# NAT（設定）

# 学習内容

1. スタティックNATの概要と設定
2. ダイナミックNATの仕組み
3. PATによるアドレス共有
4. NATテーブルの確認と管理
5. 双方向NAT（Twice NAT）の応用

01

# スタティックNATの概要と設定

# スタティックNATのコンセプト

プライベートIPアドレスを、固定のグローバルIPアドレスへ1対1で変換する仕組みです。

内部のプライベートIPと外部のグローバルIPを**固定的に紐付け**

イメージ: 社員一人ひとりに割り当てられた「**社外用の固定名刺**」

外部から内部の特定サーバーへのアクセスを許可する場合に利用される

**1対1の変換**であるため、グローバルIPアドレスを消費する

# 設定手順の全体像

設定は大きく2つのステップに分かれます。まずアドレスの対応関係を定義し、次に対象インターフェースを指定します。

**STEP 1**

内部アドレスと外部アドレスを1対1で関連付ける

**STEP 2**

変換を適用するインターフェースを「内部(inside)」「外部(outside)」として指定する

# 設定コマンド例 (1) アドレスの対応付け

ip nat inside source static コマンドで、内部と外部のアドレスを紐付けます。

| 基本コマンド書式 | 具体的な設定例 |
| --- | --- |
| (config)# ip nat inside source static <内部ローカルIP> <外部グローバルIP> | (config)# ip nat inside source static 192.168.0.1 100.1.1.1 |

# 設定コマンド例 (2) インターフェースの指定

ip nat inside と ip nat outside コマンドで、各インターフェースの役割を定義します。

| 内部インターフェースの設定 | 外部インターフェースの設定 |
| --- | --- |
| (config-if)# ip nat inside | (config-if)# ip nat outside |

# ポート番号を含むスタティック変換 (PAT)

IPアドレスだけでなく、特定のポート番号だけを変換することも可能です。これを静的PATと呼びます。

**設定コマンド例**

(config)# ip nat inside source static tcp 192.168.0.1 22 100.1.1.1 220 extendable

**動作のポイント**

外部から `100.1.1.1` のポート `220` 宛の通信が、内部の `192.168.0.1` のポート `22` に転送される

**主な用途**

Webサーバーの80番ポートや、SSHの22番ポートなど、特定のサービスのみを外部に公開する際に利用

02

# ダイナミックNATの仕組み

# ダイナミックNATのコンセプト

複数のグローバルIPアドレスをプールとして用意し、内部からの通信要求に応じて動的に割り当てます。

事前にグローバルIPアドレスの範囲（プール）を定義

イメージ: 会社で共用されている「**フリーアドレスのデスク**」

内部からの通信開始時に、プール内の**空いているアドレスが自動で割り当てられる**

同時接続数は、**プール内のアドレス数に制限**される

# ダイナミックNATの設定手順

設定は4つのステップで構成されます。プール、変換対象、関連付け、そしてインターフェース指定です。

| アドレスプール作成 | ACL定義 | ACLとプールの関連付け | インターフェース指定 |
| --- | --- | --- | --- |
| 外部に公開するグローバルアドレスの範囲を定義 | NAT変換の対象となる内部ローカルアドレスをACLで指定 | 作成したACLとアドレスプールを結びつける | inside / outside を各インターフェースに設定 |

# スタティックNATとダイナミックNATの動作比較

両者の最も大きな違いは、NATテーブルに変換情報が登録されるタイミングです。

| スタティックNAT | ダイナミックNAT |
| --- | --- |
| 設定を投入した時点でNATテーブルに変換情報が**静的に登録**される | **内部から外部への通信が発生した時点**で、初めてNATテーブルに変換情報が動的に登録される |
| 通信が発生していなくてもテーブルに表示 | 通信がない状態ではテーブルは空 |
| 外部からの通信開始が可能（サーバー公開など） | 原則、外部から直接通信を開始することはできない |

03

# PATによるアドレス共有

# PAT (Port Address Translation) のコンセプト

1つのグローバルIPアドレスを、ポート番号を使い分けることで複数の内部端末が共有する仕組みです。

NAPT (Network Address Port Translation) とも呼ばれる

イメージ: 会社や部署の「**代表電話番号と内線番号**」

送信元ポート番号をユニークな値に書き換えることで、戻りの通信を正しく振り分ける

**1つのグローバルIPで多数の端末が同時接続可能**となり、IPアドレスを大幅に節約できる

# PATの設定手順（インターフェース指定）

最も一般的な、ルータの外部インターフェースIPアドレスを利用するPATの設定手順です。

1. NAT変換対象の内部アドレスをACLで定義する
2. ACLと外部インターフェースを関連付け、末尾に `overload` を指定する
3. 各インターフェースに `ip nat inside` / `ip nat outside` を設定する

# PAT設定の最重要キーワード: overload

`overload` オプションを付けることで、1つのグローバルIPを複数端末で共有するPATとして動作します。

**コマンド設定例**

```
(config)# ip nat inside source list 1 interface GigabitEthernet0/0 overload
```

04

# NATテーブルの確認と管理

# NAT方式ごとのテーブル表示の違い

`show ip nat translations`コマンドの出力は、NATの方式によって特徴が異なります。

| NAT方式 | テーブル登録のタイミング | 出力の特徴 |
| --- | --- | --- |
| スタティックNAT | 設定投入時 | 通信がなくても常に静的なエントリが表示される |
| ダイナミックNAT | 内部からの通信開始時 | 通信中の動的なエントリのみ表示される（IPアドレスのみ） |
| PAT | 内部からの通信開始時 | 通信中の動的なエントリが表示される（**IPアドレス＋ポート番号**） |

# 統計情報の確認: show ip nat statistics

NAT変換の成功・失敗回数などの統計情報を確認できます。

Total active translations

**3**

現在の変換数

Hits

**559**

**変換成功回数**

Misses

**130**

**変換失敗回数**

# 変換エントリのクリアとタイムアウト

動的な変換エントリは手動でクリアするか、一定時間通信がない場合に自動で削除されます。

### 手動クリア

`clear ip nat translations *` コマンドで、すべての動的エントリを強制的に削除できる

### タイムアウト

通信がない状態が続くとエントリは自動削除

デフォルト値はプロトコル毎に異なる

例: UDP (300秒), DNS (60秒), TCP (24時間)

05

# 双方向NAT（Twice NAT）の応用

# 双方向NAT (Twice NAT) のコンセプト

パケットがルータを通過する際に、送信元アドレスと宛先アドレスの両方を同時に変換する技術です。

**内側→外側**: 送信元IPアドレスを変換 (通常のNAT)

**外側→内側**: 宛先IPアドレスを変換

結果として、内部ホストと外部ホストが、お互いに相手の**本当のIPアドレスを知ることなく**通信できる

IPアドレス空間が重複しているネットワーク同士を接続する際などに利用される

# 双方向NATの設定の考え方

「内側用」と「外側用」の2つのスタティックNAT設定を組み合わせることで実現します。

## ip nat inside source static

**内側ホスト**の送信元アドレスを変換するための設定

内側ホストが外側と通信する際に使用される

## ip nat outside source static

**外側ホスト**の送信元アドレスを変換するための設定

内側ホストから見た「仮想的な宛先IP」を定義する

# 双方向NATのポイント: add-route

ip nat outside source static 設定には、到達性を確保するための工夫が必要です。

**Q. なぜ `add-route` オプションやスタティックルートが必要なのですか？**

A. ルータが、内側に見せるための「仮想的なIPアドレス」を、自分自身が所有しているアドレスだと認識するために必要です。この設定がないと、ルータはその仮想IP宛のパケットをどこに送ればよいか分からなくなってしまいます。