#!/bin/bash

このスクリプトがBashシェルで実行されることを指定する特殊なコマンド

これによりどのシステムがどのプログラムを使用してスクリプトを実行するか自動的に判断すことができる

yum update -y

インストールされたパッケージを最新の状態に更新するコマンド

-y は全てのプロンプトでyesと自動的に回答することを指定している

amazon-linux-extras install nginx1 -y

systemctl start nginx

systemctl enable nginx

enableは同意という意味になる

INSTANCE\_ID=$(curl -s <http://169.254.169.254/latest/meta-data/instance-id>)

echo "<html><body><h1>Instance ID: $INSTANCE\_ID</h1></body></html>" > /usr/share/nginx/html/index.html

EC2インスタンスのメタデータからインスタンスIDを取得しHTMLファイルを生成してNginxのデフォルトのドキュメントルートに保存するコマンド

インスタンスIDをHTMLに表示するために使用されるもの。

curl URLを使ってデータを転送するためのコマンドラインツール

-s curlコマンドの実行結果を表示させないようにするオプション

この場合、結果を出力に表示せず、指定したURLからデータを取得するために使用される

やったこと　簡易版

１　ユーザーデータ用のEC2を作成

２ テラタームを起動し、コマンドテストをする

３　一旦EC2を停止させ、ユーザーデータを設定する

４　設定後反映されているか、確認する

注意点、ユーザーデータ使用時と通常時で権限が違うのでコマンドに気を付ける

sudo suで権限切り替えなど。。

参考サイト

<https://docs.aws.amazon.com/ja_jp/AWSEC2/latest/UserGuide/user-data.html#user-data-shell-scripts>

<https://docs.aws.amazon.com/ja_jp/AWSEC2/latest/WindowsGuide/ec2-windows-user-data.html>

<https://tech-lab.sios.jp/archives/29999>

自分用

他の方法も調べて、後程試しにやってみる（途中までしかできていない

nginx設定ファイルを開く

sudo nano /etc/nginx/nginx.conf

ファイルの末尾にこの設定を追加する

server {

listen 80;

location /instance-id {

default\_type text/plain;

return 200 "$host\n$instance\_id\n";

}

}

ファイル構文が正しいか確認する

sudo nginx -t

nignxを再起動して反映させる

sudo service nginx restart

[http://EC2インスタンスのパブリックDNS/instance-idにアクセスすることでインスタンスIDを取得することができる](about:blank)

nanoファイル編集コマンド

Ctrl O →エンター　ファイルを上書き

Ctrl X nanoを終了

TAB二回押す事でファイルの場所を確認できる

ls 詳細表示

cat ファイルの中身を表示する

コメントアウトする癖を付ける

# 入れたい内容

#!/bin/bash

# clean up log file

mkdir -p /usr/test/

touch /usr/test/test.log

#install nginx

echo”start nginx install” >> /usr/test/test.log

yum update -y

amazon-linux-extras install nginx1 -y

systemctl start nginx

systemctl enable nginx

# update index.html by instance-id

INSTANCE\_ID=$(curl -s http://169.254.169.254/latest/meta-data/instance-id)

echo "<html><body><h1>Instance ID: $INSTANCE\_ID</h1></body></html>" > /usr/share/nginx/html/index.html

cat /usr/test/test.log

ユーザーデータ実行後これでログを表示させる事ができる

外部的にログを保管できるサービス　クラウドウォッチログ

クラウドユニットのログファイル、自分で作ったログファイル

nginxのログファイル、アクセス、エラーログ、nginxの起動ログ

S3からやる、ユーザーデータの実行が終わったタイミングでアップロードする

S3バケットは公開で作らないようにする。

パブリックアクセスをブロックのチェックを外さないようにする

クラウドウォッチlogsまたはS3バケットにアップロードする。

インスタンスプロファイルというものを使う、権限情報となる、それおをもとにＳ３にアップロードする

誤操作を防ぐために必要なものは見れるように設計していくのがきれい

中に入るのは緊急時のみ、めったにない

インスタンスにログを保存居ていても、インスタンスを終了した時に、ログが消えてしまうので、別のサービスを使ったりする。