ILLiad Web Platform Troubleshooting

OCLC Customer Services Division

Abstract

ILLiad Web Platform: Basic Functionality, Dependencies, and General Troubleshooting

Date

2020-06-30

ILLiad Web Platform:

Basic Functionality, Dependencies, and General Troubleshooting

# What is it and what does it do?

* An API (Application Programming Interface) that provides programmatic access to your ILLiad installation in a controlled way.

<https://support.atlas-sys.com/hc/en-us/articles/360011809394-The-ILLiad-Web-Platform-API>

* A Web Service that use that API to communicate with your ILLiad system from other outside systems. At present, this is primarily ILLiad/Aeon interactivity, but other connections will be possible eventually as we release new versions with enhanced functionality.

## Current (9.1) Uses:

* ILLiad/Aeon interoperability:
	+ The integration of ILLiad and Aeon via the ILLiad and Aeon Web Platform services makes it possible for ILLiad staff to place Aeon requests directly from the ILLiad client and for Aeon staff to respond to the requests directly from the Aeon client.
	+ [https://prometheus.atlas-sys.com/display/illiad/Configuring+Web+Platform+Integration+for+ILLiad+and+Aeon](https://prometheus.atlas-sys.com/display/illiad/Configuring%2BWeb%2BPlatform%2BIntegration%2Bfor%2BILLiad%2Band%2BAeon)
* ILLiad/OCLC Article Exchange WebService interactions:
	+ Uploads scanned documents to the OCLC Article Exchange Web Service from the ILLiad Client.
	+ Clients upload to the ILLiad Web Platform which, in turn, uploads to the OCLC AE web service.
* IFM WebService interactions:
	+ Addons (like Get It Now) talk to the ILLiad Web Platform via the client scripting commands.
	+ The Web Platform, in turn, interacts with the OCLC IFM Web Service.

# How is it installed?

* On new ILLiad systems, the Web Platform is installed automatically on the ILLiad web server as a part of the server installation. For older systems, it’s added and installed as a part of the ILLiad version 9.0 server update process.
* In IIS, the Web Platform is typically installed directly beneath the Default Web Site, at the same level as the ILLiad folder itself. (See image below.)



* The files necessary for the ILLiad Web Platform are installed in the **X:\illiad\WebPlatform\** folder (where X is the drive letter where ILLiad is being installed).

# Dependencies:

* Microsoft .NET 4.0 Framework installed on your web server, and properly set up to use the .NET 4.0 Framework:
	+ Verify that there is an extension for ASP.NET v4.0.30319 and that it is set to Allowed.
		- This is done in IIS under ISAPI and CGI Restrictions
	+ Verify that the ASP.NET version is set to v4.0.30319 for each module using ASP.NET.
		- This is done in IIS under Application Pools
* An SSL Certificate. The ILLiad Web Platform is only accessible over port 443 via an https:// URL. Port 80 will not work.
* ILLiad requires you to set up an API Key in the WebPlatformConfig Table. This is accessible from the Customization Manager under System – Integration.
* Time of web and data servers must MATCH exactly!



Note: In a Shared Server/Multi-site ILLiad system, ONLY the ILL base site needs an entry in the WebPlatFormConfig table for ILLiad. If you’re also pointing the system at Aeon, for ILLiad/Aeon interoperability, this is done on a per site basis. (See the above image.)

* In addition, once the ILLiadWebPlatformConfig table is set up, use the ILLiadWebPlatformConfig customization key (System – Web Platform) to specify which record in the table each ILLiad site should be using for itself.

Note: In a Shared Server/Multi-site ILLiad system, ALL individual sites on the system need their own separate site-specific entries in the ILLiadWebPlatformConfig customization key.

## Permissions:

* The minimum permissions required for ASP.NET applications to run require read permissions on all web files for
	+ The Internet Information Services (IIS) IUSR\_
	+ The ComputerName Internet Guest Account
	+ An ASP.NET ASPNET user account (or NetworkService user account, for IIS 6.0).
	+ Everyone
* The accounts will also need read permissions to the logon.dbc file that was created using the SQL Alias Manager (SAM).

## Relevant Ports:

* **Port 443**: Web function.
* **Port 1433**: Connectivity to Database via SQL Alias Manager (SAM)/logon.dbc settings.

# Testing the Web Platform:

* In a web browser, enter a URL to go directly to the ILLiad Web Platform.
	+ If the Server ILLiad URL is:
		- <https://jkcf-alpha/illiad/>
	+ Then the ILLiad Web Platform URL is:
		- <https://kcf-alpha/ILLiadWebPlatform/>
* If you get a response that specifies the Web Platform version (as opposed to an error), the Web Platform is installed and configured correctly. (See image below.)



# Web Platform Logging:

Like most other ILLiad services, there is a means to configure logging for the ILLiad Web Platform. The ILLiadWebPlatform.log.config file for this is located in the **X:\illiad\WebPlatform\App\_Data\ILLiadWebPlatform.log.config** folder (where X is the drive letter where ILLiad is installed).

The default ILLiadWebPlatform.log.config file looks like this:

**<?xml version="1.0" encoding="utf-8"?>**

**<log4net>**

 **<root>**

 **<level value="INFO" />**

 **<appender-ref ref="RollingFile" />**

 **</root>**

 **<appender name="RollingFile" type="log4net.Appender.RollingFileAppender">**

 **<file type="log4net.Util.PatternString" value="C:\ILLiad\Logs\ILLiadWebPlatform.log" />**

 **<appendToFile value="true" />**

 **<maximumFileSize value="1MB" />**

 **<maxSizeRollBackups value="5" />**

 **<rollingStyle value="Once" />**

 **<staticLogFileName value="true" />**

 **<layout type="log4net.Layout.PatternLayout">**

 **<conversionPattern value="%date [%thread] %-5level %logger - %message%newline%exception" />**

 **</layout>**

 **</appender>**

**</log4net>**

The log file created by this configuration defaults to being created in **C:\illiad\Logs\ILLiadWebPlatform.log**, in the same location as the logs for several other ILLiad services.

By default, Web Platform Logging defaults to **INFO** mode. If more verbose logging is desired, you can change the mode to **DEBUG** in the ILLiadWebPlatform.log.config file and save the config file to apply the change.

Note: We recommend changing the .config file back to INFO mode once any troubleshooting is completed so as to not take up too much space with the logs.

# Troubleshooting Tips:

* Be sure to look at both the client logs and the web platform logs when troubleshooting. Any issues could be part of a communication process between the ILLiad client and the ILLiad web platform, or between the ILLiad web platform and something else (like another OCLC service).
	+ Troubleshooting would typically require both client and web platform logs, as the helpful part is often knowing if the client ever made it to the web platform.
		- Using timestamps, you should be able to find corresponding entries when the client sends something, and the web platform begins to work on it.
		- If it never appears in the web platform log, there may be networking issues or some other problem rendering the web platform unable to process the request.

Important: Going forward, the ILLiad web platform is going to become the middleman between the ILLiad client and database for many new features. This type of troubleshooting is going to be very important.

# Common Web Platform Issues:

## Article Exchange-related Issues

* "An error occurred while attempting to perform delivery. The remote certificate is invalid according to the validation procedure." Process fails and request is left in Awaiting Lending Scanning.:

You need an SSL Certificate, or it is invalid.

* Internal Server Error in client during Article Exchange Sending
	+ fails due to OCLC Authorization and/or Password values specified in the Customization Manager having trailing spaces in the specified values.