Installing REDCap on a Windows Server2012R2 – Strategic Approach for Reaching a Stable REDCap Production Server

Table of Contents

[i. Overview & Strategic Approach 3](#_Toc511892933)

[ii. How to Use this Document 4](#_Toc511892934)

[iii. A Note about Systems Requirements: 4](#_Toc511892935)

[iv. A Note for You – Yes, You; Guidelines for Updating this Document 5](#_Toc511892936)

[Install MySQL Server (http://dev.mysql.com/downloads/) 6](#_Toc511892937)

[Install Apache (https://www.apachelounge.com/download/) 7](#_Toc511892938)

[Install PHP ( http://www.php.net/download) 7](#_Toc511892939)

[Configure Apache and PHP 8](#_Toc511892940)

[Enable MySQLi extension in PHP 9](#_Toc511892941)

[Installing & Configuring REDCap 9](#_Toc511892942)

[Addressing Configuration Check Errors 15](#_Toc511892943)

[Directory that stores user-uploaded documents is exposed to the web 15](#_Toc511892944)

[‘innodb\_buffer\_pool\_size' could be larger – RECOMMENDED 15](#_Toc511892945)

['max\_input\_vars' could be larger – RECOMMENDED 16](#_Toc511892946)

[Your web server does NOT have the PHP Fileinfo extension installed 17](#_Toc511892947)

[PHP extension "mbstring" not installed - CRITICAL: 18](#_Toc511892948)

[OpenSSL extension not installed - RECOMMENDED: 18](#_Toc511892949)

[GD Library (version 2 or higher) is not installed – RECOMMENDED 19](#_Toc511892950)

[Check if REDCap Cron job is running 20](#_Toc511892951)

[TEST 4: Check if PHP cURL extension is installed 21](#_Toc511892952)

[TEST 5: Checking communication with REDCap Consortium server 22](#_Toc511892953)

[Not Using SSL - Critical 23](#_Toc511892954)

[NOTICE: Authentication is not enabled 23](#_Toc511892955)

[“site\_admin” user should not be an Administrator 24](#_Toc511892956)

[REDCap Base URL does not begin with ‘https’ 25](#_Toc511892957)

[REDCap is not able to send emails – Critical 26](#_Toc511892958)

[Reaching an All Green Configuration Check 28](#_Toc511892959)

[Moving REDCap to the DMZ 29](#_Toc511892960)

[Before You Begin Using REDCap 29](#_Toc511892961)

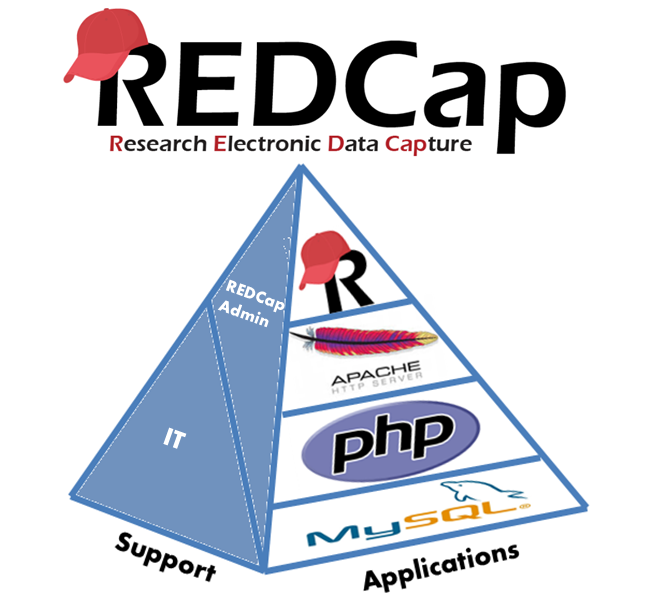
[Configuring Lightweight Directory Access Protocol (LDAP) 30](#_Toc511892962)

[Other useful references: 33](#_Toc511892963)

[Version Log in descending order 34](#_Toc511892964)

## Overview & Strategic Approach

Installing REDCap is not as easy as double-clicking an install.exe file and done. It needs a structured approach if it is to become a stable production server. Most REDCap instances are aimed to be used behind a firewall of an institution. In turn, this means that installing REDCap becomes a process in which REDCap features become the responsibility of different parties. In other words, who’s going to install the database, who’s going to make REDCap available over the inter/intra-net, what type of web-security is needed, who’s going to setup REDCap’s email and LDAP features? These are a few of the questions that need to be answered and structured to achieve a stable REDCap installation. In a few words, REDCap is a PHP application that allows for several different installation methods. In this case, a MySQL database was used, along with PHP and an Apache Server aimed to run in a Windows 2012R2 machine. The next figure gives an overview of the strategy used for installing REDCap in which responsibilities are divided among the support team to sequentially install the required applications and features.



In most cases, REDCap is fully installed by the IT department. In other cases, a clinical research department is the one initializing/starting the REDCap installation process. In the first case, all the resources needed are available within IT and installing REDCap becomes a problem of scheduling. In the second case, not all aspects of REDCap’s installation must be completed by IT personnel, which means that the approach can be divided into IT and non-IT responsibilities. In this document, the latter approach was used. However, the document can prove to be enough for IT personnel to understand all the aspects of a REDCap installation. The document doesn’t have highly technical details and information. It contains sequential steps of how to install REDCap from scratch along with tips and tricks, and references that can help any IT professional or REDCap aficionado understand what is the next installation step. The goal of the document is to arrive to a stable REDCap Production server with all its features installed. The document was created for New England Baptist Hospital, but it is voluntarily being shared with the REDCap Consortium.

## How to Use this Document

The document’s sections show the order followed for installing REDCap’s software requirements and features. The process begins by installing MySQL and 25-steps later it arrives to a stable REDCap Production server. It also contains several additional sections on optional settings. Search this document (CRTL+F) for specific error messages given to you by REDCap’s installation process and odds are that it will come up.

## A Note about Systems Requirements:

REDCap requires a server with database and web-access. Whether these are one and the same, or separate, a successful REDCap installation can be achieved nonetheless. For the most part, where REDCap’s backend is setup must be a joint decision with the IT Security Team in such a way that its installation can be protected and maintained. You can find REDCap’s tech requirements as published by project-REDCap [here](https://projectredcap.org/software/requirements/). In this case, here are the server requirements that were used:

* Database Server:
  + 8GB RAM
  + 4 Cores of CPU
  + 100GB Hard Disk
* Web Server:
  + 4GB RAM
  + 2 Cores of CPU
  + 100GB Hard Disk

## A Note for You – Yes, You; Guidelines for Updating this Document

It is the intention of this document to be shared and updated by the REDCap Community. If you come across this document and find errors, or find yourself updating or adding a section to it, then please complete your changes and re-upload the modified document to the REDCap Consortium. Try to keep things clear and provide as much detail as possible. Here are a set of guidelines to ensure that your work is successfully shared with the REDCap Community:

* Keep the document alive - there is no good or bad modification done to this document
  + Any and all modifications are welcome and I encourage you to be supportive of promoting its change and improvement in the most professional manner
* When adding new information
  + Update the document’s table of contents to make it easy for the reader to find your contribution,
  + Leave a comment of your work in the Version log section
  + Update the Table of Contents by clicking on its title and clicking “Update Entire Table”
* When fixing an error
  + Fix the error you found the best way you can,
  + Use the Version Log section to leave a comment of what was changed
  + Update the Table of Contents by clicking by on its title and clicking “Update Entire Table”
* Version Log section
  + Use today’s date ([YYYYMMDD]) in between brackets for determining the version number
  + Leave a comment of your work and use your name to sign it
  + I suggest you use a hyperlink to your LinkedIn Profile on your name so that you can be found for further questions
* Update the document’s footer with you name and document version number on the “Lasted Edited by” section
* Re-upload the updated document to REDCap’s Consortium page, using this [link](https://community.projectredcap.org/articles/39805/redcap-installation-1.html), with the version number in the file name, i.e. [20180412]REDCapInstallation.docx

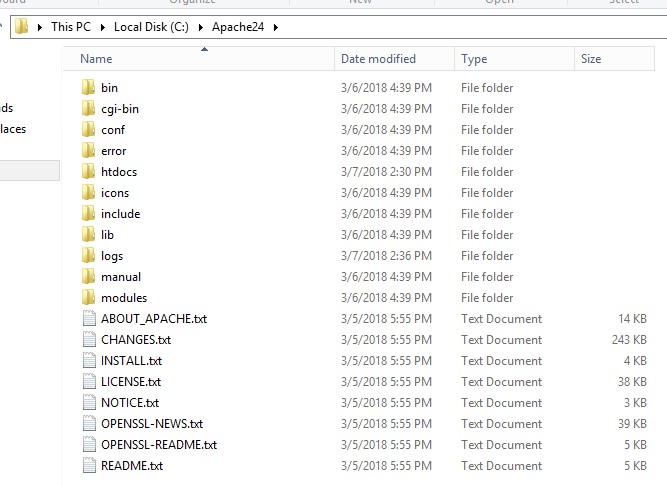
# Install MySQL Server (<http://dev.mysql.com/downloads/>)

While there are several databases that could be used for installing REDCap, we used a MySQL Server to create the database required by REDCap. The installations steps for MySQL Server are described online, but here’s a brief outline (found online) of how it can be installed.

* Install the MySQL database server on your PC. We will do this using the 'MSI' one-click installer for Windows. Go to <http://dev.mysql.com/downloads/> and download the 'MySQL Installer for Windows'. At the time of writing this was from a very obvious graphic at the top of the screen (which I missed the first time).
* Run the installation. Click...
  + Install MySQL products
  + Accept the license
  + Allow the version check (optional)
  + At 'Choose a Setup Type' accept the "Developer Default" and click Next
  + A number of downloads of required software may be identified. Click Execute and follow onscreen instructions to install them.
  + At 'Installation progress' screen, hit Execute - the MySQL software will be installed
  + At 'Configuration overview' hit Next to go to the basic configuration screen.
  + Accept all the defaults on the 'MySQL Server Configuration' and hit Next.
  + On the password screen, supply a password for the 'root' (main administrator) user. Make sure it's one you won't forget. You can also create a 'User' account just for Moodle with more restricted access if you like. This would be good practice on a public server but just using the 'root' user will suffice for testing.
  + On the Service details page, accept the defaults and hit Next and then Next a couple more times for the configuration progress.
  + Click Finish. Job done.
* MySQL Workbench will open. Under Server Administration (right hand column, double click 'Local MySQL56' (or whatever you called it). A box should pop up asking for the root password. Enter the password you supplied
* The server management screen should appear. You don't have to worry too much about this. It just shows the install is working.

# Install Apache (<https://www.apachelounge.com/download/>)

Apache installation starts by extracting all the zip-file content into the c-drive. Apache needs to be installed in C:\Apache24 – putting Apache in any other folder will cause configuration problems later on. Here’s how it should look:



The installation process requires you to manually run the install command from C:\Apache24\bin folder. Using a Command Prompt as an Administrator (right click and select Run as Administrator) do the following:

C: cd Apache\bin

httpd –k install

httpd –k start

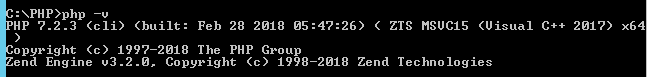
Set Apache to work on the right port so it can connect on <http://localhost> and test it.

Check the following website for reference:

* <https://httpd.apache.org/docs/2.4/platform/windows.html>

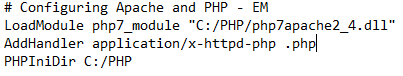
# Install PHP ( <http://www.php.net/download>)

Download, and extract php into c:\PHP. Test to see if it’s working by typing php –v in the command line under the PHP folder.

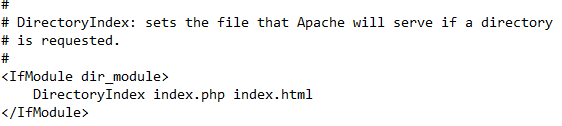


# Configure Apache and PHP

PHP is configured to work with Apache from Apache’s configuration file. The configuration file is found at: c:\Apache24\conf\httpd.conf. Edit this file (using a text editor) and add the following lines:

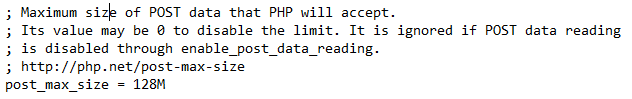


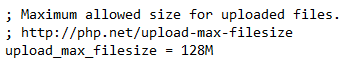
In the same way, look for DirectoryIndex and change it to:



Finally, in the PHP folder, copy the file php.ini-development and name it php.ini. In this file change the values of the following parameters:







Now restart services through Service Control, and check if [**http://localhost**](http://localhost) works.

Check the following website for reference:

* <https://docs.moodle.org/34/en/Manual_install_on_Windows_7_with_Apache_and_MySQL#Step_2:_Install_Apache>

# Enable MySQLi extension in PHP

The localhost can and will probably complain about MySQLi not being installed. MySQLi is installed at the same time as PHP. The error message means that the php.ini file doesn’t have a reference to the MySQLi extension. In order to fix this, go to the php.ini file, under the c:\PHP folder and edit it. Look for the row that says:

;extension=mysqli

Uncommenting this row will not fix the issue, because the php\_mysqli.dll file is found under the c:\PHP\ext\ folder. Add the following lines to the file (or replace the one above):



Now restart Apache by using the Apache Monitor, and check if [**http://localhost**](http://localhost) works.

# Installing & Configuring REDCap

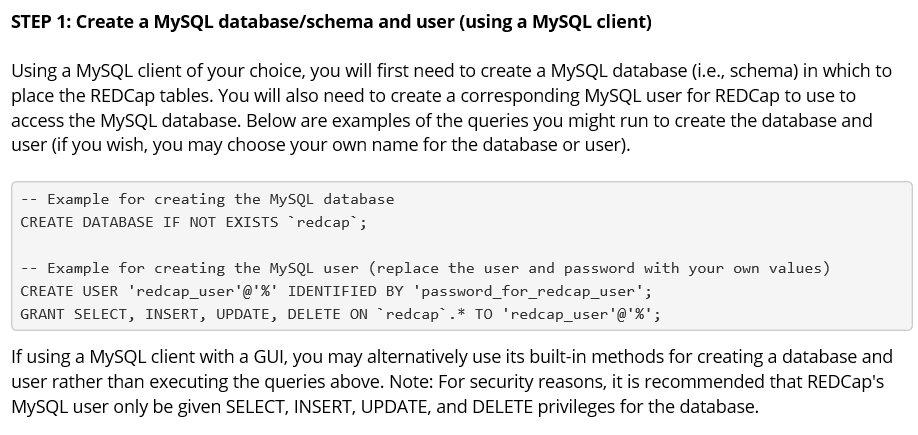
Download REDCap from the community site from the following link:

* (<https://community.projectredcap.org/users/login.html>)

and extract to the webfolder specified for Apache. In this case this webfolder is C:\Apache24\htdocs. Then, access REDCap’s installation file from the web-browser through your local host:

http://localhost/redcap/install.php

At this point REDCap is going to guide you through the remaining process. For the most part, it’s going to show you all the errors that need to be fixed. The first error needed to be fixed was that we needed to create a MySQL database/schema and user.



In the database server, open the MySQL Command Line Client and run the following SQL-statements to create the REDCap Schema:

CREATE DATABASE IF NOT EXISTS 'redcap';

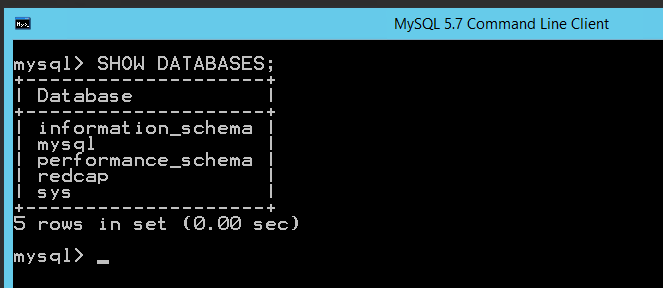
CREATE USER 'redcap\_user'@'%' IDENTIFIED BY 'password\_for\_redcap\_user';

Then create a redcap user and give it a password and grant it rights:

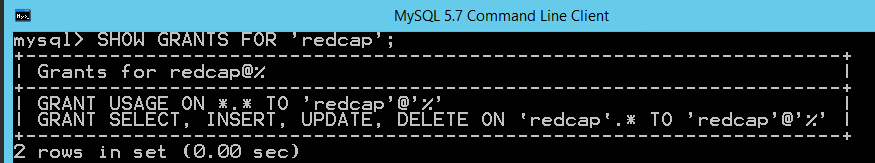
CREATE USER 'redcap\_user'@'%' IDENTIFIED BY 'password\_for\_redcap\_user';

GRANT SELECT, INSERT, UPDATE, DELETE ON `redcap`.\* TO 'redcap\_user'@'%';

Please notice that the “redcap\_user” and “password\_for\_redcap\_user” must be replaced with the actual user name and password for this root redcap account. These SQL-statements run without a problem. You can verify that the REDCap schema has been created by using:

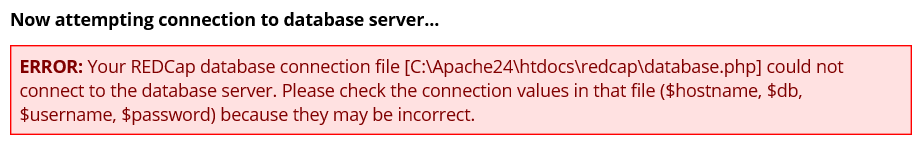


You can verify that the user was created by using:



Note: You can exit the MySQL Command Line by typing EXIT.

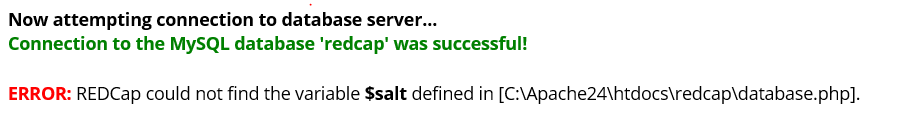
Now, you can go back to <http://localhost/redcap/install.php> and it’s going to complain that the database connection has not been setup:



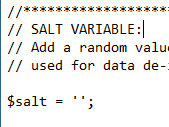
Setting-up the connection is straight-forward. The file itself states what you need to connect:

$hostname, $db, $username, and $password – these are the ones you created in MySQL Command Line.

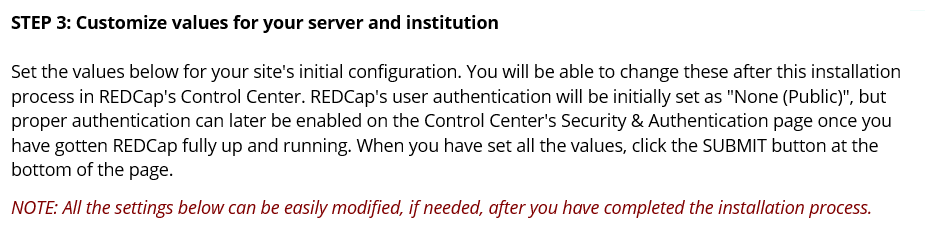
However, the database.php file also states how to set-up increased security. It is in this file where you specify the certificate to be used as well as the method for security. I’ll leave this for later. Go back and reload <http://localhost/redcap/install.php>. If successful you will see the next message:



Which means that the connection settings work, but a $Salt value needs to be specified. Go back to the database.php file and add a random alphanumeric string with at least 8 characters:



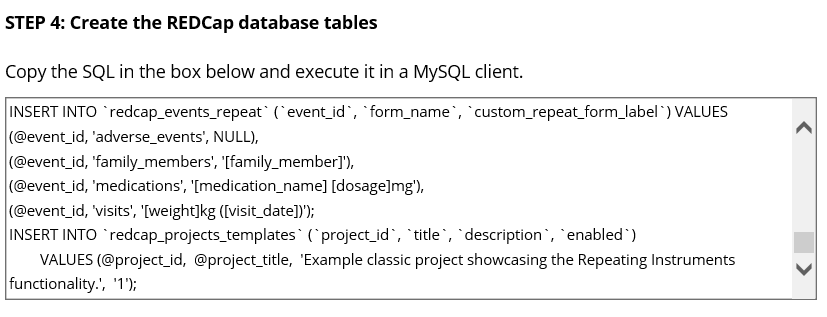
Save the file and reload <http://localhost/redcap/install.php>. If successful, the redcap installation shows you Step 3. In it, you can customize the values for the server (webserver) and institution. Here’s how it looks:



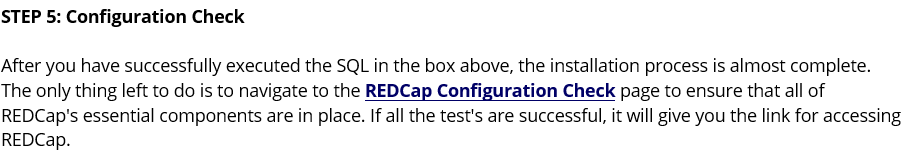
These are the settings you can specify at this point:

* Allow normal users to create new projects?
* Allow normal users to move projects to production?
* REDCap Base URL
* URL shortening service for survey links
* Date and time format
* Number format – decimal character
* Number format – thousands separator
* Contact name to display on Home Page
* Contact name email
* Name of REDCap Administrator
* Email address of REDCap Administrator
* Name of Institution
* Name of Organization at Institution

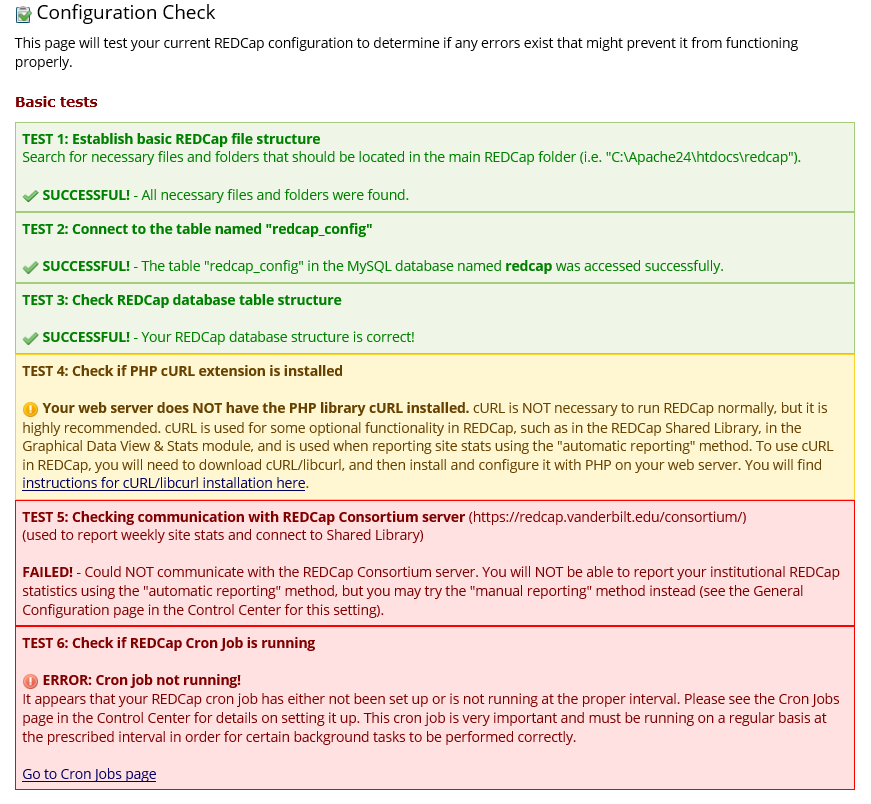
Specify all parameters and saves changes. The webpage will reload and it will show Step 4. (If for some reason you navigate away from this screen you will have to repeat Step 3, but not Step 4.) In this step the REDCap tables are created by running the SQL-Statement given in the webpage. Here’s how it looks:

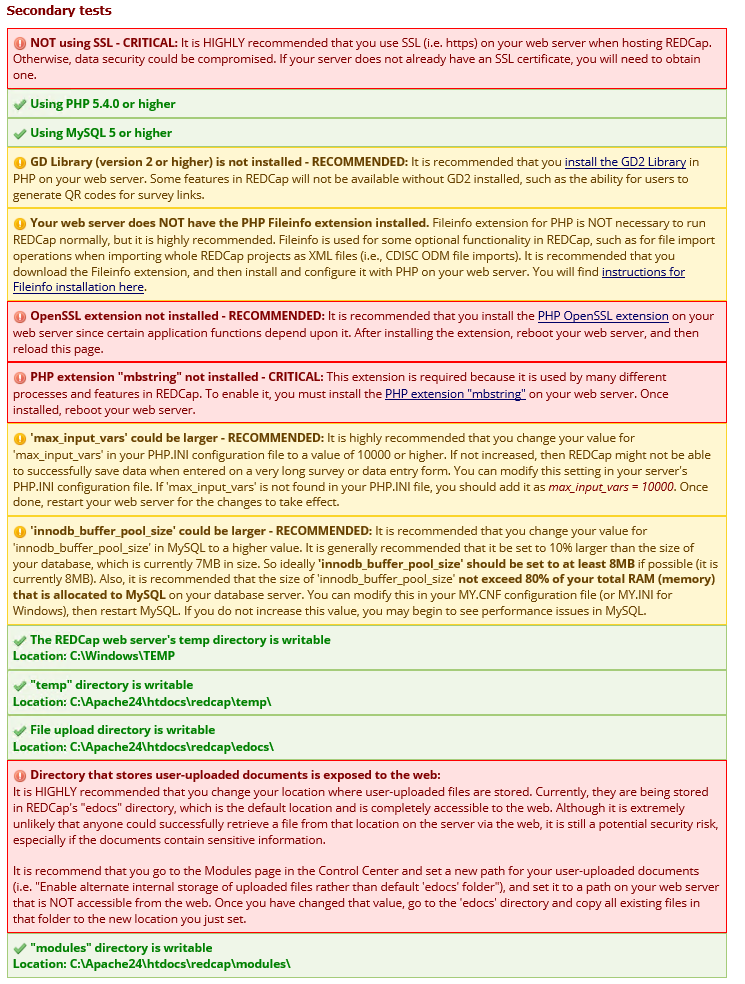


Copy the entire statement (using CRTL+A) and paste it in wordpad for now. Go to the db-server, and paste the SQL-Statement in MySQL client. The statement is very lengthy and it will execute each statement as you paste it. It will take about a minute or two for the entire command line to stop showing its work. Now, exit MySQL Command Line and go back to the webserver and click the link under Step 5 called “REDCap Configuration Check” which looks like this:

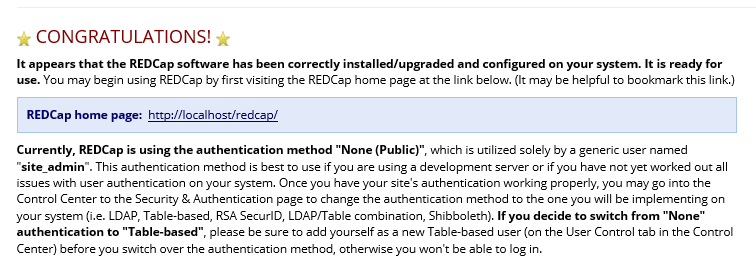


The configuration Check shows what is and is not working in the current REDCap Installation. Here’s how it looked for this case:





The installation does say that REDCap is running, but I recommend not releasing REDCap until a resolution has been reached for each of the configuration problems. Here’s how it looks:

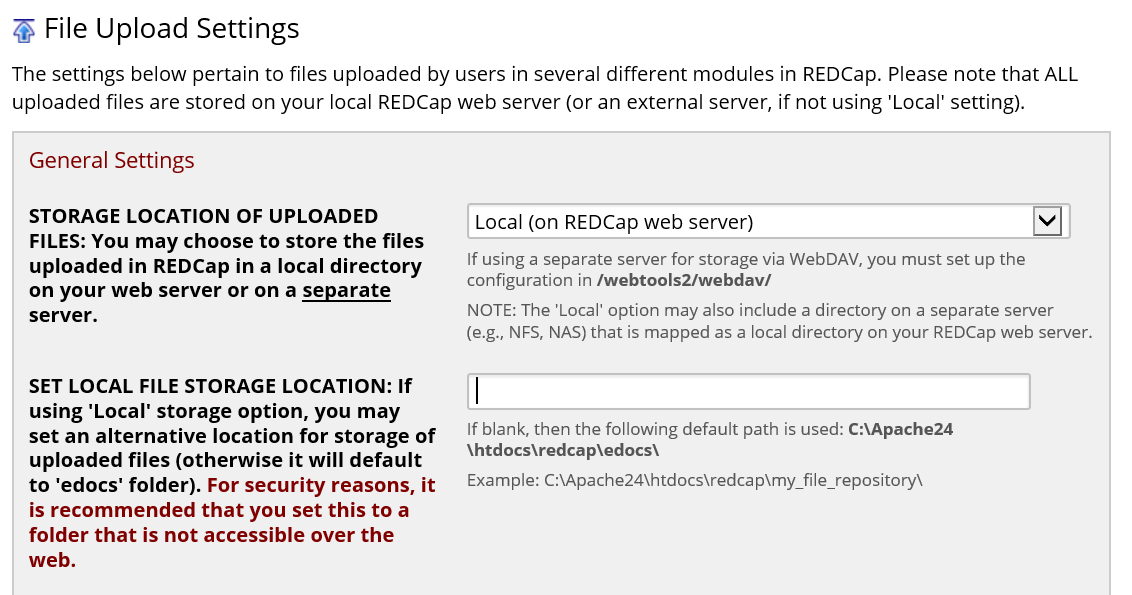


These notification errors are available from REDCap’s Control Center – Configuration Check which is available from <http://localhost/redcap/>. It is important to note that while all configuration test errors are important you don’t have to fix them all for REDCap to function properly. Go through the list of configuration errors and prioritize them according to the institution’s objectives.

# Addressing Configuration Check Errors

## Directory that stores user-uploaded documents is exposed to the web

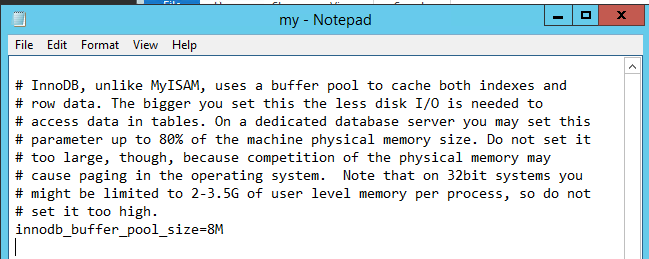
In the webserver, create a folder to store files uploaded to REDCap, through a Project’s file repository, within the REDCap folder. Once you’ve created it, copy the file in redcap/edocs into this new folder. Then go to Control Center’s File Upload Settings and set its address in the space provided. It looks like this:



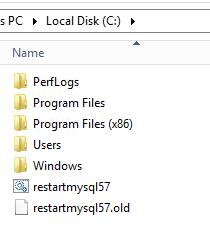
Save your changes and go the Control Center’s Configuration Check. If successful, the error message would have disappeared.

## ‘innodb\_buffer\_pool\_size' could be larger – RECOMMENDED

In the database-server, find the MySQL’s my.ini file. The value for the innodb\_buffer\_pool\_size variable needs to be changed, but it has to be 80% less than the maximum RAM available to the server and at least 10% greater than the database size. The maximum RAM memory available to MySQL can be found by looking at the My Computer Properties. It’s usually very high (8G or more). The my.ini file needs to be opened as an administrator to be able to make changes to it. Find notepad on the search bar, right click and run it as an administrator. From within notepad, find the my.ini file and modify the innodb\_buffer\_pool\_size value.



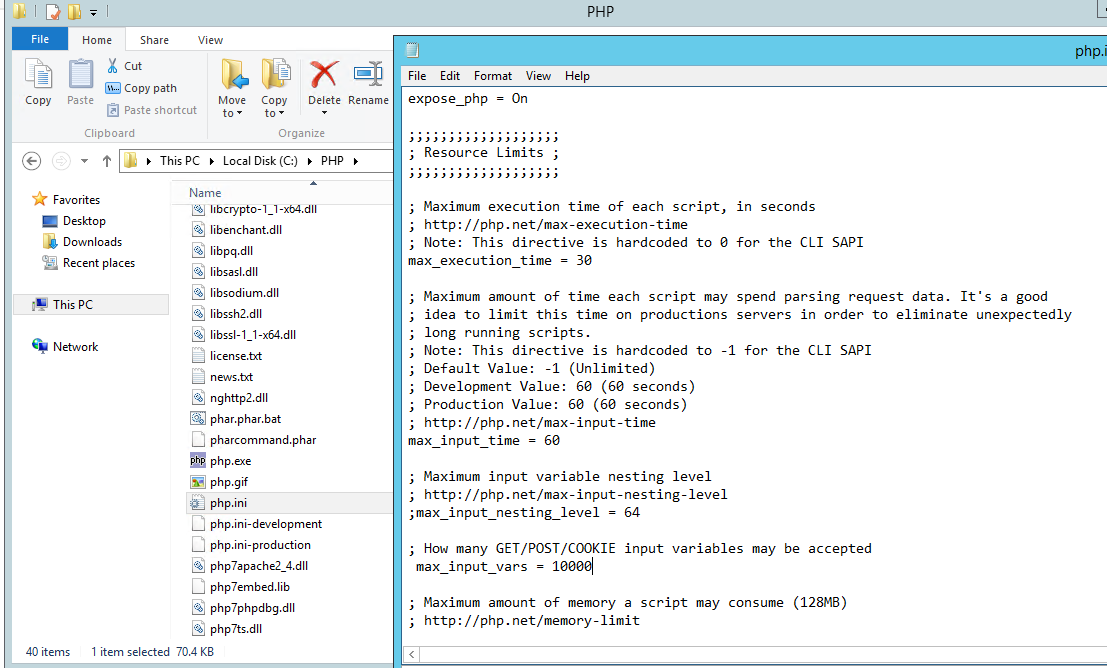
Once you’ve modified the value, restart MySQL with the restartmysql57 application and follow the prompts. The application needs to be run as an administrator (right-click and select “Run as Administrator.)



Now access REDCap from the web-server and check its Configuration Check. Notice that the error message is no longer listed and there’s no feedback that the error has been fixed – it’s just not listed anymore. If the error comes back, it’s because you haven’t set it to the right amount.

## 'max\_input\_vars' could be larger – RECOMMENDED

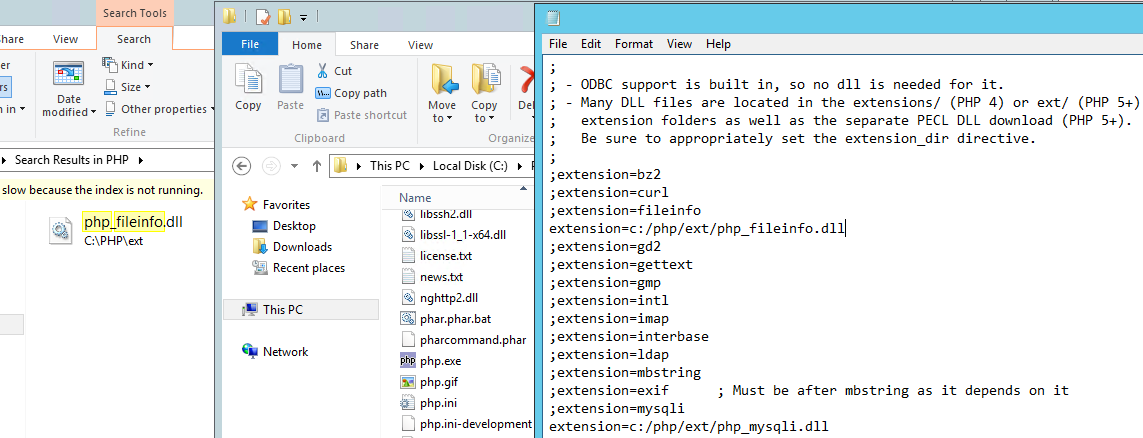
In the webserver, go to the PHP folder and find the php.ini file. Search for the variable max\_input\_vars, uncomment it and change its value to 10000 or higher.



Then restart PHP services by using the Apache Monitor to stop and start services. Again, go back to REDCap’s Configuration Check module and you’ll notice the error message is no longer present.

## Your web server does NOT have the PHP Fileinfo extension installed

The PHP Fileinfo extension has to be enabled in the php.ini file. Before it is enabled, make sure it was installed in PHP. Search for it and if you find it in your machine, then update php.ini extension (in the same section where you enabled php\_mysqli.dll). Search the php.ini file for fileinfo and update that row by, either uncommenting the row or updating it with the file’s actual location.

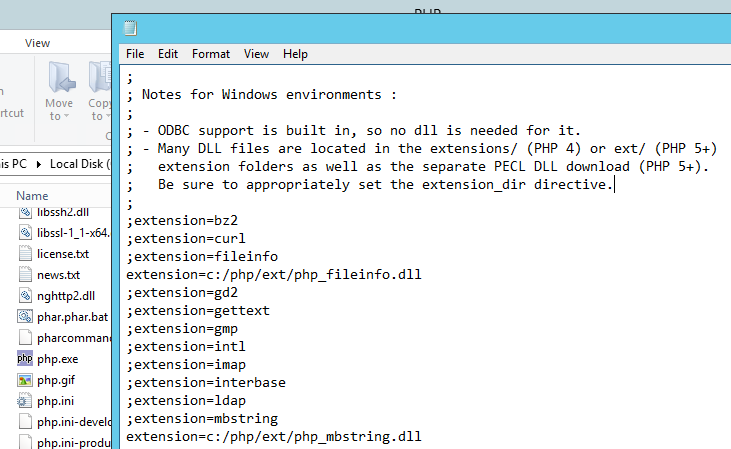


Then restart PHP services by using the Apache Monitor to stop and start services. Again, go back to REDCap’s Configuration Check module and you’ll notice the error message is no longer present. It now shows as “installed.”



## PHP extension "mbstring" not installed - CRITICAL:

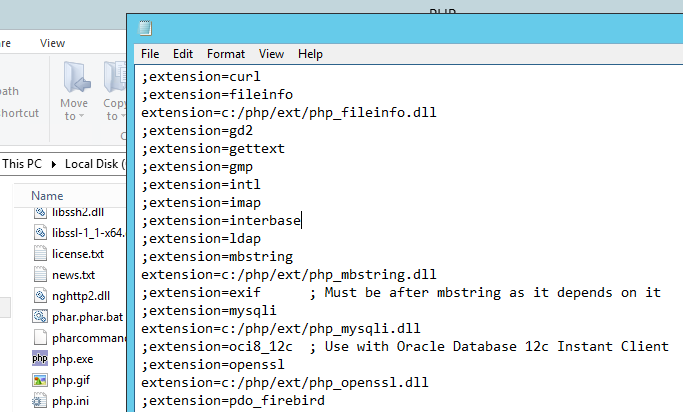
The mbstring has to be enabled in php.ini file. First, check whether it is already within PHP. If it is, then enable the extension in php.ini. If it’s not, then install it and then enable. In order to enable it, remove the comment or add a new line with the actual file location.



Then restart PHP services by using the Apache Monitor to stop and start services. Again, go back to REDCap’s Configuration Check module and you’ll notice the error message is no longer present.

## OpenSSL extension not installed - RECOMMENDED:

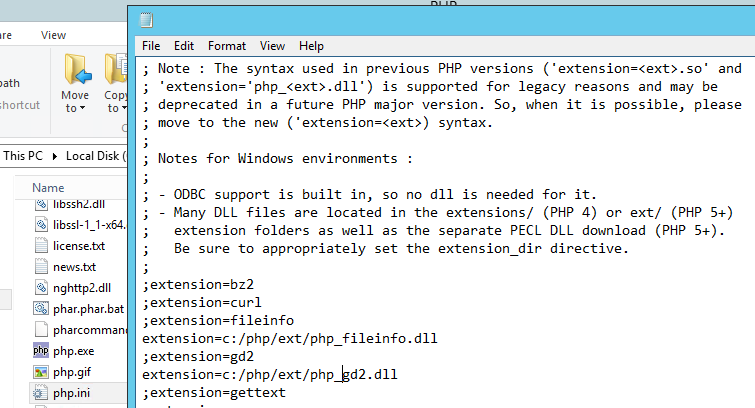
The openssl extension has to be enabled in the php.ini file. If the library exists within php, then call it from the php.ini file. In this file, search for openssl, uncomment the line or add a new line with the actual file location.



Then restart PHP services by using the Apache Monitor to stop and start services. Again, go back to REDCap’s Configuration Check module and you’ll notice the error message is no longer present.

## GD Library (version 2 or higher) is not installed – RECOMMENDED

The gd2 extension has to be enabled in the php.ini file. If the library exists within php, then call it from the php.ini file. In this file, search for gd2, uncomment the line or add a new line with the actual file location.

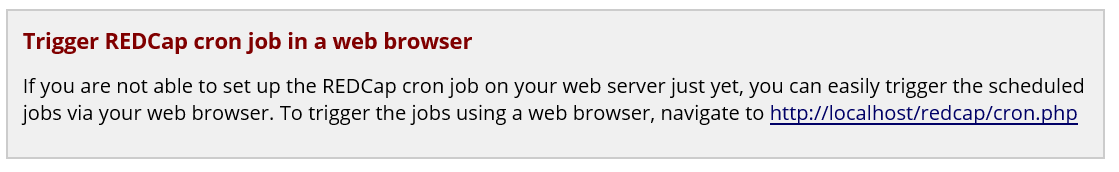


Then restart PHP services by using the Apache Monitor to stop and start services. Again, go back to REDCap’s Configuration Check module and you’ll notice the error message is no longer present. It shows that the library has been installed.



## Check if REDCap Cron job is running

In REDCap Control Center, go to System Configuration and find and click on Cron Jobs at the end of the list; it’ll take you to the Setting up the REDCap Cron Job application. There are two ways of triggering the REDCap’s Cron Job: by the command line or by a link. The easiest way of removing this error message is by clicking on the link provided in this application. The link runs the cron.php file within REDCap. You can click this link or open the file itself from the web-browser. Here’s where the link can be found:



The issue can be fixed permanently from the command line. The steps are outlined by this application. Nevertheless, here’s what you need to do. You need to schedule a task in Windows. Here’s how to do it:

* Open the command line as an Administrator (right click on it)
* and paste the code provided:

schtasks /create /tn "REDCap Cron Job" /tr "C:\php\php.exe C:\Apache24\htdocs\redcap\cron.php" /sc MINUTE /ru SYSTEM

If successful, you will see the following message:

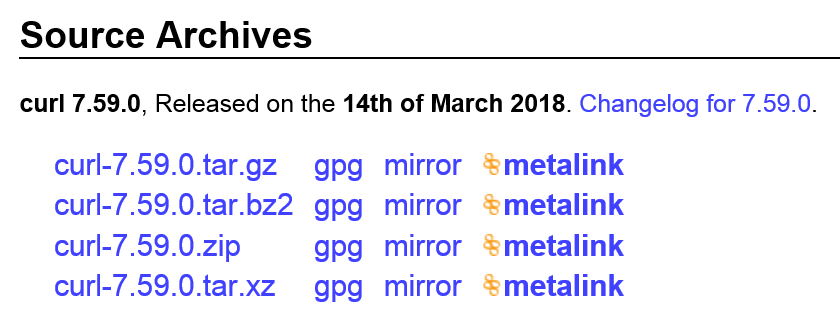


Now, go back to the Configuration Check and you will see the Cron issue resolved:

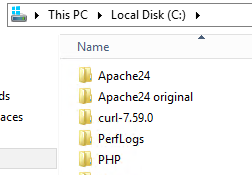


## TEST 4: Check if PHP cURL extension is installed

The error description says to install curl and its library. REDCap gives you a [link](http://us.php.net/manual/en/curl.setup.php) that takes you to the PHP documentation and guides you through the cURL documentation. CURL’s requirements states that libcurl needs to be installed. It can be downloaded from the link provided on the website. The file that was used and downloaded during this installation was curl-7.59.0.zip, because we’re working on a Windows environment. The link use in this case was: https://curl.haxx.se/download.html.



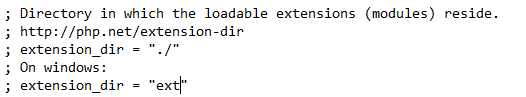
The file was extracted, and its main folder was placed on C:/ as shown in the following figure:



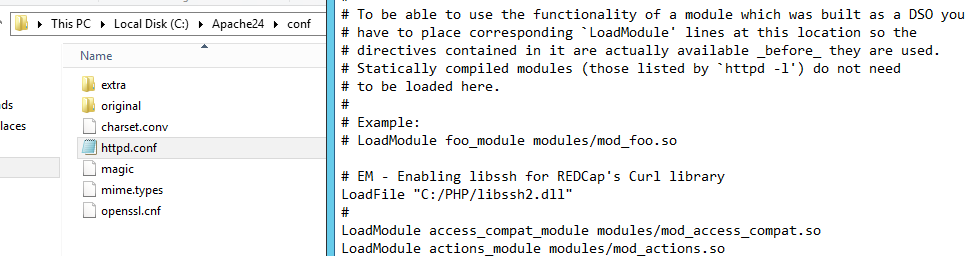
The next step is to enable the curl extension in PHP’s php.ini file. As before, the extension needs to be uncommented for it to be enabled:



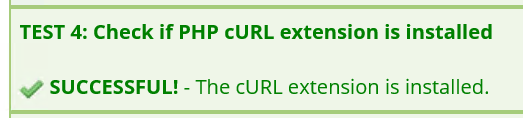
**Note:** Notice that the extension directory can be set as a constant and that avoids the need to set the extension’s file location for every extension that is to be included.



The next step is to add the PHP’s libssh library to Apache’s configuration file. In Apache’s configuration file (httpd.conf) add the following lines (LoadFile "C:/PHP/libssh2.dll") to the place in the file were modules are being loaded.

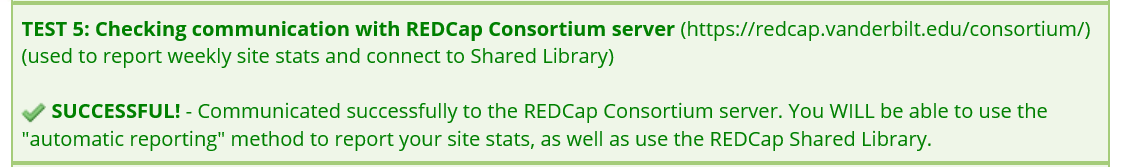


The next step is to restart Apache’s services since we made a change to its configuration file. Restart PHP services by using the Apache Monitor to stop and start services. Again, go back to REDCap’s Configuration Check module and you’ll notice the error message is no longer present. If successful the message for TEST4 would look as follows:



## TEST 5: Checking communication with REDCap Consortium server

The communication error is depended on the cURL library. Once all the software requirements for cURL were installed and the appropriate modules and extension had been enabled in php.ini and Apache’s httpd.conf file and Apache’s services are restarted, TEST 5 error message should be fixed and displayed as follows:



## Not Using SSL - Critical

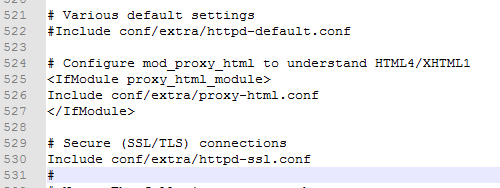
Fixed by IT – here we went from localhost/redcap to redcapwb2.vanderbilt.org, by creating a DNS host record and the SSL error message was resolved.

In short, here are the steps needed for addressing the SSL error. Using the following webpage as reference, we did:

1. <http://www.entrust.net/knowledge-base/technote.cfm?tn=6555>I
2. Move the certificate files (cert, CAroot, and key) over to the server.
3. remove encryption from cert file
4. Copy these three files in the C:\Apache24\conf folder.
5. I then edited the httpd.conf file and turned on the SSL option per the link I on step 1.
6. Restart Apache server

Here is how the modifications to httpd.conf file look:



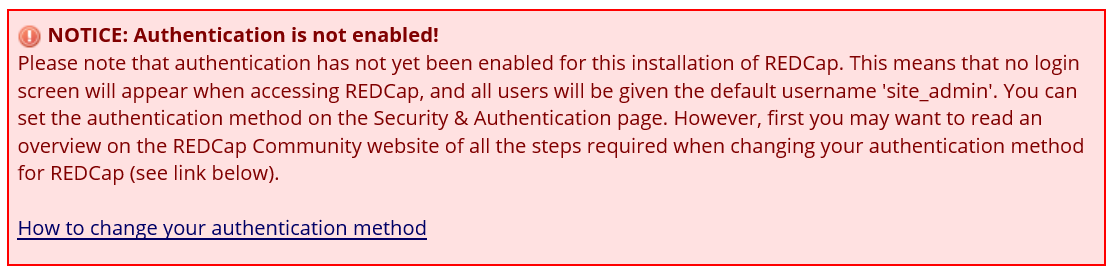


## NOTICE: Authentication is not enabled

**NOTE #1:** this error message needs to be completed after all configuration (critical) issues have been resolved.

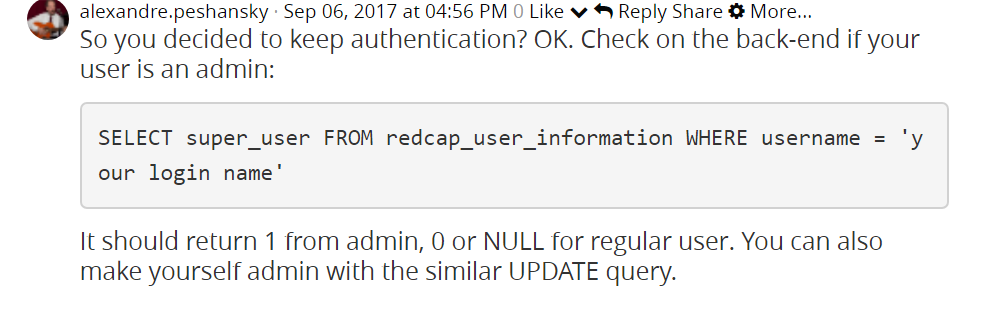
**Note# 2:** The configuration steps described in the consortium for Authentication did not work for me – this might not make sense now, but keep reading and it will: in order to avoid getting locked-out from the Control Center give administrator rights to the user you create before you try to set its password and maybe that will work.

The Authentication error message can be found under “Notifications & Reporting” as REDCap is being initially configured. Here’s how it looks:



The instructions in the consortium didn’t work and I got locked out of Redcap’s Control Center. The steps I followed were the following:

* I added a user,
* I tried to set its password but wouldn’t let me because the Authentication Method was set to Public (None).
* In Security & Authentication Configuration, I set it to LDAP & Table-based and saved it. I had to logout so I could set my password.
* After setting my password, I logged into REDCap, but it wouldn’t let me into the Control Center, because my user was not set as an Administrator.
* I enabled my user as a super user (Administrator) through MySQL database following the steps in this post found in REDCap’s consortium:
  + <https://community.projectredcap.org/questions/26465/cannot-access-controlcenter-after-enabling-authent.html>



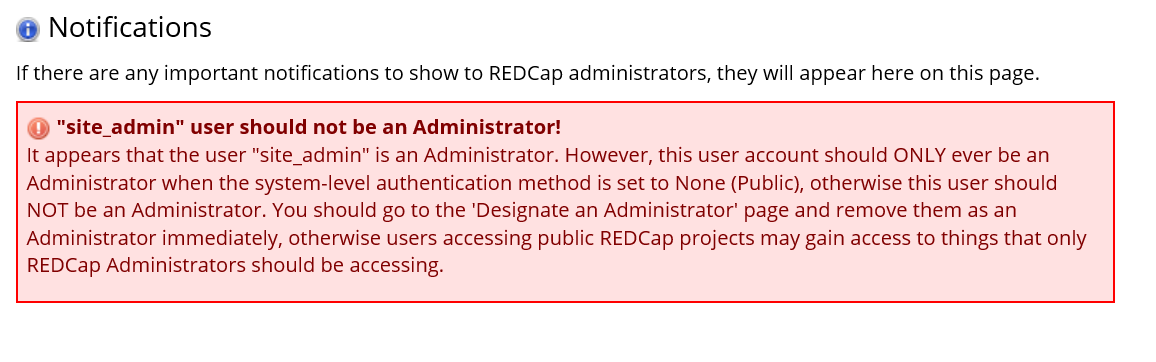
* Then I was able to continue the configuration
* At this point you have a working Administrator account and your selected Authentication method set.

Refer to his link for more information:

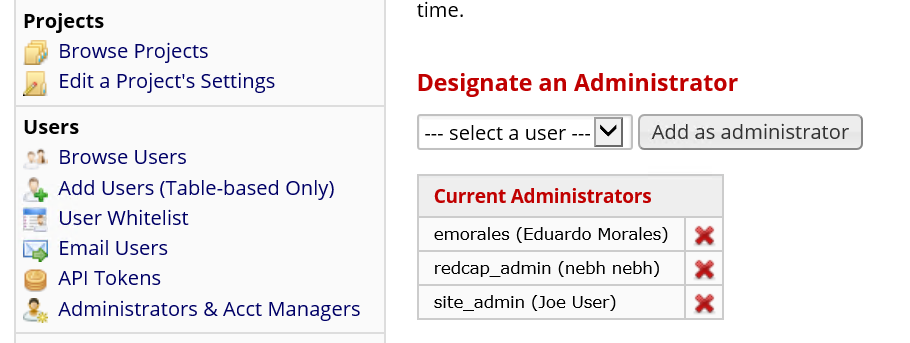
<https://community.projectredcap.org/articles/691/authentication-how-to-change-and-set-up-authentica.html>

## “site\_admin” user should not be an Administrator

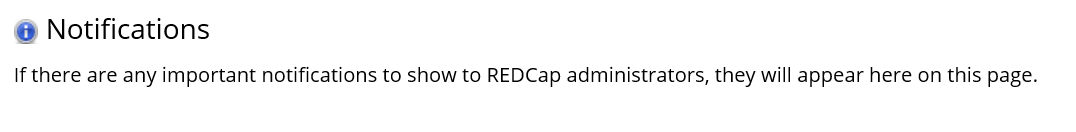
The site\_admin error message appears after you fix the SSL configuration error message. The error message means that the default “site\_admin” user must be removed from the Administrator list. Here’s how the error message looks like in the Control Center’s Notifications & Reporting:



In order to fix the issue, you have to go to the Control Center’s Administrators & Acct Managers and remove site\_admin from the list of current administrators. Here’s how it looks before the site\_admin user is removed from the list:

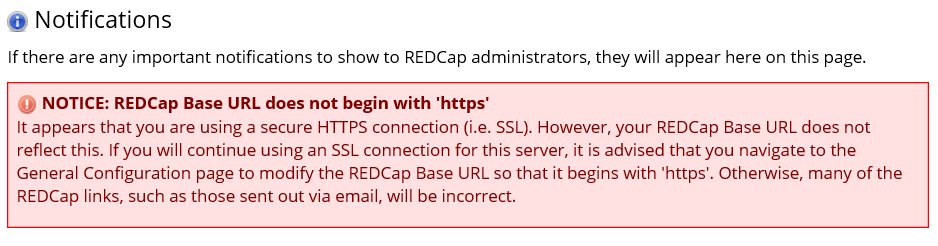


Then, you go back to Notification & Reporting and you will see that the notification error is no longer there:

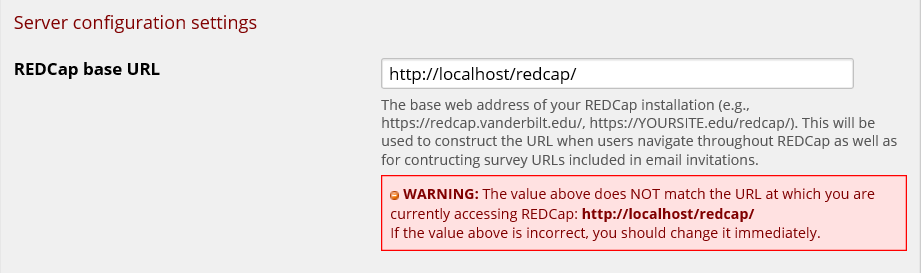
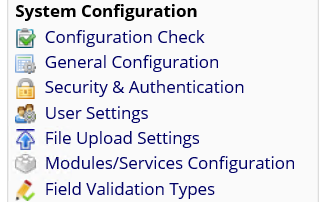


## REDCap Base URL does not begin with ‘https’

After installing the SSL certificate REDCap will display new configuration error messages. One of these errors is that REDCap was set-up as http and not https, but since at this point the https has been installed, by installing the SSL certificate, then REDCap picks up the change and asks you to update the configuration. The error message can be found in the Control Center’s Notification & Reporting app. Here’s what the error message looks like:



The error message is fixed by modifying REDCap’s base URL found in the General Configuration app found under Server Configuration Settings.

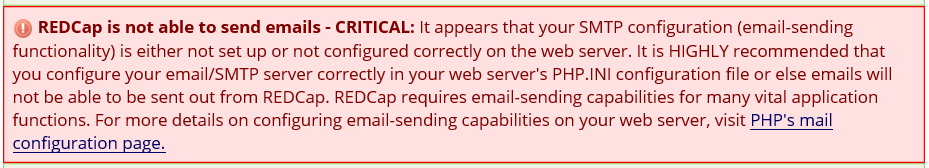


In this case, the base URL was changed to <https://redcap.vanderbilt.org/redcap> as established by the SSL connection. Save the changes and you’ll see, in Notification & Reporting, that the error message is no longer present.

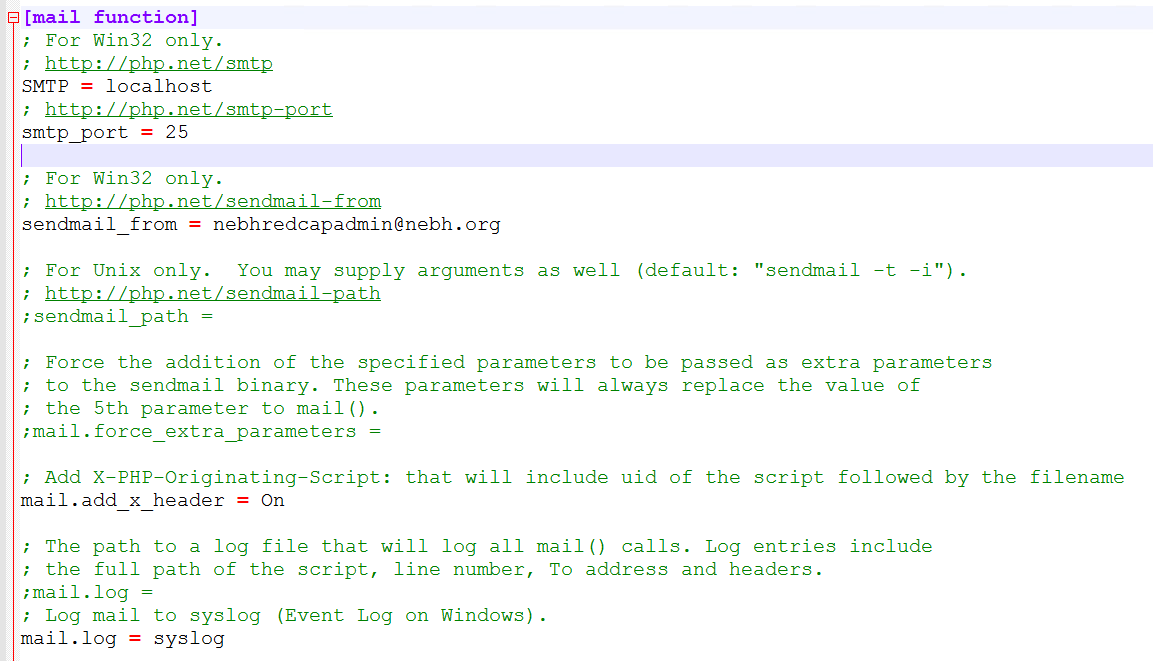


## REDCap is not able to send emails – Critical

REDCap’s emailing capability is configured within the php.ini file. REDCap doesn’t show this emailing-configuration error until later in the configuration process. Here’s how it looks:



The first step is to set the SMTP value. It expects the value of the SMTP server which in general looks like server4343.phpmail.com. Here’s a screenshot of the php.ini section for configuring REDCap emailing services. In this figure, localhost would be replaced by the SMTP value of server4343.phpmail.com.



The settings above work only if the SMTP server has already been installed and configured – which it was in this case. If not, here are hints and pointers of how to configure REDCap’s emails services if no SMTP server has been installed.

1. <https://community.projectredcap.org/questions/34315/email-setup-on-windows-server.html>
2. <https://support.office.com/en-us/article/how-to-configure-iis-for-relay-with-office-365-eb57abd2-3859-4e79-b721-2ed1f0f579c9>

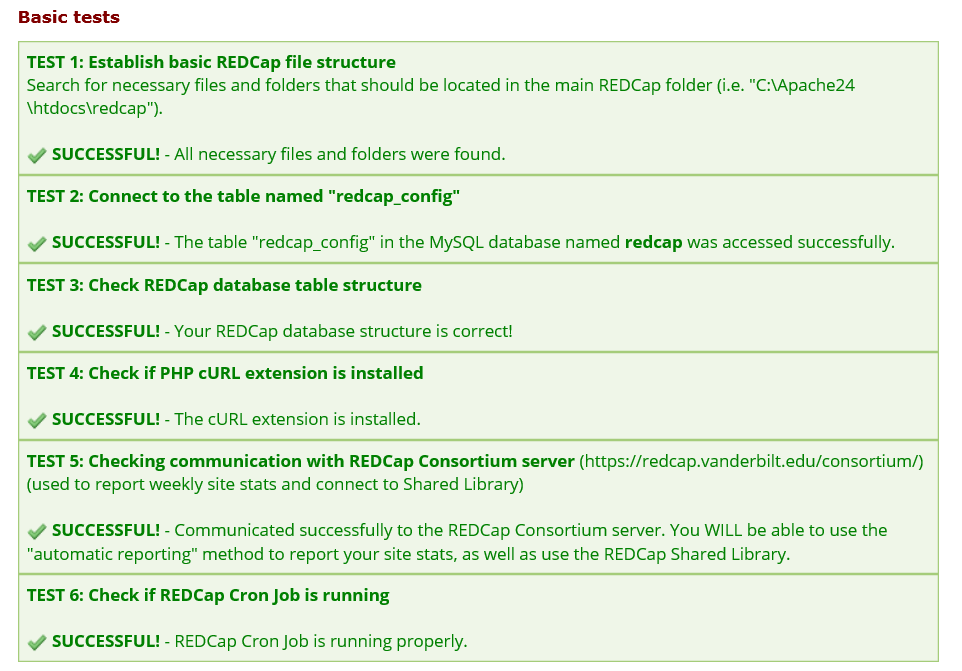
Additionally, please refer to the following link for further details about how to configure the php.ini file for enabling the email service.

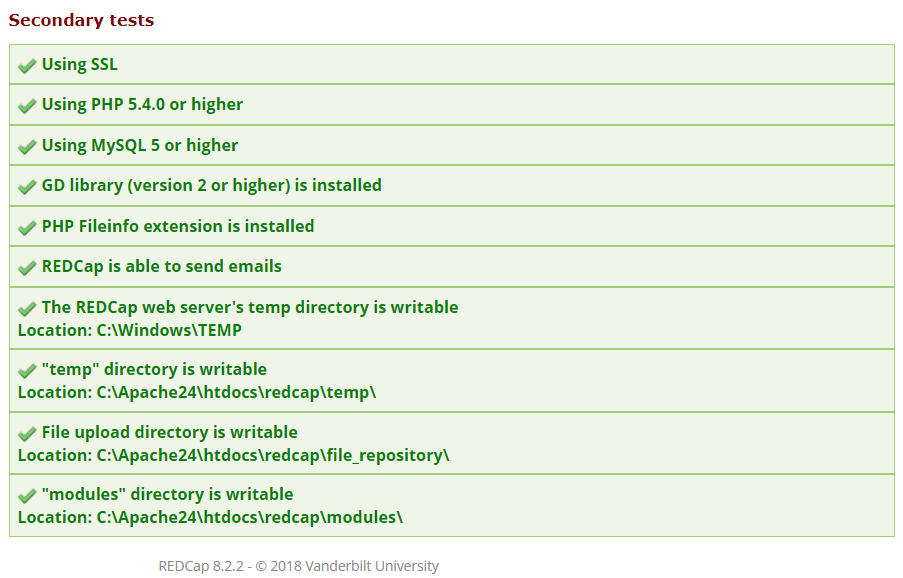
<http://php.net/manual/en/function.mail.php>

REDCap’s mailing feature can be tested by emailing existing users from the Control Panel or by adding a new user to which REDCap will email a link for setting up his/her password.

## Reaching an All Green Configuration Check

Finally, at this point all of the configuration checks shown by the Control Center have been addressed and fixed. Here’s how it looks:





At this point, I believe REDCap reaches a functional, stable configuration in which it can be used for its intended use. Nevertheless, it is important to test it thoroughly looking for odd behaviors. For instance, three remaining issues were found in this installation:

* REDCap’s internal Date Format was set to DD-MM-YYYY
* LDAP was not configured
* REDCap’s url was loading the “It works!” webpage – Redcap was only available from redcap.institition.com/redcap

These settings cannot be configured from within REDCap. They are lingering issues within the configuration of MySQL, PHP, or Apache.

# Moving REDCap to the DMZ

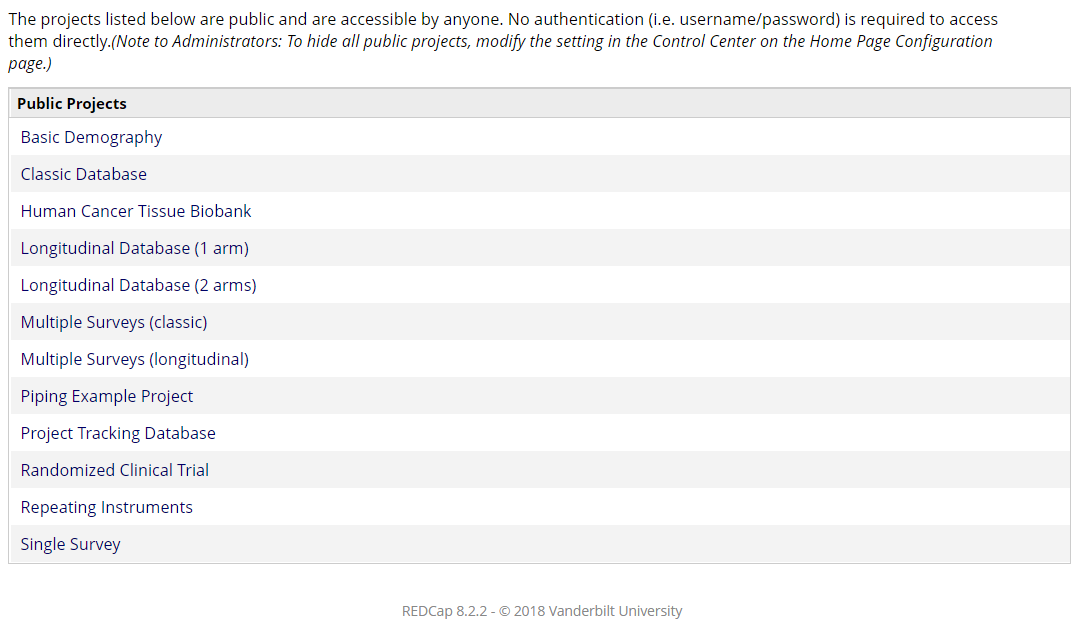
Moving REDCap to the DMZ is a task for IT. After the task was completed, REDCap configuration check reported two items failing:

1. TEST 5: Checking communication with REDCap consortium server
2. REDCap is not able to send emails – CRITICAL
3. Additionally, REDCap became very slow

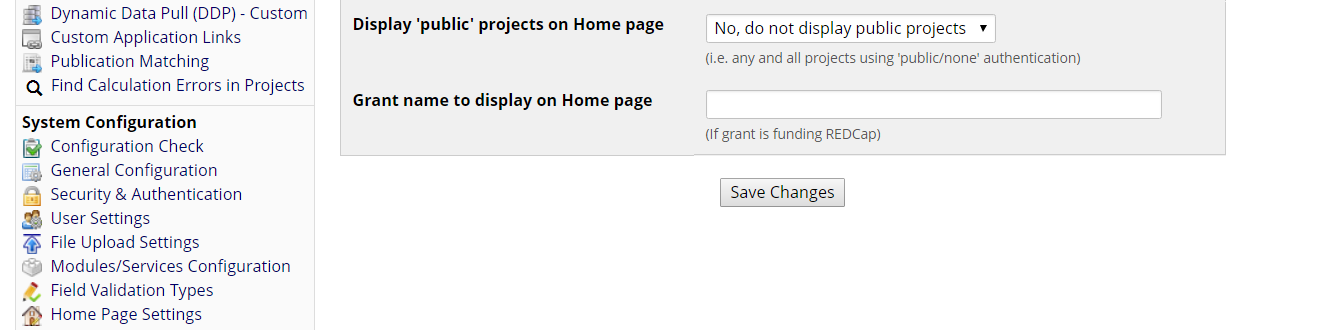
It seems that the Institution’s firewall is disabling the current configuration for both of these features. These items were addressed by IT.

# Before You Begin Using REDCap

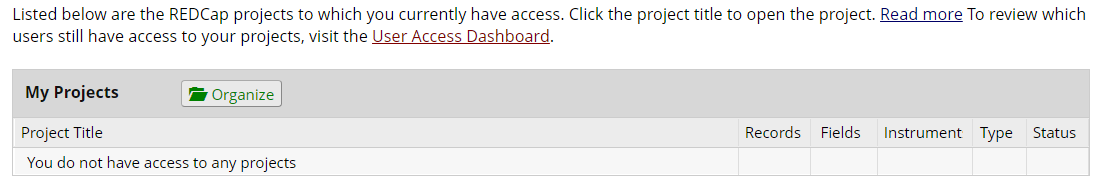
After a fresh installation of REDCap has been completed, REDCap displays Public (default) Projects that can be accessed by anyone with a valid REDCap account under their respective “My Projects” section. If these projects are not needed make sure they are removed. Here is how these project look:



These projects are removed from this list under the Control Center “Home Page Settings” app. In the section of “Display ‘public’ projects on Home Page”, select “No, do not display public projects” and save changes.



Go back to “My projects” and you’ll see that the public projects are not being displayed:



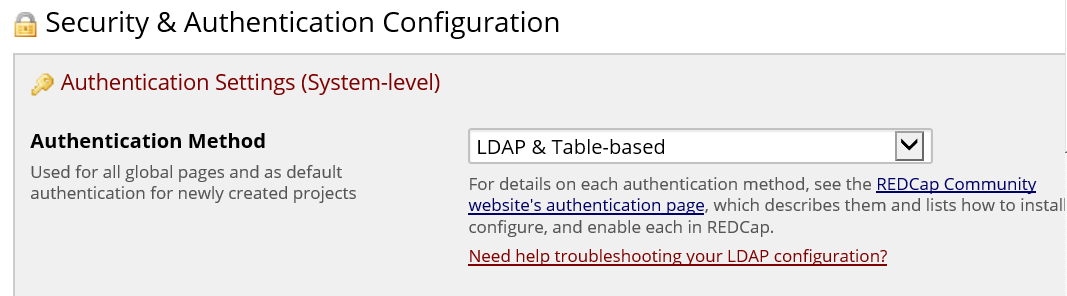
# Configuring Lightweight Directory Access Protocol (LDAP)

There are two aspects for successfully configuring LDAP in REDCap: the first is to configure PHP and Apache LDAP services and the second is requesting LDAP credentials, from the institution’s LDAP manager, and configuring it in REDCap. Enabling LDAP authentication allows REDCap users the options of using their institutional passwords for accessing REDCap too. Managing REDCap users with LDAP authorization is a topic that is not discussed here. The following paragraphs highlight how to troubleshoot and configure LDAP services in PHP and Apache.

The easiest way to figure out if LDAP has been configured in REDCap is to try to login into REDCap using credentials that are expected to work under LDAP authentication. That is, use an existing REDCap user login name and LDAP password. If LDAP is working as expected, the user should be able to login into REDCap. If not, after trying to login, REDCap will reload and state that LDAP has not been configured. Here how it looks:



REDCap provides an application that tells you the current status of the exiting LDAP configuration. It is located in the Control Center, under the Security & Authentication Configuration, in the Authentication Method when LDAP & Table-based has been selected. The link “Need help troubleshooting your LDAP configuration?” tells you the LDAP status. Notice the mentioned link in the following figure.



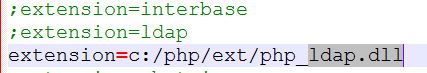
On a fresh REDCap installation, or when LDAP has not been configured, REDCap displays the following message on the LDAP troubleshooting link:



The previous message means that PHP’s LDAP extension has not been enabled. There are several steps needed to enable LDAP in a Windows Server. In a nutshell, PHP for Windows does not include a set of files needed for LDAP to run. These files exist within the Windows system, but need to be referenced to PHP. The files needed are the following: libeay32.dll and ssleay32.dll. Once PHP has access to these files, the rest of the steps require enabling an extension in PHP and a module in Apache. Then, Apache needs to be restarted for the changes to take place. Enabling LDAP in PHP must begin by referencing the system files to PHP. Here are the steps:

* Steps found at (<http://php.net/manual/en/faq.installation.php#faq.installation.addtopath>)
* On Windows 7, XP, Vista, 2008, 2012 and up:
  + Go to Control Panel and open the System icon (Start → Control Panel)
  + Go to the Advanced tab
  + Click on the 'Environment Variables' button
  + Look into the 'System Variables' pane
  + Find the Path entry (you may need to scroll to find it)
  + Double click on the Path entry
  + Enter your PHP directory at the end, including ';' before (e.g. ;C:\php)
  + Press OK

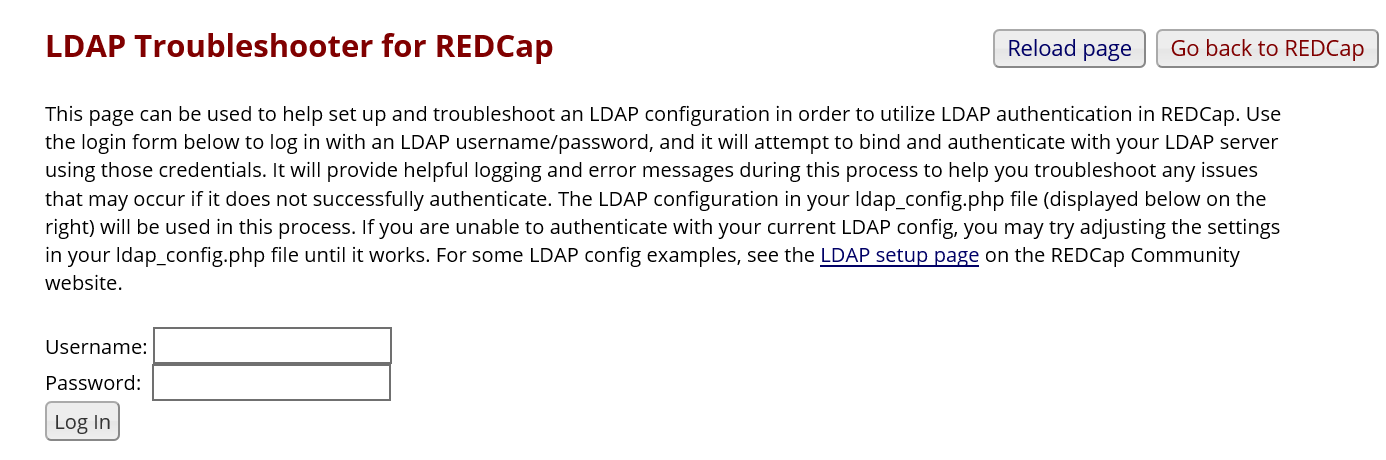
At this point, the libeay32.dll and ssleay32.dll files have been referenced to PHP. In contrast with other systems, On Windows Server2012 the machine doesn’t need to be restarted for these changes to take place. The next step is to enable the LDAP extension in PHP. As before, in the php.ini file, enable the LDAP extension as shown by the following figure:



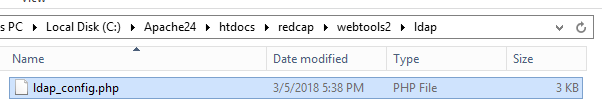
Similarly, the LDAP module needs to be enabled in Apache. The following figure shows how this looks:



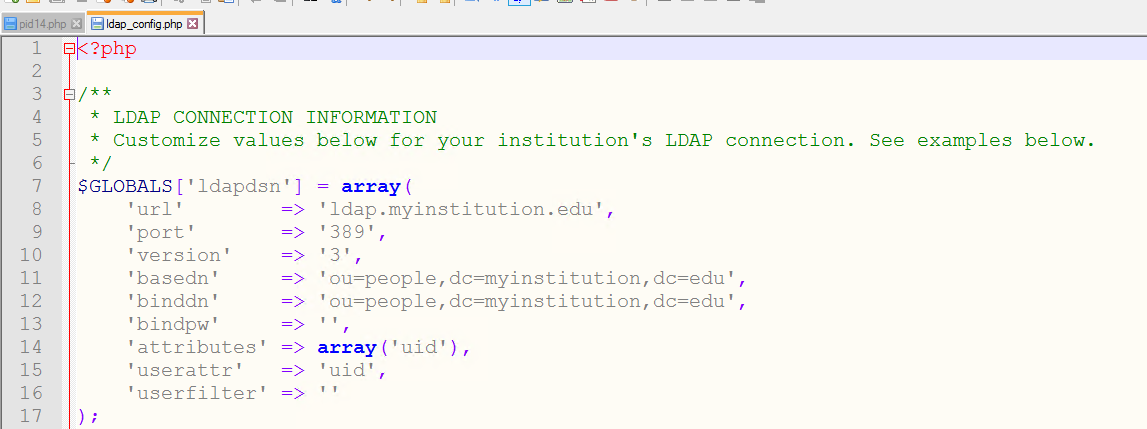
Once these changes have been made, Apache needs to be restarted using its Apache Monitor. The best way to check whether the configuration changes were successful is to go back to REDCap’s LDAP troubleshooter-link, as shown above, and see if the initial error message has been resolved. If successful, here’s how it looks after the PHP and Apache configurations for LDAP have been completed.



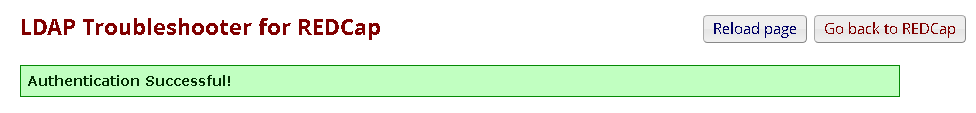
The next step is to set the LDAP credentials within REDCap. These credentials are specified by the LDAP manager at the host institution. Once these credentials have been obtained, they need to be configured in REDCap’s ldap\_config.php file found under REDCap’s webtools2 folder. Configuring this file is a similar process to setting the database connection to MySQL. Here’s a screenshot of the file location:



The ldap\_config.php file contains sample LDAP configurations as guidelines for the REDCap administrator. It is this file that needs to be adjusted with the institutions LDAP credentials. Here’s how the sample configuration looks like:



Once the appropriate credentials have been configured, REDCap’s LDAP installation can be tested by login into REDCap using LDAP credentials. If successful, the user will be able to enter REDCap, and see the corresponding REDCap projects, using their institutional password. It can also be tested by going to REDCap’s LDAP troubleshooting link and enter a user’s LDAP credentials. If successful, REDCap will display the following message:



# 

# Other useful references:

* Configuring LDAP:
  + <https://community.projectredcap.org/questions/2925/ldap-configuration-setup-details.html>
  + <https://community.projectredcap.org/questions/10864/where-is-the-ldap-troubleshooter-located.html?childToView=10866#answer-10866>
  + <http://php.net/manual/en/ldap.installation.php>
  + <http://php.net/manual/en/faq.installation.php#faq.installation.addtopath>
* Getting back into a locked REDCap:
  + <https://community.projectredcap.org/questions/21786/new-to-redcap-locked-self-site-admin-out-of-projec.html>
  + <https://community.projectredcap.org/articles/1103/reset-superuser-password-with-db-access.html>
  + <https://community.projectredcap.org/questions/15261/locked-out-of-system.html>
  + <https://community.projectredcap.org/questions/26465/cannot-access-controlcenter-after-enabling-authent.html>
* Setting REDCap’s Authentication method:
  + <https://community.projectredcap.org/articles/691/authentication-how-to-change-and-set-up-authentica.html>
  + <https://community.projectredcap.org/questions/4345/ldap-authetication-table-based-authentication.html>
* Changing the stats-date-format:
  + <https://community.projectredcap.org/questions/38655/how-do-you-change-the-date-format-on-redcap-stats.html?childToView=39413#comment-39413>

# Version Log in descending order

[20180620] Fixing Typos – [Eduardo Morales](https://www.linkedin.com/in/eduardomorales/)

[20180417] Document editing – [Kassandra Hopkins](https://www.linkedin.com/in/kassandra-hopkins-925581ba/)

[20180411] Document editing, formatting and index by [Kassandra Hopkins](https://www.linkedin.com/in/kassandra-hopkins-925581ba/)

[20180408] Installing and configuring LDAP – [Eduardo Morales](https://www.linkedin.com/in/eduardomorales/)

[20180325] Adding useful references – [Eduardo Morales](https://www.linkedin.com/in/eduardomorales/)

[20180306] Installation steps outlined by [Eduardo Morales](https://www.linkedin.com/in/eduardomorales/); professional IT support by [Tom Kerrigan](https://www.linkedin.com/in/tom-kerrigan-6a161b5/).