

JANOG35.5 WhiteBox SW検証

～サーバサイド編～

GREE株式会社 中野和貴



おまえだれよ

- 中野 和貴 (なかの かずき)
 - FB, twitter: kakky0312
- 業務内容
 - クラウド環境へのサーバ移行サポート
 - ネットワーク業務のお手伝い
 - 運用ツールの開発
- 趣味 / 好きなこと
 - 飲み、お店開拓
 - いいお店あったら教えて下さい(・∀・)
 - ドライブ
 - MT車が欲しい今日この頃
 - 音楽を聞いたり演奏したり



- OSが基本的に通常のLinuxがベースとなっている
 - 自分の好きなパッケージが入れられる
 - 通常のLinuxサーバと同じように扱う事ができる
- 筐体とOSの組み合わせを自分たちで選択できる
 - ONIE (Open Network Install Environment)に対応している筐体とOSであれば好きな様に組み合わせで使える
 - 利用OSはそのままより安い筐体を選べる費用面のメリット
 - 運用ナレッジ集約のメリット

- ONIEを利用したOSインストール
 - サーバのPXEブート相当なインストールが実現できるか
 - 実現できれば、一度に多くのSWへOSインストールを楽に出来る
- Chefを使った設定変更
 - サーバと同様に多くの機器へ同時に設定を流し込める下地があるか
 - 可能であれば、サーバ運用でのナレッジをある程度適用出来る
- OpenStackコンポーネントのインストール
 - スイッチ自体にネットワークノードを導入することが出来るか
 - 可能ならば、パケット折り返しを減らす事が可能ではないか

ONIEを利用したOSインストール

ONIEとは

- OCP (Open Compute Project)が提唱するOSインストールの規格
- HTTPやTFTP、USBメモリ経由でのOS初期導入とアップデートが可能
- Busyboxで筐体ブート、そこからOSインストールを実施する

Busybox

Busyboxで起動し、OSインストール準備状態に

DHCPでIPを取得、HTTP
でOSイメージを取得

Get Image

取得したOSイメージを使って、OSインストール

OS Install

今回試したこと

- イメージ取得用サーバを準備
 - DHCPとHTTPが使えるようにdhcpdとnginxをインストール
 - DHCPには以下の設定を入れる
 - IPアドレスの取得設定
 - next-hop設定にHTTP経由でOSイメージを取得するように
- 一度、筐体からOSを削除する
- 再度筐体を起動させ、OSインストールを実施
- インストール後に正常に起動できるかを確認

結果

- 正常にDHCPからIPを取得し、OSインストールまで出来た
- 規格に沿った形で実装されているため、OSメーカーのドキュメント通りに設定することで大きな問題もなくすんなり入った

```
Welcome to GRUB!                ONIE: OS Install Mode ...

GRUB loading.                      Version : 2014.08.0.0.3
Info: Mounting kernel filesystems... done.
Info: Mounting LABEL=ONIE-BOOT on /mnt/onie-boot ...
Running Celestica Redstone-XP platform init post_arch routines...
Info: Using eth0 MAC address: xx:xx:xx:xx:xx:xx
Info: eth0: Checking link... up.
Info: Trying DHCPv4 on interface: eth0
ONIE: Using DHCPv4 addr: eth0: 90.90.90.25 / 255.255.255.0
(省略)
Info: Fetching http://90.90.90.5/CumulusLinux-2.2.2-amd64.bin ...
ONIE: Executing installer: http://90.90.90.5/CumulusLinux-2.2.2-amd64.bin
Verifying image checksum ... OK.
Preparing image archive ... OK.
```


Chefを使った設定変更

Chefとは

- サーバ構築を自動化するフレームワーク
- 設定ファイルを記述しそれを実行することで、あらかじめ定義した状態へサーバを自動的に構築することが出来る
 - ユーザ追加/削除作業
 - パッケージインストール作業
 - コンフィグファイルの配置/編集
 - etc...
- 今回はサーバクライアント型のchefではなく、クライアント単体で動くchef-soloを利用した
- Version 11.6.2



今回試したこと

- ログインユーザの追加/削除
 - greeというログインユーザの追加及び削除
- NTPサーバの変更
 - /etc/ntp.confファイルを編集して利用するNTPサーバを変更
- ネットワークインターフェース操作
 - 新規にVLANとSVIを作成
 - SVIへIP Addressを設定
 - 物理ポートへVLANをアサイン
 - pingが通るようになったことを確認

結果 (ユーザ追加/削除)

- 問題なく追加と削除ができ、ログイン可不可が変わる事を確認できた

```
cumulus@cumulus-3$ sudo chef-solo -c solo.rb -o useradd
```

```
Starting Chef Client, version 11.6.2
```

```
[2001-04-01T22:06:41+00:00] WARN: Run List override has been provided.
```

```
[2001-04-01T22:06:41+00:00] WARN: Original Run List: []
```

```
[2001-04-01T22:06:41+00:00] WARN: Overridden Run List: [recipe[useradd]]
```

```
Compiling Cookbooks...
```

```
Converging 2 resources
```

```
Recipe: useradd::default
```

- * user[gree] action create
 - alter user user[gree]
- * user[gree] action manage (up to date)
- * user[cumulus] action manage
 - manage user user[cumulus]

```
Chef Client finished, 2 resources updated
```

結果 (NTPサーバの変更)

- ntp.confを指定のものに置き換えられたことを確認

```
cumulus@cumulus-3$ sudo chef-solo -c solo.rb -o ntp  
(中略)
```

```
Recipe: ntp::default
```

```
* service[ntp] action nothing (skipped due to action :nothing)  
* cookbook_file[/etc/ntp.conf] action create  
  - update content in file /etc/ntp.conf from fa1a33 to 142b4d  
    --- /etc/ntp.conf  2014-11-15 17:38:08.000000000 +0000  
    +++ /tmp/.ntp.conf20010401-13643-1ckz0em  2001-04-01 21:41:43.936993945 +0000  
    @@ -18,10 +18,9 @@  
    -server 0.cumulusnetworks.pool.ntp.org iburst  
    -server 1.cumulusnetworks.pool.ntp.org iburst  
    -server 2.cumulusnetworks.pool.ntp.org iburst  
    -server 3.cumulusnetworks.pool.ntp.org iburst  
    +server ntp1.jst.mfeed.ad.jp  
    +server ntp2.jst.mfeed.ad.jp  
    +server ntp3.jst.mfeed.ad.jp  
    # Access control configuration; see /usr/share/doc/ntp-doc/html/acconf.html for  
* service[ntp] action restart  
  - restart service service[ntp]
```

```
Chef Client finished, 2 resources updated
```

結果 (ネットワークインターフェース操作)

- 作成したスクリプトをChef経由で実行しVLAN追加/削除が出来た
- スクリプトはOSのドキュメントに指定された操作コマンドの羅列
 - ipコマンドやbrctlコマンド

```
cumulus@cumulus-3$ sudo chef-solo -c solo.rb -o set-vlan
Starting Chef Client, version 11.6.2
[2001-04-01T21:15:24+00:00] WARN: Run List override has been provided.
[2001-04-01T21:15:24+00:00] WARN: Original Run List: []
[2001-04-01T21:15:24+00:00] WARN: Overridden Run List: [recipe[set-vlan]]
Compiling Cookbooks...
Converging 2 resources
Recipe: set-vlan::default
  * cookbook_file[/home/cumulus/set-vlan-30.sh] action create (up to date)
  * bash[set-vlan-30.sh] action run
    - execute "bash" "/tmp/chef-script20010401-3735-b6lp9r"
Chef Client finished, 1 resources updated
```

結果 (ネットワークインターフェース操作)

- 一時的な操作に関してはスクリプトを流し込むで対応できる
- しかし、設定の永続性は担保されない
 - networkingデーモンのrestartで設定が消えてしまう
- 永続的な設定は/etc/network/interfacesファイルに記載が必要
- スクリプトでinterfacesファイルを編集するのは怖い
 - sed等での置き換え時に他のところが変に引っかかってしまうかも
 - 単純にそういう編集の仕方はプログラムの実装が面倒くさい
- インターフェース情報はDB等で管理して、毎回そこからChef経由で動的生成してファイルを置き換えてあげるほうが良さそう

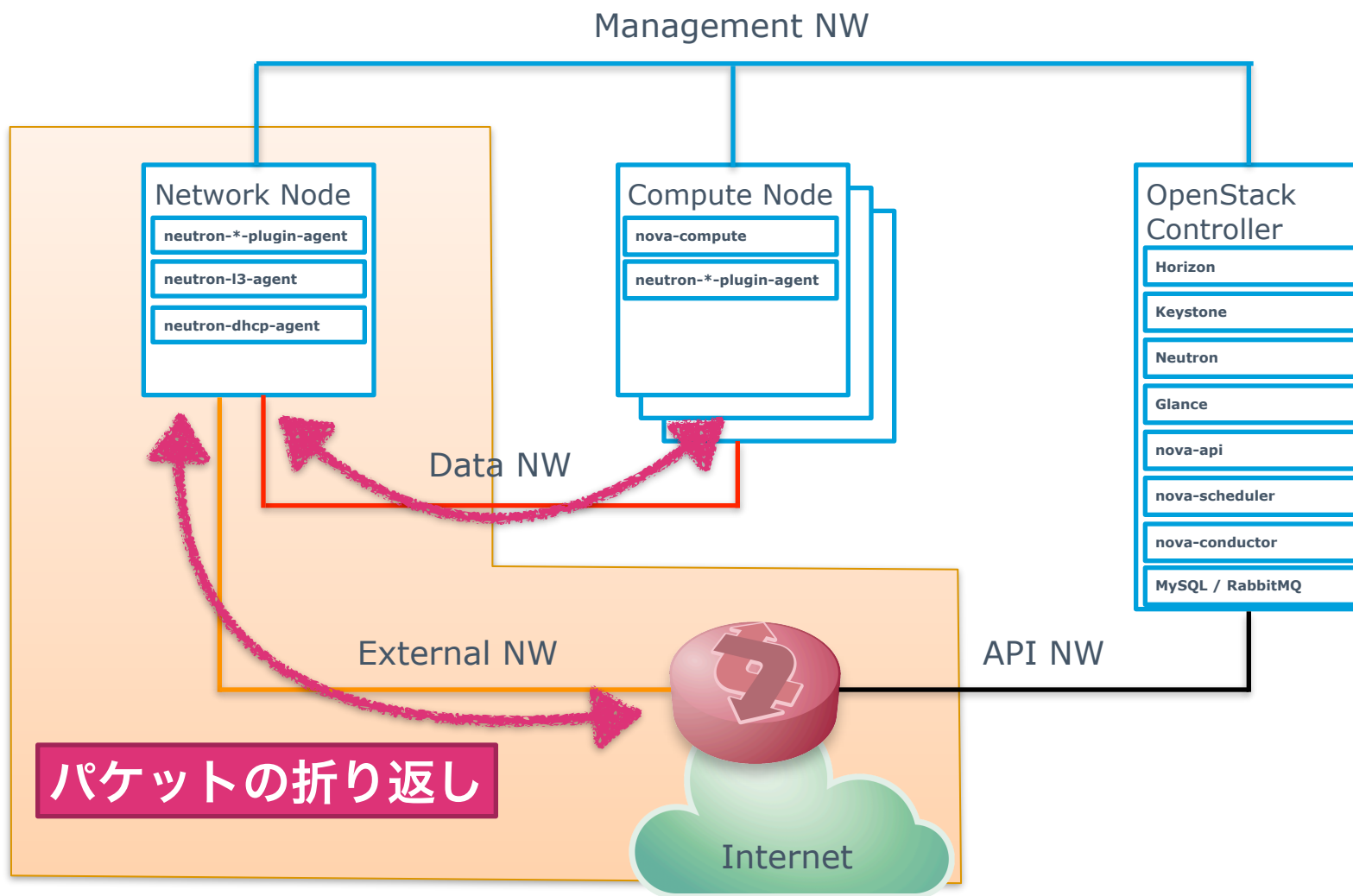
OpenStackコンポーネントのインストール

今回試したこと

- OpenStackのネットワークノードをスイッチ内にインストール
 - OpenStack内のIP割り振りや通信に関するノード
 - フォワーディングやNAT通信がここを通る事で外部ネットワークとアクセス可能になる
- 通常はサーバに構築するものをスイッチ上に直接構築することでパケットの折り返しを減らし、高速化が図れるのかを検証
- 利用したバージョンはIcehouse



今回試したこと



結果と思う所

- インストール自体が出来なかった
 - kernelが独自ビルドされたものであり、依存ライブラリのバージョンがベースOSであるWheezyと結構違った
 - そのため必要なライブラリをWheezyのレポジトリを追加してaptからインストールしようとしたが出来なかった
- ソースからビルドすればインストール出来た可能性はある
 - 今回検証時間がそれほど長く無かったためそこまでは断念
- 今後ベンダーからOpenStackパッケージが提供されてくればそれが一番幸せで確実

まとめ

- OSインストールに関して
 - 特に大きな問題も無く、インストールのやり方も何個かあるので会社の運用上適したものを選べば問題無い
- Chefを使った自動構築
 - 基本的にちゃんと動くが、Chef単体で全てをやるのは辛そう
 - 適宜DB等他ツールと連携させて使うとより捗るのではないかな
- OpenStackコンポーネントインストール
 - 自力で入れるのは難易度が高い
 - OSベンダーから提供されるようリクエストするのが吉
- 全体を通して
 - HWがPowerPC版か、x86版かでも問題の出方が違ったりしそう
 - 今回検証したような使い方をするにはそれなりにサーバサイドのナレッジは必要であり導入コストはそれなりに高い