卒業研究参考資料　Heatテンプレートファイル記述方法について

横山研究室

情報学群４年

1160304

川口貴大

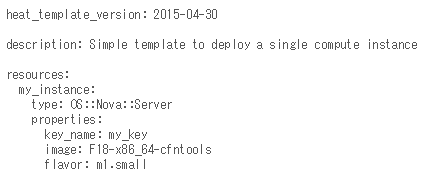
1. Heatとは

Heatとは、IaaSクラウドコンピューティングプロジェクトであるOpenStack内で提供されている機能の一つである。OpenStackとは、仮想instanceの立ち上げ、仮想networkの構築、仮想Routerの作成等々、仮想環境を立ち上げるためのもの、それらをコントロールするものである。それぞれの機能は、提供されるサービスが異なる。instanceの立ち上げではNovaというサービスを利用し、networkの立ち上げではNeutronというサービスを利用する。

Heatは、それらの機能を一括して自動で立ち上げるためのテンプレートファイルを記述し、テンプレートファイルを読み込ませるだけで仮想アーキテクチャの構築を可能にするものである。

1. テンプレートファイル内の記述について

まず、例として最小構成のテンプレートファイルで解説を行う。以下に示しているのが、instanceを一つのみ構築する最小構成のテンプレートファイルである。



このテンプレートファイル内の一番上「heat\_template\_version:2015-04-30」というのがHeatテンプレートファイルのversionである。この記述が無いとHeatテンプレートファイルとしてHeatが認識できない。descriptionの行は、説明文である。通常のコードでいうコメント部分に相当する。descriptionは入力しなくても問題ない項目である。

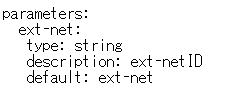
* resources

使用するリソースたちの項目である。今回はinstanceを１つのみ使用するのでmy\_instanceの記述のみである。本来ならば使用するRouterやinstanceが複数記入される。Instanceについての記入をするのならば、type選択で「OS::Nova::Server」を選択する。Type選択の上にある項目「my\_instance」というのがHeat内での識別名である。他のResources等からこのinstanceを指定する場合は「my\_instance」と記入することで指定することができる。

* properties

詳細について入力される項である。ここに、どのimage（OS選択）を使うか、flavorはどれなのか（instanceのスケール。大きければそれだけメモリ等の容量も大きくなる）を記入する。この例だと、「F18-x86\_64-cfntools」と呼ばれるimageを選択しており、flavorはsmallである。

次に、ネットワークとルーターの記述について説明する。以下に示しているのが外部ネットワーク、内部ネットワーク（サブネット）、ルーター、ルーターインターフェースについての例である。基本的に外部ネットワークに関する項目は一つのみだが、その他の内部ネットワークやルーター、ルーターインターフェースは１セットとしてまとめて取り扱う。ルーターが一つ増える毎に、外部ネットワーク以外の項目はそのまま１つずつ新しいものが増えることになる。



* parameters

今回の実験では、外部ネットワークの指定にのみ使用される。用意されている外部ネットワークは「ext-net」のみなので、ext-netについて記述する。この項に関してはどのテンプレートファイルを作成する場合でも丸写しで構わない。

* resources

instanceの説明時に説明したものと同じである。本来であれば、上記例のようにRouterやネットワークを指定した下に、そのままinstanceについての記述が続く。基本的にheatテンプレートファイルでは最初にparameterで外部ネットワークの指定をした後は、内部ネットワーク→Router→instanceの順番に記述を行っていく。

* private1

（内部の）ネットワーク。Type指定は「OS::Neutron::Net」とする。これは、OpenStack内でネットワークを管理しているのがNeutronだからである。ネットワークに関する記入であることを明らかにするためにNetと指定する。このネットワークのサブネットに関しては、以下の項目で記入されていく。

* private1-sub

ネットワークであるprivate1のサブネット部分、あるいは詳細入力部分である。サブネット部分であるため、type指定の末尾にはsubnetと記入しておく。

* depends\_on

依存するresourcesを指定する。Private1-subはprivate1のサブネットであり、private1に依存することになる。

* properties

サブネット中の詳細入力項目のことである。サブネット中で入力する項目は以下の２つである。

* + network\_id

自身が依存するprivate1のネットワークIDを取得する。{get\_resource: ネットワーク名 }で指定可能。

* + cidr

このネットワークで取り扱うIPアドレス範囲のこと。複数Routerを使用するのであれば、被りがないようにしなければならない。

* ext-router1

内部にprivate1を持つことになるRouterのことである。Type指定は末尾にRouterを指定。この項目では、内部に広がるネットワークであるprivate1が、どの外部ネットワークへ出て行くのかを記入する。

* properties
  + external\_gateway\_info

Gatewayの情報について記入する。Gatewayに当たる部分、つまり外部ネットワークを指定することになる。{get\_param: ext-net}とすることで外部ネットワークをGatewayとして指定可能である。

* ext-router1-interface

この項目には、Router-interfaceがどの要素と依存しているか、依存している物のIDを記入することになる。

* depends\_on

このinterfaceの大元であるext-routerと、サブネットであるprivate1-subを指定する。

* properties

依存するresourcesたちのIDを取得する。取得法は前述していたものと同じget\_resourceである。router\_idとsubnet\_idを取得する。

1. テンプレートファイル作成後

テンプレートファイル作成後にheatへテンプレートファイルを読みこませる。無事にテンプレートファイルが読み込まれ、記入した通りの構成が立ち上がれば作業終了である。尚実験ではテンプレートファイルの読み込み中は作業時間（所要時間）としてはカウントしない。また、テンプレートファイルを読みこませるときは川口が立ち会う。