The AWS SDK for JavaScript version 3 (v3) is a rewrite of v2 with some great new features, including modular architecture. For more information, see the AWS SDK for JavaScript v3 Developer Guide (https://docs.aws.amazon.com/sdk-for-javascript/v3/developer-guide/welcome.html) .

Tutorial: Setting Up Node.js on an Amazon EC2 Instance

PDF (/pdfs/sdk-for-javascript/v2/developer-guide/js-sdk-dv-v2.pdf#setting-up-node-on-ec2-instance) | **RSS (amazon-sdk-javascript-guide-doc-history.rss)**

A common scenario for using Node.js with the SDK for JavaScript is to set up and run a Node.js web application on an Amazon Elastic Compute Cloud (Amazon EC2) instance. In this tutorial, you will create a Linux instance, connect to it using SSH, and then install Node.js to run on that instance.

Prerequisites

This tutorial assumes that you have already launched a Linux instance with a public DNS name that is reachable from the Internet and to which you are able to connect using SSH. For more information, see Step 1: Launch an Instance (https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/EC2_GetStarted.html#ec2-launch-instance_linux) in the *Amazon EC2 User Guide for Linux Instances*.

You must also have configured your security group to allow `SSH` (port 22), `HTTP` (port 80), and `HTTPS` (port 443) connections. For more information about these prerequisites, see Setting Up with Amazon Amazon EC2 (https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/get-set-up-for-amazon-ec2.html) in the *Amazon EC2 User Guide for Linux Instances*.

Procedure

The following procedure helps you install Node.js on an Amazon Linux instance. You can use this server to host a Node.js web application.

To set up Node.js on your Linux instance

1. Connect to your Linux instance as `ec2-user` using SSH.
2. Install node version manager (nvm) by typing the following at the command line.

   > **Warning**
   >
   > AWS does not control the following code. Before you run it, be sure to verify its authenticity and integrity. More information about this code can be found in the nvm (https://github.com/nvm-sh/nvm/blob/master/README.md) GitHub repository.

   ```
   curl -o- https://raw.githubusercontent.com/nvm-
   sh/nvm/v0.39.3/install.sh | bash
   ```

   We will use nvm to install Node.js because nvm can install multiple versions of Node.js and allow you to switch between them.
3. Activate nvm by typing the following at the command line.

   ```
   . ~/.nvm/nvm.sh
   ```

4. Use nvm to install the latest version of Node.js by typing the following at the command line.

   ```
   nvm install --lts
   ```

   > **Note**
   >
   > This installs the latest LTS release of Node.js.

   > **Warning**
   >
   > Amazon Linux 2 does not currently support the current LTS release (version 18.x) of Node.js. Use Node.js version 16.x with the following command instead.
   >
   > ```
   > nvm install 16
   > ```

Installing Node.js also installs the Node Package Manager (npm) so you can install additional modules as needed.

5. Test that Node.js is installed and running correctly by typing the following at the command line.

   ```
   node -e "console.log('Running Node.js ' + process.version)"
   ```

   This displays the following message that shows the version of Node.js that is running.

   `Running Node.js` *`VERSION`*

> **Note**
>
> The node installation only applies to the current Amazon EC2 session. If you restart your CLI session you need to use nvm to enable the installed node version. If the instance is terminated, you need to install node again. The alternative is to make an Amazon Machine Image (AMI) of the Amazon EC2 instance once you have the configuration that you want to keep, as described in the following topic.

Creating an Amazon Machine Image

After you install Node.js on an Amazon EC2 instance, you can create an Amazon Machine Image (AMI) from that instance. Creating an AMI makes it easy to provision multiple Amazon EC2 instances with the same Node.js installation. For more information about creating an AMI from an existing instance, see Creating an Amazon EBS-Backed Linux AMI (https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/creating-an-ami-ebs.html) in the *Amazon EC2 User Guide for Linux Instances*.

Related Resources

For more information about the commands and software used in this topic, see the following web pages:

- node version manager (nvm): see nvm repo on GitHub (https://github.com/creationix/nvm) .
- node package manager (npm): see npm website (https://www.npmjs.com) .