Security Module How-to

Set up users, roles, and security policies.

How-to The Four Pillars User Interactions Security | Tutorial | How-to Guide | Reference

---

Overview

This guide walks you through configuring the Security module for user authentication, authorization, and access control. You'll create users, define permission groups, set security policies, and integrate with enterprise authentication systems.

Prerequisites:

- Administrator access to the solution
- Understanding of security requirements
- Active Directory/LDAP details (if integrating)

Configuration Workflow
Step 1: Define Permission Groups
Step 2: Configure Security Policies
Step 3: Create Users
Step 4: External Authentication (Optional)
Step 5: RuntimeUsers Configuration
Step 6: Security Monitor
Applying Security On Displays
Runtime Operations
Common Issues
Best Practices
Compliance Features
Security Module Links

Configuration Workflow

1. **Define Permission Groups** - Set access levels for different roles
2. **Configure Policies** - Establish password and session rules
3. **Create Users** - Add local user accounts
4. **Setup External Auth** (Optional) - Connect AD/LDAP
5. **Configure RuntimeUsers** (Optional) - Enable dynamic users
6. **Security Monitor (**Optional) - View active sessions and connections:

Step 1: Define Permission Groups

Permission groups control what users can access in both Designer and Runtime.

Using Pre-defined Groups

Group	Designer Access	Runtime Access	Typical Use
Administrator	Full	Full	System management
Engineering	Modules, no Security	Full	Solution development
Supervisor	View only	Full operations	Shift supervisors
Operator	None	Operations, no tools	Control room operators
Guest	None	View only	Anonymous access

Creating Custom Groups

1. Navigate to **Security Permissions**
2. Click first row to add new group
3. Configure permissions:

Edit Permissions (Designer access):

- **Unrestricted**: Full Designer access
- **Modules**: Select specific modules (Tags, Alarms, Historian, etc.)
- **CreateTags**: Allow tag creation
- **Publish**: Allow solution deployment

Run Permissions (Runtime access):

- **Unrestricted**: Full runtime control
- **Startup/Shutdown**: Control solution execution
- **StartTools**: Access diagnostic tools
- **CreateUsers**: Manage runtime users
- **WebAccess**: Allow web client access

Step 2: Configure Security Policies

Policies define password requirements and session behavior.

Policy Settings

1. Go to **Security Policies**
2. Select or create policy
3. Configure three main areas:

Identification (Password rules):

Setting	Default	Enhanced	Critical
Password Min Length	0	8	12
Invalid Attempts	0	5	3
Password History	0	3	5
Max Password Age (hours)	0	2160 (90 days)	720 (30 days)

E-Signature (Action confirmation):

- **Enabled**: Require password for critical actions
- **TimeoutMinutes**: How long e-signature remains valid

Session (Auto-logoff):

- **Inactivity**: Logoff after idle time
- **Duration**: Maximum session length
- **Both**: Apply both restrictions

FDA 21 CFR Part 11 Settings

For compliance, use these minimum settings:

```
Identification:
  PasswordMinLength: 8
  BlockOnInvalidAttempts: 5
  PasswordHistory: 5
  MaxPasswordAge: 2160 (90 days)

ESign:
  Enabled: True
  TimeoutMinutes: 10

Session:
  AutoLogOff: Both
  InactivityMinutes: 20
  DurationHours: 12
```

Step 3: Create Users

Default Users

User	Purpose	Action Required
Administrator	System management	Set password immediately
Guest	Anonymous access	Configure permissions
User	Generic login	Set password if using

Adding New Users

1. Navigate to **Security Users**
2. Click first row to add user
3. Configure:
    - **Name**: Unique username
    - **Password**: Meeting policy requirements
    - **Permissions**: Select group
    - **Policy**: Select security policy

- **ContactInfo**: Email, phone, full name

Bulk User Import

For multiple users, prepare CSV:

```
Name,Permissions,Policy,ContactInfo
jsmith,Operator,Enhanced,"John Smith,jsmith@company.com"
mjones,Supervisor,Critical,"Mary Jones,mjones@company.com"
```

Import via **Security Users Import**

Step 4: External Authentication (Optional)

Windows Active Directory

1. Go to **Security RuntimeUsers**
2. Enable **Windows AD Integration**
3. Configure:
    - Domain: COMPANY
    - Auto-create users: Yes
    - Default group: Operator

LDAP Server

1. Navigate to **Security RuntimeUsers**
2. Select **LDAP Provider**
3. Configure:
    - Server: ldap.company.com
    - Port: 389 (or 636 for SSL)
    - Base DN: dc=company,dc=com
    - User attribute: sAMAccountName

Step 5: RuntimeUsers Configuration

RuntimeUsers are created dynamically and stored in external databases.

Setup Database

1. Go to **Security RuntimeUsers**
2. Configure storage:
    - **Database**: RuntimeUsers (SQLite default)
    - **Encryption**: Enabled
    - **Auto-create**: Yes

Enable Dynamic Creation

In scripts, create users programmatically:

```
@Security.CreateUser(
    "newuser",
    "<yourPasswordHere>",
    "Operator",
    "Enhanced"
);
```

Step 6: Security Monitor

View active sessions and connections:

1. Go to **Security Monitor** during runtime
2. Review connected users and session times
3. Use @Server.GetAllConnections() in scripts for custom monitoring

---

Applying Security On Displays

Display Security

Entire Display:

1. In **Displays List**
2. Set **EditSecurity** and **RunSecurity** columns

Individual Elements:

1. Select display element
2. Add **Security** dynamic
3. Configure:
    - **Permission**: Required group
    - **ESign**: Require confirmation
    - **Disable**: Block interaction

Tag Security

Protect tag writes:

1. In **UNS Tags**
2. Set **WritePermission** property
3. Only authorized users can modify

Script Security

Control script execution:

```
if (@Client.UserName == "Administrator")
{
    // Admin-only operations
}
```

Runtime Operations

User Login

Manual Login:

```
@Client.LogOn("username", "password");
```

Check Current User:

```
string user = @Client.UserName;
string group = @Client.CurrentUser.Permissions;
```

Logout:

```
@Client.LogOff();  // Returns to Guest
```

Session Management

Monitor active sessions:

```
// Check session time
TimeSpan sessionTime = @Client.SessionTime;

// Force logout if needed
if (sessionTime.TotalHours > 8)
{
    @Client.LogOff();
}
```

Common Issues

User Cannot Login

- Verify credentials correct
- Check account not blocked
- Confirm user not deleted
- Review invalid attempt count

Permission Denied

- Check user's permission group
- Verify specific module access
- Review display/element security
- Confirm runtime permissions

Password Issues

- Ensure meets policy requirements
- Check password age
- Verify password history
- Review complexity rules

Session Timeout

- Check inactivity settings
- Review duration limits
- Verify policy assignment
- Monitor session properties

Best Practices

- **Set Administrator password** - Never leave default blank
- **Use groups not individuals** - Easier management
- **Regular password changes** - Enforce via policies
- **Audit user accounts** - Remove inactive users
- **Test permissions** - Verify access levels
- **Document security model** - Maintain access matrix
- **Use external auth** - Leverage enterprise systems

Compliance Features

FDA 21 CFR Part 11

- Electronic signatures
- Audit trail (via Alarms module)
- User authentication
- Password policies
- Session controls

NERC CIP

- Role-based access
- Password complexity
- Account monitoring
- Session management
- Audit logging

Security Module Links

Explanation - to understand concepts

Modules / User Interactions / Security Module

Tutorials - to learn by doing

Tutorials / User Interactions / Security Module Tutorial

How-to Guides - to accomplish specific tasks

How-to Guides / User Interactions / Security Module How-to Guide

Reference - technical details

Technical Reference / User Interactions / Security Module Reference

In this section...