

# Use Regular Expressions in BGP

## Contents

---

### [Introduction](#)

### [Prerequisites](#)

[Requirements](#)

[Components Used](#)

[Conventions](#)

### [Background Information](#)

### [Network Scenarios](#)

[Only Allow Networks that Originate from AS 4 to Enter Router 1](#)

[Only Allow Networks That Have Passed Through AS 4 to Enter AS 3](#)

[Deny Networks Originated in AS 4 to Enter AS 3 and Permit All Other Networks](#)

[Only Allow Networks Originated from AS 4, and ASs Directly Attached to AS 4, to Enter Router 1](#)

### [Related Information](#)

---

## Introduction

This document describes how to use regular expressions with Border Gateway Protocol (BGP).

## Prerequisites

### Requirements

Cisco recommends that you have knowledge of this topic:

- Basic BGP configuration

### Components Used

The information in this document is based on Cisco IOS® Software Release 12.0.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

### Conventions

For more information on document conventions, see the [Cisco Technical Tips Conventions](#).

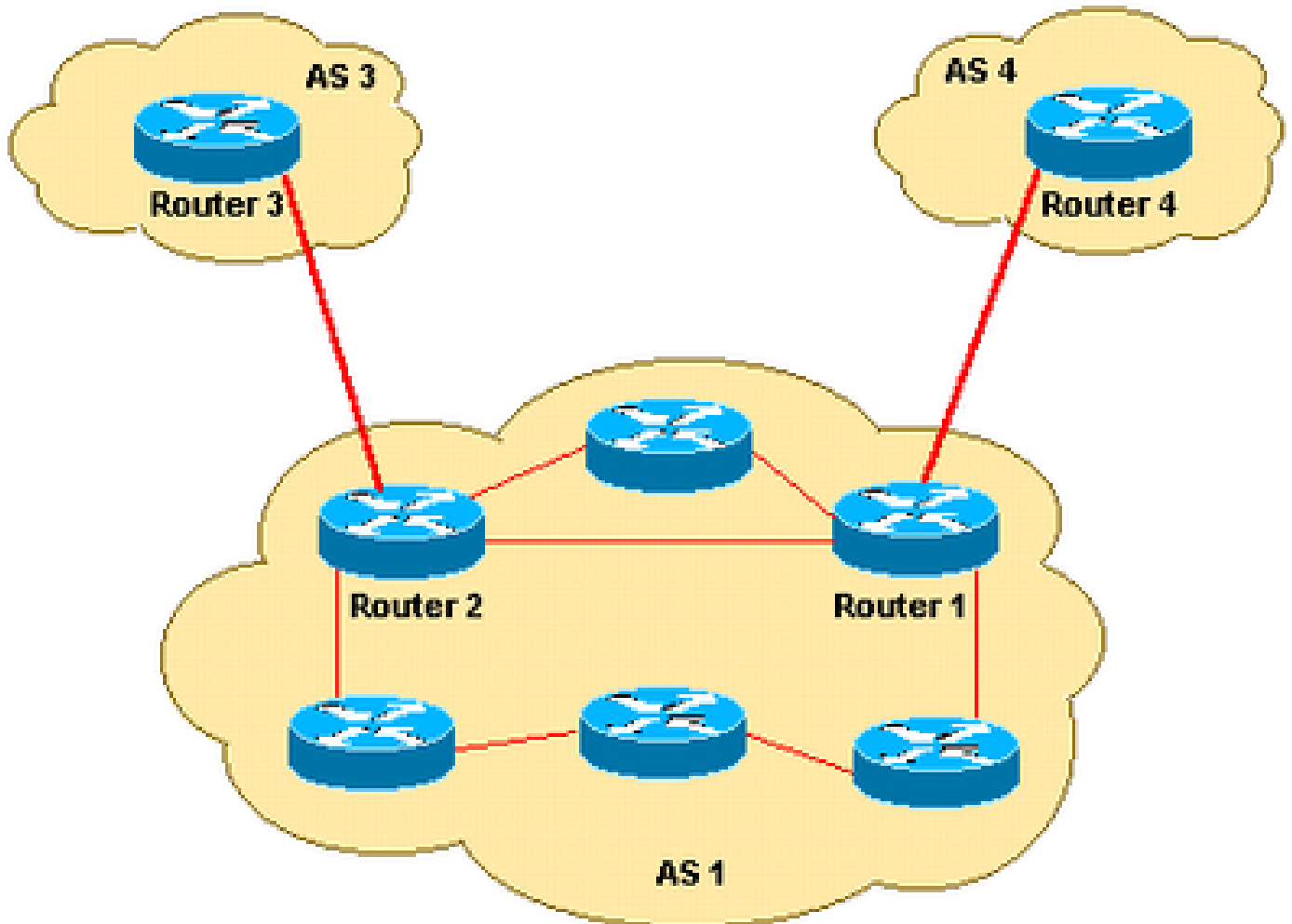
## Background Information

You can use regular expressions in the [ip as-path access-list](#) command with Border Gateway Protocol (BGP). For more general information about regular expressions, see the Cisco Documentation on [Regular Expressions](#). For more information on basic BGP configuration, see the [BGP Case Studies](#) and [Configure a Basic BGP](#)

[Network.](#)

## Network Scenarios

This is the network diagram referred to in this document.



### Only Allow Networks that Originate from AS 4 to Enter Router 1

If you would like for Router 1 to receive only the routes originated from AS 4 (and no Internet routes), you can apply an inbound access list on Router 1:

```
ip as-path access-list 1 permit ^4$

router bgp 1
 neighbor 10.4.4.4 remote-as 4
 neighbor 10.4.4.4 route-map foo in

route-map foo permit 10
 match as-path 1
```

This ensures only networks originated from AS 4 are allowed into Router 1.

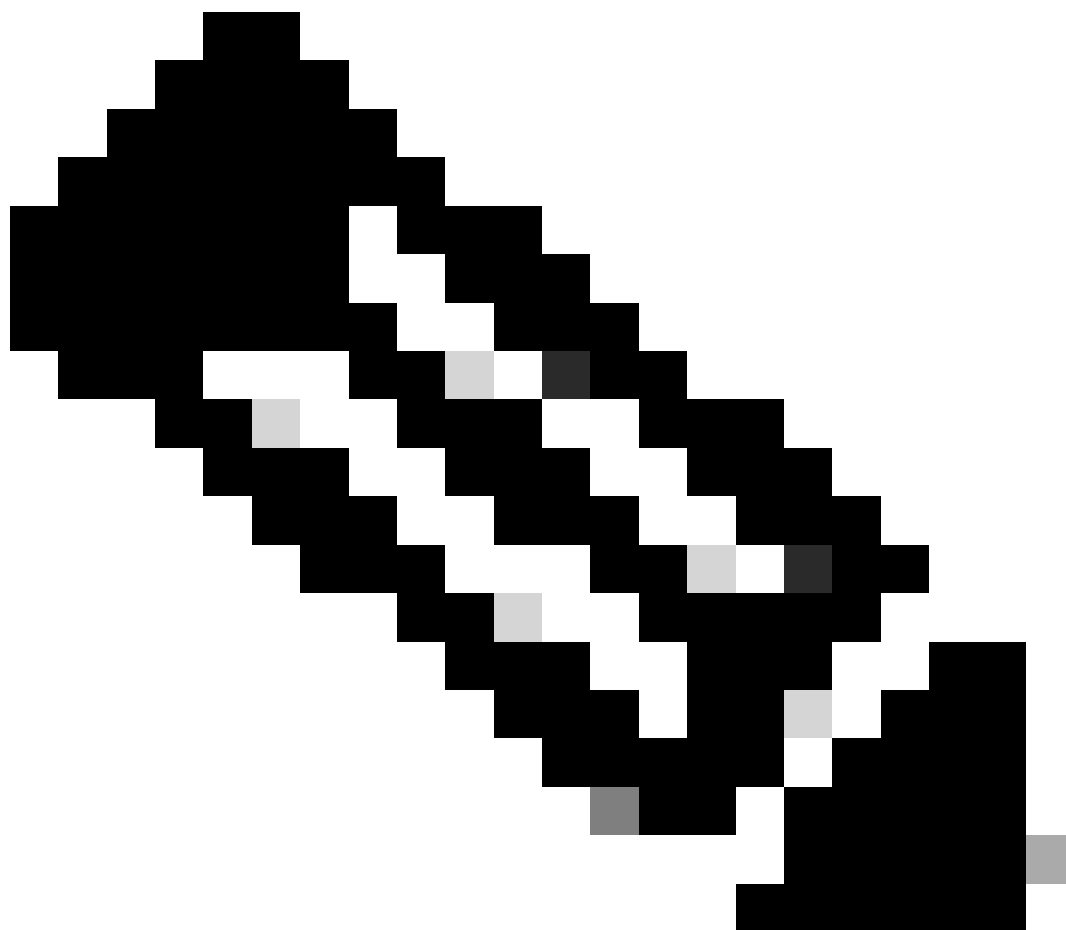
## Only Allow Networks That Have Passed Through AS 4 to Enter AS 3

If you want only the networks that have passed through AS 4 to enter AS 3 from Router 3, you can apply an inbound filter on Router 3:

```
ip as-path access-list 1 permit _4_  
  
router bgp 3  
neighbor 10.2.2.2 remote-as 1  
neighbor 10.2.2.2 route-map foo in  
  
route-map foo permit 10  
match as-path 1
```

You can use an underscore ( `_` ) as the input string and output string in the [ip as-path access-list](#) command.

---



**Note:** In this example, anchoring (for instance, there is no `^`) is not used, so it does not matter what autonomous systems come before and after AS 4.

---

## Deny Networks Originated in AS 4 to Enter AS 3 and Permit All Other Networks

If you want to deny all the networks that have originated in AS 4, and permit all other routes to enter AS 3 from Router 3, you can apply an inbound filter at Router 3:

```
ip as-path access-list 1 deny _4$
ip as-path access-list 1 permit .*

router bgp 3
 neighbor 10.2.2.2 remote-as 1
 neighbor 10.2.2.2 route-map foo in

route-map foo permit 10
 match as-path 1
```

## Only Allow Networks Originated from AS 4, and ASs Directly Attached to AS 4, to Enter Router 1

If you want AS 1 to get networks originated from AS 4, and all directly attached ASs of AS 4, apply the next inbound filter on Router 1.

```
ip as-path access-list 1 permit ^4_[0-9]*$

router bgp 1
 neighbor 10.4.4.4 remote-as 4
 neighbor 10.4.4.4 route-map foo in

route-map foo permit 10
 match as-path 1
```

In the [ip as-path access-list](#) command, the caret (^) starts the input string and designates AS" The underscore (\_) means there is a a null string in the string that comes after AS 4" The [0-9]\* specifies that any connected AS with a valid AS number can pass the filter. The advantage with the [0-9]\* syntax is that it gives you the flexibility to add any number of ASs without a modification to this command string. For additional information, see [AS-Regular Expression](#).

## Related Information

- [IP Routing Support Page](#)
- [Cisco Technical Support & Downloads](#)