

Puppet x Ansible

Mas afinal, qual é a diferença entre *Puppet* e *Ansible*?

Podemos resumir de alguns modos:

- O Ansible é uma ferramenta principalmente de **provisionamento**, ou seja, é utilizado para fornecermos ferramentas e preparar nosso ambiente para determinada tarefa.
- Outro fato sobre o Ansible é que tudo que escrevemos em nossos `playbooks` é convertido em código *python*. O que significa que devemos ter o python instalado nas máquinas em que o `playbook` será executado.
- Os `playbooks` devem ser executados em cada máquina desejada para execução do serviço, ou seja, para cada vez que desejarmos fazer um novo **provisionamento** para as máquinas, precisamos executar o `playbook` novamente.
- O Puppet, é uma ferramenta de **gerenciamento de configuração**, ou seja, utilizamos o Puppet para definir e manter as configurações de nosso ambiente.
- Com o Puppet, utilizamos arquivos de `manifest` para definir como será feita e estabelecida a configuração das máquinas que rodarão o *puppet-agent*. Para que isso funcione, devemos ter o *puppet-agent* instalado em todas as máquinas que serão gerenciadas pelo Puppet, e o *puppet-server* na máquina que será a provedora de configurações.
- Uma vez definido como as máquinas serão configuradas, executamos o comando para que as máquinas com o *puppet-agent* comecem a seguir as configurações especificadas em nosso arquivo `manifest`.

Concluindo: o Puppet é uma ferramenta de **gerenciamento de configuração** e o Ansible é uma ferramenta de **provisionamento**, ou seja, utilizamos o Puppet para validar a configuração de nosso ambiente e o Ansible para instalar e preparar o ambiente. Mas como assim, isso não seria **provisionamento** para os dois casos? Na verdade, **não** .

Vamos ver um exemplo:

Temos uma máquina e devemos construir o ambiente para nosso trabalho. Como queremos definir as configurações iniciais de uma máquina, seria interessante **provisioná-la** inicialmente, já que sequer temos o que manter de configuração. Depois de definido o ambiente, precisamos manter essas configurações. Caso algum programa ou arquivo seja removido, queremos que o estado da máquina seja restaurado para o estado original, sem afetar o funcionamento . Para garantirmos que isso aconteça, podemos utilizar o **gerenciamento de configuração** do Puppet, que consegue manter a máquina no estado padrão sem que ninguém precise reexecutar o arquivo de `manifest` . O Puppet faz essa verificação de configuração com intervalo customizável, chamamos isso de **self-healing**.