GenericUserMapper Plugin Configuration

The GenericUserMapper plugin can be configured for multiple authentication types. Each authentication type mapping should be configured in a separate options object.

Below is an example of the GenericUserMapper plugin configuration for multiple authentication types:

```
{
"Name": "Aras.OAuth.Server.Plugins.GenericUserMapper",
"Enabled": true,
"Options": [
{
"AuthenticationType": "<AuthenticationType1>",
"InnovatorUserNameFormat": "{<Claim1>}"
},
{
"AuthenticationType": "<AuthenticationType2>",
"InnovatorUserNameFormat": "{<Claim2>}"
}]
}
```

Important

Ensure the Options is a JSON array with at least one object.

The plugin allows users to specify the following parameters for the options object:

- **AuthenticationType:** The name of the authentication scheme registered in the OAuth server for which these mapping options should be applied (this value should correspond to the AuthenticationType value of the authentication plugin).
- **InnovatorUserNameFormat:** A username format string that consists of fixed text intermixed with named placeholders. A placeholder is a claim type that appears enclosed in braces. The result is a string where the corresponding claim value replaces each placeholder. An example of a username format string is:

```
"Prefix_{<Claim1>}_{<Claim2>}_Postfix"
```

Important

The maximum length of Aras Innovator username is 32 characters, so be aware of this when configuring InnovatorUserNameFormat.

- **ClaimActions:** Actions that should be performed on Claims before generating a username based on InnovatorUserNameFormat.

Claim Actions Configuration

Claim Action Configuration objects contain the following settings:

- **ActionName:** Name of action.
- **ActionOptions:** Action configuration.

The GenericUserMapper plugin supports the following types of actions:

- **CreateFrom:** Creates new claim from a value.
- **Validate:** Allows or denies claim values.

CreateFrom Action

The **CreateFrom** action allows users to get a value from one claim, edit it using a regular expression, and save it in a new claim.

The **CreateFrom** action configuration contains the following options:

- **ClaimType:** Type of new claim where a new value is set.
- **SourceClaimType:** Type of claim where a value should be used.
- **ReplacePattern:** The regular expression pattern to match.
- **Replacement:** The string to replace the match.
- **PatternOptions:** Options for matching.

> **Important**
>
> Supported PatternOptions values are described at https://docs.microsoft.com/en-us/dotnet/api/system.text.regularexpressions.regexoptions#fields.

The **CreateFrom** action uses the following algorithm:

1. Get the value from SourceClaimType claim.
2. Apply the regular expression from ReplacePattern with matching options from PatternOptions.
3. Replace the matched value with Replacement.
4. Save the new value in a new ClaimType claim.

The following is an example of configuring a **CreateFrom** action, which takes the value from the claim http://schemas.xmlsoap.org/ws/2005/05/identity/claims/emailaddress, replaces all text after at inclusive with an empty string, and saves the new value in claim username:

```
{
    "ActionName": "CreateFrom",
    "ActionOptions": {
        "ClaimType": "username",
        "SourceClaimType": "http://schemas.xmlsoap.org/ws/2005/05/identity/claims/emailaddress",
        "ReplacePattern": "@.+",
        "Replacement": "",
        "PatternOptions": [ "IgnoreCase", "Singleline" ]
    }
}
```

Important

Only the full format of claim types is supported. To validate regular expressions with different values, it is recommended to use regex tools which are available online.

Validate Action

The **Validate** action enables users to verify a claim value by checking it against specified allow and deny regular expression patterns.

The **Validate** action configuration contains the following options:

- **ClaimType:** The type of claim value to be used.
- **AllowPattern:** The regular expression pattern to match allowed values. This option might not be presented if DenyPattern is set.
- **DenyPattern:** The regular expression pattern to match denied values. This option might not be presented if AllowPattern is set.
- **PatternOptions:** The list of regular expression options used to find a match.

Important

Supported PatternOptions values are described at https://docs.microsoft.com/en-us/dotnet/api/system.text.regularexpressions.regexoptions#fields.

The **Validate** action uses the following algorithm:

1. Get the value from the ClaimType claim.
2. Apply a regular expression from AllowPattern with the matching option from PatternOptions. If there is no match, an error is returned.
3. Apply a regular expression from DenyPattern with a matching option from PatternOptions. If there is a match, an error is returned.

The following is an example of configuring a **Validate** action that allows all values and denies a value if it is equal to any of the standard Aras Innovator administrators:

```
{
    "ActionName": "Validate",
    "ActionOptions": {
        "ClaimType": "http://schemas.xmlsoap.org/ws/2005/05/identity/claims/name",
        "AllowPattern": ".*",
        "DenyPattern": "^admin$|^root$|^vadmin$|^authadmin$|^esadmin$",
        "PatternOptions": [ "IgnoreCase", "Singleline" ]
    }
}
```

Important

Only the full format for claim types is supported.

Generic User Mapper Execution

Claim actions are executed in the same order as defined in a configuration. After executing all actions, the **Generic User Mapper** creates a username using the InnovatorUserNameFormat option.

An example follows:

```
{
            "Name": "Aras.OAuth.Server.Plugins.GenericUserMapper",
            "Enabled": true,
            "Options": [
                {
                    "AuthenticationType": "Saml2-AzureAD",
                    "InnovatorUserNameFormat": "{username}",
                    "ClaimActions": [
                        {
                            "ActionName": "CreateFrom",
                            "ActionOptions": {
                                "ClaimType": "username",
                                "SourceClaimType": "http://schemas.xmlsoap.org/ws/2005/05/identity/claims
                                "ReplacePattern": "@.+",
                                "Replacement": "",
                                "PatternOptions": [ "IgnoreCase", "Singleline" ]
                            }
                        },
                        {
                            "ActionName": "Validate",
                            "ActionOptions": {
                                "ClaimType": "username",
                                "AllowPattern": ".+",
                                "DenyPattern": "^admin$|^root$|^vadmin$|^authadmin$|^esadmin$",
                                "PatternOptions": [ "IgnoreCase", "Singleline" ]
                                }
                        }
                    ]
                }
            ]
        }
}
```

The following is an example of **GenericUserMapper** plugin configuration for SAML2 authentication:

```
{
    "Name": "Aras.OAuth.Server.Plugins.GenericUserMapper",
    "Enabled": true,
    "Options": [
{
    "AuthenticationType": "Saml2",
    "InnovatorUserNameFormat": "{uid}",
    "ClaimActions": [
        {
            // Validate uid (default Innovator users are denied).
            "ActionName": "Validate",
            "ActionOptions": {
                "ClaimType": "uid",
                "AllowPattern": ".+",
                "DenyPattern": "^admin$|^root$|^vadmin$|^authadmin$|^esadmin$",
                "PatternOptions": [ "IgnoreCase", "Singleline" ]
            }
        }
    ]
}
    ]
}
```