

Utilize um upgrade do Ansible

Eleve sua experiência de mainframe com o Ansible

6 etapas 30 min

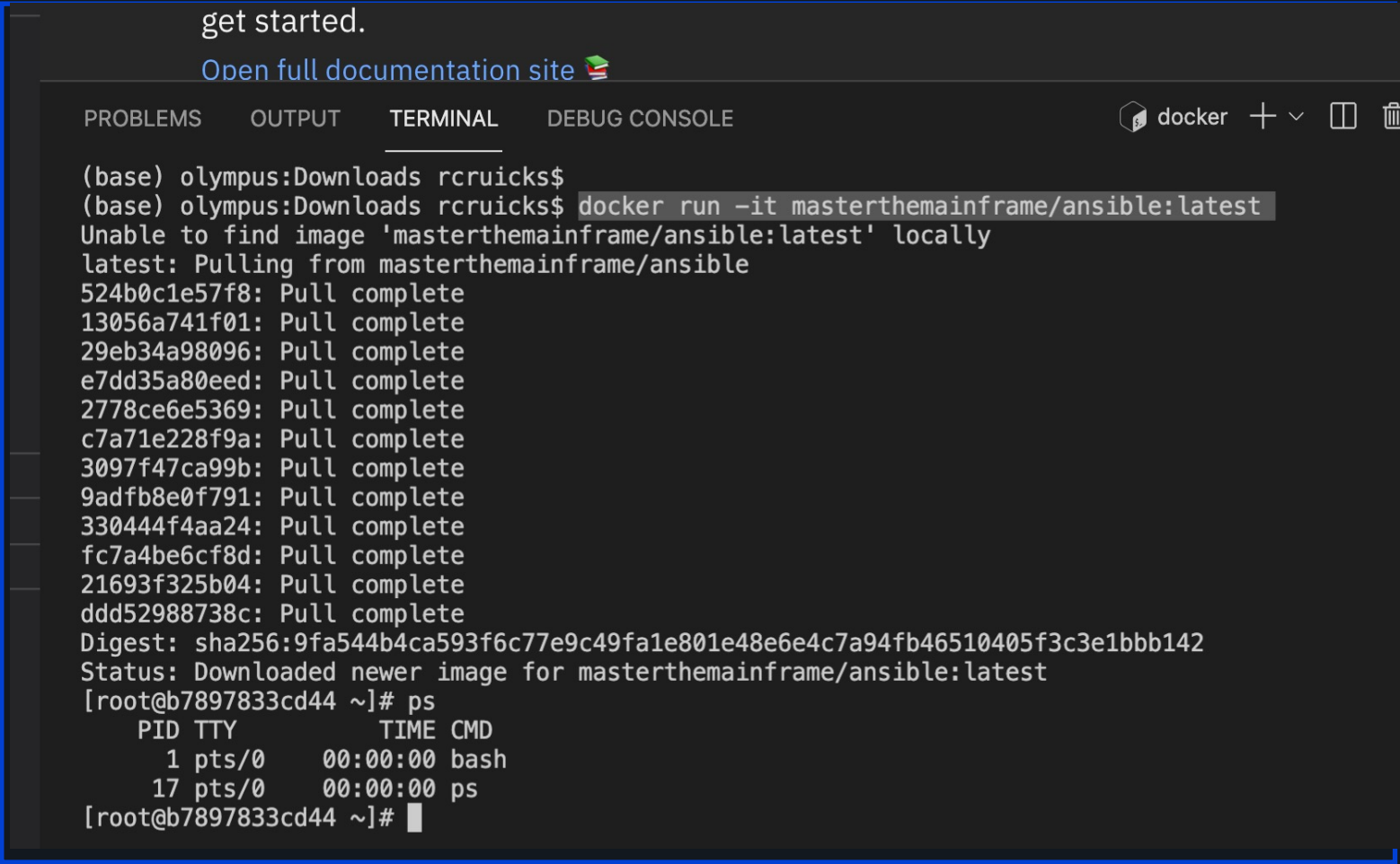
O DESAFIO

Agora é possível testar o Ansible, uma ferramenta que reúne tudo para que seja possível se concentrar no que deve ser feito. Isso é algo bem avançado e você está prestes a obter experiência prática ao utilizar isso no sistema.

ANTES DE COMEÇAR

Deixe os do Ansible por último.

Para realmente aproveitar o Ansible ao máximo, saiba o que está disponível para você EM TODOS OS LUGARES da plataforma.

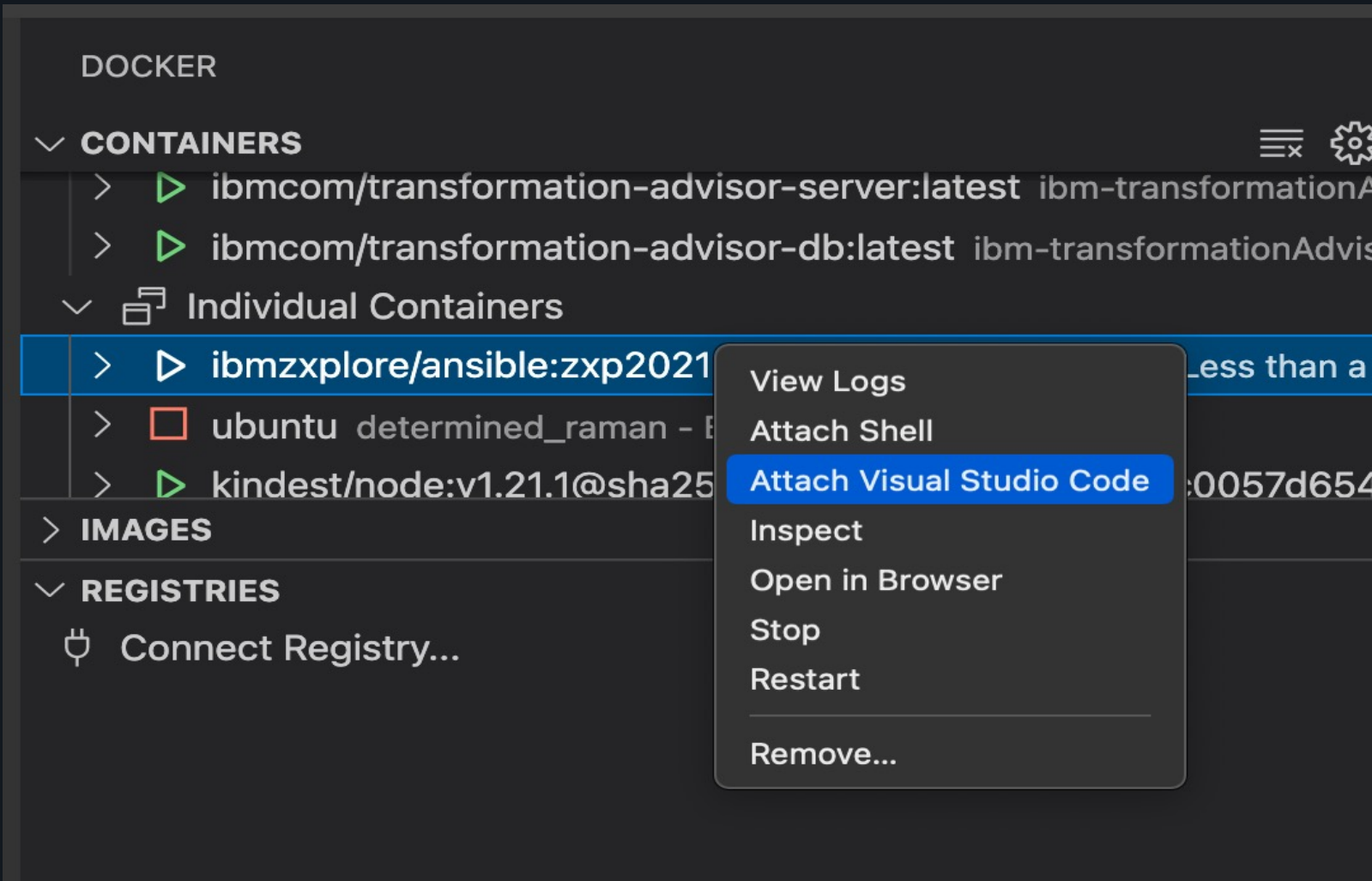


1. CARREGUE A IMAGEM DO ANSIBLE

No desafio do Docker, você configura a extensão do Docker para o VS Code e a conecta a um contêiner básico do Ubuntu.

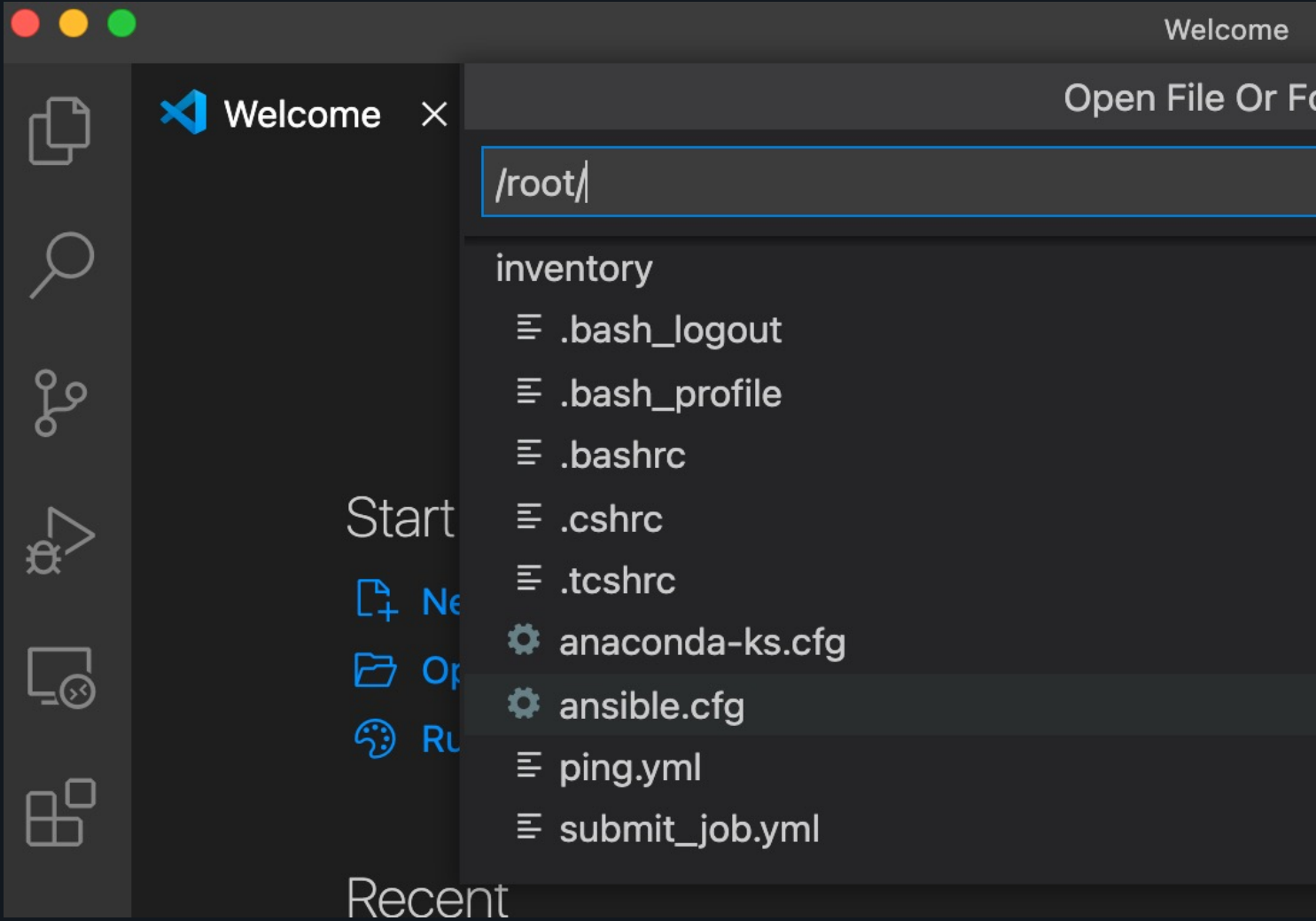
Agora é possível começar a trabalhar com o Ansible utilizando **docker run -it ibmzxplore/ansible:zxp202108**

É possível ver o Docker extrair automaticamente uma cópia da imagem para o seu tempo de execução do Docker e iniciar o contêiner.



2. CONECTE O CONSOLE DO VS CODE

Encontre o ícone no lado esquerdo do Explorador Remoto e, nele, seu contêiner do Ansible que acabamos de extrair. Clique com o botão direito e selecione "Conectar ao Visual Studio Code". Isso abrirá uma nova janela do VS Code com uma visualização sobre o que está acontecendo nessa imagem de contêiner. Esta é a visualização do contêiner, não do sistema de arquivos regular do seu computador. Observe a caixa verde na parte inferior mostrando o que a visualização é e que o ícone do Zowe está ausente.



3. ABRA /root/

Os arquivos mais relevantes serão:

anaconda-ks.cfg – este é um arquivo "inicial" feito pelo Anaconda que é usado para descrever o ambiente.

ansible.cfg – um arquivo de configuração que indica como executar o Ansible. Ele também pode ser usado para configurar caminhos de arquivo, intervalos de pesquisa e até mesmo as cores desejadas.

ping.yml – isso é um playbook, assim como submit-job. Eles descrevem as tarefas que devem ser realizadas.

inventory > hosts – uma lista de sistemas conhecidos a qual é possível se conectar.

group_vars > zxp.yml – informações sobre como interagir com sistemas específicos.


```
hosts
! zxp.yml
group_vars > ! zxp.yml
1 ---
2 ansible_port: 22
3 ansible_user: ZXXXXX
4 ansible_password: TOPSECRET
5 # ansible_ssh_pipelining: True
6 # System Related Variables
7 zoau_home: /usr/lpp/IBM/zoautil
8 python_path: /usr/lpp/IBM/cyp/v3r9/pyz
9 ansible_python_interpreter: "{{python_path}}
10 _TAG_REDIR_ERR: txt
11 _TAG_REDIR_IN: txt
```

4. VERIFIQUE O ARQUIVO zxp.yml

Confira `group_vars/zxp.yml` – ele já foi customizado para atender à configuração do sistema. Especificamente, os valores `zoau_home`, `python_path` e `ansible_python_interpreter`.

Agora basta mudar o usuário e a senha para corresponder ao seu login. Faça essas mudanças, salve o arquivo e prepare-se para finalmente emitir um comando.

Observe que esta é apenas a configuração de um sistema específico, portanto, o endereço IP dele não é inserido. Isso acontece no arquivo `inventory > hosts`.

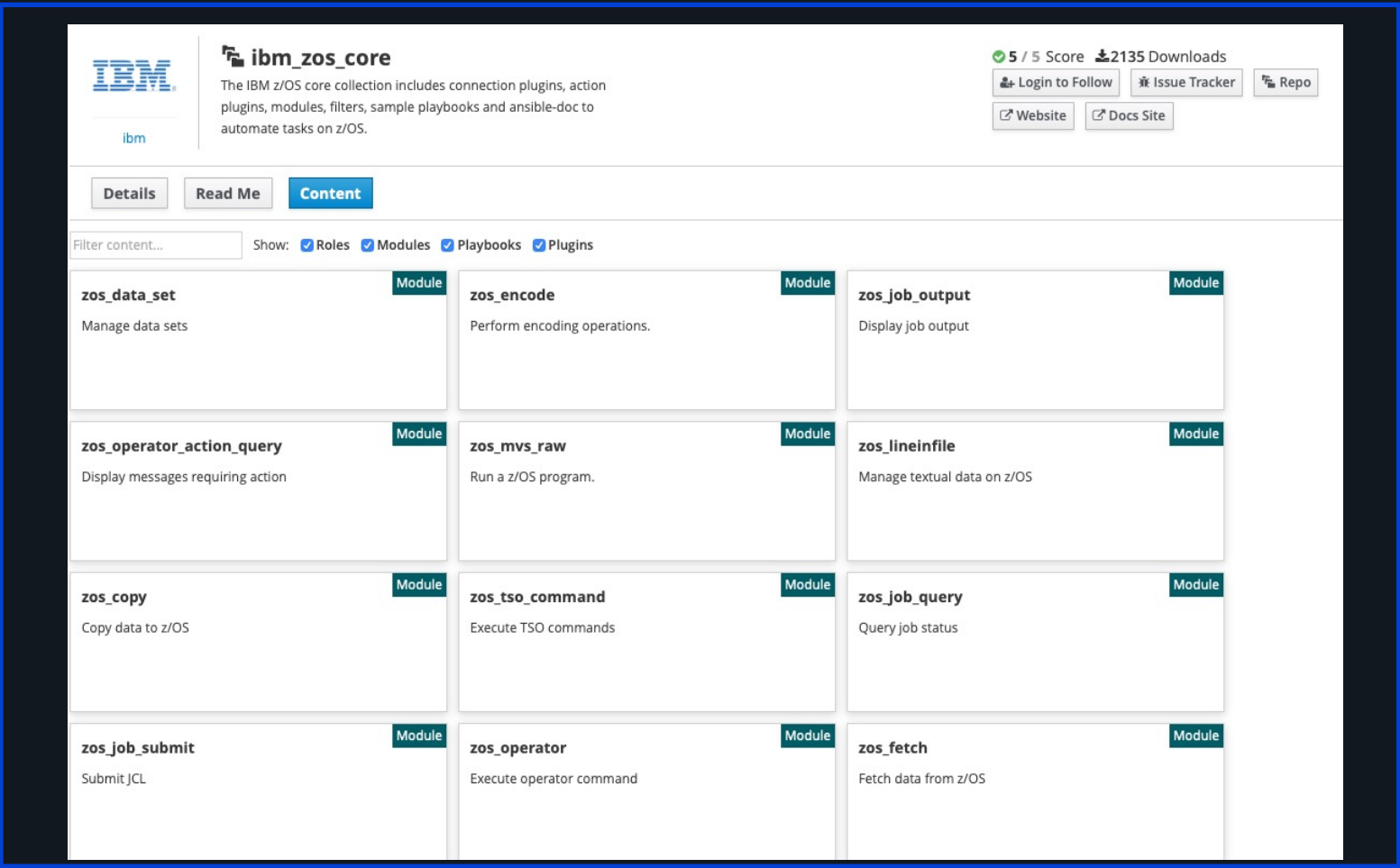
```
hosts x ! zxp.yml ! ping.yml
inventory > hosts
1 [zxp]
2 204.90.115.200
3
PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE PORTS
[root@b789783cd44 ~]# ansible-playbook ping.yml
PLAY [zxp] *****
TASK [Gathering Facts] *****
[WARNING]: Module remote_tmp /z/z99993/.ansible/tmp did not exist and was
created with a mode of 0700, this may cause issues when running as another
user. To avoid this, create the remote_tmp dir with the correct permissions
manually
ok: [204.90.115.200]
TASK [Pinging host - 204.90.115.200] *****
ok: [204.90.115.200]
PLAY RECAP *****
204.90.115.200 : ok=2 changed=0 unreachable=0 failed=0 s
kipped=0 rescued=0 ignored=0
[root@b789783cd44 ~]#
```

5. O PING DA VERDADE

Você concluiu uma longa sequência de etapas de configuração e este é o momento tão esperado.

Abra um terminal de dentro deste ambiente de contêiner e digite o comando: **ansible-playbook ping.yml**

Se tudo correr bem, você receberá uma mensagem de sucesso como a acima. A saída rosa significa apenas que esta é a primeira vez que o Ansible é executado e que ele está criando diretórios temporários. Está tudo bem.



6. MARQUE COMO CONCLUÍDO

Para marcar o desafio como concluído, envie a tarefa **CHKAANS1** em `ZXP.PUBLIC.JCL`

Para obter mais informações, visite a [página do módulo zos_operator](#) na [referência da coleção principal do z/OS](#) a fim de encontrar a sintaxe deste módulo. Consulte o arquivo `submit_job.yml` como referência e sintaxe à vontade para pegar emprestadas as seções "collections" e "environment" para que seu playbook funcione corretamente, pois ele está chamando tarefas específicas do z/OS. Ping é um genérico que funciona sem essas partes.

BOM TRABALHO! VAMOS RECAPITULAR

Estamos felizes demais. Pegamos uma rota especial para chegar aqui e esperamos que você tenha visto o suficiente para se inspirar e continuar. Agora o Ansible está configurado em seu computador (ou onde quer que você o tenha instalado) e nós verificamos que é possível chamar comandos de um playbook.

Você também aprendeu que há uma diferença entre os comandos principais criados no Ansible (como Ping) e os que você recebe ao referenciar aqueles que são específicos do z/OS. Há muito mais para ver, então vamos seguir em frente.

EM SEQUÊNCIA...

Agora que tudo está definido, é hora de usar. Vá para ANSB2 (Playbook) e terminaremos com algo um pouco diferente.

