

Vagrant で試験 ネットワーク構築して ハマった話

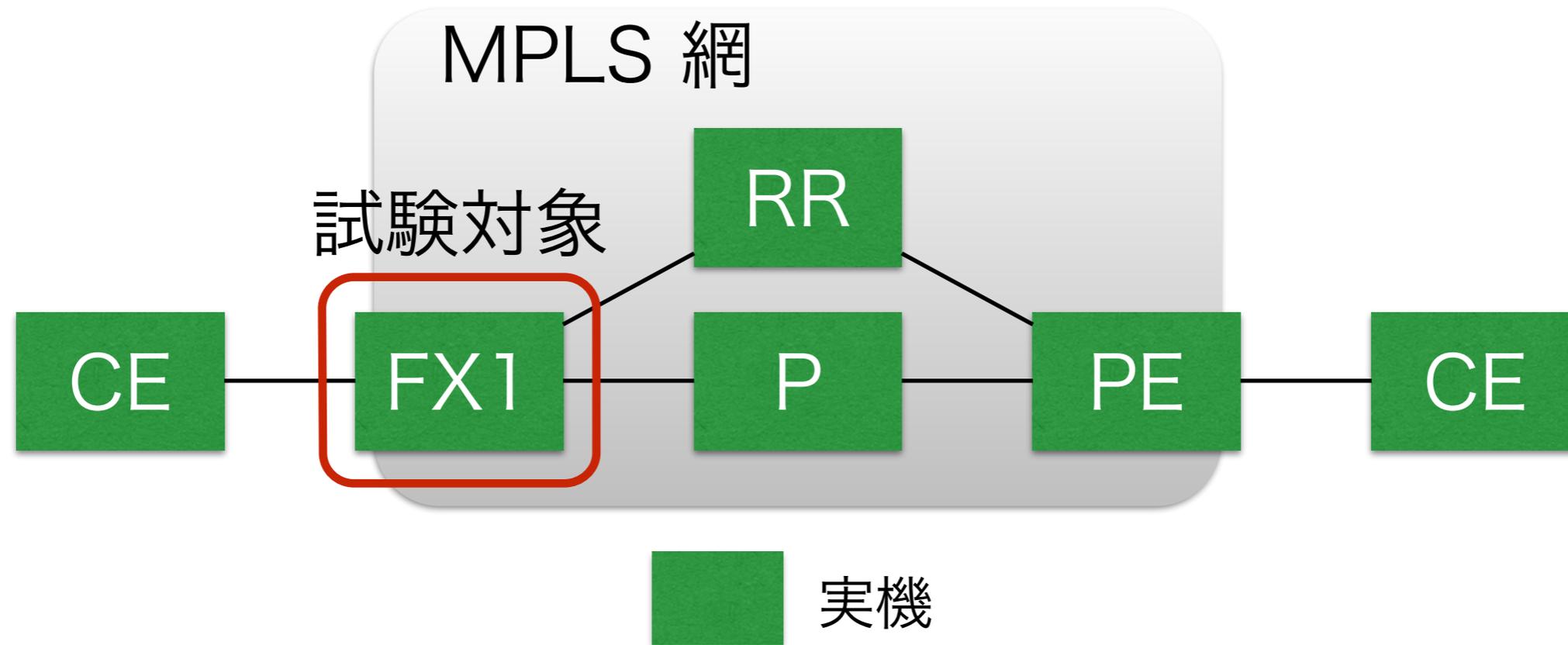
古河ネットワークソリューション株式会社
神谷 尚秀

自己紹介

- ・ 名前
 - ・ 神谷 尚秀 (かみたに なおひで)
- ・ 所属
 - ・ 古河ネットワークソリューション株式会社
- ・ 仕事内容
 - ・ 2015/7 までルータのソフトウェアを開発
 - ・ 現在はマーケティング、製品企画など
- ・ コミュニティ活動
 - ・ JANOG 運営委員
 - ・ wakamonog 運営委員

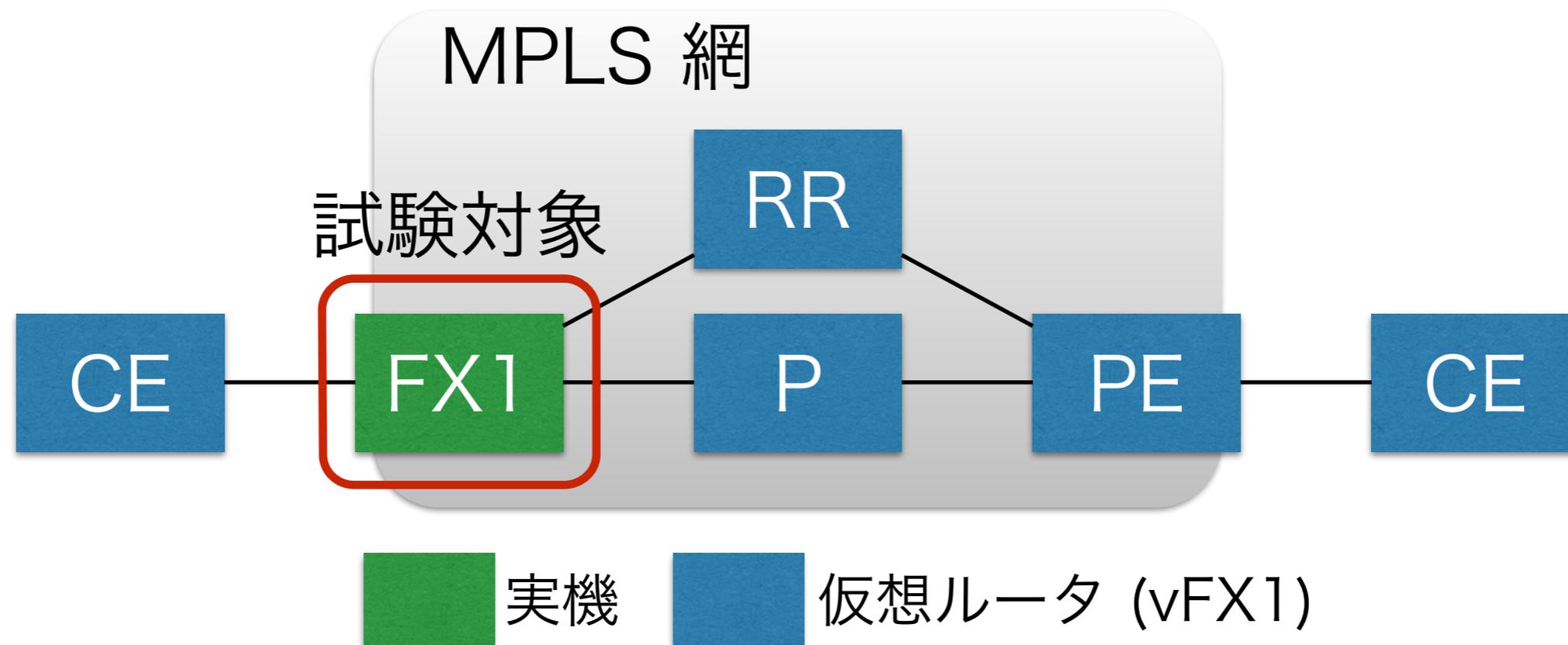


試験ネットワークの構築



- ・ ネットワークを全て実機で構築するのは難しい。。
- ・ 開発者 1 人につき実機 6 台なんて割り当てられない
- ・ コントロールプレーンの試験であれば**仮想ルータ**が利用できる

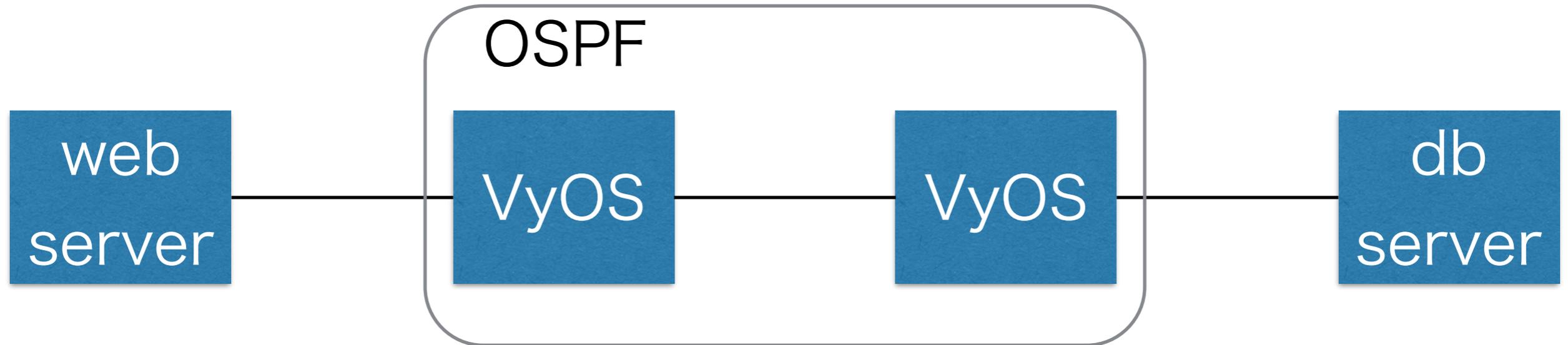
試験ネットワークの構築



- ・ 仮想ルータ (vFX1) を利用して試験ネットワークを構築
 - ・ vFX1 は FX1 のコントロールプレーン機能と同等の機能を保持
- ・ Virtualbox で vFX1 を起動するため以下を**手動**で設定
 - ・ vFX1 のクローン, リソース割当, ネットワークアダプタ割当

vFX1 の起動を自動化したい！

Vagrant で自動構築できそう



- ・ wakamonog meeting で @kakky0312 さんが検証環境を自動構築してた
- ・ Vagrant+Chef+Vyatta でお手軽検証環境構築

<https://github.com/kakky0312/vagrant-init>

Vagrant で自動構築 やってみた

box 作成

box 作成手順

1. 仮想マシンの設定 *

- ・ user/password の設定
- ・ ssh の設定
- ・ dhcp client の設定
- ・ sudo の設定

— vFX1 での設定例 —

```
#password login vagrant
#configure terminal
(config)#hostname vFX1
(config)#ssh-server version 2
(config)#interface management 1
(config-if-manage 1)#ip address dhcp
(config-if-manage 1)#end
#save boot.cfg
```

2. 仮想マシンを停止

3. vagrant package で box 作成

仮想ルータは CLI で設定管理
するため、sudo が必要ない

* <http://docs.vagrantup.com/v2/boxes/base.html>

※box とは vagrant で仮想マシンをクローンするためのベースイメージ

vagrant up でエラー続出

```
bash-3.2$ vagrant up
```

```
Bringing machine 'rtr1' up with 'virtualbox' provider...
```

```
(... snip ...)
```

The configured shell (config.ssh.shell) is invalid and unable to properly execute commands. The most common cause for this is using a shell that is unavailable on the system. Please verify you're using the full path to the shell and that the shell is executable by the SSH user.

```
bash-3.2$
```

shell (bash) を vFX1 で実行しようとしてエラー
=> shell を実行させないようにして解決

vagrant up でエラー続出

```
bash-3.2$ vagrant up
(… snip …)
==> rtr1: Configuring and enabling network interfaces...
The following SSH command responded with a non-zero exit status.
Vagrant assumes that this means the command failed!
sed -e '/^#VAGRANT-BEGIN/, $ d' /etc/network/interfaces > /tmp/
vagrant-network-interfaces.pre
Stdout from the command:
Stderr from the command:
sudo -E -H
    ^^
% Invalid input detected at '^' marker.
bash-3.2$
```

ホストオンリーアダプタ設定時に sudo を実行してエラー
=> sudo を実行させないようにして解決

vagrant up でエラー続出

```
bash-3.2$ vagrant up
Bringing machine 'rtr1' up with 'virtualbox' provider...
(… snip …)
==> rtr1: Mounting shared folders...
    rtr1: /vagrant => /Users/naohide/vFX1_01
/opt/vagrant/embedded/gems/gems/vagrant-1.7.2/plugins/guests/
linux/cap/shell_expand_guest_path.rb:15:in
`shell_expand_guest_path': undefined method `split' for nil:NilClass
(NoMethodError)
(… snip …)
bash-3.2$
```

共有フォルダを有効にしようとしてエラー
=> 共有フォルダを無効にすることで解決

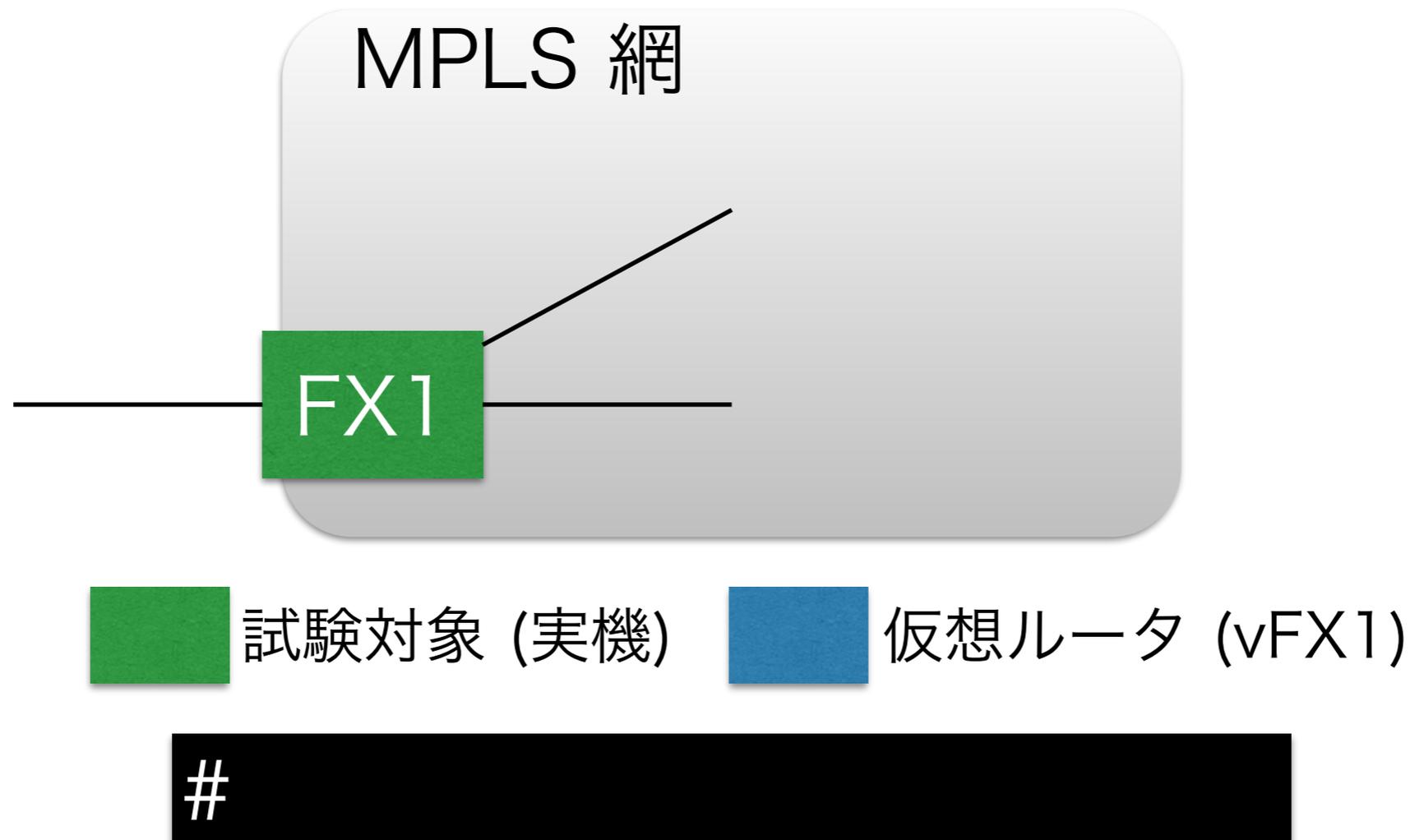
vagrant up エラーの解決方法

```
bash-3.2$ cat Vagrantfile
# -*- mode: ruby -*-
# vi: set ft=ruby :
Vagrant.configure(2) do |config|
  config.vm.box = "vFX1"
  config.vm.define "rtr1" do |rtr|
    rtr.vm.network "private_network", ip: "x.x.x.x", auto_config: false
  end
  config.ssh.shell = ""
  config.vm.synced_folder ".", "/vagrant", disabled: true
end
```

Vagrantfile を工夫する
ことで解決

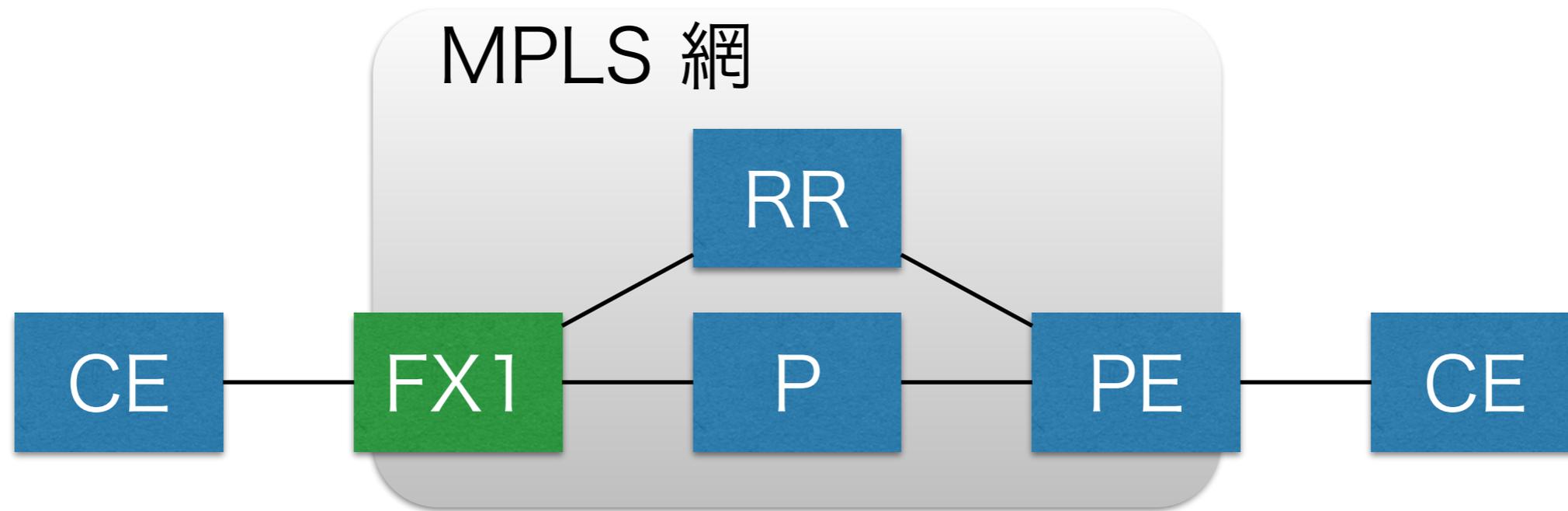
※各仮想ルータのコンフィグ設定は手動で入力して対応

試験ネットワークの構築



- ・ Vagrant により vFX1 の起動・リソース割当・ネットワークアダプタの設定まで自動化、構築工数を減らすことができた

試験ネットワークの構築



■ 試験対象 (実機) ■ 仮想ルータ (vFX1)

```
# vagrant up
```

- ・ Vagrant により vFX1 の起動・リソース割当・ネットワークアダプタの設定まで自動化、構築工数を減らすことができた

ちなみに

- ・ 他の仮想ルータでも vagrant up で同じエラーが発生
- ・ Vagrantfile を工夫する解決方法で起動確認できた仮想ルータ・ファイアウォール
 - ・ vSRX (Firefly) *
 - ・ (SEIL/x86 でもできるはず [無償版では config を保存できず box 作成ができなかった])
- ・ 本解決方法は **shell/sudo を実行できない、共有フォルダを使用できない**仮想マシンで適用可能

* vSRX は本解決方法で無くても、公開されている box を使い、Vagrant の plugin をインストールすることで vagrant up が可能 (<https://atlas.hashicorp.com/juniper>)

まとめ

- ・ Vagrant で仮想ルータの起動・リソース割当・ネットワークアダプタの設定を自動化
 - ・ 今後、環境構築工数の削減、共通の環境で試験することによる環境構築ミスをなくした試験品質の向上を目指して、仮想ルータのコンフィグ設定を含めたプロビジョニング自動化を実現予定
- ・ Vagrantfile を工夫することで vagrant up 時のエラーを解決
 - ・ shell / sudo が実行できない、共有フォルダを使用できない仮想ルータ・ファイアウォールでも Vagrant を使用できることを確認
- ・ ネットワーク自動構築でもっといい方法あるよという方、@NaohideKamitani まで是非コメントよろしくお願いします_o_