MikroTik OSPF Manual

PDF generated using the open source mwlib toolkit. See http://code.pediapress.com/ for more information.
PDF generated at: Fri, 11 May 2012 13:34:19 UTC

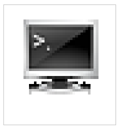
Contents

Articles

References

Manual:Routing/OSPF

Applies to RouterOS: v3, v4 +



Summary

MikroTik RouterOS implements OSPF version 2 (RFC 2328). The OSPF protocol is the link-state protocol that takes care of the routes in the dynamic network structure that can employ different paths to its subnetworks. It always chooses shortest path to the subnetwork first.

Instance

Sub-menu: `/routing ospf instance`

Since v3.17 it is possible to run multiple OSPF instances. General OSPF configuration now is moved to instances.

Properties

Property	Description
`distribute-default` (*never* | *if-installed-as-type-1* | *if-installed-as-type-2* | *always-as-type-1* | *always-as-type-2*; Default: **never**)	specifies how to distribute default route. Should be used for ABR (Area Border router) or ASBR (Autonomous System boundary router) • `never` - do not send own default route to other routers • `if-installed-as-type-1` - send the default route with type 1 metric only if it has been installed (a static default route, or route added by DHCP, PPP, etc.) • `if-installed-as-type-2` - send the default route with type 2 metric only if it has been installed (a static default route, or route added by DHCP, PPP, etc.) • `always-as-type-1` - always send the default route with type 1 metric • `always-as-type-2` - always send the default route with type 2 metric
`domain-id` (*Hex|Address*;)	MPLS related parameter. Identifies OSPF domain of the instance. This value is attached to OSPF routes redistributed in BGP as VPNv4 routes as BGP extended community attribute, and used when BGP VPNv4 routes are redistributed back OSPF to determine whether to generate inter-area or AS-external LSA for that route. By default Null domain-id is used, as described in RFC 4577.
`domain-tag` (*integer: 0..4294967295* ;)	if set, then used in route redistribution (as route-tag in all external LSAs generated by this router), and in route calculation (all external LSAs having this route tag are ignored). Needed for interoperability with older Cisco systems. By default not set.
`in-filter` (*string*;)	name of the routing filter chain used for incoming prefixes
`metric-bgp` (*integer|auto*; Default: **20**)	routes learned from the BGP protocol are redistributed with this metric. When set to *auto*, MED attribute value from BGP route will be used, if MED is not set then default value 20 is used.
`metric-connected` (*integer*; Default: **20**)	routes to directly connected networks are distributed with this metric
`metric-default` (*integer*; Default: **1**)	the default route is distributed with this metric
`metric-other-ospf` (*integer|auto*; Default: **20**)	routes learned from other OSPF instances are redistributed with this metric. If *auto* is configured, then the cost from previous instance is taken into account, otherwise cost is set to statically configured value.
`metric-rip` (*integer*; Default: **20**)	routes learned from the RIP protocol are redistributed with this metric

metric-static (*integer*; Default: **20**)	static routes are distributed with this metric
mpls-te-area (*string*;)	the area used for MPLS traffic engineering. TE Opaque LSAs are generated in this area. No more than one OSPF instance can have mpls-te-area configured.
mpls-te-router-id (*ip*;)	loopback interface from which to take IP address used as Router-ID in MPLS TE Opaque LSAs
out-filter (*string*;)	name of the routing filter chain used for outgoing prefixes
redistribute-bgp (*as-type-1* | *as-type-2* | *no*; Default: **no**)	redistribute routes learned by the BGP protocol
redistribute-connected (*as-type-1* | *as-type-2* | *no*; Default: **no**)	redistribute connected routes, i.e. routes to directly reachable networks
redistribute-other-ospf (*as-type-1* | *as-type-2* | *no*; Default: **no**)	redistribute routes learned by other OSPF instances
redistribute-rip (*as-type-1* | *as-type-2* | *no*; Default: **no**)	redistribute routes learned by the RIP protocol
redistribute-static (*as-type-1* | *as-type-2* | *no*; Default: **no**)	redistribute static routes
router-id (*IP address*; Default: **0.0.0.0**)	the OSPF Router ID. If not specified, OSPF use one of router's IP addresses.
routing-table (*name of routing table*;)	the routing table this OSPF instance operates on

Notes

OSPF protocol supports two types of metrics:

- `type1` - ospf metric is the sum of the internal OSPF cost and the external route cost
- `type2` - ospf metric is equal only to the external route cost.

Status

Command `/routing ospf monitor` will display current OSPF status.
For multi instance OSPF you have to use following command: `/routing ospf instance print status`

Available read only properties:

Property	Description
state (*down* | *running*)	shows if OSPF is running or not
effective-router-id (*IP address*)	Router-ID chosen by OSPF.
dijkstras (*integer*)	shows how many times Dijkstra's algorithm was executed (i.e. OSPF routes were recalculated)
db-exchanges (*integer*)	number of OSPF database exchanges currently going on
external-imports (*integer*)	how many external routes were imported into OSPF from this router

Area

Sub-menu: `/routing ospf area`

Description

OSPF allows collections of routers to be grouped together. Such a group is called an area. Each area runs a separate copy of the basic link-state routing algorithm. This means that each area has its own link-state database and corresponding shortest path tree.

The structure of an area is invisible from other areas. This isolation of knowledge makes the protocol more scalable if multiple areas are used; routing table calculation takes less CPU resources and routing traffic is reduced.

However, multi-area setups create additional complexity. It is not recommended separate areas with fewer than 50 routers. The maximum number of routers in one area is mostly dependent on CPU power you have for routing table calculation.

Properties

Property	Description
`area-id` (*IP address*; Default: **0.0.0.0**)	OSPF area identifier. If the router has networks in more than one area, then an area with area-id=0.0.0.0 (the backbone) must always be present. The backbone always contains all area border routers. The backbone is responsible for distributing routing information between non-backbone areas. The backbone must be contiguous, i.e. there must be no disconnected segments. However, area border routers do not need to be physically connected to the backbone - connection to it may be simulated using a virtual link.
`default-cost` (*integer*; Default: **1**)	specifies the cost for the default route originated by this stub area ABR. Applicable only for stub areas on ABRs
`inject-summary-lsas` (*yes* | *no*; Default: **yes**)	specifies whether to flood summary LSAs in this stub area. Applicable only for stub areas on ABRs
`name` (*string*; Default:)	the name of the area
`translator-role` (*translate-always* | *translate-candidate* | *translate-never*; Default: **translate-candidate**)	Parameter indicates which ABR will be used as translator from type7 to type5. Applicable only if area type is NSSA • translate-always - router will be always used as translator • translate-never - router will never be used as translator • translate-candidate - ospf ellects one of candidate routers to be a translator
`type` (*default* | *nssa* | *stub*; Default: **default**)	area type

Status

`/routing ospf area print status` will show additional read-only properties

Property	Description
interfaces (*integer*;)	count of interfaces assigned to this area
active-interfaces (*integer*;)	count of interfaces in operating state assigned to this area
neighbors (*integer*;)	count of OSPF neighbors in this area
adjacent-neighbors (*integer*;)	count of adjacent OSPF neighbors in this area

Area Range

Sub-menu: `/routing ospf area range`

Description

Prefix ranges are used to aggregate routing information on area boundaries. By default, ABR creates a summary LSA for each route in specific area, and advertises it in adjacent areas.

Using ranges allows to create only one summary LSA for multiple routes and send only single advertisement into adjacent areas, or to suppress advertisements altogether.

If a range is configured with 'advertise' parameter, a single summary LSA is advertised for each range if there are any routes under the range is the specific area. Else ('advertise' parameter disabled) no summary LSAs are created and advertised outside area boundaries at all.

Properties

Property	Description
advertise (*yes* | *no*; Default: **yes**)	whether to create summary LSA and advertise it to adjacent areas
area (*string*; Default:)	the OSPF area associated with this range
cost (*integer* | *default*; Default: **default**)	the cost of the summary LSA this range will create `default` - use the largest cost of all routes used (i.e. routes that fall within this range)
range (*IP prefix*; Default:)	the network prefix of this range

Note: For an active range (i.e. one that has at least one OSPF route from the specified area falling under it), a route with type 'unreachable' is created and installed in the routing table.

Network

Sub-menu: `/routing ospf network`

To start the OSPF protocol, you have to define the networks on which OSPF will run and associated area for each of these networks

Property	Description
area (*string*; Default: **backbone**)	the OSPF area to be associated with the specified address range
network (*IP prefix*; Default:)	the network prefix associated with the area. OSPF will be enabled on all interfaces that has at least one address falling within this range. Note that the network prefix of the address is used for this check (i.e. not the local address). For point-to-point interfaces this means the address of the remote endpoint.

Interface

Sub-menu: `/routing ospf interface`

Property	Description
authentication (*none* | *simple* | *md5*; Default: **none**)	specifies authentication method for OSPF protocol messages. • `none` - do not use authentication • `simple` - plain text authentication • `md5` - keyed Message Digest 5 authentication
authentication-key (*string*; Default: "")	authentication key to be used for simple or MD5 authentication
authentication-key-id (*integer*; Default: **1**)	key id is used to calculate message digest (used only when MD5 authentication is enabled). Value should match on all OSPF routers from the same region.
cost (*integer: 1..65535*; Default: **1**)	interface cost expressed as link state metric
dead-interval (*time*; Default: **40s**)	specifies the interval after which a neighbor is declared as dead. This interval is advertised in hello packets. This value must be the same for all routers on a specific network, otherwise adjacency between them will not form
hello-interval (*time*; Default: **10s**)	the interval between hello packets that the router sends out this interface. The smaller this interval is, the faster topological changes will be detected, but more routing traffic will ensue. This value must be the same for all routers on a specific network, otherwise adjacency between them will not form
interface (*string* | *all*; Default: **all**)	the interface name • `all` - for all interfaces without specific configuration
network-type (*broadcast* | *nbma* | *point-to-point* | *ptmp*; Default: **broadcast**)	the OSPF network type on this interface. Note that if interface configuration does not exist, the default network type is 'point-to-point' on PtP interfaces, and 'broadcast' on all other interfaces. • `broadcast` - network type suitable for Ethernet and other multicast capable link layers. Elects designated router • `nbma` - Non-Broadcast Multiple Access. Protocol packets are sent to each neighbors unicast address. Requires manual configuration of neighbors. Elects designated router • `point-to-point` - suitable for networks that consists only of two nodes. Does not elect designed router • `ptmp` - Point-to-Multipoint. Easier to configure than NBMA because it requires no manual configuration of neighbor. Does not elect designed router. This is the most robust network type and as such suitable for wireless networks, if 'broadcast' mode does not works good enough for them
passive (*yes* | *no*; Default: **no**)	if enabled, do not send or receive OSPF traffic on this interface
priority (*integer: 0..255*; Default: **1**)	router's priority. Used to determine the designated router in a broadcast network. The router with highest priority value takes precedence. Priority value 0 means the router is not eligible to become designated or backup designated router at all.
retransmit-interval (*time*; Default: **5s**)	time between retransmitting lost link state advertisements. When a router sends a link state advertisement (LSA) to its neighbor, it keeps the LSA until it receives back the acknowledgment. If it receives no acknowledgment in time, it will retransmit the LSA

transmit-delay (*time*; Default: **1s**)	link state transmit delay is the estimated time it takes to transmit a link state update packet on the interface

Status

`/routing ospf interface print status` will show additional information about used interfaces

Property	Description
ip-address (*IP address*;)	Ip address assigned to this interface
state (*backup* | *designated-router* | *point-to-point* | *passive;*)	current interface state
instance (*instance name*;)	OSPF instance that is using this interface
area (*area name*;)	area to which interface is assigned
neighbors (*integer*;)	count of OSPF neighbors found on this interface
adjacent-neighbors (*integer*;)	count of OSPF neighbors found on this interface that have formed adjacencies
designated-router (*IP address*;)	router-ID of elected designated router (DR)
backup-designated-router (*IP address*;)	router-ID of elected backup designated router (BDR)

NBMA Neighbor

Sub-menu: `/routing ospf nbma-neighbor`

Manual configuration for non-broadcast multi-access neighbors. Required only if interfaces with 'network-type=nbma' are configured.

Property	Description
address (*IP address*; Default:)	the unicast IP address of the neighbor
poll-interval (*time*; Default: **2m**)	how often to send hello messages to neighbors which are in "down" state (i.e. there is no traffic from them)
priority (*integer: 0..255*; Default: **0**)	assumed priority value of neighbors which are in "down" state

Virtual Link

Sub-menu: `/routing ospf virtual-link`

Description

As stated in OSPF RFC, the backbone area must be contiguous. However, it is possible to define areas in such a way that the backbone is no longer contiguous. In this case the system administrator must restore backbone connectivity by configuring virtual links. Virtual link can be configured between two routers through common area called transit area, one of them should have to be connected with backbone. Virtual links belong to the backbone. The protocol treats two routers joined by a virtual link as if they were connected by an unnumbered point-to-point network

Properties

Property	Description
`authentication` (*none* | *simple* | *md5*; Default: **none**)	specifies authentication method for OSPF protocol messages.
`authentication-key` (*string*; Default: **""**)	authentication key to be used for simple or MD5 authentication
`authentication-key-id` (*integer*; Default: **1**)	key id used in MD5 authentication
`neighbor-id` (*IP address*; Default: **0.0.0.0**)	specifies **router-id** of the neighbour
`transit-area` (*string*; Default: **(unknown)**)	a non-backbone area the two routers have in common

Note: Virtual link should be configured on both routers. Virtual links can not be established through stub areas.

LSA

Sub-menu: `/routing ospf lsa`

Read only properties:

Property	Description
`instance` (*string*)	Instance name where LSA is used.
`area` (*string*)	
`type` (*string*)	
`id` (*IP address*)	LSA record ID
`originator` (*IP address*)	LSA record originator
`sequence-number` (*string*)	Number of times the LSA for a link has been updated.
`age` (*integerr*)	How long ago (in seconds) the last update occurred
`checksum` (*string*)	LSA checksum
`options` (*string*)	
`body` (*string*)	

Neighbor

Sub-menu: `/routing ospf Neighbor`

Read only properties:

Property	Description
`router-id` (*IP address*)	neighbor router's **RouterID**
`address` (*IP address*)	IP address of neighbor router that is used to form OSPF connection
`interface` (*string*)	interface that neighbor router is connected to
`priority` (*integer*)	priority configured on neighbor
`dr-address` (*IP address*)	IP address of Designated Router
`backup-dr-address` (*IP address*)	IP address of Backup Designated Router

state (*down* | *attempt* | *init* | *2-way* | *ExStart* | *Exchange* | *Loading* | *full*)	• **Down** - No Hello packets has been received from neighbor. • **Attempt** - Applies only to NBMA clouds. State indicates that no recent information was received from neighbor. • **Init** - Hello packet received from the neighbor, but bidirectional communication is not established (Its own RouterID is not listed in Hello packet). • **2-way** - This state indicates that bi-directional communication is established. DR and BDR election occur during this state, routers build adjacencies based on whether router is DR or BDR, link is point-to-point or a virtual link. • **ExStart** - Routers try to establish the initial sequence number that is used for the packets information exchange. Router with higher ID becomes the master and starts the exchange. • **Exchange** - Routers exchange database description (DD) packets. • **Loading** - In this state actual link state information is exchanged. Link State Request packets are sent to neighbors to request any new LSAs that were found during Exchange state. • **Full** - Adjacency is complete, neighbor routers are fully adjacent. LSA information is synchronized between adjacent routers. Routers achieve the full state with their DR and BDR only, exception is P2P links.
state-changes (*integer*)	Total count of OSPF state changes since neighbor identification
ls-retransmits (*integer*)	
ls-requests (*integer*)	
db-summaries (*integer*)	
adjacency (*time*)	Elapsed time since adjacency was formed

OSPF Router

Sub-menu: `/routing ospf ospf-router`

List of all area border routers (ABRs).

Read only properties:

Property	Description
area (*string*)	
router-id (*IP address*)	
state (*string*)	
gateway (*IP address*)	
cost (*integer*)	

Route

Sub-menu: `/routing ospf route`

Read only properties:

Property	Description
instance (*string*)	Which OSPF instance route belongs to
dst-address (*IP prefix*)	Destination prefix
state (*intra-area* | *inter-area* | *ext-1* | *ext-2* | *imported-ext-1* | *imported-ext-2*)	State representing origin of the route
gateway (*IP address*)	used gateway
interface (*string*)	used interface
cost (*integer*)	Cost of the route
area (*external* | *backbone* | *<other area>*)	Which OSPF area this route belongs to

Sham link

Sub-menu: `/routing ospf sham-link`

Description

A sham-link is required between any two VPN sites that belong to the same OSPF area and share an OSPF backdoor link. If there is no intra-area link between the CE routers, you do not need to configure an OSPF sham link.

Sham link configuration example

Sham link must be configured on both sides.

For a sham link to be active, two conditions must be met:

- src-address is a valid local address with /32 netmask in OSPF instance's routing table.
- there is a valid route to dst-address in the OSPF instance's routing table.

When the sham link is active, hello packets are sent on it only until the neighbor reaches full state. After that, hello packet sending on the sham link is suppressed.

RouterOS does not support periodic LSA refresh suppression on sham-links yet.

Properties

Property	Description
area (*area name*)	name of area that shares an OSPF backdoor link
cost (*integer: 1..65535*)	cost of the link
dst-address (*IP address*)	loopback address of link's remote router
src-address (*IP address*)	loopback address of link's local router

See More

- OSPF case studies
- OSPF Configuration Examples

[Top | Back to Content **]**

Manual:OSPF-examples

Simple OSPF configuration

The following example illustrates how to configure single-area OSPF network. Let's assume we have the following network.

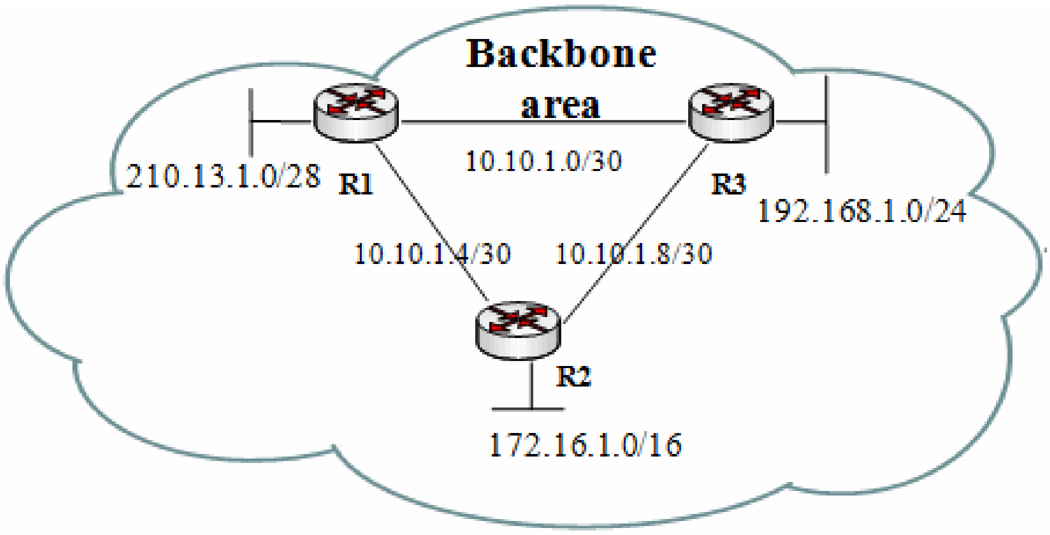


Figure 6.6. Example of single area OSPF network

Example network consists of 3 routers connected together within 10.10.1.0/24 network and each router has also one additional attached network.

In this example following IP addresses are configured:

```
[admin@MikroTikR1]/ip address add address=10.10.1.1/30 interface=ether1
 [admin@MikroTikR1]/ip address add address=10.10.1.5/30 interface=ether2
 [admin@MikroTikR1]/ip address add address=210.13.1.0/28 interface=ether3
```

```
 [admin@MikroTikR2]/ip address add address=10.10.1.6/30 interface=ether1
 [admin@MikroTikR2]/ip address add address=10.10.1.9/30 interface=ether2
 [admin@MikroTikR2]/ip address add address=172.16.1.0/16 interface=ether3
```

```
 [admin@MikroTikR3]/ip address add address=10.10.1.2 /30 interface=ether1
 [admin@MikroTikR3]/ip address add address=10.10.1.10/30 interface=ether2
 [admin@MikroTikR3]/ip address add address=192.168.1.0/24 interface=ether3
```

There are three basic elements of OSPF configuration:

- Enable OSPF instance
- OSPF area configuration
- OSPF network configuration

General information is configured in */routing ospf instance* menu. For advanced OSPF setups, it is possible to run multiple OSPF instances. Default instance configuration is good to start, we just need to enable default instance.

R1:

```
[admin@MikroTikR1] /routing ospf instance> add name=default
```

R2:

```
[admin@MikroTikR2] /routing ospf instance> add name=default
```

R3:

```
[admin@MikroTikR3] /routing ospf instance> add name=default
```

Show OSPF instance information:

```
[admin@MikroTikR1] /routing ospf instance> print
Flags: X - disabled
 0   name="default" router-id=0.0.0.0 distribute-default=never
     redistribute-connected=as-type-1 redistribute-static=as-type-1
     redistribute-rip=no redistribute-bgp=no redistribute-other-ospf=no
     metric-default=1 metric-connected=20 metric-static=20 metric-rip=20
     metric-bgp=auto metric-other-ospf=auto in-filter=ospf-in
out-filter=ospf-out
```

As you can see router-id is 0.0.0.0, it means that router will use one of router's IP addresses as router-id. In most cases it is recommended to set up *loopback* IP address as router-id. ***Loopback*** IP address is virtual, software address that is used for router identification in network. The benefits are that loopback address is always up (active) and can't be down as physical interface. OSPF protocol used it for communication among routers that identified by router-id. Loopback interface are configured as follows:

Create bridge interface named, for example, "loopback":

```
[admin@MikroTikR1] /interface bridge> add name=loopback
```

Add IP address:

```
[admin@MikroTikR1] > ip address add address=10.255.255.1/32 interface=loopback
```

Configure router-id as loopback:

```
[admin@MikroTikR1] /routing ospf instance> set 0 router-id=10.255.255.1
```

This can be done on other routers (R2, R3) as well.

Next step is to configure OSPF area. Backbone area is created during RouterOS installation and additional configuration is not required.

Note: Remember that backbone area-id is always (zero) 0.0.0.0.

And the last step is to add network to the certain OSPF area.

On R1

```
[admin@MikroTikR1] /routing ospf network> add network=210.13.1.0/28 area=backbone
[admin@MikroTikR1] /routing ospf network> add network=10.10.1.0/30 area=backbone
[admin@MikroTikR1] /routing ospf network> add network=10.10.1.4/30 area=backbone
```

Instead of typing in each network, you can aggregate networks using appropriate subnet mask. For example, to aggregate 10.10.1.0/30, 10.10.1.4/30, 10.10.1.8/30 networks, you can set up following ospf network:

```
[admin@MikroTikR1] /routing ospf network> add network=10.10.1.0/'''24''' area=backbone
```

R2:

```
[admin@MikroTikR2] /routing ospf network> add network=172.16.1.0/16 area=backbone
[admin@MikroTikR2] /routing ospf network> add network=10.10.1.0/24 area=backbone
```

R3:

```
[admin@MikroTikR3] /routing ospf network> add network=192.168.1.0/24 area=backbone
[admin@MikroTikR3] /routing ospf network> add network=10.10.1.0/24 area=backbone
```

You can verify your OSPF operation as follows:

- Look at the OSPF interface menu to verify that dynamic entry was created:

```
[admin@MikroTikR1] /routing ospf interface> print
```

- Check your OSPF neighbors, what DR and BDR is elected and adjacencies established:

```
[admin@MikroTikR1] /routing ospf neighbor> print
```

- Check router's routing table (make sure OSPF routes are present):

```
[admin@MikroTik_CE1] > ip route print
```

Simple multi-area configuration

Backbone area is the core of all OSPF network, all areas have to be connected to the backbone area. Start configuring OSPF from backbone and then expand network configuration to other areas.

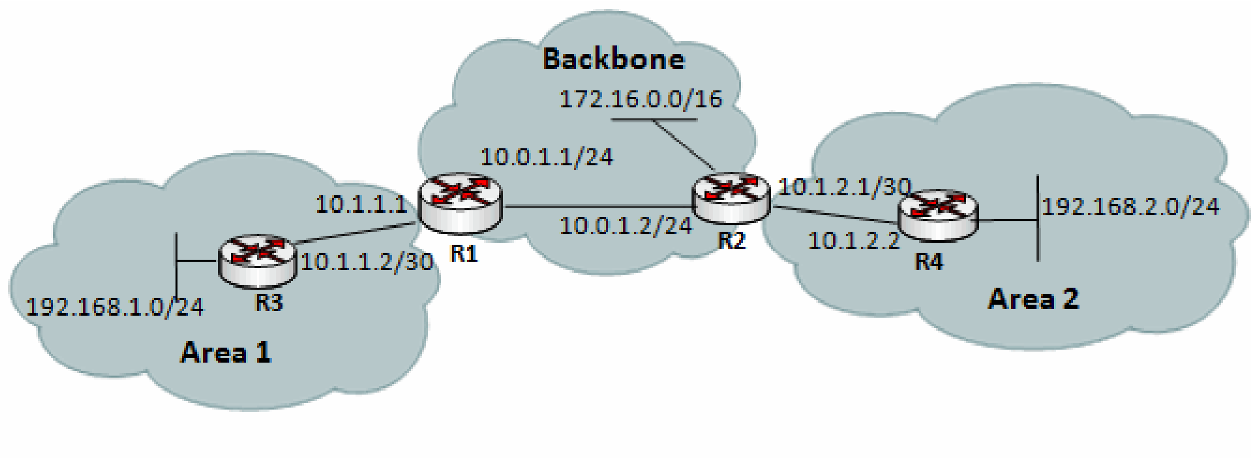


Figure 6.7. Example of multi- area OSPF network

Lets assume that IP addresses are already configured and default OSPF instance is enabled.

All we need to do is:

- create an area
- attach OSPF networks to the area

R1 configuration:

```
/routing ospf> add name=area1 area-id=0.0.0.1
/routing ospf> add network=10.0.1.0/24 area=backbone
/routing ospf> add network=10.1.1.0/30 area=area1
```

R2 configuration:

```
/routing ospf> add name=area2 area-id=0.0.0.2
/routing ospf> add network=10.0.1.0/24 area=backbone
```

```
/routing ospf> add network=10.1.2.0/30 area=area2
```

R3 configuration:

```
/routing ospf> add name=area1 area-id=0.0.0.1
/routing ospf> add network=10.1.1.0/30 area=area1
```

R4 configuration:

```
/routing ospf> add name=area2 area-id=0.0.0.2
/routing ospf> add network=10.1.2.0/30 area=area2
```

Now you can check routing table using command */ip route print*

Routing table on router R3:

```
[admin@R3] > ip route print
Flags: X - disabled, A - active, D - dynamic,
C - connect, S - static, r - rip, b - bgp, o - ospf, m - mme,
B - blackhole, U - unreachable, P - prohibit
 #      DST-ADDRESS        PREF-SRC        GATEWAY            DISTANCE
 1 ADo  10.0.1.0/24                        10.1.1.1           110
 2 ADC  10.1.1.0/30         10.1.1.2       ether1             110
 3 ADo  10.1.2.0/30                        10.1.1.1           110
 4 ADC  192.168.1.0/24      192.168.1.1    ether2             0
```

As you can see remote networks 172.16.0.0/16 and 192.168.2.0/24 are not in the routing table, because they are not distributed by OSPF. *Redistribution* feature allows different routing protocols to exchange routing information making possible, for example, to redistribute static or connected routes into OSPF. In our setup we need to redistribute connected network. We need to add following configuration on routers R1, R2 and R3.

```
[admin@R3] /routing ospf instance> set 0 redistribute-connected=as-type-1
[admin@R3] /routing ospf instance> print
Flags: X - disabled
 0   name="default" router-id=0.0.0.0 distribute-default=never
     <u>redistribute-connected=as-type-1</u> redistribute-static=no
     redistribute-rip=no redistribute-bgp=no redistribute-other-ospf=no
     metric-default=1 metric-connected=20 metric-static=20 metric-rip=20
     metric-bgp=auto metric-other-ospf=auto in-filter=ospf-in
     out-filter=ospf-out
```

Now check router R3 to see if routes 192.168.2.0/24 and 172.16.0.0/16 are installed in routing table.

```
[admin@R3] > ip route print
Flags: X - disabled, A - active, D - dynamic,
C - connect, S - static, r - rip, b - bgp, o - ospf, m - mme,
B - blackhole, U - unreachable, P - prohibit
 #      DST-ADDRESS        PREF-SRC        GATEWAY            DISTANCE
 1 ADo  10.0.1.0/24                        10.1.1.1           110
 2 ADC  10.1.1.0/30         10.1.1.2       ether1             110
 3 ADo  10.1.2.0/30                        10.1.1.1           110
 4 ADo  172.16.0.0/16                      10.1.1.1           110
 5 ADC  192.168.1.0/24      192.168.1.1    ether2             0
```

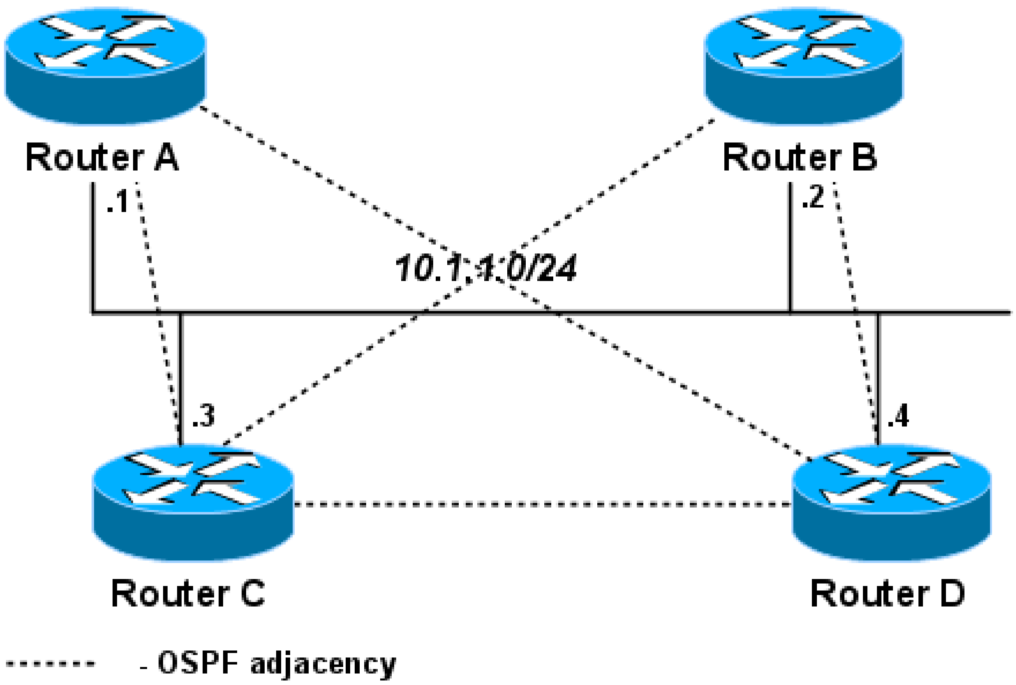
```
6 ADo  192.168.2.0/24                    10.1.1.1             110
```

NBMA networks

OSPF network type NBMA (Non-Broadcast Multiple Access) uses only unicast communications, so it is the preferred way of OSPF configuration in situations where multicast addressing is not possible or desirable for some reasons. Examples of such situations:

- in 802.11 wireless networks multicast packets are not always reliably delivered (read Multicast_and_Wireless for details); using multicast here can create OSPF stability problems;
- using multicast may be not efficient in bridged or meshed networks (i.e. large layer-2 broadcast domains).

Especially efficient way to configure OSPF is to allow only a few routers on a link to become the designated router. (But be careful - if all routers that are capable of becoming the designated router will be down on some link, OSPF will be down on that link too!) Since a router can become the DR only when priority on it's interface is not zero, this priority can be configured as zero in interface and nbma-neighbor configuration to prevent that from happening.



In this setup only C and D are allowed to become designated routers.

On all routers:

```
routing ospf network add network=10.1.1.0/24 area=backbone
routing ospf nbma-neighbor add address=10.1.1.1 priority=0
routing ospf nbma-neighbor add address=10.1.1.2 priority=0
routing ospf nbma-neighbor add address=10.1.1.3 priority=1
routing ospf nbma-neighbor add address=10.1.1.4 priority=1
```

(For simplicity, to keep configuration the same on all routers, nbma-neighbor to self is also added. Normally you wouldn't do that, but it does not cause any harm either.)

Configure interface priorities. On routers A, B:

```
routing ospf interface add interface=ether1 network-type=nbma priority=0
```

On routers C, D (they can become the designated router):

```
routing ospf interface add interface=ether1 network-type=nbma priority=1
```

Results

On Router A:

```
[admin@A] > routing ospf neighbor print
 0 router-id=10.1.1.5 address=10.1.1.5 interface=ether1 priority=1 dr-address=10.1.1.4
   backup-dr-address=10.1.1.3 state="Full" state-changes=6 ls-retransmits=0
   ls-requests=0 db-summaries=0 adjacency=4m53s

 1 router-id=10.1.1.3 address=10.1.1.3 interface=ether1 priority=1 dr-address=1.1.1.4
   backup-dr-address=10.1.1.3 state="Full" state-changes=6 ls-retransmits=0
   ls-requests=0 db-summaries=0 adjacency=4m43s

 2 address=10.1.1.2 interface=ether1 priority=0 state="Down" state-changes=2

 3 address=10.1.1.1 interface=ether1 priority=0 state="Down" state-changes=2
```

On Router D:

```
[admin@D] > routing ospf neighbor print
 0 address=10.1.1.4 interface=ether1 priority=1 state="Down" state-changes=2

 1 router-id=10.1.1.3 address=10.1.1.3 interface=ether1 priority=1 dr-address=10.1.1.4
   backup-dr-address=10.1.1.3 state="Full" state-changes=6 ls-retransmits=0
   ls-requests=0 db-summaries=0 adjacency=6m8s

 2 router-id=10.1.1.2 address=10.1.1.2 interface=ether1 priority=0 dr-address=10.1.1.4
   backup-dr-address=10.1.1.3 state="Full" state-changes=5 ls-retransmits=0
   ls-requests=0 db-summaries=0 adjacency=6m4s

 3 router-id=10.1.1.1 address=10.1.1.1 interface=ether1 priority=0 dr-address=10.1.1.4
   backup-dr-address=10.1.1.3 state="Full" state-changes=5 ls-retransmits=0
   ls-requests=0 db-summaries=0 adjacency=6m4s
```

OSPF Forwarding Address

OSPF may take extra hops at the boundary between OSPF routing domain and another Autonomous System. By looking at the following illustration you can see that even if router R3 is directly connected, packets will travel through the OSPF network and use router R1 as a gateway to other AS.

To overcome this problem, concept of OSPF *forwarding-address* was introduced. This concept allows to say "Send traffic directly to router R1". This is achieved by setting *forwarding address* other than itself in LSA updates indicating that there is an alternate next-hop. Mostly all the time forwarding address is left 0.0.0.0, suggesting that the route is reachable only through the advertising router.

Sere the full example

[Top | Back to Content **]**

Manual:OSPF Case Studies

Applies to RouterOS: v3, v4

Summary

This chapter describes the Open Shortest Path First (OSPF) routing protocol support in RouterOS.

OSPF is Interior Gateway Protocol (IGP) and distributes routing information only between routers belonging to the same Autonomous System (AS).

OSPF is based on link-state technology that has several advantages over distance-vector protocols such as RIP:

- no hop count limitations;
- multicast addressing is used to send routing information updates;
- updates are sent only when network topology changes occur;
- logical definition of networks where routers are divided into areas
- transfers and tags external routes injected into AS.

However there are few disadvantages:

- OSPF is quite CPU and memory intensive due to SPF algorithm and maintenance of multiple copies of routing information;
- more complex protocol to implement compared to RIP;

MikroTik RouterOS implements OSPF version 2 (RFC 2328) and version 3 (RFC 5340, OSPF for IPv6).

OSPF Terminology

Term definitions related to OSPF operations.

- **`Neighbor`** - connected (adjacent) router that is running OSPF with the adjacent interface assigned to the same area. Neighbors are found by Hello packets.
- **`Adjacency`** - logical connection between router and its corresponding DR and BDR. No routing information is exchanged unless adjacencies are formed.
- **`Link`** - link refers to a network or router interface assigned to any given network.
- **`Interface`** - physical interface on the router. Interface is considered as link, when it is added to OSPF. Used to build link database.
- **`LSA`** - Link State Advertisement, data packet contains link-state and routing information, that is shared among OSPF neighbors.
- **`DR`** - Designated Router, chosen router to minimize the number of adjacencies formed. Option is used in broadcast networks.
- **`BDR`** -Backup Designated Router, hot standby for the DR. BDR receives all routing updates from adjacent routers, but it does not flood LSA updates.
- **`Area`** - areas are used to establish a hierarchical network.
- **`ABR`** - Area Border Router, router connected to multiple areas.
- **`ASBR`** - Autonomous System Boundary Router, router connected to an external network (in a different AS).
- **`NBMA`** - Non-broadcast multi-access, networks allow multi-access but have no broadcast capability (for example X.25, Frame Relay). Additional OSPF neighbor configuration is required for those networks.

- `Broadcast` - Network that allows broadcasting, for example Ethernet.
- `Point-to-point` - Network type eliminates the need for DRs and BDRs
- `Router-ID` - IP address used to identify OSPF router. If the OSPF Router-ID is not configured manually, router uses one of the IP addresses assigned to the router as its Router-ID.
- `Link State` - The term link state refers to the status of a link between two routers. It defines the relationship between a router's interface and its neighboring routers.
- `Cost` - Link-state protocols assign a value to each link called cost. the cost value is depend to speed of media. A cost is associated with the outside of each router interface. This is referred to as interface output cost.
- `Autonomous System` - An autonomous system is a group of routers that use a common routing protocol to exchange routing information.

All of these terms are important for understanding the operation of the OSPF and they are used throughout the article.

OSPF Operation

OSPF is a link-state protocol. Interface of the router is considered an OSPF link and state of all the links are stored in **link-state database**.

Link-state routing protocols are distributing, replicating database that describes the routing topology. Each router in routing domain collects local routing topology and sends this information via **link-state advertisements (LSAs)**. LSAs are flooded to all other routers in routing domain and each router generates **link-state database** from received LSAs. The link-state protocol's flooding algorithm ensures that each router has identical link-state database. Each router is calculating routing table based on this link-state database.

OSPF defines several LSA types:

- **type 1** - (*Router LSA*) Sent by routers within the Area, including the list of directly attached links. Does not cross the ABR or ASBR.
- **type 2** - (*Network LSA*) Generated for every “transit network” within an area. A transit network has at least two directly attached OSPF routers. Ethernet is an example of a Transit Network. A Type 2 LSA lists each of the attached routers that make up the transit network and is generated by the DR.
- **type 3** - (*Summary LSA*) The ABR sends Type 3 Summary LSAs. A Type 3 LSA advertises any networks owned by an area to the rest of the areas in the OSPF AS. By default, OSPF advertises Type 3 LSAs for every subnet defined in the originating area, which can cause flooding problems, so it´s a good idea to use a manual summarization at the ABR.
- **type 4** - (*ASBR-Summary LSA*) It announces the ASBR address, it shows “where” the ASBR is located, announcing it´s address instead of it´s routing table.
- **type 5** - (*External LSA*) Announces the Routes learned through the ASBR. External LSAs are flooded to all areas except Stub areas. These LSAs divides in two types: external type 1 and external type2.
- **type 6** - (*Group Membership LSA*) This was defined for Multicast extensions to OSPF and is not used by ROuterOS.
- **type 7** - type 7 LSAs are used to tell the ABRs about these external routes imorted in NSSA area. Area Border Router then translates these LSAs to type 5 external LSAs and floods as normal to the rest of the OSPF network
- **type 8** - (*Link-local only LSA for OSPFv3*)
- **type 9** -
- **type 10** -
- **type 11** -

Note: If we do not have any ASBR, there´s no LSA Types 4 and 5 in the network.

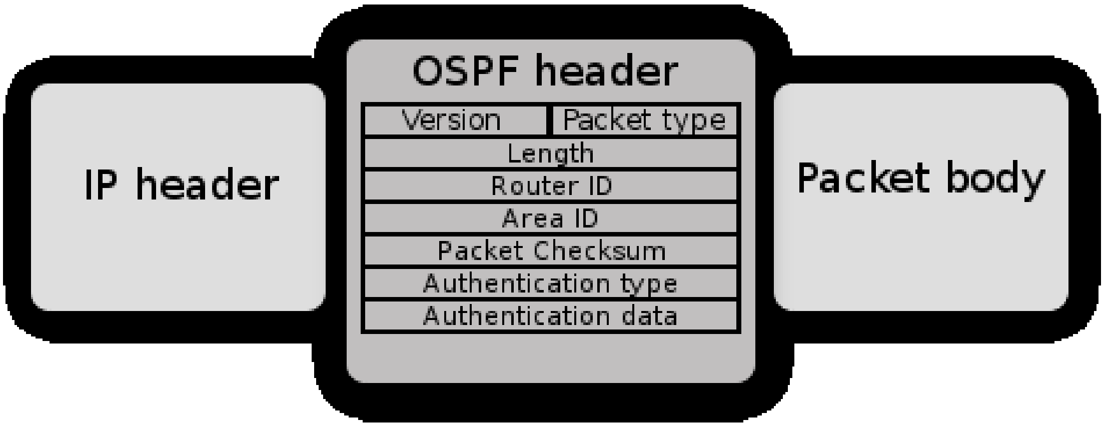
Looking at the link-state database each routing domain router knows how many other routers are in the network, how many interfaces routers have, what networks link between router connects, cost of each link and so on.

There are several steps before OSPF network becomes fully functional:

- Neighbor discovery
- Database Synchronization
- Routing calculation

Communication between OSPF routers

OSPF runs directly over the IP network layer using protocol number 89.
Destination IP address is set to neighbor's IP address or to one of the OSPF multicast addresses AllSPFRouters (224.0.0.5) or AllDRRouters (224.0.0.6). Use of these addresses are described later in this article.
Every OSPF packet begins with standard 24-byte header.



Field	Description
Packet type	There are several types of OSPF packets: Hello packet, Database Description (DD) packet, Link state request packet, link State Update packet and Link State Acknowledgment packet. All of these packets except Hello packet are used in link-state database synchronization
Router ID	one of router's IP addresses unless configured manually
Area ID	Allows OSPF router to associate the packet to the proper OSPF area.
Checksum	Allows receiving router to determine if packet was damaged in transit.
Authentication fields	These fields allow the receiving router to verify that the packet's contents was not modified and that packet really came from OSPF router which Router ID appears in the packet.

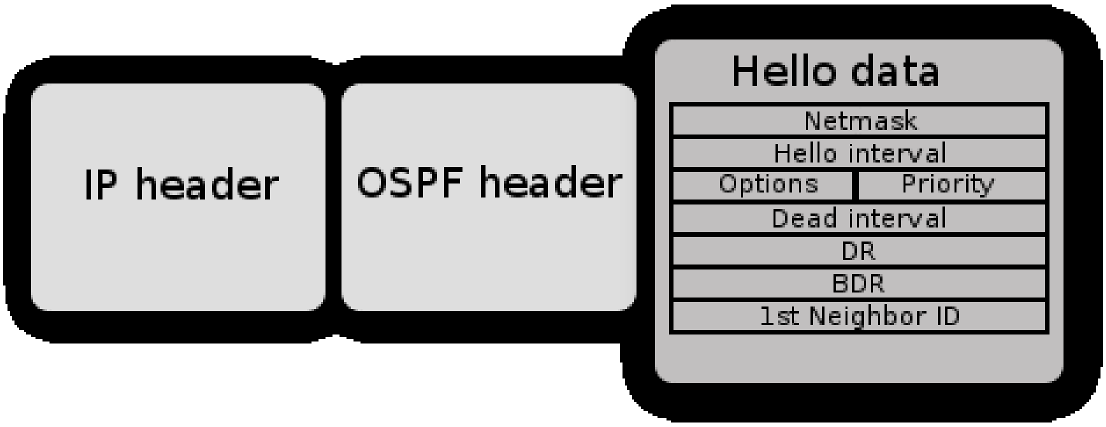
There are five different OSPF packet types used to ensure proper LSA flooding over the OSPF network.

- **Hello packet** - used to discover OSPF neighbors and build adjacencies.
- **Database Description (DD)** - check for Database synchronization between routers. Exchanged after adjacencies are built.
- **Link-State Request (LSR)** - used to request up to date pieces of the neighbor’s database. Out of date parts of routes database are determined after DD exchange.
- **Link-State Update (LSU)** - carries a collection of specifically requested link-state records.
- **Link-State Acknowledgment (LSack)** - is used to acknowledge other packet types that way introducing reliable communication.

Neighbor discovery

Neighbors are discovered by periodically sending OSPF Hello packets out of configured interfaces. By default *Hello packets* are sent out with 10 second interval. This interval can be changed by setting hello interval. Router learns the existence of a neighboring router when it receives the neighbor's Hello in return.

The transmission and reception of Hello packets also allows router to detect failure of the neighbor. If Hello packets are not received within Dead interval (which by default is 40s) router starts to route packets around the failure. Hello protocol ensures that the neighboring routers agree on the Hello interval and Dead interval parameters, preventing situations when not in time received Hello packets mistakenly bring the link down.



Field	Description
`network mask`	The IP mask of the originating router's interface IP address.
`hello interval`	period between Hello packets (default 10s)
`options`	OSPF options for neighbor information
`router priority`	an 8-bit value used to aid in the election of the DR and BDR. (Not set in p2p links)
`router dead interval`	time interval has to be received before consider the neighbor is down. (By default four times bigger than Hello interval)
`DR`	the router-id of the current DR
`BDR`	the router-id of the current BDR
`Neighbor router IDs`	a list of router-ids for all the originating router's neighbors

On each type of network segment Hello protocol works a little different. It is clear that on point-to-point segments only one neighbor is possible and no additional actions are required. However if more than one neighbor can be on the segment additional actions are taken to make OSPF functionality even more efficient.

Note: Network mask, Priority, DR and BDR fields are used only when the neighbors are connected by a broadcast or NBMA network segment.

Two routers do not become neighbors unless the following conditions are met.

- Two way communication between routers is possible. Determined by flooding Hello packets.
- Interface should belong to the same area;
- Interface should belong to the same subnet and have the same network mask, unless it has network-type configured as **point-to-point**;
- Routers should have the same authentication options, and have to exchange same password (if any);
- Hello and Dead intervals should be the same in Hello packets;
- External routing and NSSA flags should be the same in Hello packets.

Discovery on Broadcast Subnets

Attached node to the broadcast subnet can send single packet and that packet is received by all other attached nodes. This is very useful for auto-configuration and information replication. Another useful capability in broadcast subnets is multicast. This capability allows to send single packet which will be received by nodes configured to receive multicast packet. OSPF is using this capability to find OSPF neighbors and detect bidirectional connectivity.

Consider Ethernet network illustrated in image below.

Each OSPF router joins the IP multicast group **AllSPFRouters** (224.0.0.5), then router periodically multicasts its Hello packets to the IP address 224.0.0.5. All other routers that joined the same group will receive multicasted Hello packet. In that way OSPF routers maintain relationships with all other OSPF routers by sending single packet instead of sending separate packet to each neighbor on the segment.

This approach has several advantages:

- Automatic neighbor discovery by multicasting or broadcasting Hello packets.
- Less bandwidth usage compared to other subnet types. On broadcast segment there are n*(n-1)/2 neighbor relations, but those relations are maintained by sending only n Hellos.
- If broadcast has multicast capability, then OSPF operates without disturbing non-OSPF nodes on the broadcast segment. If multicast capability is not supported all routers will receive broadcasted Hello packet even if node is not OSPF router.

Discovery on NBMA Subnets

Nonbroadcast multiaccess (NBMA) segments similar to broadcast supports more than two routers, only difference is that NBMA do not support data-link broadcast capability. Due to this limitation OSPF neighbors must be discovered initially through configuration. On RouterOS NBMA configuration is possible in `/routig ospf nbma-neighbor` menu. To reduce the amount of Hello traffic, most routers attached to NBMA subnet should be assigned Router Priority of 0 (set by default in RouterOS). Routers that are eligible to become Designated Routers should have priority values other than 0. It ensures that during election of DR and BDR Hellos are sent only to eligible routers.

Discovery on PTMP Subnets

Point-to-MultiPoint treats the network as a collection of point-to-point links.

On PTMP subnets Hello protocol is used only to detect active OSPF neighbors and to detect bidirectional communication between neighbors. Routers on PTMP subnets send Hello packets to all other routers that are directly connected to them. Designated Routers and Backup Designated routers are not elected on Point-to-multipoint subnets.

Database Synchronization

Link-state Database synchronization between OSPF routers are very important.
There are two types of database synchronizations:

- initial database synchronization
- reliable flooding.

When the connection between two neighbors first come up, *initial database synchronization* will happen. Unsynchronized databases may lead to calculation of incorrect routing table, resulting in routing loops or black holes.
OSPF is using explicit database download when neighbor connections first come up. This procedure is called

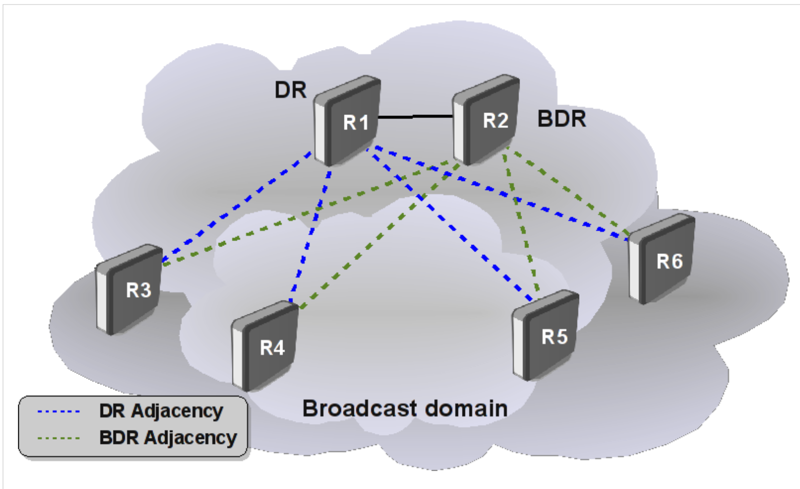
Database exchange. Instead of sending the entire database, OSPF router sends only its LSA headers in a sequence of OSPF **Database Description (DD)** packets. Router will send next DD packet only when previous packet is acknowledged. When entire sequence of DD packets has been received, router knows which LSAs it does not have and which LSAs are more recent. The router then sends **Link-State Request (LSR)** packets requesting desired LSAs, and the neighbor responds by flooding LSAs in **Link-State Update (LSU)** packets. After all updates are received neighbors are said to be **fully adjacent**.

Reliable flooding is another database synchronization method. It is used when adjacencies are already established and OSPF router wants to inform other routers about LSA changes. When OSPF router receives such Link State Update, it installs new LSA in link-state database, sends an acknowledgement packet back to sender, repackages LSA in new LSU and sends it out all interfaces except the one that received the LSA in the first place.

OSPF determines if LSAs are up to date by comparing sequence numbers. Sequence numbers start with 0×80000001, the larger the number, the more recent the LSA is. Sequence number is incremented each time the record is flooded and neighbor receiving update resets Maximum age timer. LSAs are refreshed every 30 minutes, but without a refresh LSA remains in the database for maximum age of 60 minutes.

Databases are not always synchronized between all OSPF neighbors, OSPF decides whether databases needs to be synchronized depending on network segment, for example, on point-to-point links databases are always synchronized between routers, but on ethernet networks databases are synchronized between certain neighbor pairs.

Synchronization on Broadcast Subnets



On broadcast segment there are n*(n-1)/2 neighbor relations, it will be huge amount of Link State Updates and Acknowledgements sent over the subnet if OSPF router will try to synchronize with each OSPF router on the subnet.

This problem is solved by electing one **Designated Router** and one **Backup Designated Router** for each broadcast subnet. All other routers are synchronizing and forming adjacencies only with those two elected routers. This approach reduces amount of adjacencies from n*(n-1)/2 to only 2n-1.

Image on the right illustrates adjacency formations on broadcast subnets. Routers R1 and R2 are Designated Router and Backup Designated router respectively. For example, R3 wants to flood Link State Update (LSU) to both R1 and R2, router sends LSU to IP multicast address **AllDRouters** (224.0.0.6) and only DR and BDR listens to this multicast address. Then Designated Router sends LSU addressed to AllSPFRouters, updating the rest of the routers.

DR election

DR and BDR routers are elected from data received in Hello packet. The first OSPF router on a subnet is always elected as Designated Router, when second router is added it becomes Backup Designated Router. When existing DR or BDR fails new DR or BDR is elected taking into account configured router priority. Router with the highest priority becomes the new DR or BDR.

Being Designated Router or Backup Designated Router consumes additional resources. If Router Priority is set to 0, then router is not participating in the election process. This is very useful if certain slower routers are not capable of being DR or BDR.

Synchronization on NBMA Subnets

Database synchronization on NBMA networks are similar as on broadcast networks. DR and BDR are elected, databases initially are exchanged only with DR and BDR routers and flooding always goes through the DR. The only difference is that Link State Updates must be replicated and sent to each adjacent router separately.

Synchronization on PTMP Subnets

On PTMP subnets OSPF router becomes adjacent to all other routes with which it can communicate directly.

Routing table calculation

When link-state databases are synchronized OSPF routers are able to calculate routing table.
Link state database describes the routers and links that interconnect them and are appropriate for forwarding. It also contains the cost (metric) of each link. This metric is used to calculate shortest path to destination network.
Each router can advertise a different cost for the router's own link direction, making it possible to have asymmetric links (packets to destination travels over one path, but response travels different path). Asymmetric paths are not very popular, because it makes harder to find routing problems.
The Cost in RouterOS is set to 10 on all interfaces by default. Value can be changed in ospf interface configuration menu, for example to add ether2 interface with cost of 100:

```
/routing ospf interface add interface=ether2 cost=100
```

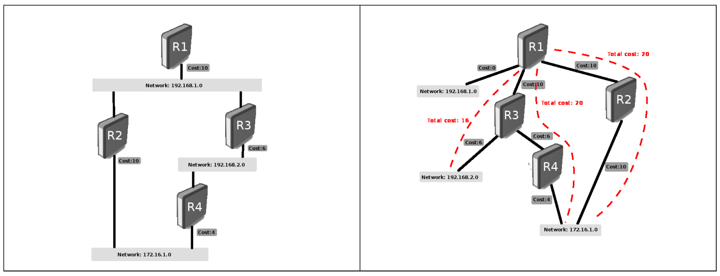
The cost of an interface on Cisco routers is inversely proportional to the bandwidth of that interface. Higher bandwidth indicates lower cost. If similar costs are necessary on RouterOS, then use following formula:

```
Cost = 100000000/bw in bps.
```

OSPF router is using Dijkstra's Shortest Path First (SPF) algorithm to calculate shortest path. The algorithm places router at the root of a tree and calculates shortest path to each destination based on the cumulative cost required to reach the destination. Each router calculates own tree even though all routers are using the same link-state database.

SPT calculation

Assume we have the following network. Network consists of 4(four) routers. OSPF costs for outgoing interfaces are shown near the line that represents the link. In order to build shortest path tree for router R1, we need to make R1 the root and calculate the smallest cost for each destination.



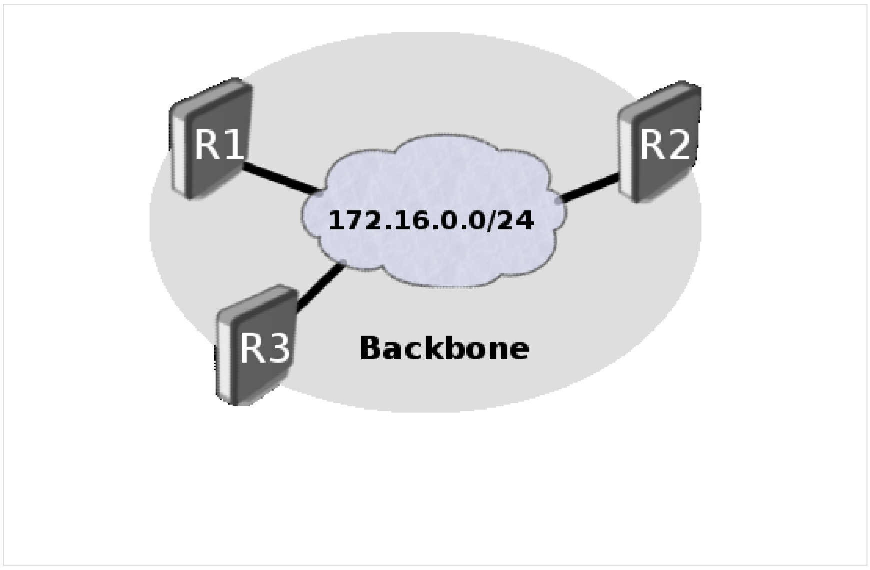
As you can see from image above multiple shortest paths have been found to 172.16.1.0 network, allowing load balancing of the traffic to that destination called equal-cost multipath (ECMP). After the shortest path tree is built, router starts to build the routing table accordingly. Networks are reached consequently to the cost calculated in the tree.

Routing table calculation looks quite simple, however when some of the OSPF extensions are used or OSPF areas are calculated, routing calculation gets more complicated.

Configuring OSPF

Let's look how to configure single-area OSPF network.

One command is required to start OSPF on MikroTik RouterOS - add network in ospf network menu.
Let's assume we have the following network.



It has only one area with three routers connected to the same network 172.16.0.0/24. Backbone area is created during RouterOS installation and additional configuration is not required for area settings.
R1 configuration:

```
/ip address add address=172.16.0.1/24 interface=ether1
/routing ospf network add network=172.16.0.0/24 area=backbone
```

R2 configuration:

```
/ip address add address=172.16.0.2/24 interface=ether1
/routing ospf network add network=172.16.0.0/24 area=backbone
```

R3 configuration:

```
/ip address add address=172.16.0.3/24 interface=ether1
/routing ospf network add network=172.16.0.0/24 area=backbone
```

To verify if OSPF instance is running on router:

```
[admin@MikroTik] /routing ospf> monitor once
             state: running
         router-id: 172.16.0.1
         dijkstras: 6
      db-exchanges: 0
   db-remote-inits: 0
    db-local-inits: 0
  external-imports: 0
```

As you can see OSPF is up and running, notice that **router-id** is set the same as IP address of the router. It was done automatically, because **router-id** was not specified during OSPF configuration.

Add a network to assign interface to the certain area. Look at the OSPF interface menu to verify that dynamic entry was created and correct network type was detected.

```
[admin@MikroTik] /routing ospf interface> print
Flags: X - disabled, I - inactive, D - dynamic, P - passive
 #    INTERFACE         COST  PRIORITY NETWORK-TYPE   AUTHENTICATION AUTHENTICATION-KEY
 0 D  ether1            10    1        broadcast      none
```

Next step is to verify, that both neighbors are found, DR and BDR is elected and adjacencies are established:

```
[admin@MikroTik] /routing ospf neighbor> print
 0 router-id=172.16.0.2 address=172.16.0.2 interface=ether1 priority=1
   dr-address=172.16.0.3 backup-dr-address=172.16.0.2 state="Full" state-changes=5
   ls-retransmits=0 ls-requests=0 db-summaries=0 adjacency=9m2s

 1 router-id=172.16.0.3 address=172.16.0.3 interface=ether1 priority=1
   dr-address=172.16.0.3 backup-dr-address=172.16.0.2 state="Full" state-changes=5
   ls-retransmits=0 ls-requests=0 db-summaries=0 adjacency=6m42s
```

Most of the properties are self explanatory, but if something is unclear, description can be found in neighbor reference manual

Last thing to check whether LSA table is generated properly.

```
[admin@MikroTik] /routing ospf lsa> print
AREA             TYPE         ID              ORIGINATOR      SEQUENCE-NUMBER AGE
backbone         router       172.16.0.1      172.16.0.1      0x80000003      587
backbone         router       172.16.0.2      172.16.0.2      0x80000003      588
backbone         router       172.16.0.3      172.16.0.3      0x80000002      592
backbone         network      172.16.0.3      172.16.0.3      0x80000002      587
```

We have three router links and one network link. All properties are explained in LSA reference manual.

Congratulations, we have fully working OSPF network at this point.

Authentication

It is possible to secure OSPF packets exchange, MikroTik RouterOS provides two authentication methods, simple and MD5. OSPF authentication is disabled by default.
Authentication is configured per interface. Add static ospf interface entry and specify authentication properties to secure OSPF information exchange. md5 authentication configuration on ether1 is shown below:

```
/routing ospf interface
  add interface=ether1 authentication=md5 authentication-key=mySampleKey authentication-key-id=2
```

Simple authentication is plain text authentication method. Method is vulnerable to passive attacks, anybody with packet sniffer can easily get password. Method should be used only to protect OSPF from mis-configurations.

MD5 is a cryptographic authentication and is more preferred. Authentication-key, key-id and OSPF packet content is used to generate message digest that is added to the packet. Unlike the simple authentication method, key is not exchanged over the network.

`Authentication-key-id` value is 1, when authentication is not set (even for router that do not allow to set key id at all).

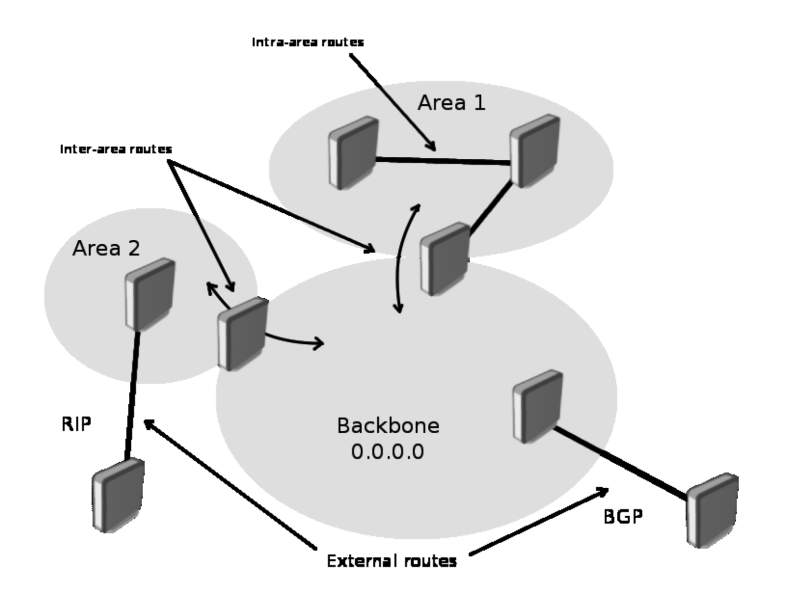
Multi-area networks

Large single area network can produce serious issues:

- Each router recalculates database every time whenever network topology change occurs, the process takes CPU resources.
- Each router holds entire link-state database, which shows the topology of the entire network, it takes memory resources.
- Complete copy of the routing table and number of routing table entries may be significantly greater than the number of networks, that can take even more memory resources.
- Updating large databases require more bandwidth.

To keep routing table size, memory and CPU demands to a manageable levels. OSPF uses a two-layer area hierarchy:

- **backbone (transit) area** - Primary function of this area is the fast and efficient movement of IP packets. Backbone area interconnects other areas and generally, end users are not found within a backbone area.
- **regular area** - Primary function of this area is to connect users and resources. To travel from one are to another, traffic must travel over the backbone, meaning that two regular areas cannot be directly connected. Regular areas have several Subtypes:
  - Standard Area
  - Stub Area
  - Totally Stubby Area
  - Not-so-stubby area (NSSA)



Each area is identified by 32-bit Area ID and has its own link-state database, consisting of router-LSAs and network-LSAs describing how all routers within that area are interconnected. Detailed knowledge of area's topology is hidden from all other areas; router-LSAs and network-LSAs are not flooded beyond the area's borders. Area Border Routers (ABRs) leak addressing information from one area into another in OSPF summary-LSAs. This allows to pick the best area border router when forwarding data to destinations from another area and is called **intra-area routing**.

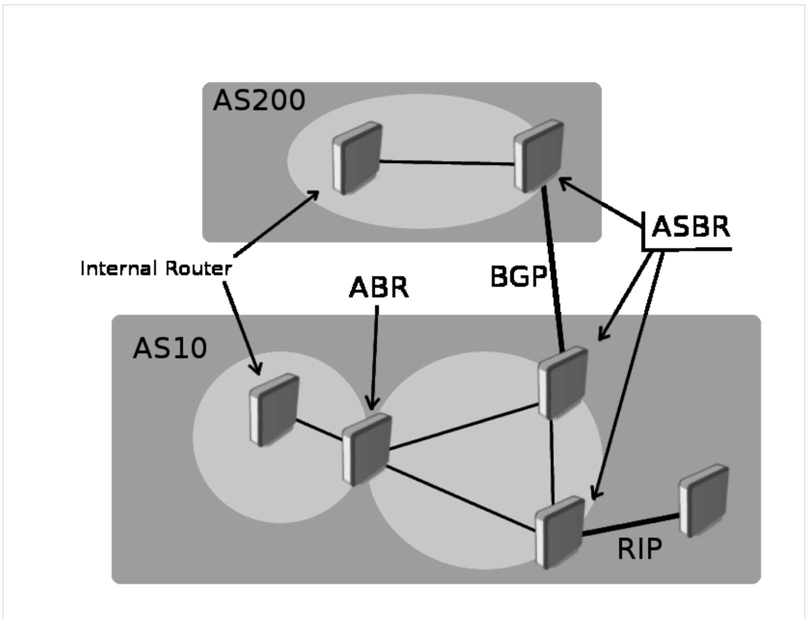
Routing information exchange between areas is essentially Distance Vector algorithm and to prevent algorithm's convergence problems, such as counting to infinity, all areas are required to attach directly to **backbone area** making simple hub-and-spoke topology. Area-ID of backbone area is always 0.0.0.0 and can not be changed.

There are several types of routing information:

- **intra-area routes** - routes generated from within an area (destination belongs to the area).
- **inter-area routes** - routes originated from other areas, also called Summary Routes.
- **external routes** - routes originated from other routing protocols and that are injected into OSPF by redistribution.

External Routing Information

On the edge of an OSPF routing domain, you can find routers called **AS boundary routers (ASBRs)** that run one of other routing protocols. The job of those routers are to import routing information learned from other routing protocols into the OSPF routing domain. External routes can be imported at two separate levels depending on metric type.



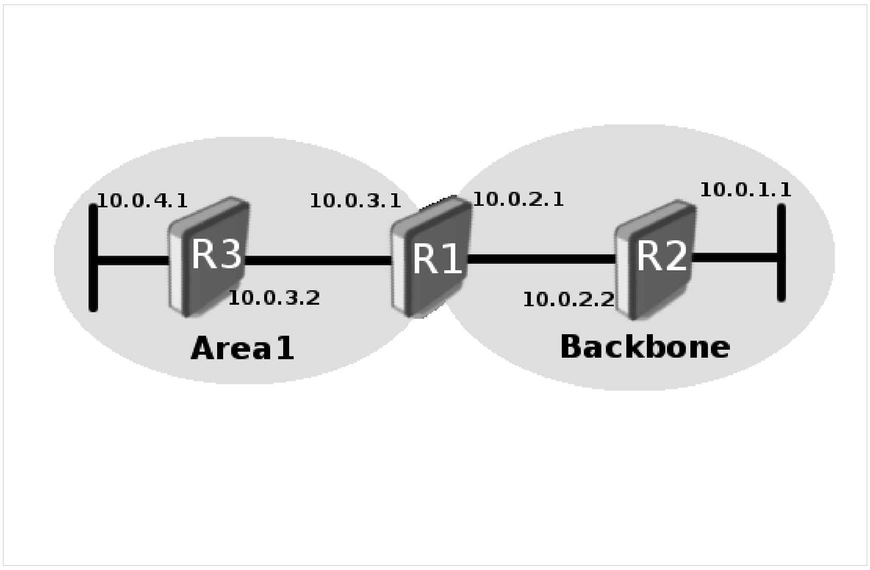
- `type1` - ospf metric is the sum of the internal OSPF cost and the external route cost
- `type2` - ospf metric is equal only to the external route cost.

OSPF provides several area types: backbone area, standard area, stub area and not-so-stubby area. All areas are covered later in the article.

Backbone area is the core of all OSPF network, all areas have to be connected to backbone area. Start configuring OSPF from backbone and then expand network configuration to other areas.

Simple multi-area network

Consider the multi-area network shown below.



R1 configuration:

```
/ip address add address=10.0.3.1/24 interface=ether1
/ip address add address=10.0.2.1/24 interface=ether2
/routing ospf area add name=area1 area-id=1.1.1.1
/routing ospf network add network=10.0.2.0/24 area=backbone
/routing ospf network add network=10.0.3.0/24 area=area1
```

R2 configuration:

```
/ip address add address=10.0.1.1/24 interface=ether2
/ip address add address=10.0.2.2/24 interface=ether1
/routing ospf network add network=10.0.2.0/24 area=backbone
```

R3 configuration:

```
/ip address add address=10.0.3.2/24 interface=ether2
/ip address add address=10.0.4.1/24 interface=ether1
/routing ospf area add name=area1 area-id=1.1.1.1
/routing ospf network add network=10.0.3.0/24 area=area1
```

Route Redistribution

OSPF external routes are routes that are being redistributed from other routing protocols or from static routes.

Remember OSPF configuration setup described in previous section. As you may notice networks 10.0.1.0/24 and 10.0.4.0/24 are not redistributed into OSPF. OSPF protocol does not redistribute external routes by default. Redistribution should be enabled in general OSPF configuration menu to do that. We need to redistribute connected routes in our case, add following configuration to routers R3 and R2:

```
/routing ospf set redistribute-connected=as-type-1
```

Check routing table to see that both networks are redistributed.

```
[admin@MikroTik] /ip route> print
```

Let's add another network to R3:

```
/ip address add address=10.0.5.1/24 interface=ether1
```

10.0.5.0/24 and 10.0.4.0/24 networks are redistributed from R3 over OSPF now. But we do not want other routers to know that 10.0.5.0/24 is reachable over router R3. To achieve it we can add rules in routing filters inside "ospf-out" chain.
Add routing filter to R3

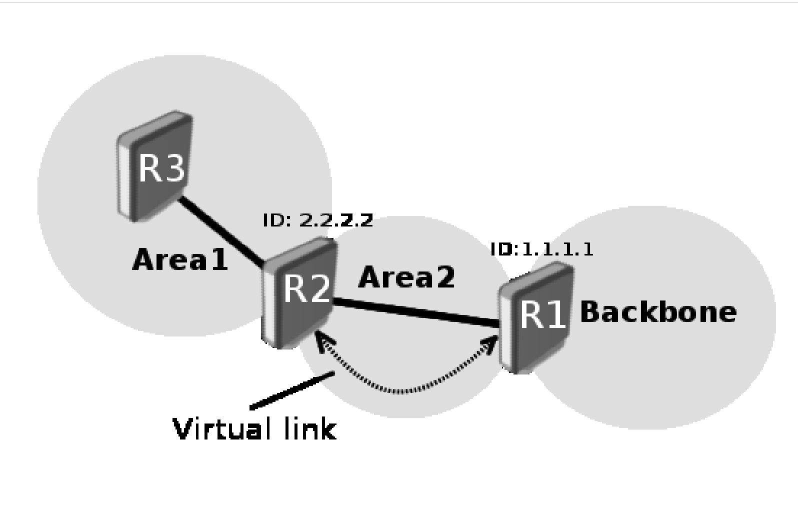
```
/routing filter add chain=ospf-out prefix=10.0.5.0/24 action=discard
```

Routing filters provide two chains to operate with OSPF routes: **ospf-in** and **ospf-out**. Ospf-in chain is used to filter incoming routes and ospf-out is used to filter outgoing routes. More about routing filters can be found in routing filters reference manual.

Virtual Link

All OSPF areas have to be attached to the backbone area, but sometimes physical connection is not possible. In this case areas can be attached logically by using **virtual links**. Also virtual links can be used to glue together fragmented backbone area.

No physical connection to backbone



Area may not have physical connection to backbone, virtual link is used to provide logical path to the backbone of the disconnected area. Link has to be established between two ABRs that have common area with one ABR connected to the backbone.

We can see that both R1 and R2 routers are ABRs and R1 is connected to backbone area. Area2 will be used as **transit area** and R1 is the **entry point** into backbone area. Virtual link has to be configured on both routers.

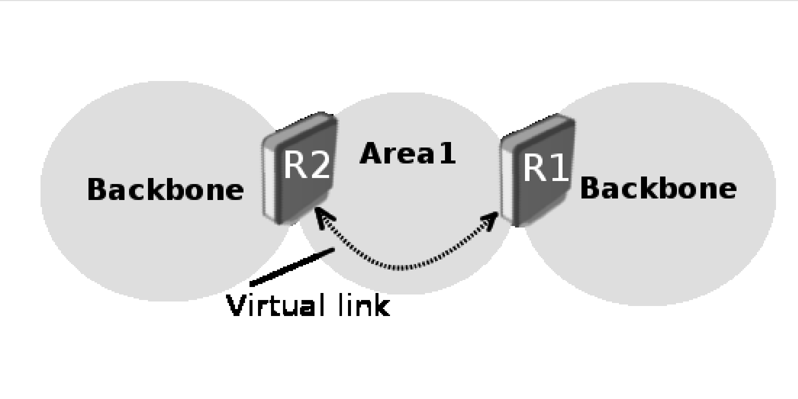
R1 configuration:

```
/routing ospf virtual-link add transit-area=area2 neighbor-id=2.2.2.2
```

R2 configuration:

```
/routing ospf virtual-link add transit-area=area2 neighbor-id=1.1.1.1
```

Partitioned backbone



OSPF allows to link discontinuous parts of the backbone area using virtual links. This might be required when two separate OSPF networks are merged into one large network. Virtual link can be configured between separate ABRs that touch backbone area from each side and have a common area.

Additional area could be created to become transit area, when common area does not exist, it is illustrated in the image above.

Virtual Links are not required for non-backbone areas, when they get partitioned. OSPF does not actively attempt to repair area partitions, each component simply becomes a separate area, when an area becomes partitioned. The backbone performs routing between the new areas. Some destinations are reachable via **intra-area** routing, the area partition requires **inter-area** routing.

However, to maintain full routing after the partition, an address range has not to be split across multiple components of the area partition.

Route Summarization

Route summarization is consolidation of multiple routes into one single advertisement. It is normally done at the area boundaries (Area Border Routers), but summarization can be configured between any two areas.

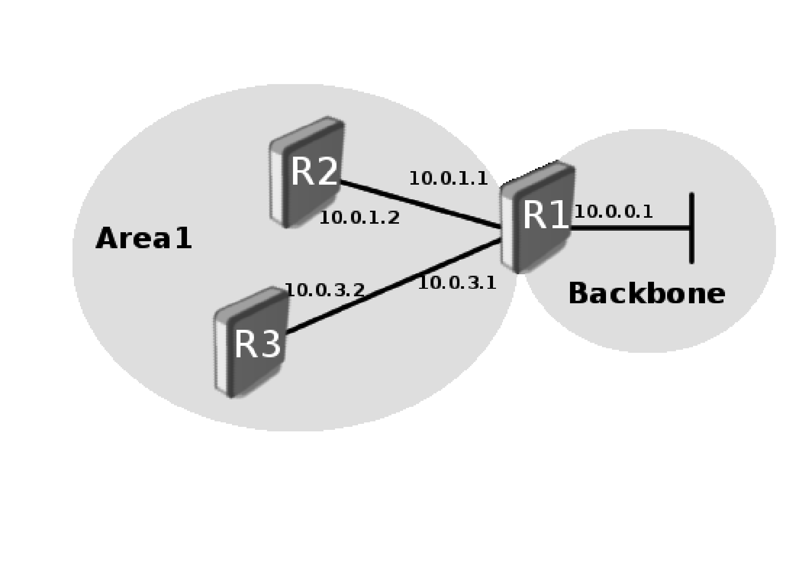
It is better to summarize in the direction to the backbone. Then way the backbone receives all the aggregate addresses and injects them into other areas already summarized. There are two types of summarization: inter-area and external route summarization.

Inter-Area Route Summarization

Inter-area route summarization is done on ABRs, it does not apply to external routes injected into OSPF via redistribution. Summarization configuration is done in OSPF area range menu.

Stub Area

Main purpose of stub areas is to keep such areas from carrying external routes. Routing from these areas to the outside world is based on a default route. Stub area reduces the database size inside an area and reduces memory requirements of routers in the area.



Stub area has few restrictions, ASBR routers cannot be internal to the area, stub area cannot be used as transit area for virtual links. The restrictions are made because stub area is mainly configured not to carry external routes.

Totally stubby area is an extension for stub area. A totally stubby area blocks external routes and summarized (inter-area) routes from going into the area. Only intra-area routes are injected into the area. inject-summary-lsa=no is used to configure totally stubby area in the RouterOS.

Let's consider the example above. Area1 is configured as stub area meaning that routers R2 and R3 will not receive any routing information from backbone area except default route.

R1 configuration:

```
/routing ospf area add name=area1 area-id=1.1.1.1 type=stub inject-summary-lsa=yes
/routing ospf network
add network=10.0.0.0/24 area=backbone
add network=10.0.1.0/24 area=area1
add network=10.0.3.0/24 area=area1
```

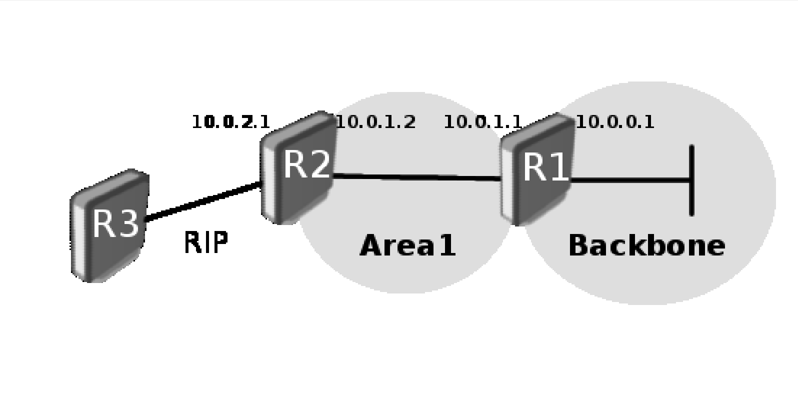
R2 configuration:

```
/routing ospf area add name=area1 area-id=1.1.1.1 type=stub inject-summary-lsa=yes
/routing ospf network
add network=10.0.1.0/24 area=area1
```

R3 configuration:

```
/routing ospf area add name=area1 area-id=1.1.1.1 type=stub inject-summary-lsa=yes
/routing ospf network
add network=10.0.3.0/24 area=area1
```

NSSA



Not-so-stubby area (NSSA) is useful when it is required to inject external routes, but injection of type 5 LSA routes is not required.

Look at the image above. There are two areas (backbone and area1) and RIP connection to area1. We need Area1 to be configured as stub area, but it is also required to inject external routes from RIP protocol. Area1 should be configured as NSSA in this case.

Configuration example does not cover RIP configuration.

R1 configuration:

```
/routing ospf area add name=area1 area-id=1.1.1.1 type=nssa
/routing ospf network
add network=10.0.0.0/24 area=backbone
add network=10.0.1.0/24 area=area1
```

R2 configuration:

```
/routing ospf set redistribute-rip=as-type-1
/routing ospf area add name=area1 area-id=1.1.1.1 type=nssa
/routing ospf network
add network=10.0.1.0/24 area=area1
```

NSSA areas have one another limitation: virtual links cannot be used over such area type.

Related Links

- OSPF Configuration Examples
- OSPF Reference Manual

Manual:OSPF and Point-to-Point interfaces

OSPF configuration on PPP interfaces often is a subject to misunderstanding. You need to keep in mind two things:

1. There is no need to explicitly configure an interface in "/routing ospf interface" to start running OSPF on it. Only "routing ospf network" configuration determines whether the interface will be active or not. If it has matching network network, i.e. the address of the interface falls within range of some network, then the interface will be running OSPF. Else it won't participate in the protocol. "/routing ospf interface" is used only if specific configuration for some interface is needed - typically to configure different link cost.
2. In case of PPP interfaces, the interface will be active if **either** local address or the address of remote are matched against some network. See sample configuration for an illustration. This counterintuitive behaviour will be changed in 3.x routing-test package. Only remote address will be considered there.

- Also remember that running OSPF on a big number of (flapping) PPP interfaces is not recommended.

Configuration example: use local address as OSPF network

Assume we have a PPPoE tunnel between two routers 10.0.0.134 and 10.0.0.133. Configure OSPF on the PPPoE interface on the first router:

```
[admin@I] > /ip address p
Flags: X - disabled, I - invalid, D - dynamic
 #   ADDRESS            NETWORK         BROADCAST       INTERFACE
 0   10.0.0.133/24      10.0.0.0        10.0.0.255      ether1
 1 D 10.1.1.254/32      10.1.1.1        0.0.0.0         pppoe-out1
[admin@I] > routing ospf network add network=10.1.1.254/32 area=backbone
```

Do the same on the second router:

```
[admin@II] > /ip address p
Flags: X - disabled, I - invalid, D - dynamic
 #   ADDRESS            NETWORK         BROADCAST       INTERFACE
 0   10.0.0.134/24      10.0.0.0        10.0.0.255      ether1
 1 D 10.1.1.1/32        10.1.1.254      0.0.0.0         <pppoe-atis>
[admin@II] > routing ospf network add network=10.1.1.1/32 area=backbone
```

An OSPF adjacency has been established; neighbor at 10.1.1.1 is in 'Full' state:

```
[admin@I] > routing ospf neighbor pr
 router-id=10.0.0.133 address=10.1.1.254 priority=1 dr-address=0.0.0.0
   backup-dr-address-id=0.0.0.0 state="2-Way" state-changes=0 ls-retransmits=0
   ls-requests=0 db-summaries=0
 router-id=10.0.0.134 address=10.1.1.1 priority=1 dr-address=0.0.0.0
   backup-dr-address-id=0.0.0.0 state="Full" state-changes=5 ls-retransmits=0
   ls-requests=0 db-summaries=0
[admin@I] >
```

External links

- OSPF in MT manual [1]
- OSPF RFC [2]

References

[1] http://www.mikrotik.com/docs/ros/2.9/routing/ospf
[2] http://rfc-ref.org/RFC-TEXTS/2328/contents.html

Article Sources and Contributors

Manual:Routing/OSPF *Source*: http://wiki.mikrotik.com/index.php?oldid=22892 *Contributors*: Janisk, Marisb, Normis, Route

Manual:OSPF-examples *Source*: http://wiki.mikrotik.com/index.php?oldid=22871 *Contributors*: Janisk, Marisb, Normis, Route

Manual:OSPF Case Studies *Source*: http://wiki.mikrotik.com/index.php?oldid=23058 *Contributors*: Janisk, Marisb

Manual:OSPF and Point-to-Point interfaces *Source*: http://wiki.mikrotik.com/index.php?oldid=17390 *Contributors*: Atis, Eep, Marisb

Image Sources, Licenses and Contributors

Image:Version.png *Source*: http://wiki.mikrotik.com/index.php?title=File:Version.png *License*: unknown *Contributors*: Normis
Image:Icon-note.png *Source*: http://wiki.mikrotik.com/index.php?title=File:Icon-note.png *License*: unknown *Contributors*: Marisb, Route
Image:image6005.gif *Source*: http://wiki.mikrotik.com/index.php?title=File:Image6005.gif *License*: unknown *Contributors*: Andriss
Image:image6006.gif *Source*: http://wiki.mikrotik.com/index.php?title=File:Image6006.gif *License*: unknown *Contributors*: Andriss
Image:ospf-nbma.png *Source*: http://wiki.mikrotik.com/index.php?title=File:Ospf-nbma.png *License*: unknown *Contributors*: Route
Image:ospf-header.png *Source*: http://wiki.mikrotik.com/index.php?title=File:Ospf-header.png *License*: unknown *Contributors*: Marisb
Image:ospf-hello.png *Source*: http://wiki.mikrotik.com/index.php?title=File:Ospf-hello.png *License*: unknown *Contributors*: Marisb
Image:ospf-adjacency.png *Source*: http://wiki.mikrotik.com/index.php?title=File:Ospf-adjacency.png *License*: unknown *Contributors*: Marisb
Image:sp-net.png *Source*: http://wiki.mikrotik.com/index.php?title=File:Sp-net.png *License*: unknown *Contributors*: Marisb
Image:sp-tree.png *Source*: http://wiki.mikrotik.com/index.php?title=File:Sp-tree.png *License*: unknown *Contributors*: Marisb
Image:ospf-basic.png *Source*: http://wiki.mikrotik.com/index.php?title=File:Ospf-basic.png *License*: unknown *Contributors*: Marisb
Image:backbone-s.png *Source*: http://wiki.mikrotik.com/index.php?title=File:Backbone-s.png *License*: unknown *Contributors*: Marisb
Image:area-br.png *Source*: http://wiki.mikrotik.com/index.php?title=File:Area-br.png *License*: unknown *Contributors*: Marisb
Image:basic-multi-area.png *Source*: http://wiki.mikrotik.com/index.php?title=File:Basic-multi-area.png *License*: unknown *Contributors*: Marisb
Image:vlink-area.png *Source*: http://wiki.mikrotik.com/index.php?title=File:Vlink-area.png *License*: unknown *Contributors*: Marisb
Image:vlink-backbone.png *Source*: http://wiki.mikrotik.com/index.php?title=File:Vlink-backbone.png *License*: unknown *Contributors*: Marisb
Image:stub-example.png *Source*: http://wiki.mikrotik.com/index.php?title=File:Stub-example.png *License*: unknown *Contributors*: Marisb
Image:nssa-example.png *Source*: http://wiki.mikrotik.com/index.php?title=File:Nssa-example.png *License*: unknown *Contributors*: Marisb