Security

Ignition and Security

Security in Ignition is simple to set up and allows many ways to control who can use it and what things they have access to. You can add as many user sources as you want, and even connect with Active Directory to add your existing users! You can even store other information like phone numbers, schedules, and more.

Ignition uses what's known as role-based security. Actions such as accessing the Gateway, logging in to a project, interacting with a client, or modifying windows in a project can require specific roles. Users are stored in a User Source, which is a list of users, roles, and other useful information such as the user's name or phone number. By default, Ignition creates User Source automatically for you to use. The User Source contains the "admin" user that initially grants access to the Gateway.

Features of Ignition Security

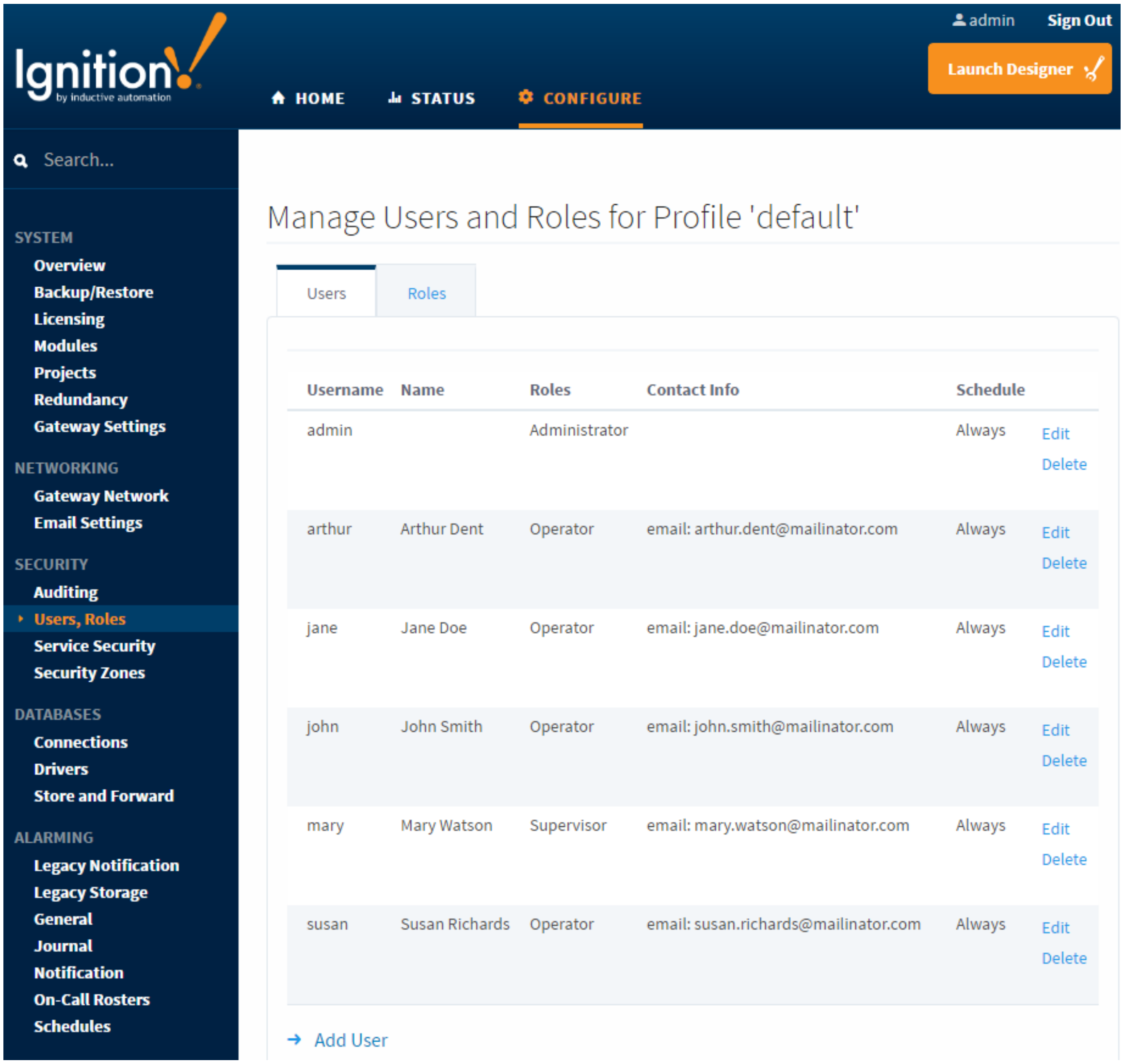
- Role-based security. Users can be assigned any number of roles, and roles can be checked at any time, in any project.
- Adding new users is simple. New users can be granted preexisting roles, and restrictions in Ignition will immediately be applied.
- Database Authentication Sources from external systems can be utilized in Ignition. Users will not need an additional password to log in to a client.
- Authenticate against Active Directory (AD). Roles can be created from AD groups, or custom roles can be assigned to users. Additionally, Ignition clients support Single Sign ON (SSO).
- Multiple User Sources. Use the same User Source for all projects, or create a separate User Source for each project.

On this page ...

Users, Roles, and User Sources (Authentication)

Role-based security works under the concept that each user may be assigned to various roles. Security policies are then defined in terms of these roles, rather than defined for specific users. This allows users to be reassigned, removed, and added without affecting the logic of the security policy.

The users and their roles are stored in **User Sources**. An Ignition Gateway may have many different User Sources defined, each governing the security of different aspects of the Gateway. For example, logging into the Gateway might be governed by one User Source, while the security in a project is governed by another.



Manage Users and Roles for Profile 'default'

Username	Name	Roles	Contact Info	Schedule	
admin		Administrator		Always	Edit Delete
arthur	Arthur Dent	Operator	email: arthur.dent@mailinator.com	Always	Edit Delete
jane	Jane Doe	Operator	email: jane.doe@mailinator.com	Always	Edit Delete
john	John Smith	Operator	email: john.smith@mailinator.com	Always	Edit Delete
mary	Mary Watson	Supervisor	email: mary.watson@mailinator.com	Always	Edit Delete
susan	Susan Richards	Operator	email: susan.richards@mailinator.com	Always	Edit Delete

There are several types of User Sources that offer various features. For example, the Internal User Source offers the ultimate in ease-of-use: you simply define the users, their passwords, and the roles within the Ignition Gateway configuration web interface. In contrast, the Active-Directory User Source offers the power of integrating Ignition with a corporate security infrastructure. Users, passwords, and roles would be managed centrally by the IT department.

Contact Information and Schedules

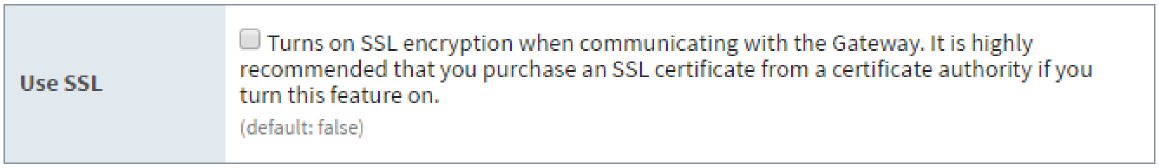
User Sources are also used for other aspects of the system besides security. For example, the alarm notification system also takes user contact information to send alarm notification messages. A schedule can be defined for a user to control when they are able to log in, and when to receive alarm notification messages. Language preferences can be defined for each user to better support individual user's preferred language (Localization).

Gateway Security

The primary purpose of Gateway security is to protect access to the two most critical areas of Ignition: the Designer, and the Gateway. Many important resources are configured in these areas, so access to each Gateway section (**Home**, **Status**, and **Configure**), as well as the Designer, can be limited by role. Furthermore, the User Source authenticating Gateway access can be separate from those authenticating project access, allowing for compartmentalized authentication management.

SSL

Ignition can use SSL with just one click by going to Configure > System > Gateway Settings! SSL (Secure Sockets Layer) is a widely adapted encryption protocol used all over the world. One of the main uses of SSL is to protect transferred data between a web-server and a web browser, but Ignition clients can also benefit from this protocol when communicating with the Gateway. Once enabled, data transferred between the Gateway and the client (such as recipe values, historical data, or customer accounts) will be encrypted. Thus the data will be unreadable by any malicious parties that may be eavesdropping in the network.

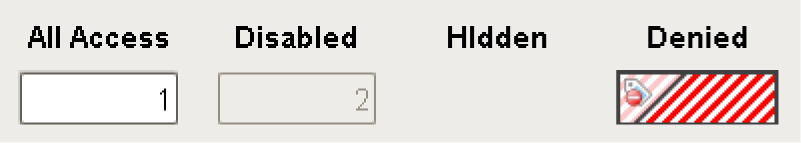


Use SSL	□ Turns on SSL encryption when communicating with the Gateway. It is highly recommended that you purchase an SSL certificate from a certificate authority if you turn this feature on. (default: false)

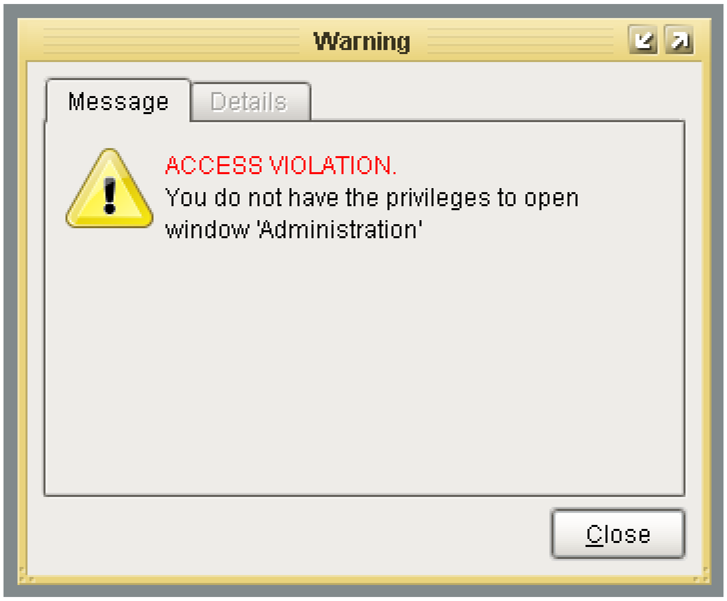
Project Security

Client Based Security

Easily control who can change values or even see your data. Security can be applied just about everywhere in a project. Individual components can check user roles from the client. Using Ignition's built-in Security Settings, the behavior of the component can change dynamically. This allows for a complex control scheme where components can be **hidden**, **disabled**, or **obscured** on a per-role basis. Project development becomes much easier! Develop once, and the Security Settings will control which components the user can interact with.



Security settings can also be applied to entire windows. Windows with administrator-level functionality can safely be added to a project without worry of unauthorized access.



ACCESS VIOLATION.
You do not have the privileges to open window 'Administration'

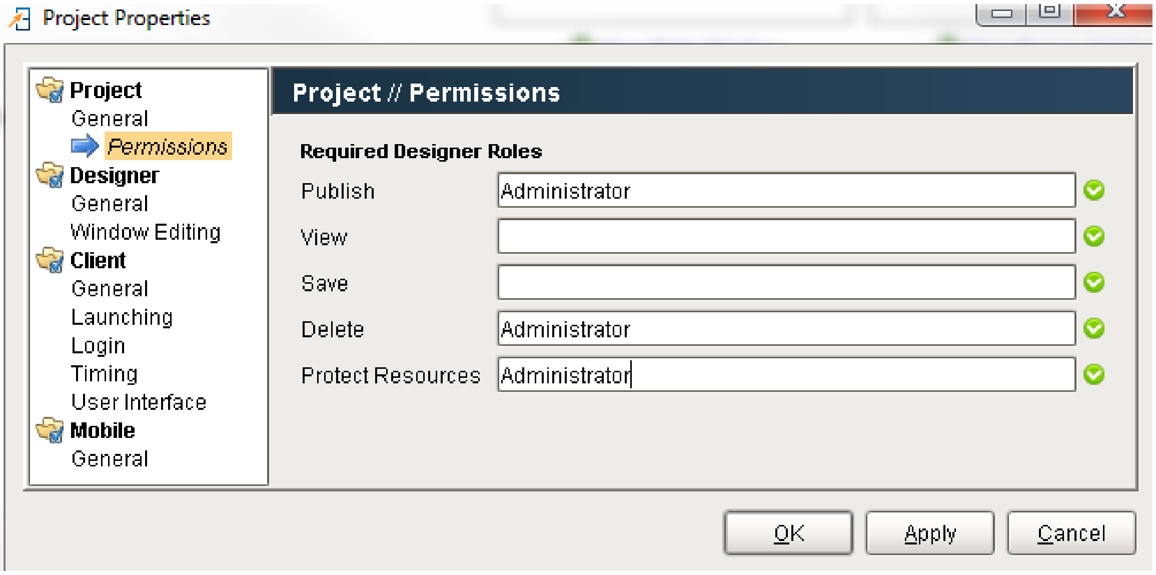
Furthermore, the entire project can be restricted to specific roles. This is commonly used when area-specific roles are given to users: i.e., operators in one location should not be able to view another site's project.

Looking for Multi-Conditional Security?

Roles can be checked from any client-based scripting event, so your security settings can go beyond just roles and account for specific times of the day, month, IP address, user names, lunar cycles, and anything else you can think of!

Project Based Access

Too many developers working on the same project could cause a number of problems: resource overlap, naming convention conflict, project creep, focus loss, and so on. To help with larger teams, Designer specific access levels can grant project permissions to certain roles. Actions such as **Publishing**, **Deleting**, or even **Viewing** the project in the designer can be restricted to certain roles.



Required Designer Roles

Publish: Administrator ✓

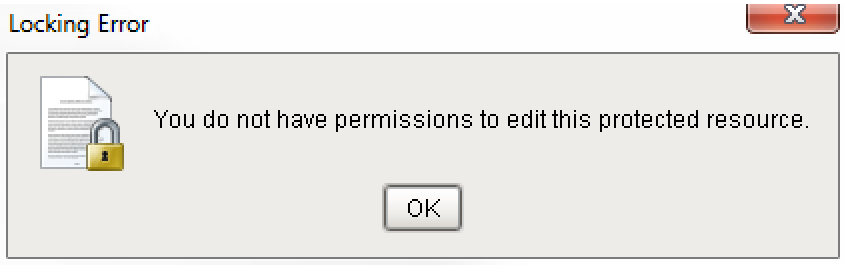
View: ✓

Save: ✓

Delete: Administrator ✓

Protect Resources: Administrator ✓

Resources can also be protected, meaning that the window, template, or other resource should not be modified. This is a great way to flag an item as "complete" to the rest of the team, without having to manually document this information.



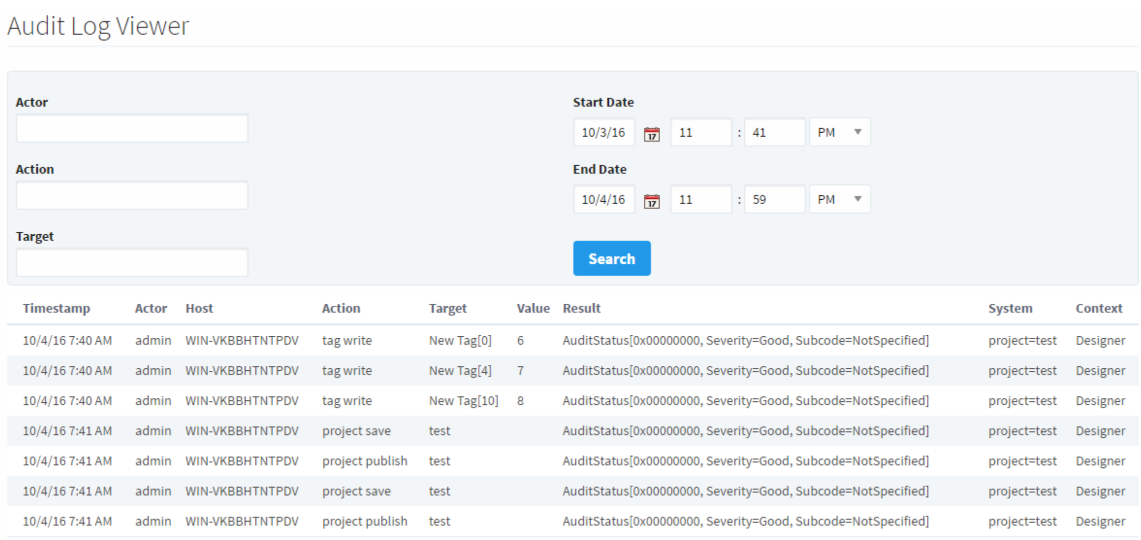
You do not have permissions to edit this protected resource.

Auditing

Automatically record any changes that are made along with who did it, when it happened, and more. Auditing enables Ignition to record specific events to a SQL database. An Audit Profile can be used to record information for various systems in Ignition:

- **Project Auditing**: Record when users log in, log out, or write to any tags.
- **Notification Profile Auditing**: Record when a notification message would be sent. This reports who the notification was sent to, as well as if the attempt was successful or not.

Ignition provides a simple interface to view the Audit Log on the Gateway, and more.

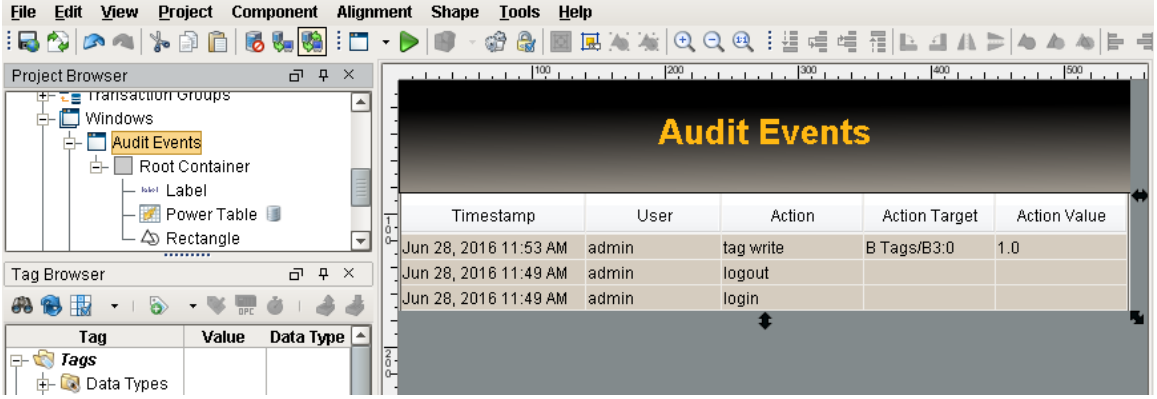


Audit Log Viewer

Actor	Start Date 10/3/16 11 : 41 PM
Action	End Date 10/4/16 11 : 59 PM
Target	

Timestamp	Actor	Host	Action	Target	Value	Result	System	Context
10/4/16 7:40 AM	admin	WIN-VKBBHTNTPDV	tag write	New Tag[0]	6	AuditStatus[0x00000000, Severity=Good, Subcode=NotSpecified]	project=test	Designer
10/4/16 7:40 AM	admin	WIN-VKBBHTNTPDV	tag write	New Tag[4]	7	AuditStatus[0x00000000, Severity=Good, Subcode=NotSpecified]	project=test	Designer
10/4/16 7:40 AM	admin	WIN-VKBBHTNTPDV	tag write	New Tag[10]	8	AuditStatus[0x00000000, Severity=Good, Subcode=NotSpecified]	project=test	Designer
10/4/16 7:41 AM	admin	WIN-VKBBHTNTPDV	project save	test		AuditStatus[0x00000000, Severity=Good, Subcode=NotSpecified]	project=test	Designer
10/4/16 7:41 AM	admin	WIN-VKBBHTNTPDV	project publish	test		AuditStatus[0x00000000, Severity=Good, Subcode=NotSpecified]	project=test	Designer
10/4/16 7:41 AM	admin	WIN-VKBBHTNTPDV	project save	test		AuditStatus[0x00000000, Severity=Good, Subcode=NotSpecified]	project=test	Designer
10/4/16 7:41 AM	admin	WIN-VKBBHTNTPDV	project publish	test		AuditStatus[0x00000000, Severity=Good, Subcode=NotSpecified]	project=test	Designer

Since the activity is recorded into a SQL database, the data from the Audit Log can also be viewed by running a simple query. This way an "auditing" window can be added to any project, and allow any users (or users restricted to specific roles) the ability to view Activity recorded in the Audit Log. Even other systems can access the data in the Audit Log by querying the database!



Audit Events

Timestamp	User	Action	Action Target	Action Value
Jun 28, 2016 11:53 AM	admin	tag write	B Tags/B3:0	1.0
Jun 28, 2016 11:49 AM	admin	logout		
Jun 28, 2016 11:49 AM	admin	login		

In This Section ...