

6.1

ENTENDENDO O

”BGP REGEXP”

(expressões regulares)

6.1 - Entendendo o regexp

- Quando temos muitos clientes de trânsito é preciso estabelecer padrões para que os administradores sejam capazes de entender as configurações
- Além de evitar ao máximo a quantidade excessiva de linhas de configuração
- Por exemplo:

Há um "range" de números de AS que são de uso privado semelhantes aos IPs 192.168.x.x . Esse intervalo vai de 64512 até 65535, e não devemos receber anúncios que contenham algum desses AS's no path

6.1 - Entendendo o regexp

- Imagine ter que configurar uma linha de deny para cada AS privado

```
AS-X-R1(config)#ip as-path access-list 1 deny 64512
AS-X-R1(config)#ip as-path access-list 1 deny 64513
AS-X-R1(config)#ip as-path access-list 1 deny 64514
...
...
AS-X-R1(config)#ip as-path access-list 1 deny 65534
AS-X-R1(config)#ip as-path access-list 1 deny 65535
AS-X-R1(config)#ip as-path access-list 1 permit .*
```

- Seria um tanto complicado
- Felizmente com expressões regulares ao invés de milhares, resolvo esse problema com apenas 2, (ou uma talvez)

6.1 - Entendendo o regexp

- Antes vamos entender a função dos principais caracteres nas expressões regulares:

"_" O caracter "underline" define um espaço em branco

- x **"1"** Um número isolado satisfaz a condição para qualquer expressão que contenha esse número

- x **exemplo1: 1** "a condição de match é qualquer AS no Path (podem haver vários AS's) que contenha o número 1 em qualquer parte, em pelo menos um dos AS's

Satisfaz: "999999999**1**999999999", "**1**23 432", "999 888 777 **1**23 998"

Não satisfaz: "355 3344 22334", "466", etc

- x **exemplo2: _1** "a condição de match é qualquer AS no Path (podem haver vários AS's) que contenha o número 1 no início de pelo menos um dos AS's e que essa ocorrência não seja a primeira"

Satisfaz: "**1**2333 3445 9999", "**1**23 432", "3333 **1**111 2222 4444"

Não satisfaz: "4333 21123", "4661" "21111 12222", etc

- x **exemplo3: _1_** "a condição de match é qualquer AS no Path (podem haver vários AS's) que contenha o número 1 sozinho em pelo menos um dos AS's e que essa ocorrência não seja nem a primeira nem a última

- x **Satisfaz:** "2333 **1** 3445 9999", "33 **1** 65554", "433 4444 **1** 433"

Não satisfaz: "333 222 555 1" autor: Rinaldo Vaz | rinaldopvaz@gmail.com

6.1 - Entendendo o regexp

- ✗ **”.”** Satisfaz a condição para qualquer número no lugar do ponto
- ✗ **exemplo: 4.5** "a condição de match é qualquer AS PATH que contenha um AS de 3 dígitos onde 4 seja o primeiro e 5 seja o último com qualquer número no meio e que contenha outros AS's antes e depois

Satisfaz: "12 **415** 43", "2 **425** 3", "6 **435** 2", "321 456 **445** 345"

Não satisfaz: "415", "123 415", etc

- ✗ **Exemplo: 4.5** "a condição de match é qualquer número de 3 ou mais dígitos que fique "entre" 4 e 5 em qualquer parte do AS PATH

Satisfaz: "23**415**", "43**425**", "**435** 3333", "**445**", "**455**87", "321 4**445**9 345"

Não satisfaz: 315, 416, etc

6.1 - Entendendo o regexp

- ✗ "\$" Satisfaz a condição para o tudo que o que for terminado com o caracter (ou expressão) ao lado esquerdo de "\$"

- ✗ **exemplo1:** **12\$** "a condição de match é qualquer AS PATH em que os dois últimos dígitos sejam 1 e 2 respectivamente"

Satisfaz: "49**12**", "650**12**", "123**12**", "111**12**", "9932 9976 649**12**", "28135 17379 543**12**" ...

Não satisfaz: 13, 22, 129, 55413, 58752 ...

- ✗ **exemplo2:** **_12\$** "a condição de math é qualquer AS PATH em que último AS seja 12"

Satisfaz: "**12**", "2335 **12**", "312 **12**", "111 **12**"

Não satisfaz: "444 112", "22", "12 229", "999 12 1111"

- ✗ **exemplo3:** **1.2\$** "a condição de match é qualquer AS PATH em que seus últimos 3 dígitos do último AS sejam: 1, qualquer coisa e 2 respectivamente"

Satisfaz: "123 45666 33 **112**", "444 122", "**132**", "1111 **132**", "444 65**192**", "60**192**"

Não satisfaz: "2333 222", 123, 912

6.1 - Entendendo o regexp

- ✗ **"^"** Satisfaz a condição para caracteres que comecem com o caracter (ou expressão) após o ^
- ✗ **exemplo1: ^1** "a condição de match é qualquer AS PATH que tenha o dígito 1 no início do primeiro AS do PATH

satisfaz: "1111 9999 55443", "12345 6776", 12222, 19999

não satisfaz: 21111, 31211, 2, 3, 43111

- ✗ **exemplo2: ^1.4** "a condição de match é qualquer AS PATH em que os 3 primeiros dígitos do primeiro as sejam: 1, qualquer coisa e 4 respectivamente

Satisfaz: "124 4321", "12444", "1449", "19455 12333", "10466 55443 9900"

Não satisfaz: 195, 2124, 22444, 9104

- ✗ **exemplo3: ^1..8\$** "a condição de match é um AS PATH contento apenas um AS de 4 dígitos, onde o primeiro dígito é 1, o segundo e o terceiro qualquer coisa e o quarto e último seja 8 respectivamente

Satisfaz: "1008", "1248", "1118", "1228", "1998", "1888"

Não satisfaz: 10008, "1008 1008", "14448"

6.1 - Entendendo o regexp

- "*" Satisfaz a condição para qualquer caractere à esquerda do "*", inclusive nenhum, por isso é preciso utilizar o caracter um espaço "_" branco em conjunto, pois sem espaço a expressão perde o sentido, por exemplo, 1* satisfaria condição para qualquer coisa, e o padrão intuitivo para "qualquer coisa/tudo" é ".*"
- x **exemplo1:** 1_* "a condição de match é qualquer coisa que contenha o dígito 1 no final de pelo menos um AS"

satisfaz: "4343**1** 9999 777 9999 8888(infinito...)", " ... 9999 999**1** 345 666 654 5556..."

não satisfaz: 8872, "9987 2332", etc

- x **exemplo2:** .* "a condição de math é qualquer AS PATH de qualquer tamanho, com qualquer combinação"

Satisfaz: TUDO

Não satisfaz: NADA

6.1 - Entendendo o regexp

- “\” Este caracter é usado quando os caracteres de condição também são caracteres de expressão regular, por exemplo, se precisássemos de uma expressão para dinheiro, onde a condição fosse “R\$ 10,01” até “R\$ 10,99”, teríamos um grande problema, já que a expressão necessária é “R\$_10,..\$” Notem que o “\$” precisa aparecer como “condição” ao mesmo tempo que é um caracter de expressão regular. Sendo assim, não funcionaria. O caracter “\” é usado para declarar que o próximo caracter deverá ser interpretado como “condição”, ao invés de sua função normal. Assim, a expressão corretá ficaria “R\\$_10,..\$”
- “Felizmente” não utilizaremos esse caracter para BGP

6.1 - Entendendo o regexp

- ✗ "[]" Satisfaz a condição para qualquer um dos caracteres dentro dos colchetes, mas apenas UM deles. Há a possibilidade de utilizar intervalos entre os colchetes seguidos de hífen "-"
- ✗ Dentro dos colchetes o caractere "^" tem função de negação, O intervalo funciona da mesma forma para as negações.
- ✗ **exemplo1:** `^9[249]5$` "a condição é um AS que contenha qualquer combinação de 3 dígitos sendo o primeiro 9, o terceiro 5 e o do meio 2, 4 ou 9"

Satisfaz: "925", "945", e "995"

Não satisfaz: 915, 955, 965, etc

- ✗ **exemplo2:** `^9[5-7]5$` "mesma lógica do exemplo 1, porém o dígito do meio pode ser de 5 à 7"

satisfaz: 955, 965 e 975

- ✗ **exemplo3:** `^9[^28]5$` "a condição de match é a mesma lógica dos exemplos 1 e 2 porém o dígito do meio não poderá ser 2 ou 8"

satisfaz: todas as possibilidades de um "9.5" "menos" 925 e 985.

- ✗ **exemplo4:** `^9[^5-8]5$` "a condição de match segue a mesma lógica dos exemplos 2 e 3 porém o dígito do meio não pode estar no intervalo entre 5 e 8 (5,6,7 ou 8)"

satisfaz todas as possibilidades "menos" 955, 965, 975 e 985

autor: Rinaldo Vaz | rinaldopvaz@gmail.com

6.1 - Entendendo o regexp

- [illegible]

6.1 - Entendendo o regexp

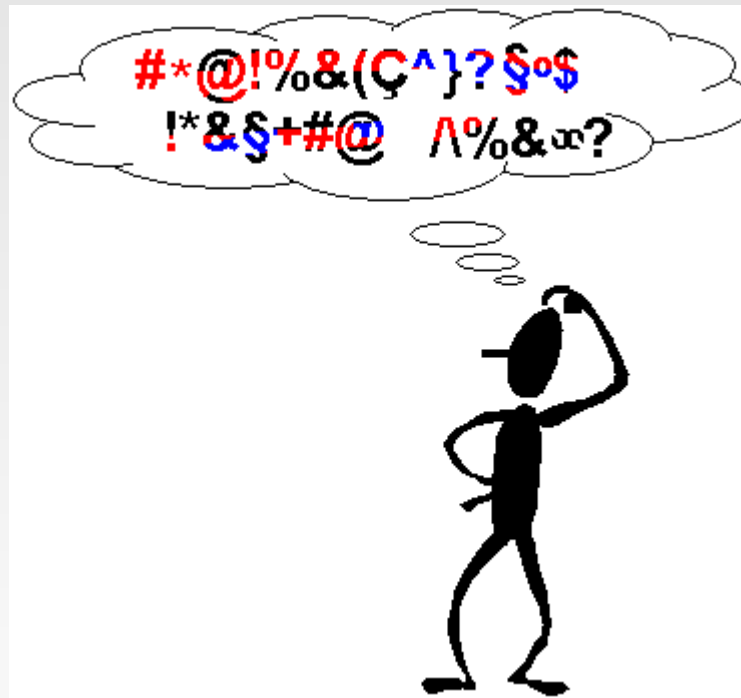
- ✗ "+" Satisfaz a condição para pelo menos uma ou mais ocorrências do caracter (ou expressão) anterior
- ✗ **exemplo1: 2+** satisfaz: 2, 22, 22222, 222222, 2222222222
- ✗ "?" Satisfaz a condição para uma ou nenhuma ou uma ocorrência do caracter (ou expressão) anterior
- ✗ **exemplo1: 2?** satisfaz: 2 ou ""
- ✗ "{}" especifica um número ou um intervalo de repetições para um caracter (ou expressão) antecedente
- ✗ **exemplo1: ^2{4}\$** satisfaz: 2222 apenas
- ✗ **exemplo2: ^2{4,6}\$** satisfaz: 2222, 22222 e 222222
- ✗ **exemplo3: ^1[0-3]{1,2}\$** satisfaz: 100, 13, 10, 132, 12, 130
nao satisfaz: 14, 1031, 15, 1033
- Para praticar segue um link interessante:
- http://tools.lymas.com.br/regexp_br.php

6.2

Filtros facilmente escaláveis

6.2 - Filtros facilmente escaláveis

- Agora vamos raciocinar um pouco e utilizar as expressões que aprendemos para bloquear o range dos AS's 64512 até 65535 utilizando o menor número possível de regras...



6.2 - Filtros facilmente escaláveis



```
AS-X-R1(config)#ip as-path access-list 1 permit 645(0.|10|11)
```

!--- aceita de 64501 a 64511

```
AS-X-R1(config)#ip as-path access-list 1 deny 6(4[5-9]..|5...)
```

!--- rejeita qualquer coisa de 64500 à 64999 e de 65001 à 65999

!--- obs, não há AS maior que 65535

```
AS-X-R1(config)#ip as-path access-list 1 permit .*
```

!--- aceita todo o resto

LAB:

O AS 99 vai gerar rotas com AS's privados, aplicar filtros de input para bloquear essas rotas:

```
AS-X-R1(config)#router bgp X
```

!--- Vamos adicionar o mesmo prefix-list na entrada do AS 100

```
AS-X-R1(config-router)#neighbor 100.X.1.1 filter-list 1 in
```

6.2 - Filtros facilmente escaláveis

- Existem infinitas aplicações, por exemplo, descubro que o AS 12345 está com lentidão, crio políticas que diminuem a prioridade de qualquer prefixo anunciado por esse AS

```
router(config)#ip as-path access-list 20 permit _12345_  
!-- cria a ACL 20 e diz que qualquer AS PATH contendo 12345 está nela
```

```
router(config)#route-map UPSTREAM-1 permit 1  
!-- Cria um route-map no topo da hierarquia  
router(config-route-map)#set local-preference 90  
!-- especifica a ação "mudar o local pref para 90"  
router(config-route-map)#match as-path 20  
!-- especifica quem irá receber a ação "set local pref 90"  
router(config-route-map)#exit  
AS-x-R1(config)#route-map UPSTREAM-1 permit 2  
!-- cria um route-map "em branco" que aceita todo o resto sem modificações  
router(config-route-map)#exit
```

6.2 - Filtros facilmente escaláveis

- Precisamos nesse caso fazer o mesmo para todos os UPSTREAMS

```
router(config)#route-map UPSTREAM-2 permit 1
!--- Cria um route-map no topo da hierarquia
router(config-route-map)#set local-preference 90
!--- especifica a ação "mudar o local pref para 90"
router(config-route-map)#match as-path 20
!--- especifica quem irá receber a ação "set local pref 90"
router(config-route-map)#exit
AS-x-R1(config)#route-map UPSTREAM-2 permit 2
!--- cria um route-map "em branco" que aceita todo o resto sem modificações
router(config-route-map)#exit
```

6.2 - Filtros facilmente escaláveis

- As expressões regulares não servem apenas para filtrar prefixos, mas também para encontrar certas rotas entre as centenas de milhares de uma tabela full
- Por exemplo:
- Quero saber quais rotas na minha RIB foram originadas pelo AS 6762 (TELECOM ITALIA SPARKLE S.p.A.)

```
BGPD-ALOG-SP# sh ip bgp regexp _6762$
```

```
...
```

Network	Next Hop	Metric	LocPrf	Weight	Path
*> 2.16.70.0/23	201.20.36.158			0	16397 6762 i
*> 2.16.140.0/23	201.20.36.158			0	16397 6762 i
*> 2.16.146.0/23	201.20.36.158			0	16397 6762 i

```
...
```

6.2 - Filtros facilmente escaláveis

- Exemplo 2:

Quais rotas foram recebidas pela Intelig através do peering com Embratel?

```
BGPD-ALOG-SP# sh ip bgp regexp _17379 4230_
```

```
...
```

Network	Next Hop	Metric	LocPrf	Weight	Path
*> 186.205.244.0/22	201.20.36.158		0	16397	17379 4230 28573 i

- Exemplo 3:

Quais ASs estão diretamente conectados com o AS 4230 (EMBRATEL)?

```
BGPD-ALOG-SP# sh ip bgp regexp _4230 ([0-9]+){1}$
```

```
...
```

Network	Next Hop	Metric	LocPrf	Weight	Path
*> 23.61.118.0/23	201.20.36.158		0	16397	17379 4230 28573 i
*> 23.61.120.0/22	201.20.36.158		0	16397	17379 4230 28573 i

```
...
```

6.2 - Filtros facilmente escaláveis

- Quais ASs estão diretamente conectados com meus upstreams/downstreams?

```
BGPD-ALOG-SP# sh ip bgp regexp ^([0-9]+_){2}$
```

Network	Next Hop	Metric	LocPrf	Weight	Path
...					
*> 216.72.166.0	201.20.36.157			0	16397 6505 i
*> 216.239.32.0/19	201.20.36.158			0	16397 15169 i
...					

6.3

BGP COMMUNITIES



The screenshot shows the Orkut website interface. At the top is the Orkut logo and navigation buttons: 'início', 'perfil', 'scraps', and 'comunidades'. The main content area displays the community 'AH B.G.P OUXXI CONHESO!!!!!!!!' with a cover photo of four women. To the right of the title is a button '+ Participar da comunidade'. Below the title, community details are listed: 'idioma: Português (Brasil)', 'categoria: Outros', 'tipo: moderada', 'visível por: público', 'criada em: 07/11/2009', 'local: PODEROLANDIA, pernambuco, Brasil', 'proprietário:', and 'moderadores:'. A link 'ocultar perfil' is also present. A description in purple text reads: 'B.G.P noix damos valor as pequenas coisas q nós ofereseem... FELINA,SAPEKA,LU E ANGINHA.'. On the left sidebar, there is a section for the community with the same title and '0 membros'. Below this are links for 'comunidade', 'fórum', 'enquetes', 'membros', 'Ações', 'Participar da comunidade', and 'Denunciar abuso'.

orkut início perfil scraps comunidades

AH B.G.P OUXXI CONHESO!!!!!!!! + Participar da comunidade

idioma: Português (Brasil) **criada em:** 07/11/2009
categoria: Outros **local:** PODEROLANDIA, pernambuco, Brasil
tipo: moderada **proprietário:**
visível por: público **moderadores:**

[ocultar perfil](#)

B.G.P noix damos valor as pequenas coisas q nós ofereseem... FELINA,SAPEKA,LU E ANGINHA.

AH B.G.P OUXXI CONHESO!!!!!!!!
0 membros

comunidade

fórum

enquetes

membros

Ações

[Participar da comunidade](#)

[Denunciar abuso](#)

6.3 - BGP Communities

- Communities são atributos que são preservados em anúncios entre AS's externos e internos
- São um tipo de carimbo inserido nos updates em forma de string
- Uma community é um atributo opcional, e sendo assim nada obriga que os fabricantes implementem no BGP de suas caixas, ou mesmo que um AS replique para seus peers o mesmo valor de community recebido por um de seus clientes

6.3 - BGP Communities

- Os modelos mais novos suportam o "novo formato" usualmente utilizado como ASN:community, por exemplo: 28135:110
- Porém há algumas communities padronizadas pela RFC 1997 e são padrão nos routers Cisco:
 - INTERNET
 - NO-EXPORT
 - NO-ADVERTISE
 - LOCAL-AS
 - NO-ADVERTISE-SUBCONFED

6.3 - BGP Communities

- **NO-EXPORT** por padrão faz o router receptor exportar o prefixo para neighbors iBGP, mas não para eBGP. É útil quando faço anúncios mais específicos para a operadora A e menos específicos para operadora B. Normalmente, a operadora B precisaria passar pela operadora A para chegar ao prefixo "específico" mesmo estabelecendo BGP diretamente com meu AS.

Enviando o mesmo bloco específico, com a community NO-EXPORT à operadora A, essa passaria a me alcançar diretamente, ao mesmo tempo que não exportaria esses prefixos.

6.3 - BGP Communities

- **NO-ADVERTISE** por padrão faz o router receptor não exportar esse prefixo para nenhum neighbor, seja ele eBGP ou iBGP
- **NO-ADVERTISE-SUBCONFED** usada para evitar que o prefixo seja exportado para AS's confederados
- **LOCAL-AS** Usada para evitar que os prefixos marcados sejam exportados para fora do AS ou confederação, geralmente usado em filtros de entrada
- **INTERNET** Na prática é a community padrão para anuncios externos, é o mesmo que não haver nenhuma

6.3 - BGP Communities

- LAB 1: Anunciar os prefixos de um dos AS's clientes com a community NO-EXPORT para dentro do AS do grupo:

```
AS-11X-R1#conf t
AS-11X-R1(config)#router bgp 11X
AS-11X-R1(config-router)#neighbor X.11X.4.1 send-community both
```

```
As-11X-R1(config)#ip bgp-community new-format
```

```
AS-11X-R1(config)#route-map AS-X-OUT permit 10
AS-11X-R1(config-route-map)#set community no-export
```

- Verifiquem nos routers R1 e R2 se os prefixos marcados como "NO-EXPORT" estão sendo anunciados para os AS's 100 e 200

6.3 - BGP Communities

- Antes de iniciar os LABs desse capítulo é **obrigatório habilitar suporte ao envio de communities** em TODOS os neighbors
- Em seguida habilitar o formato novo formato xxx:yyy para o tratamento de communities:

```
router#conf t
router(config)#router bgp X
router(config-router)#neighbor A.B.C.D send-community both
router(config-router)#exit
router(config)#ip bgp-community new-format
```

6.3 - BGP Communities

- LAB 2 opcional: verificar a diferença enviando a community NO-ADVERTISE
- LAB 3 : Utilizar communities PADRÃO e fazer com que o AS 99 chegue aos DOWNSTREAMS do AS do grupo através do AS 200 ao invés de 100, mas ao mesmo tempo que o AS 100 consiga alcança-los diretamente sem ter que "dar a volta" pelo AS 200

6.3 - BGP Communities

- Solução:



```
AS-X-R1#conf t
AS-X-R1(config)#ip as-path access-list 44 permit _1[1|2]X$
!--- lembrem que X é o AS do grupo

AS-X-R1(config)#route-map AS-100-OUT permit 5
AS-X-R1(config-route-map)#match as-path 44
AS-X-R1(config-route-map)#set community NO-EXPORT
AS-X-R1(config-route-map)#end
AS-X-R1#clear ip bgp * soft
```

- Agora verifiquem o resultado em AS-100-R1 e AS-99-R1

6.4

MANIPULANDO TRÂNSITO (NACIONAL / INTERNACIONAL) COM COMMUNITIES

6.4 - Manipulando trânsito (nacional/internacional) com communities

- É muito pequena a quantidade de communities padrão, podemos fazer pouquíssimas coisas com elas
- Porém há uma grande quantidade de communities numéricas que podem ser utilizadas.
- É importante saber que essas communities numéricas por padrão não influenciam em nada no tratamento das rotas
- É preciso uma configuração prévia para cada community

6.4 - Manipulando trânsito (nacional/internacional) com communities

- Também não há nenhum padrão para essas ações (pelo menos nenhum padrão RFC)
- Entretanto muitos AS's implementaram padrões extremamente criativos e funcionais
- O conhecimento desses padrões é uma ferramenta chave para resolver rapidamente incidentes do dia a dia
- Vamos ver nos próximos slides uma série de communities numéricas adotadas por algumas operadoras

6.4 - Manipulando trânsito (nacional/internacional) com communities

- Brasil Telecom

8167:90 Set Local Preference 90
8167:100 Set Local Preference 100
8167:110 Set Local Preference 110
* (default é Local Preference 200)

8167:3xy controle de anúncios

x: ação { 0 = não anuncia, 1,2,3 = insere x prepends)
y: peerings { 0=demais peerings, 1=internacionais, 2=Embratel, 3=Intelig, 4=Telemar, 5=Telefônica }
(exemplo: 312 = muda para 1 prepend o anúncio para a Embratel)

* (default é anunciar sem prepends para todos os peerings)

8167:555 => não exporta fora da Brasil Telecom.
8167:557 => não anuncia para sites internacionais.

8167:666 => serão injetados no BlackHole da Brasil Telecom.
8167:777 => somente anuncia para clientes da Brasil Telecom.

6.4 - Manipulando trânsito (nacional/internacional) com communities

- Embratel

4230:120 - Local Preference - marcar a rota como localpreference 120

4230:10000 - Blackhole - bloqueia todo o tráfego para a rede/endereco

4230:10002 - Blackhole - filtra o tráfego internacional nos provedores que proveem alguma especie de blackhole.

4230:10004 - Blackhole - filtra nos roteadores da Embratel nos EUA o trafego destinado a rede/endereco anunciado

6.4 - Manipulando trânsito (nacional/internacional) com communities

- Global Crossing

3549:100 - set local preference 100

3549:200 - set local preference 200

3549:275 - set local preference 275

3549:300 - set local preference 300

3549:350 - set local preference 350

3549:600 - Deny inter-continental export of tagged prefix [iBGP].

3549:666 - Deny inter-as export of tagged prefix (carry on AS 3549 only)
[eBGP]

6.4 - Manipulando trânsito (nacional/internacional) com communities

- Uma série de communities complexas possibilitam o tratamento do prepend para cada um dos peers da GLBX 3549:8...

ASN	Peer	No Export	Prepend+1	Prepend+2	Prepend+3
209	Qwest	8010	8011	8012	8013
701	MCI	8030	8031	8032	8033
1239	Sprint	8060	8061	8062	8063
1299	TeliaSonera	8250	8251	8252	8253
1668	AOL	8070	8071	8072	8073
2497	JPNIC	8080	8081	8082	8083
2516	KDDI	8100	8101	8102	8103
2914	NTT Verio	8120	8121	8122	8123
3257	Tiscali	8240	8241	8242	8243
3300	InfoNet Europe	8130	8131	8132	8133
3303	Swisscom	8140	8141	8142	8143
3320	T-Systems/DTAG	8150	8151	8152	8153
3356	Level 3	8160	8161	8162	8163
3561	Savvis	8170	8171	8172	8173
4134	ChinaNet	8230	8231	8232	8233
5511	OpenTransit	8190	8191	8192	8193
6461	AboveNet	8200	8201	8202	8203
6453	Teleglobe	8210	8211	8212	8213
7018	AT&T (US)	8220	8221	8222	8223
7738	Telemar	8290	8291	8292	8293

autor: Rinaldo Vaz | rinaldopvaz@gmail.com

6.4 - Manipulando trânsito (nacional/internacional) com communities

- Um cliente com o AS 4111 envia um prefixo com a community "3549:8011 3549:8033 3549:8190"
- O prefixo com essa marca será anunciado da seguinte forma para os peers GLBX:

Qwest (AS 209) receberá o prefixo com o AS PATH: "3549 3549 4111"

MCI (AS 701) receberá o prefixo com o AS PATH: "3549 3549 3549 3549 4111"

OpenTransit (AS 5511) não receberá o prefixo

todos os outros receberão o prefixo normalmente: "3549 4111"

6.4 - Manipulando trânsito (nacional/internacional) com communities

- Intelig

17379:23004 - Community NO-EXPORT-INTERNACIONAL - Não anuncia para peers Internacionais.

17379:23005 - Community NO-EXPORT-NACIONAL - Não anuncia para peers Nacionais.

No-export ? A rota não será anunciada para nenhum peer eBGP da Intelig.

17379:60 = set local-preference 60

17379:70 = set local-preference 70

17379:80 = set local-preference 80

17379:90 = set local-preference 90

Todo anuncio feito, sem a aplicação de communities tem por default o valor 100 de local Preference.

6.4 - Manipulando trânsito (nacional/internacional) com communities

- LAB 1:

Considerando as políticas de community para o AS 200:

- 200:70 = set local-preference 70
 - 200:80 = set local-preference 80
 - 200:90 = set local-preference 90
 - 200:110 = set local-preference 110
-
- Utilizar as communities 200:xxx em seguida acessar por telnet o router do AS 200 e verificar as mudanças

6.4 - Manipulando trânsito (nacional/internacional) com communities

- Solução:

```
AS-11X-R1#conf t
AS-11X-R1(config)#route-map AS-X-OUT permit 10
AS-11X-R1(config-route-map)#set community 200:90
AS-11X-R1(config-route-map)#end
AS-11X-R1#clear ip bgp * soft
```

```
AS-12X-R1#conf t
AS-12X-R1(config)#route-map AS-X-OUT permit 10
AS-12X-R1(config-route-map)#set community 200:70
AS-12X-R1(config-route-map)#end
AS-12X-R1#clear ip bgp * soft
```

6.4 - Manipulando trânsito (nacional/internacional) com communities

- LAB 2: De acordo com as políticas do AS 100 abaixo, utilize as communities apropriadas (100:xxxx) para fazer todas as ações:
 1. Fazer o AS 100 não anunciar o prefixo do cliente para o grupo 9
 2. Anunciar para o AS 99 com prepend 2

ASN	Peer	Nao Anuncia	Prepend +1	Prepend +2	Prepend +3
1	Grupo1	1010	1011	1012	1013
2	Grupo2	1020	1021	1022	1023
3	Grupo3	1030	1031	1032	1033
4	Grupo4	1040	1041	1042	1043
5	Grupo5	1050	1051	1052	1053
6	Grupo6	1060	1061	1062	1063
7	Grupo7	1070	1071	1072	1073
8	Grupo8	1080	1081	1082	1083
9	Grupo9	1090	1091	1092	1093
99	Internacional	9910	9911	9912	9913

6.4 - Manipulando trânsito (nacional/internacional) com communities

- **Desafio:**

Grupos de ASN par:

Utilizar as communities da tabela para fazer o AS 100 anunciar seu prefixo com prepend 3 para os ASN's ímpares e prepend 2 para o AS 99 e anunciar normalmente para todo o resto

Grupos de ASN ímpar:

Utilizar as communities da tabela para fazer o AS 100 anunciar seu prefixo com prepend 3 para os ASN's pares e prepend 2 para o AS 99 e anunciar normalmente para todo o resto

6.4 - Manipulando trânsito (nacional/internacional) com communities

- Solução Lab 2:

!--- apagando o route map do lab anterior

```
AS-11X-R1(config)#no route-map AS-X-OUT
```

```
AS-11X-R1#conf t
```

```
AS-11X-R1(config)#route-map AS-X-OUT permit 10
```

```
AS-11X-R1(config-route-map)#set community 100:1090 100:9912
```

```
AS-11X-R1(config-route-map)#end
```

```
AS-11X-R1#clear ip bgp * soft
```

!--- apagando o RM do lab anterior

```
AS-11X-R1(config)#no route-map AS-X-OUT
```

```
AS-12X-R1#conf t
```

```
AS-12X-R1(config)#route-map AS-X-OUT permit 10
```

```
AS-12X-R1(config-route-map)#set community 100:1090 100:9912
```

```
AS-12X-R1(config-route-map)#end
```

```
AS-12X-R1#clear ip bgp * soft
```

6.4 - Manipulando trânsito (nacional/internacional) com communities

- Solução Lab Desafio:(AS Ímpar)

!--- apagando o RM do lab anterior

```
AS-11X-R1(config)#no route-map AS-X-OUT
```

```
AS-11X-R1#conf t
```

```
AS-11X-R1(config)#route-map AS-X-OUT permit 10
```

```
AS-11X-R1(config-route-map)#set community 100:9912 100:1023 100:1043  
100:1063 100:1083
```

```
AS-11X-R1(config-route-map)#end
```

```
AS-11X-R1#clear ip bgp * soft
```

!--- apagando o RM do lab anterior

```
AS-11X-R1(config)#no route-map AS-X-OUT
```

```
AS-12X-R1#conf t
```

```
AS-12X-R1(config)#route-map AS-X-OUT permit 10
```

```
AS-12X-R1(config-route-map)#set community 100:9912 100:1013 100:1033  
100:1053 100:1073 100:1093
```

```
AS-12X-R1(config-route-map)#end
```

```
AS-12X-R1#clear ip bgp * soft
```

6.4 - Manipulando trânsito (nacional/internacional) com communities

- Solução Lab Desafio: (AS Par)

!--- apagando o RM do lab anterior

```
AS-11X-R1(config)#no route-map AS-X-OUT
```

```
AS-11X-R1#conf t
```

```
AS-11X-R1(config)#route-map AS-X-OUT permit 10
```

```
AS-11X-R1(config-route-map)#set community 100:9912 100:1013 100:1033 100:1053  
100:1073 100:1093
```

```
AS-11X-R1(config-route-map)#end
```

```
AS-11X-R1#clear ip bgp * soft
```

!--- apagando o RM do lab anterior

```
AS-11X-R1(config)#no route-map AS-X-OUT
```

```
AS-12X-R1#conf t
```

```
AS-12X-R1(config)#route-map AS-X-OUT permit 10
```

```
AS-12X-R1(config-route-map)#set community 100:9912 100:1013 100:1033 100:1053  
100:1073 100:1093
```

```
AS-12X-R1(config-route-map)#end
```

```
AS-12X-R1#clear ip bgp * soft
```

6.5

IDENTIFICANDO E TRATANDO ROTAS DE PEERS ESPECÍFICOS

6.5 - Identificando e tratando rotas de peers específicos

- Quando utilizamos o comando:

```
AS-11X-R1(config-route-map)#set community xxx:xxx
```

- Estamos substituindo todas as communities que haviam nesse prefixo
- Muitas vezes, é interessante adicionar mais de uma community em "pontos diferentes"

Ex: Recebo prefixos dos meus downstreams e quero inserir uma community 123:123 para controle interno, utilizando o meio convencional, eu acabaria substituindo uma eventual community que meu cliente utilizou para interagir com um outro AS externo.

- O que fazer?

6.5 - Identificando e tratando rotas de peers específicos

- Com o comando "additive" qualquer community existente é preservada

AS-**X**-R4(config-route-map)#set community 123:123 additive

- ✗ *OBS: Digitar o comando "set community" tem efeito "cumulativo" exemplo: se digitarmos a seguinte sequência de comandos:*

```
router#conf t
router(config)#route-map teste permit 10
router(config-route-map)#set community 123:123
router(config-route-map)#exit
router(config)#route-map teste permit 10
router(config-route-map)#set community 111:222
router(config-route-map)#set community 333:214
router(config-route-map)#end
router#
router#conf t
router(config)#route-map teste permit 10
router(config-route-map)#set community 555:82
```

6.5 - Identificando e tratando rotas de peers específicos

- Todas as communities digitadas serão **ACUMULADAS**:

```
router#sh running-config | begin route-map teste
route-map teste permit 10
  set community 111:222 123:123 333:214 555:82
!
```

...

- Lembrem que o "additive" não tem a ver com esse fato, mas sim com fazer com que as communities inseridas com "additive" sejam "somadas" às communities que já estavam taggeadas no(s) prefixo(s)
- Sem o comando "additive", qualquer community existente é substituída
- Sempre utilizem "additive" na hora de colocar communities em rotas recebidas de DOWNSTREAMS