

コンピュータネットワーク

経路制御プロトコル編
2011/11/9

前回の復習

- ・ 近隣探索プロトコル
 - ・ ARP
 - ・ NDP
- ・ IP の仕組みと仕様
 - ・ IP パケット
 - ・ End-to-End モデル
- ・ 経路制御の仕組み

本日の流れ

- ・ 経路制御プロトコル
 - ・ 概念と仕組み
 - ・ アルゴリズム
- ・ プロトコル例
 - ・ RIP
 - ・ OSPF
 - ・ BGP
- ・ 経路制御の階層化

連絡事項

- ・ 11/16 (水) は休講です。
 - ・ 海外出張のため
- ・ 11/23 (水) は祝日です。
- ・ 11/30 (水) は。。。休講です。
 - ・ 海外出張のため
- ・ 次回の授業は 12/7 (水) となります。
 - ・ ほんとすみません。

前回のまとめ

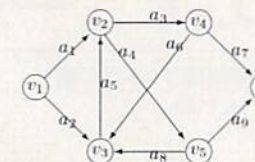
- ・ 経路制御とは
 - ・ パケットを宛先に従って転送すること
 - ・ 静的経路制御
 - ・ 動的経路制御
- ・ 経路表とは
 - ・ 宛先を記した一覧表
 - ・ Prefix と NextHop Address を対としたデータベース

経路制御プロトコル

- ・ 距離ベクトル型 (Distance Vector Model)
 - ・ RIP
- ・ リンク状態型 (Link State Model)
 - ・ OSPF
 - ・ IS-IS
- ・ パスベクトル型 (Path Vector Model)
 - ・ BGP

グラフ理論

- ・ グラフとは
 - ・ 辺 (path) と頂点をもつ
 - ・ 路 (経路): ある頂点からある頂点までの道筋
- ・ 向きを有するか
 - ・ 有向グラフ
 - ・ 無向グラフ



- ・ 最適化問題例
 - ・ 最短経路問題: 最短な経路
 - ・ 最小木問題: 無向グラフにおける最短経路
 - ・ 巡回セールスマン問題: 全ての頂点を通り、コストが最小となる経路
 - ・ 最大流問題: 入口から出口に流れる量を最大にする
 - ・ 最小費用流問題: 単位フローあたりの費用を考慮した流量

距離ベクトル型の特徴

- ・ 代表的なプロトコル: RIPv1, RIPv2
- ・ プロトコルの動作が簡単
 - ・ プログラムも簡単
 - ・ 小型ルータ/エントリクラスのL3スイッチでも搭載
- ・ 経路の収束にかかる時間が長い
 - ・ 小規模なネットワークなら高速
 - ・ 収束
 - ・ 全てのルータが正しい経路表が作成し、それ以上更新しなくなるまでの時間
- ・ ループが発生する可能性
 - ・ 確率は低いが、完全にループフリーではない
- ・ スケーラビリティ
 - ・ 大規模なネットワークほど、交換する経路情報が増す

距離ベクトル型のアルゴリズム

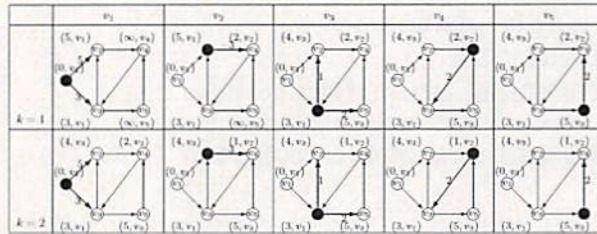
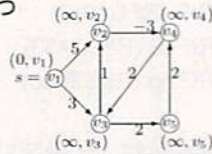
- ・ ベルマン-フォード法 (Bellman-Ford Algorithm)
 - ・ グラフ理論による最短経路解決アルゴリズム
 - ・ この分散型 (各ノードが自律的に行う) を用いて距離ベクトル型ルーティングプロトコルを実現
- ・ 各ルータは、経路情報を隣接ルータに広告
 - ・ 自分が到達可能なプレフィックス
 - ・ そこへの距離 (例えばホップ数)
- ・ 経路情報を受け取った場合の処理
 - ・ 自分が知らない経路なら、採用
 - ・ 自分が知っている経路より、短い経路なら採用
- ・ 新しい経路情報を隣接ルータに広告

11/11/09

9

Bellman – Ford Algorithm

- 単一起点最短経路問題の解法のひとつ
- 負の重みをもつ辺を扱うことができる
- 計算量
 - $O(V \cdot E)$ V : 頂点数, E : 辺数

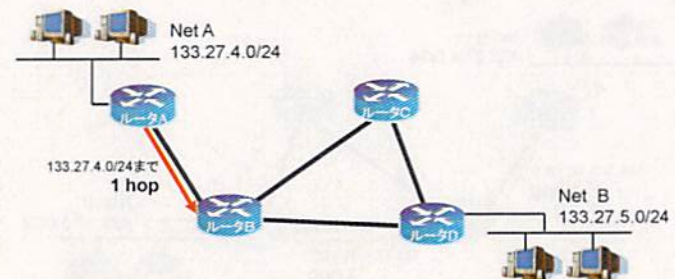


11/11/09

10

距離ベクトル型経路制御プロトコルの動作 (1/4)

- 自分が持つ経路情報を隣接ルータに広告

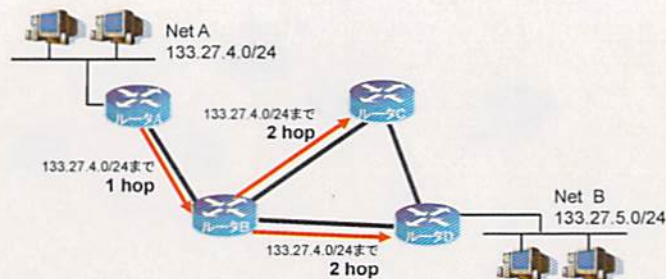


11/11/09

11

距離ベクトル型経路制御プロトコルの動作 (2/4)

- 受け取った経路情報を、更に隣接ルータに伝達

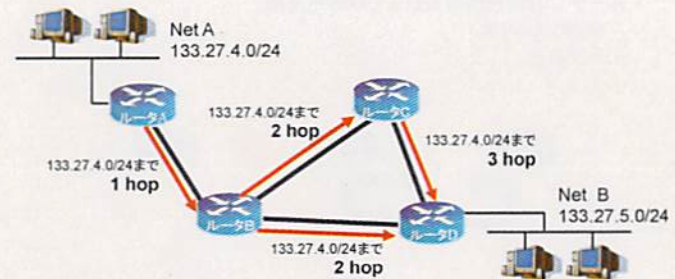


11/11/09

12

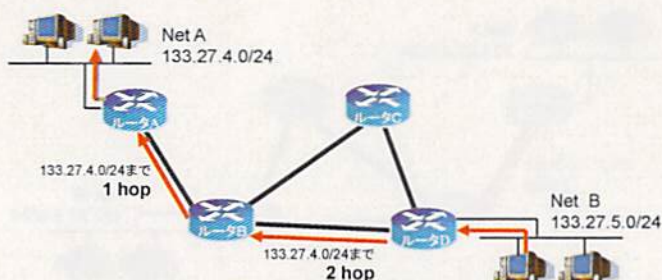
距離ベクトル型経路制御プロトコルの動作 (3/4)

- 受け取った経路情報が、自分が持つ経路情報よりホップ数が多い場合は破棄



距離ベクトル型経路制御プロトコルの動作 (4/4)

- 最短の経路でトラフィックを転送



RIP の仕組み

- RIP (RFC1058)
- RIPv2 (RFC2453)
- RIPng (RFC2080)
- Metric は 1～16 まで
 - 最大で 15 hop までのネットワーク
- 30秒ごとに広告
- 180秒間受信しなかった経路は holddown
 - Metric = 16 として一定時間保持 (60秒 or 120秒)
- 経路消滅までには 240秒 or 300秒かかる

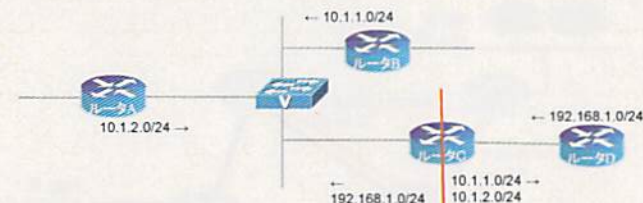
RIP におけるループ例

- ルータ B と ルータ C の間で障害発生
 - ルータ B は障害をすぐに検知可能
 - 経路を変更する
 - ルータ C は経路情報更新まで検知不可能
 - 経路はそのまま
- ループ発生



RIP の工夫 (1)

- Split Horizon
 - 受信した方向には受信した情報を送信しないという工夫
 - 経路情報の収束を早くし、ループ発生を防ぐ



11/11/09

17

RIP の工夫 (2)

- Split Horizon with Poisoned Reverse
 - ・受信した方向に対して metric 16 で投げ返す仕組み
 - ・さらに経路情報消滅時の収束を早めることができる
- Triggered Update
 - ・30秒待たずに経路情報を広告する仕組み

11/11/09

18

リンク状態型の特徴

- 代表的なリンク状態型経路制御プロトコル
 - ・ OSPF (Open Shortest Path First)
 - ・ IS-IS (Integrated System to Integrated System)
- 大規模なネットワークに適している
 - ・ Loop Free
 - ・ いったん経路が収束した後は、経路制御メッセージによるトラフィック量が少ない
 - ・ ネットワークの規模が大きくなっても、恒常的に流れる経路制御メッセージの増加量は少ない
- 距離ベクトル型に比べ、処理が複雑
 - ・ 安価なルータ、L3スイッチには搭載されていない
- 大規模なネットワークでも経路の収束が早い

11/11/09

19

リンク状態型のアルゴリズム

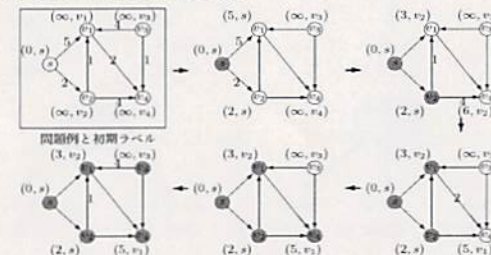
- ダイクストラ法 (Dijkstra Algorithm) を用いている
- 各ルータのリンク(インターフェース)情報をネットワーク全体で共有
 - ・ 例: R1はR2と繋がっている
 - ・ 例: R1には133.27.4.0/24が繋がっている
- ネットワーク全体にflooding
 - ・ ネットワーク内のすべてのルータにリンク情報が伝わる
 - ・ 全ルータは同一のリンク情報データベースを持つ
 - ・ 状態に変化があったリンクの情報だけを伝える
- 各ルータが、リンク情報からトポロジを再構成
 - ・ リンク情報を基に、自分がルート(根)となるツリーを作成
 - ・ ツリーにすれば、ある宛先までの最短経路がわかる
 - ・ 全ルータが同一の計算方法

11/11/09

20

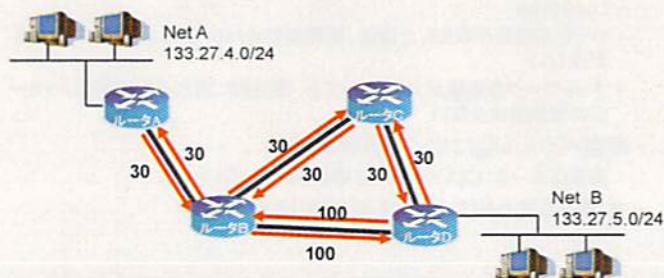
Dijkstra Algorithm

- 2点間の最短経路を求めるアルゴリズム
 - ・ インターネットの経路制御やカーナビ等にも応用
- 手法
 - ・ 始点から常に最短となる隣接頂点を見つけていき、その距離をラベリングすることで最短経路を発見する



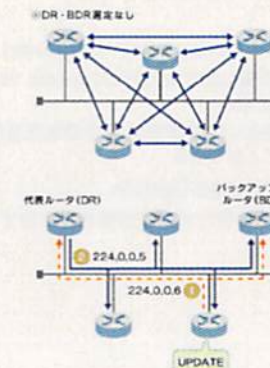
リンク状態型経路制御プロトコルの動作

- 各リンクにコストを設定
- 各ルータは、自分が持つリンクのUP/DOWNを監視
 - 各ルータは、自分が持つリンク情報をflooding
 - Link State Advertisement (LSA)



OSPF の仕組み

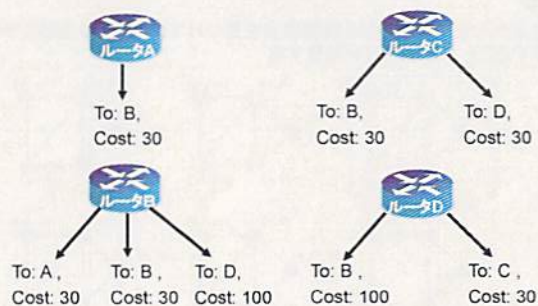
- OSPF
 - Open Shortest Path First
 - OSPFv2 : RFC2328
 - OSPFv3 : RFC2740
- DR (Designated Router) / BDR (Backup Designated Router)
 - 代表となるルータが選出され、そこに情報が集約される



出典: <http://bun.atmarkit.co.jp/bk801/kenzai/ospf24/ospf2401.html>

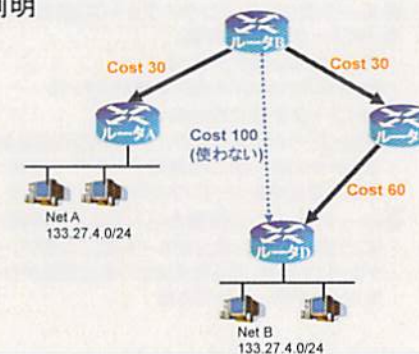
リンク状態のデータベース (LSDB)

- ルータは、自分が持つリンク情報をflooding
- 各ルータが、全ルータのリンク状態を保持



ツリーの作成

- コストが小さい順に組み合わせ、ツリーを作成
- 最短の経路が判明



11/11/09 25

LSDB の例

R1~R5: OSPF ルータ
 N1~N5: ネットワーク
 1: インターフェイスのコスト

LSDBでは、FROMがルータの行にコスト値を入力しFROMがネットワークの行には「コスト値はなし(0)」

FROM	R1	R2	R3	R4	R5	N1	N2	N3	N4
R1						0			0
R2						0			0
R3						0	0		0
R4						0	0	0	0
R5						0			0
N1	10								
N2	10	10	10						
N3			5	10					
N4		20		5	10				
N5					10				

<http://www.atmarkit.co.jp/fnetwork/rensai/iprt05/iprt01.html>

11/11/09 26

R1 から見たパスツリー

リンクコスト (トータルコスト)

FROM	R1	R2	R3	R4	R5	N1	N2	N3	N4
R1						0			0
R2						0			0
R3						0	0		0
R4						0	0	0	0
R5						0			0
N1	10								
N2	10	10	10						
N3			5	10					
N4		20		5	10				
N5					10				

11/11/09 27

OSPF エリアの分割

- バックボーンエリア
 - 基本となるエリア
- エリア分割の利点
 - 交換される情報量が減るために、データベースの肥大化を防ぐことができる
- ASR (Area Border Router)
 - エリア間でデータを交換するルータ
- ASBR (AS Border Router)
 - OSPF 以外のプロトコルと経路情報を相互交換するルータ

11/11/09 28

OSPF エリア分割概念図

バックボーンエリア(エリア0)

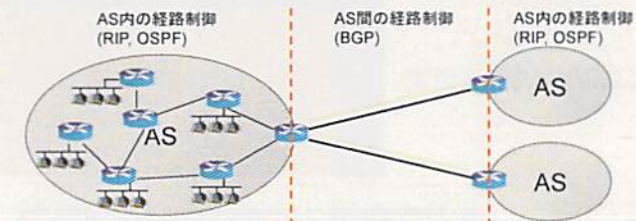
図 OSPF のエリア

各プロトコルの比較

	RIP	OSPF	IS-IS
特徴	<ul style="list-style-type: none"> ほとんどのL3機器でサポート 実装が容易でシンプル 小規模ネットワークで利用 	<ul style="list-style-type: none"> やや高価な機器でサポート IPに最適化、IAB推奨 中・大規模ネットワークで利用 	<ul style="list-style-type: none"> もともとはOSI CLNP用に設計 一部ルータでのみサポート 国内ではあまり利用されない
スケーラビリティ	△	○	○
計算量	少ない	多い (特にDR/BDR)	多い
収束時間	低速	高速	高速
アルゴリズム	距離ベクトル型	リンク状態型	リンク状態型
Neighbor生存確認	30秒	10秒 (Helloパケット)	10秒 (IS-IS Hello)
メトリック	ホップ数	コスト(帯域幅)	コスト

経路制御の階層化

- 規模性の問題
 - 全世界を一つのRIPやOSPFドメインで接続することはできない
- 障害範囲の限定
 - AS内の障害が世界中に伝播しない
- 経路数の軽減
 - 経路をAS外に広告する場合、経路を集約できる



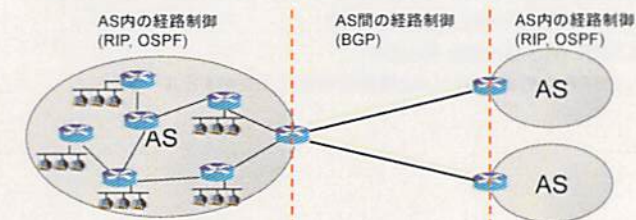
AS(Autonomous System)

- 経路制御ポリシーを共有するネットワークの集合
 - 外部からは1つのネットワークとして見える
 - 全国展開しているISPも外部から見ると1つの巨大なネットワーク
 - インターネットは各ASが相互に接続されたもの
- AS番号
 - 各ASは固有の番号を持つ
 - 日本ではJPNICがAS番号を割り当て
 - <http://www.nic.ad.jp/ja/ip/asnumber.html>



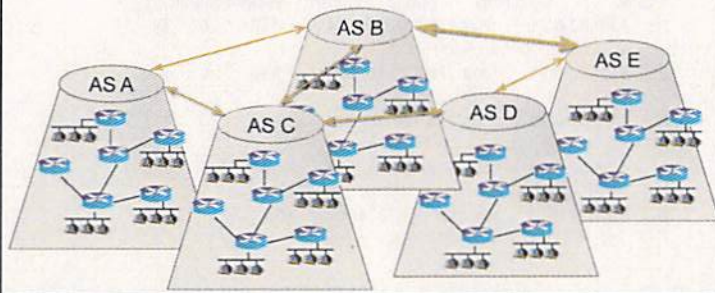
経路制御の分割

- 「AS内の経路制御」と「AS間の経路制御」
- 各ASは異なるポリシー・管理体系
- 規模性の問題
 - 全世界を一つのRIPやOSPFドメインで接続することはできない
- 障害範囲の限定
 - AS内の障害が世界中に伝播しない
- 経路数の軽減
 - 経路をAS外に広告する場合、経路を集約できる



経路制御の階層化

- ・経路制御は、大きく2階層に分かれる
 - ・AS間の経路制御
 - ・AS内の経路制御

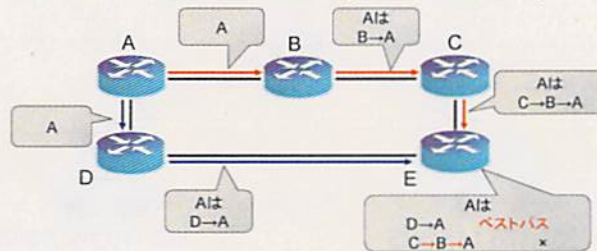


EGPとIGP

- ・EGP(Exterior Gateway Protocol)
 - ・AS間を接続するための経路制御プロトコル
 - ・AS間で共通のルーティングプロトコル
 - ・AS間の接続ポリシーを経路制御に反映
 - ・BGP4
- ・IGP(Interior Gateway Protocol)
 - ・AS内で使用する経路制御プロトコル
 - ・ASの管理者が任意の経路制御プロトコルを選択
 - ・迅速な経路制御を実現
 - ・RIP, OSPF, IS-IS ...

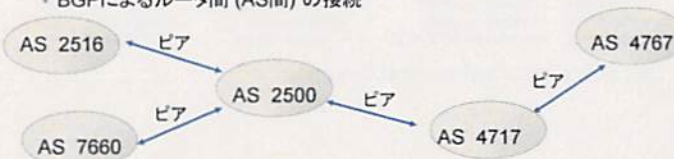
パスベクトル型の経路制御方式

- ・自分自身から目的地へ到達するまでに辿るパスを経路として広告
 - ・目的地へのホップ数や距離ではない
- ・より短いパスを選択して広告(ベストパス選択)



BGP (Border Gateway Protocol)

- ・パスベクトル型経路制御プロトコル
 - ・ループフリー
 - ・インターネット全体でループが起きたら大変
 - ・経路制御のポリシーを設定に反映しやすい
 - ・ISP間の接続の契約に従った経路制御
 - ・複数のExternalリンクをどう使いわけするか Etc...
- ・ピア
 - ・BGPによるルータ間 (AS間) の接続



From lana.org

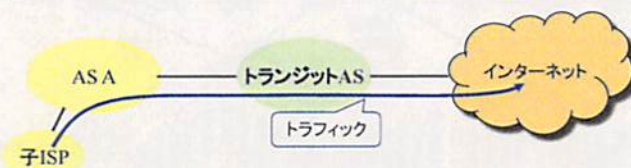


11/11/09

41

トランジット

- 自分以外のAS同士のトラフィックを、中間にあるASが中継すること
 - アップストリームASとも呼ばれる
- 通過するトラフィックはそのASのポリシーに従った転送が行われる



11/11/09

42

東京大学にとってのトランジット

- 東京大学のネットワークをトランジットしてくれている組織
 - SINET (Science Information Network)
 - AS2907
 - 学術情報ネットワーク
 - 1987年より運用開始
 - 文部科学省
 - WIDE (Widely Integrated Distributed Environment)
 - AS2500
 - 産学協同連携ネットワーク
 - 研究用ネットワーク
 - 1985年より運用開始

11/11/09

43

経路の調べ方

- traceroute コマンド
 - Windows の場合は tracert

```
sekiya[~]% traceroute www.google.co.jp
traceroute: Warning: www.google.co.jp has multiple addresses; using 72.14.203.103
traceroute to www.l.google.com (72.14.203.103), 64 hops max, 52 byte packets
 1 ra35-vlan299 (130.69.251.251)  0.681 ms  0.281 ms  0.480 ms
 2 ra36-vlan2 (133.11.127.43)  0.479 ms  0.380 ms  0.987 ms
 3 ra37-vlan3 (133.11.127.78)  0.482 ms  0.421 ms  0.996 ms
 4 foundry4.nezu.wide.ad.jp (133.11.125.238)  0.748 ms  0.410 ms  0.483 ms
 5 ve-42.foundry6.otemachi.wide.ad.jp (203.178.136.65)  0.734 ms  0.662 ms  0.732 ms
 6 ve-5.alalal.otemachi.wide.ad.jp (203.178.140.215)  2.482 ms  2.440 ms  2.484 ms
 7 as15169.dix-1e.jp (202.249.2.189)  2.468 ms  2.425 ms  2.461 ms
 8 209.85.241.68 (209.85.241.68)  2.724 ms  2.688 ms
 9 209.85.241.64 (209.85.241.64)  2.702 ms
10 209.85.250.86 (209.85.250.86)  33.718 ms  34.071 ms  33.980 ms
11 209.85.243.23 (209.85.243.23)  33.718 ms  33.598 ms
12 209.85.243.21 (209.85.243.21)  33.723 ms
13 209.85.241.154 (209.85.241.154)  37.488 ms
14 209.85.241.162 (209.85.241.162)  42.781 ms  35.206 ms
15 tx-in-f103.1e100.net (72.14.203.103)  34.213 ms  34.054 ms  34.192 ms
```

11/11/09

44

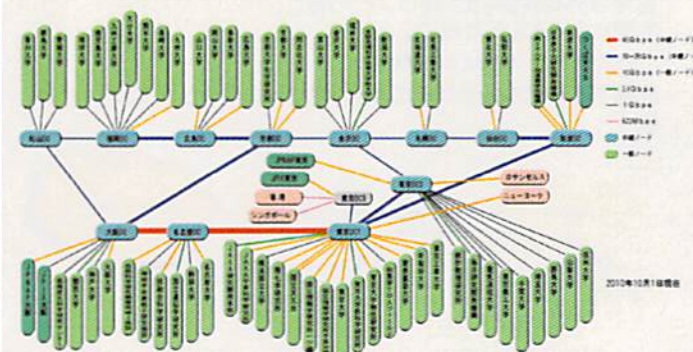
経路の調べ方 (AS)

- UNIX 系 traceroute コマンドのオプション

```
sekiya[~]% traceroute -A www.wikileaks.org
traceroute to wikileaks.org (184.72.37.90), 64 hops max, 40 byte packets
 1 ra35-vlan299.nc.u-tokyo.ac.jp (130.69.251.251) [AS2501]  2 ms  1 ms  0 ms
 2 ra36-vlan2.nc.u-tokyo.ac.jp (133.11.127.43) [AS2501]  0 ms  1 ms  1 ms
 3 ra37-vlan3.nc.u-tokyo.ac.jp (133.11.127.78) [AS2501]  0 ms  0 ms  1 ms
 4 tokyo1-dc-RM-GE-1-0-0-119.sinet.ad.jp (150.99.190.101) [AS2907]  1 ms  3 ms  1 ms
 5 tokyo2-dc-RM-AE-0-0-11.sinet.ad.jp (150.99.203.14) [AS2907]  5 ms  6 ms  6 ms
 6 lax-gate1-RM-P-7-0-0-11.sinet.ad.jp (150.99.203.62) [AS2907]  113 ms  98 ms  98 ms
 7 xe-11-3-0.edge5.LosAngeles1.Level3.net (4.59.48.1) [AS3356]  99 ms  110 ms  111 ms
 8 ae-93-90.ebr3.LosAngeles1.Level3.net (4.69.144.244) [AS3356]  123 ms  115 ms
 9 ae-73-70.ebr3.LosAngeles1.Level3.net (4.69.144.116) [AS3356]  105 ms
10 ae-2-2.ebr3.SanJose1.Level3.net (4.69.132.9) [AS3356]  119 ms  111 ms  114 ms
11 ae-63-63.csw1.SanJose1.Level3.net (4.69.134.226) [AS3356]  125 ms  115 ms
12 ae-73-73.csw2.SanJose1.Level3.net (4.69.134.230) [AS3356]  121 ms
13 ae-2-79.edge1.SanJose3.Level3.net (4.68.18.80) [AS3356]  109 ms  109 ms  109 ms
14 AMAZONCOM.edge1.SanJose3.Level3.net (4.53.208.22) [AS3356]  114 ms  111 ms  110 ms
15 * * *
16 * * *
```

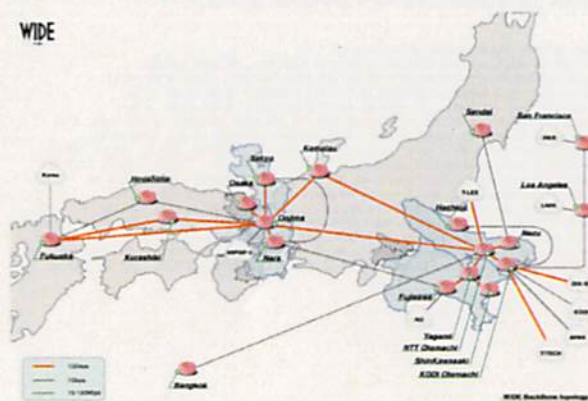

広域経路制御

SINET Backbone



WIDE Project Backbone

WIDE



パブリックピアリング IX(Internet eXchange)

- ・L2機器を共有して、複数のAS同士を相互に接続
- ・トラフィックの交換
- ・経路情報の交換

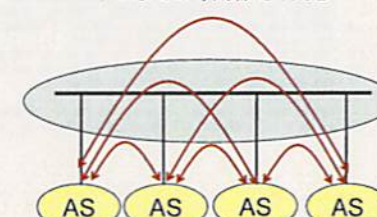
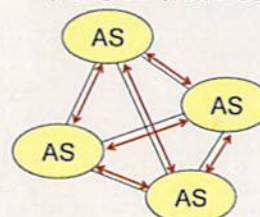
ピアリングするAS同士は対等な関係

個別の回線を用いた場合

⇒ たくさんの専用線が必要

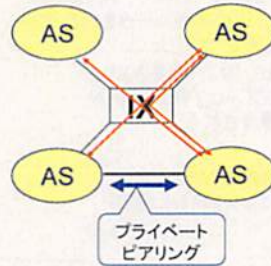
IXを使った場合

⇒ IXまでの専用線だけ用意



プライベートピアリング

- 二つのAS同士を専用のLAN回線等で接続し、他のASと設備を共有しない
 - AS同士を直接接続し、パブリックピアリングの問題を回避
 - 他ASからのトラフィックによる共有スイッチの品質変動
 - セキュリティ
 - 接続用回線やルータ機器等の設備で費用が発生する
 - 互いに利益がある場合に実施



日本の代表的な IX

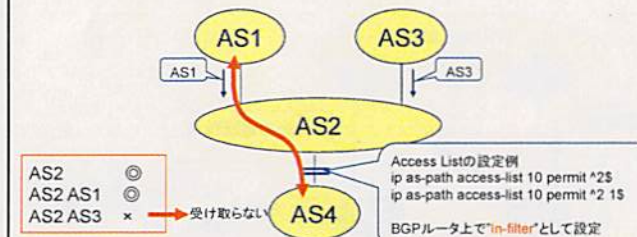
- DIX-IE (Distributed Internet eXchange In Edo)
- NSPIXP3 (Network Service Provider Internet eXchange Point 3)
- JPIX (JaPan Internet eXchange)
- JPNAP (Japan Network Access Point)
- Equinix
- BBIX

BGPによる経路制御ポリシーの実現

- 経路のフィルタリング
- BGPのアトリビュート
 - AS_PATH Prepend
 - MED(Multi Exit Discriminator)
 - Local Preference

経路のフィルタリング

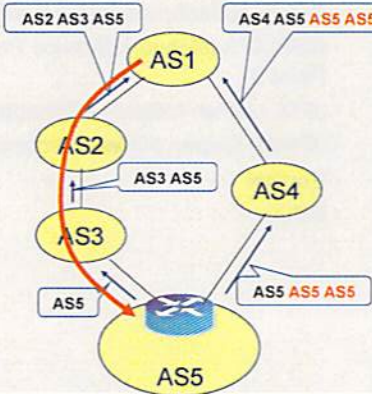
- 予め登録しておいた経路だけをやり取りする
 - In-filterによる受け取り経路の制限
 - Out-filterによるアナウンス経路の制限



AS_PATH Prependによる経路選択

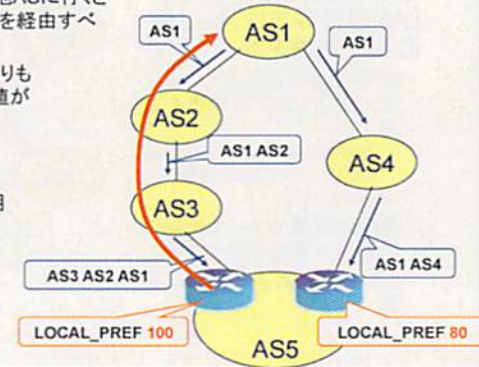
- BGPでは、AS PATHが長いほどネットワークが遠いと考えられる
- 自分のAS番号を広告する際に、AS_PATHに自分のAS番号を重複して加える
 - 隣接するAS同士にも拘らず、相手のASと自分のASとの間に複数のASがあるように見える

- ↓
- 実際のAS接続とは異なる経路をベストパスとして選択可能にする
 - 自組織向けトラフィックの制御に適用



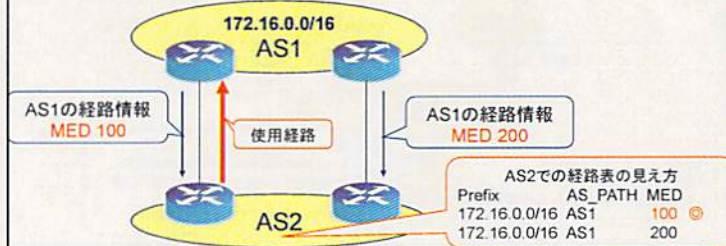
Local Preferenceによる経路選択

- 自AS内部において、他ASに行くときに、どの対外ルータを経由すべきか指定
- AS_PATHの長さよりもLOCAL_PREFの値が優先される
- 自組織から出ていくトラフィック制御に適用



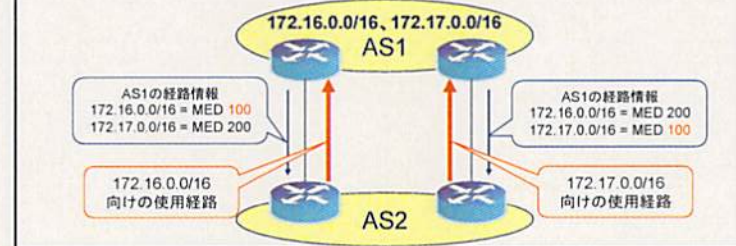
MED属性を使った経路選択

- 近隣AS同士が複数の回線で接続されている場合にMED値によって利用すべき回線を指定
 - 近隣ASのルータはMED値が小さい経路をベストパスとして選択
- 下りトラフィックの制御に適用



MED属性を使った負荷分散

- 近隣AS同士が複数の回線で接続されていて、複数のPrefixをアナウンスする場合に、Prefix単位で負荷分散を実現
- 各回線のルータでPrefixごとのMED値を互い違いに設定
- 下りトラフィックの制御に適用



11/11/09

57

Internet Routing Registry

- ・ BGPの経路情報自体に、その正しさを証明する裏付けがあるわけではない
 - ・ 意図されない経路情報が誤ってアナウンスされることもある
 - ・ 悪意による偽りの情報が流れる可能性 がある
- ・ インターネットの経路情報やその優先性に関する情報を蓄積するデータベース
 - ・ AS同士で交換される経路のフィルタ自動生成
 - ・ 経路情報の正当性、信憑性確認
 - ・ トラブルシュート時のコンタクト先情報取得
 - ・ インターネットのトポロジー情報取得