

ICMP

1999.5.21

ICMPとは

- IP層で発生した問題を、データグラム送信ホストに通告
 - » 問題解決のヒント
- ネットワーク層に関連した情報の獲得
 - » ネットワークのプロブ (probe)
 - » 遠隔地のホストの情報を得る
- IPデータグラムにカプセル化されて送られる
 - » 信頼性は保証できない
 - » ICMPに対するICMPは発生させない

Copyright ©1999, Suguru Yamaguchi, All right reserved

ICMPの機能

- 到達性チェック
 - » Echo Request / Echo Reply
- 問題報告
 - » Destination Unreachable
 - » Source Quench
 - » Redirect
 - » Time Exceeded for a Datagram
 - » Parameter Problem on a Datagram

Copyright ©1999, Suguru Yamaguchi, All right reserved

ICMPの機能

- 情報入手
 - » Timestamp Request / Timestamp Reply
 - » Address mask Request / Address mask Reply

Copyright ©1999, Suguru Yamaguchi, All right reserved

ICMPの通信形態

- 目的ホストからの返送
 - » 到達性チェック
 - » 情報入手
- 途中ゲートウェイからの返送
 - » 問題報告
 - » **P**データグラム転送処理中のエラーの報告
 - 送信ホストに対する「ヒント」

Copyright ©1999, Suguru Yamaguchi, All right reserved

ICMPの利用

- ping
 - » ICMP echo request / reply
- traceroute (tracert on Windows)
 - » UDP / ICMP time exceeded
- Path MTU Discovery
 - » 目的ホストまでの経路上の最小のMTUを発見
 - » fragmentation 発生防止
 - » DF bitを設定 / ICMP destination unreachable
 - » どの長さの **P**データグラムを出すかは課題
 - 最低保証 576 octet (**P**で規定)

Copyright ©1999, Suguru Yamaguchi, All right reserved

まとめ

- ICMPはIPの補助的機能を提供
- ICMPメッセージはIPデータグラムで転送
 - » ある意味でICMPはIPの上位層プロトコルとして実装
 - » 機能的にはICMPと不可分
- 課題
 - » セキュリティ

Copyright ©1999, Suguru Yamaguchi, All right reserved

経路制御

1999.5.21

IP層での経路制御の概念

- なぜ経路制御が必要なのか
 - » 特性の異なる複数の経路の選択
 - 遅延の大小
 - バンド幅
 - » 障害の回避
 - 迂回経路の選択
- 鉄道網で考えてみよう
 - » 本学から東京駅までの移動を考えると...
 - 新幹線や飛行機の利用はどうかんがえるのか？
 - たんなトンネルや京都駅が爆破されたらどうなるか？

Copyright ©1999, Suguru Yamaguchi, All right reserved

考えるべきポイント

- 経路を選択するためのコスト
 - » 伝送遅延
 - » バンド幅
 - » ゲートウェイ通過数
- 障害の検出方法
- 障害発生情報の伝播方法

Copyright ©1999, Suguru Yamaguchi, All right reserved

経路制御のゴール

- 最適だと思われる経路制御テーブルを作る
- $nexthop = f(destination)$
- $nexthop = f(destination, preference)$

Copyright ©1999, Suguru Yamaguchi, All right reserved

基本的な方法

- Source Routing
 - » 終点ホストまでの全ての経路を始点ホストで決定
 - » 送信者が全てを決定する
 - » どの程度の精密さで行うかはいろいろある
 - 全てのホストを決める
 - 必ず通過しなければならないホストを指定する
- Hop-by-Hop Routing
 - » いきあたりばったり方法
 - » 通過する各ゲートウェイで経路を決定

Copyright ©1999, Suguru Yamaguchi, All right reserved

TCP/IP の実装

- Source Routing , Hop-by-Hop Routing の両方を提供
- Source Routing
 - » IPオプションとして定義
 - » 送信ノードで経路を決定
 - » strict routing と non-strict routing
- Hop-by-Hop routing
 - » パケットにはIPヘッダに示された終点ホストのみ指定
 - » 各ゲートウェイ毎に経路を決定
- 現状では Hop-by-Hop routing が主流

Copyright ©1999, Suguru Yamaguchi, All right reserved

経路制御表の管理

- | | |
|---|--|
| <ul style="list-style-type: none">■ 静的経路制御<ul style="list-style-type: none">» static routing» 管理者が設定した経路情報を使用» 障害やネットワークポロジ変化への迅速な対応が不可能» ノート数に応じた管理オーバーヘッド» でも便利な状況はある | <ul style="list-style-type: none">■ 動的経路制御<ul style="list-style-type: none">» dynamic routing» 動的に経路を選択» 障害や混雑を回避する機能を組み込める» 動的に経路情報を伝播させるための経路制御プロトコル (Routing Control Protocol) が必要 |
|---|--|

Copyright ©1999, Suguru Yamaguchi, All right reserved

経路情報交換方法

■ Bellman-Ford 型

- » Distance Vector 型
- » 経路は分散的に計算
 - 各ゲートウェイで計算結果が異なる場合がある
- » 隣接ゲートウェイ間で経路情報を交換
- » 比較的オーバーヘッドが小さい
- » RIP

■ Link-State 型

- » 分散データベースを形成
- » 共通のデータベースから経路を計算
 - したがって経路のふらつきが発生が非常に少ない
- » Shortest Path First アルゴリズム
- » Routing Loop が発生しにくい
- » OSPF

Copyright ©1999, Suguru Yamaguchi, All right reserved

RIP (I)

- Routing Information Protocol
- RFC1058
- 隣接ゲートウェイ間で情報交換
- ホップ数を基本とした制御
- Distance Vector Algorithm
 - » $D(i,i) = 0$ (for all I)
 - » $D(i,j) = \min_k [d(i,k) + d(k,j)]$ (for otherwise)

Copyright ©1999, Suguru Yamaguchi, All right reserved

RIP ②

- 30秒毎に情報交換
- metric 16 が到達不能を表す

Copyright ©1999, Suguru Yamaguchi, All right reserved

RIPの問題点

- 直径15以下のネットワークしか対応できない
- ネットワークの負荷やバント幅を表現できない
- 経路情報の更新をすばやく伝える手段が必要
 - » トリガ (trigger) の実装
 - » 経路の不安定な時間を短くする

Copyright ©1999, Suguru Yamaguchi, All right reserved

経路制御の階層化 ①

■ 問題点

- » 例えばRIPは小規模ネットワークにしか適応できない
- » インターネットの規模は拡大
- » 複数の管理母体の存在

■ 経路制御の階層化

- » 外部 (exterior) と内部 (interior)
- » 内部」は単一の経路制御ポリシーを実装
- » 異なるポリシーを持った空間との経路制御は 外部」と定義

Copyright ©1999, Suguru Yamaguchi, All right reserved

経路制御の階層化 ②

■ AS (Autonomous System)

- » 単一の経路制御ポリシーを実装
- » 通常は一つのネットワーク運用母体に対応
 - ISP (Internet Service Provider)

■ 階層化

- » AS内部の経路制御は各運用母体が整合性を持って行う
- » AS間の経路制御は運用母体間の各種条件調整の上で設定

Copyright ©1999, Suguru Yamaguchi, All right reserved

経路制御の階層化 ③

- IGP: Interior Gateway Protocol
 - » RIP, RIP2, OSPF, IGRP, IS-IS
 - » 動的な経路制御と到達性の確保が最優先
- EGP: Exterior Gateway Protocol
 - » BGP-4
 - » 基本的には静的経路制御の自動管理

Copyright ©1999, Suguru Yamaguchi, All right reserved

まとめ

- 経路制御は知的な作業
- 方式
 - » Source Routing / Hop-by-hop Routing
 - » 静的経路制御 / 動的経路制御
- 経路制御の階層化
 - » インターネットの大規模化に対応
 - » AS
 - » IGP / EGP

Copyright ©1999, Suguru Yamaguchi, All right reserved