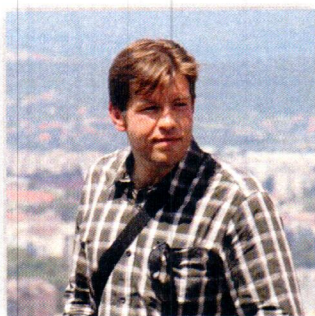


Michał Filipek



Z branżą telekomunikacyjną związany jestem od przeszło 9 lat. Specjalizuję się w dostarczaniu usług głosowych w oparciu o sieci z komutacją tarczy (TDM) jak i pakietów (VoIP). Współpracując z operatorami telekomunikacyjnymi pozyskałem wiedzę i doświadczenie, które w sposób naturalny rozwijają moją wcześniejszą fascynację nowymi technologiami oraz ścieżkę edukacyjną jaką wybrałem (Informatyka, Politechnika Warszawska).

Od początku kariery zainteresowałem się rozwiązaniami firmy Mikrotik i w mniejszym lub większym stopniu wykorzystywałem je w różnych projektach. Używałem Mikrotika między innymi jako:

- ✓ firewall
- ✓ QoS
- ✓ koncentracja VPN (największe wdrożenie ok. 1500 urządzeń)
- ✓ budowa sieci typu hotspot (CAPsMAN), instalacje o skali 100 Access Pointów per site

W rozwiązaniach Mikrotik cenię sobie, nie tylko duże możliwości konfiguracyjne, niską cenę ale przede wszystkim społeczność jaka powstała wokół systemu RouterOS. Kilukrotnie brałem udział w wydarzeniach z cyklu MUM (Mikrotik User Meeting) między innymi w Czechach, na Słowenii oraz w Mediolanie.

Piotr Wasyk



Z branżą IT związany jestem od kilkunastu lat. Skończyłem studia na Politechnice Warszawskiej oraz studia podyplomowe z zakresu zarządzania projektami IT w Łodzi. Na co dzień zajmuję się administracją i zarządzaniem:

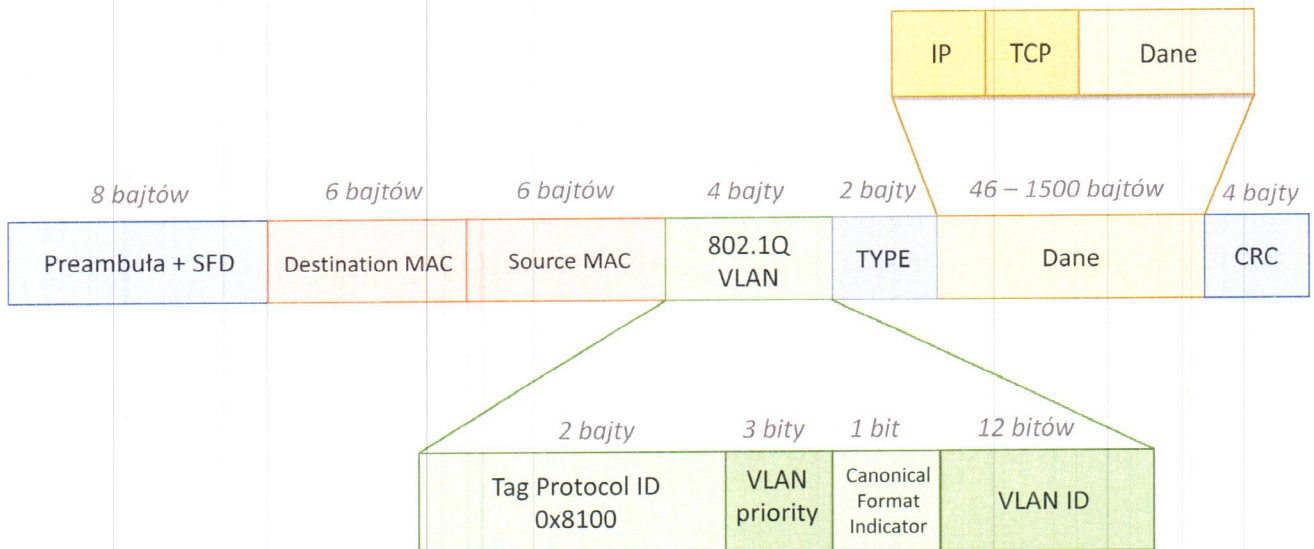
- ✓ infrastrukturą serwerową
- ✓ centralami VoIP (Asterisk)
- ✓ sieciami WLAN (Mikrotik i Ubiquiti)
- ✓ siecią LAN
- ✓ tunelami VPN
- ✓ QoS
- ✓ monitorowaniem usług

Urządzenia Mikrotik głównie wykorzystuję jako routery brzegowe, bramy VPN, kontroler sieci bezprzewodowych (CAPsMAN).

Cenię te urządzenia za ich elastyczność i możliwość konfiguracji w najmniejszych szczegółach. Dobre wsparcie dla protokołów redundancji pozwala zbudować wydajne, wysoko dostępne rozwiązanie za ułamek ceny oferowanej przez innych producentów segmentu Enterprise.

Model OSI

802.1Q



VLAN

typy portów

▪ access

Tak zwane ramki nietagowane. Tutaj podłączamy: komputer, kamera ip, drukarka

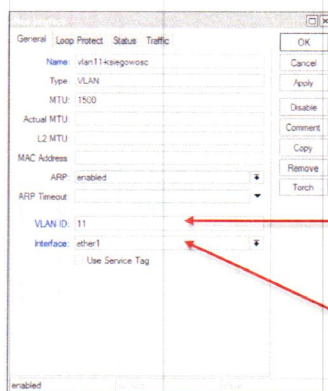
▪ trunk

Ramki ethernet'owe posiadają znacznik **vlan id**, połączenia pomiędzy: router, switch, access point

VLAN

Interface vlan w systemie RouterOS możemy:

- Zaadresować statycznie
- Zaadresować dynamicznie (dhcp-client)
- Uruchomić na nim serwer DHCP
- Dodać do bridge'a
- Uruchomić na nim serwer/klient pppoe

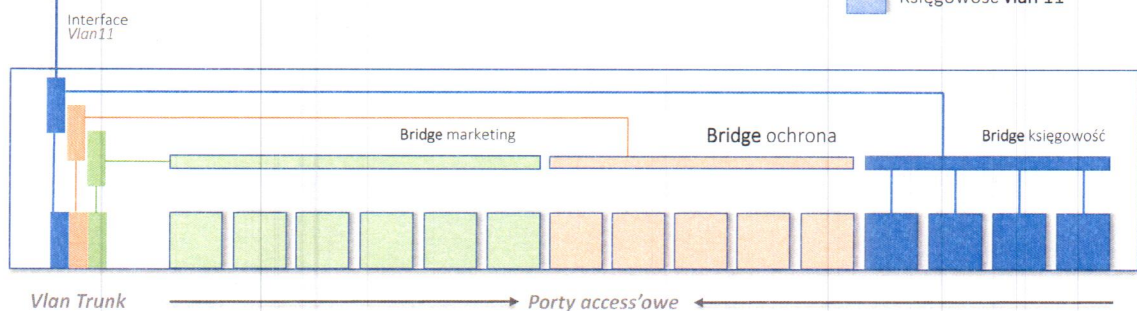


VLAN przykład 1

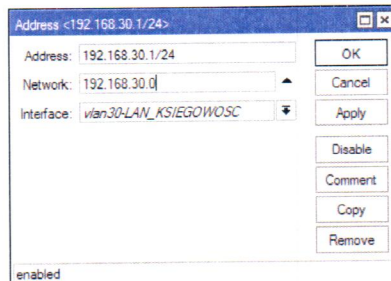
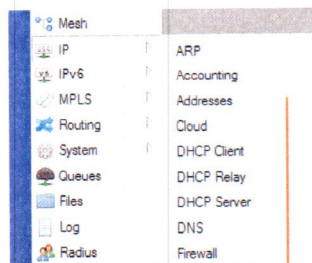
VLAN ID 12 bitowy znacznik (1-4095)

Interface L2, który pełni rolę vlan trunk, ramki Ethernet'owe pojawiające się na nim będą posiadały znacznik VLAN ID

- Marketing vlan 20
- Ochrona vlan 30
- Księgowość vlan 11



VLAN przykład 2

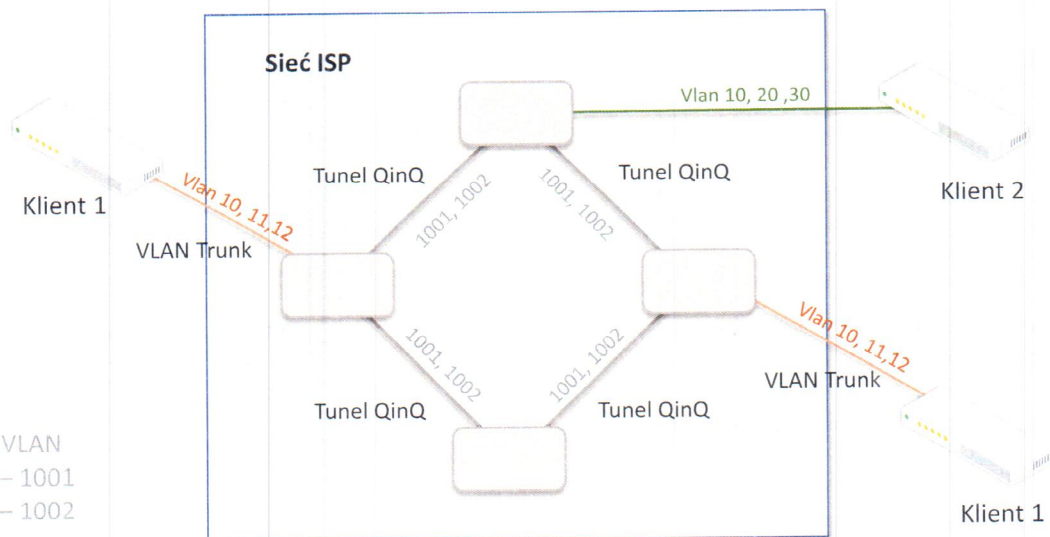


Address	Network	Interface
1.1.1.2/29	1.1.1.0	ether1
192.168.10.1/24	192.168.10.0	vlan10-LAN_IT
192.168.20.1/24	192.168.20.0	vlan20-LAN_MARKETING
192.168.30.1/24	192.168.30.0	vlan30-LAN_KSIEGOWOSC
192.168.40.1/24	192.168.40.0	vlan40-LAN_RECEPCJA
192.168.90.1/24	192.168.90.0	vlan90-DMZ

6 items (1 selected)

Adresacje na interfejsach typu **vlan** dodajemy identycznie jak na interfejsach fizycznych

VLAN QinQ



VLAN

switch chip / RouterOS < 6.41

Przykład dotyczy **RB952Ui-5ac2nD (hAP ac lite)** - Atheros8227
Dodanie portów do switch chip'a

W ten sposób komunikacja pomiędzy urządzeniami podłączonymi do portów switch'a odbywa się z pominięciem procesora

Podobnie postępujemy z innymi portami: ether2, ether4

VLAN

switch chip / RouterOS 6.41 lub nowszy

Przykład dotyczy **RB952Ui-5ac2nD (hAP ac lite)** - Atheros8227
Dodanie portów do switch chip'a

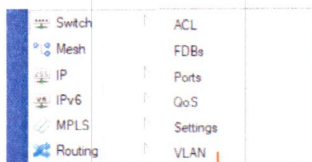
H – oznacza hardware offload, czyli ruch pomiędzy wszystkimi portami posiadającymi atrybut H, odbywa się z pominięciem CPU

#	Interface	Bridge	Horizon	Priority	H	Path Cost	Role	Root Pat...
0	all	bridge1				80	10	
1	DI1	bridge1				80	10	disabled port
2	DI2	bridge1				80	10	disabled port
3	DI3	bridge1				80	10	disabled port
4	DI4	bridge1				80	10	designated port
5	DI5	bridge1				80	10	disabled port
6	DI6	bridge1				80	10	disabled port
7	DI7	bridge1				80	10	disabled port

W nowszych wersjach RouterOS usunięto opcję master port, obecnie, aby skorzystać z funkcjonalności switch chip wystarczy utworzyć interface bridge i dodać do niego fizyczne porty.

VLAN

switch chip / CRS 125 / przykład / RouterOS 6.41 lub nowszy



Wypełniamy tablicę vlan, czyli jakie vlan id będą dostępne na konkretnych portach, bez wskazywania czy port jest typu trunk czy access. Te informacje uzupełnimy później w zakładkach: Eg. VLAN Tag, In. VLAN Tran.

Switch VLAN									
VLAN	Eg. VLAN Tag	In. VLAN Tran	Eg. VLAN Tran	1:1 VLAN Switching	MAC Based VLAN	Protocol Based VLAN			
VLAN ID	Ports			SVL	SA Learning	Road	Ingress Mirror	Comment	
101	ether1, ether2, ether3, ether4, ether5, ether6, ether7, ether8, ether9, ether10, ether11, ether12, switch1-cpu			no	yes	no	no	VLAN 101 - 192.168.101.0/24	
102	ether1, ether13, ether14, ether15, ether16, ether17, ether18, ether19, ether20, ether21, ether22, ether23, ether24, switch1-cpu			no	yes	no	no	VLAN 102 - 10.0.102.0/24	
4095	ether1, ether2, ether3, ether4, ether5, ether6, ether7, ether8, ether9, ether10, ether11, ether12, ether13, ether14, ether15, ether16, ether17, ether18, ether19, ether20, ether21, ether22, ether23, ether24, switch1-cpu			no	no	no	no		

VLAN	Eg. VLAN Tag	In. VLAN Tran
VLAN ID	Tagged Ports	
101	ether1, switch1-cpu	
102	ether1, switch1-cpu	
4095		

Oznaczamy porty, które będą tagowane

Switch VLAN									
VLAN	Eg. VLAN Tag	In. VLAN Trn	Eg. VLAN Trn	1:1 VLAN Switching	MAC Based VLAN	Protocol Based VLAN			
     									
Ports									
ether13, ether14, ether15, ether16, ether17, ether18, ether19, ether20, ether21, ether22, ether23, ether24									
ether2, ether3, ether4, ether5, ether6, ether7, ether8, ether9, ether10, ether11, ether12									

Oznaczamy porty, które będą typu access (nietagowane)

Ingress VLAN Translation - ether13, ether14, ether15, ether16, ether17, ether18, ether19, ether20, ether21, ether22, ether23, ether24

Ports: ether13, ether14, ether15, ether16, ether17, ether18, ether19, ether20, ether21, ether22, ether23, ether24

Protocol: any

Service VLAN Lookup For: any

Service VID: any

Service PCP: any

Service DEI: any

Customer VLAN Lookup For: any

Customer VID: 0

Customer PCP: any

Customer DEI: any

New Service VID: any

New Customer VID: 102

PCP Propagation: ☐

SA Learning: ☒

Bonding

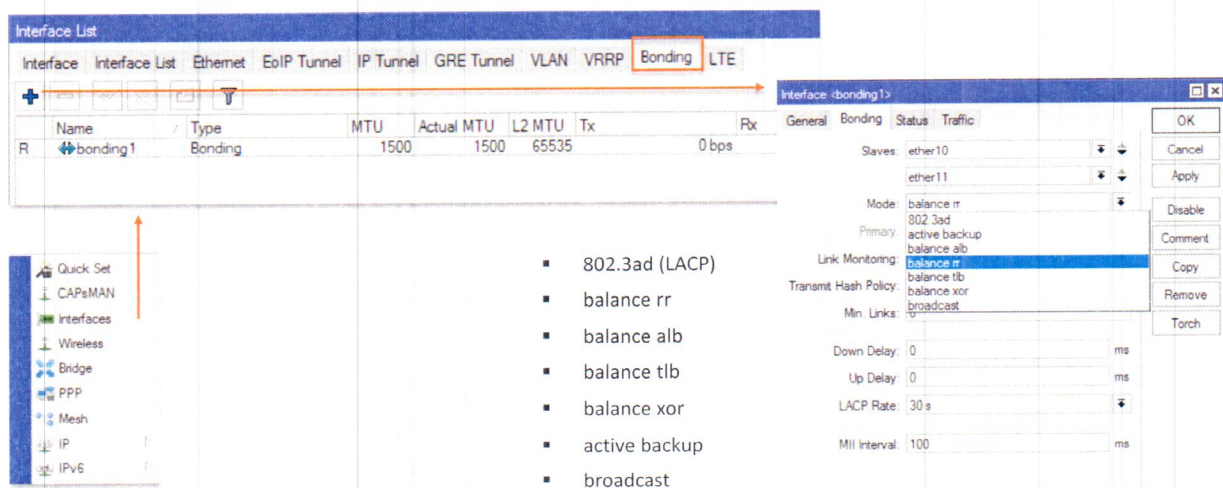
Bonding

Monitorowanie połączenia

- **ARP** - sprawdzane jest, czy występuje poprawna komunikacja **arp**. Dzięki tej metodzie wiemy, czy faktycznie można przesłać dane po linku.
- **MII** – sprawdzane jest jedynie, czy port jest w stanie up

Bonding

Tryby pracy



The screenshot shows the Mikrotik WinBox interface. On the left, the 'Interface List' window displays a table with columns: Name, Type, MTU, Actual MTU, L2 MTU, Tx, and Rx. The entry 'bonding1' is listed with Type 'Bonding'. An orange arrow points from this entry to the 'Interface: bonding1' configuration window on the right. In this window, the 'Bonding' tab is active, showing settings for Slaves (ether10, ether11), Mode (balance rr), Link Monitoring (balance rr), Transmit Hash Policy (balance xor), Min Links (1), Down Delay (0 ms), Up Delay (0 ms), LACP Rate (30 s), and MII Interval (100 ms). In the center, a list of bonding modes is provided:

- 802.3ad (LACP)
- balance rr
- balance alb
- balance tlb
- balance xor
- active backup
- broadcast

Bonding

balance rr, active backup

- **Balance rr** pakiety kierowane są przemiennie na kolejne zagregowane interface'y. Pakiety mogą docierać w innej kolejności. Jest to jedyny mechanizm pozwalający na kierowanie różnymi interface'ami ramek należących do tego samego połączenia TCP/IP. Rozwiązanie jest spotykane w agregacji linków typu Wireless.
- **Active backup** transmisja odbywa się jedynie za pomocą interface'u podstawowego, pozostałe interface'y stanowią jedynie połączenie zapasowe. W tym trybie działa jedynie MII monitoring. Tryb przydatny, gdy druga strona nie wspiera mechanizmów agregacji portów.

Bonding

balance xor, broadcast, balance tbl

- **Balance xor** mechanizm podobny do LACP, nie jest standardem, więc wsparcie po stronie innych urządzeń nie jest powszechne. Umożliwia hash'owanie layer3, layer4
- **Broadcast** pakiety są transmitowane jednocześnie wszystkimi interface'ami (ten sam pakiet pojawia się jednocześnie na innych interface'ach)
 - zapewnia odporność na awarię łącza
 - nie zwiększa przepustowości
- **Balance tbl** ruch rozkładany jest na podstawie adresu docelowego. Nie ma konieczności konfigurowania drugiej strony połączenia. Rozkładanie ruchu będzie następowało tylko w jednym kierunku.

VRRP

konfiguracja

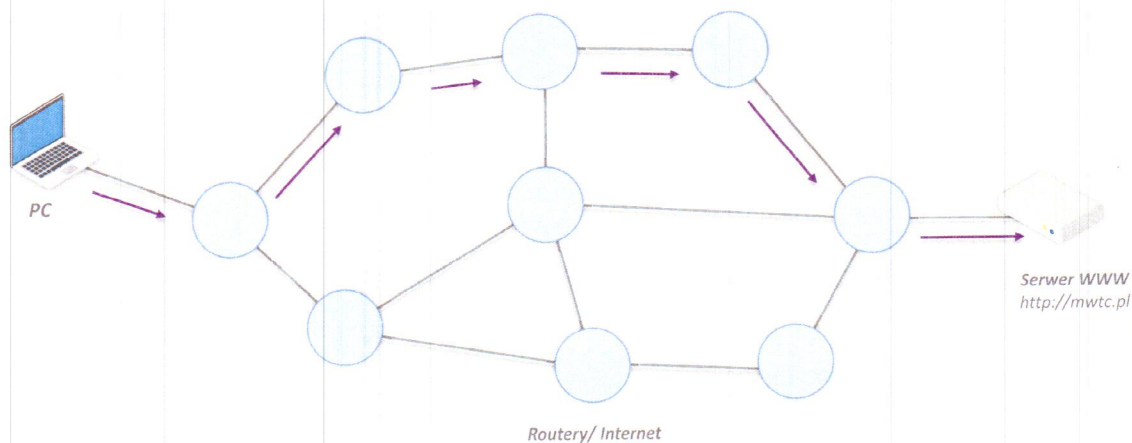
- **Preemption mode** zapobiega reelekcji mastera w przypadku, gdy pierwotny master stał się ponownie dostępny
- **Interval** – czas, co jaki będzie sprawdzany stan uczestników grupy VRRP
- **Authentication** - dodatkowe zabezpieczenie, aby nikt nie podszył się pod router pracujący w grupie VRRP
 - Brak**
 - Simple** hasło przekazywane jawnie
 - AH** hasło nie jest przekazywane jawnie

Routing statyczny

Routing

Pakiet IP przechodzi zawsze od źródła do urządzenia końcowego. Urządzenia końcowe (komputer, drukarka) posiadają stosunkowo proste tablice routingu. Routery posiadają bardziej rozbudowane tablice i mają więcej połączeń do innych routerów.

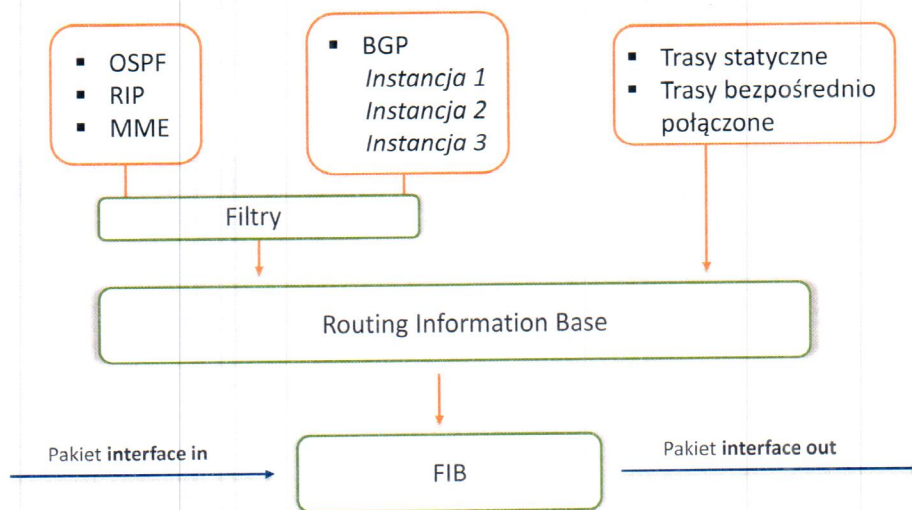
HOP – określenie routera znajdującego się pomiędzy źródłem pakietu, a celem (*poniżej: pomiędzy komputerem a serwerem mamy 5 HOP'ów*)



Routing

RIB Routing Information Base, utrzymuje różne tablice routingu

FIB Forwarding Information Base, służy do podjęcia decyzji, gdzie skierować dany pakiet



Routing

tablica routingu: flagi, parametr distance

- X** – trasa administracyjnie wyłączona
- A** – trasa aktywna
- D** – wpis został dodany dynamicznie, nie został przez nas wprowadzony ręcznie, np. adres otrzymaliśmy z DHCP
- C** – trasa bezpośrednio podłączona do naszego routera, powiązana z adresem IP jaki nadaliśmy na interface
- S** – trasa statyczna
- r** – wpis dodany za pomocą protokołu RIP
- b** – wpis dodany za pomocą protokołu BGP
- o** – wpis dodany za pomocą protokołu OSPF
- m** – wpis dodany za pomocą protokołu MME
- B** – blackhole, blokujemy dostęp do
- U** – unreachable, poinformuje stronę nadawczą za pomocą komunikatu ICMP
- P** – prohibited, poinformuje stronę nadawczą za pomocą komunikatu ICMP

Dist	Address	Gateway	Distance	Routing Mark	Pref	Source
AS	0.0.0.0/0	192.168.23.254 reachable wlan3-WAN	1		2,2,2,2	
DAC	2.2.2.0/24	bridge-sstp reachable	0			
DAC	10.8.0.1	172.16.10.100 reachable sstp-out-vult-chr-frankfurt	110			
DAo	10.250.1.0/24	172.16.10.100 reachable sstp-out-vult-chr-frankfurt	110			
DAo	127.1.1.250	172.16.10.100 reachable sstp-out-vult-chr-frankfurt	110			
DAC	157.25.99.64/29	bridge1-lan reachable	0		157.25.99.65	
DAC	172.16.10.100	sstp-out-vult-chr-frankfurt reachable	0		172.16.10.101	
DAo	172.16.10.101	172.16.10.100 reachable sstp-out-vult-chr-frankfurt	110			
DAo	172.16.100.1	172.16.10.100 reachable sstp-out-vult-chr-frankfurt	110			
DAo	172.16.100.2	172.16.10.100 reachable sstp-out-vult-chr-frankfurt	110			
DAC	172.20.20.1	korkowa-dom reachable	0		172.20.20.2	
DAo	192.168.1.0/24	172.16.10.100 reachable sstp-out-vult-chr-frankfurt	110			
DAC	192.168.5.0/24	bridge1-lan reachable	0		192.168.5.254	
DAC	192.168.23.0/24	wlan3-WAN reachable	0		192.168.23.100	
AS	192.168.50.0/24	172.20.20.1 reachable korkowa-dom	1			

Im niższa wartość parametru **distance** tym trasa ma wyższy priorytet, domyślne wartości:

- bezpośrednio połączona – distance 0
- dodana dynamicznie (dhcp, PPP) – distance 1
- statyczna, dodana ręcznie – distance 1
- OSPF – distance 110
- eBGP – distance 20
- IGRP – distance 100
- iBGP – distance 200

Routing

tablica routingu / kolejność przetwarzania wpisów

Dist	Address	Gateway	Distance	Routing Mark	Pref	Source
AS	0.0.0.0/0	192.168.23.254 reachable wlan3-WAN	1			
S	0.0.0.0/0	192.168.230.1 reachable bridge1-lan	5			
DAC	2.2.2.0/24	bridge-sstp reachable	0		2,2,2,2	
DAo	10.8.0.1	172.16.10.100 reachable sstp-out-vult-chr-frankfurt	110			
AS	10.20.100.0/23	172.16.10.100 reachable sstp-out-vult-chr-frankfurt	10			
AS	10.20.100.0/24	192.168.23.254 reachable wlan3-WAN	1			
DAo	10.250.1.0/24	172.16.10.100 reachable sstp-out-vult-chr-frankfurt	110			
DAo	127.1.1.250	172.16.10.100 reachable sstp-out-vult-chr-frankfurt	110			
DAC	157.25.99.64/29	bridge1-lan reachable	0		157.25.99.65	
DAC	172.16.10.100	sstp-out-vult-chr-frankfurt reachable	0		172.16.10.101	
DAo	172.16.10.101	172.16.10.100 reachable sstp-out-vult-chr-frankfurt	110			
DAo	172.16.100.1	172.16.10.100 reachable sstp-out-vult-chr-frankfurt	110			
DAo	172.16.100.2	172.16.10.100 reachable sstp-out-vult-chr-frankfurt	110			
DAC	172.20.20.1	korkowa-dom reachable	0		172.20.20.2	
DAo	192.168.1.0/24	172.16.10.100 reachable sstp-out-vult-chr-frankfurt	110			
DAC	192.168.5.0/24	bridge1-lan reachable	0		192.168.5.254	
DAC	192.168.23.0/24	wlan3-WAN reachable	0		192.168.23.100	
AS	192.168.50.0/24	172.20.20.1 reachable korkowa-dom	1			
DAC	192.168.230.0/24	bridge1-lan reachable	0		192.168.230.5	

Przykład 1

Do routera trafia pakiet IP z adresem docelowym 8.8.8.8, pakiet zostanie skierowany zgodnie z tablicą routingu na router o adresie: 192.168.23.254

Przykład 2

Router chce osiągnąć adres 192.168.5.113, z tablicy routingu wynika, iż najlepszym wpisem do obsłużenia tego ruchu będzie sieć własna 192.168.50.0/24 a więc router wykorzysta protokół ARP do dostarczenia pakietu.

W pierwszej kolejności brane są pod uwagę szczegółowość wpisu (czyli dla adresu 10.20.100.10 lepszym dopasowaniem jest 10.20.100.0/24 niż 10.20.100.0/23), dopiero dla wpisów z taką samą szczegółowością sprawdzany jest parametr (w naszym przykładzie 0.0.0.0/0) **distance**

Routing

Dodawanie tras routingu / check gateway

Route <0.0.0.0/0>

General | Attributes

Dist. Address: 0.0.0.0/0

Gateway: 45.32.154.209 reachable ether2

Check Gateway: ping

Type: ping

Distance: 1

Scope: 30

Target Scope: 10

Routing Mark:

Pref. Source:

enabled active static

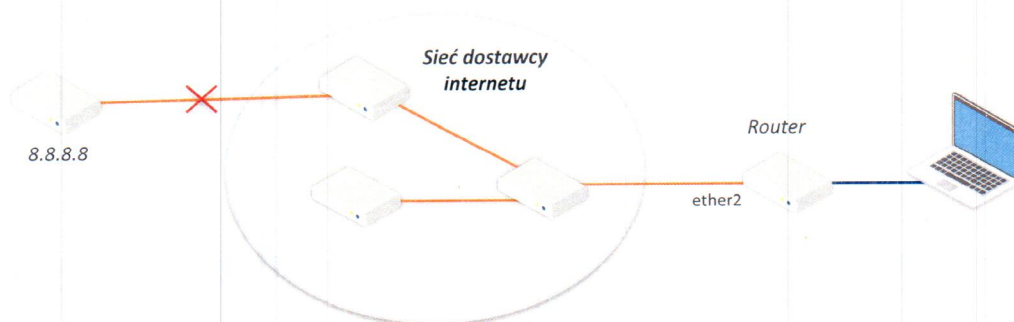
OK Cancel Apply Disable Comment Copy Remove

Check Gateway ping:

- Router wysyła co 10 sekund Echo Request (ping) do Gateway'a !
- Brak odpowiedzi na 2 komunikaty Echo Request powoduje uznanie bramy za nieosiągalną !

Routing

check gateway / recursive routing



W przypadku awarii, w sieci dostawcy usług internetowych, nasza trasa pozostanie aktywna ponieważ router operatora będzie odpowiadał na komunikaty *echo request*.

Aby rozwiązać ten problem, możemy zastosować mechanizm **recursive routing** wraz z opcją **check gateway**. Będziemy sprawdzać, czy łącze działa wysyłając komunikaty echo request, ale tym razem nie do routera operatora, ale do zewnętrznego adresu IP.