排除ACI中的意外路由洩漏故障

目錄

概觀

ACI通過部署簡單的策略來處理許多傳統複雜的路由和交換配置。這些功能包括洩露VRF之間路由的功能，以便共用服務。傳統上，這涉及許多步驟，例如定義路由目標、建立BGP地址系列、路由區分器以及在許多裝置上複製此配置。

在ACI內，路由洩漏是通過結合使用合約和在子網上設定特定共用標誌來處理的。由於合約和共用子網配置，所有路由洩漏工作所需的傳統配置都在後端處理。

但是，通過提取此配置，識別哪個合約實際上導致路由洩漏會變得更具挑戰性。在具有大量epg、vrf和合約的環境中，情況尤其如此。如果路由在VRF之間意外洩漏，管理員如何確定導致此情況的配置（合約）？

本文檔旨在演示如何確定哪個合約關係導致ACI中的路由在VRF之間洩漏。熟悉路由目標和BGP VPNv4等傳統路由洩漏概念很有幫助。

使用軟體

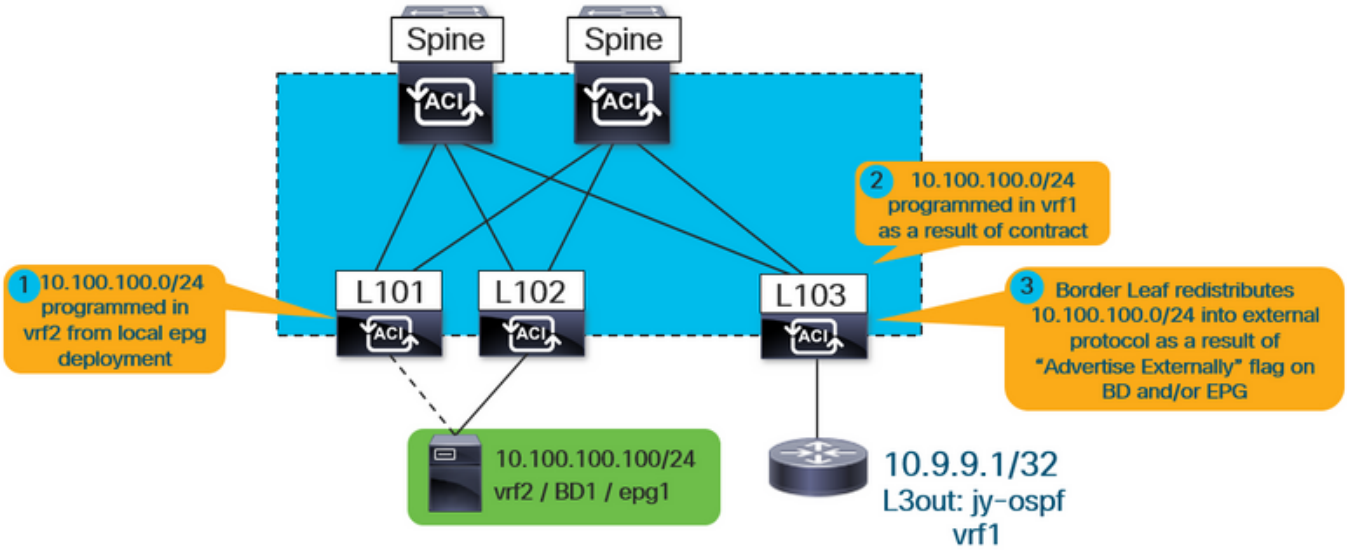
本文檔中的所有示例均基於aci軟體4.2(3j)。

為什麼在VRF y中安裝來自VRF x的網橋域/EPG子網？

本節將重點介紹BD或EPG子網意外地洩漏到其他vrf的情況。對於要洩漏的BD/EPG子網，必須配置「VRF之間的共用」標誌。更具挑戰性的部分是瞭解哪份合約導致此資訊洩露，因此本部分將解決此問題。

在高級別上，這是當BD/EPG子網在VRF之間洩漏時的工作流程。

圖1.



*請注意#3僅當通告共用的l3out時才適用。#1用和#2用始終適用，無論使用的是共用l3out還是共用服務完全是內部服務。

首先，使用者如何知道安裝的路由是否因BD或EPG子網而洩漏？

運行「**show ip route vrf <name>**」時，「**pervasive**」標誌表示該路由是BD或EPG子網。

例如，在上面的拓撲中，在外部vrf(vrf1)的邊界枝葉上會出現這種情況：

```
leaf103# show ip route 10.100.100.100 vrf jy:vrf1
IP Route Table for VRF "jy:vrf1"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%
```

```
      pervasive *via 10.3.144.68%overlay-1, [1/0], 21:29:54, static, tag 4294967292 recursive
next hop: 10.3.144.68/32%overlay-1
```

此外，可以通過運行以下命令檢視子網洩漏的目標vrf:

```
leaf103# vsh -c "show ip route 10.100.100.100 detail vrf jy:vrf1"
IP Route Table for VRF "jy:vrf1"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%
```

```
      pervasive *via 10.3.144.68%overlay-1, [1/0], 21:34:16, static, tag 4294967292 recursive
next hop: 10.3.144.68/32%overlay-1


    vrf crossing information: VNID:0x258003 ClassId:0x18 Flush#:0x2
```

* (請注意，無論目標vrf是否與查詢vrf不同，都會設定vrf交叉資訊。)

在上面的輸出中，vrf交叉的vnid設定為0x258003，或十進位制2457603。如何標識vnid 2457603所屬的vrf?

從APIC只需查詢fvCtx對象並根據該序列進行過濾。

```
apic1# moquery -c fvCtx -f 'fv.Ctx.seg=="2457603"'
Total Objects shown: 1

# fv.Ctx
name                 : vrf2
dn                   : uni/tn-jy/ctx-vrf2
pcEnfDir             : ingress
pcEnfPref            : enforced
pcTag                : 49153
scope                : 2457603
seg                  : 2457603
```

正如預期的那樣，正在從vrf2 vrf獲知路由。

目前還不知道使用的是哪份合約、提供哪些epg，以及哪些epg用於安裝此路由。在提供商和消費者關係方面，需要記住以下幾點：
1.對於vrf間合約關係，合約（以及生成的分割槽規則）僅安裝在消費者epg的vrf中。因此，提供程式vrf中的「show zoning-rule」不會顯示關係。
2.即使合約僅安裝在使用者vrf中，提供商vrf也必須獲取使用者vrf BD子網的路由，這意味著枝葉必須具有對合約的某些配置引用。

當路由意外地洩漏給使用者VRF時識別合約

枝葉上的ipCons對象安裝在枝葉上，其中引用……

a.)洩漏給消費者vrf的路由

b.)建立此種關係的合約

c.)提供商和消費者epg之間的關係。

在下面的輸出中，「jy:vrf1」是路由洩漏到的使用者vrf，而「10.100.100.0/24」是洩漏的路由。

```
leaf103# moquery -c ipCons -f 'ip.Cons.dn*"jy:vrf1/rt-\[10.100.100.0/24\]"'
Total Objects shown: 1

# ip.Cons
consDn       : cdef-[uni/tn-jy/brc-shared]/epgCont-[uni/tn-jy/out-jy-ospf/instP-all]/fr-[uni/tn-
```

```
jy/brc-shared/dirass/cons-[uni/tn-jy/out-jy-ospf/instP-all]-any-no]/to-[uni/tn-jy/brc-
shared/dirass/prov-[uni/tn-jy/ap-ap1/epg-epg1]-any-no]
subConsDn    :
childAction  :
dn           : sys/ipv4/inst/dom-jy:vrf1/rt-[10.100.100.0/24]/rsrouteToRouteDef-[bd-[uni/tn-
jy/BD-bd1]-isSvc-no/epgDn-[uni/tn-jy/ap-ap1/epg-epg1]/rt-[10.100.100.1/24]]/cons-[cdef-[uni/tn-
jy/brc-shared]/epgCont-[uni/tn-jy/out-jy-ospf/instP-all]/fr-[uni/tn-jy/brc-shared/dirass/cons-
[uni/tn-jy/out-jy-ospf/instP-all]-any-no]/to-[uni/tn-jy/brc-shared/dirass/prov-[uni/tn-jy/ap-
ap1/epg-epg1]-any-no]]-sub-[]
lcOwn        : local
modTs        : 2019-12-23T12:50:51.440-05:00
name         :
nameAlias    :
rn           : cons-[cdef-[uni/tn-jy/brc-shared]/epgCont-[uni/tn-jy/out-jy-ospf/instP-all]/fr-
[uni/tn-jy/brc-shared/dirass/cons-[uni/tn-jy/out-jy-ospf/instP-all]-any-no]/to-[uni/tn-jy/brc-
shared/dirass/prov-[uni/tn-jy/ap-ap1/epg-epg1]-any-no]]-sub-[]
status       :
```

從上述輸出中，約定名稱為「shared」，消費者epg為l3out epg「uni/tn-jy/out-jy-ospf/instP-all」，提供商epg為「uni/tn-jy/ap-ap1/epg-epg1」。

當路由意外地洩漏給提供商VRF時識別合約

consNode對象安裝在提供程式vrf中的枝葉上。它引用了被洩漏的消費者vrf中的BD子網、合約以及關係中的epg。在查詢此對象之前，請查詢配置路由的BD子網。可通過查詢apic上的fvSubnet對象來完成此操作：

```
apic1:~> moquery -c fvSubnet -f 'fv.Subnet.dn*"10.100.100"'

# fv.Subnet
ip           : 10.100.100.1/24
dn           : uni/tn-jy/BD-bd1/subnet-[10.100.100.1/24]
preferred    : no
rn           : subnet-[10.100.100.1/24]
scope        : public,shared
```

該路由配置在tn-jy/BD-bd1網橋域中。使用此命令以及提供商vrf（路由被洩漏到）的vnid運行以下命令。

```
leaf103# moquery -c consNode -f 'cons.Node.dn*"2949122"' -f 'cons.Node.dn*"tn-jy/BD-bd1"'
Total Objects shown: 1

# cons.Node
consDn       : cdef-[uni/tn-jy/brc-shared]/epgCont-[uni/tn-jy/out-jy-ospf/instP-all]/fr-[uni/tn-
jy/brc-shared/dirass/prov-[uni/tn-jy/out-jy-ospf/instP-all]-any-no]/to-[uni/tn-jy/brc-
shared/dirass/cons-[uni/tn-jy/ap-ap1/epg-epg1]-any-no]
annotation   :
childAction  :
descr        :
dn           : consroot-[bd-[uni/tn-jy/BD-bd1]-isSvc-no]-[sys/ctx-[vxlan-2949122]]/consnode-
[cdef-[uni/tn-jy/brc-shared]/epgCont-[uni/tn-jy/out-jy-ospf/instP-all]/fr-[uni/tn-jy/brc-
shared/dirass/prov-[uni/tn-jy/out-jy-ospf/instP-all]-any-no]/to-[uni/tn-jy/brc-
shared/dirass/cons-[uni/tn-jy/ap-ap1/epg-epg1]-any-no]]
extMngdBy    :
lcOwn        : local
modTs        : 2019-12-23T12:25:36.153-05:00
name         :
nameAlias    :
rn           : consnode-[cdef-[uni/tn-jy/brc-shared]/epgCont-[uni/tn-jy/out-jy-ospf/instP-
all]/fr-[uni/tn-jy/brc-shared/dirass/prov-[uni/tn-jy/out-jy-ospf/instP-all]-any-no]/to-[uni/tn-
```

```
jy/brc-shared/dirass/cons-[uni/tn-jy/ap-ap1/epg-epg1]-any-no]]
status      :
uid         : 0
```

從上述輸出中，約定名稱為「shared」，使用者epg為「uni/tn-jy/ap-ap1/epg-epg1」，提供程式epg為l3out「tn-jy/out-jy-ospf/instP-all」。

當路由被消耗的vzAny合約意外洩露時，識別該合約

從驗證角度看，vzAny示例將與傳統的提供商/消費者關係完全相同。以下示例將僅演示此情況的樣子。請注意，只有vzAny作為使用者才支援vrf間合約。

vz任意示例1:路由意外洩漏到消費者VRF

與檢視在消費者vrf中執行驗證的第一個示例相似，將再次使用ipCons對象。

```
leaf103# moquery -c ipCons -f 'ip.Cons.dn*"jy:vrf1/rt-\[10.100.100.0/24\]"'
Total Objects shown: 1

# ip.Cons
consDn      : cdef-[uni/tn-jy/brc-shared]/epgCont-[uni/tn-jy/ctx-vrf1/any]/fr-[uni/tn-jy/brc-
shared/any-[uni/tn-jy/ctx-vrf1/any]-type-cons_as_any/cons-[uni/tn-jy/ctx-vrf1/any]-any-yes]/to-
[uni/tn-jy/brc-shared/dirass/prov-[uni/tn-jy/ap-ap1/epg-epg1]-any-no]
subConsDn   :
childAction :
dn          : sys/ipv4/inst/dom-jy:vrf1/rt-[10.100.100.0/24]/rsrouteToRouteDef-[bd-[uni/tn-
jy/BD-bd1]-isSvc-no/epgDn-[uni/tn-jy/ap-ap1/epg-epg1]/rt-[10.100.100.1/24]]/cons-[cdef-[uni/tn-
jy/brc-shared]/epgCont-[uni/tn-jy/ctx-vrf1/any]/fr-[uni/tn-jy/brc-shared/any-[uni/tn-jy/ctx-
vrf1/any]-type-cons_as_any/cons-[uni/tn-jy/ctx-vrf1/any]-any-yes]/to-[uni/tn-jy/brc-
shared/dirass/prov-[uni/tn-jy/ap-ap1/epg-epg1]-any-no]]-sub-[]
lcOwn        : local
modTs        : 2019-12-23T13:11:08.077-05:00
name         :
nameAlias    :
rn           : cons-[cdef-[uni/tn-jy/brc-shared]/epgCont-[uni/tn-jy/ctx-vrf1/any]/fr-[uni/tn-
jy/brc-shared/any-[uni/tn-jy/ctx-vrf1/any]-type-cons_as_any/cons-[uni/tn-jy/ctx-vrf1/any]-any-
yes]/to-[uni/tn-jy/brc-shared/dirass/prov-[uni/tn-jy/ap-ap1/epg-epg1]-any-no]]-sub-[]
status       :
```

從上述輸出中，合約名稱為「shared」，使用者epg為vrf1 vzAny「tn-jy/ctx-vrf1/any」，提供商epg為「uni/tn-jy/ap-ap1/epg-epg1」。

vz任意示例2:路由意外洩漏到提供商VRF

與檢視在提供程式vrf中執行驗證的第二示例相似，將再次使用consNode對象。請記住獲取配置洩漏子網的BD的bd名稱以及它被洩漏到的vrf的vnid。

```
leaf103# moquery -c consNode -f 'cons.Node.dn*"vxlan-2949122"' -f 'cons.Node.dn*"tn-jy/BD-bd1"'
Total Objects shown: 1

# cons.Node
consDn      : cdef-[uni/tn-jy/brc-shared]/epgCont-[uni/tn-jy/out-jy-ospf/instP-all]/fr-[uni/tn-
jy/brc-shared/dirass/prov-[uni/tn-jy/out-jy-ospf/instP-all]-any-no]/to-[uni/tn-jy/brc-
shared/any-[uni/tn-jy/ctx-vrf2/any]-type-cons_as_any/cons-[uni/tn-jy/ctx-vrf2/any]-any-yes]
annotation  :
childAction :
descr       :
```

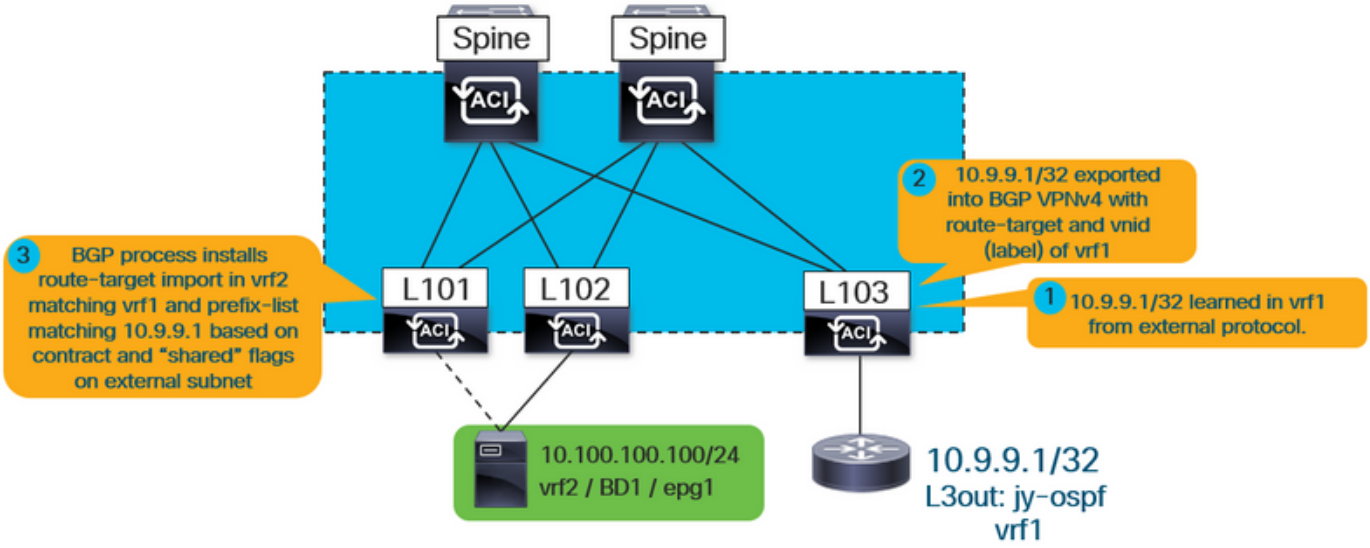
```
dn           : consroot-[bd-[uni/tn-jy/BD-bd1]-isSvc-no]-[sys/ctx-[vxlan-2949122]]/consnode-
[cdef-[uni/tn-jy/brc-shared]/epgCont-[uni/tn-jy/out-jy-ospf/instP-all]/fr-[uni/tn-jy/brc-
shared/dirass/prov-[uni/tn-jy/out-jy-ospf/instP-all]-any-no]/to-[uni/tn-jy/brc-shared/any-
[uni/tn-jy/ctx-vrf2/any]-type-cons_as_any/cons-[uni/tn-jy/ctx-vrf2/any]-any-yes]]
extMngdBy    :
lcOwn        : local
modTs        : 2019-12-23T13:06:09.016-05:00
name         :
nameAlias    :
rn           : consnode-[cdef-[uni/tn-jy/brc-shared]/epgCont-[uni/tn-jy/out-jy-ospf/instP-
all]/fr-[uni/tn-jy/brc-shared/dirass/prov-[uni/tn-jy/out-jy-ospf/instP-all]-any-no]/to-[uni/tn-
jy/brc-shared/any-[uni/tn-jy/ctx-vrf2/any]-type-cons_as_any/cons-[uni/tn-jy/ctx-vrf2/any]-any-
yes]]
status       :
uid          : 0
```

從上述輸出中，合約名稱為「shared」，使用者epg為vrf2 vzAny "tn-jy/ctx-vrf2/any"，提供程式epg為l3out "tn-jy/out-jy-ospf/instP-all"。

為什麼在VRF x中安裝來自VRF y的外部路由？

在高級別，這是在vrf之間洩露l3out-learning（外部）路由時的工作流程。

圖2.



如上所示，內部vrf（本例中為vrf2）安裝與vrf1匹配的路由目標匯入。它還會在bgp進程上安裝一個匯入對映，該對映應具有字首清單條目，與l3out中定義的所有已選擇「共用路由控制子網」標誌的條目匹配。

無論哪個epg是提供商還是消費者，驗證步驟都是相同的，因為合約始終負責導致路由目標匯入和相應的字首清單，這些字首清單將洩漏要安裝的路由。

首先，驗證路由實際上是否通過l3out獲知：

```
leaf101# show ip route 10.9.9.1 vrf jy:vrf2
IP Route Table for VRF "jy:vrf2"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
```

```
'%


       via 10.3.248.4%

    overlay-1, [200/5], 00:00:13,

    bgp-65001, internal, tag 65001
```

在上方範例中，從指向重疊中另一個枝葉的交換矩陣bgp程式得知此訊息的事實表示此訊息來自l3out。

運行以下資訊以獲取有關從哪些vrf獲知的更多資訊：

```
leaf101# vsh -c "show ip route 10.9.9.1 detail vrf jy:vrf2"
IP Route Table for VRF "jy:vrf2"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%


       rw-vnid: 0x2d0002 table-id: 0x17 rw-mac: 0
```

如本文檔前面所示，rewrite vnid 0x2d0002 / 2949122是目標vrf。在外部路由示例中，將rw-vnid值設定為非零值表示這是從其他vrf獲取的。在apic上運行**moquery -c fvCtx -f 'fv.Ctx.seg=="2949122"**將指示此屬於vrf1。

接下來，查詢路由目標匯入以及與bgp進程關聯的匯入路由對映。

```
leaf101# show bgp process vrf jy:vrf2

Information regarding configured VRFs:

BGP Information for VRF jy:vrf2
VRF Type                        : System
VRF Id                          : 23
VRF state                       : UP
VRF configured                  : yes
VRF refcount                    : 0
VRF VNID                        : 2457603
Router-ID                       : 10.100.100.1
Configured Router-ID            : 0.0.0.0
```

```
Confed-ID                       : 0
Cluster-ID                      : 0.0.0.0
MSITE Cluster-ID                : 0.0.0.0
No. of configured peers         : 0
No. of pending config peers     : 0
No. of established peers        : 0
VRF RD                          : 101:2457603
VRF EVPN RD                     : 101:2457603

    Information for address family IPv4 Unicast in VRF jy:vrf2
    Table Id                    : 17
    Table state                 : UP
    Table refcount              : 5
    Peers      Active-peers    Routes     Paths      Networks   Aggregates
    0          0               2          2          0          0

    Redistribution
        None

    Wait for IGP convergence is not configured
    Import route-map 2457603-shared-svc-leak <-- bgpRtCtrlMapP
    Export RT list:
        65001:2457603
    Import RT list:
        65001:2457603
        65001:2949122 <-- bgpRttEntry
    Label mode: per-prefix
```

上面的內部vrf正在匯出和匯入其自己的路由目標(65001:2457603)。 它還在匯入65001:2949122。 2949122 RT對應於它正在匯入的vrf vnid(vrf1)。bgpRtCtrlMapP是包含字首清單的匯入路由對映的對象名稱。bgpRttEntry是匯入路由目標的對象名稱。

接下來，使用正在學習外部vrf路由的內部vrf的vnid，查詢在shared services route-map中安裝的所有字首清單。

```
leaf101# moquery -c rtpfxEntry -f 'rtpfx.Entry.dn*"pfxlist-IPv4'.*'2457603-shared-svc-leak"' |
egrep "criteria|dn|pfx|toPfxLen"
# rtpfx.Entry
criteria         : inexact
dn               : sys/rpm/pfxlist-IPv4-2949122-24-25-2457603-shared-svc-leak/ent-2
pfx              : 0.0.0.0/0
toPfxLen         : 32
# rtpfx.Entry
criteria         : exact
dn               : sys/rpm/pfxlist-IPv4-2949122-24-25-2457603-shared-svc-leak/ent-3
pfx              : 10.9.9.1/32
toPfxLen         : 0
# rtpfx.Entry
criteria         : exact
dn               : sys/rpm/pfxlist-IPv4-2949122-24-25-2457603-shared-svc-leak/ent-1
pfx              : 10.9.9.0/24
toPfxLen         : 0
```

每個條目應對應一個外部子網。「exact / inexact」屬性指明是否在外部子網上設定了「aggregate shared」標誌。帶有不精確標誌的0.0.0.0/0字首表示它將匹配所有更具體的路由（實際上就是所有路由）。 帶有確切標誌的10.9.9.0/24字首表示它只與該/24匹配。

查詢與被意外洩露的路由匹配的條目。在這種情況下，字首10.9.9.1/32將在上述輸出中由ent-2和ent-3匹配。

使用字首清單名稱，在路由對映中查詢與其匹配的序列號。

```
leaf101# moquery -c rtmapRsRtDstAtt -f 'rtmap.RsRtDstAtt.tDn*"pfxlist-IPv4-2949122-24-25-2457603-shared-svc-leak"'
Total Objects shown: 1

# rtmap.RsRtDstAtt
tDn          : sys/rpm/pfxlist-IPv4-2949122-24-25-2457603-shared-svc-leak
childAction  :
dn           : sys/rpm/rtmap-2457603-shared-svc-leak/ent-1001/mrtdst/rsrtDstAtt-
[sys/rpm/pfxlist-IPv4-2949122-24-25-2457603-shared-svc-leak]
forceResolve : yes
lcOwn        : local
modTs        : 2019-12-24T11:17:08.668-05:00
rType        : mo
rn           : rsrtDstAtt-[sys/rpm/pfxlist-IPv4-2949122-24-25-2457603-shared-svc-leak]
state        : formed
stateQual    : none
status       :
tCl          : rtpfxRule
tSKey        : IPv4-2949122-24-25-2457603-shared-svc-leak
tType        : mo
```

上面的輸出顯示，這是路由對映條目1001。此處的最後一部分是瞭解哪個合約負責在2457603-shared-svc-leak route-map中建立路由對映條目1001。可以在fvAppEpGCons對象的枝葉上查詢此項。

```
leaf101# moquery -c fvAppEpGCons -f 'fv.AppEpGCons.dn*"rtmap-2457603-shared-svc-leak/ent-1001"'
Total Objects shown: 1

# fv.AppEpGCons
consDn       : cdef-[uni/tn-jy/brc-shared]/epgCont-[uni/tn-jy/ap-ap1/epg-epg1]/fr-[uni/tn-jy/brc-shared/dirass/prov-[uni/tn-jy/ap-ap1/epg-epg1]-any-no]/to-[uni/tn-jy/brc-shared/dirass/cons-[uni/tn-jy/out-jy-ospf/instP-all]-any-no]
childAction  :
descr        :
dn           : uni/ctxrefcont/ctxref-[sys/ctx-[vxlan-2457603]]/epgref-[uni/tn-jy/ap-ap1/epg-epg1]/epgpol-[sys/rpm/rtmap-2457603-shared-svc-leak/ent-1001]/epgcons-[cdef-[uni/tn-jy/brc-shared]/epgCont-[uni/tn-jy/ap-ap1/epg-epg1]/fr-[uni/tn-jy/brc-shared/dirass/prov-[uni/tn-jy/ap-ap1/epg-epg1]-any-no]/to-[uni/tn-jy/brc-shared/dirass/cons-[uni/tn-jy/out-jy-ospf/instP-all]-any-no]]
lcOwn        : local
modTs        : 2019-12-23T14:36:48.753-05:00
name         :
nameAlias    :
ownerKey     :
ownerTag     :
rn           : epgcons-[cdef-[uni/tn-jy/brc-shared]/epgCont-[uni/tn-jy/ap-ap1/epg-epg1]/fr-[uni/tn-jy/brc-shared/dirass/prov-[uni/tn-jy/ap-ap1/epg-epg1]-any-no]/to-[uni/tn-jy/brc-shared/dirass/cons-[uni/tn-jy/out-jy-ospf/instP-all]-any-no]]
status       :
```

上面的輸出顯示，合約名稱為「shared」，提供程式epg為「tn-jy/ap-ap1/epg-epg1」，使用者l3out epg為「tn-jy/out-jy-ospf/instP-all」

摘要

從BD/EPG子網洩露的路由

如果洩漏的路由在「show ip route」中設定了「沈浸式」標籤，則會從配置的BD/EPG子網中將其洩漏。以下兩個命令可用於檢查導致此漏洞洩露的合約關係。它們將在路由意外安裝的枝葉上運行

。

如果路由意外洩漏的vrf是使用者：
moquery -c ipCons -f 'ip.Cons.dn*"jy:vrf1/rt-\[10.100.100.0/24\]"' ***<—jy:vrf1是路由洩漏到的vrf的名稱，路由是10.100.100.0/24***

如果路由意外洩漏的vrf是提供商：
moquery -c consNode -f 'cons.Node.dn*"2949122"' -f 'cons.Node.dn*"tn-jy/BD-bd1"'
<—2949122是路由洩漏到的vrf的vnid，tn-jy/BD-bd1是配置子網的BD的名稱（在vrf中，路由被洩漏）。

從L3out洩露的路由

如果洩漏的路由是通過內部交換矩陣iBGP進程獲知，並且運行vsh -c "show ip route x.x.x.x/y detail vrf <name>"顯示非零rw-vnid值，則路由是從另一個vrf中的l3out獲知。無論哪個epg是消費者，哪一個是提供商，驗證都是相同的。

1.*識別內部vrf bgp進程上的共用服務匯入路由對映：*
show bgp process vrf jy:vrf2 | grep "Import route-map" ***<—jy:vrf2是路由洩漏到的內部vrf***

2.識別與洩漏的路由匹配的shared services route-map中的字首清單：
moquery -c rtpfxEntry -f 'rtpfx.Entry.dn*"pfxlist-IPv4'.*'2457603-shared-svc-leak"' | egrep "criteria|dn|pfx|toPfxLen" ***<—2457603本示例中內部vrf的vnid***

3.在查詢哪個字首清單引用該路由後，確定哪個路由對映序列號引用該清單：
moquery -c rtmapRsRtDstAtt -f 'rtmap.RsRtDstAtt.tDn*"pfxlist-IPv4-2949122-24-25-2457603-shared-svc-leak"' ***<—pfxlist-IPv4-2949122-24-25-2457603-shared-svc-leak是字首清單名稱***

4.使用rtmap和條目編號運行以下命令，以找出推入該路由對映條目的合約關係：
moquery -c fvAppEpGCons -f 'fv.AppEpGCons.dn*"rtmap-2457603-shared-svc-leak/ent-1001"'
<—rtmap-2457603-shared-svc-leak/ent-1001是步驟3中的路由對映名稱和條目編號。