EVPN Anycast Gateway を商用導入した話

2021年7月15日 JANOG48

NTTコミュニケーションズ株式会社

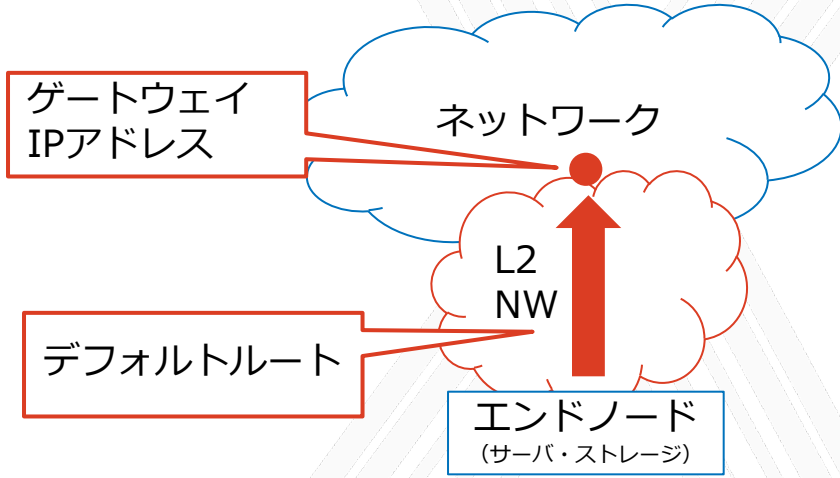
北川 拓

自己紹介

- 北川 拓 (きたがわ たく)
- NTT コミュニケーションズ
- 2017 年入社
- Smart Data Platform (SDPF) / Private Cloud
    - データセンター NW の設計・検証等を担当

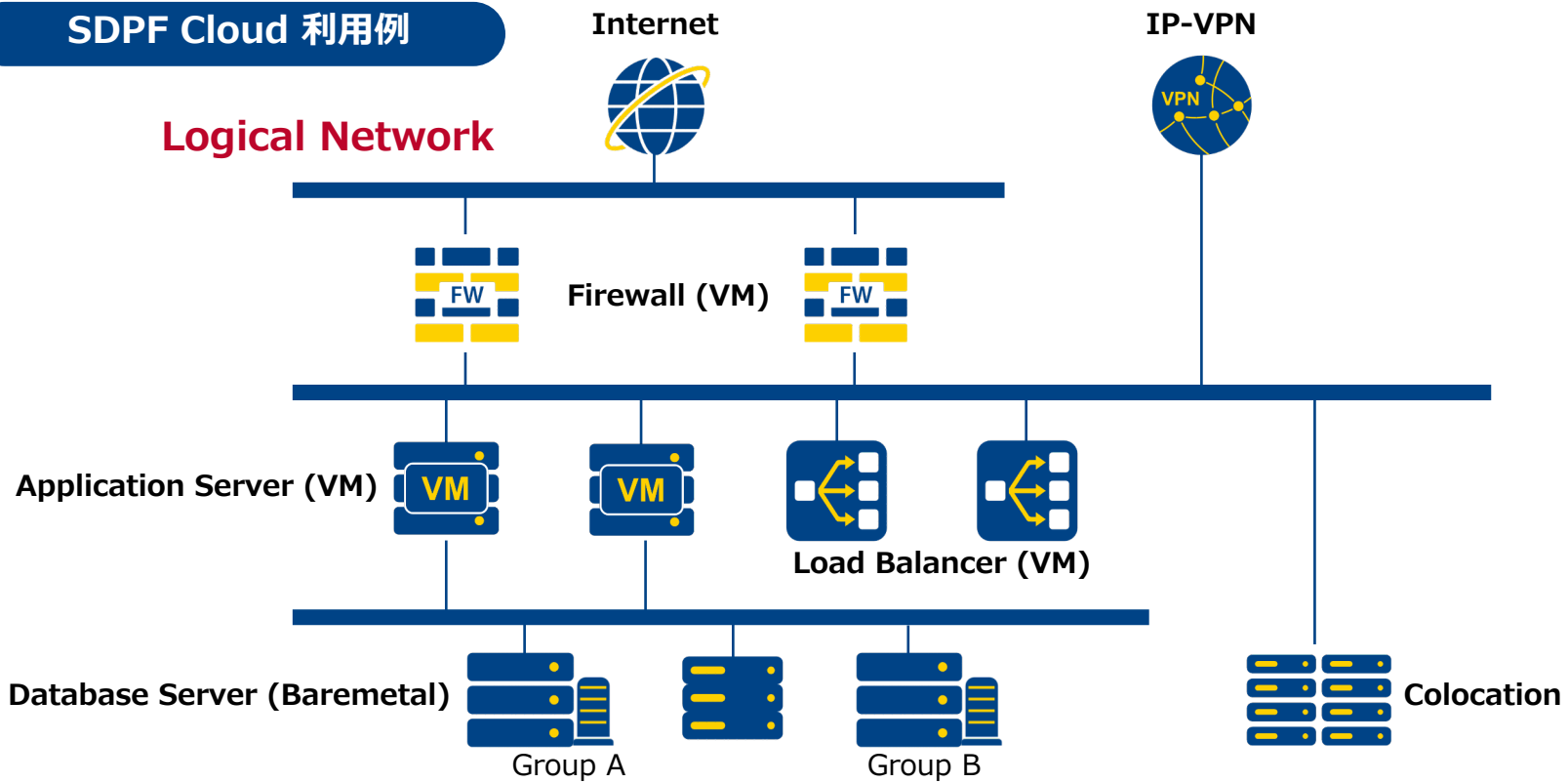
今日のお話

- JANOG47「みんなFirst-Hop Redundancyどうしてるよ？」について
  - 我々は今まで Virtual-Chassis (VC) を使ってきた
  - 現在は EVPN Anycast Gateway (EVPN MH&AG) に注目
- EVPN Anycast Gateway の商用導入向けて
  - 実施した検証のポイント
  - 検証の過程で見つけた問題や対処法
- 全体を通しての気づき
- 議論



SDPF Cloud (旧名: Enterprise Cloud 2.0)

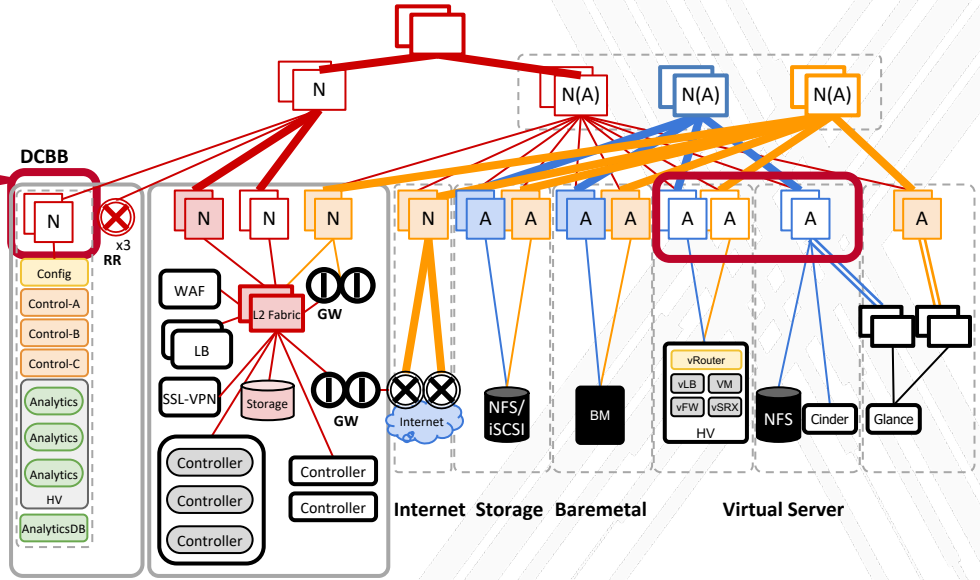
SDPF Cloud 利用例



ネットワーク構成概要

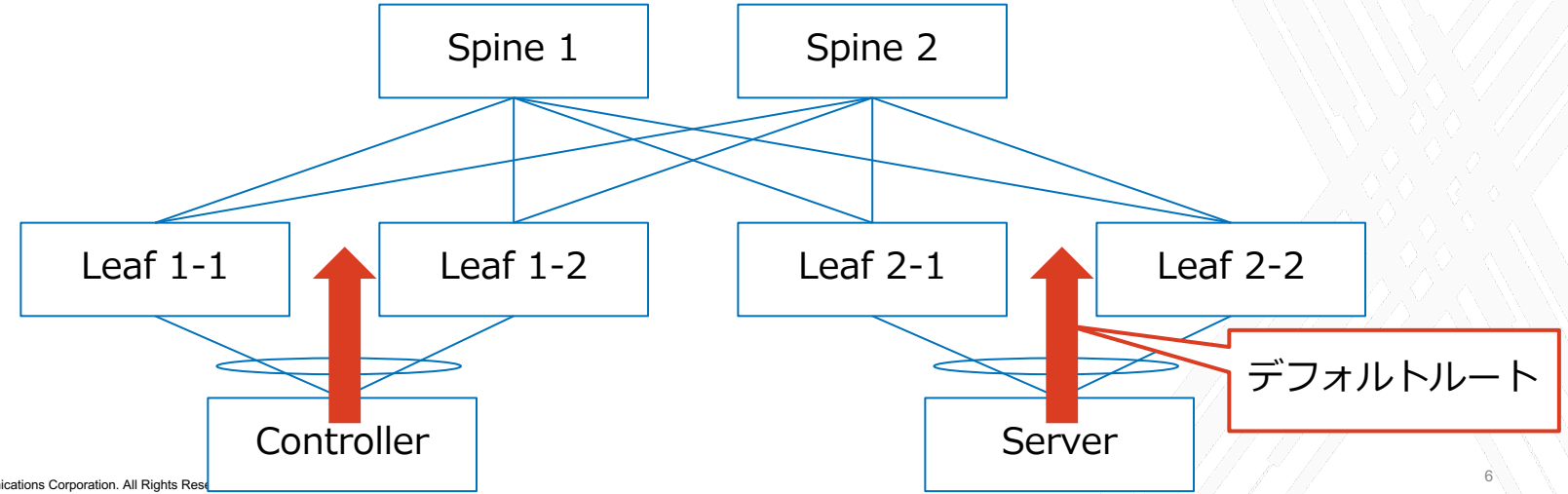
- IP CLOS (Spine/Leaf) 構成
- Leaf は大きく分けて 2 種類
  - テナント通信を終端しエンドノードを L2 で収容する Leaf
  - **HV や VM 用ストレージ、コントローラ等のエンドノードを L3 で収容する Leaf**

今日は HV や VM 用ストレージ、
コントローラ等の **L3 収容 Leaf の話**



First-hop redundancy

- エンドノードがネットワークに接続する最初の 1 歩目の冗長性のこと
- 我々のアーキテクチャでは Controller や Server 等と L3 接続する Leaf でどうデフォルトゲートウェイを冗長させるかという話
- 前回の JANOG では色々と方式を議論



前回のまとめと今日の話

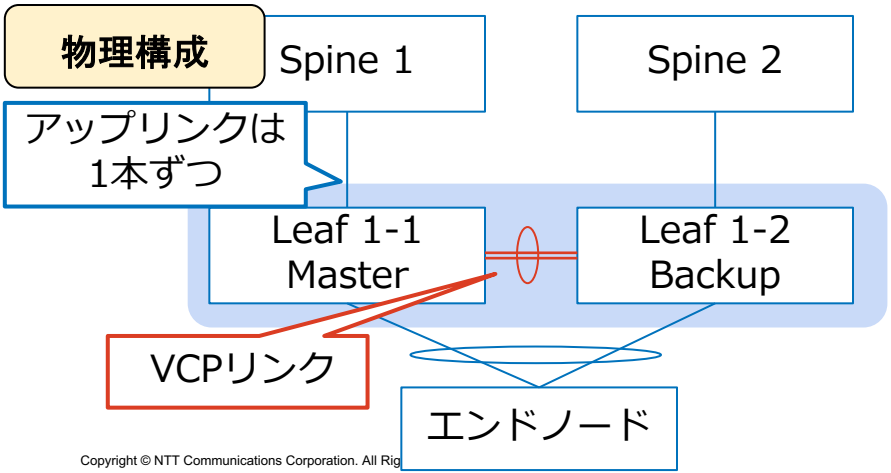
項目	VRRP	VC	MC-LAG	BGP	EVPN MH&AG
仕様	標準技術	プロプライエタリ	プロプライエタリ	標準技術	標準技術
NW機器間の結合度合い	疎結合	超密結合	密結合	結合なし	疎結合
データプレーン	Act-Stb	Act-Act	Act-Act	Act-Act	Act-Act
L2収容	必須	可能	可能	不可能	可能
BUMトラフィック	マルチキャスト	なし	なし	なし	なし
IPアドレス	3個	1個	3個	2サブネット	1個
ハートビート	半分inband	out-of-band	out-of-band	inband	inband
エンドノードに必要な技術	bonding	bonding	LACP	BGP	LCAP
バックボーンのスケーラビリティ	影響なし	影響なし	影響なし	加算される	影響なし
最大罹障範囲	収容機器	収容機器	収容機器	リンク	Spine配下
スプリットブレイン	起きる	起きる	起きるけど救える	起きない	起きない

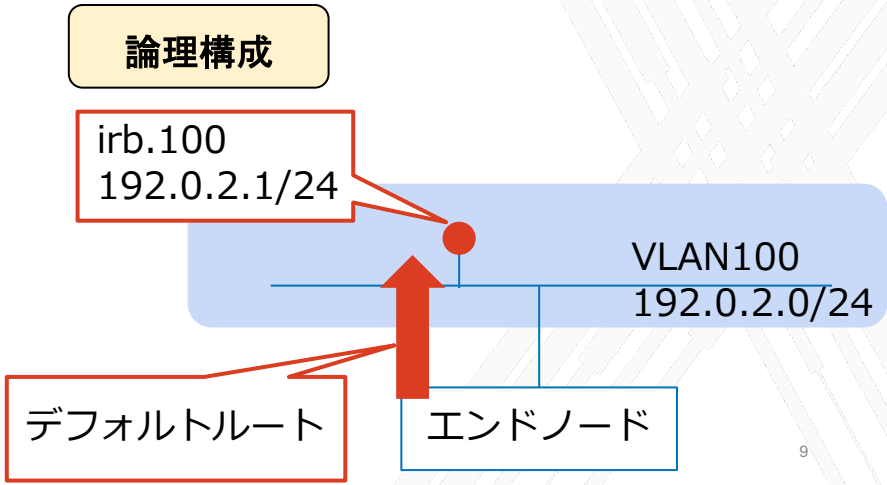
前回のまとめと今日の話

項目	VRRP	VC	MC-LAG	BGP	EVPN MH&AG
仕様	標準技術	プロプライエタリ	プロプライエタリ	標準技術	標準技術
NW機器間の結合度合い	疎結合	超密結合	密結合	結合なし	疎結合
データプレーン	Act-Stb	Act-Act	Act-Act	Act-Act	Act-Act
L2収容	必須	可能	可能	不可能	可能
BUMトラフィック	マルチキャスト	なし	なし	なし	なし
IPアドレス	3個	1個	3個	2サブネット	1個
ハートビート	半分inband	out-of-band	out-of-band	inband	inband
エンドノードに必要な技術	bonding	bonding	LACP	BGP	LCAP
バックボーンのスケーラビリティ	影響なし	影響なし	影響なし	加算される	影響なし
最大罹障範囲	収容機器	収容機器	収容機器	リンク	Spine配下
スプリットブレイン	起きる	起きる	起きるけど救える	起きない	起きない

VC (Virtual-Chassis)

- 我々がこれまで使ってきた方式
- 物理 Leaf 2 台でクラスタを組み**論理的に１つの SW** に見せている (Act/Sby)
- LAG で L2 冗長 + L3 I/F を GW として設置





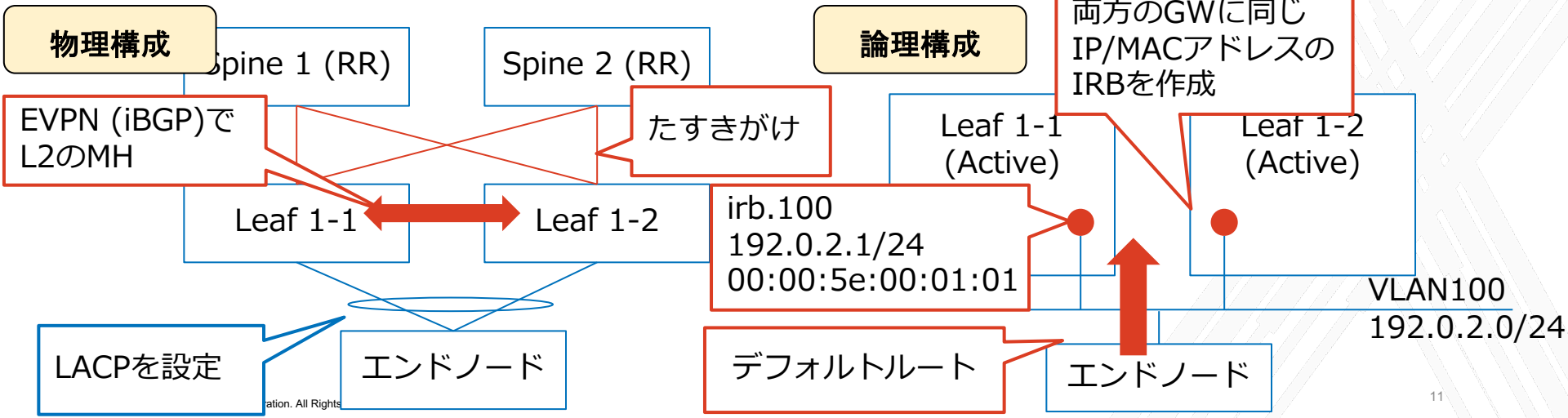
VC (Virtual-Chassis) で苦しんだこと

- 論理的に１つに見せる仕組み
  - **論理的な単一障害点になり得る**
  - クラスタは複雑

- サービス運用する中で苦しんだこと
  - **複雑なのでバグが多く影響も大きい**
    - Master 切り替わり時に状態の引き継ぎがうまくいかない等
    - 密結合の結果、片方しぬともう片方しぬ
  - **In service でメジャーバージョンアップ出来ないケースがある**
    - NSSU (Nonstop software upgrade) は複雑で難しい

EVPN MH (Multi-homing) + AG (Anycast Gateway)

- EVPN Multi-homing (MH) 機能を使って LAG で L2 冗長
- EVPN Anycast Gateway (AG) を使って Act/Act の L3GW を実現
- 直上の Spine を Route Reflector として iBGP を張り EVPN 経路を交換
- Leaf に L3 I/F を設置（EVPN Domain は Leaf のみ = オーバレイはしない)



(参考) EVPN MH + AG 設定例 (1/3)

```
set interfaces xe-0/0/0 description endnode-1_eno1
set interfaces xe-0/0/0 ether-options 802.3ad ae0

set interfaces ae0 description endnode-1_bond0
set interfaces ae0 mtu 9216
set interfaces ae0 esi auto-derive lacp
set interfaces ae0 esi all-active
set interfaces ae0 aggregated-ether-options minimum-links 1
set interfaces ae0 aggregated-ether-options local-bias
set interfaces ae0 aggregated-ether-options lacp active
set interfaces ae0 aggregated-ether-options lacp periodic slow
set interfaces ae0 aggregated-ether-options lacp system-priority 100
set interfaces ae0 aggregated-ether-options lacp system-id 02:0a:0:0:01:00
set interfaces ae0 aggregated-ether-options lacp admin-key 1
set interfaces ae0 unit 0 family ethernet-switching interface-mode trunk
set interfaces ae0 unit 0 family ethernet-switching vlan members VLAN100

set interfaces irb gratuitous-arp-reply
set interfaces irb unit 100 description VLAN100
set interfaces irb unit 100 family inet address 192.0.2.1/24:
set interfaces irb unit 100 mac 00:00:5e:00:01:01

set interfaces lo0 unit 0 family inet address 10.0.0.1/32
```

ESIで識別されるのでAE番号は何番でもいい

ESIはLACPのsystem-idを使う

LACPのsystem-idはESIに使うので、
{ペア,AE}ごとに固有の値にする。
この例では若番のLeafのLoopbackアドレスと
AE番号を16進数で埋め込んでいる

IRBにゲートウェイになるIPアドレスとMACアドレスを設定する。MACアドレスは他のVLANのIRBと重複しても良いので、すべてのペア、IRBで同じ値を使っている

Leafのペアごとに固有の値を設定する項目
ペア内の両方のLeafで同じ値を設定する項目
ペア内のそれぞれのLeafで違う値でもいい項目
ペア内のそれぞれのLeafで固有の値を設定する項目

(参考) EVPN MH + AG 設定例 (2/3)

```
set routing-options router-id 10.0.0.1

set protocols bgp group UNDERLAY_AG type internal
set protocols bgp group UNDERLAY_AG local-address 10.102.4.1
set protocols bgp group UNDERLAY_AG hold-time 40
set protocols bgp group UNDERLAY_AG family evpn signaling
set protocols bgp group UNDERLAY_AG local-as 65000
set protocols bgp group UNDERLAY_AG neighbor 10.0.0.11
set protocols bgp group UNDERLAY_AG neighbor 10.0.0.12

set protocols evpn encapsulation vxlan
set protocols evpn multicast-mode ingress-replication
set protocols evpn default-gateway do-not-advertise
set protocols evpn extended-vni-list all

set switch-options vtep-source-interface lo0.0

set vlans VLAN100 vlan-id 100
set vlans VLAN100 l3-interface irb.100
set vlans VLAN100 vxlan vni 100
```

ルートリフレクタのSpineとEVPNのiBGPを張る設定
Loopback IP でピアを張るので hold timer を設定

EVPN/VXLAN周りの設定あれこれ
LeafをGWにするのでdefault-gatewayはadvertiseしない

VLAN100のIRBを作成してVNI 100に紐付ける

Leafのペアごとに固有の値を設定する項目
ペア内の両方のLeafで同じ値を設定する項目
ペア内のそれぞれのLeafで違う値でもいい項目
ペア内のそれぞれのLeafで固有の値を設定する項目

(参考) EVPN MH + AG 設定例 (3/3)

```
set policy-options policy-statement IMPORT-EVPN term ACCEPT-ESI-IN from community COMMUNITY-EVPN-IN
set policy-options policy-statement IMPORT-EVPN term ACCEPT-ESI-IN then accept
set policy-options policy-statement IMPORT-EVPN term REJECT then reject

set policy-options community COMMUNITY-EVPN-IN members target:10.0.0.1:65000

set protocols evpn vni-options vni 100 vrf-target target:10.0.0.1:65000

set switch-options route-distinguisher 10.0.0.1:65000
set switch-options vrf-import IMPORT-EVPN
set switch-options vrf-target target:10.0.0.1:65000
```

RTはペアごとに設定する
この例では若番のLeafの
Loopbackアドレスを使用

RDは筐体ごとに設定する

Leafのペアごとに固有の値を設定する項目
ペア内の両方のLeafで同じ値を設定する項目
ペア内のそれぞれのLeafで違う値でもいい項目
ペア内のそれぞれのLeafで固有の値を設定する項目

EVPN MH + AG で何が嬉しいか

- C-Plane が標準技術で構成されている
  - ○ ナレッジが多い & 中身が分かる
  - ○ シンプルなので不具合は少なくなりそう
- 論理冗長が取れている
  - ○ 片方で不具合が起きてももう片方に伝搬しにくい（両系断になりにくい）
  - ○ 迂回できる & 異バージョンで冗長を組めるのでバージョンアップも簡単
- エンドノードの NW 要件は LAG が組めることくらい
  - ○ 既存のエンドノードに対し NW 要件を変更せずに導入可

商用導入に向けた検証

- 商用導入に向けて EVPN MH + AG を検証したので内容と結果を共有

検証のポイント

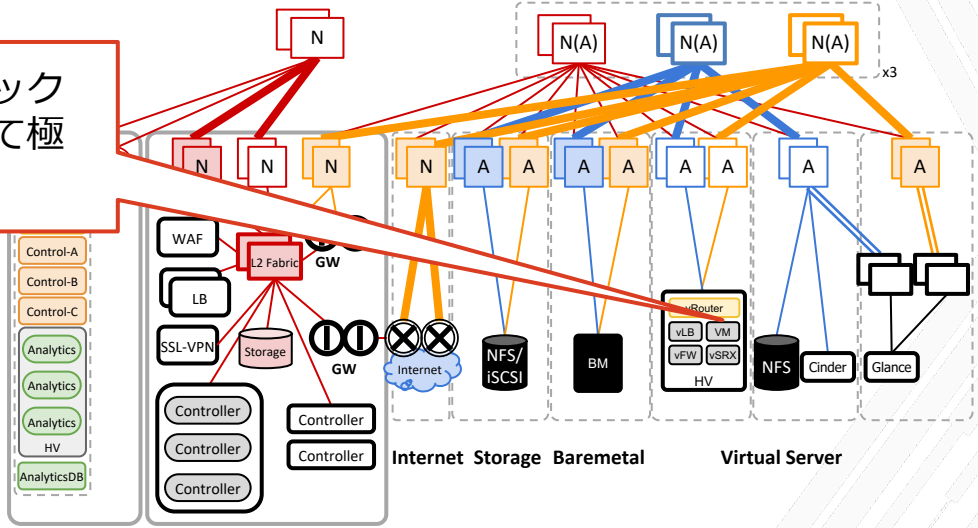
- 我々の過去の経験から意識したポイントは 3 つ
  - 【ポイント①】**幅広く**商用と同等の環境でみる
  - 【ポイント②】**集中的に**負荷をかけてみる
  - 【ポイント③】**繰り返し**何度もみる

- 重視したのは**非機能試験 & Version Up 試験**
  - 故障発生時にちゃんと切り替わってくれるかが大事
  - Version Up が簡単 & 安心にできるようになるのが最大のメリット

【ポイント①】幅広くみる

- エンドノード差分による問題に気づけるように網羅的にみる
- 商用に展開する全種別のエンドノードを総合試験環境に構築
- 全種別のエンドノードに対し背景トラフィックを流しながら試験を実施
  - ○ エンドノードのアプリ通信も流す (例: HV 上の VM 通信等)

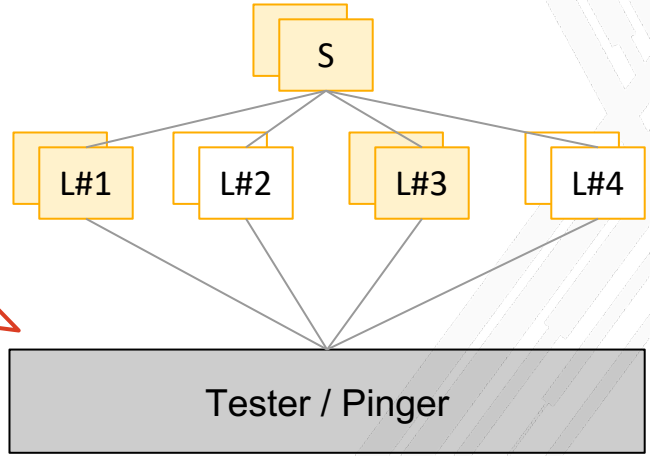
全リソースに対してトラフィックを流す。アプリの通信も流して極力商用と近づける



【ポイント②】集中的にみる

- 負荷をかけていじめることによって出現する不具合を見つける
- NW 試験専用の何でもできる TestPod という環境を作成
- Tester を用いて様々な負荷をかけたり異常状態を作ったりする
  - ○ VLAN/経路数負荷、経路更新負荷、Uplink 全断、ループ等…

Tester と Pinger を使って C/D-Plane の負荷を与える（BGP / MAC / VTEP / 様々なパターンの経路やフロー、Unicast/BUM 等）



【ポイント③】繰り返しみる

- 回数をこなすことによってレアケースの不具合を見つける
- 自動化で時間・稼働を節約して回数をこなす
  - ○ NW 検証自動化フレームワークの RENAT で試験自動化 (*)
  - ○ 断時間の測定もツールで自動化
  - ○ Tester 等の配線変更をロボット (ROME) で実施
- 延べ数千項目ほどの試験実施

```
32. IF shut - underlay leaf downlink tester for lb5bx-tpdl2103n
  ${logfile}=  START_PING_TOOL  cb0nv-ntnt0001n_vm_root_nt  QFX_MAX_01_RETRY_08_32  /root/kiban-nw-tools/jibeta_ping/ping_list/lb5/testpod_qfx.json
  START_TESTCENTER_SERVER  cb0nv-ntnt0001n_vm_root_nt  ./QFX5120_Testpod_D-Plane/QFX5120_Testpod_D-Plane.tcc  QFX_MAX_01_RETRY_08_32  11081
  START_TESTCENTER_TRAFFIC  cb0nv-ntnt0001n_vm_root_nt  QFX_MAX_01_RETRY_08_32  11081
  QFX5120_IFSHUT_NOSHUT  lb5bx-tpdl2103n  xe-0/0/1
  STOP_TESTCENTER_TRAFFIC  cb0nv-ntnt0001n_vm_root_nt  QFX_MAX_01_RETRY_08_32  11081
  STOP_TESTCENTER_SERVER  cb0nv-ntnt0001n_vm_root_nt  QFX_MAX_01_RETRY_08_32  11081
  STOP_PING_TOOL  cb0nv-ntnt0001n_vm_root_nt  QFX_MAX_01_RETRY_08_32  /root/kiban-nw-tools/jibeta_ping/ping_list/lb5/testpod_qfx.json  ${logfile}

33. IF shut - underlay leaf downlink pinger for lb5bx-tpdl2103n
  ${logfile}=  START_PING_TOOL  cb0nv-ntnt0001n_vm_root_nt  QFX_MAX_01_RETRY_08_33  /root/kiban-nw-tools/jibeta_ping/ping_list/lb5/testpod_qfx.json
  START_TESTCENTER_SERVER  cb0nv-ntnt0001n_vm_root_nt  ./QFX5120_Testpod_D-Plane/QFX5120_Testpod_D-Plane.tcc  QFX_MAX_01_RETRY_08_33  11081
  START_TESTCENTER_TRAFFIC  cb0nv-ntnt0001n_vm_root_nt  QFX_MAX_01_RETRY_08_33  11081
  QFX5120_IFSHUT_NOSHUT  lb5bx-tpdl2103n  xe-0/0/2
```

RENAT シナリオ（Tester/Pinger の断時間測定を含めた非機能試験のシナリオ）

弊社 Lab に導入した配線変更ロボット ROME

(*) RENAT(Robotframework Extension for Network AutomationTesting)を用いたネットワーク検証の自動化について, JANOG 41, https://www.janog.gr.jp/meeting/janog41/program/sp4renat

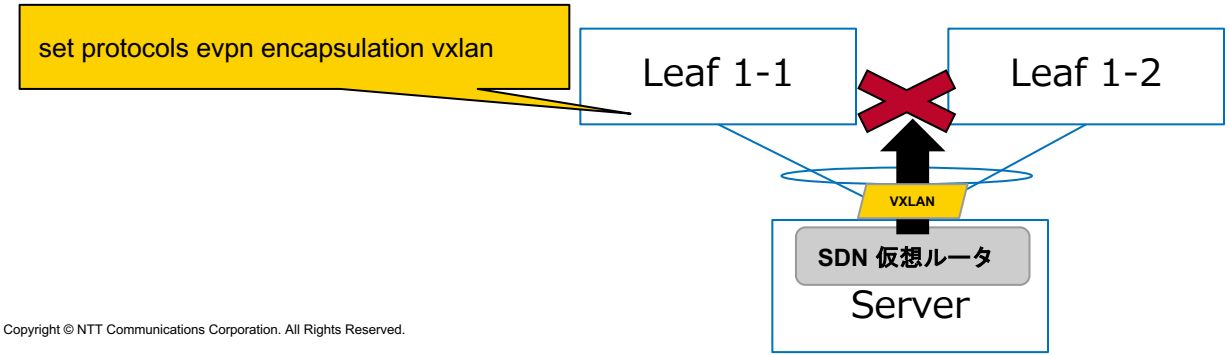
検証結果

- 切り替わり時間 0.1s 程度
- 片系故障時にもう片系に影響なし
- 片系の Uplink 全断しても core-isolation がちゃんと動いて正常に切り替わり
- Major Version Up 実施可能、不具合もなし
  - ○ 異バージョンでも C-Plane (BGP/EVPN) は問題なく動作

結論: EVPN MH + AG いい感じ。でもいくつか不具合 / 問題もみつかった。

VXLAN パケットを落とす不具合 (1/2)

- Downlink から受け取った VXLAN パケットを落としてしまいテナント通信が全断（Server 上の SDN 仮想ルータはテナント通信を VXLAN で Encap するため）
- EVPN MH+AG はペア内の L2 Overlay 通信に VXLAN を使用する設定
- VXLAN で Encap されたパケットを受信すると、VXLAN として Next-hop を Look up してしまい解決できず Drop する不具合 (現在は修正済み)

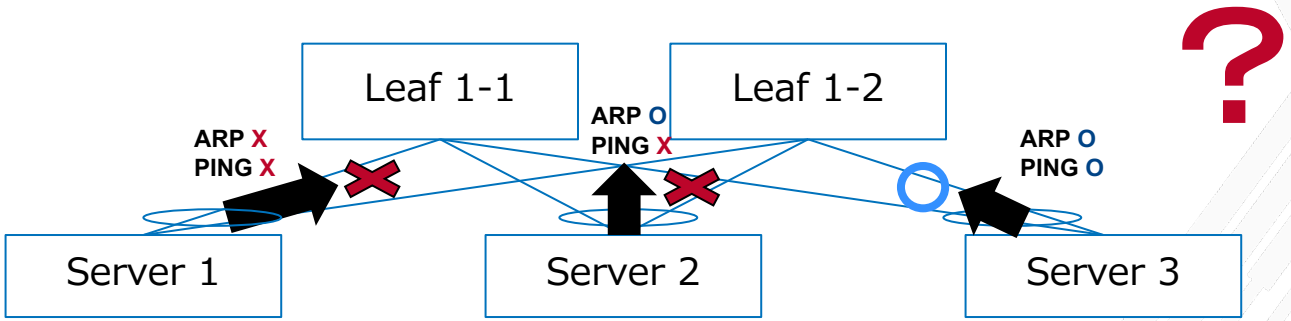


VXLAN パケットを落とす不具合 (2/2)

- 検証で気づいて直すことができた
  - ○ 商用同等環境で実際のユーザ通信 (VXLAN 通信) を流していたから
  - ○ Tester で IP トラフィックだけ流していても気づけない
  - ○ 基盤だけではなくアプリまで踏まえた幅広い動作確認が重要
- **幅広く色んなパターンをみることがやはり大事**

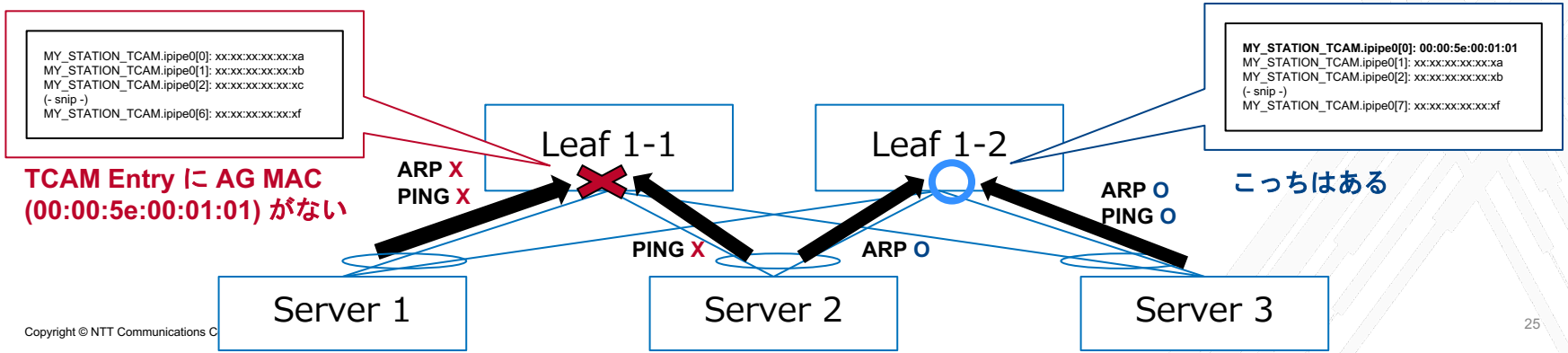
AG の MAC が消える不具合 (1/2)

- ある日突然 Server の通信が通ったり通らなかったりと不安定になった
- 特定の Leaf 収容のみで起きる
- OK な Server もいれば NG な Server もいる
- NG な Server も ARP がだめだったり Ping だけだめだったり色々
- **何が悪いか切り分けに時間がかかった**



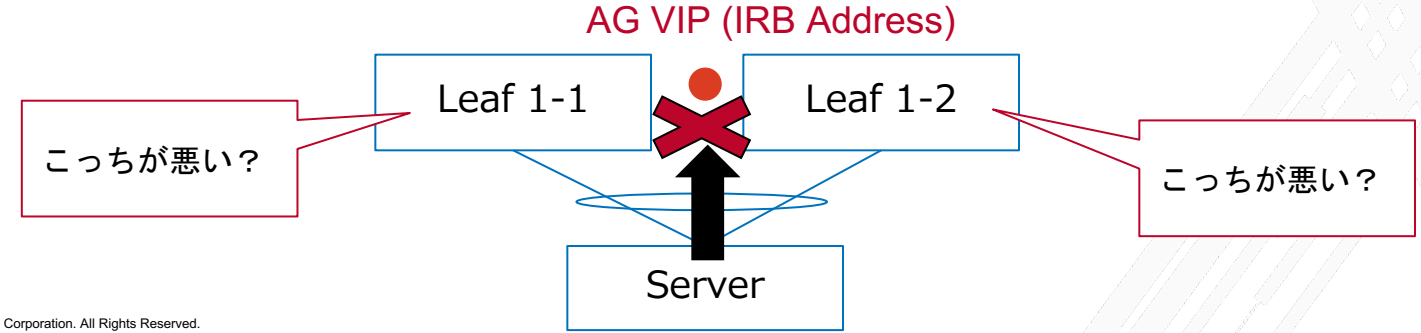
AG の MAC が消える不具合 (2/2)

- 答え: 1 系のみ AG の MAC がチップから消えていた（現在は修正済み）
- LAG のハッシュで 1 系に AG MAC 宛てのパケットが来るとドロップ
- **MAC 経路の更新が頻繁に起こると稀に生じる不具合**であった
  - ○ MAC 更新や VTEP Flap のような更新負荷を沢山かけるのがよさそう
- **集中的に負荷をかける + 繰り返し試験することもやはり大事**



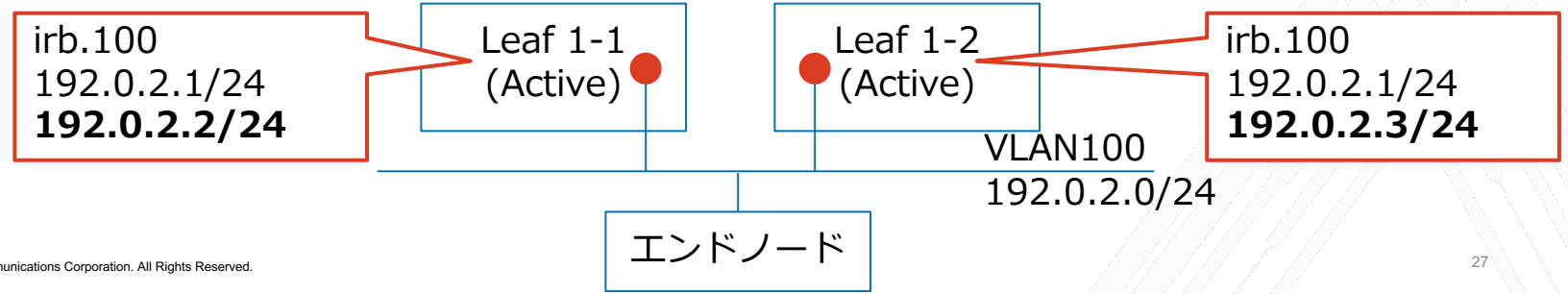
AG MAC が消える不具合をもう少し深堀り

- 商用導入を見据えて切り分けに時間がかかった問題を解きたい
- 今回のように AG VIP/MAC の通信が NG になったらどうすればよいか
- **AG VIP だけの監視だと NG には気づけても 1 系 or 2 系どっちが悪いか不明**
  - ○ **Anycast Gateway 方式の本質的な問題といえる (識別子が一つしかない)**
  - ○ 悪い方 (1 系 or 2 系) をスパッと落とすことができれば迅速に復旧対処可能
- 監視案を考えてみた



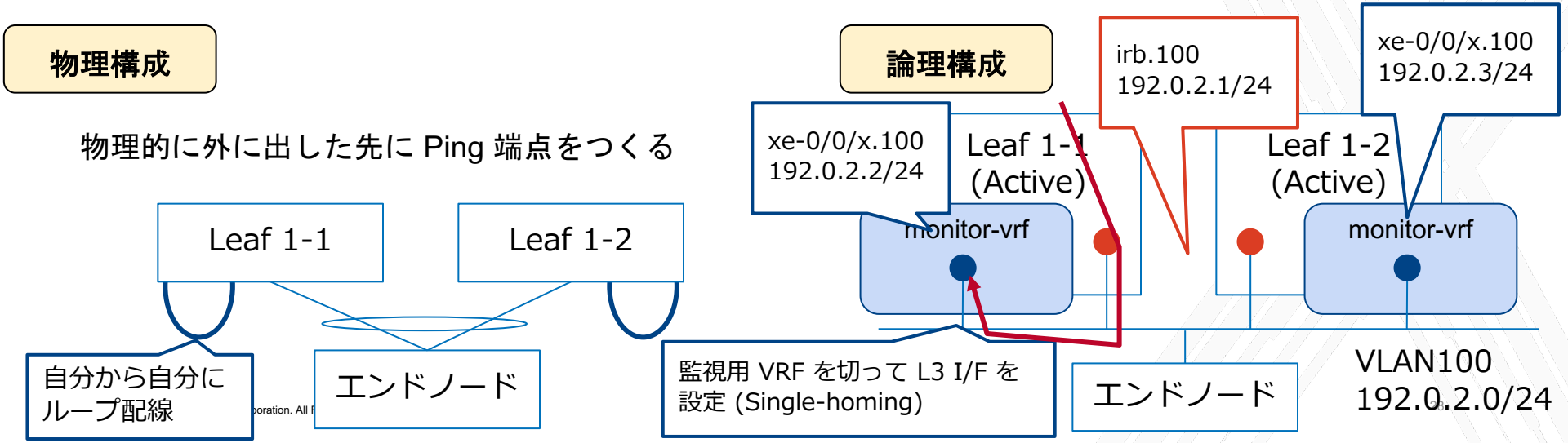
【監視案①】Real IP を付与する方式

- 1 系の IRB と 2 系の IRB に異なる Real IP を追加で付与
- Downlink からの疎通確認は向かない
  - ○ LAG Hash により Ping が Spine 経由になってしまうことがある
- Uplink からの監視は効果が薄い
  - ○ 例) Uplink から来た Real IP 宛ての ICMP に NOS レベルで反応してしまう
    => 今回のような TCAM から MAC が消えるような不具合には気づけない
    => **やはりエンドノードの通信と近い形で疎通確認するのが望ましい**



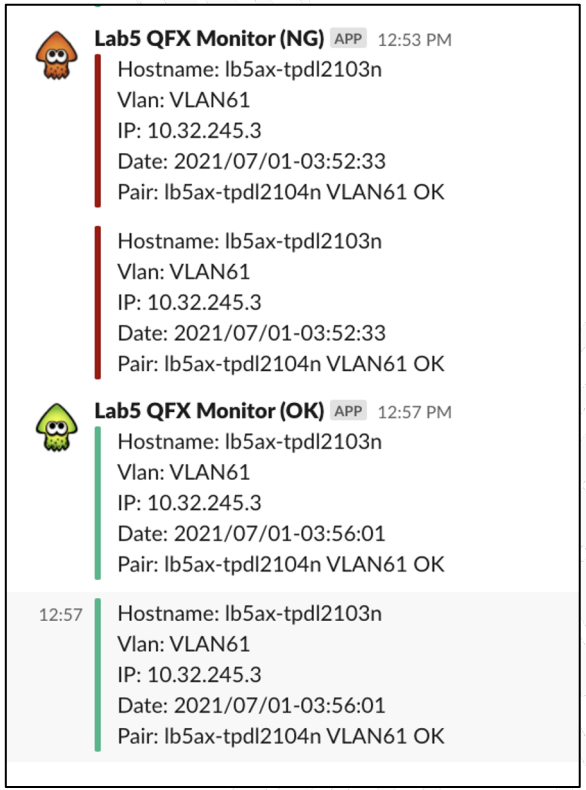
【監視案②】物理ループバック方式

- エンドノードと同様に物理 I/F の外に監視端点をつくる
- コストの節約から自分自身に配線し監視用 L3 I/F を別 VRF に作成
- 監視案① と比べて Uplink からの監視が有効になる
  - ○ 今回のような IRB の MAC が消えるチップ系の不具合もばっちり拾える



監視案②を採用

- Controller Pod から各 Routed Port の IP に対し Ping 監視を実施
- NG / OK を Slack 通知



商用へ

- 検証結果も良好、見つかった不具合も Fix、Anycast Gateway の本質的な問題も監視により克服してついに商用導入！
- 今のところ元気に動いている

まとめ

- First-hop redundancy として VC をやめて EVPN MH + AG にした
- とてもよくなった
  - ○ 不具合のリスクが減った
  - ○ Version Up も安全に実施可能になった
- 我々が過去の経験から培ってきた検証のポイントが活きた
  - ○ 幅広く、集中的に、繰り返し、が大事
- AG の本質的な問題は監視で克服
  - ○ 識別子 (IP) が 1 個しかないから問題発生時にどっちが悪いか判別困難
    => 監視する意味のある識別子を筐体単位で作って監視
- 商用で今のところ元気に動いてる

議論ポイント

- Virtual-Chassis や EVPN AG で良かったこと / 困ったこと
- こういう検証すればいいよ
- 改めて他の first hop redundancy 手法との比較
- 基盤監視ってどんなのしてる
- その他なんでも