新サービスインフラ構想におけるSONiCの活用

Using SONiC for new service infrastructure vision

IIJ
Internet Initiative Japan

Open Networking Conference Japan 2021, sponsored session
October 29, 2021

株式会社インターネットイニシアティブ
沖　勝
m-oki@iij.ad.jp

目次

自己紹介

名前　　沖　勝

所属　　株式会社インターネットイニシアティブ
　　ネットワーク本部 プロダクト開発部

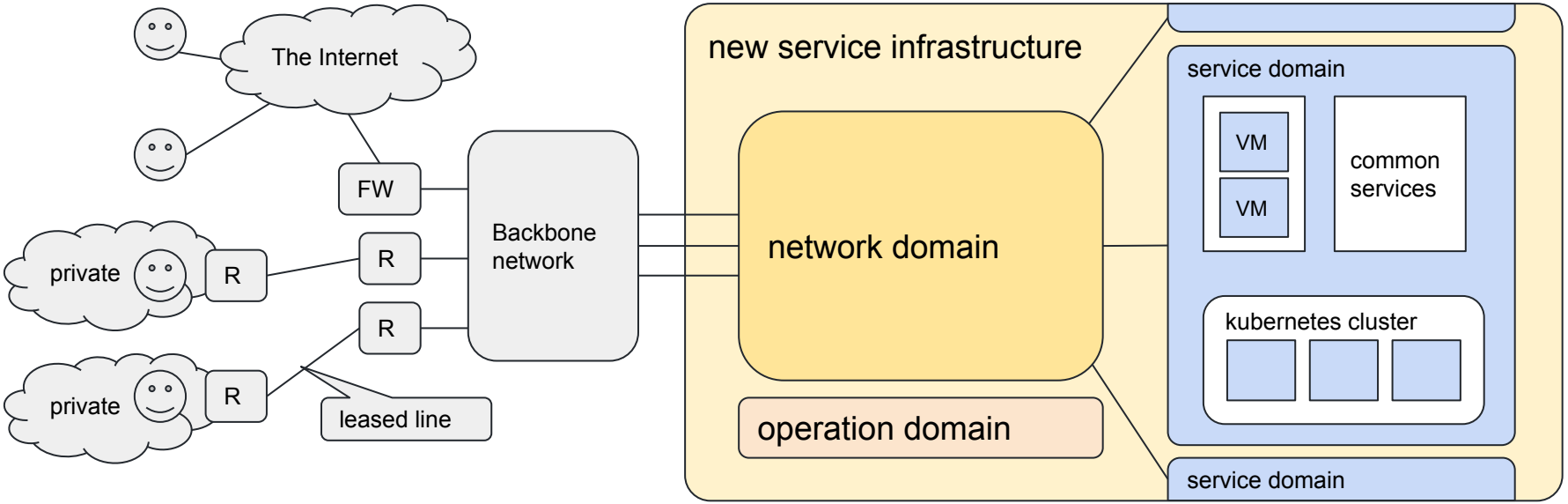
主な活動

- 1992年　LHa for UNIX (移植)
- 1993年　NetBSD/x68k 国産パソコンへのOS移植
- 1996年　日本人初のNetBSD developer
- 2001年～　IIJ SEILシリーズルータ製品のファームウェア開発
- 2013年～　Lagopus OpenFlowソフトウェアスイッチの開発
- 2018年～　新サービスインフラ構想のネットワーク検討

新サービスインフラ構想とSONiC

新サービスインフラ構想

- コンテナ、VMによるハードウェアリソースの有効活用
- スケールアウトできるネットワークアーキテクチャ
- 自動化による遅滞なきサービスデリバリーとオペレーションミスの削減
- サービス開発・運用が本来の業務に集中できる環境を目指す

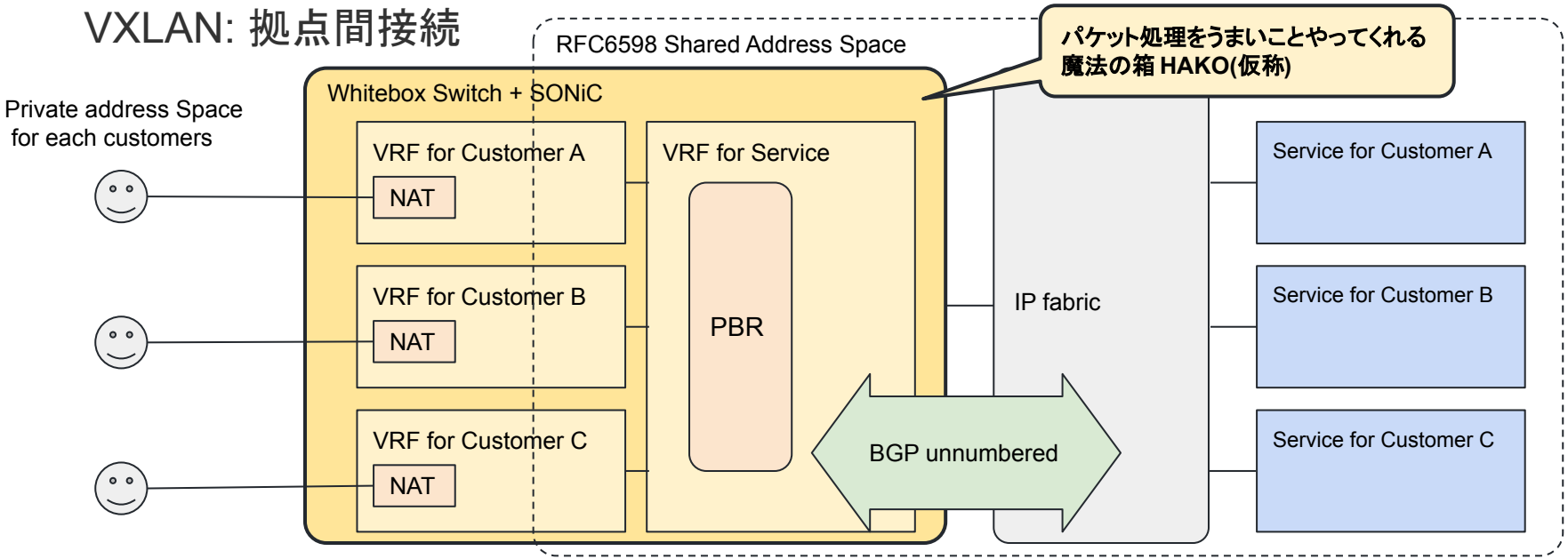


なぜSONiCなのか

- ネットワーク要件
  - ハードウェアのベンダーロックインを避ける
  - OpenFlowのように通常でないトラフィックの捻じ曲げを想定
  - ハードウェアでパケット処理し帯域を確保
- 候補
  - プログラマブルASIC、ホワイトボックススイッチ+NOS
- なぜSONiCか
  - 機能面
    - 動作するスイッチが多い (20ベンダー100モデル以上、対応ASICも多数)
    - 機能をdockerコンテナで分割していて部分更新が可能
    - OSSであり、独自の拡張や修正が必要であれば実装できる
  - 持続性
    - コミュニティが活発、動作実績がある
    - OCP, Microsoft発で開発継続中、EOLになる可能性が低い
    - ベンダーによっては有償サポートが提供されている

どのように活用するのか

VRF: 1000以上を想定 (ASICの上限は0xfff)
NAT: カスタマーにVMやコンテナのIPを知らせず到達性を確保
PBR: カスタマーIPが重複する場合の経路振り分けにsrc ipを用いる
BGP unnumbered: 設定の簡素化とIPv4アドレス消費量の抑制
VXLAN: 拠点間接続



SONiCの機能検証

検証方針

- 単体あるいはシンプルな構成で機能が動作するか検証する。
- 組み合わせ、実運用に近い構成で動作検証する。
- 問題があれば調査し、対処する。
  - 設定見直し、実装の修正、機能追加
  - (SAIの問題であればベンダーに対応をお願いする)
- 仮想環境と実機の両方で確認する。
  - 仮想で動いて実機で動かないケースがある
  - 逆に実機で動いて仮想で動かないケースもある

検証環境

- 最初の検証対象としてBroadcom Trident 3搭載スイッチ(32×100G)を用意。
  - Delta AG9032 V2A
  - DELL EMC S5232F-ON
  - (Edgecore AS7726-32X)
- 仮想マシンを使った環境も用意する。
  - Terraform+libvirtを使用し実機同様の構成をデプロイ
  - VMとVM間ネットワークをローカルマシン上でまとめて構築
- 検証のための設定はAnsibleで実行。
  - CLIを呼び出す。inventoryを切り替えることで仮想/実機の両方に対応

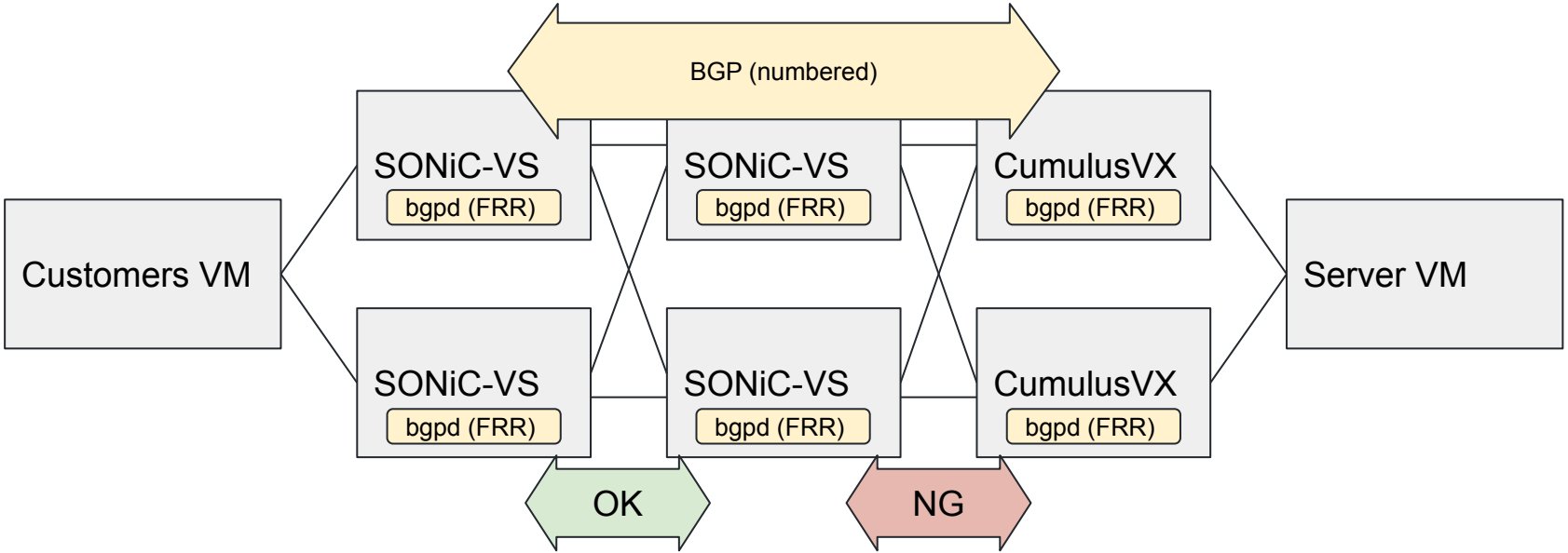
検証中に見つかった問題

問題いろいろ

- 検証してみると問題を発見。いくつかのパターンが見受けられる。
  - ドキュメントがなく使い方が手探り
  - そもそも実装がない
  - 特定条件で動作しない
- 発見した問題をいくつか紹介する。
  - BGPが動かない(仮想環境)
  - VRF+NATが動かない
  - VRF+BGP unnumberedが動かない(実機環境)

BGPが動かない(仮想環境)

- 問題
  - SONiC対向はpeerが張れるが、CumulusVX対向だとpeerが張れない
  - 実機は問題なく、仮想環境でのみ発生
- 構成



BGPが動かない(仮想環境)

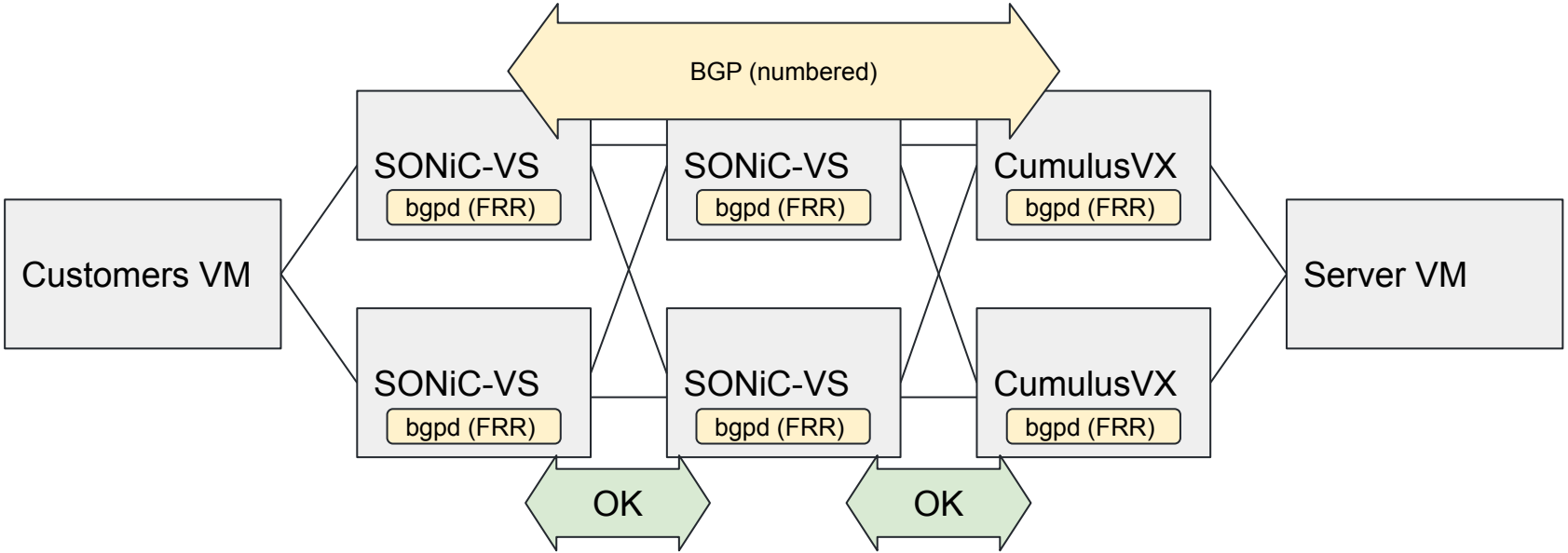
- 調査
  - パケットキャプチャしてみた
  - CumulusVXから送られるパケットの**checksumがincorrect** (!)
- 詳細を調査
  - チェックサム関連トラブルと言えばオフロード設定が定番
  - ethtool -kで設定内容を確認

```
cumulus@cumulus:mgmt:~$ ethtool -k swp1
Features for swp1:
rx-checksumming: on [fixed]
tx-checksumming: on
     tx-checksum-ipv4: off [fixed]
     tx-checksum-ip-generic: on
     tx-checksum-ipv6: off [fixed]
     tx-checksum-fcoe-crc: off [fixed]
     tx-checksum-sctp: off [fixed]
```

疑わしき設定を発見

対処

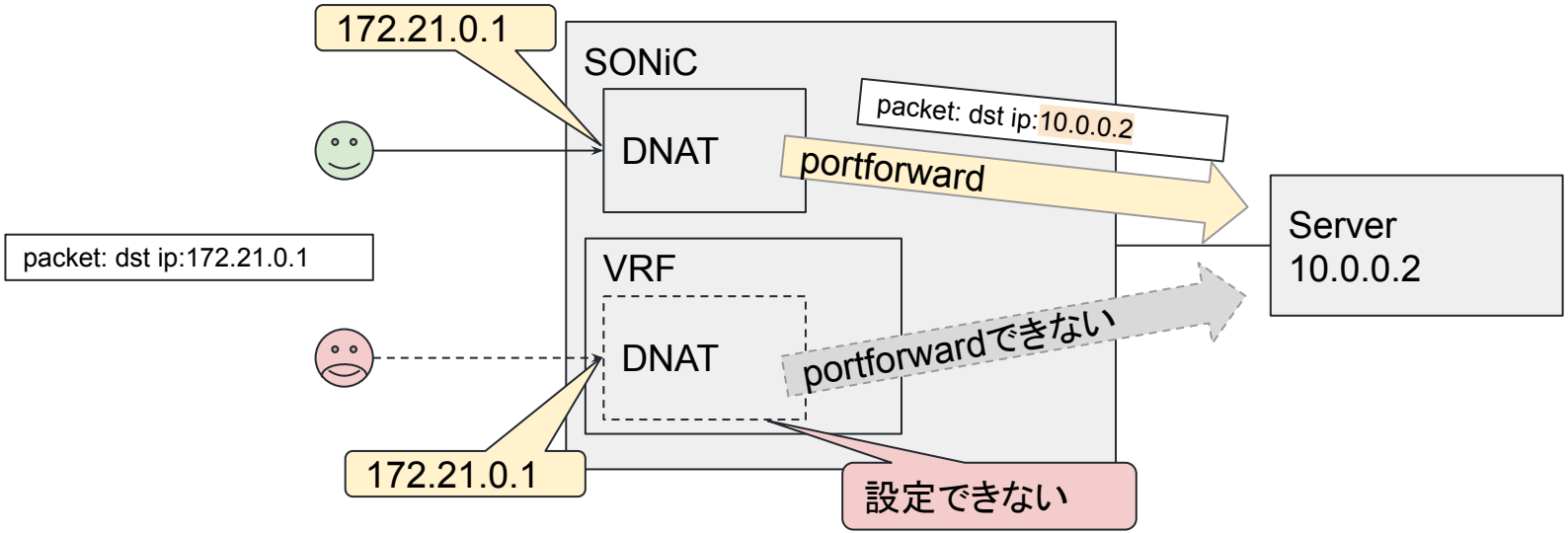
- CumulusVXで`ethtool -K swp1 tx-checksum-ip-generic off`
- 正常動作を確認。
- 備考
  - Ubuntuも同様だった。`virtio-net`の問題と思われる



VRF+NATが動かない

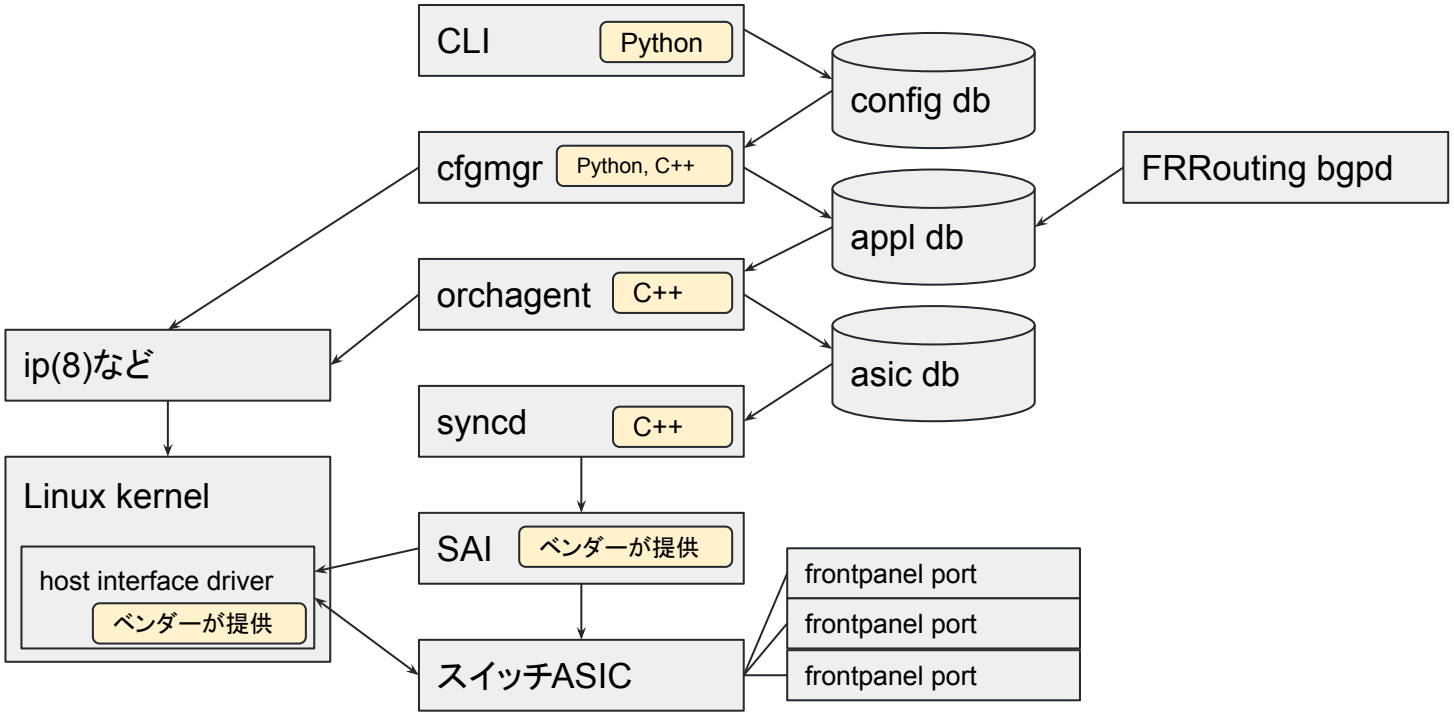
- 問題
  - CLIにVRF指定方法がなく、default VRFでのみ動作する

    ```
    config nat add static basic -nat_type dnat 172.21.0.1 10.0.0.2
    ```
  - なお、SAIやASICのNAT機能はVRFに対応している



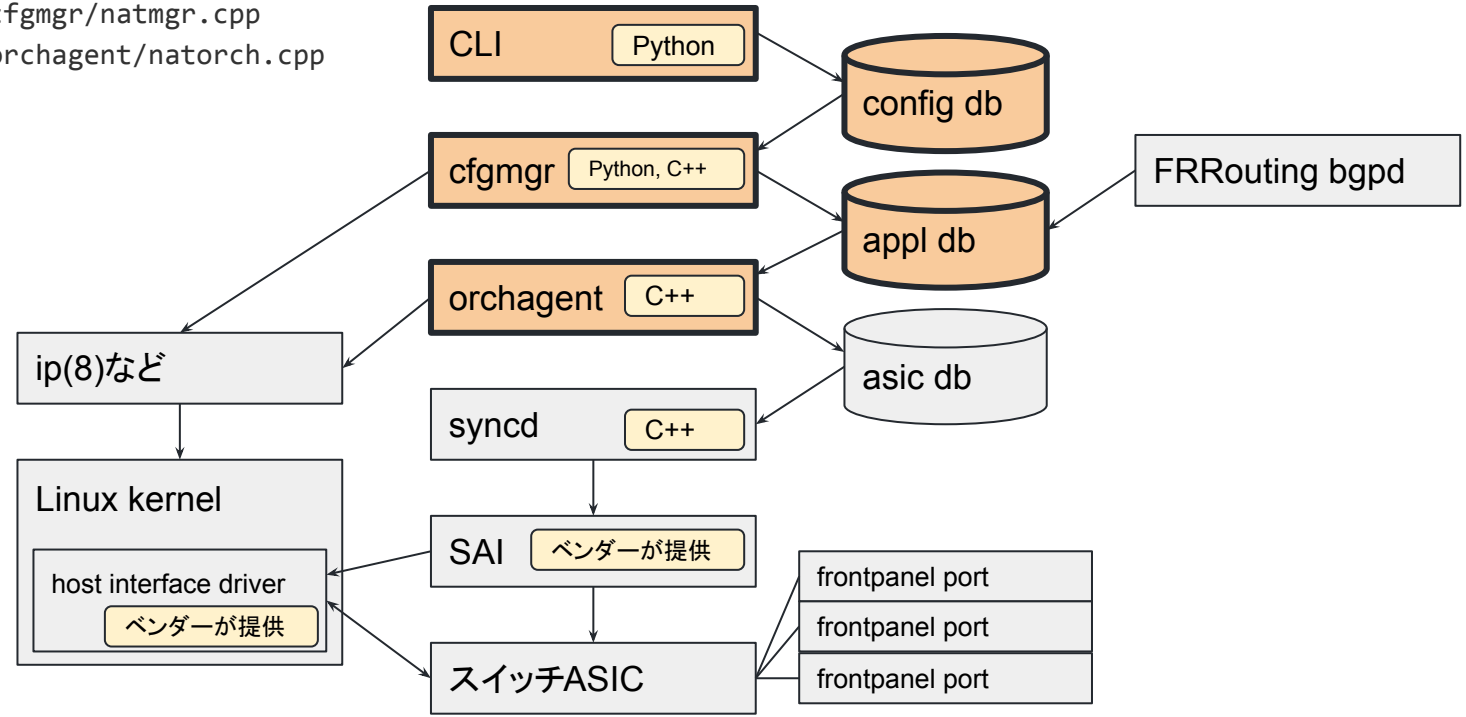
設定追加のため内部処理フローを調査

- SONiCの内部処理フローを調査し、改造に必要な部分を見極める。



改造箇所

DBスキーマにVRFを追加
src/sonic-utilitites/config/nat.py
src/sonic-swss/cfgmgr/natmgr.cpp
src/sonic-swss/orchagent/natorch.cpp



改造後

- 対処
  - VRF指定を可能とした。(指定がなければ従来どおり)

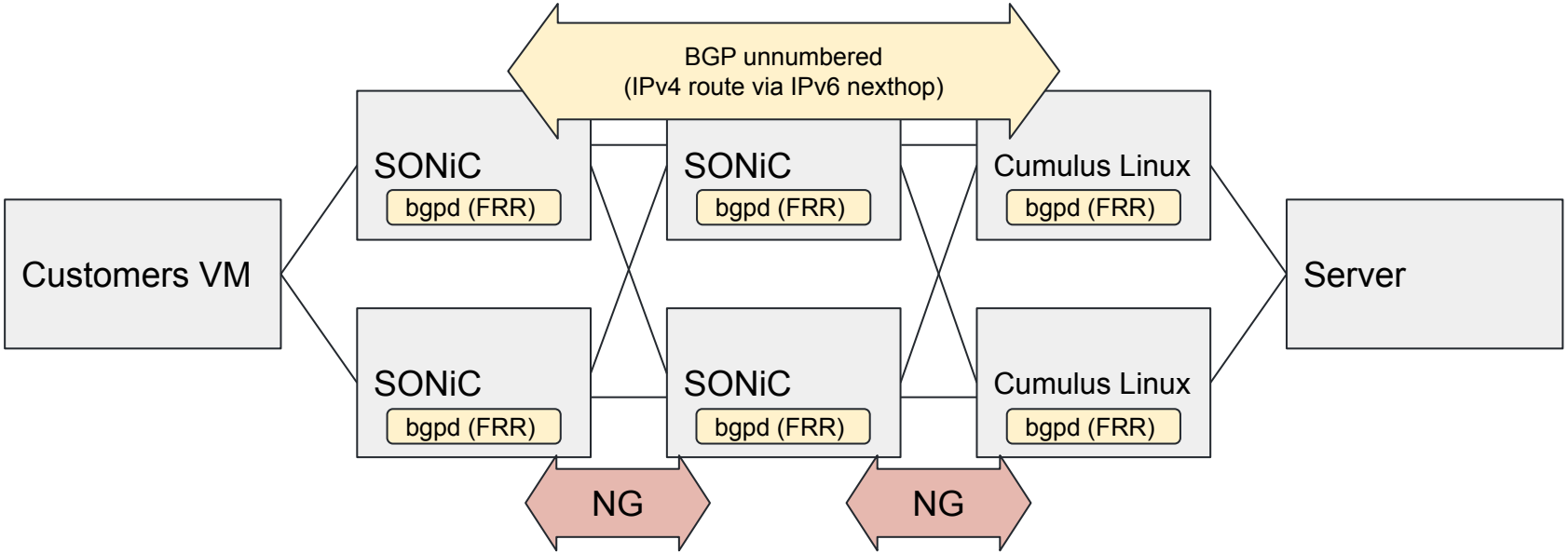
    `config nat add static basic -nat_type dnat -vrf Vrf20 172.21.0.1 10.0.0.2`
  - 参照コマンドも対応した。

```
$ show nat config

Global Values

Admin Mode      : enabled
Global Timeout : 600 secs
TCP Timeout     : 86400 secs
UDP Timeout     : 300 secs
Static Entries

Nat Type    VRF    IP Protocol    Global IP      Global Port    Local IP     Local Port     Twice-NAT Id
----------  -----  -------------  -------------  -------------  -----------  -------------  ---------------
dnat        Vrf10  all            192.168.100.2  ---            172.18.0.34  ---            ---
dnat        Vrf20  all            192.168.100.2  ---            172.18.0.35  ---            ---
```

異なるVRFであれば
同一IPアドレス指定可能

VRF+BGP unnumberedが動かない(実機環境)

- 問題
  - VRFなしだとpeerを張れるが、VRFつきにするとpeerを張れない
  - 仮想環境では問題なく、実機でのみ発生
- 構成



状況確認

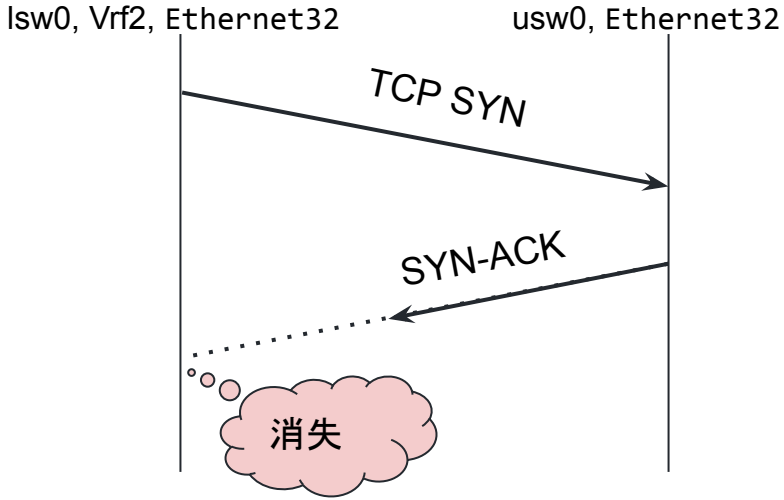
- 状況
  - 仮想環境だと動いている(FRR設定の問題ではない)
  - VRFなしだと動いている(物理の問題ではない)
  - numberedだと動いている(ASICの問題ではない)
- ログ
  - `/var/log/frr/bgpd.log`
    - listen側でacceptされてない
    - connect側はtimeoutでリトライを続けている
  - `show logging`
    - とくにエラーらしきログは出ていない

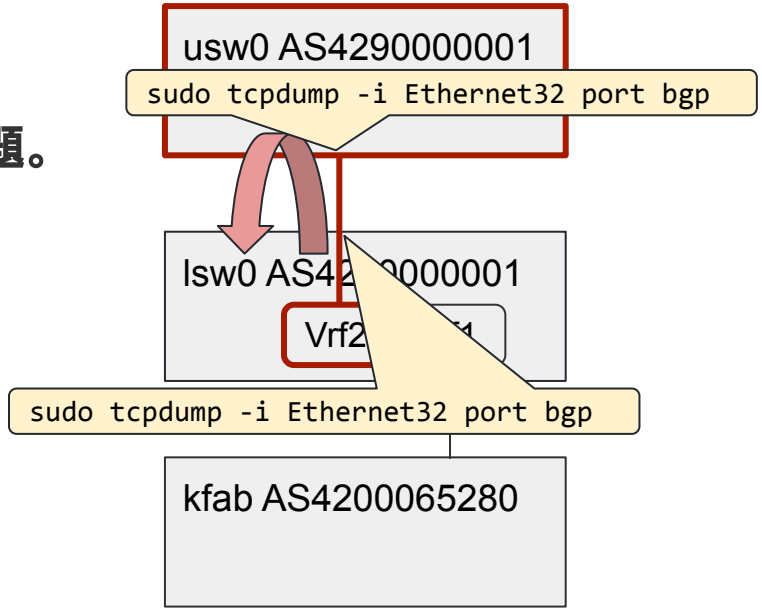
手がかりがほぼない?

- 対処できなかったときの回避策を考えておく。
  - VRFなしでBGP unnumberedを使用
    - 代わりに、マネジメントポートにManagement VRFの設定を入れて分離
    - マネジメントポートを介する通信に`sudo ip vrf exec mgmt`が必要
  - unnumbered使用をあきらめる
    - IPアドレス管理が必要になる

調査続行: パケットキャプチャでTCP接続シーケンスを確認

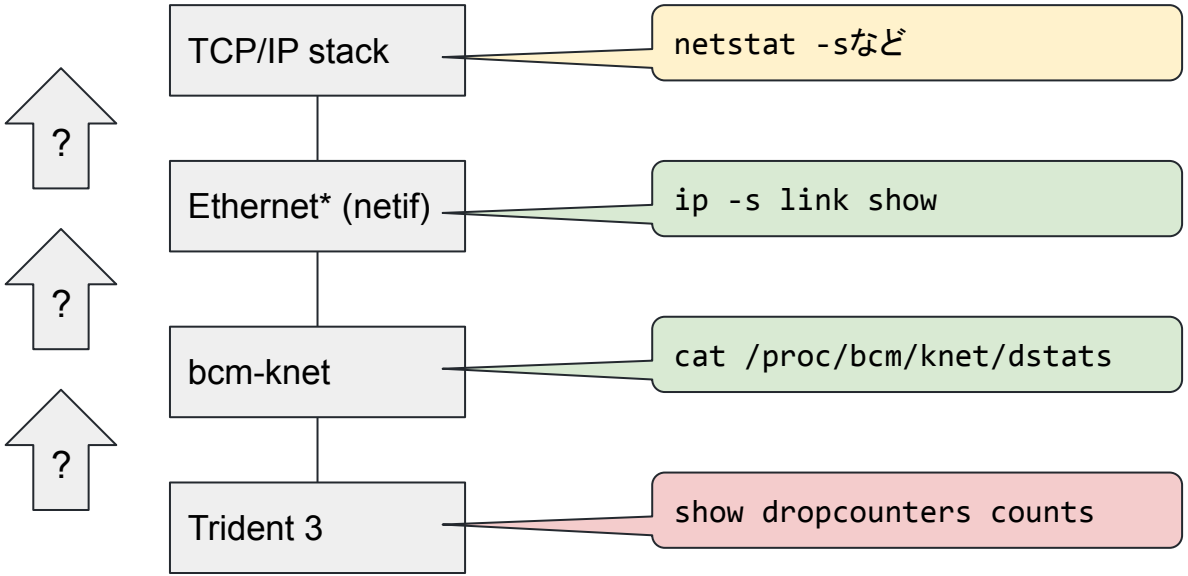
- connect側とlisten側をケーブルで直結。listen側はVRFなしでも接続できず。
- connect側がIPv6 linklocal addressでTCP SYN送信、listen側で受信。
- listen側のSYN-ACK送信をtcpdumpで観測。
- しかし、connect側でtcpdumpしても見えない。
- connect側がVRFなしならつながる。**受信の問題。**





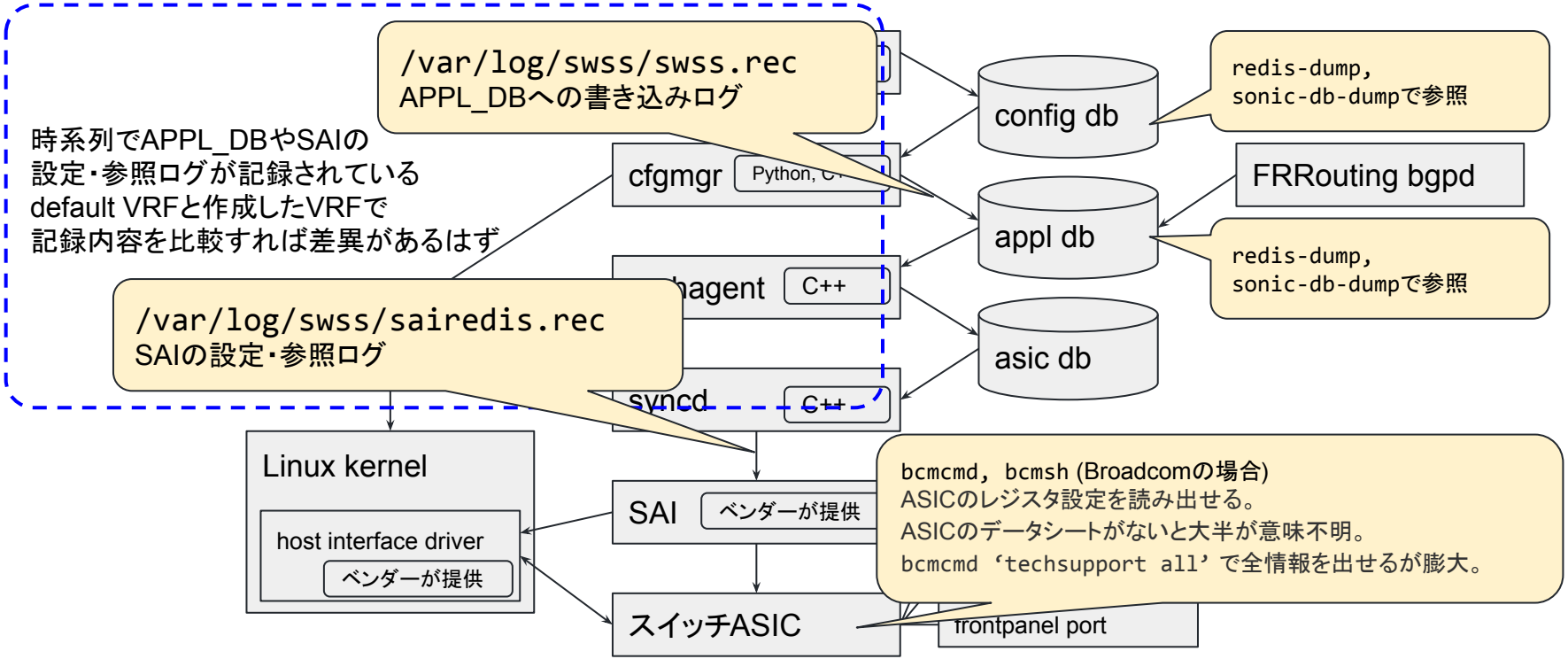
パケット受信時のカウンター確認

- bcm-knet(カーネルモジュール)やnetifではdrop count 0
- `show dropcounters`でRX_DROPSがカウントアップしていた
- ASIC内でdropしていることはわかったが理由が不明



ASICの設定内容を追跡してみる

- ASICへの設定内容の追跡に有用な情報



VRFありとVRFなしのログを比較

- connect側、sairedis.recにて差異を発見。差異のあったログは下記

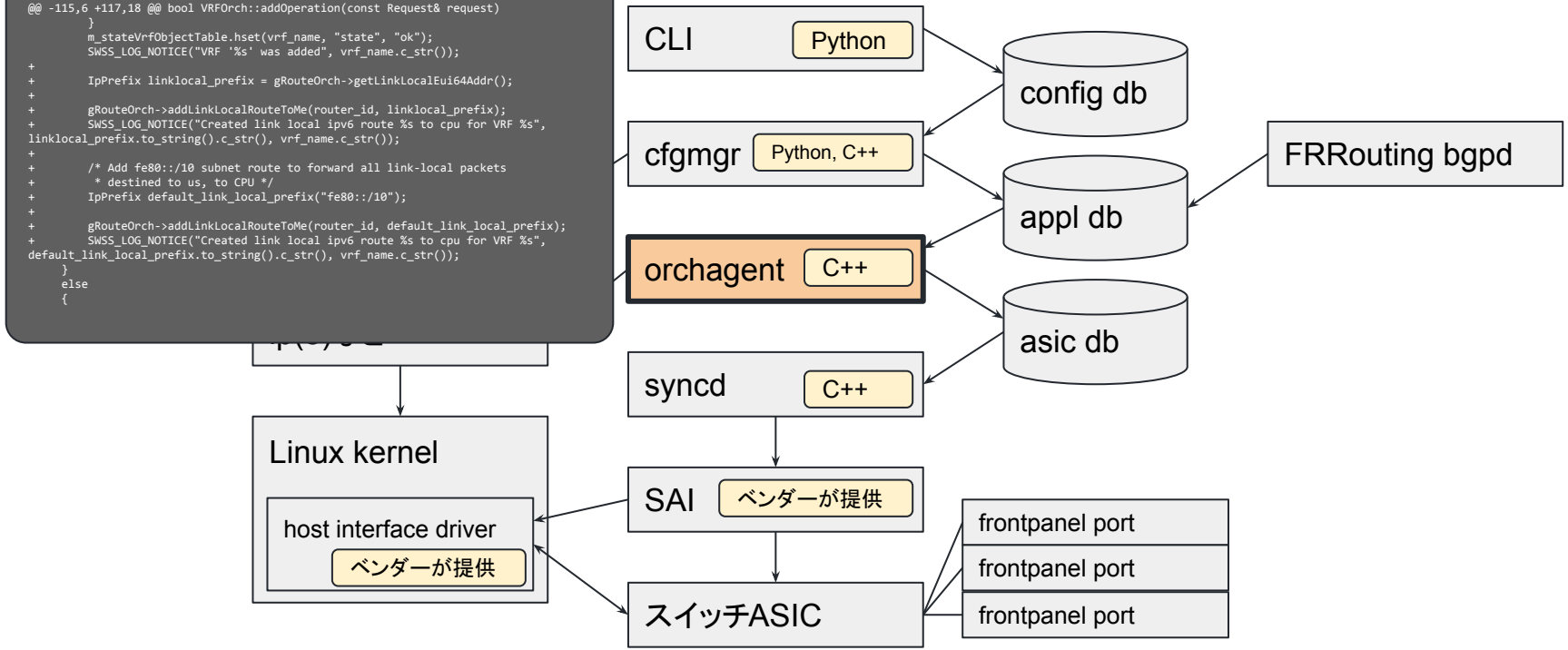
```
2021-09-23.12:02:28.800878|c|SAI_OBJECT_TYPE_ROUTE_ENTRY:{"dest":"fe80::1abe:92
ff:fed1:1f46/128","switch_id":"oid:0x21000000000000","vr":"oid:0x3000000000023"
}|SAI_ROUTE_ENTRY_ATTR_PACKET_ACTION=SAI_PACKET_ACTION_FORWARD|SAI_ROUTE_ENTRY_
ATTR_NEXT_HOP_ID=oid:0x1000000000033
```

- ログの意味
  - 自身のIPv6 link local address宛パケットをCPUにまわす経路を登録
  - oid:0x3000000000023はdefault VRF
  - oid:0x1000000000033はCPU port
- 差異
  - 設定により作成されたVRFへの経路の登録ログがなかった(!)
  - ソースコードを確認すると、登録処理がなかった

改造箇所

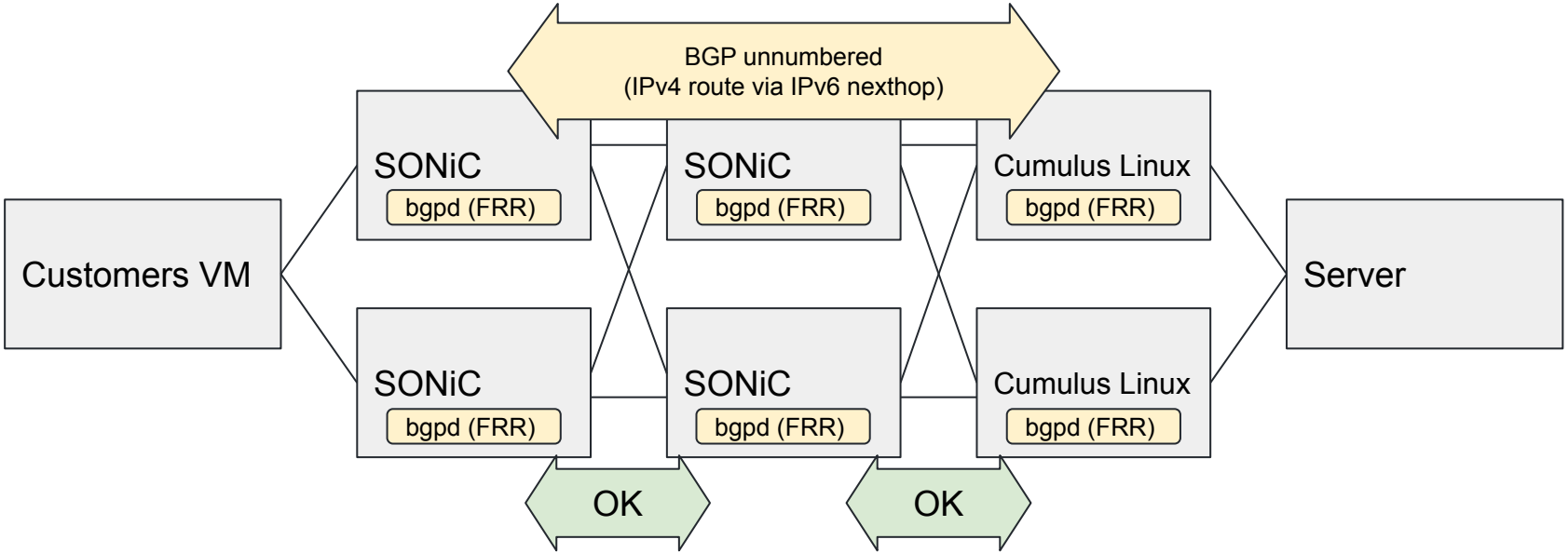
src/sonic-swss/orchagent/vrforch.cpp

```
@@ -115,6 +117,18 @@ bool VRFOrch::addOperation(const Request& request)
         }
         m_stateVrfObjectTable.hset(vrf_name, "state", "ok");
         SWSS_LOG_NOTICE("VRF '%s' was added", vrf_name.c_str());
+
+        IpPrefix linklocal_prefix = gRouteOrch->getLinkLocalEui64Addr();
+
+        gRouteOrch->addLinkLocalRouteToMe(router_id, linklocal_prefix);
+        SWSS_LOG_NOTICE("Created link local ipv6 route %s to cpu for VRF %s",
linklocal_prefix.to_string().c_str(), vrf_name.c_str());
+
+        /* Add fe80::/10 subnet route to forward all link-local packets
+         * destined to us, to CPU */
+        IpPrefix default_link_local_prefix("fe80::/10");
+
+        gRouteOrch->addLinkLocalRouteToMe(router_id, default_link_local_prefix);
+        SWSS_LOG_NOTICE("Created link local ipv6 route %s to cpu for VRF %s",
default_link_local_prefix.to_string().c_str(), vrf_name.c_str());
     }
     else
     {
```



VRF+BGP unnumberedが動かない(実機環境)

- 対処
  - VRF作成時に、IPv6 link local addressの経路登録処理を追加した
  - peerを張ることができるようになった
- フィードバック: issueとして報告、PR済み



今後の課題、まとめ

今後の課題

- 性能検証
- 最大構成検証
- 冗長構成検証
- 異常系動作検証
- 監視、ログ、テレメトリ
- コントロールプレーン

まとめ

- 新サービスインフラ構想にてSONiCを活用しています
- 動作検証を行い、問題が見つかった場合は調査し対処しています
- 構想実現に向けて、今後も様々な検証や実証実験を行います

日本のインターネットは1992年、IIJとともにはじまりました。以来、IIJグループはネットワーク社会の基盤をつくり、技術力でその発展を支えてきました。インターネットの未来を想い、新たなイノベーションに挑戦し続けていく。それは、つねに先駆者としてインターネットの可能性を切り拓いてきたIIJの、これからも変わることのない姿勢です。IIJの真ん中のIはイニシアティブ――IIJはいつもはじまりであり、未来です。