Migrate web APP to Microsoft Azure using 1-Click LAMP Migration Tool

Contents

[**Introductions** 1](#_Toc436723612)

[**Prerequisites** 2](#_Toc436723613)

[**Steps** 4](#_Toc436723614)

[**1.** **Start migration** 4](#_Toc436723615)

[**2.** **Connect to your Linux Server** 5](#_Toc436723616)

[**3.** **Check the connection to your Linux server** 6](#_Toc436723617)

[**4.** **Choose the candidate website** 7](#_Toc436723618)

[**5.** **Create new Web APP with MySQL** 7](#_Toc436723619)

[ **For people using global Microsoft Azure** 7](#_Toc436723620)

[ **For Chinese using Mooncake Azure** 10](#_Toc436723621)

[**6.** **Download the publish profile** 12](#_Toc436723622)

[ **For people using global Microsoft Azure** 12](#_Toc436723623)

[ **For Chinese using Mooncake Azure** 13](#_Toc436723624)

[**7.** **Upload the publish profile** 14](#_Toc436723625)

[**8.** **Start migration** 15](#_Toc436723626)

[**Appendix** 15](#_Toc436723627)

[**1.** **Enable PasswordAuthentication** 15](#_Toc436723628)

[**2.** **Disable Requiretty** 16](#_Toc436723629)

# **Introductions**

The 1-Click LAMP Migration Tool is a web application migration assistant which allows you to migrate your existing web applications hosted on LAMP (Linux + Apache + MySQL + PHP) environment to Microsoft Azure Web Apps Service easily.

1-Click LAMP Migration Tool support to move popular LAMP scenario into Azure Web Apps.

* Supported Web Servers: Apache running on Linux (Ubuntu, CentOS 7, OpenSUSE etc.).
* Supported PHP Frameworks: WordPress, Drupal, Joomla (We plan to support more frameworks in future based on your feedback).
* Supported Database: MySQL (or MariaDB). DB can be local or remote machine.

Using this tool, we will connect to your Linux server via SSH, and check your Apache and MySQL configuration and analyze your PHP site. Then we allow you to use existing Publish Settings profile or create a new Azure Web Apps + MySQL service on Azure with new Publish Settings profile. Finally, we will use the Publish Settings profile to migrate your website from LAMP to Microsoft Azure Web Apps automatically.

# **Prerequisites**

* SSH service is required to be enabled in the Linux side.
* A Superuser is required to run the migration script.
* Your Linux server can access to public Azure Web Apps.
* *PasswordAuthentication*  must be enabled for the user to log in via SSH using username and password if it is disabled. --- For details, see Appendix 1.
* *Requiretty*  must be disabled for the user to run command via SSH with sudo permission if it is enabled. --- For details, see Appendix 2.
* The implementation is based on a modified preview perl script provided by production team. The script can be downloaded from <http://www.movemetothecloud.net/Azure-Apache-Migration-Tool.tar.gz> .
* Some Perl libraries, such as perl LWP HTTPS, are required in Linux side to run the script. For different distros, you can follow the following command to install Perl LWP HTTPS.

#Ubuntu

sudo apt-get install liblwp-protocol-https-perl

#CentOS 7

sudo yum install perl-LWP-Protocol-https

you may also need to install some additional libraries:

sudo yum install perl-Digest-MD5

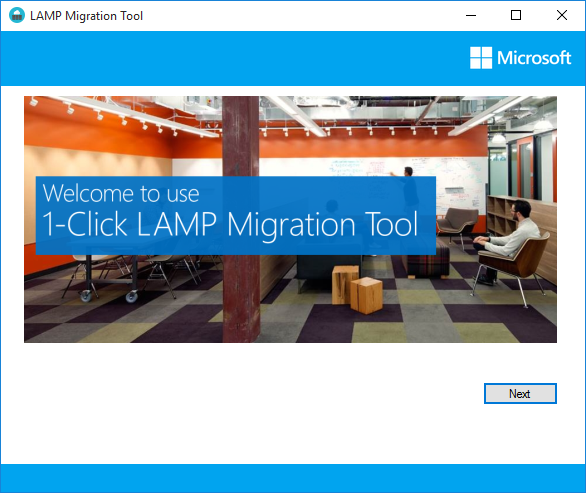
sudo yum install perl-Compress-Raw-Zlib

#OpenSUSE

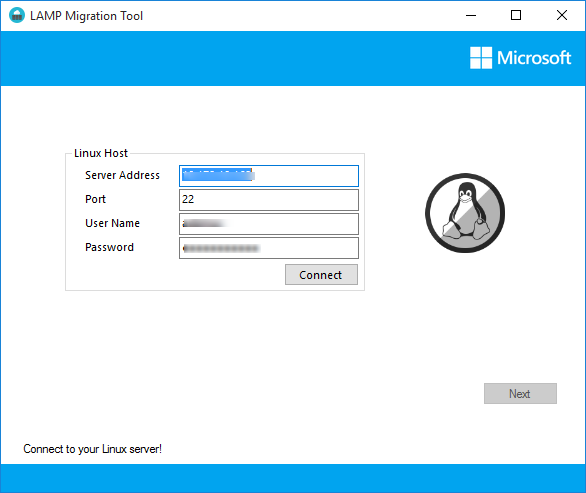
sudo zypper install perl-LWP-Protocol-https

# **Steps**

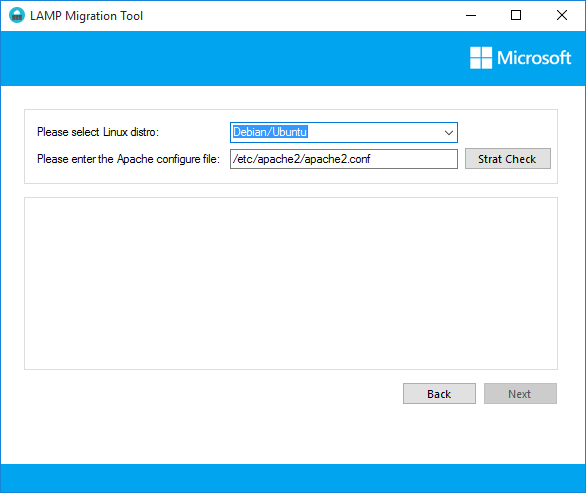
## **Start migration**



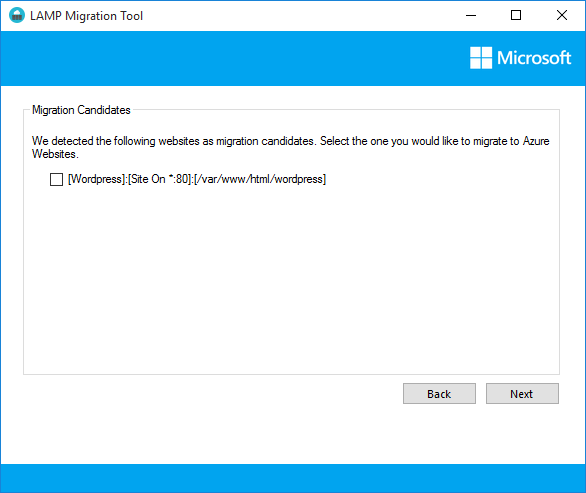
## **Connect to your Linux Server**



## **Check the connection to your Linux server**



## **Choose the candidate website**

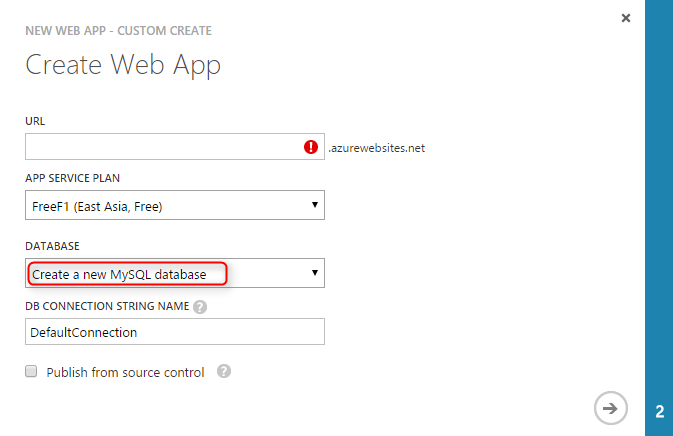


## **Create new Web APP with MySQL**

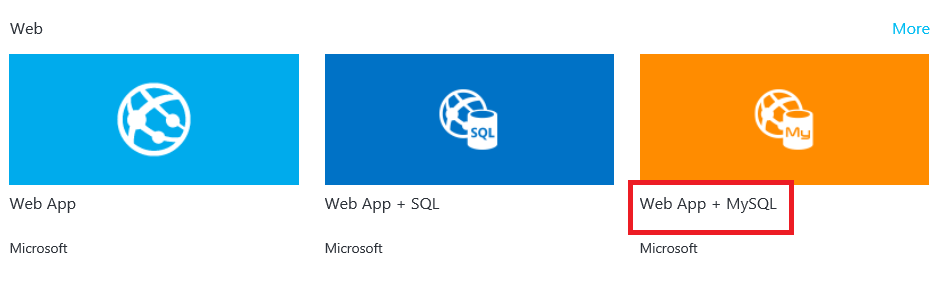
|  |
| --- |
| **NOTE:**  If you have already created a Web App with MySQL database, please jump to step 6. |

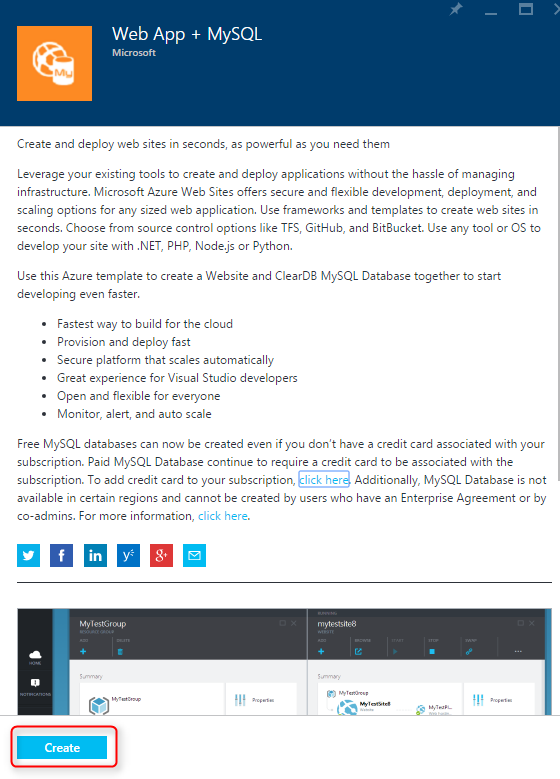
### **For people using global Microsoft Azure**

1. If you are using classic Azure portal, please go to <http://manage.windowsazure.com> and create a new Web APP with MySQL database.



1. If you are using new Azure portal, please go to <http://portal.azure.com> to create a new Web APP with MySQL database in Market place.

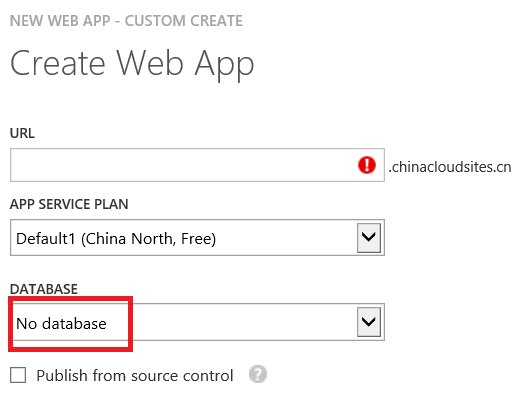




### **For Chinese using Mooncake Azure**

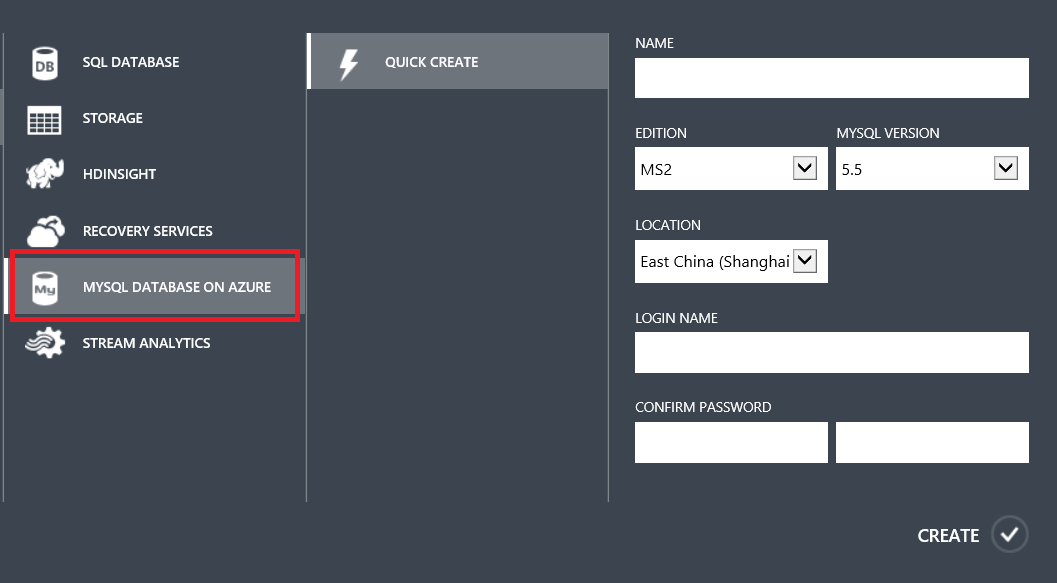
1. Please go to <https://manage.windowsazure.cn> to create a new Web APP **without database**.

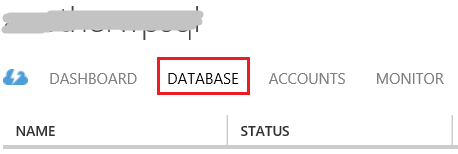
|  |
| --- |
| **NOTE:**  Since we cannot create a Web APP with MySQL directly in Mooncake, you need to add database connection string manually. |



1. Create a new MySQL database server and add a new database.

|  |
| --- |
| **NOTE:**  If you have already have a MySQL server and a database, you can skip this step. |

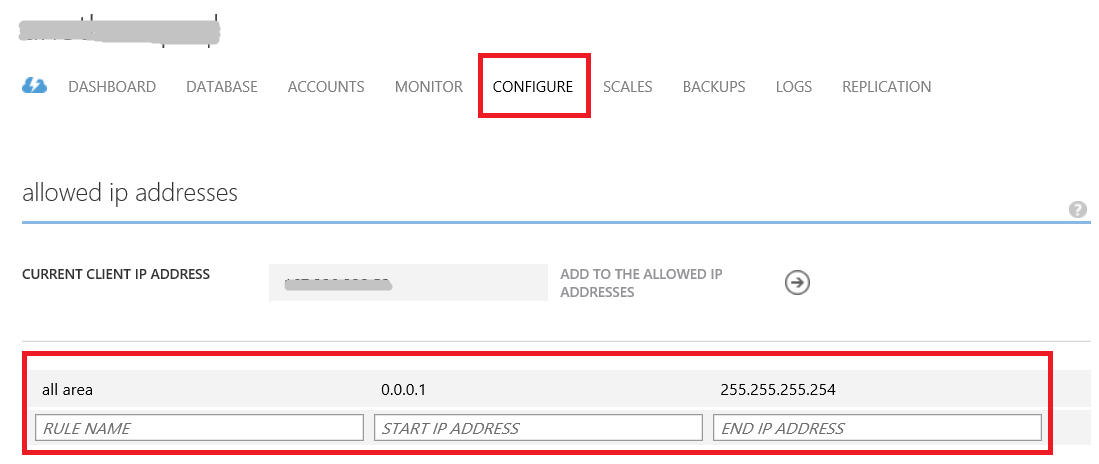






1. Set IP area that you allow to have access to your database server.

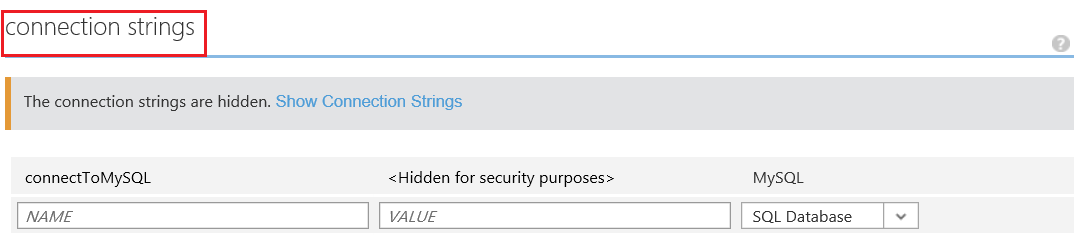
|  |
| --- |
| **NOTE:**  As we described above, we will connect to your Linux server via SSH, and all migration operations will be operated in your Linux server, so we need you to add a permission of your Linux server’s public IP to your database server.  In case that you don’t know your Linux server IP, you can add a permission like below and then delete it after the completion of migration. |



1. Add database connection string to your Web APP

|  |
| --- |
| **NOTE:**  The Format of the string VALUE is:  **Database**=*your database name*; **Data Source**=*your MySQL Server URL*; **User Id**=*your database user ID*; **Password**=*your password* |



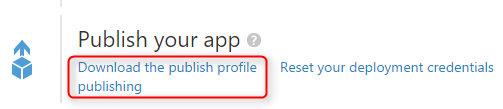


## **Download the publish profile**

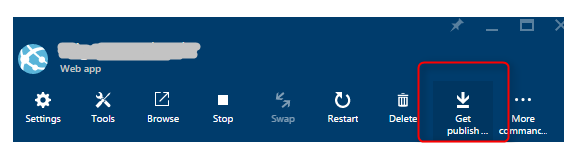
|  |
| --- |
| **NOTE:**  If you have a downloaded publish profile, please jump to step 7. |

### **For people using global Microsoft Azure**

1. If you are using classic Azure portal, please go to your Web App page and download the Publish profile.

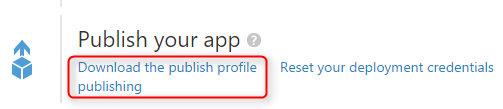


1. If you are using new Azure portal, please go to your Web Apps blade and download the Publish profile.

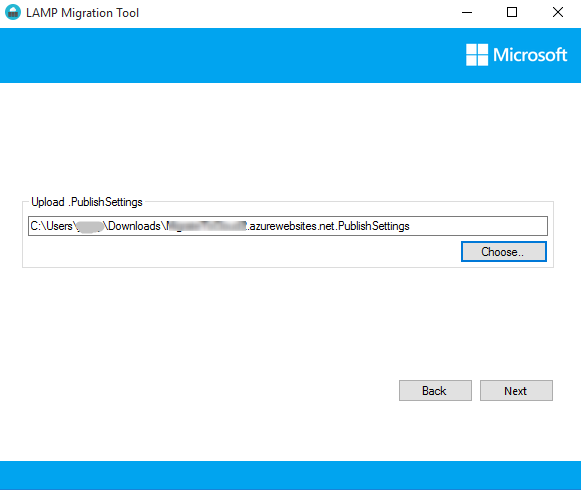


### **For Chinese using Mooncake Azure**

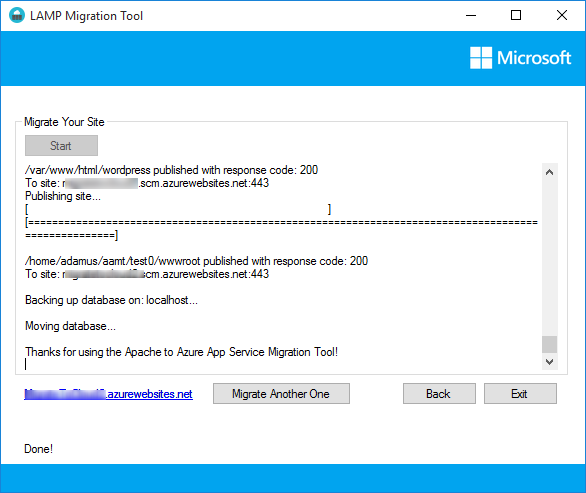
Please go to <https://manage.windowsazure.cn> and then go to your Web APP page and download the Publish profile.



## **Upload the publish profile**

****

## **Start migration**

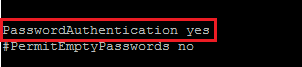


Now we have done with the migration, you can browse the new URL to get the same content.

# **Appendix**

## **Enable PasswordAuthentication**

Check the file *sshd\_config* in directory /etc/ssh/, if the setting is ‘no’, change to ‘yes’ and then restart service sshd using command ‘sudo service sshd restart’.



## **Disable Requiretty**

Check the file *sudoers* in directory /etc/, change the setting from ‘Deafults requiretty’ to ‘Deafults !requiretty’. Or for the sake of security, you can change the setting for particular user like below: Deafults:*username* !requiretty.

