Ignition Guide

Ignition!

Security Hardening Guide

Updated for Ignition 8.1

inductive automation

(800) 266-7798
www.inductiveautomation.com

Introduction

Welcome to Inductive Automation's Ignition Security Hardening Guide. Inductive Automation is committed to security and strives to make the product as secure as possible. This document is intended to provide general guidance on how to set up and secure your Ignition installation.

Included in this document are guidelines specifically for the Ignition software, as well as general suggestions regarding the hardware and network where Ignition is installed. The steps provided are recommendations rather than requirements and should be reviewed for relevance in each implementation.

This guide is best used by reading and following the steps in its entirety. Security is a complex topic and no guide can cover the complete tapestry of the subject; however, this guide should provide good information toward protecting your Ignition installation, as well as covering the basics of securing your overall device architecture. To ensure you are completely covered from a security standpoint, please consult a security firm or security experts in the field.

Step 1: Secure the Gateway

Locking down the Gateway involves restricting how Ignition is accessed from a secure connection standpoint, as well as user permissions. After installing Ignition, enabling SSL/TLS is the first and most important step towards securing the Gateway. Enabling SSL/TLS ensures that all of the next steps toward protecting your Gateway are being communicated across a secure communication channel.

SSL vs. Non-SSL

Secure Socket Layer (SSL) is a widely used security protocol for data that is transported across a network, the network traffic of the Gateway, or the Internet. On the Gateway, this means SSL will encrypt data sent over the HTTP protocol and Web Sockets, used for all traffic between the Designer, Vision Clients, and Perspective Sessions. This protects your Ignition installation from anyone spying on your data as it passes over the network. This is important if data transferred between the Gateway and Clients are sensitive in nature. This also helps to thwart a security vulnerability known as "session hijacking". Without SSL enabled, an unauthorized user may exploit a valid computer session to gain access to information or services, such as your Ignition Gateway.

Technically, SSL is the predecessor to Transport Layer Security (TLS). For historical reasons, "SSL" now colloquially refers to TLS encryption. Through the rest of this document, when we refer to SSL, know that the underpinning technology employed is TLS.

When Enabling SSL

Enable SSL communications in Ignition to set up secure communication to the Gateway webpage as well as Client/Designer communication with the Gateway. In order to encrypt this communication, you will need to acquire and install an SSL Certificate for Ignition. It is highly recommended that you purchase an SSL certificate from a certificate authority or acquire a valid SSL certificate from an IT department if you intend to enable SSL.

Steps for *installing an SSL certificate* can be found on our website.

Force SSL Redirect

1. Go to the **Configure** section of the Gateway.
2. Choose **Networking > Web Server** from the menu on the left.
3. Select the checkbox for **Force Secure Redirect** and click on **Save Changes** at the very bottom of the page. Force Secure Redirect, when enabled, will redirect all unsecure http traffic to its https counterpart.

After SSL is enabled, all Clients, Designers, and web browsers are redirected to the SSL port if they try to use the standard HTTP port. By default, the SSL port is 8043. You can change it to the standard SSL port of 443. (**Note**: In past versions, both the non-SSL and the SSL port needed to be open even when intending to communicate exclusively with SSL. In Ignition 8.0, only the SSL port needs to be opened.)

Renewing SSL Certificates

Most traditional SSL certificates have a cumbersome lifecycle that needs to be renewed often. With Ignition, the SSL certificate renewal process can be simplified and automated using the ACME protocol which is laid out in the guide, *Let's Encrypt Guide for Ignition.* ACME (Automatic Certificate Management Environment) is an automated framework for obtaining and renewing SSL certificates for your domain so that SSL can be enabled in your web server. Let's Encrypt is a "free, automated, and open certificate authority (CA)" using an ACME server that handles SSL certificates. Any domain administrator can spin up an ACME client that points to the Let's Encrypt ACME server to obtain or renew SSL certificates.

Disabling Older Cipher Suites

When SSL is enabled, cipher suites provide essential information on how to communicate secure data between the Ignition server and the user's browser. The user's browser will notify the Ignition server which cipher suites the browser supports, and the most secure cipher suite they both support will automatically be used for communication. Ignition supports many cipher suites and disables most risky suites, but there may be some suites that are enabled by default that can optionally be disabled depending on your company's security posture. It is strongly advised to review the supported cipher suites and disable any older cipher suites that are not needed in your environment. Cipher suites can be excluded in the Gateway settings under **Config > Web Server > Excluded Cipher Suites.**

The suites currently considered strong are ever-changing as security researchers discover flaws and create new algorithms. One source for information on the currently accepted list of strong suites is maintained by SSL Labs. SSL Labs provides a free online test if your Ignition server is accessible from the Internet. Even if it is not, their current list of strong cipher suites can be found in their "SSL/TLS Deployment Best Practices" guide under *2.3 User Secure Cipher Suites.* Please use this source or another to stay up to date with industry recommendations on current best practices for cipher suites support.

Configuring Strong Headers

As of Ignition 8.0.8, Gateway security headers are now set with secure default values. However, Ignition offers the opportunity to change the values of these headers to better fit your network security preferences. For example, enabling **Strict-Transport-Security** (by default is off) sets a timer for how long the browser should remember that Ignition is only accessible using HTTPS. Visitors may initially communicate with the non-encrypted version of the site (i.e., HTTP) before being redirected to HTTPS, which creates an opportunity for a man-in-the-middle attack which can redirect a visitor to a malicious site. Allowing the browser to remember to only use an encrypted connection with Ignition prevents the chances of being intercepted on a non-encrypted line.

In 8.0.10, we added support for strengthening the security of the Gateway's HTTP session cookies using the SameSite attribute. Details can be found in the *Ignition 8.0.10 release notes.*

More information on the HTTP headers and their default values can be found at support.inductiveautomation.com article, *Gateway Security HTTP Headers and Configuration.*

Step 2: Locking the Gateway

In the Gateway, there are three tabs, **Home, Status,** and **Configure.** The **Configure** and **Status** sections of the Gateway are password-protected, and this cannot be removed. For additional protection, you have the option to protect the **Home** section. You can also change the roles that are required to access any of these sections under **Configuration > Gateway Settings.**

Role-based user authentication can be used to lock down Gateway webpage sections as well as the Designer to prevent users from changing the configuration. Each of the Status, Home, and Configure pages can be restricted by roles independently.

Gateway Web Page Security

Details for setting up security for the Gateway can be found in the *user manual.*

1. Go to the **Configure** section of the Gateway.
2. Choose **Configuration > Gateway Settings** from the menu on the left.
3. Enter the roles the user must have in order to access the **Gateway Config Roles, Status Page Roles, Home Page Roles,** and **Designer Roles.**
   Note: Each option can accept any number of roles as long as they are separated by commas. If an option is left blank, any user with any role can log in (generally not recommended).

Step 3: Device, OPC, and MQTT Security

Device connections have historically been made using native device communication protocols. Most PLC manufacturers created their own protocols for communication, and a variety are popular and in heavy use today. Recently, some devices have been released that have OPC-UA and MQTT embedded directly in the devices as well. Each category is secured in a different way.

Direct connections from Ignition to OPC-UA and MQTT devices are generally the most easily-secured connections, although any connection can be secured, given the right configuration and network security.

OPC-UA Communication

OPC-UA provides built-in security whether at the server level or embedded directly on a device. One of the ways that this can be done is to encrypt all communication. Different devices/servers support different encryption levels, but when setting up endpoints, be sure to choose the *signed and encrypted option.* This ensures all data sent over OPC-UA will be encrypted.

Also, when configuring the Ignition OPC Server, trusting remote certificates is required for all secured inbound and outbound connections. **Under OPC UA > Security** trusted certificates can be imported and quarantined certificates can be marked as trusted. Some third-party OPC Servers may require additional steps such as *manually adding the client certificate.*

OPC-UA connections also support user authentication. We recommend using a strong password and changing it periodically as defined by IT standards.

MQTT

MQTT provides a handful of built-in security features. Data transferred between the Publisher and Broker, as well as between the Broker and Subscriber, should be sent over SSL. In addition to this encryption, Username/Password Authentication is supported and should be utilized to protect the data. MQTT also supports Access Control Lists (ACLs) which limit user access based on topic name space. These security measures should be implemented whether the broker is local or hosted in the cloud. For more information on securing MQTT communication, see *the MQTT modules user manual.*

Native Device Communication

In addition to encryption between Ignition and OPC-UA or MQTT devices/servers, communication between Ignition and other devices should also be protected. Since these devices often do not support encryption or certificates, a common practice is to keep them on a separate private OT network. Ignition can provide a layer of separation between the OT/private and the IT/public network to make tags available securely without exposing the devices behind the scenes. Other security options include placing Ignition and devices on a VLAN network with encryption enabled, setting up routing rules on the network or using an edge-of-network computer (such as *Ignition Edge* on an IPC) to act as a bridge between the device and the network.

We recommend consulting with a network security professional to help identify which option is best for you.

Step 4: Identity and Access Management

When securing an Application you must consider both authentication and authorization. Authentication determines who is logging in, whereas authorization determines their privileges. Ignition has the following tools to help satisfy almost any kind of authentication and authorization strategy required:

- User Identification and Authentication - User Sources and Identity Providers
- Authorization - Role-Based Access Control, Location-Based Access Control through Security Zones
- Security Levels - a hierarchical, inheritance-based access control model which builds off of Roles, Security Levels, and other attribute sources

User Identification and Authentication

Ignition manages users through Identity Providers (IdP). Ignition has a built-in IdP, but can also connect to third-party IdPs such as Okta and Duo via SAML or OpenID Connect.

Ignition Identity Provider

The Ignition IDP supports three main user sources, *Internal Authentication*, *Database Authentication*, and *Active Directory Authentication.*

Internal Authentication

From the Gateway Web Page, users can be managed directly within Ignition. These users are local to the Ignition Gateway where they are defined.

Database Authentication

The Database Authentication type uses an external database instead of storing data inside Ignition. Managing users is done via direct interaction with the database.

Active Directory Authentication

The Active Directory Authentication profile uses Microsoft's Active Directory over LDAP (Lightweight Directory Access Protocol) to store all the users, roles, and more that make up an Authentication profile. Active Directory Groups are used for Ignition's roles and user-role mappings.

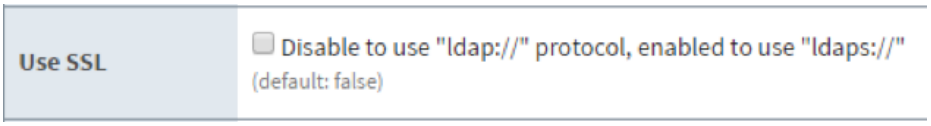
While using an Active Directory User Source, administration of users and roles is through Active Directory itself, and not manageable within Ignition. Thus, adding new users to an Active Directory User Source or modifying pre-existing users requires the modifications be made from Active Directory, usually through an AD Administrator.

In Vision, if a seamless experience is desired with no login prompt, *SSO* (Single Sign-On) can be enabled in conjunction with Active Directory. This allows the window's username to be automatically used as credentials in Ignition. User groups should be given minimum access to the application and then additional roles added as needed. This prevents users from having unintended access in the application.

In Perspective, if a seamless experience is desired with no login prompt, consider using ADFS or another IdP instead of Ignition's internal IdP. See the Third-Party Identity Provider (Perspective) section below.

LDAP Protocol Security

The active directory User Source communicates with a Microsoft Active Directory server through the LDAP protocol. To prevent snooping on authentication, encryption should be implemented. In the advanced options for a new Active Directory User Source, Ignition has a setting to force LDAPS.



Badge Authentication

Another secure method is RFID authentication support for the Ignition Identity Provider (requires Ignition 8.0.5+) allowing you to associate badges with users. Entering your credentials on a screen allows others to see your username and therefore weakens your security. With RFID enabled, users do not have to enter their username and password, and instead must have a physical badge in order to log in to the Session. If your organization makes use of RFID badges, it is recommended that Badge Authentication Method is enabled and set to default and Badge Secret is also enabled. Badge Secret will require the user to type in their password after scanning their badge. This added layer of security is useful in numerous situations. For example, in the event that a user's badge is stolen and used by another individual, the added layer of a required password would keep the thief from accessing the Session, since a password will still be required for access.

More information on the *Badge Authentication Method* can be found on our website.

Third-Party Identity Provider (Perspective)

Utilizing an external service allows you to utilize features that Ignition doesn't support natively such as multi-factor options like push notifications or biometrics. They are also valuable if IT has standardized other applications on a particular SSO solution.

Currently, third-party IdPs are only supported by Perspective Clients. In the future we intend to include support for Vision Clients, the Designer, and the Gateway Webpage.

General Authentication Suggestions

User Accounts

To ensure User Account integrity, a strong password policy should be defined including password length and complexity requirements. Establishing a password expiration schedule and quickly removing former user accounts are strongly recommended. When using Ignition's Internal Authentication, these settings can be found under the "Password Policy" section. Generic accounts should be avoided.

Group Access and Disabling Auto-Login

Generic logins pose a security risk in any system. If Auto Login is enabled, any user that launches a project is granted basic access. To mitigate this risk, Auto Login should be disabled and each user should have their own unique credentials.

Authorization

Once a user is authenticated, authorization determines what they have access to. Authorization can be determined by a variety of conditions including roles and location.

Role-Based Security

Each user is granted privileges by assigning one or many roles. Roles are not default types but rather created custom during development. Roles can be defined inside Ignition or mapped to Active Directory groups or an IdP's attributes. The level of access granted by a given role is determined in "Step 5: Define Application Security".

Location-Based Security

A Security Zone is a list of Gateways, Computers, or IP addresses that are defined and grouped together. This group now becomes a zone which can have additional policies and restrictions placed on it. While Users and Roles restrict access to specific functions within the Gateway, such as making certain controls read-only for certain users and read/write for others, Security Zones provide this functionality to the Gateway Network, location-based access control in Vision, Perspective, Alarm Notifications and Status, and History Provider and Tag Access. Security Zones allow the application to restrict access based on where the client connects from in addition to a user's role-based privileges. This allows for greater control over the type of information that is passing over the network, improving security and helping to keep different areas of the business separate, while still allowing them to interconnect.

Using Security Zones

Sometimes, in addition to knowing who the user is, it is important to know where they are sending a command from. An operator may have permissions to turn on a machine from an HMI, but if they were to log in to a project on a different Gateway in the network that had remote access to those Tags, it might not be a good idea to let the operator write to those Tags from a remote location where they can't see if the physical machine is clear to run.

This is where Security Zones come in. Security Zones themselves don't define the security, they instead define an area of the Gateway Network, breaking up Gateways and network locations into manageable zones that can then have a security policy set on them. Once there are zones defined, a security policy can be assigned to each zone, and a priority of zones can be set in the event that more than one zone applies in a given situation.

Information on how to set up *Security Zones* can be found on our website.

Security Levels (Perspective)

Security Levels are a user-set hierarchy that defines access permissions, or roles, inside a Perspective Session. This authorization system provides a way to map user roles defined from an Identity Provider (IdP) to Ignition roles. Creating security levels can be used to restrict certain people from being able to access specific functions within the Gateway, such as access to certain Perspective sessions, visibility of components in a view, or read/write permissions. This will improve security and allow better management of the information and level of control specific users are granted on the network.

Using Security Levels

When working with the Gateway, it is important that all users are able to get the information or control they need in order for operations to run smoothly. An operator may need to be able to read and write to Tags, and view the status of each plant. However, a manager may need to view the status of the plant and read the Tag values, but should not be able to write to the Tag values. Making sure that each user has the correct permissions is vital to the security of the operation.

With security levels, roles can be defined in order to allow certain users more or less control of the Gateway. Once the roles are defined, the security level rules can be defined on the Gateway in order to allow users access to specific security zones, projects, and various other attributes. These roles can also be used in the Designer in order to limit a role's viewing access and/or read/write capabilities.

Information on how to set up *Security Levels* can be found on our website.

Step 5: Define Application Security

Ignition is a software platform for creating custom applications to suit your needs. These applications could be for HMI (Human Machine Interface), SCADA (Supervisory Control And Data Acquisition), Database Front End and more. Each of the applications require customizable security. Ignition allows for security to be defined at any level from clients and projects down to individual tags.

Vision Client Security

In Clients, security settings can be applied to individual windows or components. While users with different roles may view the same project from the client, the functionality and accessibility can change based on their assigned roles. Generally, higher level access provides full functionality to all contents of a project, and lower level access is restricted to generalized read-only privileges. However, client security settings are flexible enough to accommodate any security requirements.

Information on how to set-up *Client Security* can be found on our website.

Perspective Session Security

In Sessions, security is managed through Identity Providers (IdP). Identity providers offer a way for users to log into Ignition using credentials that are stored outside of Ignition. Once the identity provider is set up in the Gateway, a security level can be assigned to a Perspective Session, which will grant only the users with the specified security level access to the Session. Generally, the higher-level access provides full functionality to all components of the project and lower-level sets more restrictions, such as read/write privileges.

Information on how to set up *Perspective Session Security* can be found on our website.

Perspective Views Security

Added security to a Perspective View allows more granularity of the security in a Perspective Session. An operator, for example, may need to view a Perspective Session but shouldn't be allowed to access a user management View. The operator can be granted access to the Session, but the user management View can be restricted to a higher level role. Adding security levels to both the Perspective Session and Views allows only the roles with proper privileges to be allowed to view or edit content and thus improving the security of the Perspective Session and control of information on a project.

Information on how to set up *Perspective Views Security* can be found on our website.

Component Scripting Security

Both Vision and Perspective contain role-based component scripting security to ensure that non-privileged users are not allowed to run scripts that can be potentially harmful to operations or manipulate information that a user does not have clearance for.

More information on *Vision* and *Perspective* component scripting can be found on our website.

Designer Security

When several users are all working on the same project, managing changes to the project can become cumbersome. By default, all users with Designer access can modify, delete, save, and publish all resources available in the Designer. In some situations, it is desirable to limit what each user can do in the Designer. Ignition has several built-in Designer restriction methods to help in these scenarios.

Our website contains instructions for restricting *editing*, *creating*, and *protecting project resources* and *global resources.*

Tag Security

Tag security is often the best way to configure security for data access. By defining security on a tag, you are applying those rules for that tag across all windows and components that use the specified tag in the project. This is opposed to configuring security on each component that controls the tag.

If a user opens a window that has components that are bound to a tag that the user doesn't have clearance to read or write to, the component will get a forbidden overlay.

You can add read/write security to individual tags through the Designer. Custom Access Rights must be set to use the Role-based permissions.

Named Queries

One of Ignition's key features is the ability to easily log, edit and retrieve data from SQL databases. By default, all database interaction is limited to defined queries on the Ignition Gateway, which may be called from clients based on the credentials of the user. These queries can be parameterized to allow for dynamic database interaction while ensuring only relevant data is accessible. It is recommended to only use parameters for individual variables rather than allowing longer SQL chunks to prevent SQL injection.

This feature can be turned off to allow any SQL query to be run directly from an open client. While this can be powerful for adding flexibility to the platform, it also leaves the data potentially exposed. If client-authored queries are enabled, be sure to use SSL and not use auto-login or any shared accounts.

Access to these named queries can be limited using the normal Ignition permission model including roles and security zones.

If upgrading from a previous version (Ignition 7.9.3 and before), unrestricted client queries will not be disabled by default for existing projects. Secure the system by either converting existing queries to named queries or limit client queries to appropriate roles and security zones.

Step 6: Set up Audit Logging

Audit Profiles allow Ignition to record details about specific events that occurred. Audit Profiles are simple to set up and immediately start to record events. By default, only tag writes, SQL UPDATE, SQL INSERT, and SQL DELETE statements are recorded. This allows you to keep track of which user wrote to which tag, or modified which table. Furthermore, a time-stamp is recorded, so you can easily track the changes, outline, and order of events.

Once some changes have been made to a tag or a database table, Ignition will begin recording.

More Information regarding *Audit Profiles* can be found on our website.

AUDIT_EVENTS_ID	EVENT_TIMESTAMP	ACTOR	ACTOR_HOST	ACTION	ACTION_TARGET	ACTION_VALUE
1	2016-07-25 17:50:09	admin	IU-WorkStation	tag write	B Tags/B3:1	1.0
2	2016-07-25 17:50:51	admin	IU-WorkStation	tag write	B Tags/B3:1	100.0
3	2016-07-25 17:50:53	admin	IU-WorkStation	tag write	B Tags/B3:1	2.0
4	2016-07-25 17:50:56	admin	IU-WorkStation	tag write	B Tags/B3:1	8.0
5	2016-07-25 17:51:20	admin	IU-WorkStation	query	update audit_events set acto...	4
6	2016-07-25 17:51:51	admin	IU-WorkStation	query	UPDATE audit_events SET `A...	1

Step 7: Protect the Database

Different databases offer different authentication options. We recommend not using a database owner account such as **root** or **sa**. A separate user account with limited privileges should be created for the database connection with the Ignition Gateway.

Most modern databases also support SSL encryption of the connection between Ignition and the database. If the database is running on a different server, SSL can be enabled by following information available for your database's JDBC driver and internal security settings. Refer to the documentation for your database for more information on enabling SSL JDBC connections from Ignition.

Step 8: Java Security

Ignition runs on the Java Virtual Machine (JVM). As of Ignition 8.0, Ignition comes pre-packaged with its own private copy of the JVM. This means that Java is not, and does not need to be, installed system-wide on any computer in order to run Ignition. By keeping Ignition up-to-date, you are also keeping the private copy of Ignition's JVM up-to-date.

Step 9: Locking down the Operating System (OS)

Removing Unnecessary Programs

Each program is a potential entry point for an attacker so removing unnecessary software and having a vetted list of allowed software can limit vulnerabilities. Not all programs require administrative access and should be run using the minimum credentials required.

Patches and Service packs

To prevent zero-day attacks and limit operating system vulnerabilities, it is recommended to keep up-to-date on OS patches and Service Packs.

Remote Services

On Windows, Remote Registry and Windows Remote Management should be disabled.

On Linux and Mac OS, disable root for everything but 'physical' console.

Ignition Gateway Service User

The Ignition Gateway runs as a service on the OS. This Ignition service should not be executed as a root or admin user. It is best to run Ignition with a user on the OS with minimal privileges.

Firewalls and ports

Firewalls should be in place to restrict network traffic. We recommend closing all ports and then only opening those that are necessary. The following ports are the *default ports* used in Ignition. Only open the ports you are using.

Ports

PORT	OPERATION	PROTOCOL	CONFIGURABLE	DESCRIPTION
8088	Listening	tcp	Yes	Default port setting to access the Ignition Gateway
8043	Listening	tcp	Yes	Default SSL port setting to access the Ignition Gateway
102	Listening	tcp	No	Siemens Step7
2222	Listening	tcp	No	Allen Bradley Drivers (Ethernet/IP I/O DMA)

PORT	OPERATION	PROTOCOL	CONFIGURABLE	DESCRIPTION
4096	Listening	tcp	Yes	Default port for Ignition OPC UA server
4446	Listening	udp/broadcast	Yes	Default receive port for a multicast messages that makes the Gateway discoverable on a local network
5060	Listening	tcp/SIP	No	Send Voice Notification to SIP server
5500	Listening	tcp	Yes	Default port for OPC browse of external tags
6501	Listening	tcp	No	Server fallback port
8000	Listening	tcp/RTP	No	Transfer/com for SIP server
8750	Listening	tcp	No	Port used for Redundancy/Network Configuration
9600	Listening	tcp	No	Omron FINS
17342	Listening	tcp	No	Receive Port for SMS with Alarming
45900	Listening	tcp	No	Callback port for the Mobile Module
44818	Listening	tcp	No	Allen Bradley Drivers (Ethernet/IP Symbolic/General)
135	Outgoing	tcp	No	Default port for DCOM communication (old OPC DA servers)
389	Outgoing	tcp	Yes	Default port for Active directory if this is being used
465	Outgoing	tcp	No	SMTP protocol used if Alarming is configured
502	Outgoing	tcp	Yes	Default Modbus port
1433	Outgoing	tcp	Yes	Default MSSQL port used for connection
1521	Outgoing	tcp	Yes	Default Oracle port used for connection
3050	Outgoing	tcp	Yes	Default Firebirdsql port used for connection
3306	Outgoing	tcp	Yes	Default MySQL port used for connection

Redundant Servers

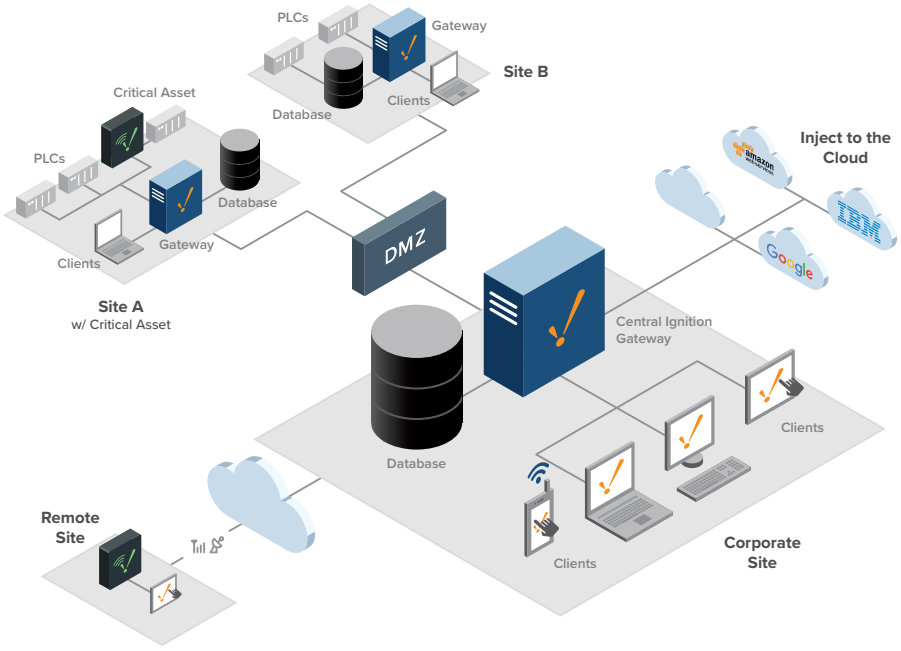
Firewalls must be set up on any server doing redundancy in order to protect the redundancy system from external attacks. The firewall on the main server should only accept incoming connections on port 8750 from the back-up server IP address on Ignition 7.9, and should only accept connections on the Gateway Network port from the back-up server IP address on Ignition 8.0.

DMZ Architecture

A DMZ Architecture (referred to as a "demilitarized zone") contains a subnetwork that accommodates the organization's exposed, outward connecting services. In general, it acts as a point of contact between the organization's internal network and untrusted networks, such as the internet.

The goal of this architecture is to add an extra layer of security to the local area's network, allowing the local network to access what is exposed in the DMZ and keeping the rest of the network protected behind a firewall.

We recommend consulting with a network security professional to help identify the best network options for your organization.



Step 10: Keep Ignition Up-to-Date

Inductive Automation recognizes that software security requires constant effort and maintenance. Security updates are released periodically to ensure continued protection and keeping up-to-date with these updates is strongly recommended.