Kaltura On-Prem Iris

Version: 0



**Kaltura Business Headquarters**

5 Union Square West, Suite 602, New York, NY, 10003, USA

Tel.: +1 800 871 5224



Copyright © 2014 Kaltura Inc. All Rights Reserved. Designated trademarks and brands are the property of their respective owners.

Use of this document constitutes acceptance of the Kaltura Terms of Use and Privacy Policy.

# Contents

[Preface 4](#_Toc391998626)

[About this Manual 4](#_Toc391998627)

[Audience 4](#_Toc391998628)

[Document Conventions 4](#_Toc391998629)

[Section 1 Overview of the Kaltura On-Prem Edition 5](#_Toc391998630)

[Kaltura On-Prem Workflows 5](#_Toc391998631)

[Single Server –Kaltura Developer Machine 6](#_Toc391998632)

[Single Server – Without Internet Access 7](#_Toc391998633)

[Multiple Servers – Typical On-Prem 7](#_Toc391998634)

[Kaltura On-Prem Installation Tasks 8](#_Toc391998635)

[Section 2 Installing Prerequisites 9](#_Toc391998636)

[Installing Prerequisites on CentOS 5.9 9](#_Toc391998637)

[Installing Prerequisites on CentOS 6.3 9](#_Toc391998638)

[installing Prerequisites on CentOS 6.4, CentOS 6.5 10](#_Toc391998639)

[Section 3 Pre-Installation Tasks 11](#_Toc391998640)

[Install Prerequisite Packages 11](#_Toc391998641)

[Install Phing 13](#_Toc391998642)

[Install git 13](#_Toc391998643)

[Configure Nagios 13](#_Toc391998644)

[Installing Pentaho Data Integration for the Kaltura Analytics Module 14](#_Toc391998645)

[Configure/Set the System Configuration 14](#_Toc391998646)

[Section 4 Preparing to Run the Installation Script 16](#_Toc391998647)

[Section 5 Installing Kaltura On-Prem 18](#_Toc391998648)

[Downloading the Installer Package 18](#_Toc391998649)

[Multiple Servers Installation Order 18](#_Toc391998650)

[Running the Installation Script 18](#_Toc391998651)

[Section 6 Post Installation Tasks 20](#_Toc391998652)

[HTTPS Environment Configuration 20](#_Toc391998653)

[Start Using Kaltura 21](#_Toc391998654)

[Section 7 Installation Options 22](#_Toc391998655)

[Install Script 22](#_Toc391998656)

[Verify Script 22](#_Toc391998657)

# Preface

|  |
| --- |
| This preface contains the following topics:   * [About this Manual](#AboutthisManual) * [Audience](#_Audience) * [Document Conventions](#_Document_Conventions) |

## About this Manual

This manual describes how to install Kaltura On-Prem version IX-9.17.0 on a single server or multiple servers environment.

|  |  |
| --- | --- |
|  | NOTE: Please refer to the official and latest product release notes for last-minute updates.  Technical support may be obtained directly from: [Kaltura Support](mailto:kalturasupport@kaltura.com). |

Contact Us:

Please send your documentation-related comments and feedback or report mistakes to [knowledge@kaltura.com](mailto:knowledge@kaltura.com).

We are committed to improving our documentation and your feedback is important to us.

## Audience

This guide is intended for IT/System administrators who install the Kaltura On-Prem platform.

## Document Conventions

Kaltura uses the following admonitions:

* Note
* Workflow

|  |  |
| --- | --- |
|  | NOTE: Identifies important information that contains helpful suggestions. |

|  |  |
| --- | --- |
|  | Workflow: Provides workflow information.   1. Step 1 2. Step 2 |

Section

# Overview of the Kaltura On-Prem Edition

Kaltura provides the world’s first Open Source Online Video Platform. Over 150,000 media and entertainment companies, enterprises, SMBs, educational institutions, service providers, platform vendors, and system integrators use Kaltura’s flexible platform to enhance their websites, web services, and web platforms with advanced customized video, photo, and audio functionalities. Kaltura’s features and products enable you to easily deploy custom work-flows involving video creation, ingestion, publishing, management, syndication, engagement, monetization, and analysis.

The Kaltura On-Prem Edition is Kaltura's full featured self-hosted video platform software installed on your own servers or cloud, supported and maintained by Kaltura.

Kaltura On-Prem version IX-9.17.0 is based on the IX release of the Kaltura online video platform. Version IX-9.17.0 provides new features, infrastructure enhancements, and new API services.

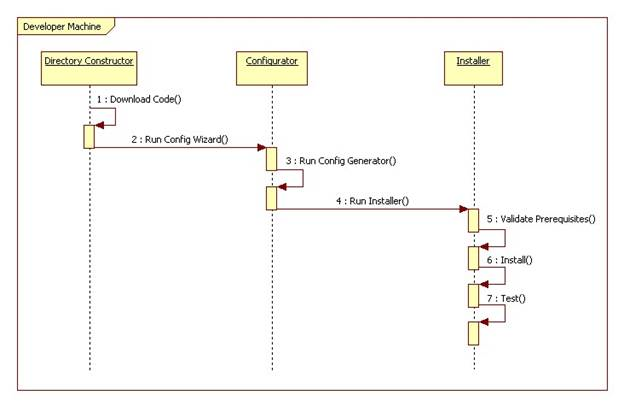
## Kaltura On-Prem Workflows

This section illustrates how the different modules, for example, the installer and packager relate to each other in the different workflows.

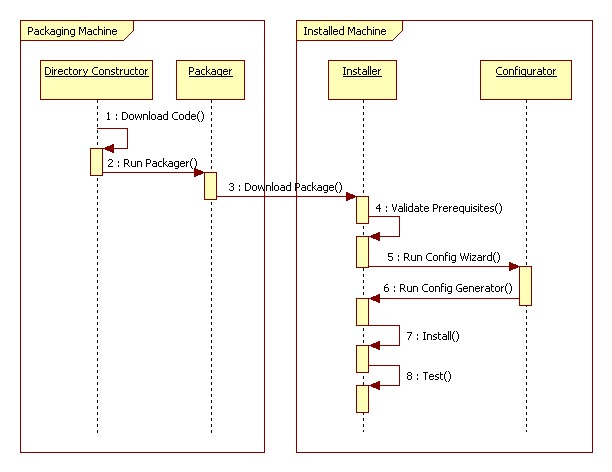
Information for the following scenarios are provided:

* [Single Server –Kaltura Developer Machine](#_Single_Server_–)
* [Single Server – Without Internet Access](#_Single_Server_–_1)
* [Multiple Servers – Typical On-Prem](#_Multiple_Servers_–)

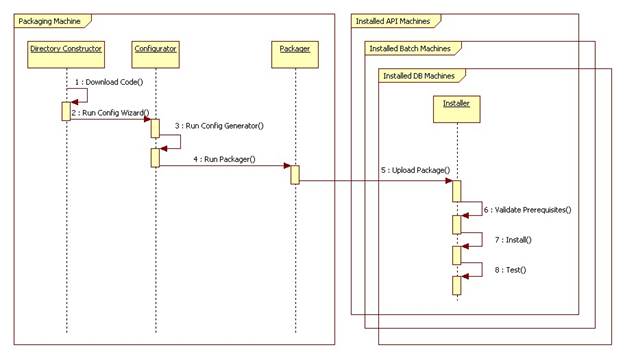
### Single Server –Kaltura Developer Machine



### Single Server – Without Internet Access



### Multiple Servers – Typical On-Prem



## Kaltura On-Prem Installation Tasks

|  |  |
| --- | --- |
|  | Workflow:   1. [Install Prerequisites](#_Installing_Prerequisites_1) 2. [Perform pre-installation tasks](#_Pre-Installation_Tasks). 3. [Prepare to run the installation script](#_Preparing_to_Run). 4. [Install Kaltura On-Prem](#_Installing_Kaltura_On-Prem_1). 5. [Perform post installation tasks](#_Post_Installation_Tasks). |

Section

# Installing Prerequisites

This section desctibes the following topics:

* [Installing Prerequisites on CentOS 5.9](#_Installing_Prerequisites_on_1)
* [Installing Prerequisites on CentOS 6.3](#_Installing_Prerequisites_on_2)
* [Installing Prerequisites on CentOS 6.4](#_installing_prerequisites_on)

## Installing Prerequisites on CentOS 5.9

Prepare additional repos:

# rpm -Uvh http://dl.fedoraproject.org/pub/epel/5/i386/epel-release-5-4.noarch.rpm

# rpm -Uvh http://rpms.famillecollet.com/enterprise/remi-release-5.rpm

Install:

# yum -y --enablerepo=remi install mysql mysql-server

# yum -y install postfix php53-gd php53-pdo php53-mysql php53-soap php53-common php53 php53-xml php53-devel php53-mbstring php53-xmlrpc php53-cli memcached glibc.i686 ImageMagick httpd dos2unix rsync mailx wget memcached php53-mcrypt sshpass redhat-lsb

# yum -y install gcc php-pear pcre-devel zlib-devel  
# yum -y install nagios nagios-plugins-ping nagios-plugins-disk nagios-plugins-load nagios-plugins-procs nagios-plugins-users nagios-plugins-dns nagios-plugins-ftp nagios-plugins-hpjd nagios-plugins-http nagios-plugins-nntp nagios-plugins-pop nagios-plugins-smtp nagios-plugins-tcp nagios-plugins-udp

Upgrade the php pear:

# pear upgrade --force Console\_Getopt

# pear upgrade --force pear

# pear upgrade-all

Build apc and memcache using pecl:

# pecl install apc memcache

Add the compiled extenstions to php.ini:

# vim /etc/php.ini

Add the following lines:

**extension=apc.so**

**extension=memcache.so**

## Installing Prerequisites on CentOS 5.10

* The commands are similar to CentOS 5.9 apart from PEAR upgrade:

# wget <http://download.pear.php.net/package/Structures_Graph-1.0.4.tgz>

# tar zxvf Structures\_Graph-1.0.4.tgz -C /usr/share/pear/

# cd /usr/share/pear

# mv Structures\_Graph-1.0.4/Structures .

# pear upgrade --force Console\_Getopt

# pear upgrade --force pear

# pear upgrade-all

## 

## Installing Prerequisites on CentOS 6.3

Install:

# yum -y install postfix mysql-server php-gd php-pdo php-mysql php-pecl-apc php-soap php-common php php-pear php-xml php-devel php-mbstring php-xmlrpc php-cli php-curl memcached zlib.i686 bzip2-libs.i686 glibc.i686 ImageMagick ncurses-libs.i686 httpd php-pecl-memcache dos2unix mysql rsync mailx cronie wget memcached sshpass redhat-lsb

Add repo for php-mcrypt and install it:

# rpm -ivh http://dl.fedoraproject.org/pub/epel/5/i386/epel-release-5-4.noarch.rpm

# yum install php53-mcrypt

Install Nagios:

# rpm -ivh http://dl.fedoraproject.org/pub/epel/6/x86\_64/epel-release-6-8.noarch.rpm  
# yum -y install nagios nagios-plugins-ping nagios-plugins-disk nagios-plugins-load nagios-plugins-procs nagios-plugins-users nagios-plugins-dns nagios-plugins-ftp nagios-plugins-hpjd nagios-plugins-http nagios-plugins-nntp nagios-plugins-pop nagios-plugins-smtp nagios-plugins-tcp nagios-plugins-udp

## installing Prerequisites on CentOS 6.4, CentOS 6.5

Install:

# yum -y install postfix mysql-server php-gd php-pdo php-mysql php-pecl-apc php-soap php-common php php-pear php-xml php-devel php-mbstring php-xmlrpc php-cli php-curl memcached zlib.i686 bzip2-libs.i686 glibc.i686 ImageMagick ncurses-libs.i686 httpd php-pecl-memcache dos2unix mysql rsync mailx cronie wget memcached sshpass redhat-lsb

Install php.mycrypt

# rpm -ivh http://dl.fedoraproject.org/pub/epel/6/x86\_64/epel-release-6-8.noarch.rpm

# yum install php-mcrypt\*

Install Nagios:

# rpm -ivh http://dl.fedoraproject.org/pub/epel/6/x86\_64/epel-release-6-8.noarch.rpm  
# yum -y install nagios nagios-plugins-ping nagios-plugins-disk nagios-plugins-load nagios-plugins-procs nagios-plugins-users nagios-plugins-dns nagios-plugins-ftp nagios-plugins-hpjd nagios-plugins-http nagios-plugins-nntp nagios-plugins-pop nagios-plugins-smtp nagios-plugins-tcp nagios-plugins-udp nrpe nagios-plugins-all openss

Section

# Pre-Installation Tasks

The following tasks should be performed prior to installing the Kaltura CE version.

* [Install Prerequisite Packages](#_Installing_CentOS_6.3_1)
* [Install Phing](#_Install_Phing)
* [Install git](#_Install_git)
* [Configure Nagios](#_Configure_Nagios)
* [Install Pentaho Data Integration for the Kaltura Analytics Module](#_Installing_Pentaho_Data_1)
* [Configure/Set System Configuration](#_Configure/Set_the_System)

## Install Prerequisite Packages

* To install prerequisites packages for Kaltura On-Prem software

1. To achieve the full potential of the Kaltura On-Prem package, install the following packages:

* MYSQL (v5.1.0 or higher)
* PHP, including PHP extensions for MYSQL, PECL, soap, pear, xml, devel, mbstring, xmlrpc, cli, memcache, pdo, gd, apc, mcrypt
* phing-2.4.13
* pear/VersionControl\_Git-0.4.4
* memcached
* zlib.i686
* glibc.i686
* bzip2
* ImageMagick
* sshpass
* httpd (apache v2.2 or higher) –if you would like your environment to work over HTTPS rather than HTTP, ensure that the ssh\_module is included in the apache configuration.
* dos2unix
* mailx
* cronie
* rsync
* java-1.6.0-openjdk
* Nagios
* Git – version 1.7.6 or above
* lsb\_release
* curl
* ntp, ntpdate
* mod\_ssl

Refer to [Installing Prerequisites](#_Installing_Prerequisites) for information about how to install the prerequisites on different Linux distributions.

1. Install **php-devel** or **php53-devel** for centro 5.x.

# yum install **php-devel**

1. If you are planning to use SFTP external storage or drop folders, the SSH2 PHP PECL extension is required. You can compile the extension using the following procedure:

# yum install make libtool

# yum install gcc php-pear libssh2 libssh2-devel

# pecl install -f ssh2

# touch /etc/php.d/ssh2.ini

# echo extension=ssh2.so > /etc/php.d/ssh2.ini

1. If you find yourself in a closed environment with only the cd-rom repo of Centos/Rhel 6.3, you will need to [install the PHP ssh2.so module manually](#php_devel_extension). This is because you won't be able to run the pecl command because it can't connect to an external website. It is also missing the libssh2-devel and php-devel packages to compile the required ssh2.so module.

* To install the PHP ssh2.so module manually

1. Run this command to get as much installed as possible:  
# yum install make libtool gcc php-pear libssh2

2. Try this command, but you should get a message that packages don't exist. If the packages exist, then there is no need for this tutorial:  
# yum install libssh2-devel

2. Install **php-devel** or **php53-devel** for centos 5.x:  
# yum php-devel

3. Download and install the RPMs onto your jump host and upload to the server. You can get them from a google search.  
php-devel-yourVersionOfPHP  
libssh2-devel-yourVersionOflibssh2  
  
# yum install \*.rpm

4. Download and extract the ssh2 PECL module:

# wget http://pecl.php.net/get/ssh2

\*For CentOS 5.10, do:

# wget [http://pecl.php.net/get/ssh2-0.11.3 -o ssh2-0.11.3.tar.gz](http://pecl.php.net/get/ssh2-0.11.3%20-o%20ssh2-0.11.3.tar.gz)

4.1 Extract the archive

(\*) For example, in CentOS 5.10 do:

# tar xzvf ssh2-0.11.3.tar.gz

5. Run the following commands:

# cd theSsh2Directory  
# phpize  
#./configure --with-ssh2  
# make  
# make install  
  
6. # echo extension=ssh2.so > /etc/php.d/ssh2.ini

7. Confirm with php -i | grep -i ssh2  
It should return something like this:

Registered PHP Streams => https, ftps, compress.zlib, compress.bzip2, php, file, glob, data, http, ftp, phar, ssh2.shell, ssh2.exec, ssh2.tunnel, ssh2.scp, ssh2.sftp, zip  
libSSH Version => libssh2/1.2.2  
ssh2  
SSH2 support => enabled  
libssh2 version => 1.2.2  
banner => SSH-2.0-libssh2\_1.2.2

1. To define the MySQL as a service running on the machine, run the following command:

# /etc/init.d/mysqld start

# chkconfig mysqld on

1. To enable the Kaltura batch mechanism to send emails using the KAsyncMailer, PostFix needs to be installed, enter the following

# yum install postfix   
# /etc/init.d/postfix start

# chkconfig postfix on

1. Although we ask the customer in our community edition requirements doc about the timezone, if they have no preference, we should use UTC by default.

Run the following:

# ln -sf /usr/share/zoneinfo/UTC /etc/localtime

# service ntpd stop

(This has to be stopped to perform the next ntpdate command)

# ntpdate 0.us.pool.ntp.org

(If this is a closed environment, make sure they have an NTP server setup! Time is important!)

Edit /etc/ntp.conf around line 20 and add:  
server 0.us.pool.ntp.org  
server 1.us.pool.ntp.org

Add to /etc/ntp/step-tickers:  
0.us.pool.ntp.org  
1.us.pool.ntp.org

# service ntpd start

# chkconfig ntpd on

## Install Phing

Required phing version is 2.4.13.

# pear channel-discover pear.phing.info

# pear install --alldeps phing/phing-2.4.13

## Install git

Install PEAR libraries of git.

# pear install pear/VersionControl\_Git-0.4.4

## Configure Nagios

Choose new password for Nagios.

# htpasswd -c /etc/nagios/passwd nagiosadmin

# mkdir /etc/nagios/conf.d.

# vim /etc/nagios/nagios.cfg

Add the following line.

cfg\_dir=/etc/nagios/conf.d

Restart Nagios.

# /etc/init.d/nagios restart

## Installing Pentaho Data Integration for the Kaltura Analytics Module

The Kaltura analytics module requires Pentaho Data Integration, version 4.2.1.

* To install the [Pentaho](http://www.pentaho.com/) Data Integration package

Run the following commands:

1. Create a new directory for Java:

# mkdir /opt/java

# cd /opt/java

1. Download Java:

# wget -O jre-6u27-linux-x64.bin http://javadl.sun.com/webapps/download/AutoDL?BundleId=52242

1. Change binary permissions, and execute it.

# chmod u+x jre-6u27-linux-x64.bin

# ./jre-6u27-linux-x64.bin

1. Create link to the new java directory.

# ln -s /opt/java/jre1.6.0\_27/bin/java /bin/java

1. Create a new directory for Pentaho:

# mkdir /usr/local/pentaho

1. Change to the new directory: cd /usr/local/pentaho/
2. Download the Pentaho Data Integration package:

# wget http://sourceforge.net/projects/pentaho/files/Data%20Integration/4.2.1-stable/pdi-ce-4.2.1-stable.tar.gz

1. Extract the Pentaho package:

# tar -zxvf pdi-ce-4.2.1-stable.tar.gz

1. Rename the data-integration root directory as *pdi*:

# mv data-integration pdi

1. Delete the downloaded file:

# rm ./pdi-ce-4.2.1-stable.tar.gz

## Configure/Set the System Configuration

* To configure the timezone

Ensure that all the servers are configured to use the same timezone.

* To configure the firewall settings

Ensure that the following ports remain open during the installation:

* 80
* 443
* 3306 – the default port used by MYSQL
* 9312 – the default port used by Sphinx
* 2049 – TCP+UDP
* Optional – used by Red5: 1935, 3590, 5080, 8088
* To set SELINUX mod to permissive

|  |  |
| --- | --- |
|  | NOTE: Kaltura On-Prem cannot run when the SELinux feature is deployed and enabled. If the SELinux feature is deployed on your server, you must set it to permissive. |

1. To edit the file, enter /etc/sysconfig/selinux
2. Set the following value: SELINUX=permissive
3. Save and exit the file.
4. To set permissive mode for the current session, run the following command:

setenforce permissive

* To configure my.cnf

|  |  |
| --- | --- |
|  | **NOTE**: If you are installing a multiple server installation with more than 1 DB machine, this process must be done on each DB machine before installation is initiated. |

1. To edit the MySQL configuration file, enter /etc/my.cnf
2. Add the following settings under [mysqld]:

* lower\_case\_table\_names = 1
* thread\_stack = 262144
* open\_files\_limit = 20000
* max\_allowed\_packet=16M
* innodb\_file\_per\_table = 1
* innodb\_log\_file\_size = 32MB

1. Verify that the following value is set: character-set-server = utf8
2. Delete the following 2 files:

/var/lib/mysql//ib\_logfile0

/var/lib/mysql//ib\_logfile1

1. Run the following command to restart the MySQL daemon service:

# /etc/init.d/mysqld restart

1. In case of multiple servers’ installation, make sure that your root user can login to the mysql server from the slave mysql machine.
2. In case of multiple servers’ installation, the database replication should be configured.  
   See mysql documentation: <http://dev.mysql.com/doc/refman/5.1/en/replication-howto.html>.

* To set services to auto-start on boot

Run the following commands:

# /etc/init.d/httpd start

# /etc/init.d/mysqld start

# /etc/init.d/memcached start

# chkconfig httpd on

# chkconfig mysqld on

# chkconfig memcached on

Section

# Preparing to Run the Installation Script

The installation script asks you for information that is required for setting up and configuring Kaltura On-Prem. The information that you provide is stored in the user\_input.ini file. Most fields have default values and can be left empty. The following fields are mandatory:

* Company name
* Host name
* Administrator email and password
* Database user name (includes create and write permissions)

You should be able to access the information when you run the Kaltura On-Prem installation script.

* To prepare to run the installation script

Locate the following information:

1. **Full path name for your Apache apachectl script** (Leave empty to use the default.)

This information is required so the Kaltura On-Prem installation script can verify your Apache setting prerequisites.

The name and location of the apachectl file is different in the apache configuration and in different Linux distributions.

For a standard apache configuration, the apachectl file usually is located at:

# /opt/lampp/bin/apachectl

For a standard LAMP configuration, the apachectl file may be available at one of the following paths:

# /usr/bin/apachectl

# /usr/bin/apache2ctl

1. **Full path name for your PHP binary file** (Leave empty to use the default.)

This information is required so the Kaltura On-Prem installation script can verify your PHP setting prerequisites.

The name and location of the PHP file is different in the apache configuration.

For a standard apache configuration, the PHP binary usually is located at:

# /opt/lampp/bin/php

For a standard LAMP configuration, the PHP binary usually is available at the following path:

# /usr/bin/php

1. **Default time zone for Kaltura application** (Leave empty to use system time zone.)

This is the default time zone that the Kaltura application uses.

If the time zone is not supplied, the time zone is detected automatically according to the settings in php.ini or (if it is not set in php.ini) according to the OS settings.

1. **Full target directory path for the Kaltura application** (Leave empty to use the default.)

The Kaltura On-Prem installation script places your Kaltura application in this directory. The target directory must be a different than the directory to which you extracted the Kaltura On‑Prem Package. The installation script creates the target directory if it does not exist. The target directory includes all Kaltura On-Prem application source code, scripts, configurations, logs, and so on. The Kaltura storage directory will be created within a web subdirectory under the target directory.

Example: /opt/kaltura

1. **The domain name/virtual host for your Kaltura On-Prem server**

The domain name (or virtual hostname) that you use for the Kaltura On-Prem installation should be set within the /etc/hosts file as follows:

127.0.0.1  mydomain.com

You are prompted to make this change at the end of the installation script.

This entry is needed so that Kaltura internal API client applications (such as the Admin Console application and the Kaltura batch application) can be directed to the relevant Kaltura web services according to their internal server URL configuration.

Examples:

kaltura.myhost.com

myhost.com

myhost.com:{port}

1. **Your primary system administrator email address**

The email address of the person who will serve as your primary system administrator for Kaltura On-Prem. This email address is set as the login username for the Kaltura On-Prem Administration Console. The person assigned to this role has full control of the video platform administration. The email address you use must be a real one, since the system uses this email address for sending system-related messages.

1. **Password for your primary administrator**
2. **Password of your system administrator for logging into the Kaltura Admin Console**
3. **Activation Key** - Kaltura OnPrem is a licensed product. In order to activate the product, you will need an Activation key. If you don’t have an Activation Key, please contact your Kaltura representative for one.
4. **Database information** (Leave empty to use the default.)

* Database hostname
* Database port
* Root user
* Root password

This is the access information to the MySQL database environment that you use for the Kaltura On-Prem. If you are planning to use a distributed setup, you can set this database on a dedicated server that can be accessed by the Kaltura On-Prem installation.

Example:

**Hostname:** localhost/<your db server hostname>

**Port:** 3306

**User:** kaltura

**Password:** kaltura

1. **Red5 installation (optional)**

This stage of the installation will ask you to determine whether to install the Red5 Media Streaming Server on the machine. Remember that this service should only be installed on the API machine(s) and not on the machines serving for batches, sphinx or the DBs.

1. **Sphinx Host (optional)**

This stage of the installation requires you to set the Sphinx host IP/Hostname- or leave it empty for default value 127.0.0.1. This value will be set in the db.ini.

1. **Work Mode**

This stage of the installation requires you to state whether the environment is going to be working over HTTP or HTTPS. Enter http/https. Default value is http.

|  |  |
| --- | --- |
|  | **NOTE: Only the apache machines should be installed with HTTPS work mode.** Batch machines and the Sphinx machines should be installed with work mode http. |

Section

# Installing Kaltura On-Prem

This section describes the following topics about installing Kaltura On-Prem:

* [Downloading the Installer Package](#_Downloading_the_Installer)
* [Multiple Servers Installation Order](#_Multiple_Servers_Installation)
* [Running the Installation Script](#_Running_the_Installation_1)

## Downloading the Installer Package

* To download the installer package

1. Run the following command to unzip the TM-Kaltura-TM-7.1.3.1.tgz file:

# tar -zxvf TM-Kaltura-TM-9.17.0.tgz

1. Run the following command to change to the new directory:

# cd TM-Kaltura-TM-9.17.0/installer

## Multiple Servers Installation Order

When installing a multiple servers’ environment, the servers should be installed according to their components dependency.

* To install the server with the API component:
  + All database servers should already be installed.
  + All sphinx servers should already be installed.
  + DWH should already be installed.
* In order to install DWH, all database servers should already be installed.
* SSL component could be installed, currently, only on the same machine that the API is installed on.

## Running the Installation Script

|  |  |
| --- | --- |
|  | NOTE: Verify that your server meets all of the prerequisites before running the installation script. See [Pre-Installation Tasks](#_Installing_CentOS_6.2). |

* To run the installation script

1. Change to a root user.
2. From the base directory of your extracted Kaltura On-Prem package, run the following command:

# php install.php

1. Follow the installation instructions and provide the required information.
2. When the installation is complete, continue with the [post installation tasks](#_Post_Installation_Tasks).

Section

# Post Installation Tasks

This section describes the following post installation tasks:

* [HTTPS Environment Configuration](#_HTTPS_Environment_Configuration)
* [Start Using Kaltura](#_Start_Using_Kaltura)

## HTTPS Environment Configuration

|  |  |
| --- | --- |
|  | NOTE: If your environment needs to run in regular HTTP mode (non-secure), change the link target of /etc/httpd/conf.d/kaltura.conf from kaltura.ssl.conf to kaltura.conf and restart the apache. |

1. **Internal API usage configuration**
   1. Tests certificate:

In case your server certificate is self-signed, invalid or expired certificate, you can internally ignore certificate errors when the API is called from the admin console, multi-accout console or the batch scheduler.

To configure the admin or the multi-account consoles, edit their configuration files, %APP\_DIR%/configurations/admin.ini, or %APP\_DIR%/configurations/var.ini

And add the following line:

settings.clientConfig.verifySSL = false

To configure the batch schduler, edit its configuration file, %APP\_DIR%/configurations/batch/batch.ini

And add the following line under [Worker] section:

clientConfig.verifySSL = false

To configure the nagios monitor tests, edit its configuration file, %APP\_DIR%/tests/monitoring/config.ini

and add the following line under [client-config] section:

verifySSL = false

* 1. Valid certificate:

When the API is called from the admin console, multi-accout console or the batch scheduler, it can be configured to use specific certificate file.

To configure the admin or the multi-account consoles, edit their configuration files, %APP\_DIR%/configurations/admin.ini, or %APP\_DIR%/configurations/var.ini

and add the following line:

settings.clientConfig.sslCertificatePath = %SSL\_CERTIFICATE\_PATH%

To configure the batch schduler, edit its configuration file, %APP\_DIR%/configurations/batch/batch.ini

and add the following line under [Worker] section:

clientConfig.sslCertificatePath = %SSL\_CERTIFICATE\_PATH%

|  |  |
| --- | --- |
|  | NOTE: If your environment was re-installed without first dropping the DB, you are required to run the Sphinx re-population scripts after the installation is complete. |

## Load Balancer Configuration

Depending on the type of load balancer used in the on-prem environment, configuration will need to be changed on the on-prem environment.

1. Local.ini – add the parameter [remote\_addr\_whitelisted\_hosts] (if it does not exist yet) and add the load balancer hostname to the list of hosts, in the following fashion:

[remote\_addr\_whitelisted\_hosts]

X = load.balancer.host.name

Wherein X is the current index off the array.

1. Uncomment the parameter:

accept\_private\_ips = 1

1. Add the following lines to the file @APP\_DIR@/configurations/apache/conf.d/enabled.kaltura.conf:

SetEnv HTTPS

RequestHeader set HTTPS on

1. Restart the apache.

## Start Using Kaltura

You can start using the Kaltura Admin Console and Kaltura Management Console (KMC).

Section

# Installation Options

This section describes the following topics:

* [Install Script](#_Install_Script)
* [Verify Script](#_Verify_Script)

## Install Script

Usage is php install.php [arguments]

* h - Show help.
* s - Silent mode, no questions will be asked.
* u - Uninstall previous installation.
* f - Force installation.
* p - Package XML path or URL.
* d - Don't validate installation.
* v - Verbose output.
* C - Comma seperated components list (api,db,sphinx,batch,dwh,admin,var,apps,red5,ssl).  
  Use \* for all default components, for example, \*,red5,ssl.

Examples:

* php install.php -s
* php install.php -C api,db,sphinx
* php install.php -C \*,red5,ssl

## Verify Script

Usage is php /root/kaltura/packages/TM-Kaltura-TM-9.17.0/installer/verify.php [arguments]

* h - Show help.
* r - Reconfigure.
* v - Verbose output.

Examples:

* php verify.php