KONAMI

P2P通信技術:NAT越え
～STUNとUPnPと、時々、TURN～

株式会社コナミデジタルエンタテインメント
プロジェクトソリューションセンター
R&D推進グループ
佐藤 良

KONAMI

目次

KONAMI

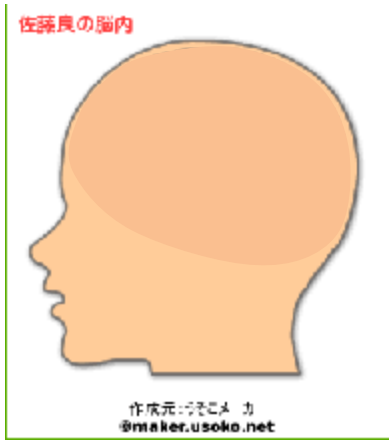
1. はじめに
2. NATの概要
3. 静的NAT　4. 動的NAT
5. NATの挙動をSTUNで調査する
6. NAT越え時のテクニック
7. 事例
8. 今後

1. はじめに

KONAMI

自己紹介

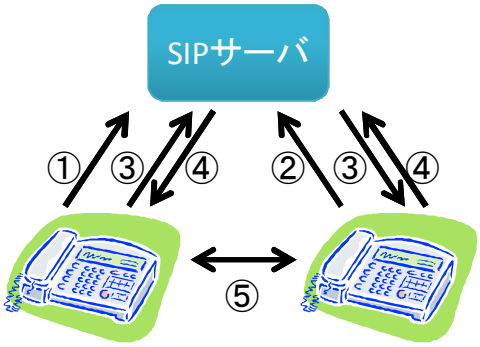
- 仕事
  - オンラインゲームのネットワーク技術開発
    - 2004～ NAT越え
      - 主な採用タイトル
        - » ウイニングイレブンシリーズ
        - » メタルギアオンラインシリーズ
    - 2006～ BitTorrent
      - 主な採用タイトル
        - » METAL GEAR ONLINE β VERSION
- 趣味
  - 習い事
    - ギターとか
  - （最近行ってないですが）パラグライダー

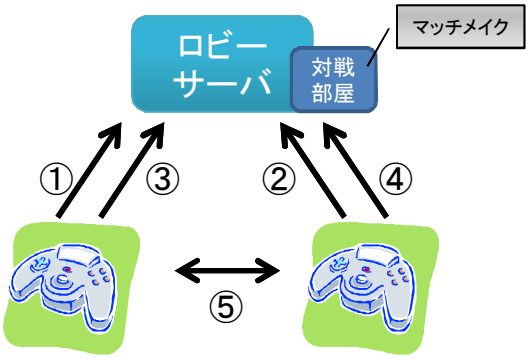


KONAMI

VoIPシステムと オンラインゲームシステムとの対応関係

① SIPサーバに登録(REGISTER)	⇔	① ロビーへ行く
② 相手もSIPサーバに登録(REGISTER)	⇔	② 相手もロビーへ行く
③ 電話を掛ける(INVITE)	⇔	③ 対戦部屋を作る
④ 受話器を取る(OK)	⇔	④ 相手が対戦部屋に入る
⑤ 通話開始	⇔	⑤ ゲーム開始

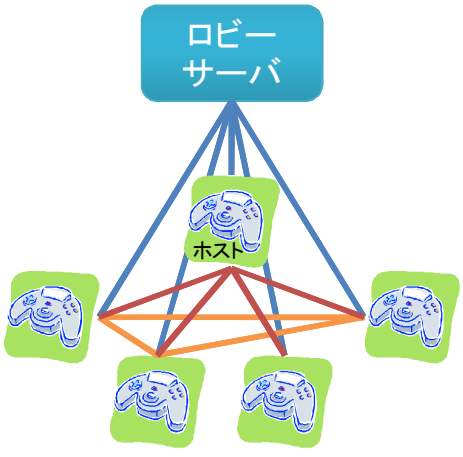




KONAMI

例）METAL GEAR ONLINE

- 2008年度第一四半期発売予定
- 16人対戦
- ネットワークトポロジ
  - クライアント/サーバ-P2Pハイブリッド
  - ホスト（スーパーノード）有り不完全メッシュ



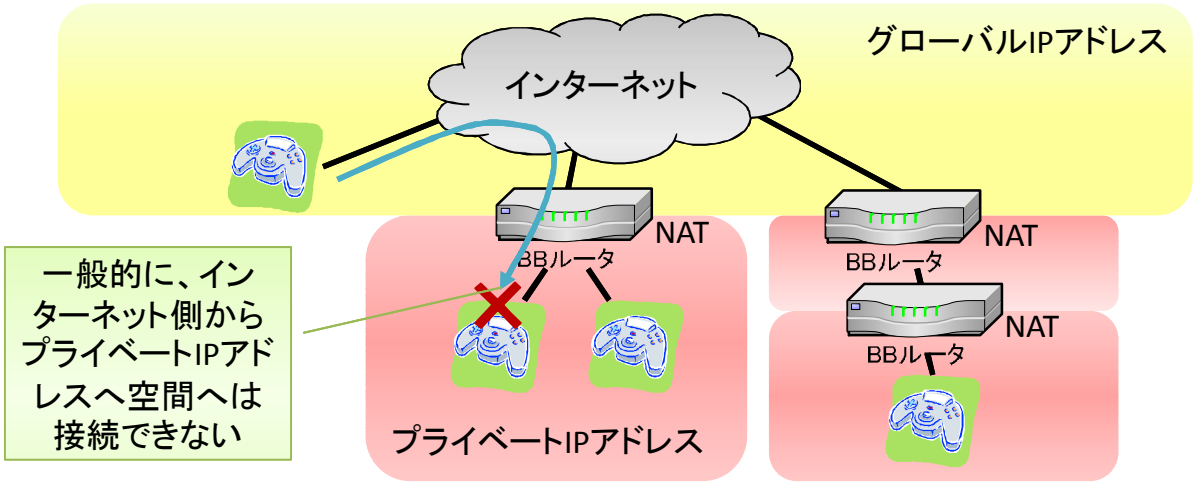
※スクリーンショットは開発中のものです

KONAMI

本日のお題

- クライアント/サーバ型では距離やトラフィック集中等で遅延が大きくなるので、直接通信させたい

「極力ユーザの手作業による設定無しに、以下の全てのクライアント同士でUDP通信できるようにせよ」



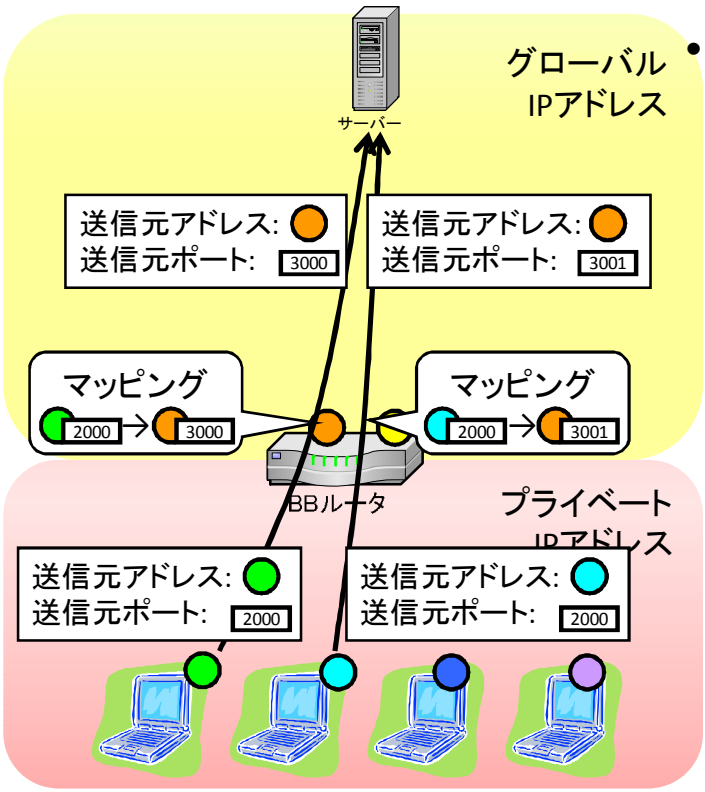
KONAMI

1. はじめに
2. NATの概要
   3. 静的NAT　4. 動的NAT
5. NATの挙動をSTUNで調査する
6. NAT越え時のテクニック
7. 事例
8. 今後

2. NATの概要

KONAMI

NAT（Network Address Translation）とは



- IPアドレス（とポート）を変換する仕組み

伝統的NAT（外方向NAT）

NAPT

（Network Address Port Translation）
IPアドレスとポート番号を変換する仕組み

ベーシックNAT

IPアドレスを変換する仕組み

KONAMI

（参考）rfc 2663での定義

NAT

伝統的NAT（外方向NAT）

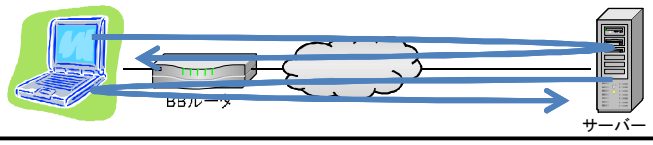
NAPT

（Network Address Port Translation）
IPアドレスとポート番号を変換する仕組み

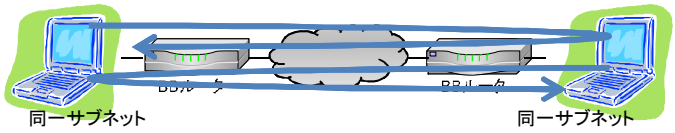
ベーシックNAT

IPアドレスを変換する仕組み

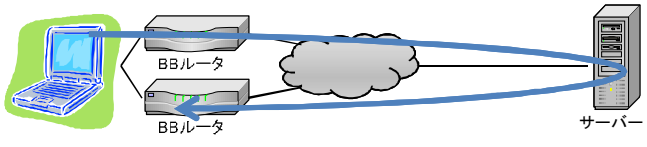
双方向NAT



両変換NAT



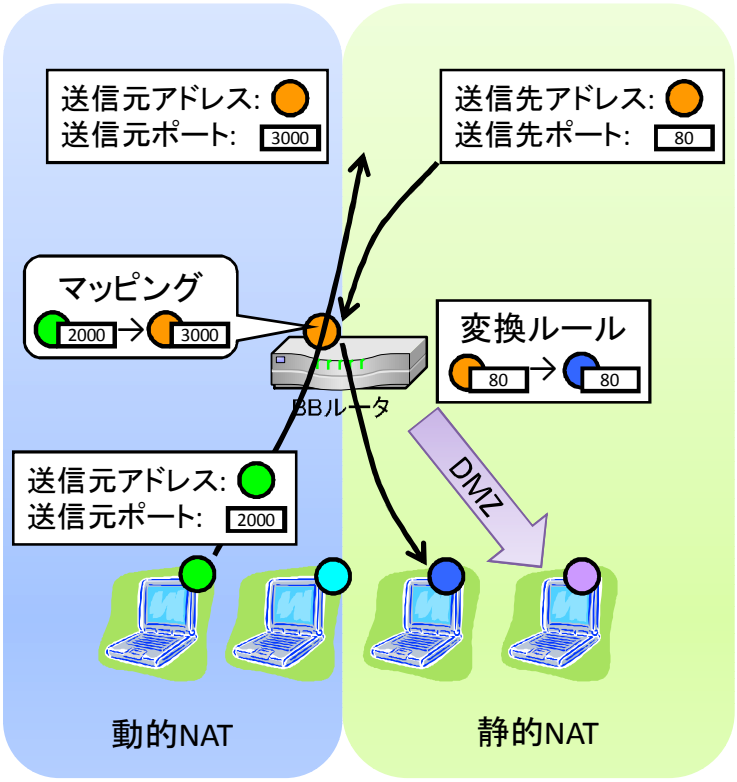
マルチホームNAT



KONAMI

その他のNAT機能とその関連機能

- 静的NAT
  - 主に外→中の通信で設定する
    - 静的ポートフォワーディング
    - バーチャルサーバ
    - スペシャルアプリケーション
- ルールに引っかからなかったパケットの送り先を指定する機能
  - DMZ
- ファイアウォール機能
  - パケットフィルタ
    - ステートフルインスペクション

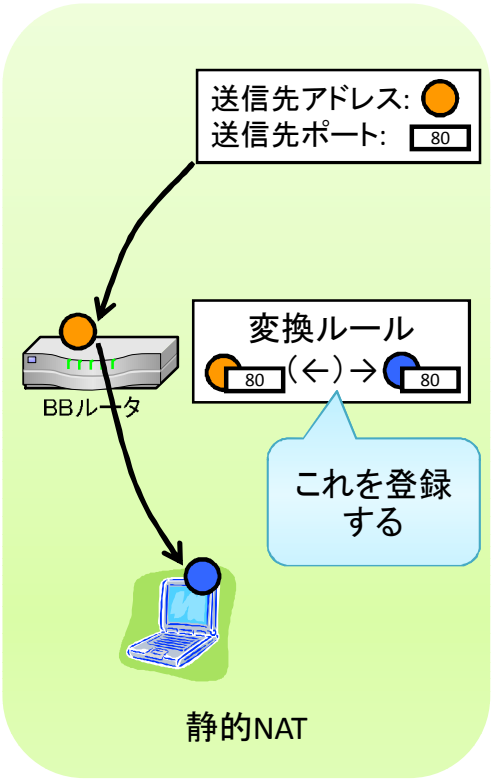


KONAMI

3. 静的NAT

KONAMI

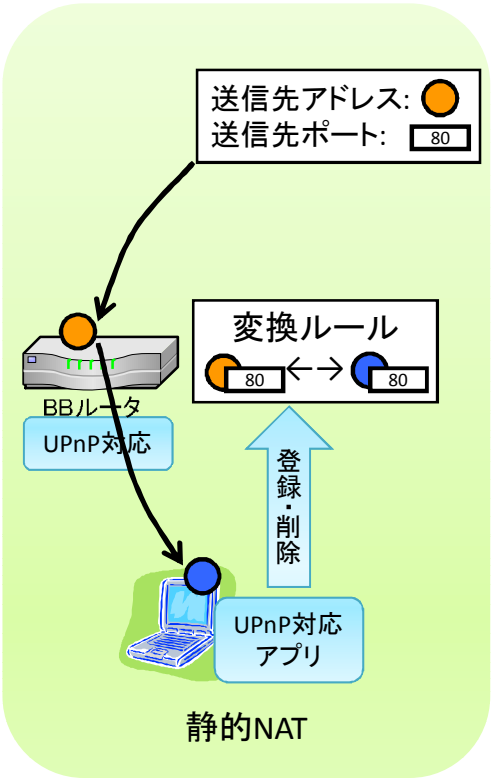
ルータに手で設定する



1. Webブラウザでルータの管理画面にログインする
2. 静的ポートフォワーディングを設定する
   - プロトコル
   - WAN側IPアドレス
   - WAN側ポート
   - LAN側IPアドレス
   - LAN側ポート

KONAMI

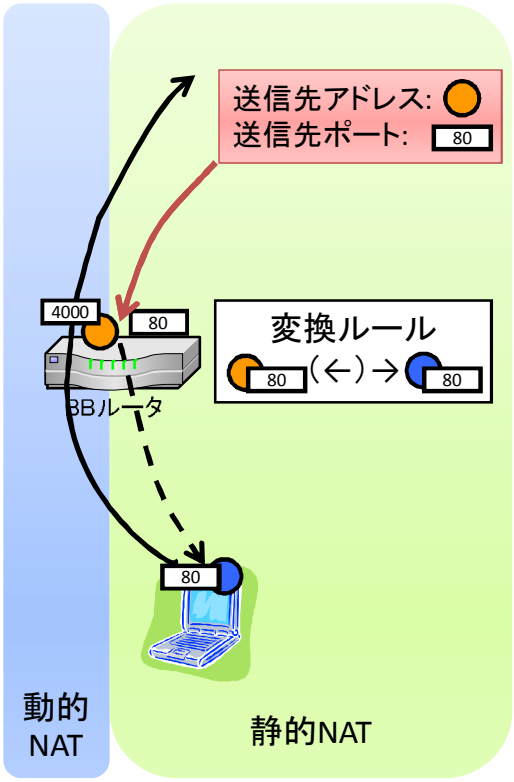
UPnPでアプリケーションから設定する



- 動作要件
  - ルータがUPnPに対応していること
  - アプリがUPnPに対応していること
- 問題点
  - 一番近いルータしか設定できないことが多い
  - 外内同じポートしか設定できないものが多い
  - ファイアウォールの設定は出来ない
  - 実装がおかしいルータが多々存在する
  - UPnP機能がデフォルトで無効になっているものがある

KONAMI

マップの有効化トリガ



- 中継パケットが設定した変換ルールに従うようになるトリガは？
  - 内向きパケット
    - これが必要なルータが希にある
  - 外向きパケット
  - どちらでも or デフォルトで有効

- 外側ポートを一発どうにかして叩いてやる

（注意点）

- クライアントのみでは解決できない
  - STUN（後述）で叩ける

KONAMI

4. 動的NAT

KONAMI

rfc 4787による マッピング特性とフィルタリング特性モデル

マッピング特性

マッピング生成規則
- 通信相手に依らず同一
- アドレス依存
- アドレス＆ポート依存

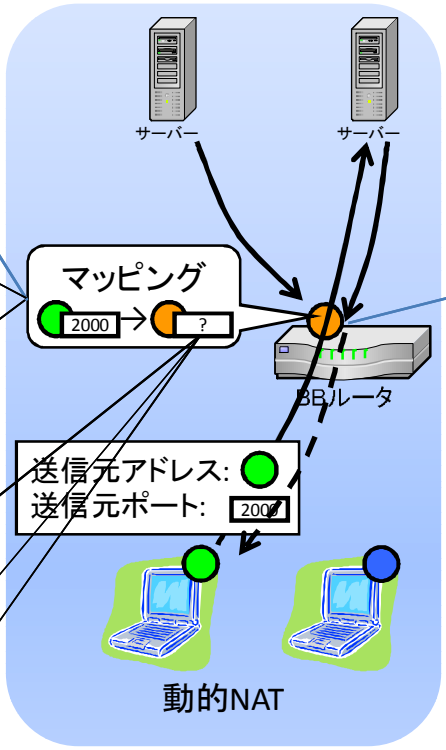
マッピング有効時間

マッピングタイマー リフレッシュ方向
- 外向き
- 内向き
- どちらでも

ポート割り当て規則
- ポート番号維持
- ポート番号多重
- ポート番号維持無し

ポートパリティ

ポート隣接性



フィルタリング特性
- 通信相手に依らず通過
- アドレス依存
- アドレス＆ポート依存

NAT下に複数の端末がある場合は、マッピング特性はより複雑になる

KONAMI

1. はじめに
2. NATの概要
3. 静的NAT　4. 動的NAT
5. NATの挙動をSTUNで調査する
6. NAT越え時のテクニック
7. 事例
8. 今後

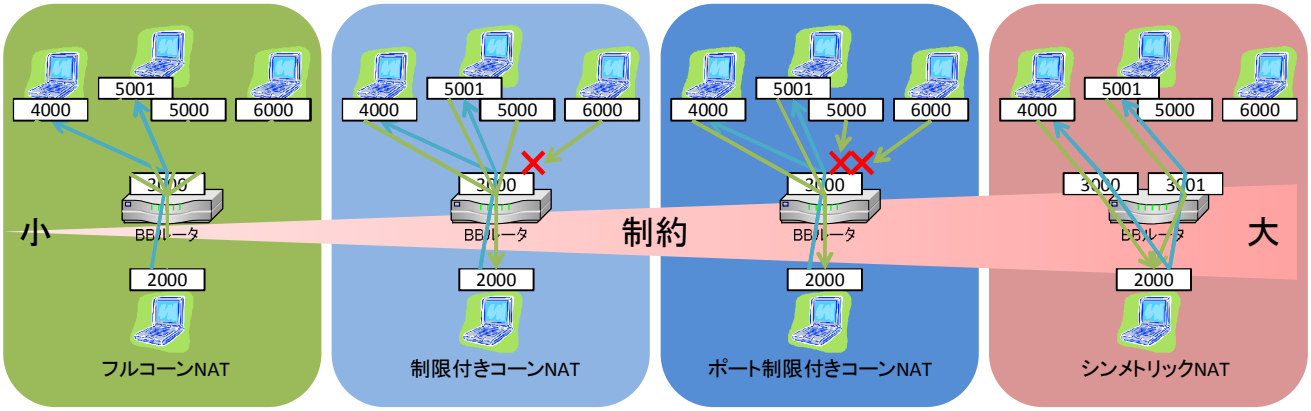
5. NATの挙動をSTUNで調査する

KONAMI

NATの分類

- rfc 4787とrfc 3489（クラシックSTUN）の分類対応表

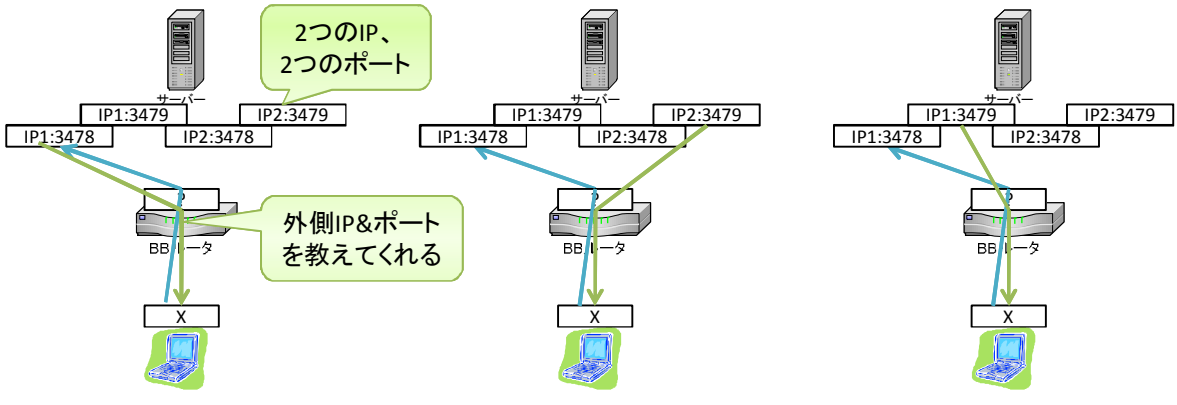
		フィルタリング特性			どこかでパケットドロップ
		通信相手に依らず通過	アドレス依存	アドレス＆ポート依存	
マッピング生成規則	通信相手に依らず同一	フルコーンNAT	制限付きコーンNAT	ポート制限付きコーンNAT	UDPブロックド
	アドレス依存	シンメトリックNAT			
	アドレス＆ポート依存				
NAT無し		オープンインターネット	シンメトリックUDPファイアウォール		



KONAMI

STUNの概要

- クライアント/サーバ型のプロトコル
  - レスポンスで、マップされたIPアドレスとポートを教えてくれる



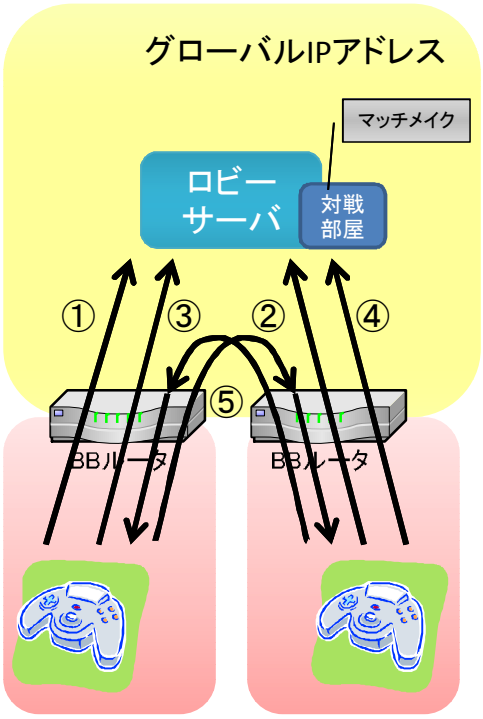
- NATが多段だった場合、一番制約の厳しいNATの結果となる
- 何度もテストする場合は、ソースポートを変更する（もしくはプロトコルを拡張する）必要がある

最終的に得られる物は、ローカルIP&ポート、グローバルIP&ポート、NATタイプ

KONAMI

STUNで調査した後

- ロビーサーバに誰と接続するのかを教えてもらう
  - ① ロビーへ行く
    - 接続情報（ローカルIPアドレス&ポート番号, グローバルIPアドレス&ポート番号, NATタイプ）を伝える
  - ② 相手もロビーへ行く
    - 接続情報（ローカルIPアドレス&ポート番号, グローバルIPアドレス&ポート番号, NATタイプ）を伝える
  - ③ 対戦部屋を作る
  - ④ 相手が対戦部屋に入る
    - それぞれの接続情報を得る
  - ⑤ ゲーム開始



KONAMI

NATタイプ別接続可否表

接続先 / 接続元	UDPブロックド	オープンインターネット	シンメトリックUDPファイアウォール	フルコーンNAT	制限付きコーンNAT	ポート制限付きコーンNAT	シンメトリックNAT	備考
UDPブロックド	×	×	×	×	×	×	×	STUNサーバが落ちている時もこれになる
オープンインターネット	×	○※	△※	○	△	△※	△※	
シンメトリックUDPファイアウォール	×	○※	×※	○	△	×※	×※	
フルコーンNAT	×	○※	△	○	△	△	△	ヘアピン問題で接続できない場合有り
制限付きコーンNAT	×	○※	△	○	△	△	△	ヘアピン問題で接続できない場合有り
ポート制限付きコーンNAT	×	○※	×※	○	△	△	×	ヘアピン問題で接続できない場合有り
シンメトリックNAT	×	○※	×※	○	△	×	×	ヘアピン問題で接続できない場合有り

△ … 要UDPホールパンチング
※ … 実際と異なる場合有り

KONAMI

rfc 3489の分類の問題点

- rfc 4787とrfc 3489（クラシックSTUN）の分類対応表 – 改訂版

<table>
<tr><th colspan="3" rowspan="2"></th><th colspan="3">フィルタリング特性</th><th rowspan="2">どこかで
パケット
ドロップ</th></tr>
<tr><th>通信相手に依らず通過</th><th>アドレス依存</th><th>アドレス＆ポート依存</th></tr>
<tr><td rowspan="3">NAT有り</td><td rowspan="3">マッピング生成規則</td><td>通信相手に依らず同一</td><td>フルコーンNAT</td><td>制限付きコーンNAT</td><td>ポート制限付きコーンNAT</td><td rowspan="5">UDP
ブロックド</td></tr>
<tr><td>アドレス依存</td><td colspan="3" rowspan="2">シンメトリックNAT</td></tr>
<tr><td>アドレス＆ポート依存</td></tr>
<tr><td rowspan="3">NAT無し</td><td rowspan="3">マッピング生成規則</td><td>通信相手に依らず同一</td><td>オープンインターネット</td><td colspan="2">~~シンメトリックUDPファイアウォール~~
↓
制限付きオープン，ポート制限付きオープン</td></tr>
<tr><td>アドレス依存</td><td rowspan="2">~~オープンインターネット~~
↓
シンメトリックオープン</td><td colspan="2" rowspan="2">~~シンメトリックUDPファイアウォール~~
↓
シンメトリックオープン</td></tr>
<tr><td>アドレス＆ポート依存</td></tr>
</table>

- ポートの割り当てられ方とフィルタリング特性が重要
  - オープンインターネットの一部は、シンメトリックオープン（勝手に命名）
  - シンメトリックUDPファイアウォールは、シンメトリックオープン、制限付きオープンとポート制限付きオープン（勝手に命名）
- 実はNATの有無はそれ程問題ではない

KONAMI

STUNの近況

Simple Traversal of UDP Through NATsから、Session Traversal Utilities for (NAT)へ

- rfc 3489（クラシックSTUN）
  - NAT挙動判定
  - バインディング有効期間検出

- I-D.ietf-behave-rfc3489bis（次期STUN）
- I-D.ietf-behave-nat-behavior-discovery
  - NATマッピング判定
  - NATフィルタリング判定
  - バインディング有効期間検出
  - NATヘアピン診断
  - フラグメントの取り扱い方の判定
  - 一般的なアプリケーションゲートウェイの検出

KONAMI

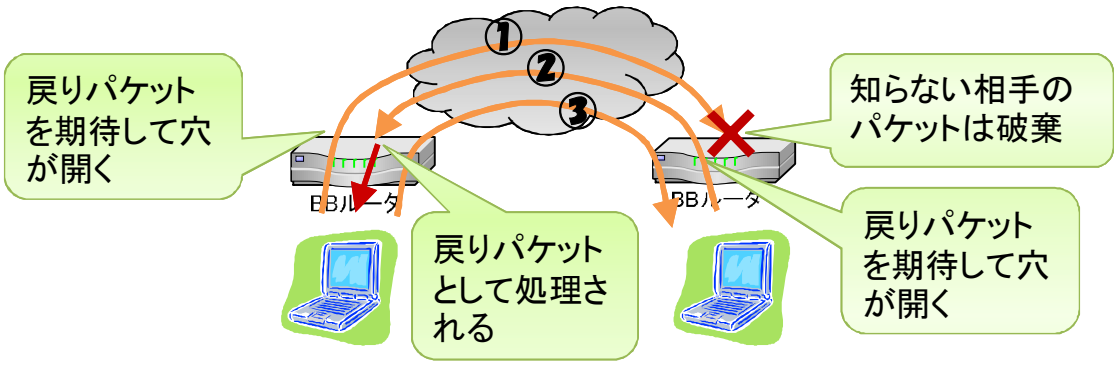
1. はじめに
2. NATの概要
3. 静的NAT　4. 動的NAT
5. NATの挙動をSTUNで調査する
6. NAT越え時のテクニック
7. 事例
8. 今後

6. NAT越え時のテクニック

KONAMI

UDPホールパンチング

- NAT下にいる端末が内部からUDPパケットを送信してNATに穴を開けて、NAT下にいる端末同士が第三者を介さずにUDP通信できるようにすること

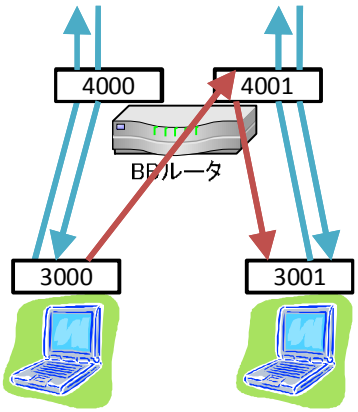


- 注意点
  - 実際のパンチングシーケンスはもう少し複雑
  - ポート制限付きコーン対シンメトリック、シンメトリック対シンメトリックには使えない

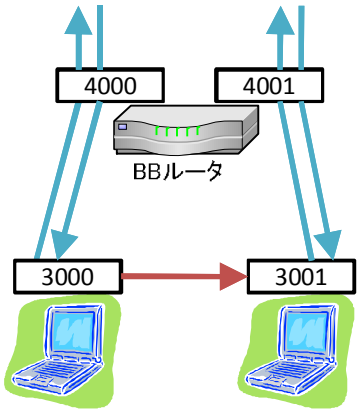
KONAMI

ヘアピンとローカルIPアドレス通信切替

- ヘアピン
  - 同じNAT下のクライアントが外側IPアドレスで通信できる挙動のこと



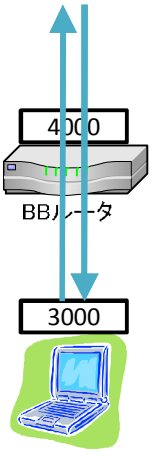
- ローカルIPアドレス通信切替
  - ヘアピンが起こらないように、ローカルIPアドレス通信を使用する



KONAMI

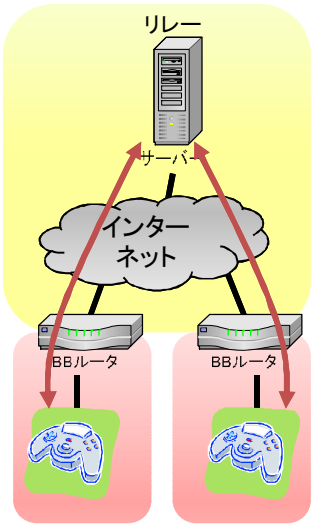
マッピングタイマーリフレッシュ（ハートビート）

- マップが消えてしまうと、別の外側ポートにマップされてしまうかもしれないので、定期的にパケットを打つ必要がある
- リフレッシュ方向
  - 外向き
  - 内向き
  - どちらでも
- ルータのデフォルトタイマー値
  - 設定値が分かるルータ、変更できるルータは少ない
  - 300秒等、意外と短い
- 推奨リフレッシュ間隔
  - TURN（後述）が使用できる場合は、LIFETIME属性を使用する
  - 私の部署では25秒を推奨している
    - サーバから指示できるようにすると良い



KONAMI

その他のNAT越え技術



- リレーサーバ(エージェント)
  - TURN
    - I.D-ietf-behave-turn
      - I.D-ietf-behave-rfc3489bis(次期STUN)を拡張する規格
    - tcp/udpプロトコル変換も含む
  - httpトンネリング
- ICE
  - I.D-ietf-mmusic-ice
    - STUN, TURNを用いたNAT越え手順を総括的にとりまとめた規格
- NAT-PMP
  - I.D.-cheshire-nat-pmp
    - Bonjour(旧称Rendezvous)で使用されているUPnPのIGDに相当する規格

KONAMI

(参考資料)その他の特有の問題(1)

- NAT
  - 同じNAT下に複数台のクライアントがいる場合
    - クライアントが増減した場合のポート番号維持の挙動
    - ポート番号多重してしまう実装
  - マッピング有効期間(ライフタイム)
    - マッピングが自動で切れたのにUPnPのマッピング情報は残っていたり…
    - ルータの電源を入れ直したのにUPnPのマッピング情報は残っていたり…
  - IPペイロード書き換え
    - ペイロードにIPアドレスバイト列を含む場合、(親切に!?)そこも書き換えてしまうNATがある
      - クラシックSTUNで誤判定を引き起こす
- STUN
  - rfc 3489問題点
    - ソースポートを固定してしまうと、誤判定を起こすケースがある

KONAMI

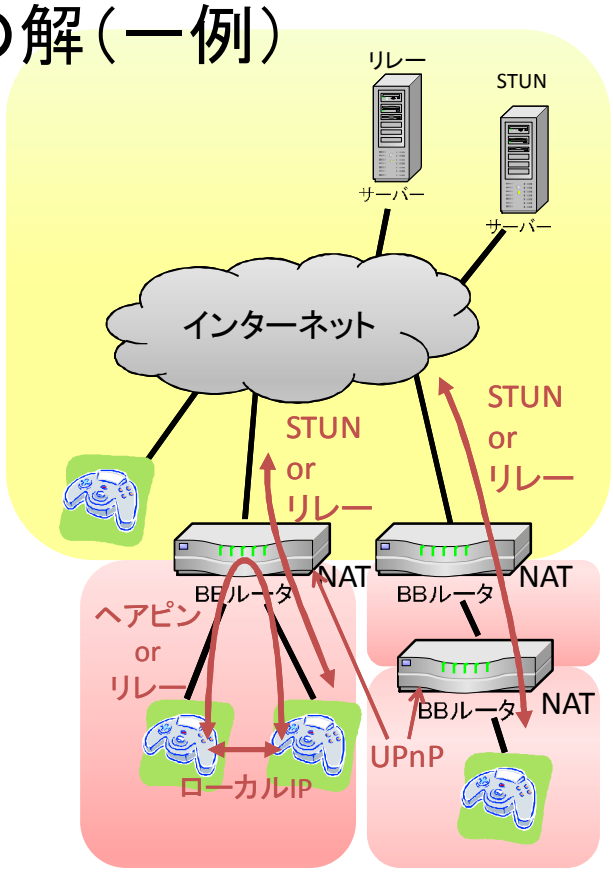
（参考資料）その他の特有の問題（2）

- UPnP
  - 必須な機能が実装されていなかったり
  - 11マッピング目からおかしくなったり
  - Windows XP/Vistaでは1900/udpが既に使われている
    - SSDPのソースポートが1900/udpでないと答えてくれない実装がある
- 無線AP
  - プライバシーセパレータ
    - グローバルIPアドレスなのに通信できなかったりする
- ファイアウォール
  - Windowsファイアウォールやセキュリティソフトのファイアウォール機能
  - ISPのセキュリティサービス

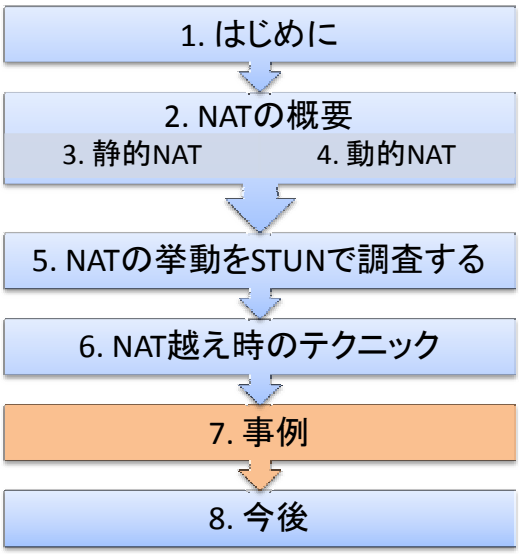
KONAMI

今日のお題の解（一例）

- 結局のところ、NATを越えるには
  どうすれば良いか
  1. UPnPでポートフォワーディングを
     設定する
  2. 次期STUNで調査する
  3. 次のいずれかを選択
     - ローカルアドレスでの通信
     - 外側IPアドレスでの通信
       - UDPホールパンチングが必要な
         場合有り
     - 誰かにリレーしてもらう
       - TURNで中継
       - TCP/UDPプロトコル変換
       - httpでトンネリングする
  4. NATを挟む場合は、要マッピング
     タイマリフレッシュ（ハートビート）



KONAMI



7. 事例

KONAMI

ウイニングイレブンシリーズの実装方法

接続元＼接続先	UDPブロックド	オープンインターネット	シンメトリックUDPファイアウォール	フルコーンNAT	制限付きコーンNAT	ポート制限付きコーンNAT	シンメトリックNAT
UDPブロックド	×	×	×	×	×	×	×
オープンインターネット	×	○※	△※	○	△	△※	△※
シンメトリックUDPファイアウォール	×	○※	×※	○	△	×※	×※
フルコーンNAT	×	○※	△	○	△	△	△
制限付きコーンNAT	×	○※	△	○	△	△	△
ポート制限付きコーンNAT	×	○※	×※	○	△	△	×
シンメトリックNAT	×	○※	×※	○	△	×	×

- 1対1の対戦
- STUNとUPnPのみで、リレーは未実装
- シンメトリック系を除外することで、全クライアントを対等にする
- UDPホールパンチングする

KONAMI

メタルギアシリーズの実装方法

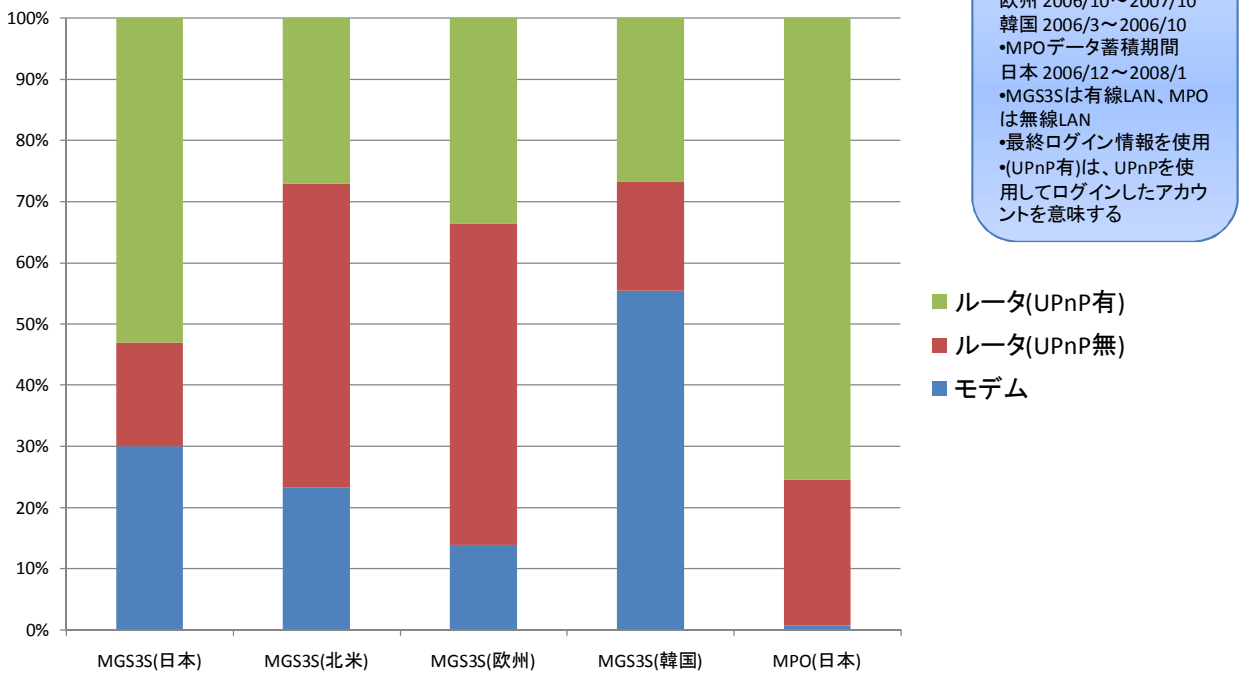
接続元＼接続先	UDPブロックド	オープンインターネット	シンメトリックUDPファイアウォール	フルコーンNAT	制限付きコーンNAT	ポート制限付きコーンNAT	シンメトリックNAT
UDPブロックド	×	×	×	×	×	×	×
オープンインターネット	×	○※	△※	○	△	△※	△※
シンメトリックUDPファイアウォール	×	○※	×※	○	△	×※	×※
フルコーンNAT	×	○※	△	○	△	△	△
制限付きコーンNAT	×	○※	△	○	△	△	△
ポート制限付きコーンNAT	×	○※	×※	○	△	△	×
シンメトリックNAT	×	○※	×※	○	△	×	×

- 多人数対戦
- STUNとUPnPのみで、リレーは未実装
- ホストをオープンインターネットとフルコーンのみにすることで、シンメトリック系も遊べるようにする
- UDPホールパンチングする必要もない
  - MGOでは、さらにUDPホールパンチングして、不完全メッシュにしている

KONAMI

実タイトルでの統計情報(1)

- モデム/ルータ比率、UPnP対応率

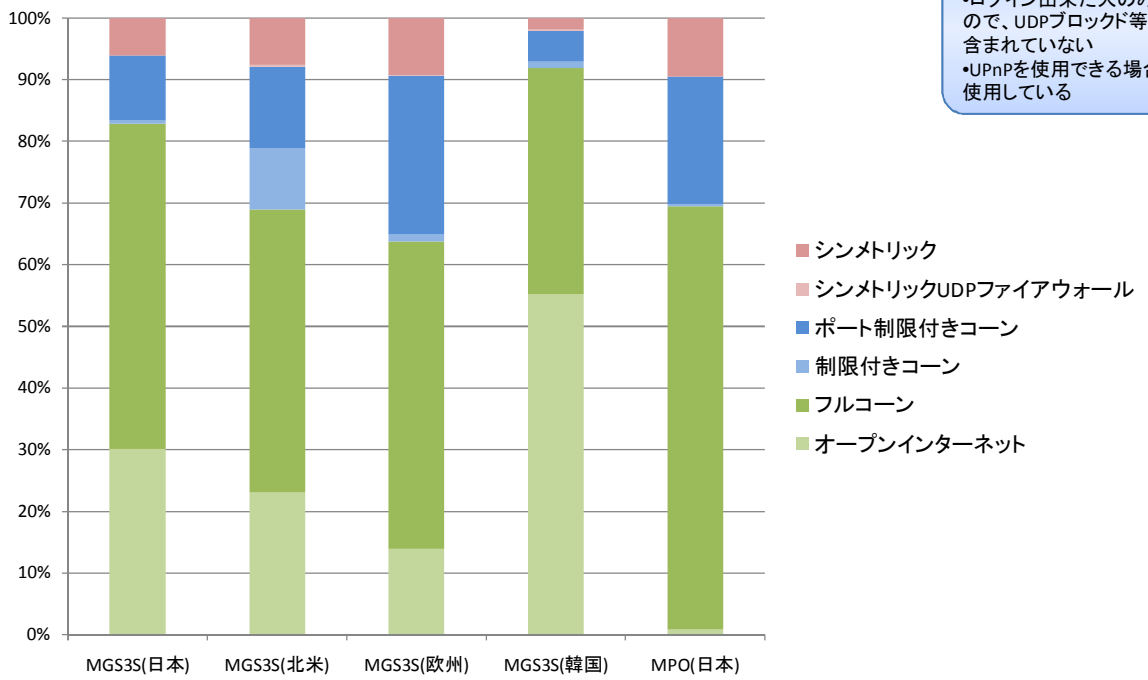


•MGS3Sデータ蓄積期間
日本 2005/12～2006/12
北米 2006/3～2007/4
欧州 2006/10～2007/10
韓国 2006/3～2006/10
•MPOデータ蓄積期間
日本 2006/12～2008/1
•MGS3Sは有線LAN、MPOは無線LAN
•最終ログイン情報を使用
•(UPnP有)は、UPnPを使用してログインしたアカウントを意味する

KONAMI

実タイトルでの統計情報(2)

- NATタイプ



•ログイン出来た人のみなので、UDPブロックド等は含まれていない
•UPnPを使用できる場合は使用している

KONAMI

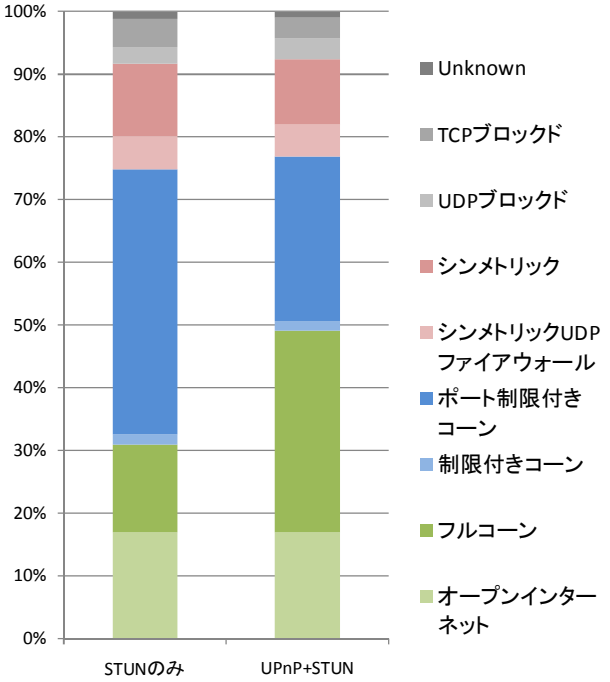
回線検証ツール

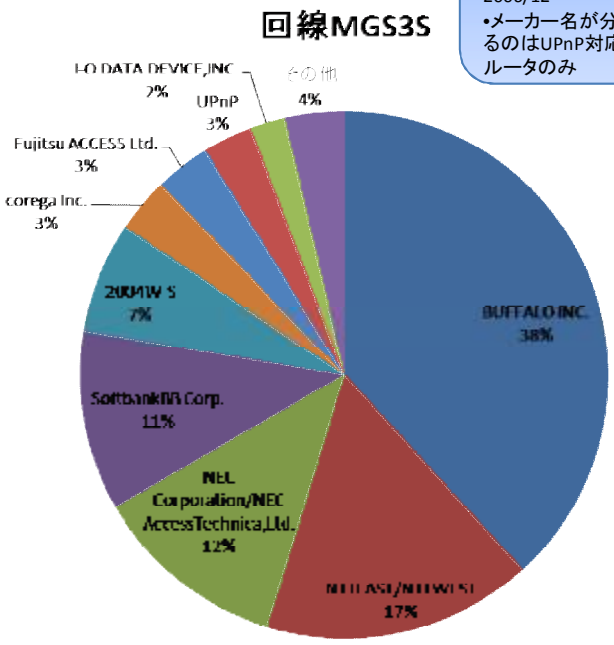
- ゲームソフト製品と同等のNAT越えアルゴリズムで動作するWindows用のプログラムを無償で提供している
- 購入前の確認や購入後のトラブルシュートに利用できる
- チェック結果をサーバへ送信することが出来る

KONAMI

日本版MGS3S用 回線検証ツール(PC)の統計情報(1)

- NATタイプとUPnPの効果
- メーカー比率



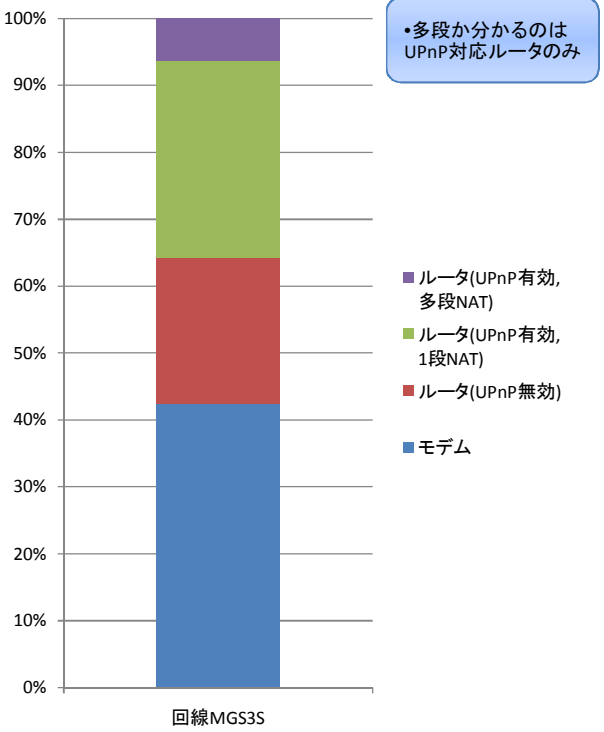


•回線検証ツールのアップロード情報をそのまま集計
•集計期間
2005/12～2006/12
•メーカー名が分かるのはUPnP対応ルータのみ

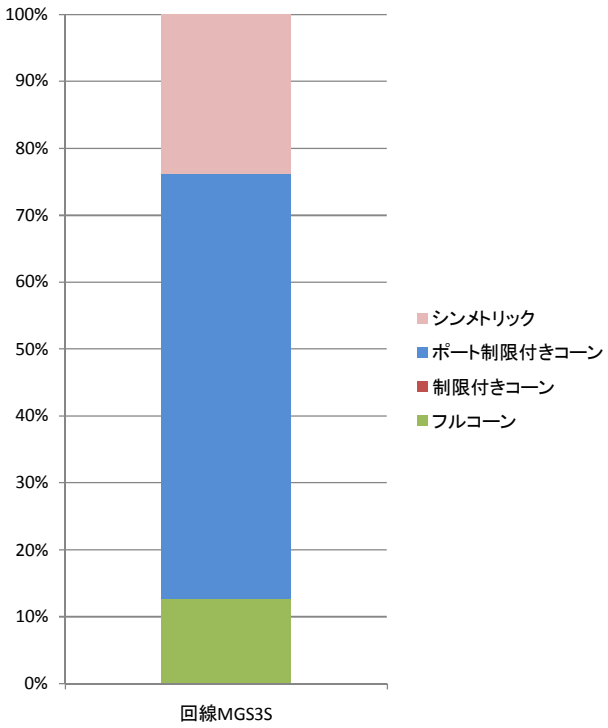
KONAMI

日本版MGS3S用 回線検証ツール(PC)の統計情報(2)

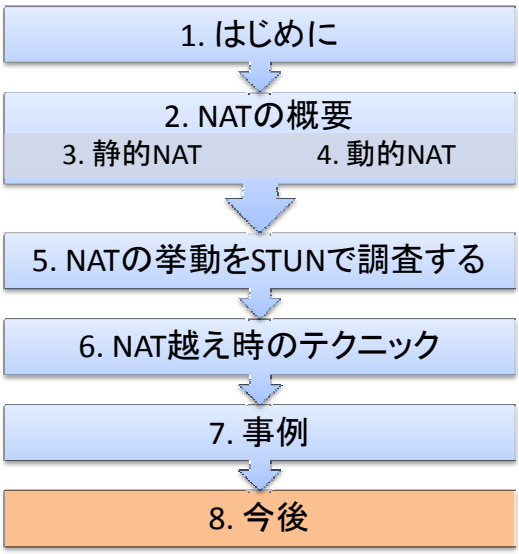
- 多段NAT率
- 多段NAT内訳



•多段か分かるのはUPnP対応ルータのみ



KONAMI



8. 今後

KONAMI

今後の弊社の取り組み

- より多くの方々が遊べるように
  - 自動設定でのNAT越え率を上げる
  - シンメトリック等のNATや、多段NAT環境でも遊べるようにする
  - リレーサーバの開発
    - TURN準拠?
- NAT越え技術で透過的に繋がる環境が出来た後は
  - P2Pアプリの実装
    - 大容量データ配信システム
      - METAL GEAR ONLINEにBitTorrent互換プロトコルを実装してます
    - ゆくゆくは、ゲームサーバの分散化(サーバレス化)...

KONAMI

（参考資料）関連ドキュメント

- RFC
  - rfc 2663 IP Network Address Translator (NAT) Terminology and Considerations
  - rfc 3489 STUN – Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs)
  - rfc 4787 Network Address Translation (NAT) Behavioral Requirements for Unicast UDP
- Internet Draft
  - I.D-ietf-behave-rfc3489bis-13
  - I.D-ietf-behave-nat-behavior-discovery-02
  - I.D-ietf-behave-turn-06
  - I.D-ietf-mmusic-ice-19
  - I.D-cheshire-nat-pmp-02 (expire中)
- UPnP FORUM
  - UPnP Device Architecture
  - Internet Gateway Device(IGD) V 1.0