System Configuration

- Guide to System Settings
- Configuring SAML SSO with OPS-COM
- Troubleshooting - Email Server Communication Errors
- Alarms System Settings
- Defining User Profile Settings
- Account Creation Preferences
- Configuring Multi-Factor Authentication on the User Portal
- Password and Security Settings
- Uploading and Managing Files
- Dashboard Layouts and Custom Items

Guide to System Settings

> System Settings in OPS-COM provide administrators with comprehensive control over the core functionalities and behaviors of their application, primarily impacting the administrative side. This centralized configuration area allows for fine-tuning various components, from general system parameters and security protocols to specific module functionalities like parking, violations, and payments, ensuring the system operates according to organizational needs.

1. Click **System Configuration**, then **System Settings** to access this area.
2. Explore the menus. Hovering over any menu item will explain with a tooltip what this setting controls.
3. Settings in **Blue** are read-only to Admins. Only a Tomahawk User can enable/disable this.

   For help with this contact support@ops-com.com.

> Only Users that have the permission to **Manage System Configuration**, will see the System Settings. If they have that permission, they can edit any system setting available.

Best Practices & Considerations

- **Review All Settings**: System settings are granular and cover many aspects of OPS-COM. Regularly review all components to ensure configurations align with your organization's current policies and operational needs.
- **Security Settings First**: Prioritize the configuration of **Security** component settings (e.g., password expiry, strength requirements, admin lockouts) to maintain a robust security posture for your admin accounts.
- **Email Configuration**: Ensure that all relevant email addresses (Default Notification Email, From Email, Appeal Notification Email, Automated Notification Email) are correctly set up to ensure timely system communications and alerts.
- **Time Zone Accuracy**: Correctly setting your **Time zone** and **Time offset** is critical for accurate timestamping of all system events, permits, and violations.
- **Impact of Toggles**: Be mindful that many settings are simple on/off toggles. Understand the full impact of enabling or disabling a module (e.g., "Enable Violations Module") or a specific feature before making changes.
- **Team Collaboration**: For settings that require OPS-COM Team access to change, communicate your needs clearly to support staff. For other settings, collaborate with your internal teams (IT, finance, enforcement) to ensure changes meet everyone's requirements.

- **Testing Changes**: For significant changes, especially those impacting user-side visibility or core workflows, consider testing in a Preview Space, before applying to your live production system.

Configuring SAML SSO with OPS-COM

What is Single Sign-On (SSO)

Single Sign-On (SSO) simplifies user access to OPS-COM by allowing them to authenticate using their existing, managed corporate accounts. This eliminates the need for separate OPS-COM usernames and passwords, enhancing convenience and security. This article details the setup and configuration of SAML-based SSO with OPS-COM, explaining the necessary fields, metadata exchange, and user synchronization. For more general information about SSO and OPS-COM refer to this wiki article.

Prerequisites and Considerations

Implementing SSO with OPS-COM, specifically using SAML (Security Assertion Markup Language), requires coordination between your organization's Identity Provider (IdP) and OPS-COM as the Service Provider (SP).

- **Paid Feature**: SSO is a paid feature. You must have the setup fee and recurring fees negotiated before proceeding. Contact your Sales Representative or email support@ops-com.com to initiate this.
- > **Login Sources**: You must first follow the instructions to set up Login Sources within OPS-COM, as SSO will be configured as a specific login source.
- **User Management Strategy**: Consider the following:
  - Will you have different Login Sources (e.g., Students/Staff use SSO, but Public Users do not)?
  - Will login sources vary by user type?
  - How do you want to initially get your users into OPS-COM (e.g., pre-import vs. on-the-fly creation)?
  - Do you want users to be created automatically upon their first SSO login?
  - Do you want to keep user information synchronized with your Identity Provider regularly, or will it be a one-time import?
  - What user profile data/fields do you want synchronized between your SSO system and OPS-COM?

    - Can you take advantage of the UserPush APIs for proactive user synchronization?

Your OPS-COM Client Success team will be happy to discuss these options to ensure a smooth and successful setup.

Once the prerequisites are addressed, the SAML setup involves configuring fields for both OPS-COM (as the Service Provider) and your external SAML system (as the Identity Provider).

Configuring SAML Setup

1. Hover over System Configuration, Users, and click Login Sources.
2. Click the pencil icon to edit your login source you created already as mentioned above. You should already have configured the login source to the point of the Unique ID field.

> The settings below must be filled out correctly and saved before you will see the Metadata tab to continue.

Service Provider Fields (Configured in OPS-COM)

These fields define how OPS-COM will interact with your Identity Provider.

- **Unique ID**: **Required** - This is a crucial part of the XML communication between OPS-COM and your SAML system. It is *supplied by your SAML system* and is the value OPS-COM uses to match against its internal `UniqueID` field to identify a user.
- **Entity ID for Service Provider**: **Required** - This value defines the unique SAML integration path within the URL in the metadata. If your OPS-COM system has more than one SAML integration, each `Entity ID` needs to be unique. The value you supply will appear in the integration path like this: `https://client.ops-com.com/auth/saml2/ENTITY_ID_FIELD/acs`. ***Only add the ENTITY_ID_FIELD not the whole URL.***
- **x509 Certificate**: (Optional) This certificate is provided by your Identity Provider (IdP) and can be generated and added to the Service Provider (OPS-COM) for secure communication.
- Private Key: (Optional)

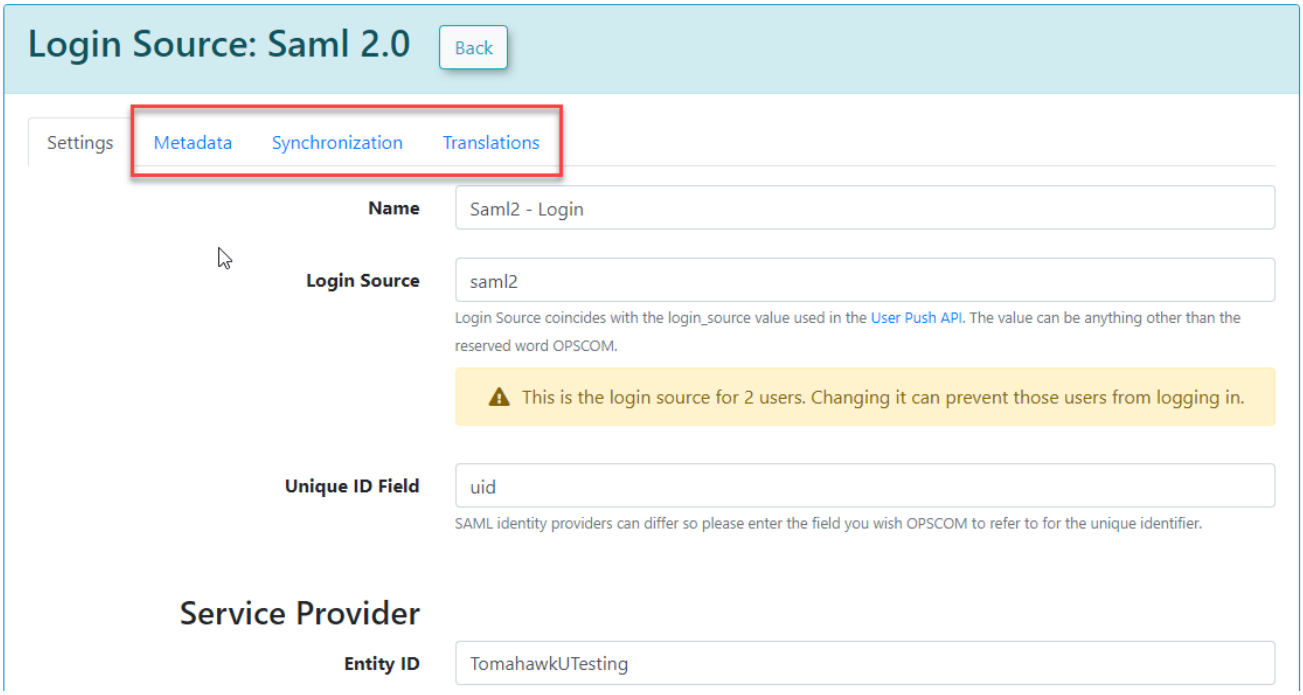
Identity Provider Fields (Configured in OPS-COM, Values from Your SAML System):

These fields capture information from your external SAML system (Identity Provider). You will find these values within your SAML system's metadata (e.g., often displayed under `Federation → Show Metadata` on your SAML installation page).

- You will input values such as the Identity Provider's `Entity ID`, `Single Sign-On URL (SSO URL)`, and `x509 Certificate` (which is often different from the one provided for the Service Provider).

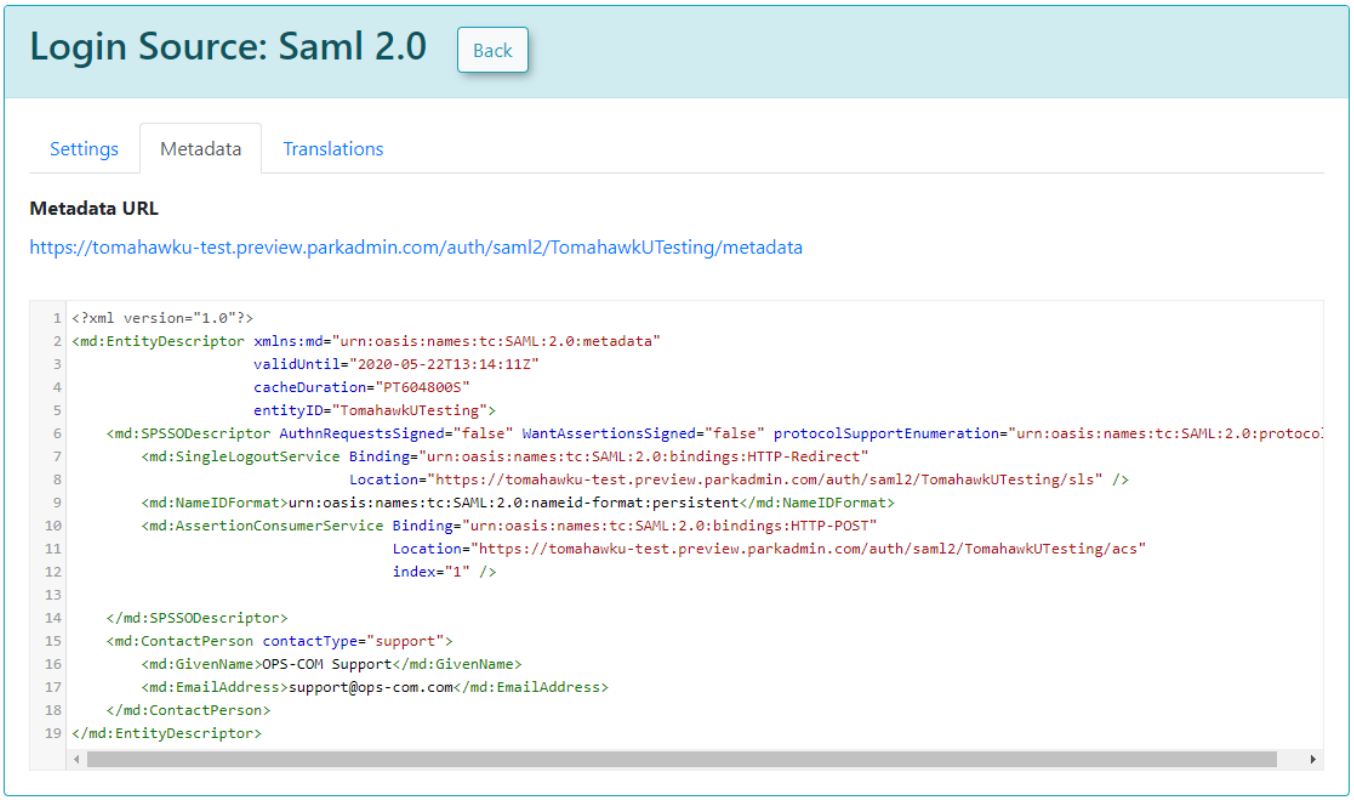
Once these settings have been completed and saved in OPS-COM, you will gain access to additional tabs: ***MetaData****,* ***Synchronization****, and* ***Translations****.*

Using this Feature



Metadata Tab

The **Metadata** tab in OPS-COM provides the XML code that you will need to provide to your Service Provider (OPS-COM, in the context of SAML communication from your IdP's perspective). This XML contains all the necessary information for your Identity Provider to communicate correctly with OPS-COM.



Sample XML File

Sample XML File Explanation: When your external system (e.g., a SimpleSAMLPhp service set up as the identity provider) sends a response back to OPS-COM, it includes an `saml:AttributeStatement` tag containing several attributes. These attributes are required for OPS-COM to match to a user within its system. The most important field in this attribute section is the value used as the permanently unique identifier for a user. For example, if the XML response shows `[uid] => Array ( [0] => 6ddf4027-3397-4e45-8628-0189f60fe91e )`, then `uid` should be entered as the **Unique ID Field** in your **Identity Provider Fields** configuration within OPS-COM. If the unique ID is something else, such as `SAMaccountName`, then that should be used instead.

... DEV-2K8 - DEBUG: Saml2 Incoming User Array ([uid] => Array ([0] => 6ddf4027-3397-4e45-8628-0189f60fe91e) [full name] => Array ([0] => Sarah Knowles) [email] => Array ([0] => sknowles@tomahawk.ca)) []

```
<? xml version = "1.0" ?>
    < samlp:Response xmlns:samlp = "urn:oasis:names:tc:SAML:2.0:protocol" xmlns:saml =
"urn:oasis:names:tc:SAML:2.0:assertion" ID = "_aa1963115aa6490e728c7376f4c8849813bbb..." >
        ...
        < saml:Assertion xmlns:xsi = "http://www.w3.org/2001/XMLSchema-instance" xmlns:xs =
"http://www.w3.org/2001/XMLSchema" ID = "_9efd79bf6425983ee9176f3d33a99d1a9176180..." >
            ...
            < saml:Subject >
                < saml:NameID SPNameQualifier = "MinionOpsComStaff" Format = "urn:oasis:names:tc:SAML:2.0:nameid-
format:transient" >_7a426e0be71f14c1f349db00d7d543b6f7dcb52baa</ saml:NameID >
                < saml:SubjectConfirmation Method = "urn:oasis:names:tc:SAML:2.0:cm:bearer" >
                    < saml:SubjectConfirmationData NotOnOrAfter = "2021-08-24T16:00:41Z" Recipient = "https://minion-
3.dev.parkadmin.com/auth/saml2/MinionOpsComStaff/acs" InResponseTo = "ONELOGIN_bb8a09203c888cf59af4c621a71cfa8f7559c016"
/>
                </ saml:SubjectConfirmation >
            </ saml:Subject >
            < saml:Conditions NotBefore = "2021-08-24T15:55:11Z" NotOnOrAfter = "2021-08-24T16:00:41Z" >
                < saml:AudienceRestriction >
                    < saml:Audience >MinionOpsComStaff</ saml:Audience >
                </ saml:AudienceRestriction >
            </ saml:Conditions >
            < saml:AuthnStatement AuthnInstant = "2021-08-24T15:34:46Z" SessionNotOnOrAfter = "2021-08-24T23:34:46Z"
SessionIndex = "_a7a68666092117d24aab8adecf1b0830622855b85..." >
                < saml:AuthnContext >
                    < saml:AuthnContextClassRef >urn:oasis:names:tc:SAML:2.0:ac:classes:PasswordProtectedTransport</
saml:AuthnContextClassRef >
                </ saml:AuthnContext >
            </ saml:AuthnStatement >


            < saml:AttributeStatement >
                < saml:Attribute Name = "uid" NameFormat = "urn:oasis:names:tc:SAML:2.0:attrname-format:basic" >
                    < saml:AttributeValue xsi:type = "xs:string" >6ddf4027-3397-4e45-8628-0189f60fe91e</ saml:AttributeValue >
                </ saml:Attribute >
                < saml:Attribute Name = "full name" NameFormat = "urn:oasis:names:tc:SAML:2.0:attrname-format:basic" >
                    < saml:AttributeValue xsi:type = "xs:string" >Sarah Knowles</ saml:AttributeValue >
                </ saml:Attribute >
                < saml:Attribute Name = "email" NameFormat = "urn:oasis:names:tc:SAML:2.0:attrname-format:basic" >
                    < saml:AttributeValue xsi:type = "xs:string" >sknowles@tomahawk.ca</ saml:AttributeValue >
                </ saml:Attribute >
            </ saml:AttributeStatement >


        </ saml:Assertion >
    </ samlp:Response >
```

Synchronization Tab

The **Synchronization** tab allows you to configure how user information is managed between your SSO system and OPS-COM.

- **Auto Create/Update User**: To begin, ensure you enable the **Auto Create/Update User** checkbox. This feature allows OPS-COM to automatically create new user profiles when they first log in via SAML, if they don't already exist in OPS-COM. It also enables the system to update existing user information.

- **User Attribute Mapping**: On this tab, you will map the user attributes from your SSO system (your Identity Provider) to the corresponding fields in OPS-COM. For example, your SSO system might send "full name" and "email" attributes, which you would map to OPS-COM's `firstName`, `lastName`, and `email` fields.
- Any field that is mapped and has a value from your SSO side should get updated to the value from SAML.

After you have provided the information in each field, click **Save Changes**.

Your users will then begin to be created or updated automatically upon their SSO login attempts. If any of the supplied fields are incorrect or don't match, the corresponding information will be blank in OPS-COM when the user logs in, or it will remain unchanged if the user already existed.

Login Source: Saml 2.0

Back

Settings | Metadata | Synchronization | Translations

✔ Auto Create/Update User

Field Mapping

Map the attributes from the Identity Provider to this service provider. In the example below, address_city and first_name are attributes supplied by the Identity Provider. OPS-COM will know to map that to the internal field name.

```
<saml:AttributeStatement>
    <saml:Attribute Name="first_name" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:basic">
        <saml:AttributeValue xsi:type="xs:string">John</saml:AttributeValue>
    </saml:Attribute>
    <saml:Attribute Name="last_name" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:basic">
        <saml:AttributeValue xsi:type="xs:string">Smith</saml:AttributeValue>
    </saml:Attribute>
    <saml:Attribute Name="address_city" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:basic">
        <saml:AttributeValue xsi:type="xs:string">Borden</saml:AttributeValue>
    </saml:Attribute>
    ...
```

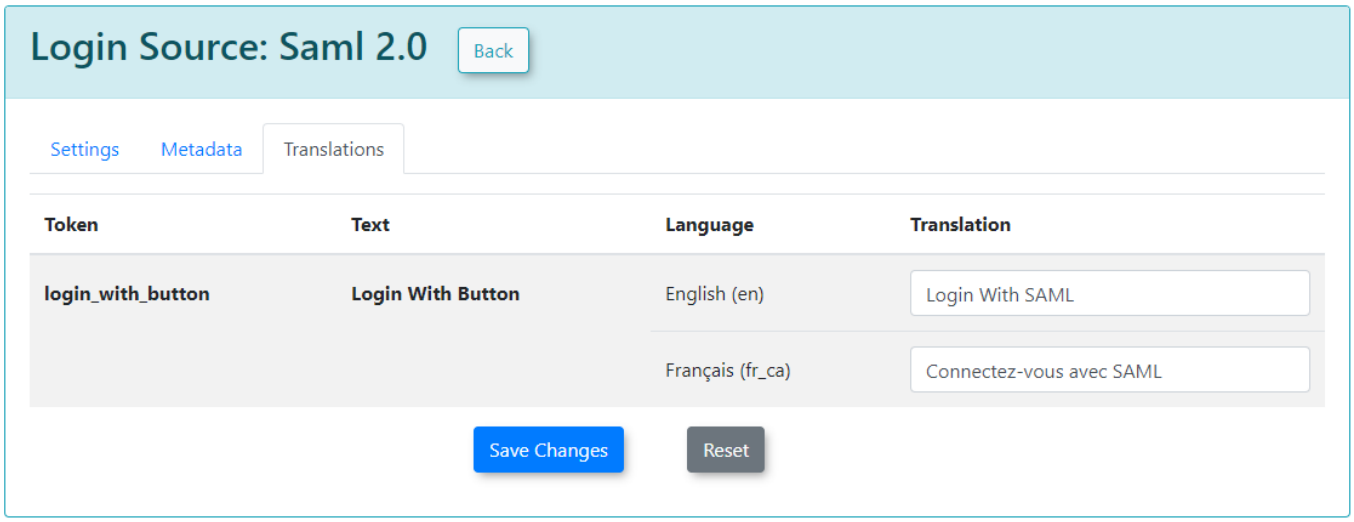
Enabled	enabled
User Type	user_type
First Name	full name
Middle Name	
Last Name	
Username	username
Email Address	email
Address	address_street
City	address_city
Province	address_province
Postal Code	address_postal
Phone Number	phone
profile.student_number	student_number
Employee Number	employee_number
Employer	employer
Building	building
Supervisor Name	supervisor_name
Supervisor Title	supervisor_title

Save Changes Reset

> The exact sample values from our test system may differ from your actual SAML system attributes.

Translations Tab

The **Translations** tab allows you to customize the text displayed on your login button from the user side. You can create as many different translations as are available in your system (e.g., English and French). This ensures that the SSO login experience is localized for your users.



Best Practices & Considerations

- **Coordinate with IT/SAML Administrator**: Successful SSO implementation requires close collaboration with your organization's IT department or the administrator of your SAML Identity Provider. They will provide the necessary metadata and attribute names.
- **Unique User Identifiers**: Ensure the **Unique Identifier** chosen for matching users is truly unique and persistent within your SSO system. Incorrect or changing identifiers will lead to duplicate accounts or login failures.
- **Attribute Mapping Accuracy**: Carefully map all desired user attributes from your Identity Provider to OPS-COM. Inaccurate mapping will result in missing or incorrect user data.
- **Test Thoroughly**: After initial configuration, conduct thorough testing with various user types and scenarios to ensure seamless login, proper user creation/updates, and correct data synchronization.
- **User Experience**: Clearly communicate the new SSO login process to your users. Provide instructions on how to access OPS-COM via SSO and address any potential questions.
- **Error Handling**: Be prepared to troubleshoot potential issues. Common problems include incorrect Entity IDs, expired certificates, or mismatched attribute names. The SSO system logs can be invaluable for diagnosing such issues.

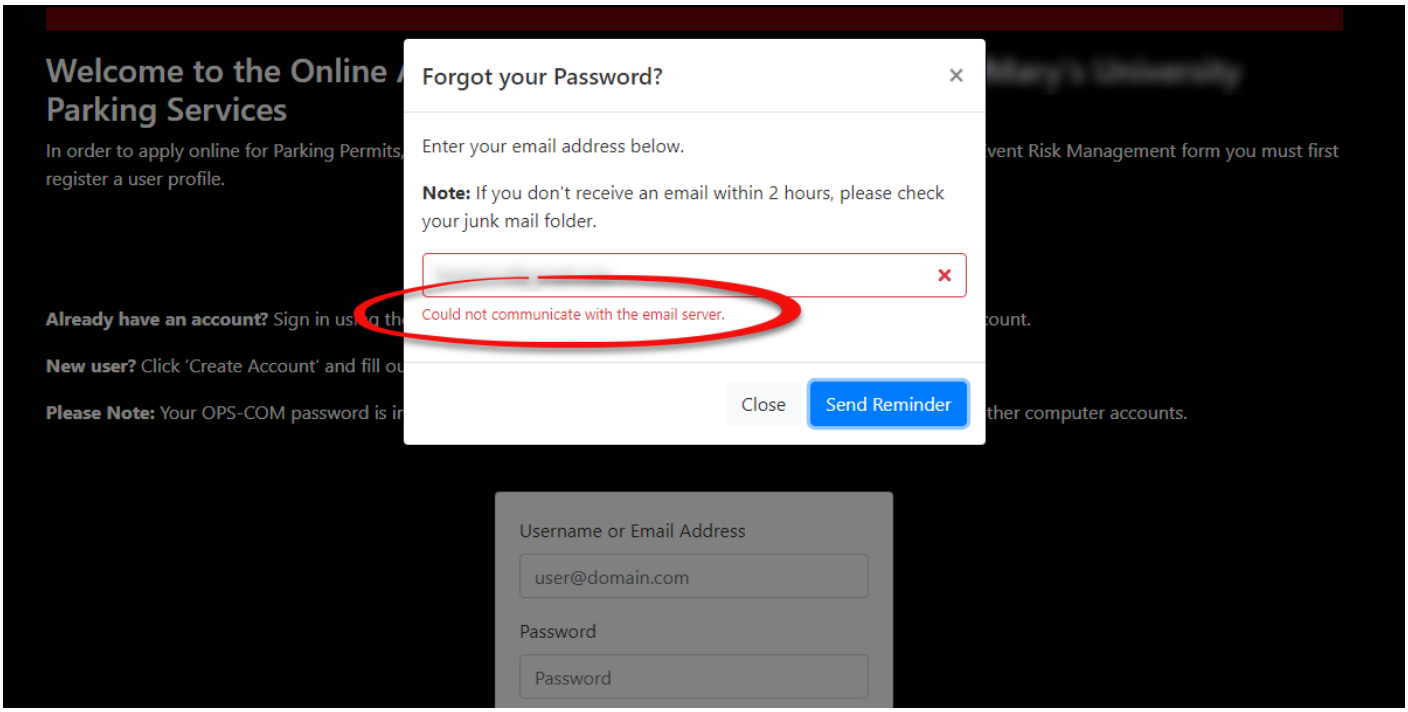
Troubleshooting - Email Server Communication Errors

> Communication errors in OPS-COM, often manifested as "Communication Error" messages to users, typically occur when essential email "From" or "Reply-to" addresses are not correctly configured in the system settings. This article helps OPS-COM administrators identify and resolve such errors, ensuring that system-generated emails (like password reset confirmations) are sent successfully.

Identifying a Communication Error

Communication errors are usually a symptom of missing or incorrect email configurations within the system settings.

A common example of a communication error occurs when a user attempts to submit the **Forgot Password** form on the user-side login page. An error message similar to the following may be displayed:



This error indicates that the system is attempting to send an email but lacks a defined "From" or "Reply-to" address from which to send it.

Fixing Communication Errors

1. Click **System Configuration,** then **System Settings.**
2. On the **General System Settings** tab, locate the **Reply-to Admin Email Address** field.
3. Enter a valid and active email address into this field. This address will serve as the system's "From" address for various automated communications.
4. **Save** your changes.

Once you have updated the email address, this communication issue should be resolved. You can test by re-attempting the action that previously triggered the error (e.g., submitting the **Forgot Password** form).

Best Practices & Considerations

- **Crucial Email Fields**: The **Reply-to Admin Email Address** (and other "From Email" settings found in **System Settings**) are critical for all system-generated email communications. Ensure they are always populated with a valid, monitored email address.
- **Troubleshooting**: If the issue persists after updating the **Reply-to Admin Email Address**, it may indicate a more complex underlying problem.
  - **Contact Support**: If the issue is still not resolved after completing these fields, please contact support@ops-com.com for further assistance. Provide details of the

error message and the steps you have already taken.

- **Monitoring System Notifications**: Regularly check the email address configured as the "Default Notification Email" in **System Settings** to catch any internal system alerts about failed communications.

Alarms System Settings

Setting up Alarms in OPS-COM

There are 3 different alarms you can setup in OPS-COM.

- People
- Vehicle/Plate
- Generic

Defining User Profile Settings

> User Profile Settings in OPS-COM enable administrators to customize the information collected from users on their profile forms. By controlling the visibility and requirement status of various fields, you can tailor the user experience to your organizational needs, ensure necessary data is captured, and streamline the registration process.

User profile item settings are configured within the **System Settings** area of OPS-COM.

1. Click **System Configuration**, then **System Settings**.
2. On the **Manage System Settings** screen, click **User Profile.**

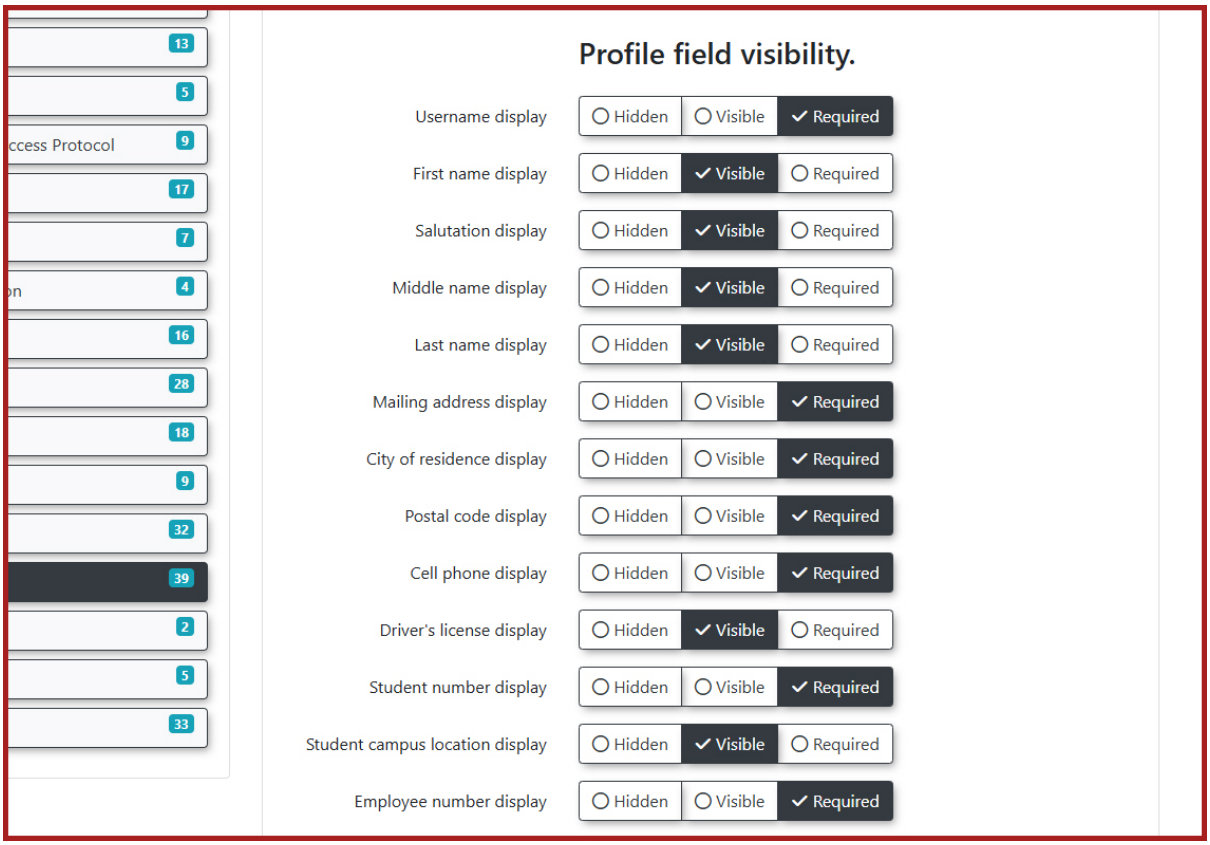
Using this Feature

On the **User Profile** settings page, items in the list can be set to one of three states, controlling their appearance and requirement on the user-side profile form:

- **Hidden**: The field is **not visible** on the user-side profile form.
- **Visible**: The field is seen on the user-side form, but entering information in it is **optional**.
- **Required**: The field is seen on the user-side form and is **mandatory**. Required fields are indicated by a **red asterisk** (*).

> **Note:** The system will not allow the user to save their profile if any required information is missing.

The state selected for each field is highlighted in black, with a checkmark indicating the active selection.



View the corresponding profile fields here.

Customizing Profile Sections

- **User Name**: A User Name is essential as it's one of the unique identifiers for system access. While a bare minimum typically includes Username, First Name, and Last Name, any of these items can be toggled on/off based on your needs.
  - > **Note:** One scenario where you might hide Username is if an external source (such as LDAP) is supplying the username.
  - **Enabling Username Edits**: By default, the username field is not editable by administrators. To enable this functionality, you must contact support@ops-com.com to request changes to the **Allow Username Edits** setting located within the **User Profile** settings list. Once activated by the OPS-COM Team, administrators will have the ability to edit usernames directly.
- **Address Information**: This section is critical if you plan to mail permits or other correspondence to end-users.
- **Phone Information**: Allows for the collection of various phone numbers.
- **License Information**: This field specifically refers to Driver's License number (not plate number). You may opt to record this information, especially if you are connected with local law enforcement.
- **Student Information**: Fields relevant to student identification (e.g., Student Number, Max/Min Student Number Digits).

- **Employee Information**: Fields relevant to employee identification (e.g., Employee Number, Max Employee Number Digits).

Considerations for Text2ParkMe Users

If your organization is using **Text2ParkMe**, a second tab will be available on this page. This tab allows you to configure additional details, including credit card information, that end-users can enter.

> **Important:** If any credit card information is entered by the user, it automatically switches all other credit card information fields to "required" for that transaction.

Best Practices & Considerations

- **Balance Data Collection and User Experience**: While it's important to collect necessary data, avoid making too many fields "Required" as this can create friction and deter users from completing their profiles. Prioritize truly essential information.
- **Understand System Overrides**: Even if you hide everything possible, the system might still require certain fundamental pieces of information (e.g., core identifiers like Username or Email) and will override your settings to ensure basic functionality.
- **Review Hidden Fields Periodically**: Ensure that fields marked "Hidden" truly remain irrelevant to your current processes. Organizational needs can change, making previously hidden data suddenly important.
- **Tailor to User Types**: Consider which information is truly necessary for different user types (e.g., students versus employees) and configure accordingly.
- **Impact on Mailing/Enforcement**: If you rely on mailing permits or recording specific ID numbers for enforcement, ensure the corresponding profile fields are set to "Required" or at least "Visible."
- The **Allow Username Edits** setting can only be toggled by the OPS-COM Support team. If you require the ability to edit usernames, please contact OPS-COM support to request this change.

Account Creation Preferences

> OPS-COM allows administrators to configure user account creation preferences, choosing between immediate auto-login or requiring email verification upon registration. Understanding and setting this preference is crucial for managing your user base effectively, balancing user convenience with security and data integrity needs.

Setup & Configuration

Account creation preferences are configured within the **User Profile** settings under **System Settings**.

1. Hover over **System Configuration,** then **System Settings,** and click the **User Profile** tab.
2. Toggle the **Auto Login After Register** setting, which controls the account creation flow.

Using this Feature

The **Auto Login After Register** setting has two states, each with distinct implications for user experience and system security:

Immediate Login (Auto Login After Register: ON)

- **Configuration**: Toggle the **Auto Login After Register** setting to **ON**.
- **Behavior**: This method allows users to instantly access their account immediately upon completing registration, without requiring them to verify their email address.
- **Reasons to Use**:
  - **Limited Email Access**: Ideal for scenarios where users might not have immediate access to their email, such as in kiosk setups or for individuals without constant mobile email access.
  - **Reduced Friction**: Provides a smoother, quicker onboarding experience, especially if your target audience is less tech-savvy or if you aim to minimize any barriers to entry.

Email Verification (Auto Login After Register: OFF)

- **Configuration**: Toggle the **Auto Login After Register** setting to **OFF**.
- **Behavior**: This method requires users to click a unique verification link sent to their registered email address before they can fully access their account.
- **Reasons to Use**:
  - **Verifying Legitimate Users**: This is generally the **preferred method** as it immediately confirms that the registration originates from a real user with a valid email address, significantly reducing bot registrations or fake accounts.
  - **Account Security and Recovery**: Email verification establishes a reliable communication channel crucial for secure password resets, account recovery procedures, and sending important notifications, thereby enhancing overall account security.
  - **Maintaining Data Integrity**: By ensuring valid email addresses from the outset, you improve the quality and accuracy of your user data in the system.

Best Practices & Considerations

- **Balance Security and Convenience**: Carefully weigh the trade-offs between user convenience (immediate login) and enhanced security/data integrity (email verification) based on your organization's risk tolerance and user base.
- **Communication**: Clearly inform users about the account creation process, especially if email verification is required. Provide instructions on checking spam folders for verification emails.
- **Email Deliverability**: If using email verification, ensure your system's email sending configurations are robust to guarantee that verification emails are delivered promptly and reliably.
- **Target Audience Analysis**: Consider the technical literacy and typical access methods of your target audience when deciding on the preferred setting.
- **Compliance**: Some data privacy regulations may implicitly favor email verification as it contributes to better data quality and user consent verification.

Configuring Multi-Factor Authentication on the User Portal

> **Multi-Factor Authentication (MFA)** adds a crucial second layer of security to user accounts in OPS-COM, significantly enhancing protection against unauthorized access. Currently, the primary method implemented is the use of **one-time passwords (OTPs)** sent via email. This article outlines how administrators can configure MFA at the system level and how users interact with this enhanced security feature on their portal.

Setup & Configuration

Implementing MFA involves administrator-side configuration within System Settings and customizing the associated email template.

Admin Side Configuration

One-time passwords will not be available on the user portal until enabled within **System Settings**.

1. Hover over **System Configuration** and click **System Settings**.
2. On the **User Profile** tab, click **Enable Multi-Factor Authentication**.

> If this setting is not available for you to change, please have your primary Admin contact support@ops-com.com to have it turned on.

This is a **ternary setting**, meaning it has three different states, allowing for flexible control over MFA implementation:

- **Hidden**: The use of one-time passwords is **disabled** site-wide. Users will not see or be able to enable MFA.
- **Visible**: The use of one-time passwords is **enabled**, but it is left **optional** for individual users to decide if they want to enable it on their account.
- **Required**: The use of one-time passwords is **mandatory** for **all** users of the website.

- If MFA is set to **Required**, users who do not have it enabled on their account will be automatically redirected to the setup page upon their next login and will be required to set it up before they can access their account.

Email Template Configuration

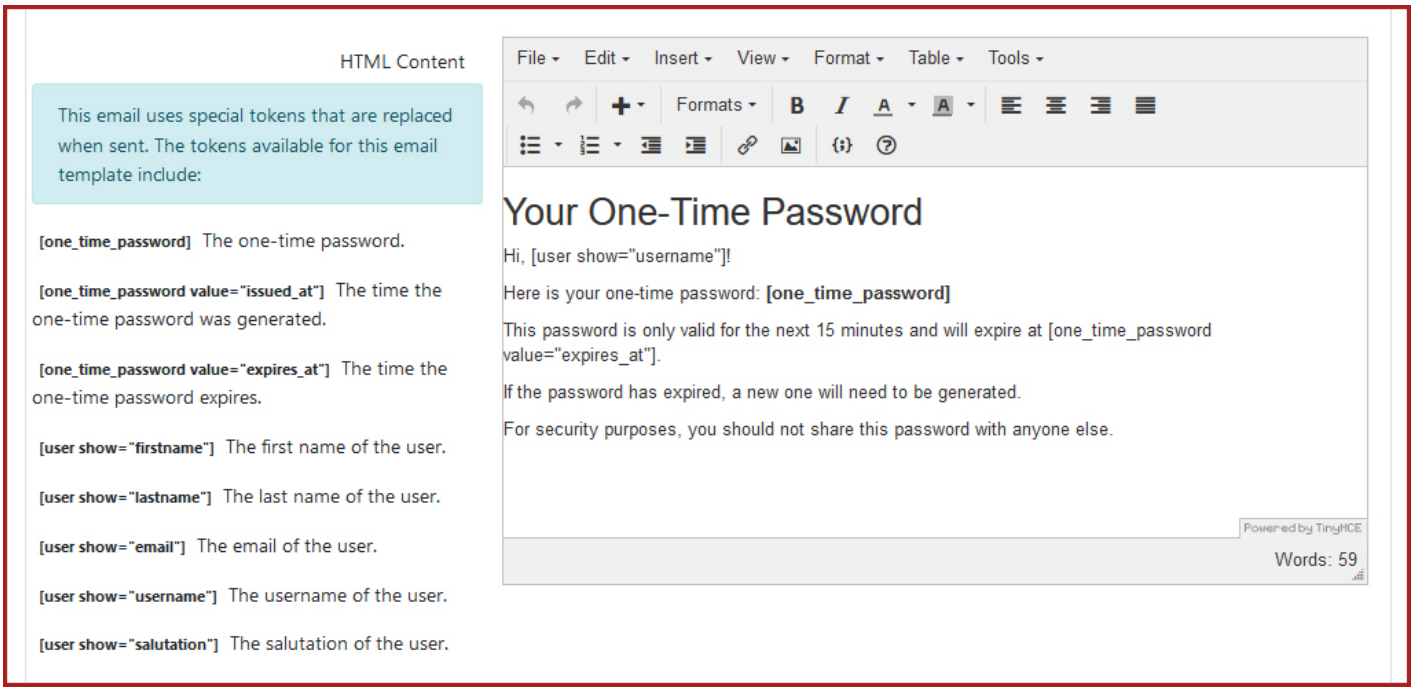
The content of the one-time password email sent to users is defined within a dedicated email template.

1. Hover over **System Configuration**, **Content & Designs** and click **Email Templates**.
2. Locate and edit the **One-Time Password Email Template**.

Here, administrators can define the message and branding of the email. In addition to general user-specific shortcodes, this template includes specific shortcodes for OTP details:

- `[one_time_password]`: Inserts the randomly generated one-time password.
- `[one_time_password value="issued_at"]`: Inserts the time the one-time password was generated.
- `[one_time_password value="expires_at"]`: Inserts the time the one-time password expires.

> One-time passwords always expire after **15 minutes**. This cannot be changed.



Using this Feature

User-Side MFA Management

Users can enable and manage their one-time password settings from their security page (formerly the passwords page). Refer to this wiki article to see the steps involved.

> The state of the user's one-time password verification is stored in the local storage of their session data. If the local storage is cleared (e.g., clearing browser cache), they will have to enter another one-time password. The MFA verification does not persist across different web browsers or devices, meaning the user will have to enter a new one-time password if they try to log in using another browser or device.

Best Practices & Considerations

- **Security Enhancement**: MFA significantly reduces the risk of unauthorized access, even if primary login credentials are compromised. It is highly recommended for all users.
- **Gradual Rollout (Visible vs. Required)**: When introducing MFA, consider starting with the **Visible** setting to allow users to opt-in voluntarily. Once accustomed, transition to **Required** for all users if your security policy mandates it.
- **Clear Communication**: Inform users about the MFA requirement, how to set it up, and how to log in using OTPs. Provide clear instructions and troubleshooting tips.
- **Email Deliverability**: Ensure that your system's email settings are correctly configured and that OTP emails are not being blocked by spam filters. Users need to receive these emails promptly to log in.
- **Template Customization**: Customize the OTP email template to include your organization's branding and any specific instructions for your users.
- **User Training**: Consider providing brief training or a guide for users on how to manage their MFA settings and log in with OTPs.
- **OTP Expiry**: Remind users that OTPs are time-sensitive (15 minutes) and that generating a new one invalidates previous ones.

Password and Security Settings

> The Security Settings in OPS-COM provide administrators with critical tools to enforce robust password policies and manage login security for all administrative accounts. Properly configuring these settings is essential for protecting sensitive system data, preventing unauthorized access, and complying with organizational security standards.

Security settings are managed within the **System Settings** area of OPS-COM.

1. Hover over **System Configuration,** and click **System Settings**.
2. Click **Security**. The **Manage System Settings** window will open, displaying all available security configurations.

Using this Feature

The **Security** component within **System Settings** allows administrators to configure various aspects of password management and account lockout policies.

Password Security Settings

- **Salted Password Hashing**:
  - **Purpose**: This setting adds an essential layer of security to stored passwords. Hashing is a one-way, irreversible process that converts a user's password into a unique, short hash value. "Salting" introduces a random string into this process, ensuring that even if two users have the same password, their stored hash values will be different. This prevents "lookup" (reverse engineering) of the original password, meaning forgotten passwords must be reset, not retrieved. This significantly limits an administrator's ability to view employee passwords and closes a critical security vulnerability.

> **Note**: Once **Hash and Salt** is enabled, it **should not be turned off**.

- **Require Password Update**:
  - **Purpose**: When activated, this setting forces users to change their passwords upon their next login.

    - **Use Case**: Ideal for ensuring compliance with regular password changes or after a password reset by an administrator.
- **Toggle Password Expiry**:
    - **Purpose**: By default, passwords in OPS-COM do not expire. For enhanced security, it is best practice to mandate regular password changes. This setting enables the use of password expiry.
    - **Configuration**: Toggle this setting **On**.
    - **Password Expiry in days**: Enter the number of days after which an administrator's password will expire, aligning with your organization's security policy (e.g., 90 days).
- **Enable Password History**:
    - **Purpose**: When toggled **On**, OPS-COM will remember passwords previously used by an administrator. The system will then prevent the reuse of those passwords for a specified period.
    - **Configuration**: Set **How long to remember old passwords** (in days) to define the duration for which old passwords are not allowed to be reused.

Password Strength Requirements

These settings allow you to enforce complexity rules for administrator passwords.

- **Minimum Password Length**: Sets the minimum number of characters required for a password.
- **Enable password strength requirements**: Toggles on or off the following specific complexity requirements:
    - **Numerical Characters**: Sets the minimum number of numbers required in the password.
    - **Lower Case Characters**: Sets the minimum number of lowercase characters required in the password.
    - **Upper Case Characters**: Sets the minimum number of uppercase characters required in the password.
    - **Non-Alpha Numeric**: Sets the minimum number of non-alphanumeric (special) characters required in the password (e.g., `!, &, #,` etc.).

Admin Account Lockout Settings

These settings provide an additional layer of security by locking an administrator out of their account after repeated incorrect password attempts.

- **Enable Admin Lockouts**: Toggles on or off the account lockout feature.
- **Lockout after X Attempts**: Sets the number of failed login attempts with an incorrect password before the system will lock out the administrator.
- **Login attempt timeframe**: Sets the timeframe (in minutes) during which incorrect login attempts are counted. For example, if an administrator fails 3 times within a 5-minute period, their account will be locked out.
- **Lock the admin out for X minutes**: Sets the duration (in minutes) that the administrator's account will remain locked. For example, setting it to `120 minutes` would

mean the administrator is locked out for 2 hours before another login attempt is permitted.

Best Practices & Considerations

- **Robust Security Policy**: Always implement a robust security policy that combines strong password requirements (length, complexity), password expiry, and lockout mechanisms.
- **Enable Hashing**: Ensure **Salted Password Hashing** is always enabled for maximum password security.
- **Regular Password Expiry**: Enforce regular password expiry (e.g., every 90 days) to mitigate the risk of compromised credentials.
- **Meaningful Lockout Settings**: Configure lockout settings to balance security with user convenience. Too aggressive settings can lead to frequent lockouts, while too lenient settings can be a security risk.
- **Communication**: Inform administrators about the security policies in place, including password strength requirements, expiry rules, and lockout procedures. This helps them comply and understand why they might be locked out.
- **Admins can see, only OPS-COM Team can change**: Several security settings (e.g., **Hash and Salt**, **Require Password Update**, **Toggle Password Expiry**, **Enable Password History**, **Enable password strength requirements**, **Enable Admin Lockouts**) are visible to administrators but can only be changed by the OPS-COM Team.

  For modifications to these specific settings, contact OPS-COM Support.

Uploading and Managing Files

> The **Manage Files** section in your OPS-COM Admin site provides a centralized repository for all files used across your OPS-COM instance, primarily images for your user and admin dashboards. This feature allows administrators to easily upload, view, organize, and manipulate these files, ensuring consistent branding and visual content throughout your system.

Using this Feature

The **Manage Files** page offers different ways to view and interact with your uploaded files.

1. To access this, click **System Configuration,** then **Manage Files.**

Viewing Files

Files can be viewed in two primary modes:

- **Grid Mode**: Displays a visual preview (thumbnail) of each image, which is useful for quickly identifying content.
- **Table Mode**: Provides a detailed list view, showing file names and other relevant information.

Managing Existing Files

By right-clicking on an image or file, a contextual menu will appear, giving you several options:

- **Download**: Save a copy of the file to your local device.
- **Rename**: Change the name of the file.
- **Delete**: Permanently remove the file from the system.

To crop an image (i.e., resize or adjust its visible dimensions to focus on a specific area), first **View** the image. Then, click the **Cropping icon** to begin selecting the desired area of the image.

Adding Files to the Repository

To add new files to this repository, click on the **Upload** tool. An **Upload files** interface will appear, prompting you to **Select files,** then **Submit**. The file will then be uploaded to your site's file storage.

> **Note:** Once an image is uploaded, it cannot be moved to a different folder. To maintain proper organization, ensure that you upload the image directly into the intended destination folder.

Adding Images to a Page (Referencing Uploaded Files)

To display an image you've uploaded onto an OPS-COM page (e.g., a page header or a custom content area):

1. Click **Insert** in the text editor toolbar.
2. Select **Image** from the dropdown menu.
3. In the **Insert/edit image** window, click the **search folder icon** next to the source field to open the **Filemanager**.
4. In the Filemanager, locate and **double-click** the image you wish to insert.
5. The image will be added to the page at the location of your cursor.

Best Practices & Considerations

- **Organize Files**: If you have many files, consider creating sub-folders within the file manager to keep your assets organized and easy to find. Be sure to upload files directly into the appropriate folder, as uploaded files cannot be moved once added.
- **Descriptive File Names**: Use clear and descriptive file names (e.g., `company-logo-header.png` instead of `image1.png`) to simplify identification and referencing.
- **Optimize Image Sizes**: Before uploading, optimize large images for web use. Smaller file sizes will improve page load times for both admin and user interfaces.
- **Backup Critical Assets**: While OPS-COM manages these files, consider maintaining local backups of critical branding assets (logos, banners) as a best practice.
- **Path Accuracy**: When linking images to pages, ensure the URL path is exactly correct, including capitalization, as file paths are often case-sensitive.

Dashboard Layouts and Custom Items

> OPS-COM's System Dashboard provides administrators with a customizable interface for quick access to key statistics and information through various widgets. This article focuses on how administrators can create and manage custom dashboard layouts, custom items, arrange widgets, and configure permissions to tailor the dashboard view for individual users or set system-wide defaults.

Setup & Configuration

Configuring dashboard layouts requires specific administrative permissions.

Assigning Dashboard Permissions

1. Click **System Configuration,** then **Admin Management,** and click **Manage Roles.**
2. On the **Manage Administrator Roles** screen, click the **Permissions** button next to the role you want to configure. The **Editing Permissions** screen will display.
3. Click **System Configuration**.
4. Ensure the following checkboxes are enabled:
   - **Change Dashboards**: Allows the administrator to view and select different active dashboard layouts.
   - **Edit Dashboard Layouts**: Allows the administrator to create, modify, and delete dashboard layouts.
   - **Edit Dashboard Items**: Allows the administrator to create and manage custom dashboard widgets (items).
5. Click **Save Permissions**.

Using this Feature

The Dashboard provides a flexible interface for administrators to arrange widgets according to their needs. To learn more about using this feature, refer to this wiki page.

Managing Layouts

Admins with the **Edit Dashboard Layouts** and **Edit Dashboard Items** permissions can manage layouts and widgets.

1. Click **System Configuration,** then **Dashboard,** and click **Dashboard Layouts.**

- **Dashboard Layouts**: This section allows you to manage or create new dashboard layouts.
- **Custom Dashboard Items**: This section is where you create and manage your own custom widgets.

Within **Dashboard Layouts**, you can perform the following actions:

- **New Layout**: Click to create a completely new dashboard layout from scratch.
- **Dashboard Items**: Click this to navigate directly to the **Custom Dashboard Items** page for widget creation.
- **Make System Default**: This option allows you to set a particular layout as the default for all users who do not have a custom dashboard assigned to their account.
- **Edit**: Click this next to an existing layout to modify its configuration.

Creating and Editing Dashboard Layouts

1. Click **New Layout** (to start fresh) or **Edit** an existing layout (to modify it). This will open the layout editor interface.
2. **Drag and drop widgets** from the **Widget Storage** sidebar onto the main dashboard area.
3. Once on the dashboard, you can **resize** and **arrange** widgets as needed to create your desired layout.
4. Widgets that are left in the **Widget Storage** sidebar will **not** appear on the dashboard. Any new widgets (whether system-created or custom-created) will initially appear in this **Widget Storage** sidebar, awaiting placement.
5. Remember to **Save** your layout changes.

Creating Custom Dashboard Items (Widgets)

Custom dashboard items allow you to add unique content widgets to your dashboards.

1. Click **System Configuration,** then **Dashboard,** and click **Custom Dashboard Items**.
2. Click **Create New** (to create a new custom widget) or **Edit** (to modify an existing one). This will open the item editor.
3. Fill in the following fields:
   - **Title**: This will be the header of your widget as it appears on the dashboard.
   - **Identifier**: A unique ID used for identifying the widget internally.
   - **Language**: Allows you to create different language versions of the widget content for multi-language sites.
   - **Content**: Add text, images, HTML, or other elements that will make up the body of your widget.

4. **Save Changes.**
5. Once saved, your custom widget will appear in the **Widget Storage** sidebar on the **Dashboard Layouts** page. You can then drag it onto a dashboard layout and save the layout.

Types of Widgets

The dashboard supports different types of widgets, identified by their color coding in the Widget Storage:

- **Blue**: Represent **System widgets** that display key administrative information and functionalities.
- **Yellow**: Represent **Stats widgets** designed to show site statistics and graphical data.
- **Green**: Represent **Custom widgets** that you create yourself through the **Custom Dashboard Items** section.

Best Practices & Considerations

- **User-Centric Design**: When creating layouts, consider the needs of the administrators who will be using them. Group related widgets together and prioritize frequently accessed information.
- **Default Layouts**: Utilize the **Make System Default** option to provide a consistent and functional starting dashboard for all new administrators or those who haven't customized their own.
- **Custom Widgets for Specific Information**: Use custom widgets to display important announcements, links to internal resources, or custom reports not available through standard widgets.
- **Permissions Management**: Ensure administrators have the correct permissions. "Change Dashboards" for viewing, "Edit Dashboard Layouts" for arranging, and "Edit Dashboard Items" for creating custom content.
- **Testing Layouts**: Always test new or modified dashboard layouts to ensure all widgets display correctly and serve their intended purpose.