

## Polarion® ALM™ Installation Guide for Windows

March 1<sup>st</sup>, 2017

ver.17

### Table of Contents

[Table of Contents](#)

[About This Guide](#)

[Evaluation Installations](#)

[Large-scale Installations](#)

[Document Conventions](#)

#### [1. System Requirements and Recommendations](#)

[Server Software](#)

[Server Hardware](#)

[Client Software](#)

[Client Hardware](#)

[Additional Recommendations](#)

[Libraries Required for Building the Demo Projects](#)

[Enable Email Notifications](#)

[Enable Support for Javadoc](#)

#### [2. Installation and Start-up](#)

[The Windows Installer](#)

[What the Installer Can Install](#)

[Using the Installer Program](#)

[One-click and Custom Installation Options](#)

[Polarion Components Installed](#)

[Installation on Windows 8.1 or 10](#)

## [Starting and Stopping the Polarion Server](#)

### [Starting the Server](#)

#### [The Express Startup Shortcut](#)

### [Starting Polarion Server as a Service](#)

#### [Notes about Running as a Service](#)

### [Starting Polarion as a Console Application](#)

### [Shutting Down the Polarion Server](#)

## [3. After Installation](#)

### [Securing the Polarion Activation Application](#)

### [Adjusting Server Memory Allocation](#)

### [Java Virtual Machine Memory Limit](#)

### [Configuring PostgreSQL Database](#)

### [Optimizing the PostgreSQL Database](#)

#### [Allowing Remote Connections to the PostgreSQL Server \(optional\)](#)

### [Enter an Error Reporting Email](#)

## [Licensing and Activation](#)

### [Using Different License Types](#)

### [Assigning Named and Concurrent Users](#)

### [License Usage Log File](#)

## [Changing Default System Passwords](#)

### [Changing the Default System Administrator Password](#)

### [Changing the Password for SVN User 'polarion'](#)

## [Configuring OLE Object Support](#)

### [Configuring Polarion for OLE](#)

## [Multiple Repository Setup](#)

## [Subversion Optimization](#)

## [Accessing the Portal](#)

## [Configuring Single Sign-on \(SSO\)](#)

### [Kerberos SSO](#)

#### [Prerequisites - Configuring the Domain](#)

#### [Configuring Polarion](#)

#### [Configuring Client Browsers](#)

##### [Internet Explorer 11, Chrome, Edge](#)

##### [Firefox](#)

### [Teamcenter SSO](#)

[Prerequisites - TcSS Configuration](#)

[Polarion Configuration](#)

[Other SSO Considerations](#)

[Subversion Access](#)

[Direct Access to Subversion](#)

[Features Requiring Special Configuration in an SSO Setup](#)

[Troubleshooting](#)

[SSL](#)

[File Encoding](#)

[Kerberos Hints](#)

[Wrong Token Provided by the Browser](#)

[Polarion Unable to Log In to KDC](#)

[LDAP Authorization](#)

[Next Steps after Installation](#)

[Removing Polarion](#)

[Manually Updating Third-party Software](#)

[1. Update Subversion](#)

[2. Update Apache](#)

[Updating Java](#)

[4. Technical Support](#)

[APPENDIX](#)

[Enabling Email Notifications](#)

[Default Parameters and Settings](#)

[Supported Microsoft Office® Versions](#)

## About This Guide

Welcome and thanks for using Polarion.

This guide covers installation information and procedures for creating a **production installation** for all Polarion products based on the Polarion Application Lifecycle Management Platform. The list of products covered by this guide currently includes:

- **Polarion® ALM™**
- **Polarion® REQUIREMENTS™**
- **Polarion® QA™**

In general, the information is applicable to all the above products. Any product-specific differences will be explicitly noted. The information covers both new installations and, where applicable, updating of existing installations.

This guide applies to installation of the above Polarion products on supported Microsoft Windows® operating systems.

If you want to install a product on a Linux system, please see the separate *Installation Guide for Linux* document provided in the download archives for our Linux product versions, and also available on the Polarion Software web site.

## Evaluation Installations

If you are installing Polarion for evaluation purposes, we recommend the **Polarion® ALM™ Trial Guide** (PDF), available for download on any product download page at <https://polarion.plm.automation.siemens.com/>. This guide focuses on getting you up and running with an evaluation installation as quickly as possible, using the 1-click install option of the Windows installer. (This is recommended for the initial stage of any evaluation.)

## Large-scale Installations

If you need a large-scale server environment with multiple clustered servers and failover capabilities, multiple repositories, etc. you should get the **Polarion® ALM™ Enterprise Setup Guide** (PDF), also available on product download pages on the Polarion web site.

## Document Conventions

- Square brackets [value] are placeholders for some actual value you must supply.
- [POLARION HOME] refers to the folder where Polarion is installed.
- System paths use `monospace font`.
- Names of properties, functions, and other system items use *italic font*.
- User input or actions used **bold font**. If user input is code snippet or path, **bold monospace** may be used

# 1. System Requirements and Recommendations

## Server Software

Requirement	Description
<b>Operating System</b>	<p>Microsoft Windows Server 2012 (or any update release), OR Microsoft Windows Server 2016 (or any update release). For evaluation purposes you can use Microsoft Windows 8.1 or 10. See <a href="#">“Installation on Windows 8.1 or 10”</a>.</p> <p>Only 64-bit versions are supported.</p>
<b>Java Runtime Environment</b>	<p>Java Platform, Standard Edition Development Kit 8 - Oracle JDK 8.. (<a href="http://www.oracle.com/technetwork/java/javase/downloads/index.html">http://www.oracle.com/technetwork/java/javase/downloads/index.html</a>)</p> <p>Note that OpenJDK 8 is not the same as Oracle Java 8 and is not officially supported. <code>JAVA_HOME/bin</code> should exist in the system paths.</p>
<b>Version control system</b>	<p>Subversion version 1.6.x, 1.7.x, 1.8.x, or 1.9.x: <a href="http://subversion.apache.org/">http://subversion.apache.org/</a></p> <p>If you are compiling Subversion yourself, compile using the <code>--with-apsx</code> or the <code>--with-httpd</code> option.</p>
<b>Web Server</b>	<p>Apache HTTPD server with <code>mod_proxy_ajp</code> and Subversion extension (WebDAV+SVN apache modules): <a href="http://httpd.apache.org/">http://httpd.apache.org/</a></p> <p>In general, the Polarion server should run with whatever Apache version is present on a Windows system provided it is at least the minimum required version (2.2), and <code>mod_proxy_ajp</code> and Subversion extension modules are also installed.</p>
<b>Database</b>	<p><b>Minimum:</b> PostgreSQL version 8.4 <b>Recommended:</b> PostgreSQL version 9.2 or higher</p>

## Server Hardware

Requirement	Description
RAM	<ul style="list-style-type: none"><li>• <b>Minimum:</b> 4 gigabytes for production installation.</li><li>• <b>Recommended:</b> 8 gigabytes or more</li></ul>
Disk Storage Space	<ul style="list-style-type: none"><li>• <b>Minimum:</b> 10 GB (gigabytes)</li><li>• <b>Recommended:</b> 40 GB (gigabytes) or more</li></ul> <p>There is no hard and fast rule for disk storage space. The actual amount you will require depends on the number and size of projects managed with Polarion. The more projects, and the larger they are, the more disk storage you will require.</p>

## Client Software

Requirement	Description
Operating System	<p>Any operating system that can run the supported web browsers with support for the Flash plugin (see below).</p> <p>If the client user will use a Polarion product supporting data interchange with Microsoft Office®, the client user must run a Windows operating system compatible with a supported version of the Microsoft Office application(s) used. For details, please see <b>Appendix: Supported Microsoft Office Versions</b>.</p>
Web Browser	<p>All you need to use Polarion is a web browser. The most current list of supported browsers and versions is provided in the Release Notes section of the <b>README.html</b> file, delivered in all download distributions of all Polarion products.</p>
Adobe Flash	<p>The Polarion web portal displays charts and other graphical data about Polarion-managed projects using Adobe Flash.</p> <p>To view these properly using a web browser, the client computer must have Adobe Flash Player installed. You can download it free at <a href="http://www.adobe.com/products/flashplayer/">http://www.adobe.com/products/flashplayer/</a>.</p>

## Client Hardware

Requirement	Description
RAM	Minimum: 2 GB (4 GB recommended)
Display Resolution	Minimum: 1280 x 800 pixels
Server Connection	Not less than 1 Mbit/S

## Additional Recommendations

The items described in this section are not critical for running and evaluating Polarion. However, the components described are needed to be able to fully utilize the product's capabilities.

### Libraries Required for Building the Demo Projects

The distribution contains several demo projects. Each of them needs its particular set of 3rd party libraries to be correctly built and have the project reports generated. Any missing libraries are automatically downloaded from the Internet during project processing, so you may need a connection to the internet when you first try building the demo projects or running reports for them.

### Enable Email Notifications

The Polarion server can send email notifications in response to various events in the system such as build completions, new work items, etc. It can also notify users about external changes.

To enable Polarion to send out email notifications you need to:

1. Configure the SMTP server to be used by the Polarion server for sending emails. You can do this from the Windows installer during the regular installation process (i.e. not One-click Installation), or after installation, but before starting Polarion server (see Appendix). You may wish to create a special account on your SMTP host for use with Polarion notifications.
2. Provide a valid email address for each user in their user account. (Administration > User Management > Users.) This can be automated through user self-creation of accounts, or integration with LDAP. See Help, Administrator's Guide: *Managing Users and Permissions*.

When this configuration is correctly set up, the system will send notification e-mails about various events according to the notification targets configuration. For information on configuring email notifications, see Help, Administrator's Guide: *Configuring Notifications*.

**NOTE**

*If a Work Item is modified outside of the Polarion portal, (e.g. manually in SVN), email notifications are sent the same as they would be if the modification has been done in the portal.*

**See also:** Appendix: [Enabling Email Notifications](#)  
**Enable Support for Javadoc**

The demo projects (and your projects) can be configured to provide Javadoc reports. However the *javadoc* executable is not included in the Polarion distribution due to licensing restrictions. To be able to run Javadoc reports you need to do the following:

1. Download the Java JDK (see link in *Server Software* earlier in this document) and install it somewhere, or access an existing installation of the JDK.
2. Copy **javadoc.exe** from the **bin** folder of the JDK to **\$POLARION\_HOME\$\bundled\java\bin**.

Javadoc must also be enabled for the **descriptors.xml** file. Access it in the Repository Browser: `Repository/.polarion/reports/descriptors.xml` . Refer to comments in the file for enabling Javadoc.



## 2. Installation and Start-up

This chapter covers installation and startup procedures for Polarion on Windows. There is also some information about advanced manual setup on Windows.

### The Windows Installer

The installer for Windows® systems is a 64-bit Windows executable (.exe) file. The distribution file name contains the product name, version number, service release ID (if applicable), the processor specification and .exe. For example:

PolarionALM\_ *NNNN* \_SRN\_x64.exe

Windows 64-bit distribution for Polarion.

... where *NNNN* is a Polarion version number (typically the year of the release) and *SRN* is a Service Release number (which does not appear if the distribution is not for a Service Release).

The distribution archive contains a setup wizard program, and everything you need to get Polarion up and running on Windows is bundled, including Apache server and Subversion. You can run the installer program from any convenient folder on the computer where Polarion will be installed.

### What the Installer Can Install

The Windows installer can be used for a **new (clean) installation only**. Installing an update for an existing installation requires a different distribution archive. The installer program will advise you of this if it detects an existing installation.

### Using the Installer Program

The installer program for Windows is a so-called "wizard" which leads you step-by-step through the installation process and several initial configuration steps.

#### IMPORTANT NOTE FOR FIREWALL USERS

*If the computer on which you are installing Polarion is running a firewall, you should disable that for the duration of the installation to enable the installer to check for the availability of needed ports. Alternatively, you may create some rules in your firewall to permit the Polarion installation executable to access ports.*

*The wizard is designed to be self-explanatory. However, there are a few issues in the installation and initial configuration that you might want to be aware of before running the installer program.*

## One-click and Custom Installation Options

The wizard offers you the following installation options:

- **One-click: Intended for evaluation installation only.** All Polarion components will be installed (see Components table below), and default and simple values will be used for ports, paths, etc.
- **Custom: Intended for production installation.** Enables you to specify used ports, host names, and SMTP parameters for email notifications, whether or not to install Java runtime, demo projects and demo project data, and whether or not to create a program group and desktop icon for it.

## Polarion Components Installed

Component	Description
<b>Polarion binaries</b>	Files needed for Polarion to run.
<b>Demo projects</b>	Repository folders containing sample projects and data. Installation of this component is recommended if you are evaluating Polarion. You may choose to omit it for production installations.
<b>Apache HTTP Server</b>	<b>IMPORTANT!</b> You should shut down any other Apache server you may be running on the computer where you are installing Polarion.
<b>Subversion</b>	The latest supported version at time of the Polarion release.
<b>Java SE JRE</b>	<p>The Java runtime environment is needed by Polarion for a number of different functions, including running Polarion itself. If you already have an installation of the same JRE version, you may opt not to install it. Polarion will prompt you to specify the location of your existing Java JRE installation.</p> <p>For evaluation purposes, it will be simplest to go ahead and install the JRE. You can uninstall Polarion after your evaluation period and the JRE will be removed as well.</p>
<b>PostgreSQL</b>	SQL database used to process complex queries that retrieve data for reports, dashboards, etc.
<b>Polarion program group in the Start menu</b>	A program group for Polarion in the start menu. (Optional.)
<b>Polarion group icon on desktop</b>	Desktop shortcut that opens the installed <code>polarion shortcuts</code> folder which contains shortcuts for starting and stopping Apache service and Polarion server. (Optional.)

For information about what versions of bundled software are distributed with Polarion, please see [Manually Updating Third-party Software](#) later in this document.

### **Installation on Windows 8.1 or 10**

For evaluation purposes you can install Polarion on computers running Microsoft Windows 8.1 or Microsoft Windows 10 (64-bit only).

**For production use it is important that you use a supported *server* operating system.**

Running Windows 8.1 in the desktop mode is required. Polarion's Flash-based features (LivePlan, etc.) will not work, and various dialogs as well as Clipboard access are blocked if the Windows 8 style user interface (sometimes called "Metro") is used.

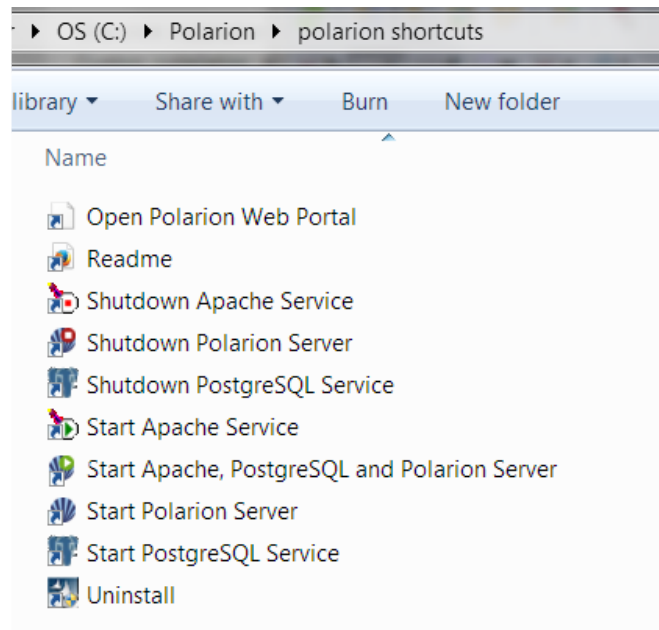
Please note that it is not recommended to install Polarion to `C:\Program Files` or `C:\Program Files (x86)`. These folders have special policies and some components would not have the necessary read or write permissions by default, so you would need to tweak configuration settings – needless for evaluation.

As mentioned in the introduction, we recommend that evaluators obtain the *Polarion ALM Trial Guide* (PDF), available for download on any product download page at <https://polarion.plm.automation.siemens.com>.

## Starting and Stopping the Polarion Server

The installer provides the option of starting the Polarion server for the first time after installation. Later on, you will need to know how to stop and restart the Polarion server manually.

The Windows installer creates and automatically configures a set of Shortcuts that enable you to start and stop the Apache service, start and stop the Polarion server, start and stop the integrated PostgreSQL database, and launch the web portal in the default browser. These are located in the folder `[POLARION_HOME]\polarion shortcuts`. The installer places an icon on the Desktop that opens this folder in Windows Explorer.



*Shortcuts for starting and stopping Apache and the Polarion server*

### Starting the Server

When starting the server, run the startup in the following sequence:

1. Start up Apache Service (run the Start Apache Service shortcut)
2. Start PostgreSQL (run the Start PostgreSQL shortcut)
3. Start up Polarion Server (run the Start Polarion Server shortcut)

Note that it is not necessary to start Apache Service again if it is already running. You can check if it is running using the Apache Monitor program

`($POLARION_HOME$\bundled\apache\bin\ApacheMonitor.exe)`.

The first time you start the Polarion server, you will see a message in the console:

```
Not enough data for startup estimation.
```

This is not an error message. On startup and on re-index operations, Polarion estimates and reports the amount of time the operation will take. You will see this estimate in the console and log file on subsequent startups, but the first time, no data exists on which to base the estimate.

The following startup phases are reported in the console and log file:

- Platform startup
- Context recognition
- Context initialization
- Revisions processing
- Build artifacts recognition
- BIR inspection
- Data indexing
- Polarion startup

### The Express Startup Shortcut

An "express" shortcut, *Start Apache, PostgreSQL and Polarion Server*, is provided that will handle all the startup operations in proper sequence. This shortcut should only be used to start Polarion when none of these services are running.

### Starting Polarion Server as a Service

By default, Polarion Server is installed as a Windows service. The service name is *Polarion*. The service description is *Polarion Server*.

When Polarion is installed as a service, the shortcuts previously described are configured to run the necessary executables with the required parameters so that Polarion is started as a service. The shortcuts are the recommended way to start the Polarion Server. If you want to start the server from the command line, you can invoke the following after making sure the Apache service is running:

```
C:\WINDOWS\system32\net.exe start Polarion
```

### Notes about Running as a Service

When the Polarion service is started, parameters from the `polarion.ini` file (memory settings etc.) are not taken into account. So if you need to change `polarion.ini` and you run Polarion as a service, you must reinstall the Polarion service after changes in `polarion.ini` are completed.

To do this, use the command line to run...

```
C:\Polarion\polarion\service.bat -uninstall
```

...and then run...

```
C:\Polarion\polarion\service.bat -install
```

## Starting Polarion as a Console Application

Polarion can be run as a console application by running...

**C:\Polarion\polarion\polarion.exe.**

However, this file is provided mainly for backward compatibility. If you installed Polarion as a service, you should run it as such, as previously described.

## Shutting Down the Polarion Server

When shutting down the server, use the following sequence:

1. Shut down Polarion server (run the *Shutdown Polarion Server* shortcut)
2. Shut down PostgreSQL database (*Shutdown PostgreSQL Service* shortcut)
3. Shut down Apache Service (run the *Shutdown Apache Service* shortcut)

Note that it is not normally necessary to shut down Apache Service if you just want to restart the Polarion Server.

### NOTE

*The Appendix provides reference information about default Polarion parameters used by the Polarion installer for Windows.*

### 3. After Installation

This chapter covers some first things to do immediately following installation and initial configuration (if applicable) of Polarion.

#### Securing the Polarion Activation Application

Beginning with version 2015, Polarion includes an activation application that makes it possible to install or update a license during runtime of the Polarion server, without need to copy the license file manually to the target machine. **Access to this application is NOT initially protected by username and password.** For production use it is highly recommended to secure access to this application directly in the Apache configuration.

Beginning with version 2015, there is a template Apache configuration file in the Polarion installation folder: `\polarion\polarion\install\polarion.activation.conf.template`

To ensure that a user name and password is requested when accessing the activation application (`\polarion\activate\online` and `\polarion\activate\offline`), copy this file to the Apache configuration folder, on Windows usually

`C:\Polarion\bundled\apache\conf\extra\` .

After copying the file, rename it to remove the extension ".template". Then open the file in any text editor and modify it according to the instruction comments provided.

The template configuration is prepared for both *users* file authentication (like Polarion uses for Subversion by default, with user passwords data in a file) and for authentication against an LDAP server.

In a multi-instance setup (coordinator + one or more instances, that can be clustered), it is necessary to use this configuration only on the coordinator server (the activation application runs only on the coordinator). For additional information about this type of setup, see the *Polarion ALM Enterprise Setup Guide*.

## Adjusting Server Memory Allocation

The default installation on Windows uses a maximum memory allocation setting in `C:\Polarion\polarion\polarion.ini` of `-Xmx640m` (640 megabytes ) which is suitable memory allocation for evaluation purposes but not for most production setups. Before beginning production use of a new installation, the server administrator should adjust the setting to use the capacity of the server to its full extent. Use MEGABYTES as the measure. E.g. for 2 gigabytes, specify `-Xmx2000m`.

### IMPORTANT

*You MUST reinstall the Windows Service after modifying the `polarion.ini` file. Make sure the Polarion server is not running, then run the utility:*

```
C:\Polarion\polarion\service.bat -install
```

## Java Virtual Machine Memory Limit

If you allocate too much memory for the Java Virtual Machine (JVM), the operating system will not initialize it. Diagnosing the issue can be difficult because the service simply does not start and no error log is written. An entry *is* written to the Windows *Application* log with the description “Could not initialize the Java Virtual Machine” but the reason is not specified.

The amount of memory you can allocate to the JVM depends on how much total memory exists on the computer. The more total memory the more you can allocate to the JVM before the operating system imposes a limit. In tests on a 4 GB system, allocating 1.1 GB to the JVM resulted in failure to initialize.

## Configuring PostgreSQL Database

This component must be installed and properly configured before you can start the Polarion server. The Windows installer attempts to install the software, configure it to run with Polarion, including some optimization for best performance. If for any reason the installer is unable to optimize PostgreSQL you should do it manually.



## Optimizing the PostgreSQL Database

Beginning with version 2015 SR2, Polarion integrates the PostgreSQL database in all new installations. After a new Polarion installation containing this database, it is highly recommended that the administrator review, and if necessary adjust some PostgreSQL settings to optimize performance.

After installation you should make the following changes in `postgresql.conf`.

Default path: `C:\Polarion\data\postgres-data\postgresql.conf`

```
max_connections = 80    # should be < 10 * number of CPUs
shared_buffers = 2GB   # should be 10% - 15% of total system RAM
work_mem = 10MB        # should be 10MB - 100MB
maintenance_work_mem = 200MB

fsync = off
synchronous_commit = off
full_page_writes = off

wal_buffers = 256kB    # should be more than size of common
                        # transaction

checkpoint_segments = 32
effective_cache_size = 4GB      # should be approx 2/3 of total
                                # system RAM
max_locks_per_transaction = 100 # specific for Polarion
# Optimal planner performance setting
# For HDD, keep default setting. Otherwise, uncomment the
# applicable setting below:
# For SSD:
# random_page_cost = 1.5
# For SAN:
# random_page_cost = 2.0
```

### Allowing Remote Connections to the PostgreSQL Server (optional)

By default the PostgreSQL database is only available on *localhost* after a fresh install. If you need to have access from outside the host machine, please follow these steps to allow remote access:

1. Change the listener addresses in the `postgresql.conf` file (see path above). Change:  
`#listen_addresses='localhost'` to `listen_addresses='*'`
2. In the same folder, you will find the `pg_hba.conf` file. Open it in a text editor and add the following line: `host all all 0.0.0.0/0 md5`

#### SECURITY ADVISORY

*If you allow external connection to the database, the external user has blanket access to the entire database content and effectively to all Polarion data - all projects, the HEAD revision as well as history. There is no further access control. All Polarion user names will also be visible. (user passwords will not be visible as those are not stored in a database or SVN.) Access is read-only.*

*Administrators are advised to consider carefully before opening database access in a system in use for production.*

### Enter an Error Reporting Email

The email for the `"error.report.email"` property in the `polarion.properties` file is empty by default for new installations. Add the email you want error reports sent to.

## Licensing and Activation

In order to use Polarion you must obtain a license. A license with the necessary key and file is normally delivered by email to the address provided by the person who purchased the license. If you need help obtaining a license please contact [sales@polarion.com](mailto:sales@polarion.com).

Polarion installs with a 30-day evaluation license. After obtaining a license for production use, you must activate your Polarion installation. The login page provides action options leading to online and offline activation instructions. You will need the information provided by Polarion to complete the activation.

### Using Different License Types

Several different license types are available - Evaluation, Site, User-limited, etc. If you begin using Polarion with one type of license key (Evaluation, for example), and want to continue using it with a different license type, simply remove the current license key file from the license folder and copy the new license key file there. If Polarion server is running, you will need to restart it for the new license to take effect. You may install multiple license keys for different license types and/or Polarion products on the same server.

### Assigning Named and Concurrent Users

If your license provides for named and/or concurrent users, you will need to add assignments for each type of user in the appropriate section of the **users** file. By default this file is located in the license folder of your Polarion installation. If you change the location for license key files, be sure to move the users file to the same folder that stores your license key file.

You can edit it in the License topic in the global Administration in the portal. The file contains comments with complete instructions on how to find the user IDs of your named/concurrent users, and make the relevant assignments.

### License Usage Log File

Administrators and managers can monitor license usage by checking the license usage log file `log4j-licensing-TIMESTAMP.log`. This file is located in the folder `[POLARION_HOME]\data\workspace\.metadata`.

When a concurrent user logs in/out, a license usage statistics report is written to the licensing log. (Note that concurrent licensing is not supported for all products.)

The following example shows one user currently using an *enterprise concurrent* license type, the greatest number of users of this license during the current server session (*peak*), and the maximum number of users allowed by the license (*limit*).

```
2008-05-14 11:12:29,609 [TP-Processor2] INFO PolarionLicensing -  
STATS:enterpriseConcurrentUsers,current:1,peak:2,limit:20
```



## Changing Default System Passwords

To help ensure the security of your Polarion system, you should change the default password of the system administrator user account, and *polarion* user in Subversion. The following two sections explain how to make these changes.

### Changing the Default System Administrator Password

The default account *System Administrator* has access to all administrative functions of Polarion, including read-write access to the Subversion repository. After installing Polarion for actual production use should change the password on the default system administrator account.

Before you change the password on the default system administrator account, you may wish to create another account with administrator rights for yourself and/or someone else.

*To change the default administrator password:*

1. Log in to the Polarion portal with the default login described above.
2. Click on *My Polarion*. The My Polarion page for the System Administrator account loads in the content area.
3. In the System Administrator's *My Polarion* page, click the link: *My Account*.
4. In the My Account page click the *Edit* button.
5. Enter the new password in the *New Password* field, and again in the *Re-enter Password* field.
6. If you want to continue using this account as the main system administrator account, you may wish to add your email address in the *Email* field, and add a description for the account in the *Description* field.
7. When finished editing the System Administrator profile, click the *Save* button. The password is now changed and you will need to use it next time you log in.

#### **DON'T LOSE THE NEW PASSWORD!**

**If you lose the changed password, you will not be able to log in as system administrator. If no other accounts exist with administrator rights, it will not be possible to change the configuration, add projects, manage user accounts, etc.**

### Changing the Password for SVN User 'polarion'

A Subversion repository user named *polarion* is created by default when you install Polarion. This user acts on behalf of the Polarion application and consequently has extensive permissions including read permission for all projects. Access to this user by unauthorized people would compromise the security of your Polarion system, so it is advisable to change this password before putting the system into production use.

The following steps assume you use *passwd* file authentication, which is the most common method.

1. Stop the Polarion server before changing this password.
2. Use the **htpasswd.exe** to change the password for the user *polarion*. The utility is located in the folder **[POLARION HOME]/bundled/apache/bin**.

Utility syntax: **htpasswd path/passwdfilename username**

Example: **htpasswd Polarion\data\svn\passwd polarion**

3. Next, change the value of the *password* property in the **polarion.properties** file to the password you set with the **htpasswd** utility. Typical location of this file is **polarion/configuration/polarion.properties**.

#### NOTE FOR LDAP USERS

*The typical setup for most Polarion users is passwd file authentication for the polarion user with failover to LDAP for company users (this is also the default Polarion setup). For such setups, you do not need to enter the polarion user to your LDAP users.*

## Configuring OLE Object Support

It is possible to import Microsoft Word documents that contain OLE objects. Polarion can display OLE Object thumbnails during Word document import. However, some additional third-party image converter software must be installed and configured before you can import such Word documents. OLE Objects in documents must contain their thumbnails in *.emf* or *.wmf* file formats, and the image converter used must support their conversion into JPEG. OLE Objects themselves are not imported, only their thumbnails.

The recommended image converter software is ImageMagick ([www.imagemagick.org/script/binary-releases.php](http://www.imagemagick.org/script/binary-releases.php)). Windows distributions are pre-configured for ImageMagick. An empty folder is provided on path `[POLARION_HOME]\imagemagick` which contains a README file describing what to download, how to unpack it and what to do next.

### Configuring Polarion for OLE

There are some system properties in the `_common.properties` system configuration file that you will need to review and set after installing the image converter. Refer to comments in the section for the `com.polarion.oleconverter.app` property that explain the settings.

There is an additional system property in the `polarion.properties` file that specifies the target image format for the OLE thumbnail conversion. Set property `com.polarion.oleconverter.convertedImageFormat=png` to have thumbnails converted to PNG, or `com.polarion.oleconverter.convertedImageFormat=jpg` to have thumbnails converted to JPEG format.

The system properties are located on path

`[POLARION_HOME]\polarion\configuration\` (Windows), or  
`[POLARION_HOME]/etc/` (Linux).

## Multiple Repository Setup

If you use Polarion in a large enterprise with many users and projects, the standard single-repository installation may not be adequate. There are two Polarion features that enable you to work with multiple repositories, but they are fundamentally different. You need to understand the basics of each feature before deciding which approach to multiple repositories best meets your needs.

The **External Repository** feature gives you the ability to link Polarion artifacts stored in Polarion's integrated repository with changes (revisions) in source code stored on one or more external SVN or Git repositories. After installation, you can configure Polarion to use one or more external repositories in addition to the SVN repository bundled and installed with Polarion. For information on this feature, see Help, Administrator's Guide: *Configuring Repositories*.

The **Clustering** feature enables you to run Polarion on multiple servers, either physical, virtual, or a combination. The topography can be set up to host multiple Polarion servers running on separate machines each with its own Polarion repository, and/or multiple machines all accessing a single Polarion repository. (Polarion servers on any node can optionally be configured to access external repositories, as described above.)

Special installation and configuration procedures beyond the scope of this guide are required to set up a clustered multi-server environment. These are covered fully in the [Polarion® ALM Enterprise Setup Guide](#) (PDF), available for download on the Polarion ALM web site.

## Subversion Optimization

Polarion uses a Subversion (SVN) repository as its main data storage. There are 2 topics in the Administrator's Guide component of Polarion online Help that provide guidance for administrators about optimizing Subversion for best performance. It is recommended that you review them before going into production with a new or updated installation.

- Administrator's Guide: Advanced Administration: Topics: Optimizing Subversion
- Administrator's Guide: System Maintenance: Topics: Maintaining Subversion (SVN)

## Accessing the Portal

A shortcut *Open Polarion Web Portal* is provided in the same folder as previously mentioned. You can use it to open the portal login screen in your default web browser after all system components are running.

On your first login after installation, you can login with the default system administrator credentials:

- User ID: **admin**
- Password: **admin**

If the installation is not yet licensed and activated, the login screen first asks you to choose between running with the built-in evaluation license, and activating the installation with a production license. See [Licensing and Activation](#) for additional information.



## Configuring Single Sign-on (SSO)

This section describes SSO authentication methods and configurations for Polarion. Starting with the Polarion 17 release, Polarion supports authentication using Kerberos tokens and Teamcenter Security Services in addition to the existing authentication methods.

### Kerberos SSO

Polarion supports standard Kerberos v5 authentication using the GSS-SPNEGO mechanism.

The authentication procedure flow is as follows:

1. The client machine logs into the domain and obtains a Kerberos token.
2. If the browser is configured to use the Kerberos token for authentication, the browser sends this token when negotiating authentication with a site that is a member of the same domain.
3. The Polarion server gets the token from the client and validates it against the Domain controller.
4. If the domain controller successfully validates the token, Polarion authenticates the user and logs them in.

### Prerequisites - Configuring the Domain

A set of prerequisites is required in order to configure Polarion to authenticate using Kerberos tokens:

1. All involved machines need to be members of the same domain, e.g. yourdomain.com
  - a) Domain server (e.g. kdc.yourdomain.com)
    - Hosting KDC (Key Distribution Center), and DNS services.
  - b) Domain clients (e.g. client123.yourdomain.com). (Polarion user client machines.)
  - c) Polarion server (e.g. polarion-server.yourdomain.com). (The server running Polarion.)
2. Client and Polarion machines need to have corresponding domain accounts (principals):  
Domain user: `user1@yourdomain.com`  
Polarion service: `HTTP/polarion-server@yourdomain.com`
3. Client browsers are configured for Kerberos authentication.

### Notes:

1. The HTTP service component in the principal name is required, otherwise a Kerberos authentication with the polarion-server will not work.
2. All user accounts must have the following checked:  
**Properties → Account → Account options → This account supports Kerberos AES 128 bit encryption.**  
The generated keytab needs to respect this configuration as well, so make sure to set the `-crypto` option below so that it's in line with the principal configuration.

## Configuring Polarion

Polarion configuration consists of a few simple steps:

1. Generate a keytab file with password key for the Polarion principal using the ktpass tool.
  - a. Run the following command as an Administrator on the Domain server:

```
ktpass -out polarion.keytab -princ HTTP/polarion-server@YOURDOMAIN.COM  
-mapuser polarion-server -kvno 0 -crypto AES128-SHA1 -pass password.123  
-ptype KRB5_NT_PRINCIPAL
```
  - b. Copy the generated **polarion.keytab** file into the **configuration** directory of the Polarion server.
  - c. On the Polarion server, create a **login.conf** file in Polarion's **configuration** directory with following data:

```
Polarion {  
  com.sun.security.auth.module.Krb5LoginModule required  
  doNotPrompt=true  
  principal="HTTP/polarion-server@YOURDOMAIN.COM"  
  useKeyTab=true  
  keyTab="C:/Polarion/polarion/configuration/polarion-server.keytab"  
  storeKey=true  
  debug=true;  
};
```

2. Configure Polarion to use SPNEGO authentication and set the following properties in **C:/Polarion/polarion/configuration/polarion.properties** :

```
com.siemens.polarion.security.auth.method=spnego  
java.security.krb5.realm=YOURDOMAIN.COM  
java.security.krb5.kdc=kdc.yourdomain.com  
java.security.auth.login.config=C:/Polarion/polarion/configuration/login.conf
```

Once you have successfully configured the domain, the domain server, the domain client and the Polarion server, users will not see the Polarion login page, but will be immediately authenticated using the provided Kerberos token and logged into Polarion as the same user with same login ID.

## Configuring Client Browsers

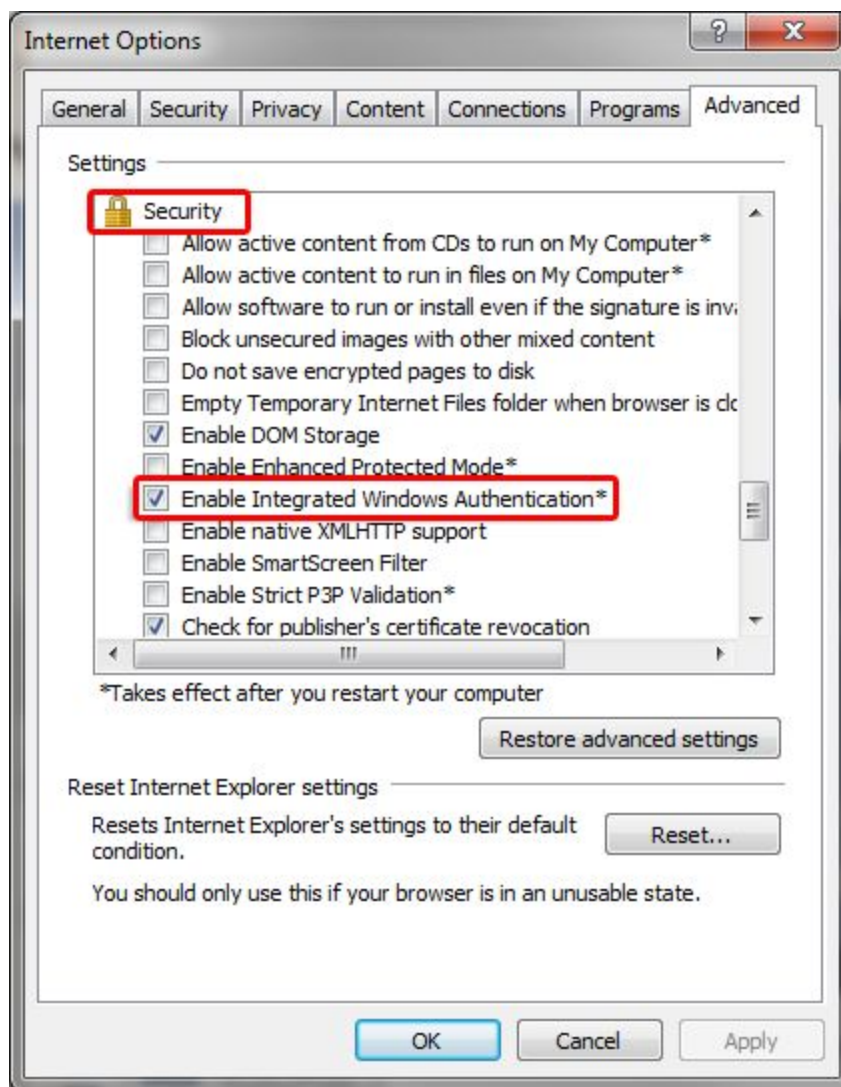
All currently supported browsers (found in the *Release Notes* section of the **README.html** file), can be configured to use Kerberos authentication with Polarion.

### Internet Explorer 11, Chrome and Edge

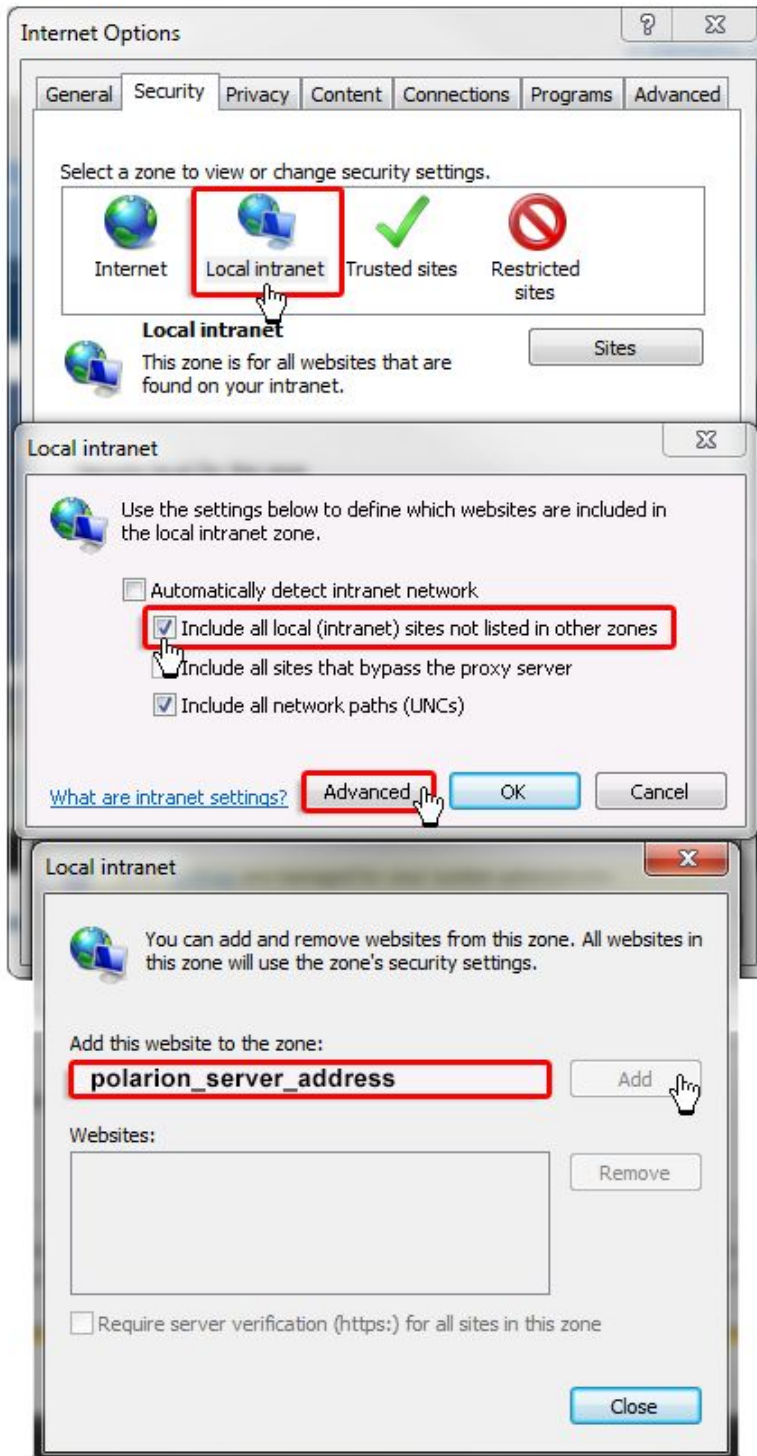
The above browsers are configured via *Internet Options*.

(Accessible from either the Windows Control Panel, or via [Internet Explorer settings](#).)

1. Enable Integrated Windows Authentication in **Internet Options** → **Advanced** → **Security**

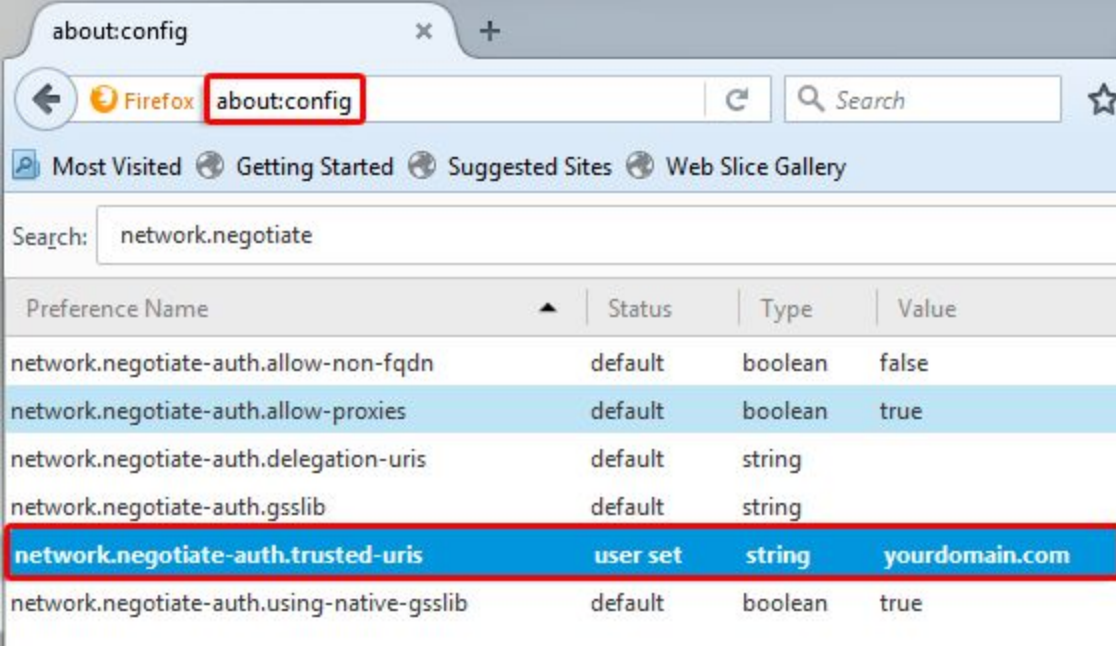


2. Ensure that your Polarion server is a member of the **Local intranet** sites in:  
**Internet Options → Security → Local intranet → Sites.**
  - a. Check "**Include all local (intranet) sites not listed in other zones**".
  - b. Add it explicitly among the site on the "**Advanced**" tab.



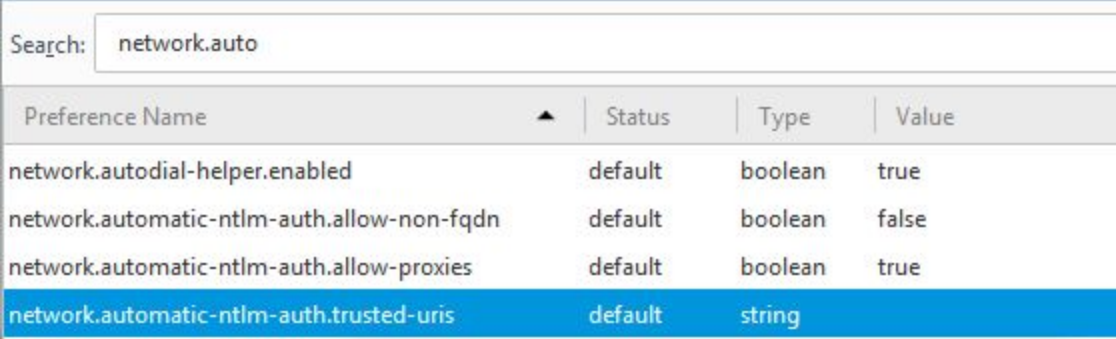
## Firefox

1. Open Firefox settings by typing `about:config` in the address bar.
2. Locate the following preferences:  
`network.negotiate-auth.trusted-uris`  
`network.automatic-ntlm-auth.trusted-uris`
3. Set them (**double-click** or **right click** and “**Modify**”) to the name of your domain (e.g. `yourdomain.com`), or the Polarion server’s hostname (e.g. `polarion.yourdomain.com`).



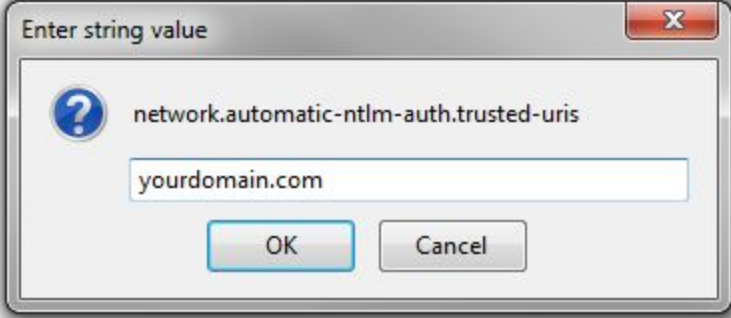
The screenshot shows the Firefox `about:config` page. The address bar contains `about:config`. The search bar contains `network.negotiate`. A table lists preferences, with `network.negotiate-auth.trusted-uris` highlighted in blue and its value set to `yourdomain.com`.

Preference Name	Status	Type	Value
<code>network.negotiate-auth.allow-non-fqdn</code>	default	boolean	false
<code>network.negotiate-auth.allow-proxies</code>	default	boolean	true
<code>network.negotiate-auth.delegation-uris</code>	default	string	
<code>network.negotiate-auth.gsslib</code>	default	string	
<code>network.negotiate-auth.trusted-uris</code>	user set	string	<code>yourdomain.com</code>
<code>network.negotiate-auth.using-native-gsslib</code>	default	boolean	true

The screenshot shows the Firefox `about:config` page. The search bar contains `network.auto`. A table lists preferences, with `network.automatic-ntlm-auth.trusted-uris` highlighted in blue.

Preference Name	Status	Type	Value
<code>network.autodial-helper.enabled</code>	default	boolean	true
<code>network.automatic-ntlm-auth.allow-non-fqdn</code>	default	boolean	false
<code>network.automatic-ntlm-auth.allow-proxies</code>	default	boolean	true
<code>network.automatic-ntlm-auth.trusted-uris</code>	default	string	

The screenshot shows a dialog box titled "Enter string value" for the preference `network.automatic-ntlm-auth.trusted-uris`. The input field contains `yourdomain.com`. The dialog has "OK" and "Cancel" buttons.

## Teamcenter SSO

Polarion supports Teamcenter Security Services (TcSS) authentication.

The authentication procedure flow is the following:

1. The client machine connects to Polarion and if a user is not authenticated, Polarion redirects them to the TcSS login service.
2. A user needs to be authenticated in TcSS to obtain an SSO token and then the TcSS login service redirects them back to Polarion.
3. The browser provides this SSO token when negotiating the authentication with Polarion.
4. The Polarion server validates the token with the TcSS identity service.
5. If the identity service successfully validates the token, Polarion authenticates the user and logs them in.

## Prerequisites - TcSS Configuration

A set of prerequisites is required in order to configure Polarion to authenticate using TcSS tokens:

1. Teamcenter Security Services needs to be installed on the network and connected to the company's directory service. See the [Security Services Installation/Customization](#) documentation.
2. The following services need to be installed and accessible by the Polarion server and client machines:
  - SSO Login Service** on e.g. `http://tcss.yourdomain.com/ssoLogin`
  - SSO Service** on e.g. `http://tcss.yourdomain.com/ssoService`
3. Configure an application ID (e.g. PolarionProd) for Polarion in the Identity Service registry, that points to the Polarion server's hostname.  
(E.g. <http://polarion.yourdomain.com/>) **NOTE:** Do not append **/polarion** - just point to the root.
4. Check that these services work correctly as described in the installation documentation above and that a user is able to login at **`http://tcss.yourdomain.com/loginservice`**.

## Polarion Configuration

Configuring Polarion to use TcSS authentication only requires that the following system properties are set in the system configuration file:

```
C:/Polarion/polarion/configuration/polarion.properties
```

```
com.siemens.polarion.security.auth.method=tcss
```

```
com.siemens.polarion.security.tcssso.serviceUrl=http://tcss.yourdomain.com/ssoService
```

```
com.siemens.polarion.security.tcssso.loginUrl=http://tcss.yourdomain.com/ssoLogin
```

```
com.siemens.polarion.security.appId=Polarion (The default value is "Polarion".)
```

Once you have successfully configured the TcSS services and the Polarion server, users will not see Polarion's login page, but will be immediately redirected to the TcSS login page where they can enter their credentials. After authenticating in TcSS, an SSO session will be created and users will be redirected back to Polarion and logged in as the same user with the same login ID.

## Other SSO Considerations

### Subversion Access

All primary Polarion data is stored in the Subversion repository. In order to ensure the traceability of commits to their authors, the ability to write to the Subversion still needs to be subject to authentication, even when SSO authentication is used with Polarion.

Because Subversion does not support standard SSO mechanisms, Polarion manages the credentials in the Subversion **passwd** file for SSO authenticated users.

The following requirements on Subversion access need to be met for SSO authentication setup:

1. Polarion is configured to access the Subversion repository on behalf of users via the http protocol by default. This configuration is controlled by the following system property:  
**repo=http://polarion-server.yourdomain.com/repo**

2. Apache needs to use file authentication (i.e. the passwd file) for the repository location as the primary authentication source. This is configured in the **polarionSVN.conf** file of the Apache configuration.

```
<Location /repo>
...
AuthBasicProvider file
...
</Location /repo>
```

3. The Polarion configuration needs to point to Apache's passwd file via the following system property: **svn.passwd.file=\${com.polarion.data}/svn/passwd**

**Note:** A Polarion update sometimes requires direct access to the Subversion repository in order to adjust or add new configuration files. For this purpose, you will need at least one user with direct access to the repository using known credentials in the password file.



### Direct Access to Subversion

For direct access to the Subversion repository via the existing *Location /repo*, configure this location to only authenticate via LDAP by removing the file authentication directive from:

```
[Apache configuration]/conf/extra/polarionSVN.conf .
```

```
<Location /repo>
...
AuthBasicProvider file ldap
...
</Location /repo>
```

Then configure a second access location for internal Polarion access that will use the authentication via the password file that is being managed by Polarion.

```
<Location /repo-internal>
# Enable Web DAV HTTP access methods
DAV svn
# Repository location
#SVNPath "C:/Polarion/data/svn/repo"
# Write requests from WebDAV clients result in automatic commits
SVNAutoversioning on

# Our access control policy
#AuthzSVNAccessFile "C:/Polarion/data/svn/access"

# No anonymous access, always require authenticated users
Require valid-user

# How to authenticate a user. (NOTE: Polarion does not currently support HTTP Digest
# access authentication.)
AuthType Basic
AuthName "Subversion repository"
#AuthUserFile "C:/Polarion/data/svn/passwd"
AuthBasicProvider file
</Location>
```

### Features Requiring Special Configuration in an SSO Setup

In a typical SSO setup, access to the server needs to be authenticated by a secure token. If there are also clients that are not able to obtain this token, Polarion provides a fallback authentication via credentials, that need to be explicitly allowed for the following features. As a prerequisite, Polarion needs to be connected to the LDAP server to validate the user credentials.

1. **Access to webservices from external client in SSO environment**
2. **e-signatures in Kerberos SSO environment**



## Troubleshooting

### SSL

To connect to the Polarion server using an HTTPS connection, make sure that Java trusts the certificate used to encrypt the communication. This especially applies to the Teamcenter Security Services Setup, Where the TCSS web application is deployed to Tomcat that runs on Java.

(For more information refer to Polarion **Help** → **Importing a Certificate to the Java Keystore**.)

### File Encoding

Support for national characters in usernames may require additional configuration. By default, Polarion encodes the Apache *passwd* and *access* files using the default Java file encoding, that matches the default OS system encoding. This default encoding may not be compatible with the character set used in LDAP. To discover what this default is set to, check for the file.encoding (e.g. file.encoding = Cp1252) property in the Polarion logs.

To start using national characters in usernames with SSO, first configure Polarion to use right charset. (On Windows Cp1252 or similar for non-Latin languages is recommended).

**To do so, follow these steps:**

1. Stop Polarion
2. Configure the following Polarion properties:  
`com.polarion.platform.repository.passwdFileEncoding=CHARSET`  
`com.polarion.platform.repository.accessFileEncoding=CHARSET`  
`svnkit.http.encoding=CHARSET`
3. Manually convert the existing *access* and *passwd* files to the target character set if the original and target character sets are not compatible. Without these steps, your existing files will be corrupted the next time they are modified by Polarion.
4. Start Polarion

### Kerberos Hints

While implementing SSO in Polarion, we ran into several pitfalls. In the majority of cases they were caused by a misconfigured Kerberos environment. (Be it principals, DNS, keytab, or the encryption type.) We did our best to provide enough details on a failed login page to aid in possible troubleshooting.

#### Wrong Token Provided by the Browser

In some cases the browser may be sending an NTLM token instead of the Kerberos one. This typically happens if you are accessing the hostname of the right server, but the Kerberos principal is configured for FQDN, or vice versa.

Check if your browser is configured correctly for Kerberos authentication and check that the principal for your Polarion service is using the right hostname and HTTP prefix.



Unexpected failure during authentication:  
`javax.security.sasl.SaslException: Unsupported authentication mechanism NTLM`

#### Polarion Unable to Log In to KDC

In other cases Polarion may not be able to login into the KDC to validate the token obtained from the client. If this is the case, the keytab file has probably been exported incorrectly and the message, or one similar it, below will appear.



Unexpected failure during authentication:  
`javax.security.sasl.SaslException: Could not initiate login  
caused by: javax.security.auth.login.LoginException: Unable to obtain password from user`

## LDAP Authorization

In a new installation, users are authorized using the Subversion integrated policy access functions (directives **AuthzSVNAccessFile** and **AuthUserFile** in **polarionSVN.conf** file). If you have LDAP infrastructure, you can make Polarion authorize users against the LDAP database.

Information on performing this configuration, together with some examples, is provided in the **polarionSVN.conf** configuration file. The file is located at:

**[POLARION\_HOME] \bundled\apache\conf\extra\polarionSVN.conf**

After modifying the configuration file, the Apache server must be restarted to reflect the changes.

For more information about the Apache LDAP modules and their capabilities, visit these web pages:

[https://httpd.apache.org/docs/2.4/mod/mod\\_authnz\\_ldap.html](https://httpd.apache.org/docs/2.4/mod/mod_authnz_ldap.html)

and [https://httpd.apache.org/docs/2.4/mod/mod\\_ldap.html](https://httpd.apache.org/docs/2.4/mod/mod_ldap.html).

You can find information on configuring Polarion to work with LDAP in the Polarion Help topic Administrator's Guide: User Management: Integrating Polarion Server with LDAP/Active Directory.

## Next Steps after Installation

Once you have installed Polarion and logged in to the Portal, consider taking a look at the demo projects (assuming you installed demo data). Click on the drop-down control in the Navigation panel on the left, select *Open Project or Project Group*, and open any project in the *Demo Projects* group. The information in the Getting Started with Projects topic of the User Guide will help you get going.

You may want to do some initial global customizations such as custom Work Item types, Workflow, Reports, SSL Support, etc. You will find topics on these configurations in the Administrator's Guide provided in the online Help system.

Once you have your Polarion system running, and any global customizations done, you are ready to begin setting up your own projects and user accounts. Look up the following topics in the Administrator's Guide: *Creating and Managing Projects* and *Managing Users and Permissions*.

## Removing Polarion

Should you ever need to remove Polarion, an uninstaller program is provided for with Windows installations. Like the installation program, it has a "wizard" that guides you step by step through the process. The main thing you need to know is where to find the shortcut to the uninstall program:

**\$POLARION\_HOME\$\polarion shortcuts\Uninstall.**

The uninstaller offers you the option of total removal of all files, or the possibility to keep data and repository files after removing Polarion.

### WARNING

Polarion's Subversion repository is stored in the [POLARION]\polarion folder. Be sure this repository does not contain production data that must be preserved! If it does, be sure to make a backup before uninstalling Polarion.

## Manually Updating Third-party Software

Polarion bundles the following versions of third party components:

- Java 1.8.0\_111
- Apache 2.4.25
- Subversion 1.9.4
- PostgreSQL 9.4.1

All are 64-bit versions. If you are running the bundled version and you would like to update to a newer service release of Subversion and Apache, please follow the steps in the following sections.

## 1. Update Subversion

After you update Subversion you must also update Apache (see the next section). Be sure you plan enough time to do both updates.

1. Visit <http://www.apachehaus.com/cgi-bin/download.plx> and download *Mod Subversion 1.9.3 for Apache 2.4.x x64*.
2. Unpack the zip archive to some temporary folder.
3. Stop the Polarion and Apache servers.
4. Stop the existing “Svnserve” windows service if it is running.  
(Ctrl +Alt +Delete ➡ Task Manager ➡ Services tab ➡ Right click on “Svnserve” Stop Service.)
5. Back up folders `$POLARION_HOME$\bundled\apache` and `$POLARION_HOME$\bundled\svn`
6. Delete the following files:
  - a. All files within the `$POLARION_HOME$\bundled\svn` folder.  
(But do not delete the folder itself.)
  - b. From folder `$POLARION_HOME$\bundled\apache\bin\` ...  
`intl3_svn.dll`  
`libdb48.dll`  
`libsvn_*.dll`
  - c. From folder `$POLARION_HOME$\bundled\apache\modules\` ...  
`mod_authz_svn.so`  
`mod_dav_svn.so`
7. Copy content of the folder within the unpacked temp folder from **Step 2** to `$POLARION_HOME$\bundled\svn`, so that there is a folder `$POLARION_HOME$\bundled\svn\bin`.
8. Copy these files...  
`$POLARION_HOME$\bundled\svn\bin\` ...  
`mod_authz_svn.so`  
`mod_dav_svn.so`  
 ...to folder `$POLARION_HOME$\bundled\apache\modules\`
9. Copy these files...  
`$POLARION_HOME$\bundled\svn\bin\libsvn_*.dll`  
 ...to folder `$POLARION_HOME$\bundled\apache\bin`
10. (Optional). Update Apache.

## 2. Update Apache

You should update Apache after updating Subversion. The Polarion server, PostgreSQL and Apache should already be stopped for that update.

1. You can download the appropriate Apache binaries at <http://www.apachehaus.com/cgi-bin/download.plx>.  
Installation Help for Apache HTTP server is available from Apache at <http://httpd.apache.org/docs/2.4/install.html>.
2. Install the downloaded Apache on any machine to get the content of the installation folder.
3. Back up the folder `$POLARION_HOME$\bundled\apache`
4. If you use Apache installed as a service, then uninstall Apache service by running:  
**`$POLARION_HOME$\bundled\apache\bin\httpd.exe -k uninstall -n Apache2Polarion`**

5. Delete content of `$POLARION_HOME$\bundled\apache` *except the following*:
  - a. Folder `$POLARION_HOME$\bundled\apache\conf`
  - b. Files in folder `$POLARION_HOME$\bundled\apache\bin` :  
`msvcr*.dll`  
`stopApache.js`  
`libsvn_*.dll`
  - c. Files in folder `$POLARION_HOME$\bundled\apache\modules` :  
`mod_authz_svn.so`  
`mod_dav_svn.so`

You may have trouble deleting `rotatelog.exe` . If so, you can kill the `rotatelog.exe` process using Windows Task Manager.

6. Copy the content of the Apache installation folder from step 2, *except* for the `conf` folder, to `$POLARION_HOME$\bundled\apache` .
7. Remove or disable the following lines in the `httpd.conf` file. (To disable, insert the `#` character at the start of the lines, as shown below.)  
`# LoadModule authn_default_module modules/mod_authn_default.so`  
`# LoadModule authz_default_module modules/mod_authz_default.so`  
`# DefaultType text/plain`
8. Add the following lines to the `httpd.conf` file:  
`LoadModule access_compat_module modules/mod_access_compat.so`

```
LoadModule authz_core_module modules/mod_authz_core.so
LoadModule authn_core_module modules/mod_authn_core.so
```

9. Not mandatory, but recommended: in

```
$POLARION_HOME$\bundled\apache\conf\extra\httpd-default.conf
```

increase the value of `MaxKeepAliveRequests` to `10000`.

10. If you use Apache installed as a service, install Apache service. Use the following command in a single line:

```
$POLARION_HOME$\bundled\apache\bin\httpd.exe -k install -n
Apache2Polarion
```

11. Start Apache, PostgreSQL and the Polarion server.

## Updating Java

Manually updating to a newer Java major version is not usually recommended. Polarion updates include new major versions of Java as soon as possible after support has been added to the system and tested.

## 4. Technical Support

Polarion is problem-free for most people... at least that's been our experience. However, it's impossible to anticipate all the conditions and environments where Polarion may be used. For those cases when issues arise, Polarion's Technical Support team maintains the Customer Self-service Portal which includes an extensive Knowledge Base of common problems and solutions and troubleshooting info, as well as the possibility to submit, manage, and review your own specific support cases.

For information about the portal and technical support options, please visit:

<https://polarion.plm.automation.siemens.com/techsupport/resources>



## APPENDIX

### Enabling Email Notifications

If you did not configure email notification settings in the installation program, you can do this after installation. by setting the host name in the *announcer.smtp.host* property in the *polarion.properties* file located in `[POLARION_HOME]\polarion\configuration` . There you should also set the *announcer.smtp.user* and *announcer.smtp.password* properties to a valid e-mail account on the SMTP host specified in *announcer.smtp.host*. You may want to create a dedicated email address on your SMTP host for use by the Polarion notifications system.

When this configuration is correctly set up, the system will send notification e-mails about various events according to the notification targets configuration. For information on configuring email notifications, see Help, Administrator's Guide: *Configuring Notifications*.

### Default Parameters and Settings

This section contains reference information about default Polarion parameters used by the Polarion installer for Windows.

1. Polarion installation root folder: `C:\Polarion`.
2. Polarion components (default folder is: `C:\Polarion\bundled`):
  - a. Apache2 HTTPD-server (`C:\Polarion\bundled\apache` )
  - b. Java Runtime Environment (`C:\Polarion\bundled\java` )
  - c. Subversion (`C:\Polarion\bundled\svn` )
  - d. PostgreSQL (`C:\Polarion\bundled\PostgreSQL` )
3. Default settings for Apache:
  - a. *Install type*: for One-click installation as Service, otherwise user-specified (Service or Console App, default: Service)
  - b. *HTTP port*: 81 for One-click installation, otherwise port user-specified (default: 80)
  - c. *Shutdown port*: 8887
  - d. *SMTP host*: localhost for One-Click Installation, otherwise the value is generated by adding mail. to the domain name taken from the registry. Remember, this is just documentation of default values. You would specify actual value for your system during installation.
4. Default settings for Polarion:
  - a. *Install type*: for One-click installation, as Console application, otherwise user specified (default: as Windows service).
5. Default system property setting for PostgreSQL database connection:  
`com.polarion.platform.internalPG=polarion:polarion@localhost:5433`  
Parameters: <connecting user name>:<password>@<hostname>:<port>

## Supported Microsoft Office® Versions

Users of a Polarion® ALM™ product supporting current or legacy Microsoft Office data interchange features for Microsoft Word™ and/or Excel®, please note the following table which outlines compatible versions of Microsoft Office applications.

				Microsoft Office Version				
Feature	Action	Extension	Format	2003	2007	2010	2013	2016
Polarion LiveDocs (2011)	Word Import & round-trip	DOCX	DOCX	<b>X</b> 1	✓	✓	✓	✓
	Excel Import & round-trip	XLSX	XLSX	<b>X</b> 1	✓	✓	✓	✓
Polarion LiveDocs (2010) <sup>2</sup>	Excel export	XLS	XML	<b>X</b> 2	<b>X</b> 2	<b>X</b> 2	<b>X</b> 2	<b>X</b> 2
	Word export	DOC	XML	<b>X</b> 2	<b>X</b> 2	<b>X</b> 2	<b>X</b> 2	<b>X</b> 2

Supported = ✓ / Unsupported = **X**

### NOTES:

1. Beginning with Polarion version 2014-SR1, Microsoft Office 2003 is no longer supported in Polarion LiveDocs™.
2. Prior to Polarion version 2011, “Live Documents” referred to Microsoft Office Word and Excel documents based on special document templates developed by Polarion, which could define and store Polarion Work Items. Beginning with version 2011, Polarion refactored the technology completely but maintained backward compatibility. Beginning with Polarion version 2014-SR1 support for this legacy format is dropped completely.

This software and related documentation are proprietary to Siemens Product Lifecycle Management Software Inc .

©2017 Polarion AG.

Polarion is a registered trademark of Polarion AG. Polarion ALM, Polarion REQUIREMENTS, Polarion QA and Polarion VARIANTS are trademarks or registered trademarks of Polarion AG.

Siemens and the Siemens logo are registered trademarks of Siemens AG. NX, Solid Edge, and Teamcenter are trademarks or registered trademarks of Siemens Product Lifecycle Management Software Inc. or their subsidiaries in the United States and in other countries. All other trademarks, registered trademarks, or service marks belong to their respective holders.

© 2017 Polarion AG