

Essential Checklist for Magento Migration from One Server to Another

When it comes to [Magento migration](#) from one server to another, it can be a complex process for you, especially when you are really unknown to this process and thinking it as a straight forward job. In this blog, you will get an essential checklist that guides you through step by step in the process of moving Magento from one server to another.

First of all, make a connection through SSH between source & destination server without requiring a password. It can be achieved by tweaking OpenSSH on source as well as destination server. This tweaking leads to secure data transfer without any need of a password. To establish such connection between two machines you should consider the following steps:

1. Installation of server's public key to another machine's authorized-keys file. One can use ssh-copy-id command to accomplish such installation.
2. Make sure to set up secure permissions for both ".ssh " directories & ".ssh" parent directories.

Upon Successful establishing the connection from your source to destination servers & vice versa without requiring any password, you can go further with next i.e., server configuration, otherwise retrace the above mentioned steps (mainly the permissions element).

Configure Magento database on the destination server. If you have phpMyAdmin available on your destination server, you can follow this procedure:

- Make a decision about the MySQL database name, database user name and password.

- Collect above mentioned data and enlist it into a document.
- As an administrative user, open the MySQL client from the destination server.
- Generate a new MySQL user & related database by using the following code (just replace the data with your own data):

DB_USER – The name of your Magento database user

DB_PASS – The password for your Magento database

DB_Name – The name of your Magento database

DB_HOST – Magento database host (most likely localhost)

Creation of two shell automation scripts is required to automating key elements of transfer. Automation scripts eliminate all the chances of human errors. Create one script by keeping in mind the need of the initial sync of Magento code between source & destination servers. Second automation script should be created with the purpose of being executed in a case of Magento store maintenance.

After creating automation scripts, it is the time to put your Magento store into maintenance mode. Also update your Magento configuration & DNS records in tune with the IP address of the destination server.

Put relevant data in place of scripts variables (like SSH & MySQL user name, host name, database name, etc.) Use first automation script to proceed with the initial sync of code and media files & second script to accomplish incremental sync of the code and media files, and also a database transfer between

source & destination server.

After moving Magento code, files and database to the destination server, you should now configure your destination server environment like adding new MySQL database name, user name, password and a DBMS host to the app/etc/local.xml (In case of dedicated DB server). And also configure crontab on the user account of your new server in order to activate Magento cron.

Before proceeding with DNS record changes, it is essential for you to check once – Is everything in an appropriate order? Test whether your destination server is able to hold file as well as database or configured to host Magento installation. You should point your domain to new IP address to perform such testing. Also go through the Magento exception log, system log & reports and finally web server software log file in order to inspect any issue.

After performing final testing, you should edit your domain records so as to point it to new IP address. You should adjust "A" type record, TXT type record, and Reverse DNS record. Make sure to change Hostname by using "hostname" command.

The above checklist and step-by-step migration procedure can help you to easily manage the Magento Migration from one server to another with less transfer downtime.

Ask Envision Ecommerce if you need any assistance in getting your Magento Store up and running.