

Sterling Selling and Fulfillment Foundation



# Password Policy Management

*Version 9.1*



Sterling Selling and Fulfillment Foundation



# Password Policy Management

*Version 9.1*

**Note**

Before using this information and the product it supports, read the information in "Notices" on page 25.

**Copyright**

This edition applies to the 9.1 Version of IBM Sterling Selling and Fulfillment Foundation and to all subsequent releases and modifications until otherwise indicated in new editions.

© **Copyright IBM Corporation 1999, 2011.**

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

---

# Contents

## Chapter 1. Password Policy Overview . . . 1

About Password Policy . . . . .	1
Guidelines for Creating Passwords . . . . .	1
Guidelines for Controlling User Access and Login Attempts . . . . .	1
Guidelines for Resetting Passwords . . . . .	2
Guidelines for Defining Rules . . . . .	2

## Chapter 2. Configuring Rules for Password Policy . . . . . 3

Rule Configuration . . . . .	3
Rule Definition Interfaces . . . . .	3

## Chapter 3. Generating Passwords . . . . . 5

About Generating Passwords . . . . .	5
Generating Random Passwords . . . . .	5
Generating Custom Passwords . . . . .	5

## Chapter 4. Resetting Passwords . . . . . 7

About Resetting Passwords . . . . .	7
Configuring Secret Questions . . . . .	7
Resetting Passwords Through E-mail . . . . .	7
Changing User Passwords . . . . .	8
Resetting Passwords through Other Protocols . . . . .	8

## Chapter 5. Controlling User Access and Login Attempts . . . . . 9

About Controlling User Access and Login Attempts . . . . .	9
Controlling Invalid Login Attempts by a User . . . . .	9
Controlling Incorrect Answers by a User . . . . .	9
Blocking an IP Address After Invalid Login Attempts . . . . .	9
Extracting an IP Address . . . . .	10
Passing the IP Address to the Interface . . . . .	10

## Chapter 6. Configuring a Password Policy . . . . . 11

About Configuring a Password Policy . . . . .	11
Defining Rules for a Password Policy . . . . .	11
Create an Assignment Rule . . . . .	12
Create a Login Rule . . . . .	13
Create a Password Change Rule . . . . .	14
Creating a Password Reset Rule . . . . .	15
Configure a Rule to Send a Confirmation to Users on Password Reset . . . . .	16
Create a Secret Answer Rule . . . . .	16
Modify a Rule Definition . . . . .	17
Modify a Rule Parameter Definition . . . . .	18
Modify a Parameter Value . . . . .	18
Delete a Rule . . . . .	18
Delete a Rule Parameter Definition . . . . .	19
Configure Password Policy Parameters . . . . .	19
Delete a Password Policy . . . . .	20
Delete a Password Policy Parameter . . . . .	20
Configuring Questions . . . . .	21
Create a Question . . . . .	21
Modify a Question . . . . .	21
Delete a Question . . . . .	22

## Chapter 7. Assigning a Password Policy . . . . . 23

Assigning a Password Policy . . . . .	23
Assigning a Password Policy to a User . . . . .	23
Assigning a Password Policy to a User Group . . . . .	23
Assigning a Password Policy to an Enterprise . . . . .	24

## Notices . . . . . 25



---

## Chapter 1. Password Policy Overview

---

### About Password Policy

The application provides an in-built and flexible password policy management for controlling password use and behavior. A password policy is a set of rules to define, control and manage user passwords. You can configure your own rules for the password policy, as applicable. The password policy is set at the organization or enterprise level.

The password policy broadly governs the following password characteristics:

- Password strength—Password strength controls the length of the password (minimum and maximum length), special characters in the password and password reuse.
- Password generation—Password generation controls generating a password during user creation, frequency of password expiration, failed login attempts and user roles that may affect the password policy.
- Password reset—Password reset controls resetting of the password through different protocols such as e-mail, SMS or any other.
- User authentication—User authentication includes authenticating users by using secret questions and answers whenever a user password is reset or changed.

Additionally, password policy configuration can be used to deny access to users in case of repeated invalid login attempts.

**Note:** If LDAP is used, the password management must be handled externally.

---

### Guidelines for Creating Passwords

A password can either be provided by the application or randomly generated by the system.

You can configure the password policy for:

- Stipulating minimum and maximum lengths for a password.
- Password History—Store recently used passwords (including invalid passwords) to prevent a user from entering the same password again. The number of recently used passwords can be configured.
- Password Strength—Ability to enforce usage of special characters for a strong password generation.

---

### Guidelines for Controlling User Access and Login Attempts

The password policy can control:

- User Access—Each user in an organization can be associated with a password policy.
- Login Attempts—Ability to deny access to users in case of repeated failure to login. Number of incorrect login attempts allowed can be configured.
- To prevent hacking, the IP address of users can be blocked after a configured number of invalid login attempts repeatedly from the same IP.

---

## Guidelines for Resetting Passwords

A password can be reset when a user forgets a password or requests a password change. The password policy defines the behavior in case of password resets.

You can reset passwords by configuring:

- Secret questions and answers—Questions can be configured in the password policy for user authentication and on success, password can be reset or changed. Repeated wrong answers lock users out. Secret questions act akin to passwords for handling user authentication.
- Using protocols such as e-mail, SMS or any other protocol. You can configure any protocol as applicable.

An e-mail can be sent to users when:

- A password is changed or reset.
- An answer to a secret question is given or changed.

---

## Guidelines for Defining Rules

A password policy is governed by a set of rules, called rule definitions. Rule definitions must be configured to provide the actual values of rule parameters.

Each rule definition has an implementation class that validates the rule definition parameter values set in the configuration. You can add your own custom classes to implement the rules.

Each rule must belong to a rule type. Following predefined rule types are available in the system:

- Password policy assignment
- Login
- Password change
- Password reset
- Password secret answer

You cannot add or modify rule types. Each rule type has an associated interface, which is implemented by the class you specify.



---

## Chapter 2. Configuring Rules for Password Policy

---

### Rule Configuration

A rule configuration consists of following:

- Rule definition
- Parameter Definition
- Parameter Configuration

#### Rule Definition

Each rule definition is identified by a unique indicative name. For example, PWD\_RULE\_DEFN.

A rule definition must include a class name for implementation. For example, com.yantra.ycp.passwordpolicy.defaultimpl.passwordchange.YCPPasswordHistValidator

#### Parameter Definition

Each rule can have any number of parameters. For example, you can define the following parameters for the rule, PWD\_RULE\_DEFN:

- MIN\_LEN and MAX\_LEN—They determine the maximum and minimum lengths of a given password.
- NUM\_SPL—Password must contain at least NUM\_SPL special characters.
- NUM\_ATTEMPTS—After NUM\_ATTEMPTS of invalid login attempts, user is locked out.
- DIC\_WORD—Password must not be based on a dictionary word (password history).

#### Parameter Configuration

You must configure these parameters with actual values for the password policy. For example, MIN\_LEN=8, NUM\_SPL=3 and so forth.

You can configure any number of rules using the Applications Manager.

---

### Rule Definition Interfaces

Each rule type has a corresponding interface associated with it. Classes specified for each rule definition implement the related interface.

The following sets of interfaces are provided:

- Application interfaces
- Custom interfaces

You can use either of these interfaces as applicable for implementing the password policy.

## Application Interfaces

The following rule definition application interfaces (corresponding to each rule type) are provided by the application:

- `IPasswordPolicyForAssignment`—This interface must be invoked when a policy is first assigned to the user.
- `IPasswordPolicyForPasswordChange`—This interface must be invoked when a password is changed.
- `IPasswordPolicyForLogin`—This interface must be invoked during login to check for login attempts.
- `IPasswordPolicyForSecretAnswers`—This interface must be invoked before login for validating answers to secret questions.
- `IPasswordPolicyForReset`—This interface must be invoked when there is a password reset request.

## Custom Interfaces

For every rule definition application interface, a corresponding custom interface is also provided. You can use your own class for implementing these interfaces.

An API template, `getUserForPasswordPolicy.xml` controls the `input.xml` for all implementations:

- `IPasswordPolicyDOM`
- `IPasswordPolicyForPasswordChangeDOM`
- `IPasswordPolicyForLoginDOM`
- `IPasswordPolicyForSecretAnswersDOM`
- `IPasswordPolicyForResetDOM`

---

## Chapter 3. Generating Passwords

---

### About Generating Passwords

There are two ways of creating user passwords in the system:

- System generates a random password
- You can generate a custom password (A user exit YCPGeneratePasswordUE is provided for this purpose).

The password policy controls the following parameters:

- Generating a password during user creation
- Password length
- Frequency of password expiration
- Number of login attempts
- Checking for commonly used passwords
- Ability to store recently used passwords

The above parameters are configured as rules.

---

### Generating Random Passwords

#### About this task

To generate random passwords for each user during user creation:

#### Procedure

- An attribute GeneratePassword is provided in the createUserHierarchy API.
- Set the attribute to true to generate a random password for the user. A random 8-digit password is generated using a mix of characters, special characters and numbers.

#### Results

Once generated, this password is available in the reset password event.

---

### Generating Custom Passwords

#### About this task

You can use the YCPGeneratePasswordUE user exit to generate custom passwords. A custom password can be generated when you require longer passwords, more number of special characters in the password or, any other. The user exit is implemented only if the attribute GeneratePassword in createUserHierarchy API is set to true. If this flag is set to true and the user exit is also implemented, the system generates a custom password.

If the user exit is not found but GeneratePassword is set to true the system generates a random password.

```
package com.yantra.ycp.japi.ue;
public interface YCPGeneratePasswordUE {
Document generatePassword(YFSEnvironment env, Document inXML);
}
```

inXML is the input XML template containing a list of required parameters to be passed to the user exit. The user exit API generates a password based on the parameters and stores the result in <UserGeneratedPassword="..." />.

When a user password changes or is reset, an event RESET\_PASSWORD.ON\_SUCCESS is triggered. You can configure this event to send an e-mail to users when there is a change in the password or when a random password is generated.

---

## Chapter 4. Resetting Passwords

---

### About Resetting Passwords

Passwords can be reset following password expiry, login failure due to invalid login attempts or in the case of a forgotten password. The password can be reset only if the password policy of a user allows it.

Resetting a password requires user authentication, which is done using secret questions and answers. You can configure the questions as required.

An event RESET\_PASSWORD.ON\_SUCCESS is triggered whenever a random password is generated or a password is changed.

You can configure the system to send an e-mail once the password is changed. Any other protocol such as SMS can also be configured.

---

### Configuring Secret Questions

#### About this task

A secret question and answer pair can be used for effective user authentication. Each secret question is configured and set at the organization level. Application access is denied to users in case of repeated incorrect answers. Custom questions pertaining to a user can also be configured.

Answers are encrypted before they are stored.

---

### Resetting Passwords Through E-mail

#### About this task

Use the rule type=password reset and the associated IPasswordPolicyForReset interface to reset passwords.

To reset a password:

- Invoke the API requestPasswordReset with the attribute ResetType as Email:

```
<ResetPassword UserKey="" ResetType="Email">
  <User Loginid=""/>
</ResetPassword>
```
- The ResetType is passed to the allowPasswordReset() method of IPasswordPolicyForReset interface. If the ResetType is blank, password reset is not allowed. The API generates a random character set and stores it as RequestId. The RequestId is passed to RESET\_PASSWORD.ON\_REQUEST event. The event RESET\_PASSWORD.ON\_REQUEST is triggered once a password reset is requested. If the attribute ResetType is set to Email for the given user ID, then the RequestId is sent to the user through e-mail.
- The changePassword API must be called to reset or change the actual password.

---

## Changing User Passwords

### About this task

Users are allowed to change their passwords once their password expires, which is determined by the password policy configuration.

To change the password:

### Procedure

- An API, `changePassword` is provided to change the actual password:

```
<User UserKey="" Loginid="" >  
  <ResetPassword RequestId="" ResetType=""/>  
</User>
```

The attribute `RequestId` is a randomly generated string that stores the password reset request ID, which is passed to the `changePassword` API to authenticate the request.

- A combination of `RequestId` and `ResetType` is used for request validation, and on successful validation, user password is changed.

---

## Resetting Passwords through Other Protocols

### About this task

You can reset passwords by configuring other protocols such as SMS. You must configure a new password policy to allow password reset through the new protocol.

To enable this:

### Procedure

1. Configure the `RESET_PASSWORD.ON_REQUEST` event to send a message.
2. Set the `ResetType` to an appropriate protocol. For example, `ResetType=SMS`.
3. Call the `requestPasswordReset` API with `ResetType=SMS`.
4. The `RESET_PASSWORD.ON_REQUEST` event processes the request and sends the message to the user along with the request ID.
5. Call `changePassword` API with `ResetType=SMS` and the request ID as given in the message to change the password.

### Results

**Note:** A Password Request purge agent is provided to clear the `PLT_PWD_REQ` table. For details on purge agents, refer to the Sterling Selling and Fulfillment Foundation Configuration Guide.

---

## Chapter 5. Controlling User Access and Login Attempts

---

### About Controlling User Access and Login Attempts

The password policy enables you to control:

- Number of invalid login attempts by a user
- Number of incorrect answers to questions
- Number of invalid login attempts from a particular URL (IP address)

---

### Controlling Invalid Login Attempts by a User

The number of login attempts allowed for a user can be configured. Failed login attempts are stored in the table `PLT_USER_LOGIN_FAILED`. An attribute `FAILURE_TYPE` in the table is used to determine the type of login failure.

In case of login failure, `FAILURE_TYPE` is set to `LOGIN`.

**Note:** A User Login Failed purge agent is provided to clear the `PLT_USER_LOGIN_FAILED` table. For details on purge agents, refer to the Sterling Selling and Fulfillment Foundation Configuration Guide.

---

### Controlling Incorrect Answers by a User

By default, the system locks out a user after a number of failed answer attempts to secret questions.

In case of failed answers, the `FAILURE_TYPE` is set to `ANSWER` in the `PLT_USER_LOGIN_FAILED` table.

- The `IPasswordPolicyForSecretAnswers` interface associated with the rule `type = password secret answer` can be used to check for failed answers.
- Use the following parameters in the implementation class for validation. You can configure the values for these parameters in the password policy as required:
  - `MaxFailedAnswers`: Number of allowed incorrect answers within the interval specified in the attribute `CheckIntervalMinutes`.
  - `CheckIntervalMinutes`: Time interval (in minutes) allowed for incorrect answers, after which the answers are not validated and the user is locked out for this duration. For example, if the interval specified is 180 mins and the number of allowed wrong answers exceeds this limit, the user is locked out for 3 hours. The user can login again after the interval lapses.

---

### Blocking an IP Address After Invalid Login Attempts

#### About this task

In case of repeated login failure from a particular IP address, you can configure the system to block the IP address from accessing the application.

To block an IP, perform the following steps:

#### Procedure

1. Extract the IP address from the header

2. Pass the IP address to the interface associated with the rule type, password secret answer

## Extracting an IP Address

### About this task

To determine the authenticity of an IP address and whether it is using a proxy, a property is provided in `yfs.properties` file:

```
yfs.clientip.reader=<classname>
```

where `<classname>` is the name of the class that is used to read and store the client IP address from the request.

To extract an IP address, perform the following steps:

### Procedure

- You can provide your own client IP reader class. This property checks the proxy-specific headers and extracts the appropriate IP address as:  

```
yfs.clientip.httprequest.ipHeaderAttribute
```
- An interface `YFSClientIPReader` is provided, which must be implemented in the given class, `<classname>`:

```
public interface YFSClientIPReader{  
    String readClientIP(HttpServletRequest req);  
}
```

## Passing the IP Address to the Interface

### About this task

Perform the following steps to pass the extracted IP address to the interface:

### Procedure

- Use the `IPasswordPolicyForSecretAnswers` interface to check for failed answers from a particular IP.
- The IP address is available in the `onAnswerFailure()` method of this interface.
- Use the following parameters in the implementation class for validation. You can configure the values for these parameters in the password policy as required:
  - `MaxFailedLogins`—Number of allowed invalid password attempts within the interval specified in the attribute, `CheckIntervalMinutes`.
  - `CheckIntervalMinutes`—Time interval (in minutes) allowed for invalid login attempts, after which the IP is blocked for this duration.



---

## Chapter 6. Configuring a Password Policy

---

### About Configuring a Password Policy

You can set up and configure a password policy for your organization and associate it to user accounts in the organization from the Applications Manager. For details on using the Applications Manager, refer to the Sterling Selling and Fulfillment Foundation Configuration Guide.

**Note:** The password policy is set at the enterprise level. You can use policy configurations of any other enterprise through enterprise inheritance.

A password policy is a set of rule configurations. Each rule configuration provides values for the parameters of a rule.

Configuring a password policy broadly consists of the following functions:

- Defining rules.
- Configuring rule parameters for implementation and validation.
- Defining a password policy that implements the defined rules.
- Setting up secret questions for user authentication.

---

### Defining Rules for a Password Policy

You can define and configure rules for the password policy in the Application. Each rule has parameters for specifying one or more policy definitions (password change, password length or password reset).

Each rule you define must belong to one of the predefined rule types. The following table lists some of the common rules that you can create, the rule type they must belong to and the associated interface that must be implemented:

Rule	Rule Type	Interface
Login	Password Rule Type Login	IPasswordPolicyForLogin
Policy assignment	Password Rule Type Assignment	IPasswordPolicyForAssignment
Password length	Password Rule Type Password Change	IPasswordPolicyForPasswordChange
Password strength	Password Rule Type Password Change	IPasswordPolicyForPasswordChange
Password reset	Password Rule Type Password Reset	IPasswordPolicyForReset
Failed login attempts	Password Rule Type Login	IPasswordPolicyForLogin
User authentication	Password Rule Type Secret Answer	IPasswordPolicyForSecretAnswers

Each rule type has an associated interface, which is implemented by the class you specify when creating the rule.

You can define any number of rules for a given rule type and each rule definition can contain any number of rule parameters.

A rule is configured when you:

- Create the rule.
- Define the parameters for the rule.
- Configure the rule parameters by specifying values.

---


## Create an Assignment Rule

### About this task


Assignment rules are applied when a password policy is assigned to a user or an enterprise.

To create an assignment rule, perform the following steps:

### Procedure


1. Open the Applications Manager and select **Password Policy Management** → **Password Rule Definition**. The Password Rule Definition List window is displayed.
2. In the list window, click . The Rule Definition Details window is displayed.
3. Specify the following:

Field	Description
Rule Name	Specify a name that is indicative, for the new rule. There is no restriction as to the length, case and type.
Rule Description	Provide a description for the rule.
Class Name	Assignment rules must implement the <code>IPasswordPolicyForAssignment</code> interface.  Each rule definition has its own class for implementation. Specify the class to be used for validating the parameters provided in the policy configuration. This class must implement the interface associated with the specified rule type.  For example,  <code>com.yantra.ycp.passwordpolicy.defaultimpl.policyassign.YCPSecurityAssignValidator</code>
Rule Type	Select <b>Assignment</b> from the drop-down list. The interface associated with this rule is specified in <b>Class Name</b> .

4. After you define a rule, you must define the rule parameters. Click  in the **Rule Parameters** panel. A pop-up window is displayed.
5. Specify the following:

Field	Description
Name	Specify a parameter name that is indicative. The parameter name must be unique as this value is used in the class that validates the rule.
Description	Provide a description for the parameter.

Field	Description
Data Type	Specify the data type (number, string, long etc) of the parameter.

- Click **OK** to close the pop-up window.
- Repeat steps 5-6 to add more rule parameters, as applicable.
- Click  in the **Rule Definition Details** panel to save the rule definition details.
- Proceed with configuring the policy parameters.


## Create a Login Rule

### About this task


Login rules are applied when a user logs into the application. Login rules can be used to validate either the number of failed login attempts by a user or the number of failed login attempts from a specific URL. The login rule can also be used to check expiry of passwords.

To create a login rule, perform the following steps:

### Procedure


- Open the Applications Manager and select **Password Policy Management** → **Password Rule Definition**. The Password Rule Definition List window is displayed.
- In the list window, click . The Rule Definition Details window is displayed.
- Specify the following:

Field	Description
Rule Name	Specify a name that is indicative, for the new rule. There is no restriction as to the length, case and type.
Rule Description	Provide a description for the rule.
Class Name	<p>Login rules must implement the IPasswordPolicyForLogin interface.</p> <p>Each rule definition has its own class for implementation. Specify the class to be used for validating the parameters provided in the policy configuration. This class must implement the interface associated with the specified rule type.</p> <p>For example,</p> <pre>com.yantra.ycp.passwordpolicy.defaultimpl.login.YCPFailedLoginValidator</pre>
Rule Type	Select <b>Login</b> from the drop-down list. The interface associated with this rule is specified in <b>Class Name</b> .

- After you define a rule, you must define the rule parameters. Click  in the **Rule Parameters** panel. A pop-up window is displayed.
- Specify the following:

Field	Description
Name	Specify a parameter name that is indicative. The parameter name must be unique as this value is used in the class that validates the rule.

Field	Description
Description	Provide a description of the parameter.
Data Type	Specify the data type (number, string, long etc) of the parameter.

6. Click **OK** to close the pop-up window.
7. Repeat steps 5-6 to add more rule parameters, as applicable.
8. Click  in the **Rule Definition Details** panel to save the rule definition details.
9. Proceed with configuring the policy parameters.


## Create a Password Change Rule

### About this task


Password change rules are applied when a password is changed by a user. Password change rules can be used to validate password length, strength, and the history that tracks previously used passwords.

To create a password change rule, perform the following steps:


### Procedure

1. Open the Applications Manager and select **Password Policy Management** → **Password Rule Definition**. The Password Rule Definition List window is displayed.
2. In the list window, click . The Rule Definition Details window is displayed.
3. Specify the following:

Field	Description
Rule Name	Specify a name that is indicative, for the new rule. There is no restriction as to the length, case and type.
Rule Description	Provide a description for the rule.
Class Name	<p>Password change rules must implement the <code>IPasswordPolicyForPasswordChange</code> interface.</p> <p>Each rule definition has its own class for implementation. Specify the class to be used for validating the parameters provided in the policy configuration. This class must implement the interface associated with the specified rule type.</p> <p>For example,</p> <pre>com.yantra.ycp.passwordpolicy.defaultimpl.passwordchange.YCPasswordHistValidator</pre>
Rule Type	Select <b>Password Change</b> from the drop-down list. The interface associated with this rule is specified in <b>Class Name</b> .

4. After you define a rule, you must define the rule parameters. Click  in the Rule Parameters panel. A pop-up window is displayed.
5. Specify the following:

Field	Description
Name	Specify a parameter name that is indicative. The parameter name must be unique as this value is used in the class that validates the rule.
Description	Provide a description of the parameter.
Data Type	Specify the data type (number, string, long etc) of the parameter.

6. Click **OK** to close the pop-up window.
7. Repeat steps 5-6 to add more rule parameters, as applicable.
8. Click  in the **Rule Definition Details** panel to save the rule definition details.
9. Proceed with configuring the policy parameters.


## Creating a Password Reset Rule

### About this task


Password reset rules are applied when a password is reset due to invalid password entry or password expiry. Password reset rules can be used to allow password resets for a user or an enterprise, and modes of password resets, such as, e-mail, SMS or any other protocol.

To create a password reset rule, perform the following steps:


### Procedure

1. Open the Applications Manager and select **Password Policy Management** → **Password Rule Definition**. The Password Rule Definition List window is displayed.
2. In the list window, click . The Rule Definition Details window is displayed.
3. Specify the following:

Field	Description
Rule Name	Specify a name that is indicative, for the new rule. There is no restriction as to the length, case and type.
Rule Description	Provide a description for the rule.
Class Name	<p>Password reset rules must implement the <code>IPasswordPolicyForReset</code> interface.</p> <p>Each rule definition has its own class for implementation. Specify the class to be used for validating the parameters provided in the policy configuration. This class must implement the interface associated with the specified rule type.</p> <p>For example,</p> <pre>com.yantra.ycp.passwordpolicy.defaultimpl.passwordreset.YCPPasswordResetValidato</pre>
Rule Type	Select <b>Password Reset</b> from the drop-down list. The interface associated with this rule is specified in <b>Class Name</b> .

4. After you define a rule, you must define the rule parameters. Click  in the **Rule Parameters** panel. A pop-up window is displayed.
5. Specify the following:

Field	Description
Name	Specify a parameter name that is indicative. The parameter name must be unique as this value is used in the class that validates the rule.
Description	Provide a description of the parameter.
Data Type	Specify the data type (number, string, long etc) of the parameter.

6. Click **OK** to close the pop-up window.
7. Repeat steps 5-6 to add more rule parameters, as applicable.
8. Click  in the **Rule Definition Details** panel to save the rule definition details.
9. Proceed with configuring the policy parameters.

## Configure a Rule to Send a Confirmation to Users on Password Reset

This topic applies to Foundation only.

### About this task


You can define a password policy rule to send a confirmation to the user when the user's password is reset.

Perform the following steps:

### Procedure

1. Open the Applications Manager and select **Password Policy Management** → **Password Policy Rules**. The Password Policy Rules window is displayed.
2. Select the **Confirmation Is Required On Password Reset** check box if you want to send a confirmation to the user when the user's password is reset.

Field	Description
Confirmation Is Required On Password Reset	Select this check box if you want to send confirmation to the user when the user's password is reset.

3. Click  to save the changes.

## Create a Secret Answer Rule

### About this task

Secret answer rules are applied during user authentication, whenever a password is changed or reset. Secret answer rules can be used to validate user answers against secret questions that are defined for the organization. Secret answer rules can also be used to validate and control the number of incorrect answers allowed to secret questions.


To create a secret answer rule, perform the following steps:

### Procedure


1. Open the Applications Manager and select **Password Policy Management** → **Password Rule Definition**. The Password Rule Definition List window is displayed.

- In the list window, click . The Rule Definition Details window is displayed.
- Specify the following:

Field	Description
Rule Name	Specify a name that is indicative, for the new rule. There is no restriction as to the length, case and type.
Rule Description	Provide a description for the rule.
Class Name	<p>Secret answer rules must implement the <code>IPasswordPolicyForSecretAnswers</code> interface.</p> <p>Each rule definition has its own class for implementation. Specify the class to be used for validating the parameters provided in the policy configuration. This class must implement the interface associated with the specified rule type.</p> <p>For example,</p> <pre>com.yantra.ycp.passwordpolicy.defaultimpl.secretanswer.YCPSecretAnswerIPValidation</pre>
Rule Type	Select <b>Password Secret Answer</b> from the drop-down list. The interface associated with this rule is specified in <b>Class Name</b> .

- After you define a rule, you must define the rule parameters. Click  in the **Rule Parameters** panel. A pop-up window is displayed.
- Specify the following:

Field	Description
Name	Specify a parameter name that is indicative. The parameter name must be unique as this value is used in the class that validates the rule.
Description	Provide a description of the parameter.
Data Type	Specify the data type (number, string, long etc) of the parameter.

- Click **OK** to close the pop-up window.
- Repeat steps 5-6 to add more rule parameters, as applicable.
- Click  in the **Rule Definition Details** panel to save the rule configuration details.
- Proceed with configuring the policy parameters.


## Modify a Rule Definition


### About this task

Once all the rules and parameters are created, they are displayed in the Password Rule Definition List panel.

To modify a rule:

### Procedure

- Open the Applications Manager and select **Password Policy Management** → **Password Rule Definition**. The **Password Rule Definition List** is displayed.
- In the list, select the rule to be modified. Click .

3. The Rule Definition Details window is displayed. Modify the details as required. You cannot modify the **Rule Name**.
4. Click  to save the details.



---

## Modify a Rule Parameter Definition

### About this task

To modify a rule parameter:

### Procedure

1. Open the Applications Manager and select **Password Policy Management** → **Password Rule Definition**. The **Password Rule Definition List** is displayed.
2. Select the rule. The Rule Definition Details window is displayed. The Rule Parameters panel displays all the parameters associated with the selected rule.
3. In the panel, select the parameter to be modified. Click .
4. The Rule Parameter Details pop-up window is displayed. Modify the details as applicable. You cannot modify the **Parameter Name**.
5. Click **OK** to close the window.
6. Click  to save the details.





---

## Modify a Parameter Value

### About this task

To modify a parameter value:

### Procedure

1. Open the Applications Manager and select **Password Policy Management** → **Password Policy**. The **Password Policy List** is displayed.
2. Select the policy and click . The Password Policy Details window is displayed, along with the associated parameters.
3. In the Policy Configurations panel, select the parameter to be modified. Click . The Configuration Details pop-up window is displayed along with the available configuration parameters.
4. In the Configuration Parameters panel, select the parameter to be modified. Click .
5. Modify the value as required. You cannot modify the **Parameter Name**. You can only change the parameter value.
6. Click **OK** to close the window.
7. Click  to save the details.

---


## Delete a Rule

### About this task

To delete a rule:



## Procedure

1. Open the Applications Manager and select **Password Policy Management** → **Password Rule Definition**. The Password Rule Definition List window is displayed.
2. Select the rule to be deleted and click  in the panel.


---

## Delete a Rule Parameter Definition

### About this task

To delete a parameter:

### Procedure

1. Open the Applications Manager and select **Password Policy Management** → **Password Rule Definition**. The Password Rule Definition List window is displayed.
2. Select the rule. The Rule Definition Details window is displayed. The **Rule Parameters** panel displays all the parameters associated with the selected rule.
3. Select the parameter to be deleted and click  .

---

## Configure Password Policy Parameters


### About this task

Creating the rule sets up the framework for rule parameters. Afterwards, you configure the password policy in order to specify the actual value of the parameters that are required by the rule. This process includes the following actions:

1. Define a password policy that governs the rule you want.
2. Select the rule type associated with the password policy.
3. Select the parameters for the rule.
4. Specify the parameter values.



To define a password policy and configure parameters, follow these steps:

### Procedure


1. Open the Applications Manager and select **Password Policy Management** → **Password Policy**. The Password Policy List window is displayed.
2. In the list window, click  . The Password Policy Details window is displayed.
3. Specify the following:

Field	Description
Policy Name	Specify a unique name for the password policy. This policy is associated to the user account. Once the name is specified, it cannot be modified.
Policy Description	Provide a brief description of the password policy.

Field	Description
Policy Status	Select the policy status from the drop-down list. The available options are: <ul style="list-style-type: none"> <li>• Disabled - If this option is set, the user account will not be validated against the rules defined in the policy during implementation.</li> <li>• Enabled - If this option is set, the user account will be associated with the password policy and the account will be validated against all the rules in the policy.</li> </ul>

4. Click  in the **Policy Configurations** panel. The Rule Parameter Details pop-up window is displayed.
5. Select the rule type you want for this policy. A list of rules that have been defined for this rule type is displayed in the **Rule Name** drop-down list.
6. Select the rule name for which the parameter values must be configured. A list of available parameters for this rule is displayed in the **Configuration Parameters** panel.
7. Select the parameter to be configured and click . The Parameter Details window is displayed.
8. Specify the following:

Field	Description
Parameter Name	The parameter name defined in the <b>Rule Parameters</b> panel is displayed here. You cannot modify the parameter name once specified.
Parameter Value	Specify the actual value for the parameter that drives the rule associated with it. This value is used by the rule definition class for validation.


9. Click **OK** to close the window.
10. Repeat steps 7-9 to configure values for other parameters associated with this rule type and name.
11. Click **OK** to close the Configuration Details window.
12. Click  in the **Password Policy Details** panel to save the rule configuration details.

## Delete a Password Policy

### About this task

To delete a password policy:

### Procedure

1. Open the Applications Manager and select **Password Policy Management** → **Password Policy**. The Password Policy List window is displayed.
2. Select the policy to be deleted and click .

## Delete a Password Policy Parameter

### About this task

To delete a password policy parameter:

## Procedure

1. Open the Applications Manager and select **Password Policy Management** → **Password Policy**. The Password Policy List window is displayed.
2. Select the policy. The Password Policy Details window is displayed.
3. In the **Policy Configurations** panel, select the parameter to be deleted and click



---

## Configuring Questions

The system allows you to define a set of questions for user authentication during a new password request following password expiry or a password reset. The questions are set at the organization or enterprise level.

You can define any number of questions for the organization.


---

## Create a Question


### About this task

To create a question:

### Procedure

1. Open the Applications Manager and select **Password Policy Management** > **Question**. The Authentication Question List window is displayed.
2. In the list, click . The Question Details window is displayed.
3. Specify the question and the sequence in which it should appear:

Field	Description
Question Text	Provide a question here. For example, Which car do you own?
Sequence	Specify the sequence in which the question must appear.

4. Repeat steps 1-4 to add more questions.
5. Click  to save the information.

---


## Modify a Question


### About this task

Once a question is specified, it cannot be modified. You can only change the sequence in which it appears.

To modify the sequence:

### Procedure

1. Open the Applications Manager and select **Password Policy Management** → **Question**. The Authentication Question List window is displayed.
2. Select the question to be modified and click . The Question Details window is displayed.
3. Modify the sequence as applicable.

4. Click  to save the details.


---

## Delete a Question

### About this task

To delete the question:

### Procedure

1. Open the Applications Manager and select **Password Policy Management** → **Question**. The Authentication Question List window is displayed.
2. Select the question to be deleted and click .

---

## Chapter 7. Assigning a Password Policy

---

### Assigning a Password Policy

A password policy can be assigned to each user or a group of users in an organization or could be set as a default at the organization or enterprise level. The password policy assigned to a user takes precedence.

Before you assign a password policy to a user, user group or an organization, you must first define and configure the password policy with rules for implementation.

**Note:** When a user's password policy is changed, the rules governing the policy take effect only when the user's password is subsequently reset or changed.

You can assign a password policy to a user, user group or an organization in the Applications Manager from the User Details, Group Details and Organization Details windows respectively.

---



### Assigning a Password Policy to a User

#### About this task

You can assign a password policy to a user from the **Primary Info** tab of the User Details window.

To assign a password policy to a user:

#### Procedure

1. Open the Applications Manager and select **Security** → **Users**. The User Search window is displayed.
2. Select the applicable search criteria and choose . The results are displayed in the **Search Results** panel.
3. Double-click a user from the **Search Results** panel. The User Details window for the selected user is displayed.
4. From the **Password Policy** drop-down list, select the password policy that you want to associate with the user.
5. Click  to save the changes.

#### What to do next

For additional information about the User Details window, refer to the Sterling Selling and Fulfillment Foundation Configuration Guide.

---

### Assigning a Password Policy to a User Group

#### About this task


Each user group can be assigned a password policy. You can assign the same password policy or different password policies to multiple user groups. However, you cannot assign multiple password policies per user group.

If a user belongs to multiple user groups and if all the user groups have conflicting rules governing their password policies, the policy with the highest priority for a user group is considered.

You can assign a password policy to a user group from the Group Details window.

To assign a password policy to a user group:

### Procedure

1. Open the Applications Manager and select **Security** → **Groups**. The Groups window is displayed.
2. Double-click a group from the Groups window. The Group Details window for the selected group is displayed.
3. From the **Password Policy** drop-down list, select the password policy that you want to associate with the user group.
4. Click  to save the changes.

### What to do next

For additional information about User Groups and Group Details window, refer to the Sterling Selling and Fulfillment Foundation.

---



## Assigning a Password Policy to an Enterprise

### About this task

You can assign a password policy to an enterprise from the **Roles & Participation** tab of the Organization Details window.

To assign a password policy to an enterprise:

### Procedure

1. Open the Applications Manager and select **Participant Modeling** → **Participant Setup**. The Organization Search window is displayed.
2. Select the applicable search criteria and choose . The results are displayed in the **Search Results** panel.
3. Double-click an organization from the **Search Results** panel. The Organization Details window for the selected organization is displayed.
4. In the **Roles & Participation** tab, select the **Enterprise Attributes** tab.  
Note: The **Enterprise Attributes** tab is displayed depending on the roles selected in the **Roles & Participation** tab.
5. From the **Password Policy** drop-down list, select the password policy that you want to associate with the enterprise.
6. Click  to save the changes.

### What to do next

For additional information about the Organization Details window, see the Sterling Selling and Fulfillment Foundation Configuration Guide.

---

## Notices

This information was developed for products and services offered in the U.S.A.

IBM® may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

*IBM Director of Licensing*

*IBM Corporation*

*North Castle Drive*

*Armonk, NY 10504-1785*

*U.S.A.*

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

*Intellectual Property Licensing*

*Legal and Intellectual Property Law*

*IBM Japan Ltd.*

*1623-14, Shimotsuruma, Yamato-shi*

*Kanagawa 242-8502 Japan*

**The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:**

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be

incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

*IBM Corporation*

*J46A/G4*

*555 Bailey Avenue*

*San Jose, CA 95141-1003*

*U.S.A.*

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.



This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

#### COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© IBM 2011. Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. 2011.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

#### **Trademarks**

IBM, the IBM logo, and [ibm.com](http://www.ibm.com)<sup>®</sup> are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at <http://www.ibm.com/legal/copytrade.shtml>.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency which is now part of the Office of Government Commerce.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.

Linear Tape-Open, LTO, the LTO Logo, Ultrium and the Ultrium Logo are trademarks of HP, IBM Corp. and Quantum in the U.S. and other countries.

Connect Control Center<sup>®</sup>, Connect:Direct<sup>®</sup>, Connect:Enterprise<sup>™</sup>, Gentran<sup>®</sup>, Gentran<sup>®</sup>:Basic<sup>®</sup>, Gentran:Control<sup>®</sup>, Gentran:Director<sup>®</sup>, Gentran:Plus<sup>®</sup>, Gentran:Realtime<sup>®</sup>, Gentran:Server<sup>®</sup>, Gentran:Viewpoint<sup>®</sup>, Sterling Commerce<sup>™</sup>, Sterling Information Broker<sup>®</sup>, and Sterling Integrator<sup>®</sup> are trademarks or registered trademarks of Sterling Commerce<sup>™</sup>, Inc., an IBM Company.

Other company, product, and service names may be trademarks or service marks of others.





Product Number:

Printed in USA