# Fehlerbehebung bei unerwarteten Route Leaking in der ACI

## Inhalt



## Übersicht

Die ACI verarbeitet viele traditionell komplexe Routing- und Switching-Konfigurationen mithilfe einfacher Richtlinien. Zu diesen Funktionen gehört die Möglichkeit, Routen zwischen VRFs zu übergeben, um gemeinsam genutzte Services zu ermöglichen. Bisher wurden hierzu viele Schritte durchgeführt, z. B. das Definieren von Routenzielen, das Erstellen von BGP-Adressfamilien, das Erstellen von Route Distinguisers und das Replizieren dieser Konfiguration auf vielen Geräten.

Innerhalb der ACI erfolgt das Route Leaking über eine Kombination von Verträgen und die Festlegung spezifischer gemeinsam genutzter Flags in Subnetzen. Die gesamte herkömmliche Konfiguration, die für die Weiterleitung von Routen erforderlich ist, wird auf dem Backend als Ergebnis des Vertrags und der gemeinsam genutzten Subnetzkonfiguration behandelt.

Wenn diese Konfiguration abstrahiert wird, kann es jedoch schwieriger sein, herauszufinden, welcher Vertrag tatsächlich das Herausgeben einer Route bewirkt. Dies gilt insbesondere für Umgebungen mit einer großen Anzahl von epgs, vrfs und Verträgen. Wenn eine Route unerwartet zwischen VRFs durchgesickert wird, wie kann ein Administrator ermitteln, welche Konfiguration (Vertrag) dies verursacht?

In diesem Dokument wird veranschaulicht, wie die Vertragsbeziehung, durch die eine Route in der ACI zwischen VRFs durchgesickert wird, identifiziert werden kann. Es ist hilfreich, sich bereits mit herkömmlichen Route-Leaking-Konzepten wie Route Targets und BGP VPNv4 vertraut zu machen.

### Verwendete Software

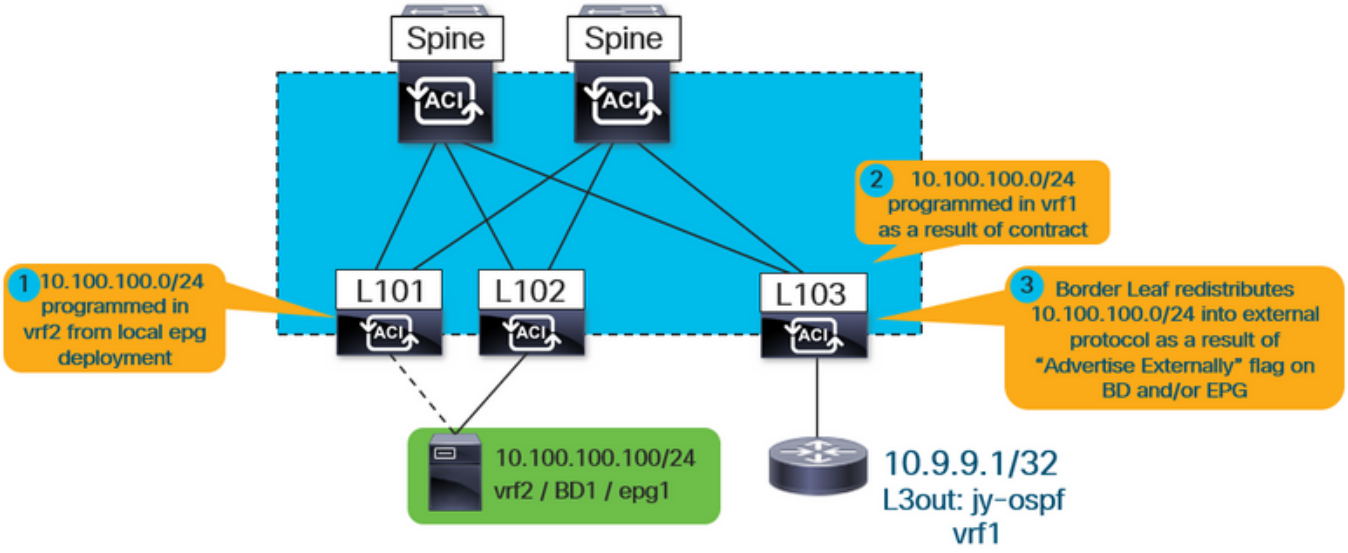
Alle Beispiele in diesem Dokument basieren auf der ACI-Software 4.2(3j).

# Warum wird eine Bridge-Domäne/ein EPG-Subnetz von VRF x in VRF y installiert?

In diesem Abschnitt wird das Szenario behandelt, in dem ein BD- oder EPG-Subnetz unerwartet an ein anderes VRF weitergeleitet wird. Damit ein BD/EPG-Subnetz durchsickert, muss das Flag "Shared Between VRFs" konfiguriert werden. Eine größere Herausforderung besteht darin, zu verstehen, welcher Vertrag die Veröffentlichung dieses Textes verursacht, damit dieser Abschnitt behandelt wird.

Auf hoher Ebene ist dies der Workflow für das, was passiert, wenn ein BD/EPG-Subnetz zwischen VRFs durchsickert.

**Abbildung 1:**




* Hinweis: Nr. 3 gilt nur, wenn ein freigegebenes l3out angegeben wird. Die Nummern 1 und 2 gelten immer unabhängig davon, ob ein gemeinsam genutztes l3out verwendet wird oder die gemeinsam genutzten Services vollständig intern sind.

Wie kann der Benutzer erstens wissen, ob die installierte Route durch ein BD- oder EPG-Subnetz übertragen wird?

Bei der Ausführung von "**show ip route vrf <name>**" gibt das "**pervasive**" Flag an, dass es sich bei der Route um ein BD- oder EPG-Subnetz handelt.

In der obigen Topologie wird dies beispielsweise auf dem Grenz-Leaf im externen VRF (VRF1) angezeigt:

```
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
```

Darüber hinaus kann das Ziel-VRF, von dem das Subnetz übertragen wurde, mithilfe des folgenden Befehls angezeigt werden:

```
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
```

*(Beachten Sie, dass die VRF-Kreuzungsinformationen unabhängig davon festgelegt werden, ob das Ziel-VRF von dem VRF für die Suche abweicht.)

In der obigen Ausgabe wird die VRF-Quer-VNID auf 0x258003 oder die Dezimalzahl 2457603 festgelegt. Wie kann das VRF identifiziert werden, zu dem vnid 2457603 gehört?

Über den APIC wird einfach das fvCtx-Objekt abgefragt und auf Basis des Segments gefiltert.

```
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
```

Wie erwartet wird die Route vom VRF2-VRF erfasst.

Es ist an diesem Punkt noch unbekannt, welcher Vertrag verwendet wird und welches epg

bereitstellt und welches epg verwendet, um diese Route zu installieren. Hinsichtlich der Anbieter- und Verbraucherbeziehung sind folgende Punkte zu beachten:
1. Für eine VRF-Vertragsbeziehung wird der Vertrag (und die daraus resultierende Zoning-Regel) nur im VRF des Consumer-EPG installiert. Daher zeigt "show zoning-regel" im Anbieter-VRF die Beziehung nicht an.
2. Obwohl der Vertrag nur im Consumer-VRF installiert ist, muss der Anbieter-VRF die Route für das Consumer-VRF-BD-Subnetz abrufen, d. h. das Leaf muss über einen Konfigurationsverweis auf den Vertrag verfügen.

## Identifizieren Sie den Vertrag, wenn die Route unerwartet an die VRF-Instanz des Verbrauchers weitergeleitet wird.

Das ipCons-Objekt auf dem Leaf wird auf dem Leaf installiert, der auf..

a) die Route, die in das VRF des Verbrauchers übertragen wird

b) den Vertrag, der die Beziehung begründet

c) Anbieter und Verbraucher in der Beziehung.

In der folgenden Ausgabe ist "jy:vrf1" das Consumer-VRF, zu dem die Route durchgesickert wird, und "10.100.100.0/24" ist die Route, die durchgesickert wird.

```
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       :
```

Aus der obigen Ausgabe lautet der Vertragsname "shared", der Consumer-epg lautet l3out epg "uni/tn-jy/out-jy-ospf/instP-all", und der Provider-epg lautet "uni/tn-jy/ap-ap1/epg-epg1".

## Identifizieren Sie den Vertrag, wenn die Route unerwartet an die VRF-Instanz des Anbieters weitergeleitet wird.

Das consNode-Objekt wird auf dem Leaf im Anbieter-VRF installiert. Es verweist auf das BD-Subnetz im Consumer-VRF, das durchgesickert wird, den Vertrag und die EPGs innerhalb der Beziehung. Bevor Sie dieses Objekt abfragen, suchen Sie das BD-Subnetz, in dem die Route

konfiguriert ist. Dies kann durch Abfragen des fvSubnet-Objekts auf dem apic erfolgen:

```
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
```

Die Route wird in der Bridge-Domäne tn-jy/BD-bd1 konfiguriert. Verwenden Sie diese und die ID des Anbieter-VRF (in den die Route gelangt ist), um den folgenden Befehl auszuführen.

```
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
```

Aus der obigen Ausgabe lautet der Vertragsname "shared", der Consumer-epg "uni/tn-jy/ap-ap1/epg-epg1" und der Provider-epg lautet l3out "tn-jy/out-jy-ospf/instP-all".

## Identifizieren Sie den Vertrag, wenn die Route unerwartet durch einen verbrauchten vzAny-Vertrag übertragen wird.

Das vzAny-Beispiel ist aus überprüfungssicht identisch mit einer traditionellen Anbieter-/Consumer-Beziehung. Die folgenden Beispiele zeigen nur, wie das aussehen würde. Beachten Sie, dass ein Inter-VRF-Vertrag nur mit dem vzAny als Consumer unterstützt wird.

### vzAny Beispiel 1: Unerwartetes Weiterleiten an Consumer-VRF

Ähnlich wie im ersten Beispiel wurde untersucht, wo die Überprüfung im Consumer-VRF durchgeführt wurde, wird das ipCons-Objekt erneut verwendet.

```
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       :
```

Aus der oben angegebenen Ausgabe lautet der Vertragsname "shared", der Consumer-epg ist der vrf1 vzAny "tn-jy/ctx-vrf1/any" und der Provider-epg lautet "uni/tn-jy/ap-ap1/epg-epg1".

### vzAny Beispiel 2: Route unerwartet an VRF-Anbieter weitergeleitet

Ähnlich wie im zweiten Beispiel wurde untersucht, wo die Überprüfung im Anbieter-VRF durchgeführt wurde, wird das consNode-Objekt erneut verwendet. Denken Sie daran, den bd-Namen des BD abzurufen, in dem das durchgesickerte Subnetz konfiguriert ist, und den vnid des VRF, in den es durchsickert.

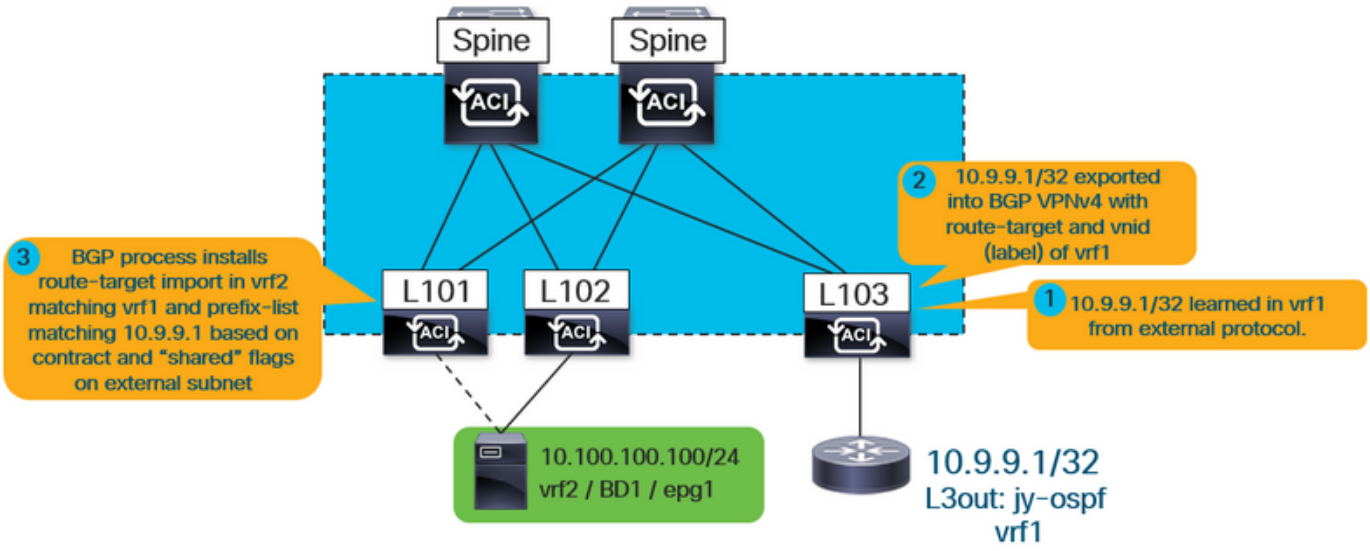
```
leaf103# moquery -c consNode -f 'cons.Node.dn*"vxlan-2949122"' -f 'cons.Node.dn*"tn-jy/BD-bd1"'
Total Objects shown: 1

# 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
```

Aus der obigen Ausgabe lautet der Vertragsname "shared", der Consumer-epg ist der vrf2 vzAny "tn-jy/ctx-vrf2/any" und der Provider-epg ist l3out "tn-jy/out-jy-ospf/instP-all".

# Warum ist eine externe Route von VRF y in VRF x installiert?

Auf einer höheren Ebene ist dies der Workflow für das, was geschieht, wenn eine l3out-gelernte (externe) Route zwischen VRFs durchsickert.

**Abbildung 2:**




Wie oben gezeigt, installiert das interne VRF (in diesem Fall VRF2) einen Route Target-Import, der VRF1 entspricht. Darüber hinaus wird eine Importzuordnung für den BGP-Prozess installiert, der Präfixlisteneinträge enthalten sollte, die mit allem übereinstimmen, was im l3out definiert ist, das das Flag "shared route control subnet" ausgewählt hat.

Unabhängig davon, welcher epg Anbieter oder Consumer ist, sind die Überprüfungsschritte identisch, da der Vertrag immer für den Import des Route-Target-Objekts und die entsprechenden Präfixlisten verantwortlich ist, die die zu installierenden Routen undicht machen.

Überprüfen Sie zunächst, ob die Route tatsächlich über ein l3out abgerufen wird:

```
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
```

Im obigen Beispiel zeigt die Tatsache, dass sie aus dem Fabric-BGP-Prozess gelernt wurde, der auf ein anderes Leaf im Overlay verweist, dass dies von einem l3out stammt.

Führen Sie die folgenden Informationen aus, um weitere Informationen darüber zu erhalten, von welchem VRF er gelernt wurde:

```
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
```

Wie in diesem Dokument weiter oben gezeigt, rewrite vnid 0x2d0002 / 2949122 ist das Ziel-VRF. Der rw-vnid-Wert, der in einem externen Routenbeispiel auf einen Wert von nicht null festgelegt wird, gibt an, dass dieser von einem anderen VRF gelernt wurde. Wenn **moquery -c fvCtx -f'fv.Ctx.seg="2949122"** auf der APIC ausgeführt wird, wird darauf hingewiesen, dass dies zu vrf1 gehört.

Als Nächstes finden Sie die Route-Target-Importe sowie die Import-Route-Map, die an den BGP-Prozess gebunden ist.

```
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
```

Das interne VRF exportiert und importiert sein eigenes Route Target (65001:2457603). Es importiert auch 65001:2949122. Der RT 2949122 entspricht dem VRF-VNID, das er importiert (VRF1). bgpRtCtrlMapP ist der Objektname für die Import-Route-Map, die die Präfixlisten enthält. bgpRttEntry ist der Objektname für das Importroute-Target.

Als Nächstes werden mithilfe des VNID des internen VRF, der die externen VRF-Routen erlernt, alle Präfixlisten abgefragt, die in der Route Map für die gemeinsam genutzten Services installiert sind.

```
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
```

Jeder Eintrag muss einem externen Subnetz entsprechen. Das Attribut "exakt/inexakt" gibt an, ob das Flag "aggregate shared" auf dem externen Subnetz festgelegt wurde. Das Präfix 0.0.0.0/0 mit dem inexakten Flag gibt an, dass es alle Routen, die spezifischerer sind (im Prinzip alles), abgleicht. Das Präfix 10.9.9.0/24 mit dem genauen Flag gibt an, dass es nur /24 entspricht.

Suchen Sie nach dem Eintrag (oder den Einträgen), der der Route entspricht, die unerwartet durchgesickert wird. In diesem Fall lautet das Präfix 10.9.9.1/32, und in den oben genannten Ausgaben werden ent-2 und ent-3 zugeordnet.

Suchen Sie mithilfe des Namens der Präfixliste die Sequenznummer in der Routenübersicht, die der Liste entspricht.

```
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
```

Die obige Ausgabe zeigt, dass es sich um den route-map-Eintrag 1001 handelt. Der letzte Teil hier ist zu verstehen, welcher Vertrag für die Erstellung des route-map-Eintrags 1001 innerhalb der route-map-Map 2457603-shared-svc-leck zuständig war. Dies kann auf dem Leaf vom fvAppEpGCons-Objekt abgefragt werden.

```
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       :
```

Die obige Ausgabe zeigt, dass der Vertragsname "shared" ist, der Provider-epg "tn-jy/ap-ap1/epg-epg1" und der Consumer-l3out epg "tn-jy/out-jy-ospf/instP-all".

# Zusammenfassung

## Route Leaked from BD/EPG Subnet

Wenn für eine durchgesickerte Route das Flag "pervasive" in "show ip route" festgelegt ist, wird sie aus einem konfigurierten BD/EPG-Subnetz übertragen. Die folgenden beiden Befehle können verwendet werden, um zu überprüfen, welche Vertragsbeziehung das Leeren dieses Befehls

verursacht. Sie werden auf dem Leaf ausgeführt, auf dem die Route unerwartet installiert wird.

*Wenn das VRF, in dem die Route unerwartet gesickert wird, der Verbraucher ist:*
**moquery -c ipCons -f 'ip.cons.dn*"jy:vrf1/rt-\[10.100.100.0/24\]"** ***<—jy:vrf1 ist der Name des VRF, zu dem die Route gelangt, und die Route lautet 10.100.100.0/24.***

*Wenn das VRF, in dem die Route unerwartet übertragen wird, der Anbieter ist:*
**moquery -c consNode -f'cons.Node.dn*"2949122" -f'cons.Node.dn*"tn-jy/BD-bd1"** ***< - 2949122 ist das VRF, an das die Route übertragen wird. tn-jy/BD-bd1 ist der Name des BD, für den das Subnetz konfiguriert ist (innerhalb des VRF, von dem die Route durchsickert wird).***

## Route Leaked von L3Out

Wenn die gesickerte Route durch den internen Fabric-iBGP-Prozess erfasst wird und **vsh -c** ausgeführt wird **"show ip route x.x.x/y detail vrf <name>"** zeigt einen Nicht-0-Rw-vnid-Wert, dann wird die Route von einem l3out in einem anderen VRF erfasst. Die Validierung ist gleich, unabhängig davon, welches epg der Verbraucher ist und welcher Anbieter.

1. *Identifizieren Sie die Route Map für den Import von Shared Services im internen VRF-BGP-Prozess:*
**show bgp process vrf jy:vrf2 | grep "Import route-map"** ***<—jy:vrf2 ist das interne VRF, an das die Route übertragen wird***

*2. Identifizieren Sie die Präfixliste, die sich in der Routenübersicht für gemeinsam genutzte Services befindet, die mit der gesickerten Route übereinstimmt:*
**moquery -c rtpfxEntry -f'rtpfx.Entry.dn*"pfxlist-IPv4".*'2457603-shared-svc-leck" | egrep "riteria|dn|pfx|toPfxLen"** ***< - 2457603 ist die vnid des internen VRF in diesem Beispiel***

*3. Nachdem Sie ermittelt haben, welche Präfixliste(n) auf die Route verweisen, geben Sie an, welche Weiterleitungssequenznummer auf die Liste(n) verweist:*
**moquery -c rtmapRsRtDstAtt -f'rtmap.RsRtDstAtt.tDn*"pfxlist-IPv4-2949122-24-25-2457603-shared-svc-leak"** ***< - pfxlist-IPv4-2949122-24-25-2457603-shared-svc-lak ist der Name der Präfixliste***

*4. Mithilfe der rtmap und der Eintragsnummer wird der folgende Befehl ausgeführt, um herauszufinden, über welche Vertragsbeziehung dieser route-map-Eintrag gesendet wurde:*
**moquery -c fvAppEpGCons -f 'fv.AppEpGCons.dn*"rtmap-2457603-shared-svc-leck/ent-1001"** ***<—rtmap-2457603-shared-svc-leck/ent-1001 ist der Name der Route-Map und die Eintragsnummer aus Schritt 3.***