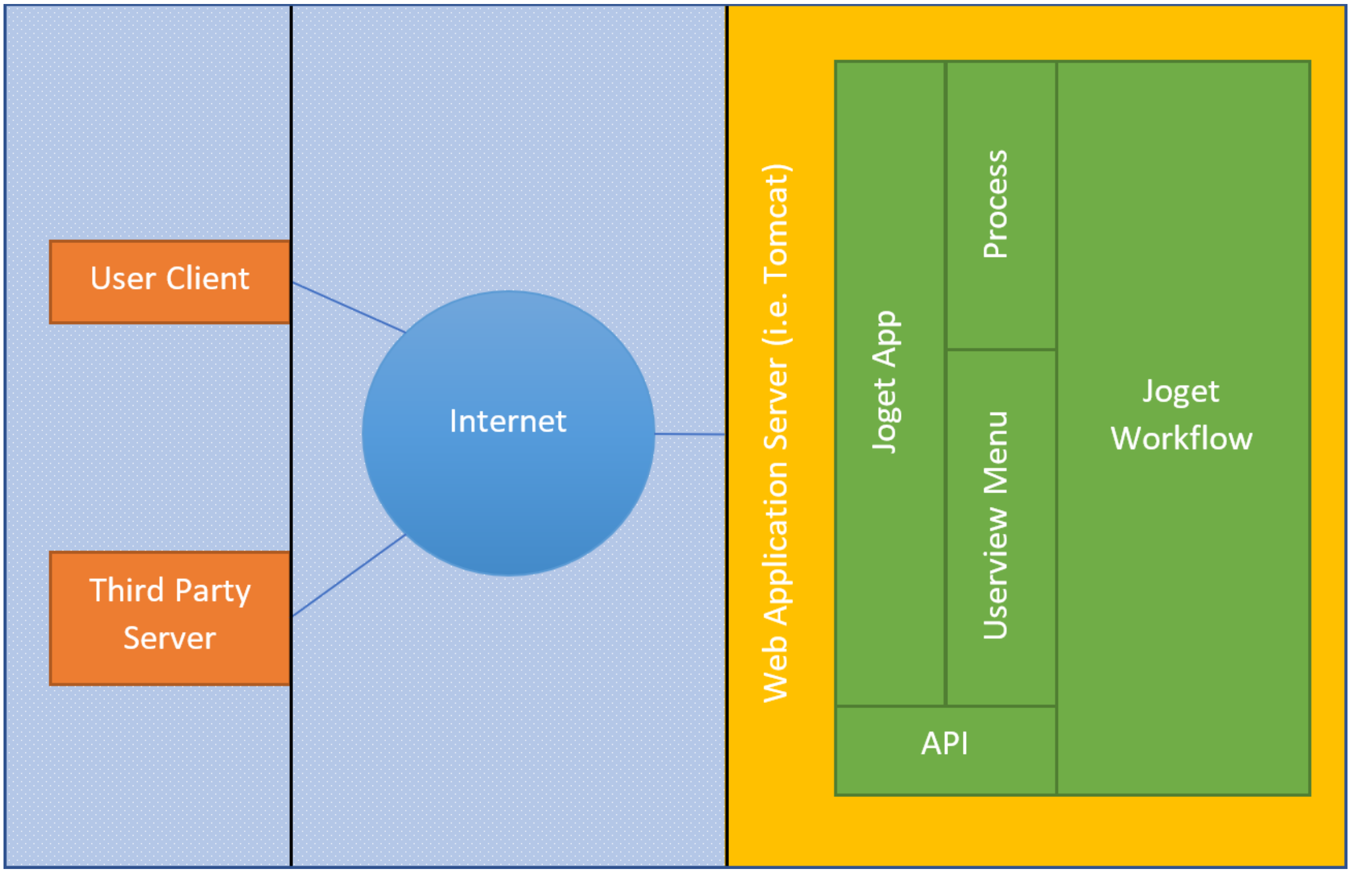
Security Best Practices



SSL

Enabling SSL would ensure that communication between the end user's browser to be server is secure. Please see Setting Up SSL on Tomcat to learn more.

ⓘ **What is SSL?**

SSL (Secure Sockets Layer) is the standard security technology for establishing an encrypted link between a web server and a browser.

Without SSL

Without the use of SSL between the end client and the server, any data sent between these 2 parties will be susceptible to data sniffing by hackers as the data packets travel from end to end.

Domain Whitelist for API Calls

Refer to API Domain Whitelist in Settings to whitelist domains that are consuming Joget's APIs.

ⓘ **Advantage**

By enabling this option, only servers white listed are able to communicate with the server.

Directory User Access Control

Maintaining good password policy management would ensure that user's password is kept safe. Security Enhanced Directory Manager is recommended to be used. The **Security Enhanced Directory Manager** features enhanced security and control on user management.

Enabling Multi-Factor Authentication using TOTP is also an added strength to it.

Advantage

By enabling this option, this will increase security of the user's login information.

Without SSL

Without the use of SSL between the end client and the server, login information will be sent in non-encrypted, clear text to the end server.

Process Start White List

Make use of this feature located under Map Participants to Users to limit on who can start a process instance.

Userview Menu Permission Control

Permission Control is used to exert control and manage access to various components in a developed Joget App. There are 4 main components/areas where permission control can be exerted. They are:-

- Userview
- Userview Category
- Form
- Form Section

Showing the App in App Center only after user is logged on

The most common practice is to list down apps in the App Center only if the user is logged in. To do so, head to the Userview Properties of your app, and locate **Permission Type** and set it to **Logged In User**.

Read more at Permission Control.

Password Encryption

During application design, any sensitive information such as password may be encrypted for security purpose. You may change the key and salt used in a Joget Workflow server to further enhance its security.

Making changes to the key and salt will render all passwords unusable in an existing server therefore it is only recommended to do during initial server installation.

Import/Export App

In an exported app, any password saved in the application design will be encrypted as well. Hence, when the app is imported into another server, be sure to reconfigure all saved password as servers with different key and salt would render the passwords unusable.

Locate the file **customApplicationContext.xml** in **\apache-tomcat-8.5.14\webapps\jw\WEB-INF\classes** and add in line 6-9 as shown below.

```
<beans xmlns="http://www.springframework.org/schema/beans"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xsi:schemaLocation="http://www.springframework.org/schema/beans http://www.springframework.org/schema
/beans/spring-beans-2.5.xsd">

    <bean id="dataEncryption" class="org.joget.apps.workflow.security.SecureDataEncryptionImpl">
        <property name="salt" value="NEW-VALUE-GOES-HERE"/>
        <property name="key" value="NEW-VALUE-GOES-HERE"/>
    </bean>

</beans>
```

Replace line 7 and 8 salt and key value to your own one.