნას, უნარებსა და ყველა რესურსს შედეგის მისაღწევად.

შეფასება განხორციელდება პროცესზე დაკვირვებით, წინასწარ განსაზღვრული შეფასების ინდიკატორების საფუძველზე.

დავალების ნიმუში და შეფასების რუბრიკა

პროცესზე დაკვირვება

- *შეასრულოს ACL-ების კონფიგურაცია, IOS-ის შეღწევის პრევენციის სისტემის (IPS) კონფიგურაცია, spanning-tree პარამეტრების დაცვა, შეასრულა პორტების უსაფრთხოების ჩართვა*
- შეასრულოს ASA-ს ბაზისური პარამეტრებისა და ინტერფეისის უსაფრთხოების დონეების კონფიგურაცია, DHCP-ის, AAA-ს და SSH-ის კონფიგურაცია, DMZ-ს, სტატიკური NAT-ის და ACL-ების კონფიგურაცია.

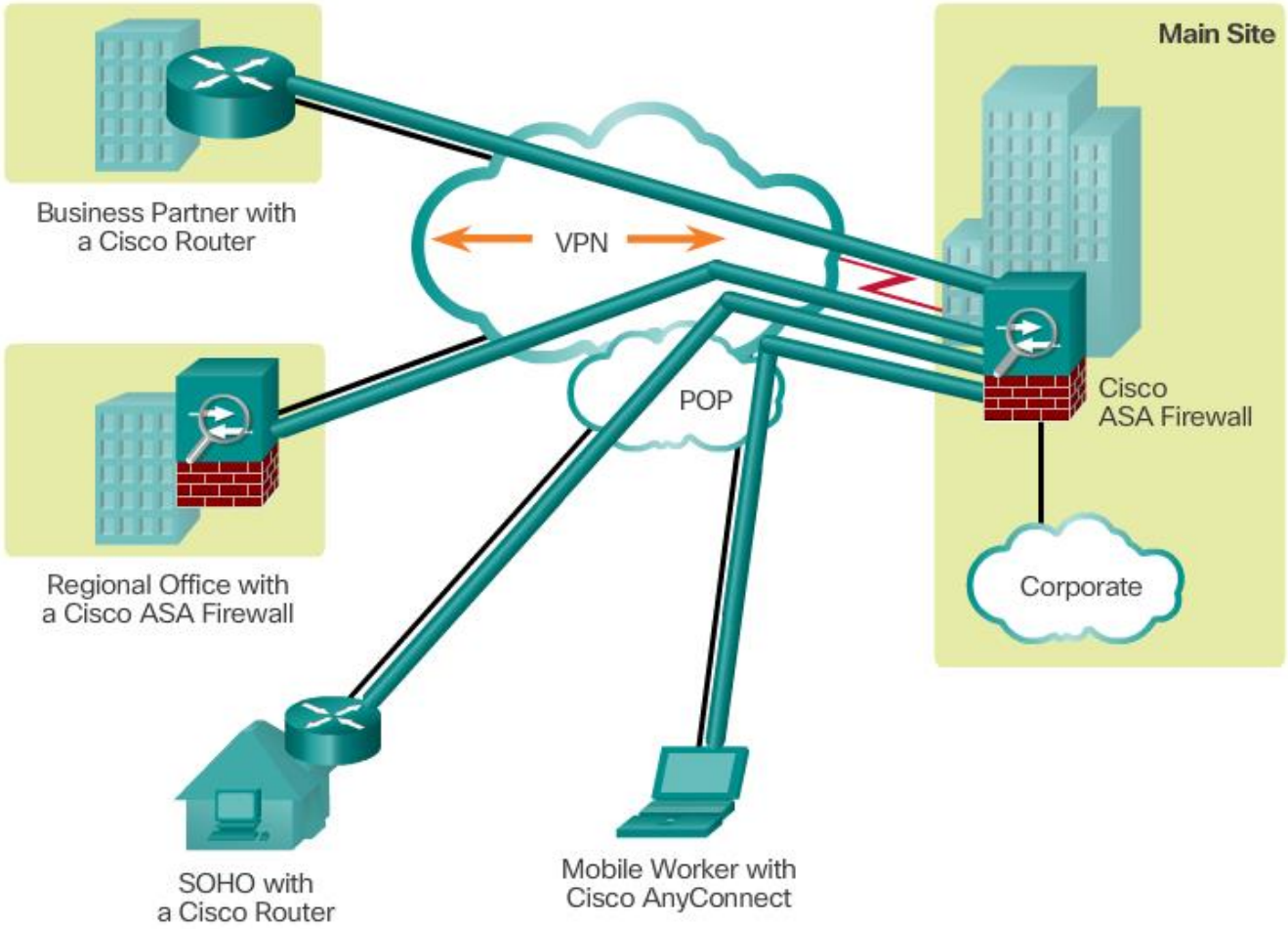
სწავლის შედეგი	N	დასახელება	შეფასება	
			კი	არა
ქსელური აპარატურის და ტექნოლოგიების უსაფრთხოება	1.	შეასრულა გაფართოებული დანომრილი ACL-ების კონფიგურაცია და შემოწმება		
	2.	შეასრულა გაფართოებული სახელდებული ACL-ების კონფიგურაცია და შემოწმება		
	3.	შეასრულა IOS-ის შეღწევის პრევენციის სისტემის (IPS) კონფიგურაცია ბრძანებათა ინტერფეისის (CLI) გამოყენებით		
	4.	შეასრულა ცენტრალური კომუტატორის დანიშვნა ძირეულ ხიდად (root bridge)		
	5.	შეასრულა spanning-tree პარამეტრების დაცვა STP manipulation შეტევების პრევენციისთვის		
	6.	შეასრულა პორტების უსაფრთხოების ჩართვა CAM table overflow შეტევების პრევენციისთვის		
	7.	შეასრულა ASA-ს ბაზისური პარამეტრებისა და ინტერფეისის უსაფრთხოების დონეების კონფიგურაცია CLI-ს გამოყენებით		
	8.	შეასრულა მარშრუტიზაციის, მისამართების თარგმნის და პოლიტიკების შემოწმება CLI-ის გამოყენებით		
	9.	შეასრულა DHCP-ის, AAA-ს და SSH-ის კონფიგურაცია		
	10.	შეასრულა DMZ-ს, სტატიკური NAT-ის და ACL-ების კონფიგურაცია		

სწავლის შედეგი ჩაითვლება მიღწეულად თუ სტუდენტმა შეძლო შედეგის მინიმუმ 8 პუნქტის შესრულება.

4. შიდა და გარე კომუნიკაციები, უსაფრთხო კავშირები და WAN ჩართვები (VPN, NAT/PAT, MPLS Applications)

4.1. ვირტუალური დაცული ქსელების (VPNs) საფუძვლები

ორგანიზაციებს სჭირდებათ დაცული, საიმედო და რენტაბელური გზები სხვადასხვა ქსელების ურთიერთდაკავშირებისთვის, როგორიცაა ფილიალებისა და მომწოდებლების დაშვება კორპორაციების სათაო ოფისების ქსელთან დასაკავშირებლად. გარდა ამისა დაშორებულად მომუშავე თანამშრომლების რიცხვის გაზრდასთან ერთად, კომპანია ზრდის უსაფრთხოების, საიმედოობისა და რენტაბელურობის მოთხოვნას მცირე ოფისი/სახლი ოფისებში (SOHO) და სხვა დაშორებულ ადგილებში მომუშავე თანამშრომლების დასაკავშირებლად კორპორატიული ადგილების რესურსებთან.



სურათზე ნაჩვენებია ტოპოლოგია, რომელსაც იყენებს ახლანდელი ქსელები დაშორებულ ადგილებთან დასაკავშირებლად. ზოგიერთ შემთხვევაში დაშორებული ადგილები უკავშირდებიან მხოლოდ სათაო ოფისებს, სხვა შემთხვევაში დაშორებული ოფისები უერთდებიან დამატებით ადგილებს.

ორგანიზაციები იყენებენ ვირტუალურ დაცულ ქსელებს ერთმანეთთან დაკავშირებული დაცული ქსელური შეერთების შესაქმნელად მესამე მხარის ქსელებზე, როგორიცაა ინტერნეტი ან ექსტრანეტი. ტუნელი გამორიცხავს მანძილის ბარიერს და საშუალებას აძლევს დაშორებულ მომხმარებლებს განახორციელონ წვდომა ცენტრალური ოფისის ქსელურ რესურსებთან. VPN არის დაცული ქსელი, რომელიც შექმნილია ტუნელირების გამოყენებით ღია ქსელზე, როგორც წესი ინტერნეტზე. VPN არის კომუნიკაციების გარემო, რომელშიც წვდომა არის მკაცრად კონტროლირებული განსაზღვრული ინტერესის მქონე საზოგადოების თანაბარუფლებიანი კავშირების დაშვება.

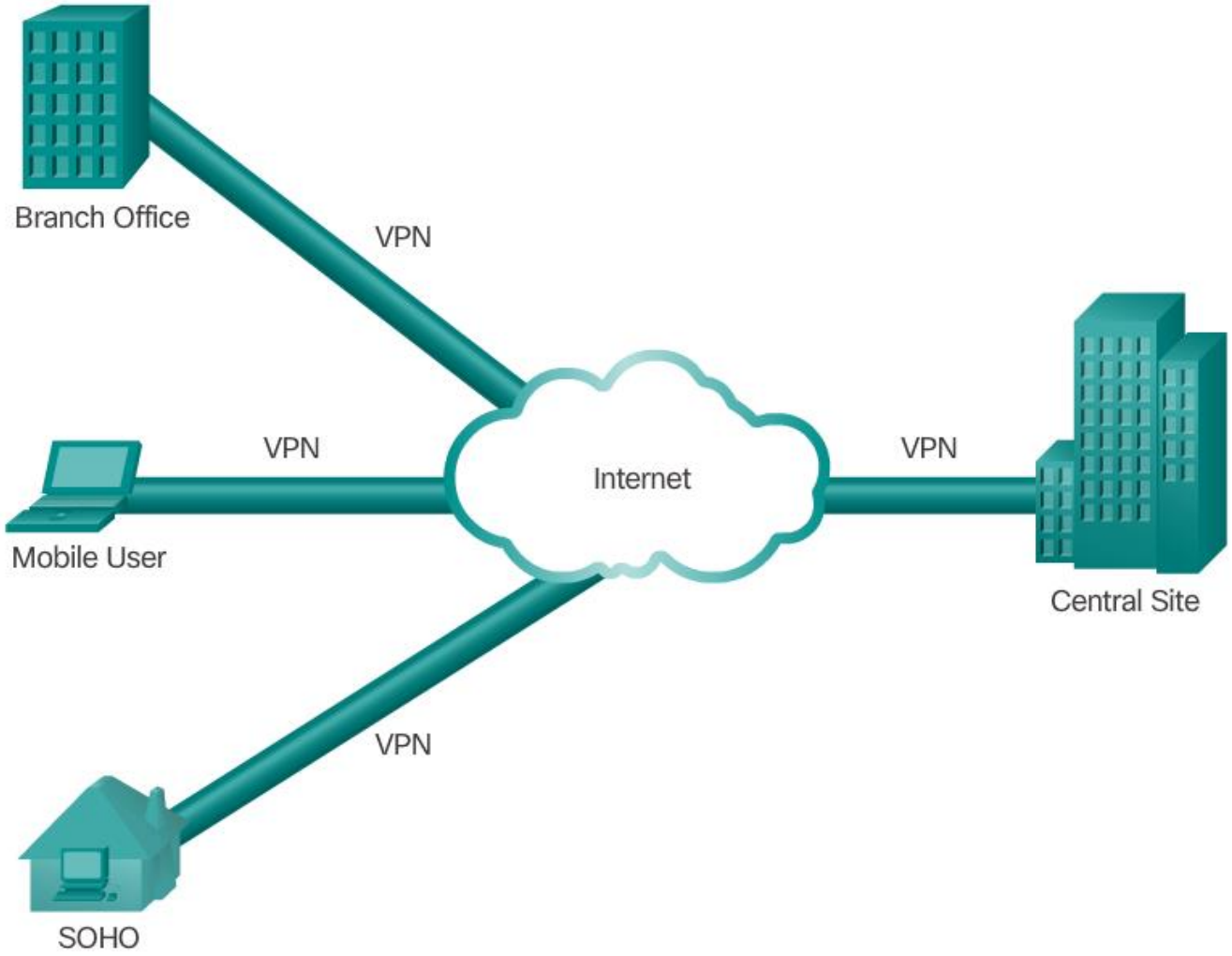
პირველი ვირტუალური დაცული ქსელები იყო მკაცრი IP ტუნელები, რომლებიც არ შეიცავდა მონაცემების აუთენტიფიკაციას ან შიფრაციას. მაგალითად საერთო მარშრუტიზაციის ენკაპსულაცია - Generic Routing Encapsulation (GRE) არის Cisco-ს მიერ განვითარებული ტუნელირების პროტოკოლი, რომელსაც შეუძლია სხვადასხვა სახის ქსელური პროტოკოლის პაკეტის ტიპების ენკაპსულაცია IP ტუნელების შიგნით. ეს ქმნის Cisco-ს მარშრუტიზატორების დაშორებულ წერტილებთან ვირტუალურ ერთმანეთთან დაკავშირებულ შეერთებას IP ქსელებს შორის.

დღეს VPN-ის უსაფრთხო დანერგვა შიფრაციით, როგორიცაა IPsec VPN-ები არის ის რაც გვინდა ვირტუალური დაცული ქსელებისაგან.

ვირტუალური დაცული ქსელების დასანერგად აუცილებელია VPN გასასვლელი (Gateway). VPN გასასვლელი შეიძლება იყოს მარშრუტიზატორი, ფაიერვოლი ან Cisco-ს ადაპტირებული უსაფრთხოების მოწყობილობა - Adaptive Security Appliance (ASA). ASA არის ავტონომიური ფაიერვოლ მოწყობილობა, რომელიც ერთი პროგრამული უზრუნველყოფის იმიჯში აერთიანებს ფაიერვოლის, VPN კონცენტრატორისა და შეტევების პრევენციის ფუნქციებს.

4.1.1 ვირტუალური დაცული ქსელების უპირატესობები

როგორც სურათზეა ნაჩვენები VPN იყენებს ვირტუალურ კავშირებს, რომელიც ინტერნეტის გამოყენებით არის მარშრუტიზებული ორგანიზაციის შიდა ქსელიდან დაშორებულ ადგილთან ან თანამშრომლის ჰოსტთან. შიდა ქსელის ინფორმაცია დაცულად არის გადაცემული ღია ქსელში, რათა შეიქმნას ვირტუალური ქსელი.



ვირტუალური დაცული ქსელის უპირატესობებია:

- **ხარჯების შემცირება** - ვირტუალური დაცული ქსელები ორგანიზაციებს აძლევს საშუალებას გამოიყენონ რენტაბელური, მესამე მხარის ინტერნეტ გადაცემა

დაშორებული ოფისებისა და მომხმარებლების დასაკავშირებლად მთავარ ოფისთან; ამიტომ მცირდება ხარჯები გამოყოფილ WAN კავშირებსა და მოდემების მარაგზე. გარდა ამისა ეკონომიკურად ეფექტური მაღალი-წარმადობის ტექნოლოგიების გამოჩენით, როგორიცაა DSL, ორგანიზაციებს შეუძლიათ გამოიყენონ ვირტუალური დაცული ქსელები შეერთების ღირებულების შემცირებისთვის რა დროსაც იმავდროულად იზრდება დაშორებული კავშირის გამტარუნარიანობა.

- **მასშტაბურობა** - ვირტუალური დაცული ქსელები ორგანიზაციებს აძლევს ინტერნეტის ინფრასტრუქტურის გამოყენების საშუალებას ინტერნეტის სერვის პროვაიდერებსა და მოწყობილობებს შორის, რაც აადვილებს ახალი მომხმარებლების დამატებას. ამიტომ ორგანიზაციებს აქვთ უნარი დაამატონ დიდი რაოდენობით მოცულობა, რაიმე მნიშვნელოვანი ინფრასტრუქტურის დამატების გარეშე.
- **ფართოზოლოვან (Broadband) ტექნოლოგიებთან თავსებადობა** - ვირტუალური დაცული ქსელები მობილურით და დისტანციურად მომუშავეებს საშუალებას აძლევს ისარგებლონ მაღალი სიჩქარის, ფართოზოლოვანი კავშირით, როგორიცაა DSL და კაბელი, თავისი ორგანიზაციების ქსელებთან წვდომისათვის. ფართოზოლოვანი კავშირები უზრუნველყოფს მოქნილობასა და ეფექტურობას. მაღალი სიჩქარის ფართოზოლოვანი კავშირები ასევე უზრუნველყოფენ რენტაბელურ გადაწყვეტას დაშორებულ ოფისებთან დაკავშირებისთვის.
- **უსაფრთხოება** - ვირტუალური დაცული ქსელები შეიცავენ უსაფრთხოების მექანიზმებს, რაც უზრუნველყოფენ მაღალი დონის უსაფრთხოებას თანამედროვე შიფრაციისა და აუთენტიფიკაციის პროტოკოლების გამოყენებით, რომელიც იცავს მონაცემებს არაავტორიზებული წვდომისაგან.

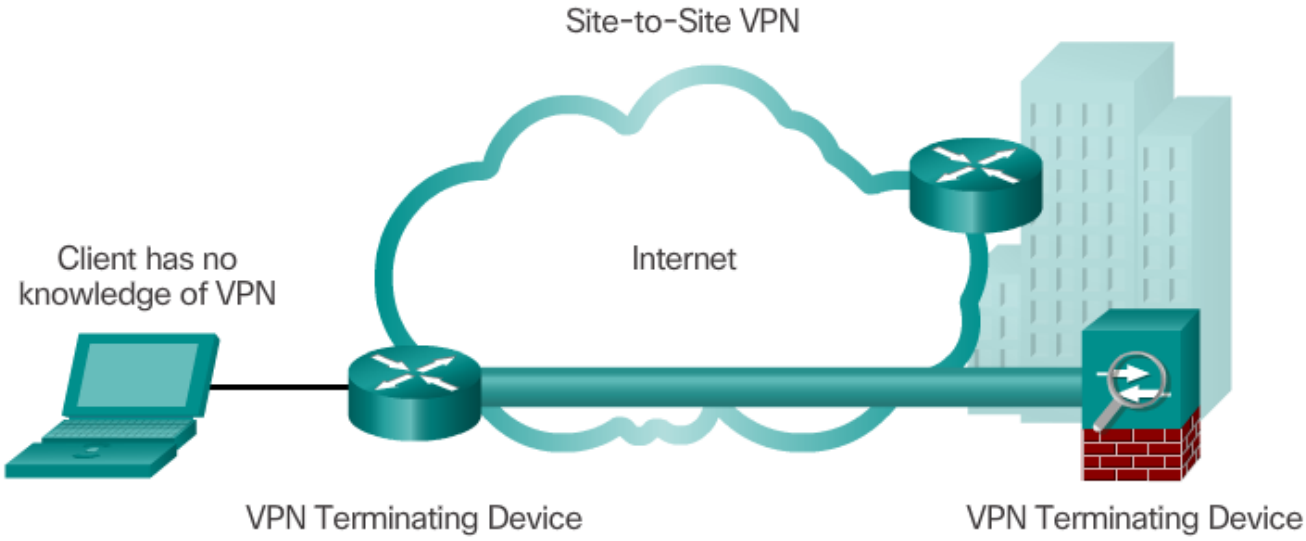
4.1.2 კვანძთაშორისი (Site-to-Site) ვირტუალური დაცული ქსელები

არსებობს ორი ტიპის VPN ქსელი:

- კვანძთაშორისი
- დაშორებული წვდომის

კვანძთაშორისი ვირტუალური დაცული ქსელი (Site-to-Site VPN)

კვანძთაშორისი VPN იქმნება, როცა VPN კავშირის ორივე მხარის მოწყობილობამ წინასწარ იცის VPN კონფიგურაცია, ისე როგორც ნაჩვენებია სურათზე.



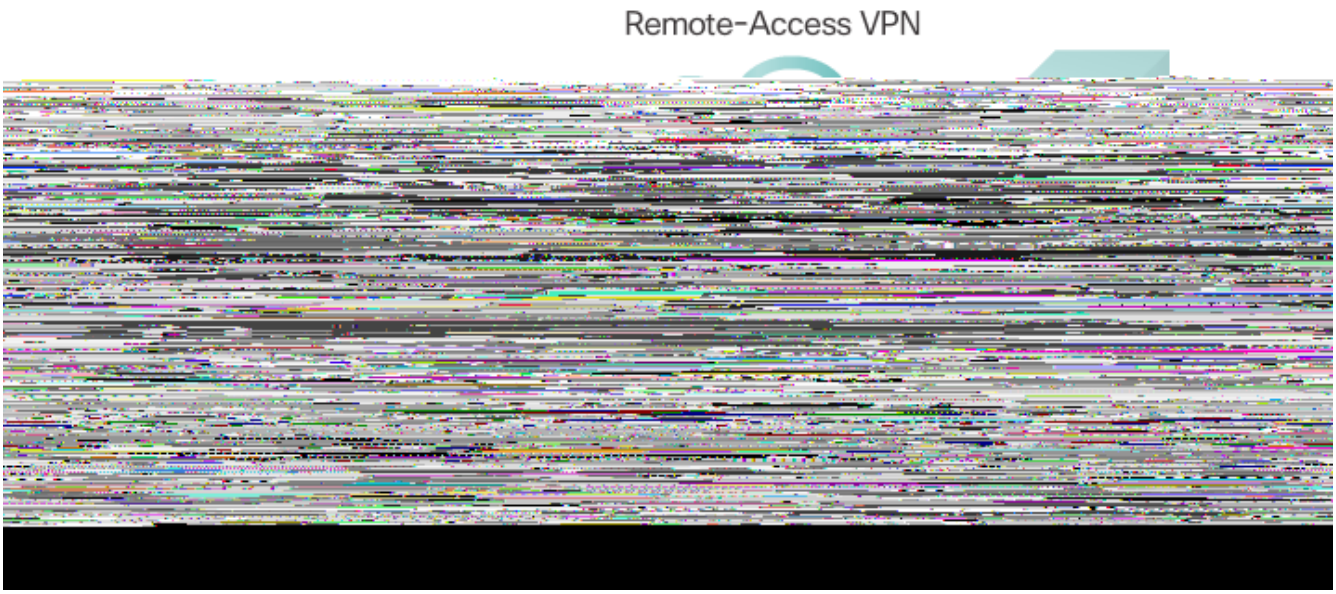
VPN რჩება უცვლელი და შიდა ჰოსტებმა არ იციან რომ VPN არსებობს. კვანძთაშორის VPN-ში საბოლოო მოწყობილობები აგზავნიან და იღებენ ჩვეულებრივ TCP/IP ტრაფიკს VPN გასასვლელის - Gateway საშუალებით. VPN გასასვლელი პასუხისმგებელია კონკრეტული ადგილიდან გამავალი ყველა ტრაფიკის ენკაპსულაციასა და შიფრაციაზე. შემდეგ VPN გასასვლელი აგზავნის მათ ინტერნეტით, VPN ტუნელის საშუალებით სამიზნე კვანძის მსგავს VPN გასასვლელთან. მიღებისთანავე თანაბარუფლებიანი (peer) VPN გასასვლელი მოაცილებს თავსართებს, მოახდენს შემცველობის დეშიფრაციას და გადასცემს პაკეტს სამიზნე ჰოსტს მის შიდა ქსელში.

კვანძთაშორისი VPN არის კლასიკური WAN ქსელის გაფართოება. კვანძთაშორისი ვირტუალური დაცული ქსელები აკავშირებენ მთელს ქსელებს ერთმანეთთან, მაგალითად,

მათ შეუძლიათ ფილიალი ოფისის დაკავშირება კომპანიის სათაო ოფისის ქსელთან. ადრე გამოყოფილი ხაზები ან Frame Relay კავშირი იყო მოთხოვნილი კვანძებს შორის კავშირისათვის, მაგრამ იმის გამო, რომ ეხლანდელ კომპანიებს ახლა უკვე აქვთ ინტერნეტთან წვდომა, მათი კავშირები შეიძლება იქნას შეცვლილი კვანძთაშორისი (Site-to-Site) ვირტუალური დაცული ქსელებით (VPNs).

4.1.3 დაშორებული წვდომის ვირტუალური დაცული ქსელები

მაშინ როცა კვანძთაშორისი VPN გამოიყენება მთელი ქსელების დასაკავშირებლად, დაშორებული წვდომის VPN მხარს უჭერს დაშორებული მომხმარებლების, მობილურ მომხმარებლების, ექსტრანეტის და მომხმარებელ-ბიზნესის ტრაფიკის მოთხოვნებს. დაშორებული წვდომის VPN იქმნება მაშინ, როდესაც VPN ინფორმაცია არ არის სტატიკურად მომართული, მაგრამ მის სანაცვლოდ იძლევა ინფორმაციის დინამიურად ცვლილების საშუალებას და შესაძლებელია მისი ჩართვა და გამორთვა. დაშორებული წვდომის ვირტუალური დაცული ქსელები მხარს უჭერენ კლიენტ/სერვერულ არქიტექტურას, სადაც VPN კლიენტი (დაშორებული ჰოსტი) იღებს დაცულ წვდომას ორგანიზაციის ქსელთან VPN სერვერ მოწყობილობის საშუალებით ქსელის მეორე მხარეს.



დაშორებული წვდომის ვირტუალური დაცული ქსელები გამოიყენება ცალკეულ ჰოსტებთან დასაკავშირებლად, რომლებსაც უნდა ჰქონდეთ უსაფრთხო წვდომა კომპანიის ქსელთან ინტერნეტის საშუალებით. დაშორებული მომხმარებლების მიერ გამოყენებული ინტერნეთ კავშირი როგორც წესი არის: ფართოზოლოვანი (Broadband), DSL, უკაბელო ან კაბელური შეერთება, როგორც ნაჩვენებია სურათზე.

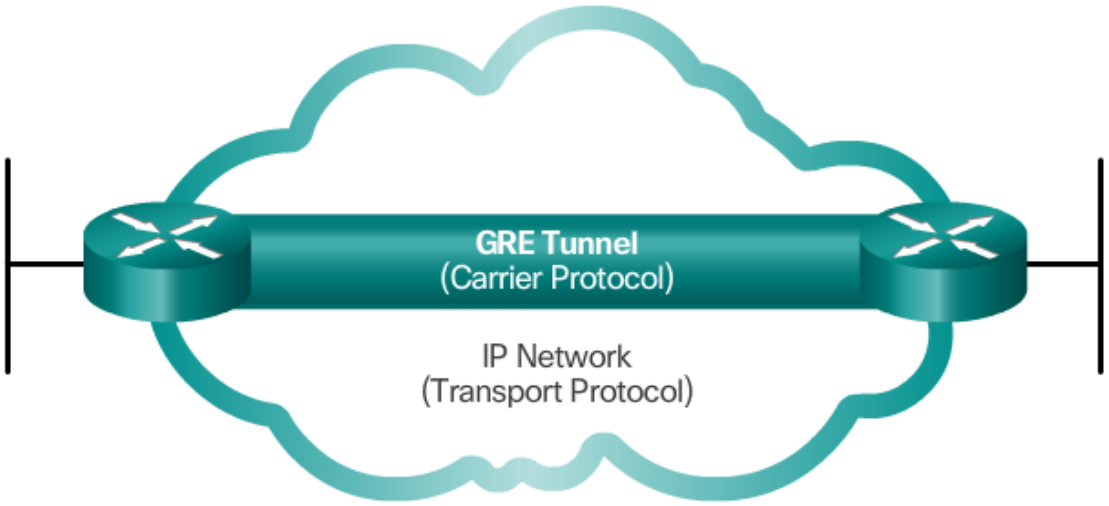
მობილური მომხმარებლების საბოლოო მოწყობილობებზე შეიძლება საჭირო გახდეს VPN კლიენტ პროგრამული უზრუნველყოფის ინსტალაცია; მაგალითად თითოეულ ჰოსტს შეიძლება ჰქონდეს ინსტალირებული Cisco AnyConnect Secure Mobility Client პროგრამული უზრუნველყოფა. როცა ჰოსტი ეცდება ნებისმიერი ტრაფიკის გაგზავნას, Cisco AnyConnect VPN კლიენტი პროგრამული უზრუნველყოფა მოახდენს ამ ტრაფიკის ენკაპსულაციას და შიფრაციას. დაშიფრული მონაცემები შემდეგ ინტერნეტის საშუალებით გაიგზავნება სამიზნე ქსელის მხარეს არსებულ VPN გასასვლელისკენ - Gateway. მიღებისთანავე VPN gateway იქცევა ისე, როგორც კვანძთაშორისი ვირტუალური დაცული ქსელების (Site-to-Site VPNs) შემთხვევაში იქცეოდა.

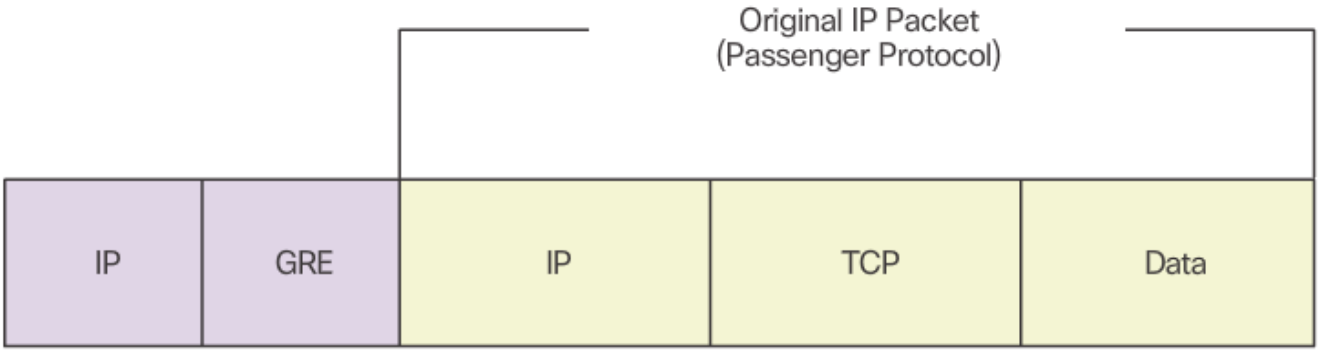
შენიშვნა: Cisco AnyConnect Secure Mobility Client პროგრამული უზრუნველყოფა ეფუძნება წინამორბედ Cisco AnyConnect VPN Client და Cisco VPN Client შეთავაზებებს, რათა გააუმჯობესოს მუდმივი VPN გამოცდილება ლეპტოპებსა და სმარტფონებზე დაფუძნებულ მობილური მოწყობილობებზე. მოცემული კლიენტი მხარს უჭერს IPv6-ს.

4.1.4 Generic Routing Encapsulation (GRE)-ს საფუძვლები

GRE არის ბაზისური, დაუცველი, კვანძთაშორისი კავშირის VPN ტუნელირების პროტოკოლის ერთი მაგალითი. GRE არის Cisco-ს მიერ განვითარებული ტუნელირების პროტოკოლი, რომელსაც შეუძლია მრავალფეროვანი პაკეტის ტიპის პროტოკოლის ენკაპსულაცია IP ტუნელების შიგნით. GRE ქმნის ვირტუალურ წერტილიდან-წერტილამდე (Point-to-point) კავშირს Cisco-ს მარშრუტიზატორებისათვის დაშორებულ წერტილებში, IP გაერთიანებულ ქსელებზე (Internetwork).

GRE არის შექმნილი მრავალპროტოკოლიანი და IP ჯგუფური მაუწყებლობის (Multicast) ტრაფიკის გასაცვლელად ორ ან მეტ კვანძს შორის, რომლებსაც უნდა ჰქონდეთ მხოლოდ IP კავშირი. მას შეუძლია მრავალი პროტოკოლის პაკეტის ტიპების ენკაპსულაცია IP ტუნელის შიგნით.





როგორც სურათზეა ნაჩვენები, ტუნელური ინტერფეისი მხარს უჭერს თავსართს თითოეული ქვემოთ ჩამოთვლილისათვის:

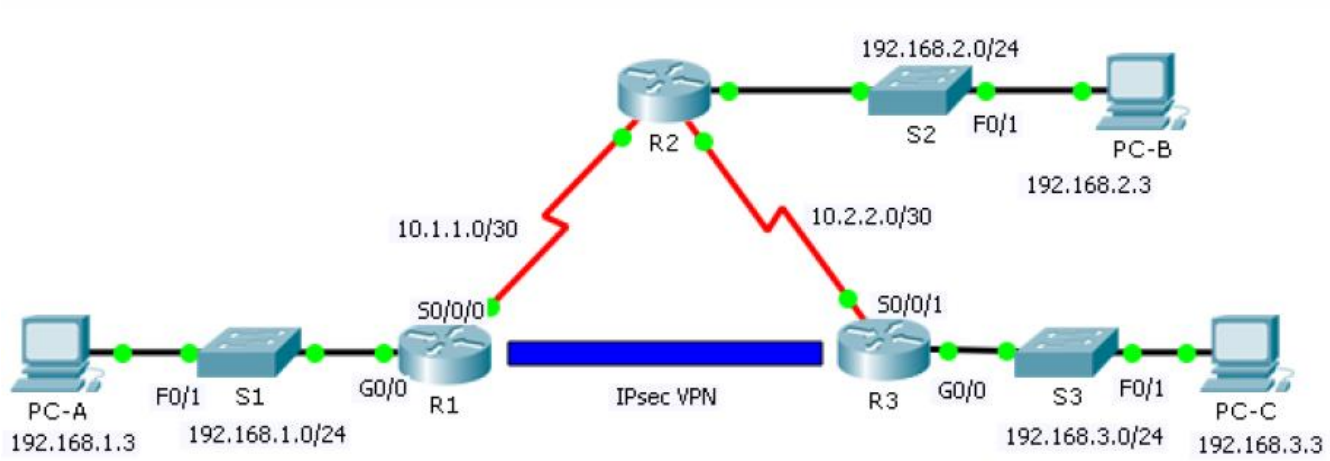
- ენკაპსულირებული პროტოკოლი (ან გამვლელი - passenger პროტოკოლი), როგორიცაა IPv4, IPv6, AppleTalk, DECnet ან IPX.
- ენკაპსულაციის პროტოკოლი (ან სატრანსპორტო), როგორიცაა GRE.

მიწოდების გადაცემის პროტოკოლი, როგორიცაა IP, რომელიც არის პროტოკოლი, ვინც ატარებს ენკაპსულირებულ პროტოკოლს.

4.2. VPN (Virtual Private Network) -ის კონფიგურირება

4.2.1. *VPN-ების კონფიგურაცია*

ტოპოლოგია



მისამართების ცხრილი

მოწყობილობა	ინტერფეისი	IP მისამართი	ქვექსელის ნიღაბი	ნაგულისხმევი გასასვლელი
R1	G0/0	192.168.1.1	255.255.255.0	N/A
	S0/0/0	10.1.1.2	255.255.255.252	N/A
R2	G0/0	192.168.2.1	255.255.255.0	N/A
	S0/0/0	10.1.1.1	255.255.255.252	N/A
	S0/0/1	10.2.2.1	255.255.255.252	N/A
R3	G0/0	192.168.3.1	255.255.255.0	N/A
	S0/0/1	10.2.2.2	255.255.255.252	N/A
PC-A	ქსელის ადაპტერი	192.168.1.3	255.255.255.0	192.168.1.1

PC-B	ქსელის ადაპტერი	192.168.2.3	255.255.255.0	192.168.2.1
PC-C	ქსელის ადაპტერი	192.168.3.3	255.255.255.0	192.168.3.1

ISAKMP ფაზა 1 პოლიტიკის პარამეტრები

პარამეტრები		R1	R3
გასაღების განაწილების მეთოდი	ხელით ან **ISAKMP**	ISAKMP	ISAKMP
შიფრაციის ალგორითმი	**DES**, 3DES ან AES	AES	AES
ჰეშირების ალგორითმი	MD5 ან **SHA-1**	SHA-1	SHA-1
აუთენტიკაციის მეთოდი	წინასწარ განსაზღვრული გასაღებები ან **RSA**	წინასწარ განსაზღვრული	წინასწარ განსაზღვრული
გასაღების გაცვლა	DH ჯგუფი **1**,2, ან 5	DH 2	DH 2
IKE SA მოქმედების ვადა	86400 წამი ან ნაკლები	86400	86400
ISAKMP გასაღები		cisco	cisco

მუქად მონიშნულები არის ნაგულისხმევი პარამეტრები. სხვა პარამეტრებს სჭირდებათ პირდაპირი კონფიგურაცია.

IPsec ფაზა 2 პოლიტიკის პარამეტრები

პარამეტრები	R1	R3
გარდაქმნის ნაკრები	VPN-SET	VPN-SET
Peer ჰოსტის სახელი	R3	R1
Peer IP მისამართი	10.2.2.2	10.1.1.2
დასაშიფრი ქსელი	192.168.1.0/24	192.168.3.0/24
კრიპტოგრაფიული რუკის სახელი	VPN-MAP	VPN-MAP

SA დადგენა	ipsec-isakmp	ipsec-isakmp

შესასრულებელი სამუშაოები

ნაწილი №1: უსაფრთხოების ფუნქციების ჩართვა

ნაწილი №2: IPsec პარამეტრების კონფიგურაცია R1 მარშრუტიზატორზე

ნაწილი №3: IPsec პარამეტრების კონფიგურაცია R3 მარშრუტიზატორზე

ნაწილი №4: IPsec VPN-ის შემოწმება

სცენარი

მოცემულ დავალებაში თქვენ უნდა დააკონფიგუროთ ორი მარშრუტიზატორი site-to site IPsec VPN-ის მხარდაჭერისთვის მათი შესაბამისი ლოკალური ქსელებიდან ტრაფიკის გასატარებლად. IPsec VPN ტრაფიკი გაივლის სხვა მარშუტიზატორის დახმარებით, რომელიც არ არის ნაცნობი VPN-სთვის. IPsec უზრუნველყოფს კონფიდენციალური ინფორმაციის უსაფრთხო გადაცემას დაუცველ ქსელებში, ინტერნეტის ჩათვლით. IPsec მოქმედებს ქსელის დონეზე, იცავს IP პაკეტებს და ახდენს მათ აუთენტიკაციას მონაწილე IPsec მოწყობილობებს შორის, Cisco მარშრუტიზატორების ჩათვლით.

ნაწილი №1: უსაფრთხოების ფუნქციების ჩართვა

პირველი ეტაპი: securityk9 მოდულის აქტივაცია

ამ დავალების შესასრულებლად აუცილებელია უსაფრთხოების ტექნოლოგიების პაკეტის ლიცენზიის ჩართვა.

შენიშვნა: მომხმარებლის EXEC და პრივილეგირებული EXEC რეჟიმების პაროლი არის cisco.

ა. გაუშვით **show version** ბრძანება მომხმარებლის EXEC ან პრივილეგირებული EXEC რეჟიმებიდან, რათა შევამოწმოთ აქტივირებულია თუ არა უსაფრთხოების ტექნოლოგიების პაკეტის ლიცენზია.

```
------------------------------------------------------------------
Technology    Technology-package          Technology-package
              Current       Type          Next reboot
------------------------------------------------------------------
ipbase        ipbasek9      Permanent     ipbasek9
security      None          None          None
uc            None          None          None
data          None          None          None

Configuration register is 0x2102
```

ბ. თუ არა, გააქტიურეთ **security9** მოდული მარშრუტიზატორის შემდეგი ჩატვირთვისთვის, დაეთანხმეთ ლიცენზიას, შეინახეთ კონფიგურაცია და გადატვირთეთ.

R1(config)# **license boot module c2900 technology-package security9**

R1(config) # **end**

R1# **copy running-config startup-config**

R1# **reload**

გ. გადატვირთვის დასრულების შემდეგ, გაუშვით **show version** ბრძანება ხელახლა, უსაფრთხოების ტექნოლოგიის პაკეტის ლიცენზიის აქტივაციის შესამოწმებლად.

```
Technology Package License Information for Module:'c2900'

------------------------------------------------------------------
Technology    Technology-package          Technology-package
              Current       Type          Next reboot
------------------------------------------------------------------
ipbase        ipbasek9      Permanent     ipbasek9
security      securityk9    Evaluation    securityk9
uc            None          None          None
data          None          None          None
```

დ. გაიმეორეთ 1ა-1გ ეტაპები **R3** მარშრუტიზატორზე

ნაწილი №2: IPsec პარამეტრების კონფიგურაცია R1 მარშრუტიზატორზე

პირველი ეტაპი: კავშირის შემოწმება.

PC-A-დან დაპინგეთ **PC-C**.

მეორე ეტაპი: საინტერესო ტრაფიკის იდენტიფიკაცია R1 მარშრუტიზატორზე.

დააკონფიგურეთ წვდომის კონტროლის სია (ACL) 110 **R1** მარშრუტიზატორის ლოკალური ქსელიდან **R3** მარშრუტიზატორის ლოკალური ქსელში საინტერესოდ მიჩნეული ტრაფიკის იდენტიფიკაციისათვის. სხვა ყველა ტრაფიკი, რომელიც მოდის ლოკალური ქსელებიდან არ იქნება დაშიფრული. შეგახსენებთ რომ ბუნდოვანი deny any - ნებისმიერის აკრძალვა წესის დროს, არ არის აუცილებელი შესაბამისი დადგენილების დამატება სიაში.

R1(config)# **access-list 110 permit ip 192.168.1.0 0.0.0.255 192.168.3.0 0.0.0.255**

მესამე ეტაპი: ISAKMP ფაზა 1 პარამეტრების კონფიგურაცია R1 მარშრუტიზატორზე.

დააკონფიგურეთ კრიპტოგრაფიის ISAKMP პოლიტიკა **10** პარამეტრები **R1** მარშრუტიზატორზე, გაზიარებული კრიპტოგრაფიის **cisco** გასაღებთან ერთად. გამოიყენეთ ISAKMP ფაზა 1-ის ცხრილი კონფიგურაციისთვის საჭირო სპეციფიური პარამეტრების მისათითებლად. ნაგულისხმევი მნიშვნელობები არ უნდა იქნას მომართული, ამიტომ მხოლოდ შიფრაცია, გასაღების გაცვლის მეთოდი და DH მეთოდი უნდა დაკონფიგურდეს.

R1(config)# **crypto isakmp policy 10**

R1(config-isakmp)# **encryption aes**

R1(config-isakmp)# **authentication pre-share**

R1(config-isakmp)# **group 2**

R1(config-isakmp)# **exit**

R1(config)# **crypto isakmp key cisco address 10.2.2.2**

მეოთხე ეტაპი: ISAKMP ფაზა 2-ის პარამეტრების კონფიგურაცია R1 მარშრუტიზატორზე.

შექმენით **VPN-SET** გარდაქმნის ნაკრები - transform set, რათა გამოიყენოთ **esp-3des** და **esp-sha-hmac**. შემდეგ შექმენით კრიპტოგრაფიის რუკა **VPN-MAP**, რომელიც შეკრავს ფაზა 2-ის ყველა პარამეტრს ერთად. გამოიყენეთ რიგითი ნომერი **10** და მოახდინეთ მისი როგორც **ipsec-isakmp** რუკის იდენტიფიკაცია.

R1 (config) # **crypto ipsec transform-set VPN-SET esp-3des esp-sha-hmac**

R1 (config) # **crypto map VPN-MAP 10 ipsec-isakmp**

R1 (config-crypto-map) # **description vpn connection to R3**

R1 (config-crypto-map) # **set peer 10.2.2.2**

R1 (config-crypto-map) # **set transform-set VPN-SET**

R1 (config-crypto-map) # **match address 110**

R1 (config-crypto-map) # **exit**

მეხუთე ეტაპი: კრიპტოგრაფიის რუკის კონფიგურაცია გამავალ ინტერფეისზე

ბოლოს დააკავშირეთ **VPN-MAP** კრიპტოგრაფიული რუქა გამავალ Serial 0/0/0 ინტერფეისთან.

R1 (config) # **interface s0/0/0**

R1 (config-if) # **crypto map VPN-MAP**

მესამე ნაწილი: IPsec პარამეტრების კონფიგურაცია R3 მარშრუტიზატორზე

პირველი ეტაპი: R3 მარშრუტიზატორის კონფიგურაცია R1 მარშრუტიზატორთან site-to-site VPN-ის მხარდასაჭერად.

დააკონფიგურეთ უკუქცევითი პარამეტრები **R3** მარშრუტიზატორზე. დააკონფიგურეთ ACL **110 R3** მარშრუტიზატორის ლოკალური ქსელიდან **R1** მარშრუტიზატორის ლოკალურ ქსელში საინტერეოდ მიჩნეული ტრაფიკის იდენტიფიკაციისათვის.

R3 (config) # **access-list 110 permit ip 192.168.3.0 0.0.0.255 192.168.1.0 0.0.0.255**

მეორე ეტაპი: კრიპტოგრაფიის ISAKMP პოლიტიკა 10-ის პარამეტრების კონფიგურაცია

დააკონფიგურეთ კრიპტოგრაფიის ISAKMP პოლიტიკა **10**-ის პარამეტრები **R3** მარშრუტიზატორზე გაზიარებულ შიფრაციის **cisco** გასაღებთან ერთად.

R3 (config) # **crypto isakmp policy 10**

R3 (config-isakmp) # **encryption aes**

R3 (config-isakmp) # **authentication pre-share**

R3 (config-isakmp) # **group 2**

R3 (config-isakmp) # **exit**

R3 (config) # **crypto isakmp key cisco address 10.1.1.2**

მესამე ეტაპი: ISAKMP ფაზა 2-ის პარამეტრების კონფიგურაცია R1 მარშრუტიზატორზე

როგორც **R1** მარშრუტიზატორზე გააკეთეთ, შექმენით გარდაქმნის ნაკრები - transform-set **VPN-SET** რათა გამოიყენოთ **esp-3des** და **esp-sha-hmac**. შემდეგ შექმენით კრიპტოგრაფიის რუკა **VPN-MAP**, რომელიც შეკრავს ყველა ფაზა 2-ის პარამეტრებს ერთად. გამოიყენეთ რიგითი ნომერი **10** და მოახდინეთ მისი იდენტიფიკაცია როგორც **ipsec-isakmp** რუკა.

R3 (config) # **crypto ipsec transform-set VPN-SET esp-3des esp-sha-hmac**

R3 (config) # **crypto map VPN-MAP 10 ipsec-isakmp**

R3 (config-crypto-map) # **description vpn connection to R1**

R3 (config-crypto-map) # **set peer 10.1.1.2**

R3 (config-crypto-map) # **set transform-set VPN-SET**

R3 (config-crypto-map) # **match address 110**

R3 (config-crypto-map) # **exit**

მეოთხე ეტაპი: კრიპტოგრაფიის რუკის კონფიგურაცია გამავალ ინტერფეისზე

ბოლოს დააკავშირეთ **VPN-MAP** კრიპტოგრაფიული რუქა გამავალ Serial 0/0/1 ინტერფეისთან.

R1 (config) # **interface s0/0/1**

R1 (config-if) # **crypto map VPN-MAP**

ნაწილი №4: IPsec VPN-ის შემოწმება

პირველი ეტაპი: წინა საინტერესო ტრაფიკის ტუნელის შემოწმება.

გაუშვით **show crypto ipsec sa** ბრძანება **R1** მარშრუტიზატორზე. შენიშვნა: ენკაპსულირებული, შიფრირებული, დეკაპსულირებული და დეშიფრირებული პაკეტების რაოდენობა არის ნული.

R1# **show crypto ipsec sa**

```
interface: Serial0/0/0
    Crypto map tag: VPN-MAP, local addr 10.1.1.2

   protected vrf: (none)
   local  ident (addr/mask/prot/port): (192.168.1.0/255.255.255.0/0/0)
   remote  ident (addr/mask/prot/port): (192.168.3.0/255.255.255.0/0/0)
   current_peer 10.2.2.2 port 500
    PERMIT, flags={origin_is_acl,}
   #pkts encaps: 0, #pkts encrypt: 0, #pkts digest: 0
   #pkts decaps: 0, #pkts decrypt: 0, #pkts verify: 0
   #pkts compressed: 0, #pkts decompressed: 0
   #pkts not compressed: 0, #pkts compr. failed: 0
   #pkts not decompressed: 0, #pkts decompress failed: 0
   #send errors 0, #recv errors 0

     local crypto endpt.: 10.1.1.2, remote crypto endpt.:10.2.2.2
     path mtu 1500, ip mtu 1500, ip mtu idb Serial0/0/0
     current outbound spi: 0x0(0)
<output omitted>
```

მეორე ეტაპი: საინტერესო ტრაფიკის შექმნა.

PC-A-დან დაპინგეთ **PC-C.**

მესამე ეტაპი: ტუნელის შემოწმება საინტერესო ტრაფიკის შემდეგ.

R1 მარშრუტიზატორზე ხელახლა გაუშვით **show crypto ipsec sa** ბრძანება. გამოტანილი ინფორმაციიდან ვიგებთ რომ პაკეტების რიცხვი არის 0-ზე მეტი, რაც მიუთითებს რომ IPsec VPN ტუნელი მუშაობს.

R1# **show crypto ipsec sa**

```
interface: Serial0/0/0
    Crypto map tag: VPN-MAP, local addr 10.1.1.2

   protected vrf: (none)
   local  ident (addr/mask/prot/port): (192.168.1.0/255.255.255.0/0/0)
   remote  ident (addr/mask/prot/port): (192.168.3.0/255.255.255.0/0/0)
   current_peer 10.2.2.2 port 500
    PERMIT, flags={origin_is_acl,}
   #pkts encaps: 3, #pkts encrypt: 3, #pkts digest: 0
   #pkts decaps: 3, #pkts decrypt: 3, #pkts verify: 0
   #pkts compressed: 0, #pkts decompressed: 0
   #pkts not compressed: 0, #pkts compr. failed: 0
   #pkts not decompressed: 0, #pkts decompress failed: 0
   #send errors 1, #recv errors 0

     local crypto endpt.: 10.1.1.2, remote crypto endpt.:10.2.2.2
     path mtu 1500, ip mtu 1500, ip mtu idb Serial0/0/0
     current outbound spi: 0x0A496941(172583233)
<output omitted>
```

მეოთხე ეტაპი: არასაინტერესო ტრაფიკის შექმნა.

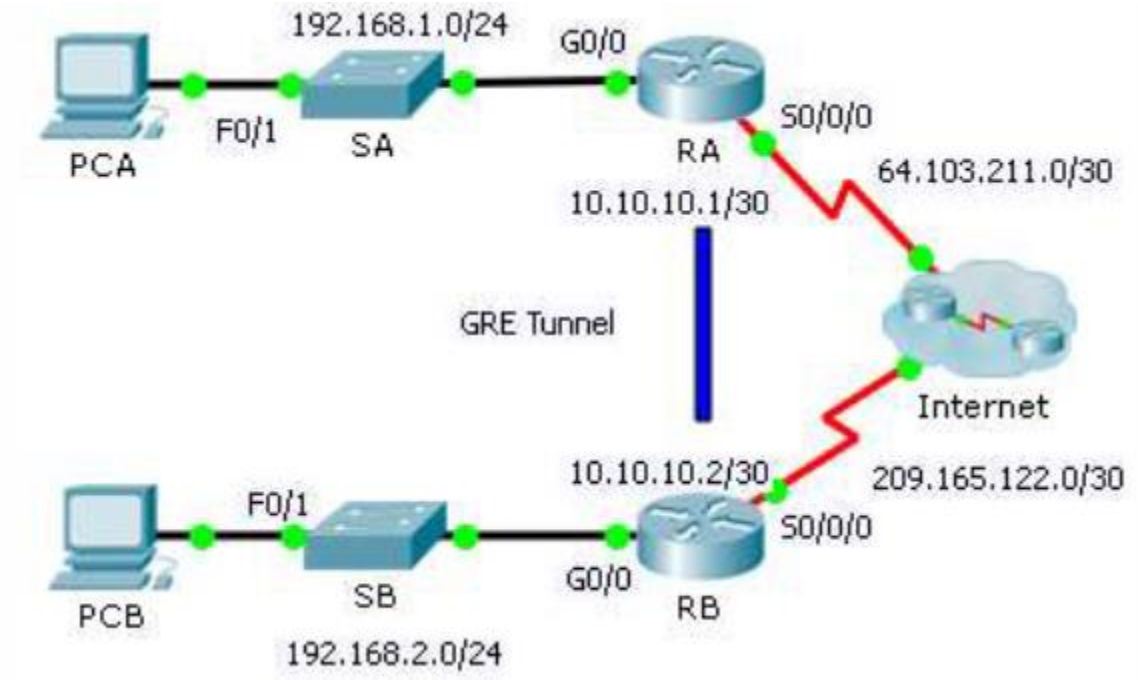
PC-A-დან დაპინგეთ **PC-B**.

მეხუთე ეტაპი: ტუნელის შემოწმება.

R1 მარშრუტიზატორზე ხელახლა გაუშვით **show crypto ipsec sa** ბრძანება. მას შემდეგ რაც, შევიტყობთ რომ პაკეტების რიცხვი არ შეცვლილა, შევამოწმოთ რომ არასაინტერესო ტრაფიკი არ არის შიფრირებული.

4.2.2. *GRE (Generic Routing Encapsulation)-ს კონფიგურაცია*

ტოპოლოგია



მისამართების ცხრილი

მოწყობილობა	ინტერფეისი	IP მისამართი	ქვექსელის ნიღაბი	ნაგულისხმევი გასავლელი
RA	G0/0	192.168.1.1	255.255.255.0	N/A
	S0/0/0	64.103.211.2	255.255.255.252	N/A
	Tunnel 0	10.10.10.1	255.255.255.252	N/A
RB	G0/0	192.168.2.1	255.255.255.0	N/A
	S0/0/0	209.165.122.2	255.255.255.252	N/A
	Tunnel 0	10.10.10.2	255.255.255.252	N/A
PC-A	ქსელის ადაპტერი	192.168.1.2	255.255.255.0	192.168.1.1
PC-C	ქსელის ადაპტერი	192.168.2.2	255.255.255.0	192.168.2.1

შესასრულებელი დავალებები:

ნაწილი №1: მარშრუტიზატორის კავშირის შემოწმება

ნაწილი №2: GRE ტუნელების კონფიგურაცია

ნაწილი №3: PC კავშირის შემოწმება

სცენარი

თქვენ ხართ კომპანიის ქსელის ადმინისტრატორი. კომპანიას სურს მომართოთ GRE ტუნელი დაშორებულ ოფისთან. ორივე ქსელი ლოკალურად არის კონფიგურირებული და საჭიროა მხოლოდ ტუნელის კონფიგურაცია.

ნაწილი №1: მარშრუტიზატორის კავშირის შემოწმება

პირველი ეტაპი: RB მარშრუტიზატორიდან დაპინგეთ RA მარშრუტიზატორი

ა. გამოიყენეთ **show ip interface brief** ბრძანება **RA** მარშრუტიზატორზე S0/0/0 პორტის IP მისამართის დასადგენად.

ბ. **RB** მარშრუტიზატორიდან დაპინგეთ **RA** მარშრუტიზატორის S0/0/0 IP მისამართი.

მეორე ეტაპი: PCB-დან დაპინგეთ PCA

PCB-დან სცადეთ **PCA** -ს IP მისამართის დაპინგვა. ჩვენ ისევ გავიმეორებთ ამ ტესტს GRE ტუნელის კონფიგურაციის შემდეგ. რა არის პინგ ბრძანების შედეგი? რატომ?

_____

ნაწილი №2: GRE ტუნელების კონფიგურაცია

პირველი ეტაპი: Tunnel 0 ინტერფეისის კონფიგურაცია RA მარშრუტიზატორზე

ა. შედით კონფიგურაციის რეჟიმში **RA** მარშრუტიზატორის Tunnel 0-სთვის.

RA (config) # **interface Tunnel 0**

ბ. მომართეთ IP მისამართი ისე როგორც მითითებულია მისამართების ცხრილში.

RA (config-if)# **ip address 10.10.10.1 255.255255.252**

გ. მომართეთ წყარო და ადრესატი Tunnel 0-ის საბოლოო წერტილისათვის.

RA (config-if)# **tunnel source s0/0/0**

RA (config-if)# **tunnel destination 209.165.122.2**

დ. დააკონფიგურეთ Tunnel 0 IP ტრაფიკის GRE-ზე დასაფარად.

RA (config-if)# **tunnel mode gre ip**

ე. Tunnel 0 შეიძლება უკვე აქტივირებული იყოს. იმ შემთხვევაში, თუ ეს ასე არაა, ჩართეთ ის, ნებისმიერ სხვა ინტერფეისის მსგავსად.

RA (config-if)# **no shutdown**

მეორე ეტაპი: დააკონფიგურეთ RB მარშრუტიზატორის Tunnel 0 ინტერფეისი.

გაიმეორეთ 1ა-1ე ეტაპები **RB** მარშრუტიზატორისთვის. არ დაგავიწყდეთ IP მისამართების შეცვლა ცხრილის შესაბამისად.

მესამე ეტაპი: დააკონფიგურეთ მარშრუტი კერძო IP ტრაფიკისათვის.

განსაზღვრეთ მარშრუტი 192.168.X.X ქსელებს შორის, 10.10.10.0/30 ადრესატის მისამართის გამოყენებით.

RA (config)# **ip route 192.168.2.0 255.255.255.0 10.10.10.2**

RB (config)# **ip route 192.168.1.0 255.255.255.0 10.10.10.1**

მესამე ნაწილი: მარშრუტიზატორის კავშირის შემოწმება

პირველი ეტაპი: PCB-დან დაპინგეთ PCA

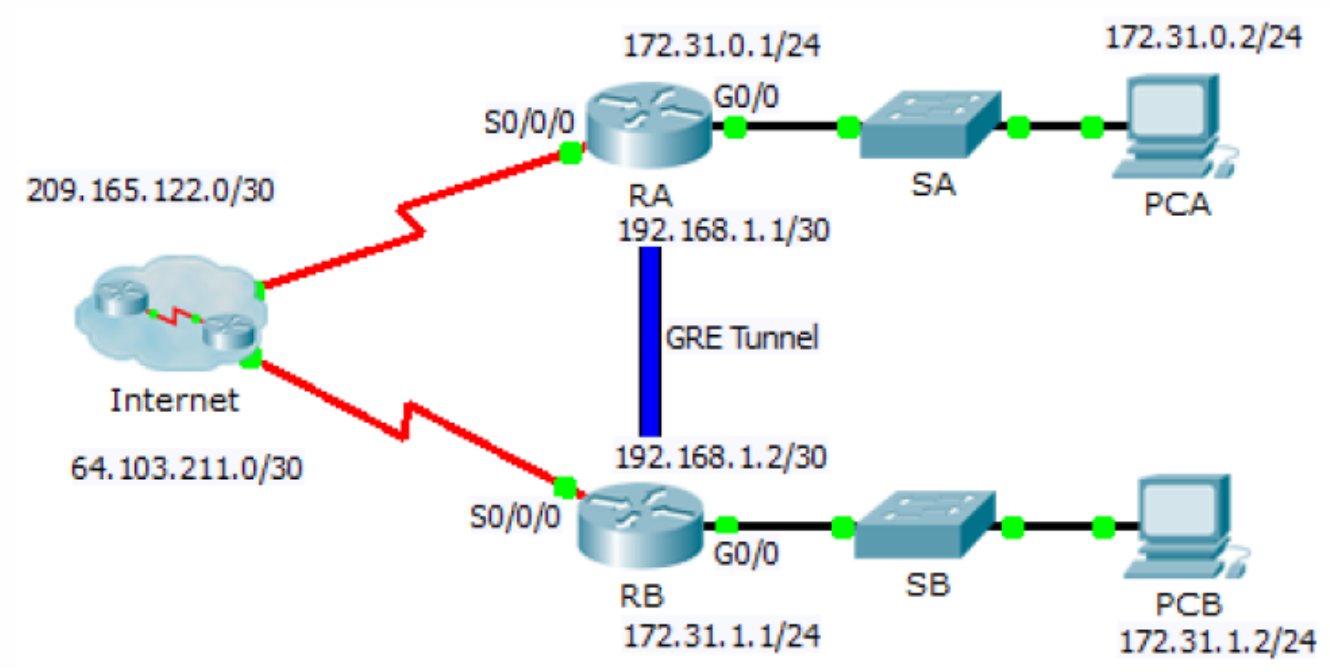
PCB-დან სცადეთ **PCA** -ს IP მისამართის დაპინგვა. პინგი უნდა იყოს წარმატებული.

მეორე ეტაპი: PCA-დან PCB-სკენ მიმავალი გზის თვალთვალი (trace)

სცადეთ **PCA**-დან **PCB**-სკენ მიმავალი გზის თვალთვალი (trace). აღსანიშნავია საზოგადო IP მისამართების არქონა გასასვლელზე.

4.2.3. *GRE (Generic Routing Encapsulation)-ს პრობლემის გადაწყვეტა*

ტოპოლოგია



მისამართების ცხრილი:

მოწყობილობა	ინტერფეისი	IP მისამართი	ქვექსელის ნიღაბი	ნაგულისხმევი გასავლელი
RA	G0/0	172.31.0.1	255.255.255.0	N/A
	S0/0/0	209.165.122.2	255.255.255.252	N/A
	Tunnel 0	192.168.1.1	255.255.255.252	N/A
RB	G0/0	172.31.1.1	255.255.255.0	N/A
	S0/0/0	64.103.211.2	255.255.255.252	N/A
	Tunnel 0	192.168.1.2	255.255.255.252	N/A
PC-A	ქსელის ადაპტერი	172.31.0.2	255.255.255.0	172.31.0.1

PC-C	ქსელის ადაპტერი	172.31.1.2	255.255.255.0	172.31.1.1

შესასრულებელი დავალებები:

- **ყველა ქსელური შეცდომის მოძებნა და გამოსწორება**
- **კავშირის შემოწმება**

სცენარი

დაქირავებულ იქნა უმცროსი ქსელის ადმინისტრატორი, რომელსაც დავალებული აქვს GRE ტუნელის გამართვა ორ ადგილს შორის, მაგრამ ვერ შეძლო ამ დავალების შესრულება. თქვენ შეთავაზეთ კონფიგურაციის შეცდომების გამოსწორება კომპანიის ქსელში.

ნაწილი №1: ქსელის ყველა შეცდომის მოძებნა და გამოსწორება

მოწყობილობა	შეცდომა	გამოსწორება

ნაწილი №2: კავშირის შემოწმება

პირველი ეტაპი: PCB-დან PCA-ს დაპინგვა.

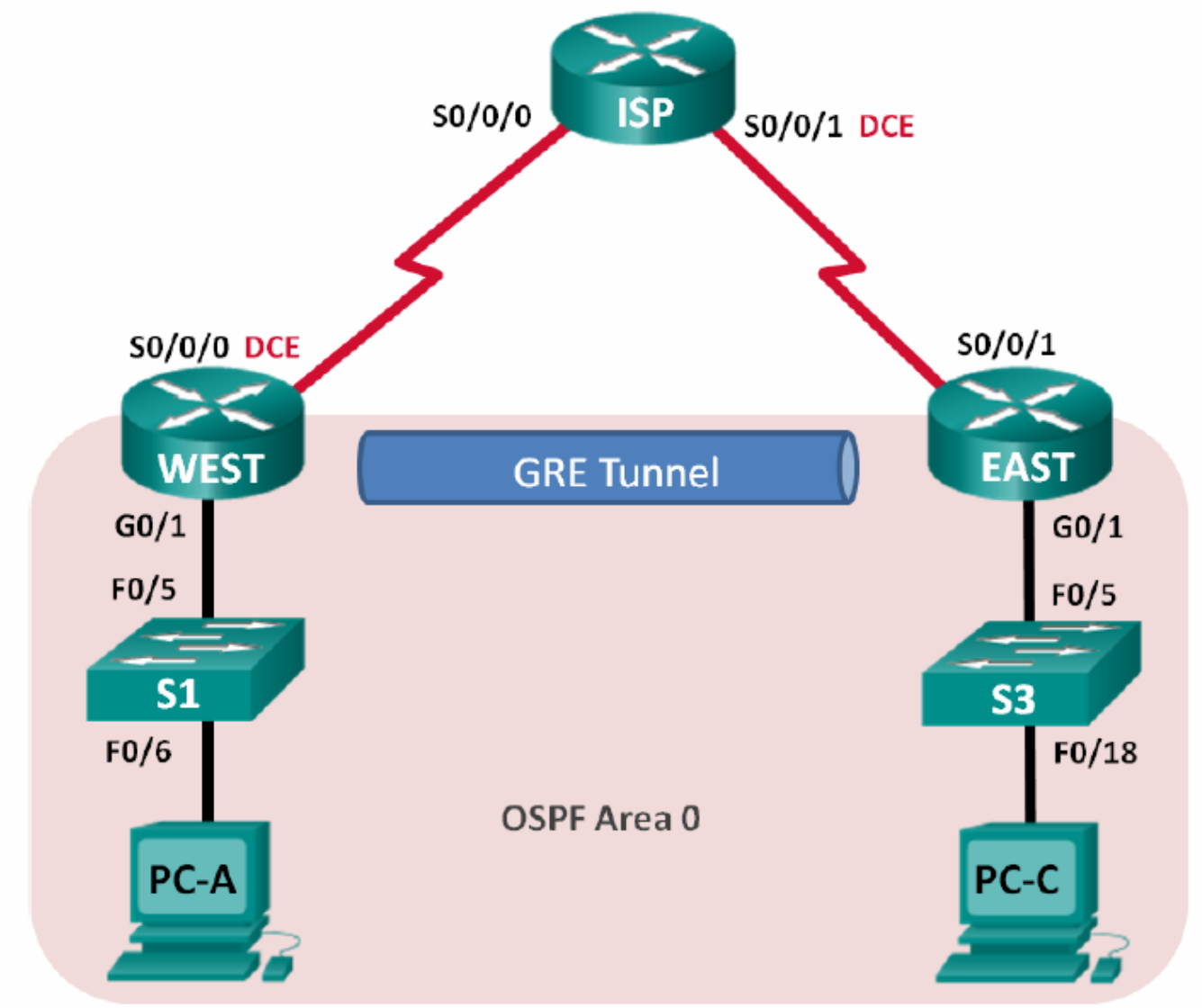
PCB-დან სცადეთ **PCA**-ს IP მისამართის დაპინგვა. პინგი უნდა იყოს წარმატებული

მეორე ეტაპი: PCA-დან PCB-მდე გზის თვალყურის დევნება (trace).

სცადეთ **PCA**-დან **PCB**-მდე გზის თვალყურის დევნება (trace). მიაქციეთ ყურადღება გარე IP მისამართების არქონას გასასვლელზე (output).

4.2.4. *წერტილიდან წერტილამდე (Point-to-Point) GRE VPN ტუნელის კონფიგურაცია*

ტოპოლოგია



მისამართების ცხრილი

მოწყობილობა	ინტერფეისი	IP მისამართი	ქვექსელის ნიღაბი	ნაგულისხმევი გასავლელი
WEST	G0/1	172.16.1.1	255.255.255.0	N/A
	S0/0/0 (DCE)	10.1.1.1	255.255.255.252	N/A

	Tunnel0	172.16.12.1	255.255.255.252	N/A
ISP	S0/0/0	10.1.1.2	255.255.255.252	N/A
	S0/0/1 (DCE)	10.2.2.2	255.255.255.252	N/A
EAST	G0/1	172.16.2.1	255.255.255.0	N/A
	S0/0/1	10.2.2.1	255.255.255.252	N/A
	Tunnel0	172.16.12.2	255.255.255.252	N/A
PC-A	ქსელის ადაპტერი	172.16.1.3	255.255.255.0	172.16.1.1
PC-C	ქსელის ადაპტერი	172.16.2.3	255.255.255.0	172.16.2.1

შესასრულებელი დავალებები:

ნაწილი №1: მოწყობილობის ბაზისური პარამეტრების კონფიგურაცია

ნაწილი №2: GRE ტუნელის კონფიგურაცია

ნაწილი №3: მარშრუტიზაციის ჩართვა GRE ტუნელის საშუალებით

ზოგადი ინფორმაცია / სცენარი

საერთო მარშრუტიზაციის ენკაპსულაცია (GRE) არის ტუნელირების პროტოკოლი, რომელსაც შეუძლია სხვადასხვა ქსელური დონის პროტოკოლების ენკაპსულაცია ორ ადგილს შორის ღია ქსელის გამოყენებით, ისეთი როგორიცაა ინტერნეტი.

GRE შეიძლება გამოყენებულ იქნას:

- IPv6 ქსელების დასაკავშირებლად IPv4 ქსელებთან
- მრავალმისამართიან (Multicast) პაკეტებთან, როგორიცაა OSPF, EIGRP და ნაკადურ აპლიკაციებთან (Streaming Applications)

მოცემულ ლაბორატორიულ დავალებაში თქვენ უნდა დააკონფიგუროთ დაუშიფრავი წერტილიდან-წერტილამდე GRE VPN ტუნელი და დარწმუნდეთ რომ ქსელის ტრაფიკი იყენებს ამ ტუნელს. თქვენ ასევე უნდა დააკონფიგუროთ OSPF მარშრუტიზაციის პროტოკოლი GRE VPN ტუნელში. GRE ტუნელი არის WEST და EAST მარშრუტიზატორებს შორის OSPF სივრცე 0-ში. ISP არის უცნობი GRE ტუნელისათვის. WEST და EAST მარშრუტიზატორებს და ISP-ს შორის კომუნიკაცია სრულდება ნაგულისხმევი სტატიკური მარშრუტის გამოყენებით.

შენიშვნა: მარშრუტიზატორები რომლებიც გამოყენებულია CCNA პრაქტიკულ ლაბორატორიულ სამუშაოებში, არის Cisco 1941 ინტეგრირებული სერვისების მარშრუტიზატორები (ISRs) Cisco IOS Release 15.2(4)M3 (universalk9 image)-ით. გამოყენებული კომუტატორები არის Cisco IOS Catalyst 2960s მოდელები Cisco IOS Release 15.0(2) (lanbasek9 image)-ით. შესაძლებელია სხვა მოდელის მარშრუტიზატორების, კომუტატორების და Cisco IOS ვერსიების გამოყენებაც. იმის მიხედვით რა მოდელი და Cisco IOS ვერსიაა გამოყენებული, ხელმისაწვდომი ბრძანებები და შედეგები შეიძლება იყოს განსხვავებული იმისგან რაც მოცემულია ამ ლაბორატორიულ დავალებაში. გაეცანით მარშრუტიზატორის ინტერფეისის მოკლე მიმოხილვის ცხრილში, ამ ლაბორატორიული სამუშაოს ბოლოში ინტერფეისების სწორი იდენტიფიკატორებისთვის.

შენიშვნა: დარწმუნდით რომ მარშრუტიზატორები და კომუტატორები არის წაშლილი და არ აქვთ საწყისი კონფიგურაცია. თუ ეს ასე არაა, მიმართეთ ინსტრუქტორს.

მოთხოვნილი რესურსები

- 3 მარშრუტიზატორი (Cisco 1941 მოდელი Cisco Release 15.2(4)M3 უნივერსალი იმიჯით ან მსგავსი)
- 2 კომუტატორი (Cisco 2960 მოდელი Cisco IOS Release 15.0(2) lanbasek9 იმიჯით ან მსგავსი)
- 2 პერსონალური კომპიუტერი (Windows 7, Windows Vista ან XP ოპერაციული სისტემით, ტერმინალის ემულაციის პროგრამასთან ერთად, როგორიცაა Tera Term)

- კონსოლის კაბელები კონსოლის პორტებიდან Cisco IOS მოწყობილობების კონფიგურაციისთვის
- Ethernet და სერიალური კაბელები, ისეთი როგორიც ნაჩვენებია ტოპოლოგიაზე

ნაწილი №1: მოწყობილობის ბაზისური პარამეტრების კონფიგურაცია

პირველ ნაწილში თქვენ ააწყობთ ქსელურ ტოპოლოგიას და დააკონფიგურებთ მარშრუტიზატორის ბაზისურ პარამეტრებს, როგორიცაა ინტერფეისების IP მისამართები, მარშრუტიზაცია, მოწყობილობასთან წვდომა და პაროლები.

პირველი ეტაპი: კაბელების შეერთება ტოპოლოგიაზე ნაჩვენები სქემის მიხედვით.

მეორე ეტაპი: მარშრუტიზატორებისა და კომუტატორების ინიციალიზაცია და ხელახალი ჩატვირთვა

მესამე ეტაპი: თითოეული მარშრუტიზატორის ბაზისური პარამეტრების კონფიგურაცია

ა. გათიშეთ DNS-ის ძიება (lookup).

ბ. მოწყობილობებს მიანიჭეთ სახელები.

გ. დაშიფრეთ ყველა ღია ტექსტით მოცემული პაროლი.

დ. შექმენით მომხმარებლებისათვის გამაფრთხილებელი დღის შეტყობინების (MOTD) ბანერი ტექსტით: „არაავტორიზებული წვდომა აკრძალულია - unauthorized access is prohibited".

ე. პრივილეგირებულ EXEC რეჟიმზე დააყენეთ შიფრირებული პაროლი - **class**.

ვ. კონსოლისა და vty ხაზებზე დააყენეთ პაროლი **cisco** და გააქტიურეთ შესვლა.

ზ. დააყენეთ კონსოლის ლოგირება სინქრონულ რეჟიმში.

თ. მიანიჭეთ IP მისამართები Serial და Gigabit Ethernet ინტერფეისებს ცხრილში მოცემული ინფორმაციის მიხედვით და გააქტიურეთ ფიზიკური ინტერფეისები. ჯერჯერობით არ დააკონფიგუროთ Tunnel0 ინტერფეისი.

ი. დააყენეთ ტაქტური სიხშირე **128000** მნიშვნელობაზე DCE სერიალ ინტერფეისებისათვის.

მეოთხე ეტაპი: დააკონფიგურეთ ISP მარშრუტიზატორის ნაგულისხმევი მარშრუტები.

WEST (config) # **ip route 0.0.0.0 0.0.0.0 10.1.1.2**

EAST (config) # **ip route 0.0.0.0 0.0.0.0 10.2.2.2**

მეხუთე ეტაპი: პერსონალური კომპიუტერების კონფიგურაცია.

ცხრილის მიხედვით განუსაზღვრეთ IP მისამართები და ნაგულისხმევი გასასვლელები პერსონალურ კომპიუტერებს.

მეექვსე ეტაპი: კავშირის შემოწმება.

ამ ეტაპზე პერსონალურ კომპიუტერებს არ შეუძლიათ ერთმანეთის დაპინგვა. თითოეულ კომპიუტერს უნდა შეეძლოს საკუთარი ნაგულისხმევი გასასვლელის (Default Gateway) დაპინგვა. მარშრუტიზატორებს შეუძლიათ ტოპოლოგიის სხვა მარშრუტიზატორების სერიალური ინტერფეისების დაპინგვა. თუ ასე არაა, მაშინ მოაგვარეთ პრობლემა.

მეშვიდე ეტაპი: შეინახეთ თქვენი მიერ გაშვებული კონფიგურაცია.

ნაწილი №2: GRE ტუნელის კონფიგურაცია

მეორე ნაწილში თქვენ უნდა დააკონფიგუროთ GRE ტუნელი WEST და EAST მარშრუტიზატორებს შორის.

პირველი ეტაპი: GRE ტუნელის ინტერფეისის კონფიგურაცია.

ა. დააკონფიგურეთ tunnel ინტერფეისი WEST მარშრუტიზატორზე. გამოიყენეთ WEST მარშრუტიზატორის S0/0/0 პორტი როგორც ტუნელის წყარო (source) ინტერფეისი და 10.2.2.1 მისამართი, როგორც ტუნელის დანიშნულების ადგილი EAST მარშრუტიზატორზე.

WEST (config) # **interface tunnel 0**

WEST (config-if) # **ip iddress 172.16.12.1 255.255.255.252**

WEST (config-if) # **tunnel source s0/0/0**

WEST (config-if) # **tunnel destination 10.2.2.1**

ბ. დააკონფიგურეთ tunnel ინტერფეისი EAST მარშრუტიზატორზე. გამოიყენეთ EAST მარშრუტიზატორის S0/0/1 პორტი როგორც ტუნელის წყარო (source) ინტერფეისი და 10.1.1.1 მისამართი, როგორც ტუნელის დანიშნულების ადგილი WEST მარშრუტიზატორზე.

EAST (config) # **interface tunnel 0**

EAST (config-if) # **ip address 172.16.2.2 255.255.255.252**

EAST (config-if) # **tunnel source 10.2.2.1**

EAST (config-if) # **tunnel destination 10.1.1.1**

შენიშვნა: **tunnel source** ბრძანებისთვის წყაროდ შესაძლოა გამოყენებულ იქნას როგორც ინტერფეისის სახელი ისე IP მისამართი.

მეორე ეტაპი: GRE ტუნელის ფუნქციურობის შემოწმება.

ა. შეამოწმეთ tunnel ინტერფეისის მდგომარეობა WEST და EAST მარშრუტიზატორებზე.

```
WEST# show ip interface brief
Interface                  IP-Address      OK? Method Status                Protocol
Embedded-Service-Engine0/0 unassigned      YES unset  administratively down down
GigabitEthernet0/0         unassigned      YES unset  administratively down down
GigabitEthernet0/1         172.16.1.1      YES manual up                    up
```

```
Serial0/0/0                10.1.1.1        YES manual up                    up
Serial0/0/1                unassigned      YES unset  administratively down down
Tunnel0                    172.16.12.1     YES manual up                    up

EAST# show ip interface brief
Interface                  IP-Address      OK? Method Status                Protocol
Embedded-Service-Engine0/0 unassigned      YES unset  administratively down down
GigabitEthernet0/0         unassigned      YES unset  administratively down down
GigabitEthernet0/1         172.16.2.1      YES manual up                    up
Serial0/0/0                unassigned      YES unset  administratively down down
Serial0/0/1                10.2.2.1        YES manual up                    up
Tunnel0                    172.16.12.2     YES manual up                    up
```

ბ. გაუშვით **show interfaces tunnel 0** ბრძანება ამ ტუნელში გამოყენებული ტუნელირების პროტოკოლის, ტუნელის წყაროს და ტუნელის ადრესატის შესამოწმებლად.

ტუნელირების რა პროტოკოლია გამოყენებული? რა არის ტუნელის წყაროს და ადრესატის IP მისამართები, რომელიც დაკავშირებულია თითოეული მარშრუტიზატორის GRE ტუნელთან?

_____

_____

_____

გ. დაპინგეთ ტუნელი WEST მარშრუტიზატორიდან EAST მარშრუტიზატორამდე tunnel ინტერფეისის IP მისამართის გამოყენებით.

```
WEST# ping 172.16.12.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.12.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 32/34/36 ms
```

დ. გამოიყენეთ **traceroute** ბრძანება WEST მარშრუტიზატორზე, EAST მარშრუტიზატორზე tunnel ინტერფეისის გზის განსასაზღვრად. რა არის EAST მარშრუტიზატორამდე გზა?

_____

ე. ტუნელის საშუალებით დაპინგეთ და თვალყური ადევნეთ მარშრუტს EAST მარშრუტიზატორიდან WEST მარშრუტიზატორამდე tunnel ინტერფეისის IP მისამართის გამოყენებით.

რა არის EAST მარშრუტიზატორიდან WEST მარშრუტიზატორამდე გზა? _____

_____

რომელ ინტერფეისებთანაა დაკავშირებული ეს IP მისამართები? რატომ? _____

_____

ზ. **ping** და **traceroute** ბრძანებები უნდა იყოს წარმატებული. თუ არა, შემდეგ ეტაპზე გადასვლის წინ მოაგვარეთ პრობლემა.

ნაწილი №3: მარშრუტიზაციის ჩართვა GRE ტუნელზე

მესამე ნაწილში თქვენ უნდა დააკონფიგუროთ OSPF მარშრუტიზაცია ისე რომ WEST და EAST მარშრუტიზატორების ლოკალურ ქსელებს შეეძლოთ ერთმანეთთან დაკავშირება GRE ტუნელის გამოყენებით.

GRE ტუნელის გამართვის შემდეგ შესაძლებელია მარშრუტიზაციის პროტოკოლის რეალიზაცია. GRE ტუნელისათვის ქსელის უწყისი სერიალ ინტერფეისთან მინიჭებული ქსელის ნაცვლად უნდა შეიცავდეს ტუნელის IP ქსელს. ისევე როგორც აკეთებთ სხვა ინტერფეისებზე, როგორიცაა სერიალი და Ethernet. შეგახსენებთ რომ ISP მარშრუტიზატორი არ მონაწილეობს მარშრუტიზაციის ამ პროცესში.

პირველი ეტაპი: OSPF მარშრუტიზაციის კონფიგურაცია area 0-სთვის ტუნელზე.

ა. area 0-ის გამოყენებით WEST მარშრუტიზატორზე დააკონფიგურეთ OSPF process ID 1 172.16.1.0/24 და 172.16.12.0/24 ქსელებისთვის.

WEST (config) # **router ospf 1**

WEST (config-router) # **network 172.16.1.0 0.0.0.255 area 0**

WEST (config-router) # **network 172.16.12.0 0.0.0.3 area 0**

ბ. area 0-ის გამოყენებით EAST მარშრუტიზატორზე დააკონფიგურეთ OSPF process ID 1 172.16.2.0/24 და 172.16.12.0/24 ქსელებისთვის.

EAST (config) # **router ospf 1**

EAST (config-router) # **network 172.16.2.0 0.0.0.255 area 0**

EAST (config-router) # **network 172.16.12.0 0.0.0.3 area 0**

მეორე ეტაპი: OSPF მარშრუტიზაციის შემოწმება

ა. WEST მარშრუტიზატორიდან გაუშვით **show ip route** ბრძანება, EAST მარშრუტიზატორზე 172.16.2.0/24 ლოკალური ქსელთან მარშრუტის შესამოწმებლად.

```
WEST# show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       + - replicated route, % - next hop override

Gateway of last resort is 10.1.1.2 to network 0.0.0.0

S*    0.0.0.0/0 [1/0] via 10.1.1.2
      10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        10.1.1.0/30 is directly connected, Serial0/0/0
L        10.1.1.1/32 is directly connected, Serial0/0/0
      172.16.0.0/16 is variably subnetted, 5 subnets, 3 masks
C        172.16.1.0/24 is directly connected, GigabitEthernet0/1
L        172.16.1.1/32 is directly connected, GigabitEthernet0/1
O        172.16.2.0/24 [110/1001] via 172.16.12.2, 00:00:07, Tunnel0
C        172.16.12.0/30 is directly connected, Tunnel0
L        172.16.12.1/32 is directly connected, Tunnel0
```

რა არის გასასვლელი ინტერფეისი და IP მისამართი 172.16.2.0/24 ქსელთან დასაკავშირებლად? _____

ბ. EAST მარშრუტიზატორიდან გაუშვით show ip route ბრძანება, WEST მარშრუტიზატორზე 172.16.1.0/24 ლოკალური ქსელთან მარშრუტის შესამოწმებლად.

რა არის გასასვლელი ინტერფეისი და IP მისამართი 172.16.1.0/24 ქსელთან დასაკავშირებლად? _____

მესამე ეტაპი: ერთმანეთთან კავშირის შემოწმება

ა. PC-A-დან დაპინგეთ PC-C. პინგი უნდა იყოს წარმატებული, თუ არა, მოაგვარეთ პრობლემა სანამ არ გექნებათ ბოლომდე დასრულებული კავშირი.

შენიშვნა: შესაძლოა აუცილებელი გახდეს პერსონალური კომპიუტერის ფაიერვოლის გათიშვა კომპიუტერებს შორის პინგის გასაშვებად.

ბ. PC-A-დან განახორციელეთ PC-C-ს მარშრუტის თვალთვალი (Traceroute). რა არის PC-A-დან PC-C-მდე გზა? _____

ასახვა

1. სხვა რა კონფიგურაციებია საჭირო დაცული GRE ტუნელის შესაქმნელად?

_____

2. თუ თქვენ დაამატებთ WEST ან EAST მარშრუტიზატორების ლოკალურ ქსელებს, რა უნდა გააკეთოთ იმისათვის რომ ტრაფიკისთვის ქსელმა გამოიყენოს GRE ტუნელი?

_____

_____

_____

მარშრუტიზატორის ინტერფეისის შემაჯამებელი ცხრილი:

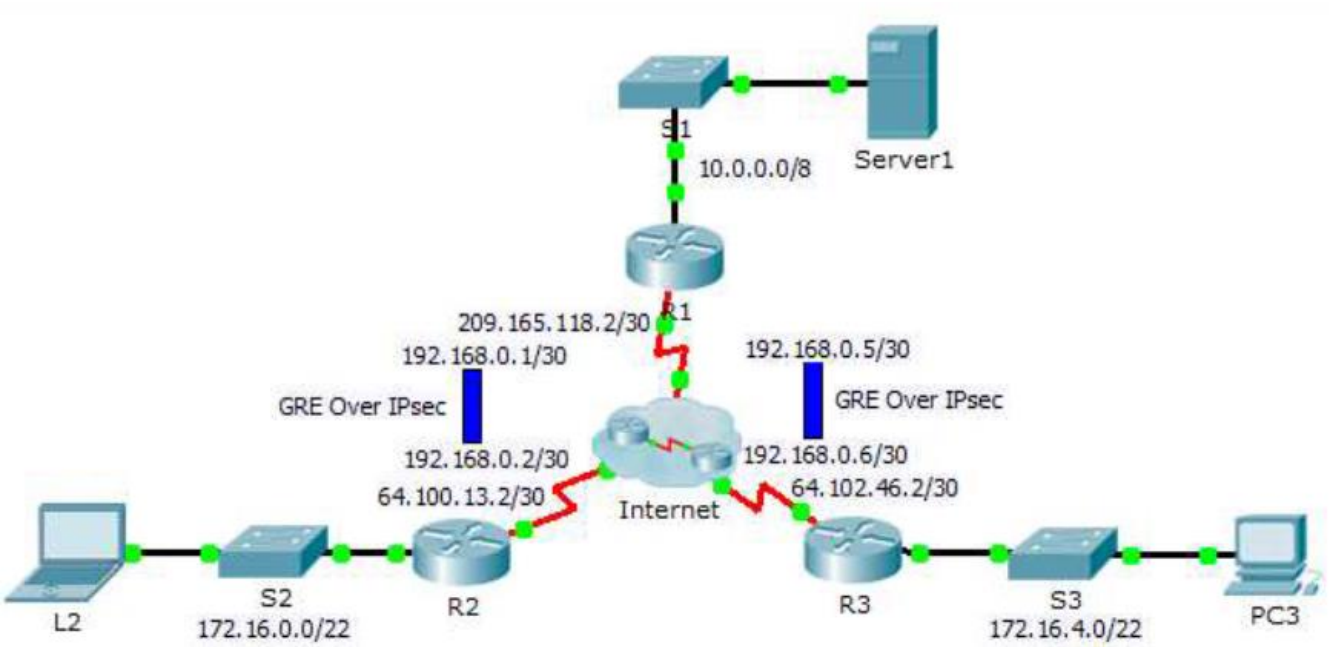
მარშრუტიზატორის ინტერფეისის შეჯამება				
მარშრუტიზატორის მოდელი	Ethernet interface №1	Ethernet interface №2	Serial interface №1	Serial interface №2
1800	FastEthernet 0/0 (F0/0)	FastEthernet 0/1 (F0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)

1900	GigabitEthernet 0/0 (G0/0)	GigabitEthernet 0/1 (G0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
2801	FastEthernet 0/0 (F0/0)	FastEthernet 0/1 (F0/1)	Serial 0/1/0 (S0/0/0)	Serial 0/1/1 (S0/0/1)
2811	FastEthernet 0/0 (F0/0)	FastEthernet 0/1 (F0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
2900	GigabitEthernet 0/0 (G0/0)	GigabitEthernet 0/1 (G0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)

შენიშვნა: თუ გინდათ მოძებნოთ თუ როგორაა მარშრუტიზატორი კონფიგურირებული, დაათვალიერეთ ინტერფეისები რათა მოახდინოთ მარშრუტიზატორის იდენტიფიკაცია და დაადგინოთ რამდენი ინტერფეისი აქვს მას. აქ არ არის იმის საშუალება, რომ ჩამოიწეროს თითოეული მარშრუტიზატორის კონფიგურაციების კომბინაციები. მოცემული ცხრილი მოიცავს მოწყობილობის **Ethernet** და **Serial** ინტერფეისების შესაძლო კომბინაციების იდენტიფიკატორებს. ეს ცხრილი არ მოიცავს ნებისმიერი სხვა ტიპის ინტერფეისს, რადგან კონკრეტული მარშრუტიზატორი შეიძლება შეიცავდეს მხოლოდ ერთს. ამის მაგალითი შეიძლება იყოს **ISDN BRI** ინტერფეისი. ფრჩხილებში მოცემული ცვლადი არის ლეგალური აბრევიატურა, რომელიც შეიძლება გამოყენებულ იქნას **Cisco IOS** ბრძანებებში, ინტერფეისის წარმოსადგენად.

4.2.5. *GRE-ს კონფიგურაცია IPsec-ზე (არჩევითი)*

ტოპოლოგია



მისამართების ცხრილი

მოწყობილობა	ინტერფეისი	IP მისამართი	ქვექსელის ნიღაბი	ნაგულისხმევი გასავლელი
R1	G0/0	10.0.0.1	255.0.0.0	N/A
	S0/0/0	209.165.118.2	255.255.255.252	N/A
	Tunnel0	192.168.0.1	255.255.255.252	N/A
	Tunnel1	192.168.0.5	255.255.255.252	N/A
R2	G0/0	172.16.0.1	255.255.252.0	N/A
	S0/0/0	64.100.13.2	255.255.255.252	N/A
	Tunnel0	192.168.0.2	255.255.255.252	N/A
R3	G0/0	172.16.4.1	255.255.252.0	N/A
	S0/0/0	64.102.46.2	255.255.255.252	N/A

	Tunnel0	192.168.0.6	255.255.255.252	N/A
Server1	ქსელის ადაპტერი	10.0.0.2	255.0.0.0	10.0.0.1
L2	ქსელის ადაპტერი	172.16.0.2	255.255.252.0	172.16.0.1
PC3	ქსელის ადაპტერი	172.16.4.3	255.255.252.0	172.16.4.1

შესასრულებელი დავალებები

ნაწილი №1: მარშრუტიზატორის კავშირის შემოწმება

ნაწილი №2: უსაფრთხოების ფუნქციების ჩართვა

ნაწილი №3: IPSec პარამეტრების კონფიგურაცია

ნაწილი №4: GRE ტუნელის კონფიგურაცია IPSec-ზე

ნაწილი №5: კავშირის შემოწმება

სცენარი

თქვენ ხართ იმ კომპანიის ქსელის ადმინისტრატორი, რომელსაც სურს GRE ტუნელის მომართვა IPsec-ზე დაშორებული ოფისებისთვის. ყველა ქსელი არის ლოკალურად კონფიგურირებული და საჭიროებს მხოლოდ ტუნელის და შიფრაციის კონფიგურაციას.

ნაწილი №1: მარშრუტიზატორის კავშირის შემოწმება

პირველი ეტაპი: R1 მარშრუტიზატორიდან R2 და R3 მარშრუტიზატორების დაპინგვა

ა. R1 მარშრუტიზატორიდან დაპინგეთ R2 მარშრუტიზატორის S0/0/0 ინტერფეისის IP მისამართი.

ბ. R1 მარშრუტიზატორიდან დაპინგეთ R3 მარშრუტიზატორის S0/0/0 ინტერფეისის IP მისამართი.

მეორე ეტაპი: L2 და PC3 კომპიუტერებიდან Server1-ის დაპინგვა.

სცადეთ L2 ლეპტოპიდან Server1-ის IP მისამართის დაპინგვა. ჩვენ გავიმეორებთ ამ ტესტს GRE ტუნელის IPsec-ზე კონფიგურაციის შემდეგაც. რა არის პინგის შედეგი? რატომ?_____

_____

მესამე ეტაპი: PC3 კომპიუტერის დაპინგვა L2 ლეპტოპიდან.

სცადეთ L2 ლეპტოპიდან PC3-ის IP მისამართის დაპინგვა. ჩვენ გავიმეორებთ ამ ტესტს GRE ტუნელის IPsec-ზე კონფიგურაციის შემდეგაც. რა არის პინგის შედეგი? რატომ?_____

_____

ნაწილი №2: უსაფრთხოების ფუნქციების ჩართვა

პირველი ეტაპი: security9 მოდულის აქტივაცია.

ამ დავალების შესასრულებლად საჭიროა უსაფრთხოების ტექნოლოგიის პაკეტის ლიცენზიის ჩართვა.

ა. გაუშვით show version ბრძანება მომხმარებლის EXEC ან პრივილეგირებულ EXEC რეჟიმში, რათა შევამოწმოთ ჩართულია თუ არა უსაფრთხოების ტექნოლოგიის პაკეტის ლიცენზია.

```
-------------------------------------------------------------------
Technology    Technology-package          Technology-package
              Current       Type          Next reboot
-------------------------------------------------------------------
ipbase        ipbasek9      Permanent     ipbasek9
security      None          None          None
uc            None          None          None
data          None          None          None

Configuration register is 0x2102
```

ბ. თუ არ არის ჩართული, გააქტიურეთ securityk9 მოდული მარშრუტიზატორის შემდგომი ჩატვირთვისთვის, დაეთანხმეთ ლიცენზიას, შეინახეთ კონფიგურაცია და გადატვირთეთ.

R1 (config) # license boot module c2900 technology-packege securityk9

<Accept the License>

R1 (config) # end

R1 # copy running-config startup-config

R1 # reload

გ. ხელახალი ჩატვირთვის პროცესის დასრულების შემდეგ, გაუშვით show version ბრძანება ხელახლა უსაფრთხოების ტექნოლოგიის პაკეტის ლიცენზიის აქტივაციის შესამოწმებლად.

```
Technology Package License Information for Module:'c2900'

-----------------------------------------------------------------
Technology    Technology-package          Technology-package
              Current       Type          Next reboot
-----------------------------------------------------------------
ipbase        ipbasek9      Permanent     ipbasek9
security      securityk9    Evaluation    securityk9
uc            None          None          None
data          None          None          None
```

დ. გაიმეორეთ 1ა-1გ ეტაპები R2 და R3 მარშრუტიზატორებისათვის.

ნაწილი №3: IPsec პარამეტრების კონფიგურაცია

პირველი ეტაპი: საინტერესო ტრაფიკის იდენტიფიკაცია R1 მარშრუტიზატორზე.

ა. დააკონფიგურეთ 101 წვდომის კონტროლის სია (ACL) R1 მარშრუტიზატორის ლოკალური ქსელიდან R2 და R3 მარშტურიზატორების ლოკალურ ქსელამდე საინტერესოდ მიჩნეული ტრაფიკის იდენტიფიცირებისათვის. მოცემული საინტერესო ტრაფიკი გამოიწვევს IPsec VPN-ის რეალიზაციას, როგორც კი იქნება ტრაფიკი R1 და R2-R3 მარშრუტიზატორების ლოკალურ ქსელებს შორის. სხვა ლოკალური ქსელიდან

გაშვებული ტრაფიკი არ დაიშიფრება. შეგახსენებთ, იმის გამო რომ გაშვებულია ნაგულისხმევი ნებისმიერის აკრძალვა (deny any) წესი, არ არის საჭირო შესაბამისი უწყისის დამატება სიაში.

R1 (config) # access-list 101 permit ip 10.0.0.0 0.255.255.255 172.16.0.0 0.0.3.255

ბ. გაიმეორეთ 1ა ეტაპი ACL 101-ის კონფიგურაციისთვის, რათა მოხდეს R3 მარშრუტიზატორის ლოკალურ ქსელზე არსებული ტრაფიკის საინტერესოდ ცნობისთვის.

მეორე ეტაპი: ISAKMP ფაზა 1-ის თვისებების დაკონფიგურება R1 მარშრუტიზატორზე.

ა. დააკონფიგურეთ კრიპტოგრაფიული ISAKMP პოლიტიკა 101-ის თვისებები R1 მარშრუტიზატორზე გაზიარებულ კრიპტოგრაფიულ გასაღებ-cisco-სთან ერთად. ნაგულისხმევი მნიშვნელობები არ არის აუცილებელი იყოს კონფიგურირებული, ამიტომ მხოლოდ შიფრაცია, გასაღების გაცვლის მეთოდი და DH მეთოდი უნდა იქნას დაკონფიგურებული.

R1 (config) # crypto isakmp policy 101

R1 (config-isakmp) # encryption aes

R1 (config-isakmp) # authentication pre-share

R1 (config-isakmp) # group 5

R1 (config-isakmp) # exit

ბ. შექმენით isakmp გასაღებები R1 მარშრუტიზატორის თითოეული წევრისათვის.

R1 (config) # crypto isakmp key cisco address 64.100.13.2

R1 (config) # crypto isakmp key cisco address 64.102.46.2

მესამე ეტაპი: დააკონფიგურეთ ISAKMP ფაზა 2-ის თვისებები R1 მარშრუტიზატორზე.

ა. შექმენით გარდაქმნის ნაკრები (transform-set) VPN-SET, esp-aes და esp-sha-hmac-ის გამოსაყენებლად. შემდეგ შექმენით კრიპტოგრაფიული რუქა VPN-MAP, რომელიც შეკრავს ყვეა ფაზა 2-ის პარამეტრებს ერთად. გამოიყენეთ 101 რიგითი ნომერი და განსაზღვრეთ ის, როგორც ipsec-isakmp რუკა.

R1 (config) # crypto ipsec transform-set R1_Set esp-aes esp-sha-hmac

R1 (config) # crypto map R1_Map 101 ipsec-isakmp

R1 (config-crypto-map) # set peer 64.100.13.2

R1 (config-crypto-map) # set peer 64.102.46.2

R1 (config-crypto-map) # set transform-set R1_Set

R1 (config-crypto-map) # match address 101

R1 (config-crypto-map) # exit

მეოთხე ეტაპი: კრიპტოგრაფიული რუკის კონფიგურაცია გამავალ ინტერფეისზე.

ბოლოს მიაბით R1-Map კრიპტოგრაფიული რუკა გამავალ S0/0/0 ინტერფეისს.

R1 (config) # interface S0/0/0

R1 (config-if) # crypto map R1_Map

მეხუთე ეტაპი: IPsec პარამეტრების კონფიგურაცია R2 და R3 მარშრუტიზატორებზე

გაიმეორეთ 1-4 ეტაპები R2 და R3 მარშრუტიზატორებზე. შეცვალეთ R1 მარშრუტიზატორის ნაკრების და რუკის სახელები R2 და R3 მარშრუტიზატორებზე. გამოიყენეთ იგივე გაფართოებული წვდომის კონტროლის სიის (ACL) ნომერი 101. შევნიშნავთ რომ თითოეულ მარშრუტიზატორს სჭირდება მხოლოდ ერთი დაშიფრული კავშირი R1 მარშრუტიზატორთან. არ არსებობს დაშიფრული კავშირი R2 და R3 მარშრუტიზატორებს შორის.

ნაწილი №4: GRE ტუნელების IPSec-ზე კონფიგურაცია

პირველი ეტაპი: Tunnel ინტერფეისის კონფიგურაცია R1 მარშრუტიზატორზე.

ა. შედით R1 მარშრუტიზატორის Tunnel 0 ინტერფეისის კონფიგურაციის რეჟიმში.

R1 (config) # interface tunnel 0

ბ. მომართეთ IP მისამართი ისე როგორც ნაჩენებია მისამართების ცხრილში.

R1 (config-if) # ip address 192.168.0.1 255.255.255.252

გ. მომართეთ tunnel 0-ის საბოლოო წერტილებისათვის წყარო და ადრესატი.

R1 (config-if) # tunnel source s0/0/0

R1 (config-if) # tunnel destination 64100.13.2

დ. დაკონფიგურეთ Tunnel 0 IP ტრაფიკის გადასაცემად GRE-ს საშუალებით.

R1 (config-if) # tunnel mode gre ip

ე. Tunnel 0 ინტერფეისი შეიძლება უკვე აქტივირებული იყოს. თუ ეს ასე არ აღმოჩნდა, გამოასწორეთ სხვა ინტერფეისების მსგავსად.

ვ. გაიმეორეთ 1ა-1ე ეტაპები R3 მარშრუტიზატორზე Tunnel 1 ინტერფეისის შესაქმნელად. შეცვალეთ დამისამართება საჭიროების შემთხვევაში.

მეორე ეტაპი: Tunnel 0 ინტერფეისის კონფიგურაცია R2 და R3 მარშრუტიზატორებზე.

ა. გაიმეორეთ 1ა-ე ეტაპები R2 მარშრუტიზატორისთვის. დარწმუნდით რომ IP დამისამართება შეცვალეთ ცხრილის შესაბამისად.

ბ. გაიმეორეთ 1ა-ე ეტაპები R3 მარშრუტიზატორისთვის. დარწმუნდით რომ IP დამისამართება შეცვალეთ ცხრილის შესაბამისად.

მესამე ეტაპი: მარშრუტის კონფიგურაცია კერძო IP ტრაფიკისთვის.

ა. განსაზღვრეთ მარშრუტი R1 მარშრუტიზატორიდან 172.16.0.0 და 172.16.4.0 ქსელებში tunnel ინტერფეისის შემდეგი-ნახტომი (Next-hop) მისამართის გამოყენებით.

ბ. განსაზღვრეთ მარშრუტი R2 და R3 მარშრუტიზატორებიდან 10.0.0.0 ქსელში tunnel ინტერფეისის შემდეგი-ნახტომი (Next-hop) მისამართის გამოყენებით.

ნაწილი №5: კავშირის შემოწმება

პირველი ეტაპი: L2 და PC3 კომპიუტერებიდან Server1-ის დაპინგვა.

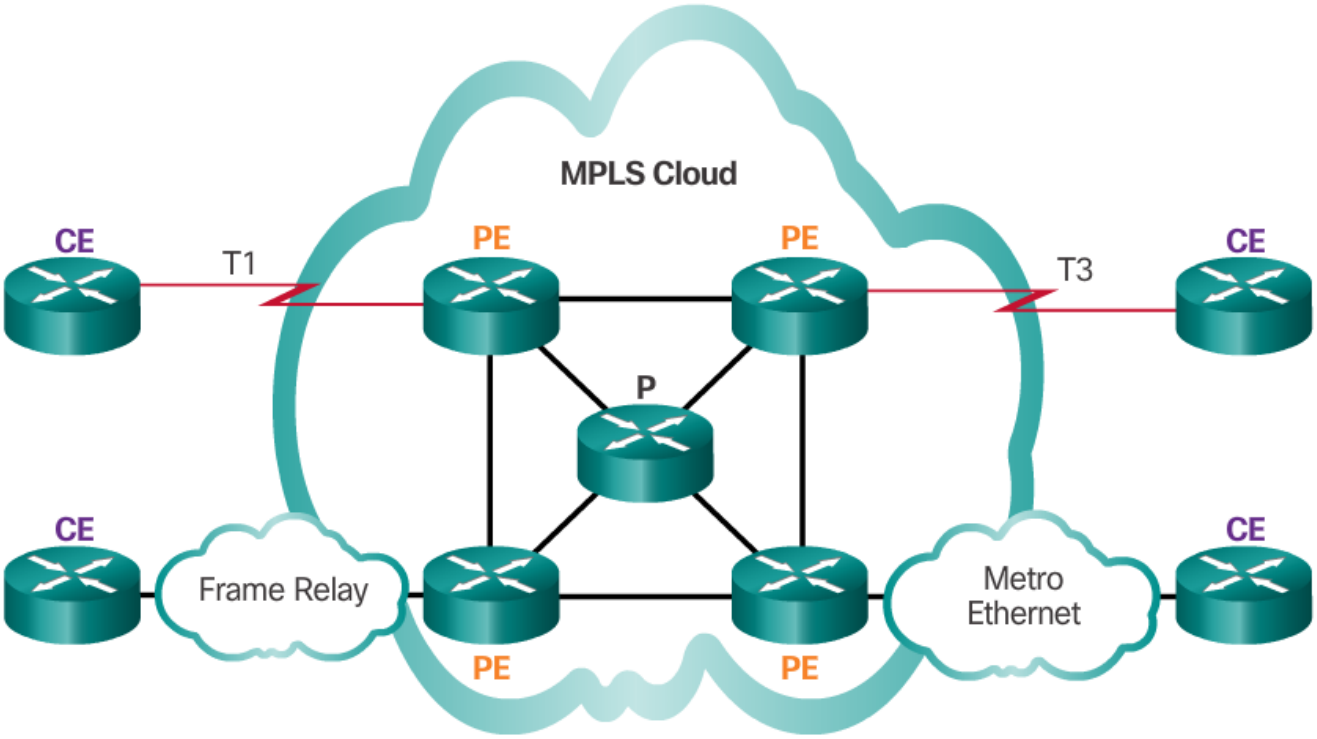
ა. სცადეთ L2 ლეპტოპიდან და PC3 კომპიუტერიდან Server1-ის IP მისამართის დაპინგვა. პინგი უნდა იყოს წარმატებული.

ბ. სცადეთ L2 ლეპტოპის დაპინგვა PC3 კომპიუტერიდან. პინგი უნდა იყოს წარუმატებელი, რადგან არ არსებობს ტუნელი ორ ქსელს შორის.

4.3. MPLS-ის საჭიროებისა და მისი დანიშნულების შეფასება

Multiprotocol Label Switching (MPLS) არის მრავალპროტოკოლიანი მაღალი წარმადობის WAN ტექნოლოგია, რომელიც აგზავნის მონაცემებს ერთი მარშრუტიზატორიდან მეორეში მოკლე გზის მარკირებების საფუძველზე და არა IP ქსელის მისამართებით.

MPLS აქვს რამდენიმე განმსაზღვრელი მახასიათებელი. ის არის მრავალპროტოკოლიანი რაც იმას ნიშნავს, რომ მას აქვს უნარი გადაიტანოს ნებისმიერი „ტვირთი“ IPv4, IPv6, Ethernet, ATM, DSL, და Frame Relay ტრაფიკის ჩათვლით. ის იყენებს მარკერებს, რომელიც ეუბნევა მარშრუტიზატორს რა უნდა გააკეთოს პაკეტებთან. მარკერები განსაზღვრავენ გზებს დაშორებულ მარშრუტიზატორებს შორის, და არა საბოლოო მოწყობილობების გზებს და სანამ MPLS ჩვეულებრივ ახდენს IPv4 და IPv6 პაკეტების მარშრუტიზაციას, ყველა დანარჩენი არის კომუტირებული.



სურ. 4.3.1 - მარტივი MPLS ტოპოლოგია

MPLS არის სერვისების მომწოდებლის ტექნოლოგია. გამოყოფილი ხაზები აწვდიან ბიტებს ადგილებს შორის, და Frame Relay და Ethernet WAN გადასცემენ ფრეიმებს ადგილებს

შორის. თუმცა MPLS შეუძლია მიაწოდოს ნებისმიერი ტიპის პაკეტი ადგილებს შორის. MPLS-ს შეუძლია პაკეტების სხვადასხვა ქსელური პროტოკოლების ენკაპსულაცია ის მხარს უჭერს WAN ტექნოლოგიების ფართო არეალს როგორიცაა: T-carrier / E-carrier links, Carrier Ethernet, ATM, Frame Relay, და DSL.

სურათზე გამოსახული მარტივი ტოპოლოგია გვიჩვენებს თუ როგორ გამოიყენება MPLS. აღსანიშნავია რომ სხვა ადგილებს შეუძლიათ დაკავშირება MPLS clou-თან განსხვავებული წვდომის ტექნოლოგიების გამოყენებით. სურათზე CE მიმართავს მომხმარებლის მხარეს, PE არის პროვაიდერის მხარის მარშრუტიზატორი, რომელიც ამატებს და შლის მარკერებს, ხოლო P არის პროვაიდერის შიდა მარშრუტიზატორი, რომელიც ახდენს MPLS-ის მიერ მარკირებული პაკეტების კომუტირებას.

შენიშვნა: MPLS პირველ რიგში არის სერვისების პროვაიდერის WAN ტექნოლოგია.

პრაქტიკული სავარჯიშო

1. შეასრულეთ IPsec პარამეტრების კონფიგურაცია მარშრუტიზატორზე
2. მოახდინეთ IPsec VPN-ის შემოწმება
3. შეასრულეთ GRE ტუნელების კონფიგურაცია
4. გადაწყვეტეთ GRE (Generic Routing Encapsulation)-ს პრობლემები
5. განახორციელეთ წერტილიდან წერტილამდე (Point-to-Point) GRE VPN ტუნელის კონფიგურაცია
6. შეასრულეთ GRE-ს კონფიგურაცია IPsec-ზე

ცოდნის შეფასება

სტუდენტებს მიეცემათ პრაქტიკული დავალება

- შეასრულონ securityk9 მოდულის აქტივაცია, IPsec პარამეტრების კონფიგურაცია მარშრუტიზატორზე, ISAKMP ფაზა 1 პარამეტრების კონფიგურაცია მარშრუტიზატორზე, ISAKMP ფაზა 2-ის პარამეტრების კონფიგურაცია მარშრუტიზატორზე, კრიპტოგრაფიის რუკის კონფიგურაცია გამავალ ინტერფეისზე, IPsec პარამეტრების კონფიგურაცია მარშრუტიზატორზე, კრიპტოგრაფიის ISAKMP პოლიტიკა 10-ის პარამეტრების კონფიგურაცია;
- მოახდინონ მარშრუტიზატორის კავშირის შემოწმება, GRE ტუნელების კონფიგურაცია, Tunnel 0 ინტერფეისის კონფიგურაცია მარშრუტიზატორზე;
- განახორციელონ კაბელების შეერთება ტოპოლოგიაზე ნაჩვენები სქემის მიხედვით, მარშრუტიზატორებისა და კომუტატორების ინიციალიზაცია და ხელახალი ჩატვირთვა, თითოეული მარშრუტიზატორის ბაზისური პარამეტრების კონფიგურაცია, GRE ტუნელის კონფიგურაცია, GRE ტუნელის ინტერფეისის კონფიგურაცია, მარშრუტიზაციის ჩართვა GRE ტუნელზე;
- შეასრულონ უსაფრთხოების ფუნქციების ჩართვა, IPsec პარამეტრების კონფიგურაცია, საინტერესო ტრაფიკის იდენტიფიკაცია R1 მარშრუტიზატორზე, ISAKMP ფაზა 1-ის თვისებების დაკონფიგურება მარშრუტიზატორზე, ISAKMP ფაზა 2-ის თვისებების კონფიგურაცია მარშრუტიზატორზე, GRE ტუნელების IPSec-ზე კონფიგურაცია, მარშრუტის კონფიგურაცია კერძო IP ტრაფიკისთვის.

შემფასებელი აკვირდება შესაფასებელ პირის მუშაობას პროფესიული სტანდარტით (პროგრამით / მოდულით) განსაზღვრული ამოცანების შესრულების პროცესში. დაკვირვება ხორციელდება კომპიუტერებით აღჭურვილ ლაბორატორიაში, სადაც შესაფასებელი პირი პრაქტიკულ საქმიანობას ეწევა. შემფასებელმა წინასწარ უნდა დაგეგმოს დაკვირვების პროცესი, იმის დასადგენად, თუ რამდენად სწორად იყენებს შესაფასებელი პირი ცოდნას, უნარებსა და ყველა რესურსს შედეგის მისაღწევად.

შეფასება განხორციელდება პროცესზე დაკვირვებით, წინასწარ განსაზღვრული შეფასების ინდიკატორების საფუძველზე.

დავალების ნიმუში და შეფასების რუბრიკა

პროცესზე დაკვირვება

- *შეასრულოს IPsec პარამეტრების კონფიგურაცია R1 მარშრუტიზატორზე, IPsec VPN-ის შემოწმება, შეასრულეთ GRE ტუნელების კონფიგურაცია*
- მოახდინონ GRE (Generic Routing Encapsulation)-ს პრობლემების გადაწყვეტა, წერტილიდან წერტილამდე (Point-to-Point) GRE VPN ტუნელის კონფიგურაცია.

სწავლის შედეგი	N	დასახელება	შეფასება	
			კი	არა
მიდა და გარე კომუნიკაციები, უსაფრთხო კავშირები და WAN ჩართვები (VPN, MPLS Applications)	1.	შეასრულა IPsec პარამეტრების კონფიგურაცია მარშრუტიზატორზე		
	2.	შეასრულა securityk9 მოდულის აქტივაცია		
	3.	შეასრულა ISAKMP ფაზა 1, ფაზა 2 პარამეტრების კონფიგურაცია მარშრუტიზატორზე		
	4.	შეასრულა IPsec VPN-ის შემოწმება		
	5.	შეასრულა GRE ტუნელების კონფიგურაცია		
	6.	შეასრულა GRE (Generic Routing Encapsulation)-ს პრობლემები		
	7.	შეასრულა წერტილიდან წერტილამდე (Point-to-Point) GRE VPN ტუნელის კონფიგურაცია		
	8.	შეასრულა GRE-ს კონფიგურაცია IPsec-ზე		
	9.	შეასრულა Tunnel 0 ინტერფეისის კონფიგურაცია მარშრუტიზატორზე		
	10.	შეასრულა მარშრუტის კონფიგურაცია კერძო IP ტრაფიკისთვის		

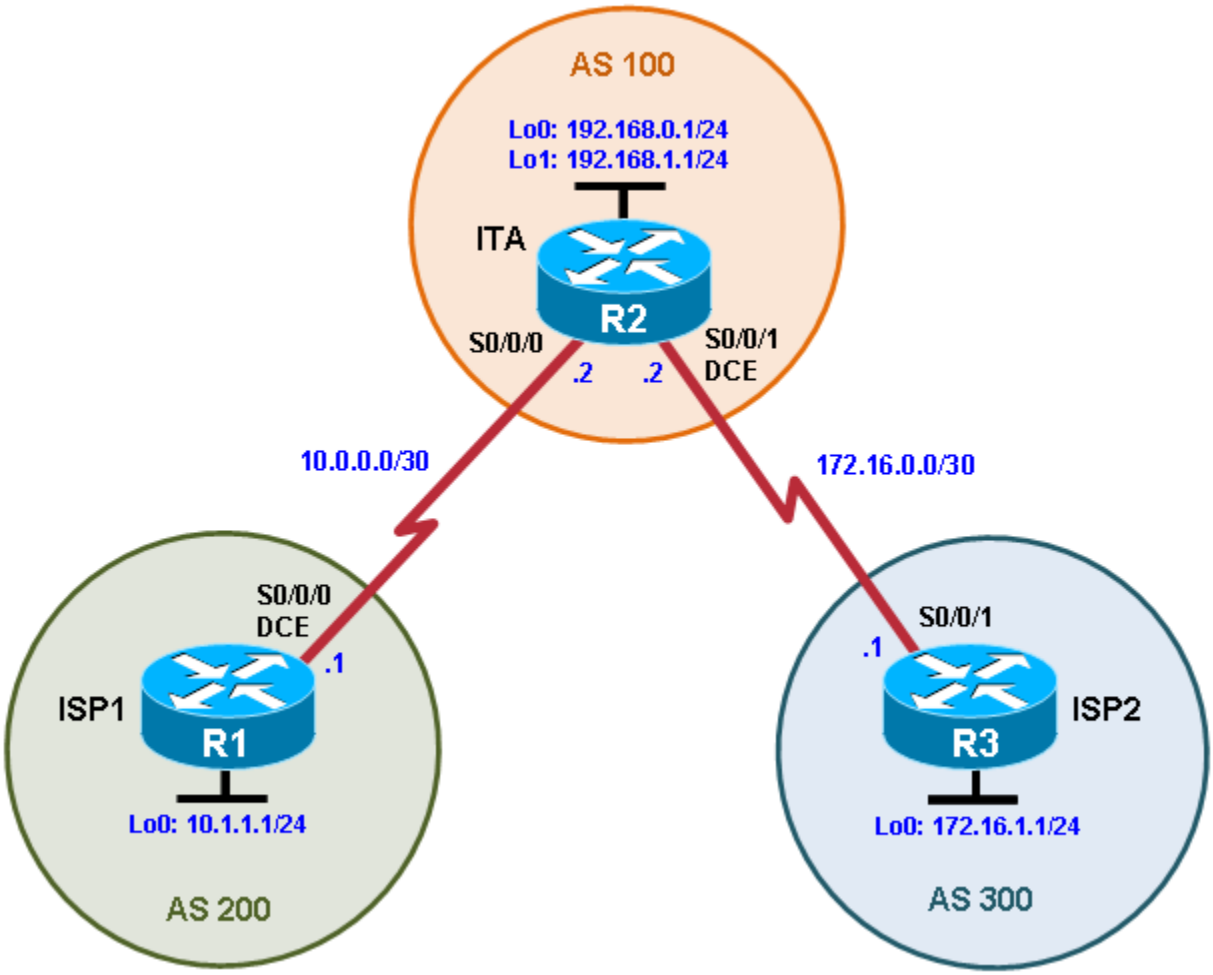
სწავლის შედეგი ჩაითვლება მიღწეულად თუ სტუდენტმა შეძლო შედეგის მინიმუმ 8 პუნქტის შესრულება.

5. მესამე დონის მარშრუტიზაციის პროტოკოლი BGP

5.1. BGP პროტოკოლის კონფიგურირება.

5.1.1. *BGP-ის კონფიგურაცია ნაგულისხმევი მარშრუტით*

ტოპოლოგია



შესასრულებელი დავალებები:

- BGP-ის კონფიგურაცია ორ ინტერნეტ-პროვაიდერს შორის მარშრუტიზაციის ინფორმაციის გასაცვლელად.

ზოგადი ინფორმაცია

საერთაშორისო ტურისტული სააგენტო - International Travel Agency (ITA) აქტიურად ემყარება ინტერნეტ გაყიდვებს. ამ მიზეზით, ITA-მ გადაწყვიტა მრავალქსელიანი ISP კავშირის გადაწყვეტის შექმნა და გააფორმა კონტრაქტი ორ ინტერნეტ პროვაიდერთან მდგრადი (Fault tolerance) ინტერნეტ კავშირისთვის. რადგან ITA უკავშირდება ორ სხვადასხვა სერვის პროვაიდერს, თქვენ უნდა დააკონფიგუროთ BGP, რომელიც გადის სასაზღვრო მარშრუტიზატორსა და ორ ინტერნეტ-პროვაიდერის მარშრუტიზატორს შორის.

შენიშვნა: მოცემული ლაბორატორიული სამუშაო იყენებს Cisco IOS 15.4 გამოშვების მქონე Cisco 1941 მარშრუტიზატორებს IP ბაზით. კომუტატორები არის Cisco WS-C2960-24TT-L Fast Ethernet ინტერფეისებით, ამიტომ მაშრუტიზატორი გამოიყენებს მარშუტიზაციის მაჩვენებლებს (Metrics), რომელიც დაკავშირებულია 100მბ/წმ-იან ინტერფეისთან. მარშრუტიზატორის ან კომუტატორის და Cisco IOS პროგრამული უზრუნველყოფის ვერსიის მიხედვით, ხელმისაწვდომი ბრძანებები და მიღებული შედეგები შესაძლოა განსხვავებული იყოს იმისგან რაც მოცემულია ამ ლაბორატორიულ დავალებაში.

მოთხოვნილი რესურსები:

- 3 მარშრუტიზატორი (Cisco IOS Release 15.2 ან მსგავსი)
- სერიალური და Ethernet კაბელები

ეტაპი №0: შემოთავაზებული საწყისი კონფიგურაციები.

ა. გამოიყენეთ ქვემოთ მოცემული კონფიგურაცია თითოეული მარშრუტიზატორის-თვის, შესაბამის **hostname**-თან ერთად. **Exec-timeout 0 0** ბრძანების გამოყენება უნდა მოხდეს მხოლოდ ლაბორატორიულ გარემოში.

```
Router(config)# no ip domain-lookup
Router(config)# line con 0
Router(config-line)# logging synchronous
```

Router(config-line)# **exec-timeout 0 0**

პირველი ეტაპი: ინტერფეისის მისამართების კონფიგურაცია.

ა. დიაგრამაზე მოცემული მისამართების სქემის გამოყენებით შექმენით უკუკავშირის მისამართის (loopback) ინტერფეისები და გამოიყენეთ IPv4 მისამართები მათთვის და სერიალური ინტერფეისებისათვის, ISP1 (R1), ISP2 (R3), და ITA (R2) მარშრუტიზატორებზე. ISP loopback-ები იმიტირებენ რეალურ ქსელებს, რომლებიც შეიძლება იქნენ დაკავშირებულნი ISP-ით. ITA მარშრუტიზატორის ორი loopback-ი ახდენს ITA სასაზღვრო მარშრუტიზატორსა და მათ ძირითად (Core) მარშრუტიზატორებს შორის კავშირის სიმულაციას. მომართეთ ტაქტური სიხშირე (clock rate) DCE სერიალურ ინტერფეისებზე.

ISP1(config)# **interface Lo0**

ISP1(config-if)# **description ISP1 Internet Network**

ISP1(config-if)# **ip address 10.1.1.1 255.255.255.0**

ISP1(config-if)# **exit**

ISP1(config)# **interface Serial0/0/0**

ISP1(config-if)# **description ISP1 -> ITA**

ISP1(config-if)# **ip address 10.0.0.1 255.255.255.252**

ISP1(config-if)# **clock rate 128000**

ISP1(config-if)# **no shutdown**

ISP1(config-if)# **end**

ISP1#

ITA(config)# **interface Lo0**

ITA(config-if)# **description Core router network link 1**

ITA(config-if)# **ip address 192.168.0.1 255.255.255.0**

ITA(config)# **exit**

ITA(config-if)# **interface Lo1**

ITA(config-if)# **description Core router network link 2**

ITA(config-if)# **ip address 192.168.1.1 255.255.255.0**

ITA(config-if)# **exit**

ITA(config)# **interface Serial0/0/0**

ITA(config-if)# **description ITA -> ISP1**

ITA(config-if)# **ip address 10.0.0.2 255.255.255.252**

ITA(config-if)# **no shutdown**

ITA(config-if)# **exit**

ITA(config)# **interface Serial0/0/1**

ITA(config-if)# **description ITA -> ISP2**

ITA(config-if)# **ip address 172.16.0.2 255.255.255.252**

ITA(config-if)# **clock rate 128000**

ITA(config-if)# **no shutdown**

ITA(config-if)# **end**

ITA#

ISP2(config)# **interface Lo0**

ISP2(config-if)# **description ISP2 Internet Network**

ISP2(config-if)# **ip address 172.16.1.1 255.255.255.0**

ISP2(config)# **exit**

ISP2(config-if)# **interface Serial0/0/1**

ISP2(config-if)# **description ISP2 -> ITA**

ISP2(config-if)# **ip address 172.16.0.1 255.255.255.252**

ISP2(config-if)# **no shutdown**

ISP2(config-if)# **end**

ISP2#

ბ. გამოიყენეთ **ping** ბრძანება პირდაპირ დაკავშირებულ მარშრუტიზატორებს შორის კავშირის შესამოწმებლად. შენიშვნა: ISP1 მარშრუტიზატორს არ შეუძლია ISP2-თან წვდომის განხორციელება.

მეორე ეტაპი: BGP-ს კონფიგურაცია ISP მარშრუტიზატორებზე.

ISP1 და ISP2 მარშრუტიზატორებზე დააკონფიგურეთ BGP, რათა გაუთანაბრდნენ ITA სასაზღვრო მარშრუტიზატორს და განათავსოს ISP loopback ქსელები.

ISP1(config)# **router bgp 200**
ISP1(config-router)# **neighbor 10.0.0.2 remote-as 100**
ISP1(config-router)# **network 10.1.1.0 mask 255.255.255.0**

ISP2(config)# **router bgp 300**
ISP2(config-router)# **neighbor 172.16.0.2 remote-as 100**
ISP2(config-router)# **network 172.16.1.0 mask 255.255.255.0**

მესამე ეტაპი: BGP-ის კონფიგურაცია ITA სასაზღვრო მარშრუტიზატორზე.

ა. დააკონფიგურეთ ITA მარშრუტიზატორი ორივე სერვის-პროვაიდერთან BGP-ის გასაშვებად.

ITA(config)# **router bgp 100**
ITA(config-router)# **neighbor 10.0.0.1 remote-as 200**
ITA(config-router)# **neighbor 172.16.0.1 remote-as 300**
ITA(config-router)# **network 192.168.0.0**
ITA(config-router)# **network 192.168.1.0**

თქვენ უნდა დაინახოთ BGP მეზობელთან წვდომის შეტყობინებები კონსოლზე, ქვემოთ მოცემულის მსგავსად:

*Sep 8 16:00:21.587: %BGP-5-ADJCHANGE: neighbor 10.0.0.1 Up

ბ. კონფიგურაციის შესამოწმებლად, დაათვალიერეთ ITA-ს მარშრუტიზაციის ცხრილი **show ip route** ბრძანების გამოყენებით.

```
ITA# show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override
Gateway of last resort is not set
      10.0.0.0/8 is variably subnetted, 3 subnets, 3 masks
C        10.0.0.0/30 is directly connected, Serial0/0/0
L        10.0.0.2/32 is directly connected, Serial0/0/0
B        10.1.1.0/24 [20/0] via 10.0.0.1, 00:01:10
      172.16.0.0/16 is variably subnetted, 3 subnets, 3 masks
C        172.16.0.0/30 is directly connected, Serial0/0/1
L        172.16.0.2/32 is directly connected, Serial0/0/1
B        172.16.1.0/24 [20/0] via 172.16.0.1, 00:00:53
      192.168.0.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.0.0/24 is directly connected, Loopback0
L        192.168.0.1/32 is directly connected, Loopback0
      192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.1.0/24 is directly connected, Loopback1
L        192.168.1.1/32 is directly connected, Loopback1
ITA#
```

ITA-ს აქვს BGP loopback ქსელების მარშრუტები თითოეულ ISP მარშრუტიზატორთან.

გ. პინგების წარუმატებლობის შემთხვევაში გაუშვით ქვემოთ მოცემული Tcl სკრიპტი ყველა მარშრუტიზატორზე კავშირის შესამოწმებლად, მოაგვარეთ პრობლემა. Tcl script-დან გამოსასვლელად გამოიყენეთ **exit** ბრძანება.

შენიშვნა: გლობალური ქსელის (WAN) ქვექსელები, რომლებიც აკავშირებს ITA (R2)-ს ინტერნეტის სერვისის მომწოდებლებთან ISPs (R1 და R3) არ არიან განთავსებულნი BGP-ში, ამიტომ ინტერნეტ-მომწოდებლებიso (ISPs) ვერ შეძლებენ ერთმანეთის სერიალური ინტერფეისის მისამართის დაპინგვას.

```
ITA# tclsh
foreach address {
10.0.0.1
10.0.0.2
10.1.1.1
172.16.0.1
172.16.0.2
172.16.1.1
192.168.0.1
192.168.1.1
} {
ping $address }
```

მეოთხე ეტაპი: BGP-ს შემოწმება მარშრუტიზატორებზე.

ა. BGP-ის მუშაობის შესამოწმებლად გაუშვით **show ip bgp** ბრძანება.

```
ITA# show ip bgp
BGP table version is 5, local router ID is 192.168.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
```

x best-external, a additional-path, c RIB-compressed,

Origin codes: i - IGP, e - EGP, ? - incomplete

RPKI validation codes: V valid, I invalid, N Not found

	Network	Next Hop	Metric	LocPrf	Weight	Path
*>	10.1.1.0/24	10.0.0.1	0		0	200 i
*>	172.16.1.0/24	172.16.0.1	0		0	300 i
*>	192.168.0.0	0.0.0.0	0		32768	i
*>	192.168.1.0	0.0.0.0	0		32768	i

ITA#

რა არის ლოკალური მარშრუტიზატორის ID?

ცხრილის რომელი ვერსიაა ნაჩვენები?

მარშრუტიზატორის შემდეგ ვარსკვლავი (*) მიუთითებს რომ ის არის მოქმედი. next to a route indicates that it is valid. კვადრატული ფრჩხილი (>) მიუთითებს რომ მარშრუტი არჩეულია, როგორც საუკეთესო მარშრუტი.

ბ. ISP1-ის ფუნქციურობის შესამოწმებლად გაუშვით **show ip bgp** ბრძანება.

ISP1# **show ip bgp**

BGP table version is 5, local router ID is 10.1.1.1

Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,

r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,

x best-external, a additional-path, c RIB-compressed,

Origin codes: i - IGP, e - EGP, ? - incomplete

RPKI validation codes: V valid, I invalid, N Not found

	Network	Next Hop	Metric	LocPrf	Weight	Path

```
*> 10.1.1.0/24      0.0.0.0                  0         32768 i
*> 172.16.1.0/24  10.0.0.2                  0             0 100 300 i
*> 192.168.0.0      10.0.0.2                 0             0 100 i
*> 192.168.1.0      10.0.0.2                 0             0 100 i
ISP1#
```

ცხრილის რომელი ვერსიაა ნაჩვენები და არის თუ არა იგივე BGP ცხრილის ვერსია ITA-სთვის?

ISP1-დან, რა არის 172.16.1.0/24 ქსელამდე გზა?

ბ. ISP1 მარშრუტიზატორზე გაუშვით **shutdown** ბრძანება Loopback0-ზე. შემდეგ ITA-ზე ხელახლა გაუშვით **show ip bgp** ბრძანება.

```
ISP1(config)# interface loopback 0
ISP1(config-if)# shutdown
ISP1(config-if)#

ITA# show ip bgp
BGP table version is 6, local router ID is 192.168.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found
     Network          Next Hop            Metric LocPrf Weight Path
*> 172.16.1.0/24    172.16.0.1               0             0 300 i
*> 192.168.0.0      0.0.0.0                  0         32768 i
*> 192.168.1.0      0.0.0.0                  0         32768 i
```

ITA#

ცხრილის რომელი ვერსიაა ნაჩვენები? რატომ?

_____

_____

რა მოხდა მარშრუტზე 10.1.1.0/24 ქსელისთვის?

_____

გ. დააბრუნეთ ISP1 მარშრუტიზატორის Loopback0 ინტერფეისის მდგომარეობა უკან **no shutdown** ბრძანების გამოყენებით.

ISP1(config)# **interface loopback 0**

ISP1(config-if)# **no shutdown**

ISP1(config-if)#

დ. ITA-ზე, გაუშვით **show ip bgp neighbors** ბრძანება. ქვემოთ მოცემული არის 172.16.0.1 მეზობლის ჩვენების შედეგის ნაწილობრივი ნიმუში.

ITA# **show ip bgp neighbors**

BGP neighbor is 10.0.0.1, remote AS 200, external link

 BGP version 4, remote router ID 10.1.1.1

 BGP state = Established, up for 00:20:47

 Last read 00:00:49, last write 00:00:41, hold time is 180, keepalive interval is 60 seconds

 Neighbor sessions:

  1 active, is not multisession capable (disabled)

 Neighbor capabilities:

  Route refresh: advertised and received(new)

  Four-octets ASN Capability: advertised and received

  Address family IPv4 Unicast: advertised and received

  Enhanced Refresh Capability: advertised and received

Multisession Capability:

Stateful switchover support enabled: NO for session 1

Message statistics:

InQ depth is 0

OutQ depth is 0

	Sent	Rcvd
Opens:	1	1
Notifications:	0	0
Updates:	5	1
Keepalives:	15	17
Route Refresh:	0	0
Total:	21	19

Default minimum time between advertisement runs is 30 seconds

<output omitted>

მოცემული ბრძანების შედეგის მიხედვით, როგორია BGP-ის მდგომარეობა ამ მარშრუტიზატორსა და ISP2-ს შორის?

_____ _____

_____

რა ხნის განმავლობაში იყო მოცემული კავშირი up-ში?

_____

მეხუთე ეტაპი: მარშრუტის ფილტრების კონფიგურაცია.

ა. შეამოწმეთ ISP2-ის მარშრუტიზაციის ცხრილი **show ip route** ბრძანების გამოყენებით. ISP2-ს უნდა ჰქონდეს მარშრუტი, რომელიც ეკუთვნის ISP1-ს, 10.1.1.0 ქსელს.

ISP2# **show ip route**

<output omitted>

10.0.0.0/24 is subnetted, 1 subnets

B 10.1.1.0 [20/0] via 172.16.0.2, 00:09:26

172.16.0.0/16 is variably subnetted, 4 subnets, 3 masks

C 172.16.0.0/30 is directly connected, Serial0/0/1

L 172.16.0.1/32 is directly connected, Serial0/0/1

C 172.16.1.0/24 is directly connected, Loopback0

L 172.16.1.1/32 is directly connected, Loopback0

B 192.168.0.0/24 [20/0] via 172.16.0.2, 00:28:05

B 192.168.1.0/24 [20/0] via 172.16.0.2, 00:28:05

ISP2#

თუ ITA გამოაქვეყნებს ISP1-ის კუთვნილ მარშრუტს, ISP2 შეიტანს ამ მარშრუტს თავის ცხრილში. ISP2-მა შემდეგ შეიძლება სცადოს გადასატანი ტრაფიკის გადაგზავნა ITA მეშვეობით. ეს აქცევს ITA-ს სატრანზიტო მარშრუტიზატორად. ქვემოთ მოცემულია ISP1-ის Lo0 ინტერფეისის მარშრუტის კვალზე მიყოლის (traceroute) პროცესი.

ISP2# **traceroute 10.1.1.1**

Type escape sequence to abort.

Tracing the route to 10.1.1.1

VRF info: (vrf in name/id, vrf out name/id)

1 172.16.0.2 8 msec 4 msec 8 msec

2 * * *

3 * * *

4 * * * <control-shift-6 to break>

ISP2#

traceroute 10.1.1.1 ჩავარდება რადგან ISP1-ს არ აქვს სათვალთვალო წყარო IPv4 მისამართის მარშრუტი, 172.16.0.1. როგორც წესი BGP ქსელებში არ არის გამოქვეყნებული BGP-ში არსებული პროვაიდერებს შორის კავშირები. მარშრუტის თვალთვალი (traceroute) ISP2 Lo0 ინტერფეისის წყარო IPv4 მისამართის გამოყენებით

არის წარმატებული, რაც გვიჩვენებს რომ ITA არის მოცემული ქსელის სატრანზიტო მარშრუტიზატორი.

ISP2# **traceroute 10.1.1.1 source loopback0**

Type escape sequence to abort.

Tracing the route to 10.1.1.1

VRF info: (vrf in name/id, vrf out name/id)

1 172.16.0.2 8 msec 4 msec 8 msec

2 10.0.0.1 12 msec * 12 msec

ISP2#

ბ. დააკონფიგურეთ ITA მარშრუტიზატორი რადგან ის აქვეყნებს მხოლოდ ორივე ინტერნეტ-მომწოდებლის 192.168.0.0 და 192.168.1.0 ITA ქსელებს. ITA მარშრუტიზატორზე დააკონფიგურეთ ქვემოთ მოცემული დაშვების სია.

ITA(config)# **access-list 1 permit 192.168.0.0 0.0.1.255**

გ. გამოიყენეთ მოცემული დაშვების სია როგორც მარშრუტის ფილტრი **distribute-list** საკვანძო სიტყვის გამოყენებით BGP **neighbor** ჩვენებასთან ერთად.

ITA(config)# **router bgp 100**

ITA(config-router)# **neighbor 10.0.0.1 distribute-list 1 out**

ITA(config-router)# **neighbor 172.16.0.1 distribute-list 1 out**

დ. ხელახლა შეამოწმეთ ISP2-ის მარშრუტიზაციის ცხრილი. 10.1.1.0, ISP1-ის მარშრუტი კვლავ უნდა იყოს ცხრილში.

ISP2# **show ip route**

<output omitted>

10.0.0.0/24 is subnetted, 1 subnets

B 10.1.1.0 [20/0] via 172.16.0.2, 00:25:14

172.16.0.0/16 is variably subnetted, 4 subnets, 3 masks

C 172.16.0.0/30 is directly connected, Serial0/0/1

L 172.16.0.1/32 is directly connected, Serial0/0/1

C 172.16.1.0/24 is directly connected, Loopback0

L 172.16.1.1/32 is directly connected, Loopback0

B 192.168.0.0/24 [20/0] via 172.16.0.2, 00:43:53

B 192.168.1.0/24 [20/0] via 172.16.0.2, 00:43:53

ISP2#

ე. დაუბრუნდით ITA მარშრუტიზატორს და გაუშვით **clear ip bgp *** ბრძანება. დაიცადეთ სანამ მარშრუტიზატორები მიაღწევენ დადგენილ მდგომარეობას, რომელსაც შეიძლება დასჭირდეს რამდენიმე წამი, და შემდეგ ხელახლა შეამოწმეთ ISP2 მარშრუტიზაციის ცხრილი. ISP1-თან მარშრუტი, 10.1.1.0 ქსელი, აღარ უნდა იყოს ISP2-ის მარშრუტიზაციის ცხრილში და ISP2-თან მარშრუტი, 172.16.1.0 ქსელი, არ უნდა იყოს ISP1-ის მარშრუტიზაციის ცხრილში.

ITA# **clear ip bgp ***

ITA#

*Sep 8 16:47:25.179: %BGP-5-ADJCHANGE: neighbor 10.0.0.1 Down User reset

*Sep 8 16:47:25.179: %BGP_SESSION-5-ADJCHANGE: neighbor 10.0.0.1 IPv4 Unicast topology base removed from session User reset

*Sep 8 16:47:25.179: %BGP-5-ADJCHANGE: neighbor 172.16.0.1 Down User reset

*Sep 8 16:47:25.179: %BGP_SESSION-5-ADJCHANGE: neighbor 172.16.0.1 IPv4 Unicast topology base removed from session User reset

*Sep 8 16:47:25.815: %BGP-5-ADJCHANGE: neighbor 10.0.0.1 Up

*Sep 8 16:47:25.819: %BGP-5-ADJCHANGE

ITA#: neighbor 172.16.0.1 Up

ITA#

შენიშვნა: **clear ip bgp *** ბრძანება არის გამანადგურებელი, რადგან ის სრულად ანულებს ყველა BGP მეზობლობებს. ეს დასაშვებია ლაბორატორიულ გარემოში, მაგრამ შეიძლება იყოს პრობლემატური შექმნილ ქსელში. ამის ნაცვლად, თუ მხოლოდ შემომავალი/გამავალი მარშრუტიზაციის პროტოკოლების შესრულებაა

საჭირო, საკმარისია **clear ip bgp * in** ან **clear ip bgp * out** ბრძანებების გაშვება. ეს ბრძანებები ასრულებენ მხოლოდ ახალი BGP მონაცემთა ბაზის სინქრონიზაციას BGP მეზობლობის სრულად განულების გამანადგურებელი ეფექტების გარეშე. ყველა არსებული Cisco IOS ვერსია მხარს უჭერს მარშრუტების განახლების შესაძლებლობას, რაც ცვლის შემომავალ პროგრამულ რეკონფიგურაციის ფუნქციებს, რომელიც წინასწარ იქნა კონფიგურირებული თითოეული მეზობლის საფუძველზე.

ISP2# **show ip route**

<output omitted>

172.16.0.0/16 is variably subnetted, 4 subnets, 3 masks

C 172.16.0.0/30 is directly connected, Serial0/0/1

L 172.16.0.1/32 is directly connected, Serial0/0/1

C 172.16.1.0/24 is directly connected, Loopback0

L 172.16.1.1/32 is directly connected, Loopback0

B 192.168.0.0/24 [20/0] via 172.16.0.2, 00:00:06

B 192.168.1.0/24 [20/0] via 172.16.0.2, 00:00:06

ISP2#

ISP1# **show ip route**

<output omitted>

10.0.0.0/8 is variably subnetted, 4 subnets, 3 masks

C 10.0.0.0/30 is directly connected, Serial0/0/0

L 10.0.0.1/32 is directly connected, Serial0/0/0

C 10.1.1.0/24 is directly connected, Loopback0

L 10.1.1.1/32 is directly connected, Loopback0

B 192.168.0.0/24 [20/0] via 10.0.0.2, 00:00:42

B 192.168.1.0/24 [20/0] via 10.0.0.2, 00:00:42

ISP1#

მექვსე ეტაპი: ძირითადი და სარეზერვო მარშრუტების კონფიგურაცია მოძრავი სტატიკური მარშრუტების საშუალებით.

ორმხრივ კავშირთან ერთად, რომელიც დადგენილია თითოეულ პროვაიდერთან BGP-ით, დააკონფიგურეთ ძირითადი და სარეზერვო მარშრუტები. configure the primary and backup routes. ეს შეიძლება გაკეთდეს მოძრავი სტატიკური მარშრუტებით ან BGP-ით.

ა. გაუშვით **show ip route** ბრძანება ITA მარშრუტიზატორზე.

ITA# **show ip route**

<output omitted>

Gateway of last resort is not set

10.0.0.0/8 is variably subnetted, 3 subnets, 3 masks

C 10.0.0.0/30 is directly connected, Serial0/0/0

L 10.0.0.2/32 is directly connected, Serial0/0/0

B 10.1.1.0/24 [20/0] via 10.0.0.1, 00:03:51

172.16.0.0/16 is variably subnetted, 3 subnets, 3 masks

C 172.16.0.0/30 is directly connected, Serial0/0/1

L 172.16.0.2/32 is directly connected, Serial0/0/1

B 172.16.1.0/24 [20/0] via 172.16.0.1, 00:03:51

192.168.0.0/24 is variably subnetted, 2 subnets, 2 masks

C 192.168.0.0/24 is directly connected, Loopback0

L 192.168.0.1/32 is directly connected, Loopback0

192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks

C 192.168.1.0/24 is directly connected, Loopback1

L 192.168.1.1/32 is directly connected, Loopback1

ITA#

შეგახსენებთ რომ არ არის განსაზღვრული ბოლო მიმართვის გასასვლელი. ეს არის პრობლემა რადგან ITA არის სასაზღვრო მარშრუტიზატორი ორგანიზაციის ქსელისათვის.

ბ. დააკოფიგურეთ სტატიკური მარშრუტები პოლიტიკების ასახვისთვის, ISP1 არის ძირითადი მომწოდებელი და ISP2 მოქმედებს როგორც სარეზერვო, სარეზერვო ISP2 მარშრუტთან (მანძლის მეტრიკა 220) შედარებით დაბალი მანძილის მეტრიკის მითითებით ISP1 მარშრუტისთვის (210.

ITA(config)# **ip route 0.0.0.0 0.0.0.0 10.0.0.1 210**

ITA(config)# **ip route 0.0.0.0 0.0.0.0 172.16.0.1 220**

გ. **show ip route** ბრძანების გამოყენებით დარწმუნდით რომ ნაგულისხმევი მარშრუტი არის განსაზღვრული.

ITA# **show ip route**

<output omitted>

Gateway of last resort is 10.0.0.1 to network 0.0.0.0

S* 0.0.0.0/0 [210/0] via 10.0.0.1

10.0.0.0/8 is variably subnetted, 3 subnets, 3 masks

C 10.0.0.0/30 is directly connected, Serial0/0/0

L 10.0.0.2/32 is directly connected, Serial0/0/0

B 10.1.1.0/24 [20/0] via 10.0.0.1, 00:05:38

172.16.0.0/16 is variably subnetted, 3 subnets, 3 masks

C 172.16.0.0/30 is directly connected, Serial0/0/1

L 172.16.0.2/32 is directly connected, Serial0/0/1

B 172.16.1.0/24 [20/0] via 172.16.0.1, 00:05:38

192.168.0.0/24 is variably subnetted, 2 subnets, 2 masks

C 192.168.0.0/24 is directly connected, Loopback0

L 192.168.0.1/32 is directly connected, Loopback0

192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks

C 192.168.1.0/24 is directly connected, Loopback1

L 192.168.1.1/32 is directly connected, Loopback1

ITA#

დ. გატესტეთ მოცემული ნაგულისხმევი მარშრუტი არაგამოქვეყნებადი loopback-ის შექმნით ISP1 მარშრუტიზატორზე.

ISP1# **config t**

ISP1(config)# **interface loopback 100**

ISP1(config-if)# **ip address 192.168.100.1 255.255.255.0**

ე. გაუშვით **show ip route** ბრძანება რათა დარწმუნდეთ რომ ახლად დამატებული 192.168.100.0 /24 ქსელი არ ჩანს მარშრუტიზაციის ცხრილში.

ITA# **show ip route**

<output omitted>

Gateway of last resort is 10.0.0.1 to network 0.0.0.0

S* 0.0.0.0/0 [210/0] via 10.0.0.1

10.0.0.0/8 is variably subnetted, 3 subnets, 3 masks

C 10.0.0.0/30 is directly connected, Serial0/0/0

L 10.0.0.2/32 is directly connected, Serial0/0/0

B 10.1.1.0/24 [20/0] via 10.0.0.1, 00:07:08

172.16.0.0/16 is variably subnetted, 3 subnets, 3 masks

C 172.16.0.0/30 is directly connected, Serial0/0/1

L 172.16.0.2/32 is directly connected, Serial0/0/1

B 172.16.1.0/24 [20/0] via 172.16.0.1, 00:07:08

192.168.0.0/24 is variably subnetted, 2 subnets, 2 masks

C 192.168.0.0/24 is directly connected, Loopback0

L 192.168.0.1/32 is directly connected, Loopback0

192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks

C 192.168.1.0/24 is directly connected, Loopback1

L 192.168.1.1/32 is directly connected, Loopback1

ITA#

3. გაფართოებულ პინგის რეჟიმში, დაპინგეთ ISP1 loopback 1 ინტერფეისი 192.168.100.1 ITA loopback 1 ინტერფეისიდან 192.168.1.1.

ITA# **ping**

Protocol [ip]:

Target IP address: **192.168.100.1**

Repeat count [5]:

Datagram size [100]:

Timeout in seconds [2]:

Extended commands [n]: **y**

Source address or interface: **192.168.1.1**

Type of service [0]:

Set DF bit in IP header? [no]:

Validate reply data? [no]:

Data pattern [0xABCD]:

Loose, Strict, Record, Timestamp, Verbose[none]:

Sweep range of sizes [n]:

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 192.168.100.1, timeout is 2 seconds:

Packet sent with a source address of 192.168.1.1

!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 12/14/16 ms

ITA#

შენიშვნა: თქვენ შეგიძლიათ გვერდი აუაროთ გაფართოებულ პინგის რეჟიმს და დაპინგოთ წყარო მისამართის მითითებით, ქვემოთ მოცემული ერთ-ერთი შემოკლებული ბრძანების გამოყენებით:

ITA# **ping 192.168.100.1 source 192.168.1.1**

ან

ITA# **ping 192.168.100.1 source Lo1**

შენიშვნა: ნაგულისხმევი მარშრუტის ტესტირება ISP1-ზე გამოუქვეყნებელი ქსელის შექმნით და მისი დაპინგვა მუშაობს მხოლოდ იმიტომ, რომ ნაგულისხმევი მარშრუტი მიუთითებს ISP1-ის მიმართულებით. თუ სასურველი ნაგულისხმევი მარშრუტი მიუთითებს ISP2-სკენ, პინგი ამ გამოუქვეყნებელ ქსელთან ISP1-თან არ იქნება წარმატებული. თუ ISP1-სთან კავშირი ჩავარდა, მაშინ ISP2-ის ნაგულისხმევი მარშრუტი გააქტიურდება, მაგრამ პინგები იქნება წარმატებული მხოლოდ იმ შემთხვევაში თუ ISP1-ს და ISP2-ს აქვთ სხვა სამუშაო ურთიერთკავშირი და შესაბამისი BGP წვდომა მათ შორის, რაც მოცემული მომენტისათვის არ ხდება.

მეშვიდე ეტაპი: BGP-ის გამოყენება ნაგულისხმევი მარშრუტების გასავრცელებლად

ა. ინტერნეტ-პროვაიდერის მარშრუტიზატორი იქნება გამოყენებული ნაგულისხმევი მარშრუტის მისაცემად BGP-ით. პირველ რიგში გააუქმეთ მიმდინარე ნაგულისხმევი მარშრუტები ITA -ზე.

ITA(config)# **no ip route 0.0.0.0 0.0.0.0 10.0.0.1 210**

ITA(config)# **no ip route 0.0.0.0 0.0.0.0 172.16.0.1 220**

ბ. შემდეგ დააკონფიგურეთ ISP1 მარშრუტიზატორი, რათა გაუგზავნოს ნაგულისხმევი მარშრუტები თავის მეზობელს, ITA მარშრუტიზატორს. მოცემული ბრძანება არ მოითხოვს 0.0.0.0 -ის არსებობას ლოკალურ ISP1 მარშრუტიზატორში.

ISP1(config)# **router bgp 200**

ISP1(config-router)# **neighbor 10.0.0.2 default-originate**

ISP1(config-router)#

გ. დარწმუნდით რომ ნაგულისხმევი მარშრუტი მიღებულ იქნა ITA-სგან BGP-ის გამოყენებით.

ITA# **show ip route**

<output omitted>

Gateway of last resort is 10.0.0.1 to network 0.0.0.0

B* 0.0.0.0/0 [20/0] via 10.0.0.1, 00:01:43

10.0.0.0/8 is variably subnetted, 3 subnets, 3 masks

C 10.0.0.0/30 is directly connected, Serial0/0/0

L 10.0.0.2/32 is directly connected, Serial0/0/0

B 10.1.1.0/24 [20/0] via 10.0.0.1, 00:06:51

172.16.0.0/16 is variably subnetted, 3 subnets, 3 masks

C 172.16.0.0/30 is directly connected, Serial0/0/1

L 172.16.0.2/32 is directly connected, Serial0/0/1

B 172.16.1.0/24 [20/0] via 172.16.0.1, 00:06:51

192.168.0.0/24 is variably subnetted, 2 subnets, 2 masks

C 192.168.0.0/24 is directly connected, Loopback0

L 192.168.0.1/32 is directly connected, Loopback0

192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks

C 192.168.1.0/24 is directly connected, Loopback1

L 192.168.1.1/32 is directly connected, Loopback1

ITA#

მოწყობილობის კონფიგურაციები

ბაზისური კონფიგურაციები

მარშრუტიზატორი ISP1 (R1)

```
hostname ISP1
!
interface Lo0
 description ISP1 Internet Network
 ip address 10.1.1.1 255.255.255.0
!
interface Serial0/0/0
 description ISP1 -> ITA
 ip address 10.0.0.1 255.255.255.252
 clock rate 128000
 no shutdown
!
end
```

მარშრუტიზატორი ITA (R2)

```
hostname ITA
!
interface Lo0
 description Core router network link 1
 ip address 192.168.0.1 255.255.255.0
!
interface Lo1
 description Core router network link 2
```

```
 ip address 192.168.1.1 255.255.255.0
!
interface Serial0/0/0
 description ITA -> ISP1
 ip address 10.0.0.2 255.255.255.252
 no shutdown
interface Serial0/0/1
 description ITA -> ISP2
 ip address 172.16.0.2 255.255.255.252
 clock rate 128000
 no shutdown
!
end
```

მარშრუტიზატორი ISP2 (R3)

```
hostname ISP2
!
interface Lo0
 description ISP2 Internet Network
 ip address 172.16.1.1 255.255.255.0
!
interface Serial0/0/1
 description ISP2 -> ITA
 ip address 172.16.0.1 255.255.255.252
 no shutdown
!
end
```

მოწყობილობის კონფიგურაციები

მარშრუტიზატორი ISP1 (R1)

```
hostname ISP1
!
interface Loopback0
 description ISP1 Internet network
 ip address  10.1.1.1 255.255.255.0
!
interface Loopback100
 ip address 192.168.100.1 255.255.255.0
!
interface Serial0/0/0
 description ISP1 -> ITA
 ip address 10.0.0.1 255.255.255.252
 clock rate 128000
 no shutdown
!
router bgp 200
 bgp log-neighbor-changes
 network 10.1.1.0 mask 255.255.255.0
 neighbor 10.0.0.2 remote-as 100
 neighbor 10.0.0.2 default-originate
!
end
```

მარშრუტიზატორი ITA (R2)

```
hostname ITA
!
interface Loopback0
 description Core router network link 1
 ip address 192.168.0.1 255.255.255.0
!
interface Loopback1
 description Core router network link 2
 ip address 192.168.1.1 255.255.255.0
!
interface Serial0/0/0
 description ITA -> ISP1
 ip address 10.0.0.2 255.255.255.252
 no shutdown
!
interface Serial0/0/1
 description ITA -> ISP2
 ip address 172.16.0.2 255.255.255.252
 clock rate 128000
 no shutdown
!
router bgp 100
 bgp log-neighbor-changes
 network 192.168.0.0
 network 192.168.1.0
 neighbor 10.0.0.1 remote-as 200
 neighbor 10.0.0.1 distribute-list 1 out
```

```
 neighbor 172.16.0.1 remote-as 300
 neighbor 172.16.0.1 distribute-list 1 out
!
access-list 1 permit 192.168.0.0 0.0.1.255
!
end
```

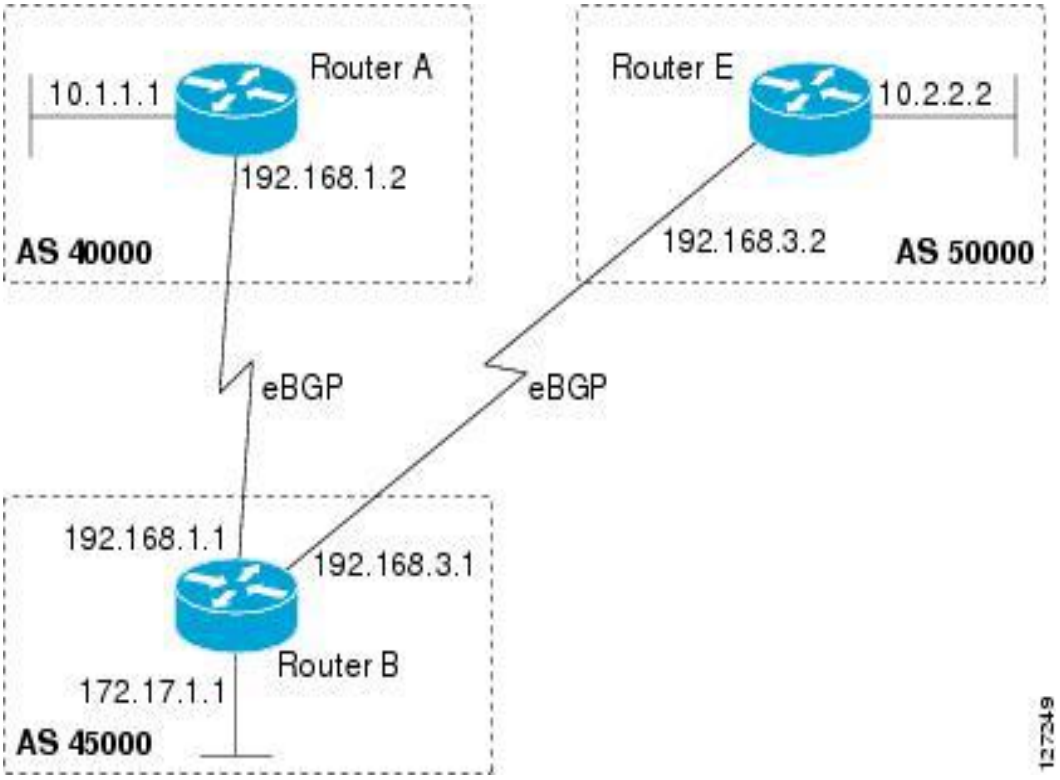
მარშრუტიზატორი ISP2 (R3)

```
hostname ISP2
!
interface Loopback0
 description ISP2 Internet Network
 ip address 172.16.1.1 255.255.255.0
!
interface Serial0/0/1
 description ISP2 -> ITA
 ip address 172.16.0.1 255.255.255.252
 no shutdown
!
router bgp 300
 bgp log-neighbor-changes
 network 172.16.1.0 mask 255.255.255.0
 neighbor 172.16.0.2 remote-as 100
!
end
```

5.2. VRF-ების კონფიგურაცია

5.2.1. *BGP კვანძის კონფიგურაცია IPv4 VRF მისამართების ოჯახისათვის*

შეასრულეთ მოცემული არჩევითი დავალება ორ მოწყობილობას (კვანძს) შორის BGP-ის კონფიგურაციისათვის, რომელმაც უნდა გაცვალოს IPv4 VRF ინფორმაცია, რადგან ისინი არიან VPN-ში. აქ დაკონფიგურებული მისამართის ოჯახი არის IPv4 VRF მისამართის ოჯახი, და კონფიგურაცია გაკეთებულია Router B მარშრუტიზატორზე, ქვემოთ მოცემულ სურათზე, მეზობელ 192.168.3.2 Router E მარშრუტიზატორთან ერთად 50000 ავტონომიურ სისტემაში. არ დაგავიწყდეთ შეასრულოთ მოცემული დავალება ნებისმიერი მეზობელი მოწყობილობისათვის, რომელიც უნდა იყოს BGP IPv4 VRF მისამართის ოჯახის კვანძი.



დავალების დაწყებამდე

ამ დავალების დაწყებამდე, შეასრულეთ, BGP მარშრუტიზაციის პროცესის კონფიგურაცია.

შემაჯამებელი ეტაპები

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **vrf forwarding** *vrf-name*
5. **ip address** *ip-address mask* [**secondary** [**vrf** *vrf-name*]]
6. **exit**
7. **ip vrf** *vrf-name*
8. **rd** *route-distinguisher*
9. **route-target** {**import** | **export** | **both**} *route-target-ext-community*
10. **exit**
11. **router bgp** *autonomouse-system-number*
12. **address-family ipv4** [**unicast** | **multicast** | **vrf** vfr-name]
13. **neighbor** *ip-address* **remote-as** *autonomouse-system-number*
14. **neighbor** {*ip-address* | *peer-group-name*} **maximum-prefix** *maximum* [**warning-only**]
15. **neighbor** *ip-address* **activate**
16. **end**

დეტალურად აღწერილი ნაბიჯები

ეტაპი	ბრძანება ან მოქმედება	მიზანი
ეტაპი №1	**enable** **მაგალითი:** Device> enable	რთავს პრივილეგირებულ EXEC რეჟიმს. • მოთხოვნის შემთხვევაში შეიყვანეთ თქვენი პაროლი
ეტაპი №2	**configure terminal** **მაგალითი:** Device# configure terminal	შეყავს გლობალური კონფიგურაციის რეჟიმში
ეტაპი №3	**interface** *type number* **მაგალითი:**	შედის ინტერფეისის კონფიგურაციის რეჟიმში
ეტაპი №4	**vtf forwarding** *vrf-name* **მაგალითი:** Device (config-if)# vrf forwarding vpn1	ასოციაციით აკავშირებს VPN VRF მოთხოვნას ინტერფეისთან ან ქვეინტერფეისთან
ეტაპი №5	**ip address** *ip address mask* [**secondary** [**vrf** *vrf-name*]] **მაგალითი:** Device (config-if)# ip address 192.168.3.1 255.255.255.0	ახდენს IP მისამართის მომართვას ინტერფეისზე
ეტაპი №6	**exit** **მაგალითი:** Device (config-if)# exit	ბრძანებას გამოყავს ინტერფეისის კონფიგურაციის რეჟიმიდან და შეყავს გლობალური კონფიგურაციის რეჟიმში
ეტაპი №7	**ip vrf** *vrf-name* **მაგალითი:** Device (config) # ip vrf vpn1	აკონფიგურებს VRF მარშრუტიზაციის ცხრილს და შედის VRF კონფიგურაციის რეჟიმში

		• სახელის მისათითებლად გამოიყენეთ *vrf-name* არგუმენტი, VRF-სთვის მისანიჭებლად
ეტაპი №8	**rd** *route-distinguisher* **მაგალითი:** Device (config-vrf) # rd 45000 : 5	ქმნის მარშრუტიზაციისა და გადამისამართების ცხრილებს და უთითებს ნაგულისხმევ მარშრუტის მდგენელს VPN-სთვის. • გამოიყენეთ *route-distinguisher* არგუმენტი IPv4 პრეფიქსზე 8 ბაიტიანი მნიშვნელობის დასამატებლად, რათა შეიქმნას უნიკალური VPN IPv4 პრეფიქსი
ეტაპი №9	**route-target** {**import** | **export** | **both**} *route-target-ext-community* **მაგალითი:** Device (config-vrf) # route-target both 45000 : 100	ქმნის სამიზნე მარშრუტს, რომელიც ვრცელდება VRF-ის გაერთიანებაში. • გამოიყენეთ **import** საკვანძო სიტყვა სამიზნე VPN-ის გაფართოებული გაერთიანებიდან მარშრუტიზაციის ინფორმაციის იმპორტისათვის • გამოიყენეთ **export** საკვანძო სიტყვა სამიზნე VPN-ის გაფართოებული გაერთიანებაში მარშრუტიზაციის ინფორმაციის ექსპორტისათვის. • გამოიყენეთ **both** სიტყვაკოდი ორივე იმპორტისა და ექსპორტის მარშრუტიზაციის ინფორმაციის იმპორტისათვის სამიზნე VPN-ის გაფართოებული გაერთიანებაში

		• გამოიყენეთ *route-target-ext-community* არგუმენტი იმპორტის, ექსპორტის ან ორივეს (იმპორტი და ექსპორტი) VRF სიის სამიზნე მარშრუტის გაფართოებული გაერთიანებების სამიზნე გაფართოებული გაერთიანებული ატრიბუტების დასამატებლად.
ეტაპი №10	**exit** **მაგალითი:** Device (config-vrf)# exit	გამოყავს VRF კონფიგურაციის რეჟიმიდან და შეყავს გლობალური კონფიგურაციის რეჟიმში
ეტაპი №11	**route bgp** *autonomous-system-number* **მაგალითი:** Device (config) # router bgp 45000	შეყავს მარშრუტიზატორის კონფიგურაციის რეჟიმში კონკრეტული მარშრუტიზაციის პროცესისათვის.
ეტაპი №12	**address-family ipv4** [**unicast** |**multicast** | **vrf** *vrf-name*] **მაგალითი:** Device (config-router) # address-family ipv4 vrf vpn1	უთითებს IPv4 მისამართის ოჯახს და შეყავს მისამართის ოჯახის კონფიგურაციის რეჟიმში. • გამოიყენეთ **unicast** სიტყვაკოდი ერთმისამართიანი IPv4 მისამართის ოჯახის მისათითებლად. ნაგულისხმევად მოწყობილობა განთავსებულია კონფიგურაციის რეჟიმში IPv4 ერთმისამართიანი მისამართის ოჯახისათვის თუ **unicast** სიტყვაკოდი არ არის მითითებული **address-family ipv4** ბრძანებასთან ერთად.

		• გამოიყენეთ **multicast** სიტყვაკოდი IPv4 მრავალმისამართიანი მისამართის პრეფიქსების მისათითებლად. • გამოიყენეთ **vrf** სიტყვაკოდი და *vrf-name* არგუმენტი vrf მოთხოვნის სახელის მისათითებლად, რათა მოხდეს ასოციაციით დაკავშირება შემდგომ IPv4 მისამართის ოჯახის კონფიგურაციის რეჟიმის ბრძანებებთან.
ეტაპი №13	**neighbor** *ip-address* **remote-as** *autonomous-system-number* **მაგალითი:** Device (config-router-af) # neighbor 192.168.3.2 remote-as 50000	ამატებს მეზობლის IP მისამართს განსაზღვრულ ავტონომიურ სისტემაში, ლოკალური მოწყობილობის IPv4 მულტიპროტოკოლურ BGP მეზობლის ცხრილში.
ეტაპი №14	**neighbor** {*ip-address* | *peer-group-name*} **maximum-prefix** *maximum* [*threshold*] [**restart** *restart-interval*] [**warning-only**] **მაგალითი:** Device (config-router-af) # neighbor 192.168.3.2 maximum-prefix 10000 warning-only	აკონტროლებს რამდენი პრეფიქსის მიღება შეუძლია მეზობლიდან. • გამოიყენეთ *maximum* არგუმენტი კონკრეტული მეზობლიდან დაშვებული მაქსიმუმი რაოდენობის პრეფიქსის მისათითებლად. პრეფიქსების რიცხვი, რომელიც შეიძლება იქნეს კონფიგურირებული არის შეზღუდული მხოლოდ მოწყობილობის ხელმისაწვდომი სისტემური რესურსებით.

		• გამოიყენეთ *threshold* არგუმენტი მაქსიმუმი პრეფიქსის ლიმიტის მთელი რიცხვის პროცენტულად წარმოდგენისათვის, რომლის შემდეგაც მოწყობილობა დაიწყებს გამაფრთხილებელი შეტყობინების შექმნას. • გამოიყენეთ **warining-only** სიტყვაკოდი რათა საშუალება მიეცეს მოწყობილობას შექმნას ლოგ შეტყობინება, როდესაც გადააჭარბებს მაქსიმუმ პრეფიქსის ლიმიტს, საკვანძო სესიის დასრულების ნაცვლად.
ეტაპი №15	**neighbor** *ip-address* **activate** **მაგალითი:** Device (config-router-af) # neighbor 192.168.3.2 activate	საშუალებას აძლევს მეზობელს გაცვალოს პრეფიქსები IPv4 VRF მისამართის ოჯახისთვის ლოკალურ მოწყობილობასთან ერთად.
ეტაპი №16	**end** **მაგალითი:** Device (config-router-af) # end	გამოყავს მისამართის ოჯახის კონფიგურაციის რეჟიმიდან და შეყავს პრივილეგირებულ EXEC რეჟიმში.

პრობლემის აღმოფხვრა

გამოიყენეთ **ping vrf** ბრძანება რათა დარწმუნდეთ BGP მოწყობილობებს შორის ბაზისურ ქსელის კავშირში და გამოიყენეთ **show ip vrf** ბრძანება რათა დავრწმუნდეთ, რომ შეიქმნა VRF მოთხოვნა.

5.3. Route-map-ების გამოყენება ფილტრაციით.

ტრაფიკის ფილტრაცია BGP Route-Map-ში Continue Clauses გამოყენებით

შემაჯამებელი ეტაპები

1. enable

2. configure terminal

3. router bgp *autonomouse-system-number*

4. neighbor {*ip-address* | *peer-group-name*} **remote-as** *autonomous-system-number*

5. **neighbor** {*ip-address* | *peer-group-name*} **route-map** *map-name* {**in** | **out**}

6. **exit**

7. **route-map** *map-name* {**permit** | **deny**} [*sequence-number*]

8. **match ip address** {*access-list-number* | *access-list-name*} [*...access-list-number* | *... access-list-name*]

9. **set community** *community-number* [**additive**] [*well-known-community*] | **none**}

10. **continue** [*sequence-number*]

11. **end**

12. **show route-map** [*map-name*]

დეტალურად აღწერილი ნაბიჯები

ეტაპი	ბრძანება ან მოქმედება	მიზანი
ეტაპი №1	**enable** **მაგალითი:** Device> enable	რთავს პრივილეგირებულ EXEC რეჟიმს. • მოთხოვნის შემთხვევაში შეიყვანეთ თქვენი პაროლი
ეტაპი №2	**configure terminal** **მაგალითი:** Device# configure terminal	შეყავს გლობალური კონფიგურაციის რეჟიმში
ეტაპი №3	**router bgp** *autonomous-system-number* **მაგალითი:** Device (config) # router bgp 50000	შეყავს მარშრუტიზატორის კონფიგურაციის და ქმნის BGP მარშრუტიზაციის პროცესს
ეტაპი №4	**neighbor** {*ip-address* | *peer-group-name*} **remote-as** *autonomous-system-number* **მაგალითი:** Device (config-router) # neighbor 10.0.0.1 remote-as 50000	ამატებს მეზობლის IP მისამართს ან კვანძის ჯგუფის სახელს განსაზღვრულ ავტონომიურ სისტემაში, ლოკალური მოწყობილობის IPv4 მულტიპროტოკოლურ BGP მეზობლის ცხრილში.
ეტაპი №5	**neighbor** {*ip-address* | *peer-group-name*} **route-map** *map-name* {**in** | **out**} **მაგალითი:** Device (config-router) # neighbor 10.0.0.1 route-map ROUTE-MAP-NAME in	იყენებს შემომავალ მარშრუტის რუკას, იმ მარშრუტებისათვის, რომლებიც მიღებულია განსაზღვრული მეზობლიდან ან იყენებს გამავალი მარშრუტების მარშრუტის რუკას, რომელიც გამოქვეყნებულია განსაზღვრულ მეზობელზე.

ეტაპი №6	**exit** **მაგალითი:** Device (config-if)# exit	ბრძანებას გამოყავს ინტერფეისის კონფიგურაციის რეჟიმიდან და შეყავს გლობალური კონფიგურაციის რეჟიმში
ეტაპი №7	**route-map** *map-name* {**permit** | **deny**} [*sequence-number*] **მაგალითი:** Device (config) # route-map ROUTE-MAP-NAME permit 10	შეყავს router-map კონფიგურაციის რეჟიმში, route-map-ის შესაქმნელად ან დასაკონფიგურებლად.
ეტაპი №8	**match ip address** {*access-list-number* | *access-list-name*} [*...access-list-number* | *... access-list-name*] **მაგალითი:** Device (config-route-map) # match ip address 1	აკონფიგურებს **match** ბრძანებას, რომელიც მიუთითებს პირობებს, რომლითაც ხდება მარშრუტიზაცის პოლიტიკები და მარშრუტების ფილტრაციები. • შესაძლებელია მრავალი **match** ბრძანებების დაკონფიგურება. თუ **match** ბრძანება არის კონფიგურირებული, დამთხვევა უნდა მოხდეს იმისათვის, რომ მდგომარეობამ განაგრძოს შესრულება. თუ **match** ბრძანება არ არის დაკონფიგურებული, ნაკრები (set) და გაგრძელებული დებულებები (continue clauses) იქნება გამოყენებული. **შენიშვნა:** ამ დავალებაში გამოყენებული **match** და **set** ბრძანებები არის მაგალითები, რომელიც გვეხმარება **continue** ბრძანების ფუნქციის აღწერაში.

		კონკრეტული **match** და **set** ბრძანებების სიისათვის, იხილეთ **continue** ბრძანება *Cisco IOS IP Routing : BGP Command Reference*-ში.
ეტაპი №9	**set community community-number** [**additive**] [*well-known-community*] | **none**} **მაგალითი:** Device (Config-route-map) # set community 10:1	აკონფიგურებს **set** ბრძანებას, რომელიც განსაზღვრავს მარშრუტიზაციის მოქმედებას შესრულებისთვის თუ შეხვდა **match** ბრძანებებით იძულებით უზრუნველყოფილი კრიტერიუმები. • შესაძლებელია მრავალი **set** ბრძანების კონფიგურაცია. • მოცემულ მაგალითში, დებულება (clause) არის შექმნილი კონკრეტული ერთობის მოსამართად.
ეტაპი №10	**continue** [*sequence-number*] **მაგალითი:** Device (config-route-map) # continue	აკონფიგურებს მარშრუტის რუკას რათა გააგრძელოს match მდგომარეობების შეფასება და გამოყენება, წარმატებული დამთხვევის მოხდენის შემდეგ. • თუ რიგითი ნომერი არის კონფიგურირებული, continue clause წავა მარშრუტის რუკისკენ მითითებული რიგითობის ნომრით. • თუ რიგითი ნომერი არ არის განსაზღვრული, continue clause წავა მარშრუტის რუკისკენ მომდევნო რიგითი ნომრით. ასეთ

		ქცევას უწოდებენ „ნაგულისხმევ გაგრძელებას - implied continue”. **შენიშვნა:** გამავალი მარშრუტის რუკების დებულებების გაგრძელება (continue clauses) არის მხარდაჭერილი Cisco IOS XE Release 2.1 და შემდეგ გამოშვებებში.
ეტაპი №11	**end** **მაგალითი:** Device (config-router-af) # end	გამოყავს მისამართის ოჯახის კონფიგურაციის რეჟიმიდან და შეყავს პრივილეგირებულ EXEC რეჟიმში.
ეტაპი №12	**show route-map** [*map-name*] **მაგალითი:** Device# show route-map	(არჩევითი) აჩვენებს ლოკალურად დაკონფიგურებულ მარშრუტის რუკებს. შედეგის გასაფილტრად ამ ბრძანების სინტაქსში შესაძლოა მითითებული იყოს მარშრუტის რუკის სახელი.

მაგალითები

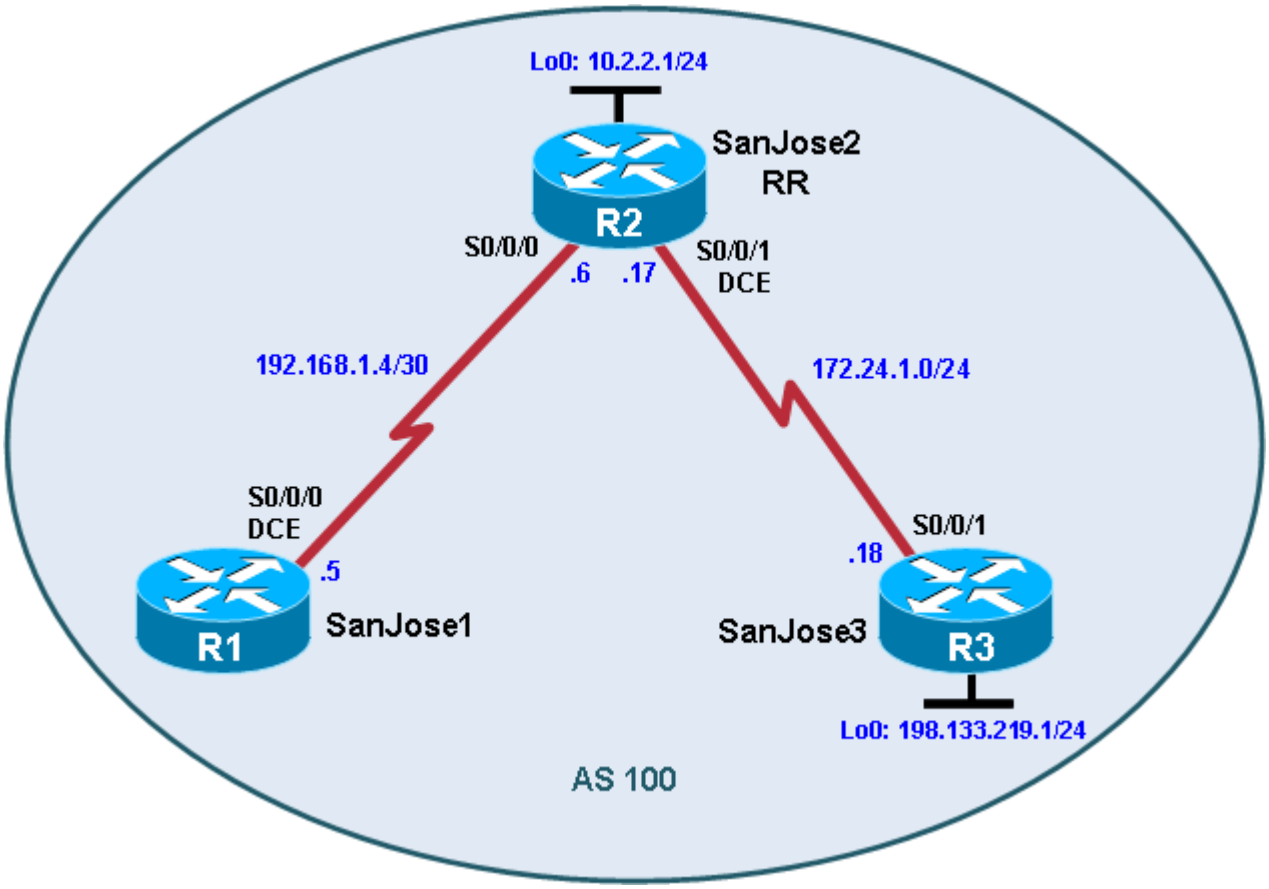
ქვემოთ მოცემული მარტივი შედეგი გვიჩვენებს თუ როგორ უნდა შევამოწმოთ დებულებების გაგრძელება (continue clauses) **show route-map** ბრძანები გამოყენებით. შედეგი გვიჩვენებს დაკონფიგურებულ მარშრუტების რუკებს დამთხვევების, მომართვის და დებულებების გაგრძელების ჩათვლით.

```
Device# show route-map

route-map MARKETING, permit, sequence 10
  Match clauses:
    ip address (access-lists): 1
    metric 10
  Continue: sequence 40
  Set clauses:
    as-path prepend 10
  Policy routing matches: 0 packets, 0 bytes
route-map MARKETING, permit, sequence 20
  Match clauses:
    ip address (access-lists): 2
    metric 20
  Set clauses:
    as-path prepend 10 10
  Policy routing matches: 0 packets, 0 bytes
route-map MARKETING, permit, sequence 30
  Match clauses:
  Continue: to next entry 40
  Set clauses:
    as-path prepend 10 10 10
  Policy routing matches: 0 packets, 0 bytes
route-map MARKETING, permit, sequence 40
  Match clauses:
    community (community-list filter): 10:1
  Set clauses:
    local-preference 104
  Policy routing matches: 0 packets, 0 bytes
route-map MKTG-POLICY-MAP, permit, sequence 10
  Match clauses:
  Set clauses:
    community 655370
  Policy routing matches: 0 packets, 0 bytes
```

5.4. მარშრუტების ფილტრაცია რედისტრიბუციით და შეჯამებით.

BGP მარშრუტების ამრეკლავები (Reflectors) და მარშრუტების ფილტრები

ტოპოლოგია



შესასრულებელი დავალებები:

- IBGP მარშრუტიზატორების კონფიგურაცია მარშრუტის ამრეკლისა და მარტივი მარშრუტის ფილტრის გამოსაყენებლად.

ზოგადი ინფორმაცია

საერთაშორისო ტურისტული სააგენტო მხარს უჭერს სრულ-ბადისებრ IBGP ქსელს, რომელიც სწრაფად არის მასშტაბირებული 100 მარშრუტიზატორს შორის. კომპანიას სურს დანერგოს მარშრუტის რეფლექტორები, სრულ-ბადისებრ IBGP მოთხოვნების გარშემო სამუშაოდ. დააკონფიგურეთ მცირე კლასტერი და დააკვირდით როგორ

მუშაობს BGP ამ კონფიგურაციაში. გამოიყენეთ IP პრეფიქს ფილტრები IBGP კვანძებს შორის განახლებების სამართავად.

შენიშვნა: მოცემული ლაბორატორიული დავალება იყენებს Cisco 1841 მარშრუტიზატორებს Cisco IOS 12.4(24)T1 გამოშვებასთან ერთად და გაფართოებულ IP სერვისების იმიჯს c1841-advipservicesk9-mz.124-24.T1.bin. თქვენ შეგიძლიათ გამოიყენოთ სხვა მარშრუტიზატორები (როგორიცაა 2801 ან 2811) და Cisco IOS პროგრამული უზრუნველყოფის ვერსიები თუ მათ აქვთ მსგავსი შესაძლებლობები და ფუნქციები. მარშრუტიზატორის ან კომუტატორის და Cisco IOS პროგრამული უზრუნველყოფის ვერსიის მიხედვით, ხელმისაწვდომი ბრძანებები და წარმოებული შედეგები შეიძლება იყოს განსხვავებული იმისგან რაც მოცემულია ამ ლაბორატორიულ დავალებაში.

მოთხოვნილი რესურსები:

- 3 მარშრუტიზატორი (Cisco 1841 Cisco IOS Release 12.4(24)T1 გაფართოებული IP სერვისებით ან მსგავსი)
- სერიალური და კონსოლის კაბელები

პირველი ეტაპი: მარშრუტიზატორების მომზადება ლაბორატორიული დავალებისათვის.

დააკავშირეთ კაბელებით მოწყობილობები ისე როგორც ნაჩვენებია ტოპოლოგიის დიაგრამაზე. გაასუფთავეთ თითოეული მარშრუტიზატორის საწყისი კონფიგურაცია და ხელახლა ჩატვირთეთ ისინი, წინა კონფიგურაციების წასაშლელად. ჯერჯერობით არ დააკონფიგუროთ Loopback 0 ინტერფეისი SanJose3 მარშრუტიზატორზე.

მეორე ეტაპი: სახელისა და ინტერფეისის მისამართების კონფიგურაცია.

თქვენ შეგიძლიათ ქვემოთ მოცემული კონფიგურაციის ასლის აღება და თქვენს მაშრუტიზატორებში ჩასმა.

მარშრუტიზატორი R1 (სახელი SanJose1)

```
hostname SanJose1
```

```
!
interface Serial0/0/0
 ip address 192.168.1.5 255.255.255.252
 clock rate 128000
 no shutdown
```

მარშრუტიზატორი R2 (სახელი SanJose2)

```
hostname SanJose2
!
interface Loopback0
 ip address 10.2.2.1 255.255.255.0
!
interface Serial0/0/0
 ip address 192.168.1.6 255.255.255.252
 no shutdown
!
interface Serial0/0/1
 ip address 172.24.1.17 255.255.255.0
 clock rate 128000
 no shutdown
```

მარშრუტიზატორი R3 (სახელი SanJose3)

```
hostname SanJose3
!
interface Serial0/0/1
 ip address 172.24.1.18 255.255.255.0
 no shutdown
```

შენიშვნა: ჯერჯერობით არ დააკონფიგუროთ R3 (SanJose3) მარშრუტიზატორი loopback 0-ით. ამის გაკეთება მოგიწევთ შემდგომში.

მესამე ეტაპი: RIPv2-ის კონფიგურაცია.

ა. ააწყვეთ და დააკონფიგურეთ ქსელი დიაგრამის მიხედვით. გამოიყენეთ RIPv2 როგორც IGP. არ დააკონფიგუროთ 198.133.219.0 ქსელი RIP პროცესის ქვეშ.

SanJose1(config)# **router rip**

SanJose1(config-router)# **version 2**

SanJose1(config-router)# **no auto-summary**

SanJose1(config-router)# **network 192.168.1.0**

SanJose2(config)# **router rip**

SanJose2(config-router)# **version 2**

SanJose2(config-router)# **no auto-summary**

SanJose2(config-router)# **network 172.24.0.0**

SanJose2(config-router)# **network 192.168.1.0**

SanJose2(config-router)# **network 10.0.0.0**

SanJose3(config)# **router rip**

SanJose3(config-router)# **version 2**

SanJose3(config-router)# **no auto-summary**

SanJose3(config-router)# **network 172.24.0.0**

ბ. გაუშვით **show ip route** ბრძანება მარშრუტიზატორებზე იმის შესამოწმებლად, რომ თითოეულ მარშრუტიზატორს აქვს სრული მარშრუტიზაციის ცხრილი.

SanJose1# **show ip route**

Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP

D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area

N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2

E1 - OSPF external type 1, E2 - OSPF external type 2

i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2

ia - IS-IS inter area, * - candidate default, U - per-user static route

o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

172.24.0.0/24 is subnetted, 1 subnets

R 172.24.1.0 [120/1] via 192.168.1.6, 00:00:21, Serial0/0/0

10.0.0.0/24 is subnetted, 1 subnets

R 10.2.2.0 [120/1] via 192.168.1.6, 00:00:21, Serial0/0/0

192.168.1.0/30 is subnetted, 1 subnets

C 192.168.1.4 is directly connected, Serial0/0/0

გ. გაუშვით ქვემოთ მოცემული Tcl სკრიპტი ყველა მარშრუტიზატორზე კავშირის შესამოწმებლად.

SanJose1# **tclsh**

foreach address {

10.2.2.1

192.168.1.5

192.168.1.6

172.24.1.17

172.24.1.18

} {

ping $address }

მეოთხე ეტაპი: IBGP კვანძებისა და მარშრუტის რეფლექტორების კონფიგურაცია.

მოცემულ ლაბორატორიულ დავალებაში თქვენ დააკონფიგურებთ მარშრუტის რეფლექტორს. როგორც წესი, მარშრუტიზატორი რომელიც იღებს EBGP მარშრუტს, აქვეყნებს მას თავის EBGP და IBGP კვანძებში. თუმცა თუ ის იღებს მას IBGP-ის საშუალებით, მაშინ არ ახდენს მის გამოქვეყნებას თავის IBGP კვანძებში, როგორც ციკლის პრევენციის მექანიზმი. ციკლის პრევენციის შესანარჩნებლად მარშრუტის რეფლექტორი ამატებს ორ არჩევით, არატრანზიტულ BGP ატრიბუტს თითოეული

რეფლექტული მარშრუტისთვის, ORIGINATOR_ID და CLUSTER_LIST. ის იყენებს ამ ატრიბუტებს AS_PATH სიის მსგავსად, რათა აკრძალოს მარშრუტიზაციის მოსალოდნელი ჩაციკვლისაგან. დანატებითი ინფორმაციისათვის იხილეთ http://tools.ietf.org/html/rfc4456.

თუმცა ამ ქცევისთვის, ერთადერთი საშუალება ყველა IBGP მარშრუტიზატორისთვის მიიღოს მარშრუტი AS-ში წარმოშობის შემდეგ არის სრული-ბადისებრი IBGP კვანძების ქონა. ეს შეიძლება იყოს რთული დიდი რაოდენობის კვანძებთან. მარშრუტის რეფლექტორი საშუალებას აძლევს ტოპოლოგიას მიიღოს შეზღუდვა IBGP-ის გარშემო სრული ბადის ფლობით. ამის გასაკეთებლად მარშრუტის რეფლექტორი უთითებს მისი მეზობლებიდან ზოგიერთს როგორც მარშრუტის რეფლექტორ კლიენტებს. როცა მარშრუტის რეფლექტორი მიიღებს განახლებას კლიენტი მარშრუტის რეფლექტორიდან, მას შეუძლია გადასცეს ის თავის სხვა კლიენტებს. მარშრუტის რეფლექტორს ასევე შეუძლია რომ გადასცეს კლიენტ-შესწავლილი მარშრუტი თავის არა-კლიენტ კვანძებს (ორივე IBGP და EBGP კვანძები). ანალოგიურად, არა-კლიენტის მიერ შესწავლილი კვანძი (ისევ IBGP ან EBGP კვანძიდან ერთ-ერთი) შესაძლოა გადაცემულ იქნას თავის კლიენტ კვანძებზე. ეს მნიშვნელოვნად ამარტივებს კონფიგურაციას რადგან მხოლოდ მარშრუტის რეფლექტორია საჭირო ყველა სხვა კვანძების ცოდნისთვის. კლიენტებმა ასევე არ იციან რომ ისინი კლიენტებად ითვლებიან. მათთვის ეს არის უბრალოდ ჩვეულებრივი IBGP კვანძების ურთიერთკავშირი. დუბლირებისათვის (redundancy) თქვენ ასევე შეგიძლიათ მომართოთ რამდენიმე მარშრუტის რეფლექტორი უფრო გაფართოებულ კონფიგურაციაში.

ა. დააკონფიგურეთ IBGP კვანძები BGP-სთვის. შემდეგ თქვენ დააკონფიგურებთ SanJose2-ს, როგორც მარშრუტის რეფლექტორს. თუმცა პირველ რიგში დააკონფიგურეთ ის რათა გამოჩნდეს ერთდროულად ორ სხვა მარშრუტიზატორზე.

```
SanJose2(config)# router bgp 100
SanJose2(config-router)# neighbor 192.168.1.5 remote-as 100
SanJose2(config-router)# neighbor 172.24.1.18 remote-as 100
```

SanJose2 კონფიგურაციის შემდეგ, დააკონფიგურეთ ორი სხვა მარშრუტიზატორი, როგორც მარშრუტის რეფლექტორი კლიენტები. გახსოვდეთ, რომ კლიენტების მარტივად მოსამართად, დააკონფიგურეთ კლიენტსა და სერვერს შორის წვდომა. IBGP არ საჭიროებს სრულ ბადეში კონფიგურაციას.

ბ. გაუშვით ქვემოთ მოცემული ბრძანებები SanJose1 მარშრუტიზატორზე:

SanJose1(config)# **router bgp 100**

SanJose1(config-router)# **neighbor 192.168.1.6 remote-as 100**

გ. გაუშვით ქვემოთ მოცემული ბრძანებები SanJose3 მარშრუტიზატორზე:

SanJose3(config)# **router bgp 100**

SanJose3(config-router)# **neighbor 172.24.1.17 remote-as 100**

დ. გამოიყენეთ **show ip bgp neighbors** ბრძანება იმის შესამოწმებლად, რომ SanJose2-მ ჩამოაყალიბა თანაბარი (peering) ურთიერთობა SanJose1-სა და SanJose3-თან. აუცილებლობის შემთხვევაში მოაგვარეთ პრობლემა.

SanJose2# **show ip bgp neighbors**

BGP neighbor is 172.24.1.18, remote AS 100, internal link

BGP version 4, remote router ID 172.24.1.18

BGP state = Established, up for 00:02:10

<output omitted>

BGP neighbor is 192.168.1.5, remote AS 100, internal link

BGP version 4, remote router ID 192.168.1.5

BGP state = Established, up for 00:04:15

SanJose1 და SanJose3 მარშრუტიზატორებმა ვერჩამოაყალიბეს კავშირი. რატომ?

_____

_____

SanJose1 და SanJose3 მარშრუტიზატორები არ იქნა დაკონფიგურებული შესაბამისი BGP **neighbor** ბრძანებით. როგორც მარშრუტის რეფლექტორ კლიენტები, SanJose1-სა და SanJose3-ს არ სჭირდებათ მიაღწიონ დადგენილ მდგომარეობას.

მეხუთე ეტაპი: ქსელის შეყვანა BGP-ში.

ა. მარშრუტის რეფლექტორის გამოყენებით სრული ეფექტის სანახავად, დააკონფიგურეთ SanJose3 გარე მარშრუტიზაციის ინფორმაციის BGP-ში შესაყვანად.

SanJose3(config)# **interface loopback 0**

SanJose3(config-if)# **ip address 198.133.219.1 255.255.255.0**

SanJose3(config-if)# **router bgp 100**

SanJose3(config-router)# **network 198.133.219.0**

მოცემული კონფიგურაცია აიძულებს SanJose3 მარშრუტიზატორს შეიყვანოს გარე 198.133.219.0 მარშრუტი BGP. გამოიყენეთ **show ip route** ბრძანება იმის შესამოწმებლად აიღო თუ არა SanJose2-მა მარშრუტი BGP-ს დახმარებით. SanJose2-ს უნდა ჰქონდეს მარშრუტი 198.133.219.0-თან.

SanJose2# **show ip route**

Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP

D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area

N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2

E1 - OSPF external type 1, E2 - OSPF external type 2

i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2

ia - IS-IS inter area, * - candidate default, U - per-user static route

o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

172.24.0.0/24 is subnetted, 1 subnets

C 172.24.1.0 is directly connected, Serial0/0/1

10.0.0.0/24 is subnetted, 1 subnets

C 10.2.2.0 is directly connected, Loopback0

B 198.133.219.0/24 [200/0] via 172.24.1.18, 00:01:48

C 10.2.2.0 is directly connected, Loopback0

192.168.1.0/30 is subnetted, 1 subnets

C 192.168.1.4 is directly connected, Serial0/0/0

რა არის შემდეგი ნახტომი (Next-hop) ამ მარშრუტისათვის? ახსენით.

_____

_____

ბ. დარწმუნდით რომ შეგიძლიათ 198.133.219.1-ის დაპინგვა SanJose2-დან. თუ არა მოაგვარეთ პრობლემა.

გ. შეამოწმეთ SanJose1-ის მარშრუტიზაციის ცხრილი. აქ არ უნდა იყოს 198.133.219.0-თან მარშრუტი. რატომ?

_____

_____

დ. გახსოვდეთ რომ SanJose1 არ არის კონფიგურირებული SanJose3-თან წვდომისათვის (peer). სრული IBGP ბადის საჭიროების მოსაგვარელად, SanJose2 უნდა იყოს კონფიგურირებული როგორც მარშრუტის რეფლექტორი. გაუშვით ქვემოთ მოცემული ბრძანებები SanJose2 მარშრუტიზატორზე:

SanJose2(config)# **router bgp 100**

SanJose2(config-router)# **neighbor 192.168.1.5 route-reflector-client**

SanJose2(config-router)# **neighbor 172.24.1.18 route-reflector-client**

*Mar 9 19:02:27.831: %BGP-5-ADJCHANGE: neighbor 192.168.1.5 Down RR client conf ig change

*Mar 9 19:02:27.931: %BGP-5-ADJCHANGE: neighbor 172.24.1.18 Down RR client conf ig change

*Mar 9 19:02:32.387: %BGP-5-ADJCHANGE: neighbor 172.24.1.18 Up

*Mar 9 19:02:37.507: %BGP-5-ADJCHANGE: neighbor 192.168.1.5 Up

ე. SanJose2 მარშრუტიზატორზე **show ip protocols** ბრძანების გაშვებით დარწმუნდით რომ IBGP კლასტერი შეიქმნა წარმატებულად. ამ ბრძანების შედეგმა უნდა აჩვენოს, რომ SanJose2 არის მარშრუტის რეფლექტორი.

SanJose2# **show ip protocols**

Routing Protocol is "rip"

Outgoing update filter list for all interfaces is not set

Incoming update filter list for all interfaces is not set

Sending updates every 30 seconds, next due in 26 seconds

Invalid after 180 seconds, hold down 180, flushed after 240

Redistributing: rip

Default version control: send version 2, receive version 2

Interface	Send	Recv	Triggered RIP	Key-chain
Serial0/0/0	2	2		
Serial0/0/1	2	2		
Loopback0	2	2		

Automatic network summarization is not in effect

Maximum path: 4

Routing for Networks:

10.0.0.0

172.24.0.0

192.168.1.0

Routing Information Sources:

Gateway	Distance	Last Update

Distance: (default is 120)

Routing Protocol is "bgp 100"

Outgoing update filter list for all interfaces is not set

Incoming update filter list for all interfaces is not set

Route Reflector for address family IPv4 Unicast, 2 clients

Route Reflector for address family IPv6 Unicast, 2 clients

Route Reflector for address family IPv4 MDT, 2 clients

Route Reflector for address family VPNv4 Unicast, 2 clients

Route Reflector for address family VPNv6 Unicast, 2 clients

Route Reflector for address family IPv4 Multicast, 2 clients

Route Reflector for address family IPv6 Multicast, 2 clients

Route Reflector for address family NSAP Unicast, 2 clients

IGP synchronization is disabled

Automatic route summarization is disabled

Neighbor(s):

Address	FiltIn	FiltOut	DistIn	DistOut	Weight	RouteMap
172.24.1.18						
192.168.1.5						

Maximum path: 1

Routing Information Sources:

Gateway	Distance	Last Update
172.24.1.18	200	00:01:43

Distance: external 20 internal 200 local 200

რამდენი კლიენტი აქვს SanJose2 მარშრუტიზატორს?

---

3. გაუშვით **show ip protocols** ბრძანება SanJose1-ზე. მოცემული ბრძანების შედეგი არ შეიცავს ინფორმაციას მარშრუტის რეფლექტორებზე. შეგახსენებთ, რომ SanJose1 არის კლიენტი და არა მარშრუტის რეფლექტორი სერვერი, ამიტომ ის არ ფლობს ინფორმაციას მარშრუტის არეკვლის (reflection) ინფორმაციას.

ზ. ბოლოს, დარწმუნდით რომ მარშრუტის არეკვლა მუშაობს, SanJose1-ზე მარშრუტიზაციის ცხრილის შემოწმებით. SanJose1-ს უნდა ჰქონდეს მარშრუტი 198.133.219.0 ქსელთან.

SanJose1# **show ip route**

Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP

D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area

N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2

E1 - OSPF external type 1, E2 - OSPF external type 2

i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2

ia - IS-IS inter area, * - candidate default, U - per-user static route

o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

172.24.0.0/24 is subnetted, 1 subnets

R 172.24.1.0 [120/1] via 192.168.1.6, 00:00:08, Serial0/0/0

10.0.0.0/24 is subnetted, 1 subnets

R 10.2.2.0 [120/1] via 192.168.1.6, 00:00:08, Serial0/0/0

B 198.133.219.0/24 [200/0] via 172.24.1.18, 00:01:25

192.168.1.0/30 is subnetted, 1 subnets

C 192.168.1.4 is directly connected, Serial0/0/0

არის თუ არა 172.24.1.18 IP მისამართი SanJose1 ცხრილში ამ მარშრუტის შემდეგი ნახტომის (next hop)? ახსენით.

_____

_____

აღსანიშნავია, რომ SanJose1 არ არის პირდაპირ დაკავშირებული შემდეგი ნახტომის (next hop) IP ქსელთან. რატომ?

მინიშნება: რომელი მარშრუტიზატორიდან შეისწავლა SanJose1-მ მარშრუტი?

თ. SanJose1-დან დაპინგეთ 198.133.219.1 SanJose1. პინგი უნდა იყოს წარმატებული.

მიაქციეთ ყურადღება, რომ SanJose1-ს პინგები R3 198.133.219.1-თან არის წარმატებული მიუხედავად იმისა, რომ შემდეგი ნახტომის (next-hop) მისამართი არ არის პირდაპირ დაკავშირებული ქსელი. მაგალითად შემდეგი ნახტომის (next-hop) მისამართი უნდა იყოს 192.168.1.6 R2 მარშრუტიზატორზე თუ ის არ IBGP ქმედებისთვის.

მეექვსე ეტაპი: შემაჯამებელი მისამართის შეყვანა BGP-ში.

ა. ამ ლაბორატორიული დავალებისათვის დააკონფიგურეთ SanJose3, BGP-ში შემაჯამებელი მისამართის შესაყვანად.

SanJose3(config)# **router bgp 100**

SanJose3(config-router)# **aggregate-address 198.0.0.0 255.0.0.0**

BGP-მ ახლა აუციებლად უნდა გააგზავნოს supernet მარშრუტი 198.0.0.0.0/8 SanJose2-თან ATOMIC_AGGREGATE ატრიბუტების ნაკრებთან ერთად.

შენიშვნა: ნაგულისხმევად BGP Cisco მარშრუტიზატორებზე აქვეყნებს როგორც მარშრუტების გაერთიანებას ისე ცალკეული კომპონენტის მარშრუტებს. თუ მხოლოდ მარშრუტების გაერთიანება არ არის ამოქვეყნებული, გამოიყენეთ **aggregate-address** *network mask* **summary-only** ბრძანება.

ბ. SanJose2 მარშრუტიზატორზე გაუშვით ქვემოთ მოცემული ბრძანება:

SanJose2# **show ip bgp 198.0.0.0**

BGP routing table entry for 198.0.0.0/8, version 8

Paths: (1 available, best #1, table Default-IP-Routing-Table)

Flag: 0x820

Advertised to update-groups:

1

Local, (aggregated by 100 172.24.1.18), (Received from a RR-client)

172.24.1.18 from 172.24.1.18 (172.24.1.18)

Origin IGP, metric 0, localpref 100, valid, internal, atomic-aggregate, best

ამ ბრძანების შედეგის მიხედვით, რომელი მისამართი აერთიანებს მოცემულ მარშრუტს?

რა მიუთითებს იმას, რომ მარშრუტის არეკვლა (reflection) არის ჩართული ამ პროცესში?

არსებობს რაიმე ნიშანი იმისა, რომ ATOMIC_AGGREGATE ატრიბუტი არის მომართული?

გ. SanJose2 თავის მხრივ უნდა ასახავდეს მარშრუტს SanJose1-თან. ამაში დარწმუნებისათვის, შეამოწმეთ როგორც მარშრუტიზაციის ცხრილი ისე BGP ცხრილი SanJose1 მარშრუტიზატორზე. ორივე მარშრუტი 198.133.219.0-თან და supernet მარშრუტი 198.0.0.0 უნდა იყოს შეტანილი SanJose1-ს მარშრუტიზაციის ცხრილში და BGP ცხრილში.

SanJose1# **show ip route**

Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP

D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area

N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2

E1 - OSPF external type 1, E2 - OSPF external type 2

i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2

ia - IS-IS inter area, * - candidate default, U - per-user static route

o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

172.24.0.0/24 is subnetted, 1 subnets

R 172.24.1.0 [120/1] via 192.168.1.6, 00:00:20, Serial0/0/0

10.0.0.0/24 is subnetted, 1 subnets

R 10.2.2.0 [120/1] via 192.168.1.6, 00:00:20, Serial0/0/0

B 198.133.219.0/24 [200/0] via 172.24.1.18, 00:08:34

192.168.1.0/30 is subnetted, 1 subnets

C 192.168.1.4 is directly connected, Serial0/0/0

B 198.0.0.0/8 [200/0] via 172.24.1.18, 00:04:19

საერთაშორისო ტურისტულმა სააგენტომ გადაწყვიტა გაფილტროს კონკრეტული მარშრუტები 198.0.0.0/8 მისამართების სივრცეში. დააკონფიგურეთ მარშრუტის ფილტრი, რათა აუკრძალოთ SanJose2-ს 198.133.219.0/24 მარშრუტის გაგზავნა თავის სხვა კლიენტებთან, ამ შემთხვევაში SanJose1-თან.

დ. გაუშვით ქვემოთ მოცემული ბრძანებები SanJose2მარშრუტიზატორზე:

SanJose2(config)# **ip prefix-list SUPERNETONLY permit 198.0.0.0/8**

SanJose2(config)# **router bgp 100**

SanJose2(config-router)# **neighbor 192.168.1.5 prefix-list SUPERNETONLY out**

ე. დაუბრუნდით SanJose1-ს, გაუშვით **clear ip bgp * soft** ბრძანება და **show ip bgp** ბრძანების გაშვებით შეამოწმეთ, რომ პრეფიქსის სია არის გაშვებული შესრულებაზე. პრობლემის არსებობის შემთხვევაში, მოგვარეთ ისინი.

წინასგან განსხვავებით, სადაც 198.133.219.0 და 198.0.0.0 მარშრუტები იყო წარმოდგენილი, ახლა მხოლოდ ერთი მარშრუტი 198.0.0.0 ჩანს მარშრუტიზაციის და BGP ცხრილებში. აუცილებლობის შემთხვევაში მოაგვარეთ პრობლემები.

SanJose1# **show ip route**

Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP

D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area

N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2

E1 - OSPF external type 1, E2 - OSPF external type 2

i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2

ia - IS-IS inter area, * - candidate default, U - per-user static route

o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

172.24.0.0/24 is subnetted, 1 subnets

R 172.24.1.0 [120/1] via 192.168.1.6, 00:00:20, Serial0/0/0

10.0.0.0/24 is subnetted, 1 subnets

R 10.2.2.0 [120/1] via 192.168.1.6, 00:00:20, Serial0/0/0

192.168.1.0/30 is subnetted, 1 subnets

C 192.168.1.4 is directly connected, Serial0/0/0

B 198.0.0.0/8 [200/0] via 172.24.1.18, 00:04:19

3. გაუშვით ქვემოთ მოცემული Tcl სკრიპტი ყველა მარშრუტიზატორზე, სრული კავშირის შესამოწმებლად. ყველა პინგი უნდა იყოს წარმატებული.

SanJose1# **tclsh**

foreach address {

10.2.2.1

198.133.219.1

192.168.1.5

192.168.1.6

172.24.1.17

172.24.1.18

} {

ping $address }

მარშრუტიზატორის ინტერფეისის შემაჯამებელი ცხრილი:

მარშრუტიზატორის ინტერფეისის შეჯამება				
მარშრუტიზატორის მოდელი	Ethernet interface №1	Ethernet interface №2	Serial interface №1	Serial interface №2
1700	FastEthernet 0 (FA0)	FastEthernet 1 (FA1)	Serial 0 (S0)	Serial 1 (S1)
1800	FastEthernet 0/0 (FA0/0)	FastEthernet 0/1 (FA0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
2600	FastEthernet 0/0 (FA0/0)	FastEthernet 0/1 (FA0/1)	Serial 0/0 (S0/0)	Serial 0/1 (S0/1)
2800	FastEthernet 0/0 (FA0/0)	FastEthernet 0/1 (FA0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)

შენიშვნა: თუ გინდათ მოძებნოთ თუ როგორაა მარშრუტიზატორი კონფიგურირებული, დაათვალიერეთ ინტერფეისები რათა მოახდინოთ მარშრუტიზატორის იდენტიფიკაცია და დაადგინოთ რამდენი ინტერფეისი აქვს მას. აქ არ არის იმის საშუალება, რომ ჩამოიწეროს თითოეული მარშრუტიზატორის კონფიგურაციების კომბინაციები. მოცემული ცხრილი მოიცავს მოწყობილობის **Ethernet** და **Serial** ინტერფეისების შესაძლო კომბინაციების იდენტიფიკატორებს. ეს ცხრილი არ მოიცავს ნებისმიერი სხვა ტიპის ინტერფეისს, რადგან კონკრეტული მარშრუტიზატორი შეიძლება შეიცავდეს მხოლოდ ერთს. ამის მაგალითი შეიძლება იყოს **ISDN BRI** ინტერფეისი. ფრჩხილებში მოცემული ცვლადი არის ლეგალური აბრევიატურა, რომელიც შეიძლება გამოყენებულ იქნას **Cisco IOS** ბრძანებებში, ინტერფეისის წარმოსადგენად.

მოწყობილობის კონფიგურაციები

მარშრუტიზატორი SanJose1 (R1)

```
hostname SanJose1
!
interface Serial0/0/0
 ip address 192.168.1.5 255.255.255.252
 clock rate 128000
 no shutdown
!
router rip
 version 2
 network 192.168.1.0
 no auto-summary
!
router bgp 100
 no synchronization
 neighbor 192.168.1.6 remote-as 100
 no auto-summary
!
end
```

მარშრუტიზატორი SanJose2 (R2)

```
hostname SanJose2
!
interface Loopback0
 ip address 10.2.2.1 255.255.255.0
!
```

```
interface Serial0/0/0
 ip address 192.168.1.6 255.255.255.252
 no shutdown
!
interface Serial0/0/1
 ip address 172.24.1.17 255.255.255.0
 clock rate 128000
 no shutdown
!
router rip
 version 2
 network 172.24.0.0
 network 192.168.1.0
 network 10.0.0.0
 no auto-summary
!
router bgp 100
 no synchronization
 neighbor 172.24.1.18 remote-as 100
 neighbor 172.24.1.18 route-reflector-client
 neighbor 192.168.1.5 remote-as 100
 neighbor 192.168.1.5 route-reflector-client
 neighbor 192.168.1.5 prefix-list SUPERNETONLY out
 no auto-summary
!
ip prefix-list SUPERNETONLY seq 5 permit 198.0.0.0/8
ip prefix-list SUPERNETONLY seq 10 permit 172.24.1.0/24
ip prefix-list SUPERNETONLY seq 15 permit 10.2.2.0/24
```

```
!
end
```

მარშრუტიზატორი SanJose3 (R3)

```
hostname SanJose3
!
interface Loopback0
 ip address 198.133.219.1 255.255.255.0
!
interface Serial0/0/1
 ip address 172.24.1.18 255.255.255.0
 no shutdown
!
router rip
 version 2
 network 172.24.0.0
 no auto-summary
!
router bgp 100
 no synchronization
 network 198.133.219.0
 aggregate-address 198.0.0.0 255.0.0.0
 neighbor 172.24.1.17 remote-as 100
 no auto-summary
end
```

პრაქტიკული სავარჯიშო

11. შეასრულეთ ინტერფეისის მისამართების კონფიგურაცია;
12. შეასრულეთ BGP-ს კონფიგურაცია ISP მარშრუტიზატორებზე;
13. მოახდინეთ BGP-ის კონფიგურაცია ITA სასაზღვრო მარშრუტიზატორზე;
14. შეამოწმეთ BGP მარშრუტიზატორებზე;
15. მოახდინეთ მარშრუტის ფილტრების კონფიგურაცია;
16. შეასრულეთ ძირითადი და სარეზერვო მარშრუტების კონფიგურაცია მოძრავი სტატიკური მარშრუტების საშუალებით;
17. გამოიყენეთ BGP ნაგულისხმევი მარშრუტების გასავრცელებლად;
18. შეასრულეთ BGP კვანძის კონფიგურაცია IPv4 VRF მისამართების ოჯახისათვის;
19. მოახდინეთ ტრაფიკის ფილტრაცია BGP Route-Map-ში Continue Clauses გამოყენებით;
20. შეასრულეთ IBGP კვანძებისა და მარშრუტის რეფლექტორების კონფიგურაცია;
21. მოახდინეთ შემაჯამებელი მისამართის შეყვანა BGP-ში.

ცოდნის შეფასება

სტუდენტებს მიეცემათ პრაქტიკული დავალება

- შეასრულონ ინტერფეისის მისამართების კონფიგურაცია;
- შეასრულონ BGP-ს კონფიგურაცია ISP მარშრუტიზატორებზე;
- მოახდინონ BGP-ის კონფიგურაცია ITA სასაზღვრო მარშრუტიზატორზე;
- შეამოწმონ BGP მარშრუტიზატორებზე;
- დააკონფიგურონ მარშრუტის ფილტრები;
- მოახდინონ ძირითადი და სარეზერვო მარშრუტების კონფიგურაცია მოძრავი სტატიკური მარშრუტების საშუალებით;
- გამოიყენონ BGP ნაგულისხმევი მარშრუტების გასავრცელებლად;
- შეასრულონ BGP კვანძის კონფიგურაცია IPv4 VRF მისამართების ოჯახისათვის;

- მოახდინონ ტრაფიკის ფილტრაცია BGP Route-Map-ში Continue Clauses გამოყენებით;
- დააკონფიგურონ IBGP კვანძები და მარშრუტის რეფლექტორები;
- შეასრულონ შემაჯამებელი მისამართის შეყვანა BGP-ში.

შემფასებელი აკვირდება შესაფასებელ პირის მუშაობას პროფესიული სტანდარტით (პროგრამით / მოდულით) განსაზღვრული ამოცანების შესრულების პროცესში. დაკვირვება ხორციელდება კომპიუტერებით აღჭურვილ ლაბორატორიაში, სადაც შესაფასებელი პირი პრაქტიკულ საქმიანობას ეწევა. შემფასებელმა წინასწარ უნდა დაგეგმოს დაკვირვების პროცესი, იმის დასადგენად, თუ რამდენად სწორად იყენებს შესაფასებელი პირი ცოდნას, უნარებსა და ყველა რესურსს შედეგის მისაღწევად.

შეფასება განხორციელდება პროცესზე დაკვირვებით, წინასწარ განსაზღვრული შეფასების ინდიკატორების საფუძველზე.

დავალების ნიმუში და შეფასების რუბრიკა

პროცესზე დაკვირვება

- შეასრულოს BGP-ს კონფიგურაცია ISP მარშრუტიზატორებზე, მოახდინოს BGP-ის კონფიგურაცია ITA სასაზღვრო მარშრუტიზატორზე, მოახდინოს ძირითადი და სარეზერვო მარშრუტების კონფიგურაცია მოძრავი სტატიკური მარშრუტების საშუალებით.
- გამოიყენოს BGP ნაგულისხმევი მარშრუტების გასავრცელებლად, მოახდინოს ტრაფიკის ფილტრაცია BGP Route-Map-ში Continue Clauses გამოყენებით, დააკონფიგუროს IBGP კვანძები და მარშრუტის რეფლექტორები, შეასრულოს შემაჯამებელი მისამართის შეყვანა BGP-ში.

სწავლის შედეგი	N	დასახელება	შეფასება	
			კი	არა
მესამე დონის მარშრუტიზაციის პროტოკოლი BGP	11.	შეასრულა ინტერფეისის მისამართების კონფიგურაცია		
	12.	შეასრულა BGP-ს კონფიგურაცია ISP მარშრუტიზატორებზე		
	13.	შეასრულა BGP-ის კონფიგურაცია ITA სასაზღვრო მარშრუტიზატორზე;		
	14.	შეასრულა BGP-ს შემოწმება მარშრუტიზატორებზე		
	15.	შეასრულა მარშრუტის ფილტრების კონფიგურაცია		
	16.	შეასრულა ძირითადი და სარეზერვო მარშრუტების კონფიგურაცია მოძრავი სტატიკური მარშრუტების საშუალებით		
	17.	გამოყენება BGP ნაგულისხმევი მარშრუტების გასავრცელებლად		
	18.	შეასრულა BGP კვანძის კონფიგურაცია IPv4 VRF მისამართების ოჯახისათვის		
	19.	ტრაფიკის ფილტრაცია BGP Route-Map-ში Continue Clauses გამოყენებით		
	20.	IBGP კვანძებისა და მარშრუტის რეფლექტორების კონფიგურაცია და შემაჯამებელი მისამართის შეყვანა BGP-ში		

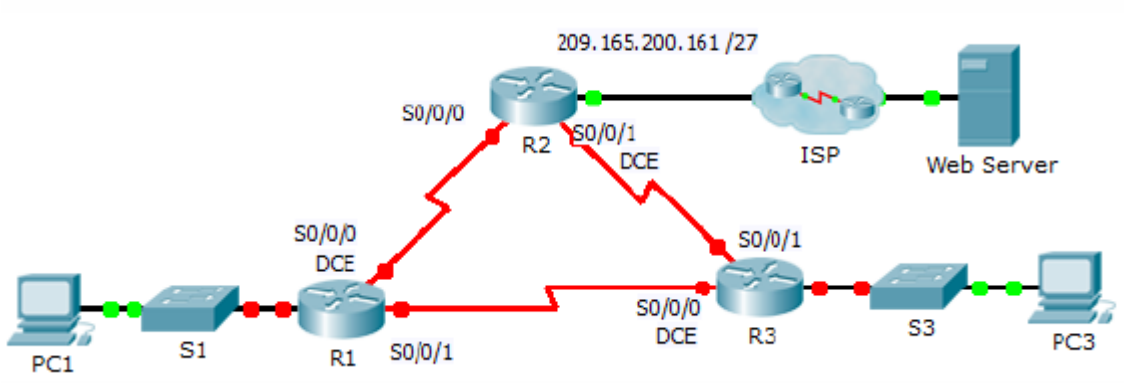
სწავლის შედეგი ჩაითვლება მიღწეულად თუ სტუდენტმა შეძლო შედეგის მინიმუმ 8 პუნქტის შესრულება.

6. ქსელური ინფრასტრუქტურის გამართული მუშაობის უზრუნველყოფა და მონიტორინგი

6.1. მეორე დონის პროტოკოლების დიაგნოსტიკა პრობლემების აღმოფხვრით

PPP-ის პრობლემის აღმოფხვრა აუთენტიფიკაციასთან ერთად

ტოპოლოგია



მისამართების ცხრილი

მოწყობილობა	ინტერფეისი	IP მისამართი	ქვექსელის ნიღაბი	ნაგულისხმევი გასავლელი
R1	G0/1	10.0.0.1	255.255.255.128	N/A
	S0/0/0	172.16.0.1	255.255.255.252	N/A
	S0/0/1	172.16.0.9	255.255.255.252	N/A
R2	G0/1	209.165.200.161	255.255.252.224	N/A
	S0/0/0	172.16.0.2	255.255.255.252	N/A
	S0/0/1	172.16.0.5	255.255.255.252	N/A
R3	G0/1	10.0.0.129	255.255.252.128	N/A
	S0/0/0	172.16.0.10	255.255.255.252	N/A
	S0/0/1	172.16.0.6	255.255.255.252	N/A

ISP	G0/1	209.165.200.162	255.255.255.224	N/A
PC1	ქსელის ადაპტერი	10.0.0.10	255.255.255.128	10.0.0.1
PC3	ქსელის ადაპტერი	10.0.0.139	255.255.255.128	10.0.0.129
Web Server	ქსელის ადაპტერი	209.165.200.2	255.255.255.252	209.165.200.1

შესასრულებელი სამუშაო

ნაწილი №1: ფიზიკური დონის დიაგნოსტიკისა და შეკეთების პროცედურის ჩატარება

ნაწილი №2: არხის დონის დიაგნოსტიკისა და შეკეთების პროცედურის ჩატარება

ნაწილი №3: ქსელის დონის დიაგნოსტიკისა და შეკეთების პროცედურის ჩატარება

სცენარი

თქვენი კომპანიის მარშრუტიზატორები დაკონფიგურებულ იქნა გამოუცდელი ქსელის ინჟინერის მიერ. რამდენიმე შეცდომამ კონფიგურაციაში გამოიწვია კავშირის პრობლემა. თქვენმა უფროსმა გთხოვათ პრობლემის მოძიება, კონფიგურაციის შეცდომების გასწორება და თქვენი სამუშაოს დოკუმენტირება. PPP-სთან მუშაობის თქვენი გამოცდილებით და სტანდარტული ტესტირების მეთოდებით იპოვეთ და გაასწორეთ შეცდომები. დარწმუნდით რომ ყველა სერიალ ლინკი იყენებს PPP CHAP აუთენტიკაციას და დარწმუნდით რომ ყველა ქსელი არის ხელმისაწვდომი. პაროლები არის **cisco** და **class**.

ნაწილი №1: ფიზიკური დონის დიაგნოსტიკისა და შეკეთების პროცედურის გაშვება

პირველი ეტაპი: კაბელირების დიაგნოსტიკა და შეკეთება.

ა. გადახედეთ მისამართების ცხრილს და განსაზღვრეთ ყველა შეერთების ადგილმდებარეობა.

ბ. შეამოწმეთ კაბელები შეერთებულია თუ არა მითითებისამებრ.

გ. ჩაატარეთ ნებისმიერი არააქტიური ინტერფეისის დიაგნოსტიკისა და შეკეთების პროცედურა.

ნაწილი №2: არხის დონის დიაგნოსტიკისა და შეკეთების პროცედურის გაშვება

პირველი ეტაპი: DCE მოწყობილობის გამოკვლევა და ტაქტური სიხშირის დაყენება.

შეისწავლეთ თითოეული მარშრუტიზატორის კონფიგურაცია, რათა დარწმუნდეთ რომ ტაქტური სიხშირე ინტერფეისებზე მომართულია სწორად. მომართეთ ტაქტური სიხშირე ყველა სერიალურ ინტერფეისზე, რომელსაც სჭირდება.

მეორე ეტაპი: DCE მოწყობილობაზე ენკაპსულაციის შესწავლა

ყველა სერიალური ინტერფეისი უნდა იყენებდეს PPP-ს, როგორც ენკაპსულაციის ტიპს. იმ ინტერფეისებზე, სადაც სხვანაირადაა მომართული, შეცვალეთ ენკაპსულაციის ტიპი PPP-ით.

მესამე ეტაპი: CHAP მომხმარებლის სახელებისა და პაროლების შესწავლა და მომართვა.

შეისწავლეთ თითოეული ლინკი რათა შეამოწმოთ სწორად შედიან თუ არა მარშრუტიზატორები ერთმანეთთან. ყველა CHAP პაროლი არის cisco. გამოიყენეთ debug ppp authentication ბრძანება აუცილებლობის შემთხვევაში. გაასწორეთ ან მომართეთ ნებისმიერი საჭირო მომხმარებლის სახელი და პაროლი.

ნაწილი №3: ქსელის დონის დიაგნოსტიკისა და შეკეთების პროცედურის ჩატარება

პირველი ეტაპი: IP დამისამართების შემოწმება

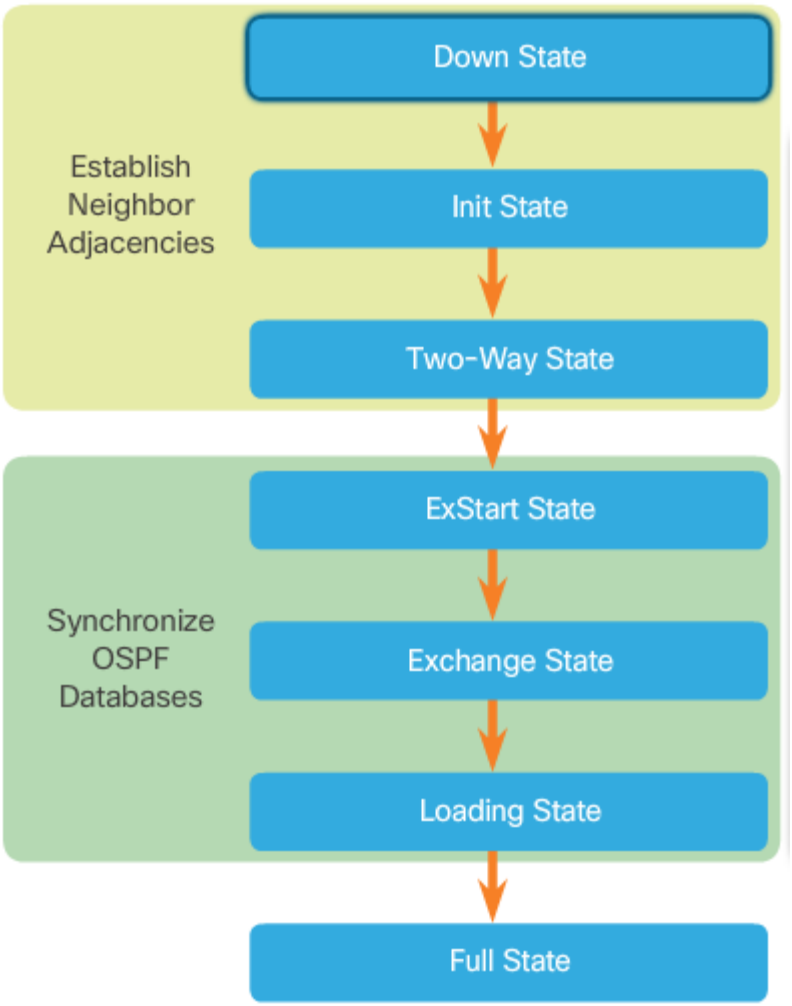
შეამოწმეთ IP მისამართები მისამართების ცხრილის მიხედვით და დარწმუნდით რომ დაკავშირებული ინტერფეისებია იმყოფებიან სწორ ქვექსელში. გაასწორეთ ნებისმიერი IP მისამართი, რომელიც ახდენს გადაფარვას, არის არასწორ ინტერფეისზე, აქვს არასწორი ქვექსელის მისამართი, ან მომართულია ჰოსტის ან ფართომაუწყებლობითი მისამართით.

მეორე ეტაპი: შეამოწმეთ სრული კავშირი PC1-დან და PC3-დან Web Server-თან გზის თვალყურის დევნების საშუალებით.

6.2. მესამე დონის მარშრუტიზაციის პროტოკოლების დიაგნოსტიკა პრობლემების აღმოფხვრით

6.2.1. *OSPF-ის მდგომარეობები*

OSPF-ის პრობლემის მოსაგვარებლად აუცილებელია იმის გაგება თუ როგორ გადიან OSPF მარშრუტიზატორები OSPF-ის განსხვავებულ მდგომარეობას, როცა დადგინდება საზღვრები.



სურათი გვიჩვენებს OSPF-ის მდგომარეობებს და გვაძლევს თითოეული მდგომარეობის ფუნქციებზე ინფორმაციას.

OSPF მეზობლების პრობლემის აღმოფხვრისას დარწმუნებული იყავით რომ FULL ან 2WAY მდგომარეობები არის ნორმალური. ყველა დანარჩენი მდგომარეობა არის

დროებითი; ანუ მარშრუტიზატორი არ უნდა დარჩეს რომელიმე მდგომარეობაში ხანგრძლივი დროის პერიოდში.

6.2.2. *OSPF-ის პრობლემის მოძიებისა და აღმოფხვრის ბრძანებები*

არსებობს OSPF-ის ბევრი სხვადასხვა ბრძანება, რომლებიც შეიძლება გამოყენებულ იქნან პრობლემის მოძიებისა და აღმოფხვრის პროცესში. ქვემოთ მოცემულია ყველაზე გავრცელებული ბრძანებები:

- **show ip protocols** (პირველი სურათი) - გამოიყენება OSPF-ის სასიცოცხლო მნიშვნელობის მქონე კონფიგურაციის ინფორმაციის შესამოწმებლად, მათ შორის OSPF პროცესის იდენტიფიკატორი (ID), მარშრუტიზატორის იდენტიფიკატორი (ID), ქსელები აფიშირებენ თუ არა მარშრუტიზატორს, საიდან იღებენ მარშრუტიზატორის მეზობლები განახლებებს და ნაგულისხმევი ადმინისტრაციული მანძილი, რომელიც OSPF-სთვის არის 110.

```
R1# show ip protocols
*** IP Routing is NSF aware ***

Routing Protocol is "ospf 10"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Router ID 1.1.1.1
  Number of areas in this router is 1. 1 normal 0 stub 0 nssa
  Maximum path: 4
  Routing for Networks:
    172.16.1.1 0.0.0.0 area 0
    172.16.3.1 0.0.0.0 area 0
    192.168.10.5 0.0.0.0 area 0
  Passive Interface(s):
    GigabitEthernet0/0
  Routing Information Sources:
    Gateway         Distance      Last Update
    3.3.3.3              110      00:08:35
    2.2.2.2              110      00:08:35
  Distance: (default is 110)

R1#
```

- **show ip ospf neighbor** (მეორე სურათი) - გამოიყენება იმის შესამოწმებლად, რომ მარშრუტიზატორმა ჩამოაყალიბა მოსაზღვრეობა მეზობელ მარშრუტიზატორებ-თან. გვიჩვენებს მეზობელი მარშრუტიზატორის იდენტიფიკატორს, მეზობლის პრიორიტეტებს, OSPF-ის მდგომარეობას, Dead timer, მეზობლის ინტერფეისის IP მისამართს და ინტერფეისს, რომლითაც მეზობელი არის პირდაპირ წვდომადი. თუ მოსაზღვრე მარშრუტიზატორის იდენტიფიკატორი არ არის ნაჩვენები, ან თუ არ უჩვენებს FULL ან 2WAY მდგომარეობას, ე.ი ორ მარშრუტიზატორს არ აქვს ჩამოყალიბებული OSPF მოსაზღვრეობა. თუ ორი მარშრუტიზატორი ვერ აყალიბებს მოსაზღვრეობას, მაშინ არხის მდგომარეობის შესახებ ინფორმაცია არ გაიცვლება. არასრულმა არხის მდგომარეობის მონაცემთა ბაზამ შეიძლება გამოიწვიოს SPF კალაპოტებისა (trees) და მარშრუტიზაციის ცხრილების უზუსტობა. დანიშნულების ქსელების მარშრუტები შეიძლება არ არსებობდეს ან არ იყოს ყველაზე ოპტიმალური გზა.

```
R1# show ip ospf neighbor

Neighbor ID Pri State        Dead Time Address     Interface
2.2.2.2       1 FULL/BDR     00:00:30  192.168.1.2 GigabitEthernet0/0
3.3.3.3       0 FULL/DROTHER 00:00:38  192.168.1.3 GigabitEthernet0/0
R1#
```

- **show ip ospf interface** (მესამე სურათი) - გამოიყენება ინტერფეისზე დაკონფიგურებული OSPF პარამეტრების საჩვენებლად, როგორიცაა, OSPF პროცესის იდენტიფიკატორი (ID), რომელიც ინტერფეისზეა მინიჭებული, სივრცე რომელშიც იმყოფებიან ინტერფეისები, ინტერფეისის ღირებულება (cost) და Hello და Dead ინტერვალები. ინტერფეისის სახელისა და ნომრის დამატება ბრძანებაზე აჩვენებს შედეგს კონკრეტული ინტერფეისისთვის.

```
R1# show ip ospf interface Serial 0/0/0
Serial0/0/0 is up, line protocol is up
  Internet Address 172.16.3.1/30, Area 0, Attached via Network
Statement
  Process ID 10, Router ID 1.1.1.1, Network Type POINT_TO_POINT,
Cost: 64
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           64        no          no            Base
  Transmit Delay is 1 sec, State POINT_TO_POINT
  Timer intervals configured, Hello 5, Dead 20, Wait 20, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:02
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Index 2/2, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 2.2.2.2
  Suppress hello for 0 neighbor(s)
  Message digest authentication enabled
    Youngest key id is 1
R1#
```

- **show ip ospf** (მეოთხე სურათი) - გამოიყენება OSPF პროცესისა იდენტიფიკატორისა და მარშრუტიზატორის იდენტიფიკატორის შესასწავლად. დამატებით, მოცემული ბრძანება უჩვენებს OSPF სივრცის ინფორმაციას, ასევე SPF ალგორითმის გამოთვლის ბოლო დროს.

```
R1# show ip ospf
Routing Process "ospf 10" with ID 1.1.1.1
 Start time: 00:02:19.116, Time elapsed: 00:01:00.796
 Supports only single TOS(TOS0) routes
 Supports opaque LSA
 Supports Link-local Signaling (LLS)
 Supports area transit capability
 Supports NSSA (compatible with RFC 3101)
 Event-log enabled, Maximum number of events: 1000, Mode: cyclic
 Router is not originating router-LSAs with maximum metric
 Initial SPF schedule delay 5000 msecs
 Minimum hold time between two consecutive SPFs 10000 msecs
 Maximum wait time between two consecutive SPFs 10000 msecs
 Incremental-SPF disabled
 Minimum LSA interval 5 secs
 Minimum LSA arrival 1000 msecs
 LSA group pacing timer 240 secs
 Interface flood pacing timer 33 msecs
 Retransmission pacing timer 66 msecs
 Number of external LSA 1. Checksum Sum 0x00A1FF
 Number of opaque AS LSA 0. Checksum Sum 0x000000
 Number of DCbitless external and opaque AS LSA 0
 Number of DoNotAge external and opaque AS LSA 0
 Number of areas in this router is 1. 1 normal 0 stub 0 nssa
 External flood list length 0
 IETF NSF helper support enabled
 Cisco NSF helper support enabled
 Reference bandwidth unit is 100 mbps
    Area BACKBONE(0)
        Number of interfaces in this area is 3
Area has no authentication
SPF algorithm last executed 00:00:36.936 ago
SPF algorithm executed 3 times
Area ranges are
Number of LSA 3. Checksum Sum 0x016D60
Number of opaque link LSA 0. Checksum Sum 0x000000
Number of DCbitless LSA 0
Number of indication LSA 0
Number of DoNotAge LSA 0
Flood list length 0

R1#
```

- **show ip route ospf** (მეხუთე სურათი) - გამოიყენება მხოლოდ OSPF-ად ცნობილი მარშრუტების საჩვენებლად მარშრუტიზაციის ცხრილში. შედეგიდან ჩანს რომ R1 მარშრუტიზატორმა OSPF-ით შეიტყო ოთხი დაშორებული ქსელის შესახებ.

```
R1# show ip route ospf
Codes:L - local,C - connected,S - static,R - RIP,M - mobile,B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS,su - IS-IS summary,L1 - IS-IS level-1,L2-IS-IS level-2
      ia - IS-IS inter area,*-candidate default,U-per-user static route
      o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
      + - replicated route, % - next hop override

Gateway of last resort is 172.16.3.2 to network 0.0.0.0

O*E2  0.0.0.0/0 [110/1] via 172.16.3.2, 00:33:17, Serial0/0/0
      172.16.0.0/16 is variably subnetted, 5 subnets, 3 masks
O       172.16.2.0/24 [110/65] via 172.16.3.2, 00:33:17,Serial0/0/0
O     192.168.1.0/24 [110/65] via 192.168.10.6, 00:30:43,Serial0/0/1
      192.168.10.0/24 is variably subnetted, 3 subnets, 2 masks
O       192.168.10.8/30[110/128] via 192.168.10.6,00:30:43,Serial0/0/1
                        [110/128] via 172.16.3.2,00:33:17,Serial0/0/0
R1#
```

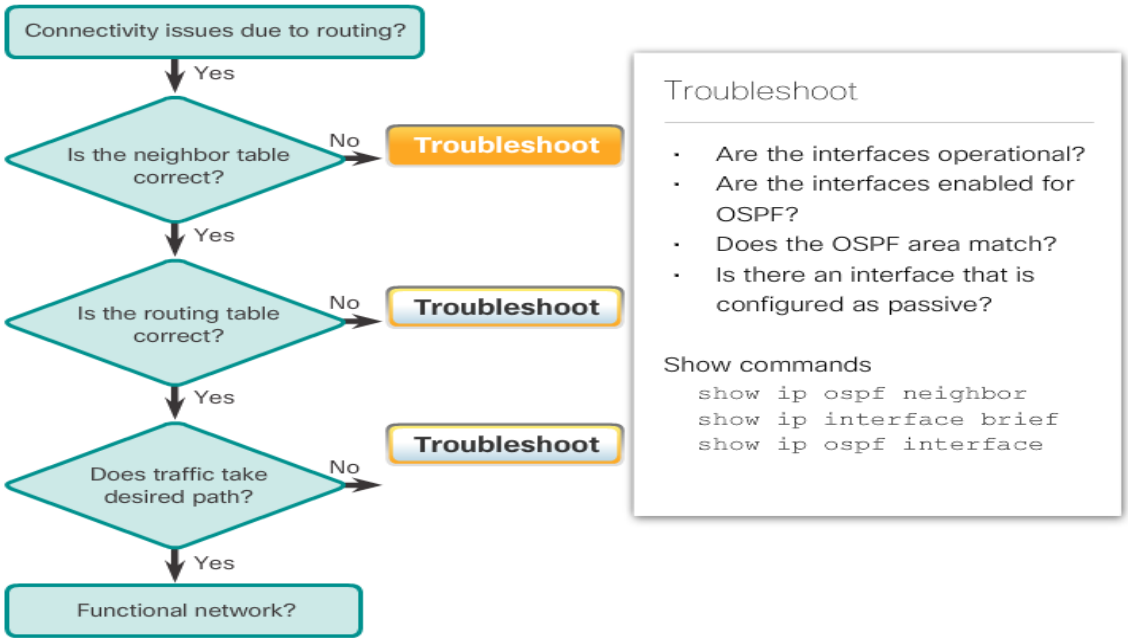
- **clear ip ospf [process-id] process** - გამოიყენება OSPFv2 მეზობელი მოსაზღვრეობის გასაუქმებლად.

6.2.3. *OSPF-ის პრობლემის მოძიებისა და აღმოფხვრის პროცესის კომპონენტები*

როგორც სურათზეა ნაჩვენები OSPF პრობლემები როგორც წესი დაკავშირებულია:

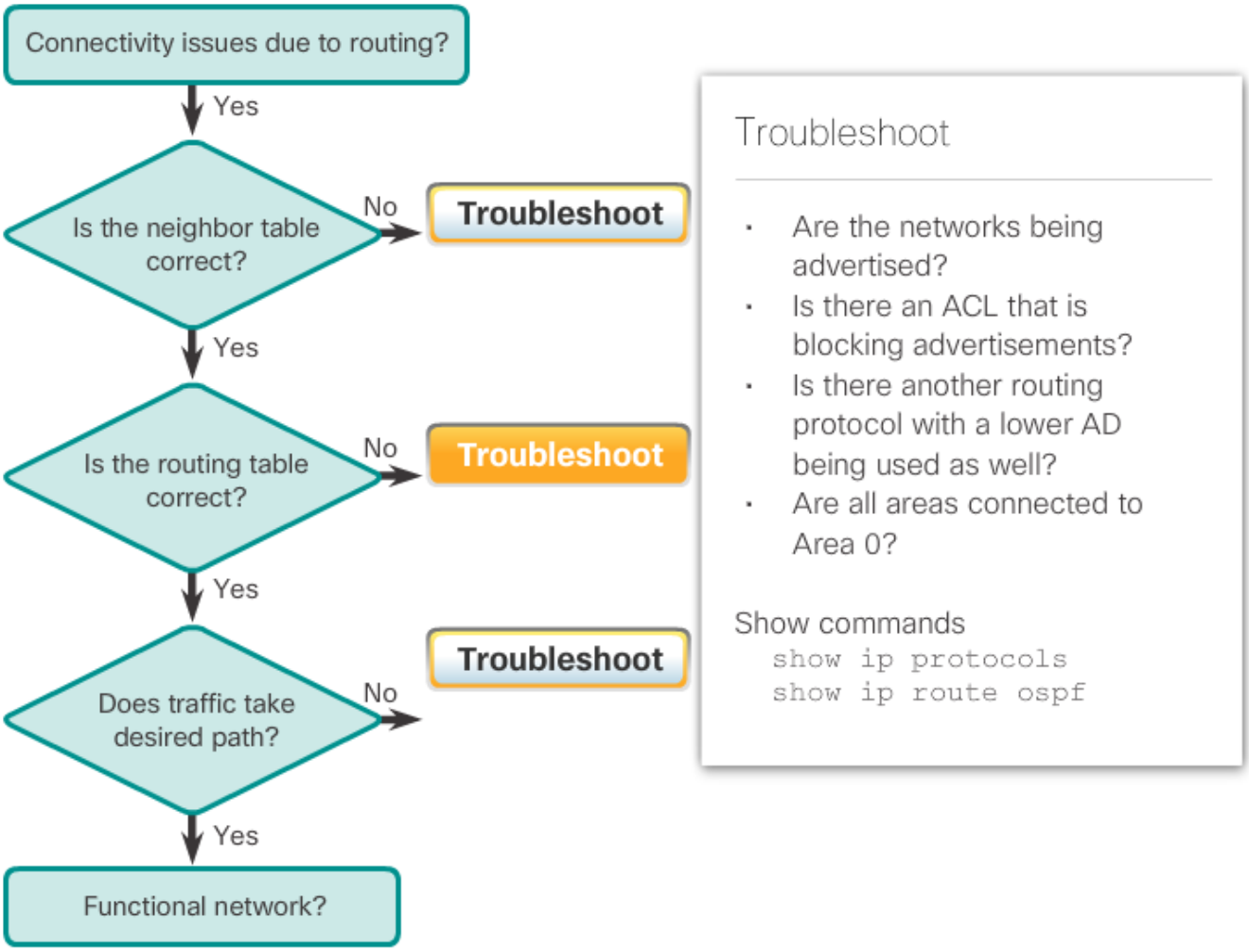
- მეზობელ მოსაზღვრეობასთან
- დაკარგულ მარშრუტებთან
- გზის არჩევასთან

როდესაც აგვარებთ მეზობლობის პრობლემებს, show ip ospf neighbor ბრძანების გამოყენებით შეამოწმეთ მარშრუტიზატორს აქვს თუ არა შექმნილი მოსაზღვრეობა მეზობელ მარშრუტიზატორებთან. თუ არ არსებობს მოსაზღვრეობა, მაშინ მარშრუტიზატორები ვერ გაცვლიან მარშრუტებს. show ip interface brief და show ip ospf interface ბრძანებების გამოყენებით შეამოწმეთ OSPF-სთვის ინტერფეისები ჩართულია თუ არა და თუ ფუნქციონირებს. თუ OSPF-სთვის ინტერფეისები აქტიურია და ფუნქციური, მაშინ დარწმუნდით რომ ორივე მარშრუტიზატორზე ინტერფეისები დაკონფიგურებულია იგივე OSPF სივრცისთვის და ინტერფეისები არ არის დაკონფიგურებული, როგორც პასიური ინტერფეისები.

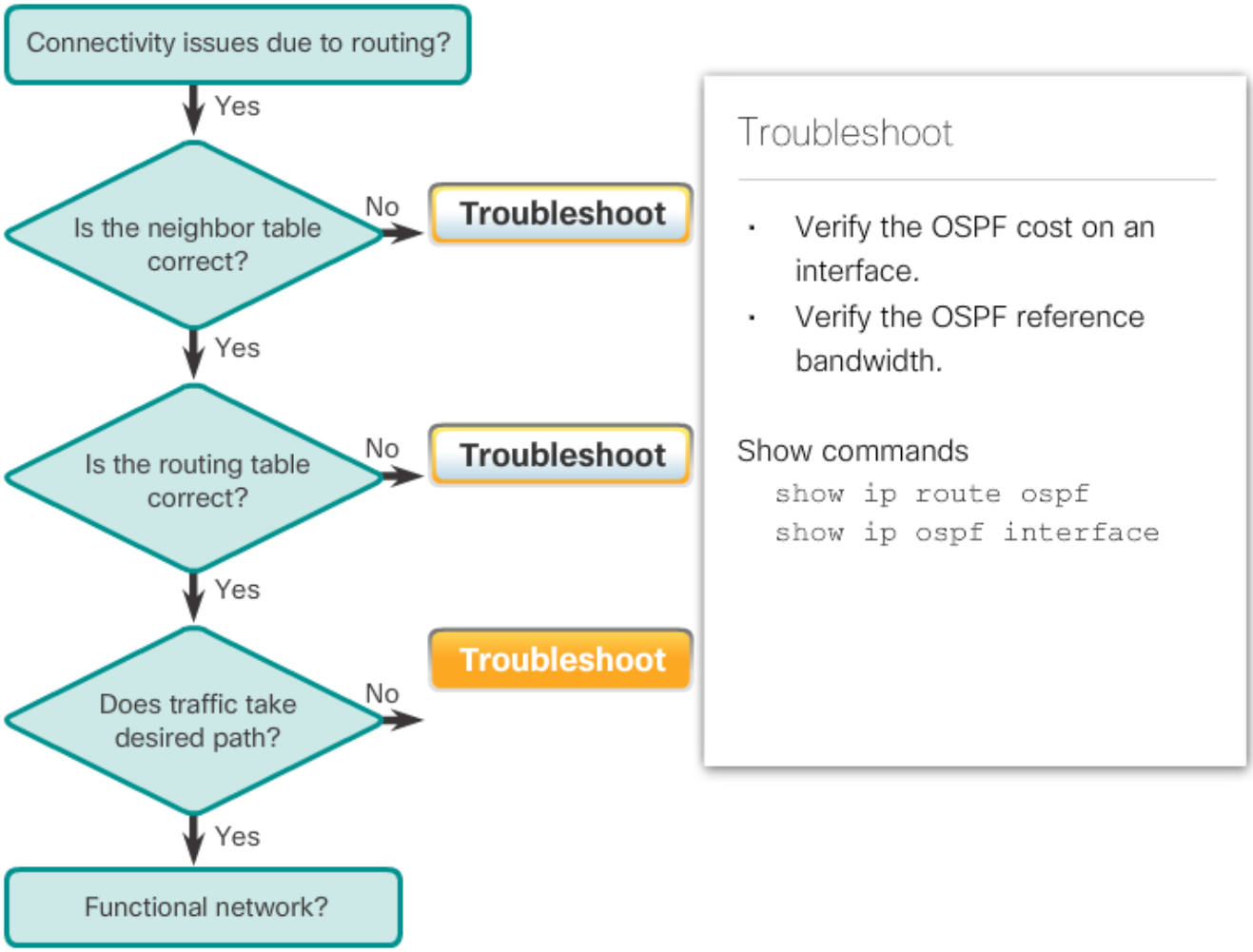


თუ ორ მარშრუტიზატორს შორის ჩამოყალიბებულია მოსაზღვრეობა, show ip route ospf ბრძანების გამოყენებით შეამოწმეთ რომ მარშრუტიზაციის ცხრილში არსებობს OSPF

მარშრუტები. თუ არ არსებობს OSPF მარშრუტები, დარწმუნდით რომ ქსელში არ არის გაშვებული სხვა მარშრუტიზაციის პროტოკოლი დაბალი ადმინისტრაციული მანძილით. შეამოწმეთ ყველა მოთხოვნილი ქსელი არის თუ არა აფიშირებული OSPF-ში. ასევე შეამოწმეთ არის თუ არა დაკონფიგურებული წვდომის სია მარშრუტიზატორზე, რომელიც ფილტრავს როგორც შემომავალ ისე გამავალ მარშრუტიზაციის განახლებებს.

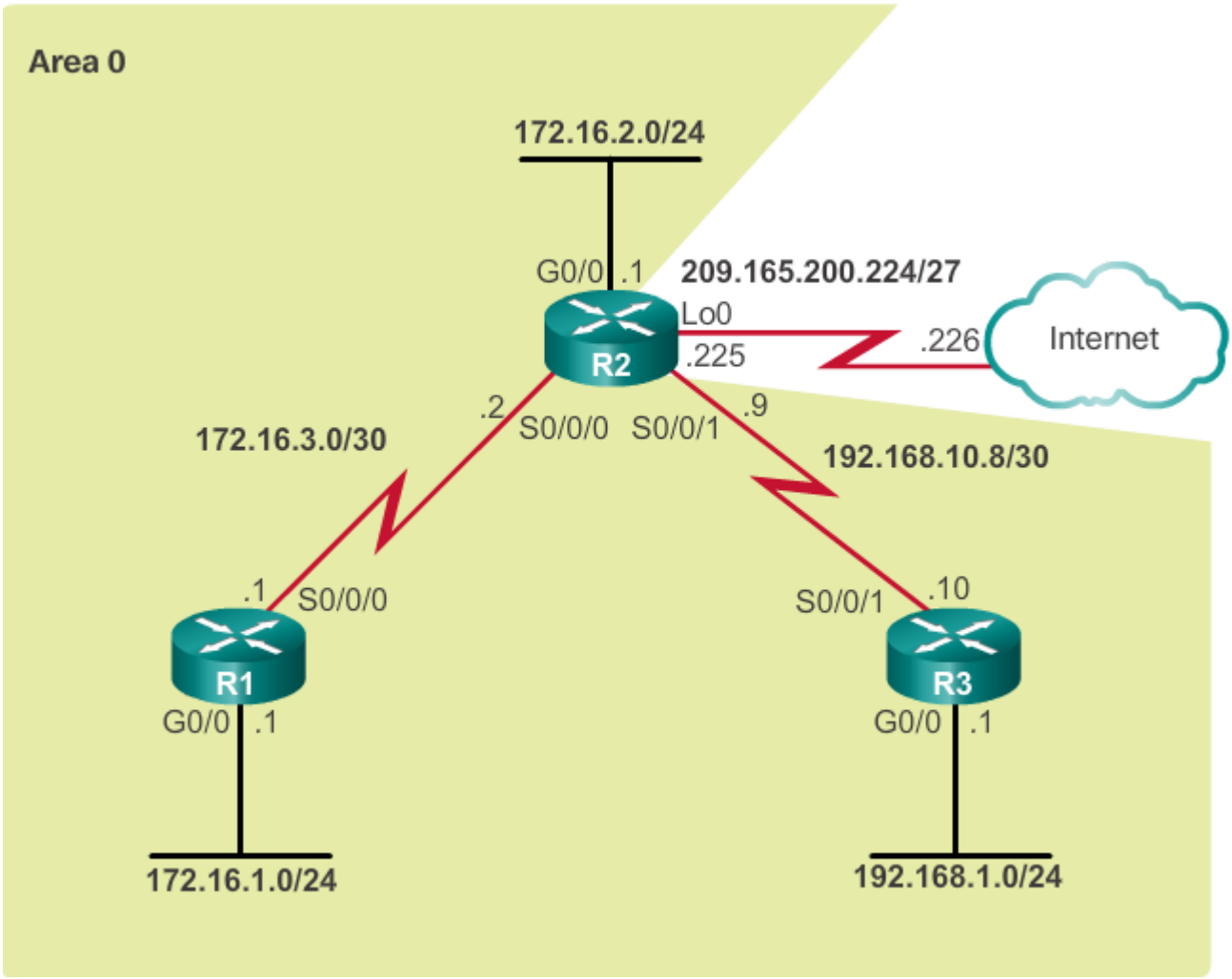


თუ ყველა მოთხოვნილი მარშრუტი არის მარშრუტიზაციის ცხრილში, მაგრამ გზა რომელსაც ტრაფიკი ქმნის არ არის სწორი, შეამოწმეთ გზაზე ინტერფეისების OSPF ღირებულება (cost). ასევე იყავით ფრთხილად იმ შემთხვევაში, როცა ინტერფეისები არის 100მგ/წმ-ზე მეტი სიჩქარის, რადგან ყველა ინტერფეისს ამ გამტარობაზე მეტი მნიშვნელობის შემთხვევაში ნაგულისხმევად აქვს იგივე OSPF ღირებულება.


Connectivity issues due to routing?
Yes
Is the neighbor table correct?
No
Troubleshoot
Yes
Is the routing table correct?
No
Troubleshoot
Yes
Does traffic take desired path?
No
Troubleshoot
Yes
Functional network?
Troubleshoot
Verify the OSPF cost on an interface.
Verify the OSPF reference bandwidth.
Show commands
show ip route ospf
show ip ospf interface

6.2.4. *მეზობლობის პრობლემების მოძიება და აღმოფხვრა*

მოცემული მაგალითი გვიჩვენებს თუ როგორ ვიპოვოთ და გამოვასწოროთ მეზობლობის პრობლემები. პირველ სურათზე მოცემულ ტოპოლოგიაში ყველა მარშრუტიზატორი დაკონფიგურებულია ისე, რომ მხარი დაუჭირონ OSPF მარშრუტიზაციას.



მეორე სურათზე ნაჩვენები R1 მარშრუტიზატორის მარშრუტიზაციის ცხრილის სწრაფი დათვალიერებით გამოვლინდა, რომ არანაირი OSPF მარშრუტი არ არის დამატებული. არსებობს ამის რამოდენიმე მიზეზი. თუმცა ორ მარშრუტიზატორს შორის მეზობლური ურთიერთობის ჩამოყალიბების წინაპირობა არის OSI მოდელის მესამე დონის შეერთება.

```
R1# show ip route
Codes: L -  local, C - connected, S - static, R - RIP, M -
            mobile, B - BGP
       D -  EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF
            inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA
            external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i -  IS-IS, su - IS-IS summary, L1 - IS-IS level-1,
            L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U -
            per-user static route
       o -  ODR, P - periodic downloaded static route, H -
            NHRP, l - LISP
       + -  replicated route, % - next hop override

Gateway of last resort is not set

   172.16.0.0/16 is variably subnetted, 4 subnets, 3 masks
C   172.16.1.0/24 is directly connected, GigabitEthernet0/0
L   172.16.1.1/32 is directly connected, GigabitEthernet0/0
C   172.16.3.0/30 is directly connected, Serial0/0/0
L   172.16.3.1/32 is directly connected, Serial0/0/0
R1#
```

მესამე სურათზე ნაჩვენები შედეგი ადასტურებს რომ S0/0/0 ინტერფეისი არის up-ში და არის აქტიური. წარმატებული პინგი ასევე ადასტურებს, რომ R2 მარშრუტიზატორის სერიალური ინტერფეისი არის აქტიური. წარმატებული პინგი არ ნიშნავს რომ ჩამოყალიბდება მოსაზღვრეობა, რადგან შესაძლოა ადგილი ჰქონდეს ქვექსელების გადაფარვას. თქვენ კვლავ უნდა გადაამოწმოთ, რომ დაკავშირებული მოწყობილობების ინტერფეისები არიან იგივე ქვექსელში. თუ პინგი არ არის წარმატებული, შეამოწმეთ კაბელების შეერთება და დარწმუნდით, რომ დაკავშირებულ მოწყობილობებზე ინტერფეისები დაკონფიგურებულია სწორად და არიან ფუნქციურნი.

```
R1# show ip interface brief
Interface                  IP-Address    OK?  Method  Status
Embedded-Service-Engine0/0 unassigned    YES  unset   administr
GigabitEthernet0/0         172.16.1.1    YES  manual  up
GigabitEthernet0/1         unassigned    YES  unset   administr
Serial0/0/0                172.16.3.1    YES  manual  up
Serial0/0/1                unassigned    YES  TFTP    up
R1#
R1# ping 172.16.3.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.3.2, timeout is 2 seco
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 12/
R1#
```

ინტერფეისისთვის უნდა იყოს დაშვებული OSPF, შესაბამისი network ბრძანება უნდა იყოს კონფიგურირებული OSPF მარშრუტიზაციის პროცესის ქვეშ. აქტიური OSPF ინტერფეისები შეიძლება შემოწმდეს show ip ospf interface ბრძანების გამოყენებით. მეოთხე სურათზე ნაჩვენები შედეგი ადასტურებს რომ სერიალური 0/0/0 ინტერფეისი დაშვებულია OSPF-სთვის. თუ ორ მარშრუტიზატორზე დაკავშირებული ინტერფეისები არაა დაშვებული OSPF-სთვის, მეზობლები ვერ ჩამოაყალიბებენ მეზობლობას.

```
R1# show ip ospf interface serial 0/0/0
Serial0/0/0 is up, line protocol is up
  Internet Address 172.16.3.1/30, Area 0, Attached via
  Network Statement
  Process ID 10, Router ID 1.1.1.1, Network Type
  POINT_TO_POINT, Cost: 64
  Topology-MTID    Cost  Disabled  Shutdown  Topology Name
        0           64     no          no       Base
  Transmit Delay is 1 sec, State POINT_TO_POINT
  Timer intervals configured, Hello 5, Dead 20, Wait 20,
  Retransmit 5
    oob-resync timeout 40
    No Hellos (Passive interface)
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Index 2/2, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 0, Adjacent neighbor count is 0
  Suppress hello for 0 neighbor(s)
  Message digest authentication enabled
    Youngest key id is 1
```

show ip protocols ბრძანების გამოყენებით შეამოწმეთ OSPF-ის პარამეტრები. მეხუთე სურათზე ნაჩვენები შედეგი ადასტურებს, რომ OSPF ჩართულია და ასევე ჩამოთვლილია ქსელები, რომლებიც არიან აფიშირდებიან როგორც დაშვებულნი, network ბრძანებით. თუ IP მისამართი ინტერფეისზე მოდის ქსელში, რომელიც დაშებულ იქნა OSPF-სთვის, ინტერფეისი დაშვებული იქნება OSPF-სთვის.

```
R1# show ip protocols
*** IP Routing is NSF aware ***

Routing Protocol is "ospf 10"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Router ID 1.1.1.1
  Number of areas in this router is 1. 1 normal 0 stub 0 nssa
  Maximum path: 4
  Routing for Networks:
    172.16.1.1 0.0.0.0 area 0
    172.16.3.1 0.0.0.0 area 0
  Passive Interface(s):
    GigabitEthernet0/0
    Serial0/0/0
  Routing Information Sources:
    Gateway         Distance      Last Update
    3.3.3.3              110      00:50:03
    2.2.2.2              110      04:27:25
  Distance: (default is 110)

R1#
```

თუმცა აღსანიშნავია, რომ სერიალური 0/0/0 ინტერფეისი არის წარმოდგენილი როგორც პასიური. შეგახსენებთ, რომ passive-interface ბრძანება აჩერებს როგორც გამავალ, ისე შემომავალ მარშრუტიზაციის განახლებებს, რადგან ბრძანება მარშრუტიზატორზე იწვევს Hello პაკეტების გაგზავნისა და მიღების შეჩერებას ინტერფეისზე. ამ მიზეზით მარშრუტიზატორები ვერ ხდებიან მეზობლები.

პასიური ინტერფეისის გასათიშად გამოიყენეთ no passive-interface მარშრუტიზატორის კონფიგურაციის რეჟიმის ბრძანება, ისე როგორც ნაჩვენებია მეექვსე სურათზე. მას შემდეგ რაც გათიშავთ პასიურ ინტერფეისს, მარშრუტიზატორები გახდებიან მოსაზღვრეები, როგორც მითითებულია ავტომატურად შექმნილი საინფორმაციო შეტყობინების მიერ.

```
R1(config)# router ospf 10
R1(config-router)# no passive-interface s0/0/0
R1(config-router)#
*Apr  9 13:14:15.454: %OSPF-5-ADJCHG: Process 10, Nbr
2.2.2.2 on Serial0/0/0 from LOADING to FULL, Loading Done
R1(config-router)# end
R1#
```

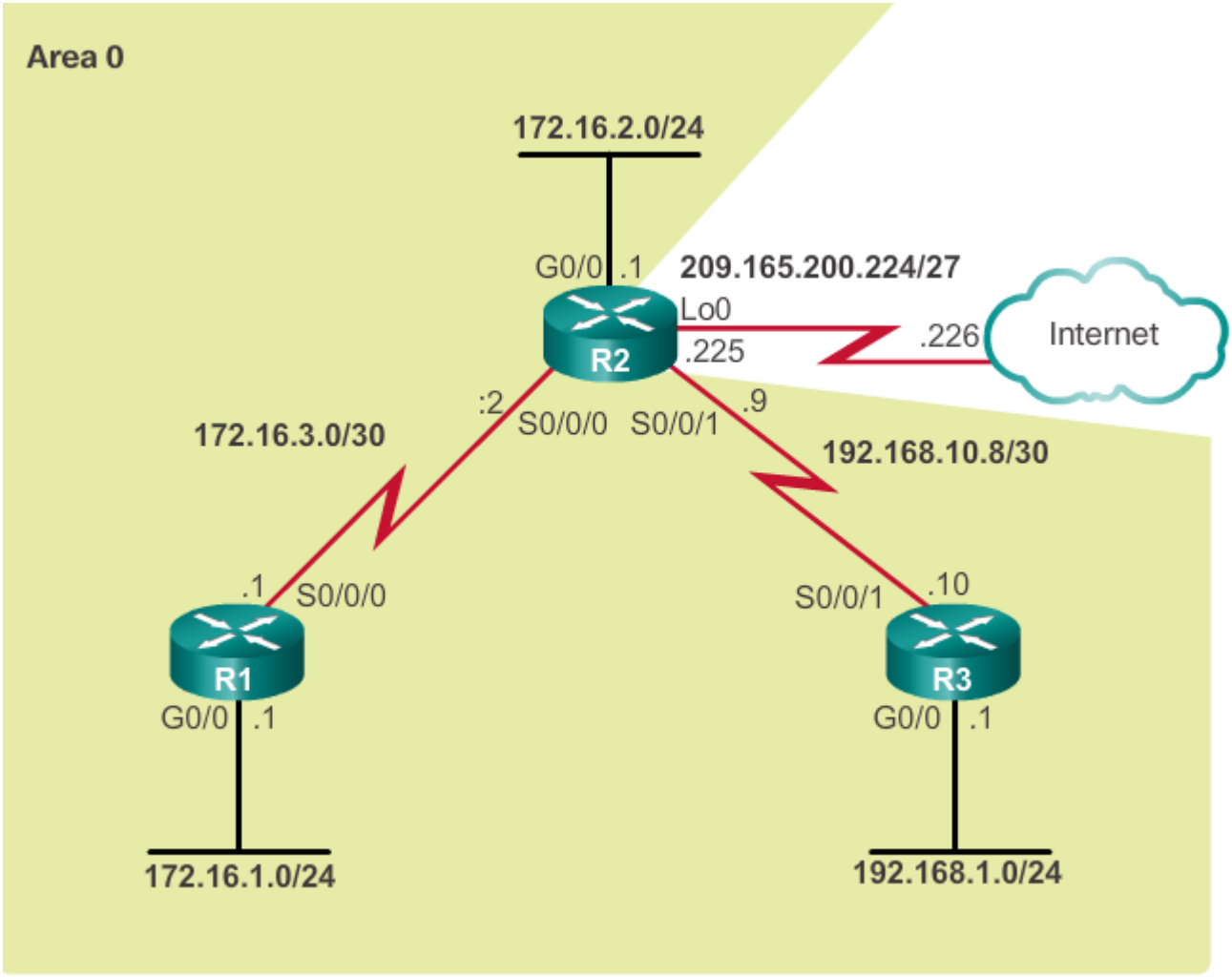
ქვემოთ სურათზე ნაჩვენები მარშრუტიზაციის ცხრილის სწრაფი შემოწმება ადასტურებს, რომ OSPF ახლა უკვე ცვლის მარშრუტიზაციის ინფორმაციას.

```
R1# show ip route ospf
Codes: L -  local, C - connected, S - static, R - RIP, M -
            mobile, B - BGP
       D -  EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF
            inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA
            external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i -  IS-IS, su - IS-IS summary, L1 - IS-IS level-1,
            L2 -IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U -
            per-user static route
       o -  ODR, P - periodic downloaded static route, H -
            NHRP, l - LISP
       + -  replicated route, % - next hop override

Gateway of last resort is 172.16.3.2 to network 0.0.0.0

O*E2  0.0.0.0/0 [110/1] via 172.16.3.2, 00:00:18,
Serial0/0/0
      172.16.0.0/16 is variably subnetted, 5 subnets, 3
masks
O      172.16.2.0/24 [110/65] via 172.16.3.2, 00:00:18,
       Serial0/0/0
O     192.168.1.0/24 [110/129] via 172.16.3.2, 00:00:18,
      Serial0/0/0
      192.168.10.0/30 is subnetted, 1 subnets
O        192.168.10.8 [110/128] via 172.16.3.2, 00:00:18,
         Serial0/0/0
R1#
```

სხვა პრობლემა რომელიც შეიძლება წარმოიქმნეს არის, როცა ორი მეზობელ მარშრუტიზატორს აქვს არათავსებადი MTU ზომები თავიანთ დაკავშირებულ ინტერფეისებზე. MTU ზომა არის უდიდესი ქსელის დონი პაკეტი, რომელსაც მარშრუტიზატორი აწვდის თითოეულ ინტერფეისს. მარშრუტიზატორების ნაგულისხმევი MTU ზომა არის 1500 ბაიტი. თუმცა ეს მნიშვნელობა შეიძლება შეიცვალოს IPv4 პაკეტებისთვის ip mtu *size* ინტერფეისის კონფიგურაციის ბრძანების გამოყენებით ან ipv6 mtu *size* ინტერფეისის ბრძანებით, IPv6 პაკეტებისათვის. თუ ორ დაკავშირებულ მარშრუტიზატორს აქვს შეუთავსებელი MTU მნიშვნელობები, ისინი მაინც ეცდებიან მეზობლობის ჩამოყალიბებას, მაგრამ ისინი ვერ გაცვლიან თავიანთ არხის მდგომარეობის მონაცემთა ბაზას (Link State Database – LSDB) და მეზობლური კავშირი ჩავარდება.

6.2.5. *OSPF მარშრუტიზაციის ცხრილის პრობლემების მოძიება და აღმოფხვრა*

პირველ სურათზე ნაჩვენებ ტოპოლოგიაში ყველა მარშრუტიზატორი დაკონფიგურებულია ისე რომ მხარი დაუჭირონ OSPF მარშრუტიზაციას.



R1 მარშრუტიზატორის მარშრუტიზაციის სწრაფი დათვალიერებისას გამოვლინდა (მეორე სურათი), რომ ის იღებს მარშრუტიზაციის ნაგულისხმევ ინფორმაციას, R2 მარშრუტიზატორის ლოკალური ქსელი (LAN) (172.16.2.0/24) და R2 და R3 მარშრუტიზატორებს შორის კავშირი (192.168.10.8/30). თუმცა ის ვერ იღებს R3 ლოკალური ქსელის OSPF მარშრუტს.

```
R1# show ip route
Codes: L -  local, C - connected, S - static, R - RIP, M -
            mobile, B - BGP
       D -  EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF
            inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA
            external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i -  IS-IS, su - IS-IS summary, L1 - IS-IS level-1,
            L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U -
            per-user static route
       o -  ODR, P - periodic downloaded static route, H -
            NHRP, l - LISP
       + -  replicated route, % - next hop override

Gateway of last resort is 172.16.3.2 to network 0.0.0.0

O*E2  0.0.0.0/0 [110/1] via 172.16.3.2, 00:05:26,
      Serial0/0/0
      172.16.0.0/16 is variably subnetted, 5 subnets, 3
      masks
C        172.16.1.0/24 is directly connected,
         GigabitEthernet0/0
L        172.16.1.1/32 is directly connected,
         GigabitEthernet0/0
O        172.16.2.0/24 [110/65] via 172.16.3.2, 00:05:26,
         Serial0/0/0
C        172.16.3.0/30 is directly connected, Serial0/0/0
L        172.16.3.1/32 is directly connected, Serial0/0/0
      192.168.10.0/30 is subnetted, 1 subnets
O        192.168.10.8 [110/128] via 172.16.3.2, 00:05:26,
         Serial0/0/0
R1#
```

მესამე სურათზე მოცემული შედეგი ამოწმებს OSPF-ის პარამეტრებს R3 მარშრუტიზატორზე. აღსანიშნავია, რომ R3 მარშრუტიზატორი აფიშირებს მხოლოდ R3 და R2 მარშრუტიზატორებს შორის კავშირს. ის არ აფიშირებს R3 ლოკალურ ქსელს (192.168.1.0/24).

```
R3# show ip protocols
*** IP Routing is NSF aware ***
Routing Protocol is "ospf 10"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Router ID 3.3.3.3
  Number of areas in this router is 1. 1 normal 0 stub 0
  nssa
  Maximum path: 4
  Routing for Networks:
    192.168.10.8 0.0.0.3 area 0
  Passive Interface(s):
    Embedded-Service-Engine0/0
    GigabitEthernet0/0
    GigabitEthernet0/1
    GigabitEthernet0/3
    RG-AR-IF-INPUT1
  Routing Information Sources:
    Gateway         Distance      Last Update
    1.1.1.1              110      00:02:48
    2.2.2.2              110      00:02:48
  Distance: (default is 110)

R3#
```

ინტერფეისისთვის უნდა იყოს დაშვებული OSPF, შესაბამისი network ბრძანება უნდა იყოს კონფიგურირებული OSPF მარშრუტიზაციის პროცესის ქვეშ. მეოთხე სურათზე გამოტანილი ინფორმაცია ადასტურებს რომ R3 მარშრუტიზატორის ლოკალური ქსელი არ არის აფიშირებული OSPF-ში.

```
R3# show running-config | section router ospf
router ospf 10
 router-id 3.3.3.3
 passive-interface default
 no passive-interface Serial0/0/1
 network 192.168.10.8 0.0.0.3 area 0
R3#
```

მეხუთე სურათზე მოცემულ მაგალითში დამატებულია network ბრძანება R3 მარშრუტიზატორის ლოკალური ქსელისათვის. R3 მარშრუტიზატორმა ახლა უკვე უნდა მოახდინოს R3 ლოკალური ქსელის აფიშირება თავისი OSPF მეზობლებისათვის.

```
R3# conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)# router ospf 10
R3(config-router)# network 192.168.1.0 0.0.0.255 area 0
R3(config-router)# end
R3#
*Apr 10 11:03:11.115: %SYS-5-CONFIG_I: Configured from
console by console
R3#
```

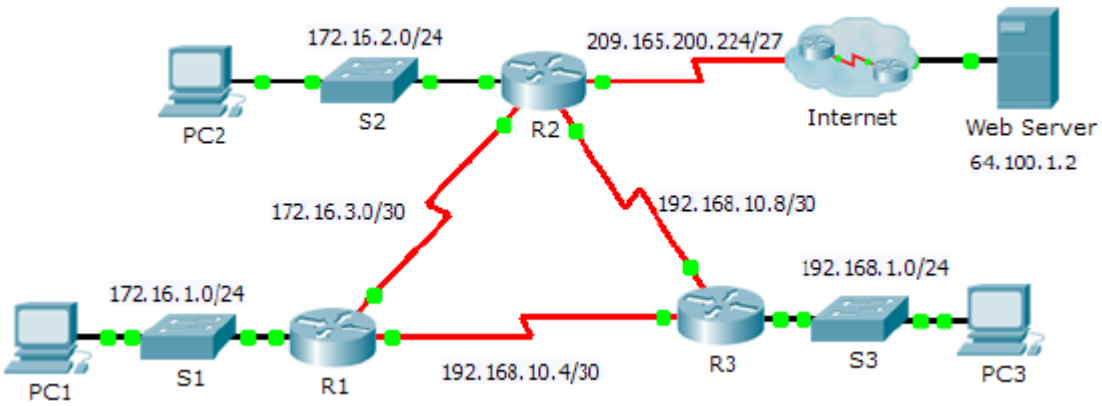
მეექვსე სურათზე მოცემული ინფორმაცია ამოწმებს იმას, რომ R3 ლოკალური ქსელი ახლა უკვე არის R1 მარშრუტიზატორის მარშრუტიზაციის ცხრილში.

```
R1# show ip route ospf
Codes: L -  local, C - connected, S - static, R - RIP, M -
            mobile, B - BGP
       D -  EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF
            inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA
            external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i -  IS-IS, su - IS-IS summary, L1 - IS-IS level-1,
            L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U -
            per-user static route
       o -  ODR, P - periodic downloaded static route, H -
            NHRP, l - LISP
       + -  replicated route, % - next hop override

Gateway of last resort is 172.16.3.2 to network 0.0.0.0
```

```
O*E2  0.0.0.0/0 [110/1] via 172.16.3.2, 00:08:38,
      Serial0/0/0
      172.16.0.0/16 is variably subnetted, 5 subnets, 3
      masks
O        172.16.2.0/24 [110/65] via 172.16.3.2, 00:08:38,
         Serial0/0/0
O     192.168.1.0/24 [110/129] via 172.16.3.2, 00:00:37,
      Serial0/0/0
      192.168.10.0/30 is subnetted, 1 subnets
O        192.168.10.8 [110/128] via 172.16.3.2, 00:08:38,
         Serial0/0/0
R1#
```

6.3. ერთ სივრციანი (Single-Area) OSPFv2-ის პრობლემის მოძიება და აღმოფხვრა

ტოპოლოგია



მისამართების ცხრილი

მოწყობილობა	ინტერფეისი	IP მისამართი	ქვექსელის ნიღაბი	ნაგულისხმევი გასავლელი
R1	G0/0	172.16.1.1	255.255.255.0	N/A
	S0/0/0	172.16.3.1	255.255.255.252	N/A
	S0/0/1	192.168.10.5	255.255.255.252	N/A
R2	G0/0	172.16.2.1	255.255.252.0	N/A
	S0/0/0	172.16.3.2	255.255.255.252	N/A
	S0/0/1	192.168.10.9	255.255.255.252	N/A
	S0/1/0	209.165.200.225	255.255.255.224	N/A
R3	G0/0	192.168.1.1	255.255.252.0	N/A
	S0/0/0	192.168.10.6	255.255.255.252	N/A
	S0/0/1	192.168.10.10	255.255.255.252	N/A

PC1	ქსელის ადაპტერი	172.16.1.2	255.255.255.0	172.16.1.1
PC2	ქსელის ადაპტერი	172.16.2.2	255.255.255.0	172.16.2.1
PC3	ქსელის ადაპტერი	192.168.1.2	255.255.255.0	192.168.1.1

სცენარი

მოცემულ დავალებაში თქვენ უნდა მოაგვაროთ OSPF მარშრუტიზაციის პრობლემები **ping** და **show** ბრძანებების დახმარებით, რათა მოახდინოთ ქსელის კონფიგურაციის შეცდომების იდენტიფიკაცია. შემდეგ უნდა მოახდინოთ თქვენს მიერ აღმოჩენილი შეცდომების და შესაბამისი გადაწყვეტის გზების დოკუმეტირება. ბოლოს თქვენ შეამოწმებთ აღდგა თუ არა ერთმანეთთან კავშირი.

პრობლემის აღმოფხვრის პროცესი

1. გამოიყენეთ შემმოწმებელი ბრძანებები ქსელში კავშირის პრობლემების აღმოსაჩენად და მოახდინეთ პრობლემის დოკუმენტირება დოკუმენტაციის ცხრილში.

2. გამოიყენეთ ვერიფიკაციის ბრძანებები პრობლემის წყაროს დასადგენად და შეიმუშავეთ ამ პრობლემის გადაჭრის გეგმა.

3. შეასრულეთ თითოეული პრობლემის გადაწყვეტა ერთხელ და შეამოწმეთ მოგვარდა თუ არა პრობლემა. შეიტანეთ შესაბამისი გადაწყვეტილება დოკუმენტაციის ცხრილში.

4. თუ პრობლემა არ მოგვარდა, აუცილებელია პირველ რიგში გაუქმდეს შესრულებული გადაწყვეტები, სანამ დაუბრუნდებით მეორე ეტაპს.

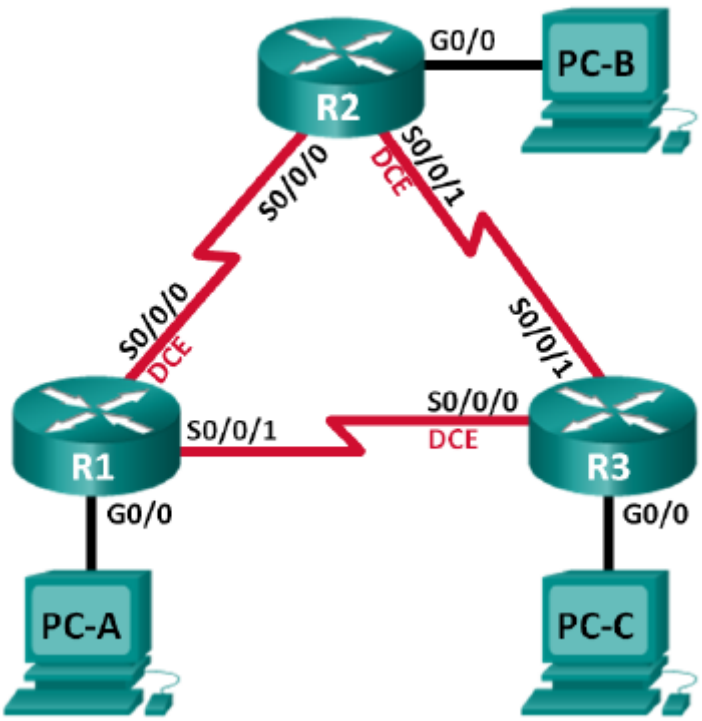
5. როდესაც ყველა გამოვლენილი პრობლემა მოგვარდება, შეამოწმეთ ერთმანეთს შორის კავშირი.

დოკუმენტაციის ცხრილი

მოწყობილობა	გამოვლენილი პრობლემა	შეთავაზებული გადაწყვეტა	მოგვარდა?

6.3.1. *- ერთსივრციანი (Single-Area) OSPFv2-სა და OSPFv3-ის მთავარი პრობლემების მოძიება და გამოსწორება*

ტოპოლოგია



მისამართების ცხრილი

მოწყობილობა	OSPF მარშრუტიზატორის ID	ინტერფეისი	IP მისამართი	ნაგულისხმევი გასასვლელი
R1	1.1.1.1	G0/0	192.168.1.1/24 2001:DB8:ACAD:A::1/64 FE80::1 link-local	N/A
		S0/0/0	192.168.12.1/30 2001:DB8:ACAD:12::1/64 FE80::1 link-local	N/A

		S0/0/1	192.18.13.1/30 2001:DB8:ACAD:13::1/64	N/A
R2	2.2.2.2	G0/0	192.168.2.1/24 2001:DB8:ACAD:B::2/64 FE80::2 link-local	N/A
		S0/0/0	192.168.12.2/30 2001:DB8:ACAD:12::2/64 FE80::2 link-local	N/A
		S0/0/1	192.168.23.1/30 2001:DB8:ACAD:23::2/64 FE80::2 link-local	N/A
R3	3.3.3.3	G0/0	192.168.3.1/24 2001:DB8:ACAD:C::3/64 FE80::3 link-local	N/A
		S0/0/0	192.168.13.2/30 2001:DB8:ACAD:13::3/64 FE80::3 link-local	N/A
		S0/0/1	192.168.23.2/30 2001:DB8:ACAD:23::3/64 FE80::3 link-local	N/A
PC-A		ქსელის ადაპტერი	192.168.1.3/24 2001:DB8:ACAD:A::A/64	192.168.1.1 FE80::1

PC-B		ქსელის ადაპტერი	192.168.2.3/24 2001:DB8:ACAD:B::B/64	192.168.2.1 FE80::2
PC-C		ქსელის ადაპტერი	192.168.3.3/24 2001:DB8:ACAD:C::C/64	192.168.3.1 FE80::3

შესასრულებელი სამუშაოები

ნაწილი №1: ქსელის აწყობა და მოწყობილობის კონფიგურაციების ჩატვირთვა

ნაწილი №2: მესამე დონის კავშირის პრობლემის მოგვარება

ნაწილი №3: OSPFv2-ის პრობლემის აღმოფხვრა

ნაწილი №4: OSPFv3-ის პრობლემის აღმოფხვრა

ძირითადი ინფორმაცია / სცენარი

Open Shortest Path First (OSPF) არის არხის-მდგომარეობის (link-state) მარშრუტიზაციის პროტოკოლი IP ქსელებისათვის. OSPFv2 განსაზღვრულია IPv4 ქსელებისათვის, ხოლო OSPFv3 – Ipv6 ქსელებისათვის. OSPFv2 და OSPFv3 არის სრულად იზოლირებული მარშრუტიზაციის პროტოკოლები, OSPFv2-ში რაიმე ცვლილების შეტანა გავლენას არ ახდენს OSPFv3 მარშრუტიზაციაზე და პირიქით.

მოცემულ ლაბორატორიულ დავალებაში ერთსივრციან (single-area) OSPF ქსელში გაშვებულ OSPFv2 და OSPFv3 აქვთ პრობლემები. თქვენ დავალებული გაქვთ მოძებნოთ პრობლემები და გამოასწოროთ ისინი.

შენიშვნა: მარშრუტიზატორები, რომლებიც გამოიყენება **CCNA**-ს პრაქტიკული სამუშაოებისთვის, არის **Cisco 1941** ინტეგრირებული სერვისების მარშრუტიზატორები **(ISRs) Cisco IOS Release 15.2(4)M3 (universalk9 image)** ვერსიასთან ერთად. შესაძლოა გამოყენებულ იქნას სხვა მარშრუტიზატორები და **Cisco IOS** ვერსიებიც. მოდელისა და **Cisco IOS** ვერსიის მიხედვით ხელმისაწვდომი ბრძანებები და მიღებული შედეგები

შეიძლება იყოს განსხვავებული იმისგან, რაც ნაჩვენებია ამ ლაბორატორიულ სამუშაოში. მიაქციეთ ყურადღება მარშრუტიზატორის ინტერფეისის შემაჯამებელ ცხრილს ამ დავალების ბოლოში, სწორი ინტერფეისის იდენტიფიკატორებისათვის.

შენიშვნა: დარწმუნდით, რომ მარშრუტიზატორები და კომუტატორები წაშლილია და არ აქვთ საწყისი კონფიგურაციები. თუ არ ხართ დარწმუნებული დაუკავშირდით თქვენს ინსტრუქტორს.

მოთხოვნილი რესურსები

- 3 მარშრუტიზატორი (Cisco 1941 მოდელი Cisco Release 15.2(4)M3 უნივერსალი იმიჯით ან მსგავსი)
- 3 პერსონალური კომპიუტერი (Windows 7, Windows Vista ან XP ოპერაციული სისტემით, ტერმინალის ემულაციის პროგრამასთან ერთად, როგორიცაა Tera Term)
- კონსოლის კაბელები კონსოლის პორტებიდან Cisco IOS მოწყობილობების კონფიგურაციისთვის
- Ethernet და სერიალური კაბელები, ისეთი როგორიც ნაჩვენებია ტოპოლოგიაზე

ნაწილი №1: ქსელის აწყობა და მოწყობილობის კონფიგურაციების ჩატვირთვა

პირველ ნაწილში თქვენ უნდა მომართოთ ქსელის ტოპოლოგია და დააკონფიგუროთ ბაზისური პარამეტრები პერსონალურ კომპიუტერებსა და მარშრუტიზატორებზე.

პირველი ეტაპი: ქსელის კაბელებით აწყობა ტოპოლოგიის შესაბამისად.

მეორე ეტაპი: PC ჰოსტების კონფიგურაცია.

მესამე ეტაპი: მარშრუტიზატორის კონფიგურაციების ჩატვირთვა

ჩატვირთეთ ქვემოთ მოცემული კონფიგურაციები შესაბამის მარშრუტიზატორში. ყველა მარშრუტიზატორს აქვს ერთიდაიგივე პაროლი. პრივილეგირებული რეჟიმის პაროლი არის **cisco**. კონსოლისა და vty ხაზებისათვის პაროლი არის **class**.

R1 მარშრუტიზატორის კონფიგურაცია:

```
 conf t
 service password-encryption
 no ip domain lookup
 hostname R1
 enable secret class
 line con 0
  logging synchronous
 password cisco
 login
line vty 0
 password cisco
 login
banner motd @Unauthorized Access is Prohibited!@
ipv6 unicast-routing
ipv6 router ospf 1
 router-id 1.1.1.1
 passive-interface g0/0
interface g0/0
 ip address 192.168.1.1 255.255.255.0
 ipv6 address 2001:db8:acad:a::1/64
 ipv6 address fe80::1 link-local
interface s0/0/0
 clock rate 128000
 ip address 192.168.12.1 255.255.255.0
 ipv6 address 2001:db8:acad:12::1/64
 ipv6 address fe80::1 link-local
 ipv6 ospf 1 area 0
 no shutdown
interface s0/0/1
 ip address 192.168.13.1 255.255.255.0
 ipv6 address 2001:db8:acad:13::1/64
 ipv6 address fe80::1 link-local
 ipv6 ospf 1 area 0
 no shutdown
router ospf 1
 network 192.168.1.0 0.0.0.255 area 0
 network 129.168.12.0 0.0.0.3 area 0
 network 192.168.13.0 0.0.0.3 area 0
 passive-interface g0/0
end
```

R2 მარშრუტიზატორის კონფიგურაცია:

```
 conf t
 service password-encryption
 no ip domain lookup
 hostname R2
 enable secret class
 line con 0
  logging synchronous
  password cisco
  login
 line vty 0
  password cisco
 login
banner motd @Unauthorized Access is Prohibited!@
ipv6 unicast-routing
ipv6 router ospf 1
 router-id 2.2.2.2
interface g0/0
ip address 192.168.2.1 255.255.255.0
 ipv6 address 2001:db8:acad:B::2/64
 ipv6 address fe80::1 link-local
 no shutdown
interface s0/0/0
 ip address 192.168.12.2 255.255.255.252
 ipv6 address 2001:db8:acad:12::2/64
 ipv6 address fe80::2 link-local
 ipv6 ospf 1 area 0
 no shutdown
interface s0/0/1
 clock rate 128000
 ipv6 address 2001:db8:acad:23::2/64
 ipv6 address fe80::2 link-local
 no shutdown
router ospf 1
 network 192.168.2.0 0.0.0.255 area 0
 network 192.168.12.0 0.0.0.3 area 0
 network 192.168.23.0 0.0.0.3 area 0
end
```

R3 მარშრუტიზატორის კონფიგურაცია:

```
conf t
service password-encryption
no ip domain lookup
enable secret class
hostname R3
line con 0
 logging synchronous
 password cisco
 login
line vty 0
 password cisco
 login
banner motd @Unauthorized Access is Prohibited!@
interface g0/0
 ipv6 address 2001:db8:acad:c::3/64
 ipv6 address fe80::3 link-local
interface s0/0/0
 clock rate 128000
 ip address 192.168.13.1 255.255.255.252
 ipv6 address 2001:db8:acad:13::3/64
 ipv6 address fe80::3 link-local
 no shutdown
interface s0/0/1
 ip address 192.168.23.2 255.255.255.252
 ipv6 address 2001:db8:acad:23::3/64
 ipv6 address fe80::3 link-local
router ospf 1
 network 192.168.3.0 0.0.0.255 area 0
 passive-interface g0/0
end
```

ნაწილი №2: მესამე დონის კავშირის პრობლემის აღმოფხვრა

მეორე ნაწილში თქვენ უნდა დარწმუნდეთ რომ მესამე დონის კავშირი მომართულია ყველა ინტერფეისზე. თქვენ ასევე უნდა შეამოწმოთ Ipv4 და IPv6 კავშირები ყველა მოწყობილობის ინტერფეისზე.

პირველი ეტაპი: დარწმუნდით რომ მისამართების ცხრილში ჩამოთვლილი ინტერფეისები აქტიურია და კონფიგურირებულია სწორი IP მისამართის ინფორმაციით.

ა. გაუშვით **show ip interface brief** ბრძანება ყველა მარშრუტიზატორზე, იმის შესამოწმებლად რომ ინტერფეისები იმყოფებიან up/up მდგომარეობაში. ჩაიწერეთ მიღებული შედეგები.

_____

_____

_____

ბ. გაუშვით **show run interface** ბრძანება მარშრუტიზატორის ყველა ინტერფეისზე მინიჭებული IP მისამართების შესამოწმებლად. შეუდარეთ ინტერფეისის IP მისამართები მისამართების ცხრილს და შეამოწმეთ მინიჭებული ქვექსელის ნიღბები. IPv6-სთვის დარწმუნდით რომ link-local მისამართი არის მინიჭებული. ჩაიწერეთ მიღებული შედეგები.

_____

_____

_____

გ. მოაგვარეთ ყველა დაფიქსირებული შეცდომა. ჩაიწერეთ ყველა ის ბრძანება, რომელსაც გამოიყენებთ პრობლემის გამოსწორებისას.

_____

_____

დ. **Ping** ბრძანების გამოყენებით შეამოწმეთ აქვს თუ არა თითოეულ მარშრუტიზატორს სერიალ ინტერფეისებით კავშირი მეზობელ მარშრუტიზატორებზე. დარწმუნდით რომ პერსონალურ კომპიუტერებს შეუძლიათ თავიანთი ნაგულისხმევი გასასვლელების დაპინგვა. თუ პრობლემები ისევ არსებობს, გააგრძელეთ მესამე დონის პრობლემების მოძიებისა და აღმოფხვრის პროცესი.

ნაწილი №3: OSPFv2-ის პრობლემის მოძიება და აღმოფხვრა

მესამე ნაწილში თქვენ უნდა მოიძიოთ და აღმოფხვრათ OSPFv2 პრობლემები და მოახდინოთ აუცილებელი ცვლილებები, რომელიც საჭიროა OSPFv2 მარშრუტებისა და IPv4 საბოლოო წერტილების (end-to-end) კავშირის დასამყარებლად.

შენიშვნა: ლოკალური ქსელის G0/0 ინტერფეისები შეიძლება არ იყოს განთავსებული OSPFv2 მარშრუტიზაციის ინფორმაციაში, მაგრამ ამ ქსელების მარშრუტები უნდა იყოს მარშრუტიზაციის ცხრილში.

პირველი ეტაპი: IPv4 საბოლოო წერტილებს შორის კავშირის შემოწმება.

თითოეული კომპიუტერიდან დაპინგეთ სხვა ჰოსტ კომპიუტერები, საბოლოო წერტილებს შორის კავშირის შესამოწმებლად.

შენიშვნა: ჰოსტებს შორის პინგის გასაშვებად შეიძლება აუცილებელი გახდეს პერსონალური ფაიერვოლის გათიშვა.

ა. PC-A-დან დაპინგეთ PC-B. წარმატებით დასრულდა პინგი?_____

ბ. PC-A-დან დაპინგეთ PC-C. წარმატებით დასრულდა პინგი? _____

გ. PC-B-დან დაპინგეთ PC-C. წარმატებით დასრულდა პინგი? _____

მეორე ეტაპი: დარწმუნდით რომ ყველა ინტერფეისი არის მინიჭებული OSPFv2 area 0-თან R1 მარშრუტიზატორზე.

ა. გაუშვით **show ip protocols** ბრძანება იმის დასადგენად ჩართულია თუ არა OSPF და რომ ყველა ქსელი განთავსებულია area 0-ში. შეამოწმეთ სწორადაა თუ არა მომართული მარშრუტიზატორის ID. ჩაიწერეთ მიღებული შედეგები.

_____

_____

_____

ბ. **Show ip protocols** ბრანების შედეგების საფუძველზე განახორციელეთ კონფიგურაციის აუცილებელი ცვლილებები R1 მარშრუტიზატორზე. ჩაიწერეთ ყველა ის ბრძანება, რომელსაც გამოიყენებთ პრობლემის გამოსწორებისას.

_____

_____

_____

გ. აუცილებლობის შემთხვევაში გაუშვით **clear ip ospf process** ბრძანება.

დ. ხელახლა გაუშვით **show ip protocols** ბრძანება იმის დასადგენად ჰქონდა თუ არა სასურველი ეფექტი თქვენს მიერ განხორციელებულ ცვლილებებს.

ე. გაუშვით **show ip ospf interface brief** ბრძანება რათა დავრწმუნდეთ რომ ყველა ინტერფეისი OSPF ქსელების სიაში არის მინიჭებული area 0-თან.

ვ. გაუშვით **show ip ospf interface g0/0** ბრძანება რათა დავრწმუნდეთ რომ G0/0 არის პასიური ინტერფეისი.

შენიშვნა: მოცემული ინფორმაცია ასევე არის **show ip protocols** ბრძანებაში.

ზ. მოაგვარეთ R1 მარშრუტიზატორზე დაფიქსირებული ნებისმიერი პრობლემა. ჩამოწერეთ R1 მარშრუტიზატორზე განხორციელებული ნებისმიერი დამატებითი ცვლილება. თუ მოწყობილობაზე არ იქნა ნაპოვნი არც ერთი პრობლემა, მაშინ ვპასუხობთ „პრობლემები არ იქნა ნაპოვნი“.

_____

_____

მესამე ეტაპი: დარწმუნდით რომ ყველა ინტერფეისი არის მინიჭებული OSPFv2 area 0-თან R2 მარშრუტიზატორზე.

ა. გაუშვით **show ip protocols** ბრძანება იმის დასადგენად ჩართულია თუ არა OSPF და რომ ყველა ქსელი განთავსებულია area 0-ში. შეამოწმეთ სწორადაა თუ არა მომართული მარშრუტიზატორის ID. ჩაიწერეთ მიღებული შედეგები.

_____

_____

_____

ბ. **Show ip protocols** ბრანების შედეგების საფუძველზე განახორციელეთ კონფიგურაციის აუცილებელი ცვლილებები R2 მარშრუტიზატორზე. ჩაიწერეთ ყველა ის ბრძანება, რომელსაც გამოიყენებთ პრობლემის გამოსწორებისას.

_____

_____

_____

გ. აუცილებლობის შემთხვევაში გაუშვით **clear ip ospf process** ბრძანება.

დ. ხელახლა გაუშვით **show ip protocols** ბრძანება იმის დასადგენად ჰქონდა თუ არა სასურველი ეფექტი თქვენს მიერ განხორციელებულ ცვლილებებს.

ე. გაუშვით **show ip ospf interface brief** ბრძანება რათა დავრწმუნდეთ რომ ყველა ინტერფეისი OSPF ქსელების სიაში არის მინიჭებული area 0-თან.

ვ. გაუშვით **show ip ospf interface g0/0** ბრძანება რათა დავრწმუნდეთ რომ G0/0 არის პასიური ინტერფეისი.

შენიშვნა: მოცემული ინფორმაცია ასევე არის **show ip protocols** ბრძანებაში.

ზ. მოაგვარეთ R2 მარშრუტიზატორზე დაფიქსირებული ნებისმიერი პრობლემა. ჩამოწერეთ R2 მარშრუტიზატორზე განხორციელებული ნებისმიერი დამატებითი ცვლილება. თუ მოწყობილობაზე არ იქნა ნაპოვნი არც ერთი პრობლემა, მაშინ ვპასუხობთ „პრობლემები არ იქნა ნაპოვნი“.

_____

_____

მეოთხე ეტაპი: დარწმუნდით რომ ყველა ინტერფეისი არის OSPFv2 area 0-თან R3 მარშრუტიზატორზე.

ა. გაუშვით **show ip protocols** ბრძანება იმის დასადგენად ჩართულია თუ არა OSPF და რომ ყველა ქსელი განთავსებულია area 0-ში. შეამოწმეთ სწორადაა თუ არა მომართული მარშრუტიზატორის ID. ჩაიწერეთ მიღებული შედეგები.

_____

_____

_____

ბ. **Show ip protocols** ბრანების შედეგების საფუძველზე განახორციელეთ კონფიგურაციის აუცილებელი ცვლილებები R3 მარშრუტიზატორზე. ჩაიწერეთ ყველა ის ბრძანება, რომელსაც გამოიყენებთ პრობლემის გამოსწორებისას.

_____

_____

_____

გ. აუცილებლობის შემთხვევაში გაუშვით **clear ip ospf process** ბრძანება.

დ. ხელახლა გაუშვით **show ip protocols** ბრძანება იმის დასადგენად ჰქონდა თუ არა სასურველი ეფექტი თქვენს მიერ განხორციელებულ ცვლილებებს.

ე. გაუშვით **show ip ospf interface brief** ბრძანება რათა დავრწმუნდეთ რომ ყველა ინტერფეისი OSPF ქსელების სიაში არის მინიჭებული area 0-თან.

ვ. გაუშვით **show ip ospf interface g0/0** ბრძანება რათა დავრწმუნდეთ რომ G0/0 არის პასიური ინტერფეისი.

შენიშვნა: მოცემული ინფორმაცია ასევე არის **show ip protocols** ბრძანებაში.

ზ. მოაგვარეთ R3 მარშრუტიზატორზე დაფიქსირებული ნებისმიერი პრობლემა. ჩამოწერეთ R3 მარშრუტიზატორზე განხორციელებული ნებისმიერი დამატებითი ცვლილება. თუ მოწყობილობაზე არ იქნა ნაპოვნი არც ერთი პრობლემა, მაშინ ვპასუხობთ „პრობლემები არ იქნა ნაპოვნი“.

_____

_____

მეხუთე ეტაპი: მეზობელი OSPF ინფორმაციის დადგენა.

ა. გაუშვით **show ip ospf neighbor** ბრძანება ყველა მარშრუტიზატორზე, მეზობელი OSPF ინფორმაციის დასადგენად.

მეექვსე ეტაპი: OSPFv2 მარშრუტიზაციის ინფორმაციის შემოწმება.

ა. გაუშვით **show ip route ospf** ბრძანება იმის შესამოწმებლად, რომ თითოეულ მარშრუტიზატორს აქვს ყველა არამოსაზღვრე ქსელების OSPFv2 მარშრუტები.

ხელმისაწვდომია ყველა OSPFv2 მარშრუტი? _____

თუ ნებისმიერი OSPFv2 მარშრუტი არის მიუწვდომელი, რა არის დაკარგული?

_____

ბ. თუ მიუწვდომელია რომელიმე მარშრუტიზაციის ინფორმაცია, მოაგვარეთ ეს პრობლემები.

მეშვიდე ეტაპი: IPv4 საბოლოო მოწყობილობების კავშირის შემოწმება

თითოეული პერსონალური კომპიუტერიდან დარწმუნდით რომ არსებობს IPv4 საბოლოო მოწყობილობებს შორის კავშირი. კომპიუტერებს უნდა შეეძლოთ ტოპოლოგიის სხვა ჰოსტების დაპინგვა. თუ არ არსებობს IPv4 საბოლოო მოწყობილობებს შორის კავშირი, მაშინ გააგრძელეთ პრობლემის მოძიებისა და აღმოფხვრის პროცესი და მოაგვარეთ ნებისმიერი დარჩენილი პრობლემა.

შენიშვნა: კომპიუტერებს შორის პინგის გასაშვებად შესაძლოა საჭირო გახდეს პერსონალური ფაირვოლის გათიშვა.

ნაწილი №4: OSPFv3-ის პრობლემის მოძიება და აღმოფხვრა

მეოთხე ნაწილში თქვენ უნდა მოიძიოთ და აღმოფხვრათ OSPFv3 პრობლემები და მოახდინოთ აუცილებელი ცვლილებები, რომელიც საჭიროა OSPFv3 მარშრუტებისა და IPv6 საბოლოო წერტილების (end-to-end) კავშირის დასამყარებლად.

შენიშვნა: ლოკალური ქსელის G0/0 ინტერფეისები შეიძლება არ იყოს განთავსებული OSPFv3 მარშრუტიზაციის ინფორმაციაში, მაგრამ ამ ქსელების მარშრუტები უნდა იყოს მარშრუტიზაციის ცხრილში.

პირველი ეტაპი: IPv6 საბოლოო წერტილებს შორის კავშირის შემოწმება.

თითოეული კომპიუტერიდან დაპინგეთ ტოპოლოგიის სხვა ჰოსტ კომპიუტერების IPv6 მისამართები, IPv6 საბოლოო წერტილებს შორის კავშირის შესამოწმებლად.

შენიშვნა: ჰოსტებს შორის პინგის გასაშვებად შეიძლება აუცილებელი გახდეს პერსონალური ფაიერვოლის გათიშვა.

მეორე ეტაპი: დარწმუნდით რომ IPv6 ერთმისამართიანი (unicast) მარშრუტიზაცია ჩართულია ყველა მარშრუტიზატორზე.

ა. რომ შევამოწმოთ IPv6 მარშრუტიზაცია ჩართულია თუ არა მარშრუტიზატორზე, ყველაზე მარტივი გზაა **show run | section ipv6 unicast** ბრძანების გამოყენება. **Show run**

ბრძანებაში სექციის მილის (|) დამატებით, **IPv6 unicast-routing** ბრძანება აჩვენებს ჩართულია თუ არა IPv6 მარშრუტიზაცია.

შენიშვნა: show run ბრძანება შეიძლება გაშვებულ იქნას ყოველგვარი მილის (|) გარეშე და შემდეგ შესაძლებელია შესრულდეს **ipv6 unicats-routing** ბრძანების ხელით ძებნა.

გაუშვით ბრძანება თითოეულ მარშრუტიზატორზე. ჩაიწერეთ მიღებული შედეგები.

_____

_____

_____

ბ. თუ IPv6 ერთმისამართიანი (unicast) მარშრუტიზაცია არ არის ჩართული ერთ ან რამდენიმე მარშრუტიზატორზე, ჩართეთ ისინი. ჩაიწერეთ ყველა ბრძანება რომელიც გამოყენებულ იქნა პრობლემების მოგვარების დროს.

_____

_____

მესამე ეტაპი: დარწმუნდით რომ ყველა ინტერფეისი არის მინიჭებული OSPFv3 area 0-თან R1 მარშრუტიზატორზე.

ა. გაუშვით **show ipv6 protocols** ბრძანება იმის დასადგენად სწორადაა თუ არა მომართული მარშრუტიზატორის ID. ასევე დარწმუნდით, რომ სავარაუდო ინტერფეისები ჩანს area 0-ის ქვეშ.

შენიშვნა: თუ ამ ბრძანების გაშვებამ არ მოგვცა შედეგი, ე.ი OSPFv3 პროცესი არ ყოფილა დაკონფიგურებული.

ჩაიწერეთ მიღებული შედეგები.

_____

_____

_____

ბ. განახორციელეთ კონფიგურაციის აუცილებელი ცვლილებები R1 მარშრუტიზატორზე. ჩაიწერეთ ყველა ის ბრძანება, რომელსაც გამოიყენებთ პრობლემის გამოსწორებისას.

_____

_____

_____

გ. აუცილებლობის შემთხვევაში გაუშვით **clear ipv6 ospf process** ბრძანება.

დ. ხელახლა გაუშვით **show ipv6 protocols** ბრძანება იმის დასადგენად ჰქონდა თუ არა სასურველი ეფექტი თქვენს მიერ განხორციელებულ ცვლილებებს.

ე. გაუშვით **show ipv6 ospf interface brief** ბრძანება რათა დავრწმუნდეთ რომ ყველა ინტერფეისი OSPF ქსელების სიაში არის მინიჭებული area 0-თან.

ვ. გაუშვით **show ipv6 ospf interface g0/0** ბრძანება რათა დავრწმუნდეთ რომ მოცემული ინტერფეისი მომართულია ისე რომ არ მოახდინოს OSPFv3 მარშრუტების აფიშირება.

ზ. მოაგვარეთ R1 მარშრუტიზატორზე დაფიქსირებული ნებისმიერი პრობლემა. ჩამოწერეთ R1 მარშრუტიზატორზე განხორციელებული ნებისმიერი დამატებითი ცვლილება. თუ მოწყობილობაზე არ იქნა ნაპოვნი არც ერთი პრობლემა, მაშინ ვპასუხობთ „პრობლემები არ იქნა ნაპოვნი“.

_____

_____

მეოთხე ეტაპი: დარწმუნდით რომ ყველა ინტერფეისი არის მინიჭებული OSPFv3 area 0-თან R2 მარშრუტიზატორზე.

ა. გაუშვით **show ipv6 protocols** ბრძანება იმის დასადგენად სწორადაა თუ არა მომართული მარშრუტიზატორის ID. ასევე დარწმუნდით, რომ სავარაუდო ინტერფეისები ჩანს area 0-ის ქვეშ.

შენიშვნა: თუ ამ ბრძანების გაშვებამ არ მოგვცა შედეგი, ე.ი OSPFv3 პროცესი არ ყოფილა დაკონფიგურებული.

ჩაიწერეთ მიღებული შედეგები.

_____

_____

_____

ბ. განახორციელეთ კონფიგურაციის აუცილებელი ცვლილებები R2 მარშრუტიზატორზე. ჩაიწერეთ ყველა ის ბრძანება, რომელსაც გამოიყენებთ პრობლემის გამოსწორებისას.

_____

_____

_____

გ. აუცილებლობის შემთხვევაში გაუშვით **clear ipv6 ospf process** ბრძანება.

დ. ხელახლა გაუშვით **show ipv6 protocols** ბრძანება იმის დასადგენად ჰქონდა თუ არა სასურველი ეფექტი თქვენს მიერ განხორციელებულ ცვლილებებს.

ე. გაუშვით **show ipv6 ospf interface brief** ბრძანება რათა დავრწმუნდეთ რომ ყველა ინტერფეისი OSPF ქსელების სიაში არის მინიჭებული area 0-თან.

ვ. გაუშვით **show ipv6 ospf interface g0/0** ბრძანება რათა დავრწმუნდეთ რომ მოცემული ინტერფეისი მომართულია ისე რომ არ მოახდინოს OSPFv3 მარშრუტების აფიშირება.

ზ. ჩამოწერეთ R2 მარშრუტიზატორზე განხორციელებული ნებისმიერი დამატებითი ცვლილება. თუ მოწყობილობაზე არ იქნა ნაპოვნი არც ერთი პრობლემა, მაშინ ვპასუხობთ „პრობლემები არ იქნა ნაპოვნი“.

_____

_____

მეხუთე ეტაპი: დარწმუნდით რომ ყველა ინტერფეისი არის მინიჭებული OSPFv3 area 0-თან R3 მარშრუტიზატორზე.

ა. გაუშვით **show ipv6 protocols** ბრძანება იმის დასადგენად სწორადაა თუ არა მომართული მარშრუტიზატორის ID. ასევე დარწმუნდით, რომ სავარაუდო ინტერფეისები ჩანს area 0-ის ქვეშ.

შენიშვნა: თუ ამ ბრძანების გაშვებამ არ მოგვცა შედეგი, ე.ი OSPFv3 პროცესი არ ყოფილა დაკონფიგურებული.

ჩაიწერეთ მიღებული შედეგები.

_____

_____

_____

ბ. განახორციელეთ კონფიგურაციის აუცილებელი ცვლილებები R3 მარშრუტიზატორზე. ჩაიწერეთ ყველა ის ბრძანება, რომელსაც გამოიყენებთ პრობლემის გამოსწორებისას.

_____

_____

_____

გ. აუცილებლობის შემთხვევაში გაუშვით **clear ipv6 ospf process** ბრძანება.

დ. ხელახლა გაუშვით **show ipv6 protocols** ბრძანება იმის დასადგენად ჰქონდა თუ არა სასურველი ეფექტი თქვენს მიერ განხორციელებულ ცვლილებებს.

ე. გაუშვით **show ipv6 ospf interface brief** ბრძანება რათა დავრწმუნდეთ რომ ყველა ინტერფეისი OSPF ქსელების სიაში არის მინიჭებული area 0-თან.

ვ. გაუშვით **show ipv6 ospf interface g0/0** ბრძანება რათა დავრწმუნდეთ რომ მოცემული ინტერფეისი მომართულია ისე რომ არ მოახდინოს OSPFv3 მარშრუტების აფიშირება.

ზ. მოაგვარეთ R3 მარშრუტიზატორზე დაფიქსირებული ნებისმიერი პრობლემა. ჩამოწერეთ R3 მარშრუტიზატორზე განხორციელებული ნებისმიერი დამატებითი ცვლილება. თუ მოწყობილობაზე არ იქნა ნაპოვნი არც ერთი პრობლემა, მაშინ ვპასუხობთ „პრობლემები არ იქნა ნაპოვნი“.

_____

_____

მეექვსე ეტაპი: დარწმუნდით რომ ყველა მარშრუტიზატორს აქვს მეზობლის მოსაზღვრეობის სწორი ინფორმაცია.

ა. გაუშვით **show ipv6 ospf neighbor** ბრძანება იმის შესამოწმებლად, რომ მოსაზღვრეობა არის ჩამოყალიბებული მეზობელ მარშრუტიზატორებს შორის.

ბ. მოაგვარეთ ნებისმიერი არსებული OSPFv3 მესაზღვრეობის პრობლემა.

მეშვიდე ეტაპი: OSPFv3 მარშრუტიზაციის ინფორმაციის შემოწმება.

ა. გაუშვით **show ipv6 route ospf** ბრძანება იმის შესამოწმებლად, რომ თითოეულ მარშრუტიზატორს აქვს ყველა არამოსაზღვრე ქსელების OSPFv3 მარშრუტები.

ხელმისაწვდომია ყველა OSPFv3 მარშრუტი? _____

თუ ნებისმიერი OSPFv3 მარშრუტი არის მიუწვდომელი, რა არის დაკარგული?

_____

_____

ბ. მოაგვარეთ ნებისმიერი არსებული მარშრუტიზაციის პრობლემა.

მერვე ეტაპი: IPv6 საბოლოო მოწყობილობების კავშირის შემოწმება

თითოეული პერსონალური კომპიუტერიდან დარწმუნდით რომ არსებობს IPv6 საბოლოო მოწყობილობებს შორის კავშირი. კომპიუტერებს უნდა შეეძლოთ ტოპოლოგიის სხვა ჰოსტების დაპინგვა. თუ არ არსებობს IPv6 საბოლოო მოწყობილობებს

შორის კავშირი, მაშინ გააგრძელეთ პრობლემის მოძიებისა და აღმოფხვრის პროცესი და მოაგვარეთ ნებისმიერი დარჩენილი პრობლემა.

შენიშვნა: კომპიუტერებს შორის პინგის გასაშვებად შესაძლოა საჭირო გახდეს პერსონალური ფაიერვოლის გათიშვა.

ასახვა

რატომ ასრულებთ OSPFv2-სა და OSPFv3-ის პრობლემების მოძიებისა და აღმოფხვრის პროცესს ცალ-ცალკე?

_____

_____

მარშრუტიზატორის ინტერფეისის შემაჯამებელი ცხრილი:

მარშრუტიზატორის ინტერფეისის შეჯამება				
მარშრუტიზატორის მოდელი	Ethernet interface №1	Ethernet interface №2	Serial interface №1	Serial interface №2
1800	FastEthernet 0/0 (F0/0)	FastEthernet 0/1 (F0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
1900	GigabitEthernet 0/0 (G0/0)	GigabitEthernet 0/1 (G0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
2801	FastEthernet 0/0 (F0/0)	FastEthernet 0/1 (F0/1)	Serial 0/1/0 (S0/0/0)	Serial 0/1/1 (S0/0/1)
2811	FastEthernet 0/0 (F0/0)	FastEthernet 0/1 (F0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
2900	GigabitEthernet 0/0 (G0/0)	GigabitEthernet 0/1 (G0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
შენიშვნა: თუ გინდათ მოძებნოთ თუ როგორაა მარშრუტიზატორი კონფიგურირებული, დაათვალიერეთ ინტერფეისები რათა მოახდინოთ მარშრუტიზატორის იდენტიფიკაცია და დაადგინოთ რამდენი ინტერფეისი აქვს მას. აქ არ არის იმის საშუალება, რომ				

ჩამოიწეროს თითოეული მარშრუტიზატორის კონფიგურაციების კომბინაციები. მოცემული ცხრილი მოიცავს მოწყობილობის **Ethernet** და **Serial** ინტერფეისების შესაძლო კომბინაციების იდენტიფიკატორებს. ეს ცხრილი არ მოიცავს ნებისმიერი სხვა ტიპის ინტერფეისს, რადგან კონკრეტული მარშრუტიზატორი შეიძლება შეიცავდეს მხოლოდ ერთს. ამის მაგალითი შეიძლება იყოს **ISDN BRI** ინტერფეისი. ფრჩხილებში მოცემული ცვლადი არის ლეგალური აბრევიატურა, რომელიც შეიძლება გამოყენებულ იქნას **Cisco IOS** ბრძანებებში, ინტერფეისის წარმოსადგენად.

6.4. მონიტორინგის, ინციდენტების და სხვადასხვა სერვისების დიაგნოსტიკა პრობლემების აღმოფხვრით

6.4.1. *SNMP პროტოკოლის კონფიგურირება.*

SNMP განვითარდა იმისათვის, რომ ადმინისტრატორებს ჰქონდეთ საშუალება მართონ სერვერები, სამუშაო სადგურები (Workstations), მარშრუტიზატორები, კომუტატორები და უსაფრთხოების ტექნიკა, IP ქსელში. ის საშუალებას აძლევს ქსელის ადმინისტრატორებს მართონ ქსელის წარმადობა, იპოვონ და აღმოფხვრან ქსელური პრობლემები და დაგეგმონ ქსელის გაზრდა.

SNMP არის გამოყენებითი დონის პროტოკოლი, რომელიც უზრუნველყოფს შეტყობინების ფორმატს, მენეჯერებსა და აგენტებს შორის კომუნიკაციისთვის. SNMP სისტემა შედგება სამი ელემენტისაგან:

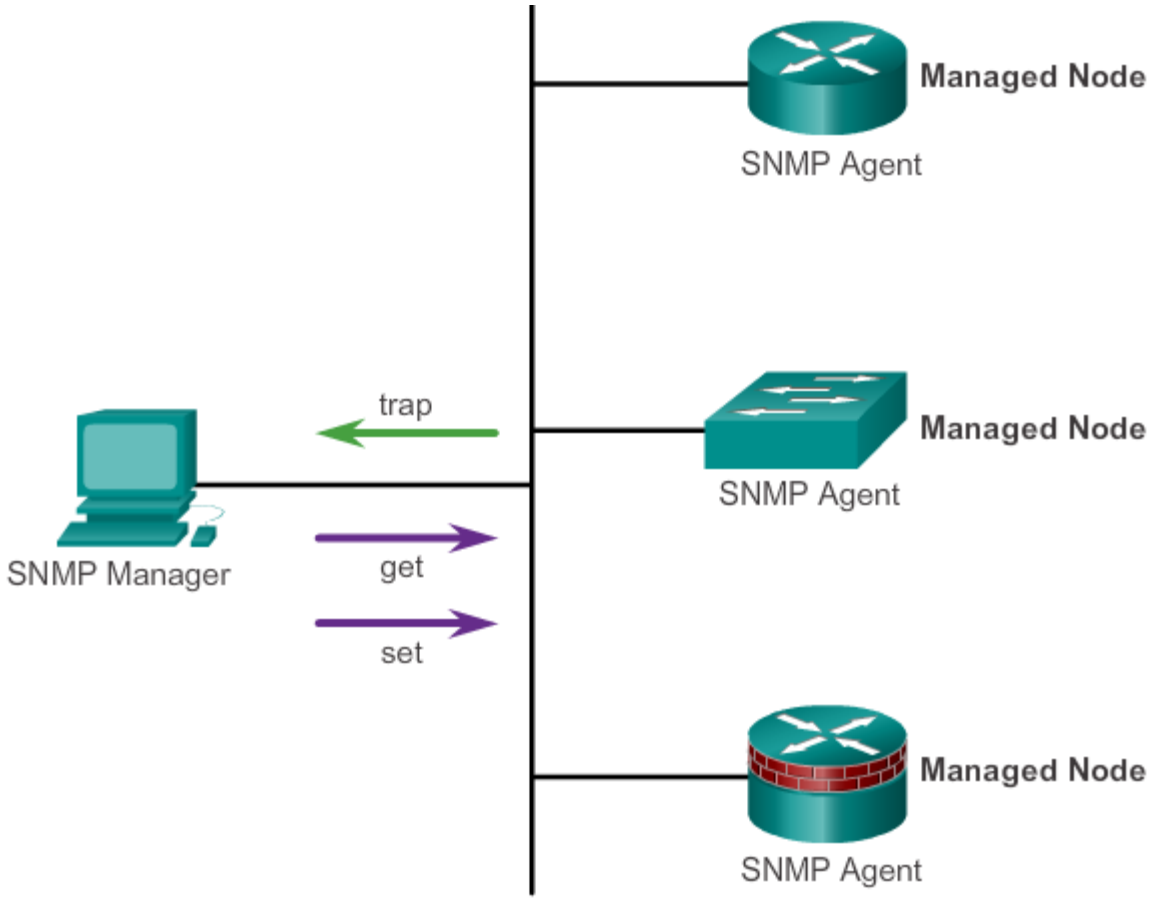
- SNMP მენეჯერი
- SNMP აგენტი (მართვადი კვანძი)
- მართვის საინფორმაციო ბაზა (MIB)

ქსელურ მოწყობილობაზე SNMP-ს კონფიგურაციისთვის, პირველ რიგში აუცილებელია განისაზღვროს ურთიერთობა მენეჯერსა და აგენტს შორის.

SNMP მენეჯერი არის ქსელის მართვის სისტემის (NMS) ნაწილი. SNMP მენეჯერი უშვებს SNMP-ის მართვის პროგრამულ უზრუნველყოფას. როგორც 5.1 სურათზეა მოცემული, SNMP მენეჯერს შეუძლია შეაგროვოს ინფორმაცია SNMP აგენტებიდან „get (მიღება)“ მოქმედების გამოყენებით და შეუძლი კონფიგურაციის შეცვლა აგენტზე „Set (გაშვება)“ მოქმედების გამოყენებით. დამატებით, SNMP აგენტებს შეუძლიათ ინფორმაციის გადაგზავნა პირდაპირ ქსელის მართვის სისტემასთან (NMS), „traps (მახეები)”-ის გამოყენებით.

SNMP აგენტი და მართვის საინფორმაციო ბაზა (MIB) მიეკუთვნება ქსელური მოწყობილობების კლიენტებს. ის ქსელური მოწყობილობები, რომელთა მართვაც შეიძლება,

კომუტატორების, მარშრუტიზატორების, სერვერების, ფაიერვოლების და სამუშაო სადგურების ჩათვლით, აღჭურვილია SNMP აგენტი პროგრამული უზრუნველყოფის მოდულით. მართვის საინფორმაციო ბაზა (MIB) ინახავს მონაცემებს მოწყობილობის მუშაობის შესახებ და განკუთვნილია იმისთვის რომ იყოს ხელმისაწვდომი ავტორიზებული დაშორებული მომხმარებლებისთვის. SNMP აგენტი პასუხისმგებელია ლოკალური მართვის საინფორმაციო ბაზის წვდომის უზრუნველყოფაზე ობიექტებთან, რომლებიც ასახავენ რესურსებსა და საქმიანობას.



სურ. 6.4.1 მარტივი ქსელის მართვის პროტოკოლი (SNMP)

SNMP განსაზღვრავს თუ როგორ იცვლება სამართავი ინფორმაცია ქსელის მართვის აპლიკაციებსა და მართვად აგენტებს შორის. SNMP იყენებს UDP პროტოკოლს, პორტის ნომრით 162, რათა მიიღოს და გააგზავნოს მართვის ინფორმაცია.

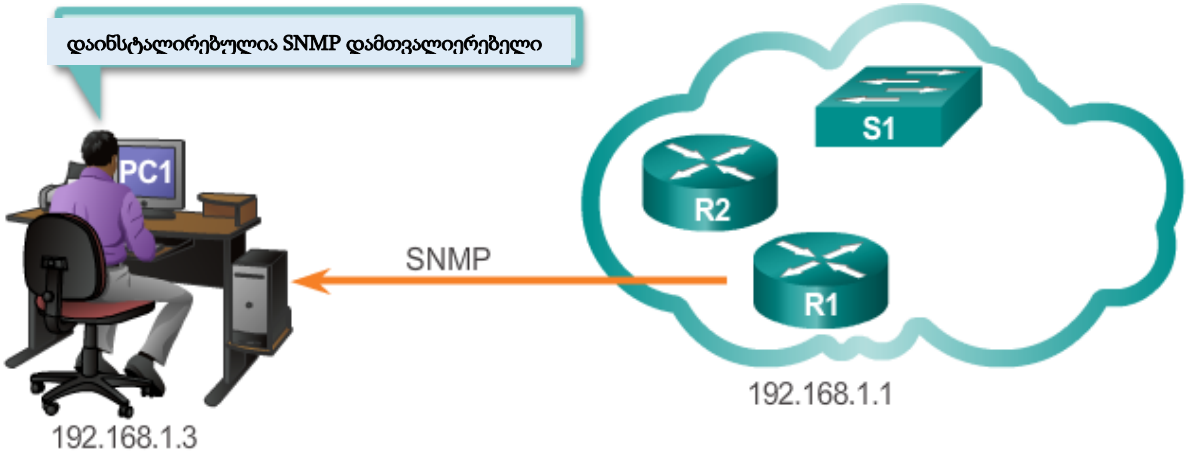
6.4.2. *SNMP კონფიგურაციის ეტაპები*

ქსელის ადმინისტრატორს შეუძლია SNMPv2-ის კონფიგურაცია ქსელური მოწყობილობებიდან ქსელის ინფორმაციის მისაღებად. როგორც 6.3.2 სურათზეა ნაჩვენები, SNMP-ს კონფიგურაციის ყველა ბაზისური ეტაპი არის საერთო კონფიგურაციის რეჟიმში.

პირველი ეტაპი. (აუცილებელი) დააკონფიგურეთ მწკრივების ერთობა (Community String) და დაშვების დონე (მხოლოდ ნახვა ან ნახვა-ჩაწერა) **snmp-server community** *string* **ro | rw** ბრძანებით.

მეორე ეტაპი. (დამატებითი) მოახდინეთ მოწყობილობის ადგილმდებარეობის დოკუმენტირება **snmp-server** location *text* ბრძანების გამოყენებით.

მესამე ეტაპი. (დამატებითი) მოახდინეთ სისტემური კონტაქტების დოკუმენტირება **snmp-server** contact *text* ბრძანებით.




```
R1(config)# snmp-server community batonaug ro SNMP_ACL
R1(config)# snmp-server location NOC_SNMP_MANAGER
R1(config)# snmp-server contact Wayne World
R1(config)# snmp-server host 192.168.1.3 version 2c batonaug
R1(config)# snmp-server enable traps
R1(config)# ip access-list standard SNMP_ACL
R1(config-std-nacl)# permit 192.168.1.3
```

სურ. 6.4.2. SNMP მენეჯერის კონფიგურაციის მხარაჭერა

მეოთხე ეტაპი. (დამატებითი) აკრძალეთ SNMP-ს წვდომა ქსელის მართვის სისტემის (NMS) ჰოსტებთან (SNMP მენეჯერები), რომლებიც დაშვებულნი არიან ACL-ის მიერ: განსაზღვრეთ ACL და შემდეგ მიუთითეთ ACL **snmp-server community** *string access-list-number-or-name* ბრძანების გამოყენებით. მოცემული ბრძანება გამოიყენება როგორც მწკრივების მისათითებლად, ისე SNMP წვდომის აკრძალვისთვის ACL-ების მეშვეობით. სურვილის შემთხვევაში პირველი და მეოთხე ეტაპი შეიძლება გაერთიანდეს ერთ ეტაპად. Cisco ქსელური მოწყობილობა აერთიანებს ორ ბრძანებას ერთში, თუ ისინი შეტანილია ცალ-ცალკე.

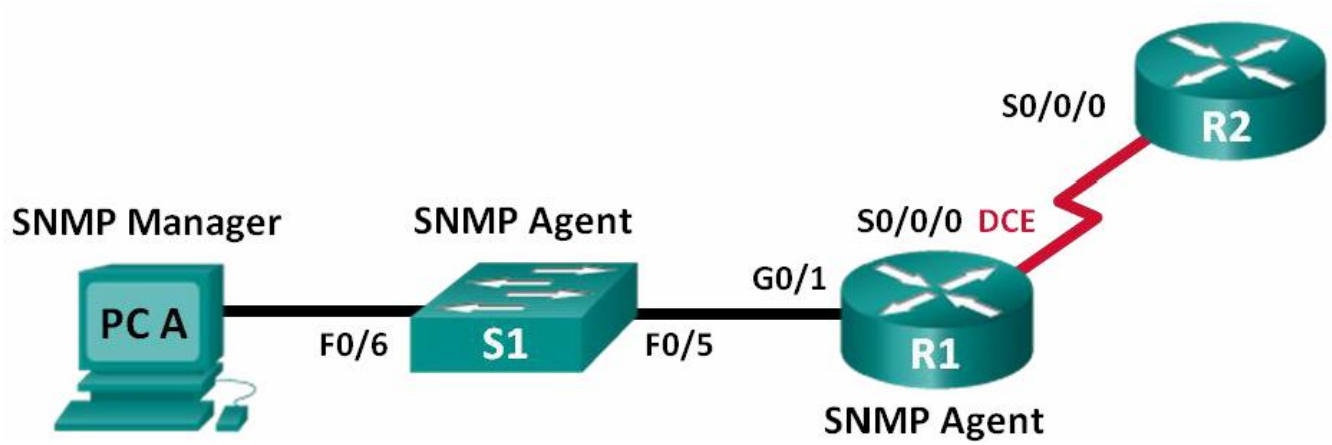
მეხუთე ეტაპი. (დამატებითი) მიუთითეთ SNMP trap ოპერაციების მიმღები **snmp-server host** *host-id* [version {1 | 2c | 3 [auth | noauth | priv]}] *community-string* ბრძანების გამოყენებით. ნაგულისხმევად trap მენეჯერ არ არის მითითებული.

მეექვსე ეტაპი. (დამატებითი) ჩართეთ traps (მახეები) SNMP აგენტზე **snmp-server enable traps** *notification-types* ბრძანებით. თუ მოცემულ ბრძანებაში არცერთი trap შეტყობინების ტიპი არ არის მითითებული, მაშინ ყველა ტიპის trap-ი იქნება გაგზავნილი. ამ ბრძანების განმეორებითი გამოყენება მოითხოვება, მაშინ თუ განსაზღვრული ტიპის trap ქვეჯგუფებია სასურველი.

შენიშვნა: ნაგულისხმევად, SNMP-ს არ აქვს არანაირი trap-ები მომართული. ამ ბრძანების გარეშე, SNMP მენეჯერებს შეუძლიათ ამოირჩიონ ყველა მართებული ინფორმაცია.

ლაბორატორიული სამუშაო - SNMP-ს კონფიგურაცია

ტოპოლოგია:



მისამართების ცხრილი:

მოწყობილობა	ინტერფეისი	IP მისამართი	ქვექსელის ნიღაბი	ნაგულისხმევი გასასვლელი
R1	G0/1	192.168.1.1	255.255.255.0	N/A
	S0/0/0	192.168.2.1	255.255.255.252	N/A
R2	S0/0/0	192.168.2.2	255.255.255.252	N/A
S1	VLAN 1	192.168.1.2	255.255.255.0	N/A
PC-A	NIC	192.168.1.3	255.255.255.0	192.168.1.1

შესასრულებელი დავალებები:

ნაწილი №1: ქსელის აწყობა და მოწყობილობის ბაზისური კონფიგურაცია

ნაწილი №2: SNMP მმართველისა და აგენტის კონფიგურაცია

ნაწილი №3: OID კოდების კონვერტაცია Cisco SNMP Object Navigator-თან

ზოგადი ინფორმაცია / სცენარი

Simple Network Management Protocol (SNMP) არის ქსელის მართვის პროტოკოლი და **IETF** სტანდარტი, რომელიც შეიძლება გამოყენებულ იქნას ქსელში კლიენტების მონიტორინგისა და მართვისათვის. **SNMP** შეიძლება გამოყენებულ იქნას ცვლადების მიღებისა და დაყენებისათვის, რომლებიც დაკავშირებულია ქსელური ჰოსტების მდგომარეობასა და კონფიგურაციაზე, როგორიცაა მარშრუტიზატორები და კომუტატორები, ასევე კლიენტი კომპიუტერების ქსელი. **SNMP** მმართველმა შეიძლება შეაგროვოს **SNMP** აგენტები მონაცემებისათვის, ან მონაცემები შეიძლება ავტომატურად იქნას გაგზავნილი **SNMP** მმართველთან, **SNMP** აგენტებზე **trap**-ების კონფიგურაციით.

ამ ლაბორატორიულ დავალებაში თქვენ გადმოიწერთ, დააინსტალირებთ და დააკონფიგურებთ **SNMP** მართვის პროგრამულ უზრუნველყოფას **PC-A**-ზე. თქვენ ასევე დააკონფიგურებთ **Cisco** მარშრუტიზატორებს და **Cisco** კომუტატორებს, როგორც **SNMP** აგენტებს. **SNMP** აგენტიდან მოსული **SNMP** შეტყობინების დაჭერის შემდეგ, თქვენ უნდა მოახდენთ **MIB/Object ID** კოდების კონვერტაციას, შეტყობინების დეტალების შესასწავლად **Cisco SNMP Object Navigator**-ის გამოყენებით.

შენიშვნა: მარშრუტიზატორები, რომლებიც გამოიყენება **CCNA**-ს პრაქტიკული სამუშაოებისთვის, არის **Cisco 1941** ინტეგრირებული სერვისების მარშრუტიზატორები **(ISRs) Cisco IOS Release 15.2(4)M3 (universalk9 image)** ვერსიასთან ერთად. გამოყენებული კომუტატორები არის **Cisco Catalyst 2960s** ვერსია, **Cisco IOS Release 15.0(2) (lanbasek9 image)** ოპერაციული სისტემით. შესაძლოა გამოყენებულ იქნას სხვა მარშრუტიზატორები, კომუტატორები და **Cisco IOS** ვერსიებიც. მოდელისა და **Cisco IOS** ვერსიის მიხედვით ხელმისაწვდომი ბრძანებები და მიღებული შედეგები შეიძლება იყოს განსხვავებული იმისგან, რაც ნაჩვენებია ამ ლაბორატორიულ სამუშაოში. მიაქციეთ ყურადღება მარშრუტიზატორის ინტერფეისის შემაჯამებელ ცხრილს ამ დავალების ბოლოში, სწორი ინტერფეისის იდენტიფიკატორებისათვის.

შენიშვნა: დარწმუნდით, რომ მარშრუტიზატორები და კომუტატორები წაშლილია და არ აქვთ საწყისი კონფიგურაციები. თუ არ ხართ დარწმუნებული დაუკავშირდით თქვენს ინსტრუქტორს.

შენიშვნა: **snmp-server** ბრძანებები ამ ლაბორატორიულ დავალებაში გამოიწვევს **Cisco 2960** კომუტატორზე გამაფრთხილებელი შეტყობინების გაშვებას, კონფიგურაციის ფაილის **NVRAM**-ში შენახვის დროს. გამაფრთხილებელი შეტყობინების თავიდან ასაცილებლად შეამოწმეთ კომუტატორი იყენებს თუ არა **lanbase-routing** შაბლონს. **IOS** შაბლონი იმართება კომუტატორის მონაცემთა ბაზის მმართველის (**Switch Database Manager - SDM**)-ის მიერ. სასურველი შაბლონის შეცვლის შემდეგ ახალი შაბლონი გამოყენებული იქნება გადატვირთვის შემდეგ, მაშინაც კი თუ კონფიგურაცია არ არის შენახული.

S1# **show sdm prefer**

გამოიყენეთ ქვემოთ მოცემული ბრძანებები lanbase-routing შაბლონის ნაგულისხმევ SDM შაბლონად მითითებისთვის.

S1# **configure terminal**

S1 (config) # **sdm prefer lanbase-routing**

S1 (config) # **end**

S1 # **reload**

მოთხოვნილი რესურსები:

- ორი მარშრუტიზატორი (**Cisco 1941 Cisco IOS Release 15.2(4)M3** უნივერსალი იმიჯით ან მსგავსით)
- ერთი კომუტატორი (**Cisco 2960 Cisco IOS Release 15.0(2) lanbasek9** იმიჯით ან მსგავსით)
- ერთი პერსონალური კომპიუტერი (**Windows** ოპერაციული სისტემა ტერმინალის ემულაციის პროგრამასთან ერთად, როგორიცაა **Tera Term**)

- ერთი პერსონალური კომპიუტერი (**Windows** ოპერაციული სისტემა ინტერნეტთან წვდომით)
- კონსოლის კაბელები **Cisco IOS** მოწყობილობების კონსოლის პორტებით კონფიგურაციისათვის.
- **Ethernet** და სერიალური კაბელები, როგორც ნაჩვენებია ტოპოლოგიაზე
- **SNMP** მართვის პროგრამული უზრუნველყოფა (**PowerSNMP** უფასო მმართველი **Dart Communication**-სგან, ან **SolarWinds Kiwi Syslog Server**-ის 30 დღიანი საცდელი ვერსია).

ნაწილი №1: ქსელის აწყობა და მოწყობილობის ბაზისური პარამეტრების კონფიგურაცია

ამ ნაწილში თქვენ მომართავთ ქსელის ტოპოლოგიას და დააკონფიგურებთ მოწყობილობას ბაზისური პარამეტრებით.

პირველი ეტაპი: ქსელის კაბელებით დაკავშირება, როგორც ნაჩვენებია ტოპოლოგიაზე.

მეორე ეტაპი: PC ჰოსტის კონფიგურაცია

მესამე ეტაპი: მარშრუტიზატორებისა და კომუტატორის ინიციალიზაცია და ხელახლა ჩატვირთვა აუცილებლობის შემთხვევაში

მეოთხე ეტაპი: ბაზისური პარამეტრების კონფიგურაცია მარშრუტიზატორებისა და კომუტატორისთვის.

ა. გათიშეთ **DNS lookup**

ბ. მომართეთ მოწყობილობების სახელები ისე როგორც ნაჩვენებია ტოპოლოგიაზე

გ. დააკონფიგურეთ **IP** მისამართები მისამართების ცხრილის მიხედვით. (ჯერჯერობით არ დააკონფიგუროთ S0/0/0 ინტერფეისი R1 მარშრუტიზატორზე.)

დ. დააყენეთ **cisco** კონსოლისა და **vty**-ის პაროლად და ჩართეთ შესვლა (**login**).

ე. პრივილეგირებული **EXEC** რეჟიმის შიფრირებულ პაროლად დააყენეთ **class**

ვ. დააკონფიგურეთ **logging Synchronous**, რათა აკრძალულ იქნას კონსოლის შეტყობინებები წყვეტის ბრძანებების ჩანაწერიდან.

ზ. **Ping** ბრძანების გაშვებით შეამოწმეთ **LAN** მოწყობილობებს შორის წარმატებული კავშირი.

თ. გადაიტანეთ გაშვებული კონფიგურაციის ასლი საწყის კონფიგურაციაში.

ნაწილი №2: SNMP მმართველისა და აგენტების კონფიგურაცია

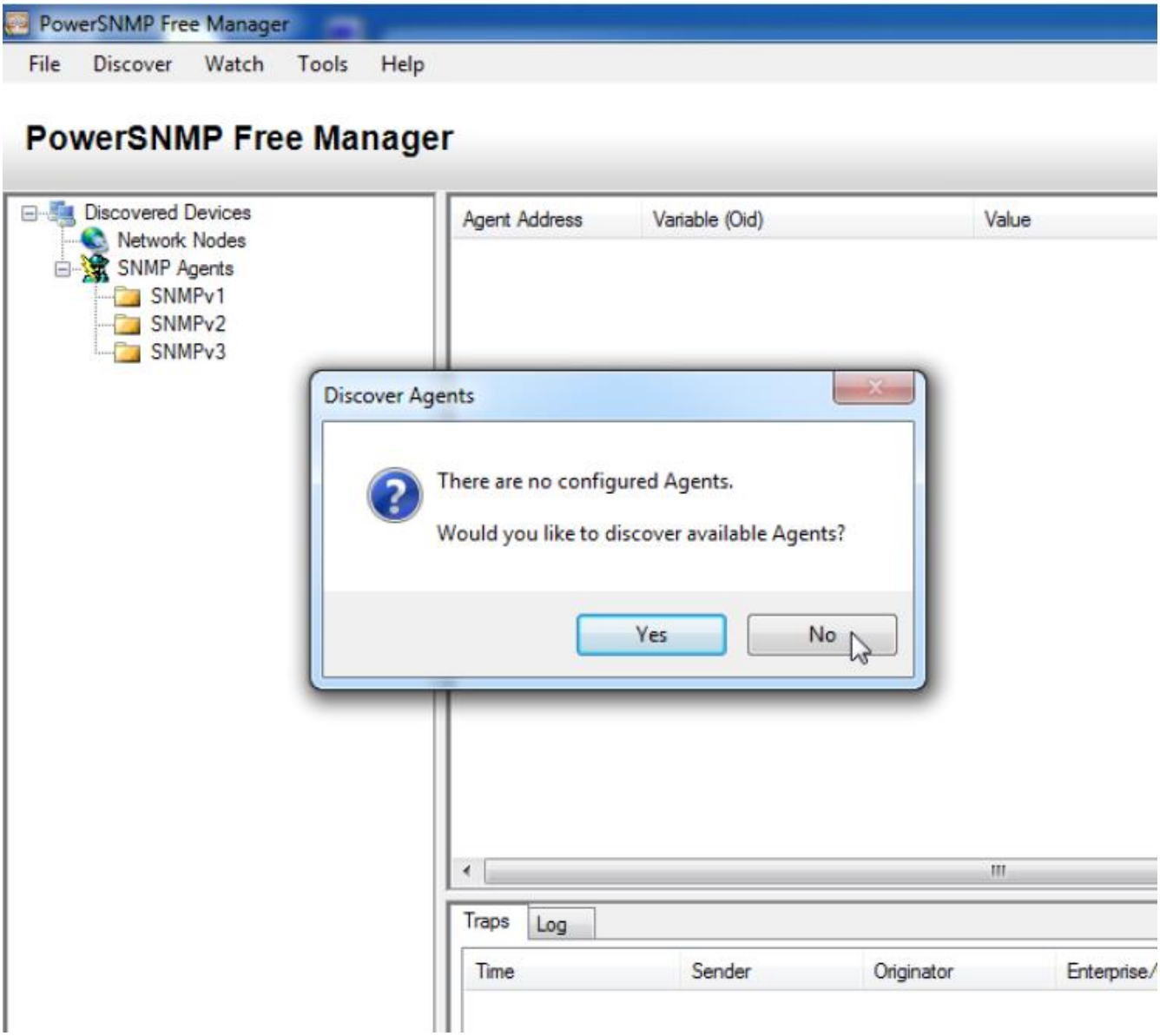
მეორე ნაწილში, **PC-A**-ზე მოხდება **SNMP** მართვის პროგრამული უზრუნველყოფის ინსტალაცია და კონფიგურაცია, ასევე **R1** და **S1** იქნება დაკონფიგურებული, როგორც **SNMP** აგენტები.

პირველი ეტაპი: SNMP მართვის პროგრამის ინსტალაცია

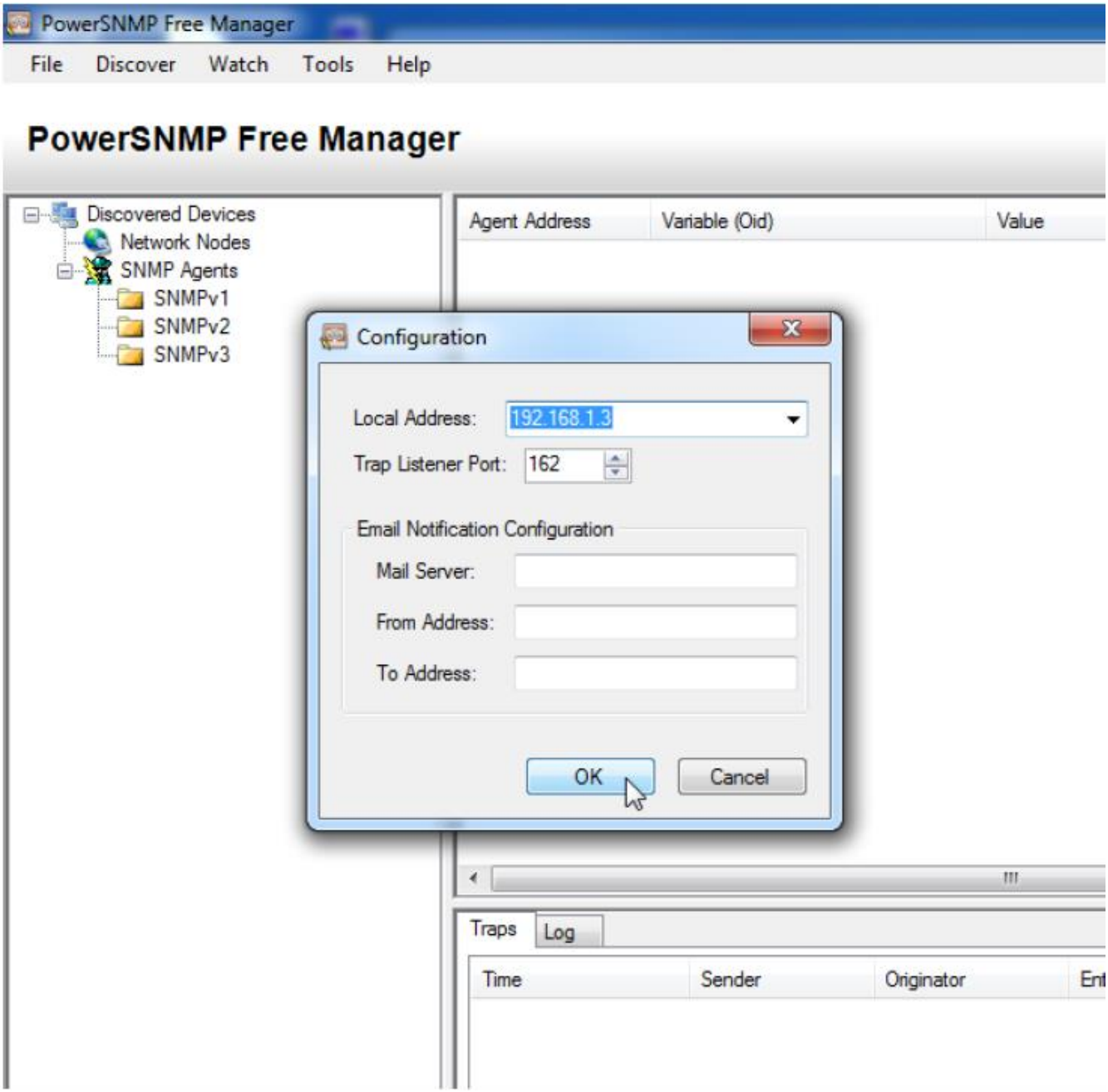
ა. გადმოწერეთ და დააინსტალირეთ **Dart Communications**-ის მიერ გამოშვებული **PowerSNMP Free Manager** პროგრამა ქვემოთ მოცემული მისამართიდან: http://www.dart.com/snmp-free-manager.aspx.

ბ. გაუშვით **PowerSNMP Free Manager** პროგრამა.

გ. დააჭირეთ **No** ღილაკს თუ შემოთავაზებულ იქნა ხელმისაწვდომი **SNMP** აგენტების აღმოჩენა. თქვენ აღმოაჩენთ **SNMP** აგენტებს, **R1** მარშრუტიზატორზე **SNMP**-ს კონფიგურაციის შემდეგ. **PowerSNMP Free Manager** მხარს უჭერს **SNMP**-ს 1, 2 და 3 ვერსიებს. მოცემულ ლაბორატორიულ სამუშაოში გამოყენებულია **SNMPv2**.



დ. გამოტანილ კონფიგურაციის ფანჯარაში (თუ ფანჯარა არ გამოჩნდა, მაშინ გადადით **Tools** განყოფილებაში და აირჩიეთ **configuration** ბრძანება), მომართეთ ლოკალური **IP** მისამართი **192.168.1.3**, მოსმენისათვის და დააჭირეთ **OK** ღილაკს.



შენიშვნა: თუ შემოთავაზებულ იქნა ხელმისაწვდომი **SNMP** აგენტების აღმოჩენა, დააჭირეთ **No** ღილაკს და გადადით ამ დავალების შემტებ ეტაპზე.

მეორე ეტაპი: **SNMP** აგენტის კონფიგურაცია

ა. **R1** მარშრუტიზატორზე გლობალური კონფიგურაციის რეჟიმიდან შეიყვანეთ ქვემოთ მოცემული ბრძანებები, მარშრუტიზატორის როგორც **SNMP** აგენტის კონფიგურაციისათვის. პირველ სტრიქონზე **SNMP** რიგების ერთობა (**Community**

string) არის **ciscolab**, მხოლოდ დათვალიერების პრივილეგიებით და **SNMP_ACL** სახელის მქონე წვდომის სია, რომელიც განსაზღვრავს თუ რომელი ჰოსტები არიან დაშვებული რომ მიიღონ **SNMP** ინფორმაცია **R1** მარშრუტიზატორიდან. მეორე და მესამე სტრიქონებზე **SNMP** მმართველის **location** და **contact** ბრძანებები იძლევიან აღწერილობით საკონტაქტო ინფორმაციას. მეოთხე სტრიქონი განსაზღვრავს ჰოსტის **IP** მისამართს, რომელიც მიიღებს **SNMP** შეტყობინებებს, **SNMP** ვერსიას და რიგების ერთობას (**Community string**). მეხუთე სტრიქონი რთავს ყველა ნაგულისხმევ **SNMP trap**-ს, ხოლო მე-6 და მე-7 სტრიქონები ქმნიან დასახელებულ წვდომის სიას, რათა აკონტროლოს თუ რომელი ჰოსტები არიან დაშვებული მარშრუტიზატორიდან **SNMP** ინფორმაციის მისაღებად.

R1 (config) # **snmp-server community ciscolab ro SNMP_ACL**

R1 (config) # **snmp-server location snmp_manager**

R1 (config) # **snmp-server contact ciscolab_admin**

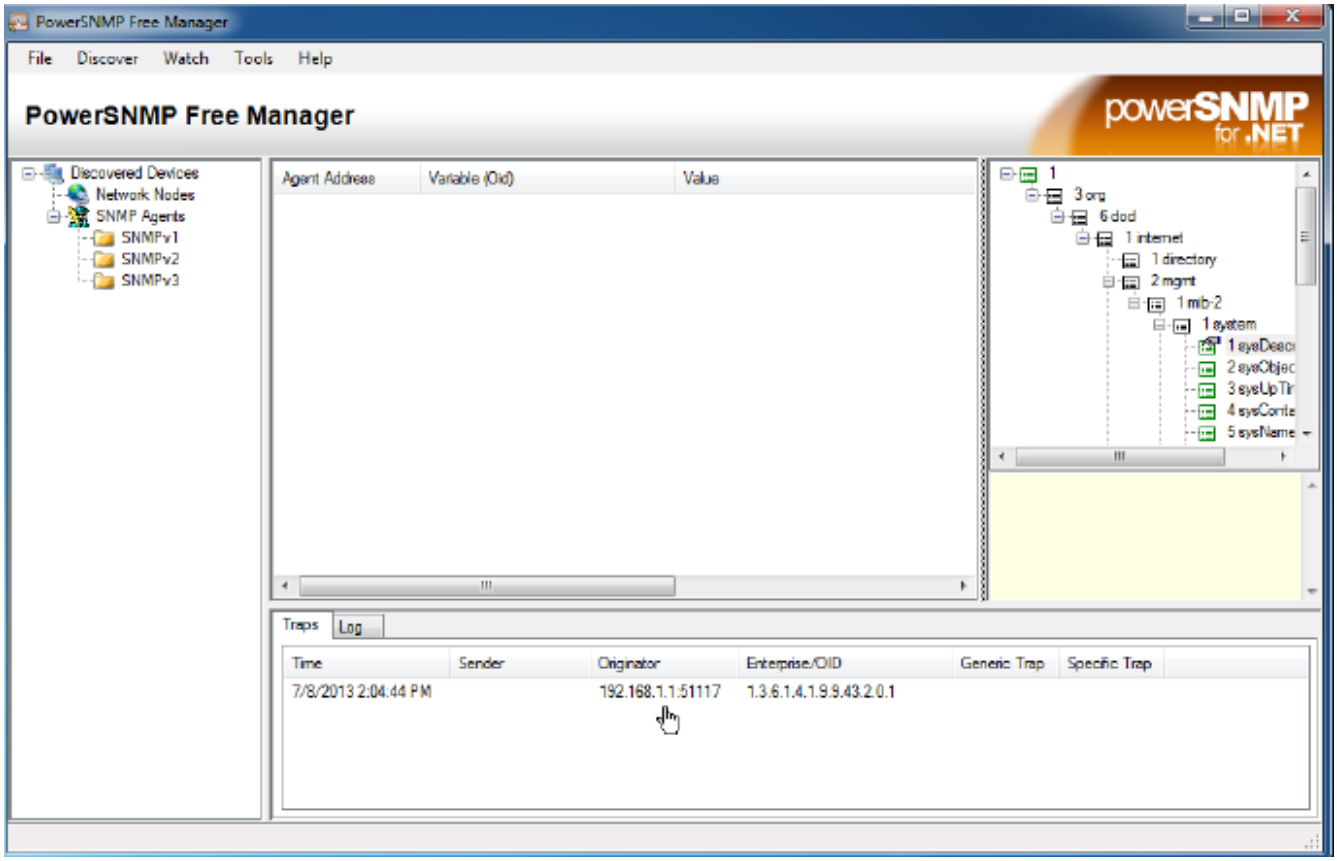
R1 (config) # **snmp-server host 192.168.1.3 version 2c ciscolab**

R1 (config) # **snmp-server enable traps**

R1 (config) # **ip access-list standard SNMP_ACL**

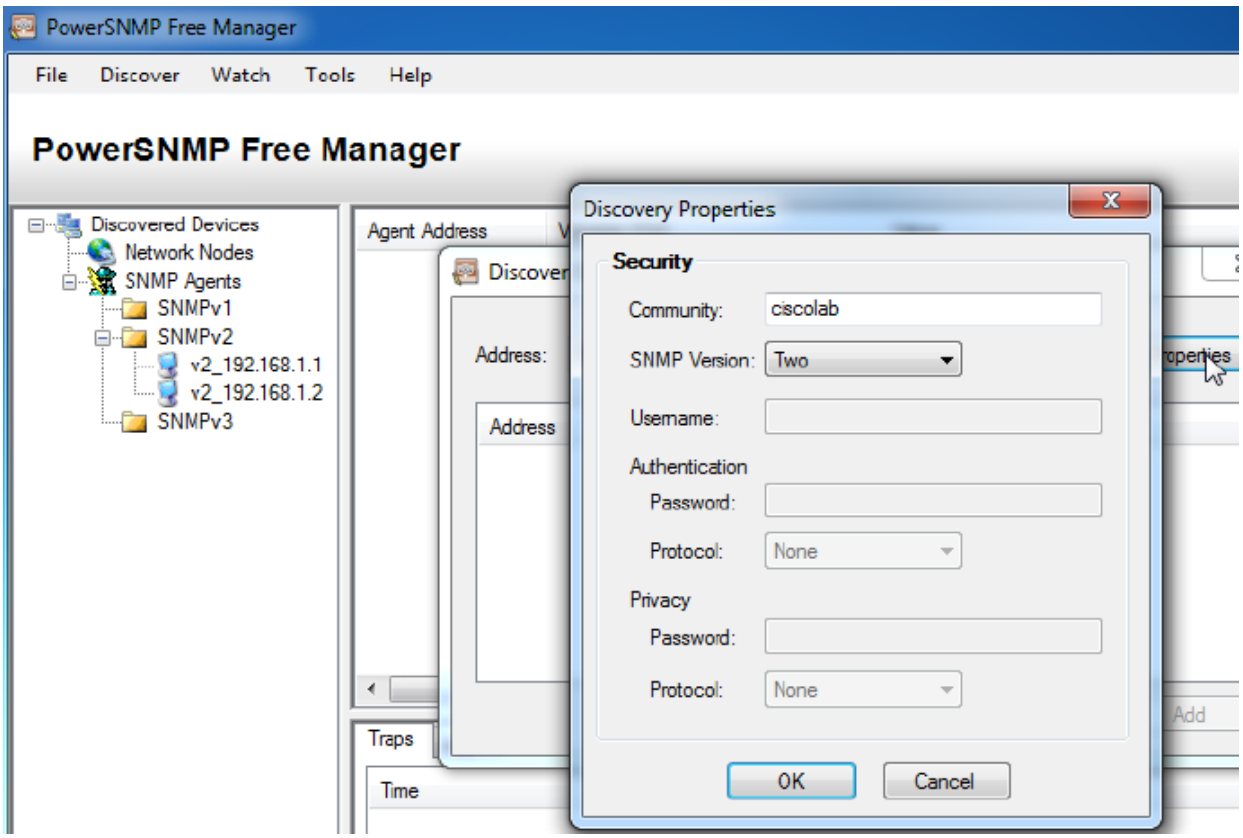
R1 (config-std-nacl) # **permit 192.168.1.3**

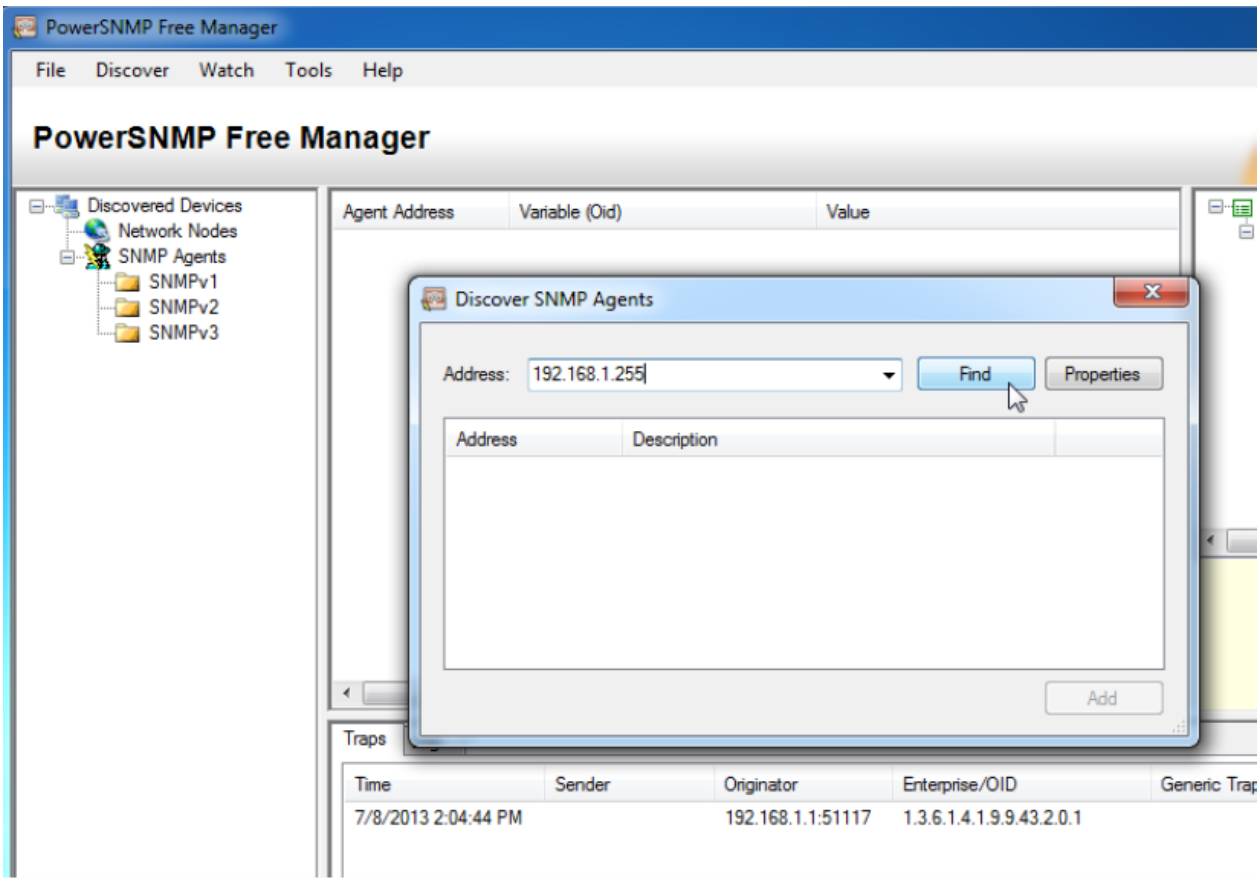
ბ. ამ ეტაპზე თქვენ შეიძლება გაფრთხილებულ იქნათ, რომ **PowerSNMP Free Manager** იღებს შეტყობინებებს **R1** მარშრუტიზატორიდან. თუ ასე არაა, მაშინ თქვენ შეგიძლიათ აიძულოთ **SNMP** შეტყობინება, რომ გაგზავნილ იქნეს **copy run start** ბრძანების შეყვანით **R1** მარშრუტიზატორზე. წარუმატებლობის შემთხვევაში გადადით შემდეგ ეტაპზე.

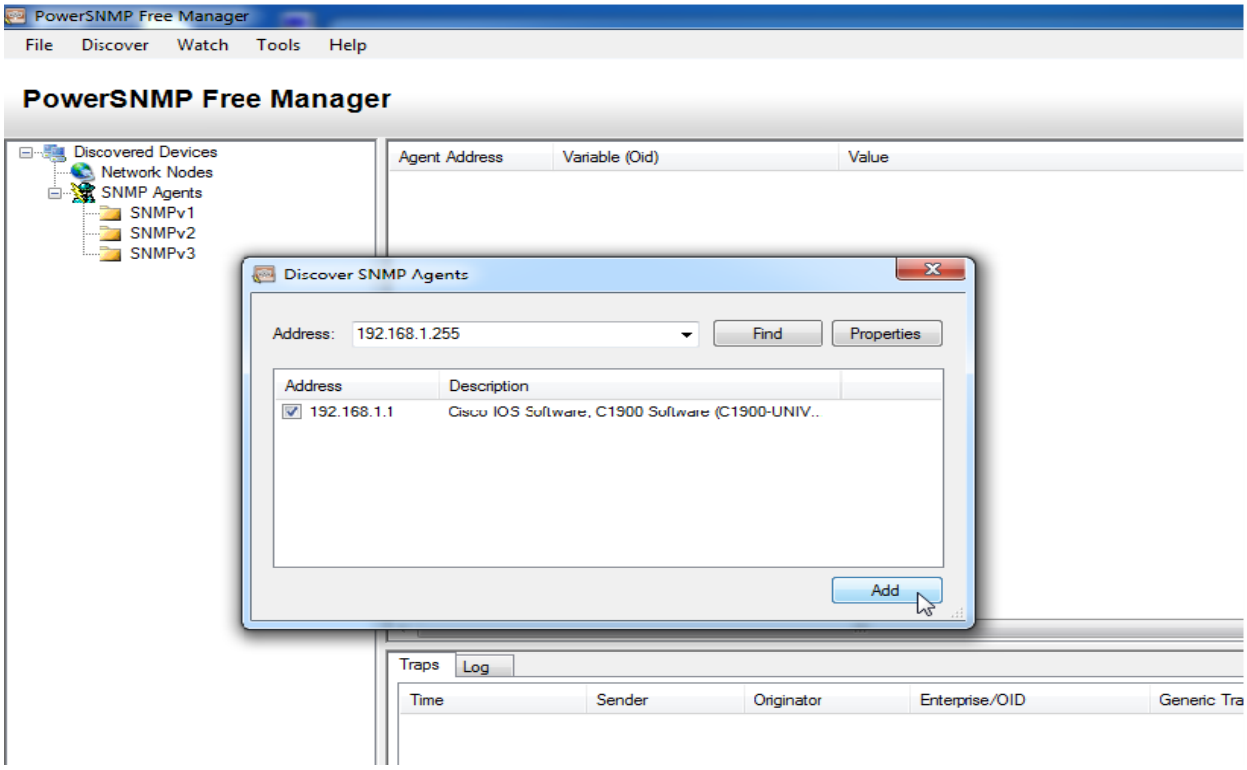


მესამე ეტაპი: SNMP აგენტის აღმოჩენა

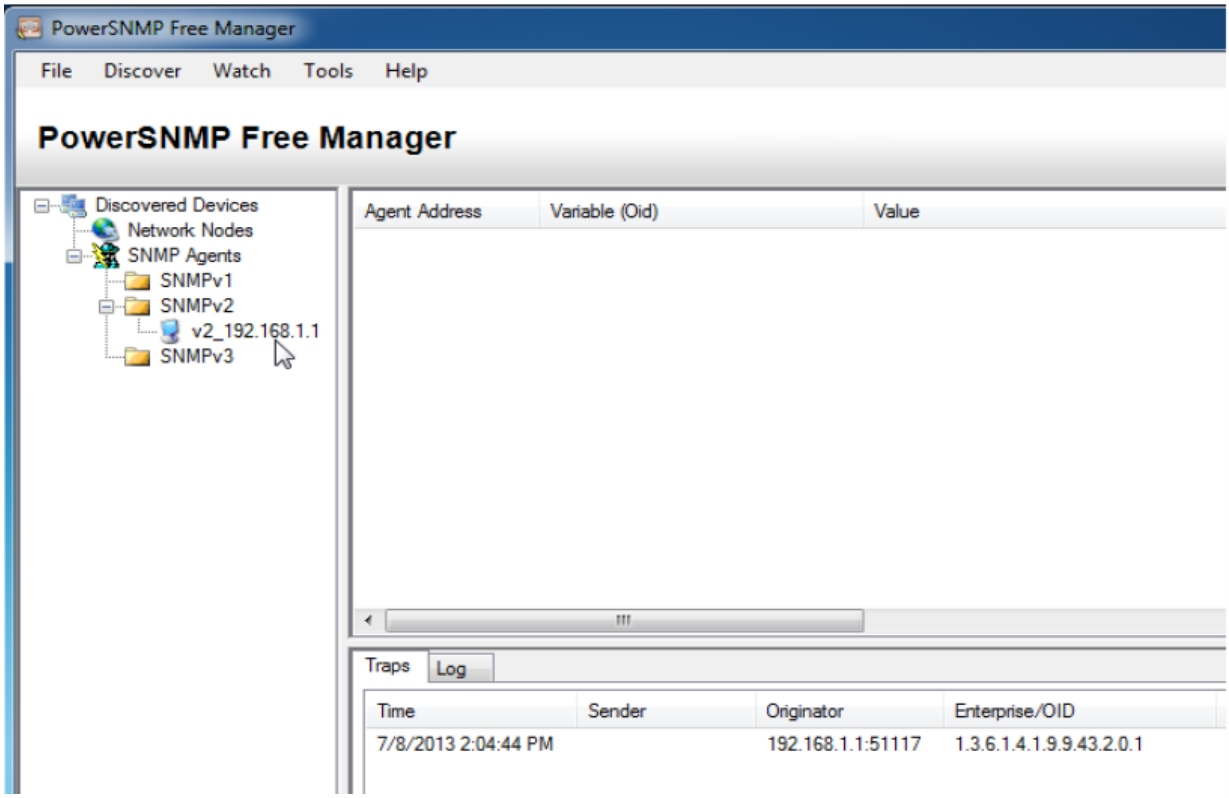
ა. **PC-A**-ზე დაყენებული **PowerSNMP Free Manager** პროგრამიდან, გახსენით **Discover** > **SNMP Agents** ფანჯარა. შეიყვანეთ **IP** მისამართი **192.168.1.255**. იგივე ფანჯარაში დააჭირეთ Properties ღილაკს და მომართეთ **ciscolab community** და **SNMP** ვერსია **ორი**, შემდეგ დააჭირეთ **OK** ღილაკს. ახლა თქვენ შეგიძლიათ დააწვეთ **Find** ღილაკს, ყველა **SNMP** აგენტის აღმოსაჩენად **192.168.1.0** ქსელში. **PowerSNMP Free Manager**-მა შეიძლება იპოვოს **R1** მარშრუტიზატორი **192.168.1.1**-ზე. დააწექით მონიშვნას და შემდეგ **Add** ღილაკს, **R1** მარშრუტიზატორის როგორც **SNMP** აგენტის დასამატებლად.


PowerSNMP Free Manager
File
Discover
Watch
Tools
Help
PowerSNMP Free Manager
Discovered Devices
Network Nodes
SNMP Agents
SNMPv1
SNMPv2
v2_192.168.1.1
v2_192.168.1.2
SNMPv3
Agent Address
Discover
Address:
Address
Traps
Time
Discovery Properties
Security
Community:
ciscolab
SNMP Version:
Two
Username:
Authentication
Password:
Protocol:
None
Privacy
Password:
Protocol:
None
OK
Cancel
Add


PowerSNMP Free Manager
File
Discover
Watch
Tools
Help
PowerSNMP Free Manager
Discovered Devices
Network Nodes
SNMP Agents
SNMPv1
SNMPv2
SNMPv3
Agent Address
Variable (Oid)
Value
Discover SNMP Agents
Address:
192.168.1.255
Find
Properties
Address
Description
Add
Traps
Time
Sender
Originator
Enterprise/OID
Generic Trap
7/8/2013 2:04:44 PM
192.168.1.1:51117
1.3.6.1.4.1.9.9.43.2.0.1



ბ. **PowerSNMP Free Manager**-ში, **R1** მარშრუტიზატორი დაემატა **SNMPv2** ხელმისაწვდომი აგენტების სიაში.



გ. დააკონფიგურეთ **S1**, როგორც **SNMP** აგენტი. თქვენ შეგიძლიათ გამოიყენოთ იგივე **snmp-server** ბრძანებები, რომლებიც გამოიყენეთ **R1** მარშრუტიზატორის კონფიგურაციისას.

დ. **S1**-ის კონფიგურაციის შემდეგ, **SNMP** შეტყობინებები **192.168.1.2**-დან ნაჩვენებია **PowerSNMP Free Manager** პროგრამის **trap**-ების ფანჯარაში. **PowerSNMP Free Manager**-ში დაამატეთ **S1**, როგორც **SNMP** აგენტი იგივე პროცესის გამოყენებით, რომელიც გამოიყენეთ **R1**-ის აღმოჩენის დროს.

ნაწილი №3: OID კოდების კონვერტაცია Cisco SNMP Object Navigator-ის საშუალებით

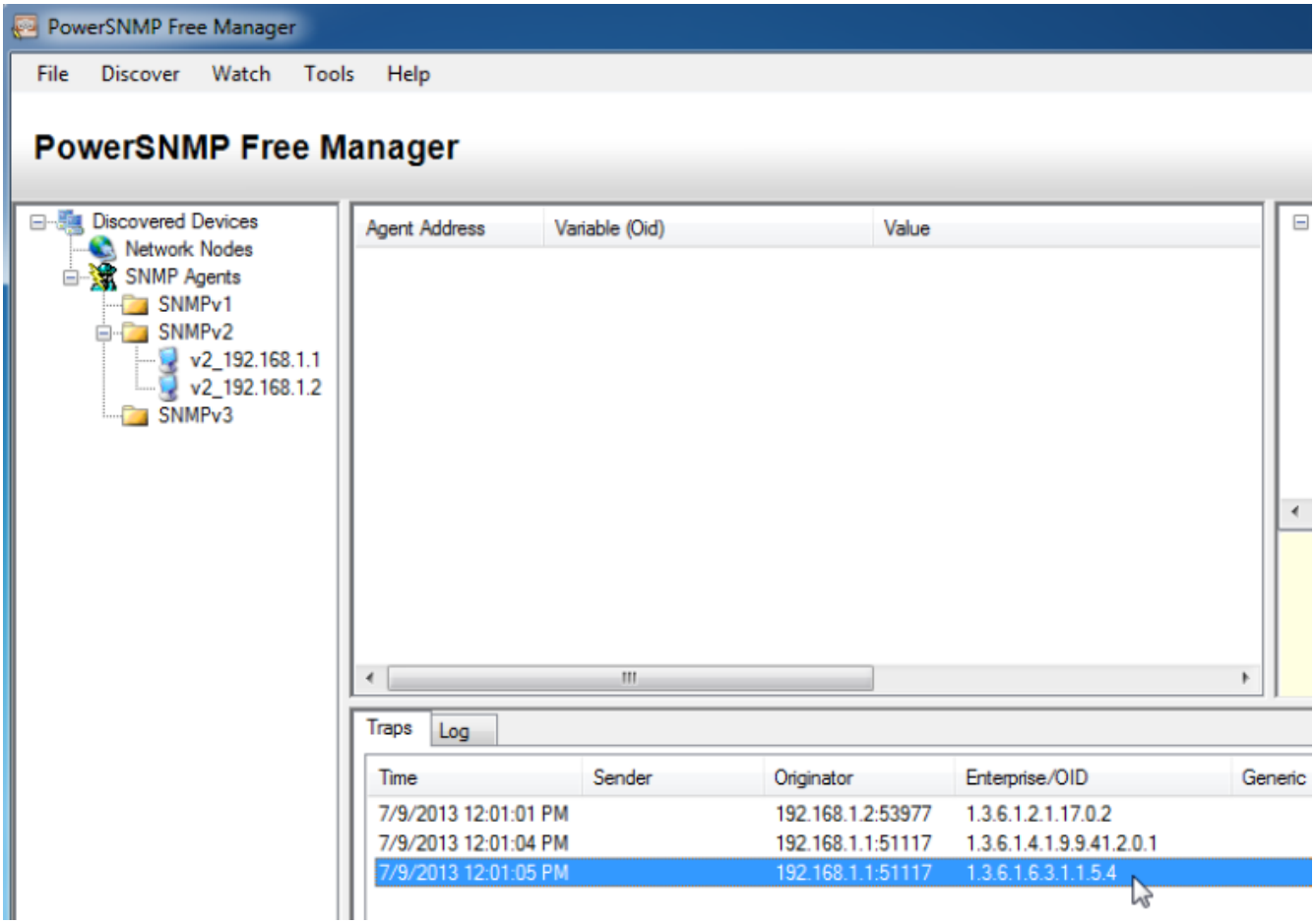
მესამე ნაწილში თქვენ აიძულებთ **SNMP** შეტყობინებებს რომ გაიგზავნონ **SNMP** მმართველზე, რომელიც განთავსებულია **PC-A**-ზე. შემდეგ თქვენ უნდა მოახდინოთ მიღებული **OID** კოდების კონვერტაცია სახელებით, რათა შესწავლილ იქნას შეტყობინებების „ბუნება“. **MIB/OID** კოდები შეიძლება მარტივად დაკონვერტირდეს **Cisco SNMP Object Navigator**-ის გამოყენებით, რომელის განთავსებულია http://www.cisco.com -ზე

პირველი ეტაპი: მიმდინარე SNMP შეტყობინებების წაშლა.

PowerSNMP Free Manager პროგრამაში, მარჯვენა ღილაკით დააჭირეთ **Traps** ფანჯარას და აირჩიეთ **Clear** ბრძანება, **SNMP** შეტყობინებების წასაშლელად.

მეორე ეტაპი: SNMP trap-სა და შეტყობინების შექმნა.

R1 მარშრუტიზატორზე დააკონფიგურეთ **S0/0/0** ინტერფეისი ლაბორატორიული სამუშაოს თავში მოცემული მისამართების ცხრილის მიხედვით. განახორციელეთ წვდომა გლობალური კონფიგურაციის რეჟიმში და **SNMP trap** შეტყობინებების შექმნისათვის ჩართეთ ინტერფეისი, რათა გაგზავნილ იქნას ისინი **SNMP** მმართველთან **PC-A**-ზე. გაითვალისწინეთ **Enterprise/OID** კოდური ნომრები, რომლის ნახვაც შესაძლებელია **traps** ფანჯარაში.



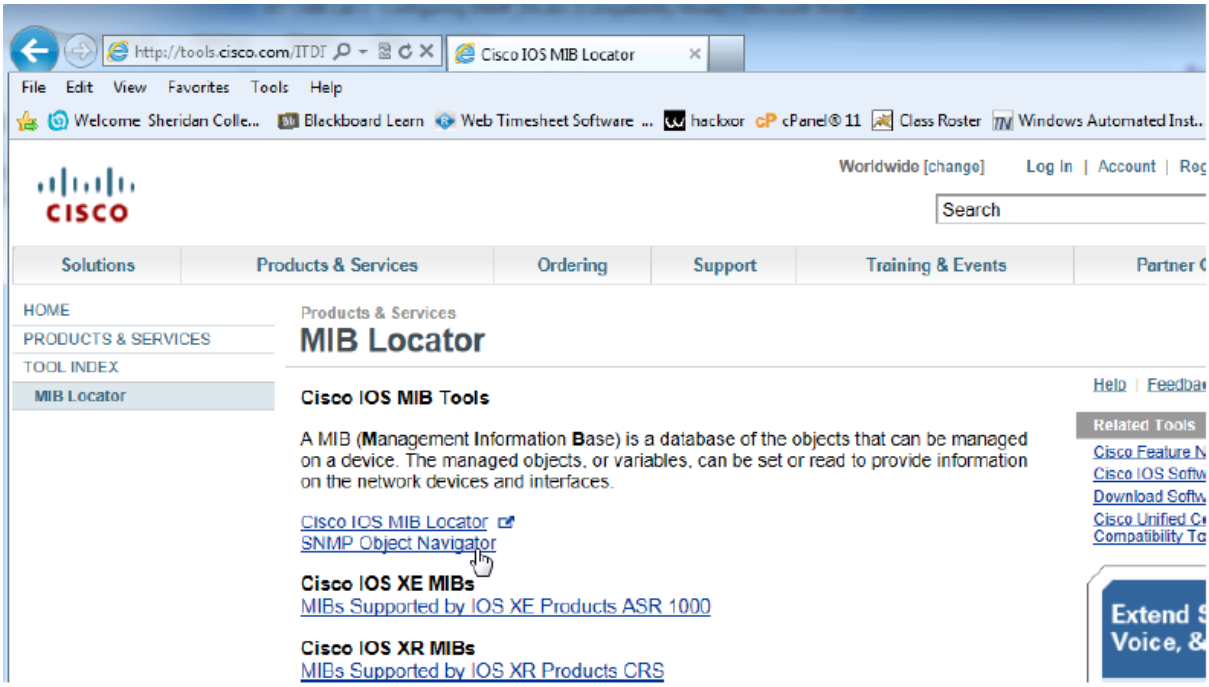
მესამე ეტაპი: SNMP MIB/OID შეტყობინებების დეკოდირება

ინტერნეტიანი კომპიუტერიდან გახსენით ბრაუზერი და გადადით http://www.cisco.com ვებ-გვერდზე.

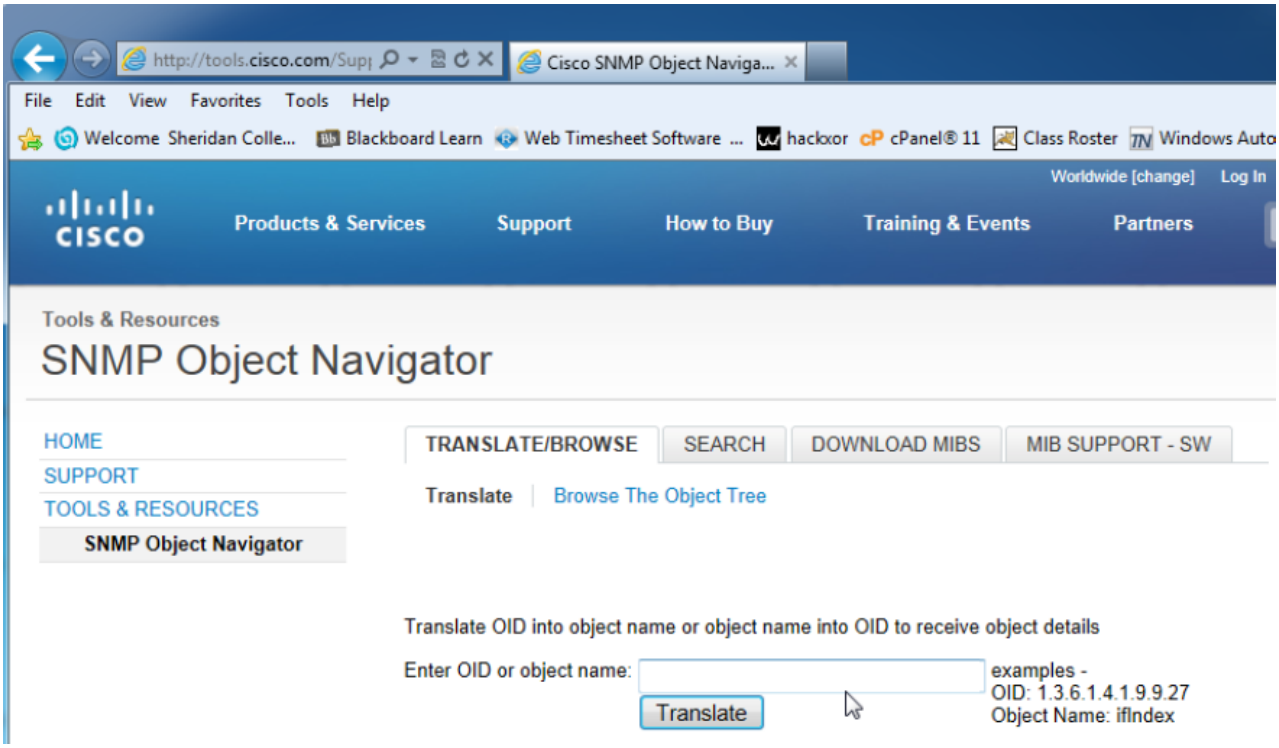
ა. ფანჯრის ზედა ნაწილში არსებული ძებნის უტილიტის დახმარებით, მოძებნეთ **SNMP Object Navigarot**-ი.

ბ. მიღებული შედეგიდან აირჩიეთ **SNMP Object Navigator MIB Download MIBs OID OIDs**.

გ. გადადით **MIB Locator** გვერდზე. დააჭირეთ **SNMP Object Navigrator** ბმულს



დ. **SNMP Object Navigator** გვერდის გამოყენებით, მოახდინეთ **OID** კოდური ნომრების დეკოდირება **PowerSNMP Free Manager**-დან, რომელიც შექმნილია მესამე ნაწილის მეორე ეტაპზე. შეიყვანეთ **OID** კოდური ნომერი და დააწექით **Translate** ღილაკს.



ე. ჩაწერეთ ქვემოთ მოცემულ ველებში **OID** კოდური ნომრები და მათი შესაბამისი შეტყობინების თარგმანი.

_____

_____

_____

ასახვა (Reflection)

1. რა პოტენციური უპირატესობები გააჩნია ქსელის **SNMP**-თი მონიტორინგს? _____

_____

_____

2. რატომ არის სასურველი მხოლოდ წაკითხვის წვდომის რეჟიმის გამოყენება **SNMPv2**-თან მუშაობის დროს? _____

_____

_____

მარშრუტიზატორის ინტერფეისის შემაჯამებელი ცხრილი:

მარშრუტიზატორის ინტერფეისის შეჯამება				
მარშრუტიზატორის მოდელი	Ethernet interface №1	Ethernet interface №2	Serial interface №1	Serial interface №2
1800	FastEthernet 0/0 (F0/0)	FastEthernet 0/1 (F0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
1900	GigabitEthernet 0/0 (G0/0)	GigabitEthernet 0/1 (G0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
2801	FastEthernet 0/0 (F0/0)	FastEthernet 0/1 (F0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
2811	FastEthernet 0/0 (F0/0)	FastEthernet 0/1 (F0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
2900	GigabitEthernet 0/0 (G0/0)	GigabitEthernet 0/1 (G0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)

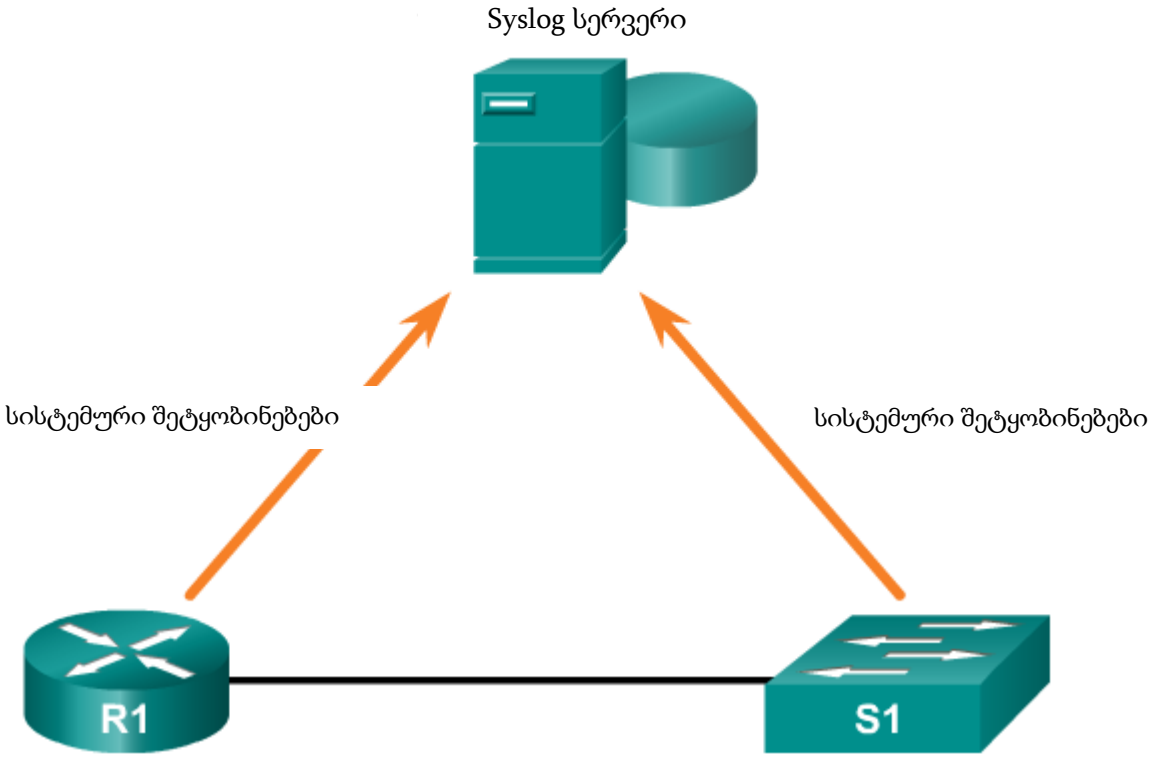
შენიშვნა: თუ გინდათ მოძებნოთ თუ როგორაა მარშრუტიზატორი კონფიგურირებული, დაათვალიერეთ ინტერფეისები რათა მოახდინოთ მარშრუტიზატორის იდენტიფიკაცია და დაადგინოთ რამდენი ინტერფეისი აქვს მას. აქ არ არის იმის საშუალება, რომ ჩამოიწეროს თითოეული მარშრუტიზატორის კონფიგურაციების კომბინაციები. მოცემული ცხრილი მოიცავს მოწყობილობის **Ethernet** და **Serial** ინტერფეისების შესაძლო კომბინაციების იდენტიფიკატორებს. ეს ცხრილი არ მოიცავს ნებისმიერი სხვა ტიპის ინტერფეისს, რადგან კონკრეტული მარშრუტიზატორი შეიძლება შეიცავდეს მხოლოდ ერთს. ამის მაგალითი შეიძლება იყოს **ISDN BRI** ინტერფეისი. ფრჩხილებში მოცემული ცვლადი არის ლეგალური აბრევიატურა, რომელიც შეიძლება გამოყენებულ იქნას **Cisco IOS** ბრძანებებში, ინტერფეისის წარმოსადგენად.

6.3.2 Syslog, NTP, Netflow პროტოკოლის კონფიგურირება

6.3.2.1 Syslog-ის გაცნობა

როდესაც რაიმე მოვლენა ხდება ქსელში, ქსელის მოწყობილობებს აქვთ სანდო მექანიზმი ადმინისტრატორისთვის დეტალური სისტემური შეტყობინებების გასაცნობად. ეს შეტყობინებები შეიძლება იყოს ან არაკრიტიკული ან მნიშვნელოვანი. ქსელის ადმინისტრატორებს აქვთ შენახვის, განმარტების და ჩვენების შეტყობინებების სხვადასხვა ვარიანტები, ასევე მიმდინარეობს გაფრთხილება იმ შეტყობინებებით, რომელთაც შეიძლება დიდი გავლენა იქონიონ ქსელის ინფრასტრუქტურაზე.

ყველაზე გავრცელებული მეთოდი, იმ სისტემურ შეტყობინებებზე წვდომისთვის, რომელთაც იძლევიან ქსელის მოწყობილობები, არის Syslog პროტოკოლის გამოყენება.



სურ. 6.3.2.1 Syslog

Syslog არის ტერმინი, რომელიც გამოიყენება სტანდარტის აღსაწერად. ის ასევე გამოყენებულია იმ პროტოკოლის აღსაწერად, რომელიც განვითარდა ამ სტანდარტისთვის. Syslog პროტოკოლი განვითარდა UNIX სისტემებისთვის 1980 წელში, მაგრამ პირველად მისი

დოკუმენტირება IETF-ის მიერ, როგორც RFC 3164, 2001 წელს. Syslog იყენებს UDP პორტის ნომერს 514 მოვლენის შესახებ შეტყობინების გასაგზავნად მთელს IP ქსელებში, მოვლენების შეტყობინების შემგროვებლისათვის, ისე როგორც ნაჩვენებია 5.2.1 სურათზე:

ბევრი ქსელური მოწყობილობა უჭერს მხარს Syslog-ს, მარშრუტიზატორების, კომუტატორების, აპლიკაციების სერვერების, ფაიერვოლების და სხვა ქსელური მოწყობილობების ჩათვლით. Syslog პროტოკოლი საშუალებას აძლევს ქსელურ მოწყობილობებს გააგზავნონ თავისი სისტემური შეტყობინებები მთელს ქსელში, Syslog სერვერებისთვის. შესაძლებელია სპეციალური out-of-band (OOB) ქსელის აწყობა ამ მიზნებისთვის.

არსებობს რამდენიმე განსხვავებული Syslog სერვერის პროგრამული უზრუნველყოფის პაკეტები Windows და UNIX სისტემებისათვის. ბევრი მათგანი არის უფასო.

Syslog აღრიცხვის ჟურნალის სერვისი უზრუნველყოფს სამ მთავარ ფუნქციას:

- ჩანაწერების ინფორმაციის შეგროვების შესაძლებლობა მონიტორინგისა და პრობლემის მოძებნისთვის;
- რეგისტრირებული ჩანაწერების ინფორმაციიდან ამორჩევის შესაძლებლობა;
- რეგისტრირებული Syslog შეტყობინებების ადრესატების განსაზღვრის შესაძლებლობა.
-

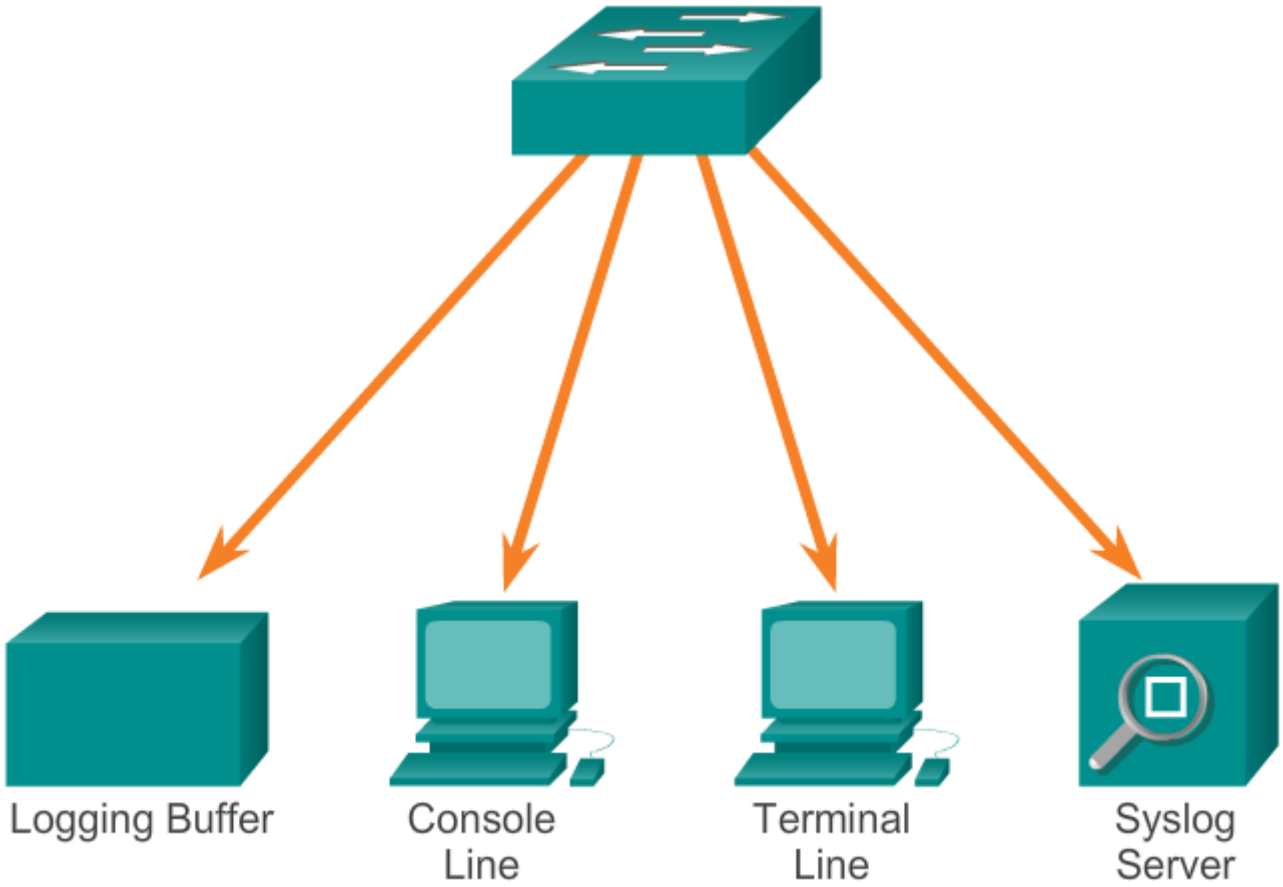
6.3.2.2 Syslog ოპერაცია

Cisco ქსელურ მოწყობილობებზე Syslog პროტოკოლი იწყება სისტემური შეტყობინებების გაგზავნით და debug შედეგით ლოკალური რეგისტრაციის პროცესიდან, მოწყობილობის შიგნით. თუ როგორ მართავს ლოგირების პროცესი ამ შეტყობინებებს და შედეგებს დამოკიდებულია მოწყობილობის კონფიგურაციაზე. მაგალითად, Syslog შეტყობინებები შეიძლება გაგზავნილ იქნას მთელი ქსელის მასშტაბით, გარე Syslog სერვერზე. ეს შეტყობინებები შეიძლება მიღებულ იქნას მოქმედ მოწყობილობასთან წვდომის საჭიროების გარეშე. აღრიცხვის (Log) შეტყობინებები და შედეგები, რომლებიც

ინახება გარე სერვერზე, შეიძლება გადმოტანილ იქნას სხვადასხვა ანგარიშებში, უფრო მარტივად წაკითხვისთვის.

ალტერნატიულად, Syslog შეტყობინებები შეიძლება გაგზავნილ იქნას შიდა ბუფერში. შიდა ბუფერში გაგზავნილი შეტყობინებების ნახვა შესაძლებელია მხოლოდ მოწყობილობის CLI-დან.

ბოლოს, ქსელის ადმინისტრატორს შეუძლია მიუთითოს, რომ მხოლოდ განსაზღვრული ტიპის სისტემური შეტყობინებები იგზავნება დანიშნულების სხვადასხვა პუნქტებში. მაგალითად, მოწყობილობა შეიძლება იყოს კონფიგურირებული ისე, რომ გადააგზავნოს ყველა სისტემური შეტყობინება გარე syslog სერვერზე. თუმცა, გამართვის დონის (Debug-level) შეტყობინებები იგზავნება შიდა ბუფერში და ადმინისტრატორს მასთან წვდომა შეუძლია მხოლოდ CLI-ით.



სურ. 6.3.2.2 Syslog შეტყობინების დანიშნულების ადგილის ვარიანტები

როგორც 6.3.2.2 სურათზეა მოცემული, Syslog შეტყობინებების პოპულარული დანიშნულების ადგილები მოიცავს:

- აღრიცხვის ბუფერი (მარშრუტიზატორის ან კომუტატორის ოპერატიული მეხსიერების შიდა ნაწილი)
- კონსოლის ხაზი
- ტერმინალის ხაზი
- Syslog სერვერი

შესაძლებელია სისტემური შეტყობინებების დაშორებულად მონიტორინგი, ლოგების დათვალიერებით Syslog სერვერზე, ან მოწყობილობასთან წვდომით Telnet-ის, SSH-ის ან კონსოლის პორტის გამოყენებით.

6.3.2.3 Syslog-ის შეტყობინების ფორმატი

Cisco-ს მოწყობილობები იძლევა Syslog შეტყობინებებს ქსელის რაიმე მოვლენის მიხედვით. თითოეული Syslog შეტყობინება შეიცავს მკაცრი წესების დონეს და შესაძლებლობებს.

Severity Name	Severity Level	Explanation
Emergency	Level 0	System Unusable
Alert	Level 1	Immediate Action Needed
Critical	Level 2	Critical Condition
Error	Level 3	Error Condition
Warning	Level 4	Warning Condition
Notification	Level 5	Normal, but Significant Condition
Informational	Level 6	Informational Message
Debugging	Level 7	Debugging Message

სურ. 6.3.2.3 Syslog სიმკაცრის დონე

მცირე ნომრიანი დონე არის მეტად კრიტიკული Syslog-ის განგაშის სიგნალი. შეტყობინებების მკაცრი წესების დონე შეიძლება დაყენებულ იქნას კონტროლისთვის, თუ

სად უნდა გამოჩნდეს თითოეული ტიპის შეტყობინება (ანუ კონსოლზე თუ სხვა დანიშნულების ადგილზე). Syslog დონეების სრული სია მოცემულია 6.3.2.3 სურათზე.

Syslog-ის თითოეულ დონეს აქვს თავისი მნიშვნელობა:

- გაფრთხილების დონე (Warning Level) - გადაუდებელი დონე (Emergency Level) - მოცემული შეტყობინებები არის შეცდომის შეტყობინებები პროგრამული ან ტექნიკური უზრუნველყოფის ცუდათ ფუნქციონირების შესახებ; ამ ტიპის შეტყობინებები ნიშნავს იმას, რომ მოწყობილობა არის დაზიანებული. პრობლემის სირთულე განსაზღვრავს გამოყენებული Syslog-ის დონეს.
- გამართვის დონე (Debugging Level) - მოცემული დონე მიუთითებს, რომ შეტყობინებები შედეგი, რომელიც გენერირებულია სხვადასხვა გამომავალი debug ბრძანებებიდან.
- შეტყობინების დონე (Notification Level) - შეტყობინების დონე არის მხოლოდ ინფორმაციისთვის, მოწყობილობის ფუნქციონირება არ არის დაზიანებული. ინტერფეისის up-ში ან down-ში გადასვლები, სისტემის გადატვირთვის შეტყობინებები ჩნდება შეტყობინების დონეზე.

დამატებით, სიმკაცრის (Severity) მისათითებლად, Syslog შეტყობინებები ასევე შეიცავს ინფორმაციას შესაძლებლობებზე. Syslog-ის შესაძლებლობები არის სერვისის იდენტიფიკატორები, რომელიც ამოიცნობს და კატეგორიებად ყოფს სისტემის მდგომარეობის მონაცემებს შეცდომებისა და მოვლენების შეტყობინების ანგარიშებისთვის. ხელმისაწვდომი ლოგირების შესაძლებლობის პარამეტრები ინდივიდუალურია ქსელური მოწყობილობისათვის. მაგალითად, Cisco 2960 კომუტატორზე, რომელზეც გაშვებულია Cisco IOS Release 15.0(2) და Cisco 1941 მარშრუტიზატორზე, რომელზეც გაშვებულია Cisco IOS Release 15.2(4) სისტემა, მხარს უჭერენ 24 შესაძლებლობის პარამეტრს, რომელიც დაყოფილია კატეგორიებად 12 ტიპის შესაძლებლობის მიხედვით.

ზოგიერთი გავრცელებული Syslog შეტყობინების შესაძლებლობები, რომლებიც ამოღებულია Cisco IOS მარშრუტიზატორებიდან, მოიცავს:

- IP

- OSPF პროტოკოლს
- SYS ოპერაციულ სისტემას
- IP Security (IPsec)
- Interface IP (IF)

ნაგულისხმევად, syslog შეტყობინებების ფორმატი Cisco IOS პროგრამულ უზრუნველყოფაზე არის შემდეგნაირი: seq no: timestamp: %facility-severity-MNEMONIC: description.

ველები, რომლებსაც მოიცავს Cisco IOS პროგრამული უზრუნველყოფაში, Syslog შეტყობინება, აღწერილია 6.3.2.4 სურათზე:

Field	Explanation
seq no	Stamps log messages with a sequence number only if the **service sequence-numbers** global configuration comman is configured.
timestamp	Date and time of the message or event, which appears only if the **service timestamps** global configuration command is configured.
facility	The facility to which the message refers.
severity	Single-digit code from 0 to 7 that is the severity of the message.
MNEMONIC	Text string that uniquely describes the message.
description	Text string containing detailed information about the event bein reported.

სურ. 6.3.2.4. Syslog-ის შეტყობინების ფორმატი

მაგალითად, მარტივი შედეგი Cisco კომუტატორზე EtherChannel ხაზის მდგომარეობის შეცვლა up-ში, არის: 00:00:46: %LINK-3-UPDOWN: Interface Port-channell, changed state to up.

აქ შესაძლებლობა არის LINK და სიმკაცრის დონე არის 3, UPDOWN MNEMONIC-ით.

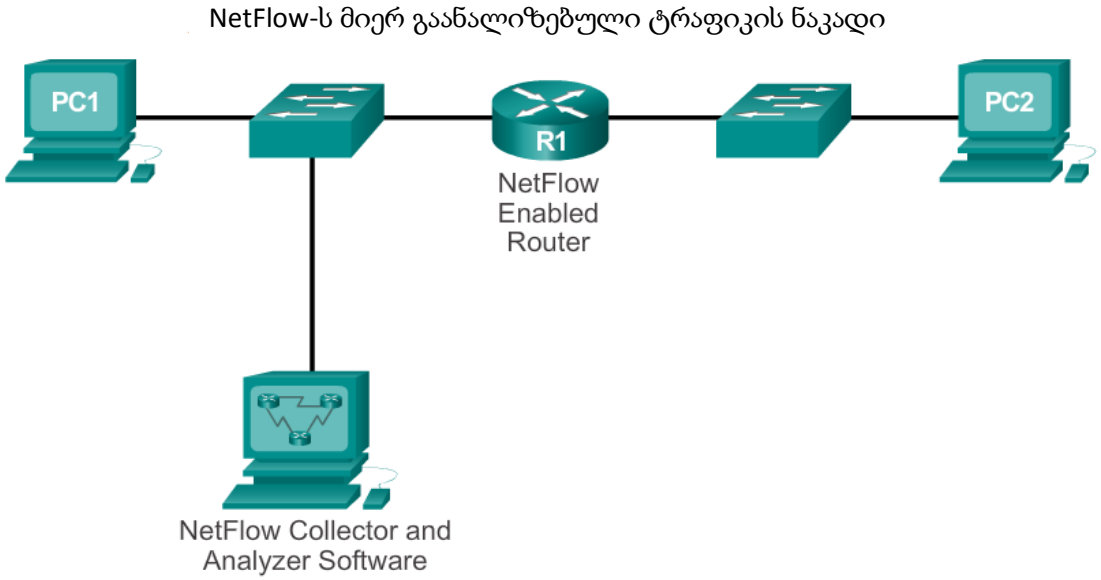
ყველაზე გავრცელებული შეტყობინებები არის ხაზის up და down მდგომარეობაში გადასვლის შეტყობინებები, და შეტყობინებები, რომლებიც მოწყობილობას გამოაქვს, როდესაც ხდება კონფიგურაციის რეჟიმიდან გამოსვლა. თუ ACL ლოგირება

კონფიგურირებულია, მაშინ მოწყობილობა ახდენს Syslog შეტყობინებების გენერაციას, როცა პაკეტები ემთხვევა პარამეტრის პირობებს.

6.3.2.4 NetFlow-ს მიმოხილვა

NetFlow არის Cisco IOS ტექნოლოგია, რომელიც იძლევა სტატისტიკას პაკეტების ნაკადზე, Cisco მარშრუტიზატორის ან მრავალდონიანი კომუტატორის საშუალებით. NetFlow არის სტანდარტი IP ქსელებიდან IP ექსპლუატაციის მონაცემების შესაგროვებლად.

ისტორიულად, NetFlow ტექნოლოგია განვითარდა იმიტომ, რომ ქსელის პროფესიონალებს სჭირდებოდათ მარტივი და მოხერხებული მეთოდი ქსელში TCP/IP ნაკადების თვალყურის დევნების, და SNMP არ იყო საკმარისი ამ მოთხოვნებისთვის. მაშინ, როცა SNMP ცდილობს ქსელის მართვის მახასიათებლებისა და პარამეტრების ძალიან ფართო დიაპაზონის მოწოდებას, NetFlow ამ დროს ფოკუსირებულია IP პაკეტების ნაკადების სტატისტიკის მოწოდებაზე, ქსელური მოწყობილობების დახმარებით.



სურ. 6.3.2.5 NetFlow ქსელში

NetFlow გვაწვდის ინფორმაციას, რაც იძლევა ქსელის და უსაფრთხოების მონიტორინგის, ქსელის დაგეგმვის, ტრაფიკის ანალიზის და IP ანგარიშის შესაძლებლობას. მაგალითად 6.3.2.5 სურათზე PC1 უკავშირდება PC2-ს აპლიკაციის დახმარებით, როგორიცაა HTTPS. NetFlow-ს შეუძლია ამ აპლიკაციის კავშირის მონიტორინგი, აპლიკაციის

ინდივიდუალური ნაკადის ბაიტებისა და პაკეტების დათვლა. შემდეგ ის უშვებს სტატისტიკას გარე სერვერზე, რომელსაც NetFlow კოლექტორი ეწოდება.

NetFlow გახდა მონიტორინგის სტანდარტი და ახლა ის ფართოდაა მხარდაჭერილი ქსელურ ინდუსტრიაში.

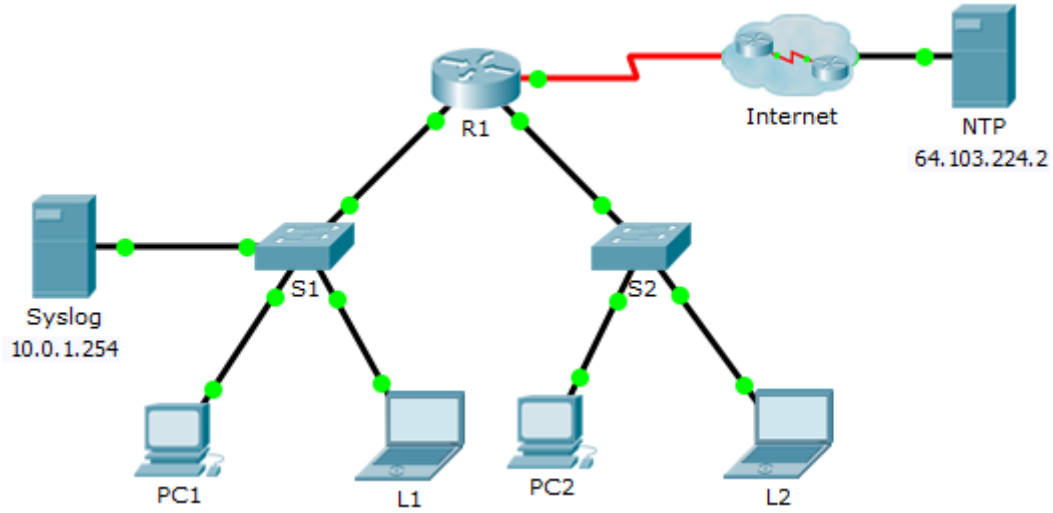
მოქნილი NetFlow არის NetFlow-ს ბოლო ტექნოლოგია. მოქნილი NetFlow აუმჯობესებს „ორიგინალ NetFlow“-ს, ტრაფიკის პარამეტრების მომართვის შესაძლებლობების დამატებით, რომელიც გამოიყენება ქსელის ადმინისტრატორის კონკრეტული მოთხოვნებისთვის.

მოქნილი NetFlow ამარტივებს უფრო რთული ტრაფიკის ანალიზის და მონაცემთა ექსპორტის კონფიგურაციის შექმნას, მრავალჯერადი კონფიგურაციის კომპონენტების გამოყენებით.

მოქნილი NetFlow იყენებს მე-9 ვერსიის ექსპორტის ფორმატს. მე-9 ვერსიის NetFlow-ს ექსპორტის ფორმატის განმასხვავებელი მახასიათებელი არის ის, რომ ის არის შაბლონზე დაფუძნებული. შაბლონები იძლევიან ჩაწერის ფორმატის გაფართოებულ დიზაინს, ფუნქცია, რომელიც უზრუნველყოფს NetFlow მომსახურების მომავალ გაუმჯობესებას, არ საჭიროებს ერთდროულ ცვლილებას ჩაწერის ნაკადის ძირითად ფორმატში. აღსანიშნავია, რომ მოქნილი NetFlow ბევრი საჭირო ბრძანება შეტანილ იქნა Cisco IOS-ის 15.1 ვერსიაში.

ლაბორატორიული სამუშაო - Syslog-ის და NTP-ს კონფიგურაცია

ტოპოლოგია:



შესასრულებელი ამოცანები:

ნაწილი №1: Syslog სერვისის კონფიგურაცია

ნაწილი №2: რეგისტრირებული ღონისძიებების შექმნა

ნაწილი №3: კომუტატორის საათების (Switch Clocks) ხელით მომართვა

ნაწილი №4: NTP სერვისის კონფიგურაცია

ნაწილი №5: დაფიქსირებული სარეგისტრაციო ჟურნალისების შემოწმება

სცენარი

მოცემულ ლაბორატორიულ დავალებაში თქვენ ჩართავთ და გამოიყენებთ **Syslog** და **NTP** სერვისებს, იმისათვის, რომ ქსელის ადმინისტრატორმა უფრო ეფექტურად შეძლოს ქსელის მონიტორინგი.

ნაწილი №1: Syslog სერვისის კონფიგურაცია

პირველი ეტაპი: Syslog სერვისის ჩართვა

ა. დააჭირეთ **Syslog**-ს, შემდეგ გადადით **Services** ჩანართში.

ბ. ჩართეთ **Syslog** სერვისი და გადაადგილეთ ფანჯარა ისე, რომ შეძლოთ საქმიანობის მონიტორინგი.

მეორე ეტაპი: შუალედური მოწყობილობების კონფიგურაცია Syslog სერვისის გამოსაყენებლად

ა. **R1** მარშრუტიზატორის კონფიგურაცია ღონისძიებების ჟურნალის **(log events)** გასაგზავნად **Syslog** სერვერზე.

R1(config)# **logging 10.0.1.254**

ბ. დააკონფიგურეთ **S1** კომუტატორი ღონისძიებების ჟურნალის გასაგზავნად **Syslog** სერვერზე.

გ. დააკონფიგურეთ **S2** კომუტატორი ღონისძიებების ჟურნალის გასაგზავნად **Syslog** სერვერზე.

ნაწილი №2: რეგისტრირებული ღონისძიებების (Logged Events) შექმნა

პირველი ეტაპი: ინტერფეისების სტატუსის შეცვლა ღონისძიებების ჟურნალის შესაქმნელად.

ა. დააკონფიგურეთ **Loopback 0** ინტერფეისი **R1** მარშრუტიზატორზე, შემდეგ გათიშეთ.

ბ. გათიშეთ **PC1** და **PC2**. შემდეგ ხელახლა ჩართეთ

მეორე ეტაპი: Syslog ღონისძიებების შესწავლა

ა. დაათვალიერეთ **Syslog** ღონისძიებები. შენიშვნა: ჩაწერილ იქნა ყველა ღონისძიება; თუმცა დროის შტამპები **(Time stamps)** არასწორია.

ბ. გაასუფთავეთ სარეგისტრაციო ჟურნალები (logs) შემდეგი ნაწილის დაწყებამდე.

ნაწილი №3: კომუტატორის საათების ხელით დაყენება

პირველი ეტაპი: საათების ხელით მომართვა კომუტატორებზე.

S1 და **S2** კომუტატორებზე ხელით მომართეთ საათი მიმდინარე თარიღით და მიახლოებითი დროით. ქვემოთ მოცემულია მაგალითი:

S1# **clock set 11:47:00 July 10 2013**

მეორე ეტაპი: ჟურნალირების დროითი შტამპების (logging timestamp) სერვისის დაყენება კომუტატორებზე.

დააკონფიგურეთ **S1** და **S2** კომუტატორი იმისათვის, რომ გააგზავნონ თავიანთი დროითი შტამპები სარეგისტრაციო ჟურნალებთან ერთად, რომლებსაც გზავნიან **Syslog** სერვერთან.

S1 (config) # **service timestamps log datetime msec**

ნაწილი №4: NTP სერვისის კონფიგურაცია

პირველი ეტაპი: NTP სერვისის ჩართვა

ამ დავალებაში უნდა ვივარაუდოთ რომ **NTP** სერვისი გამართულია საერთო ინტერნეტ სერვერზე. თუ **NTP** სერვერი არის კერძო, შესაძლოა გამოყენებულ იქნას აუთენტიფიკაციაც.

ა. გახსენით **NTP** სერვერის **Services** ჩანართი

ბ. ჩართეთ **NTP** სერვისი და ჩაინიშნეთ ის დრო და თარიღი, რომელიც ნაჩვენებია ეკრანზე.

მეორე ეტაპი: საათის ავტომატური მომართვა მარშრუტიზატორზე

მომართეთ **R1** მარშრუტიზატორის საათი თარიღსა და დროსთან ერთად, **NTP** სერვერის შესაბამისად.

R1 (config) # **ntp server 64.103.224.2**

მესამე ეტაპი: ჟურნალირების დროითი შტამპების ჩართვა მარშრუტიზატორზე

დააკონფიგურეთ **R1** მარშრუტიზატორი იმისათვის, რომ გააგზავნონ თავიანთი დროითი შტამპები სარეგისტრაციო ჟურნალებთან (Log) ერთად, რომლებსაც გზავნიან **Syslog** სერვერთან.

ნაწილი №5: დროით მარკირებული სარეგისტრაციო ჟურნალების (Timestamped Logs) შემოწმება

ა. ხელახლა ჩართეთ და შემდეგ გამორთეთ **Loopback 0** ინტერფეისი **R1** მარშრუტიზატორზე

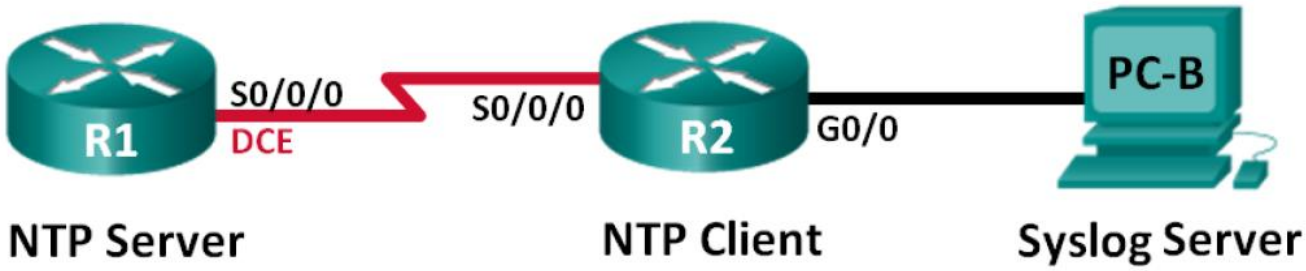
ბ. გათიშეთ **L1** და **L2** ლეპტოპი. შემდეგ ხელახლა ჩართეთ

მეორე ეტაპი: შეისწავლეთ Syslog ღონისძიებები

დაათვალიერეთ **Syslog** ღონისძიებები. **შენიშვნა:** ყველა ღონისძიება ჩაწერილია და დროითი შტამპებიც სწორია, კონფიგურაციის შესაბამისად. **შენიშვნა: R1** იყენებს საათის პარამეტრებს **NTP** სერვერიდან, ხოლო **S1** და **S2** იყენებენ საათის პარამეტრებს თქვენს მიერ **ნაწილი №3**-ის კონფიგურაციის შესაბამისად.

ლაბორატორიული სამუშაო - Syslog-ის და NTP-ს კონფიგურაცია

ტოპოლოგია



მისამართების ცხრილი:

მოწყობილობები	ინტერფეისი	IP მისამართი	ქვექსელის ნიღაბი	ნაგულისხმევი გასასვლელი
R1	S0/0/0 (DCE)	10.1.1.1	255.255.255.252	N/A
R2	S0/0/0	10.1.1.2	255.255.255.252	N/A
	G0/0	172.16.2.1	255.255.255.0	N/A
PC-B	NIC	172.16.2.3	255.255.255.0	172.16.2.1

შესასრულებელი დავალებები:

ნაწილი №1: მოწყობილობის ბაზისური პარამეტრების კონფიგურაცია

ნაწილი №2: NTP-ს კონფიგურაცია

ნაწილი №3: Syslog-ის კონფიგურაცია

ზოგადი ინფორმაცია / სცენარი

Syslog შეტყობინებები, რომლებიც შექმნილია ქსელური მოწყობილობების მიერ, შეიძლება შეგროვდეს და დაარქივდეს Syslog სერვერზე. ინფორმაცია შესაძლოა გამოყენებულ იქნას მონიტორინგის, გამართვისა (**Debugging**) და პრობლემის მოძიებისა და აღმოფხვრის მიზნებისათვის. ადმინისტრატორს შეუძლია აკონტროლოს თუ სად არის შენახული და ნაჩვენები შეტყობინებები. **Syslog** შეტყობინებები შეიძლება იყოს დროით-მარკირებული ქსელური ღონისძიებების თანმიმდევრობის ანალიზისთვის. ამიტომ

აუცილებელია საათის სინქრონიზაცია ქსელური მოწყობილობებს შორის ქსელური დროის პროტოკოლის (**NTP**) სერვერით.

ამ ლაბორატორიულ დავალებაში თქვენ დააკონფიგურებთ **R1** მარშრუტიზატორს, როგორც **NTP** სერვერი და **R2** მარშრუტიზატორს, როგორც **Syslog** და **NTP** კლიენტი. **Syslog** სერვერის აპლიკაცია, **Tftp32d**-ის ან მსგავსი პროგრამის ჩათვლით, გაშვებულ იქნება **PC-B** კომპიუტერზე. გარდა ამისა თქვენ აკონტროლებთ სარეგისტრაციო ჟურნალების შეტყობინებების „სირთულის“ დონეს, რომელიც შეგროვებულია და დაარქივებული **Syslog** სერვერზე.

შენიშვნა: მარშრუტიზატორები, რომლებიც გამოიყენება **CCNA**-ს პრაქტიკული სამუშაოებისთვის, არის **Cisco 1941** ინტეგრირებული სერვისების მარშრუტიზატორები **(ISRs) Cisco IOS Release 15.2(4)M3 (universalk9 image)** ვერსიასთან ერთად. შესაძლოა გამოყენებულ იქნას სხვა მარშრუტიზატორები და **Cisco IOS** ვერსიებიც. მოდელისა და **Cisco IOS** ვერსიის მიხედვით ხელმისაწვდომი ბრძანებები და მიღებული შედეგები შეიძლება იყოს განსხვავებული იმისგან, რაც ნაჩვენებია ამ ლაბორატორიულ სამუშაოში. მიაქციეთ ყურადღება მარშრუტიზატორის ინტერფეისის შემაჯამებელ ცხრილს ამ დავალების ბოლოში, სწორი ინტერფეისის იდენტიფიკატორებისათვის.

შენიშვნა: დარწმუნდით, რომ მარშრუტიზატორები წაიშალა და არ აქვთ საწყისი კონფიგურაციები. თუ არ ხართ დარწმუნებული დაუკავშირდით თქვენს ინსტრუქტორს.

მოთხოვნილი რესურსები:

- 2 მარშრუტიზატორი **(Cisco 1941 with Cisco IOS Release 15.2.(4)M3** უნივერსალი ან მსგავსი იმიჯით)
- ერთი პერსონალური კომპიუტერი (**Windows** ოპერაციული სისტემით ტერმინალის ემულაციის პროგრამასთან ერთად, **Tera Term**-ის ჩათვლით და Syslog პროგრამული უზრუნველყოფით, **tftpd32**-ის ჩათვლით)
- კონსოლის კაბელი **Cisco IOS** მოწყობილობების კონსოლის პორტებით კონფიგურაციისათვის
- ტოპოლოგიაზე ნაჩვენები **Ethernet** და სერიალური კაბელები

ნაწილი №1: მოწყობილობილობის ბაზისური პარამეტრების კონფიგურაცია

პირველ ნაწილში თქვენ მომართავთ ქსელის ტოპოლოგიას და დააკონფიგურებთ ბაზისურ პარამეტრებს, ინტერფეისის **IP** მისამართების, მარშრუტიზაციის, მოწყობილობებთან წვდომის და პაროლების ჩათვლით.

პირველი ეტაპი: კაბელების შეერთება ისე, როგორც ნაჩვენებია ტოპოლოგიაზე

მეორე ეტაპი: მარშრუტიზატორის ინიციალიზაცია და ხელახლა ჩატვირთვა, აუცილებლობის შემთხვევაში

მესამე ეტაპი: თითოული მარშრუტიზატორის ბაზისური პარამეტრების კონფიგურაცია

ა. გათიშეთ **DNS lookup**

ბ. დააკონფიგურეთ მოწყობილობის სახელი

გ. დაშიფრეთ ღია ტექსტის პაროლები

დ. შექმენით დღის შეტყობინების (**MOTD**) გამაფრთხილებელი ბანერი მომხმარებლებისათვის, რომლებიც ახორციელებენ არაავტორიზებულ წვდომას.

ე. დააყენეთ **class**, როგორც დაშიფრული პრივილეგირებული **EXEC** რეჟიმის პაროლი

ვ. დანიშნეთ **cisco** როგორც კონსოლის და **vty** პაროლი და ჩართეთ შესვლა (**login**)

ზ. მომართეთ კონსოლის ჟურნალირება სინქრონული რეჟიმისთვის

თ. მისამართების ცხრილის მიხედვით მომართეთ **IP** მისამართები სერიალური და გიგაბიტური **Ethernet** ინტერფეისებისათვის და გააქტიურეთ ფიზიკური ინტერფეისები.

ი. დააყენეთ ტაქტური სიხშირე **128000**-ზე **DCE** სერიალ ინტერფეისისთვის

მეოთხე ეტაპი: მარშრუტიზაციისკონფიგურაცია

ჩართეთ **single-area OSPF** მარშრუტიზატორებზე **process ID 1**-ით. დაამატეთ ყველა ქსელი **OSPF** პროცესში **area 0**-სთვის.

მეხუთე ეტაპი: PC-B კომპიუტერის კონფიგურაცია

მისამართების ცხრილის მიხედვით დააკონფიგურეთ **PC-B** კომპიუტერის **IP** მისამართი და ნაგულისხმევი გასასვლელი.

მეექვსე ეტაპი: სრული ციკლის კავშირის შემოწმება

შეამოწმეთ თითოეულ მოწყობილობას წარმატებით შეუძლია თუ არა **ping**-ის გაშვება ქსელის სხვა მოწყობილობებთან. თუ არა მოძებნეთ და აღმოფხვერით პრობლემა, სანამ არ მიიღწევა სრულკავშირიანი ციკლი.

მეშვიდე ეტაპი: შეინახეთ გაშვებული კონფიგურაცია საწყის კონფიგურაციაში

ნაწილი №2: NTP-ს კონფიგურაცია

მეორე ნაწილში თქვენ დააკონფიგურებთ **R1** მარშრუტიზატორს, როგორც **NTP** სერვერს და **R2** მარშრუტიზატორს, როგორც **R1**-ის **NTP** კლიენტს. სინქრონიზებული დრო აუცილებელია **Syslog** და **debug** ფუნქციებისათვის. თუ დრო არ არის სინქრონული, რთულია იმის განსაზღვრა რომელმა ქსელურმა ღონისძიებამ გამოიწვია შეტყობინება.

პირველი ეტაპი: მიმდინარე თარიღის ჩვენება

გაუშვით **show clock** ბრძანება **R1** მარშრუტიზატორზე მიმდინარე თარიღის საჩვენებლად.

R1# **show clock**

*12:30:06.147 UTC Tue May 14 2013

ჩაიწერეთ მიმდინარე თარიღის ინფორმაცია ქვემოთ მოცემულ ცხრილში

თარიღი	
დრო	
სასაათო ზონა	

მეორე ეტაპი: დროის დაყენება

გამოიყენეთ clock set ბრძანება R1 მარშრუტიზატორზე დროის მოსამართად. ქვემოთ მოცემული არის თარიღისა და დროის პარამეტრების მაგალითი.

R1# **clock set 9:39:00 05 july 2013**

R1#

*Jul 5 09:39:00.000: %SYS-6-CLOCKUPDATE: System clock has been updated from 12:30:54 UTC Tue May 14 2013 to 09:39:00 UTC Fri Jul 5 2013, configured from console by console.

შენიშვნა: დრო შეიძლება ასევე დაყენებულ იქნას **clock timezone** ბრძანების გამოყენებით გლობალური კონფიგურაციის რეჟიმში. ამ ბრძანებასთან მიმართებაში მეტი ინფორმაციისთვის, მოძებნეთ **clock timezone** ბრძანება www.cisco.com საიტზე, თქვენი რეგიონის ზონის განსასაზღვრად.

მესამე ეტაპი: NTP მმართველის კონფიგურაცია

დააკონფიგურეთ **R1** როგორც **NTP** მმართველი **ntp master** *<შრის (stratum)-ნომერი>* ბრძანების გამოყენებით გლობალური კონფიგურაციის რეჟიმში. შრის ნომერი (**stratum number**) მიუთითებს **NTP** გადასვლების (**hops**) რიცხვს აუთენტური დროის წყაროდან. მოცემულ ლაბორატორიულ დავალებაში რიცხვი 5 არის ამ **NTP** სერვერის შრის დონე.

R1 (config) # **ntp master 5**

მეოთხე ეტაპი: NTP კლიენტის კონფიგურაცია

ა. გაუშვით **show clock** ბრძანება **R2** მარშრუტიზატორზე. ჩაიწერეთ **R2**-ზე ნაჩვენები მიმდინარე თარიღი, ქვემოთ მოცემულ ცხრილში.

თარიღი	
დრო	
სასაათო ზონა	

ბ. დააკონფიგურეთ **R2** მარშრუტიზატორი, როგორც **NTP** კლიენტი. გამოიყენეთ **ntp server** ბრძანება **NTP** სერვერის **IP** მისამართის ან ჰოსტის სახელის მისათითებლად. **ntp update-calendar** ბრძანება პერიოდულად განაახლებს კალენდარს **NTP** დროით.

R2 (config) # **ntp server 10.1.1.1**

R2 (config) # **ntp update-calendar**

მეხუთე ეტაპი: NTP კონფიგურაციის შემოწმება

ა. გამოიყენეთ **show ntp associations** ბრძანება იმის შესამოწმებლად აქვს თუ არა **R2**-ს **NTP** კავშირი **R1**-თან.

R2# **show ntp associations**

address	ref clock	st	when	poll	reach	delay	offset	disp
*~10.1.1.1	127.127.1.1	5	11	64	177	11.312	-0.018	4.298

*sys. peer, #selected, +candidate, -outlyer, x falseticker, ~configured

ბ. გაუშვით **show clock** ბრძანება **R1** და **R2** მარშრუტიზატორებზე, დროის მარკირების შესადარებლად.

შენიშვნა: რამდენიმე წუთია საჭირო **R2** მარშრუტიზატორის დროითი მარკირების სინქრონიზაციისთვის **R1**-თან.

R1# **show clock**

09:43:32.799 UTC Fri Jul 5 2013

R2# **show clock**

09:43:37.122 UTC Fri Jul 5 2013

ნაწილი №3: Syslog-ის კონფიგურაცია

ქსელური მოწყობილობებიდან **Syslog** შეტყობინებები შეიძლება შეგროვდეს და დაარქივდეს **syslog** სერვერზე. მოცემულ ლაბორატორიულ სამუშაოში, **Tftpd32** პროგრამული უზრუნველყოფა გამოიყენება როგორც **syslog** სერვერი. ქსელის ადმინისტრატორს შეუძლია იმ შეტყობინებების ტიპების კონტროლი, რომლებიც შეიძლება გაიგზავნოს **syslog** სერვერზე.

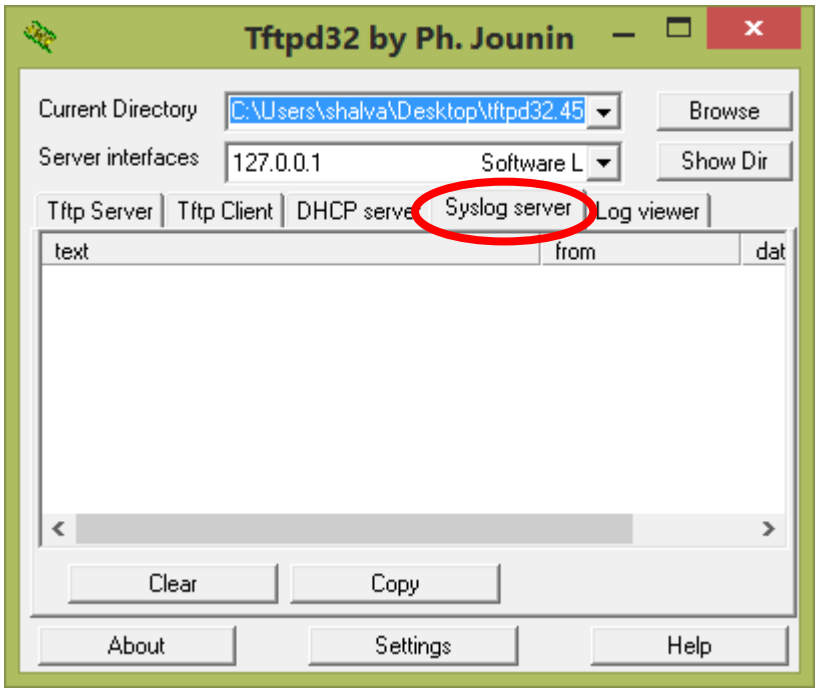
პირველი ეტაპი: (არააუცილებელი) syslog სერვერის ინსტალაცია

თუ **syslog** სერვერი ჯერ კიდევ არ არის ინსტალირებულის კომპიუტერზე, გადმოწერეთ და დააინსტალირეთ კომპიუტერზე **syslog** სერვერის ბოლო ვერსია, ისეთი როგორიცაა **Tftpd32**. **Tftpd32**-ის ბოლო ვერსია შეგიძლიათ გადმოწეროთ მოცემული ბმულიდან:

http://tftpd32.jounin.net/

მეორე ეტაპი: syslog სერვერის გაშვება PC-B კომპიუტერზე

Tftpd32 აპლიკაციის გაშვების შემდეგ, დააჭირეთ **syslog server** ჩანართს.



მესამე ეტაპი: შეამოწმეთ ჩართულია თუ არა დროითი მარკირების სერვისი R2 მარშრუტიზატორზე.

გამოიყენეთ **show run** ბრძანება იმის შესამოწმებლად, რომ დროითი მარკირების სერვისი ჟურნალირებისთვის (**Logging**) ჩართულია თუ არა **R2** მარშრუტიზატორზე.

R2# **show run | include timestamp**

service timestamps debug datetime msec

service timestamps log datetime msec

თუ დროითი მარკირების სერვისი არაა ჩართული, გამოიყენეთ ქვემოთ მოცემული ბრძანება მის ჩასართავად.

R2 (config) # **service timestamps log datetime msec**

მეოთხე ეტაპი: R2 მარშრუტიზატორის ჟურნალირების შეტყობინებების კონფიგურაცია syslog სერვერზე.

დააკონფიგურეთ **R2** მარშრუტიზატორი **syslog** შეტყობინებების გასაგზავნად syslog სერვერზე, **PC-B. PC-B syslog** სერვერის **IP** მისამართი არის 172.16.2.3.

R2(config)# **logging host 172.16.2.3**

მეხუთე ეტაპი: ნაგულისხმევი ჟურნალირების პარამეტრების ჩვენება

გამოიყენეთ **show logging** ბრძანება, ნაგულისხმევი ჟურნალირების პარამეტრების გამოსატანად.

R2# **show logging**

Syslog logging: enabled (0 messages dropped, 2 messages rate-limited, 0 flushes, 0 overruns, xml disabled, filtering disabled)

No Activity Mesaage Discriminator.

No Inactive Message Discriminator.

Console logging: level debugging, 47 messages logged, xml disabled,
filtering disabled

Monitor logging: level debugging, 0 messages logged, xml disabled,
filtering disabled

Buffer logging: level debugging, 47 messages logged, xml disabled,
filtering disabled

Exception Logging: size (4096 bytes)

Count and timestamp logging messages: disabled

Persistent logging: disabled

No active filter modules.

Trap logging: level informational, 49 message lines logged

Logging to 172.16.2.3 (udp port 514, audit disabled,
link up),
6 message lines logged,

0 message lines rate-limited,

0 message lines dropped-by-MD,

xml disabled, sequence number disabled

filtering disabled

Logging Source-Interface: VRF Name:

რა არის **syslog** სერვერის **IP** მისამართი? _____

რომელ პორტს და პროტოკოლს იყენებს **syslog**-ი?_____

რომელ დონეზეა ჩართული **trap** ჟურნალირება (**Logging**)? _____

მეექვსე ეტაპი: კონფიგურაცია და ჟურნალირების „სირთულის“ დონეების დაკვირვება R2 მარშრუტიზატორზე.

ა. გამოიყენეთ **logging trap ?** ბრძანება სხვადასხვა **trap** დონეების ხელმისაწვდომობის განსასაზღვრად. როდესაც ვაკონფიგურებთ დონეს, **syslog** სერვერზე გაგზავნილი შეტყობინებები არის **trap** დონეზე და ნებისმიერ უფრო დაბალ დონეზე მომართული.

R2 (config) # **logging trap ?**

<0-7>	Logging severity level	
alerts	Immediate action needed	{severity=1}
critical	Critical conditions	{severity=2}
debugging	Debugging messages	{severity=7}
emergencies	System is unusable	{severity=0}
errors	Error conditions	{severity=3}
informational	Informational messages	{severity=6}
notifications	Normal but significant conditions	{severity=5}
warnings	Warning conditions	{severity=4}
<cr>		

თუ გაშვებულ იქნა **logging trap warnings** ბრძანება, როგორი სირთულის დონის შეტყობინებები იქნება ჟურნალირებული?_____

_____

ბ. ჟურნალირების (**Logging**) სირთულის დონის შეცვლა 4-მდე.

R2 (config) # **logging trap warnings**

or

R2 (config) # **logging trap 4**

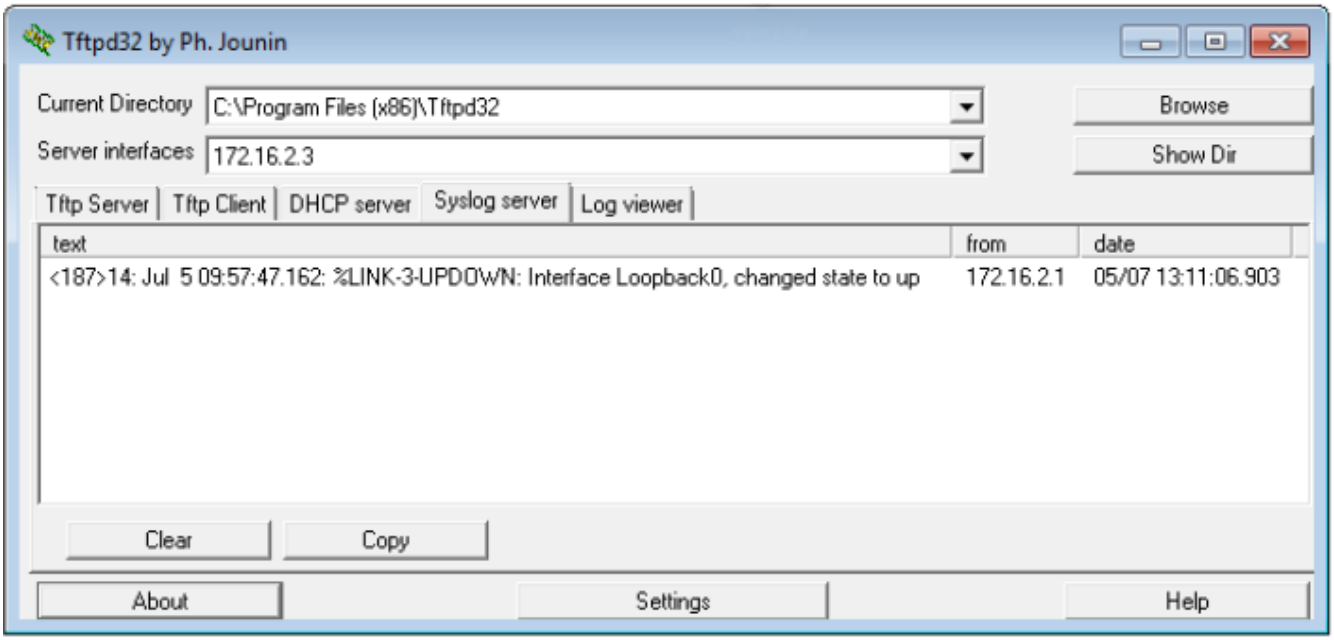
გ. **Loopback0** ინტერფეისის შექმნა **R2** მარშრუტიზატორზე და სარეგისტრაციო ჟურნალის შეტყობინებებზე დაკვირვება როგორც ტერმინალის ფანჯარაზე, ისე **PC-B** კომპიუტერის **syslog** სერვერის ფანჯარაზე.

R2 (config) # **interface lo 0**

R2 (config-if)#

Jul 5 09:57:47.162: %LINK-3-UPDOWN: Interface Loopback0, changed state to up

Jul 5 09:57:48.162: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up



დ. გააუქმეთ **Loopback0** ინტერფეისი **R2** მარშრუტიზატორზე და დააკვირდით სარეგისტრაციო ჟურნალის შეტყობინებებს.

R2 (config-if)# **no interface lo 0**

R2 (config) #

Jul 5 10:02:58.910: %LINK-5-CHANGED: Interface Loopback0, changed state to administratively down

Jul 5 10:02:59.910: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to down

სირთულის მეოთხე დონეზე, არის რაიმე სარეგისტრაციო ჟურნალის შეტყობინებები **syslog** სერვერზე? თუ რაიმე სარეგისტრაციო ჟურნალის შეტყობინება გამოჩნდა, ახსენით რა და რატომ გამოჩნდა. _____

_____

_____

ე. შეცვალეთ ჟურნალირების სირთულის დონე 6-მდე.

R2 (config) # **logging trap informational**

or

R2 (config) # **logging trap 6**

ვ. წაშალეთ syslog ჩანაწერები **PC-B** კომპიუტერზე. დააჭირეთ **Clear** ღილაკს **Tftpd32** დიალოგურ ფანჯარაში.

ზ. შექმენით **Loopback 1** ინტერფეისი **R2** მარშრუტიზატორზე.

R2 (config)# **interface lo 1**

Jul 5 10:05:46.650: %LINK-3-UPDOWN: Interface Loopback1, changed state to up

Jul 5 10:05:47.650: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback1, changed state to up

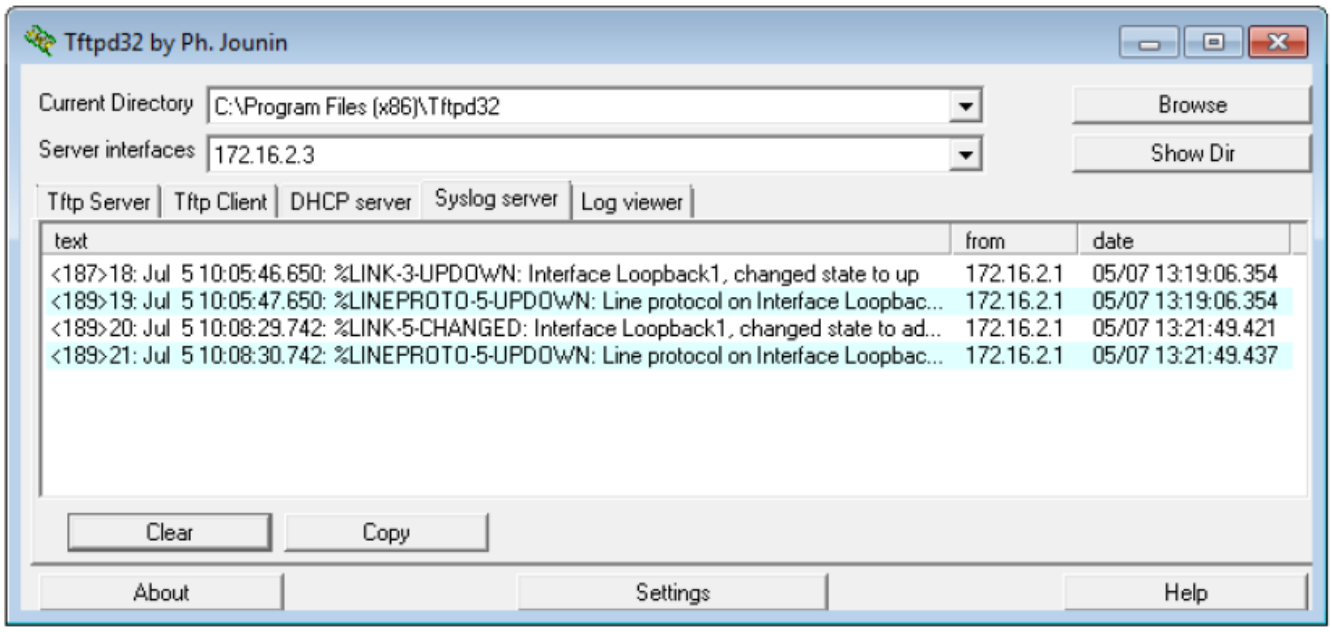
თ. გააუქმეთ **Loopback 1** ინტერფეისი **R2** მარშრუტიზატორზე

R2 (config-if) # **no interface lo 1**

R2 (config-if) #

Jul 5 10:08:29.742: %LINK-5-CHANGED: Interface Loopback1, changed state to administratively down

Jul 5 10:08:30.742: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback1, changed state to down



ი. დააკვირდით **syslog** სერვერის შედეგს. შეუდარეთ მოცემული შედეგი **trapping** მეოთხე დონის შედეგებს. თქვენი დასკვნა? _____

_____

_____

_____

ასახვა (Reflection)

Syslog-სთვის რა არის პრობლემა, სირთულის დონის პარამეტრის ძალიან მაღალზე (დაბალი დონის რიცხვი) თუ ძალიან დაბალზე დაყენება (მაღალი დონის რიცხვი).

_____

_____

_____

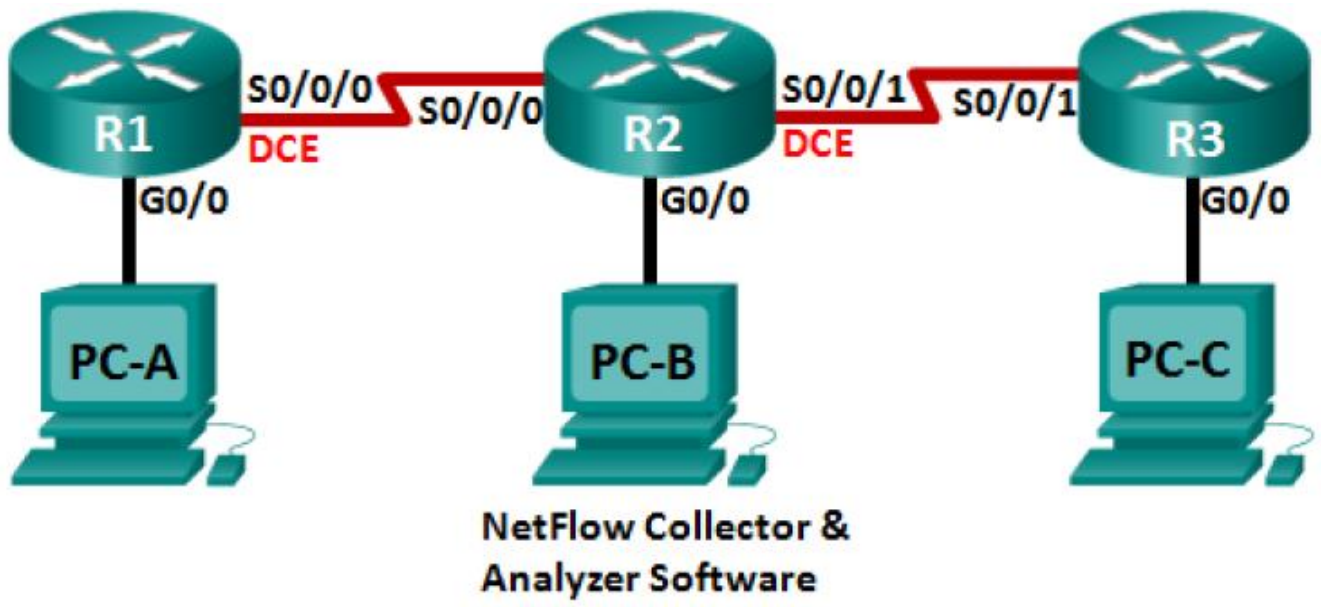
მარშრუტიზატორის ინტერფეისის შემაჯამებელი ცხრილი:

მარშრუტიზატორის ინტერფეისის შეჯამება				
მარშრუტიზატორის მოდელი	Ethernet interface №1	Ethernet interface №2	Serial interface №1	Serial interface №2
1800	FastEthernet 0/0 (F0/0)	FastEthernet 0/1 (F0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
1900	GigabitEthernet 0/0 (G0/0)	GigabitEthernet 0/1 (G0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
2801	FastEthernet 0/0 (F0/0)	FastEthernet 0/1 (F0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
2811	FastEthernet 0/0 (F0/0)	FastEthernet 0/1 (F0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
2900	GigabitEthernet 0/0 (G0/0)	GigabitEthernet 0/1 (G0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)

შენიშვნა: თუ გინდათ მოძებნოთ თუ როგორაა მარშრუტიზატორი კონფიგურირებული, დაათვალიერეთ ინტერფეისები რათა მოახდინოთ მარშრუტიზატორის იდენტიფიკაცია და დაადგინოთ რამდენი ინტერფეისი აქვს მას. აქ არ არის იმის საშუალება, რომ ჩამოიწეროს თითოეული მარშრუტიზატორის კონფიგურაციების კომბინაციები. მოცემული ცხრილი მოიცავს მოწყობილობის **Ethernet** და **Serial** ინტერფეისების შესაძლო კომბინაციების იდენტიფიკატორებს. ეს ცხრილი არ მოიცავს ნებისმიერი სხვა ტიპის ინტერფეისს, რადგან კონკრეტული მარშრუტიზატორი შეიძლება შეიცავდეს მხოლოდ ერთს. ამის მაგალითი შეიძლება იყოს **ISDN BRI** ინტერფეისი. ფრჩხილებში მოცემული ცვლადი არის ლეგალური აბრევიატურა, რომელიც შეიძლება გამოყენებულ იქნას **Cisco IOS** ბრძანებებში, ინტერფეისის წარმოსადგენად.

ლაბორატორიული სამუშაო - Netflow მონაცემების შეგროვება და ანალიზი

ტოპოლოგია:



მისამართების ცხრილი:

მოწყობილობა	ინტერფეისი	IP მისამართი	ნაგულისხმევი გასასვლელი
R1	G0/0	192.168.1.1/24	N/A
	S0/0/0 (DCE)	192.168.12.1/30	N/A
R2	G0/0	192.168.2.1/24	N/A
	S0/0/0	192.168.12.2/30	N/A
	S0/0/1 (DCE)	192.168.23.1/30	N/A
R3	G0/0	192.168.3.1/24	N/A
	S0/0/1	192.168.23.2/30	N/A
PC-A	NIC	192.168.1.3	192.168.1.1
PC-B	NIC	192.168.2.3	192.168.2.1
PC-C	NIC	192.168.3.3	192.168.3.1

შესასრულებელი ამოცანები:

ნაწილი №1: ქსელის აწყობა და მოწყობილობის პარამეტრების ბაზისური კონფიგურაცია

ნაწილი №2: NetFlow-ს კონფიგურაცია მარშრუტიზატორზე

ნაწილი №3: NetFlow-ს ანალიზი CLI-ს გამოყენებით

ნაწილი №4: NetFlow კოლექტორის შესწავლა და ანალიზატორი პროგრამული უზრუნველყოფა

ზოგადი ინფორმაცია/ სცენარი

NetFlow არის **Cisco IOS** ტექნოლოგია, რომელიც იძლევა **Cisco** მარშრუტიზატორიდან ან მრავალდონიანი კომუტატორიდან წამოსული პაკეტების სტატისტიკას. **NetFlow** იძლევა ქსელისა და უსაფრთხოების მონიტორინგის, ქსელის დაგეგმვის, ტრაფიკის ანალიზის და **IP** ანგარიშის საშუალებას. მნიშვნელოვანია, რომ არ აურიოთ **NetFlow**-ს დანიშნულება და შედეგები, პაკეტების დაჭერის ტექნიკურ და პროგრამულ უზრუნველყოფასთან. პაკეტების დამჭერი იწერს ყველა შესაძლო ინფორმაციას, რომლებიც გამოდის ან შედის ქსელურ მოწყობილობაზე, შემდგომი ანალიზისათვის, **NetFlow**-ს კონკრეტული დავალებებია სტატისტიკური ინფორმაცია.

მოქნილი **NetFlow** არის **NetFlow**-ს ბოლო ტექნოლოგია. ის გაუმჯობესებულია ორიგინალ **NetFlow**-სთან შედარებით, რადგან დამატებული აქვს ტრაფიკის ანალიზის პარამეტრების მომართვის შესაძლებლობა. მოქნილი **NetFlow** იყენებს მე-9 ვერსიის ექსპორტის ფორმატს. დაწყებული **Cisco IOS Relase 15.1**-დან, მოქნილი **NetFlow**-ს მრავალი საჭირო ბრძანებაა მხარდაჭერილი.

მოცემულ ლაბორატორიულ სამუშაოში თქვენ დააკონფიგურებთ **NetFlow**-ს, როგორც შემომავალი ისე გამავალი პაკეტების დასაჭერად. თქვენ გამოიყენებთ **show** ბრძანებებს, იმის შესამოწმებლად, რომ **NetFlow** ფუნქციონირებს და იღებს სტატისტიკურ ინფორმაციას. თქვენ ასევე შეისწავლით **NetFlow** შეგროვებისა და ანალიზის პროგრამული უზრუნველყოფის ხელმისაწვდომ პარამეტრებს.

შენიშვნა: მარშრუტიზატორები, რომლებიც გამოიყენება **CCNA**-ს პრაქტიკული სამუშაოებისთვის, არის **Cisco 1941** ინტეგრირებული სერვისების მარშრუტიზატორები **(ISRs) Cisco IOS Release 15.2(4)M3 (universalk9 image)** ვერსიასთან ერთად. შესაძლოა გამოყენებულ იქნას სხვა მარშრუტიზატორები და **Cisco IOS** ვერსიებიც. მოდელისა და **Cisco IOS** ვერსიის მიხედვით ხელმისაწვდომი ბრძანებები და მიღებული შედეგები შეიძლება იყოს განსხვავებული იმისგან, რაც ნაჩვენებია ამ ლაბორატორიულ სამუშაოში. მიაქციეთ ყურადღება მარშრუტიზატორის ინტერფეისის შემაჯამებელ ცხრილს ამ დავალების ბოლოში, სწორი ინტერფეისის იდენტიფიკატორებისათვის.

შენიშვნა: დარწმუნდით, რომ მარშრუტიზატორები წაიშალა და არ აქვთ საწყისი კონფიგურაციები. თუ არ ხართ დარწმუნებული დაუკავშირდით თქვენს ინსტრუქტორს.

მოთხოვნილი რესურსები:

- 3 მარშრუტიზატორი **(Cisco 1941 with Cisco IOS Release 15.2.(4)M3** უნივერსალი ან მსგავსი იმიჯით)
- სამი პერსონალური კომპიუტერი (**Windows** ოპერაციული სისტემით ტერმინალის ემულაციის პროგრამასთან ერთად, **Tera Term**-ის ჩათვლით)
- კონსოლის კაბელი **Cisco IOS** მოწყობილობების კონსოლის პორტებით კონფიგურაციისათვის
- ტოპოლოგიაზე ნაჩვენები **Ethernet** და სერიალური კაბელები

ნაწილი №1: მოწყობილობილობის ბაზისური პარამეტრების კონფიგურაცია

პირველ ნაწილში თქვენ მომართავთ ქსელის ტოპოლოგიას და დააკონფიგურებთ ბაზისურ პარამეტრებს **PC** ჰოსტებსა და მარშრუტიზატორებზე.

პირველი ეტაპი: კაბელების შეერთება ისე, როგორც ნაჩვენებია ტოპოლოგიაზე

მეორე ეტაპი: მარშრუტიზატორის ინიციალიზაცია და ხელახლა ჩატვირთვა, აუცილებლობის შემთხვევაში

მესამე ეტაპი: თითოული მარშრუტიზატორის ბაზისური პარამეტრების კონფიგურაცია

ა. გათიშეთ **DNS lookup**

ბ. დააკონფიგურეთ მოწყობილობის სახელი ისე როგორც ნაჩვენებია ტოპოლოგიაზე

გ. დააყენეთ **class**, როგორც დაშიფრული პრივილეგირებული **EXEC** რეჟიმის პაროლი

დ. დანიშნეთ **cisco** როგორც კონსოლის და **vty** პაროლი და ჩართეთ შესვლა (**login**)

ე. დაშიფრეთ ღია ტექსტის პაროლები

ვ. შექმენით დღის შეტყობინების (**MOTD**) გამაფრთხილებელი ბანერი მომხმარებლებისათვის, რომლებიც ახორციელებენ არაავტორიზებულ წვდომას.

ზ. მომართეთ **logging Synchronous** კონსოლის ხაზისთვის.

თ. დააყენეთ ტაქტური სიხშირე **128000**-ზე **DCE** სერიალ ინტერფეისისთვის

ი. დააკონფიგურეთ **IP** მისამართები მისამართების ცხრილის მიხედვით

კ. მომართეთ **OSPF Process ID 1**-ის გამოყენებით და შეატყობინეთ ყველა ქსელს. Ethernet ინტერფეისები უნდა იყოს პასიური

ლ. შექმენით ლოკალური მონაცემთა ბაზა **R3** მარშრუტიზატორზე, admin მომხმარებლის სახელითა და cisco პაროლით, პრივილეგირებულ დონე 15-ზე.

მ. **R3** მარშრუტიზატორზე ჩართეთ **HTTP** სერვისი და მოახდინეთ **HTTP** მომხმარებლების აუთენტიფიკაცია ლოკალური ბაზის გამოყენებით.

ნ. გადაიტანეთ გაშვებული კონფიგურაციის ასლი საწყის კონფიგურაციაში.

მეოთხე ეტაპი: PC ჰოსტების კონფიგურაცია

მეხუთე ეტაპი: ერთმანეთთან კავშირის შემოწმება

ყველა მოწყობილობას უნდა შეეძლოს ტოპოლოგიის სხვა მოწყობილობების დაპინგვა (**Ping**). აუცილებელია პრობლემის მოძიება და აღმოფხვრა, სანამ არ მიიღწევა სრული ციკლის კავშირი.

შენიშვნა: შეიძლება აუცილებელი იყოს პერსონალური კომპიუტერის ფაიერვოლის გათიშვა კომპიუტერებს შორის წარმატებული **ping**-სთვის.

ნაწილი №2: NetFlow-ს კონფიგურაცია მარშრუტიზატორზე

მეორე ნაწილში თქვენ დააკონფიგურებთ **NetFlow**-ს **R2** მარშრუტზატორზე. **NetFlow** გამოიჭერს ყველა შემომავალ და გამავალ ტრაფიკს **R2** მარშრუტიზატორის სერიალურ ინტერფეისებზე და გაიტანს მონაცემებს **NetFlow** კოლექტორში, **PC-B**. **NetFlow** კოლექტორში ექსპორტირებისათვის გამოყენებული იქნება მოქნილი **NetFlow**-ს მე-9 ვერსია.

პირველი ეტაპი: NetFlow გამოჭერის (Capture) კონფიგურაცია

მომართეთ **NetFlow** მონაცემთა დაჭერა ორივე სერიალურ ინტერფეისზე. დააკავეთ მონაცემები შემომავალი და გამავალი პაკეტებიდან.

R2 (config)# **interface s0/0/0**

R2 (config-if) # **ip flow ingress**

R2 (config-if)# **ip flow egress**

R2 (config-if) # **interface s0/0/1**

R2 (config-if) # **ip flow ingress**

R2 (config-if) # **ip flow egress**

მეორე ეტაპი: NetFlow-ს მონაცემთა ექსპორტის კონფიგურაცია

გამოიყენეთ **ip flow-export destination** ბრძანება **NetFlow** კოლექტორის **IP** მისამართისა და **UDP** პორტის იდენტიფიცირებისათვის, რომლითაც მარშრუტიზატორს შეუძლია **NetFlow**-ს მონაცემების ექსპორტი. ამ კონფიგურაციისათვის გამოყენებულ იქნება **9996 UDP** პორტის ნომერი.

R2 (config) # **ip flow-export destination 192.168.2.3 9996**

მესამე ეტაპი: NetFlow-ს ექსპორტის ვერსიის კონფიგურაცია

Cisco მარშრუტიზარტორები, რომელზეც გაშვებულია **IOS 15.1**, მხარს უჭერენ **NetFLow**-ს 1, 5 და 9 ვერსიებს. მე-9 ვერსია არის მოქნილი მონაცემთა ექსპორტის ფორმატი, მაგრამ არ არის თავსებადი წინამორბედ ძველ ვერსიებთან. გამოიყენეთ ip flow-export version ბრძანება NetFlow-ს ვერსიის მოსამართად.

R2 (config) # **ip flow-export version 9**

მეოთხე ეტაპი: NetFlow-ს კონფიგურაციის შემოწმება

ა. გაუშვით **show ip flow interface** ბრძანება **NetFlow** დაჭერის ინტერფეისის ინფორმაციის მიმოხილვისათვის.

R2 # **show ip flow interface**

Serial0/0/0

ip flow ingress

ip flow egress

Serial0/0/1

ip flow ingress

ip flow egress

ბ. გაუშვით **show ip flow export** ბრძანება, **NetFlow**-ს მონაცემთა ექსპორტის ინფორმაციის მიმოხილვისათვის.

R2# **show ip flow export**

Flow export v9 is enabled for main cache

Export source and destination details :

VRF ID : Default

Destination(1) 192.168.2.3 (9996)

Version 9 flow records

388 flow exported in 63 udp datagrams

0 flows failed due to lack of export packet

0 export packets were sent up to process level

0 export packets were dropped due to no fib

0 export packets were dropped due to adjacency issues

0 export packets were droped due to fragmentation failures

0 export packets were dropped due to encapsulation fixup failures

ნაწილი №3: NetFlow-ს ანალიზი CLI-ს გამოყენებით

მესამე ნაწილში თქვენ შექმნით მონაცემთა ტრაფიკს **R1** და **R3** მარშრუტიზატორებს შორის, NetFlow-ს ტექნოლოგიაზე დაკვირვებისათვის.

პირველი ეტაპი: მონაცემთა ტრაფიკის შექმნა R1 და R3 მარშრუტიზატორებს შორის.

ა. დაკავშირდით **Telnet**-ით **R1**-დან **R3**-ში 1**92.168.3.1 IP** მისამართის გამოყენებით. შეიყვანეთ პაროლი cisco მომხმარებლის **EXEC** რეჟიმში შესასვლელად. შეიყვანეთ პაროლი **class**, გლობალური **EXEC** რეჟიმის ჩასართავად. გაუშვით **show run** ბრძანება რაიმე **Telnet** ტრაფიკის შესაქმნელად. დატოვეთ **Telnet** სესია აქტიური.

ბ. **R3** მარშრუტიზატორიდან გაუშვით **ping 192.168.1.1 repeat 1000** ბრძანება, **R1 G0/0** ინტერფეისის „დასაპინგად“. ეს შექმნის **ICMP** ტრაფიკს **R2** მარშრუტიზატორზე.

გ. **PC-A**-დან დაათვალიერეთ **R3** მარშტურიზატორი **192.168.3.1 IP** მისამართის გამოყენებით. შედით როგორც **admin** მომხმარებელი **cisco** პაროლთან ერთად. დატოვეთ ბრაუზერი გახსნილი, მას შემდეგ რაც შეხვალთ **R3** მარშრუტიზატორში.

შენიშვნა: დარწმუნდით რომ **pop-up blocker** გათიშულია თქვენს ბრაუზერში.

მეორე ეტაპი: NetFlow-ს შემაჯამებელი ანგარიშის სტატისტიკის ჩვენება.

R2 მარშრუტიზატორზე გაუშვით **show ip cache flow** ბრძანება, **NetFlow** შემაჯამებელი მონაცემების ცვლილებების სანახავად, პაკეტის ზომის განაწილების, **IP** ნაკადის ინფორმაციის, დაჭერილი პროტოკოლების და ინტერფეისის აქტივობის ჩათვლით. მიაქციეთ ყურადღება, რომ პროტოკოლები ახლა ნაჩვენებია შემაჯამებელ მონაცემებში.

R2# **show ip cache flow**

IP packet size distribution (5727 total ackets):

1-32	64	96	128	160	192	224	256	288	320	352	384	416	448	480
.000	.147	.018	.700	.000	.001	.001	.001	.001	.011	.009	.001	.002	.000	.001

512	544	576	1024	1536	2048	2560	3072	3584	4096	4608
.001	.001	.097	.000	.000	.000	.000	.000	.000	.000	.000

IP Flow Switching cache, 278544 bytes

2 active, 4094 inactive, 114 added

1546 ager polls, 0 flow alloc failures

Active flows timeout in 15 seconds

IP Sub Flow Cache, 34056 bytes

0 active, 1024 inactive, 112 added, 112 added to flow

0 alloc failures, 0 force free

1 chunk, 1 chunk added

last clearing of statistics 00:07;35

Protocol	total	Flows	Packets	Bytes	Packets	Active (sec)	Idle (sec)
---------	Flows	/sec	/flow	/pkt	/Sec	/Flow	/Flow
TCP-Telnet	4	0.0	27	43	0.2	5.0	15.7
TCP-WWW	104	0.2	14	275	3.4	2.1	1.5
ICMP	4	0.0	1000	100	8.8	27.9	15.4

SrcIf	SrcIPaddress	DstIf	DstIPaddress	Pr SrcP DstP	Pkts
Total:	112	0.2	50 146 12.5	3.1	2.5
SrcIf	SrcIPaddress	DstIf	DstIPaddress	pr SrcP DstP	Pkts
Se0/0/0	192.168.12.1	Null	224.0.0.5	59 0000 0000	43
Se0/0/1	192.168.23.2	Null	224.0.0.5	59 0000 0000	40

მესამე ეტაპი: Telnet და ბრაუზერის სესიების დახურვა

ა. გაუშვით **exit** ბრძანება **R1** მარშრუტიზატორზე, რათა გაითიშოს **R3** მარშრუტიზატორის **Telnet** სესიიდან გასათიშად.

ბ. დახურეთ ბრაუზერის სესია **PC-A** კომპიუტერზე

მეოთხე ეტაპი: NetFlow სააღრიცხვო სტატისტიკის გასუფთავება

ა. **R2** მარშრუტიზატორზე გაუშვით **clear ip flow stats** ბრძანება, **NetFlow** სააღრიცხვო სტატისტიკის გასასუფთავებლად.

R2# **clear ip flow stats**

ბ. ხელახლა გაუშვით **show ip cache flow** ბრძანება რათა შემოწმდეს **Netflow** სააღრცხვო სტატისტიკა განულდა თუ არა. მიაქციეთ ყურადღება, რომ თუ თქვენ აღარ შექმნით მონაცემებს **R2**-ის გავლით, მაშინ მონაცემები არჩეული იქნება **NetFLow**-ს მიერ. ქვემოთ მოცემულ მაგალითში, ადრესატის მისამართი ამ ტრაფიკისათვის არის მულიტიკასტმისამართი 224.0.0.5, ან **OSPF LSA** მონაცემი.

R2# **show ip cache flow**

IP packet size distribution (124 total packets) :

1-32	64	96	128	160	192	224	256	288	320	352	384	416	448	480
.000	.000	1.00	.000	.000	.000	.000	.000	.000	.000	.000	.000	.000	.000	.000

512	544	576	1024	1536	2048	2560	3072	3584	4096	4608
.000	.000	.000	.000	.000	.000	.000	.000	.000	.000	.000

IP Flow Switching cache, 278544 bytes

2 active, 4094 inactive, 2 added

1172 ager polls, 0 flow allow alloc failures

Active flows timeout in 30 minutes

Inactive flows timeout in 15 seconds

IP Sub Flow Cache, 34056 bytes

2 active, 1022 inactive, 2 added, 2 added to flow

0 alloc failures, 0 force free

1 chunk, 0 chunks added

last clearing of statistics 00:09:48

Protocol	total Flows	Flows /sec	Packets /flow	Bytes /pkt	Packets /Sec	Active (sec) /Flow	Idle (sec) /Flow
---------							
IP-other Total:	2	0.0	193	79	0.6	1794.8	5.7
	2	0.0	193	79	0.6	1794.8	5.7

SrcIf	SrcIPaddress	DstIf	DstIPaddress	Pr SrcP DstP	Pkts
Se0/0/0	192.168.12.1	Null	224.0.0.5	59 0000 0000	35

SrcIf	SrcIPaddress	DstIf	DstIPaddress	pr SrcP DstP	Pkts
Se0/0/1	192.168.23.2	Null	224.0.0.5	59 0000 0000	33

ნაწილი №4: NetFlow

მარშრუტიზატორის ინტერფეისის შემაჯამებელი ცხრილი:

მარშრუტიზატორის ინტერფეისის შეჯამება				
მარშრუტიზატორის მოდელი	Ethernet interface №1	Ethernet interface №2	Serial interface №1	Serial interface №2
1800	FastEthernet 0/0 (F0/0)	FastEthernet 0/1 (F0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
1900	GigabitEthernet 0/0 (G0/0)	GigabitEthernet 0/1 (G0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
2801	FastEthernet 0/0 (F0/0)	FastEthernet 0/1 (F0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/1/1 (S0/1/1)
2811	FastEthernet 0/0 (F0/0)	FastEthernet 0/1 (F0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
2900	GigabitEthernet 0/0 (G0/0)	GigabitEthernet 0/1 (G0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)

შენიშვნა: თუ გინდათ მოძებნოთ თუ როგორაა მარშრუტიზატორი კონფიგურირებული, დაათვალიერეთ ინტერფეისები რათა მოახდინოთ მარშრუტიზატორის იდენტიფიკაცია და დაადგინოთ რამდენი ინტერფეისი აქვს მას. აქ არ არის იმის საშუალება, რომ ჩამოიწეროს თითოეული მარშრუტიზატორის კონფიგურაციების კომბინაციები. მოცემული ცხრილი მოიცავს მოწყობილობის **Ethernet** და **Serial** ინტერფეისების შესაძლო კომბინაციების იდენტიფიკატორებს. ეს ცხრილი არ მოიცავს ნებისმიერი სხვა ტიპის ინტერფეისს, რადგან კონკრეტული მარშრუტიზატორი შეიძლება შეიცავდეს მხოლოდ ერთს. ამის მაგალითი შეიძლება იყოს **ISDN BRI** ინტერფეისი. ფრჩხილებში მოცემული ცვლადი არის ლეგალური აბრევიატურა, რომელიც შეიძლება გამოყენებულ იქნას **Cisco IOS** ბრძანებებში, ინტერფეისის წარმოსადგენად.

პრაქტიკული სავარჯიშო

1. შეასრულეთ ერთ სივრციანი (Single-Area) OSPFv2-ის პრობლემის მოძიება და აღმოფხვრა;
2. შეასრულეთ ერთსივრციანი (Single-Area) OSPFv2-სა და OSPFv3-ის მთავარი პრობლემების მოძიება და გამოსწორება;
3. განახორციელეთ SNMP მმართველისა და აგენტის კონფიგურაცია;
4. მოახდინეთ OID კოდების კონვერტაცია Cisco SNMP Object Navigator-თან;
5. განახორციელეთ Syslog სერვისის კონფიგურაცია;
6. შეასრულეთ კომუტატორის საათების (Switch Clocks) ხელით მომართვა;
2. შეასრულეთ NTP სერვისის კონფიგურაცია;
3. განახორციელეთ დაფიქსირებული სარეგისტრაციო ჟურნალისების შემოწმება;
4. მოახდინეთ Syslog-ის და NTP-ს კონფიგურაცია;
5. განახორციელეთ NetFlow-ს კონფიგურაცია მარშრუტიზატორზე;
21. შეასრულეთ NetFlow-ს ანალიზი CLI-ს გამოყენებით;
22. შეასრულეთ NetFlow კოლექტორის შესწავლა და ანალიზატორი პროგრამული უზრუნველყოფა.

ცოდნის შეფასება

სტუდენტებს მიეცემათ პრაქტიკული დავალება

- შეასრულონ ქსელის აწყობა და მოწყობილობის კონფიგურაციების ჩატვირთვა, მესამე დონის კავშირის პრობლემის მოგვარება, მეზობელი OSPF ინფორმაციის დადგენა, OSPFv2 მარშრუტიზაციის ინფორმაციის შემოწმება, OSPFv2-ის პრობლემის აღმოფხვრა, OSPFv3-ის პრობლემის აღმოფხვრა; IPv6 საბოლოო წერტილებს შორის კავშირის შემოწმება;

- განახორციელონ SNMP მმართველისა და აგენტების კონფიგურაცია, SNMP მართვის პროგრამის ინსტალაცია, SNMP აგენტის აღმოჩენა, OID კოდების კონვერტაცია Cisco SNMP Object Navigator-ის საშუალებით, მიმდინარე SNMP შეტყობინებების წაშლა, SNMP trap-სა და შეტყობინების შექმნა, SNMP MIB/OID შეტყობინებების დეკოდირება;
- შეასრულონ Syslog სერვისის ჩართვა, შუალედური მოწყობილობების კონფიგურაცია Syslog სერვისის გამოსაყენებლად, რეგისტრირებული ღონისძიებების (Logged Events) შექმნა, საათების ხელით მომართვა კომუტატორებზე, ჟურნალირების დროითი შტამპების (logging timestamp) სერვისის დაყენება კომუტატორებზე, NTP სერვისის ჩართვა, დროით მარკირებული სარეგისტრაციო ჟურნალების (Timestamped Logs) შემოწმება;
- განახორციელონ NTP-ს კონფიგურაცია, NTP მმართველის კონფიგურაცია, NTP კონფიგურაციის შემოწმება, Syslog-ის კონფიგურაცია, syslog სერვერის ინსტალაცია, კონფიგურაცია და ჟურნალირების „სირთულის“ დონეების დაკვირვება მარშრუტიზატორზე;
- შეასრულონ NetFlow-ს კონფიგურაცია მარშრუტიზატორზე, NetFlow გამოჭერის (Capture) კონფიგურაცია, NetFlow-ს მონაცემთა ექსპორტის კონფიგურაცია, NetFlow-ს ექსპორტის ვერსიის კონფიგურაცია, NetFlow-ს კონფიგურაციის შემოწმება, NetFlow-ს შემაჯამებელი ანგარიშის სტატისტიკის ჩვენება, Telnet და ბრაუზერის სესიების დახურვა, NetFlow სააღრიცხვო სტატისტიკის გასუფთავება, NetFlow კოლექტორისა და ანალიზერის პროგრამული უზრუნველყოფის შესწავლა.

შემფასებელი აკვირდება შესაფასებელ პირის მუშაობას პროფესიული სტანდარტით (პროგრამით / მოდულით) განსაზღვრული ამოცანების შესრულების პროცესში. დაკვირვება ხორციელდება კომპიუტერებით აღჭურვილ ლაბორატორიაში, სადაც შესაფასებელი პირი პრაქტიკულ საქმიანობას ეწევა. შემფასებელმა წინასწარ უნდა დაგეგმოს დაკვირვების პროცესი, იმის დასადგენად, თუ რამდენად სწორად იყენებს შესაფასებელი პირი ცოდნას, უნარებსა და ყველა რესურსს შედეგის მისაღწევად.

შეფასება განხორციელდება პროცესზე დაკვირვებით, წინასწარ განსაზღვრული შეფასების ინდიკატორების საფუძველზე.

დავალების ნიმუში და შეფასების რუბრიკა

პროცესზე დაკვირვება

- *შეასრულოს ერთ სივრციანი (Single-Area) OSPFv2-ისა და OSPFv3-ის პრობლემის მოძიება და აღმოფხვრა.*
- *გამოიყენოს დაყენებული პრაქტიკული ამოცანებისთვის (Syslog სერვისის კონფიგურაცია, NTP სერვისის კონფიგურაცია, NetFlow-ს კონფიგურაცია მარშრუტიზატორზე)*

სწავლის შედეგი	N	დასახელება	შეფასება	
			კი	არა
მონიტორინგის, ინციდენტების და სხვადასხვა სერვისების დიაგნოსტიკა პრობლემების აღმოფხვრით	1.	შეასრულა ერთ სივრციანი (Single-Area) OSPFv2-ის პრობლემის მოძიება და აღმოფხვრა		
	2.	შეასრულა SNMP მმართველისა და აგენტის კონფიგურაცია		
	3.	შეასრულა OID კოდების კონვერტაცია Cisco SNMP Object Navigator-თან		
	4.	შეასრულა Syslog სერვისის კონფიგურაცია		
	5.	შეასრულა კომუტატორის საათების (Switch Clocks) ხელით მომართვა		
	6.	შეასრულა NTP სერვისის კონფიგურაცია		
	7.	შეასრულა დაფიქსირებული სარეგისტრაციო ჟურნალისების შემოწმება		
	8.	შეასრულა Syslog-ის და NTP-ს კონფიგურაცია		
	9.	შეასრულა NetFlow-ს ანალიზი CLI-ს გამოყენებით		
	10.	შეასრულა NetFlow კოლექტორის შესწავლა და ანალიზატორი პროგრამული უზრუნველყოფა		

სწავლის შედეგი ჩაითვლება მიღწეულად თუ სტუდენტმა შეძლო შედეგის მინიმუმ 8 პუნქტის შესრულება.

დასკვნა

კომპიუტერული ქსელის ფიზიკური და ლოგიკური გამართვა კომპიუტერული ქსელის ადმინისტრატორის ძირითადი ამოცანაა.

ქსელის მაშტაბების ზრდასთან ერთად, თანამედროვე ქსელის ადმინისტრატორი მოეთხოვება უფრო მეტი ცოდნა, როგორც უშუალო პროფესიულ მოვალეობებში ასევე უსაფრთხოებისა და მონიტორინგის საკითხებში

ქსელის ტიპების წარმოდგენის ყველაზე თვალსაჩინო მაგალითია - ლოკალური და გლობალური ქსელები. მოცემული სახელმძღვანელო ძირითადად ეხმაურება გლობალური ქსელების ტექნოლოგიებს და ქსელთაშორის მარშრუტიზაციას, მასში განხილული საკითხები უთუოდ საინტერესო იქნება გარკვეული გამოცდილების მქონე ქსელის ადმინისტრირებით დაინტერესებული მკითხველისთვის.

მოცემული სახელმძღვანელო შექმნილია „კომპიუტერული ქსელის ადმინისტრირება“ პროფესიული სასწავლო პროგრამის სტუდენტებისათვის და მოიცავს II ეტაპზე სწავლებად მოდულებს.

გამოყენებული ლიტერატურა

1. Cisco.netacad.com
2. CCNA R&S: Routing and Switching Essentials (Cisco.netacad.com)
3. CCNA R&S: Scaling Networks (Cisco.netacad.com)
4. http://www.cisco.com/c/en/us/support/docs/ip/network-address-translation-nat/4606-8.html
5. http://www.cisco.com/c/en/us/products/security/vpn-endpoint securityclients/index.html
6. http://www.cisco.com/c/en/us/support/docs/cloud-systems-management/configurationprofessional/113337-ccp-vpn-routerA-routerB-config-00.html
7. http://www.cisco.com/c/en/us/support/docs/security/vpn-client/71461-routervpnclient-pi-stick.html
8. http://www.cisco.com/c/en/us/td/docs/ios/12_2sr/12_2srb/feature/guide/tbgp_c/brbcln s. html
9. კაპანაძე დავით, „ინფორმაციული უსაფრთხოების საფუძვლები“, 2009