Propel

Reach your servers

**Configuration Management**

Contents

[Development Phases 4](#_Toc97972047)

[First time deployment 5](#_Toc97972048)

[Node.js 5](#_Toc97972049)

[MongoDB 5](#_Toc97972050)

[Additional tools and steps 9](#_Toc97972051)

[ActiveDirectory Module 9](#_Toc97972052)

[AWS Tools for Windows Powershell 9](#_Toc97972053)

[Deploying Propel 11](#_Toc97972054)

Figures

[Figure 1 Keep this unchecked: "Tools for Native Modules" 5](file:///C:\DATA\Dev\Propel\propel\doc\Propel%20-%20Configuration%20Management.docx#_Toc97973413)

[Figure 2 - Checking Node.js versions 5](file:///C:\DATA\Dev\Propel\propel\doc\Propel%20-%20Configuration%20Management.docx#_Toc97973414)

[Figure 3 - Installing Mongo DB as a service 6](file:///C:\DATA\Dev\Propel\propel\doc\Propel%20-%20Configuration%20Management.docx#_Toc97973415)

[Figure 4 - Choose complete setup type 6](file:///C:\DATA\Dev\Propel\propel\doc\Propel%20-%20Configuration%20Management.docx#_Toc97973416)

[Figure 5 - Mongo DB site 6](file:///C:\DATA\Dev\Propel\propel\doc\Propel%20-%20Configuration%20Management.docx#_Toc97973417)

[Figure 6 - Modified Mongod.cfg file with alternative folders. 7](file:///C:\DATA\Dev\Propel\propel\doc\Propel%20-%20Configuration%20Management.docx#_Toc97973418)

[Figure 7 - Granting full access to the service account. 7](file:///C:\DATA\Dev\Propel\propel\doc\Propel%20-%20Configuration%20Management.docx#_Toc97973419)

[Figure 8 Script to add the Mongo DB path to the PATH system environment variable 7](file:///C:\DATA\Dev\Propel\propel\doc\Propel%20-%20Configuration%20Management.docx#_Toc97973420)

[Figure 9 - Getting the Mongo engine version 8](file:///C:\DATA\Dev\Propel\propel\doc\Propel%20-%20Configuration%20Management.docx#_Toc97973421)

[Figure 10 - Enabling auth for all MongoDB databases. 8](file:///C:\DATA\Dev\Propel\propel\doc\Propel%20-%20Configuration%20Management.docx#_Toc97973422)

[Figure 11 - Creating the new DBA user 8](file:///C:\DATA\Dev\Propel\propel\doc\Propel%20-%20Configuration%20Management.docx#_Toc97973423)

[Figure 12 - Installing the PowerShell ActiveDirectory module. 9](file:///C:\DATA\Dev\Propel\propel\doc\Propel%20-%20Configuration%20Management.docx#_Toc97973424)

[Figure 13 - Installing AWS Tools for PowerShell 10](file:///C:\DATA\Dev\Propel\propel\doc\Propel%20-%20Configuration%20Management.docx#_Toc97973425)

[Figure 14 -Dist folder 11](file:///C:\DATA\Dev\Propel\propel\doc\Propel%20-%20Configuration%20Management.docx#_Toc97973426)

[Figure 15 - Installer file 11](file:///C:\DATA\Dev\Propel\propel\doc\Propel%20-%20Configuration%20Management.docx#_Toc97973427)

# Development Phases

Two phases planned so far:

**Phase I**: The tool will be accessible only from a specific server via Remote Desktop technology, reducing any security risk to the minimum.

**Phase II**: The tool will be accessible from the internet adding security to both APP and API.

**This document will focus on the configuration management aspect of Phase I only**.

# First time deployment

This topic is going to cover everything required to deploy the solution for the first time. We will take care of all the project dependencies that we need to install and configure.

## Node.js

Navigate to Node.js site and download the current LTS, (Long Term Support), version. Then install using all by defect options, with the exception of the “Tools for native modules check that need to be checked manually as you can see in Figure 2.

After Node is installed, you can check it’s version and also the one for Npm, (Node Package Manager), by running the command you can see in Figure in a PowerShell console.

## MongoDB

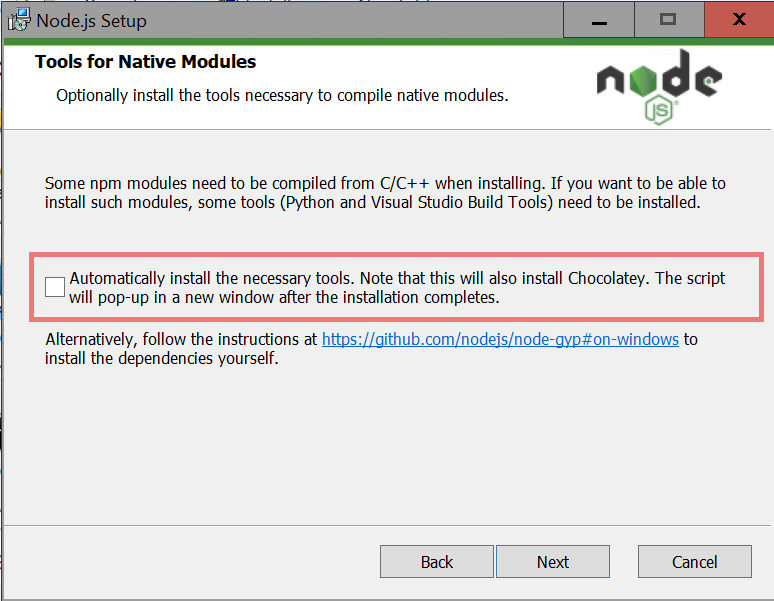


Figure 1 Keep this unchecked: "Tools for Native Modules"

Download the Mongo DB community server edition and install it with the default values. Choose the version based on the support of the OS. From time-to-time Mongo is dropping the support for different Server versions.

Choose the most up to date version that support your target OS. When the installation prompt for a setup type be sure to select the “Complete” option as you can see in Figure 4.

Next step is to configure MongoDB to run as a service.

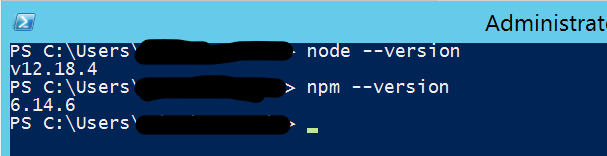


Figure 2 - Checking Node.js versions

Last step in the installer is prompting about to install Mongo DB compass, uncheck the option. You can install later if needed. Also, optionally you can try [Robo 3T](https://robomongo.org/), (a.k.a. RoboMongo), is a light Mongo DB GUI.

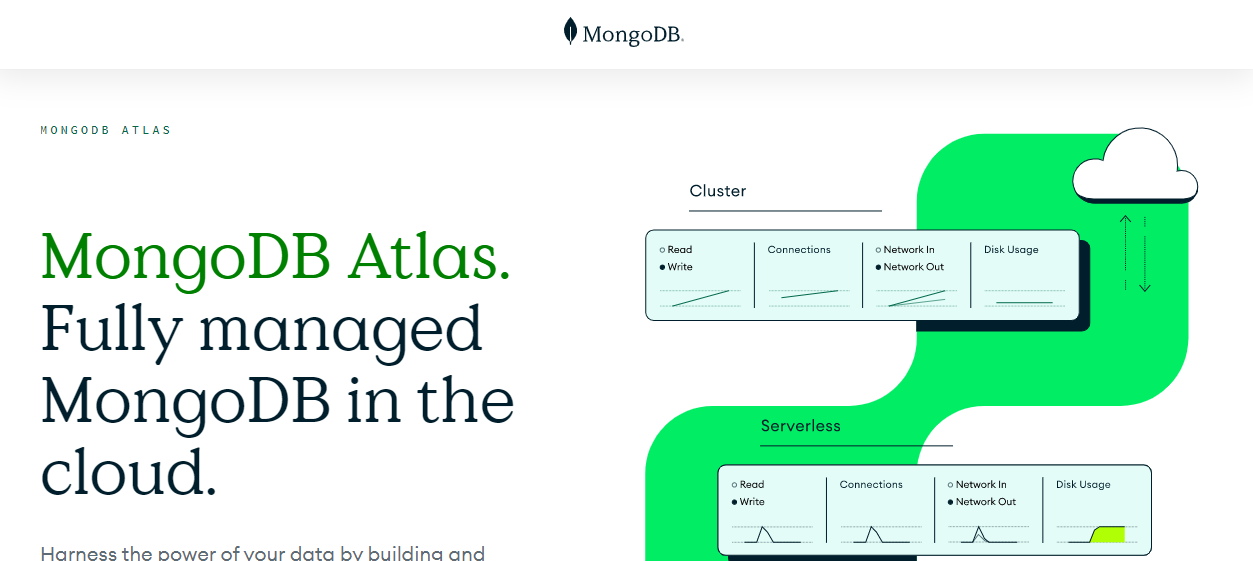


Figure - Mongo DB site

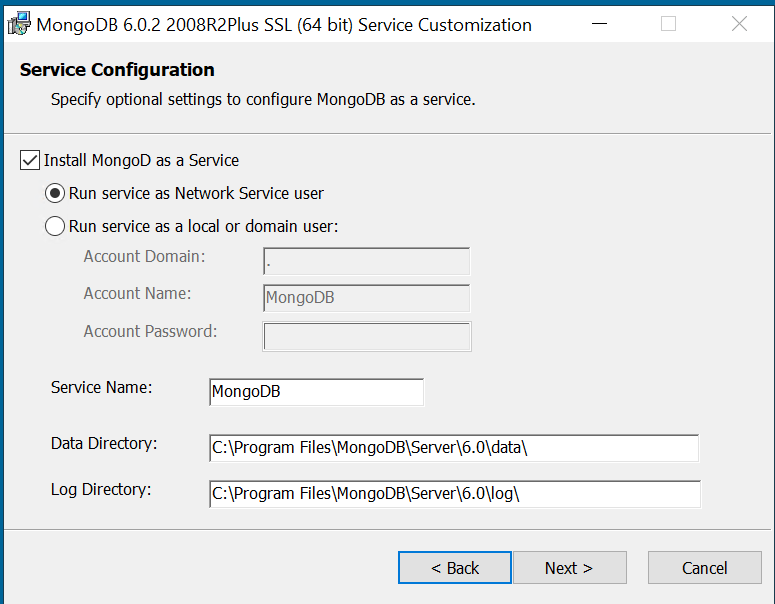


Figure - Installing Mongo DB as a service

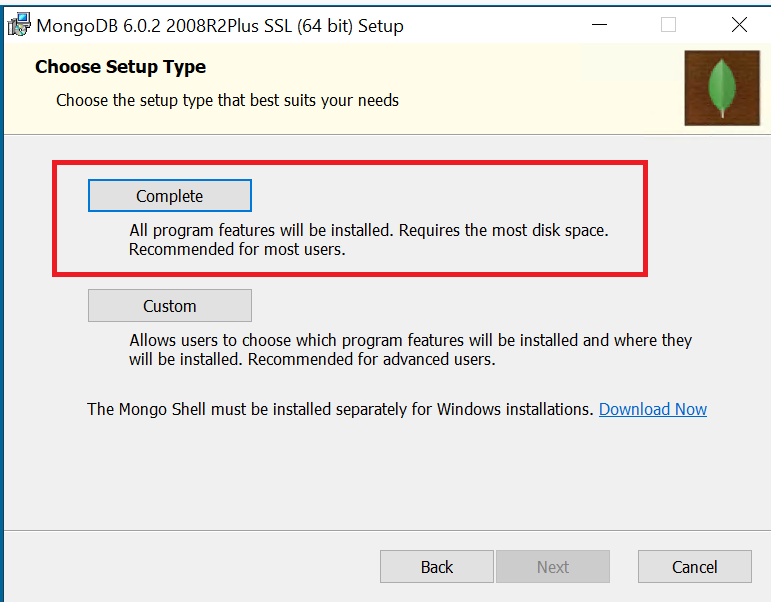


Figure - Choose complete setup type

If after the installation you want to change the folders the engine is going to use for data and logging, you must proceed in this way:

**1st -** Stop the “MongoDB” service.

**2nd** - Open the folder “*C:\Program Files\MongoDB\Server\4.0\bin*” with a file explorer instance.

**3rd** - Make a backup of current “mongod.cfg” file.

**4th** - Update the mongod.cfg file by modifying the “storage.dbPath” and “systemLog.path” values as detailed in Figure 6.

**5th** - Grant full access to the local built in account “NETWORK SERVICE” over those folders,(Figure 7).

**6th** - And as last step, start the service.

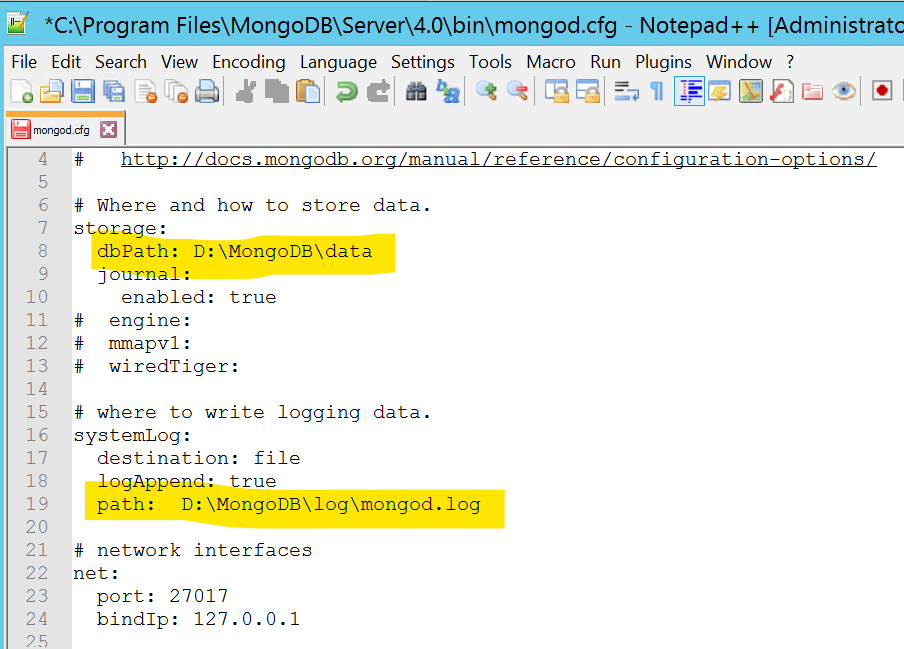


Figure - Modified Mongod.cfg file with alternative folders.

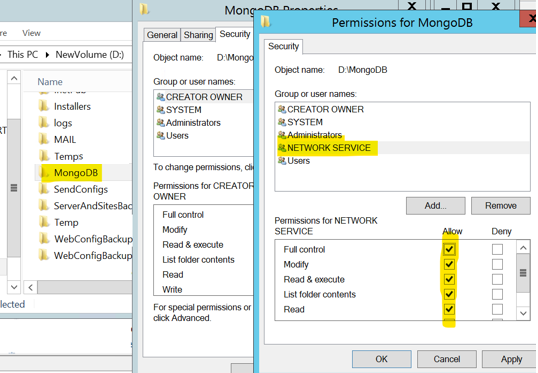


Figure - Granting full access to the service account.

Also, you maybe would like to install the [Mongo DB tools](https://www.mongodb.com/try/download/database-tools). These are helper commands to for example make backups, extract or import data into the db, etc.

To avoid later that the Propel installer failed, there is one more single step required. This is to add Mongo DB runtime folder to the global path. You can do it by running the following script.

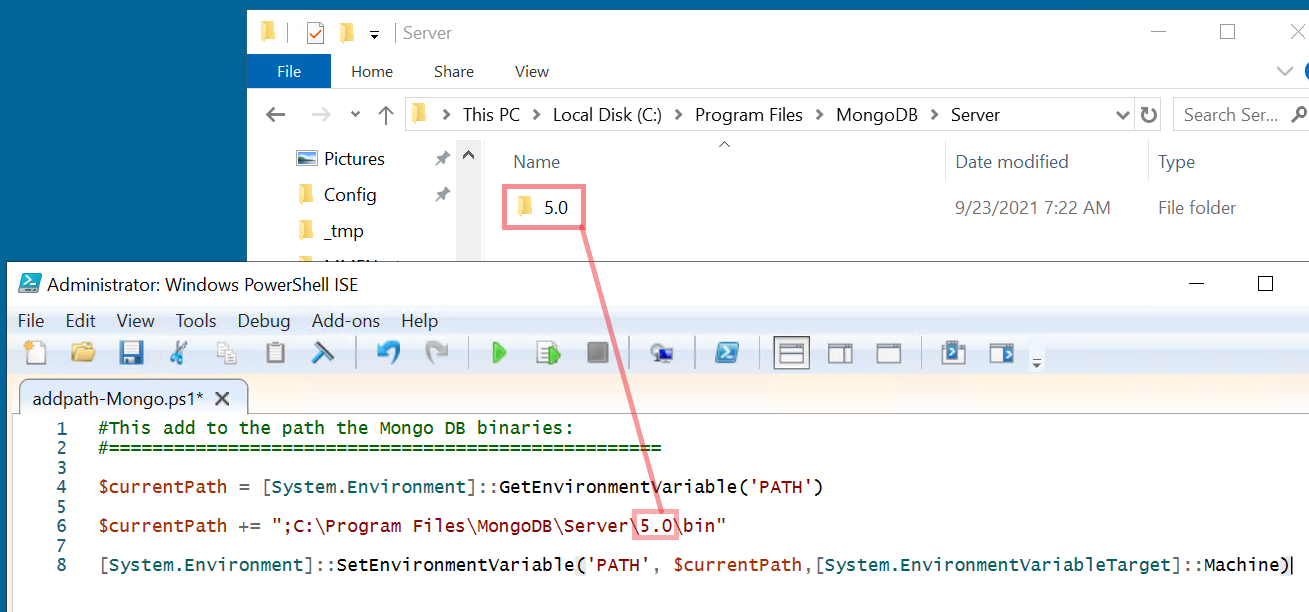


Figure Script to add the Mongo DB path to the PATH system environment variable

Now you can run the command you can see in Figure 9 from any folder to verify the path was set correctly. Be aware **this require to restart the computer to apply the changes**.

Next step is to secure the database. To do that, you can use the following script from Propel v2.0 repository:

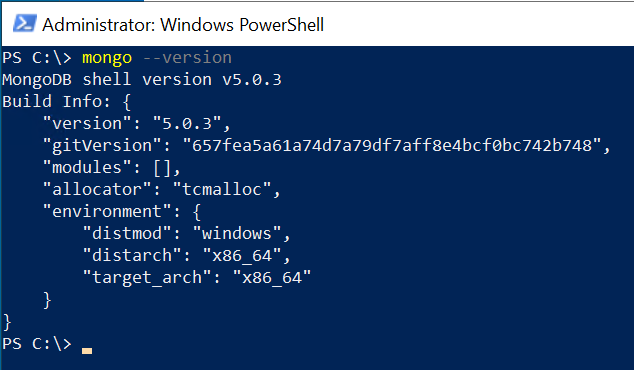


Figure - Getting the Mongo engine version

"*.\distrib\cutover\create-admin-user.js*"

Yu can copy in the target machine and run it as you can see in Figure 10. This script is going to create a user named “*DBA”* in the Admin database that will have administrator privileges in any database.

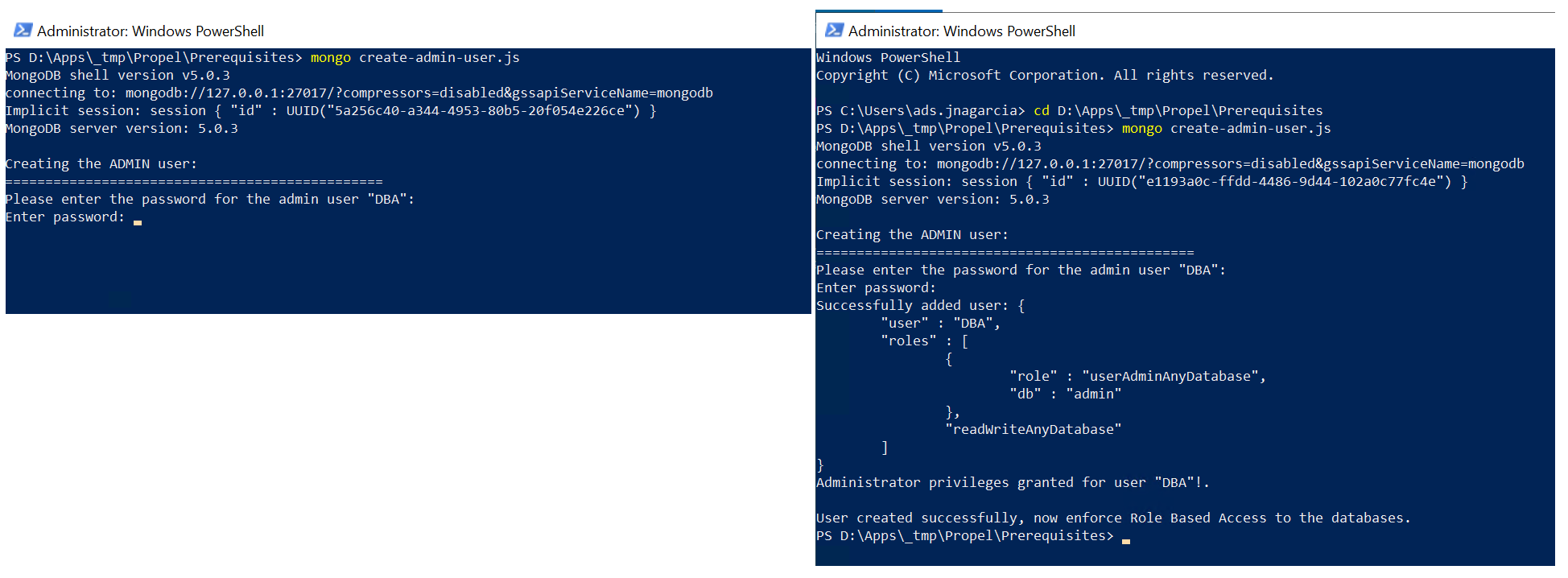


Figure - Creating the new DBA user

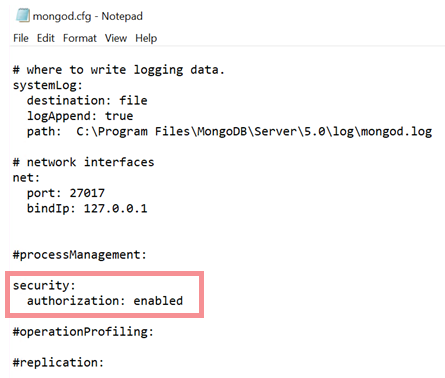


Figure - Adding the Authorization security feature.

After doing that our next steps is enforce user authentication in the MongoDB engine.

So, now we need to stop the MongoDB service, locate the mongod config file located at “*C:\Program Files\MongoDB\Server\{Here the server version folder}\bin\mongod.cfg*” and edit it to enable the authorization security feature as depicted in Figure 11.

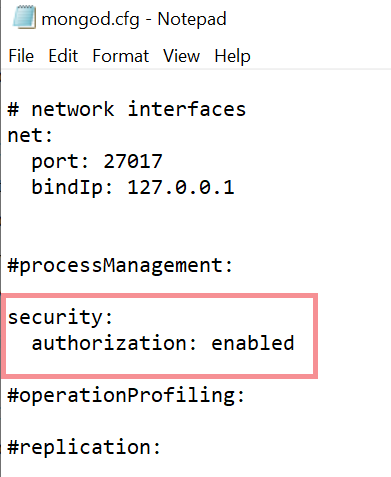


Figure - Enabling auth for all MongoDB databases.

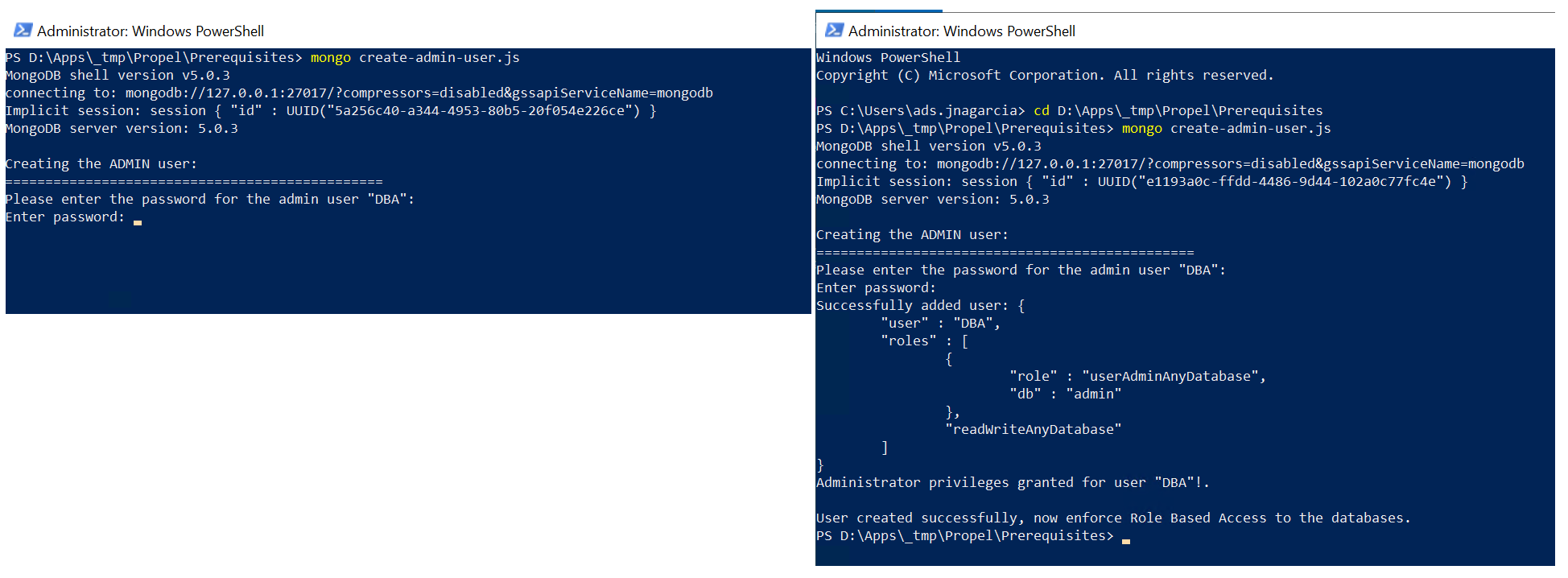


Figure - Creating the new DBA user

After that we can restart the Mongo DB Service.

# Additional tools and steps

Depending on the kind of scripts you are going to execute, maybe you will need to take care of installing and configuring come of the following PowerShell tools and modules.

## ActiveDirectory Module

In order to be able to run LDAP Queries to Microsoft Active Directory, you will need installed the “*ActiveDirectory*” PowerShell module. To do this, you will need to open the Server Manager and add the feature as depicted in Figure 12

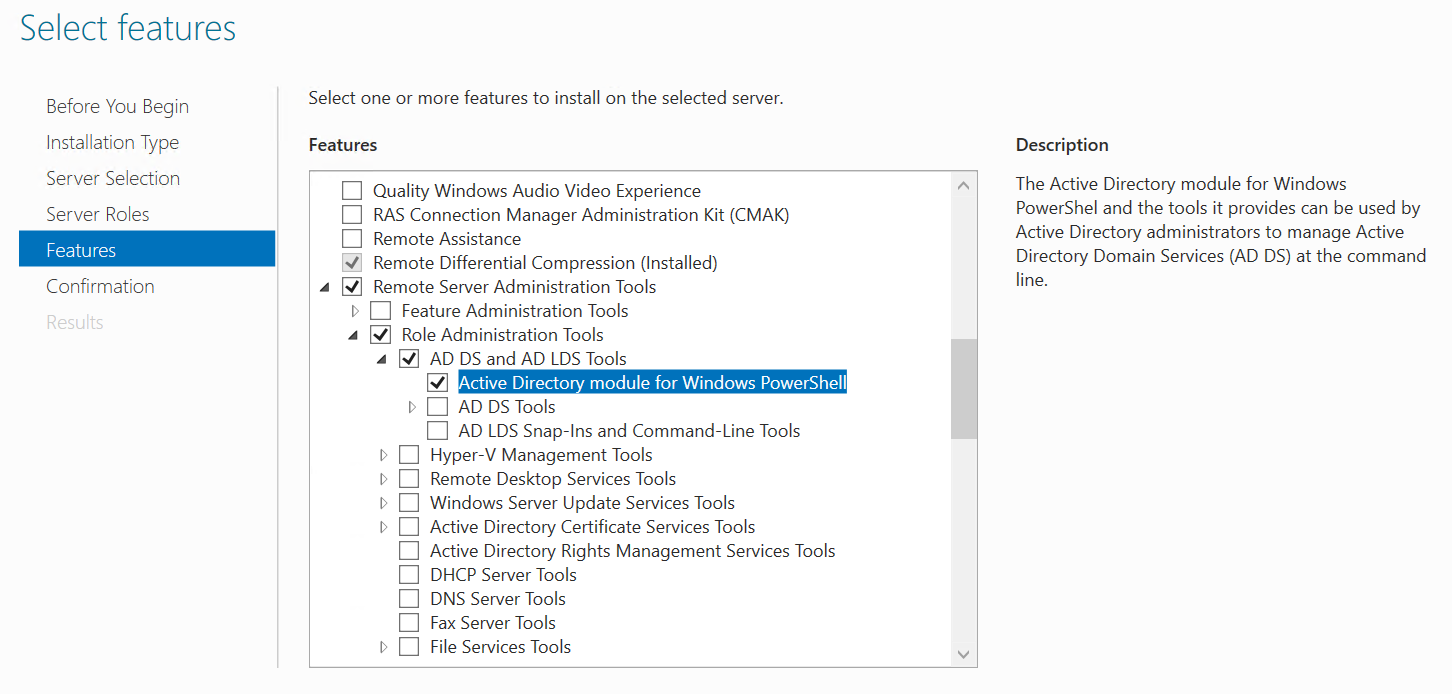


Figure 12 - Installing the PowerShell ActiveDirectory module.

## AWS Tools for Windows Powershell

In the case you have this dependency for your scripts the way to install it is by opening an elevated PowerShell console session and execute this:

Install-Module -Name AWSPowerShell -Scope AllUsers

You can see more details in Figure 13.

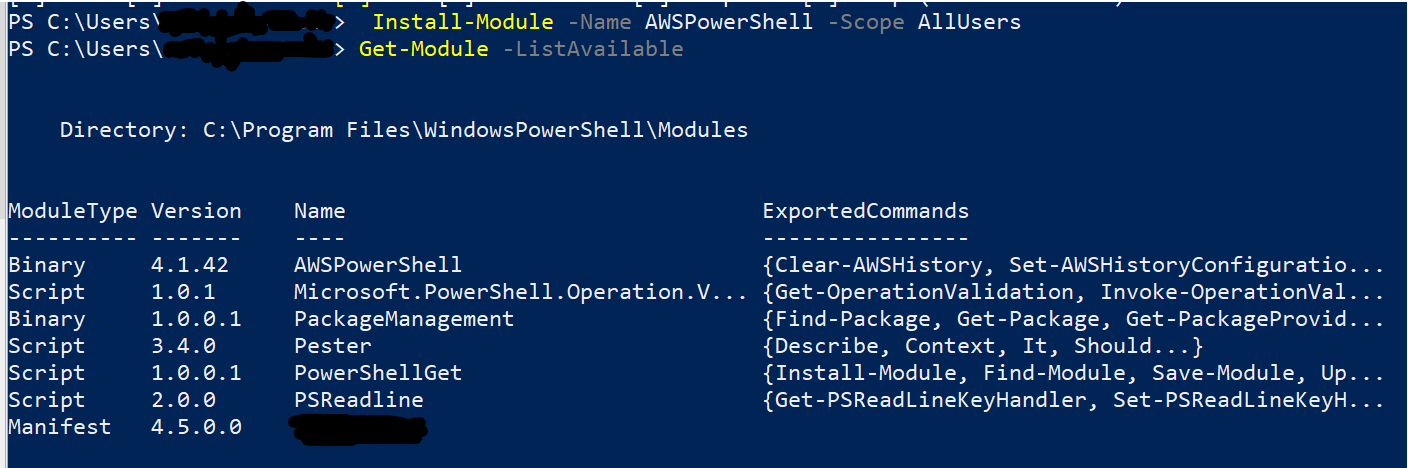


Figure 13 - Installing AWS Tools for PowerShell

For more details about this visit the [AWS Tools for Powershell](https://docs.aws.amazon.com/powershell/latest/userguide/pstools-getting-started.html) site.

# Preparing development environment

This section is intended to guide you in the requirements to setup a development environment to be able to clone/download Propel code from our GIT repository and also be able to build, run test, etc.

## Cloning the repo in your local

You will need some sort of IDE to follow the steps in this section. In this document [Visual Studio Code](https://code.visualstudio.com/) is the selected IDE, but you can use whatever you want.

First navigate to our [GIT repository](https://github.com/Elmosoftware/propel-api) and get the clone URL

Graphical user interface, text, application, email

Description automatically generated

Figure - Getting the clone URL from Git repo

Let’s open now the parent folder we want to use for our repository in a powershell console and run the following command:

**git clone {Here the url} {Here the folder name to create}**

You can see the run of this command in Figure 17.

Also if is a new Git installation, recall you will need to [setup your user name and email](https://git-scm.com/book/en/v2/Getting-Started-First-Time-Git-Setup) before to be able to commit.

## Deploying Propel

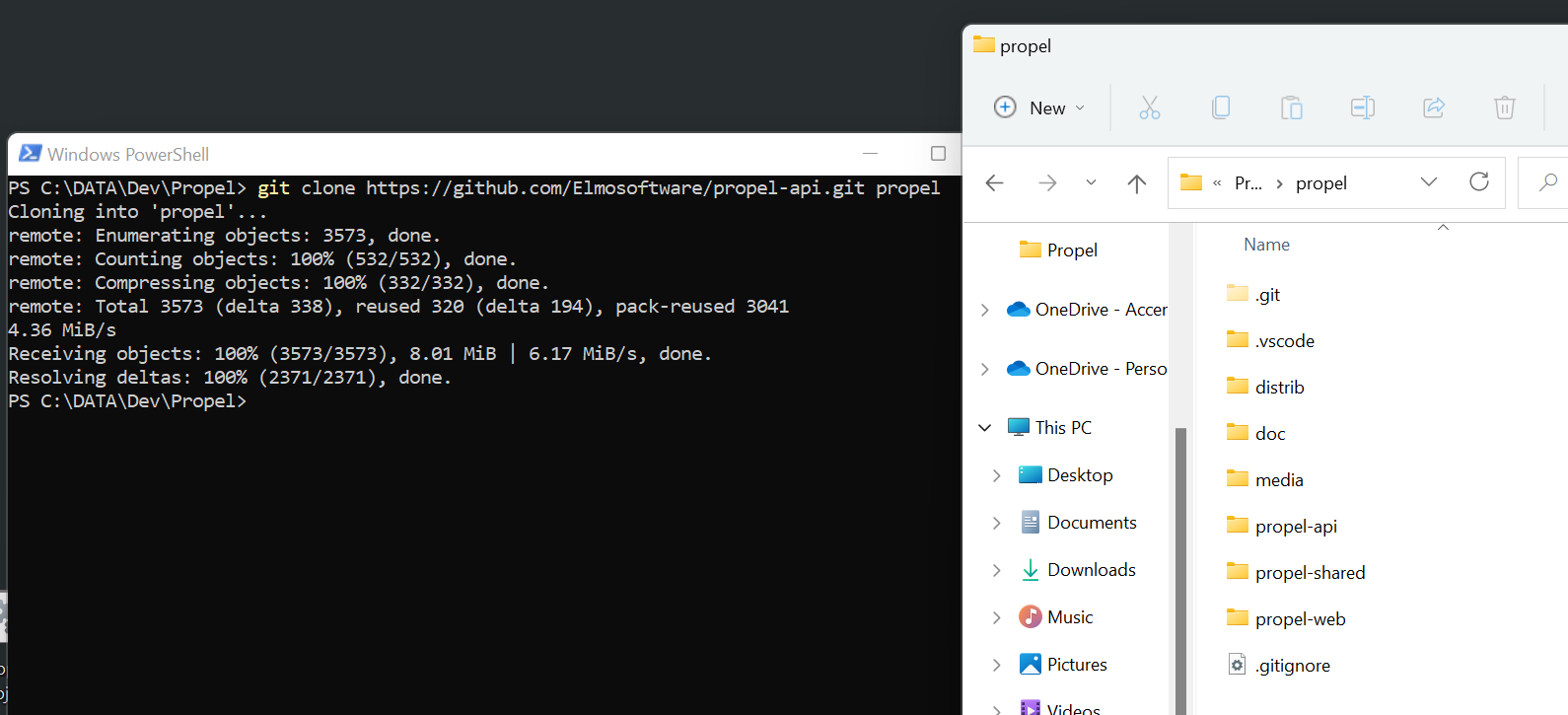


Figure - Cloning the Repository in VS Code

In VS Code click on the “Run Task” menu option of the “Terminal” menu and select the “Build Production” task. After it runs you will have in “.\Distrib\dist” folder the new build. Next step is to compress the folder content and copy them to the production server in a temporal location. **If it’s a first-time deployment, ensure the destination folder “C:\Propel” exists**, otherwise please create it. That’s the folder where Propel is being installed.

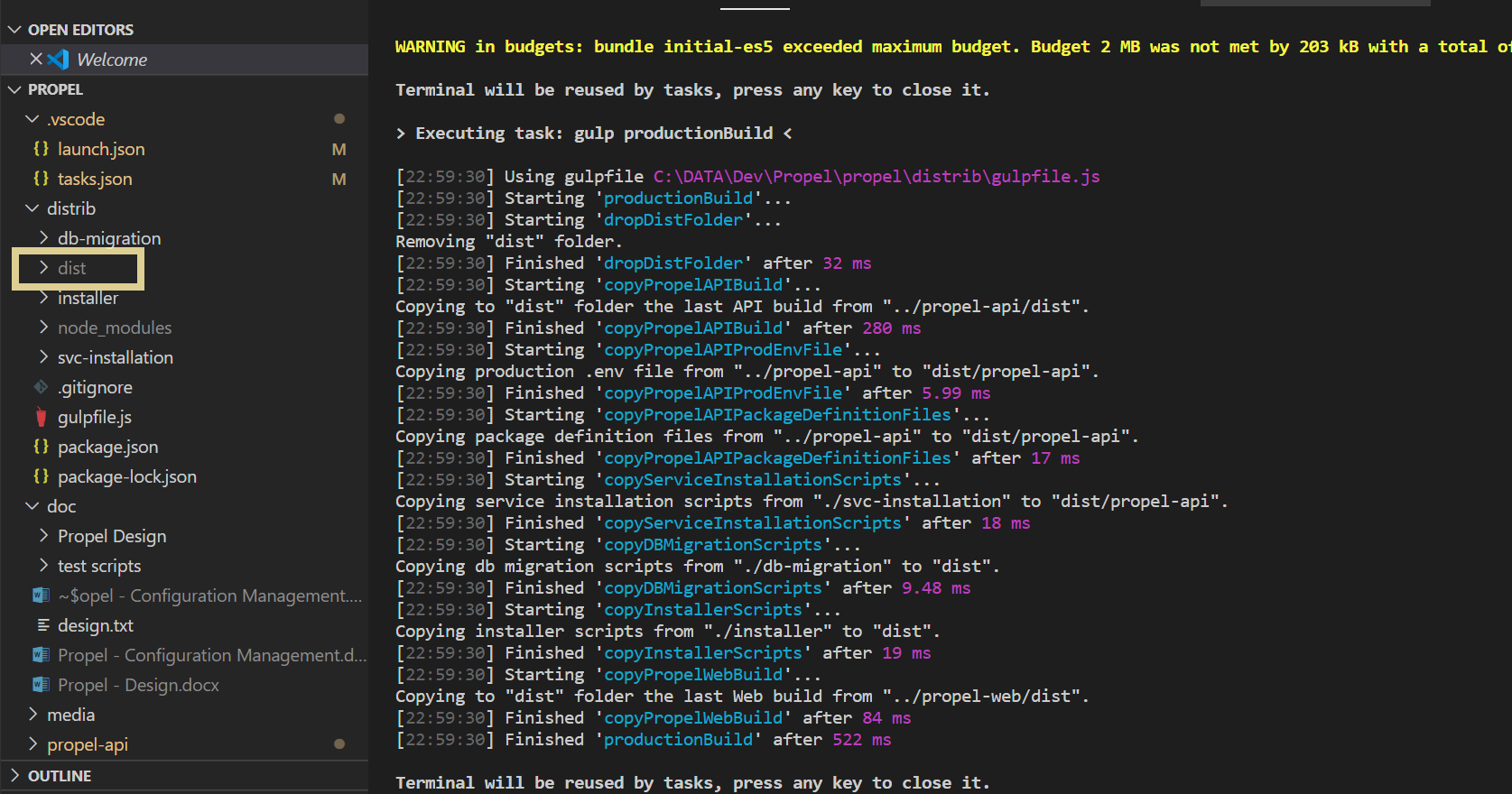


Figure -Dist folder

In the temporal installer path, please locate the file “install.ps1” and run it with PowerShell.

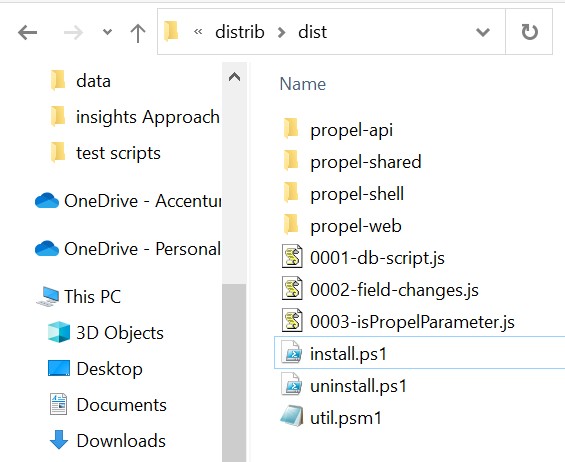


Figure - Installer file

The installer will take care of:

Stop the Propel service if it is running, install the new version and restart the service.

It will also apply any required database migration script. And will allows you to configure a specific account to use for impersonation during remote script execution, (otherwise remote scripts will run with the credentials set to the service).