FutureNet NXR シリーズ

設定例集

Ver 1.2.0

※上記写真は NXR-130/C です。

センチュリー・システムズ株式会社

目次

はじめに

- FutureNet はセンチュリー・システムズ株式会社の登録商標です。
- 本書に記載されている会社名，製品名は、各社の商標および登録商標です。
- 本ガイドは、以下の FutureNet NXR 製品に対応しております。
  - NXR-130/C
- 本書の内容の一部または全部を無断で転載することを禁止しています。
- 本書の内容については、将来予告なしに変更することがあります。
- 本書の内容については万全を期しておりますが、ご不審な点や誤り、記載漏れ等お気づきの点がありましたらお手数ですが、ご一報下さいますようお願い致します。
- 本書は FutureNet NXR シリーズ NXR-130/C の以下のバージョンをベースに作成しております。

  第１章～第６章 FutureNet NXR シリーズ NXR-130/C Ver5.1.0

  第７章 FutureNet NXR シリーズ NXR-130/C Ver5.1.1

  第８章～第９章 FutureNet NXR シリーズ NXR-130/C Ver5.1.2

  各種機能において、ご使用されている製品およびファームウェアのバージョンによっては、一部機能，コマンドおよび設定画面が異なっている場合もありますので、その場合は各製品のユーザーズガイドを参考に、適宜読みかえてご参照および設定を行って下さい。
- 本バージョンでは IPv4 のみを対象とし、IPv6 の設定に関しては本バージョンでは記載しておりません。
- 設定した内容の復帰（流し込み）を行う場合は、CLI では「copy」コマンド，GUI では設定の復帰を行う必要があります。
- 本書を利用し運用した結果発生した問題に関しましては、責任を負いかねますのでご了承下さい。

改版履歴

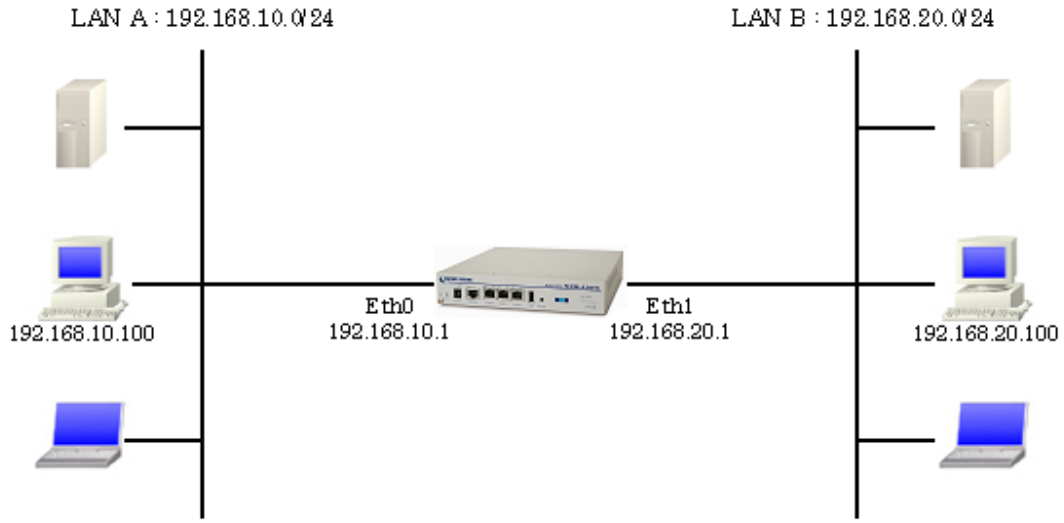
Version	更新内容
1.0.0	初版
1.1.0	IPsec 設定例追加
1.2.0	L2TPv3,VPN 応用設定例追加

1. インタフェース設定

1-1. ローカルルータ設定

LAN A「192.168.10.0/24」と LAN B「192.168.20.0/24」のネットワークを接続し、ローカルルータとして利用するための設定をします。

1-1-1. 構成図



1-1-2. 設定例

```
nxr130#configure terminal
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
nxr130(config-if)#exit
nxr130(config)#interface ethernet 1
nxr130(config-if)#ip address 192.168.20.1/24
nxr130(config-if)#exit
nxr130(config)#exit
nxr130#save config
```

【 解説 】

- Ethernet0 インタフェース側を LAN A「192.168.10.0/24」, Ethernet1 インタフェース側を LAN B「192.168.20.0/24」とします。
- インタフェース設定でそれぞれのネットワークに属する IPv4 アドレスをルータに設定します。
- IP アドレスの設定を変更した場合、その設定した IP アドレスが即反映されます。

＜Ethernet0 インタフェース設定＞

```
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.1/24」を設定します。

＜Ethernet1 インタフェース設定＞

```
nxr130(config)#interface ethernet 1
nxr130(config-if)#ip address 192.168.20.1/24
```

Ethernet1 インタフェースの IPv4 アドレスとして「192.168.20.1/24」を設定します。

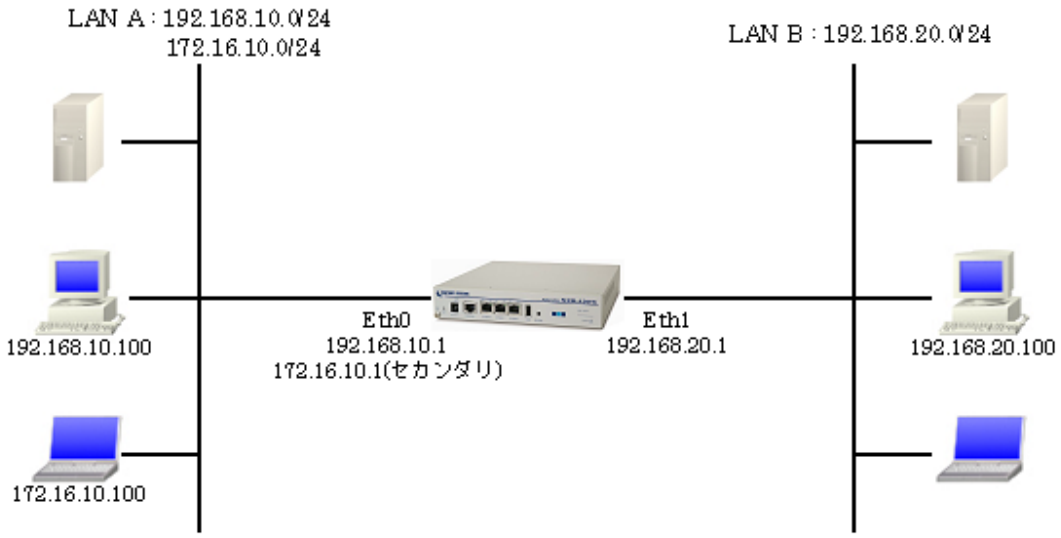
1-1-3. パソコンの設定例

	LAN A のパソコン	LAN B のパソコン
IPv4 アドレス	192.168.10.100	192.168.20.100
サブネットマスク	255.255.255.0	255.255.255.0
デフォルトゲートウェイ	192.168.10.1	192.168.20.1

1-2. セカンダリアドレス設定

物理的に 1 つのインタフェースに、複数の IPv4 アドレスを割り当てることができます。

1-2-1. 構成図



1-2-2. 設定例

```
nxr130#configure terminal
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
nxr130(config-if)#ip address 172.16.10.1/24 secondary
nxr130(config-if)#exit
nxr130(config)#interface ethernet 1
nxr130(config-if)#ip address 192.168.20.1/24
nxr130(config-if)#exit
nxr130(config)#exit
nxr130#save config
```

【 解説 】

- Ethernet0 インタフェースで「172.16.10.0/24」のネットワークアドレスが利用できるようにセカンダリアドレスを設定します。
- IP アドレスの設定を変更した場合、その設定した IP アドレスが即反映されます。

＜Ethernet0 インタフェース設定＞

```
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.1/24」を設定します。

```
nxr130(config-if)#ip address 172.16.10.1/24 secondary
```

Ethernet0 インタフェースのセカンダリ IPv4 アドレスとして「172.16.10.1/24」を設定します。

＜Ethernet1 インタフェース設定＞

```
nxr130(config)#interface ethernet 1
nxr130(config-if)#ip address 192.168.20.1/24
```

Ethernet1 インタフェースの IPv4 アドレスとして「192.168.20.1/24」を設定します。

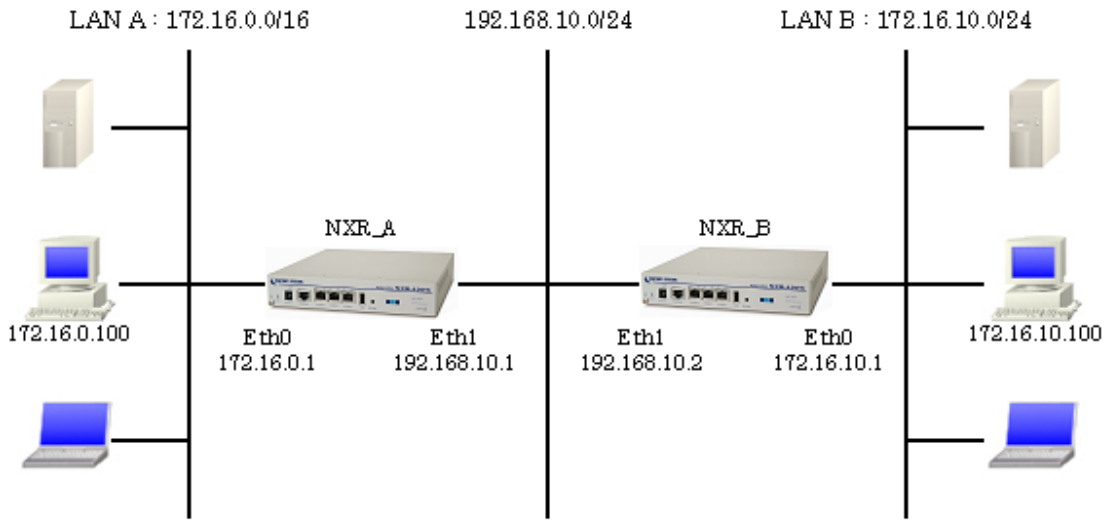
1-2-3. パソコンの設定例

	LAN A のパソコン (192.168.10.0/24)	LAN A のパソコン (172.16.10.0/24)	LAN B のパソコン
IPv4 アドレス	192.168.10.100	172.16.10.100	192.168.20.100
サブネットマスク	255.255.255.0	255.255.255.0	255.255.255.0
デフォルトゲートウェイ	192.168.10.1	172.16.10.1	192.168.20.1

1-3. プロキシ ARP 設定

プロキシ ARP は他のホスト宛への ARP 要求に対して、ルータが代理で ARP 応答する機能です。
サブネットマスクを設定することができないホストが存在し、そのホストが通信を行う際に、利用されます。

1-3-1. 構成図



1-3-2. 設定例

[NXR_A の設定]

```
nxr130#configure terminal
nxr130(config)#hostname NXR_A
NXR_A(config)#interface ethernet 0
NXR_A(config-if)#ip address 172.16.0.1/16
NXR_A(config-if)#ip proxy-arp
NXR_A(config-if)#exit
NXR_A(config)#interface ethernet 1
NXR_A(config-if)#ip address 192.168.10.1/24
NXR_A(config-if)#exit
NXR_A(config)#ip route 172.16.10.0/24 192.168.10.2
NXR_A(config)#exit
NXR_A#save config
```

【 解説 】

- Ethernet0 インタフェースでプロキシ ARP を有効にし、ARP の代理応答を可能にします。
- LAN B「172.16.10.0/24」宛のスタティックルートを設定します。

＜ホスト名の設定＞

```
nxr130(config)#hostname NXR_A
```

ホスト名として「NXR_A」を設定します。

＜Ethernet0 インタフェース設定＞

```
NXR_A(config)#interface ethernet 0
NXR_A(config-if)#ip address 172.16.0.1/16
```

Ethernet0 インタフェースの IPv4 アドレスとして「172.16.0.1/16」を設定します。

```
NXR_A(config-if)#ip proxy-arp
```

Ethernet0 インタフェースでプロキシ ARP を有効に設定します。

＜Ethernet1 インタフェース設定＞

```
NXR_A(config)#interface ethernet 1
NXR_A(config-if)#ip address 192.168.10.1/24
```

Ethernet1 インタフェースの IPv4 アドレスとして「192.168.10.1/24」を設定します。

＜スタティックルート設定＞

```
NXR_A(config)#ip route 172.16.10.0/24 192.168.10.2
```

LAN B の「172.16.10.0/24」宛のパケットは 192.168.10.2 宛に転送するように設定します。

[NXR_B の設定]

```
nxr130#configure terminal
nxr130(config)#hostname NXR_B
NXR_B(config)#interface ethernet 0
NXR_B(config-if)#ip address 172.16.10.1/24
NXR_B(config-if)#exit
NXR_B(config)#interface ethernet 1
NXR_B(config-if)#ip address 192.168.10.2/24
NXR_B(config-if)#exit
NXR_B(config)#ip route 172.16.0.0/16 192.168.10.1
NXR_B(config)#exit
NXR_B#save config
```

＜ホスト名の設定＞

```
nxr130(config)#hostname NXR_B
```

ホスト名として「NXR_B」を設定します。

＜Ethernet0 インタフェース設定＞

```
NXR_B(config)#interface ethernet 0
NXR_B(config-if)#ip address 172.16.10.1/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「172.16.10.1/24」を設定します。

＜Ethernet1 インタフェース設定＞

```
NXR_B(config)#interface ethernet 1
NXR_B(config-if)#ip address 192.168.10.2/24
```

Ethernet1 インタフェースの IPv4 アドレスとして「192.168.10.2/24」を設定します。

＜スタティックルート設定＞

```
NXR_B(config)#ip route 172.16.0.0/16 192.168.10.1
```

LAN A の「172.16.0.0/16」宛のパケットは 192.168.10.1 宛に転送するように設定します。

1-3-3. パソコンの設定例

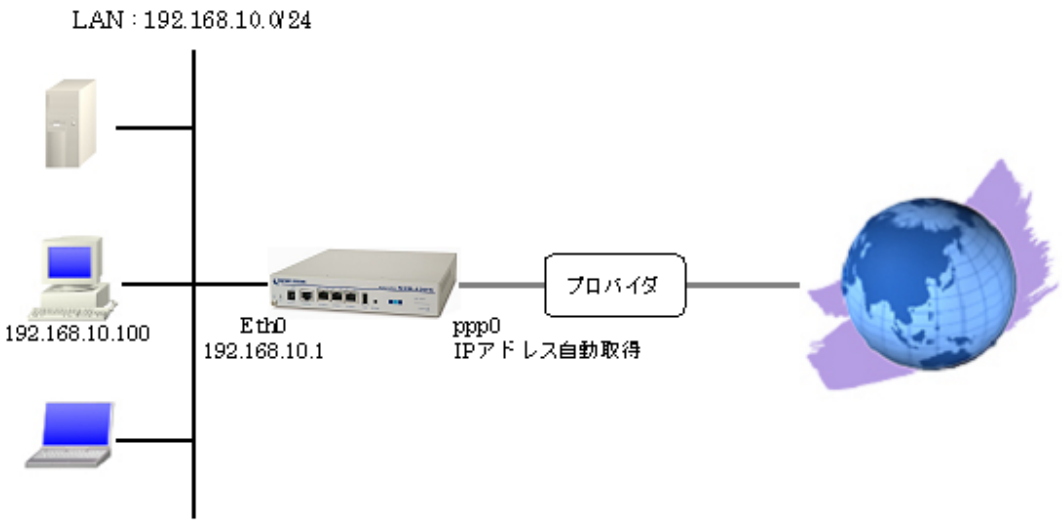
	LAN A のパソコン	LAN B のパソコン
IPv4 アドレス	172.16.0.100	172.16.10.100
サブネットマスク	255.255.0.0	255.255.255.0
デフォルトゲートウェイ	172.16.0.1	172.16.10.1

2. PPPoE 設定

2-1. 端末型接続設定

フレッツ ADSL やＢフレッツなど PPPoE 接続を必要とする環境で、IP アドレスを 1 つ利用できるサービスで利用可能な設定です。

2-1-1. 構成図



2-1-2. 設定例

```
nxr130#configure terminal
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
nxr130(config-if)#exit
nxr130(config)#interface ppp 0
nxr130(config-ppp)#ip address negotiated
nxr130(config-ppp)#ip masquerade
nxr130(config-ppp)#ip spi-filter
nxr130(config-ppp)#ip tcp adjust-mss auto
nxr130(config-ppp)#no ip redirects
nxr130(config-ppp)#ppp username test1@centurysys password test1pass
nxr130(config-ppp)#exit
nxr130(config)#interface ethernet 1
nxr130(config-if)#no ip address
nxr130(config-if)#pppoe-client ppp 0
nxr130(config-if)#exit
nxr130(config)#ip route 0.0.0.0/0 ppp 0
nxr130(config)#dns
nxr130(dns-config)#service enable
nxr130(dns-config)#exit
nxr130(config)#exit
nxr130#save config
```

【 解説 】

- Ethernet1 インタフェースで ppp0 インタフェースを利用します。
- ppp0 インタフェースで IP マスカレード,ステートフルパケットインスペクションを有効にします。
- DNS サービスを有効にします。

＜Ethernet0 インタフェース設定＞

```
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.1/24」を設定します。

＜ppp0 インタフェース設定＞

```
nxr130(config)#interface ppp 0
```

ppp0 インタフェースに関する設定をします。

```
nxr130(config-ppp)#ip address negotiated
```

IPCP で IP アドレスを取得するように設定します。

```
nxr130(config-ppp)#ip masquerade
```

IP マスカレードを設定します。

```
nxr130(config-ppp)#ip spi-filter
```

ステートフルパケットインスペクションを設定します。

```
nxr130(config-ppp)#ip tcp adjust-mss auto
```

TCP MSS の調整機能をオートに設定します。

```
nxr130(config-ppp)#no ip redirects
```

ICMP リダイレクト機能を無効に設定します。

```
nxr130(config-ppp)#ppp username test1@centurysys password test1pass
```

PPPoE 接続で使用するユーザ ID とパスワードを設定します。

＜Ethernet1 インタフェース設定＞

```
nxr130(config)#interface ethernet 1
```

Ethernet1 インタフェースに関する設定をします。

```
nxr130(config-if)#no ip address
```

Ethernet1 インタフェースに IPv4 アドレスを割り当てない設定をします。

```
nxr130(config-if)#pppoe-client ppp 0
```

Ethernet1 インタフェース上で ppp0 インタフェースを使用するための設定をします。

＜スタティックルート設定＞

```
nxr130(config)#ip route 0.0.0.0/0 ppp 0
```

デフォルトルートを設定します。

＜DNS 設定＞

```
nxr130(config)#dns
```

DNS に関する設定をします。

```
nxr130(dns-config)#service enable
```

DNS サービスを有効にします。

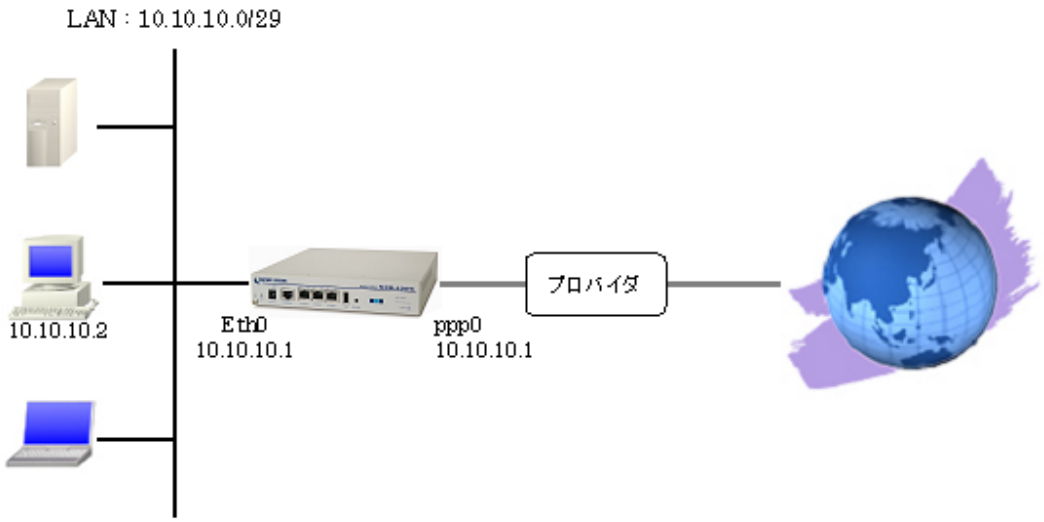
2-1-3. パソコンの設定例

IPv4 アドレス	192.168.10.100
サブネットマスク	255.255.255.0
デフォルトゲートウェイ	192.168.10.1
DNS サーバの IPv4 アドレス	192.168.10.1

2-2. LAN 型接続設定

フレッツ ADSL やＢフレッツなど PPPoE 接続を必要とする環境で、IPv4 アドレスを複数利用可能な場合、ルータの LAN 側にもグローバル IPv4 アドレスを割り当てることができます。

2-2-1. 構成図



2-2-2. 設定例

```
nxr130#configure terminal
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 10.10.10.1/29
nxr130(config-if)#exit
nxr130(config)#interface ppp 0
nxr130(config-ppp)#ip address 10.10.10.1/32
nxr130(config-ppp)#ip tcp adjust-mss auto
nxr130(config-ppp)#no ip redirects
nxr130(config-ppp)#ppp username test1@centurysys password test1pass
nxr130(config-ppp)#exit
nxr130(config)#interface ethernet 1
nxr130(config-if)#no ip address
nxr130(config-if)#pppoe-client ppp 0
nxr130(config-if)#exit
nxr130(config)#ip route 0.0.0.0/0 ppp 0
nxr130(config)#dns
nxr130(dns-config)#service enable
nxr130(dns-config)#exit
nxr130(config)#exit
nxr130#save config
```

【 解説 】

- Ethernet1 インタフェースで ppp0 インタフェースを利用します。
- ppp0 インタフェースで IP アドレスを固定で割り当て、ppp0 インタフェースで割り当てた IP アドレスと同じアドレスを Ethernet0 インタフェースにも設定します。
- ステートフルパケットインスペクションを無効に設定していますので、個別にフィルタリングルールを作成する必要があります。※ここではフィルタリングルールは作成していません。
- DNS サービスを有効にします。

＜Ethernet0 インタフェース設定＞

```
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 10.10.10.1/29
```

Ethernet0 インタフェースの IPv4 アドレスとして「10.10.10.1/29」を設定します。

＜ppp0 インタフェース設定＞

```
nxr130(config)#interface ppp 0
```

ppp0 インタフェースに関する設定をします。

```
nxr130(config-ppp)#ip address 10.10.10.1/32
```

ppp0 インタフェースの IPv4 アドレスとして「10.10.10.1/32」を設定します。

```
nxr130(config-ppp)#ip tcp adjust-mss auto
```

TCP MSS の調整機能をオートに設定します。

```
nxr130(config-ppp)#no ip redirects
```

ICMP リダイレクト機能を無効に設定します。

```
nxr130(config-ppp)#ppp username test1@centurysys password test1pass
```

PPPoE 接続で使用するユーザ ID とパスワードを設定します。

＜Ethernet1 インタフェース設定＞

```
nxr130(config)#interface ethernet 1
```

Ethernet1 インタフェースに関する設定をします。

```
nxr130(config-if)#no ip address
```

Ethernet1 インタフェースに IPv4 アドレスを割り当てない設定をします。

```
nxr130(config-if)#pppoe-client ppp 0
```

Ethernet1 インタフェース上で ppp0 インタフェースを使用するための設定をします。

＜スタティックルート設定＞

```
nxr130(config)#ip route 0.0.0.0/0 ppp 0
```

デフォルトルートを設定します。

＜DNS 設定＞

```
nxr130(config)#dns
```

DNS に関する設定をします。

```
nxr130(dns-config)#service enable
```

DNS サービスを有効にします。

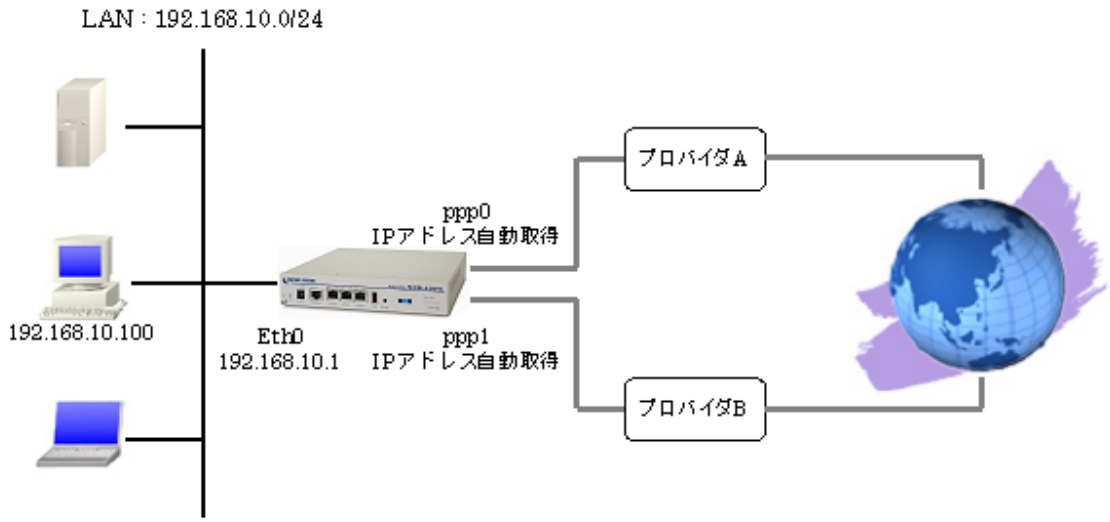
2-2-3. パソコンの設定例

IPv4 アドレス	10.10.10.2
サブネットマスク	255.255.255.248
デフォルトゲートウェイ	10.10.10.1
DNS サーバの IPv4 アドレス	10.10.10.1

2-3. マルチセッション接続設定

Ｂフレッツなどでは同時に複数の PPPoE 接続を行うことが可能です。これにより複数のプロバイダに接続し利用することも可能です。

2-3-1. 構成図



2-3-2. 設定例

```
nxr130#configure terminal
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
nxr130(config-if)#exit
nxr130(config)#interface ppp 0
nxr130(config-ppp)#ip address negotiated
nxr130(config-ppp)#ip masquerade
nxr130(config-ppp)#ip spi-filter
nxr130(config-ppp)#ip tcp adjust-mss auto
nxr130(config-ppp)#no ip redirects
nxr130(config-ppp)#ppp username test1@centurysys password test1pass
nxr130(config-ppp)#exit
nxr130(config)#interface ppp 1
nxr130(config-ppp)#ip address negotiated
nxr130(config-ppp)#ip masquerade
nxr130(config-ppp)#ip spi-filter
nxr130(config-ppp)#ip tcp adjust-mss auto
nxr130(config-ppp)#no ip redirects
nxr130(config-ppp)#ppp username test2@centurysys password test2pass
nxr130(config-ppp)#exit
nxr130(config)#interface ethernet 1
nxr130(config-if)#no ip address
nxr130(config-if)#pppoe-client ppp 0
nxr130(config-if)#pppoe-client ppp 1
nxr130(config-if)#exit
nxr130(config)#ip route 0.0.0.0/0 ppp 0
nxr130(config)#ip route 10.100.0.0/24 ppp 1
nxr130(config)#dns

----- 次のページに続きがあります -----
```

```
----- 前のページからの続きです -----

nxr130(dns-config)#service enable
nxr130(dns-config)#exit
nxr130(config)#exit
nxr130#save config
```

【 解説 】

- Ethernet1 インタフェースで ppp0, 1 インタフェースを利用します。
- ppp0, 1 インタフェースでそれぞれ IP マスカレード,ステートフルパケットインスペクションを有効にします。
- ここでは ppp0 インタフェースをデフォルトルートとし、宛先 IP アドレスが「10.100.0.0/24」の時には ppp1 インタフェースを利用するように設定します。
- DNS サービスを有効にします。

＜Ethernet0 インタフェース設定＞

```
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.1/24」を設定します。

＜ppp0 インタフェース設定＞

```
nxr130(config)#interface ppp 0
nxr130(config-ppp)#ip address negotiated
nxr130(config-ppp)#ip masquerade
nxr130(config-ppp)#ip spi-filter
nxr130(config-ppp)#ip tcp adjust-mss auto
nxr130(config-ppp)#no ip redirects
nxr130(config-ppp)#ppp username test1@centurysys password test1pass
```

ppp0 インタフェースで PPPoE に関する設定をします。

＜ppp1 インタフェース設定＞

```
nxr130(config)#interface ppp 1
nxr130(config-ppp)#ip address negotiated
nxr130(config-ppp)#ip masquerade
nxr130(config-ppp)#ip spi-filter
nxr130(config-ppp)#ip tcp adjust-mss auto
nxr130(config-ppp)#no ip redirects
nxr130(config-ppp)#ppp username test2@centurysys password test2pass
```

ppp1 インタフェースで PPPoE に関する設定をします。

＜Ethernet1 インタフェース設定＞

```
nxr130(config)#interface Ethernet 1
```

Ethernet1 インタフェースに関する設定をします。

```
nxr130(config-if)#no ip address
```

Ethernet1 インタフェースに IPv4 アドレスを割り当てない設定をします。

```
nxr130(config-if)#pppoe-client ppp 0
nxr130(config-if)#pppoe-client ppp 1
```

Ethernet1 インタフェース上で ppp0, 1 インタフェースを使用するための設定をします。

＜スタティックルート設定＞

```
nxr130(config)#ip route 0.0.0.0/0 ppp 0
```

ppp0 インタフェースをデフォルトルートとする設定をします。

```
nxr130(config)#ip route 10.100.0.0/24 ppp 1
```

宛先 IP アドレスが「10.100.0.0/24」の時には ppp1 インタフェースを利用するように設定します。

＜DNS 設定＞

```
nxr130(config)#dns
nxr130(dns-config)#service enable
```

DNS サービスを有効にします。

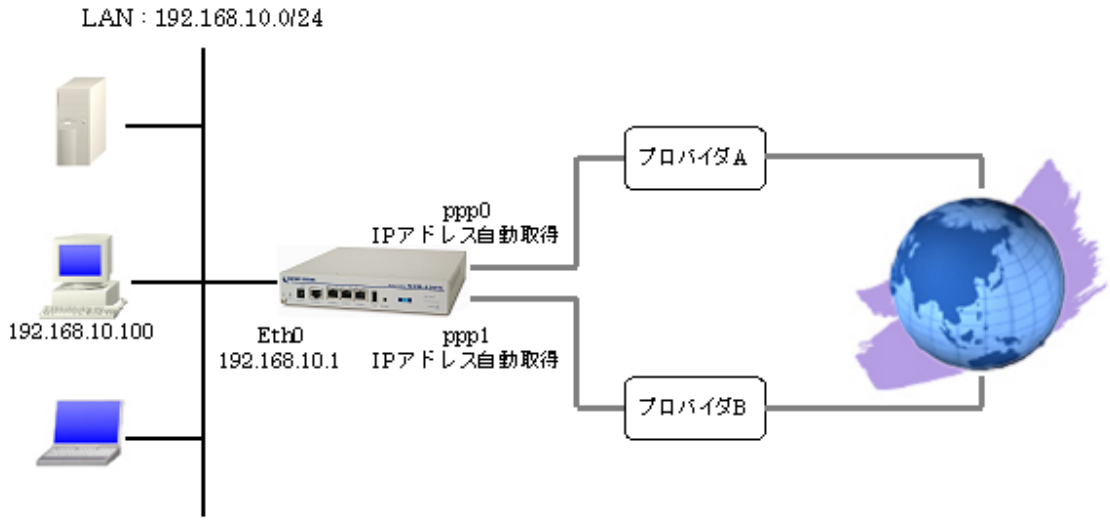
2-3-3. パソコンの設定例

IPv4 アドレス	192.168.10.100
サブネットマスク	255.255.255.0
デフォルトゲートウェイ	192.168.10.1
DNS サーバの IPv4 アドレス	192.168.10.1

2-4. ECMP 設定

同じ宛先に対して同じコストのルートを複数設定する ECMP (Equal Cost Multi Path) を利用することが可能です。
ルートは送信元/宛先の組み合わせによって決定します。
どちらかの回線で通信障害が発生した場合は、通信可能な回線だけ利用して通信します。

2-4-1. 構成図



2-4-2. 設定例

```
nxr130#configure terminal
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
nxr130(config-if)#exit
nxr130(config)#interface ppp 0
nxr130(config-ppp)#ip address negotiated
nxr130(config-ppp)#ip masquerade
nxr130(config-ppp)#ip spi-filter
nxr130(config-ppp)#ip tcp adjust-mss auto
nxr130(config-ppp)#no ip redirects
nxr130(config-ppp)#ppp username test1@centurysys password test1pass
nxr130(config-ppp)#exit
nxr130(config)#interface ppp 1
nxr130(config-ppp)#ip address negotiated
nxr130(config-ppp)#ip masquerade
nxr130(config-ppp)#ip spi-filter
nxr130(config-ppp)#ip tcp adjust-mss auto
nxr130(config-ppp)#no ip redirects
nxr130(config-ppp)#ppp username test2@centurysys password test2pass
nxr130(config-ppp)#exit
nxr130(config)#interface ethernet 1
nxr130(config-if)#no ip address
nxr130(config-if)#pppoe-client ppp 0
nxr130(config-if)#pppoe-client ppp 1
nxr130(config-if)#exit

----- 次のページに続きがあります -----
```

```
----- 前のページからの続きです -----

nxr130(config)#ip route 0.0.0.0/0 ppp 0 1
nxr130(config)#ip route 0.0.0.0/0 ppp 1 1
nxr130(config)#dns
nxr130(dns-config)#service enable
nxr130(dns-config)#exit
nxr130(config)#exit
nxr130#save config
```

【 解説 】

- Ethernet1 インタフェースで ppp0, 1 インタフェースを利用します。
- ppp0, 1 インタフェースでそれぞれ IP マスカレード,ステートフルパケットインスペクションを有効にします。
- デフォルトルートを ppp0, 1 インタフェースで設定し、共にディスタンス値を「1」とします。
- DNS サービスを有効にします。

＜Ethernet0 インタフェース設定＞

```
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.1/24」を設定します。

＜ppp0 インタフェース設定＞

```
nxr130(config)#interface ppp 0
nxr130(config-ppp)#ip address negotiated
nxr130(config-ppp)#ip masquerade
nxr130(config-ppp)#ip spi-filter
nxr130(config-ppp)#ip tcp adjust-mss auto
nxr130(config-ppp)#no ip redirects
nxr130(config-ppp)#ppp username test1@centurysys password test1pass
```

ppp0 インタフェースで PPPoE に関する設定をします。

＜ppp1 インタフェース設定＞

```
nxr130(config)#interface ppp 1
nxr130(config-ppp)#ip address negotiated
nxr130(config-ppp)#ip masquerade
nxr130(config-ppp)#ip spi-filter
nxr130(config-ppp)#ip tcp adjust-mss auto
nxr130(config-ppp)#no ip redirects
nxr130(config-ppp)#ppp username test2@centurysys password test2pass
```

ppp1 インタフェースで PPPoE に関する設定をします。

＜Ethernet1 インタフェース設定＞

```
nxr130(config)#interface Ethernet 1
```

Ethernet1 インタフェースに関する設定をします。

```
nxr130(config-if)#no ip address
```

Ethernet1 インタフェースに IPv4 アドレスを割り当てない設定をします。

```
nxr130(config-if)#pppoe-client ppp 0
nxr130(config-if)#pppoe-client ppp 1
```

Ethernet1 インタフェース上で ppp0，1 インタフェースを使用するための設定をします。

＜スタティックルート設定＞

```
nxr130(config)#ip route 0.0.0.0/0 ppp 0 1
nxr130(config)#ip route 0.0.0.0/0 ppp 1 1
```

デフォルトルートを ppp0，1 インタフェースで設定し、共にディスタンス値を「1」とします。

＜DNS 設定＞

```
nxr130(config)#dns
nxr130(dns-config)#service enable
```

DNS サービスを有効にします。

2-4-3. パソコンの設定例

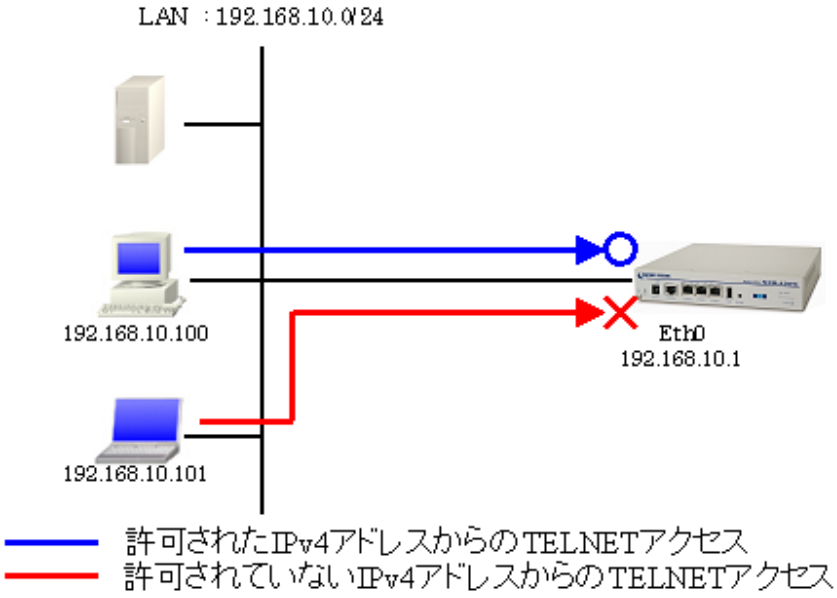
IPv4 アドレス	192.168.10.100
サブネットマスク	255.255.255.0
デフォルトゲートウェイ	192.168.10.1
DNS サーバの IPv4 アドレス	192.168.10.1

3. フィルタ設定

3-1. 入力フィルタ設定

入力フィルタではルータ宛に送信されたパケットのうち、ルータ自身で受信し処理するものを対象とします。ここでは LAN 内で特定の IPv4 アドレスからルータへの TELNET アクセスは許可するが、それ以外の IPv4 アドレスからの TELNET アクセスは破棄する設定です。

3-1-1. 構成図



3-1-2. 設定例

```
nxr130#configure terminal
nxr130(config)#ip access-list eth0_in permit 192.168.10.100 192.168.10.1 tcp any 23
nxr130(config)#ip access-list eth0_in deny any 192.168.10.1 tcp any 23
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
nxr130(config-if)#ip access-group in eth0_in
nxr130(config-if)#exit
nxr130(config)#exit
nxr130#save config
```

【 解説 】

- 送信元 IPv4 アドレス「192.168.10.100」, 宛先 IPv4 アドレス「192.168.10.1」への TELNET アクセスは許可し、その他の IPv4 アドレスから宛先 IPv4 アドレス「192.168.10.1」への TELNET アクセスは破棄する「eth0_in」という IPv4 アクセスリストを作成します。
- 作成した「eth0_in」の IPv4 アクセスリストを Ethernet0 インタフェースの「in」フィルタに適用します。

＜IPv4 アクセスリスト設定＞

```
nxr130(config)#ip access-list eth0_in permit 192.168.10.100 192.168.10.1 tcp any 23
```

IPv4 アクセスリスト名を「eth0_in」とし、送信元 IPv4 アドレス「192.168.10.100」, 宛先 IPv4 アドレス「192.168.10.1」への TELNET アクセスは許可します。

```
nxr130(config)#ip access-list eth0_in deny any 192.168.10.1 tcp any 23
```

IPv4 アクセスリスト名「eth0_in」に、宛先 IPv4 アドレス「192.168.10.1」への TELNET アクセスは破棄するルールを登録します。

＜Ethernet0 インタフェース設定＞

```
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.1/24」を設定します。

```
nxr130(config-if)#ip access-group in eth0_in
```

IPv4 アクセスリスト設定で設定した「eth0_in」を Ethernet0 インタフェースの「in」フィルタに適用します。

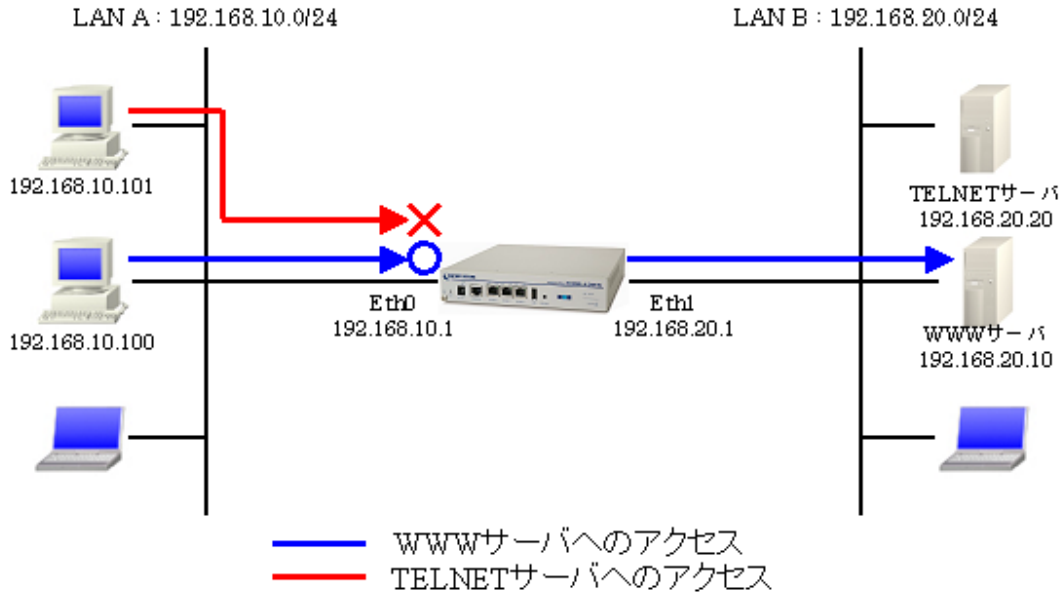
3-1-3. パソコンの設定例

IPv4 アドレス	192.168.10.100
サブネットマスク	255.255.255.0

3-2. 転送フィルタ設定

転送フィルタでは本装置で内部転送（本装置がルーティング）するパケットを制御するときに利用します。ここでは LAN B に設置されている WWW サーバ，TELNET サーバに対して IPv4 での WWW サーバへのアクセスは許可するが、IPv4 での TELNET サーバへのアクセスは破棄する設定です。

3-2-1. 構成図



3-2-2. 設定例

```
nxr130#configure terminal
nxr130(config)#ip access-list eth0_forward-in permit any 192.168.20.10 tcp any 80
nxr130(config)#ip access-list eth0_forward-in deny any 192.168.20.20 tcp any 23
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
nxr130(config-if)#ip access-group forward-in eth0_forward-in
nxr130(config-if)#exit
nxr130(config)#interface ethernet 1
nxr130(config-if)#ip address 192.168.20.1/24
nxr130(config-if)#exit
nxr130(config)#exit
nxr130#save config
```

【 解説 】

- 宛先 IPv4 アドレス「192.168.20.10」，TCP ポート番号「80」（HTTP）へのアクセスは許可し、宛先 IPv4 アドレス「192.168.20.20」，TCP ポート番号「23」（TELNET）へのアクセスは破棄する「eth0_forward-in」という IPv4 アクセスリストを作成します。
- 作成した「eth0_forward-in」の IPv4 アクセスリストを Ethernet0 インタフェースの「forward-in」フィルタに適用します。

＜IPv4 アクセスリスト設定＞

```
nxr130(config)#ip access-list eth0_forward-in permit any 192.168.20.10 tcp any 80
```

IPv4 アクセスリスト名を「eth0_forward-in」とし、宛先 IPv4 アドレス「192.168.20.10」, TCP ポート番号「80」(HTTP) へのアクセスは許可します。

```
nxr130(config)#ip access-list eth0_forward-in deny any 192.168.20.20 tcp any 23
```

IPv4 アクセスリスト名「eth0_forward-in」に、宛先 IPv4 アドレス「192.168.20.20」, TCP ポート番号「23」(TELNET) へのアクセスは破棄するルールを登録します。

＜Ethernet0 インタフェース設定＞

```
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.1/24」を設定します。

```
nxr130(config-if)#ip access-group forward-in eth0_forward-in
```

IPv4 アクセスリスト設定で設定した「eth0_forward-in」を Ethernet0 インタフェースの「forward-in」フィルタに適用します。

＜Ethernet1 インタフェース設定＞

```
nxr130(config)#interface ethernet 1
nxr130(config-if)#ip address 192.168.20.1/24
```

Ethernet1 インタフェースの IPv4 アドレスとして「192.168.20.1/24」を設定します。

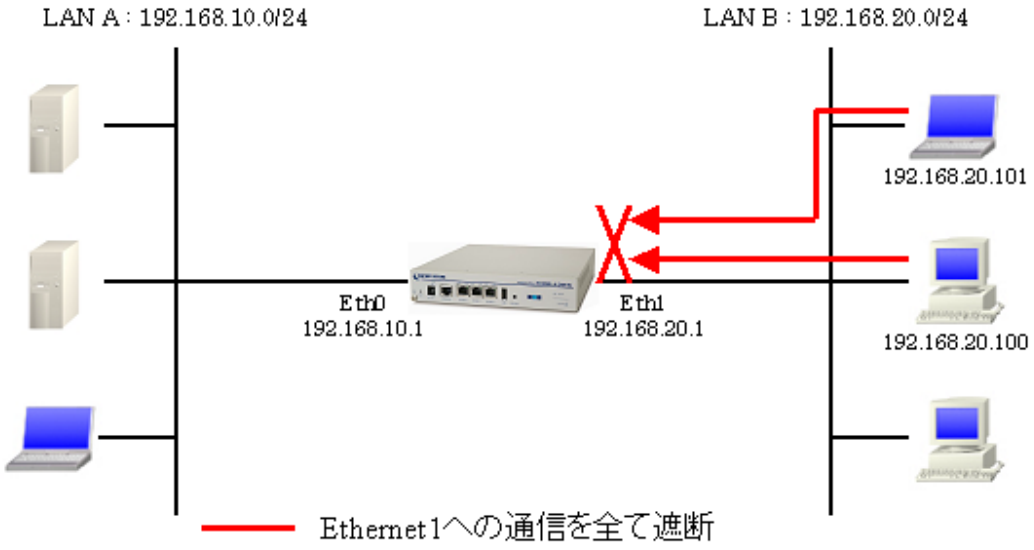
3-2-3. サーバ，パソコンの設定例

	LAN A の パソコン	LAN B の WWW サーバ	LAN B の TELNET サーバ	LAN B の パソコン
IPv4 アドレス	192.168.10.100	192.168.20.10	192.168.20.20	192.168.20.100
サブネットマスク	255.255.255.0	255.255.255.0	255.255.255.0	255.255.255.0
デフォルトゲートウェイ	192.168.10.1	192.168.20.1	192.168.20.1	192.168.20.1

3-3. 動的フィルタ（ステートフルパケットインスペクション）設定

ステートフルパケットインスペクションは、パケットを監視してパケットフィルタリング項目を随時変更する機能で、動的パケットフィルタリングともいわれる機能です。ここではEthernet1インタフェース側からの接続要求を全て遮断する設定です。

3-3-1. 構成図



3-3-2. 設定例

```
nxr130#configure terminal
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
nxr130(config-if)#exit
nxr130(config)#interface ethernet 1
nxr130(config-if)#ip address 192.168.20.1/24
nxr130(config-if)#ip spi-filter
nxr130(config-if)#exit
nxr130(config)#exit
nxr130#save config
```

【 解説 】

- Ethernet1インタフェースで動的フィルタ（ステートフルパケットインスペクション）を有効にし、Ethernet1インタフェース側からの接続要求を遮断します。

＜Ethernet0インタフェース設定＞

```
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
```

Ethernet0インタフェースのIPv4アドレスとして「192.168.10.1/24」を設定します。

＜Ethernet1 インタフェース設定＞

```
nxr130(config)#interface ethernet 1
nxr130(config-if)#ip address 192.168.20.1/24
```

Ethernet1 インタフェースの IPv4 アドレスとして「192.168.20.1/24」を設定します。

```
nxr130(config-if)#ip spi-filter
```

Ethernet1 インタフェースの動的フィルタ(ステートフルパケットインスペクション)を有効にします。

3-3-3. パソコンの設定例

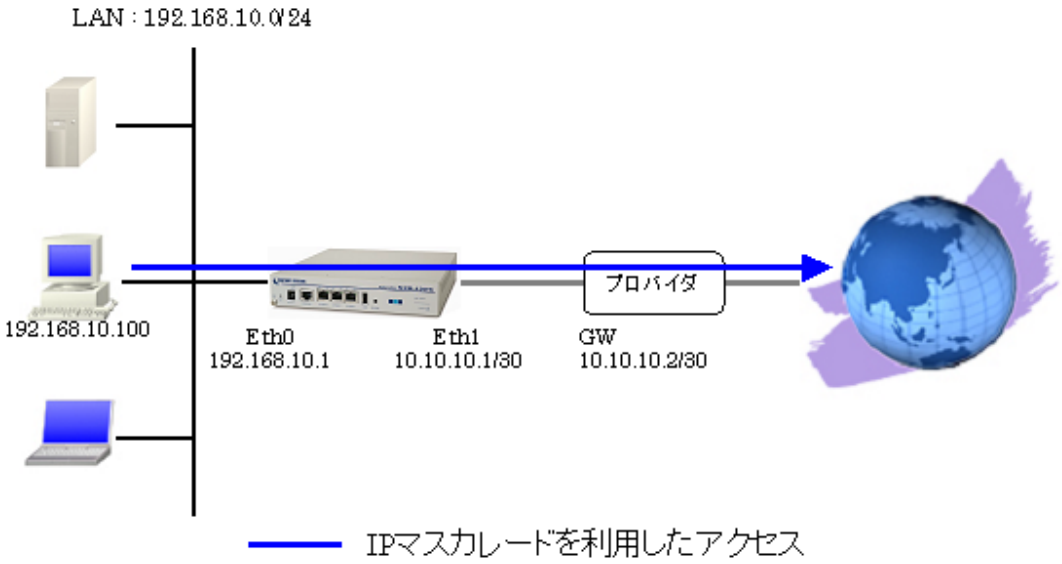
	LAN A のパソコン	LAN B のパソコン
IPv4 アドレス	192.168.10.100	192.168.20.100
サブネットマスク	255.255.255.0	255.255.255.0
デフォルトゲートウェイ	192.168.10.1	192.168.20.1

4. NAT 設定

4-1. IP マスカレード設定

プライベート IPv4 アドレスのネットワーク内にある端末がインターネットへアクセスする際など、送信元 IPv4 アドレスを IP マスカレードの設定を有効にしたインタフェースの IPv4 アドレスに変換することができます。

4-1-1. 構成図



4-1-2. 設定例

```
nxr130#configure terminal
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
nxr130(config-if)#exit
nxr130(config)#interface ethernet 1
nxr130(config-if)#ip address 10.10.10.1/30
nxr130(config-if)#ip masquerade
nxr130(config-if)#ip spi-filter
nxr130(config-if)#no ip redirects
nxr130(config-if)#exit
nxr130(config)#ip route 0.0.0.0/0 10.10.10.2
nxr130(config)#dns
nxr130(dns-config)#service enable
nxr130(dns-config)#root enable
nxr130(dns-config)#exit
nxr130(config)#exit
nxr130#save config
```

【 解説 】

- Ethernet0 インタフェースを LAN 側，Ethernet1 インタフェースを WAN 側とします。
- Ethernet1 インタフェースで IP マスカレード，ステートフルパケットインスペクションを有効にしています。
- DNS サービスを有効にし、root DNS サーバを有効にしています。

＜Ethernet0 インタフェース設定＞

```
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.1/24」を設定します。

＜Ethernet1 インタフェース設定＞

```
nxr130(config)#interface ethernet 1
```

Ethernet1 インタフェースに関する設定をします。

```
nxr130(config-if)#ip address 10.10.10.1/30
```

Ethernet1 インタフェースの IPv4 アドレスとして「10.10.10.1/30」を設定します。

```
nxr130(config-if)#ip masquerade
```

IP マスカレードを設定します。これにより Ethernet1 インタフェースより出力されるパケットの送信元 IPv4 アドレスを Ethernet1 インタフェースの IPv4 アドレスに変換して通信することができます。

```
nxr130(config-if)#ip spi-filter
```

ステートフルパケットインスペクションを設定します。

```
nxr130(config-if)#no ip redirects
```

ICMP リダイレクト機能を無効に設定します。

＜スタティックルート設定＞

```
nxr130(config)#ip route 0.0.0.0/0 10.10.10.2
```

デフォルトルートを設定します。

＜DNS 設定＞

```
nxr130(config)#dns
nxr130(dns-config)#root enable
nxr130(dns-config)#service enable
```

DNS サービスを有効にし、ここでは root DNS サーバを有効にします。

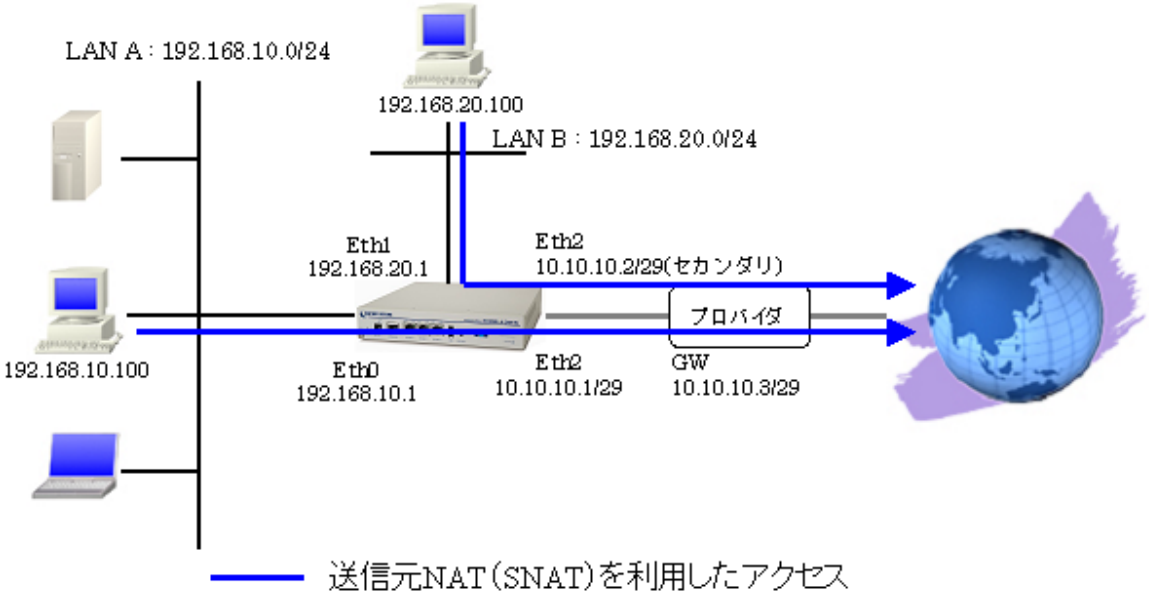
4-1-3. パソコンの設定例

IPv4 アドレス	192.168.10.100
サブネットマスク	255.255.255.0
デフォルトゲートウェイ	192.168.10.1
DNS サーバの IPv4 アドレス	192.168.10.1

4-2. 送信元 NAT（SNAT）設定

ある特定のネットワークやホストを指定し、送信元 IPv4 アドレスの変換を行うことができます。
例えばセグメント毎に異なるグローバル IPv4 アドレスを利用する際に使用します。

4-2-1. 構成図



4-2-2. 設定例

```
nxr130#configure terminal
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
nxr130(config-if)#exit
nxr130(config)#interface ethernet 1
nxr130(config-if)#ip address 192.168.20.1/24
nxr130(config-if)#exit
nxr130(config)#ip snat eth2_snat ip 192.168.10.0/24 any 10.10.10.1
nxr130(config)#ip snat eth2_snat ip 192.168.20.0/24 any 10.10.10.2
nxr130(config)#interface ethernet 2
nxr130(config-if)#ip address 10.10.10.1/29
nxr130(config-if)#ip address 10.10.10.2/29 secondary
nxr130(config-if)#ip snat eth2_snat
nxr130(config-if)#ip spi-filter
nxr130(config-if)#no ip redirects
nxr130(config-if)#exit
nxr130(config)#ip route 0.0.0.0/0 10.10.10.3
nxr130(config)#dns
nxr130(dns-config)#service enable
nxr130(dns-config)#root enable
nxr130(dns-config)#exit
nxr130(config)#exit
nxr130#save config
```

【 解説 】

- Ethernet0，1 インタフェースを LAN 側，Ethernet2 インタフェースを WAN 側とします。
- Ethernet0，1 インタフェースが属するネットワークからのパケットで Ethernet2 インタフェース

から出力されるパケットの送信元 IP アドレスを変換します。

- Ethernet2 インタフェースでステートフルパケットインスペクションを有効にしています。
- DNS サービスを有効にし、root DNS サーバを有効にしています。

＜Ethernet0 インタフェース設定＞

```
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.1/24」を設定します。

＜Ethernet1 インタフェース設定＞

```
nxr130(config)#interface ethernet 1
nxr130(config-if)#ip address 192.168.20.1/24
```

Ethernet1 インタフェースの IPv4 アドレスとして「192.168.20.1/24」を設定します。

＜SNAT 設定＞

```
nxr130(config)#ip snat eth2_snat ip 192.168.10.0/24 any 10.10.10.1
```

SNAT 名を「eth2_snat」とし、送信元 IPv4 アドレスが「192.168.10.0/24」のパケットは、IPv4 アドレス「10.10.10.1」に変換します。

```
nxr130(config)#ip snat eth2_snat ip 192.168.20.0/24 any 10.10.10.2
```

SNAT 名「eth2_snat」に、送信元 IPv4 アドレスが「192.168.20.0/24」のパケットは IPv4 アドレス「10.10.10.2」に変換するルールを登録します。

＜Ethernet2 インタフェース設定＞

```
nxr130(config)#interface ethernet 2
```

Ethernet2 インタフェースに関する設定をします。

```
nxr130(config-if)#ip address 10.10.10.1/29
```

Ethernet2 インタフェースの IPv4 アドレスとして「10.10.10.1/29」を設定します。

```
nxr130(config-if)#ip address 10.10.10.2/29 secondary
```

Ethernet2 インタフェースのセカンダリアドレスとして「10.10.10.2/29」を設定します。

```
nxr130(config-if)#ip snat eth2_snat
```

SNAT 設定で設定した「eth2_snat」を Ethernet2 インタフェースに適用します。これにより Ethernet2 インタフェースからのパケット送信時に送信元 IPv4 アドレスが「192.168.10.0/24」のパケットは IPv4 アドレス「10.10.10.1」に変換、送信元 IPv4 アドレスが「192.168.20.0/24」のパケットは IPv4 アドレス「10.10.10.2」に変換されます。

```
nxr130(config-if)#ip spi-filter
```

ステートフルパケットインスペクションを設定します。

```
nxr130(config-if)#no ip redirects
```

ICMP リダイレクト機能を無効に設定します。

＜スタティックルート設定＞

```
nxr130(config)#ip route 0.0.0.0/0 10.10.10.3
```

デフォルトルートを設定します。

＜DNS 設定＞

```
nxr130(config)#dns
nxr130(dns-config)#root enable
nxr130(dns-config)#service enable
```

DNS サービスを有効にし、ここでは root DNS サーバを有効にします。

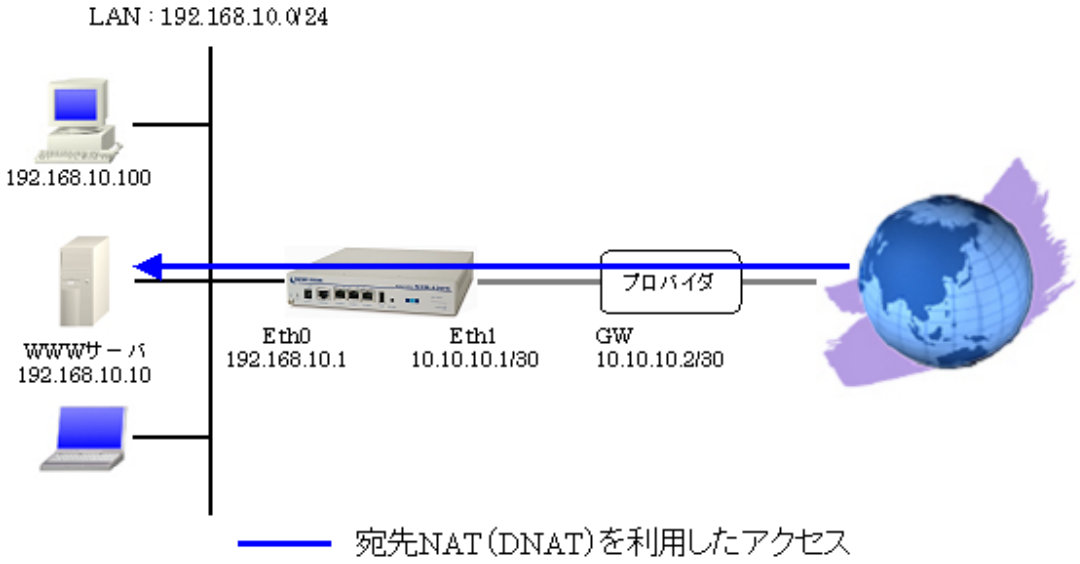
4-2-3. パソコンの設定例

	LAN A のパソコン	LAN B のパソコン
IPv4 アドレス	192.168.10.100	192.168.20.100
サブネットマスク	255.255.255.0	255.255.255.0
デフォルトゲートウェイ	192.168.10.1	192.168.20.1
DNS サーバの IPv4 アドレス	192.168.10.1	192.168.20.1

4-3. 宛先 NAT（DNAT）設定

プライベート IPv4 アドレスのネットワーク内にあるサーバをインターネット経由でアクセスさせる場合、宛先 NAT（DNAT）によりルータ経由でのアクセスが可能になります。
ここでは WWW サーバを DNAT で外部に公開する設定をします。

4-3-1. 構成図



4-3-2. 設定例

```
nxr130#configure terminal
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
nxr130(config-if)#exit
nxr130(config)#ip dnat eth1_dnat tcp any any 10.10.10.1 80 192.168.10.10
nxr130(config)#ip access-list eth1_forward-in permit any 192.168.10.10 tcp any 80
nxr130(config)#interface ethernet 1
nxr130(config-if)#ip address 10.10.10.1/30
nxr130(config-if)#ip dnat eth1_dnat
nxr130(config-if)#ip masquerade
nxr130(config-if)#ip access-group forward-in eth1_forward-in
nxr130(config-if)#ip spi-filter
nxr130(config-if)#no ip redirects
nxr130(config-if)#exit
nxr130(config)#ip route 0.0.0.0/0 10.10.10.2
nxr130(config)#dns
nxr130(dns-config)#service enable
nxr130(dns-config)#root enable
nxr130(dns-config)#exit
nxr130(config)#exit
nxr130#save config
```

【 解説 】

- Ethernet0 インタフェースを LAN 側，Ethernet1 インタフェースを WAN 側とします。
- Ethernet1 インタフェースで宛先 IPv4 アドレス「10.10.10.1」 TCP ポート番号「80」のパケットを受信した場合は、パケットの宛先 IPv4 アドレスを「192.168.10.10」に変換します。

- Ethernet1 インタフェースで宛先 IPv4 アドレス「192.168.10.10」, TCP ポート番号「80」へのアクセスは許可します。
- Ethernet1 インタフェースでステートフルパケットインスペクションを有効にしています。
- DNS サービスを有効にし、ここでは root DNS サーバを有効にします。

＜Ethernet0 インタフェース設定＞

```
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.1/24」を設定します。

＜DNAT 設定＞

```
nxr130(config)#ip dnat eth1_dnat tcp any any 10.10.10.1 80 192.168.10.10
```

DNAT 名を「eth1_dnat」とし、宛先 IPv4 アドレス「10.10.10.1」, TCP ポート番号「80」のパケットの宛先 IPv4 アドレスを「192.168.10.10」に変換します。

＜IPv4 アクセスリスト設定＞

```
nxr130(config)#ip access-list eth1_forward-in permit any 192.168.10.10 tcp any 80
```

IPv4 アクセスリスト名を「eth1_forward-in」とし、宛先 IPv4 アドレス「192.168.10.10」, TCP ポート番号「80」のパケットは許可します。

＜Ethernet1 インタフェース設定＞

```
nxr130(config)#interface ethernet 1
nxr130(config-if)#ip address 10.10.10.1/30
```

Ethernet1 インタフェースの IPv4 アドレスとして「10.10.10.1/30」を設定します。

```
nxr130(config-if)#ip dnat eth1_dnat
```

DNAT 設定で設定した「eth1_dnat」を Ethernet1 インタフェースに適用します。
これにより Ethernet1 インタフェースで宛先 IPv4 アドレス「10.10.10.1」, TCP ポート番号「80」のパケットの宛先 IPv4 アドレスは「192.168.10.10」に変換されます。

```
nxr130(config-if)#ip masquerade
```

IP マスカレードを設定します。

```
nxr130(config-if)#ip access-group forward-in eth1_forward-in
```

IPv4 アクセスリスト設定で設定した「eth1_forward-in」を Ethernet1 インタフェースの「forward-in」フィルタに適用します。
これにより Ethernet1 インタフェースで宛先 IPv4 アドレス「192.168.10.10」, TCP ポート番号「80」のパケットは許可されます。

```
nxr130(config-if)#ip spi-filter
```
ステートフルパケットインスペクションを設定します。

```
nxr130(config-if)#no ip redirects
```
ICMP リダイレクト機能を無効に設定します。

＜スタティックルート設定＞

```
nxr130(config)#ip route 0.0.0.0/0 10.10.10.2
```
デフォルトルートを設定します。

＜DNS 設定＞

```
nxr130(config)#dns
nxr130(dns-config)#root enable
nxr130(dns-config)#service enable
```
DNS サービスを有効にし、ここでは root DNS サーバを有効にします。

4-3-3. サーバ，パソコンの設定例

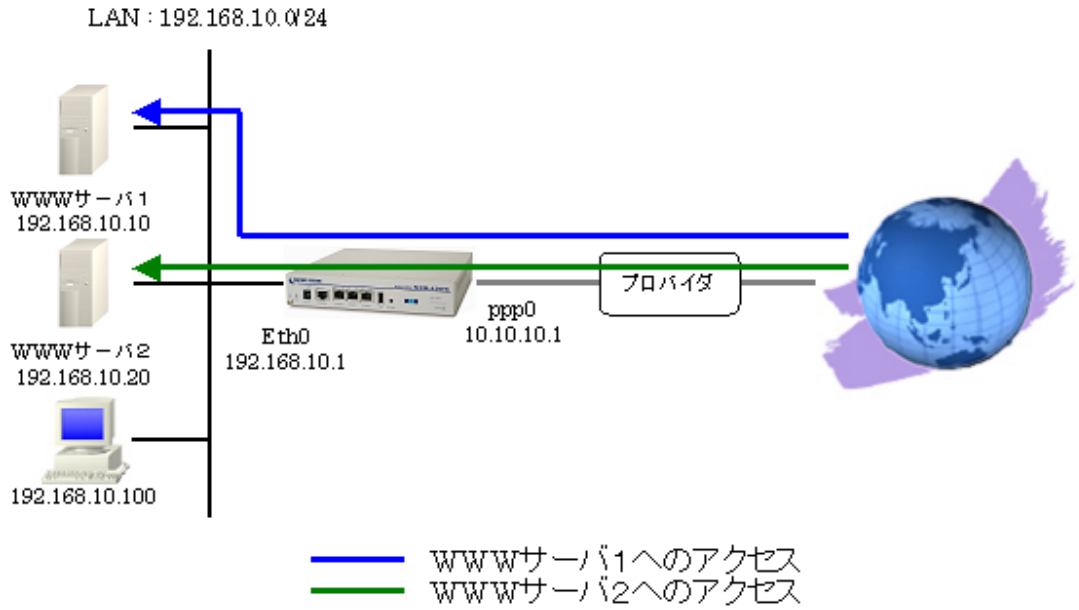
	WWW サーバ	パソコン
IPv4 アドレス	192.168.10.10	192.168.10.100
サブネットマスク	255.255.255.0	255.255.255.0
デフォルトゲートウェイ	192.168.10.1	192.168.10.1
DNS サーバの IPv4 アドレス	－	192.168.10.1

5. NAT/フィルタ応用設定

5-1. NAT でのサーバ公開１（ポートマッピング）設定

DNAT 機能では NAT 変換時にその前後でポート番号を変換することが可能です。ここでは WAN 側で受信時のポート番号を分けておくこととで、グローバル IPv4 アドレスが１つでも複数の WEB サーバに対してのアクセスが可能になる設定です。

5-1-1. 構成図



5-1-2. 設定例

```
nxr130#configure terminal
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
nxr130(config-if)#exit
nxr130(config)#ip dnat ppp0_dnat tcp any any 10.10.10.1 80 192.168.10.10 80
nxr130(config)#ip dnat ppp0_dnat tcp any any 10.10.10.1 8080 192.168.10.20 80
nxr130(config)#ip access-list ppp0_forward-in permit any 192.168.10.10 tcp any 80
nxr130(config)#ip access-list ppp0_forward-in permit any 192.168.10.20 tcp any 80
nxr130(config)#interface ppp 0
nxr130(config-ppp)#ip address 10.10.10.1/32
nxr130(config-ppp)#ip dnat ppp0_dnat
nxr130(config-ppp)#ip masquerade
nxr130(config-ppp)#ip access-group forward-in ppp0_forward-in
nxr130(config-ppp)#ip spi-filter
nxr130(config-ppp)#ip tcp adjust-mss auto
nxr130(config-ppp)#no ip redirects
nxr130(config-ppp)#ppp username test1@centurysys password test1pass
nxr130(config-ppp)#exit
nxr130(config)#interface ethernet 1
nxr130(config-if)#no ip address
nxr130(config-if)#pppoe-client ppp 0
nxr130(config-if)#exit
nxr130(config)#ip route 0.0.0.0/0 ppp 0
nxr130(config)#dns
nxr130(dns-config)#service enable

----- 次のページに続きがあります -----
```

```
----- 前のページからの続きです -----

nxr130(dns-config)#exit
nxr130(config)#exit
nxr130#save config
```

【 解説 】

- ppp0 インタフェースで宛先 IPv4 アドレス「10.10.10.1」, TCP ポート番号「80」(WWW サーバ１)のパケットを受信した場合は、パケットの宛先 IPv4 アドレスを「192.168.10.10」に変換します。
- ppp0 インタフェースで宛先 IPv4 アドレス「10.10.10.1」, TCP ポート番号「8080」(WWW サーバ２)のパケットを受信した場合は、パケットの宛先 IPv4 アドレスを「192.168.10.20」, TCP ポート番号「80」に変換します。
- ppp0 インタフェースで宛先 IPv4 アドレス「192.168.10.10」および「192.168.10.20」, TCP ポート番号「80」へのアクセスは許可します。
- ppp0 インタフェースでステートフルパケットインスペクションを有効にしています。

＜Ethernet0 インタフェース設定＞

```
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.1/24」を設定します。

＜DNAT 設定＞

```
nxr130(config)#ip dnat ppp0_dnat tcp any any 10.10.10.1 80 192.168.10.10 80
```

DNAT 名を「ppp0_dnat」とし、「10.10.10.1」, TCP ポート番号「80」のパケットは、宛先 IPv4 アドレス「192.168.10.10」, TCP ポート番号「80」に変換します。

```
nxr130(config)#ip dnat ppp0_dnat tcp any any 10.10.10.1 8080 192.168.10.20 80
```

DNAT 名「ppp0_dnat」に、宛先 IPv4 アドレス「10.10.10.1」, TCP ポート番号「8080」のパケットは、宛先 IPv4 アドレス「192.168.10.20」, TCP ポート番号「80」に変換するルールを登録します。

＜IPv4 アクセスリスト設定＞

```
nxr130(config)#ip access-list ppp0_forward-in permit any 192.168.10.10 tcp any 80
```

IPv4 アクセスリスト名を「ppp0_forward-in」とし、宛先 IPv4 アドレス「192.168.10.10」, TCP ポート番号「80」のパケットは許可します。

```
nxr130(config)#ip access-list ppp0_forward-in permit any 192.168.10.20 tcp any 80
```

IPv4 アクセスリスト名「ppp0_forward-in」に、宛先 IPv4 アドレス「192.168.10.20」, TCP ポート番号「80」のパケットは許可するルールを登録します。

＜ppp0 インタフェース設定＞

```
nxr130(config)#interface ppp 0
```

ppp0 インタフェースに関する設定をします。

```
nxr130(config-ppp)#ip address 10.10.10.1/32
```
IP アドレスを「10.10.10.1/32」で設定します。

```
nxr130(config-ppp)#ip dnat ppp0_dnat
```
DNAT 設定で設定した「ppp0_dnat」を ppp0 インタフェースに適用します。

```
nxr130(config-ppp)#ip masquerade
```
IP マスカレードを設定します。

```
nxr130(config-ppp)#ip access-group forward-in ppp0_forward-in
```
IPv4 アクセスリスト設定で設定した「ppp0-forward-in」を ppp0 インタフェースの「forward-in」フィルタに適用します。

```
nxr130(config-ppp)#ip spi-filter
```
ステートフルパケットインスペクションを設定します。

```
nxr130(config-ppp)#ip tcp adjust-mss auto
```
TCP MSS の調整機能をオートに設定します。

```
nxr130(config-ppp)#no ip redirects
```
ICMP リダイレクト機能を無効に設定します。

```
nxr130(config-ppp)#ppp username test1@centurysys password test1pass
```
PPPoE 接続で使用するユーザ ID とパスワードを設定します。

＜Ethernet1 インタフェース設定＞

```
nxr130(config)#interface Ethernet 1
```
Ethernet1 インタフェースに関する設定をします。

```
nxr130(config-if)#no ip address
```
Ethernet1 インタフェースに IPv4 アドレスを割り当てない設定をします。

```
nxr130(config-if)#pppoe-client ppp 0
```
Ethernet1 インタフェース上で ppp0 インタフェースを使用するための設定をします。

＜スタティックルート設定＞

```
nxr130(config)#ip route 0.0.0.0/0 ppp 0
```
デフォルトルートを設定します。

＜DNS 設定＞

```
nxr130(config)#dns
nxr130(dns-config)#service enable
```

DNS サービスを有効にします。

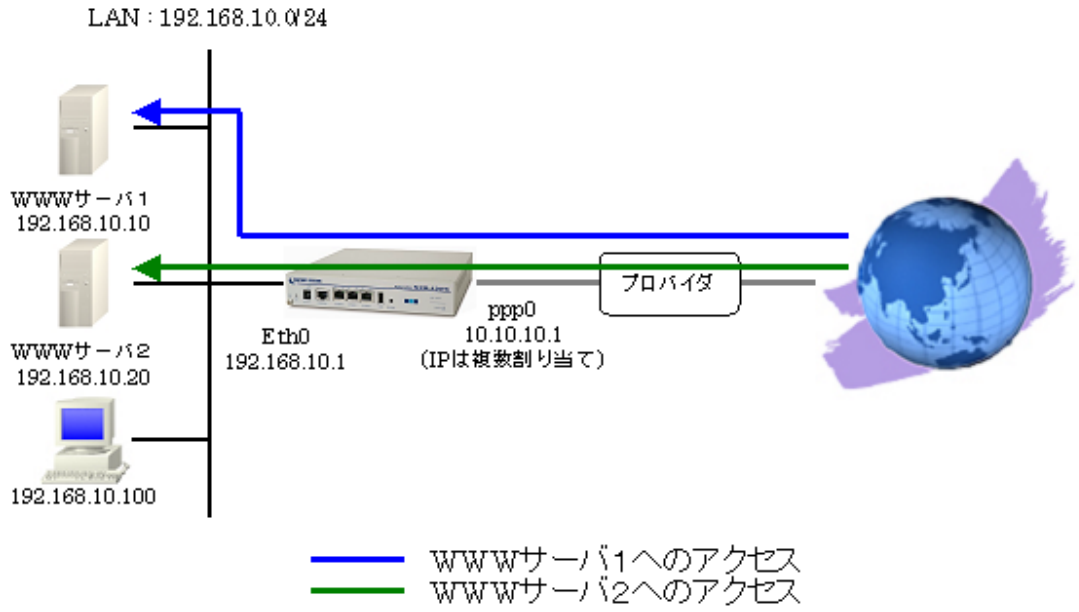
5-1-3. サーバ，パソコンの設定例

	WWW サーバ１	WWW サーバ２	パソコン
IPv4 アドレス	192.168.10.10	192.168.10.20	192.168.10.100
サブネットマスク	255.255.255.0	255.255.255.0	255.255.255.0
デフォルトゲートウェイ	192.168.10.1	192.168.10.1	192.168.10.1
DNS サーバの IPv4 アドレス	－	－	192.168.10.1

5-2. NAT でのサーバ公開２（複数 IP+PPPoE）設定

複数のグローバル IPv4 アドレスが割り当てられる場合、それぞれのグローバル IPv4 アドレス毎に LAN 内のプライベート IP アドレスを持ったサーバに対して DNAT 設定をすることにより、異なるグローバル IP アドレスでそれぞれのサーバに対してアクセスさせることができます。ここでは WAN 回線に PPPoE を利用した例になります。

5-2-1. 構成図



5-2-2. 設定例

```
nxr130#configure terminal
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
nxr130(config-if)#exit
nxr130(config)#ip dnat ppp0_dnat tcp any any 10.10.10.1 80 192.168.10.10
nxr130(config)#ip dnat ppp0_dnat tcp any any 10.10.10.2 80 192.168.10.20
nxr130(config)#ip access-list ppp0_forward-in permit any 192.168.10.10 tcp any 80
nxr130(config)#ip access-list ppp0_forward-in permit any 192.168.10.20 tcp any 80
nxr130(config)#interface ppp 0
nxr130(config-ppp)#ip address 10.10.10.1/32
nxr130(config-ppp)#ip dnat ppp0_dnat
nxr130(config-ppp)#ip masquerade
nxr130(config-ppp)#ip access-group forward-in ppp0_forward-in
nxr130(config-ppp)#ip spi-filter
nxr130(config-ppp)#ip tcp adjust-mss auto
nxr130(config-ppp)#no ip redirects
nxr130(config-ppp)#ppp username test1@centurysys password test1pass
nxr130(config-ppp)#exit
nxr130(config)#interface ethernet 1
nxr130(config-if)#no ip address
nxr130(config-if)#pppoe-client ppp 0
nxr130(config-if)#exit
nxr130(config)#ip route 0.0.0.0/0 ppp 0
nxr130(config)#dns
nxr130(dns-config)#service enable
nxr130(dns-config)#exit
nxr130(config)#exit
nxr130#save config
```

【 解説 】

- ppp0 インタフェースで宛先 IPv4 アドレス「10.10.10.1」, TCP ポート番号「80」(WWW サーバ１)のパケットを受信した場合は、パケットの宛先 IPv4 アドレスを「192.168.10.10」に変換します。
- ppp0 インタフェースで宛先 IPv4 アドレス「10.10.10.2」, TCP ポート番号「80」(WWW サーバ２)のパケットを受信した場合は、パケットの宛先 IPv4 アドレスを「192.168.10.20」に変換します。
- ppp0 インタフェースで宛先 IPv4 アドレス「192.168.10.10」および「192.168.10.20」, TCP ポート番号「80」へのアクセスは許可します。
- ppp0 インタフェースでステートフルパケットインスペクションを有効にしています。

＜Ethernet0 インタフェース設定＞

```
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.1/24」を設定します。

＜DNAT 設定＞

```
nxr130(config)#ip dnat ppp0_dnat tcp any any 10.10.10.1 80 192.168.10.10
nxr130(config)#ip dnat ppp0_dnat tcp any any 10.10.10.2 80 192.168.10.20
```

DNAT 名を「ppp0_dnat」とし、宛先 IPv4 アドレス「10.10.10.1」, TCP ポート番号「80」のパケットは宛先 IPv4 アドレス「192.168.10.10」に変換し、宛先 IPv4 アドレス「10.10.10.2」, TCP ポート番号「80」のパケットは宛先 IPv4 アドレスを「192.168.10.20」に変換します。

＜IPv4 アクセスリスト設定＞

```
nxr130(config)#ip access-list ppp0_forward-in permit any 192.168.10.10 tcp any 80
nxr130(config)#ip access-list ppp0_forward-in permit any 192.168.10.20 tcp any 80
```

IPv4 アクセスリスト名を「ppp0_forward-in」とし、宛先 IPv4 アドレス「192.168.10.10」, TCP ポート番号「80」のパケットおよび宛先 IPv4 アドレス「192.168.10.20」, TCP ポート番号「80」のパケットは許可します。

＜ppp0 インタフェース設定＞

```
nxr130(config)#interface ppp 0
nxr130(config-ppp)#ip address 10.10.10.1/32
nxr130(config-ppp)#ip dnat ppp0_dnat
nxr130(config-ppp)#ip masquerade
nxr130(config-ppp)#ip access-group forward-in ppp0_forward-in
nxr130(config-ppp)#ip spi-filter
nxr130(config-ppp)#ip tcp adjust-mss auto
nxr130(config-ppp)#no ip redirects
nxr130(config-ppp)#ppp username test1@centurysys password test1pass
```

ppp0 インタフェースに関する設定をします。

DNAT 設定で設定した「ppp0_dnat」を ppp0 インタフェースに適用します。

IPv4 アクセスリスト設定で設定した「ppp0-forward-in」を ppp0 インタフェースの「forward-in」フィルタに適用します。

＜Ethernet1 インタフェース設定＞

```
nxr130(config)#interface ethernet 1
nxr130(config-if)#no ip address
nxr130(config-if)#pppoe-client ppp 0
```

Ethernet1 インタフェースに関する設定をします。

＜スタティックルート設定＞

```
nxr130(config)#ip route 0.0.0.0/0 ppp 0
```

デフォルトルートを設定します。

＜DNS 設定＞

```
nxr130(config)#dns
nxr130(dns-config)#service enable
```

DNS サービスを有効にします。

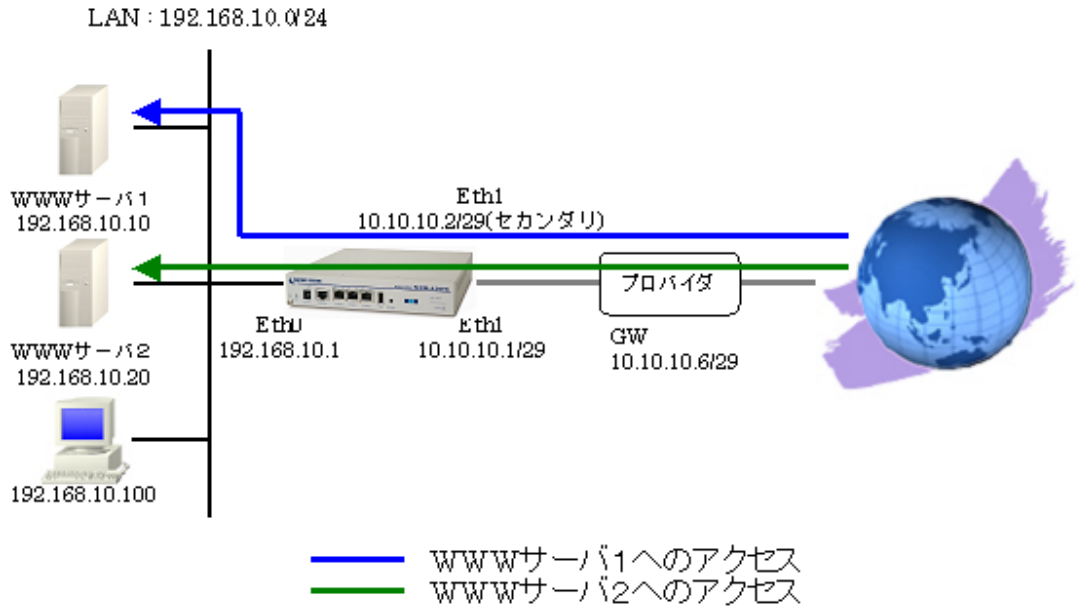
5-2-3. サーバ，パソコンの設定例

	WWW サーバ１	WWW サーバ２	パソコン
IPv4 アドレス	192.168.10.10	192.168.10.20	192.168.10.100
サブネットマスク	255.255.255.0	255.255.255.0	255.255.255.0
デフォルトゲートウェイ	192.168.10.1	192.168.10.1	192.168.10.1
DNS サーバの IPv4 アドレス	－	－	192.168.10.1

5-3. NAT でのサーバ公開３（複数 IP+Ethernet）設定

複数のグローバル IPv4 アドレスが割り当てられる場合、それぞれのグローバル IPv4 アドレス毎に LAN 内のプライベート IP アドレスを持ったサーバに対して DNAT 設定をすることにより、異なるグローバル IP アドレスでそれぞれのサーバに対してアクセスさせることができます。ここでは WAN 回線に Ethernet を利用した例になります。

5-3-1. 構成図



5-3-2. 設定例

```
nxr130#configure terminal
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
nxr130(config-if)#exit
nxr130(config)#ip dnat eth1_dnat tcp any any 10.10.10.1 80 192.168.10.10
nxr130(config)#ip dnat eth1_dnat tcp any any 10.10.10.2 80 192.168.10.20
nxr130(config)#ip access-list eth1_forward-in permit any 192.168.10.10 tcp any 80
nxr130(config)#ip access-list eth1_forward-in permit any 192.168.10.20 tcp any 80
nxr130(config)#interface ethernet 1
nxr130(config-if)#ip address 10.10.10.1/29
nxr130(config-if)#ip address 10.10.10.2/29 secondary
nxr130(config-if)#ip dnat eth1_dnat
nxr130(config-if)#ip masquerade
nxr130(config-if)#ip access-group forward-in eth1_forward-in
nxr130(config-if)#ip spi-filter
nxr130(config-if)#no ip redirects
nxr130(config-if)#exit
nxr130(config)#ip route 0.0.0.0/0 10.10.10.6
nxr130(config)#dns
nxr130(dns-config)#service enable
nxr130(dns-config)#exit
nxr130(config)#exit
nxr130#save config
```

【 解説 】

- Ethernet1 インタフェースで宛先 IPv4 アドレス「10.10.10.1」, TCP ポート番号「80」(WWW サーバ１）のパケットを受信した場合は、パケットの宛先 IPv4 アドレスを「192.168.10.10」に変換します。
- Ethernet1 インタフェースで宛先 IPv4 アドレス「10.10.10.2」, TCP ポート番号「80」(WWW サーバ２）のパケットを受信した場合は、パケットの宛先 IPv4 アドレスを「192.168.10.20」, TCP ポート番号「80」に変換します。
- Ethernet1 インタフェースで宛先 IPv4 アドレス「192.168.10.10」および「192.168.10.20」, TCP ポート番号「80」へのアクセスは許可します。
- Ethernet1 インタフェースでステートフルパケットインスペクションを有効にしています。

＜Ethernet0 インタフェース設定＞

```
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.1/24」を設定します。

＜DNAT 設定＞

```
nxr130(config)#ip dnat eth1_dnat tcp any any 10.10.10.1 80 192.168.10.10
nxr130(config)#ip dnat eth1_dnat tcp any any 10.10.10.2 80 192.168.10.20
```

DNAT 名を「eth1_dnat」とし、宛先 IPv4 アドレス「10.10.10.1」, TCP ポート番号「80」のパケットは宛先 IPv4 アドレス「192.168.10.10」に変換し、宛先 IPv4 アドレス「10.10.10.2」, TCP ポート番号「80」のパケットは宛先 IPv4 アドレス「192.168.10.20」に変換します。

＜IPv4 アクセスリスト設定＞

```
nxr130(config)#ip access-list eth1_forward-in permit any 192.168.10.10 tcp any 80
nxr130(config)#ip access-list eth1_forward-in permit any 192.168.10.20 tcp any 80
```

IPv4 アクセスリスト名を「eth1_forward-in」とし、宛先 IPv4 アドレス「192.168.10.10」, TCP ポート番号「80」のパケットおよび宛先 IPv4 アドレス「192.168.10.20」, TCP ポート番号「80」のパケットは許可します。

＜Ethernet1 インタフェース設定＞

```
nxr130(config)#interface ethernet 1
```

Ethernet1 インタフェースに関する設定をします。

```
nxr130(config-if)#ip address 10.10.10.1/29
nxr130(config-if)#ip address 10.10.10.2/29 secondary
```

Ethernet1 インタフェースの IPv4 アドレスとして「10.10.10.1/29」を、セカンダリ IPv4 アドレスとして「10.10.10.2/29」を設定します。

```
nxr130(config-if)#ip dnat eth1_dnat
nxr130(config-if)#ip masquerade
nxr130(config-if)#ip access-group forward-in eth1_forward-in
nxr130(config-if)#ip spi-filter
nxr130(config-if)#no ip redirects
```

DNAT 設定で設定した「eth1_dnat」を Ethernet1 インタフェースに適用します。

IPv4 アクセスリスト設定で設定した「eth1_forward-in」を Ethernet1 インタフェースの「forward-in」フィルタに適用します。

＜スタティックルート設定＞

```
nxr130(config)#ip route 0.0.0.0/0 10.10.10.6
```

デフォルトルートを設定します。

＜DNS 設定＞

```
nxr130(config)#dns
nxr130(dns-config)#service enable
```

DNS サービスを有効にします。

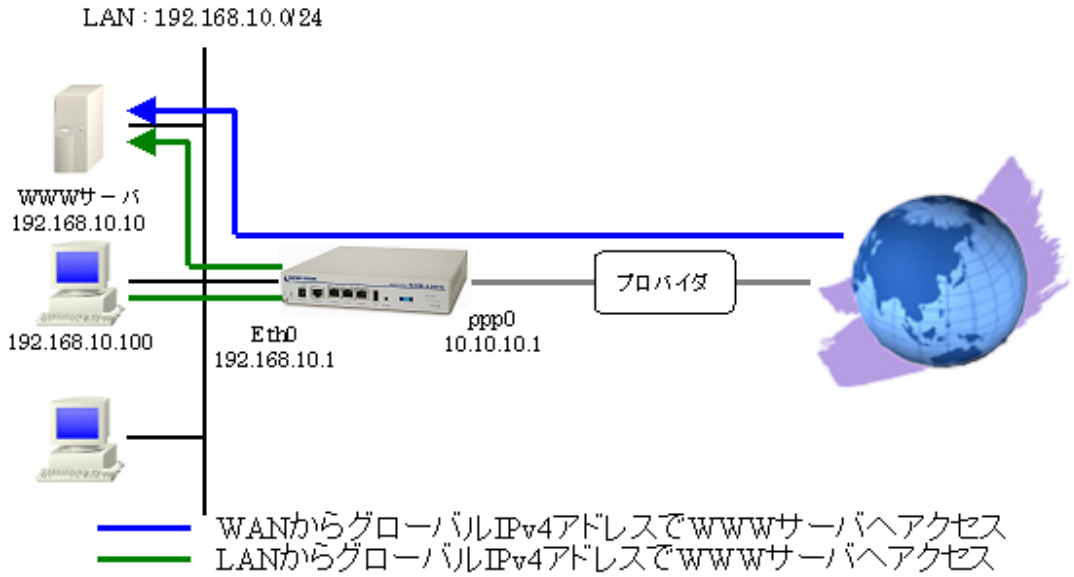
5-3-3. サーバ，パソコンの設定例

	WWW サーバ１	WWW サーバ２	パソコン
IPv4 アドレス	192.168.10.10	192.168.10.20	192.168.10.100
サブネットマスク	255.255.255.0	255.255.255.0	255.255.255.0
デフォルトゲートウェイ	192.168.10.1	192.168.10.1	192.168.10.1
DNS サーバの IPv4 アドレス	－	－	192.168.10.1

5-4. NAT でのサーバ公開４（LAN 内のサーバにグローバル IPv4 アドレスでアクセス）設定

NAT で外部に公開しているサーバに対しては、プライベート IPv4 アドレスでのアクセスだけでなく、グローバル IPv4 アドレスでのアクセスも可能です。ここでは NAT で公開している LAN 内の WWW サーバに対してグローバル IPv4 アドレスでアクセスする設定です。

5-4-1. 構成図



5-4-2. 設定例

```
nxr130#configure terminal
nxr130(config)#ip dnat ppp0_dnat tcp any any 10.10.10.1 80 192.168.10.10 80
nxr130(config)#ip dnat eth0_dnat tcp 192.168.10.0/24 any 10.10.10.1 80 192.168.10.10
nxr130(config)#ip snat eth0_snat tcp 192.168.10.0/24 any 192.168.10.10 80 192.168.10.1
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
nxr130(config-if)#ip dnat eth0_dnat
nxr130(config-if)#ip snat eth0_snat
nxr130(config-if)#exit
nxr130(config)#ip access-list ppp0_forward-in permit any 192.168.10.10 tcp any 80
nxr130(config)#interface ppp 0
nxr130(config-ppp)#ip address 10.10.10.1/32
nxr130(config-ppp)#ip dnat ppp0_dnat
nxr130(config-ppp)#ip masquerade
nxr130(config-ppp)#ip access-group forward-in ppp0_forward-in
nxr130(config-ppp)#ip spi-filter
nxr130(config-ppp)#ip tcp adjust-mss auto
nxr130(config-ppp)#no ip redirects
nxr130(config-ppp)#ppp username test1@centurysys password test1pass
nxr130(config-ppp)#exit
nxr130(config)#interface ethernet 1
nxr130(config-if)#no ip address
nxr130(config-if)#pppoe-client ppp 0
nxr130(config-if)#exit
nxr130(config)#ip route 0.0.0.0/0 ppp 0

----- 次のページに続きがあります -----
```

```
----- 前のページからの続きです -----

nxr130(config)#dns
nxr130(dns-config)#service enable
nxr130(dns-config)#exit
nxr130(config)#exit
nxr130#save config
```

【 解説 】

- ppp0 インタフェースで宛先 IPv4 アドレス「10.10.10.1」, TCP ポート番号「80」(WWW サーバ) のパケットを受信した場合は、パケットの宛先 IPv4 アドレスを「192.168.10.10」に変換します。
- Ethernet0 インタフェースで宛先 IPv4 アドレス「10.10.10.1」, TCP ポート番号「80」のパケットを受信した場合は、パケットの送信元 IPv4 アドレス「192.168.10.1」, 宛先 IPv4 アドレスを「192.168.10.10」に変換します。

＜DNAT 設定＞

```
nxr130(config)#ip dnat ppp0_dnat tcp any any 10.10.10.1 80 192.168.10.10 80
```

DNAT 名を「ppp0_dnat」とし、「10.10.10.1」, TCP ポート番号「80」のパケットは、宛先 IPv4 アドレス「192.168.10.10」, TCP ポート番号「80」に変換します。

```
nxr130(config)#ip dnat eth0_dnat tcp 192.168.10.0/24 any 10.10.10.1 80 192.168.10.10
```

DNAT 名を「eth0_dnat」とし、送信元 IPv4 アドレスが「192.168.10.0/24」,宛先 IPv4 アドレス「10.10.10.1」, TCP ポート番号「80」のパケットは、宛先 IPv4 アドレス「192.168.10.10」に変換します。

＜SNAT 設定＞

```
nxr130(config)#ip snat eth0_snat tcp 192.168.10.0/24 any 192.168.10.10 80 192.168.10.1
```

SNAT 名を「eth0_snat」とし、送信元 IPv4 アドレスが「192.168.10.0/24」, 宛先 IPv4 アドレス「192.168.10.10」, TCP ポート番号「80」のパケットは、送信元 IPv4 アドレス「192.168.10.1」に変換します。

＜Ethernet0 インタフェース設定＞

```
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.1/24」を設定します。

```
nxr130(config-if)#ip dnat eth0_dnat
nxr130(config-if)#ip snat eth0_snat
```

DNAT 設定で設定した「eth0_dnat」、SNAT 設定で設定した「eth0_snat」を Ethernet0 インタフェースに適用します。

＜IPv4 アクセスリスト設定＞

```
nxr130(config)#ip access-list ppp0_forward-in permit any 192.168.10.10 tcp any 80
```

IPv4 アクセスリスト名を「ppp0_forward-in」とし、宛先 IPv4 アドレス「192.168.10.10」, TCP ポート番号「80」のパケットは許可します。

＜ppp0 インタフェース設定＞

```
nxr130(config)#interface ppp 0
nxr130(config-ppp)#ip address 10.10.10.1/32
nxr130(config-ppp)#ip dnat ppp0_dnat
nxr130(config-ppp)#ip masquerade
nxr130(config-ppp)#ip access-group forward-in ppp0_forward-in
nxr130(config-ppp)#ip spi-filter
nxr130(config-ppp)#ip tcp adjust-mss auto
nxr130(config-ppp)#no ip redirects
nxr130(config-ppp)#ppp username test1@centurysys password test1pass
```

ppp0 インタフェースに関する設定をします。

＜Ethernet1 インタフェース設定＞

```
nxr130(config)#interface ethernet 1
nxr130(config-if)#no ip address
nxr130(config-if)#pppoe-client ppp 0
```

Ethernet1 インタフェースに関する設定をします。

＜スタティックルート設定＞

```
nxr130(config)#ip route 0.0.0.0/0 ppp 0
```

デフォルトルートを設定します。

＜DNS 設定＞

```
nxr130(config)#dns
nxr130(dns-config)#service enable
```

DNS サービスを有効にします。

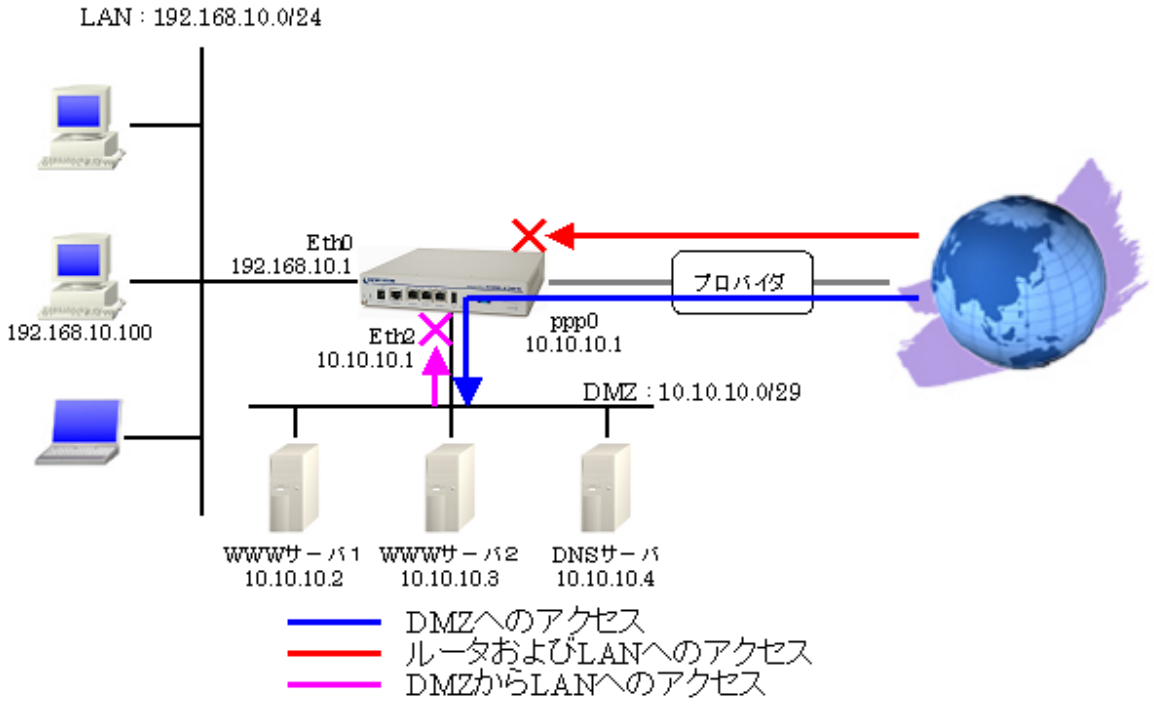
5-4-3. サーバ，パソコンの設定例

	WWW サーバ	パソコン
IPv4 アドレス	192.168.10.10	192.168.10.100
サブネットマスク	255.255.255.0	255.255.255.0
デフォルトゲートウェイ	192.168.10.1	192.168.10.1
DNS サーバの IPv4 アドレス	－	192.168.10.1

5-5. DMZ 構築例（PPPoE）設定

NXR-130/C のように３ポート（３セグメント）以上を有する製品では、インターネットに公開するサーバ群（DMZ）と社内 LAN を物理的に分けて構築することが可能です。

5-5-1. 構成図



5-5-2. 設定例

```
nxr130#configure terminal
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
nxr130(config-if)#exit
nxr130(config)#ip snat ppp0_snat ip 192.168.10.0/24 any 10.10.10.1
nxr130(config)#ip access-list ppp0_in deny any 10.10.10.1 icmp 8 0
nxr130(config)#ip access-list ppp0_forward-in permit any 10.10.10.2 tcp any 80
nxr130(config)#ip access-list ppp0_forward-in permit any 10.10.10.3 tcp any 80
nxr130(config)#ip access-list ppp0_forward-in permit any 10.10.10.4 udp any 53
nxr130(config)#ip access-list ppp0_forward-in permit any 10.10.10.0/29 icmp
nxr130(config)#interface ppp 0
nxr130(config-ppp)#ip address 10.10.10.1/32
nxr130(config-ppp)#ip snat ppp0_snat
nxr130(config-ppp)#ip access-group in ppp0_in
nxr130(config-ppp)#ip access-group forward-in ppp0_forward-in
nxr130(config-ppp)#ip spi-filter
nxr130(config-ppp)#ip tcp adjust-mss auto
nxr130(config-ppp)#no ip redirects
nxr130(config-ppp)#ppp authentication pap
nxr130(config-ppp)#ppp username test1@centurysys password test1pass
nxr130(config-ppp)#exit
nxr130(config)#interface ethernet 1

----- 次のページに続きがあります -----
```

```
----- 前のページからの続きです -----

nxr130(config-if)#no ip address
nxr130(config-if)#pppoe-client ppp 0
nxr130(config-if)#exit
nxr130(config)#interface ethernet 2
nxr130(config-if)#ip address 10.10.10.1/29
nxr130(config-if)#ip spi-filter
nxr130(config-if)#exit
nxr130(config)#ip route 0.0.0.0/0 ppp 0
nxr130(config)#dns
nxr130(dns-config)#address 10.10.10.4
nxr130(dns-config)#service enable
nxr130(dns-config)#exit
nxr130(config)#exit
nxr130#save config
```

【 解説 】

- Ethernet0 を LAN 側, Ethernet1 を WAN 側, Ethernet2 を DMZ 側とします。
- 送信元 IPv4 アドレスが「192.168.10.0/24」のパケットは、IPv4 アドレス「10.10.10.1」に変換します。
- ppp0 インタフェースで宛先 IPv4 アドレス「10.10.10.1」の ICMP Echo Request は破棄しますが、それ以外の「10.10.10.0/29」宛の ICMP パケットは許可します。
- ppp0 インタフェースで宛先 IPv4 アドレス「10.10.10.2」および「10.10.10.3」, TCP ポート番号「80」と「10.10.10.4」, UDP ポート番号「53」へのアクセスは許可します。
- Ethernet2,ppp0 インタフェースでステートフルパケットインスペクションを有効にしています。

＜Ethernet0 インタフェース設定＞

```
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.1/24」を設定します。

＜SNAT 設定＞

```
nxr130(config)#ip snat ppp0_snat ip 192.168.10.0/24 any 10.10.10.1
```

SNAT 名を「ppp0_snat」とし、送信元 IPv4 アドレス「192.168.10.0/24」のパケットは、IPv4 アドレス「10.10.10.1」に変換します。

＜IPv4 アクセスリスト設定＞

```
nxr130(config)#ip access-list ppp0_in deny any 10.10.10.1 icmp 8 0
```

IPv4 アクセスリスト名を「ppp0_in」とし、宛先 IPv4 アドレス「10.10.10.1」, ICMP Type「8」Code「0」(Echo Request)のパケットは破棄します。

```
nxr130(config)#ip access-list ppp0_forward-in permit any 10.10.10.2 tcp any 80
nxr130(config)#ip access-list ppp0_forward-in permit any 10.10.10.3 tcp any 80
nxr130(config)#ip access-list ppp0_forward-in permit any 10.10.10.4 udp any 53
nxr130(config)#ip access-list ppp0_forward-in permit any 10.10.10.0/29 icmp
```

IPv4 アクセスリスト名を「ppp0_forward-in」とし、以下のルールで設定します。

動作	送信元 IPv4 アドレス	宛先 IPv4 アドレス	プロトコル	送信元ポート	宛先ポート
許可	any	10.10.10.2	tcp	any	80
許可	any	10.10.10.3	tcp	any	80
許可	any	10.10.10.4	udp	any	53
許可	any	10.10.10.0/29	icmp	-	-

＜ppp0 インタフェース設定＞

```
nxr130(config)#interface ppp 0
nxr130(config-ppp)#ip address 10.10.10.1/32
nxr130(config-ppp)#ip snat ppp0_snat
nxr130(config-ppp)#ip access-group in ppp0_in
nxr130(config-ppp)#ip access-group forward-in ppp0_forward-in
nxr130(config-ppp)#ip spi-filter
nxr130(config-ppp)#ip tcp adjust-mss auto
nxr130(config-ppp)#no ip redirects
nxr130(config-ppp)#ppp authentication pap
nxr130(config-ppp)#ppp username test1@centurysys password test1pass
```

ppp0 インタフェースに関する設定をします。

＜Ethernet1 インタフェース設定＞

```
nxr130(config)#interface ethernet 1
nxr130(config-if)#no ip address
nxr130(config-if)#pppoe-client ppp 0
```

Ethernet1 インタフェースに関する設定をします。

＜Ethernet2 インタフェース設定＞

```
nxr130(config)#interface ethernet 2
nxr130(config-if)#ip address 10.10.10.1/29
nxr130(config-if)#ip spi-filter
```

Ethernet2 インタフェースに関する設定をします。

ステートフルパケットインスペクションを設定することにより DMZ→LAN,DMZ→WAN への不要な通信を遮断します。

＜スタティックルート設定＞

```
nxr130(config)#ip route 0.0.0.0/0 ppp 0
```

デフォルトルートを設定します。

＜DNS 設定＞

```
nxr130(config)#dns
nxr130(dns-config)#address 10.10.10.4
nxr130(dns-config)#service enable
```

DNS サーバの IPv4 アドレス「10.10.10.4」を設定し、DNS サービスを有効にします。

5-5-3. サーバ，パソコンの設定例

DMZ	WWW サーバ１	WWW サーバ２	DNS サーバ
IPv4 アドレス	10.10.10.2	10.10.10.3	10.10.10.4
サブネットマスク	255.255.255.248	255.255.255.248	255.255.255.248
デフォルトゲートウェイ	10.10.10.1	10.10.10.1	10.10.10.1

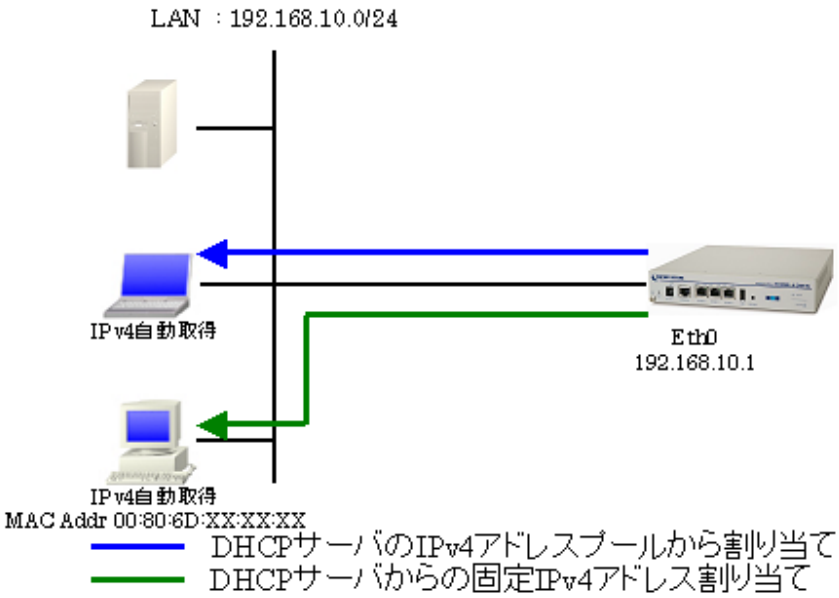
LAN	パソコン
IPv4 アドレス	192.168.10.100
サブネットマスク	255.255.255.0
デフォルトゲートウェイ	192.168.10.1
DNS サーバの IPv4 アドレス	192.168.10.1

6. DHCP 設定

6-1. DHCP サーバ設定

DHCP サーバ機能では、LAN 内の端末に自動的に IPv4 アドレス等の設定をすることが可能です。
また IPv4 アドレスの固定割り当ての設定を行うことにより、特定の端末に対して常に同じ IPv4 アドレスを割り当てることができます。

6-1-1. 構成図



6-1-2. 設定例

```
nxr130#configure terminal
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
nxr130(config-if)#exit
nxr130(config)#dhcp-server bind 00:80:6D:XX:XX:XX 192.168.10.250
nxr130(config)#dhcp-server 1
nxr130(dhcps-config)#network 192.168.10.0/24 range 192.168.10.200 192.168.10.210
nxr130(dhcps-config)#gateway 192.168.10.1
nxr130(dhcps-config)#dns-server 192.168.10.1 10.10.10.100
nxr130(dhcps-config)#exit
nxr130(config)#exit
nxr130#save config
```

【 解説 】

- DHCP サーバで IPv4 アドレス，ゲートウェイアドレス，DNS サーバの IPv4 アドレスを払い出す設定をします。
- MAC アドレス「00:80:6D:XX:XX:XX」の端末にのみ IPv4 アドレスを固定割り当てし、その他の端末は IP アドレスプールから割り当てるように設定します。

＜Ethernet0 インタフェース設定＞

```
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.1/24」を設定します。

＜DHCP サーバ設定＞

```
nxr130(config)#dhcp-server bind 00:80:6D:XX:XX:XX 192.168.10.250
```

MAC アドレス「00:80:6D:XX:XX:XX」の端末に IPv4 アドレス「192.168.10.250」を固定割り当てするための設定をします。

```
nxr130(config)#dhcp-server 1
```

DHCP サーバのサーバナンバ「１」に設定します。

```
nxr130(dhcps-config)#network 192.168.10.0/24 range 192.168.10.200 192.168.10.210
nxr130(dhcps-config)#gateway 192.168.10.1
nxr130(dhcps-config)#dns-server 192.168.10.1 10.10.10.100
```

DHCP サーバで配布するアドレス情報を設定します。

項目	設定内容
IPv4 アドレスリース範囲	192.168.10.200～210
デフォルトゲートウェイアドレス	192.168.10.1
DNS サーバの IPv4 アドレス	192.168.10.1（プライマリ） 10.10.10.100（セカンダリ）

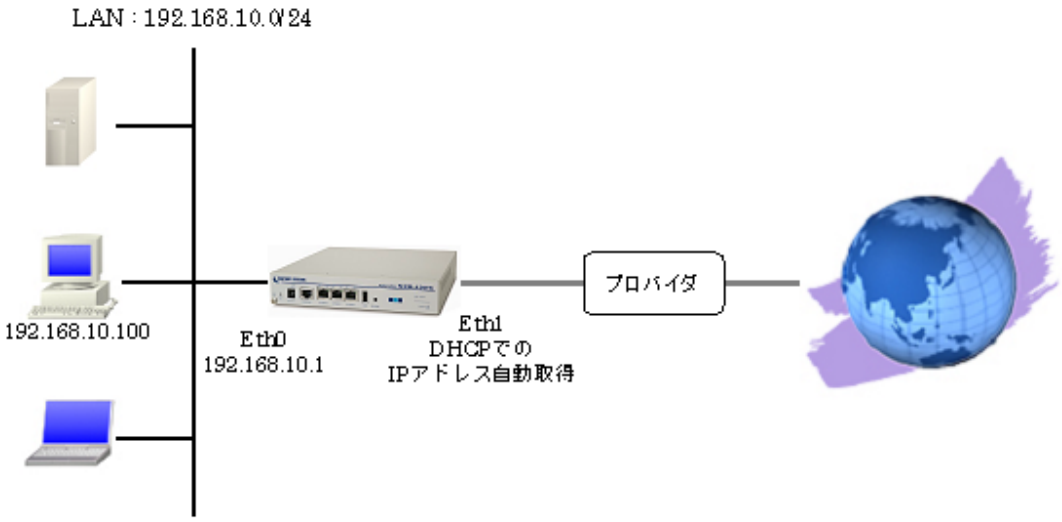
6-1-3. パソコンの設定例

	パソコン
IPv4 アドレス	DHCP サーバから自動取得
サブネットマスク	
デフォルトゲートウェイ	
DNS サーバの IPv4 アドレス	

6-2. DHCP クライアント設定

CATV など IP アドレスが DHCP で払い出される場合には、DHCP クライアントの設定をします。

6-2-1. 構成図



6-2-2. 設定例

```
nxr130#configure terminal
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
nxr130(config-if)#exit
nxr130(config)#interface ethernet 1
nxr130(config-if)#ip address dhcp
nxr130(config-if)#ip masquerade
nxr130(config-if)#ip spi-filter
nxr130(config-if)#no ip redirects
nxr130(config-if)#exit
nxr130(config)#dns
nxr130(dns-config)#service enable
nxr130(dns-config)#exit
nxr130(config)#exit
nxr130#save config
```

【 解説 】

- Ethernet0 を LAN 側, Ethernet1 を WAN 側とします。
- Ethernet1 インタフェースを DHCP クライアントとして設定します。
- IP マスカレード,ステートフルパケットインスペクションを有効にしています。
- DNS サービスを有効にします。

＜Ethernet0 インタフェース設定＞

```
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.1/24」を設定します。

＜Ethernet1 インタフェース設定＞

```
nxr130(config)#interface ethernet 1
nxr130(config-if)#ip address dhcp
nxr130(config-if)#ip masquerade
nxr130(config-if)#ip spi-filter
nxr130(config-if)#no ip redirects
```

Ethernet1 インタフェースに関する設定をします。

DHCP クライアントの設定をし、IP アドレスを自動取得できるようにします。

＜DNS 設定＞

```
nxr130(config)#dns
nxr130(dns-config)#service enable
```

DNS サービスを有効にします。

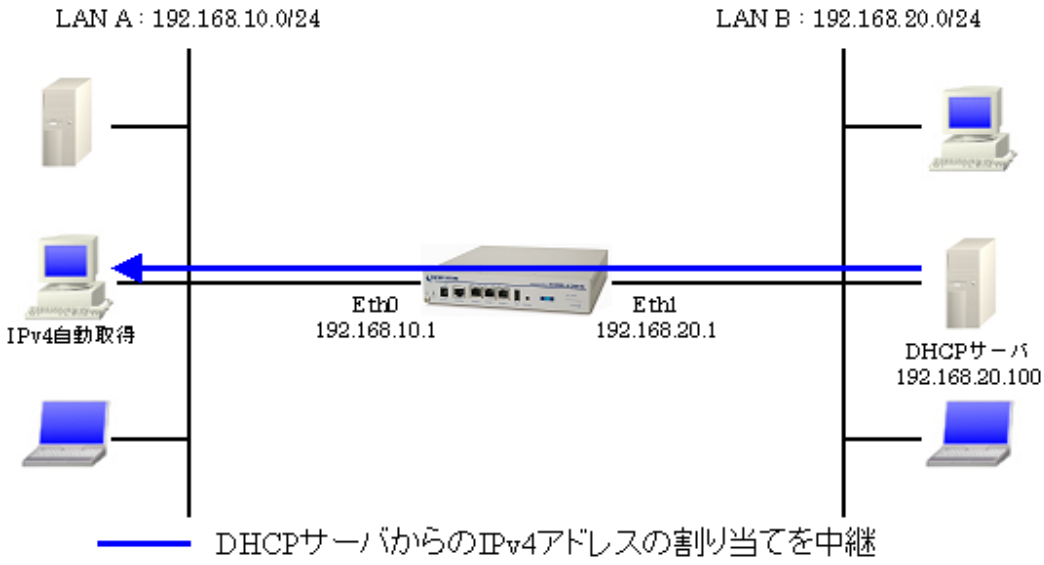
6-2-3. パソコンの設定例

IPv4 アドレス	192.168.10.100
サブネットマスク	255.255.255.0
デフォルトゲートウェイ	192.168.10.1
DNS サーバの IPv4 アドレス	192.168.10.1

6-3. DHCP リレー設定

DHCP リレー機能では異なるネットワークにある DHCP サーバで IP アドレスを一括管理している場合など、ルータ経由で端末に IP アドレスを払い出す必要がある場合に利用することができます。

6-3-1. 構成図



6-3-2. 設定例

```
nxr130#configure terminal
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
nxr130(config-if)#exit
nxr130(config)#interface ethernet 1
nxr130(config-if)#ip address 192.168.20.1/24
nxr130(config-if)#exit
nxr130(config)#dhcp-relay
nxr130(dhcpr-config)#address 192.168.20.100
nxr130(dhcpr-config)#exit
nxr130(config)#exit
nxr130#save config
```

【 解説 】

- DHCP で IPv4 アドレスの取得要求があった場合に、DHCP サーバの IP アドレス「192.168.20.100」にパケットを転送します。

＜Ethernet0 インタフェース設定＞

```
nxr130(config)#interface ethernet 0
nxr130(config-if)#ip address 192.168.10.1/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.1/24」を設定します。

＜Ethernet1 インタフェース設定＞

```
nxr130(config)#interface ethernet 1
nxr130(config-if)#ip address 192.168.20.1/24
```

Ethernet1 インタフェースの IPv4 アドレスとして「192.168.20.1/24」を設定します。

＜DHCP リレー設定＞

```
nxr130(config)#dhcp-relay
nxr130(dhcpr-config)#address 192.168.20.100
```

中継する DHCP サーバの IPv4 アドレスとして「192.168.20.100」を設定します。

6-3-3. パソコンの設定例

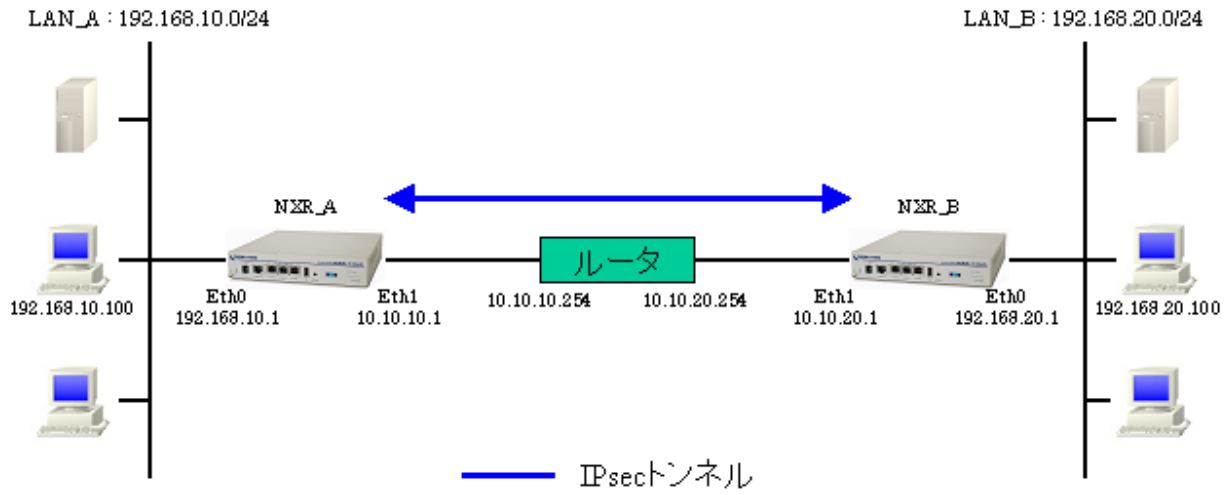
	パソコン
IPv4 アドレス	DHCP サーバから自動取得
サブネットマスク	
デフォルトゲートウェイ	
DNS サーバの IPv4 アドレス	

7. IPsec 設定

7-1. 固定 IPv4 アドレスでの接続設定例（MainMode の利用）

LAN A「192.168.10.0/24」と LAN B「192.168.20.0/24」のネットワークにある NXR_A, NXR_B 間で IPsec トンネルを構築し、LAN 間通信を可能にします。IPsec を使用するルータの WAN 側 IPv4 アドレスはともに固定 IP アドレスになります。

7-1-1. 構成図



7-1-2. 設定例

[NXR_A の設定]

```
nxr130#configure terminal
nxr130(config)#hostname NXR_A
NXR_A(config)#interface ethernet 0
NXR_A(config-if)#ip address 192.168.10.1/24
NXR_A(config-if)#exit
NXR_A(config)#ip route 0.0.0.0/0 10.10.10.254
NXR_A(config)#ipsec access-list LAN_B ip 192.168.10.0/24 192.168.20.0/24
NXR_A(config)#ipsec local policy 1
NXR_A(config-ipsec-local)#address ip
NXR_A(config-ipsec-local)#exit
NXR_A(config)#ipsec isakmp policy 1
NXR_A(config-ipsec-isakmp)#description NXR_B
NXR_A(config-ipsec-isakmp)#authentication pre-share ipseckey
NXR_A(config-ipsec-isakmp)#hash sha1
NXR_A(config-ipsec-isakmp)#encryption aes128
NXR_A(config-ipsec-isakmp)#group 5
NXR_A(config-ipsec-isakmp)#lifetime 10800
NXR_A(config-ipsec-isakmp)#isakmp-mode main
NXR_A(config-ipsec-isakmp)#remote address ip 10.10.20.1
NXR_A(config-ipsec-isakmp)#keepalive 30 3 periodic restart
NXR_A(config-ipsec-isakmp)#local policy 1
NXR_A(config-ipsec-isakmp)#exit
NXR_A(config)#ipsec tunnel policy 1

----- 次のページに続きがあります -----
```

```
----- 前のページからの続きです -----

NXR_A(config-ipsec-tunnel)#description NXR_B
NXR_A(config-ipsec-tunnel)#negotiation-mode auto
NXR_A(config-ipsec-tunnel)#set transform esp-aes128 esp-sha1-hmac
NXR_A(config-ipsec-tunnel)#set pfs group5
NXR_A(config-ipsec-tunnel)#set sa lifetime 3600
NXR_A(config-ipsec-tunnel)#set key-exchange isakmp 1
NXR_A(config-ipsec-tunnel)#match address LAN_B
NXR_A(config-ipsec-tunnel)#exit
NXR_A(config)#interface ethernet 1
NXR_A(config-if)#ip address 10.10.10.1/24
NXR_A(config-if)#ipsec policy 1
NXR_A(config-if)#exit
NXR_A(config)#exit
NXR_A#save config
```

【 解説 】

- フェーズ１ではメインモード，フェーズ２ではクイックモードを利用します。
- ISAKMP ポリシー（フェーズ１）で利用するプロポーザルは以下のとおりです。

認証アルゴリズム	SHA-1
暗号化アルゴリズム	AES-128
Diffie-Hellman(DH)グループ	group5
対向の認証方式	事前共有鍵（Pre-Shared Key）
ライフタイム	10800(s)

- tunnel ポリシー（フェーズ２）で利用するプロポーザルは以下のとおりです。

認証アルゴリズム	ESP-SHA1-HMAC
暗号化アルゴリズム	ESP-AES128
Diffie-Hellman(DH)グループ	group5
ライフタイム	3600(s)

- 事前共有鍵は対向機器と同一のもの（ここでは ipseckey）を使用する必要があります。

＜ホスト名の設定＞

```
nxr130(config)#hostname NXR_A
```

ホスト名として「NXR_A」を設定します。

＜Ethernet0 インタフェース設定＞

```
NXR_A(config)#interface ethernet 0
NXR_A(config-if)#ip address 192.168.10.1/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.1/24」を設定します。

＜スタティックルート設定＞

```
NXR_A(config)#ip route 0.0.0.0/0 10.10.10.254
```

デフォルトルートを設定します。

＜IPsec アクセスリスト設定＞

```
NXR_A(config)#ipsec access-list LAN_B ip 192.168.10.0/24 192.168.20.0/24
```

IPsec アクセスリスト名を「LAN_B」とし、送信元 IPv4 アドレス「192.168.10.0/24」, 宛先 IPv4 アドレス「192.168.20.0/24」を設定します。
またこの送信元, 宛先 IPv4 アドレスが IPsec でのカプセル化対象となります。

＜IPsec ローカルポリシー設定＞

```
NXR_A(config)#ipsec local policy 1
```

IPsec のローカルポリシー「１」を設定します。

```
NXR_A(config-ipsec-local)#address ip
```

IPsec トンネルの送信元 IP として IPv4 アドレスを設定します。

＜IPsec ISAKMP ポリシー設定＞

```
NXR_A(config)#ipsec isakmp policy 1
```

IPsec の ISAKMP ポリシー「１」を設定します。

```
NXR_A(config-ipsec-isakmp)#description NXR_B
```

ISAKMP ポリシー「１」の名前を「NXR_B」と設定します。

```
NXR_A(config-ipsec-isakmp)#authentication pre-share ipseckey
```

事前共有鍵(Pre-Shared Key)として「ipseckey」を設定します。

```
NXR_A(config-ipsec-isakmp)#hash sha1
```

認証アルゴリズムとして「sha1」を設定します。

```
NXR_A(config-ipsec-isakmp)#encryption aes128
```

暗号化アルゴリズムとして「aes128」を設定します。

```
NXR_A(config-ipsec-isakmp)#group 5
```

Diffie-Hellman(DH)グループとして「group 5」を設定します。

```
NXR_A(config-ipsec-isakmp)#lifetime 10800
```

ISAKMP SA のライフタイムとして「10800」秒を設定します。

```
NXR_A(config-ipsec-isakmp)#isakmp-mode main
```

IPsec を使用するルータの WAN 側 IPv4 アドレスがともに固定 IP アドレスのため、フェーズ１のネゴシエーションモードとして「main」を設定します。

```
NXR_A(config-ipsec-isakmp)#remote address ip 10.10.20.1
```

対向機器の IPv4 アドレス「10.10.20.1」を設定します。

```
NXR_A(config-ipsec-isakmp)#keepalive 30 3 periodic restart
```

IKE KeepAlive(DPD)を設定します。監視は 30 秒間隔で 3 回リトライ（間隔およびリトライ回数はデフォルト値）を行い、keepalive 失敗時に SA を削除し、IKE の ネゴシエーションを開始します。

```
NXR_A(config-ipsec-isakmp)#local policy 1
```

使用する IPsec ローカルポリシーとして「1」(ipsec local policy 1) を設定します。

＜IPsec tunnel ポリシー設定＞

```
NXR_A(config)#ipsec tunnel policy 1
```

IPsec の tunnel ポリシー「1」を設定します。

```
NXR_A(config-ipsec-tunnel)#description NXR_B
```

tunnel ポリシー「1」の名前を「NXR_B」とします。

```
NXR_A(config-ipsec-tunnel)#negotiation-mode auto
```

ネゴシエーションモードを「auto」に設定します。これによりこちらからネゴシエーションを開始することができます。

```
NXR_A(config-ipsec-tunnel)#set transform esp-aes128 esp-sha1-hmac
```

トランスフォームとして暗号化アルゴリズム「esp-aes128」, 認証アルゴリズム「esp-sha1-hmac」を設定します。

```
NXR_A(config-ipsec-tunnel)#set pfs group5
```

PFS を有効とし、DH グループとして「group5」を設定します。

```
NXR_A(config-ipsec-tunnel)#set sa lifetime 3600
```

IPsec SA のライフタイムとして「3600」秒を設定します。

```
NXR_A(config-ipsec-tunnel)#set key-exchange isakmp 1
```

使用する IPsec ISAKMP ポリシーとして「1」(ipsec isakmp policy 1) を設定します。

```
NXR_A(config-ipsec-tunnel)#match address LAN_B
```

使用する IPsec アクセスリストとして「LAN_B」を設定します。

＜Ethernet1 インタフェース設定＞

```
NXR_A(config)#interface ethernet 1
NXR_A(config-if)#ip address 10.10.10.1/24
```

Ethernet1 インタフェースの IPv4 アドレスとして「10.10.10.1/24」を設定します。

```
NXR_A(config-if)#ipsec policy 1
```

IPsec ローカルポリシー「１」を適用します。これによりこのインタフェースが IPsec トンネルのエンドポイントとなります。

[NXR_B の設定]

```
nxr130#configure terminal
nxr130(config)#hostname NXR_B
NXR_B(config)#interface ethernet 0
NXR_B(config-if)#ip address 192.168.20.1/24
NXR_B(config-if)#exit
NXR_B(config)#ip route 0.0.0.0/0 10.10.20.254
NXR_B(config)#ipsec access-list LAN_A ip 192.168.20.0/24 192.168.10.0/24
NXR_B(config)#ipsec local policy 1
NXR_B(config-ipsec-local)#address ip
NXR_B(config-ipsec-local)#exit
NXR_B(config)#ipsec isakmp policy 1
NXR_B(config-ipsec-isakmp)#description NXR_A
NXR_B(config-ipsec-isakmp)#authentication pre-share ipseckey
NXR_B(config-ipsec-isakmp)#hash sha1
NXR_B(config-ipsec-isakmp)#encryption aes128
NXR_B(config-ipsec-isakmp)#group 5
NXR_B(config-ipsec-isakmp)#lifetime 10800
NXR_B(config-ipsec-isakmp)#isakmp-mode main
NXR_B(config-ipsec-isakmp)#remote address ip 10.10.10.1
NXR_B(config-ipsec-isakmp)#keepalive 30 3 periodic restart
NXR_B(config-ipsec-isakmp)#local policy 1
NXR_B(config-ipsec-isakmp)#exit
NXR_B(config)#ipsec tunnel policy 1
NXR_B(config-ipsec-tunnel)#description NXR_A
NXR_B(config-ipsec-tunnel)#negotiation-mode auto
NXR_B(config-ipsec-tunnel)#set transform esp-aes128 esp-sha1-hmac
NXR_B(config-ipsec-tunnel)#set pfs group5
NXR_B(config-ipsec-tunnel)#set sa lifetime 3600
NXR_B(config-ipsec-tunnel)#set key-exchange isakmp 1
NXR_B(config-ipsec-tunnel)#match address LAN_A
NXR_B(config-ipsec-tunnel)#exit
NXR_B(config)#interface ethernet 1
NXR_B(config-if)#ip address 10.10.20.1/24
NXR_B(config-if)#ipsec policy 1
NXR_B(config-if)#exit
NXR_B(config)#exit
NXR_B#save config
```

【 解説 】

- フェーズ１ではメインモード，フェーズ２ではクイックモードを利用します。
- ISAKMP ポリシー(フェーズ１)，tunnel ポリシー(フェーズ２)で利用するプロポーザルは「NXR_A」と同一です。
- 事前共有鍵は対向機器と同一のものを使用する必要があります。

＜ホスト名の設定＞

```
nxr130(config)#hostname NXR_B
```

ホスト名として「NXR_B」を設定します。

＜Ethernet0 インタフェース設定＞

```
NXR_B(config)#interface ethernet 0
NXR_B(config-if)#ip address 192.168.20.1/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.20.1/24」を設定します。

＜スタティックルート設定＞

```
NXR_B(config)#ip route 0.0.0.0/0 10.10.20.254
```

デフォルトルートを設定します。

＜IPsec アクセスリスト設定＞

```
NXR_B(config)#ipsec access-list LAN_A ip 192.168.20.0/24 192.168.10.0/24
```

IPsec アクセスリスト名を「LAN_A」とし、送信元 IPv4 アドレス「192.168.20.0/24」, 宛先 IPv4 アドレス「192.168.10.0/24」を設定します。

またこの送信元, 宛先 IPv4 アドレスが IPsec でのカプセル化対象となります。

＜IPsec ローカルポリシー設定＞

```
NXR_B(config)#ipsec local policy 1
NXR_B(config-ipsec-local)#address ip
```

IPsec のローカルポリシー「１」を設定します。

＜IPsec ISAKMP ポリシー設定＞

```
NXR_B(config)#ipsec isakmp policy 1
NXR_B(config-ipsec-isakmp)#description NXR_A
NXR_B(config-ipsec-isakmp)#authentication pre-share ipseckey
NXR_B(config-ipsec-isakmp)#hash sha1
NXR_B(config-ipsec-isakmp)#encryption aes128
NXR_B(config-ipsec-isakmp)#group 5
NXR_B(config-ipsec-isakmp)#lifetime 10800
NXR_B(config-ipsec-isakmp)#isakmp-mode main
NXR_B(config-ipsec-isakmp)#remote address ip 10.10.10.1
NXR_B(config-ipsec-isakmp)#keepalive 30 3 periodic restart
NXR_B(config-ipsec-isakmp)#local policy 1
```

IPsec の ISAKMP ポリシー「１」を設定します。

IPsec を使用するルータの WAN 側 IPv4 アドレスがともに固定 IP アドレスのため、フェーズ１のネゴシエーションモードとして「main」を設定します。

＜IPsec tunnel ポリシー設定＞

```
NXR_B(config)#ipsec tunnel policy 1
NXR_B(config-ipsec-tunnel)#description NXR_A
NXR_B(config-ipsec-tunnel)#negotiation-mode auto
NXR_B(config-ipsec-tunnel)#set transform esp-aes128 esp-sha1-hmac
NXR_B(config-ipsec-tunnel)#set pfs group5
NXR_B(config-ipsec-tunnel)#set sa lifetime 3600
NXR_B(config-ipsec-tunnel)#set key-exchange isakmp 1
NXR_B(config-ipsec-tunnel)#match address LAN_A
```

IPsec の tunnel ポリシー「１」を設定します。

＜Ethernet1 インタフェース設定＞

```
NXR_B(config)#interface ethernet 1
NXR_B(config-if)#ip address 10.10.20.1/24
NXR_B(config-if)#ipsec policy 1
```

Ethernet1 インタフェースの IPv4 アドレスとして「10.10.20.1/24」を設定します。

IPsec ローカルポリシー「１」を適用します。これによりこのインタフェースが IPsec トンネルのエンドポイントとなります。

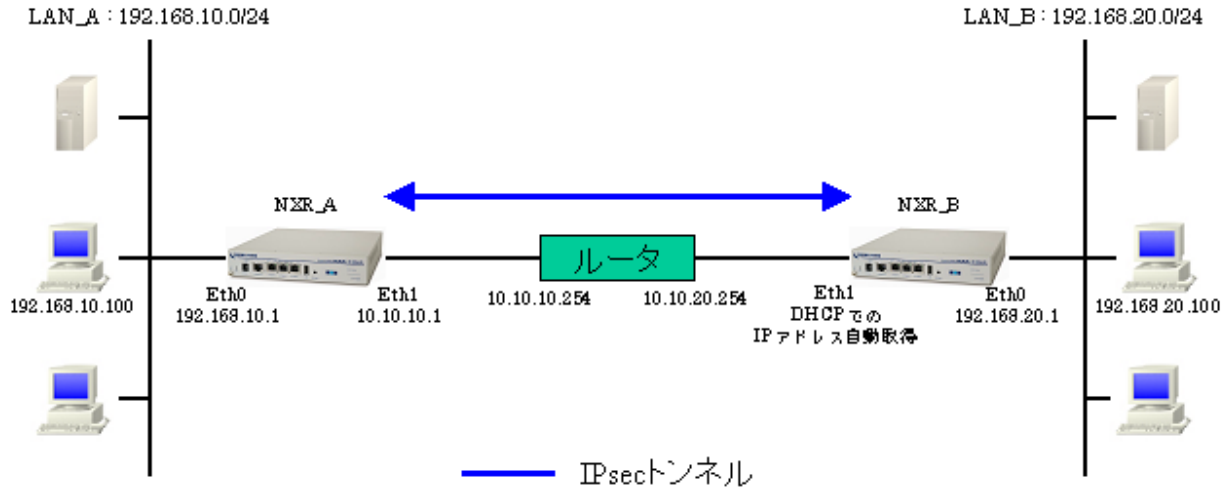
1-1-3. パソコンの設定例

	LAN A のパソコン	LAN B のパソコン
IPv4 アドレス	192.168.10.100	192.168.20.100
サブネットマスク	255.255.255.0	255.255.255.0
デフォルトゲートウェイ	192.168.10.1	192.168.20.1

7-2. 動的 IPv4 アドレスでの接続設定例（AggressiveMode の利用）

片方の機器の WAN 側 IPv4 アドレスが動的 IP アドレスの場合でも IPsec を利用することは可能です。ただし、この時対向機器の WAN 側 IPv4 アドレスは固定 IP アドレスが必須となります。

7-2-1. 構成図



7-2-2. 設定例

```
nxr130#configure terminal
nxr130(config)#hostname NXR_A
NXR_A(config)#interface ethernet 0
NXR_A(config-if)#ip address 192.168.10.1/24
NXR_A(config-if)#exit
NXR_A(config)#ip route 0.0.0.0/0 10.10.10.254
NXR_A(config)#ipsec access-list LAN_B ip 192.168.10.0/24 192.168.20.0/24
NXR_A(config)#ipsec local policy 1
NXR_A(config-ipsec-local)#address ip
NXR_A(config-ipsec-local)#exit
NXR_A(config)#ipsec isakmp policy 1
NXR_A(config-ipsec-isakmp)#description NXR_B
NXR_A(config-ipsec-isakmp)#authentication pre-share ipseckey
NXR_A(config-ipsec-isakmp)#hash sha1
NXR_A(config-ipsec-isakmp)#encryption aes128
NXR_A(config-ipsec-isakmp)#group 5
NXR_A(config-ipsec-isakmp)#lifetime 10800
NXR_A(config-ipsec-isakmp)#isakmp-mode aggressive
NXR_A(config-ipsec-isakmp)#remote address ip any
NXR_A(config-ipsec-isakmp)#remote identity fqdn nxrb
NXR_A(config-ipsec-isakmp)#keepalive 30 3 periodic clear
NXR_A(config-ipsec-isakmp)#local policy 1
NXR_A(config-ipsec-isakmp)#exit
NXR_A(config)#ipsec tunnel policy 1
NXR_A(config-ipsec-tunnel)#description NXR_B
NXR_A(config-ipsec-tunnel)#negotiation-mode manual
NXR_A(config-ipsec-tunnel)#set transform esp-aes128 esp-sha1-hmac
NXR_A(config-ipsec-tunnel)#set pfs group5

----- 次のページに続きがあります -----
```

```
----- 前のページからの続きです -----

NXR_A(config-ipsec-tunnel)#set sa lifetime 3600
NXR_A(config-ipsec-tunnel)#set key-exchange isakmp 1
NXR_A(config-ipsec-tunnel)#match address LAN_B
NXR_A(config-ipsec-tunnel)#exit
NXR_A(config)#interface ethernet 1
NXR_A(config-if)#ip address 10.10.10.1/24
NXR_A(config-if)#ipsec policy 1
NXR_A(config-if)#exit
NXR_A(config)#exit
NXR_A#save config
```

【 解説 】

- フェーズ１ではアグレッシブモード，フェーズ２ではクイックモードを利用します。
- ISAKMP ポリシー（フェーズ１）で利用するプロポーザルは以下のとおりです。

認証アルゴリズム	SHA-1
暗号化アルゴリズム	AES-128
Diffie-Hellman(DH)グループ	group5
対向の認証方式	事前共有鍵（Pre-Shared Key）
ライフタイム	10800(s)

- tunnel ポリシー（フェーズ２）で利用するプロポーザルは以下のとおりです。

認証アルゴリズム	ESP-SHA1-HMAC
暗号化アルゴリズム	ESP-AES128
Diffie-Hellman(DH)グループ	group5
ライフタイム	3600(s)

- 対向機器の IP アドレスを ID として利用することができない（動的 IPv4 アドレス）ため、ISAKMP ポリシーで ID（ID Type ： FQDN）を設定します。
- 事前共有鍵は対向機器と同一のもの（ここでは ipseckey）を使用する必要があります。

＜ホスト名の設定＞

```
nxr130(config)#hostname NXR_A
```

ホスト名として「NXR_A」を設定します。

＜Ethernet0 インタフェース設定＞

```
NXR_A(config)#interface ethernet 0
NXR_A(config-if)#ip address 192.168.10.1/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.1/24」を設定します。

＜スタティックルート設定＞

```
NXR_A(config)#ip route 0.0.0.0/0 10.10.10.254
```

デフォルトルートを設定します。

＜IPsec アクセスリスト設定＞

```
NXR_A(config)#ipsec access-list LAN_B ip 192.168.10.0/24 192.168.20.0/24
```

IPsec アクセスリスト名を「LAN_B」とし、送信元 IPv4 アドレス「192.168.10.0/24」, 宛先 IPv4 アドレス「192.168.20.0/24」を設定します。
またこの送信元, 宛先 IPv4 アドレスが IPsec でのカプセル化対象となります。

＜IPsec ローカルポリシー設定＞

```
NXR_A(config)#ipsec local policy 1
```

IPsec のローカルポリシー「１」を設定します。

```
NXR_A(config-ipsec-local)#address ip
```

IPsec トンネルの送信元 IP として IPv4 アドレスを設定します。

＜IPsec ISAKMP ポリシー設定＞

```
NXR_A(config)#ipsec isakmp policy 1
```

IPsec の ISAKMP ポリシー「１」を設定します。

```
NXR_A(config-ipsec-isakmp)#description NXR_B
```

ISAKMP ポリシー「１」の名前を「NXR_B」と設定します。

```
NXR_A(config-ipsec-isakmp)#authentication pre-share ipseckey
```

事前共有鍵(Pre-Shared Key)として「ipseckey」を設定します。

```
NXR_A(config-ipsec-isakmp)#hash sha1
```

認証アルゴリズムとして「sha1」を設定します。

```
NXR_A(config-ipsec-isakmp)#encryption aes128
```

暗号化アルゴリズムとして「aes128」を設定します。

```
NXR_A(config-ipsec-isakmp)#group 5
```

Diffie-Hellman(DH)グループとして「group 5」を設定します。

```
NXR_A(config-ipsec-isakmp)#lifetime 10800
```

ISAKMP SA のライフタイムとして「10800」秒を設定します。

```
NXR_A(config-ipsec-isakmp)#isakmp-mode aggressive
```

対向機器の WAN 側 IPv4 アドレスが動的 IP アドレスのため、フェーズ１のネゴシエーションモードとして「aggressive」を設定します。

```
NXR_A(config-ipsec-isakmp)#remote address ip any
```

対向機器の IPv4 アドレスは動的 IPv4 アドレスのため、「any」を設定します。

```
NXR_A(config-ipsec-isakmp)#remote identity fqdn nxrb
```
対向機器の IP アドレスを ID として利用することができない(動的 IPv4 アドレス)ため、ID として「nxrb」を設定します。(fqdn 方式で設定しています)

```
NXR_A(config-ipsec-isakmp)#keepalive 30 3 periodic clear
```
IKE KeepAlive(DPD)を設定します。監視は 30 秒間隔で 3 回リトライ（間隔およびリトライ回数はデフォルト値）を行い、keepalive 失敗時に SA を削除します。

```
NXR_A(config-ipsec-isakmp)#local policy 1
```
使用する IPsec ローカルポリシーとして「１」(ipsec local policy 1) を設定します。

＜IPsec tunnel ポリシー設定＞
```
NXR_A(config)#ipsec tunnel policy 1
```
IPsec の tunnel ポリシー「１」を設定します。

```
NXR_A(config-ipsec-tunnel)#description NXR_B
```
tunnel ポリシー「１」の名前を「NXR_B」とします。

```
NXR_A(config-ipsec-tunnel)#negotiation-mode manual
```
ネゴシエーションモードを「manual」に設定します。これによりこちらからはネゴシエーションを開始しなくなります。(対向機器が動的 IPv4 アドレスのため)

```
NXR_A(config-ipsec-tunnel)#set transform esp-aes128 esp-sha1-hmac
```
トランスフォームとして暗号化アルゴリズム「esp-aes128」, 認証アルゴリズム「esp-sha1-hmac」を設定します。

```
NXR_A(config-ipsec-tunnel)#set pfs group5
```
PFS を有効とし、DH グループとして「group5」を設定します。

```
NXR_A(config-ipsec-tunnel)#set sa lifetime 3600
```
IPsec SA のライフタイムとして「3600」秒を設定します。

```
NXR_A(config-ipsec-tunnel)#set key-exchange isakmp 1
```
使用する IPsec ISAKMP ポリシーとして「１」(ipsec isakmp policy 1) を設定します。

```
NXR_A(config-ipsec-tunnel)#match address LAN_B
```
使用する IPsec アクセスリストとして「LAN_B」を設定します。

＜Ethernet1 インタフェース設定＞

```
NXR_A(config)#interface ethernet 1
NXR_A(config-if)#ip address 10.10.10.1/24
```

Ethernet1 インタフェースの IPv4 アドレスとして「10.10.10.1/24」を設定します。

```
NXR_A(config-if)#ipsec policy 1
```

IPsec ローカルポリシー「１」を適用します。これによりこのインタフェースが IPsec トンネルのエンドポイントとなります。

[NXR_B の設定]

```
nxr130#configure terminal
nxr130(config)#hostname NXR_B
NXR_B(config)#interface ethernet 0
NXR_B(config-if)#ip address 192.168.20.1/24
NXR_B(config-if)#exit
NXR_B(config)#ipsec access-list LAN_A ip 192.168.20.0/24 192.168.10.0/24
NXR_B(config)#ipsec local policy 1
NXR_B(config-ipsec-local)#address ip
NXR_B(config-ipsec-local)#self-identity fqdn nxrb
NXR_B(config-ipsec-local)#exit
NXR_B(config)#ipsec isakmp policy 1
NXR_B(config-ipsec-isakmp)#description NXR_A
NXR_B(config-ipsec-isakmp)#authentication pre-share ipseckey
NXR_B(config-ipsec-isakmp)#hash sha1
NXR_B(config-ipsec-isakmp)#encryption aes128
NXR_B(config-ipsec-isakmp)#group 5
NXR_B(config-ipsec-isakmp)#isakmp-mode aggressive
NXR_B(config-ipsec-isakmp)#remote address ip 10.10.10.1
NXR_B(config-ipsec-isakmp)#keepalive 30 3 periodic restart
NXR_B(config-ipsec-isakmp)#local policy 1
NXR_B(config-ipsec-isakmp)#exit
NXR_B(config)#ipsec tunnel policy 1
NXR_B(config-ipsec-tunnel)#description NXR_A
NXR_B(config-ipsec-tunnel)#negotiation-mode auto
NXR_B(config-ipsec-tunnel)#set transform esp-aes128 esp-sha1-hmac
NXR_B(config-ipsec-tunnel)#set pfs group5
NXR_B(config-ipsec-tunnel)#set key-exchange isakmp 1
NXR_B(config-ipsec-tunnel)#match address LAN_A
NXR_B(config-ipsec-tunnel)#exit
NXR_B(config)#interface ethernet 1
NXR_B(config-if)#ip address dhcp
NXR_B(config-if)#ipsec policy 1
NXR_B(config-if)#exit
NXR_B(config)#exit
NXR_B#save config
```

【 解説 】

- フェーズ１ではアグレッシブモード，フェーズ２ではクイックモードを利用します。
- ISAKMP ポリシー(フェーズ１)，tunnel ポリシー(フェーズ２)で利用するプロポーザルは「NXR_A」と同一です。
- 機器の IP アドレスを ID として利用することができない（動的 IPv4 アドレス）ため、ID（ID Type ： FQDN）を設定します。
- 事前共有鍵は対向機器と同一のもの（ここでは ipseckey）を使用する必要があります。

＜ホスト名の設定＞

```
nxr130(config)#hostname NXR_B
```

ホスト名として「NXR_B」を設定します。

＜Ethernet0 インタフェース設定＞

```
NXR_B(config)#interface ethernet 0
NXR_B(config-if)#ip address 192.168.20.1/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.20.1/24」を設定します。

＜IPsec アクセスリスト設定＞

```
NXR_B(config)#ipsec access-list LAN_A ip 192.168.20.0/24 192.168.10.0/24
```

IPsec アクセスリスト名を「LAN_A」とし、送信元 IPv4 アドレス「192.168.20.0/24」, 宛先 IPv4 アドレス「192.168.10.0/24」を設定します。
またこの送信元, 宛先 IPv4 アドレスが IPsec でのカプセル化対象となります。

＜IPsec ローカルポリシー設定＞

```
NXR_B(config)#ipsec local policy 1
```

IPsec のローカルポリシー「１」を設定します。

```
NXR_B(config-ipsec-local)#address ip
```

IPsec トンネルの送信元 IP として IPv4 アドレスを設定します。

```
NXR_B(config-ipsec-local)#self-identity fqdn nxrb
```

機器の IP アドレスを ID として利用することができない（動的 IPv4 アドレス）ため、ID として「nxrb」を設定します。(fqdn 方式で設定しています)

＜IPsec ISAKMP ポリシー設定＞

```
NXR_B(config)#ipsec isakmp policy 1
NXR_B(config-ipsec-isakmp)#description NXR_A
NXR_B(config-ipsec-isakmp)#authentication pre-share ipseckey
NXR_B(config-ipsec-isakmp)#hash sha1
NXR_B(config-ipsec-isakmp)#encryption aes128
NXR_B(config-ipsec-isakmp)#group 5
NXR_B(config-ipsec-isakmp)#isakmp-mode aggressive
NXR_B(config-ipsec-isakmp)#remote address ip 10.10.10.1
NXR_B(config-ipsec-isakmp)#keepalive 30 3 periodic restart
NXR_B(config-ipsec-isakmp)#local policy 1
```

IPsec の ISAKMP ポリシー「１」を設定します。
機器の WAN 側 IPv4 アドレスが動的 IP アドレスのため、フェーズ１のネゴシエーションモードとして「aggressive」を設定します。

＜IPsec tunnel ポリシー設定＞

```
NXR_B(config)#ipsec tunnel policy 1
NXR_B(config-ipsec-tunnel)#description NXR_A
NXR_B(config-ipsec-tunnel)#negotiation-mode auto
NXR_B(config-ipsec-tunnel)#set transform esp-aes128 esp-sha1-hmac
NXR_B(config-ipsec-tunnel)#set pfs group5
NXR_B(config-ipsec-tunnel)#set key-exchange isakmp 1
NXR_B(config-ipsec-tunnel)#match address LAN_A
```

IPsec の tunnel ポリシー「１」を設定します。

＜Ethernet1 インタフェース設定＞

```
NXR_B(config)#interface ethernet 1
NXR_B(config-if)#ip address dhcp
NXR_B(config-if)#ipsec policy 1
```

Ethernet1 インタフェースに関する設定をします。

IPv4 アドレスが動的 IP のため、DHCP クライアントに設定します。

IPsec ローカルポリシー「１」を適用します。これによりこのインタフェースが IPsec トンネルのエンドポイントとなります。

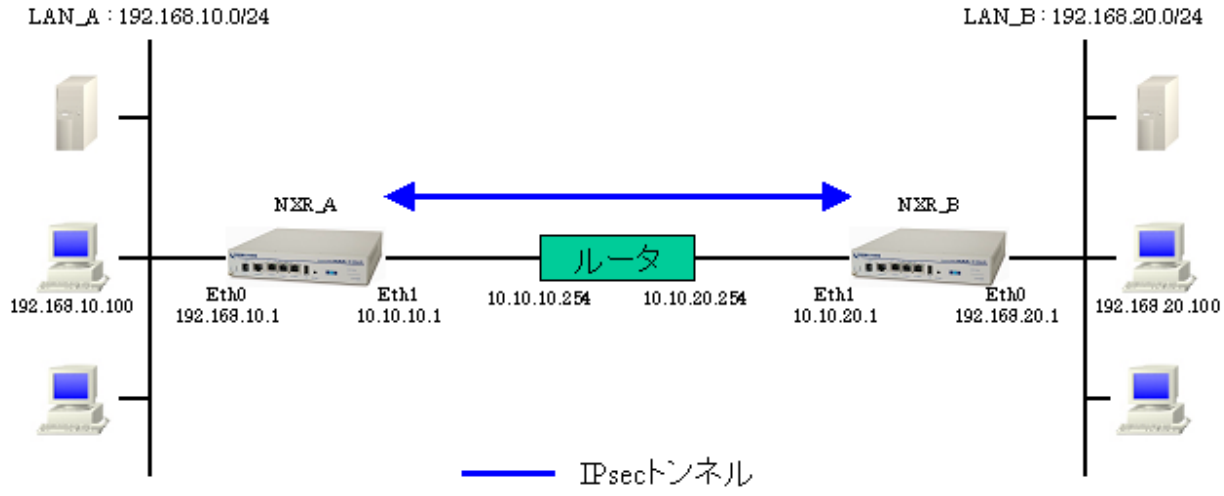
7-2-3. パソコンの設定例

	LAN A のパソコン	LAN B のパソコン
IPv4 アドレス	192.168.10.100	192.168.20.100
サブネットマスク	255.255.255.0	255.255.255.0
デフォルトゲートウェイ	192.168.10.1	192.168.20.1

7-3. RSA 公開鍵暗号方式での接続設定例

IKE のフェーズ１で対向機器の認証に RSA 公開鍵暗号方式を利用することができます。RSA 公開鍵暗号方式を利用する場合は IKE のフェーズ１でメインモードを使用する必要があります。

7-3-1. 構成図



7-3-2. 設定例

[NXR_A の設定]

```
nxr130#configure terminal
nxr130(config)#hostname NXR_A
NXR_A(config)#interface ethernet 0
NXR_A(config-if)#ip address 192.168.10.1/24
NXR_A(config-if)#exit
NXR_A(config)#ip route 0.0.0.0/0 10.10.10.254
NXR_A(config)#ipsec access-list LAN_B ip 192.168.10.0/24 192.168.20.0/24
NXR_A(config)#ipsec generate rsa-sig-key 1024
RSA-SIG KEY generating...
NXR_A(config)#exit
NXR_A#show ipsec rsa-pub-key
RSA public key :
0sAQNyjiS2aqmmPHvKp6GvDIVG6eC6yclJxWRk+syfUozTqPW70R3TcFP74gjNZp3p16GN3SET2/M9+qVQIySERsh3
rBrzEuwzJQ/ShSv7XwJw7Awb2hlsZ8NvFKkIQ9AEGPOF223KT3T807QjxuX5wNToUWqJKZgURAoW1pY9ufM2qw==
NXR_A#configure terminal
NXR_A(config)#ipsec local policy 1
NXR_A(config-ipsec-local)#address ip
NXR_A(config-ipsec-local)#self-identity fqdn nxra
NXR_A(config-ipsec-local)#exit
NXR_A(config)#ipsec isakmp policy 1
NXR_A(config-ipsec-isakmp)#description NXR_B
NXR_A(config-ipsec-isakmp)#authentication rsa-sig 0sAQOZe2V6nfz4pY9P/I5X0NiGgTDjY6yUZ+cPSI
np9dAZqe9QLQwtDitiHZMUo2Liz2/8Nlvq78+Vz7/rdNhoKAPD07cqndlbPR1EnmaLfyNRC2Je19CJyHjCCz0vOL5q
Ob+eFKbAK3icFzi1ryr3tRCA2VIox57Wn2W7KkD96j5urQ==
NXR_A(config-ipsec-isakmp)#hash sha1
NXR_A(config-ipsec-isakmp)#encryption aes128
NXR_A(config-ipsec-isakmp)#group 5

----- 次のページに続きがあります -----
```

```
----- 前のページからの続きです -----

NXR_A(config-ipsec-isakmp)#lifetime 10800
NXR_A(config-ipsec-isakmp)#isakmp-mode main
NXR_A(config-ipsec-isakmp)#remote address ip 10.10.20.1
NXR_A(config-ipsec-isakmp)#remote identity fqdn nxrb
NXR_A(config-ipsec-isakmp)#keepalive 30 3 periodic restart
NXR_A(config-ipsec-isakmp)#local policy 1
NXR_A(config-ipsec-isakmp)#exit
NXR_A(config)#ipsec tunnel policy 1
NXR_A(config-ipsec-tunnel)#description NXR_B
NXR_A(config-ipsec-tunnel)#negotiation-mode auto
NXR_A(config-ipsec-tunnel)#set transform esp-aes128 esp-sha1-hmac
NXR_A(config-ipsec-tunnel)#set pfs group5
NXR_A(config-ipsec-tunnel)#set sa lifetime 3600
NXR_A(config-ipsec-tunnel)#set key-exchange isakmp 1
NXR_A(config-ipsec-tunnel)#match address LAN_B
NXR_A(config-ipsec-tunnel)#exit
NXR_A(config)#interface ethernet 1
NXR_A(config-if)#ip address 10.10.10.1/24
NXR_A(config-if)#ipsec policy 1
NXR_A(config-if)#exit
NXR_A(config)#exit
NXR_A#save config
```

【 解説 】

- フェーズ１ではメインモード，フェーズ２ではクイックモードを利用します。
- 公開鍵は作成後、対向機器の設定で使用します。
- RSA 公開鍵暗号方式を利用する場合は、ID 設定は必須になります。

＜ホスト名の設定＞

```
nxr130(config)#hostname NXR_A
```

ホスト名として「NXR_A」を設定します。

＜Ethernet0 インタフェース設定＞

```
NXR_A(config)#interface ethernet 0
NXR_A(config-if)#ip address 192.168.10.1/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.1/24」を設定します。

＜スタティックルート設定＞

```
NXR_A(config)#ip route 0.0.0.0/0 10.10.10.254
```

デフォルトルートを設定します。

＜IPsec アクセスリスト設定＞

```
NXR_A(config)#ipsec access-list LAN_B ip 192.168.10.0/24 192.168.20.0/24
```

IPsec アクセスリスト名を「LAN_B」とし、送信元 IPv4 アドレス「192.168.10.0/24」，宛先 IPv4 アドレス「192.168.20.0/24」を設定します。

またこの送信元，宛先 IPv4 アドレスが IPsec でのカプセル化対象となります。

＜RSA Signature Key の作成＞

```
NXR_A(config)#ipsec generate rsa-sig-key 1024
```

RSA Signature Key を作成します。ここでは 1024bit で作成します。

＜RSA 公開鍵の確認＞

```
NXR_A#show ipsec rsa-pub-key
RSA public key :
0sAQNyjiS2aqmmPHvKp6GvDIVG6eC6yclJxWRk+syfUozTqPW70R3TcFP74gjNZp3p16GN3SET2/M9+qVQIySERsh3
rBrzEuwzJQ/ShSv7XwJw7Awb2hlsZ8NvFKkIQ9AEGPOF223KT3T807QjxuX5wNToUWqJKZgURAoWlpY9ufM2qw==
```

作成した RSA 公開鍵を確認します。ここで表示した公開鍵は対向機器の IPsec ISAKMP ポリシー設定で利用します。

＜IPsec ローカルポリシー設定＞

```
NXR_A(config)#ipsec local policy 1
NXR_A(config-ipsec-local)#address ip
```

IPsec のローカルポリシー「１」を設定し、IPsec トンネルの送信元 IP として IPv4 アドレスを設定します。

```
NXR_A(config-ipsec-local)#self-identity fqdn nxra
```

機器の ID として「nxra」を設定します（fqdn 方式で設定しています)。

なお RSA 公開鍵暗号方式を利用する場合は、ID 設定は必須になります。

＜IPsec ISAKMP ポリシー設定＞

```
NXR_A(config)#ipsec isakmp policy 1
```

IPsec の ISAKMP ポリシー「１」を設定します。

```
NXR_A(config-ipsec-isakmp)#description NXR_B
```

ISAKMP ポリシー「１」の名前を「NXR_B」と設定します。

```
NXR_A(config-ipsec-isakmp)#authentication rsa-sig 0sAQOZe2V6nfz4pY9P/I5X0NiGgTDjY6yUZ+cPSI
np9dAZqe9QLQwtDitiHZMUo2Liz2/8N1vq78+Vz7/rdNhoKAPD07cqndlbPR1EnmaLfyNRC2Je19CJyHjCCz0vOL5q
0b+eFKbAK3icFzi1ryr3tRCA2VIox57Wn2W7KkD96j5urQ==
```

対向機器の公開鍵を設定します。この設定の前までに対向機器の公開鍵は作成しておく必要があります。

```
NXR_A(config-ipsec-isakmp)#hash sha1
```

認証アルゴリズムとして「sha1」を設定します。

```
NXR_A(config-ipsec-isakmp)#encryption aes128
```

暗号化アルゴリズムとして「aes128」を設定します。

```
NXR_A(config-ipsec-isakmp)#group 5
```

Diffie-Hellman(DH)グループとして「group 5」を設定します。

```
NXR_A(config-ipsec-isakmp)#lifetime 10800
```
ISAKMP SA のライフタイムとして「10800」秒を設定します。

```
NXR_A(config-ipsec-isakmp)#isakmp-mode main
```
RSA 公開鍵暗号方式を利用するため、フェーズ１のネゴシエーションモードとして「main」を設定します。

```
NXR_A(config-ipsec-isakmp)#remote address ip 10.10.20.1
```
対向機器の IPv4 アドレス「10.10.20.1」を設定します。

```
NXR_A(config-ipsec-isakmp)#remote identity fqdn nxrb
```
対向機器の ID として「nxrb」を設定します。(fqdn 方式で設定しています)

```
NXR_A(config-ipsec-isakmp)#keepalive 30 3 periodic restart
```
IKE KeepAlive(DPD)を設定します。監視は 30 秒間隔で 3 回リトライ（間隔およびリトライ回数はデフォルト値）を行い、keepalive 失敗時に SA を削除し、IKE の ネゴシエーションを開始します。

```
NXR_A(config-ipsec-isakmp)#local policy 1
```
使用する IPsec ローカルポリシーとして「１」(ipsec local policy 1) を設定します。

＜IPsec tunnel ポリシー設定＞

```
NXR_A(config)#ipsec tunnel policy 1
```
IPsec の tunnel ポリシー「１」を設定します。

```
NXR_A(config-ipsec-tunnel)#description NXR_B
```
tunnel ポリシー「１」の名前を「NXR_B」とします。

```
NXR_A(config-ipsec-tunnel)#negotiation-mode auto
```
ネゴシエーションモードを「auto」に設定します。これによりこちらからネゴシエーションを開始することができます。

```
NXR_A(config-ipsec-tunnel)#set transform esp-aes128 esp-sha1-hmac
```
トランスフォームとして暗号化アルゴリズム「esp-aes128」, 認証アルゴリズム「esp-sha1-hmac」を設定します。

```
NXR_A(config-ipsec-tunnel)#set pfs group5
```
PFS を有効とし、DH グループとして「group5」を設定します。

```
NXR_A(config-ipsec-tunnel)#set sa lifetime 3600
```
IPsec SA のライフタイムとして「3600」秒を設定します。

```
NXR_A(config-ipsec-tunnel)#set key-exchange isakmp 1
```

使用する IPsec ISAKMP ポリシーとして「1」(ipsec isakmp policy 1) を設定します。

```
NXR_A(config-ipsec-tunnel)#match address LAN_B
```

使用する IPsec アクセスリストとして「LAN_B」を設定します。

＜Ethernet1 インタフェース設定＞

```
NXR_A(config)#interface ethernet 1
NXR_A(config-if)#ip address 10.10.10.1/24
```

Ethernet1 インタフェースの IPv4 アドレスとして「10.10.10.1/24」を設定します。

```
NXR_A(config-if)#ipsec policy 1
```

IPsec ローカルポリシー「1」を適用します。これによりこのインタフェースが IPsec トンネルのエンドポイントとなります。

[NXR_B の設定]

```
nxr130#configure terminal
nxr130(config)#hostname NXR_B
NXR_B(config)#interface ethernet 0
NXR_B(config-if)#ip address 192.168.20.1/24
NXR_B(config-if)#exit
NXR_B(config)#ip route 0.0.0.0/0 10.10.20.254
NXR_B(config)#ipsec access-list LAN_A ip 192.168.20.0/24 192.168.10.0/24
NXR_B(config)#ipsec generate rsa-sig-key 1024
RSA-SIG KEY generating...
NXR_B(config)#exit
NXR_B#show ipsec rsa-pub-key
RSA public key :
0sAQOZe2V6nfz4pY9P/I5XONiGgTDjY6yUZ+cPSInp9dAZqe9QLQwtDitiHZMUo2Liz2/8Nlvq78+Vz7/rdNhoKAPD
07cqndlbPR1EnmaLfyNRC2Je19CJyHjCCz0vOL5qOb+eFKbAK3icFzi1ryr3tRCA2VIox57Wn2W7KkD96j5urQ==
NXR_B#configure terminal
NXR_B(config)#ipsec local policy 1
NXR_B(config-ipsec-local)#address ip
NXR_B(config-ipsec-local)#self-identity fqdn nxrb
NXR_B(config-ipsec-local)#exit
NXR_B(config)#ipsec isakmp policy 1
NXR_B(config-ipsec-isakmp)#description NXR_A
NXR_B(config-ipsec-isakmp)#authentication rsa-sig 0sAQNyjiS2aqmmPHvKp6GvDIVG6eC6yclJxWRk+s
yfUozTqPW70R3TcFP74gjNZp3p16GN3SET2/M9+qVQIySERsh3rBrzEuwzJQ/ShSv7XwJw7Awb2hlsZ8NvFKkIQ9AE
GPOF223KT3T807QjxuX5wNToUWqJKZgURAoWlpY9ufM2qw==
NXR_B(config-ipsec-isakmp)#hash sha1
NXR_B(config-ipsec-isakmp)#encryption aes128
NXR_B(config-ipsec-isakmp)#group 5
NXR_B(config-ipsec-isakmp)#lifetime 10800
NXR_B(config-ipsec-isakmp)#isakmp-mode main
NXR_B(config-ipsec-isakmp)#remote address ip 10.10.10.1
NXR_B(config-ipsec-isakmp)#remote identity fqdn nxra
NXR_B(config-ipsec-isakmp)#keepalive 30 3 periodic restart
NXR_B(config-ipsec-isakmp)#local policy 1
NXR_B(config-ipsec-isakmp)#exit
NXR_B(config)#ipsec tunnel policy 1
NXR_B(config-ipsec-tunnel)#description NXR_A
NXR_B(config-ipsec-tunnel)#negotiation-mode auto
NXR_B(config-ipsec-tunnel)#set transform esp-aes128 esp-sha1-hmac
NXR_B(config-ipsec-tunnel)#set pfs group5
NXR_B(config-ipsec-tunnel)#set sa lifetime 3600
NXR_B(config-ipsec-tunnel)#set key-exchange isakmp 1
NXR_B(config-ipsec-tunnel)#match address LAN_A
NXR_B(config-ipsec-tunnel)#exit
NXR_B(config)#interface ethernet 1
NXR_B(config-if)#ip address 10.10.20.1/24
NXR_B(config-if)#ipsec policy 1
NXR_B(config-if)#exit
NXR_B(config)#exit
NXR_B#save config
```

【 解説 】

- フェーズ１ではメインモード，フェーズ２ではクイックモードを利用します。
- ISAKMP ポリシー(フェーズ１)，tunnel ポリシー(フェーズ２)で利用するプロポーザルは「NXR_A」と同一です。
- 公開鍵は作成後、対向機器の設定で使用します。
- RSA 公開鍵暗号方式を利用する場合は、ID 設定は必須になります。

＜ホスト名の設定＞

```
nxr130(config)#hostname NXR_B
```

ホスト名として「NXR_B」を設定します。

＜Ethernet0 インタフェース設定＞

```
NXR_B(config)#interface ethernet 0
NXR_B(config-if)#ip address 192.168.20.1/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.20.1/24」を設定します。

＜スタティックルート設定＞

```
NXR_B(config)#ip route 0.0.0.0/0 10.10.20.254
```

デフォルトルートを設定します。

＜IPsec アクセスリスト設定＞

```
NXR_B(config)#ipsec access-list LAN_A ip 192.168.20.0/24 192.168.10.0/24
```

IPsec アクセスリスト名を「LAN_A」とし、送信元 IPv4 アドレス「192.168.20.0/24」, 宛先 IPv4 アドレス「192.168.10.0/24」を設定します。

またこの送信元, 宛先 IPv4 アドレスが IPsec でのカプセル化対象となります。

＜RSA Signature Key の作成＞

```
NXR_B(config)#ipsec generate rsa-sig-key 1024
```

RSA Signature Key を作成します。ここでは 1024bit で作成します。

＜RSA 公開鍵の確認＞

```
NXR_B#show ipsec rsa-pub-key
RSA public key :
0sAQOZe2V6nfz4pY9P/I5X0NiGgTDjY6yUZ+cPSInp9dAZqe9QLQwtDitiHZMUo2Liz2/8Nlvq78+Vz7/rdNhoKAPD
07cqndlbPR1EnmaLfyNRC2Je19CJyHjCCz0vOL5qOb+eFKbAK3icFzi1ryr3tRCA2VIox57Wn2W7KkD96j5urQ==
```

作成した RSA 公開鍵を確認します。ここで表示した公開鍵は対向機器の IPsec ISAKMP ポリシー設定で利用します。

＜IPsec ローカルポリシー設定＞

```
NXR_B(config)#ipsec local policy 1
NXR_B(config-ipsec-local)#address ip
```

IPsec のローカルポリシー「１」を設定します。

```
NXR_B(config-ipsec-local)#self-identity fqdn nxrb
```

機器の ID として「nxrb」を設定します（fqdn 方式で設定しています)。

なお RSA 公開鍵暗号方式を利用する場合は、ID 設定は必須になります。

＜IPsec ISAKMP ポリシー設定＞

```
NXR_B(config)#ipsec isakmp policy 1
NXR_B(config-ipsec-isakmp)#description NXR_A
NXR_B(config-ipsec-isakmp)#authentication rsa-sig 0sAQNyjiS2aqmmPHvKp6GvDIVG6eC6yclJxWRk+s
yfUozTqPW70R3TcFP74gjNZp3p16GN3SET2/M9+qVQIySERsh3rBrzEuwzJQ/ShSv7XwJw7Awb2hlsZ8NvFKkIQ9AE
GPOF223KT3T807QjxuX5wNToUWqJKZgURAoWlpY9ufM2qw==
NXR_B(config-ipsec-isakmp)#hash sha1
NXR_B(config-ipsec-isakmp)#encryption aes128
NXR_B(config-ipsec-isakmp)#group 5
NXR_B(config-ipsec-isakmp)#lifetime 10800
NXR_B(config-ipsec-isakmp)#isakmp-mode main
NXR_B(config-ipsec-isakmp)#remote address ip 10.10.10.1
NXR_B(config-ipsec-isakmp)#remote identity fqdn nxra
NXR_B(config-ipsec-isakmp)#keepalive 30 3 periodic restart
NXR_B(config-ipsec-isakmp)#local policy 1
```

IPsec の ISAKMP ポリシー「１」を設定します。

＜IPsec tunnel ポリシー設定＞

```
NXR_B(config)#ipsec tunnel policy 1
NXR_B(config-ipsec-tunnel)#description NXR_A
NXR_B(config-ipsec-tunnel)#negotiation-mode auto
NXR_B(config-ipsec-tunnel)#set transform esp-aes128 esp-sha1-hmac
NXR_B(config-ipsec-tunnel)#set pfs group5
NXR_B(config-ipsec-tunnel)#set sa lifetime 3600
NXR_B(config-ipsec-tunnel)#set key-exchange isakmp 1
NXR_B(config-ipsec-tunnel)#match address LAN_A
```

IPsec の tunnel ポリシー「１」を設定します。

＜Ethernet1 インタフェース設定＞

```
NXR_B(config)#interface ethernet 1
NXR_B(config-if)#ip address 10.10.20.1/24
NXR_B(config-if)#ipsec policy 1
```

Ethernet1 インタフェースの IPv4 アドレスとして「10.10.20.1/24」を設定します。

IPsec ローカルポリシー「１」を適用します。これによりこのインタフェースが IPsec トンネルのエンドポイントとなります。

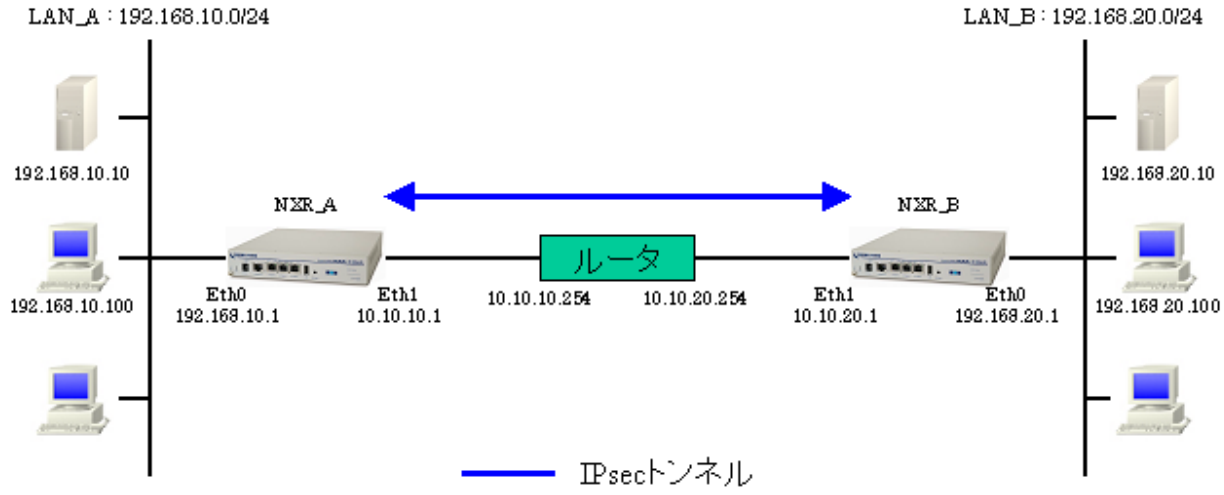
7-3-3. パソコンの設定例

	LAN A のパソコン	LAN B のパソコン
IPv4 アドレス	192.168.10.100	192.168.20.100
サブネットマスク	255.255.255.0	255.255.255.0
デフォルトゲートウェイ	192.168.10.1	192.168.20.1

7-4. X.509（デジタル署名認証）方式での接続設定例

IKE のフェーズ１で対向機器の認証に X.509 認証（デジタル署名認証）方式を利用することができます。認証で利用する証明書や鍵は、CA 等で事前に用意しておく必要があります。（NXR では証明書の発行を行うことはできません）X.509（デジタル署名認証）方式を利用する場合は IKE のフェーズ１でメインモードを使用する必要があります。

7-4-1. 構成図



7-4-2. 設定例

［NXR_A の設定］

```
nxr130#configure terminal
nxr130(config)#hostname NXR_A
NXR_A(config)#interface ethernet 0
NXR_A(config-if)#ip address 192.168.10.1/24
NXR_A(config-if)#exit
NXR_A(config)#interface ethernet 1
NXR_A(config-if)#ip address 10.10.10.1/24
NXR_A(config-if)#exit
NXR_A(config)#ip route 0.0.0.0/0 10.10.10.254
NXR_A(config)#ipsec access-list LAN_B ip 192.168.10.0/24 192.168.20.0/24
NXR_A(config)#ipsec x509 enable
NXR_A(config)#ipsec x509 ca-certificate nxr ftp://192.168.10.10/nxrCA.pem
NXR_A(config)#ipsec x509 crl nxr ftp://192.168.10.10/nxrCRL.pem
NXR_A(config)#ipsec x509 certificate nxra ftp://192.168.10.10/nxraCert.pem
NXR_A(config)#ipsec x509 private-key nxra key ftp://192.168.10.10/nxraKey.pem
NXR_A(config)#ipsec x509 private-key nxra password hidden nxrapass
NXR_A(config)#ipsec local policy 1
NXR_A(config-ipsec-local)#address ip
NXR_A(config-ipsec-local)#x509 certificate nxra
NXR_A(config-ipsec-local)#self-identity dn /C=JP/CN=nxra/E=nxra@example.com
NXR_A(config-ipsec-local)#exit
NXR_A(config)#ipsec isakmp policy 1
NXR_A(config-ipsec-isakmp)#description NXR_B

----- 次のページに続きがあります -----
```

```
----- 前のページからの続きです -----

NXR_A(config-ipsec-isakmp)#authentication rsa-sig
NXR_A(config-ipsec-isakmp)#hash sha1
NXR_A(config-ipsec-isakmp)#encryption aes128
NXR_A(config-ipsec-isakmp)#group 5
NXR_A(config-ipsec-isakmp)#lifetime 10800
NXR_A(config-ipsec-isakmp)#isakmp-mode main
NXR_A(config-ipsec-isakmp)#remote address ip 10.10.20.1
NXR_A(config-ipsec-isakmp)#remote identity dn /C=JP/CN=nxrb/E=nxrb@example.com
NXR_A(config-ipsec-isakmp)#keepalive 30 3 periodic restart
NXR_A(config-ipsec-isakmp)#local policy 1
NXR_A(config-ipsec-isakmp)#exit
NXR_A(config)#ipsec tunnel policy 1
NXR_A(config-ipsec-tunnel)#description NXR_B
NXR_A(config-ipsec-tunnel)#negotiation-mode auto
NXR_A(config-ipsec-tunnel)#set transform esp-aes128 esp-sha1-hmac
NXR_A(config-ipsec-tunnel)#set pfs group5
NXR_A(config-ipsec-tunnel)#set sa lifetime 3600
NXR_A(config-ipsec-tunnel)#set key-exchange isakmp 1
NXR_A(config-ipsec-tunnel)#match address LAN_B
NXR_A(config-ipsec-tunnel)#exit
NXR_A(config)#interface ethernet 1
NXR_A(config-if)#ipsec policy 1
NXR_A(config-if)#exit
NXR_A(config)#exit
NXR_A#save config
```

【 解説 】

- フェーズ１ではメインモード，フェーズ２ではクイックモードを利用します。
- 認証で利用する証明書や鍵は、CA 等で事前に用意しておく必要があります。(NXR では証明書の発行を行うことはできません)
- 証明書を保管しているサーバ（CA）を「192.168.10.10」とします。
  このサーバには NXR_A のルータに関係する証明書として以下の証明書が保管されています。

証明書名	ファイル名
CA 証明書	nxrCA.pem
CRL	nxrCRL.pem
NXR_A 用証明書	nxraCert.pem
NXR_A 用秘密鍵	nxraKey.pem

＜ホスト名の設定＞

```
nxr130(config)#hostname NXR_A
```

ホスト名として「NXR_A」を設定します。

＜Ethernet0 インタフェース設定＞

```
NXR_A(config)#interface ethernet 0
NXR_A(config-if)#ip address 192.168.10.1/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.1/24」を設定します。

＜Ethernet1 インタフェース設定＞

```
NXR_A(config)#interface ethernet 1
NXR_A(config-if)#ip address 10.10.10.1/24
```

Ethernet1 インタフェースの IPv4 アドレスとして「10.10.10.1/24」を設定します。

＜スタティックルート設定＞

```
NXR_A(config)#ip route 0.0.0.0/0 10.10.10.254
```

デフォルトルートを設定します。

＜IPsec アクセスリスト設定＞

```
NXR_A(config)#ipsec access-list LAN_B ip 192.168.10.0/24 192.168.20.0/24
```

IPsec アクセスリスト名を「LAN_B」とし、送信元 IPv4 アドレス「192.168.10.0/24」, 宛先 IPv4 アドレス「192.168.20.0/24」を設定します。

またこの送信元, 宛先 IPv4 アドレスが IPsec でのカプセル化対象となります。

＜X.509 の有効化＞

```
NXR_A(config)#ipsec x509 enable
```

X.509 を利用する場合は、有効にします。

＜CA 証明書の設定＞

```
NXR_A(config)#ipsec x509 ca-certificate nxr ftp://192.168.10.10/nxrCA.pem
```

FTP サーバ「192.168.10.10」にある CA 証明書ファイル「nxrCA.pem」を取得し、設定します。

＜CRL の設定＞

```
NXR_A(config)#ipsec x509 crl nxr ftp://192.168.10.10/nxrCRL.pem
```

FTP サーバ「192.168.10.10」にある CRL ファイル「nxrCRL.pem」を取得し、設定します。

＜NXR_A 用証明書の設定＞

```
NXR_A(config)#ipsec x509 certificate nxra ftp://192.168.10.10/nxraCert.pem
```

FTP サーバ「192.168.10.10」にある NXR_A 用証明書ファイル「nxraCert.pem」を取得し、設定します。

＜NXR_A 用秘密鍵の設定＞

```
NXR_A(config)#ipsec x509 private-key nxra key ftp://192.168.10.10/nxraKey.pem
```

FTP サーバ「192.168.10.10」にある NXR_A 用秘密鍵ファイル「nxraKey.pem」を取得し、設定します。

＜NXR_A 用秘密鍵パスフレーズの設定＞

```
NXR_A(config)#ipsec x509 private-key nxra password hidden nxrapass
```

NXR_A 用秘密鍵のパスフレーズとして「nxrapass」を設定し、パスフレーズを暗号化するため hidden オプションを設定しています。

＜IPsec ローカルポリシー設定＞

```
NXR_A(config)#ipsec local policy 1
NXR_A(config-ipsec-local)#address ip
```

IPsec のローカルポリシー「１」を設定し、IPsec トンネルの送信元 IP として IPv4 アドレスを設定します。

```
NXR_A(config-ipsec-local)#x509 certificate nxra
```

X.509 で利用する証明書を設定します。ここでは「NXR_A 用証明書の設定」で設定した certificate name「nxra」を設定します。

```
NXR_A(config-ipsec-local)#self-identity dn /C=JP/CN=nxra/E=nxra@example.com
```

X.509（デジタル署名）方式を利用する場合は、機器の ID は DN(Distinguished Name)方式で設定しますので、設定前に証明書の DN または subject 等をご確認下さい。
ここでは「/C=JP/CN=nxra/E=nxra@example.com」を設定します。
なお X.509（デジタル署名）方式を利用する場合は、ID 設定は必須になります。

＜IPsec ISAKMP ポリシー設定＞

```
NXR_A(config)#ipsec isakmp policy 1
NXR_A(config-ipsec-isakmp)#description NXR_B
```

IPsec の ISAKMP ポリシー「１」で、名前を「NXR_B」と設定します。

```
NXR_A(config-ipsec-isakmp)#authentication rsa-sig
```

X.509 方式を利用する場合は、「rsa-sig」を設定する必要があります。

```
NXR_A(config-ipsec-isakmp)#hash sha1
```

認証アルゴリズムとして「sha1」を設定します。

```
NXR_A(config-ipsec-isakmp)#encryption aes128
```

暗号化アルゴリズムとして「aes128」を設定します。

```
NXR_A(config-ipsec-isakmp)#group 5
```

Diffie-Hellman(DH)グループとして「group 5」を設定します。

```
NXR_A(config-ipsec-isakmp)#lifetime 10800
```

ISAKMP SA のライフタイムとして「10800」秒を設定します。

```
NXR_A(config-ipsec-isakmp)#isakmp-mode main
```

X.509 を利用するため、フェーズ１のネゴシエーションモードとして「main」を設定します。

```
NXR_A(config-ipsec-isakmp)#remote address ip 10.10.20.1
```

対向機器の IPv4 アドレス「10.10.20.1」を設定します。

```
NXR_A(config-ipsec-isakmp)#remote identity dn /C=JP/CN=nxrb/E=nxrb@example.com
```

対向機器の ID として「/C=JP/CN=nxra/E=nxra@example.com」を設定します（DN(Distinguished Name)方式で設定します)。

```
NXR_A(config-ipsec-isakmp)#keepalive 30 3 periodic restart
```

IKE KeepAlive(DPD)を設定します。監視は 30 秒間隔で 3 回リトライ（間隔およびリトライ回数はデフォルト値）を行い、keepalive 失敗時に SA を削除し、IKE の ネゴシエーションを開始します。

```
NXR_A(config-ipsec-isakmp)#local policy 1
```

使用する IPsec ローカルポリシーとして「１」(ipsec local policy 1) を設定します。

＜IPsec tunnel ポリシー設定＞

```
NXR_A(config)#ipsec tunnel policy 1
```

IPsec の tunnel ポリシー「１」を設定します。

```
NXR_A(config-ipsec-tunnel)#description NXR_B
```

tunnel ポリシー「１」の名前を「NXR_B」とします。

```
NXR_A(config-ipsec-tunnel)#negotiation-mode auto
```

ネゴシエーションモードを「auto」に設定します。これによりこちらからネゴシエーションを開始することができます。

```
NXR_A(config-ipsec-tunnel)#set transform esp-aes128 esp-sha1-hmac
```

トランスフォームとして暗号化アルゴリズム「esp-aes128」, 認証アルゴリズム「esp-sha1-hmac」を設定します。

```
NXR_A(config-ipsec-tunnel)#set pfs group5
```

PFS を有効とし、DH グループとして「group5」を設定します。

```
NXR_A(config-ipsec-tunnel)#set sa lifetime 3600
```

IPsec SA のライフタイムとして「3600」秒を設定します。

```
NXR_A(config-ipsec-tunnel)#set key-exchange isakmp 1
```

使用する IPsec ISAKMP ポリシーとして「１」(ipsec isakmp policy 1) を設定します。

```
NXR_A(config-ipsec-tunnel)#match address LAN_B
```

使用する IPsec アクセスリストとして「LAN_B」を設定します。

＜Ethernet1 インタフェース設定＞

```
NXR_A(config)#interface ethernet 1
NXR_A(config-if)#ipsec policy 1
```

IPsec ローカルポリシー「１」を適用します。これによりこのインタフェースが IPsec トンネルのエンドポイントとなります。

[NXR_B の設定]

```
nxr130#configure terminal
nxr130(config)#hostname NXR_B
NXR_B(config)#interface ethernet 0
NXR_B(config-if)#ip address 192.168.20.1/24
NXR_B(config-if)#exit
NXR_B(config)#interface ethernet 1
NXR_B(config-if)#ip address 10.10.20.1/24
NXR_B(config-if)#exit
NXR_B(config)#ip route 0.0.0.0/0 10.10.20.254
NXR_B(config)#ipsec access-list LAN_A ip 192.168.20.0/24 192.168.10.0/24
NXR_B(config)#ipsec x509 enable
NXR_B(config)#ipsec x509 ca-certificate nxr ftp://192.168.20.10/nxrCA.pem
NXR_B(config)#ipsec x509 crl nxr ftp://192.168.20.10/nxrCRL.pem
NXR_B(config)#ipsec x509 certificate nxrb ftp://192.168.20.10/nxrbCert.pem
NXR_B(config)#ipsec x509 private-key nxrb key ftp://192.168.20.10/nxrbKey.pem
NXR_B(config)#ipsec x509 private-key nxrb password hidden nxrbpass
NXR_B(config)#ipsec local policy 1
NXR_B(config-ipsec-local)#address ip
NXR_B(config-ipsec-local)#x509 certificate nxrb
NXR_B(config-ipsec-local)#self-identity dn /C=JP/CN=nxrb/E=nxrb@example.com
NXR_B(config-ipsec-local)#exit
NXR_B(config)#ipsec isakmp policy 1
NXR_B(config-ipsec-isakmp)#description NXR_A
NXR_B(config-ipsec-isakmp)#authentication rsa-sig
NXR_B(config-ipsec-isakmp)#hash sha1
NXR_B(config-ipsec-isakmp)#encryption aes128
NXR_B(config-ipsec-isakmp)#group 5
NXR_B(config-ipsec-isakmp)#lifetime 10800
NXR_B(config-ipsec-isakmp)#isakmp-mode main
NXR_B(config-ipsec-isakmp)#remote address ip 10.10.10.1
NXR_B(config-ipsec-isakmp)#remote identity dn /C=JP/CN=nxra/E=nxra@example.com
NXR_B(config-ipsec-isakmp)#keepalive 30 3 periodic restart
NXR_B(config-ipsec-isakmp)#local policy 1
NXR_B(config-ipsec-isakmp)#exit
NXR_B(config)#ipsec tunnel policy 1
NXR_B(config-ipsec-tunnel)#description NXR_A
NXR_B(config-ipsec-tunnel)#negotiation-mode auto
NXR_B(config-ipsec-tunnel)#set transform esp-aes128 esp-sha1-hmac
NXR_B(config-ipsec-tunnel)#set pfs group5
NXR_B(config-ipsec-tunnel)#set sa lifetime 3600
NXR_B(config-ipsec-tunnel)#set key-exchange isakmp 1
NXR_B(config-ipsec-tunnel)#match address LAN_A
NXR_B(config-ipsec-tunnel)#exit
NXR_B(config)#interface ethernet 1
NXR_B(config-if)#ipsec policy 1
NXR_B(config-if)#exit
NXR_B(config)#exit
NXR_B#save config
```

【 解説 】

- フェーズ１ではメインモード，フェーズ２ではクイックモードを利用します。
- 認証で利用する証明書や鍵は、CA 等で事前に用意しておく必要があります。(NXR では証明書の発行を行うことはできません)
- 証明書を保管しているサーバ（CA）を「192.168.20.10」とします。
  このサーバには NXR_B のルータに関係する証明書として以下の証明書が保管されています。

証明書名	ファイル名
CA 証明書	nxrCA.pem
CRL	nxrCRL.pem
NXR_B 用証明書	nxrbCert.pem
NXR_B 用秘密鍵	nxrbKey.pem

＜ホスト名の設定＞

```
nxr130(config)#hostname NXR_B
```

ホスト名として「NXR_B」を設定します。

＜Ethernet0 インタフェース設定＞

```
NXR_B(config)#interface ethernet 0
NXR_B(config-if)#ip address 192.168.20.1/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.20.1/24」を設定します。

＜Ethernet1 インタフェース設定＞

```
NXR_B(config)#interface ethernet 1
NXR_B(config-if)#ip address 10.10.20.1/24
```

Ethernet1 インタフェースの IPv4 アドレスとして「10.10.20.1/24」を設定します。

＜スタティックルート設定＞

```
NXR_B(config)#ip route 0.0.0.0/0 10.10.20.254
```

デフォルトルートを設定します。

＜IPsec アクセスリスト設定＞

```
NXR_B(config)#ipsec access-list LAN_A ip 192.168.20.0/24 192.168.10.0/24
```

IPsec アクセスリスト名を「LAN_A」とし、送信元 IPv4 アドレス「192.168.20.0/24」, 宛先 IPv4 アドレス「192.168.10.0/24」を設定します。

またこの送信元, 宛先 IPv4 アドレスが IPsec でのカプセル化対象となります。

＜X.509 の有効化＞

```
NXR_B(config)#ipsec x509 enable
```

X.509 を利用する場合は、有効にします。

＜CA 証明書の設定＞

```
NXR_B(config)#ipsec x509 ca-certificate nxr ftp://192.168.20.10/nxrCA.pem
```

FTP サーバ「192.168.20.10」にある CA 証明書ファイル「nxrCA.pem」を取得し、設定します。

＜CRL の設定＞

```
NXR_B(config)#ipsec x509 crl nxr ftp://192.168.20.10/nxrCRL.pem
```

FTP サーバ「192.168.20.10」にある CRL ファイル「nxrCRL.pem」を取得し、設定します。

＜NXR_B 用証明書の設定＞

```
NXR_B(config)#ipsec x509 certificate nxrb ftp://192.168.20.10/nxrbCert.pem
```

FTP サーバ「192.168.20.10」にある NXR_B 用証明書ファイル「nxrbCert.pem」を取得し、設定します。

＜NXR_B 用秘密鍵の設定＞

```
NXR_B(config)#ipsec x509 private-key nxrb key ftp://192.168.20.10/nxrbKey.pem
```

FTP サーバ「192.168.20.10」にある NXR_B 用秘密鍵ファイル「nxrbKey.pem」を取得し、設定します。

＜NXR_B 用秘密鍵パスフレーズの設定＞

```
NXR_B(config)#ipsec x509 private-key nxrb password hidden nxrbpass
```

NXR_B 用秘密鍵のパスフレーズとして「nxrbpass」を設定し、パスフレーズを暗号化するため hidden オプションを設定しています。

＜IPsec ローカルポリシー設定＞

```
NXR_B(config)#ipsec local policy 1
NXR_B(config-ipsec-local)#address ip
```

IPsec のローカルポリシー「１」を設定し、IPsec トンネルの送信元 IP として IPv4 アドレスを設定します。

```
NXR_B(config-ipsec-local)#x509 certificate nxrb
```

X.509 で利用する証明書を設定します。ここでは「NXR_B 用証明書の設定」で設定した certificate name 「nxrb」を設定します。

```
NXR_B(config-ipsec-local)#self-identity dn /C=JP/CN=nxrb/E=nxrb@example.com
```

X.509（デジタル署名）方式を利用する場合は、機器の ID は DN(Distinguished Name)方式で設定しますので、設定前に証明書の DN または subject 等をご確認下さい。
ここでは「/C=JP/CN=nxrb/E=nxrb@example.com」を設定します
なお X.509（デジタル署名）方式を利用する場合は、ID 設定は必須になります。

＜IPsec ISAKMP ポリシー設定＞

```
NXR_B(config)#ipsec isakmp policy 1
NXR_B(config-ipsec-isakmp)#description NXR_A
```

IPsec の ISAKMP ポリシー「１」で、名前を「NXR_A」と設定します。

```
NXR_B(config-ipsec-isakmp)#authentication rsa-sig
```

X.509 方式を利用する場合は、「rsa-sig」を設定する必要があります。

```
NXR_B(config-ipsec-isakmp)#hash sha1
NXR_B(config-ipsec-isakmp)#encryption aes128
NXR_B(config-ipsec-isakmp)#group 5
NXR_B(config-ipsec-isakmp)#lifetime 10800
NXR_B(config-ipsec-isakmp)#isakmp-mode main
NXR_B(config-ipsec-isakmp)#remote address ip 10.10.10.1
```

対向機器の IPv4 アドレスやフェーズ 1 のネゴシエーションモード,認証アルゴリズム等の各プロポーザルを設定します。

```
NXR_B(config-ipsec-isakmp)#remote identity dn /C=JP/CN=nxra/E=nxra@example.com
```

対向機器の ID として「/C=JP/CN=nxra/E=nxra@example.com」を設定します (DN(Distinguished Name)方式で設定します)。

```
NXR_B(config-ipsec-isakmp)#keepalive 30 3 periodic restart
```

IKE KeepAlive(DPD)を設定します。監視は 30 秒間隔で 3 回リトライ (間隔およびリトライ回数はデフォルト値) を行い、keepalive 失敗時に SA を削除し、IKE の ネゴシエーションを開始します。

```
NXR_B(config-ipsec-isakmp)#local policy 1
```

使用する IPsec ローカルポリシーとして「1」(ipsec local policy 1) を設定します。

＜IPsec tunnel ポリシー設定＞

```
NXR_B(config)#ipsec tunnel policy 1
NXR_B(config-ipsec-tunnel)#description NXR_A
NXR_B(config-ipsec-tunnel)#negotiation-mode auto
NXR_B(config-ipsec-tunnel)#set transform esp-aes128 esp-sha1-hmac
NXR_B(config-ipsec-tunnel)#set pfs group5
NXR_B(config-ipsec-tunnel)#set sa lifetime 3600
NXR_B(config-ipsec-tunnel)#set key-exchange isakmp 1
NXR_B(config-ipsec-tunnel)#match address LAN_A
```

IPsec の tunnel ポリシー「1」を設定します。

＜Ethernet1 インタフェース設定＞

```
NXR_A(config)#interface ethernet 1
NXR_A(config-if)#ipsec policy 1
```

IPsec ローカルポリシー「1」を適用します。これによりこのインタフェースが IPsec トンネルのエンドポイントとなります。

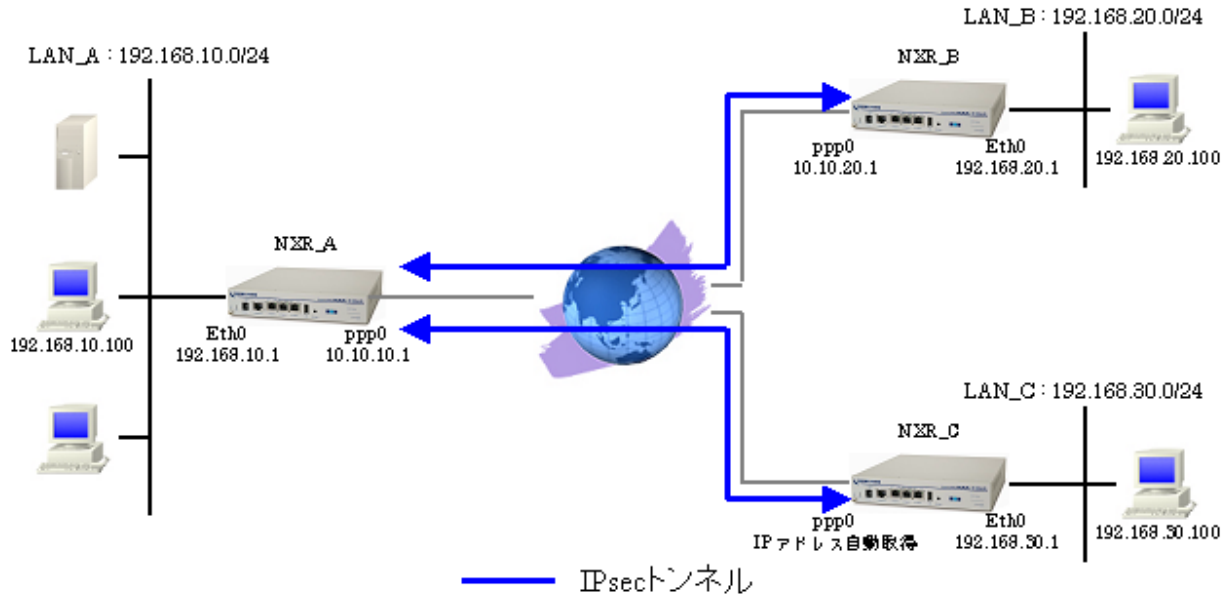
7-4-3. パソコンの設定例

	LAN A のパソコン	LAN B のパソコン
IPv4 アドレス	192.168.10.100	192.168.20.100
サブネットマスク	255.255.255.0	255.255.255.0
デフォルトゲートウェイ	192.168.10.1	192.168.20.1

7-5. PPPoE を利用した IPsec 接続設定例

PPPoE 上でも IPsec を利用することは可能です。ここではフェーズ１で NXR_A（センタ） － NXR_B（拠点）間はメインモードを NXR_A（センタ） － NXR_C（拠点）間はアグレッシブモードを利用して接続しています。なおここでは拠点間の IPsec 経由での通信は行いません。
またここでは、各拠点からのインターネットアクセスを可能にするために、フィルタ設定（SPI），NAT 設定（IP マスカレード），DNS 設定を行っています。

7-5-1. 構成図



7-5-2. 設定例

[NXR_A の設定]

```
nxr130#configure terminal
nxr130(config)#hostname NXR_A
NXR_A(config)#interface ethernet 0
NXR_A(config-if)#ip address 192.168.10.1/24
NXR_A(config-if)#exit
NXR_A(config)#ip access-list ppp0_in permit any 10.10.10.1 udp 500 500
NXR_A(config)#ip access-list ppp0_in permit any 10.10.10.1 50
NXR_A(config)#ipsec access-list LAN_B ip 192.168.10.0/24 192.168.20.0/24
NXR_A(config)#ipsec access-list LAN_C ip 192.168.10.0/24 192.168.30.0/24
NXR_A(config)#interface ppp 0
NXR_A(config-ppp)#ip address 10.10.10.1/32
NXR_A(config-ppp)#ip masquerade
NXR_A(config-ppp)#ip access-group in ppp0_in
NXR_A(config-ppp)#ip spi-filter
NXR_A(config-ppp)#ip tcp adjust-mss auto
NXR_A(config-ppp)#no ip redirects
NXR_A(config-ppp)#ppp authentication pap
NXR_A(config-ppp)#ppp username test1@centurysys password test1pass
NXR_A(config-ppp)#exit
NXR_A(config)#interface ethernet 1
NXR_A(config-if)#no ip address
NXR_A(config-if)#pppoe-client ppp 0

----- 次のページに続きがあります -----
```

```
----- 前のページからの続きです -----

NXR_A(config-if)#exit
NXR_A(config)#ip route 0.0.0.0/0 ppp 0
NXR_A(config)#ipsec local policy 1
NXR_A(config-ipsec-local)#address ip
NXR_A(config-ipsec-local)#exit
NXR_A(config)#ipsec isakmp policy 1
NXR_A(config-ipsec-isakmp)#description NXR_B
NXR_A(config-ipsec-isakmp)#authentication pre-share ipseckey1
NXR_A(config-ipsec-isakmp)#hash sha1
NXR_A(config-ipsec-isakmp)#encryption aes128
NXR_A(config-ipsec-isakmp)#group 5
NXR_A(config-ipsec-isakmp)#lifetime 10800
NXR_A(config-ipsec-isakmp)#isakmp-mode main
NXR_A(config-ipsec-isakmp)#remote address ip 10.10.20.1
NXR_A(config-ipsec-isakmp)#keepalive 30 3 periodic restart
NXR_A(config-ipsec-isakmp)#local policy 1
NXR_A(config-ipsec-isakmp)#exit
NXR_A(config)#ipsec tunnel policy 1
NXR_A(config-ipsec-tunnel)#description NXR_B
NXR_A(config-ipsec-tunnel)#negotiation-mode auto
NXR_A(config-ipsec-tunnel)#set transform esp-aes128 esp-sha1-hmac
NXR_A(config-ipsec-tunnel)#set pfs group5
NXR_A(config-ipsec-tunnel)#set sa lifetime 3600
NXR_A(config-ipsec-tunnel)#set key-exchange isakmp 1
NXR_A(config-ipsec-tunnel)#match address LAN_B
NXR_A(config-ipsec-tunnel)#exit
NXR_A(config)#ipsec isakmp policy 2
NXR_A(config-ipsec-isakmp)#description NXR_C
NXR_A(config-ipsec-isakmp)#authentication pre-share ipseckey2
NXR_A(config-ipsec-isakmp)#hash sha1
NXR_A(config-ipsec-isakmp)#encryption aes128
NXR_A(config-ipsec-isakmp)#group 5
NXR_A(config-ipsec-isakmp)#lifetime 10800
NXR_A(config-ipsec-isakmp)#isakmp-mode aggressive
NXR_A(config-ipsec-isakmp)#remote address ip any
NXR_A(config-ipsec-isakmp)#remote identity fqdn nxrc
NXR_A(config-ipsec-isakmp)#keepalive 30 3 periodic clear
NXR_A(config-ipsec-isakmp)#local policy 1
NXR_A(config-ipsec-isakmp)#exit
NXR_A(config)#ipsec tunnel policy 2
NXR_A(config-ipsec-tunnel)#description NXR_C
NXR_A(config-ipsec-tunnel)#negotiation-mode manual
NXR_A(config-ipsec-tunnel)#set transform esp-aes128 esp-sha1-hmac
NXR_A(config-ipsec-tunnel)#set pfs group5
NXR_A(config-ipsec-tunnel)#set sa lifetime 3600
NXR_A(config-ipsec-tunnel)#set key-exchange isakmp 2
NXR_A(config-ipsec-tunnel)#match address LAN_C
NXR_A(config-ipsec-tunnel)#exit
NXR_A(config)#interface ppp 0
NXR_A(config-ppp)#ipsec policy 1
NXR_A(config-ppp)#exit
NXR_A(config)#dns
NXR_A(dns-config)#service enable
NXR_A(dns-config)#exit
NXR_A(config)#exit
NXR_A#save config
```

【 解説 】

- フェーズ１では NXR_B 向けにはメインモード，NXR_C 向けにはアグレッシブモードをフェーズ２ではクイックモードを利用します。
- 事前共有鍵は NXR_B 向けには「ipseckey1」，NXR_C 向けには「ipseckey2」を使用します。
- IPsec のネゴシエーションパケットおよび通信用（ESP）のパケットを受信可能にするために以下のアクセスリストを作成し、ppp0 インタフェースの IN フィルタに登録します。

動作	送信元 IPv4 アドレス	宛先 IPv4 アドレス	プロトコル	送信元ポート	宛先ポート
許可	any	10.10.10.1	UDP	500	500
許可	any	10.10.10.1	50(ESP)	-	-

＜ホスト名の設定＞

```
nxr130(config)#hostname NXR_A
```

ホスト名として「NXR_A」を設定します。

＜Ethernet0 インタフェース設定＞

```
NXR_A(config)#interface ethernet 0
NXR_A(config-if)#ip address 192.168.10.1/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.1/24」を設定します。

＜IPv4 アクセスリスト設定＞

```
NXR_A(config)#ip access-list ppp0_in permit any 10.10.10.1 udp 500 500
NXR_A(config)#ip access-list ppp0_in permit any 10.10.10.1 50
```

IPv4 アクセスリスト名を「ppp0_in」とし、送信元 UDP ポート番号「500」，宛先 IPv4 アドレス「10.10.10.1」，UDP ポート番号「500」のパケットおよび宛先 IPv4 アドレス「10.10.10.1」，プロトコル番号 50（ESP）のパケットは許可します。

＜IPsec アクセスリスト設定＞

```
NXR_A(config)#ipsec access-list LAN_B ip 192.168.10.0/24 192.168.20.0/24
NXR_A(config)#ipsec access-list LAN_C ip 192.168.10.0/24 192.168.30.0/24
```

IPsec アクセスリスト名を「LAN_B」とし、送信元 IPv4 アドレス「192.168.10.0/24」，宛先 IPv4 アドレス「192.168.20.0/24」を設定します。

IPsec アクセスリスト名を「LAN_C」とし、送信元 IPv4 アドレス「192.168.10.0/24」，宛先 IPv4 アドレス「192.168.30.0/24」を設定します。

またこの送信元，宛先 IPv4 アドレスが IPsec でのカプセル化対象となります。

＜ppp0 インタフェース設定＞

```
NXR_A(config)#interface ppp 0
NXR_A(config-ppp)#ip address 10.10.10.1/32
NXR_A(config-ppp)#ip masquerade
NXR_A(config-ppp)#ip access-group in ppp0_in
NXR_A(config-ppp)#ip spi-filter
NXR_A(config-ppp)#ip tcp adjust-mss auto
NXR_A(config-ppp)#no ip redirects
NXR_A(config-ppp)#ppp authentication pap
NXR_A(config-ppp)#ppp username test1@centurysys password test1pass
```

ppp0 インタフェースに関する設定をします。

IPv4 アドレスが固定のため、「10.10.10.1/32」を設定します。

IPv4 アクセスリスト設定で設定した「ppp0-in」を ppp0 インタフェースの「in」フィルタに適用します。

＜Ethernet1 インタフェース設定＞

```
NXR_A(config)#interface ethernet 1
NXR_A(config-if)#no ip address
NXR_A(config-if)#pppoe-client ppp 0
```

Ethernet1 インタフェースに関する設定をします。

＜スタティックルート設定＞

```
NXR_A(config)#ip route 0.0.0.0/0 ppp 0
```

デフォルトルートを設定します。

＜IPsec ローカルポリシー設定＞

```
NXR_A(config)#ipsec local policy 1
NXR_A(config-ipsec-local)#address ip
```

IPsec のローカルポリシー「１」を設定します。

＜IPsec ISAKMP ポリシー設定１＞

```
NXR_A(config)#ipsec isakmp policy 1
NXR_A(config-ipsec-isakmp)#description NXR_B
NXR_A(config-ipsec-isakmp)#authentication pre-share ipseckey1
NXR_A(config-ipsec-isakmp)#hash sha1
NXR_A(config-ipsec-isakmp)#encryption aes128
NXR_A(config-ipsec-isakmp)#group 5
NXR_A(config-ipsec-isakmp)#lifetime 10800
NXR_A(config-ipsec-isakmp)#isakmp-mode main
NXR_A(config-ipsec-isakmp)#remote address ip 10.10.20.1
NXR_A(config-ipsec-isakmp)#keepalive 30 3 periodic restart
NXR_A(config-ipsec-isakmp)#local policy 1
```

IPsec の ISAKMP ポリシー「１」を設定します。

事前共有鍵(Pre-Shared Key)として「ipseckey1」を設定します。

フェーズ１のネゴシエーションモードとして「main」を設定します。

＜IPsec tunnel ポリシー設定１＞

```
NXR_A(config)#ipsec tunnel policy 1
NXR_A(config-ipsec-tunnel)#description NXR_B
NXR_A(config-ipsec-tunnel)#negotiation-mode auto
NXR_A(config-ipsec-tunnel)#set transform esp-aes128 esp-sha1-hmac
NXR_A(config-ipsec-tunnel)#set pfs group5
NXR_A(config-ipsec-tunnel)#set sa lifetime 3600
NXR_A(config-ipsec-tunnel)#set key-exchange isakmp 1
NXR_A(config-ipsec-tunnel)#match address LAN_B
```

IPsec の tunnel ポリシー「１」を設定します。

使用する IPsec アクセスリストとして「LAN_B」を設定します。

＜IPsec ISAKMP ポリシー設定２＞

```
NXR_A(config)#ipsec isakmp policy 2
NXR_A(config-ipsec-isakmp)#description NXR_C
NXR_A(config-ipsec-isakmp)#authentication pre-share ipseckey2
NXR_A(config-ipsec-isakmp)#hash sha1
NXR_A(config-ipsec-isakmp)#encryption aes128
NXR_A(config-ipsec-isakmp)#group 5
NXR_A(config-ipsec-isakmp)#lifetime 10800
NXR_A(config-ipsec-isakmp)#isakmp-mode aggressive
NXR_A(config-ipsec-isakmp)#remote address ip any
NXR_A(config-ipsec-isakmp)#remote identity fqdn nxrc
NXR_A(config-ipsec-isakmp)#keepalive 30 3 periodic clear
NXR_A(config-ipsec-isakmp)#local policy 1
```

IPsec の ISAKMP ポリシー「２」を設定します。

事前共有鍵(Pre-Shared Key)として「ipseckey2」を設定します。

フェーズ１のネゴシエーションモードとして「aggressive」を設定します。

＜IPsec tunnel ポリシー設定２＞

```
NXR_A(config)#ipsec tunnel policy 2
NXR_A(config-ipsec-tunnel)#description NXR_C
NXR_A(config-ipsec-tunnel)#negotiation-mode manual
NXR_A(config-ipsec-tunnel)#set transform esp-aes128 esp-sha1-hmac
NXR_A(config-ipsec-tunnel)#set pfs group5
NXR_A(config-ipsec-tunnel)#set sa lifetime 3600
NXR_A(config-ipsec-tunnel)#set key-exchange isakmp 2
NXR_A(config-ipsec-tunnel)#match address LAN_C
```

IPsec の tunnel ポリシー「２」を設定します。

使用する IPsec アクセスリストとして「LAN_C」を設定します。

＜ppp0 インタフェース設定＞

```
NXR_A(config)#interface ppp 0
NXR_A(config-ppp)#ipsec policy 1
```

IPsec ローカルポリシー「１」を適用します。これによりこのインタフェースが IPsec トンネルのエンドポイントとなります。

＜DNS 設定＞

```
NXR_A(config)#dns
NXR_A(dns-config)#service enable
```

DNS に関する設定をします。

[NXR_B の設定]

```
nxr130#configure terminal
nxr130(config)#hostname NXR_B
NXR_B(config)#interface ethernet 0
NXR_B(config-if)#ip address 192.168.20.1/24
NXR_B(config-if)#exit
NXR_B(config)#ip access-list ppp0_in permit 10.10.10.1 10.10.20.1 udp 500 500
NXR_B(config)#ip access-list ppp0_in permit 10.10.10.1 10.10.20.1 50
NXR_B(config)#ipsec access-list LAN_A ip 192.168.20.0/24 192.168.10.0/24
NXR_B(config)#interface ppp 0
NXR_B(config-ppp)#ip address 10.10.20.1/32
NXR_B(config-ppp)#ip masquerade
NXR_B(config-ppp)#ip access-group in ppp0_in
NXR_B(config-ppp)#ip spi-filter
NXR_B(config-ppp)#ip tcp adjust-mss auto
NXR_B(config-ppp)#no ip redirects
NXR_B(config-ppp)#ppp authentication pap
NXR_B(config-ppp)#ppp username test2@centurysys password test2pass
NXR_B(config-ppp)#exit
NXR_B(config)#interface ethernet 1
NXR_B(config-if)#no ip address
NXR_B(config-if)#pppoe-client ppp 0
NXR_B(config-if)#exit
NXR_B(config)#ip route 0.0.0.0/0 ppp 0
NXR_B(config)#ipsec local policy 1
NXR_B(config-ipsec-local)#address ip
NXR_B(config-ipsec-local)#exit
NXR_B(config)#ipsec isakmp policy 1
NXR_B(config-ipsec-isakmp)#description NXR_A
NXR_B(config-ipsec-isakmp)#authentication pre-share ipseckey1
NXR_B(config-ipsec-isakmp)#hash sha1
NXR_B(config-ipsec-isakmp)#encryption aes128
NXR_B(config-ipsec-isakmp)#group 5
NXR_B(config-ipsec-isakmp)#lifetime 10800
NXR_B(config-ipsec-isakmp)#isakmp-mode main
NXR_B(config-ipsec-isakmp)#remote address ip 10.10.10.1
NXR_B(config-ipsec-isakmp)#keepalive 30 3 periodic restart
NXR_B(config-ipsec-isakmp)#local policy 1
NXR_B(config-ipsec-isakmp)#exit
NXR_B(config)#ipsec tunnel policy 1
NXR_B(config-ipsec-tunnel)#description NXR_A
NXR_B(config-ipsec-tunnel)#negotiation-mode auto
NXR_B(config-ipsec-tunnel)#set transform esp-aes128 esp-sha1-hmac
NXR_B(config-ipsec-tunnel)#set pfs group5
NXR_B(config-ipsec-tunnel)#set sa lifetime 3600
NXR_B(config-ipsec-tunnel)#set key-exchange isakmp 1
NXR_B(config-ipsec-tunnel)#match address LAN_A
NXR_B(config-ipsec-tunnel)#exit
NXR_B(config)#interface ppp 0
NXR_B(config-ppp)#ipsec policy 1
NXR_B(config-ppp)#exit
NXR_B(config)#dns
NXR_B(dns-config)#service enable
NXR_B(dns-config)#exit
NXR_B(config)#exit
NXR_B#save config
```

【 解説 】

- フェーズ１ではメインモード，フェーズ２ではクイックモードを利用します。
- ISAKMP ポリシー(フェーズ１)，tunnel ポリシー(フェーズ２)で利用するプロポーザルは「NXR_A」

と同一です。

・ 事前共有鍵は対向機器と同一のもの（ここでは ipseckey1）を使用します。

＜ホスト名の設定＞

```
nxr130(config)#hostname NXR_B
```

ホスト名として「NXR_B」を設定します。

＜Ethernet0 インタフェース設定＞

```
NXR_B(config)#interface ethernet 0
NXR_B(config-if)#ip address 192.168.20.1/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.20.1/24」を設定します。

＜IPv4 アクセスリスト設定＞

```
NXR_B(config)#ip access-list ppp0_in permit 10.10.10.1 10.10.20.1 udp 500 500
NXR_B(config)#ip access-list ppp0_in permit 10.10.10.1 10.10.20.1 50
```

IPv4 アクセスリスト名を「ppp0_in」とし、以下のルールで設定します。

動作	送信元 IPv4 アドレス	宛先 IPv4 アドレス	プロトコル	送信元ポート	宛先ポート
許可	10.10.10.1	10.10.20.1	UDP	500	500
許可	10.10.10.1	10.10.20.1	50(ESP)	-	-

＜IPsec アクセスリスト設定＞

```
NXR_B(config)#ipsec access-list LAN_A ip 192.168.20.0/24 192.168.10.0/24
```

IPsec アクセスリスト名を「LAN_A」とし、送信元 IPv4 アドレス「192.168.20.0/24」, 宛先 IPv4 アドレス「192.168.10.0/24」を設定します。

またこの送信元, 宛先 IPv4 アドレスが IPsec でのカプセル化対象となります。

＜ppp0 インタフェース設定＞

```
NXR_B(config)#interface ppp 0
NXR_B(config-ppp)#ip address 10.10.20.1/32
NXR_B(config-ppp)#ip masquerade
NXR_B(config-ppp)#ip access-group in ppp0_in
NXR_B(config-ppp)#ip spi-filter
NXR_B(config-ppp)#ip tcp adjust-mss auto
NXR_B(config-ppp)#no ip redirects
NXR_B(config-ppp)#ppp authentication pap
NXR_B(config-ppp)#ppp username test2@centurysys password test2pass
```

ppp0 インタフェースに関する設定をします。

IPv4 アドレスが固定のため、「10.10.20.1/32」を設定します。

IPv4 アクセスリスト設定で設定した「ppp0-in」を ppp0 インタフェースの「in」フィルタに適用します。

＜Ethernet1 インタフェース設定＞

```
NXR_B(config)#interface ethernet 1
NXR_B(config-if)#no ip address
NXR_B(config-if)#pppoe-client ppp 0
```

Ethernet1 インタフェースに関する設定をします。

＜スタティックルート設定＞

```
NXR_B(config)#ip route 0.0.0.0/0 ppp 0
```

デフォルトルートを設定します。

＜IPsec ローカルポリシー設定＞

```
NXR_B(config)#ipsec local policy 1
NXR_B(config-ipsec-local)#address ip
```

IPsec のローカルポリシー「１」を設定します。

＜IPsec ISAKMP ポリシー設定１＞

```
NXR_B(config)#ipsec isakmp policy 1
NXR_B(config-ipsec-isakmp)#description NXR_A
NXR_B(config-ipsec-isakmp)#authentication pre-share ipseckey1
NXR_B(config-ipsec-isakmp)#hash sha1
NXR_B(config-ipsec-isakmp)#encryption aes128
NXR_B(config-ipsec-isakmp)#group 5
NXR_B(config-ipsec-isakmp)#lifetime 10800
NXR_B(config-ipsec-isakmp)#isakmp-mode main
NXR_B(config-ipsec-isakmp)#remote address ip 10.10.10.1
NXR_B(config-ipsec-isakmp)#keepalive 30 3 periodic restart
NXR_B(config-ipsec-isakmp)#local policy 1
```

IPsec の ISAKMP ポリシー「１」を設定します。

事前共有鍵(Pre-Shared Key)として「ipseckey1」を設定します。

フェーズ１のネゴシエーションモードとして「main」を設定します。

＜IPsec tunnel ポリシー設定１＞

```
NXR_B(config)#ipsec tunnel policy 1
NXR_B(config-ipsec-tunnel)#description NXR_A
NXR_B(config-ipsec-tunnel)#negotiation-mode auto
NXR_B(config-ipsec-tunnel)#set transform esp-aes128 esp-sha1-hmac
NXR_B(config-ipsec-tunnel)#set pfs group5
NXR_B(config-ipsec-tunnel)#set sa lifetime 3600
NXR_B(config-ipsec-tunnel)#set key-exchange isakmp 1
NXR_B(config-ipsec-tunnel)#match address LAN_A
```

IPsec の tunnel ポリシー「１」を設定します。

使用する IPsec アクセスリストとして「LAN_A」を設定します。

＜ppp0 インタフェース設定＞

```
NXR_B(config)#interface ppp 0
NXR_B(config-ppp)#ipsec policy 1
```

IPsec ローカルポリシー「１」を適用します。これによりこのインタフェースが IPsec トンネルのエンドポイントとなります。

＜DNS 設定＞

```
NXR_B(config)#dns
NXR_B(dns-config)#service enable
```

DNS に関する設定をします。

[NXR_C の設定]

```
nxr130#configure terminal
nxr130(config)#hostname NXR_C
NXR_C(config)#interface ethernet 0
NXR_C(config-if)#ip address 192.168.30.1/24
NXR_C(config-if)#exit
NXR_C(config)#ip access-list ppp0_in permit 10.10.10.1 any udp 500 500
NXR_C(config)#ip access-list ppp0_in permit 10.10.10.1 any 50
NXR_C(config)#ipsec access-list LAN_A ip 192.168.30.0/24 192.168.10.0/24
NXR_C(config)#interface ppp 0
NXR_C(config-ppp)#ip address negotiated
NXR_C(config-ppp)#ip masquerade
NXR_C(config-ppp)#ip access-group in ppp0_in
NXR_C(config-ppp)#ip spi-filter
NXR_C(config-ppp)#ip tcp adjust-mss auto
NXR_C(config-ppp)#no ip redirects
NXR_C(config-ppp)#ppp authentication pap
NXR_C(config-ppp)#ppp username test3@centurysys password test3pass
NXR_C(config-ppp)#exit
NXR_C(config)#interface ethernet 1
NXR_C(config-if)#no ip address
NXR_C(config-if)#pppoe-client ppp 0
NXR_C(config-if)#exit
NXR_C(config)#ip route 0.0.0.0/0 ppp 0
NXR_C(config)#ipsec local policy 1
NXR_C(config-ipsec-local)#address ip
NXR_C(config-ipsec-local)#self-identity fqdn nxrc
NXR_C(config-ipsec-local)#exit
NXR_C(config)#ipsec isakmp policy 1
NXR_C(config-ipsec-isakmp)#description NXR_A
NXR_C(config-ipsec-isakmp)#authentication pre-share ipseckey2
NXR_C(config-ipsec-isakmp)#hash sha1
NXR_C(config-ipsec-isakmp)#encryption aes128
NXR_C(config-ipsec-isakmp)#group 5
NXR_C(config-ipsec-isakmp)#lifetime 10800
NXR_C(config-ipsec-isakmp)#isakmp-mode aggressive
NXR_C(config-ipsec-isakmp)#remote address ip 10.10.10.1
NXR_C(config-ipsec-isakmp)#keepalive 30 3 periodic restart
NXR_C(config-ipsec-isakmp)#local policy 1
NXR_C(config-ipsec-isakmp)#exit
NXR_C(config)#ipsec tunnel policy 1
NXR_C(config-ipsec-tunnel)#description NXR_A
NXR_C(config-ipsec-tunnel)#negotiation-mode auto
NXR_C(config-ipsec-tunnel)#set transform esp-aes128 esp-sha1-hmac
NXR_C(config-ipsec-tunnel)#set pfs group5
NXR_C(config-ipsec-tunnel)#set sa lifetime 3600
NXR_C(config-ipsec-tunnel)#set key-exchange isakmp 1
NXR_C(config-ipsec-tunnel)#match address LAN_A
NXR_C(config-ipsec-tunnel)#exit
NXR_C(config)#interface ppp 0
NXR_C(config-ppp)#ipsec policy 1
NXR_C(config-ppp)#exit
NXR_C(config)#dns
NXR_C(dns-config)#service enable
NXR_C(dns-config)#exit
NXR_C(config)#exit
NXR_C#save config
```

【 解説 】

・ フェーズ１ではアグレッシブモード，フェーズ２ではクイックモードを利用します。

- ISAKMP ポリシー(フェーズ１), tunnel ポリシー(フェーズ２)で利用するプロポーザルは「NXR_A」と同一です。
- 機器の IP アドレスを ID として利用することができない（動的 IPv4 アドレス）ため、ID（ID Type ： FQDN）を設定します。
- 事前共有鍵は対向機器と同一のもの（ここでは ipseckey2）を使用します。

＜ホスト名の設定＞

```
nxr130(config)#hostname NXR_C
```

ホスト名として「NXR_C」を設定します。

＜Ethernet0 インタフェース設定＞

```
NXR_C(config)#interface ethernet 0
NXR_C(config-if)#ip address 192.168.30.1/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.30.1/24」を設定します。

＜IPv4 アクセスリスト設定＞

```
NXR_C(config)#ip access-list ppp0_in permit 10.10.10.1 any udp 500 500
NXR_C(config)#ip access-list ppp0_in permit 10.10.10.1 any 50
```

IPv4 アクセスリスト名を「ppp0_in」とし、以下のルールで設定します。

動作	送信元 IPv4 アドレス	宛先 IPv4 アドレス	プロトコル	送信元ポート	宛先ポート
許可	10.10.10.1	any	UDP	500	500
許可	10.10.10.1	any	50(ESP)	-	-

＜IPsec アクセスリスト設定＞

```
NXR_C(config)#ipsec access-list LAN_A ip 192.168.30.0/24 192.168.10.0/24
```

IPsec アクセスリスト名を「LAN_A」とし、送信元 IPv4 アドレス「192.168.30.0/24」, 宛先 IPv4 アドレス「192.168.10.0/24」を設定します。

またこの送信元, 宛先 IPv4 アドレスが IPsec でのカプセル化対象となります。

＜ppp0 インタフェース設定＞

```
NXR_C(config)#interface ppp 0
NXR_C(config-ppp)#ip address negotiated
NXR_C(config-ppp)#ip masquerade
NXR_C(config-ppp)#ip access-group in ppp0_in
NXR_C(config-ppp)#ip spi-filter
NXR_C(config-ppp)#ip tcp adjust-mss auto
NXR_C(config-ppp)#no ip redirects
NXR_C(config-ppp)#ppp authentication pap
NXR_C(config-ppp)#ppp username test3@centurysys password test3pass
```

ppp0 インタフェースに関する設定をします。

IPv4 アクセスリスト設定で設定した「ppp0-in」を ppp0 インタフェースの「in」フィルタに適用します。

＜Ethernet1 インタフェース設定＞

```
NXR_C(config)#interface ethernet 1
NXR_C(config-if)#no ip address
NXR_C(config-if)#pppoe-client ppp 0
```

Ethernet1 インタフェースに関する設定をします。

＜スタティックルート設定＞

```
NXR_C(config)#ip route 0.0.0.0/0 ppp 0
```

デフォルトルートを設定します。

＜IPsec ローカルポリシー設定＞

```
NXR_C(config)#ipsec local policy 1
NXR_C(config-ipsec-local)#address ip
NXR_C(config-ipsec-local)#self-identity fqdn nxrc
```

IPsec のローカルポリシー「１」を設定します。

＜IPsec ISAKMP ポリシー設定１＞

```
NXR_C(config)#ipsec isakmp policy 1
NXR_C(config-ipsec-isakmp)#description NXR_A
NXR_C(config-ipsec-isakmp)#authentication pre-share ipseckey2
NXR_C(config-ipsec-isakmp)#hash sha1
NXR_C(config-ipsec-isakmp)#encryption aes128
NXR_C(config-ipsec-isakmp)#group 5
NXR_C(config-ipsec-isakmp)#lifetime 10800
NXR_C(config-ipsec-isakmp)#isakmp-mode aggressive
NXR_C(config-ipsec-isakmp)#remote address ip 10.10.10.1
NXR_C(config-ipsec-isakmp)#keepalive 30 3 periodic restart
NXR_C(config-ipsec-isakmp)#local policy 1
```

IPsec の ISAKMP ポリシー「１」を設定します。

事前共有鍵(Pre-Shared Key)として「ipseckey2」を設定します。

フェーズ１のネゴシエーションモードとして「aggressive」を設定します。

＜IPsec tunnel ポリシー設定１＞

```
NXR_C(config)#ipsec tunnel policy 1
NXR_C(config-ipsec-tunnel)#description NXR_A
NXR_C(config-ipsec-tunnel)#negotiation-mode auto
NXR_C(config-ipsec-tunnel)#set transform esp-aes128 esp-sha1-hmac
NXR_C(config-ipsec-tunnel)#set pfs group5
NXR_C(config-ipsec-tunnel)#set sa lifetime 3600
NXR_C(config-ipsec-tunnel)#set key-exchange isakmp 1
NXR_C(config-ipsec-tunnel)#match address LAN_A
```

IPsec の tunnel ポリシー「１」を設定します。

使用する IPsec アクセスリストとして「LAN_A」を設定します。

＜ppp0 インタフェース設定＞

```
NXR_C(config)#interface ppp 0
NXR_C(config-ppp)#ipsec policy 1
```

IPsec ローカルポリシー「１」を適用します。これによりこのインタフェースが IPsec トンネルのエンドポイントとなります。

＜DNS 設定＞

```
NXR_C(config)#dns
NXR_C(dns-config)#service enable
```

DNS に関する設定をします。

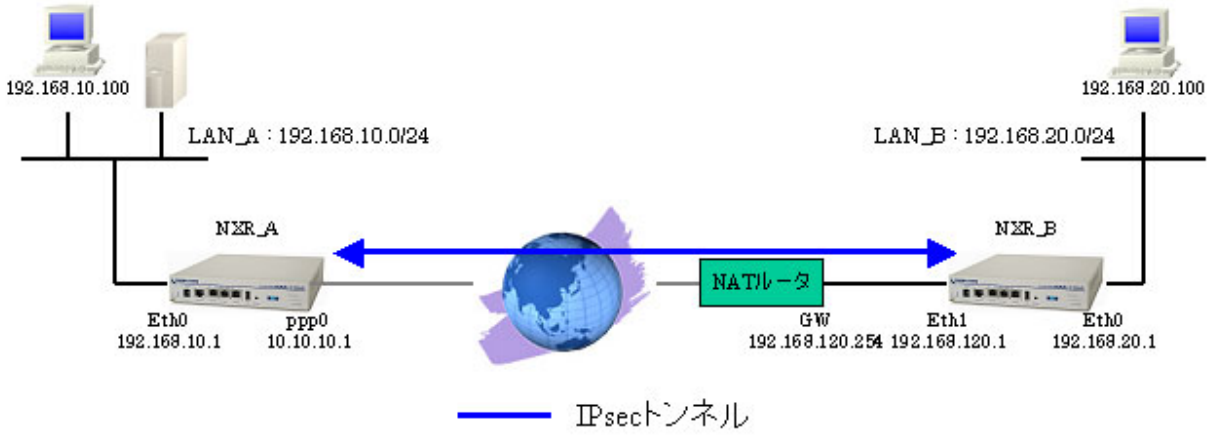
7-5-3. パソコンの設定例

	LAN A のパソコン	LAN B のパソコン	LAN C のパソコン
IPv4 アドレス	192.168.10.100	192.168.20.100	192.168.30.100
サブネットマスク	255.255.255.0	255.255.255.0	255.255.255.0
デフォルトゲートウェイ	192.168.10.1	192.168.20.1	192.168.30.1

7-6. IPsec NAT-Traversal 接続設定例

NXR の上位にインターネット接続用のルータがあり、そのルータが NAPT として動作している場合は、NXR では IPsec 接続として NAT-Traversal を利用します。

7-6-1. 構成図



7-6-2. 設定例

[NXR_A の設定]

```
nxr130#configure terminal
nxr130(config)#hostname NXR_A
NXR_A(config)#interface ethernet 0
NXR_A(config-if)#ip address 192.168.10.1/24
NXR_A(config-if)#exit
NXR_A(config)#ip access-list ppp0_in permit any 10.10.10.1 udp any 500
NXR_A(config)#ip access-list ppp0_in permit any 10.10.10.1 udp any 4500
NXR_A(config)#ipsec access-list LAN_B ip 192.168.10.0/24 192.168.20.0/24
NXR_A(config)#interface ppp 0
NXR_A(config-ppp)#ip address 10.10.10.1/32
NXR_A(config-ppp)#ip masquerade
NXR_A(config-ppp)#ip access-group in ppp0_in
NXR_A(config-ppp)#ip spi-filter
NXR_A(config-ppp)#ip tcp adjust-mss auto
NXR_A(config-ppp)#no ip redirects
NXR_A(config-ppp)#ppp authentication pap
NXR_A(config-ppp)#ppp username test1@centurysys password test1pass
NXR_A(config-ppp)#exit
NXR_A(config)#interface ethernet 1
NXR_A(config-if)#no ip address
NXR_A(config-if)#pppoe-client ppp 0
NXR_A(config-if)#exit
NXR_A(config)#ip route 0.0.0.0/0 ppp 0
NXR_A(config)#ipsec nat-traversal enable
NXR_A(config)#ipsec local policy 1
NXR_A(config-ipsec-local)#address ip
NXR_A(config-ipsec-local)#exit
```

----- 次のページに続きがあります -----

```
----- 前のページからの続きです -----

NXR_A(config)#ipsec isakmp policy 1
NXR_A(config-ipsec-isakmp)#description NXR_B
NXR_A(config-ipsec-isakmp)#authentication pre-share ipseckey
NXR_A(config-ipsec-isakmp)#hash sha1
NXR_A(config-ipsec-isakmp)#encryption aes128
NXR_A(config-ipsec-isakmp)#group 5
NXR_A(config-ipsec-isakmp)#lifetime 10800
NXR_A(config-ipsec-isakmp)#isakmp-mode aggressive
NXR_A(config-ipsec-isakmp)#remote address ip any
NXR_A(config-ipsec-isakmp)#remote identity fqdn nxrb
NXR_A(config-ipsec-isakmp)#keepalive 30 3 periodic clear
NXR_A(config-ipsec-isakmp)#local policy 1
NXR_A(config-ipsec-isakmp)#exit
NXR_A(config)#ipsec tunnel policy 1
NXR_A(config-ipsec-tunnel)#description NXR_B
NXR_A(config-ipsec-tunnel)#negotiation-mode manual
NXR_A(config-ipsec-tunnel)#set transform esp-aes128 esp-sha1-hmac
NXR_A(config-ipsec-tunnel)#set pfs group5
NXR_A(config-ipsec-tunnel)#set sa lifetime 3600
NXR_A(config-ipsec-tunnel)#set key-exchange isakmp 1
NXR_A(config-ipsec-tunnel)#match address LAN_B
NXR_A(config-ipsec-tunnel)#exit
NXR_A(config)#interface ppp 0
NXR_A(config-ppp)#ipsec policy 1
NXR_A(config-ppp)#exit
NXR_A(config)#dns
NXR_A(dns-config)#service enable
NXR_A(dns-config)#exit
NXR_A(config)#exit
NXR_A#save config
```

【 解説 】

- フェーズ１ではアグレッシブモードをフェーズ２ではクイックモードを利用します。
- NAT-Traversal の設定を有効にし、IPsec Tunnel Policy 内で NAT-Traversal を使用する IPsec アクセスリストを設定します。
- NAT-Traversal 利用時は IPsec のパケットは UDP でカプセル化 (ESP ではない) されますので、UDP でフィルタを許可する必要があります。(設定内容は後述)
- インターネットアクセスを可能にするために、フィルタ設定 (SPI), NAT 設定 (IP マスカレード), DNS 設定を行っています。

＜ホスト名の設定＞

```
nxr130(config)#hostname NXR_A
```

ホスト名として「NXR_A」を設定します。

＜Ethernet0 インタフェース設定＞

```
NXR_A(config)#interface ethernet 0
NXR_A(config-if)#ip address 192.168.10.1/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.1/24」を設定します。

＜IPv4 アクセスリスト設定＞

```
NXR_A(config)#ip access-list ppp0_in permit any 10.10.10.1 udp any 500
NXR_A(config)#ip access-list ppp0_in permit any 10.10.10.1 udp any 4500
```

IPv4 アクセスリスト名を「ppp0_in」とし、以下のルールで設定します。

動作	送信元 IPv4 アドレス	宛先 IPv4 アドレス	プロトコル	送信元ポート	宛先ポート
許可	any	10.10.10.1	UDP	any	500
許可	any	10.10.10.1	UDP	any	4500

＜IPsec アクセスリスト設定＞

```
NXR_A(config)#ipsec access-list LAN_B ip 192.168.10.0/24 192.168.20.0/24
```

IPsec アクセスリスト名を「LAN_B」とし、送信元 IPv4 アドレス「192.168.10.0/24」, 宛先 IPv4 アドレス「192.168.20.0/24」を設定します。

またこの送信元, 宛先 IPv4 アドレスが IPsec でのカプセル化対象となります。

＜ppp0 インタフェース設定＞

```
NXR_A(config)#interface ppp 0
NXR_A(config-ppp)#ip address 10.10.10.1/32
NXR_A(config-ppp)#ip masquerade
NXR_A(config-ppp)#ip access-group in ppp0_in
NXR_A(config-ppp)#ip spi-filter
NXR_A(config-ppp)#ip tcp adjust-mss auto
NXR_A(config-ppp)#no ip redirects
NXR_A(config-ppp)#ppp authentication pap
NXR_A(config-ppp)#ppp username test1@centurysys password test1pass
```

ppp0 インタフェースに関する設定をします。

IPv4 アドレスが固定のため、「10.10.10.1/32」を設定します。

IPv4 アクセスリスト設定で設定した「ppp0-in」を ppp0 インタフェースの「in」フィルタに適用します。

＜Ethernet1 インタフェース設定＞

```
NXR_A(config)#interface ethernet 1
NXR_A(config-if)#no ip address
NXR_A(config-if)#pppoe-client ppp 0
```

Ethernet1 インタフェースに関する設定をします。

＜スタティックルート設定＞

```
NXR_A(config)#ip route 0.0.0.0/0 ppp 0
```

デフォルトルートを設定します。

＜IPsec NAT-Traversal の有効化＞

```
NXR_A(config)#ipsec nat-traversal enable
```

NAT-Traversal を利用する場合は、有効にします。

＜IPsec ローカルポリシー設定＞

```
NXR_A(config)#ipsec local policy 1
NXR_A(config-ipsec-local)#address ip
```

IPsec のローカルポリシー「１」を設定します。

＜IPsec ISAKMP ポリシー設定１＞

```
NXR_A(config)#ipsec isakmp policy 1
NXR_A(config-ipsec-isakmp)#description NXR_B
NXR_A(config-ipsec-isakmp)#authentication pre-share ipseckey
NXR_A(config-ipsec-isakmp)#hash sha1
NXR_A(config-ipsec-isakmp)#encryption aes128
NXR_A(config-ipsec-isakmp)#group 5
NXR_A(config-ipsec-isakmp)#lifetime 10800
NXR_A(config-ipsec-isakmp)#isakmp-mode aggressive
NXR_A(config-ipsec-isakmp)#remote address ip any
NXR_A(config-ipsec-isakmp)#remote identity fqdn nxrb
NXR_A(config-ipsec-isakmp)#keepalive 30 3 periodic clear
NXR_A(config-ipsec-isakmp)#local policy 1
```

IPsec の ISAKMP ポリシー「１」を設定します。

事前共有鍵(Pre-Shared Key)として「ipseckey」を設定します。

フェーズ１のネゴシエーションモードとして「aggressive」を設定します。

＜IPsec tunnel ポリシー設定１＞

```
NXR_A(config)#ipsec tunnel policy 1
NXR_A(config-ipsec-tunnel)#description NXR_B
NXR_A(config-ipsec-tunnel)#negotiation-mode manual
NXR_A(config-ipsec-tunnel)#set transform esp-aes128 esp-sha1-hmac
NXR_A(config-ipsec-tunnel)#set pfs group5
NXR_A(config-ipsec-tunnel)#set sa lifetime 3600
NXR_A(config-ipsec-tunnel)#set key-exchange isakmp 1
```

IPsec の tunnel ポリシー「１」を設定します。

```
NXR_A(config-ipsec-tunnel)#match address LAN_B
```

NAT-Traversal で使用する IPsec アクセスリストとして「LAN_B」を設定します。

＜ppp0 インタフェース設定＞

```
NXR_A(config)#interface ppp 0
NXR_A(config-ppp)#ipsec policy 1
```

IPsec ローカルポリシー「１」を適用します。これによりこのインタフェースが IPsec トンネルのエンドポイントとなります。

＜DNS 設定＞

```
NXR_A(config)#dns
NXR_A(dns-config)#service enable
```

DNS に関する設定をします。

[NXR_B の設定]

```
nxr130#configure terminal
nxr130(config)#hostname NXR_B
NXR_B(config)#interface ethernet 0
NXR_B(config-if)#ip address 192.168.20.1/24
NXR_B(config-if)#exit
NXR_B(config)#ip route 0.0.0.0/0 192.168.120.254
NXR_B(config)#ipsec access-list LAN_A ip 192.168.20.0/24 192.168.10.0/24
NXR_B(config)#ipsec nat-traversal enable
NXR_B(config)#ipsec local policy 1
NXR_B(config-ipsec-local)#address ip
NXR_B(config-ipsec-local)#self-identity fqdn nxrb
NXR_B(config-ipsec-local)#exit
NXR_B(config)#ipsec isakmp policy 1
NXR_B(config-ipsec-isakmp)#description NXR_A
NXR_B(config-ipsec-isakmp)#authentication pre-share ipseckey
NXR_B(config-ipsec-isakmp)#hash sha1
NXR_B(config-ipsec-isakmp)#encryption aes128
NXR_B(config-ipsec-isakmp)#group 5
NXR_B(config-ipsec-isakmp)#isakmp-mode aggressive
NXR_B(config-ipsec-isakmp)#remote address ip 10.10.10.1
NXR_B(config-ipsec-isakmp)#keepalive 30 3 periodic restart
NXR_B(config-ipsec-isakmp)#local policy 1
NXR_B(config-ipsec-isakmp)#exit
NXR_B(config)#ipsec tunnel policy 1
NXR_B(config-ipsec-tunnel)#description NXR_A
NXR_B(config-ipsec-tunnel)#negotiation-mode auto
NXR_B(config-ipsec-tunnel)#set transform esp-aes128 esp-sha1-hmac
NXR_B(config-ipsec-tunnel)#set pfs group5
NXR_B(config-ipsec-tunnel)#set key-exchange isakmp 1
NXR_B(config-ipsec-tunnel)#match address LAN_A
NXR_B(config-ipsec-tunnel)#exit
NXR_B(config)#interface ethernet 1
NXR_B(config-if)#ip address 192.168.120.1/24
NXR_B(config-if)#ipsec policy 1
NXR_B(config-if)#exit
NXR_B(config)#dns
NXR_B(dns-config)#service enable
NXR_B(dns-config)#exit
NXR_B(config)#exit
NXR_B#save config
```

【 解説 】

- フェーズ１ではアグレッシブモードをフェーズ２ではクイックモードを利用します。
- ISAKMP ポリシー(フェーズ１), tunnel ポリシー(フェーズ２)で利用するプロポーザルは「NXR_A」と同一です。
- 機器の IP アドレスを ID として利用せず、ID（ID Type ： FQDN）を設定します。
- 事前共有鍵は対向機器と同一のもの（ここでは ipseckey）を使用する必要があります。

＜ホスト名の設定＞

```
nxr130(config)#hostname NXR_B
```

ホスト名として「NXR_B」を設定します。

＜Ethernet0 インタフェース設定＞

```
NXR_B(config)#interface ethernet 0
NXR_B(config-if)#ip address 192.168.20.1/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.20.1/24」を設定します。

＜スタティックルート設定＞

```
NXR_B(config)#ip route 0.0.0.0/0 192.168.120.254
```

デフォルトルートを設定します。

＜IPsec アクセスリスト設定＞

```
NXR_B(config)#ipsec access-list LAN_A ip 192.168.20.0/24 192.168.10.0/24
```

IPsec アクセスリスト名を「LAN_A」とし、送信元 IPv4 アドレス「192.168.20.0/24」, 宛先 IPv4 アドレス「192.168.10.0/24」を設定します。

またこの送信元, 宛先 IPv4 アドレスが IPsec でのカプセル化対象となります。

＜IPsec NAT-Traversal の有効化＞

```
NXR_B(config)#ipsec nat-traversal enable
```

NAT-Traversal を利用する場合は、有効にします。

＜IPsec ローカルポリシー設定＞

```
NXR_B(config)#ipsec local policy 1
```

IPsec のローカルポリシー「１」を設定します。

```
NXR_B(config-ipsec-local)#address ip
```

IPsec トンネルの送信元 IP として IPv4 アドレスを設定します。

```
NXR_B(config-ipsec-local)#self-identity fqdn nxrb
```

機器の IP アドレスを ID として利用せず、ID として「nxrb」を設定します。(fqdn 方式で設定しています)

＜IPsec ISAKMP ポリシー設定＞

```
NXR_B(config)#ipsec isakmp policy 1
NXR_B(config-ipsec-isakmp)#description NXR_A
NXR_B(config-ipsec-isakmp)#authentication pre-share ipseckey
NXR_B(config-ipsec-isakmp)#hash sha1
NXR_B(config-ipsec-isakmp)#encryption aes128
NXR_B(config-ipsec-isakmp)#group 5
NXR_B(config-ipsec-isakmp)#isakmp-mode aggressive
NXR_B(config-ipsec-isakmp)#remote address ip 10.10.10.1
NXR_B(config-ipsec-isakmp)#keepalive 30 3 periodic restart
NXR_B(config-ipsec-isakmp)#local policy 1
```

IPsec の ISAKMP ポリシー「１」を設定します。

機器の WAN 側 IPv4 アドレスが動的 IP アドレスのため、フェーズ１のネゴシエーションモードとして「aggressive」を設定します。

＜IPsec tunnel ポリシー設定＞

```
NXR_B(config)#ipsec tunnel policy 1
NXR_B(config-ipsec-tunnel)#description NXR_A
NXR_B(config-ipsec-tunnel)#negotiation-mode auto
NXR_B(config-ipsec-tunnel)#set transform esp-aes128 esp-sha1-hmac
NXR_B(config-ipsec-tunnel)#set pfs group5
NXR_B(config-ipsec-tunnel)#set key-exchange isakmp 1
NXR_B(config-ipsec-tunnel)#match address LAN_A
```

IPsec の tunnel ポリシー「１」を設定します。

＜Ethernet1 インタフェース設定＞

```
NXR_B(config)#interface ethernet 1
NXR_B(config-if)#ip address 192.168.120.1/24
NXR_B(config-if)#ipsec policy 1
```

Ethernet1 インタフェースの IPv4 アドレスとして「192.168.120.1/24」を設定します。

IPsec ローカルポリシー「１」を適用します。これによりこのインタフェースが IPsec トンネルのエンドポイントとなります。

＜DNS 設定＞

```
NXR_B(config)#dns
NXR_B(dns-config)#service enable
```

DNS に関する設定をします。

7-6-3. パソコンの設定例

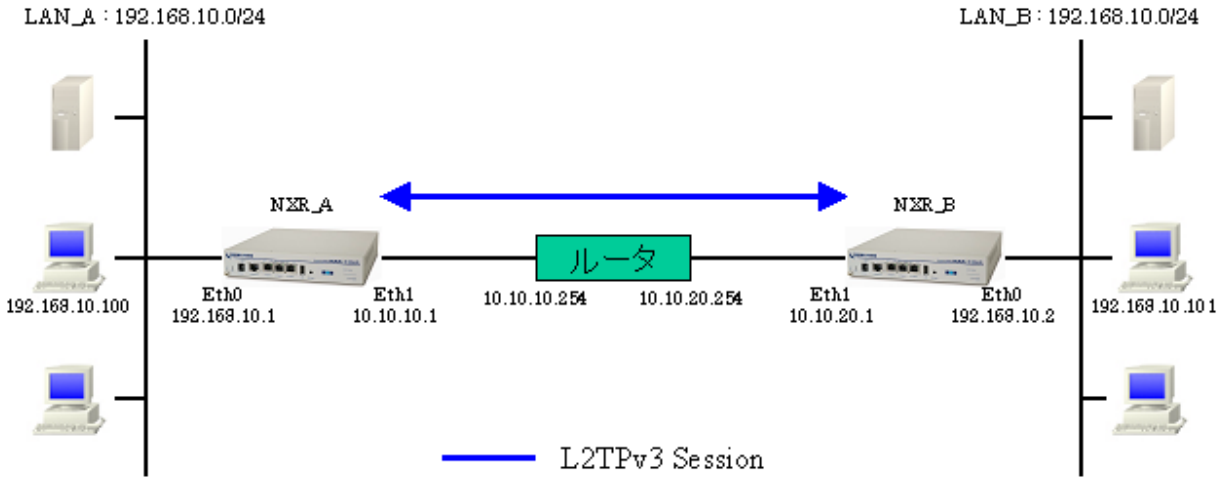
	LAN A のパソコン	LAN B のパソコン
IPv4 アドレス	192.168.10.100	192.168.20.100
サブネットマスク	255.255.255.0	255.255.255.0
デフォルトゲートウェイ	192.168.10.1	192.168.20.1

8. L2TPv3 設定

8-1. L2TPv3 での LAN 間接続設定例

同一ネットワークである LAN A「192.168.10.0/24」と LAN B「192.168.10.0/24」にそれぞれ属する NXR_A, NXR_B 間で L2TPv3 によりトンネルを構築することにより、同一ネットワークである LAN 間の通信を可能にします。

8-1-1. 構成図



8-1-2. 設定例

[NXR_A の設定]

```
nxr130#configure terminal
nxr130(config)#hostname NXR_A
NXR_A(config)#interface ethernet 0
NXR_A(config-if)#ip address 192.168.10.1/24
NXR_A(config-if)#exit
NXR_A(config)#interface ethernet 1
NXR_A(config-if)#ip address 10.10.10.1/24
NXR_A(config-if)#exit
NXR_A(config)#ip route 0.0.0.0/0 10.10.10.254
NXR_A(config)#l2tpv3 hostname nxra
NXR_A(config)#l2tpv3 router-id 172.20.10.1
NXR_A(config)#l2tpv3 mac-learning
NXR_A(config)#l2tpv3 mac-aging 300
NXR_A(config)#l2tpv3 path-mtu-discovery
NXR_A(config)#l2tpv3 tunnel 1
NXR_A(config-l2tpv3-tunnel)#description NXR_B
NXR_A(config-l2tpv3-tunnel)#tunnel address 10.10.20.1
NXR_A(config-l2tpv3-tunnel)#tunnel hostname nxrb
NXR_A(config-l2tpv3-tunnel)#tunnel router-id 172.20.20.1
NXR_A(config-l2tpv3-tunnel)#tunnel vendor ietf
NXR_A(config-l2tpv3-tunnel)#exit
NXR_A(config)#l2tpv3 xconnect 1
NXR_A(config-l2tpv3-xconnect)#description NXR_B

----- 次のページに続きがあります -----
```

```
----- 前のページからの続きです -----

NXR_A(config-l2tpv3-xconnect)#tunnel 1
NXR_A(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_A(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_A(config-l2tpv3-xconnect)#retry-interval 30
NXR_A(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
NXR_A(config-l2tpv3-xconnect)#exit
NXR_A(config)#exit
NXR_A#save config
```

【 解説 】

- L2TPv3 の Xconnect インタフェースを「ethernet0」とし、このインタフェースが L2 フレームを送受信するインタフェースとなります。
- L2TPv3 トンネル/セッションが切断されたときに自動再接続できるように、リトライインターバルを設定します。

＜ホスト名の設定＞

```
nxr130(config)#hostname NXR_A
```

ホスト名として「NXR_A」を設定します。

＜Ethernet0 インタフェース設定＞

```
NXR_A(config)#interface ethernet 0
NXR_A(config-if)#ip address 192.168.10.1/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.1/24」を設定します。

＜Ethernet1 インタフェース設定＞

```
NXR_A(config)#interface ethernet 1
NXR_A(config-if)#ip address 10.10.10.1/24
```

Ethernet1 インタフェースの IPv4 アドレスとして「10.10.10.1/24」を設定します。

＜スタティックルート設定＞

```
NXR_A(config)#ip route 0.0.0.0/0 10.10.10.254
```

デフォルトルートを設定します。

＜L2TPv3 設定＞

```
NXR_A(config)#l2tpv3 hostname nxra
```

L2TPv3 のホスト名として「nxra」を設定します。省略時は「hostname」コマンドで設定した値が使用されます。

```
NXR_A(config)#l2tpv3 router-id 172.20.10.1
```

ルータ ID として「172.20.10.1」を設定します。

```
NXR_A(config)#l2tpv3 mac-learning
```

MAC アドレス学習機能を有効にします。(デフォルト値は有効に設定されています)

```
NXR_A(config)#l2tpv3 mac-aging 300
```
MAC アドレス学習機能での MAC アドレスエージングタイムを「300」秒に設定します。(デフォルト値は 300 秒に設定されています)

```
NXR_A(config)#l2tpv3 path-mtu-discovery
```
Path MTU Discovery を有効にします。

＜L2TPv3 トンネル設定＞
```
NXR_A(config)#l2tpv3 tunnel 1
```
L2TPv3 トンネル「１」を設定します。

```
NXR_A(config-l2tpv3-tunnel)#description NXR_B
```
L2TPv3 トンネル「１」の名前を「NXR_B」と設定します。

```
NXR_A(config-l2tpv3-tunnel)#tunnel address 10.10.20.1
```
対向機器（対向 LCCE）の IPv4 アドレス「10.10.20.1」を設定します。

```
NXR_A(config-l2tpv3-tunnel)#tunnel hostname nxrb
```
対向機器の L2TPv3 ホスト名「nxrb」を設定します。対向機器で設定した「l2tpv3 hostname」コマンドの値と同一にする必要があります。

```
NXR_A(config-l2tpv3-tunnel)#tunnel router-id 172.20.20.1
```
対向機器の L2TPv3 ルータ ID「172.20.20.1」を設定します。対向機器で設定した「l2tpv3 router-id」コマンドの値と同一にする必要があります。

```
NXR_A(config-l2tpv3-tunnel)#tunnel vendor ietf
```
対向機器のベンダーID として「ietf」を設定します。対向機器で設定した「tunnel vendor」コマンドの値と同一にする必要があります。

＜L2TPv3 Xconnect 設定＞
```
NXR_A(config)#l2tpv3 xconnect 1
```
L2TPv3 Xconnect「１」を設定します。

```
NXR_A(config-l2tpv3-xconnect)#description NXR_B
```
L2TPv3 Xconnect「１」の名前を「NXR_B」と設定します。

```
NXR_A(config-l2tpv3-xconnect)#tunnel 1
```
使用する L2TPv3 トンネルとして「１」(l2tpv3 tunnel 1) を設定します。

```
NXR_A(config-l2tpv3-xconnect)#xconnect ethernet 0
```

Xconnect インタフェースとして「ethernet 0」を設定します。

※このインタフェースが L2 フレームを送受信するインタフェースとなります。

```
NXR_A(config-l2tpv3-xconnect)#xconnect end-id 1
```

リモート End-ID として「1」を設定します。Xconnect インタフェースを識別する際に使用する ID のため、対向機器の設定した「xconnect end-id」コマンドの値と同一にする必要があります。

```
NXR_A(config-l2tpv3-xconnect)#retry-interval 30
```

リトライインターバルとして「30」秒を設定します。

リトライインターバルは、トンネル/セッションが切断したときに自動再接続を開始するまでの間隔を設定します。

```
NXR_A(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
```

TCP MSS の調整機能をオートに設定します。

[NXR_B の設定]

```
nxr130#configure terminal
nxr130(config)#hostname NXR_B
NXR_B(config)#interface ethernet 0
NXR_B(config-if)#ip address 192.168.10.2/24
NXR_B(config-if)#exit
NXR_B(config)#interface ethernet 1
NXR_B(config-if)#ip address 10.10.20.1/24
NXR_B(config-if)#exit
NXR_B(config)#ip route 0.0.0.0/0 10.10.20.254
NXR_B(config)#l2tpv3 hostname nxrb
NXR_B(config)#l2tpv3 router-id 172.20.20.1
NXR_B(config)#l2tpv3 mac-learning
NXR_B(config)#l2tpv3 mac-aging 300
NXR_B(config)#l2tpv3 path-mtu-discovery
NXR_B(config)#l2tpv3 tunnel 1
NXR_B(config-l2tpv3-tunnel)#description NXR_A
NXR_B(config-l2tpv3-tunnel)#tunnel address 10.10.10.1
NXR_B(config-l2tpv3-tunnel)#tunnel hostname nxra
NXR_B(config-l2tpv3-tunnel)#tunnel router-id 172.20.10.1
NXR_B(config-l2tpv3-tunnel)#tunnel vendor ietf
NXR_B(config-l2tpv3-tunnel)#exit
NXR_B(config)#l2tpv3 xconnect 1
NXR_B(config-l2tpv3-xconnect)#description NXR_A
NXR_B(config-l2tpv3-xconnect)#tunnel 1
NXR_B(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_B(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_B(config-l2tpv3-xconnect)#retry-interval 45
NXR_B(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
NXR_B(config-l2tpv3-xconnect)#exit
NXR_B(config)#exit
NXR_B#save config
```

【 解説 】

- L2TPv3 の Xconnect インタフェースを「ethernet0」とし、このインタフェースが L2 フレームを送受信するインタフェースとなります。
- L2TPv3 トンネル/セッションが切断されたときに自動再接続できるように、リトライインターバルを設定します。

＜ホスト名の設定＞

```
nxr130(config)#hostname NXR_B
```

ホスト名として「NXR_B」を設定します。

＜Ethernet0 インタフェース設定＞

```
NXR_B(config)#interface ethernet 0
NXR_B(config-if)#ip address 192.168.10.2/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.2/24」を設定します。

＜Ethernet1 インタフェース設定＞

```
NXR_B(config)#interface ethernet 1
NXR_B(config-if)#ip address 10.10.20.1/24
```

Ethernet1 インタフェースの IPv4 アドレスとして「10.10.20.1/24」を設定します。

＜スタティックルート設定＞

```
NXR_B(config)#ip route 0.0.0.0/0 10.10.20.254
```

デフォルトルートを設定します。

＜L2TPv3 設定＞

```
NXR_B(config)#l2tpv3 hostname nxrb
```

L2TPv3 のホスト名として「nxrb」を設定します。省略時は「hostname」コマンドで設定した値が使用されます。

```
NXR_B(config)#l2tpv3 router-id 172.20.20.1
```

ルータ ID として「172.20.20.1」を設定します。

```
NXR_B(config)#l2tpv3 mac-learning
```

MAC アドレス学習機能を有効にします。(デフォルト値は有効に設定されています)

```
NXR_B(config)#l2tpv3 mac-aging 300
```

MAC アドレス学習機能での MAC アドレスエージングタイムを「300」秒に設定します。(デフォルト値は 300 秒に設定されています)

```
NXR_B(config)#l2tpv3 path-mtu-discovery
```

Path MTU Discovery を有効にします。

＜L2TPv3 トンネル設定＞

```
NXR_B(config)#l2tpv3 tunnel 1
```

L2TPv3 トンネル「１」を設定します。

```
NXR_B(config-l2tpv3-tunnel)#description NXR_A
```

L2TPv3 トンネル「１」の名前を「NXR_A」と設定します。

```
NXR_B(config-l2tpv3-tunnel)#tunnel address 10.10.10.1
```

対向機器（対向 LCCE）の IPv4 アドレス「10.10.10.1」を設定します。

```
NXR_B(config-l2tpv3-tunnel)#tunnel hostname nxra
```

対向機器の L2TPv3 ホスト名「nxra」を設定します。対向機器で設定した「l2tpv3 hostname」コマンドの値と同一にする必要があります。

```
NXR_B(config-l2tpv3-tunnel)#tunnel router-id 172.20.10.1
```

対向機器の L2TPv3 ルータ ID「172.20.10.1」を設定します。対向機器で設定した「l2tpv3 router-id」コマンドの値と同一にする必要があります。

```
NXR_B(config-l2tpv3-tunnel)#tunnel vendor ietf
```

対向機器のベンダーID として「ietf」を設定します。対向機器で設定した「tunnel vendor」コマンドの値と同一にする必要があります。

＜L2TPv3 Xconnect 設定＞

```
NXR_B(config)#l2tpv3 xconnect 1
```

L2TPv3 Xconnect「１」を設定します。

```
NXR_B(config-l2tpv3-xconnect)#description NXR_A
```

L2TPv3 Xconnect「１」の名前を「NXR_A」と設定します。

```
NXR_B(config-l2tpv3-xconnect)#tunnel 1
```

使用する L2TPv3 トンネルとして「１」(l2tpv3 tunnel 1) を設定します。

```
NXR_B(config-l2tpv3-xconnect)#xconnect Ethernet 0
```

Xconnect インタフェースとして「ethernet 0」を設定します。
※このインタフェースが L2 フレームを送受信するインタフェースとなります。

```
NXR_B(config-l2tpv3-xconnect)#xconnect end-id 1
```

リモート End-ID として「1」を設定します。Xconnect インタフェースを識別する際に使用する ID のため、対向機器の設定した「xconnect end-id」コマンドの値と同一にする必要があります。

```
NXR_B(config-l2tpv3-xconnect)#retry-interval 45
```

リトライインターバルとして「45」秒を設定します。
リトライインターバルは、トンネル/セッションが切断したときに自動再接続を開始するまでの間隔を設定します。

```
NXR_B(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
```

TCP MSS の調整機能をオートに設定します。

8-1-3. パソコンの設定例

	LAN A のパソコン	LAN B のパソコン
IPv4 アドレス	192.168.10.100	192.168.10.101
サブネットマスク	255.255.255.0	255.255.255.0
デフォルトゲートウェイ	192.168.10.1	192.168.10.2

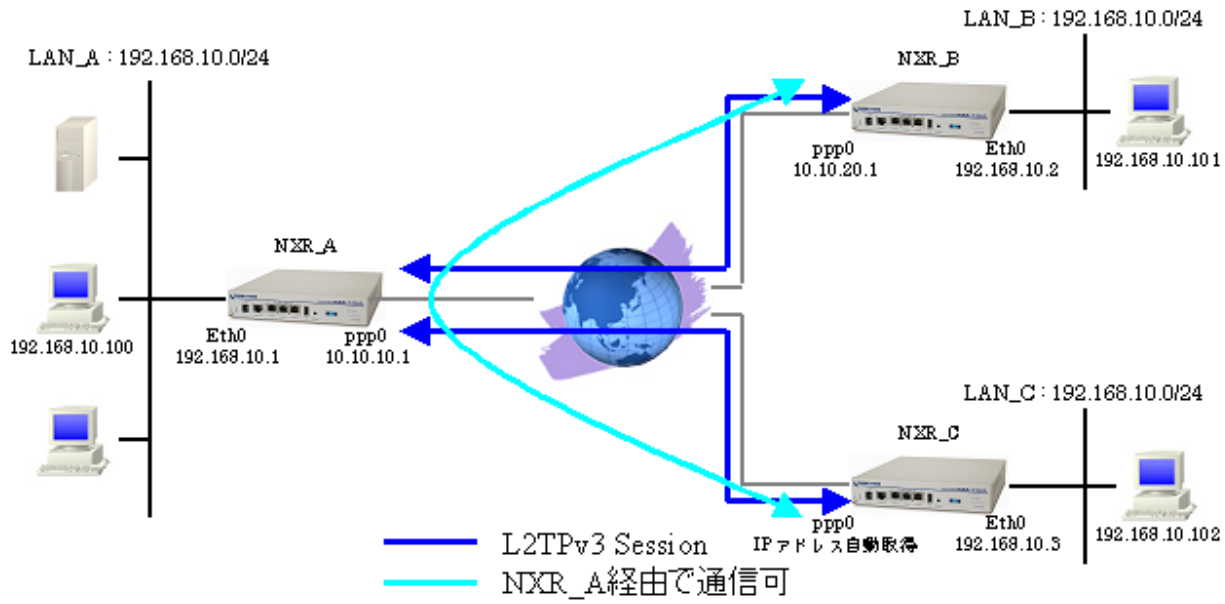
8-2. PPPoE を利用した L2TPv3 接続設定例

PPPoE 上でも L2TPv3 を利用することは可能です。ここでは NXR_A(センタ)－NXR_B(拠点)間および NXR_A（センタ）－NXR_C（拠点）間で L2TPv3 接続しています。なお、この設定例では NXR_A を経由した拠点間通信を行うことは可能です。

※センター拠点間の通信のみで、拠点間通信を行わない場合は、「補足 スプリットホライズンの設定例」もご参照下さい。

なおここでは、各拠点からのインターネットアクセスを可能にするために、フィルタ設定（SPI），NAT 設定（IP マスカレード），DNS 設定を行っています。

8-2-1. 構成図



8-2-2. 設定例

[NXR_A の設定]

```
nxr130#configure terminal
nxr130(config)#hostname NXR_A
NXR_A(config)#interface ethernet 0
NXR_A(config-if)#ip address 192.168.10.1/24
NXR_A(config-if)#exit
NXR_A(config)#ip access-list ppp0_in permit any 10.10.10.1 115
NXR_A(config)#interface ppp 0
NXR_A(config-ppp)#ip address 10.10.10.1/32
NXR_A(config-ppp)#ip masquerade
NXR_A(config-ppp)#ip access-group in ppp0_in
NXR_A(config-ppp)#ip spi-filter
NXR_A(config-ppp)#ip tcp adjust-mss auto
NXR_A(config-ppp)#no ip redirects
NXR_A(config-ppp)#ppp authentication pap
NXR_A(config-ppp)#ppp username test1@centurysys password test1pass
NXR_A(config-ppp)#exit
NXR_A(config)#interface ethernet 1
NXR_A(config-if)#no ip address
NXR_A(config-if)#pppoe-client ppp 0

----- 次のページに続きがあります -----
```

```
----- 前のページからの続きです -----

NXR_A(config-if)#exit
NXR_A(config)#ip route 0.0.0.0/0 ppp 0
NXR_A(config)#l2tpv3 hostname nxra
NXR_A(config)#l2tpv3 router-id 172.20.10.1
NXR_A(config)#l2tpv3 mac-learning
NXR_A(config)#l2tpv3 mac-aging 300
NXR_A(config)#l2tpv3 path-mtu-discovery
NXR_A(config)#l2tpv3 tunnel 1
NXR_A(config-l2tpv3-tunnel)#description NXR_B
NXR_A(config-l2tpv3-tunnel)#tunnel address 10.10.20.1
NXR_A(config-l2tpv3-tunnel)#tunnel hostname nxrb
NXR_A(config-l2tpv3-tunnel)#tunnel router-id 172.20.20.1
NXR_A(config-l2tpv3-tunnel)#tunnel vendor ietf
NXR_A(config-l2tpv3-tunnel)#exit
NXR_A(config)#l2tpv3 xconnect 1
NXR_A(config-l2tpv3-xconnect)#description NXR_B
NXR_A(config-l2tpv3-xconnect)#tunnel 1
NXR_A(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_A(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_A(config-l2tpv3-xconnect)#retry-interval 30
NXR_A(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
NXR_A(config-l2tpv3-xconnect)#exit
NXR_A(config)#l2tpv3 tunnel 2
NXR_A(config-l2tpv3-tunnel)#description NXR_C
NXR_A(config-l2tpv3-tunnel)#tunnel hostname nxrc
NXR_A(config-l2tpv3-tunnel)#tunnel router-id 172.20.30.1
NXR_A(config-l2tpv3-tunnel)#tunnel vendor ietf
NXR_A(config-l2tpv3-tunnel)#exit
NXR_A(config)#l2tpv3 xconnect 2
NXR_A(config-l2tpv3-xconnect)#description NXR_C
NXR_A(config-l2tpv3-xconnect)#tunnel 2
NXR_A(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_A(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_A(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
NXR_A(config-l2tpv3-xconnect)#exit
NXR_A(config)#dns
NXR_A(dns-config)#service enable
NXR_A(dns-config)# exit
NXR_A(config)#exit
NXR_A#save config
```

【 解説 】

- L2TPv3 の Xconnect インタフェースを「ethernet0」とし、このインタフェースが L2 フレームを送受信するインタフェースとなります。
- L2TPv3 トンネル/セッションが切断されたときに自動再接続できるように、リトライインターバルを設定します。ただし NXR_C に対しては WAN 側 IP アドレスが動的のため、こちらからはネゴシエーションを行わないようにします。

＜ホスト名の設定＞

```
nxr130(config)#hostname NXR_A
```

ホスト名として「NXR_A」を設定します。

＜Ethernet0 インタフェース設定＞

```
NXR_A(config)#interface ethernet 0
NXR_A(config-if)#ip address 192.168.10.1/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.1/24」を設定します。

＜IPv4 アクセスリスト設定＞

```
NXR_A(config)#ip access-list ppp0_in permit any 10.10.10.1 115
```

IPv4 アクセスリスト名を「ppp0_in」とし、宛先 IPv4 アドレス「10.10.10.1」, プロトコル番号 115 (L2TP) のパケットは許可します。

動作	送信元 IPv4 アドレス	宛先 IPv4 アドレス	プロトコル	送信元ポート	宛先ポート
許可	any	10.10.10.1	115(L2TP)		

＜ppp0 インタフェース設定＞

```
NXR_A(config)#interface ppp 0
NXR_A(config-ppp)#ip address 10.10.10.1/32
NXR_A(config-ppp)#ip masquerade
NXR_A(config-ppp)#ip access-group in ppp0_in
NXR_A(config-ppp)#ip spi-filter
NXR_A(config-ppp)#ip tcp adjust-mss auto
NXR_A(config-ppp)#no ip redirects
NXR_A(config-ppp)#ppp authentication pap
NXR_A(config-ppp)#ppp username test1@centurysys password test1pass
```

ppp0 インタフェースに関する設定をします。

IPv4 アドレスが固定のため、「10.10.10.1/32」を設定します。

IPv4 アクセスリスト設定で設定した「ppp0_in」を ppp0 インタフェースの「in」フィルタに適用します。

＜Ethernet1 インタフェース設定＞

```
NXR_A(config)#interface ethernet 1
NXR_A(config-if)#no ip address
NXR_A(config-if)#pppoe-client ppp 0
```

Ethernet1 インタフェースに関する設定をします。

＜スタティックルート設定＞

```
NXR_A(config)#ip route 0.0.0.0/0 ppp 0
```

デフォルトルートを設定します。

＜L2TPv3 設定＞

```
NXR_A(config)#l2tpv3 hostname nxra
```

L2TPv3 のホスト名として「nxra」を設定します。省略時は「hostname」コマンドで設定した値が使用されます。

```
NXR_A(config)#l2tpv3 router-id 172.20.10.1
```

ルータ ID として「172.20.10.1」を設定します。

```
NXR_A(config)#l2tpv3 mac-learning
```

MAC アドレス学習機能を有効にします。(デフォルト値は有効に設定されています)

```
NXR_A(config)#l2tpv3 mac-aging 300
```
MAC アドレス学習機能での MAC アドレスエージングタイムを「300」秒に設定します。(デフォルト値は 300 秒に設定されています)

```
NXR_A(config)#l2tpv3 path-mtu-discovery
```
Path MTU Discovery を有効にします。

＜L2TPv3 トンネル設定１＞
```
NXR_A(config)#l2tpv3 tunnel 1
```
L2TPv3 トンネル「１」を設定します。

```
NXR_A(config-l2tpv3-tunnel)#description NXR_B
```
L2TPv3 トンネル「１」の名前を「NXR_B」と設定します。

```
NXR_A(config-l2tpv3-tunnel)#tunnel address 10.10.20.1
```
対向機器（対向 LCCE）の IPv4 アドレス「10.10.20.1」を設定します。

```
NXR_A(config-l2tpv3-tunnel)#tunnel hostname nxrb
```
対向機器の L2TPv3 ホスト名「nxrb」を設定します。対向機器で設定した「l2tpv3 hostname」コマンドの値と同一にする必要があります。

```
NXR_A(config-l2tpv3-tunnel)#tunnel router-id 172.20.20.1
```
対向機器の L2TPv3 ルータ ID「172.20.20.1」を設定します。対向機器で設定した「l2tpv3 router-id」コマンドの値と同一にする必要があります。

```
NXR_A(config-l2tpv3-tunnel)#tunnel vendor ietf
```
対向機器のベンダーID として「ietf」を設定します。対向機器で設定した「tunnel vendor」コマンドの値と同一にする必要があります。

＜L2TPv3 Xconnect 設定１＞
```
NXR_A(config)#l2tpv3 xconnect 1
```
L2TPv3 Xconnect「１」を設定します。

```
NXR_A(config-l2tpv3-xconnect)#description NXR_B
```
L2TPv3 Xconnect「１」の名前を「NXR_B」と設定します。

```
NXR_A(config-l2tpv3-xconnect)#tunnel 1
```
使用する L2TPv3 トンネルとして「１」(l2tpv3 tunnel 1) を設定します。

```
NXR_A(config-l2tpv3-xconnect)#xconnect ethernet 0
```
Xconnect インタフェースとして「ethernet 0」を設定します。
※このインタフェースが L2 フレームを送受信するインタフェースとなります。

```
NXR_A(config-l2tpv3-xconnect)#xconnect end-id 1
```
リモート End-ID として「1」を設定します。Xconnect インタフェースを識別する際に使用する ID のため、対向機器の設定した「xconnect end-id」コマンドの値と同一にする必要があります。

```
NXR_A(config-l2tpv3-xconnect)#retry-interval 30
```
リトライインターバルとして「30」秒を設定します。
リトライインターバルは、トンネル/セッションが切断したときに自動再接続を開始するまでの間隔を設定します。

```
NXR_A(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
```
TCP MSS の調整機能をオートに設定します。

＜L2TPv3 トンネル設定２＞
```
NXR_A(config)#l2tpv3 tunnel 2
```
L2TPv3 トンネル「２」を設定します。

```
NXR_A(config-l2tpv3-tunnel)#description NXR_C
```
L2TPv3 トンネル「２」の名前を「NXR_C」と設定します。

```
NXR_A(config-l2tpv3-tunnel)#tunnel hostname nxrc
```
対向機器の L2TPv3 ホスト名「nxrc」を設定します。対向機器で設定した「l2tpv3 hostname」コマンドの値と同一にする必要があります。

```
NXR_A(config-l2tpv3-tunnel)#tunnel router-id 172.20.30.1
```
対向機器の L2TPv3 ルータ ID「172.20.30.1」を設定します。対向機器で設定した「l2tpv3 router-id」コマンドの値と同一にする必要があります。

```
NXR_A(config-l2tpv3-tunnel)#tunnel vendor ietf
```
対向機器のベンダーID として「ietf」を設定します。対向機器で設定した「tunnel vendor」コマンドの値と同一にする必要があります。

＜L2TPv3 Xconnect 設定２＞
```
NXR_A(config)#l2tpv3 xconnect 2
```
L2TPv3 Xconnect「２」を設定します。

```
NXR_A(config-l2tpv3-xconnect)#description NXR_C
```
L2TPv3 Xconnect「２」の名前を「NXR_C」と設定します。

```
NXR_A(config-l2tpv3-xconnect)#tunnel 2
```

使用する L2TPv3 トンネルとして「２」(l2tpv3 tunnel 2) を設定します。

```
NXR_A(config-l2tpv3-xconnect)#xconnect ethernet 0
```

Xconnect インタフェースとして「ethernet 0」を設定します。

※このインタフェースが L2 フレームを送受信するインタフェースとなります。

```
NXR_A(config-l2tpv3-xconnect)#xconnect end-id 1
```

リモート End-ID として「1」を設定します。Xconnect インタフェースを識別する際に使用する ID のため、対向機器の設定した「xconnect end-id」コマンドの値と同一にする必要があります。

```
NXR_A(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
```

TCP MSS の調整機能をオートに設定します。

＜DNS 設定＞

```
NXR_A(config)#dns
NXR_A(dns-config)#service enable
```

DNS に関する設定をします。

[NXR_B の設定]

```
nxr130#configure terminal
nxr130(config)#hostname NXR_B
NXR_B(config)#interface ethernet 0
NXR_B(config-if)#ip address 192.168.10.2/24
NXR_B(config-if)#exit
NXR_B(config)#ip access-list ppp0_in permit 10.10.10.1 10.10.20.1 115
NXR_B(config)#interface ppp 0
NXR_B(config-ppp)#ip address 10.10.20.1/32
NXR_B(config-ppp)#ip masquerade
NXR_B(config-ppp)#ip access-group in ppp0_in
NXR_B(config-ppp)#ip spi-filter
NXR_B(config-ppp)#ip tcp adjust-mss auto
NXR_B(config-ppp)#no ip redirects
NXR_B(config-ppp)#ppp authentication pap
NXR_B(config-ppp)#ppp username test2@centurysys password test2pass
NXR_B(config-ppp)#exit
NXR_B(config)#interface ethernet 1
NXR_B(config-if)#no ip address
NXR_B(config-if)#pppoe-client ppp 0
NXR_B(config-if)#exit
NXR_B(config)#ip route 0.0.0.0/0 ppp 0
NXR_B(config)#l2tpv3 hostname nxrb
NXR_B(config)#l2tpv3 router-id 172.20.20.1
NXR_B(config)#l2tpv3 mac-learning
NXR_B(config)#l2tpv3 mac-aging 300
NXR_B(config)#l2tpv3 path-mtu-discovery
NXR_B(config)#l2tpv3 tunnel 1
NXR_B(config-l2tpv3-tunnel)#description NXR_A
NXR_B(config-l2tpv3-tunnel)#tunnel address 10.10.10.1
NXR_B(config-l2tpv3-tunnel)#tunnel hostname nxra
NXR_B(config-l2tpv3-tunnel)#tunnel router-id 172.20.10.1
NXR_B(config-l2tpv3-tunnel)#tunnel vendor ietf
NXR_B(config-l2tpv3-tunnel)#exit
NXR_B(config)#l2tpv3 xconnect 1
NXR_B(config-l2tpv3-xconnect)#description NXR_A
NXR_B(config-l2tpv3-xconnect)#tunnel 1
NXR_B(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_B(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_B(config-l2tpv3-xconnect)#retry-interval 45
NXR_B(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
NXR_B(config-l2tpv3-xconnect)#exit
NXR_B(config)#dns
NXR_B(dns-config)#service enable
NXR_B(dns-config)#exit
NXR_B(config)#exit
NXR_B#save config
```

【 解説 】

- L2TPv3 の Xconnect インタフェースを「ethernet0」とし、このインタフェースが L2 フレームを送受信するインタフェースとなります。
- L2TPv3 トンネル/セッションが切断されたときに自動再接続できるように、リトライインターバルを設定します。

＜ホスト名の設定＞

```
nxr130(config)#hostname NXR_B
```

ホスト名として「NXR_B」を設定します。

＜Ethernet0 インタフェース設定＞

```
NXR_B(config)#interface ethernet 0
NXR_B(config-if)#ip address 192.168.10.2/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.2/24」を設定します。

＜IPv4 アクセスリスト設定＞

```
NXR_B(config)#ip access-list ppp0_in permit 10.10.10.1 10.10.20.1 115
```

IPv4 アクセスリスト名を「ppp0_in」とし、以下のルールで設定します。

動作	送信元 IPv4 アドレス	宛先 IPv4 アドレス	プロトコル	送信元ポート	宛先ポート
許可	10.10.10.1	10.10.20.1	115(L2TP)		

＜ppp0 インタフェース設定＞

```
NXR_B(config)#interface ppp 0
NXR_B(config-ppp)#ip address 10.10.20.1/32
NXR_B(config-ppp)#ip masquerade
NXR_B(config-ppp)#ip access-group in ppp0_in
NXR_B(config-ppp)#ip spi-filter
NXR_B(config-ppp)#ip tcp adjust-mss auto
NXR_B(config-ppp)#no ip redirects
NXR_B(config-ppp)#ppp authentication pap
NXR_B(config-ppp)#ppp username test2@centurysys password test2pass
```

ppp0 インタフェースに関する設定をします。

IPv4 アドレスが固定のため、「10.10.20.1/32」を設定します。

IPv4 アクセスリスト設定で設定した「ppp0_in」を ppp0 インタフェースの「in」フィルタに適用します。

＜Ethernet1 インタフェース設定＞

```
NXR_B(config)#interface ethernet 1
NXR_B(config-if)#no ip address
NXR_B(config-if)#pppoe-client ppp 0
```

Ethernet1 インタフェースに関する設定をします。

＜スタティックルート設定＞

```
NXR_B(config)#ip route 0.0.0.0/0 ppp 0
```

デフォルトルートを設定します。

＜L2TPv3 設定＞

```
NXR_B(config)#l2tpv3 hostname nxrb
NXR_B(config)#l2tpv3 router-id 172.20.20.1
NXR_B(config)#l2tpv3 mac-learning
NXR_B(config)#l2tpv3 mac-aging 300
NXR_B(config)#l2tpv3 path-mtu-discovery
```

L2TPv3 のホスト名やルータ ID 等を設定します。

＜L2TPv3 トンネル設定＞

```
NXR_B(config)#l2tpv3 tunnel 1
NXR_B(config-l2tpv3-tunnel)#description NXR_A
NXR_B(config-l2tpv3-tunnel)#tunnel address 10.10.10.1
NXR_B(config-l2tpv3-tunnel)#tunnel hostname nxra
NXR_B(config-l2tpv3-tunnel)#tunnel router-id 172.20.10.1
NXR_B(config-l2tpv3-tunnel)#tunnel vendor ietf
```

L2TPv3 トンネル「１」を設定します。

＜L2TPv3 Xconnect 設定＞

```
NXR_B(config)#l2tpv3 xconnect 1
NXR_B(config-l2tpv3-xconnect)#description NXR_A
NXR_B(config-l2tpv3-xconnect)#tunnel 1
NXR_B(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_B(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_B(config-l2tpv3-xconnect)#retry-interval 45
NXR_B(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
```

L2TPv3 Xconnect「１」を設定します。

＜DNS 設定＞

```
NXR_C(config)#dns
NXR_C(dns-config)#service enable
```

DNS に関する設定をします。

[NXR_C の設定]

```
nxr130#configure terminal
nxr130(config)#hostname NXR_C
NXR_C(config)#interface ethernet 0
NXR_C(config-if)#ip address 192.168.10.3/24
NXR_C(config-if)#exit
NXR_C(config)#ip access-list ppp0_in permit 10.10.10.1 any 115
NXR_C(config)#interface ppp 0
NXR_C(config-ppp)#ip address negotiated
NXR_C(config-ppp)#ip masquerade
NXR_C(config-ppp)#ip access-group in ppp0_in
NXR_C(config-ppp)#ip spi-filter
NXR_C(config-ppp)#ip tcp adjust-mss auto
NXR_C(config-ppp)#no ip redirects
NXR_C(config-ppp)#ppp authentication pap
NXR_C(config-ppp)#ppp username test3@centurysys password test3pass
NXR_C(config-ppp)#exit
NXR_C(config)#interface ethernet 1
NXR_C(config-if)#no ip address
NXR_C(config-if)#pppoe-client ppp 0
NXR_C(config-if)#exit
NXR_C(config)#ip route 0.0.0.0/0 ppp 0
NXR_C(config)#l2tpv3 hostname nxrc
NXR_C(config)#l2tpv3 router-id 172.20.30.1
NXR_C(config)#l2tpv3 mac-learning
NXR_C(config)#l2tpv3 mac-aging 300
NXR_C(config)#l2tpv3 path-mtu-discovery
NXR_C(config)#l2tpv3 tunnel 1
NXR_C(config-l2tpv3-tunnel)#description NXR_A
NXR_C(config-l2tpv3-tunnel)#tunnel address 10.10.10.1
NXR_C(config-l2tpv3-tunnel)#tunnel hostname nxra
NXR_C(config-l2tpv3-tunnel)#tunnel router-id 172.20.10.1
NXR_C(config-l2tpv3-tunnel)#tunnel vendor ietf
NXR_C(config-l2tpv3-tunnel)#exit
NXR_C(config)#l2tpv3 xconnect 1
NXR_C(config-l2tpv3-xconnect)#description NXR_A
NXR_C(config-l2tpv3-xconnect)#tunnel 1
NXR_C(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_C(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_C(config-l2tpv3-xconnect)#retry-interval 30
NXR_C(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
NXR_C(config-l2tpv3-xconnect)#exit
NXR_C(config)#dns
NXR_C(dns-config)#service enable
NXR_C(dns-config)#exit
NXR_C(config)#exit
NXR_C#save config
```

【 解説 】

- L2TPv3 の Xconnect インタフェースを「ethernet0」とし、このインタフェースが L2 フレームを送受信するインタフェースとなります。
- L2TPv3 トンネル/セッションが切断されたときに自動再接続できるように、リトライインターバルを設定します。

＜ホスト名の設定＞

```
nxr130(config)#hostname NXR_C
```

ホスト名として「NXR_C」を設定します。

＜Ethernet0 インタフェース設定＞

```
NXR_C(config)#interface ethernet 0
NXR_C(config-if)#ip address 192.168.10.3/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.3/24」を設定します。

＜IPv4 アクセスリスト設定＞

```
NXR_C(config)#ip access-list ppp0_in permit 10.10.10.1 any 115
```

IPv4 アクセスリスト名を「ppp0_in」とし、以下のルールで設定します。

動作	送信元 IPv4 アドレス	宛先 IPv4 アドレス	プロトコル	送信元ポート	宛先ポート
許可	10.10.10.1	any	115(L2TP)		

＜ppp0 インタフェース設定＞

```
NXR_C(config)#interface ppp 0
NXR_C(config-ppp)#ip address negotiated
NXR_C(config-ppp)#ip masquerade
NXR_C(config-ppp)#ip access-group in ppp0_in
NXR_C(config-ppp)#ip spi-filter
NXR_C(config-ppp)#ip tcp adjust-mss auto
NXR_C(config-ppp)#no ip redirects
NXR_C(config-ppp)#ppp authentication pap
NXR_C(config-ppp)#ppp username test3@centurysys password test3pass
```

ppp0 インタフェースに関する設定をします。

IPv4 アクセスリスト設定で設定した「ppp0_in」を ppp0 インタフェースの「in」フィルタに適用します。

＜Ethernet1 インタフェース設定＞

```
NXR_C(config)#interface ethernet 1
NXR_C(config-if)#no ip address
NXR_C(config-if)#pppoe-client ppp 0
```

Ethernet1 インタフェースに関する設定をします。

＜スタティックルート設定＞

```
NXR_C(config)#ip route 0.0.0.0/0 ppp 0
```

デフォルトルートを設定します。

＜L2TPv3 設定＞

```
NXR_C(config)#l2tpv3 hostname nxrc
NXR_C(config)#l2tpv3 router-id 172.20.30.1
NXR_C(config)#l2tpv3 mac-learning
NXR_C(config)#l2tpv3 mac-aging 300
NXR_C(config)#l2tpv3 path-mtu-discovery
```

L2TPv3 のホスト名やルータ ID 等を設定します。

＜L2TPv3 トンネル設定＞

```
NXR_C(config)#l2tpv3 tunnel 1
NXR_C(config-l2tpv3-tunnel)#description NXR_A
NXR_C(config-l2tpv3-tunnel)#tunnel address 10.10.10.1
NXR_C(config-l2tpv3-tunnel)#tunnel hostname nxra
NXR_C(config-l2tpv3-tunnel)#tunnel router-id 172.20.10.1
NXR_C(config-l2tpv3-tunnel)#tunnel vendor ietf
```

L2TPv3 トンネル「１」を設定します。

＜L2TPv3 Xconnect 設定＞

```
NXR_C(config)#l2tpv3 xconnect 1
NXR_C(config-l2tpv3-xconnect)#description NXR_A
NXR_C(config-l2tpv3-xconnect)#tunnel 1
NXR_C(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_C(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_C(config-l2tpv3-xconnect)#retry-interval 30
NXR_C(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
```

L2TPv3 Xconnect「１」を設定します。

＜DNS 設定＞

```
NXR_C(config)#dns
NXR_C(dns-config)#service enable
```

DNS に関する設定をします。

8-2-3. パソコンの設定例

	LAN A のパソコン	LAN B のパソコン	LAN C のパソコン
IPv4 アドレス	192.168.10.100	192.168.10.101	192.168.10.102
サブネットマスク	255.255.255.0	255.255.255.0	255.255.255.0
デフォルトゲートウェイ	192.168.10.1	192.168.10.2	192.168.10.3

8-2-4. 補足 スプリットホライズンの設定例

NXR ではセンター拠点間の通信のみで、拠点間通信を行わない設定をすることも可能です。それを実現するスプリットホライズン機能とは、Xconnect を共有するセッション間において L2TP セッションより受信したフレームを、他の L2TP セッションへ転送を行わない機能です。
この設定例では NXR_A（センタ）でスプリットホライズンを有効にし、拠点間通信を行わないようにします。

[NXR_A の設定]

```
nxr130#configure terminal
nxr130(config)#hostname NXR_A
NXR_A(config)#interface ethernet 0
NXR_A(config-if)#ip address 192.168.10.1/24
NXR_A(config-if)#exit
NXR_A(config)#ip access-list ppp0_in permit any 10.10.10.1 115
NXR_A(config)#interface ppp 0
NXR_A(config-ppp)#ip address 10.10.10.1/32
NXR_A(config-ppp)#ip masquerade
NXR_A(config-ppp)#ip access-group in ppp0_in
NXR_A(config-ppp)#ip spi-filter
NXR_A(config-ppp)#ip tcp adjust-mss auto
NXR_A(config-ppp)#no ip redirects
NXR_A(config-ppp)#ppp authentication pap
NXR_A(config-ppp)#ppp username test1@centurysys password test1pass
NXR_A(config-ppp)#exit
NXR_A(config)#interface ethernet 1
NXR_A(config-if)#no ip address
NXR_A(config-if)#pppoe-client ppp 0
NXR_A(config-if)#exit
NXR_A(config)#ip route 0.0.0.0/0 ppp 0
NXR_A(config)#l2tpv3 hostname nxra
NXR_A(config)#l2tpv3 router-id 172.20.10.1
NXR_A(config)#l2tpv3 mac-learning
NXR_A(config)#l2tpv3 mac-aging 300
NXR_A(config)#l2tpv3 path-mtu-discovery
NXR_A(config)#l2tpv3 tunnel 1
NXR_A(config-l2tpv3-tunnel)#description NXR_B
NXR_A(config-l2tpv3-tunnel)#tunnel address 10.10.20.1
NXR_A(config-l2tpv3-tunnel)#tunnel hostname nxrb
NXR_A(config-l2tpv3-tunnel)#tunnel router-id 172.20.20.1
NXR_A(config-l2tpv3-tunnel)#tunnel vendor ietf
NXR_A(config-l2tpv3-tunnel)#exit
NXR_A(config)#l2tpv3 xconnect 1
NXR_A(config-l2tpv3-xconnect)#description NXR_B
NXR_A(config-l2tpv3-xconnect)#tunnel 1
NXR_A(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_A(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_A(config-l2tpv3-xconnect)#retry-interval 30
NXR_A(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
NXR_A(config-l2tpv3-xconnect)#split-horizon enable
NXR_A(config-l2tpv3-xconnect)#exit
NXR_A(config)#l2tpv3 tunnel 2
NXR_A(config-l2tpv3-tunnel)#description NXR_C
NXR_A(config-l2tpv3-tunnel)#tunnel hostname nxrc
NXR_A(config-l2tpv3-tunnel)#tunnel router-id 172.20.30.1
NXR_A(config-l2tpv3-tunnel)#tunnel vendor ietf
NXR_A(config-l2tpv3-tunnel)#exit

----- 次のページに続きがあります -----
```

```
----- 前のページからの続きです -----

NXR_A(config)#l2tpv3 xconnect 2
NXR_A(config-l2tpv3-xconnect)#description NXR_C
NXR_A(config-l2tpv3-xconnect)#tunnel 2
NXR_A(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_A(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_A(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
NXR_A(config-l2tpv3-xconnect)#split-horizon enable
NXR_A(config-l2tpv3-xconnect)#exit
NXR_A(config)#dns
NXR_A(dns-config)#service enable
NXR_A(dns-config)# exit
NXR_A(config)#exit
NXR_A#save config
```

【 解説 】

- L2TPv3 Xconnect 設定１，２それぞれでスプリットホライズンを有効にします。
- インターネットアクセスを可能にするために、フィルタ設定(SPI),NAT 設定(IP マスカレード),DNS 設定を行っています。

＜L2TPv3 Xconnect 設定１＞

```
NXR_A(config)#l2tpv3 xconnect 1
NXR_A(config-l2tpv3-xconnect)#description NXR_B
NXR_A(config-l2tpv3-xconnect)#tunnel 1
NXR_A(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_A(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_A(config-l2tpv3-xconnect)#retry-interval 30
NXR_A(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
NXR_A(config-l2tpv3-xconnect)#split-horizon enable
```

L2TPv3 Xconnect「１」でスプリットホライズンを有効に設定します。

＜L2TPv3 Xconnect 設定２＞

```
NXR_A(config)#l2tpv3 xconnect 2
NXR_A(config-l2tpv3-xconnect)#description NXR_C
NXR_A(config-l2tpv3-xconnect)#tunnel 2
NXR_A(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_A(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_A(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
NXR_A(config-l2tpv3-xconnect)#split-horizon enable
```

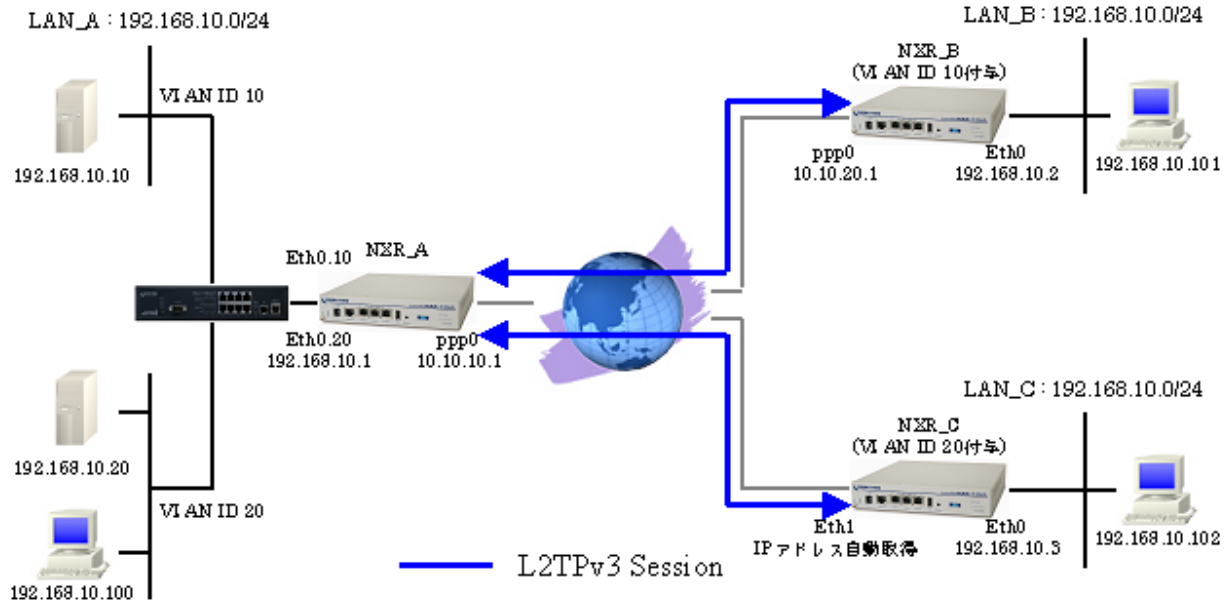
L2TPv3 Xconnect「２」でスプリットホライズンを有効に設定します。

※L2TPv3 Xconnect 設定以外は NXR_A の設定は同一です。

8-3. L2TPv3 での接続設定例（VLAN タグの利用）

NXR では Xconnect インタフェースとして VLAN インタフェースも指定することができます。また NXR では配下に VLAN（802.1Q）に対応していない L2 スイッチがあり、かつ対向ルータで VLAN ID を利用しているような場合に対応するために、NXR では VLAN タグ付与機能を実装しています。これにより L2TP セッションからのパケット送信時に NXR で Ethernet フレームに VLAN タグを付与し、セッションからのパケット受信時に VLAN タグを取り除いて通信することができます。
なおここでは、各拠点からのインターネットアクセスを可能にするために、フィルタ設定（SPI），NAT 設定（IP マスカレード），DNS 設定を行っています。

8-3-1. 構成図



8-3-2. 設定例

[NXR_A の設定]

```
nxr130#configure terminal
nxr130(config)#hostname NXR_A
NXR_A(config)#interface ethernet 0 vid 10
NXR_A(config-vlan)#no ip address
NXR_A(config-vlan)#exit
NXR_A(config)#interface ethernet 0 vid 20
NXR_A(config-vlan)#ip address 192.168.10.1/24
NXR_A(config-vlan)#exit
NXR_A(config)#ip access-list ppp0_in permit any 10.10.10.1 115
NXR_A(config)#interface ppp 0
NXR_A(config-ppp)#ip address 10.10.10.1/32
NXR_A(config-ppp)#ip masquerade
NXR_A(config-ppp)#ip access-group in ppp0_in
NXR_A(config-ppp)#ip spi-filter
NXR_A(config-ppp)#ip tcp adjust-mss auto
NXR_A(config-ppp)#no ip redirects
NXR_A(config-ppp)#ppp authentication pap

----- 次のページに続きがあります -----
```

```
----- 前のページからの続きです -----

NXR_A(config-ppp)#ppp username test1@centurysys password test1pass
NXR_A(config-ppp)#exit
NXR_A(config)#interface ethernet 1
NXR_A(config-if)#no ip address
NXR_A(config-if)#pppoe-client ppp 0
NXR_A(config-if)#exit
NXR_A(config)#ip route 0.0.0.0/0 ppp 0
NXR_A(config)#l2tpv3 hostname nxra
NXR_A(config)#l2tpv3 router-id 172.20.10.1
NXR_A(config)#l2tpv3 mac-learning
NXR_A(config)#l2tpv3 mac-aging 300
NXR_A(config)#l2tpv3 path-mtu-discovery
NXR_A(config)#l2tpv3 tunnel 1
NXR_A(config-l2tpv3-tunnel)#description NXR_B
NXR_A(config-l2tpv3-tunnel)#tunnel address 10.10.20.1
NXR_A(config-l2tpv3-tunnel)#tunnel hostname nxrb
NXR_A(config-l2tpv3-tunnel)#tunnel router-id 172.20.20.1
NXR_A(config-l2tpv3-tunnel)#tunnel vendor ietf
NXR_A(config-l2tpv3-tunnel)#exit
NXR_A(config)#l2tpv3 xconnect 1
NXR_A(config-l2tpv3-xconnect)#description NXR_B
NXR_A(config-l2tpv3-xconnect)#tunnel 1
NXR_A(config-l2tpv3-xconnect)#xconnect ethernet 0 vid 10
NXR_A(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_A(config-l2tpv3-xconnect)#retry-interval 30
NXR_A(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
NXR_A(config-l2tpv3-xconnect)#exit
NXR_A(config)#l2tpv3 tunnel 2
NXR_A(config-l2tpv3-tunnel)#description NXR_C
NXR_A(config-l2tpv3-tunnel)#tunnel hostname nxrc
NXR_A(config-l2tpv3-tunnel)#tunnel router-id 172.20.30.1
NXR_A(config-l2tpv3-tunnel)#tunnel vendor ietf
NXR_A(config-l2tpv3-tunnel)#exit
NXR_A(config)#l2tpv3 xconnect 2
NXR_A(config-l2tpv3-xconnect)#description NXR_C
NXR_A(config-l2tpv3-xconnect)#tunnel 2
NXR_A(config-l2tpv3-xconnect)#xconnect ethernet 0 vid 20
NXR_A(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_A(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
NXR_A(config-l2tpv3-xconnect)#exit
NXR_A(config)#dns
NXR_A(dns-config)#service enable
NXR_A(dns-config)#exit
NXR_A(config)#exit
NXR_A#save config
```

【 解説 】

- Ethernet0 インタフェースに VLAN ID「10」(eth0.10),「20」(eth0.20) のインタフェースを作成します。「eth0.10」には IPv4 アドレスを割り当てず、「eth0.20」には「192.168.10.1」の IPv4 アドレスを割り当てます。
- Xconnect インタフェースとして「eth0.10」,「eth0.20」を設定します。
- L2TPv3 トンネル/セッションが切断されたときに自動再接続できるように、リトライインターバルを設定します。ただし NXR_C に対しては WAN 側 IP アドレスが動的のため、こちらからはネゴシエーションを行わないようにします。

＜ホスト名の設定＞

```
nxr130(config)#hostname NXR_A
```

ホスト名として「NXR_A」を設定します。

＜Ethernet0 インタフェース設定＞

```
NXR_A(config)#interface ethernet 0 vid 10
NXR_A(config-vlan)#no ip address
```

Ethernet0 インタフェースで VLAN ID「10」を設定します。なおこのインタフェースには IP アドレスは割り当てません。

```
NXR_A(config)#interface ethernet 0 vid 20
NXR_A(config-vlan)#ip address 192.168.10.1/24
```

Ethernet0 インタフェースで VLAN ID「20」を設定します。IP アドレスとして「192.168.10.1」を設定します。

＜IPv4 アクセスリスト設定＞

```
NXR_A(config)#ip access-list ppp0_in permit any 10.10.10.1 115
```

IPv4 アクセスリスト名を「ppp0_in」とし、以下のルールで設定します。

動作	送信元 IPv4 アドレス	宛先 IPv4 アドレス	プロトコル	送信元ポート	宛先ポート
許可	any	10.10.10.1	115(L2TP)	-	-

＜ppp0 インタフェース設定＞

```
NXR_A(config)#interface ppp 0
NXR_A(config-ppp)#ip address 10.10.10.1/32
NXR_A(config-ppp)#ip masquerade
NXR_A(config-ppp)#ip access-group in ppp0_in
NXR_A(config-ppp)#ip spi-filter
NXR_A(config-ppp)#ip tcp adjust-mss auto
NXR_A(config-ppp)#no ip redirects
NXR_A(config-ppp)#ppp authentication pap
NXR_A(config-ppp)#ppp username test1@centurysys password test1pass
```

ppp0 インタフェースに関する設定をします。

IPv4 アドレスが固定のため、「10.10.10.1/32」を設定します。

IPv4 アクセスリスト設定で設定した「ppp0_in」を ppp0 インタフェースの「in」フィルタに適用します。

＜Ethernet1 インタフェース設定＞

```
NXR_A(config)#interface ethernet 1
NXR_A(config-if)#no ip address
NXR_A(config-if)#pppoe-client ppp 0
```

Ethernet1 インタフェースに関する設定をします。

＜スタティックルート設定＞

```
NXR_A(config)#ip route 0.0.0.0/0 ppp 0
```

デフォルトルートを設定します。

＜L2TPv3 設定＞

```
NXR_A(config)#l2tpv3 hostname nxra
NXR_A(config)#l2tpv3 router-id 172.20.10.1
NXR_A(config)#l2tpv3 mac-learning
NXR_A(config)#l2tpv3 mac-aging 300
NXR_A(config)#l2tpv3 path-mtu-discovery
```

L2TPv3 のホスト名やルータ ID 等を設定します。

＜L2TPv3 トンネル設定１＞

```
NXR_A(config)#l2tpv3 tunnel 1
NXR_A(config-l2tpv3-tunnel)#description NXR_B
NXR_A(config-l2tpv3-tunnel)#tunnel address 10.10.20.1
NXR_A(config-l2tpv3-tunnel)#tunnel hostname nxrb
NXR_A(config-l2tpv3-tunnel)#tunnel router-id 172.20.20.1
NXR_A(config-l2tpv3-tunnel)#tunnel vendor ietf
```

L2TPv3 トンネル「１」を設定します。

＜L2TPv3 Xconnect 設定１＞

```
NXR_A(config)#l2tpv3 xconnect 1
NXR_A(config-l2tpv3-xconnect)#description NXR_B
NXR_A(config-l2tpv3-xconnect)#tunnel 1
NXR_A(config-l2tpv3-xconnect)#xconnect ethernet 0 vid 10
NXR_A(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_A(config-l2tpv3-xconnect)#retry-interval 30
NXR_A(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
```

L2TPv3 Xconnect「１」を設定します。

Xconnect インタフェースとして「ethernet 0 vid 10」を設定します。

＜L2TPv3 トンネル設定２＞

```
NXR_A(config)#l2tpv3 tunnel 2
NXR_A(config-l2tpv3-tunnel)#description NXR_C
NXR_A(config-l2tpv3-tunnel)#tunnel hostname nxrc
NXR_A(config-l2tpv3-tunnel)#tunnel router-id 172.20.30.1
NXR_A(config-l2tpv3-tunnel)#tunnel vendor ietf
```

L2TPv3 トンネル「２」を設定します。

＜L2TPv3 Xconnect 設定２＞

```
NXR_A(config)#l2tpv3 xconnect 2
NXR_A(config-l2tpv3-xconnect)#description NXR_C
NXR_A(config-l2tpv3-xconnect)#tunnel 2
NXR_A(config-l2tpv3-xconnect)#xconnect ethernet 0 vid 20
NXR_A(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_A(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
```

L2TPv3 Xconnect「２」を設定します。

Xconnect インタフェースとして「ethernet 0 vid 20」を設定します。

＜DNS 設定＞

```
NXR_A(config)#dns
NXR_A(dns-config)#service enable
```

DNS に関する設定をします。

[NXR_B の設定]

```
nxr130#configure terminal
nxr130(config)#hostname NXR_B
NXR_B(config)#interface ethernet 0
NXR_B(config-if)#ip address 192.168.10.2/24
NXR_B(config-if)#exit
NXR_B(config)#ip access-list ppp0_in permit 10.10.10.1 10.10.20.1 115
NXR_B(config)#interface ppp 0
NXR_B(config-ppp)#ip address 10.10.20.1/32
NXR_B(config-ppp)#ip masquerade
NXR_B(config-ppp)#ip access-group in ppp0_in
NXR_B(config-ppp)#ip spi-filter
NXR_B(config-ppp)#ip tcp adjust-mss auto
NXR_B(config-ppp)#no ip redirects
NXR_B(config-ppp)#ppp authentication pap
NXR_B(config-ppp)#ppp username test2@centurysys password test2pass
NXR_B(config-ppp)#exit
NXR_B(config)#interface ethernet 1
NXR_B(config-if)#no ip address
NXR_B(config-if)#pppoe-client ppp 0
NXR_B(config-if)#exit
NXR_B(config)#ip route 0.0.0.0/0 ppp 0
NXR_B(config)#l2tpv3 hostname nxrb
NXR_B(config)#l2tpv3 router-id 172.20.20.1
NXR_B(config)#l2tpv3 mac-learning
NXR_B(config)#l2tpv3 mac-aging 300
NXR_B(config)#l2tpv3 path-mtu-discovery
NXR_B(config)#l2tpv3 tunnel 1
NXR_B(config-l2tpv3-tunnel)#description NXR_A
NXR_B(config-l2tpv3-tunnel)#tunnel address 10.10.10.1
NXR_B(config-l2tpv3-tunnel)#tunnel hostname nxra
NXR_B(config-l2tpv3-tunnel)#tunnel router-id 172.20.10.1
NXR_B(config-l2tpv3-tunnel)#tunnel vendor ietf
NXR_B(config-l2tpv3-tunnel)#exit
NXR_B(config)#l2tpv3 xconnect 1
NXR_B(config-l2tpv3-xconnect)#description NXR_A
NXR_B(config-l2tpv3-xconnect)#tunnel 1
NXR_B(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_B(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_B(config-l2tpv3-xconnect)#retry-interval 45
NXR_B(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
NXR_B(config-l2tpv3-xconnect)#vlan-id 10
NXR_B(config-l2tpv3-xconnect)#exit
NXR_B(config)#dns
NXR_B(dns-config)#service enable
NXR_B(dns-config)#exit
NXR_B(config)#exit
NXR_B#save config
```

【 解説 】

- Xconnect インタフェース設定で VLAN ID「10」を付与する設定をします。これにより L2TP セッションにデータを送信する場合は NXR で 802.1Q ヘッダを付与し、L2TP セッションよりデータ受信し LAN 側に送信する場合は、802.1Q ヘッダを取り除いて通信を行います。

＜ホスト名の設定＞

```
nxr130(config)#hostname NXR_B
```

ホスト名として「NXR_B」を設定します。

＜Ethernet0 インタフェース設定＞

```
NXR_B(config)#interface ethernet 0
NXR_B(config-if)#ip address 192.168.10.2/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.2/24」を設定します。

＜IPv4 アクセスリスト設定＞

```
NXR_B(config)#ip access-list ppp0_in permit 10.10.10.1 10.10.20.1 115
```

IPv4 アクセスリスト名を「ppp0_in」とし、以下のルールで設定します。

動作	送信元 IPv4 アドレス	宛先 IPv4 アドレス	プロトコル	送信元ポート	宛先ポート
許可	10.10.10.1	10.10.20.1	115(L2TP)	-	-

＜ppp0 インタフェース設定＞

```
NXR_B(config)#interface ppp 0
NXR_B(config-ppp)#ip address 10.10.20.1/32
NXR_B(config-ppp)#ip masquerade
NXR_B(config-ppp)#ip access-group in ppp0_in
NXR_B(config-ppp)#ip spi-filter
NXR_B(config-ppp)#ip tcp adjust-mss auto
NXR_B(config-ppp)#no ip redirects
NXR_B(config-ppp)#ppp authentication pap
NXR_B(config-ppp)#ppp username test2@centurysys password test2pass
```

ppp0 インタフェースに関する設定をします。

IPv4 アドレスが固定のため、「10.10.20.1/32」を設定します。

IPv4 アクセスリスト設定で設定した「ppp0_in」を ppp0 インタフェースの「in」フィルタに適用します。

＜Ethernet1 インタフェース設定＞

```
NXR_B(config)#interface ethernet 1
NXR_B(config-if)#no ip address
NXR_B(config-if)#pppoe-client ppp 0
```

Ethernet1 インタフェースに関する設定をします。

＜スタティックルート設定＞

```
NXR_B(config)#ip route 0.0.0.0/0 ppp 0
```

デフォルトルートを設定します。

＜L2TPv3 設定＞

```
NXR_B(config)#l2tpv3 hostname nxrb
NXR_B(config)#l2tpv3 router-id 172.20.20.1
NXR_B(config)#l2tpv3 mac-learning
NXR_B(config)#l2tpv3 mac-aging 300
NXR_B(config)#l2tpv3 path-mtu-discovery
```

L2TPv3 のホスト名やルータ ID 等を設定します。

＜L2TPv3 トンネル設定＞

```
NXR_B(config)#l2tpv3 tunnel 1
NXR_B(config-l2tpv3-tunnel)#description NXR_A
NXR_B(config-l2tpv3-tunnel)#tunnel address 10.10.10.1
NXR_B(config-l2tpv3-tunnel)#tunnel hostname nxra
NXR_B(config-l2tpv3-tunnel)#tunnel router-id 172.20.10.1
NXR_B(config-l2tpv3-tunnel)#tunnel vendor ietf
```

L2TPv3 トンネル「１」を設定します。

＜L2TPv3 Xconnect 設定＞

```
NXR_B(config)#l2tpv3 xconnect 1
NXR_B(config-l2tpv3-xconnect)#description NXR_A
NXR_B(config-l2tpv3-xconnect)#tunnel 1
NXR_B(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_B(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_B(config-l2tpv3-xconnect)#retry-interval 45
NXR_B(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
```

L2TPv3 Xconnect「１」を設定します。

```
NXR_B(config-l2tpv3-xconnect)#vlan-id 10
```

Xconnect インタフェースで VLAN ID「10」を付与する設定をします。

＜DNS 設定＞

```
NXR_B(config)#dns
NXR_B(dns-config)#service enable
```

DNS に関する設定をします。

[NXR_C の設定]

```
nxr130#configure terminal
nxr130(config)#hostname NXR_C
NXR_C(config)#interface ethernet 0
NXR_C(config-if)#ip address 192.168.10.3/24
NXR_C(config-if)#exit
NXR_C(config)#ip access-list eth1_in permit 10.10.10.1 any 115
NXR_C(config)#interface ethernet 1
NXR_C(config-if)#ip address dhcp
NXR_C(config-if)#ip masquerade
NXR_C(config-if)#ip access-group in eth1_in
NXR_C(config-if)#ip spi-filter
NXR_C(config-if)#no ip redirects
NXR_C(config-if)#exit
NXR_C(config)#l2tpv3 hostname nxrc
NXR_C(config)#l2tpv3 router-id 172.20.30.1
NXR_C(config)#l2tpv3 mac-learning
NXR_C(config)#l2tpv3 mac-aging 300
NXR_C(config)#l2tpv3 path-mtu-discovery
NXR_C(config)#l2tpv3 tunnel 1
NXR_C(config-l2tpv3-tunnel)#description NXR_A
NXR_C(config-l2tpv3-tunnel)#tunnel address 10.10.10.1
NXR_C(config-l2tpv3-tunnel)#tunnel hostname nxra
NXR_C(config-l2tpv3-tunnel)#tunnel router-id 172.20.10.1
NXR_C(config-l2tpv3-tunnel)#tunnel vendor ietf
NXR_C(config-l2tpv3-tunnel)#exit
NXR_C(config)#l2tpv3 xconnect 1
NXR_C(config-l2tpv3-xconnect)#description NXR_A
NXR_C(config-l2tpv3-xconnect)#tunnel 1
NXR_C(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_C(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_C(config-l2tpv3-xconnect)#retry-interval 30
NXR_C(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
NXR_C(config-l2tpv3-xconnect)#vlan-id 20
NXR_C(config-l2tpv3-xconnect)#exit
NXR_C(config)#dns
NXR_C(dns-config)#service enable
NXR_C(dns-config)#exit
NXR_C(config)#exit
NXR_C#save config
```

【 解説 】

- Xconnect インタフェース設定で VLAN ID「20」を付与する設定をします。これにより L2TP セッションにデータを送信する場合は NXR で 802.1Q ヘッダを付与し、L2TP セッションよりデータ受信し LAN 側に送信する場合は、802.1Q ヘッダを取り除いて通信を行います。

＜ホスト名の設定＞

```
nxr130(config)#hostname NXR_C
```

ホスト名として「NXR_C」を設定します。

＜Ethernet0 インタフェース設定＞

```
NXR_C(config)#interface ethernet 0
NXR_C(config-if)#ip address 192.168.10.3/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.3/24」を設定します。

＜IPv4 アクセスリスト設定＞

```
NXR_C(config)#ip access-list eth1_in permit 10.10.10.1 any 115
```

IPv4 アクセスリスト名を「eth1_in」とし、以下のルールで設定します。

動作	送信元 IPv4 アドレス	宛先 IPv4 アドレス	プロトコル	送信元ポート	宛先ポート
許可	10.10.10.1	any	115(L2TP)	-	-

＜Ethernet1 インタフェース設定＞

```
NXR_C(config)#interface ethernet 1
NXR_C(config-if)#ip address dhcp
NXR_C(config-if)#ip masquerade
NXR_C(config-if)#ip access-group in eth1_in
NXR_C(config-if)#ip spi-filter
NXR_C(config-if)#no ip redirects
```

Ethernet1 インタフェースに関する設定をします。

IPv4 アドレスが動的のため、DHCP クライアントに設定します。

IPv4 アクセスリスト設定で設定した「eth1_in」を Ethernet1 インタフェースの「in」フィルタに適用します。

＜L2TPv3 設定＞

```
NXR_C(config)#l2tpv3 hostname nxrc
NXR_C(config)#l2tpv3 router-id 172.20.30.1
NXR_C(config)#l2tpv3 mac-learning
NXR_C(config)#l2tpv3 mac-aging 300
NXR_C(config)#l2tpv3 path-mtu-discovery
```

L2TPv3 のホスト名やルータ ID 等を設定します。

＜L2TPv3 トンネル設定＞

```
NXR_C(config)#l2tpv3 tunnel 1
NXR_C(config-l2tpv3-tunnel)#description NXR_A
NXR_C(config-l2tpv3-tunnel)#tunnel address 10.10.10.1
NXR_C(config-l2tpv3-tunnel)#tunnel hostname nxra
NXR_C(config-l2tpv3-tunnel)#tunnel router-id 172.20.10.1
NXR_C(config-l2tpv3-tunnel)#tunnel vendor ietf
```

L2TPv3 トンネル「１」を設定します。

＜L2TPv3 Xconnect 設定＞

```
NXR_C(config)#l2tpv3 xconnect 1
NXR_C(config-l2tpv3-xconnect)#description NXR_A
NXR_C(config-l2tpv3-xconnect)#tunnel 1
NXR_C(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_C(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_C(config-l2tpv3-xconnect)#retry-interval 30
NXR_C(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
```

L2TPv3 Xconnect「１」を設定します。

```
NXR_C(config-l2tpv3-xconnect)#vlan-id 20
```

Xconnect インタフェースで VLAN ID「20」を付与する設定をします。

＜DNS 設定＞

```
NXR_C(config)#dns
NXR_C(dns-config)#service enable
```

DNS に関する設定をします。

8-3-3. パソコンの設定例

LAN A	VLAN ID 10 の サーバ	VLAN ID 20 の サーバ	VLAN ID 20 の パソコン
IPv4 アドレス	192.168.10.10	192.168.10.20	192.168.10.100
サブネットマスク	255.255.255.0	255.255.255.0	255.255.255.0
デフォルトゲートウェイ		192.168.10.1	192.168.10.1

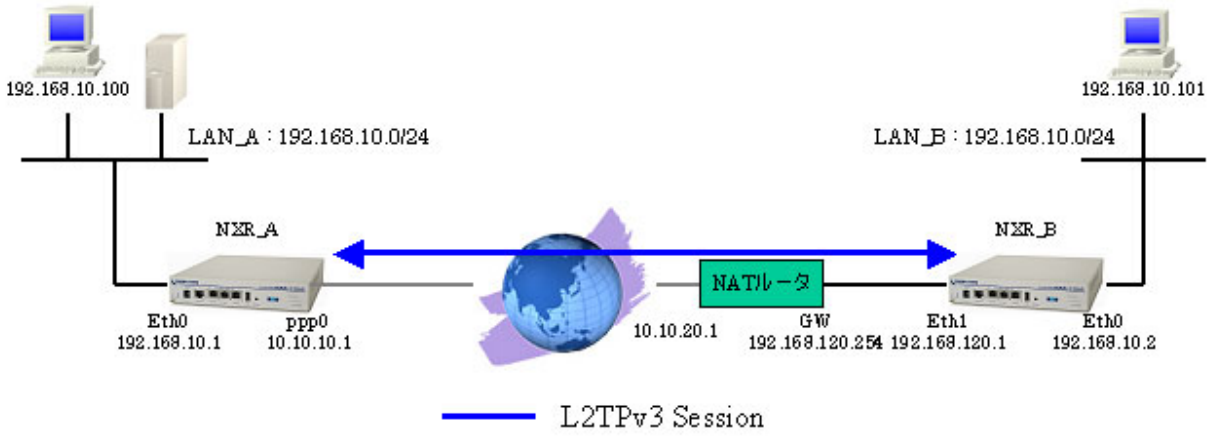
	LAN B のパソコン	LAN C のパソコン
IPv4 アドレス	192.168.10.101	192.168.10.102
サブネットマスク	255.255.255.0	255.255.255.0
デフォルトゲートウェイ	192.168.10.2	192.168.10.3

8-4. L2TPv3 over UDP 設定例

NXR では L2TP パケットを UDP でカプセル化する L2TPv3 over UDP 機能を利用することが可能です。これにより NAT を経由する構成や L2TP パケットを通過させられない環境でも、UDP が許可されていれば L2TPv3 を利用できるようになります。

※ここでは NAT ルータで UDP ポート「1701」に関して NAT 可能で、かつフィルタが許可されていることを前提としています。

8-4-1. 構成図



8-4-2. 設定例

[NXR_A の設定]

```
nxr130#configure terminal
nxr130(config)#hostname NXR_A
NXR_A(config)#interface ethernet 0
NXR_A(config-if)#ip address 192.168.10.1/24
NXR_A(config-if)#exit
NXR_A(config)#ip access-list ppp0_in permit 10.10.20.1 10.10.10.1 udp 1701 1701
NXR_A(config)#interface ppp 0
NXR_A(config-ppp)#ip address 10.10.10.1/32
NXR_A(config-ppp)#ip masquerade
NXR_A(config-ppp)#ip access-group in ppp0_in
NXR_A(config-ppp)#ip spi-filter
NXR_A(config-ppp)#ip tcp adjust-mss auto
NXR_A(config-ppp)#no ip redirects
NXR_A(config-ppp)#ppp authentication pap
NXR_A(config-ppp)#ppp username test1@centurysys password test1pass
NXR_A(config-ppp)#exit
NXR_A(config)#interface ethernet 1
NXR_A(config-if)#no ip address
NXR_A(config-if)#pppoe-client ppp 0
NXR_A(config-if)#exit

----- 次のページに続きがあります -----
```

```
----- 前のページからの続きです -----

NXR_A(config)#ip route 0.0.0.0/0 ppp 0
NXR_A(config)#l2tpv3 hostname nxra
NXR_A(config)#l2tpv3 router-id 172.20.10.1
NXR_A(config)#l2tpv3 mac-learning
NXR_A(config)#l2tpv3 mac-aging 300
NXR_A(config)#l2tpv3 udp source-port 1701
NXR_A(config)#l2tpv3 udp path-mtu-discovery
NXR_A(config)#l2tpv3 tunnel 1
NXR_A(config-l2tpv3-tunnel)#description NXR_B
NXR_A(config-l2tpv3-tunnel)#tunnel hostname nxrb
NXR_A(config-l2tpv3-tunnel)#tunnel address 10.10.20.1
NXR_A(config-l2tpv3-tunnel)#tunnel router-id 172.20.20.1
NXR_A(config-l2tpv3-tunnel)#tunnel vendor ietf
NXR_A(config-l2tpv3-tunnel)#tunnel protocol udp
NXR_A(config-l2tpv3-tunnel)#tunnel udp port 1701
NXR_A(config-l2tpv3-tunnel)#exit
NXR_A(config)#l2tpv3 xconnect 1
NXR_A(config-l2tpv3-xconnect)#description NXR_B
NXR_A(config-l2tpv3-xconnect)#tunnel 1
NXR_A(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_A(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_A(config-l2tpv3-xconnect)#retry-interval 30
NXR_A(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
NXR_A(config-l2tpv3-xconnect)#exit
NXR_A(config)#dns
NXR_A(dns-config)#service enable
NXR_A(dns-config)#exit
NXR_A(config)#exit
NXR_A#save config
```

【 解説 】

- L2TPv3 over UDP を利用するため、L2TPv3 トンネル設定にて送信プロトコルとして「UDP」を選択します。それにともない L2TPv3 設定，L2TPv3 トンネル設定で送信元，宛先のポート番号を設定します。ここでは送信元，宛先ともに「1701」を設定します。
- L2TPv3 トンネル/セッションが切断されたときに自動再接続できるように、リトライインターバルを設定します。
- インターネットアクセスを可能にするために、フィルタ設定(SPI)，NAT 設定(IP マスカレード)，DNS 設定を行っています。

＜ホスト名の設定＞

```
nxr130(config)#hostname NXR_A
```

ホスト名として「NXR_A」を設定します。

＜Ethernet0 インタフェース設定＞

```
NXR_A(config)#interface ethernet 0
NXR_A(config-if)#ip address 192.168.10.1/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.1/24」を設定します。

＜IPv4 アクセスリスト設定＞

```
NXR_A(config)#ip access-list ppp0_in permit 10.10.20.1 10.10.10.1 udp 1701 1701
```

IPv4 アクセスリスト名を「ppp0_in」とし、以下のルールで設定します。

動作	送信元 IPv4 アドレス	宛先 IPv4 アドレス	プロトコル	送信元ポート	宛先ポート
許可	10.10.20.1	10.10.10.1	UDP	1701	1701

＜ppp0 インタフェース設定＞

```
NXR_A(config)#interface ppp 0
NXR_A(config-ppp)#ip address 10.10.10.1/32
NXR_A(config-ppp)#ip masquerade
NXR_A(config-ppp)#ip access-group in ppp0_in
NXR_A(config-ppp)#ip spi-filter
NXR_A(config-ppp)#ip tcp adjust-mss auto
NXR_A(config-ppp)#no ip redirects
NXR_A(config-ppp)#ppp authentication pap
NXR_A(config-ppp)#ppp username test1@centurysys password test1pass
```

ppp0 インタフェースに関する設定をします。

IPv4 アドレスが固定のため、「10.10.10.1/32」を設定します。

IPv4 アクセスリスト設定で設定した「ppp0_in」を ppp0 インタフェースの「in」フィルタに適用します。

＜Ethernet1 インタフェース設定＞

```
NXR_A(config)#interface ethernet 1
NXR_A(config-if)#no ip address
NXR_A(config-if)#pppoe-client ppp 0
```

Ethernet1 インタフェースに関する設定をします。

＜スタティックルート設定＞

```
NXR_A(config)#ip route 0.0.0.0/0 ppp 0
```

デフォルトルートを設定します。

＜L2TPv3 設定＞

```
NXR_A(config)#l2tpv3 hostname nxra
NXR_A(config)#l2tpv3 router-id 172.20.10.1
NXR_A(config)#l2tpv3 mac-learning
NXR_A(config)#l2tpv3 mac-aging 300
```

L2TPv3 のホスト名やルータ ID 等を設定します。

```
NXR_A(config)#l2tpv3 udp source-port 1701
```

L2TPv3 over UDP 利用時の送信元ポート番号「1701」を設定します。

```
NXR_A(config)#l2tpv3 udp path-mtu-discovery
```

L2TPv3 over UDP 利用時の Path MTU Discovery (PMTUD over UDP) を有効にします。

＜L2TPv3 トンネル設定＞

```
NXR_A(config)#l2tpv3 tunnel 1
NXR_A(config-l2tpv3-tunnel)#description NXR_B
NXR_A(config-l2tpv3-tunnel)#tunnel hostname nxrb
NXR_A(config-l2tpv3-tunnel)#tunnel address 10.10.20.1
NXR_A(config-l2tpv3-tunnel)#tunnel router-id 172.20.20.1
NXR_A(config-l2tpv3-tunnel)#tunnel vendor ietf
```

L2TPv3 トンネル「１」を設定します。

```
NXR_A(config-l2tpv3-tunnel)#tunnel protocol udp
```

L2TPv3 over UDP を利用するため、送信時のプロトコルとして「UDP」を設定します。

```
NXR_A(config-l2tpv3-tunnel)#tunnel udp port 1701
```

L2TPv3 over UDP 利用時の宛先ポート番号「1701」を設定します。

＜L2TPv3 Xconnect 設定＞

```
NXR_A(config)#l2tpv3 xconnect 1
NXR_A(config-l2tpv3-xconnect)#description NXR_B
NXR_A(config-l2tpv3-xconnect)#tunnel 1
NXR_A(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_A(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_A(config-l2tpv3-xconnect)#retry-interval 30
NXR_A(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
```

L2TPv3 Xconnect「１」を設定します。

＜DNS 設定＞

```
NXR_A(config)#dns
NXR_A(dns-config)#service enable
```

DNS に関する設定をします。

[NXR_B の設定]

```
nxr130#configure terminal
nxr130(config)#hostname NXR_B
NXR_B(config)#interface ethernet 0
NXR_B(config-if)#ip address 192.168.10.2/24
NXR_B(config-if)#exit
NXR_B(config)#interface ethernet 1
NXR_B(config-if)#ip address 192.168.120.1/24
NXR_B(config-if)#no ip redirects
NXR_B(config-if)#exit
NXR_B(config)#ip route 0.0.0.0/0 192.168.120.254
NXR_B(config)#l2tpv3 hostname nxrb
NXR_B(config)#l2tpv3 router-id 172.20.20.1
NXR_B(config)#l2tpv3 mac-learning
NXR_B(config)#l2tpv3 mac-aging 300
NXR_B(config)#l2tpv3 udp source-port 1701
NXR_B(config)#l2tpv3 udp path-mtu-discovery
NXR_B(config)#l2tpv3 tunnel 1
NXR_B(config-l2tpv3-tunnel)#description NXR_A
NXR_B(config-l2tpv3-tunnel)#tunnel hostname nxra
NXR_B(config-l2tpv3-tunnel)#tunnel address 10.10.10.1
NXR_B(config-l2tpv3-tunnel)#tunnel router-id 172.20.10.1
NXR_B(config-l2tpv3-tunnel)#tunnel vendor ietf
NXR_B(config-l2tpv3-tunnel)#tunnel protocol udp
NXR_B(config-l2tpv3-tunnel)#tunnel udp port 1701
NXR_B(config-l2tpv3-tunnel)#exit
NXR_B(config)#l2tpv3 xconnect 1
NXR_B(config-l2tpv3-xconnect)#description NXR_A
NXR_B(config-l2tpv3-xconnect)#tunnel 1
NXR_B(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_B(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_B(config-l2tpv3-xconnect)#retry-interval 45
NXR_B(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
NXR_B(config-l2tpv3-xconnect)#exit
NXR_B(config)#dns
NXR_B(dns-config)#service enable
NXR_B(dns-config)#exit
NXR_B(config)#exit
NXR_B#save config
```

【 解説 】

- L2TPv3 over UDP を利用するため、L2TPv3 トンネル設定にて送信プロトコルとして「UDP」を選択します。それにともない L2TPv3 設定，L2TPv3 トンネル設定で送信元，宛先のポート番号を設定します。ここでは送信元，宛先ともに「1701」を設定します。
- L2TPv3 トンネル/セッションが切断されたときに自動再接続できるように、リトライインターバルを設定します。

＜ホスト名の設定＞

```
nxr130(config)#hostname NXR_B
```

ホスト名として「NXR_B」を設定します。

＜Ethernet0 インタフェース設定＞

```
NXR_B(config)#interface ethernet 0
NXR_B(config-if)#ip address 192.168.10.2/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.2/24」を設定します。

＜Ethernet1 インタフェース設定＞

```
NXR_B(config)#interface ethernet 1
NXR_B(config-if)#ip address 192.168.120.1/24
NXR_B(config-if)#no ip redirects
```

Ethernet1 インタフェースの IPv4 アドレスとして「192.168.120.1/24」を設定します。

＜スタティックルート設定＞

```
NXR_B(config)#ip route 0.0.0.0/0 192.168.120.254
```

デフォルトルートを設定します。

＜L2TPv3 設定＞

```
NXR_B(config)#l2tpv3 hostname nxrb
NXR_B(config)#l2tpv3 router-id 172.20.20.1
NXR_B(config)#l2tpv3 mac-learning
NXR_B(config)#l2tpv3 mac-aging 300
NXR_B(config)#l2tpv3 udp source-port 1701
NXR_B(config)#l2tpv3 udp path-mtu-discovery
```

L2TPv3 のホスト名やルータ ID 等を設定します。

L2TPv3 over UDP 利用時の送信元ポート番号「1701」を設定します。

L2TPv3 over UDP 利用時の Path MTU Discovery (PMTUD over UDP) を有効にします。

＜L2TPv3 トンネル設定＞

```
NXR_B(config)#l2tpv3 tunnel 1
NXR_B(config-l2tpv3-tunnel)#description NXR_A
NXR_B(config-l2tpv3-tunnel)#tunnel hostname nxra
NXR_B(config-l2tpv3-tunnel)#tunnel address 10.10.10.1
NXR_B(config-l2tpv3-tunnel)#tunnel router-id 172.20.10.1
NXR_B(config-l2tpv3-tunnel)#tunnel vendor ietf
NXR_B(config-l2tpv3-tunnel)#tunnel protocol udp
NXR_B(config-l2tpv3-tunnel)#tunnel udp port 1701
```

L2TPv3 トンネル「１」を設定します。

L2TPv3 over UDP を利用するため、送信時のプロトコルとして「UDP」を設定します。

L2TPv3 over UDP 利用時の宛先ポート番号「1701」を設定します。

＜L2TPv3 Xconnect 設定＞

```
NXR_B(config)#l2tpv3 xconnect 1
NXR_B(config-l2tpv3-xconnect)#description NXR_A
NXR_B(config-l2tpv3-xconnect)#tunnel 1
NXR_B(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_B(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_B(config-l2tpv3-xconnect)#retry-interval 45
NXR_B(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
```

L2TPv3 Xconnect「1」を設定します。

＜DNS 設定＞

```
NXR_B(config)#dns
NXR_B(dns-config)#service enable
```

DNS に関する設定をします。

8-4-3. パソコンの設定例

	LAN A のパソコン	LAN B のパソコン
IPv4 アドレス	192.168.10.100	192.168.10.101
サブネットマスク	255.255.255.0	255.255.255.0
デフォルトゲートウェイ	192.168.10.1	192.168.10.2

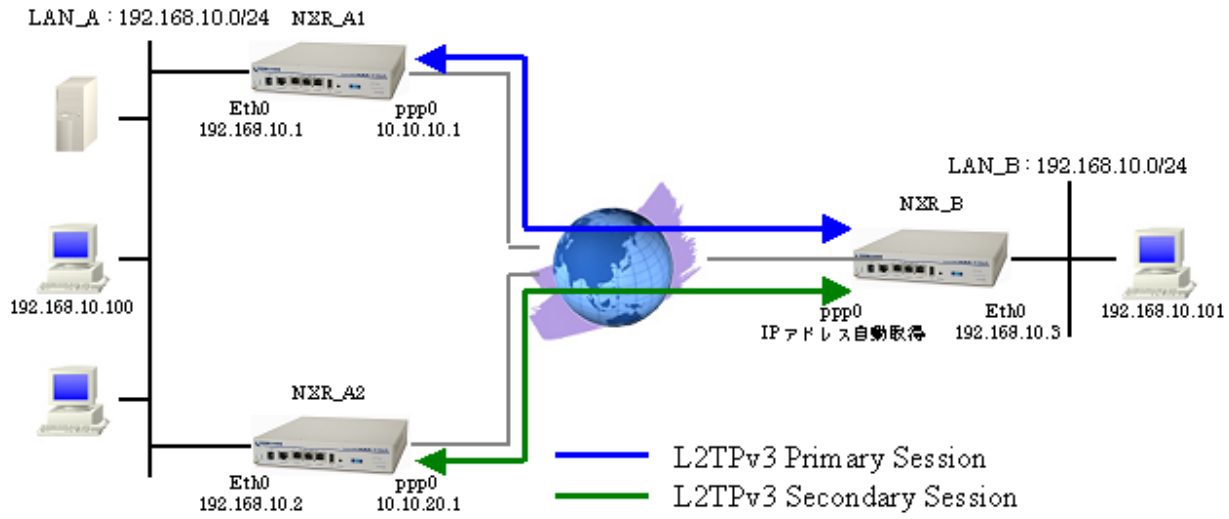
8-5. L2TPv3 Group 機能二重化設定例

センタでの WAN 側回線障害や機器障害に備えて、NXR では L2TPv3 Group 機能を搭載しています。この機能によりプライマリセッションに障害が発生した場合、セカンダリセッションを利用して通信経路を確保します。

ここでは、セカンダリセッションがアクティブセッションとなっている状態でプライマリセッションが確立した場合、セカンダリセッションがアクティブな状態を維持し続けるように設定します。

※セカンダリセッションがアクティブセッションとなっている状態でプライマリセッションが確立した場合に、プライマリセッションをアクティブセッションとする設定に関しましては、「補足 Preempt の設定例」をご参照下さい。

8-5-1. 構成図



8-2-2. 設定例

[NXR_A1 の設定]

```
nxr130#configure terminal
nxr130(config)#hostname NXR_A1
NXR_A1(config)#interface ethernet 0
NXR_A1(config-if)#ip address 192.168.10.1/24
NXR_A1(config-if)#exit
NXR_A1(config)#ip access-list ppp0_in permit any 10.10.10.1 115
NXR_A1(config)#interface ppp 0
NXR_A1(config-ppp)#ip address 10.10.10.1/32
NXR_A1(config-ppp)#ip masquerade
NXR_A1(config-ppp)#ip access-group in ppp0_in
NXR_A1(config-ppp)#ip spi-filter
NXR_A1(config-ppp)#ip tcp adjust-mss auto
NXR_A1(config-ppp)#no ip redirects
NXR_A1(config-ppp)#ppp authentication pap
NXR_A1(config-ppp)#ppp username test1@centurysys password test1pass
NXR_A1(config-ppp)#exit
NXR_A1(config)#interface ethernet 1

----- 次のページに続きがあります -----
```

```
----- 前のページからの続きです -----

NXR_A1(config-if)#no ip address
NXR_A1(config-if)#pppoe-client ppp 0
NXR_A1(config-if)#exit
NXR_A1(config)#ip route 0.0.0.0/0 ppp 0
NXR_A1(config)#l2tpv3 hostname nxra1
NXR_A1(config)#l2tpv3 router-id 172.20.10.1
NXR_A1(config)#l2tpv3 mac-learning
NXR_A1(config)#l2tpv3 mac-aging 300
NXR_A1(config)#l2tpv3 path-mtu-discovery
NXR_A1(config)#no l2tpv3 loop-detect
NXR_A1(config)#l2tpv3 send-known-unicast
NXR_A1(config)#l2tpv3 tunnel 1
NXR_A1(config-l2tpv3-tunnel)#description NXR_B
NXR_A1(config-l2tpv3-tunnel)#tunnel hostname nxrb
NXR_A1(config-l2tpv3-tunnel)#tunnel router-id 172.20.30.1
NXR_A1(config-l2tpv3-tunnel)#tunnel vendor ietf
NXR_A1(config-l2tpv3-tunnel)#exit
NXR_A1(config)#l2tpv3 xconnect 1
NXR_A1(config-l2tpv3-xconnect)#description NXR_B
NXR_A1(config-l2tpv3-xconnect)#tunnel 1
NXR_A1(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_A1(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_A1(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
NXR_A1(config-l2tpv3-xconnect)#send-known-unicast enable
NXR_A1(config-l2tpv3-xconnect)#exit
NXR_A1(config)#dns
NXR_A1(dns-config)#service enable
NXR_A1(dns-config)#exit
NXR_A1(config)#exit
NXR_A1#save config
```

【 解説 】

- Known Unicast 送信機能を「有効」に設定し、Xconnect インタフェースより受信したユニキャストフレームの宛先 MAC アドレスが L2TPv3 ローカル MAC テーブルに存在する場合でも、セッション側へフレームを転送します。
  LAN_A 側のルータでは、Xconnect インタフェース側で LAN_B から送信されたフレームを受信してしまう場合があるため、この設定をしない場合は L2TPv3 ローカル MAC テーブルに登録された該当 MAC アドレスが削除される（エージングタイマが切れる）まで通信できない場合があります。
- 対向ルータの NXR_B は WAN 側 IP アドレスが動的のため、こちらからはネゴシエーションを行わないようにしています。

＜ホスト名の設定＞

```
nxr130(config)#hostname NXR_A1
```

ホスト名として「NXR_A1」を設定します。

＜Ethernet0 インタフェース設定＞

```
NXR_A1(config)#interface ethernet 0
NXR_A1(config-if)#ip address 192.168.10.1/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.1/24」を設定します。

＜IPv4 アクセスリスト設定＞

```
NXR_A1(config)#ip access-list ppp0_in permit any 10.10.10.1 115
```

IPv4 アクセスリスト名を「ppp0_in」とし、以下のルールで設定します。

動作	送信元 IPv4 アドレス	宛先 IPv4 アドレス	プロトコル	送信元ポート	宛先ポート
許可	any	10.10.10.1	115(L2TP)	-	-

＜ppp0 インタフェース設定＞

```
NXR_A1(config)#interface ppp 0
NXR_A1(config-ppp)#ip address 10.10.10.1/32
NXR_A1(config-ppp)#ip masquerade
NXR_A1(config-ppp)#ip access-group in ppp0_in
NXR_A1(config-ppp)#ip spi-filter
NXR_A1(config-ppp)#ip tcp adjust-mss auto
NXR_A1(config-ppp)#no ip redirects
NXR_A1(config-ppp)#ppp authentication pap
NXR_A1(config-ppp)#ppp username test1@centurysys password test1pass
```

ppp0 インタフェースに関する設定をします。

IPv4 アドレスが固定のため、「10.10.10.1/32」を設定します。

IPv4 アクセスリスト設定で設定した「ppp0_in」を ppp0 インタフェースの「in」フィルタに適用します。

＜Ethernet1 インタフェース設定＞

```
NXR_A1(config)#interface ethernet 1
NXR_A1(config-if)#no ip address
NXR_A1(config-if)#pppoe-client ppp 0
```

Ethernet1 インタフェースに関する設定をします。

＜スタティックルート設定＞

```
NXR_A1(config)#ip route 0.0.0.0/0 ppp 0
```

デフォルトルートを設定します。

＜L2TPv3 設定＞

```
NXR_A1(config)#l2tpv3 hostname nxra1
NXR_A1(config)#l2tpv3 router-id 172.20.10.1
NXR_A1(config)#l2tpv3 mac-learning
NXR_A1(config)#l2tpv3 mac-aging 300
NXR_A1(config)#l2tpv3 path-mtu-discovery
```

L2TPv3 のホスト名やルータ ID 等を設定します。

```
NXR_A1(config)#no l2tpv3 loop-detect
```

LoopDetect 機能を「無効」に設定します。(デフォルト値は「無効」に設定されています)

```
NXR_A1(config)#l2tpv3 send-known-unicast
```

Known Unicast 送信機能を「有効」に設定します。

これにより Xconnect インタフェースより受信したユニキャストフレームの宛先 MAC アドレスが L2TPv3 ローカル MAC テーブルに存在する場合でも、セッション側へフレームを転送します。

＜L2TPv3 トンネル設定＞

```
NXR_A1(config)#l2tpv3 tunnel 1
NXR_A1(config-l2tpv3-tunnel)#description NXR_B
NXR_A1(config-l2tpv3-tunnel)#tunnel hostname nxrb
NXR_A1(config-l2tpv3-tunnel)#tunnel router-id 172.20.30.1
NXR_A1(config-l2tpv3-tunnel)#tunnel vendor ietf
```

L2TPv3 トンネル「１」を設定します。

＜L2TPv3 Xconnect 設定＞

```
NXR_A1(config)#l2tpv3 xconnect 1
NXR_A1(config-l2tpv3-xconnect)#description NXR_B
NXR_A1(config-l2tpv3-xconnect)#tunnel 1
NXR_A1(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_A1(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_A1(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
```

L2TPv3 Xconnect「１」を設定します。

```
NXR_A1(config-l2tpv3-xconnect)#send-known-unicast enable
```

Known Unicast 送信機能を「有効」に設定します。

この設定を有効にするには、L2TPv3 設定において「send-known-unicast」が設定されている必要があります。

＜DNS 設定＞

```
NXR_A1(config)#dns
NXR_A1(dns-config)#service enable
```

DNS に関する設定をします。

[NXR_A2 の設定]

```
nxr130#configure terminal
nxr130(config)#hostname NXR_A2
NXR_A2(config)#interface ethernet 0
NXR_A2(config-if)#ip address 192.168.10.2/24
NXR_A2(config-if)#exit
NXR_A2(config)#ip access-list ppp0_in permit any 10.10.20.1 115
NXR_A2(config)#interface ppp 0
NXR_A2(config-ppp)#ip address 10.10.20.1/32
NXR_A2(config-ppp)#ip masquerade
NXR_A2(config-ppp)#ip access-group in ppp0_in
NXR_A2(config-ppp)#ip spi-filter
NXR_A2(config-ppp)#ip tcp adjust-mss auto
NXR_A2(config-ppp)#no ip redirects
NXR_A2(config-ppp)#ppp authentication pap
NXR_A2(config-ppp)#ppp username test2@centurysys password test2pass
NXR_A2(config-ppp)#exit
NXR_A2(config)#interface ethernet 1
NXR_A2(config-if)#no ip address
NXR_A2(config-if)#pppoe-client ppp 0
NXR_A2(config-if)#exit
NXR_A2(config)#ip route 0.0.0.0/0 ppp 0
NXR_A2(config)#l2tpv3 hostname nxra2
NXR_A2(config)#l2tpv3 router-id 172.20.20.1
NXR_A2(config)#l2tpv3 mac-learning
NXR_A2(config)#l2tpv3 mac-aging 300
NXR_A2(config)#l2tpv3 path-mtu-discovery
NXR_A2(config)#no l2tpv3 loop-detect
NXR_A2(config)#l2tpv3 send-known-unicast
NXR_A2(config)#l2tpv3 tunnel 1
NXR_A2(config-l2tpv3-tunnel)#description NXR_B
NXR_A2(config-l2tpv3-tunnel)#tunnel hostname nxrb
NXR_A2(config-l2tpv3-tunnel)#tunnel router-id 172.20.30.1
NXR_A2(config-l2tpv3-tunnel)#tunnel vendor ietf
NXR_A2(config-l2tpv3-tunnel)#exit
NXR_A2(config)#l2tpv3 xconnect 1
NXR_A2(config-l2tpv3-xconnect)#description NXR_B
NXR_A2(config-l2tpv3-xconnect)#tunnel 1
NXR_A2(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_A2(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_A2(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
NXR_A2(config-l2tpv3-xconnect)#send-known-unicast enable
NXR_A2(config-l2tpv3-xconnect)#exit
NXR_A2(config)#dns
NXR_A2(dns-config)#service enable
NXR_A2(dns-config)#exit
NXR_A2(config)#exit
NXR_A2#save config
```

【 解説 】

- Known Unicast 送信機能を「有効」に設定し、Xconnect インタフェースより受信したユニキャストフレームの宛先 MAC アドレスが L2TPv3 ローカル MAC テーブルに存在する場合でも、セッション側へフレームを転送します。

  LAN_A 側のルータでは、Xconnect インタフェース側で LAN_B から送信されたフレームを受信してしまう場合があるため、この設定をしない場合は L2TPv3 ローカル MAC テーブルに登録された該当 MAC アドレスが削除される（エージングタイマが切れる）まで通信できない場合があります。
- 対向ルータの NXR_B は WAN 側 IP アドレスが動的のため、こちらからはネゴシエーションを行わな

いようにしています。

＜ホスト名の設定＞

```
nxr130(config)#hostname NXR_A2
```

ホスト名として「NXR_A2」を設定します。

＜Ethernet0 インタフェース設定＞

```
NXR_A2(config)#interface ethernet 0
NXR_A2(config-if)#ip address 192.168.10.2/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.2/24」を設定します。

＜IPv4 アクセスリスト設定＞

```
NXR_A2(config)#ip access-list ppp0_in permit any 10.10.20.1 115
```

IPv4 アクセスリスト名を「ppp0_in」とし、以下のルールで設定します。

動作	送信元 IPv4 アドレス	宛先 IPv4 アドレス	プロトコル	送信元ポート	宛先ポート
許可	any	10.10.20.1	115(L2TP)	-	-

＜ppp0 インタフェース設定＞

```
NXR_A2(config)#interface ppp 0
NXR_A2(config-ppp)#ip address 10.10.20.1/32
NXR_A2(config-ppp)#ip masquerade
NXR_A2(config-ppp)#ip access-group in ppp0_in
NXR_A2(config-ppp)#ip spi-filter
NXR_A2(config-ppp)#ip tcp adjust-mss auto
NXR_A2(config-ppp)#no ip redirects
NXR_A2(config-ppp)#ppp authentication pap
NXR_A2(config-ppp)#ppp username test2@centurysys password test2pass
```

ppp0 インタフェースに関する設定をします。

IPv4 アドレスが固定のため、「10.10.20.1/32」を設定します。

IPv4 アクセスリスト設定で設定した「ppp0_in」を ppp0 インタフェースの「in」フィルタに適用します。

＜Ethernet1 インタフェース設定＞

```
NXR_A2(config)#interface ethernet 1
NXR_A2(config-if)#no ip address
NXR_A2(config-if)#pppoe-client ppp 0
```

Ethernet1 インタフェースに関する設定をします。

＜スタティックルート設定＞

```
NXR_A2(config)#ip route 0.0.0.0/0 ppp 0
```

デフォルトルートを設定します。

＜L2TPv3 設定＞

```
NXR_A2(config)#l2tpv3 hostname nxra2
NXR_A2(config)#l2tpv3 router-id 172.20.20.1
NXR_A2(config)#l2tpv3 mac-learning
NXR_A2(config)#l2tpv3 mac-aging 300
NXR_A2(config)#l2tpv3 path-mtu-discovery
```

L2TPv3 のホスト名やルータ ID 等を設定します。

```
NXR_A2(config)#no l2tpv3 loop-detect
```

LoopDetect 機能を「無効」に設定します。(デフォルト値は「無効」に設定されています)

```
NXR_A2(config)#l2tpv3 send-known-unicast
```

Known Unicast 送信機能を「有効」に設定します。

これにより Xconnect インタフェースより受信したユニキャストフレームの宛先 MAC アドレスが L2TPv3 ローカル MAC テーブルに存在する場合でも、セッション側へフレームを転送します。

＜L2TPv3 トンネル設定＞

```
NXR_A2(config)#l2tpv3 tunnel 1
NXR_A2(config-l2tpv3-tunnel)#description NXR_B
NXR_A2(config-l2tpv3-tunnel)#tunnel hostname nxrb
NXR_A2(config-l2tpv3-tunnel)#tunnel router-id 172.20.30.1
NXR_A2(config-l2tpv3-tunnel)#tunnel vendor ietf
```

L2TPv3 トンネル「１」を設定します。

＜L2TPv3 Xconnect 設定＞

```
NXR_A2(config)#l2tpv3 xconnect 1
NXR_A2(config-l2tpv3-xconnect)#description NXR_B
NXR_A2(config-l2tpv3-xconnect)#tunnel 1
NXR_A2(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_A2(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_A2(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
NXR_A2(config-l2tpv3-xconnect)#send-known-unicast enable
```

L2TPv3 Xconnect「１」を設定します。

Known Unicast 送信機能を「有効」に設定します。

この設定を有効にするには、L2TPv3 設定において「send-known-unicast」が設定されている必要があります。

＜DNS 設定＞

```
NXR_A2(config)#dns
NXR_A2(dns-config)#service enable
```

DNS に関する設定をします。

[NXR_B の設定]

```
nxr130#configure terminal
nxr130(config)#hostname NXR_B
NXR_B(config)#interface ethernet 0
NXR_B(config-if)#ip address 192.168.10.3/24
NXR_B(config-if)#exit
NXR_B(config)#ip access-list ppp0_in permit 10.10.10.1 any 115
NXR_B(config)#ip access-list ppp0_in permit 10.10.20.1 any 115
NXR_B(config)#interface ppp 0
NXR_B(config-ppp)#ip address negotiated
NXR_B(config-ppp)#ip masquerade
NXR_B(config-ppp)#ip access-group in ppp0_in
NXR_B(config-ppp)#ip spi-filter
NXR_B(config-ppp)#ip tcp adjust-mss auto
NXR_B(config-ppp)#no ip redirects
NXR_B(config-ppp)#ppp authentication pap
NXR_B(config-ppp)#ppp username test3@centurysys password test3pass
NXR_B(config-ppp)#exit
NXR_B(config)#interface ethernet 1
NXR_B(config-if)#no ip address
NXR_B(config-if)#pppoe-client ppp 0
NXR_B(config-if)#exit
NXR_B(config)#ip route 0.0.0.0/0 ppp 0
NXR_B(config)#l2tpv3 hostname nxrb
NXR_B(config)#l2tpv3 router-id 172.20.30.1
NXR_B(config)#l2tpv3 mac-learning
NXR_B(config)#l2tpv3 mac-aging 300
NXR_B(config)#l2tpv3 path-mtu-discovery
NXR_B(config)#l2tpv3 tunnel 1
NXR_B(config-l2tpv3-tunnel)#description NXR_A1
NXR_B(config-l2tpv3-tunnel)#tunnel address 10.10.10.1
NXR_B(config-l2tpv3-tunnel)#tunnel hostname nxra1
NXR_B(config-l2tpv3-tunnel)#tunnel router-id 172.20.10.1
NXR_B(config-l2tpv3-tunnel)#tunnel vendor ietf
NXR_B(config-l2tpv3-tunnel)#exit
NXR_B(config)#l2tpv3 xconnect 1
NXR_B(config-l2tpv3-xconnect)#description NXR_A1
NXR_B(config-l2tpv3-xconnect)#tunnel 1
NXR_B(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_B(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_B(config-l2tpv3-xconnect)#retry-interval 30
NXR_B(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
NXR_B(config-l2tpv3-xconnect)#exit
NXR_B(config)#l2tpv3 tunnel 2
NXR_B(config-l2tpv3-tunnel)#description NXR_A2
NXR_B(config-l2tpv3-tunnel)#tunnel address 10.10.20.1
NXR_B(config-l2tpv3-tunnel)#tunnel hostname nxra2
NXR_B(config-l2tpv3-tunnel)#tunnel router-id 172.20.20.1
NXR_B(config-l2tpv3-tunnel)#tunnel vendor ietf
NXR_B(config-l2tpv3-tunnel)#exit
NXR_B(config)#l2tpv3 xconnect 2
NXR_B(config-l2tpv3-xconnect)#description NXR_A2
NXR_B(config-l2tpv3-xconnect)#tunnel 2
NXR_B(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_B(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_B(config-l2tpv3-xconnect)#retry-interval 30
NXR_B(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
NXR_B(config-l2tpv3-xconnect)#exit
NXR_B(config)#l2tpv3 group 1
NXR_B(config-l2tpv3-group)#xconnect 1 2
```

----- 次のページに続きがあります -----

```
----- 前のページからの続きです -----

NXR_B(config-l2tpv3-group)#no preempt enable
NXR_B(config-l2tpv3-group)#exit
NXR_B(config)#dns
NXR_B(dns-config)#service enable
NXR_B(dns-config)#exit
NXR_B(config)#exit
NXR_B#save config
```

【 解説 】

- L2TPv3 Group 機能を設定することにより L2TPv3 セッションの二重化を行います。
- L2TPv3 Group 機能の preempt モードを「無効」に設定します。これによりセカンダリセッションがアクティブセッションとなっている状態でプライマリセッションが確立した場合、セカンダリセッションがアクティブな状態を維持し続けるようになります。
- L2TPv3 トンネル/セッションが切断されたときに自動再接続できるように、リトライインターバルを設定します。

＜ホスト名の設定＞

```
nxr130(config)#hostname NXR_B
```

ホスト名として「NXR_B」を設定します。

＜Ethernet0 インタフェース設定＞

```
NXR_B(config)#interface ethernet 0
NXR_B(config-if)#ip address 192.168.10.3/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.3/24」を設定します。

＜IPv4 アクセスリスト設定＞

```
NXR_B(config)#ip access-list ppp0_in permit 10.10.10.1 any 115
NXR_B(config)#ip access-list ppp0_in permit 10.10.20.1 any 115
```

IPv4 アクセスリスト名を「ppp0_in」とし、以下のルールで設定します。

動作	送信元 IPv4 アドレス	宛先 IPv4 アドレス	プロトコル	送信元ポート	宛先ポート
許可	10.10.10.1	any	115(L2TP)	-	-
許可	10.10.20.1	any	115(L2TP)	-	-

＜ppp0 インタフェース設定＞

```
NXR_B(config)#interface ppp 0
NXR_B(config-ppp)#ip address negotiated
NXR_B(config-ppp)#ip masquerade
NXR_B(config-ppp)#ip access-group in ppp0_in
NXR_B(config-ppp)#ip spi-filter
NXR_B(config-ppp)#ip tcp adjust-mss auto
NXR_B(config-ppp)#no ip redirects
NXR_B(config-ppp)#ppp authentication pap
NXR_B(config-ppp)#ppp username test3@centurysys password test3pass
```

ppp0 インタフェースに関する設定をします。

IPv4 アクセスリスト設定で設定した「ppp0_in」を ppp0 インタフェースの「in」フィルタに適用します。

＜Ethernet1 インタフェース設定＞

```
NXR_B(config)#interface ethernet 1
NXR_B(config-if)#no ip address
NXR_B(config-if)#pppoe-client ppp 0
```

Ethernet1 インタフェースに関する設定をします。

＜スタティックルート設定＞

```
NXR_B(config)#ip route 0.0.0.0/0 ppp 0
```

デフォルトルートを設定します。

＜L2TPv3 設定＞

```
NXR_B(config)#l2tpv3 hostname nxrb
NXR_B(config)#l2tpv3 router-id 172.20.30.1
NXR_B(config)#l2tpv3 mac-learning
NXR_B(config)#l2tpv3 mac-aging 300
NXR_B(config)#l2tpv3 path-mtu-discovery
```

L2TPv3 のホスト名やルータ ID 等を設定します。

＜L2TPv3 トンネル設定１＞

```
NXR_B(config)#l2tpv3 tunnel 1
NXR_B(config-l2tpv3-tunnel)#description NXR_A1
NXR_B(config-l2tpv3-tunnel)#tunnel address 10.10.10.1
NXR_B(config-l2tpv3-tunnel)#tunnel hostname nxra1
NXR_B(config-l2tpv3-tunnel)#tunnel router-id 172.20.10.1
NXR_B(config-l2tpv3-tunnel)#tunnel vendor ietf
```

L2TPv3 トンネル「１」を設定します。

＜L2TPv3 Xconnect 設定１＞

```
NXR_B(config)#l2tpv3 xconnect 1
NXR_B(config-l2tpv3-xconnect)#description NXR_A1
NXR_B(config-l2tpv3-xconnect)#tunnel 1
NXR_B(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_B(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_B(config-l2tpv3-xconnect)#retry-interval 30
NXR_B(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
```

L2TPv3 Xconnect「１」を設定します。

＜L2TPv3 トンネル設定２＞

```
NXR_B(config)#l2tpv3 tunnel 2
NXR_B(config-l2tpv3-tunnel)#description NXR_A2
NXR_B(config-l2tpv3-tunnel)#tunnel address 10.10.20.1
NXR_B(config-l2tpv3-tunnel)#tunnel hostname nxra2
NXR_B(config-l2tpv3-tunnel)#tunnel router-id 172.20.20.1
NXR_B(config-l2tpv3-tunnel)#tunnel vendor ietf
```

L2TPv3 トンネル「２」を設定します。

＜L2TPv3 Xconnect 設定２＞

```
NXR_B(config)#l2tpv3 xconnect 2
NXR_B(config-l2tpv3-xconnect)#description NXR_A2
NXR_B(config-l2tpv3-xconnect)#tunnel 2
NXR_B(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_B(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_B(config-l2tpv3-xconnect)#retry-interval 30
NXR_B(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
```

L2TPv3 Xconnect「２」を設定します。

＜L2TPv3 Group 設定＞

```
NXR_B(config)#l2tpv3 group 1
```

L2TPv3 Group「１」を設定します。

```
NXR_B(config-l2tpv3-group)#xconnect 1 2
```

使用する Xconnect ナンバを指定します。

ここではプライマリセッションを「１」，セカンダリセッションを「２」として設定します。

```
NXR_B(config-l2tpv3-group)#no preempt enable
```

Group の preempt モードを設定します。

ここでは preempt モードを「無効」に設定します。(デフォルト値は「無効」に設定されています)

これによりセカンダリセッション（Xconnect 2）がアクティブセッションとなっている状態でプライマリセッション（Xconnect 1）が確立した場合、セカンダリセッションがアクティブな状態を維持し続けるようになります。

＜DNS 設定＞

```
NXR_C(config)#dns
NXR_C(dns-config)#service enable
```

DNS に関する設定をします。

8-5-3. パソコンの設定例

	LAN A のパソコン	LAN B のパソコン
IPv4 アドレス	192.168.10.100	192.168.10.101
サブネットマスク	255.255.255.0	255.255.255.0
デフォルトゲートウェイ	192.168.10.1	192.168.10.3

8-5-4. 補足 Preempt の設定例

L2TPv3 Group 機能の Preempt 設定を「有効」にした場合は、セカンダリセッションがアクティブセッションとなっている状態でプライマリセッションが確立した場合、プライマリセッションがアクティブセッションとなります。

[NXR_B の設定]

```
nxr130#configure terminal
nxr130(config)#hostname NXR_B
NXR_B(config)#interface ethernet 0
NXR_B(config-if)#ip address 192.168.10.3/24
NXR_B(config-if)#exit
NXR_B(config)#ip access-list ppp0_in permit 10.10.10.1 any 115
NXR_B(config)#ip access-list ppp0_in permit 10.10.20.1 any 115
NXR_B(config)#interface ppp 0
NXR_B(config-ppp)#ip address negotiated
NXR_B(config-ppp)#ip masquerade
NXR_B(config-ppp)#ip access-group in ppp0_in
NXR_B(config-ppp)#ip spi-filter
NXR_B(config-ppp)#ip tcp adjust-mss auto
NXR_B(config-ppp)#no ip redirects
NXR_B(config-ppp)#ppp authentication pap
NXR_B(config-ppp)#ppp username test3@centurysys password test3pass
NXR_B(config-ppp)#exit
NXR_B(config)#interface ethernet 1
NXR_B(config-if)#no ip address
NXR_B(config-if)#pppoe-client ppp 0
NXR_B(config-if)#exit
NXR_B(config)#ip route 0.0.0.0/0 ppp 0
NXR_B(config)#l2tpv3 hostname nxrb
NXR_B(config)#l2tpv3 router-id 172.20.30.1
NXR_B(config)#l2tpv3 mac-learning
NXR_B(config)#l2tpv3 mac-aging 300
NXR_B(config)#l2tpv3 path-mtu-discovery
NXR_B(config)#l2tpv3 tunnel 1
NXR_B(config-l2tpv3-tunnel)#description NXR_A1
NXR_B(config-l2tpv3-tunnel)#tunnel address 10.10.10.1
NXR_B(config-l2tpv3-tunnel)#tunnel hostname nxra1
NXR_B(config-l2tpv3-tunnel)#tunnel router-id 172.20.10.1
NXR_B(config-l2tpv3-tunnel)#tunnel vendor ietf
NXR_B(config-l2tpv3-tunnel)#exit
NXR_B(config)#l2tpv3 xconnect 1
NXR_B(config-l2tpv3-xconnect)#description NXR_A1
NXR_B(config-l2tpv3-xconnect)#tunnel 1
NXR_B(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_B(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_B(config-l2tpv3-xconnect)#retry-interval 30
NXR_B(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
NXR_B(config-l2tpv3-xconnect)#exit
NXR_B(config)#l2tpv3 tunnel 2
NXR_B(config-l2tpv3-tunnel)#description NXR_A2
NXR_B(config-l2tpv3-tunnel)#tunnel address 10.10.20.1
NXR_B(config-l2tpv3-tunnel)#tunnel hostname nxra2
NXR_B(config-l2tpv3-tunnel)#tunnel router-id 172.20.20.1
NXR_B(config-l2tpv3-tunnel)#tunnel vendor ietf
NXR_B(config-l2tpv3-tunnel)#exit
NXR_B(config)#l2tpv3 xconnect 2
NXR_B(config-l2tpv3-xconnect)#description NXR_A2

----- 次のページに続きがあります -----
```

```
----- 前のページからの続きです -----

NXR_B(config-l2tpv3-xconnect)#tunnel 2
NXR_B(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_B(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_B(config-l2tpv3-xconnect)#retry-interval 30
NXR_B(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
NXR_B(config-l2tpv3-xconnect)#exit
NXR_B(config)#l2tpv3 group 1
NXR_B(config-l2tpv3-group)#xconnect 1 2
NXR_B(config-l2tpv3-group)#preempt enable
NXR_B(config-l2tpv3-group)#exit
NXR_B(config)#dns
NXR_B(dns-config)#service enable
NXR_B(dns-config)#exit
NXR_B(config)#exit
NXR_B#save config
```

【 解説 】

- セカンダリセッションがアクティブセッションとなっている状態でプライマリセッションが確立した場合に、プライマリセッションをアクティブセッションとします。

＜L2TPv3 Group 設定＞

```
NXR_B(config)#l2tpv3 group 1
NXR_B(config-l2tpv3-group)#xconnect 1 2
NXR_B(config-l2tpv3-group)#preempt enable
```

L2TPv3 Group「１」を設定します。

Group の preempt モードを「有効」に設定します。

これによりセカンダリセッション（Xconnect 2）がアクティブセッションとなっている状態でプライマリセッション（Xconnect 1）が確立した場合、プライマリセッションがアクティブセッションとなります。

※L2TPv3 Group 設定以外は NXR_B の設定は同一です。

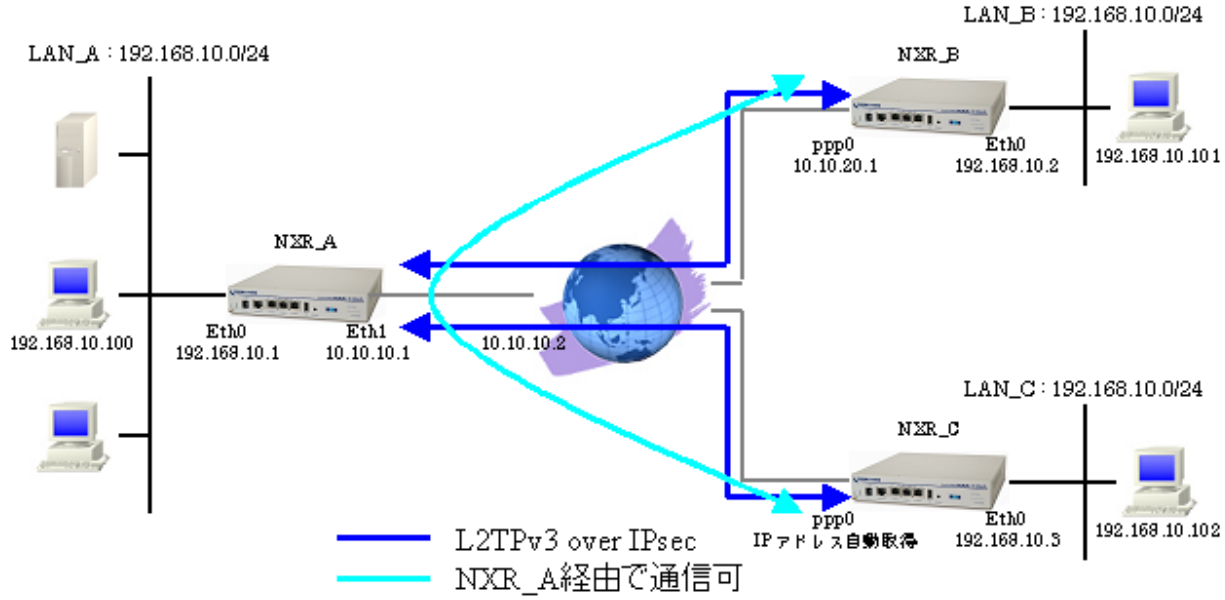
9. VPN 応用設定

9-1. L2TPv3 over IPsec 設定例

L2TPv3 だけでは通信内容を暗号化することができませんでしたが、IPsec を併用することにより暗号化することが可能です。これによりインターネット網を経由する場合でもよりセキュアな環境を実現できます。

なおここでは、各拠点からのインターネットアクセスを可能にするために、フィルタ設定（SPI），NAT 設定（IP マスカレード），DNS 設定を行っています。

9-1-1. 構成図



9-1-2. 設定例

[NXR_A の設定]

```
nxr130#configure terminal
nxr130(config)#hostname NXR_A
NXR_A(config)#interface ethernet 0
NXR_A(config-if)#ip address 192.168.10.1/24
NXR_A(config-if)#exit
NXR_A(config)#ip access-list eth1_in permit any 10.10.10.1 udp 500 500
NXR_A(config)#ip access-list eth1_in permit any 10.10.10.1 50
NXR_A(config)#ip access-list eth1_out deny 10.10.10.1 10.10.20.1 115
NXR_A(config)#ipsec access-list NXR_B ip host host
NXR_A(config)#ipsec access-list NXR_C ip host host
NXR_A(config)#interface ethernet 1
NXR_A(config-if)#ip address 10.10.10.1/29
NXR_A(config-if)#ip masquerade
NXR_A(config-if)#ip access-group in eth1_in
NXR_A(config-if)#ip access-group out eth1_out
NXR_A(config-if)#ip spi-filter
NXR_A(config-if)#no ip redirects
NXR_A(config-if)#exit
NXR_A(config)#ip route 0.0.0.0/0 10.10.10.2
NXR_A(config)#ipsec local policy 1

----- 次のページに続きがあります -----
```

----- 前のページからの続きです -----

```
NXR_A(config-ipsec-local)#address ip
NXR_A(config-ipsec-local)#exit
NXR_A(config)#ipsec isakmp policy 1
NXR_A(config-ipsec-isakmp)#description NXR_B
NXR_A(config-ipsec-isakmp)#authentication pre-share ipseckey1
NXR_A(config-ipsec-isakmp)#hash sha1
NXR_A(config-ipsec-isakmp)#encryption aes128
NXR_A(config-ipsec-isakmp)#group 5
NXR_A(config-ipsec-isakmp)#lifetime 10800
NXR_A(config-ipsec-isakmp)#isakmp-mode main
NXR_A(config-ipsec-isakmp)#remote address ip 10.10.20.1
NXR_A(config-ipsec-isakmp)#keepalive 30 3 periodic restart
NXR_A(config-ipsec-isakmp)#local policy 1
NXR_A(config-ipsec-isakmp)#exit
NXR_A(config)#ipsec tunnel policy 1
NXR_A(config-ipsec-tunnel)#description NXR_B
NXR_A(config-ipsec-tunnel)#negotiation-mode auto
NXR_A(config-ipsec-tunnel)#set transform esp-aes128 esp-sha1-hmac
NXR_A(config-ipsec-tunnel)#set pfs group5
NXR_A(config-ipsec-tunnel)#set sa lifetime 3600
NXR_A(config-ipsec-tunnel)#set key-exchange isakmp 1
NXR_A(config-ipsec-tunnel)#match address NXR_B
NXR_A(config-ipsec-tunnel)#exit
NXR_A(config)#ipsec isakmp policy 2
NXR_A(config-ipsec-isakmp)#description NXR_C
NXR_A(config-ipsec-isakmp)#authentication pre-share ipseckey2
NXR_A(config-ipsec-isakmp)#hash sha1
NXR_A(config-ipsec-isakmp)#encryption aes128
NXR_A(config-ipsec-isakmp)#group 5
NXR_A(config-ipsec-isakmp)#lifetime 10800
NXR_A(config-ipsec-isakmp)#isakmp-mode aggressive
NXR_A(config-ipsec-isakmp)#remote address ip any
NXR_A(config-ipsec-isakmp)#remote identity fqdn nxrc
NXR_A(config-ipsec-isakmp)#keepalive 30 3 periodic clear
NXR_A(config-ipsec-isakmp)#local policy 1
NXR_A(config-ipsec-isakmp)#exit
NXR_A(config)#ipsec tunnel policy 2
NXR_A(config-ipsec-tunnel)#description NXR_C
NXR_A(config-ipsec-tunnel)#negotiation-mode manual
NXR_A(config-ipsec-tunnel)#set transform esp-aes128 esp-sha1-hmac
NXR_A(config-ipsec-tunnel)#set pfs group5
NXR_A(config-ipsec-tunnel)#set sa lifetime 3600
NXR_A(config-ipsec-tunnel)#set key-exchange isakmp 2
NXR_A(config-ipsec-tunnel)#match address NXR_C
NXR_A(config-ipsec-tunnel)#exit
NXR_A(config)#interface ethernet 1
NXR_A(config-if)#ipsec policy 1
NXR_A(config-if)#exit
NXR_A(config)#l2tpv3 hostname nxra
NXR_A(config)#l2tpv3 router-id 172.20.10.1
NXR_A(config)#l2tpv3 mac-learning
NXR_A(config)#l2tpv3 mac-aging 300
NXR_A(config)#l2tpv3 path-mtu-discovery
NXR_A(config)#l2tpv3 tunnel 1
NXR_A(config-l2tpv3-tunnel)#description NXR_B
NXR_A(config-l2tpv3-tunnel)#tunnel address 10.10.20.1
NXR_A(config-l2tpv3-tunnel)#tunnel hostname nxrb
NXR_A(config-l2tpv3-tunnel)#tunnel router-id 172.20.20.1
NXR_A(config-l2tpv3-tunnel)#tunnel vendor ietf
```

----- 次のページに続きがあります -----

```
----- 前のページからの続きです -----

NXR_A(config-l2tpv3-tunnel)#exit
NXR_A(config)#l2tpv3 xconnect 1
NXR_A(config-l2tpv3-xconnect)#description NXR_B
NXR_A(config-l2tpv3-xconnect)#tunnel 1
NXR_A(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_A(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_A(config-l2tpv3-xconnect)#retry-interval 30
NXR_A(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
NXR_A(config-l2tpv3-xconnect)#exit
NXR_A(config)#l2tpv3 tunnel 2
NXR_A(config-l2tpv3-tunnel)#description NXR_C
NXR_A(config-l2tpv3-tunnel)#tunnel address 0.0.0.0
NXR_A(config-l2tpv3-tunnel)#tunnel hostname nxrc
NXR_A(config-l2tpv3-tunnel)#tunnel router-id 172.20.30.1
NXR_A(config-l2tpv3-tunnel)#tunnel vendor ietf
NXR_A(config-l2tpv3-tunnel)#exit
NXR_A(config)#l2tpv3 xconnect 2
NXR_A(config-l2tpv3-xconnect)#description NXR_C
NXR_A(config-l2tpv3-xconnect)#tunnel 2
NXR_A(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_A(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_A(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
NXR_A(config-l2tpv3-xconnect)#exit
NXR_A(config)#dns
NXR_A(dns-config)#service enable
NXR_A(dns-config)#root enable
NXR_A(dns-config)#exit
NXR_A(config)#exit
NXR_A#save config
```

【 解説 】

- L2TP のパケットを IPsec でカプセル化できるように IPsec の Tunnel ポリシー設定は本ルータおよび対向ルータの WAN 側 IPv4 アドレスを設定します。
- IPsec SA が確立する前に L2TPv3 のネゴシエーションおよび通信を防止するために WAN 側インタフェースから出力される L2TP パケットを破棄しています。

＜ホスト名の設定＞

```
nxr130(config)#hostname NXR_A
```

ホスト名として「NXR_A」を設定します。

＜Ethernet0 インタフェース設定＞

```
NXR_A(config)#interface ethernet 0
NXR_A(config-if)#ip address 192.168.10.1/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.1/24」を設定します。

＜IPv4 アクセスリスト設定＞

```
NXR_A(config)#ip access-list eth1_in permit any 10.10.10.1 udp 500 500
NXR_A(config)#ip access-list eth1_in permit any 10.10.10.1 50
NXR_A(config)#ip access-list eth1_out deny 10.10.10.1 10.10.20.1 115
```

IPv4 アクセスリスト名を「eth1_in」とし、以下のルールで設定します。

動作	送信元 IPv4 アドレス	宛先 IPv4 アドレス	プロトコル	送信元ポート	宛先ポート
許可	any	10.10.10.1	UDP	500	500
許可	any	10.10.10.1	50(ESP)	-	-

IPv4 アクセスリスト名を「eth1_out」とし、以下のルールで設定します。
これにより IPsec SA が確立する前に L2TPv3 のネゴシエーションおよび通信を防止するために WAN 側インタフェースから出力される L2TP パケットを破棄します。

動作	送信元 IPv4 アドレス	宛先 IPv4 アドレス	プロトコル	送信元ポート	宛先ポート
破棄	10.10.10.1	10.10.20.1	115(L2TP)	-	-

＜IPsec アクセスリスト設定＞

```
NXR_A(config)#ipsec access-list NXR_B ip host host
```

IPsec アクセスリスト名を「NXR_B」とし、送信元 IPv4 アドレス「host」, 宛先 IPv4 アドレス「host」を設定します。これにより L2TPv3 パケットを IPsec でカプセル化します。

```
NXR_A(config)#ipsec access-list NXR_C ip host host
```

IPsec アクセスリスト名を「NXR_C」とし、送信元 IPv4 アドレス「host」, 宛先 IPv4 アドレス「host」を設定します。これにより L2TPv3 パケットを IPsec でカプセル化します。

＜Ethernet1 インタフェース設定＞

```
NXR_A(config)#interface ethernet 1
NXR_A(config-if)#ip address 10.10.10.1/29
NXR_A(config-if)#ip masquerade
NXR_A(config-if)#ip access-group in eth1_in
NXR_A(config-if)#ip access-group out eth1_out
NXR_A(config-if)#ip spi-filter
NXR_A(config-if)#no ip redirects
```

Ethernet1 インタフェースに関する設定をします。
IPv4 アドレスが固定のため、「10.10.10.1/29」を設定します。
IPv4 アクセスリスト設定で設定した「eth1_in」を Ethernet1 インタフェースの「in」フィルタに適用します。
IPv4 アクセスリスト設定で設定した「eth1_out」を Ethernet1 インタフェースの「out」フィルタに適用します。

＜スタティックルート設定＞

```
NXR_A(config)#ip route 0.0.0.0/0 10.10.10.2
```

デフォルトルートを設定します。

＜IPsec ローカルポリシー設定＞

```
NXR_A(config)#ipsec local policy 1
NXR_A(config-ipsec-local)#address ip
```

IPsec のローカルポリシー「１」を設定します。

＜IPsec ISAKMP ポリシー設定１＞

```
NXR_A(config)#ipsec isakmp policy 1
NXR_A(config-ipsec-isakmp)#description NXR_B
NXR_A(config-ipsec-isakmp)#authentication pre-share ipseckey1
NXR_A(config-ipsec-isakmp)#hash sha1
NXR_A(config-ipsec-isakmp)#encryption aes128
NXR_A(config-ipsec-isakmp)#group 5
NXR_A(config-ipsec-isakmp)#lifetime 10800
NXR_A(config-ipsec-isakmp)#isakmp-mode main
NXR_A(config-ipsec-isakmp)#remote address ip 10.10.20.1
NXR_A(config-ipsec-isakmp)#keepalive 30 3 periodic restart
NXR_A(config-ipsec-isakmp)#local policy 1
```

IPsec の ISAKMP ポリシー「１」を設定します。

事前共有鍵(Pre-Shared Key)として「ipseckey1」を設定します。

フェーズ１のネゴシエーションモードとして「main」を設定します。

＜IPsec tunnel ポリシー設定１＞

```
NXR_A(config)#ipsec tunnel policy 1
NXR_A(config-ipsec-tunnel)#description NXR_B
NXR_A(config-ipsec-tunnel)#negotiation-mode auto
NXR_A(config-ipsec-tunnel)#set transform esp-aes128 esp-sha1-hmac
NXR_A(config-ipsec-tunnel)#set pfs group5
NXR_A(config-ipsec-tunnel)#set sa lifetime 3600
NXR_A(config-ipsec-tunnel)#set key-exchange isakmp 1
NXR_A(config-ipsec-tunnel)#match address NXR_B
```

IPsec の tunnel ポリシー「１」を設定します。

使用する IPsec アクセスリストとして「NXR_B」を設定します。

＜IPsec ISAKMP ポリシー設定２＞

```
NXR_A(config)#ipsec isakmp policy 2
NXR_A(config-ipsec-isakmp)#description NXR_C
NXR_A(config-ipsec-isakmp)#authentication pre-share ipseckey2
NXR_A(config-ipsec-isakmp)#hash sha1
NXR_A(config-ipsec-isakmp)#encryption aes128
NXR_A(config-ipsec-isakmp)#group 5
NXR_A(config-ipsec-isakmp)#lifetime 10800
NXR_A(config-ipsec-isakmp)#isakmp-mode aggressive
NXR_A(config-ipsec-isakmp)#remote address ip any
NXR_A(config-ipsec-isakmp)#remote identity fqdn nxrc
NXR_A(config-ipsec-isakmp)#keepalive 30 3 periodic clear
NXR_A(config-ipsec-isakmp)#local policy 1
```

IPsec の ISAKMP ポリシー「２」を設定します。

事前共有鍵(Pre-Shared Key)として「ipseckey2」を設定します。

フェーズ１のネゴシエーションモードとして「aggressive」を設定します。

＜IPsec tunnel ポリシー設定２＞

```
NXR_A(config)#ipsec tunnel policy 2
NXR_A(config-ipsec-tunnel)#description NXR_C
NXR_A(config-ipsec-tunnel)#negotiation-mode manual
NXR_A(config-ipsec-tunnel)#set transform esp-aes128 esp-sha1-hmac
NXR_A(config-ipsec-tunnel)#set pfs group5
NXR_A(config-ipsec-tunnel)#set sa lifetime 3600
NXR_A(config-ipsec-tunnel)#set key-exchange isakmp 2
NXR_A(config-ipsec-tunnel)#match address NXR_C
```

IPsec の tunnel ポリシー「２」を設定します。

使用する IPsec アクセスリストとして「NXR_C」を設定します。

＜Ethernet1 インタフェース設定＞

```
NXR_A(config)#interface ethernet 1
NXR_A(config-if)#ipsec policy 1
```

IPsec ローカルポリシー「１」を適用します。これによりこのインタフェースが IPsec トンネルのエンドポイントとなります。

＜L2TPv3 設定＞

```
NXR_A(config)#l2tpv3 hostname nxra
NXR_A(config)#l2tpv3 router-id 172.20.10.1
NXR_A(config)#l2tpv3 mac-learning
NXR_A(config)#l2tpv3 mac-aging 300
NXR_A(config)#l2tpv3 path-mtu-discovery
```

L2TPv3 のホスト名やルータ ID 等を設定します。

＜L2TPv3 トンネル設定１＞

```
NXR_A(config)#l2tpv3 tunnel 1
NXR_A(config-l2tpv3-tunnel)#description NXR_B
NXR_A(config-l2tpv3-tunnel)#tunnel address 10.10.20.1
NXR_A(config-l2tpv3-tunnel)#tunnel hostname nxrb
NXR_A(config-l2tpv3-tunnel)#tunnel router-id 172.20.20.1
NXR_A(config-l2tpv3-tunnel)#tunnel vendor ietf
```

L2TPv3 トンネル「１」を設定します。

＜L2TPv3 Xconnect 設定１＞

```
NXR_A(config)#l2tpv3 xconnect 1
NXR_A(config-l2tpv3-xconnect)#description NXR_B
NXR_A(config-l2tpv3-xconnect)#tunnel 1
NXR_A(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_A(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_A(config-l2tpv3-xconnect)#retry-interval 30
NXR_A(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
```

L2TPv3 Xconnect「１」を設定します。

＜L2TPv3 トンネル設定２＞

```
NXR_A(config)#l2tpv3 tunnel 2
NXR_A(config-l2tpv3-tunnel)#description NXR_C
NXR_A(config-l2tpv3-tunnel)#tunnel address 0.0.0.0
NXR_A(config-l2tpv3-tunnel)#tunnel hostname nxrc
NXR_A(config-l2tpv3-tunnel)#tunnel router-id 172.20.30.1
NXR_A(config-l2tpv3-tunnel)#tunnel vendor ietf
```

L2TPv3 トンネル「２」を設定します。

＜L2TPv3 Xconnect 設定２＞

```
NXR_A(config)#l2tpv3 xconnect 2
NXR_A(config-l2tpv3-xconnect)#description NXR_C
NXR_A(config-l2tpv3-xconnect)#tunnel 2
NXR_A(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_A(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_A(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
```

L2TPv3 Xconnect「２」を設定します。

＜DNS 設定＞

```
NXR_A(config)#dns
NXR_A(dns-config)#service enable
NXR_A(dns-config)#root enable
```

DNS に関する設定をします。

ここでは DNS のルートサーバを使用する設定にしています。

[NXR_B の設定]

```
nxr130#configure terminal
nxr130(config)#hostname NXR_B
NXR_B(config)#interface ethernet 0
NXR_B(config-if)#ip address 192.168.10.2/24
NXR_B(config-if)#exit
NXR_B(config)#ip access-list ppp0_in permit 10.10.10.1 10.10.20.1 udp 500 500
NXR_B(config)#ip access-list ppp0_in permit 10.10.10.1 10.10.20.1 50
NXR_B(config)#ip access-list ppp0_out deny 10.10.20.1 10.10.10.1 115
NXR_B(config)#ipsec access-list NXR_A ip host host
NXR_B(config)#interface ppp 0
NXR_B(config-ppp)#ip address 10.10.20.1/32
NXR_B(config-ppp)#ip masquerade
NXR_B(config-ppp)#ip access-group in ppp0_in
NXR_B(config-ppp)#ip access-group out ppp0_out
NXR_B(config-ppp)#ip spi-filter
NXR_B(config-ppp)#ip tcp adjust-mss auto
NXR_B(config-ppp)#no ip redirects
NXR_B(config-ppp)#ppp authentication pap
NXR_B(config-ppp)#ppp username test2@centurysys password test2pass
NXR_B(config-ppp)#exit
NXR_B(config)#interface ethernet 1
NXR_B(config-if)#no ip address
NXR_B(config-if)#pppoe-client ppp 0
NXR_B(config-if)#exit
NXR_B(config)#ip route 0.0.0.0/0 ppp 0
NXR_B(config)#ipsec local policy 1
NXR_B(config-ipsec-local)#address ip
NXR_B(config-ipsec-local)#exit
NXR_B(config)#ipsec isakmp policy 1
NXR_B(config-ipsec-isakmp)#description NXR_A
NXR_B(config-ipsec-isakmp)#authentication pre-share ipseckey1
NXR_B(config-ipsec-isakmp)#hash sha1
NXR_B(config-ipsec-isakmp)#encryption aes128
NXR_B(config-ipsec-isakmp)#group 5
NXR_B(config-ipsec-isakmp)#lifetime 10800
NXR_B(config-ipsec-isakmp)#isakmp-mode main
NXR_B(config-ipsec-isakmp)#remote address ip 10.10.10.1
NXR_B(config-ipsec-isakmp)#keepalive 30 3 periodic restart
NXR_B(config-ipsec-isakmp)#local policy 1
NXR_B(config-ipsec-isakmp)#exit
NXR_B(config)#ipsec tunnel policy 1
NXR_B(config-ipsec-tunnel)#description NXR_A
NXR_B(config-ipsec-tunnel)#negotiation-mode auto
NXR_B(config-ipsec-tunnel)#set transform esp-aes128 esp-sha1-hmac
NXR_B(config-ipsec-tunnel)#set pfs group5
NXR_B(config-ipsec-tunnel)#set sa lifetime 3600
NXR_B(config-ipsec-tunnel)#set key-exchange isakmp 1
NXR_B(config-ipsec-tunnel)#match address NXR_A
NXR_B(config-ipsec-tunnel)#exit
NXR_B(config)#interface ppp 0
NXR_B(config-ppp)#ipsec policy 1
NXR_B(config-ppp)#exit
NXR_B(config)#l2tpv3 hostname nxrb
NXR_B(config)#l2tpv3 router-id 172.20.20.1
NXR_B(config)#l2tpv3 mac-learning
NXR_B(config)#l2tpv3 mac-aging 300
NXR_B(config)#l2tpv3 path-mtu-discovery
NXR_B(config)#l2tpv3 tunnel 1
NXR_B(config-l2tpv3-tunnel)#description NXR_A
```

----- 次のページに続きがあります -----

```
----- 前のページからの続きです -----

NXR_B(config-l2tpv3-tunnel)#tunnel address 10.10.10.1
NXR_B(config-l2tpv3-tunnel)#tunnel hostname nxra
NXR_B(config-l2tpv3-tunnel)#tunnel router-id 172.20.10.1
NXR_B(config-l2tpv3-tunnel)#tunnel vendor ietf
NXR_B(config-l2tpv3-tunnel)#exit
NXR_B(config)#l2tpv3 xconnect 1
NXR_B(config-l2tpv3-xconnect)#description NXR_A
NXR_B(config-l2tpv3-xconnect)#tunnel 1
NXR_B(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_B(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_B(config-l2tpv3-xconnect)#retry-interval 45
NXR_B(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
NXR_B(config-l2tpv3-xconnect)#exit
NXR_B(config)#dns
NXR_B(dns-config)#service enable
NXR_B(dns-config)#exit
NXR_B(config)#exit
NXR_B#save config
```

【 解説 】

- L2TP のパケットを IPsec でカプセル化できるように IPsec の Tunnel ポリシー設定は本ルータおよび対向ルータの WAN 側 IPv4 アドレスを設定します。
- IPsec SA が確立する前に L2TPv3 のネゴシエーションおよび通信を防止するために WAN 側インタフェースから出力される L2TP パケットを破棄しています。

＜ホスト名の設定＞

```
nxr130(config)#hostname NXR_B
```

ホスト名として「NXR_B」を設定します。

＜Ethernet0 インタフェース設定＞

```
NXR_B(config)#interface ethernet 0
NXR_B(config-if)#ip address 192.168.10.2/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.2/24」を設定します。

＜IPv4 アクセスリスト設定＞

```
NXR_B(config)#ip access-list ppp0_in permit 10.10.10.1 10.10.20.1 udp 500 500
NXR_B(config)#ip access-list ppp0_in permit 10.10.10.1 10.10.20.1 50
NXR_B(config)#ip access-list ppp0_out deny 10.10.20.1 10.10.10.1 115
```

IPv4 アクセスリスト名を「ppp0_in」とし、以下のルールで設定します。

動作	送信元 IPv4 アドレス	宛先 IPv4 アドレス	プロトコル	送信元ポート	宛先ポート
許可	10.10.10.1	10.10.20.1	UDP	500	500
許可	10.10.10.1	10.10.20.1	50(ESP)	-	-

IPv4 アクセスリスト名を「ppp0_out」とし、以下のルールで設定します。

動作	送信元 IPv4 アドレス	宛先 IPv4 アドレス	プロトコル	送信元ポート	宛先ポート
破棄	10.10.20.1	10.10.10.1	115(L2TP)	-	-

＜IPsec アクセスリスト設定＞

```
NXR_B(config)#ipsec access-list NXR_A ip host host
```

IPsec アクセスリスト名を「NXR_A」とし、送信元 IPv4 アドレス「host」, 宛先 IPv4 アドレス「host」を設定します。これにより L2TPv3 パケットを IPsec でカプセル化します。

＜ppp0 インタフェース設定＞

```
NXR_B(config)#interface ppp 0
NXR_B(config-ppp)#ip address 10.10.20.1/32
NXR_B(config-ppp)#ip masquerade
NXR_B(config-ppp)#ip access-group in ppp0_in
NXR_B(config-ppp)#ip access-group out ppp0_out
NXR_B(config-ppp)#ip spi-filter
NXR_B(config-ppp)#ip tcp adjust-mss auto
NXR_B(config-ppp)#no ip redirects
NXR_B(config-ppp)#ppp authentication pap
NXR_B(config-ppp)#ppp username test2@centurysys password test2pass
```

ppp0 インタフェースに関する設定をします。

IPv4 アドレスが固定のため、「10.10.20.1/32」を設定します。

IPv4 アクセスリスト設定で設定した「ppp0_in」を ppp0 インタフェースの「in」フィルタに適用します。

IPv4 アクセスリスト設定で設定した「ppp0_out」を ppp0 インタフェースの「out」フィルタに適用します。

＜Ethernet1 インタフェース設定＞

```
NXR_B(config)#interface ethernet 1
NXR_B(config-if)#no ip address
NXR_B(config-if)#pppoe-client ppp 0
```

Ethernet1 インタフェースに関する設定をします。

＜スタティックルート設定＞

```
NXR_B(config)#ip route 0.0.0.0/0 ppp 0
```

デフォルトルートを設定します。

＜IPsec ローカルポリシー設定＞

```
NXR_B(config)#ipsec local policy 1
NXR_B(config-ipsec-local)#address ip
```

IPsec のローカルポリシー「１」を設定します。

＜IPsec ISAKMP ポリシー設定＞

```
NXR_B(config)#ipsec isakmp policy 1
NXR_B(config-ipsec-isakmp)#description NXR_A
NXR_B(config-ipsec-isakmp)#authentication pre-share ipseckey1
NXR_B(config-ipsec-isakmp)#hash sha1
NXR_B(config-ipsec-isakmp)#encryption aes128
NXR_B(config-ipsec-isakmp)#group 5
NXR_B(config-ipsec-isakmp)#lifetime 10800
NXR_B(config-ipsec-isakmp)#isakmp-mode main
NXR_B(config-ipsec-isakmp)#remote address ip 10.10.10.1
NXR_B(config-ipsec-isakmp)#keepalive 30 3 periodic restart
NXR_B(config-ipsec-isakmp)#local policy 1
```

IPsec の ISAKMP ポリシー「１」を設定します。

事前共有鍵(Pre-Shared Key)として「ipseckey1」を設定します。

フェーズ１のネゴシエーションモードとして「main」を設定します。

＜IPsec tunnel ポリシー設定＞

```
NXR_B(config)#ipsec tunnel policy 1
NXR_B(config-ipsec-tunnel)#description NXR_A
NXR_B(config-ipsec-tunnel)#negotiation-mode auto
NXR_B(config-ipsec-tunnel)#set transform esp-aes128 esp-sha1-hmac
NXR_B(config-ipsec-tunnel)#set pfs group5
NXR_B(config-ipsec-tunnel)#set sa lifetime 3600
NXR_B(config-ipsec-tunnel)#set key-exchange isakmp 1
NXR_B(config-ipsec-tunnel)#match address NXR_A
```

IPsec の tunnel ポリシー「１」を設定します。

使用する IPsec アクセスリストとして「NXR_A」を設定します。

＜ppp0 インタフェース設定＞

```
NXR_B(config)#interface ppp 0
NXR_B(config-ppp)#ipsec policy 1
```

IPsec ローカルポリシー「１」を適用します。これによりこのインタフェースが IPsec トンネルのエンドポイントとなります。

＜L2TPv3 設定＞

```
NXR_B(config)#l2tpv3 hostname nxrb
NXR_B(config)#l2tpv3 router-id 172.20.20.1
NXR_B(config)#l2tpv3 mac-learning
NXR_B(config)#l2tpv3 mac-aging 300
NXR_B(config)#l2tpv3 path-mtu-discovery
```

L2TPv3 のホスト名やルータ ID 等を設定します。

＜L2TPv3 トンネル設定＞

```
NXR_B(config)#l2tpv3 tunnel 1
NXR_B(config-l2tpv3-tunnel)#description NXR_A
NXR_B(config-l2tpv3-tunnel)#tunnel address 10.10.10.1
NXR_B(config-l2tpv3-tunnel)#tunnel hostname nxra
NXR_B(config-l2tpv3-tunnel)#tunnel router-id 172.20.10.1
NXR_B(config-l2tpv3-tunnel)#tunnel vendor ietf
```

L2TPv3 トンネル「１」を設定します。

＜L2TPv3 Xconnect 設定＞

```
NXR_B(config)#l2tpv3 xconnect 1
NXR_B(config-l2tpv3-xconnect)#description NXR_A
NXR_B(config-l2tpv3-xconnect)#tunnel 1
NXR_B(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_B(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_B(config-l2tpv3-xconnect)#retry-interval 45
NXR_B(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
```

L2TPv3 Xconnect「１」を設定します。

＜DNS 設定＞

```
NXR_B(config)#dns
NXR_B(dns-config)#service enable
```

DNS に関する設定をします。

[NXR_C の設定]

```
nxr130#configure terminal
nxr130(config)#hostname NXR_C
NXR_C(config)#interface ethernet 0
NXR_C(config-if)#ip address 192.168.10.3/24
NXR_C(config-if)#exit
NXR_C(config)#ip access-list ppp0_in permit 10.10.10.1 any udp 500 500
NXR_C(config)#ip access-list ppp0_in permit 10.10.10.1 any 50
NXR_C(config)#ip access-list ppp0_out deny any 10.10.10.1 115
NXR_C(config)#ipsec access-list NXR_A ip host host
NXR_C(config)#interface ppp 0
NXR_C(config-ppp)#ip address negotiated
NXR_C(config-ppp)#ip masquerade
NXR_C(config-ppp)#ip access-group in ppp0_in
NXR_C(config-ppp)#ip access-group out ppp0_out
NXR_C(config-ppp)#ip spi-filter
NXR_C(config-ppp)#ip tcp adjust-mss auto
NXR_C(config-ppp)#no ip redirects
NXR_C(config-ppp)#ppp authentication pap
NXR_C(config-ppp)#ppp username test3@centurysys password test3pass
NXR_C(config-ppp)#exit
NXR_C(config)#interface ethernet 1
NXR_C(config-if)#no ip address
NXR_C(config-if)#pppoe-client ppp 0
NXR_C(config-if)#exit
NXR_C(config)#ip route 0.0.0.0/0 ppp 0
NXR_C(config)#ipsec local policy 1
NXR_C(config-ipsec-local)#address ip
NXR_C(config-ipsec-local)#self-identity fqdn nxrc
NXR_C(config-ipsec-local)#exit
NXR_C(config)#ipsec isakmp policy 1
NXR_C(config-ipsec-isakmp)#description NXR_A
NXR_C(config-ipsec-isakmp)#authentication pre-share ipseckey2
NXR_C(config-ipsec-isakmp)#hash sha1
NXR_C(config-ipsec-isakmp)#encryption aes128
NXR_C(config-ipsec-isakmp)#group 5
NXR_C(config-ipsec-isakmp)#lifetime 10800
NXR_C(config-ipsec-isakmp)#isakmp-mode aggressive
NXR_C(config-ipsec-isakmp)#remote address ip 10.10.10.1
NXR_C(config-ipsec-isakmp)#keepalive 30 3 periodic restart
NXR_C(config-ipsec-isakmp)#local policy 1
NXR_C(config-ipsec-isakmp)#exit
NXR_C(config)#ipsec tunnel policy 1
NXR_C(config-ipsec-tunnel)#description NXR_A
NXR_C(config-ipsec-tunnel)#negotiation-mode auto
NXR_C(config-ipsec-tunnel)#set transform esp-aes128 esp-sha1-hmac
NXR_C(config-ipsec-tunnel)#set pfs group5
NXR_C(config-ipsec-tunnel)#set sa lifetime 3600
NXR_C(config-ipsec-tunnel)#set key-exchange isakmp 1
NXR_C(config-ipsec-tunnel)#match address NXR_A
NXR_C(config-ipsec-tunnel)#exit
NXR_C(config)#interface ppp 0
NXR_C(config-ppp)#ipsec policy 1
NXR_C(config-ppp)#exit
NXR_C(config)#l2tpv3 hostname nxrc
NXR_C(config)#l2tpv3 router-id 172.20.30.1
NXR_C(config)#l2tpv3 mac-learning
NXR_C(config)#l2tpv3 mac-aging 300
NXR_C(config)#l2tpv3 path-mtu-discovery
NXR_C(config)#l2tpv3 tunnel 1

----- 次のページに続きがあります -----
```

```
----- 前のページからの続きです -----

NXR_C(config-l2tpv3-tunnel)#description NXR_A
NXR_C(config-l2tpv3-tunnel)#tunnel address 10.10.10.1
NXR_C(config-l2tpv3-tunnel)#tunnel hostname nxra
NXR_C(config-l2tpv3-tunnel)#tunnel router-id 172.20.10.1
NXR_C(config-l2tpv3-tunnel)#tunnel vendor ietf
NXR_C(config-l2tpv3-tunnel)#exit
NXR_C(config)#l2tpv3 xconnect 1
NXR_C(config-l2tpv3-xconnect)#description NXR_A
NXR_C(config-l2tpv3-xconnect)#tunnel 1
NXR_C(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_C(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_C(config-l2tpv3-xconnect)#retry-interval 30
NXR_C(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
NXR_C(config-l2tpv3-xconnect)#exit
NXR_C(config)#dns
NXR_C(dns-config)#service enable
NXR_C(dns-config)#exit
NXR_C(config)#exit
NXR_C#save config
```

【 解説 】

- L2TP のパケットを IPsec でカプセル化できるように IPsec の Tunnel ポリシー設定は本ルータおよび対向ルータの WAN 側 IPv4 アドレスを設定します。
- IPsec SA が確立する前に L2TPv3 のネゴシエーションおよび通信を防止するために WAN 側インタフェースから出力される L2TP パケットを破棄しています。

＜ホスト名の設定＞

```
nxr130(config)#hostname NXR_C
```

ホスト名として「NXR_C」を設定します。

＜Ethernet0 インタフェース設定＞

```
NXR_C(config)#interface ethernet 0
NXR_C(config-if)#ip address 192.168.10.3/24
```

Ethernet0 インタフェースの IPv4 アドレスとして「192.168.10.3/24」を設定します。

＜IPv4 アクセスリスト設定＞

```
NXR_C(config)#ip access-list ppp0_in permit 10.10.10.1 any udp 500 500
NXR_C(config)#ip access-list ppp0_in permit 10.10.10.1 any 50
NXR_C(config)#ip access-list ppp0_out deny any 10.10.10.1 115
```

IPv4 アクセスリスト名を「ppp0_in」とし、以下のルールで設定します。

動作	送信元 IPv4 アドレス	宛先 IPv4 アドレス	プロトコル	送信元ポート	宛先ポート
許可	10.10.10.1	any	UDP	500	500
許可	10.10.10.1	any	50(ESP)	-	-

IPv4 アクセスリスト名を「ppp0_out」とし、以下のルールで設定します。
IPsec SA が確立する前に L2TPv3 のネゴシエーションおよび通信を防止するために WAN 側インタフェースから出力される L2TP パケットを破棄しています。

動作	送信元 IPv4 アドレス	宛先 IPv4 アドレス	プロトコル	送信元ポート	宛先ポート
破棄	any	10.10.10.1	115(L2TP)	-	-

＜IPsec アクセスリスト設定＞

```
NXR_C(config)#ipsec access-list NXR_A ip host host
```

IPsec アクセスリスト名を「NXR_A」とし、送信元 IPv4 アドレス「host」, 宛先 IPv4 アドレス「host」を設定します。これにより L2TPv3 パケットを IPsec でカプセル化します。

＜ppp0 インタフェース設定＞

```
NXR_C(config)#interface ppp 0
NXR_C(config-ppp)#ip address negotiated
NXR_C(config-ppp)#ip masquerade
NXR_C(config-ppp)#ip access-group in ppp0_in
NXR_C(config-ppp)#ip access-group out ppp0_out
NXR_C(config-ppp)#ip spi-filter
NXR_C(config-ppp)#ip tcp adjust-mss auto
NXR_C(config-ppp)#no ip redirects
NXR_C(config-ppp)#ppp authentication pap
NXR_C(config-ppp)#ppp username test3@centurysys password test3pass
```

ppp0 インタフェースに関する設定をします。
IPv4 アクセスリスト設定で設定した「ppp0_in」を ppp0 インタフェースの「in」フィルタに適用します。
IPv4 アクセスリスト設定で設定した「ppp0_out」を ppp0 インタフェースの「out」フィルタに適用します。

＜Ethernet1 インタフェース設定＞

```
NXR_C(config)#interface ethernet 1
NXR_C(config-if)#no ip address
NXR_C(config-if)#pppoe-client ppp 0
```

Ethernet1 インタフェースに関する設定をします。

＜スタティックルート設定＞

```
NXR_B(config)#ip route 0.0.0.0/0 ppp 0
```

デフォルトルートを設定します。

＜IPsec ローカルポリシー設定＞

```
NXR_C(config)#ip route 0.0.0.0/0 ppp 0
NXR_C(config)#ipsec local policy 1
NXR_C(config-ipsec-local)#address ip
NXR_C(config-ipsec-local)#self-identity fqdn nxrc
```

IPsec のローカルポリシー「１」を設定します。

＜IPsec ISAKMP ポリシー設定＞

```
NXR_C(config)#ipsec isakmp policy 1
NXR_C(config-ipsec-isakmp)#description NXR_A
NXR_C(config-ipsec-isakmp)#authentication pre-share ipseckey2
NXR_C(config-ipsec-isakmp)#hash sha1
NXR_C(config-ipsec-isakmp)#encryption aes128
NXR_C(config-ipsec-isakmp)#group 5
NXR_C(config-ipsec-isakmp)#lifetime 10800
NXR_C(config-ipsec-isakmp)#isakmp-mode aggressive
NXR_C(config-ipsec-isakmp)#remote address ip 10.10.10.1
NXR_C(config-ipsec-isakmp)#keepalive 30 3 periodic restart
NXR_C(config-ipsec-isakmp)#local policy 1
```

IPsec の ISAKMP ポリシー「１」を設定します。

事前共有鍵(Pre-Shared Key)として「ipseckey2」を設定します。

フェーズ１のネゴシエーションモードとして「aggressive」を設定します。

＜IPsec tunnel ポリシー設定＞

```
NXR_C(config)#ipsec tunnel policy 1
NXR_C(config-ipsec-tunnel)#description NXR_A
NXR_C(config-ipsec-tunnel)#negotiation-mode auto
NXR_C(config-ipsec-tunnel)#set transform esp-aes128 esp-sha1-hmac
NXR_C(config-ipsec-tunnel)#set pfs group5
NXR_C(config-ipsec-tunnel)#set sa lifetime 3600
NXR_C(config-ipsec-tunnel)#set key-exchange isakmp 1
NXR_C(config-ipsec-tunnel)#match address NXR_A
```

IPsec の tunnel ポリシー「１」を設定します。

使用する IPsec アクセスリストとして「NXR_A」を設定します。

＜ppp0 インタフェース設定＞

```
NXR_C(config)#interface ppp 0
NXR_C(config-ppp)#ipsec policy 1
```

IPsec ローカルポリシー「１」を適用します。これによりこのインタフェースが IPsec トンネルのエンドポイントとなります。

＜L2TPv3 設定＞

```
NXR_C(config)#l2tpv3 hostname nxrc
NXR_C(config)#l2tpv3 router-id 172.20.30.1
NXR_C(config)#l2tpv3 mac-learning
NXR_C(config)#l2tpv3 mac-aging 300
NXR_C(config)#l2tpv3 path-mtu-discovery
```

L2TPv3 のホスト名やルータ ID 等を設定します。

＜L2TPv3 トンネル設定＞

```
NXR_C(config)#l2tpv3 tunnel 1
NXR_C(config-l2tpv3-tunnel)#description NXR_A
NXR_C(config-l2tpv3-tunnel)#tunnel address 10.10.10.1
NXR_C(config-l2tpv3-tunnel)#tunnel hostname nxra
NXR_C(config-l2tpv3-tunnel)#tunnel router-id 172.20.10.1
NXR_C(config-l2tpv3-tunnel)#tunnel vendor ietf
```

L2TPv3 トンネル「１」を設定します。

＜L2TPv3 Xconnect 設定＞

```
NXR_C(config)#l2tpv3 xconnect 1
NXR_C(config-l2tpv3-xconnect)#description NXR_A
NXR_C(config-l2tpv3-xconnect)#tunnel 1
NXR_C(config-l2tpv3-xconnect)#xconnect ethernet 0
NXR_C(config-l2tpv3-xconnect)#xconnect end-id 1
NXR_C(config-l2tpv3-xconnect)#retry-interval 30
NXR_C(config-l2tpv3-xconnect)#ip tcp adjust-mss auto
```

L2TPv3 Xconnect「１」を設定します。

＜DNS 設定＞

```
NXR_C(config)#dns
NXR_C(dns-config)#service enable
```

DNS に関する設定をします。

9-1-3. パソコンの設定例

	LAN A のパソコン	LAN B のパソコン	LAN C のパソコン
IPv4 アドレス	192.168.10.100	192.168.10.101	192.168.10.102
サブネットマスク	255.255.255.0	255.255.255.0	255.255.255.0
デフォルトゲートウェイ	192.168.10.1	192.168.10.2	192.168.10.3

10. 付録

10-1. PPPoE 接続確認方法

PPPoE の接続状態は、「show ppp」コマンドで確認することができます。

実行例

```
nxr130#show ppp 0

PPP 0 session state is connected, line type is PPPoE, time since change 00:10:15
```

また PPPoE 接続時に利用する IP アドレスは「show interface」コマンドで確認することができます。

実行例

```
nxr130# show interface ppp 0
ppp0
          Link encap:Point-to-Point Protocol
          inet addr:10.67.15.1  P-t-P:10.255.0.1  Mask:255.255.255.255
          UP POINTOPOINT RUNNING NOARP MULTICAST  MTU:1454  Metric:1
          RX packets:4 errors:0 dropped:0 overruns:0 frame:0
          TX packets:4 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:3
          RX bytes:46 (46.0 b)  TX bytes:52 (52.0 b)
```

PPPoE 接続完了時には以下のようなログ(※)が表示されます。ログは「show syslog message」コマンドで確認できます。

```
nxr130 pppd[XXXX]: Plugin /etc/ppp/plugins/rp-pppoe.so loaded.
nxr130 pppd[XXXX]: RP-PPPoE plugin version 3.3 compiled againstpppd 2.4.4
nxr130 pppd[XXXX]: pppd 2.4.4 started
nxr130 pppd[XXXX]: PPP session is 1
nxr130 pppd[XXXX]: Using interface ppp0
nxr130 pppd[XXXX]: Connect: ppp0 <--> eth1
nxr130 pppd[XXXX]: Remote message: Login ok
nxr130 pppd[XXXX]: PAP authentication succeeded
nxr130 pppd[XXXX]: peer from calling number 00:80:6D:51:00:63 authorized
nxr130 pppd[XXXX]: local  IP address 10.67.15.1
nxr130 pppd[XXXX]: remote IP address 10.255.0.1
```

※PAP 認証時

10-2. フィルタの状態確認方法

フィルタの状態(アクセスリスト)を確認には、「show ip access-list」コマンドで確認することができます。

実行例

```
nxr130#show ip access-list
Chain +ppp0_forward-in (1 references)
 No.    packts    bytes target   prot       sourceIP            destIP  port
   1        0        0 permit    tcp      0.0.0.0/0        10.10.10.2  spt:any dpt:80
   2        6      767 permit    tcp      0.0.0.0/0        10.10.10.3  spt:any dpt:80
   3        2      124 permit    udp      0.0.0.0/0        10.10.10.4  spt:any dpt:53
   4        8      480 permit   icmp      0.0.0.0/0     10.10.10.0/29

Chain +ppp0_in (1 references)
 No.    packts    bytes target   prot       sourceIP            destIP  port
   1        4      240   deny   icmp      0.0.0.0/0        10.10.10.1  type:8 code:0
```

10-3. NAT の状態確認方法

DNAT/SNAT の状態を確認には、それぞれ以下のコマンドを実行することにより確認することができます。

DNAT 実行例

```
nxr130#show ip dnat
Chain *eth1_dnat (1 references)
 No.    packts    bytes   prot       sourceIP sport       destIP dport       DNAT
   1         1       52    tcp      0.0.0.0/0 any      10.10.10.1 80         192.168.10.10
```

SNAT 実行例

```
nxr130#show ip snat
Chain +eth2_snat (1 references)
 No.    packts    bytes   prot          sourceIP sport         destIP dport    SNAT
   1         2      120    all   192.168.10.0/24 any       0.0.0.0/0 any      10.10.10.1
   2         4      240    all   192.168.20.0/24 any       0.0.0.0/0 any      10.10.10.2
```

10-4. DHCP サーバによるリース状況確認方法

DHCP サーバによる IP アドレスのリース状況は、「show dhcp lease」コマンドで確認することができます。

実行例

```
nxr130#show dhcp lease
   [IP address]        [start]              [end]           [MACaddress]
 ------------------------------------------------------------------------------
 192.168.10.200    09/05/22 17:02:30    09/05/22 23:02:30    00:XX:XX:XX:XX:XX
```

10-5. IPsec 接続確認方法

IPsec の各トンネル状況を一覧で確認する場合は、「show ipsec status brief」コマンドを使用します。このコマンドでは IPsec SA が確立している(established)ものを「up」,それ以外を「down」と表示します。

実行例

```
nxr130#show ipsec status brief
TunnelName           Status
tunnel1              up
tunnel2              down
```

IPsec の SA 確立状況等を確認する場合は、「show ipsec status」コマンドを使用します。
また「show ipsec status」コマンドの後に「tunnel <ポリシー番号>」を指定することにより tunnel ポリシー毎にステータスを表示させることができます。これは多拠点構成で個々のポリシーを確認するのに有効です。

実行例

```
nxr130#show ipsec status
000 "tunnel1": 192.168.20.0/24===10.10.20.1[@nxrb]...10.10.10.1===192.168.10.0/24; erouted;
eroute owner: #2
000 "tunnel1":   newest ISAKMP SA: #1; newest IPsec SA: #2;
000
000 #2: "tunnel1" STATE_QUICK_I2 (sent QI2, IPsec SA established); EVENT_SA_REPLACE in 3213s;
newest IPSEC; eroute owner
000 #2: "tunnel1" esp.8d6d7079@10.10.10.1 (0 bytes) esp.c7d9d22b@10.10.20.1 (0 bytes); tunnel
000 #1: "tunnel1" STATE_AGGR_I2 (sent AI2, ISAKMP SA established); EVENT_SA_REPLACE in 10249s;
newest ISAKMP
000
```

ログ(※)では IPsec 接続完了時に以下のようなに表示されます。ログは「show syslog message」コマンドで確認できます。

ログ出力例

```
pluto[XXXX]: added connection description "tunnel1"
pluto[XXXX]: "tunnel1" #1: initiating Aggressive Mode #1,connection "tunnel1"
pluto[XXXX]: "tunnel1" #1: ignoring Vendor ID payload [strongSwan 4.2.9]
pluto[XXXX]: "tunnel1" #1: received Vendor ID payload [Dead Peer Detection]
pluto[XXXX]: "tunnel1" #1: Aggressive mode peer ID is ID_IPV4_ADDR: '10.10.10.1'
pluto[XXXX]: "tunnel1" #1: Aggressive mode peer ID is ID_IPV4_ADDR: '10.10.10.1'
pluto[XXXX]: "tunnel1" #1: sent AI2, ISAKMP SA established
pluto[XXXX]: "tunnel1" #1: Dead Peer Detection (RFC 3706): enabled
pluto[XXXX]: "tunnel1" #2: initiating Quick Mode PSK+ENCRYPT+TUNNEL+PFS+UP+0x8000000 {using
isakmp#1}
pluto[XXXX]: "tunnel1" #2: sent QI2, IPsec SA established{ESP=>0x8d6d7079 <0xc7d9d22b DPD}
```

※IPsec Aggressive mode Initiator log(IPv4)

10-6. L2TPv3 接続確認方法

L2TPv3 の情報を表示する場合は、「show l2tpv3」コマンドを使用します。

実行例

```
nxr130# #show l2tpv3

******* Global Information *******
 MAC Learning enable, LoopDetect disable, known-unicast drop
 RouterID is 172.20.10.1, Hostname is nxra
 snmp disable(disconnect) Trap disable

******* Interface Information *******
NumXconnectInterfaces 1
Interface name is ethernet0, Interface is up, link status is up
 LoopDetect is disable, known-unicast drop
 L2RP disable
          0 Frame Sent,          0 dropped,          0 errors
        294   received,          0 dropped,          0 known-unicast Frame


******* MAC Table Information *******
Interface ethernet0, NumMACs 0
MAC Table Entry is empty


******* FDB Information *******
attached Interface ethernet0, NumMACs 0
MAC Table Entry is empty
```

```
******* Group Information *******
NumL2TPGroups 1
Group ID 56988
  preempt is disable
  hold is disable
  Primary Xconnect     : PeerID(172.20.20.1), RemoteEND ID(1)
  Secondary Xconnect   : n/a
  Primary Session ID   : 115032585
  Secondary Session ID : n/a
  Active  Session ID   : 115032585


******* Tunnel/Session Information *******
NumL2TPTunnels 1
Tunnel MyID 1264197391 AssignedID 211192647 NumSessions 1 PeerIP 10.10.20.1 Stat
e established
 Session LAC(S) MyID 115032585 AssignedID 3821800887 State established
  Interface name is ethernet0, type is Ethernet
  Circuit state is UP (local is up, Remote is up)
  Group ID 56988, Group State is Active
            294 Packets sent,          0 dropped,          0 errors
              0     received,          0 dropped,          0 errors
```

また「show l2tpv3」コマンドで表示される項目のうち、一部の項目のみ表示させることも可能です。

以下はL2TPv3のセッションの確立状況を確認する「show l2tpv3 session」コマンドを使用した例になります。

なお「show l2tpv3 session」コマンドの後に「detail」を指定することによりより詳細なステータスを表示させることができます。

実行例

```
nxr130#show l2tpv3 session

Session Information Total tunnels 1 sessions 1

Tunnel MyID 1264197391 AssignedID 211192647
 Session LAC(S) MyID 115032585 AssignedID 3821800887 State established
  Interface name is ethernet0, type is Ethernet
  Circuit state is UP (local is up, Remote is up)
  Group ID 56988, Group State is Active
            289 Packets sent,          0 dropped,          0 errors
              0     received,          0 dropped,          0 errors
```

L2TPv3 接続完了時には以下のようなログ(※)が表示されます。ログは「show syslog message」コマンドで確認できます。

```
l2tpv3[XXXX]: L2TP Session Established
l2tpv3[XXXX]:   Peer IP = 10.10.20.1
l2tpv3[XXXX]:   Peer ID = 172.20.20.1
l2tpv3[XXXX]:   Remote END ID = 1
l2tpv3[XXXX]:   Local  Tunnel/Session ID = 3255229736/3984545304
l2tpv3[XXXX]:   Remote Tunnel/Session ID = 110781248/2309376460
```

11. サポートデスクへのお問い合わせ

11-1. サポートデスクへのお問い合わせに関して

サポートデスクにお問い合わせ頂く際は、以下の情報をお知らせ頂けると効率よく対応させて頂くことが可能ですので、ご協力をお願い致します。

- ご利用頂いている NXR 製品の機種名,バージョン番号
- ご利用頂いている NXR 製品を含んだネットワーク構成
- 不具合の内容および不具合の再現手順（何を行った場合にどのような問題が発生したのかをできるだけ具体的にお知らせ下さい）
- ご利用頂いている NXR 製品での不具合発生時のログ（show syslog message）
- ご利用頂いている NXR 製品の設定ファイル,「show tech-support」コマンドの実行結果

11-2. サポートデスクのご利用に関して

電話サポート

電話番号：０４２２－３７－８９２６

電話での対応は以下の時間帯で行います。

月曜日 ～ 金曜日 10:00 AM - 5:00 PM

ただし、国の定める祝祭日、弊社の定める年末年始は除きます。

電子メールサポート

E-mail： support@centurysys.co.jp

ＦＡＸサポート

FAX 番号：０４２２－５５－３３７３

電子メール、FAX は 毎日 24 時間受け付けております。

ただし、システムのメンテナンスやビルの電源点検のため停止する場合があります。 その際は弊社ホームページ等にて事前にご連絡いたします。

FutureNet NXR シリーズ

設定例集

Ver 1.2.0

2009 年 9 月

発行　センチュリー・システムズ株式会社