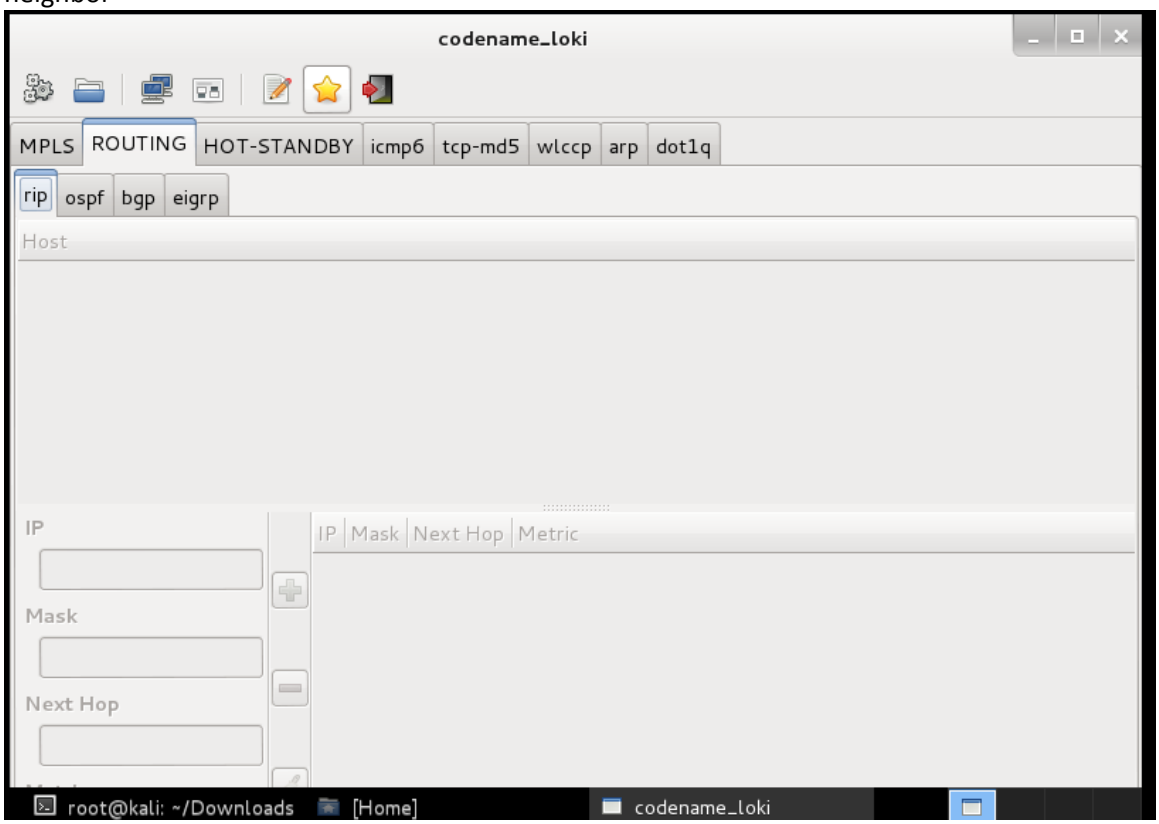
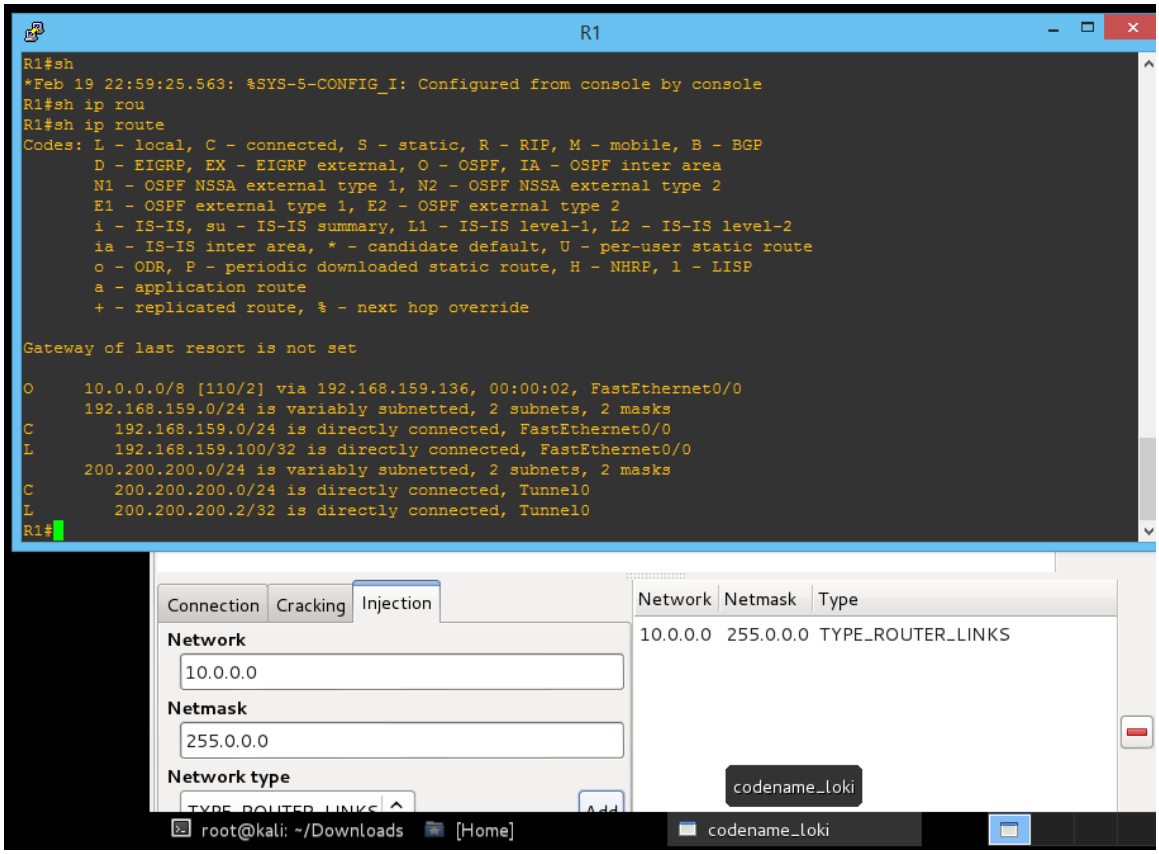
Introduction to Routers Authentication

Router Authentication means not any router can be a neighbor to any other router, plus all routes updates will be send after authentication happens.

But why Authentication is important?
Have a look to this tool “Loki” its python tool can be installed in Linux distros such as Kali OS
Using Loki I can for example tell a OSPF router that I am OSPF router too and force him to be my neighbor



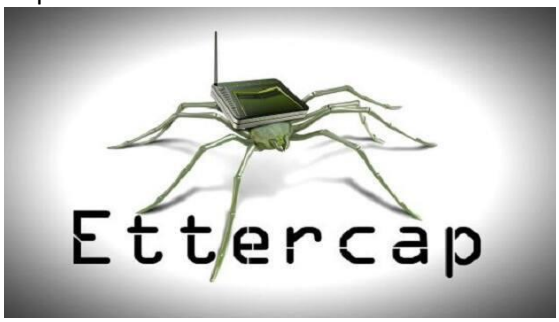
I can even send him fake routes or even flood his routing table.



Authentication means both routers should agree to be neighbor if they had the same password.

This password will be exchanged between both routers , it can be exchanged as plain text or as hashed characters , but sending it as plain text is not a good option since its vulnerable to sniffing attacks.

Sniffing attacks means someone between the two routers can sniff traffic (have a copy of it using tools like ettercap & wireshark) and he will easily read what the password is since it is sent as plain text.



So we should secure our routers by configuring IGP (RIPv2,OSPF,EIGRP) or EGP (BGP) authentication.

How Neighbor (Router) Authentication Works
When neighbor authentication has been configured on a router, the router authenticates the source of each routing update packet that it receives. This is accomplished by the exchange of an authenticating key (sometimes referred to as a password) that is known to both the sending and the receiving router.

There are two types of neighbor authentication used: plain text authentication and Message Digest Algorithm Version 5 (MD5) authentication. Both forms work in the same way, with the exception that MD5 sends a "message digest" instead of the authenticating key itself. The message digest is created using the key and a message, but the key itself is not sent, preventing it from being read while it is being transmitted. Plain text authentication sends the authenticating key itself over the wire.

Key and key chain Concept

Before we start with configuration commands, let me explain the concept of **Key and key chain**

Each Password will be tied to Key , Key needs a number as identifier
One or more Key will be tied to Key Chain , key chain need a name as identifier

Picture above shows that I have Key chain named “MYHOME”
This key chain had three keys
Key number 1 has password “cisco”
Key number 2 has password “cisco123”
Key number 3 has password “CISCO”

We will use this “Key and key-chain” concept with RIPv2 and EIGRP

RIPv2 Authentication

RIPv2 support Two Types of Authentication:

- Plain-text authentication
- MD5-based authentication

Plain-text authentication Configuration (in both routers)

```
key chain MYHOME
key 1   < key id can be from 0 to 2147483647
key-string cisco  < string(password) from 1 to 80 uppercase and lowercase alphanumeric characters, except that the first character cannot be a number.

int f0/0
ip rip authentication key-chain MYHOME
```

MD5-based authentication Configuration (in both routers)

```
key chain MYHOME
key 1
key-string cisco

int f0/0
ip rip authentication key-chain MYHOME
ip rip authentication mode md5
```

So, RIP authentication is configured in three steps:

1. Define a global key chain.
2. Enable authentication mode (Plain-text or MD5) on the RIP interfaces.
3. Apply the key chain to the interface.

Above configuration is nearly identical to both EIGRP and IS-IS authentication, with minor syntax differences at the interface level of how the authentication is enabled.

Unlike Plain-text Authentication, For MD5-based authentication, **the key number in the key chain must match between the neighbors (if you have more than one key id but if not it will work fine such as R1 had only key 2 with string cisco and R2 had only key 1 with string cisco BUT if R1 for instance had key 1 with string CISCO and key 2 with string cisco, authentication will failed),** because this value is used as a salt for the MD5 hash.

To verify successful authentication we can use "debug ip rip"command and you should see a message like the one below

***Jul 10 23:24:39.491: RIP: received packet with MD5 authentication**

RIPng does not have an authentication mechanism of its own; it relies on the authentication feature built into IPv6.

EIGRP Authentication

Authentication with EIGRP depends on which way of EIGRP configuration we are going to use , Traditional or Named mode.

Traditional Mode

EIGRP authentication works similar to RIP and IS-IS authentication in that it is key-chain based. Unlike RIP and IS-IS authentication, though, **Traditional EIGRP only supports MD5**, not Plain-text (aka Clear-text).

MD5-based authentication Configuration (in both routers)

```
key chain MYHOME
key 1
key-string cisco

int f0/0
ip authentication mode eigrp 100 md5
ip authentication key-chain eigrp 100 MYHOME
```

In EIGRPv6 Traditional mode for IPv6 networks it will be same commands but using "ipv6" instead of "ip"

Something extra here we should mention here , we can pre-stage the routers to periodically change their passwords used for authentication in RIPv2 , EIGRP and IS-IS.

Example

KEY1 should be sent between 00:00:00 Jan 1 2013 and 00:00:00 Jan 1 2014.
KEY1 should be accepted between 00:00:00 Jan 1 2013 and 01:00:00 Jan 1 2014.
KEY2 should be sent between 00:00:00 Jan 1 2014 and 00:00:00 Jan 1 2015.
KEY2 should be accepted between 23:00:00 Dec 31 2013 and 01:00:00 Jan 1 2015.
KEY3 should be sent between 00:00:00 Jan 1 2015 and 00:00:00 Jan 1 2016.
KEY3 should be accepted between 23:00:00 Dec 31 2014 and 01:00:00 Jan 1 2016.

```
key chain MYHOME
 key 1
  key-string cisco
  accept-lifetime 00:00:00 Jan 1 2013 01:00:00 Jan 1 2014
  send-lifetime 00:00:00 Jan 1 2013 00:00:00 Jan 1 2014
 key 2
  key-string cisco123
  accept-lifetime 23:00:00 Dec 31 2013 01:00:00 Jan 1 2015
  send-lifetime 00:00:00 Jan 1 2014 00:00:00 Jan 1 2015
 key 3
  key-string CISCO
  accept-lifetime 23:00:00 Dec 31 2014 01:00:00 Jan 1 2016
  send-lifetime 00:00:00 Jan 1 2015 00:00:00 Jan 1 2016
```

accept-lifetime start-time {**infinite** | end-time | duration seconds}

infinite
Indicates the key is valid for use on received packets from the start-time value on.

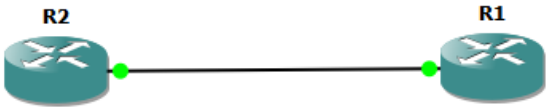
end-time
Indicates the key is valid for use on received packets from the start-time value until the end-time value. The syntax is the same as that for the start-time value. The end-time value must be after the start-time value. The default end time is an infinite time period.

seconds
Length of time (in seconds) that the key is valid for use on received packets.
The range is from 1 to 2147483646.

send-lifetime start-time {infinite | end-time | duration seconds}

specify the time period during which this key can be used for sending packets using the key-chain-key configuration command. Table 3-4 describes the parameters for this command.

Tradtional Mode EIGRPv6 Authentication Lab



Basic Configuration
R1
ipv6 unicast-routing
int f0/0
ipv6 add 2001:12:12:12::1/64
int loop 0
ipv6 add 2001:1:1:1::1/64

R2
ipv6 unicast-routing
int f0/0
ipv6 add 2001:12:12:12::2/64

MD5 Authentication Configuration
R1
ipv6 router eigrp 100
eigrp router-id 0.0.0.1
no shut
int loop 0
ipv6 eigrp 100

key chain MYHOME
key 1
key-string cisco
int f0/0
ipv6 eigrp 100
ipv6 authentication mode eigrp 100 md5
ipv6 authentication key-chain eigrp 100 MYHOME

R2
ipv6 router eigrp 100
eigrp router-id 0.0.0.2
no shut

key chain MYHOME
key 1
key-string cisco
int f0/0
ipv6 eigrp 100
ipv6 authentication mode eigrp 100 md5
ipv6 authentication key-chain eigrp 100 MYHOME

```
R2#sh ipv6 route eigrp
D   2001:1:1:1::/64 [90/156160]
     via FE80::C801:16FF:FE50:0, FastEthernet0/0
```

Named mode

EIGRP **supports MD5 authentication and SHA-256** in Multi-AF (Named) Mode.
For MD5 authentication in both Classic and Named modes, the key chain is defined globally.
Note that the key ID must match for authentication to occur, because this number is exchanged in the hello packets.
In Tradtional Mode, the authentication is applied at the link level, whereas in Named Mode it is applied at the af-interface mode under the address-family.

MD5-based authentication Configuration (in both routers)

```
key chain MYHOME
 key 1
  key-string cisco
!
router eigrp TEST
address-family ipv4 unicast autonomous-system 100
 af-interface GigabitEthernet0/0
  authentication mode md5
  authentication key-chain MYHOME
 exit-af-interface
```

SHA-256-based authentication Configuration (in both routers)

```
router eigrp TEST
address-family ipv4 unicast autonomous-system 100
af-interface g0/0
 authentication mode hmac-sha-256 cisco
 exit-af-interface
```

Notice that SHA-256 configuration doesn't need Key chain commands

In EIGRPv6 named mode for IPv6 networks it will be same configuration but we will use address-family **ipv6** unicast autonomous-system 100 **Instead of using** address-family **ipv4** unicast autonomous-system 100

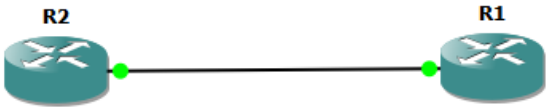
Verify Successful MD5 Authentication

```
R1#debug eigrp packets
*Apr 21 16:38:51.745: EIGRP: received packet with MD5 authentication, key id = 1
```

Verify unsuccessful MD5 Authentication

```
R2#debug eigrp packets
*Apr 21 16:50:18.749: EIGRP: pkt key id = 2, authentication mismatch
*Apr 21 16:50:18.749: EIGRP: Serial0/0/1: ignored packet from 192.168.1.101, opcode = 5
  (invalid authentication)
*Apr 21 16:50:18.749: EIGRP: Dropping peer, invalid authentication
*Apr 21 16:50:18.753: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 100: Neighbor 192.168.1.101
(Serial0/0/1) is down: Auth failure
```

Named Mode EIGRPv6 Authentication Lab



Basic Configuration

R1

```
ipv6 unicast-routing
int f0/0
ipv6 add 2001:12:12:12::1/64

int loop 0
ipv6 add 2001:1:1:1::1/64
```

R2

```
ipv6 unicast-routing
int f0/0
ipv6 add 2001:12:12:12::2/64
```

Named Mode Support MD5 or SHA Authentication , let try each one of them

MD5 Authentication Configuration

R1

```
key chain MYHOME
key 1
key-string cisco

router eigrp MYEIGRP
address-family ipv6 autonomous-system 100
no shutdown
eigrp router-id 0.0.0.1
af-interface f0/0
authentication mode md5
authentication key-chain MYHOME
exit
af-interface  loop 0
exit
exit
exit
```

R2

```
key chain MYHOME
key 1
key-string cisco

router eigrp MYEIGRP
address-family ipv6 autonomous-system 100
```

```
no shutdown
eigrp router-id 0.0.0.2
af-interface f0/0
authentication mode md5
authentication key-chain MYHOME
exit
exit
exit

R2#sh ipv6 route eigrp
D   2001:1:1:1::/64 [90/103040]
    via FE80::C801:EFF:FE94:0, FastEthernet0/0
```

EIGRP/SAF HMAC-SHA-256 Authentication Configuration

R1

```
router eigrp MYEIGRP
address-family ipv6 autonomous-system 100
no shutdown
eigrp router-id 0.0.0.1
af-interface f0/0
authentication mode hmac-sha-256 0 cisco
authentication key-chain MYHOME
exit
af-interface  loop 0
exit
exit
exit
```

R2

```
router eigrp MYEIGRP
address-family ipv6 autonomous-system 100
no shutdown
eigrp router-id 0.0.0.2
af-interface f0/0
authentication mode hmac-sha-256 0 cisco
authentication key-chain MYHOME
exit
exit
exit

R2#sh ipv6 route eigrp
D   2001:1:1:1::/64 [90/103040]
    via FE80::C801:EFF:FE94:0, FastEthernet0/0
```

Notice with EIGRP Named Mode Authentication using SHA we do not need Key Chain

OSPFv2 Authentication

There are three different types of authentication available for OSPF version 2:
1) Null authentication(Type 0): Null authentication means that there is no authentication, which is the default on Cisco routers. It means no password will be used.
under interface we type **ip ospf authentication null**

Null authentication means that basically authentication is disabled.

The use case for this authentication type is when for example you have globally configured clear-text or MD5/SHA authentication for one OSPF area, but want one or multiple interfaces in that area to actually use no authentication. Because the interface level configuration overrides the area level configuration, interfaces for which ip ospf authentication null is configured will require no OSPF authentication.

2) Plain-text [Clear text] authentication (Type 1): 64 bit passwords will be used but are exchanged in clear text on the network.

3) MD5 authentication (Type 2): passwords will be used but are exchanged in Message Digest type 5 format , since we are using MD5 , password format will be represented in 128 bit , each password will tied to key id number .
the MD5 key number must match between the neighbors, because it is a salt for the MD5 hash

Authentication can be enabled in one of two ways:

1. Under router configuration mode globally applies to all interfaces in that area.
2. Under interface applied to this interface only.

Plain-text [Clear text] authentication example:
per area authentication example:

```
router ospf 1
area0 authentication

int f0/0
ip ospf authentication-key cisco
```

md5 authentication Under interface example:

```
int f0/0
ip ospf authentication
ip ospf authentication-key cisco
```

Note: The authentication key, however, is always configured at the interface level.

```
Router# debug ip ospf packet
OSPF: rcv. v:2 t:1 l:48 rid:200.0.0.117
      aid:0.0.0.0 chk:6AB2 aut:0 auk:
```

md5 authentication Under router configuration example:
per area authentication example:
router ospf 1
area 0 authentication message-digest

int f0/0
ip ospf message-digest-key 1 md5 cisco

md5 authentication Under interface example:
int f0/0
ip ospf authentication message-digest
ip ospf message-digest-key 1 md5 cisco

When OSPF authentication is enabled at the interface level, it overrides the configuration of the global process, but only for that particular interface.

If keys is mismatch , and we run debug ip ospf adj

*Jul 12 16:26:10.191: OSPF-100 ADJ Fa0/0: Send with youngest Key 2
*Jul 12 16:26:15.579: OSPF-100 ADJ Fa0/0: Rcv pkt from 10.13.13.1 : Mismatched Authentication Key - No message digest key 1 on interface

```
Router# debug ip ospf packet
OSPF: rcv. v:2 t:1 l:48 rid:200.0.0.116
      aid:0.0.0.0 chk:0 aut:2 keyid:1 seq:0x0
```

Field	Description
v:	OSPF version.
t:	OSPF packet type. Possible packet types follow: • 1--Hello • 2--Data description • 3--Link state request • 4--Link state update • 5--Link state acknowledgment
l:	OSPF packet length in bytes.
rid:	OSPF router ID.
aid:	OSPF area ID.
chk:	OSPF checksum.
aut:	OSPF authentication type. Possible authentication types follow: • 0--No authentication • 1--Simple password • 2--MD5
keyid:	MD5 key ID.
seq:	Sequence number.

Also we should remember that you can use "show ip ospf interface" command to know if authentication enabled or not.

```
R1# show ip ospf interface
FastEthernet0/0 is up, line protocol is up
  Internet Address 192.168.0.1/30, Area 0
  Process ID 1, Router ID 1.1.1.1, Network Type BROADCAST, Cost: 10
  Enabled by interface config, including secondary ip addresses
  Transmit Delay is 1 sec, State BDR, Priority 1
  Designated Router (ID) 2.2.2.2, Interface address 192.168.0.2
  Backup Designated router (ID) 1.1.1.1, Interface address 192.168.0.1
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:07
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 2.2.2.2  (Designated Router)
  Suppress hello for 0 neighbor(s)
  Message digest authentication enabled
      No key configured, using default key id 0
```

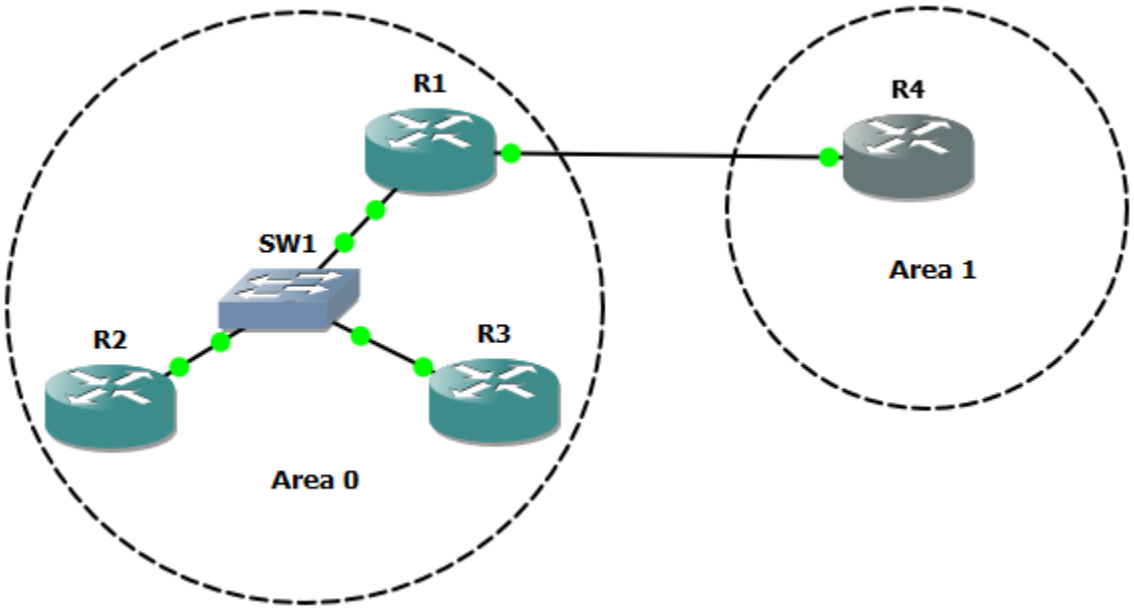
If everything is ok and configured , you should see in the above output

```
Youngest key id is 1
```

Instead of

```
Message digest authentication enabled
    No key configured, using default key id 0
```

OSPFv2 IPv4 Authentication Lab



Basic Configuration

R1

int f0/0
ip add 10.123.123.1 255.255.255.0
int f1/0
ip add 10.14.14.1 255.255.255.0

router ospf 100
router-id 0.0.0.1
network 10.123.123.0 0.0.0.255 area 0
network 10.14.14.0 0.0.0.255 area 1

R2

int f0/0
ip add 10.123.123.2 255.255.255.0
router ospf 100
router-id 0.0.0.2
network 10.123.123.0 0.0.0.255 area 0

R3

int f0/0
ip add 10.123.123.3 255.255.255.0

router ospf 100
router-id 0.0.0.3
network 10.123.123.0 0.0.0.255 area 0

R4
int f1/0
ip add 10.14.14.4 255.255.255.0
router ospf 100
router-id 0.0.0.4
network 10.14.14.0 0.0.0.255 area 1

- **Configure area 0 routers with md5 authentication per area**
- **configure R1 and R4 with md5 authentication per interface**

R1
router ospf 100
area 0 authentication message-digest

int f0/0
ip ospf message-digest-key 1 md5 cisco

int f1/0
ip ospf authentication message-digest
ip ospf message-digest-key 1 md5 cisco

R4
int f1/0
ip ospf authentication message-digest
ip ospf message-digest-key 1 md5 cisco

R2
router ospf 100
area 0 authentication message-digest

int f0/0
ip ospf message-digest-key 1 md5 cisco

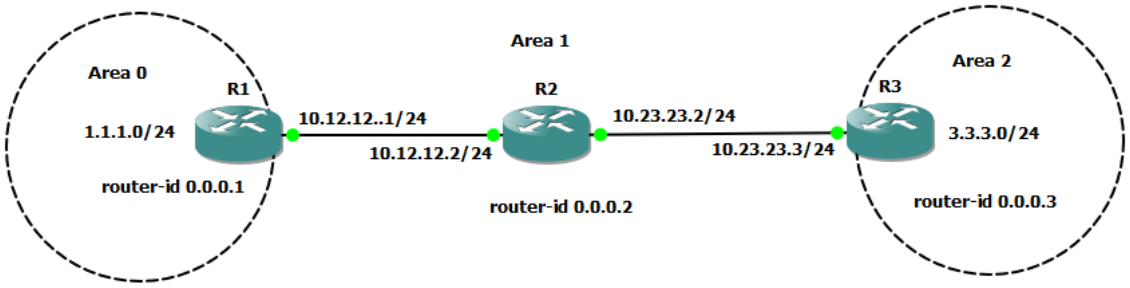
R3
router ospf 100
area 0 authentication message-digest

int f0/0
ip ospf message-digest-key 1 md5 cisco

R3#sh ip route ospf
O IA 10.14.14.0/24 [110/2] via 10.123.123.1, 00:06:41, FastEthernet0/0
R1#sh ip ospf interface f1/0 | i Message digest
Message digest authentication enabled
R1#sh ip ospf 100 | i Area |authentication
Area BACKBONE(0)
Area has message digest authentication
Area 1
Area has no authentication

OSPF virtual link authentication

Virual link is a interface in area 0.
This mean if you enable authentication on Area 0 it will automatically turn authentication on virtual link but as discussed above password(Key) must need to enable on interface.
As we know Virtual link doesnt have any interface on which you can configure authentication,authentication on virtual link can be configure using"**area virtual-link**" command under OSPF process.



Basic Configuration

R1

```
interface Loopback0
 ip address 1.1.1.1 255.255.255.0

router ospf 100
 router-id 0.0.0.1
 area 1 virtual-link 0.0.0.3
 network 1.1.1.0 0.0.0.255 area 0
 network 10.12.12.0 0.0.0.255 area 1
```

R2

```
router ospf 100
 router-id 0.0.0.2
 network 10.12.12.0 0.0.0.255 area 1
 network 10.23.23.0 0.0.0.255 area 1
```

R3

```
interface Loopback0
 ip address 3.3.3.3 255.255.255.0

router ospf 100
 router-id 0.0.0.3
 area 1 virtual-link 0.0.0.1
 network 3.3.3.0 0.0.0.255 area 2
 network 10.23.23.0 0.0.0.255 area 1
```

Virtual Link Authentication Configuration

R1

```
int loop0
 ip ospf message-digest-key 1 md5 cisco

router ospf 100
area 0 authentication message-digest
 area 1 virtual-link 0.0.0.3 message-digest-key 1 md5 cisco
```

R3

```
router ospf 100
area 0 authentication message-digest
area 1 virtual-link 0.0.0.1 message-digest-key 1 md5 cisco
```

OSPF SHA Authentication

Lately Type 2 OSPF authentication was extended to offer support for HMAC-SHA based authentication

SHA types available to be used are :

- HMAC-SHA-1 (160 bits digest)
- HMAC-SHA-256 (256 bits digest)
- HMAC-SHA-384 (384 bits digest)
- HMAC-SHA-512 (512 bits digest)

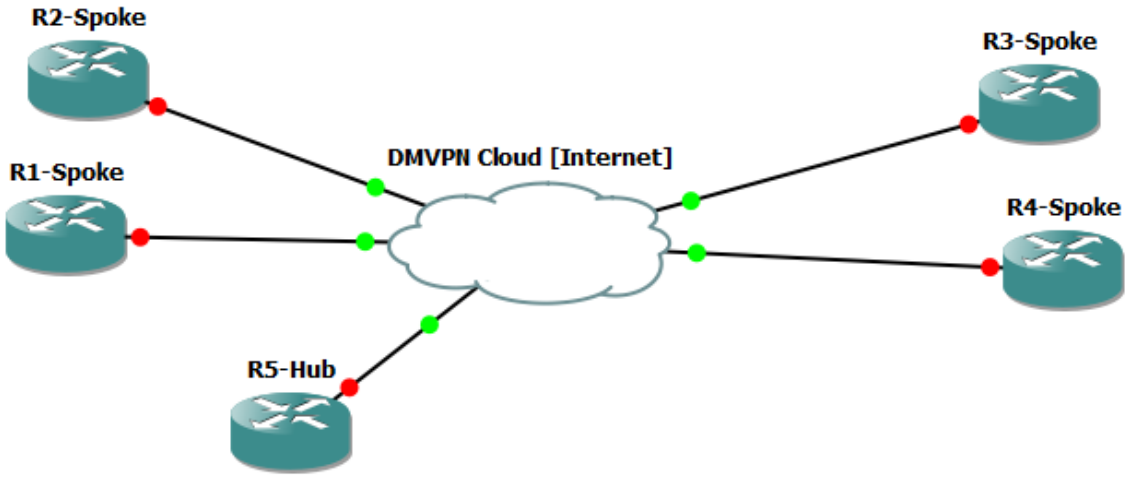
HMAC-SHA-1 uses the first version of SHA, while all the other methods use SHA-2.

With the current IOS code ,IOS does not allow you to enable SHA authentication at the area level, only at the interface level
With the current IOS code, SHA authentication is not supported for virtual-links
OSPF Type 2 authentication is now called a "Cryptographic Authentication" for both MD5 and SHA.
To configure SHA authentication you will need to configure Key Chain same as RIP & EIGRP

```
key chain MYHOME
 key 1
  key-string cisco
  cryptographic-algorithm hmac-sha-256
!
interface GigabitEthernet1.79
 ip ospf authentication key-chain MYHOME
```

OSPF MD5 Authentication with Multiple Keys

In this design, multiple keys are used to authenticate different neighbors on the same interface. Such as DMVPN Hub interface.



R1 - R2:
interface Tunnel0
 ip ospf authentication message-digest
 ip ospf message-digest-key **1** md5 **KEYONE**

R3 - R4:
interface Tunnel0
 ip ospf authentication message-digest
 ip ospf message-digest-key **2** md5 **KEYTWO**

R5:
interface Tunnel0
 ip ospf authentication message-digest
 ip ospf message-digest-key **1** md5 **KEYONE**
 ip ospf message-digest-key **2** md5 **KEYTWO**

OSPFv3 Authentication

OSPFv3 uses **IPsec** for both authentication and encryption, as IPsec is already embedded into the IPv6 specification.
So all what you learned about IPsec will apply here.
IPsec support both Authentication Header (AH) for authentication, and Encapsulating Security Payload (ESP) for authentication and encryption.

But OSPFv3 doesn't use ISAKMP for dynamic key negotiation in Phase 1, the Phase 2 authentication and encryption keys must be manually configured.

This is why the SPI and key strings in the configuration should be written manually while in normal VPN IPsec tunnel configuration these values are normally automatically negotiated by the routers through a process such as the Diffie Hellman (DH) exchange.

The Security Parameter Indexes SPI (as per RFC 2401) is a required part of an IPsec Security Association (SA) because it enables the receiving system to select the SA under which a received packet will be processed….SPI hexa decimal digits will be used should be the same in both routers.

OSPFv3 support SHA1 and MD5 , as we can see below:

```
ipv6 ospf authentication ipsec spi 256 ?
  md5   Use MD5 authentication
  sha1  Use SHA-1 authentication
```

Example using IPsec AH with SPI 1000 and SHA1 for authentication, and a key of 0123456789012345678901234567890123456789.

R1

```
interface GigabitEthernet0/0
 ipv6 ospf 1 area 0
 ipv6 ospf authentication ipsec spi 1000 sha1 0123456789012345678901234567890123456789
```

R3

```
interface FastEthernet0/0.13
 ipv6 ospf 1 area 0
 ipv6 ospf authentication ipsec spi 1000 sha1 0123456789012345678901234567890123456789
```

Example using IPsec ESP with SPI 2000 and md5 for authentication and a key of 0123456789ABCDEF0123456789ABCDEF , AES 256 for encryption, and a key of 0123456789ABCDEF0123456789ABCDEF0123456789ABCDEF0123456789ABCDEF.

R2

```
interface GigabitEthernet0/0
 ipv6 ospf 1 area 1
 ipv6 ospf encryption ipsec spi 2000 esp aes-cbc 256
0123456789ABCDEF0123456789ABCDEF0123456789ABCDEF0123456789ABCDEF md5
0123456789ABCDEF0123456789ABCDEF
```

R3

```
interface FastEthernet0/0.23
 ipv6 ospf 1 area 1
 ipv6 ospf encryption ipsec spi 2000 esp aes-cbc 256
0123456789ABCDEF0123456789ABCDEF0123456789ABCDEF0123456789ABCDEF md5
0123456789ABCDEF0123456789ABCDEF
```

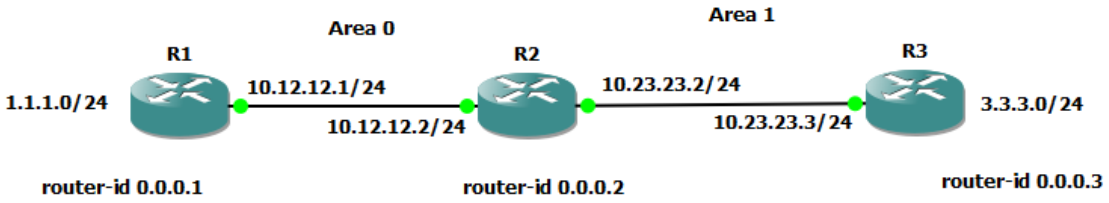
OSPFv3 IPv4/IPv6 Address Families Authentication

Newer version of OSPFv3 nowadays can support IPv4 and IPv6 as well.
So if in exam I asked to configure IPv4 with OSPFv3 or I asked to authenticate IPv4 OSPF routers with SHA and Ipsec , answer will be using OSPFv3 Address Families concept.

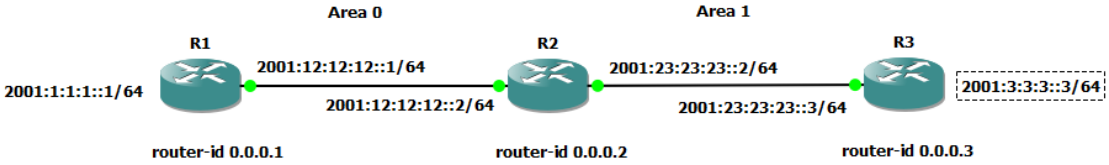
Address Family in OSPFv3 working same as we normally use it in Named mode EIGRP or BGP.

Lets do a lab for this one to fully understand the concept behind it.

Topology from IPv4 Prespective



Topology from IPv6 Prespective



Basic Configuration

R1

```
ipv6 uni
int loop 0
ip add 1.1.1.1 255.255.255.0
ipv6 add 2001:1:1:1::1/64
int f0/0
ipv6 add 2001:12:12:12::1/64
ip add 10.12.12.1 255.255.255.0
```

R2

```
ipv6 uni
int f0/0
ipv6 add 2001:12:12:12::2/64
ip add 10.12.12.2 255.255.255.0
int f1/0
ipv6 add 2001:23:23:23::2/64
ip add 10.23.23.2 255.255.255.0
```

R3

```
ipv6 uni
int loop 0
ipv6 add 2001:3:3:3::3/64
ip add 3.3.3.3 255.255.255.0
int f1/0
ipv6 add 2001:23:23:23::3/64
ip add 10.23.23.3 255.255.255.0
```

OSPFv3 Configuration for both IPv4 and IPv6

R1

```
int f0/0
ipv6 enable   < enabling IPv6 command is mandatory here.
 ospfv3 4 area 0 ipv4
 ospfv3 4 area 0 ipv6

int loop0
ipv6 enable
 ospfv3 4 area 0 ipv4
 ospfv3 4 area 0 ipv6

router ospfv3 100
 router-id 0.0.0.1
 address-family ipv6 unicast
 exit-address-family
 address-family ipv4 unicast
 exit-address-family
```

R2

```
int f0/0
ipv6 enable
 ospfv3 4 area 0 ipv4
 ospfv3 4 area 0 ipv6

int f1/0
ipv6 enable
 ospfv3 4 area 1 ipv4
 ospfv3 4 area 1 ipv6

router ospfv3 100
 router-id 0.0.0.2
 address-family ipv6 unicast
 exit-address-family
 address-family ipv4 unicast
 exit-address-family
```

R3

```
int loop 0
ipv6 enable
 ospfv3 4 area 1 ipv4
 ospfv3 4 area 1 ipv6

int f1/0
ipv6 enable
 ospfv3 4 area 1 ipv4
 ospfv3 4 area 1 ipv6

router ospfv3 100
 router-id 0.0.0.3
 address-family ipv6 unicast
 exit-address-family
 address-family ipv4 unicast
 exit-address-family
```

once we configuration completed , you will see in each router two messages per each neghbor indicating that IPv4 and IPv4 Adj is established

*Aug 1 10:01:35.695: %OSPFv3-5-ADJCHG: Process 4, **IPv4**, **Nbr 0.0.0.3** on FastEthernet0/0 from LOADING to FULL, Loading Done

R2(config)#
*Aug 1 10:01:37.339: %OSPFv3-5-ADJCHG: Process 4, **IPv6**, **Nbr 0.0.0.3** on FastEthernet0/0 from LOADING to FULL, Loading Done

OSPFv3 Authentication for both IPv4 and IPv6 routers

Command used under interface had the following syntax
{ospfv3 authentication ipsec spi {md5 | sha1} key-encryption-type key | null}

If we check avaliable options we had in each part of this command , we find it like the following:

```
R1(config-if)#ospfv3 authentication ?
  ipsec  Use IPsec authentication
  null   Use no authentication

R1(config-if)#ospfv3 authentication ipsec ?
  spi  Set the SPI (Security Parameters Index)

R1(config-if)#ospfv3 authentication ipsec spi ?
  <256-4294967295>  SPI

R1(config-if)#ospfv3 authentication ipsec spi 256 ?
  md5   Use MD5 authentication
  sha1  Use SHA-1 authentication
```

```
R1(config-if)#ospfv3 authentication ipsec spi 256 md5 ?
  0         The key is not encrypted (plain text)
  7         The key is encrypted
  Hex-string  MD5 key (32 chars)

R1(config-if)#ospfv3 authentication ipsec spi 256 md5 0 ?
  Hex-string  MD5 key (32 chars)
```

R1-R2

```
Int f0/0
ospfv3 authentication ipsec spi 256 md5 0 27576134094768132473302031209727
```

R2-R3

```
int f1/0
ospfv3 authentication ipsec spi 512 md5 0 1234567890abcdef1234567890abcdef
```

Verification

R1

```
R1#sh ip route
     1.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C       1.1.1.0/24 is directly connected, Loopback0
L       1.1.1.1/32 is directly connected, Loopback0
     3.0.0.0/8 is variably subnetted, 3 subnets, 2 masks
C       3.3.3.0/24 is directly connected, Loopback1
L       3.3.3.1/32 is directly connected, Loopback1
O IA    3.3.3.3/32 [110/2] via 10.12.12.2, 00:09:36, FastEthernet0/0
     10.0.0.0/8 is variably subnetted, 3 subnets, 2 masks
C       10.12.12.0/24 is directly connected, FastEthernet0/0
L       10.12.12.1/32 is directly connected, FastEthernet0/0
O IA    10.23.23.0/24 [110/2] via 10.12.12.2, 00:09:36, FastEthernet0/0

R1#sh ipv6 route
C   2001:1:1:1::/64 [0/0]
    via Loopback0, directly connected
L   2001:1:1:1::1/128 [0/0]
    via Loopback0, receive
OI  2001:3:3:3::3/128 [110/2]
    via FE80::C802:6FF:FEA8:0, FastEthernet0/0
C   2001:12:12:12::/64 [0/0]
    via FastEthernet0/0, directly connected
L   2001:12:12:12::1/128 [0/0]
    via FastEthernet0/0, receive
OI  2001:23:23:23::/64 [110/2]
    via FE80::C802:6FF:FEA8:0, FastEthernet0/0
L   FF00::/8 [0/0]
    via Null0, receive
```

BGP Authentication

- BGP uses TCP for transport.
- TCP already has a specification for authentication which is **TCP Option 19**, which is the **MD5** Signature Option.
- BGP uses TCP authentication instead of a separate internal mechanism.
- To verify that BGP authentication is working use show ip bgp summary command and ensure that the peers are up.

R1:
```
router bgp 1
 network 150.1.1.1 mask 255.255.255.255
 neighbor 136.1.13.3 remote-as 3
 neighbor 136.1.13.3 password CISCO
```

R3:
```
router bgp 3
 network 150.1.3.3 mask 255.255.255.255
 neighbor 136.1.13.1 remote-as 1
 neighbor 136.1.13.1 password CISCO
```

IGP/EGP Troubleshooting Tricks

- Miss match Key string
- Miss match key id (if one or both routers had more than one key id) with RIPv2 and EIGRP
- Miss match key id with OSPF
- Miss match Key chain name
- Miss match time and data in accept-lifetime and send-lifetime commands or NTP sync missing
- Miss match authentication mode or type (plain-text , md5 , hmac-sha-256)
- Miss Match OSPF authentication area id with OSPF

Other troubleshooting commands can lead to unsuccessful authentication:

- If interface is configured as backup interface or passive interface
- If duplicate routing protocol router id exists

Good Luck
CCIE & CCSI: Yasser Auda
https://www.facebook.com/YasserRamzyAuda
https://learningnetwork.cisco.com/people/yasserramzy
https://www.youtube.com/user/yasserramzyauda