Configure authentication policies via Windows PowerShell

Windows PowerShell enables greater flexibility in using various factors of access control and the authentication mechanism that are available in AD FS in Windows Server 2012 R2 to configure authentication policies and authorization rules that are necessary to implement true conditional access for your AD FS-secured resources.

Membership in Administrators, or equivalent, on the local computer is the minimum requirement to complete these procedures. Review details about using the appropriate accounts and group memberships at Local and Domain Default Groups (http://go.microsoft.com/fwlink/?LinkId=83477).

To configure an additional authentication method via Windows PowerShell

---

1. On your federation server, open the Windows PowerShell command window and run the following command.

   Copy

   ```
   Set-AdfsGlobalAuthenticationPolicy -
   AdditionalAuthenticationProvider
   CertificateAuthentication
   ```

Warning

To verify that this command ran successfully, you can run the `Get-AdfsGlobalAuthenticationPolicy` command.

To configure MFA per-relying party trust that is based on a user's group membership data

---

1. On your federation server, open the Windows PowerShell command window and run the following command:

   Copy

   ```
   $rp = Get-AdfsRelyingPartyTrust -
   Name relying_party_trust
   ```

Warning

Ensure to replace *<relying_party_trust>* with the name of your relying party trust.

2. In the same Windows PowerShell command window, run the following command.

   Copy

```
$MfaClaimRule = "c:[Type ==
'"http://schemas.microsoft.com/ws/20
08/06/identity/claims/groupsid'",
Value =~ '"^(?i) <group_SID>$'"] =>
issue(Type =
'"http://schemas.microsoft.com/ws/20
08/06/identity/claims/authentication
method'", Value =
'"http://schemas.microsoft.com/claim
s/multipleauthn'");"

Set-AdfsRelyingPartyTrust -
TargetRelyingParty $rp -
AdditionalAuthenticationRules
$MfaClaimRule
```

Note

Ensure to replace <group_SID> with the value of the security identifier (SID) of your Active Directory (AD) group.

To configure MFA globally based on users' group membership data

---

1. On your federation server, open the Windows PowerShell command window and run the following command.

   Copy

```
$MfaClaimRule = "c:[Type == '"
http://schemas.microsoft.com/ws/2008
/06/identity/claims/groupsid'",
Value == '"group_SID'"]
 => issue(Type =
'"http://schemas.microsoft.com/ws/20
08/06/identity/claims/authentication
method'", Value =
'"http://schemas.microsoft.com/claim
s/multipleauthn'");"

Set-AdfsAdditionalAuthenticationRule
$MfaClaimRule
```

Note

Ensure to replace *<group_SID>* with the value of the SID of your AD group.

To configure MFA globally based on user's location

---

1. On your federation server, open the Windows PowerShell command window and run the following command.

   Copy

   ```
   $MfaClaimRule = "c:[Type == '"
   http://schemas.microsoft.com/ws/2012
   /01/insidecorporatenetwork'", Value
   == '"true_or_false'"]
    => issue(Type =
   '"http://schemas.microsoft.com/ws/20
   08/06/identity/claims/authentication
   method'", Value =
   '"http://schemas.microsoft.com/claim
   s/multipleauthn'");"

   Set-AdfsAdditionalAuthenticationRule
   $MfaClaimRule
   ```

Note

Ensure to replace *<true_or_false>* with either `true` or `false`. The value depends on your specific rule condition that is based on whether the access request comes from the extranet or the intranet.

To configure MFA globally based on user's device data

---

1. On your federation server, open the Windows PowerShell command window and run the following command.

   Copy

   ```
   $MfaClaimRule = "c:[Type == '"
   http://schemas.microsoft.com/2012/01
   /devicecontext/claims/isregisteredus
   er'", Value == '"true_or_false"']
   ```

```
 => issue(Type =
'"http://schemas.microsoft.com/ws/20
08/06/identity/claims/authentication
method'", Value =
'"http://schemas.microsoft.com/claim
s/multipleauthn'");"

Set-AdfsAdditionalAuthenticationRule
$MfaClaimRule
```

Note

Ensure to replace *<true_or_false>* with either `true` or `false`. The value depends on your specific rule condition that is based on whether the device is workplace-joined or not.

To configure MFA globally if the access request comes from the extranet and from a non-workplace-joined device

---

1. On your federation server, open the Windows PowerShell command window and run the following command.

   Copy

```
Set-AdfsAdditionalAuthenticationRule
"c:[Type ==
`"http://schemas.microsoft.com/2012/
01/devicecontext/claims/isregistered
user`", Value == `"true_or_false`"]
&& c2:[Type ==
`"http://schemas.microsoft.com/ws/20
12/01/insidecorporatenetwork`",
Value == `" true_or_false `"] =>
issue(Type =
`"http://schemas.microsoft.com/ws/20
08/06/identity/claims/authentication
method`", Value
=`"http://schemas.microsoft.com/clai
ms/multipleauthn`");"
```

Note

Ensure to replace both instances of *<true_or_false>* with either `true` or `false`, which depends on your specific rule conditions. The rule conditions are based on whether the device is workplace-joined or not and whether the access request comes from the extranet or intranet.

To configure MFA globally if access comes from an extranet user that belongs to a certain group

1. On your federation server, open the Windows PowerShell command window and run the following command.

   Copy

```
Set-AdfsAdditionalAuthenticationRule
"c:[Type ==
`"http://schemas.microsoft.com/ws/20
08/06/identity/claims/groupsid`",
Value == `"group_SID`"]
 && c2:[Type ==
`"http://schemas.microsoft.com/ws/20
12/01/insidecorporatenetwork`",
Value == `"true_or_false`"] =>
issue(Type =
`"http://schemas.microsoft.com/ws/20
08/06/identity/claims/authentication
method`", Value
=`"http://schemas.microsoft.com/clai
ms/multipleauthn`");"
```

Note

Ensure to replace *<group_SID>* with the value of the group SID and *<true_or_false>* with either `true` or `false`, which depends on your specific rule condition that is based on whether the access request comes from the extranet or intranet.

To grant access to an application based on user data via Windows PowerShell

1. On your federation server, open the Windows PowerShell command window and run the following command.

   Copy

```
$rp = Get-AdfsRelyingPartyTrust -
Name relying_party_trust
```

Note

Ensure to replace *<relying_party_trust>* with the value of your relying party trust.

2. In the same Windows PowerShell command window, run the following command.

   Copy

```
$GroupAuthzRule = "@RuleTemplate =
`"Authorization`" @RuleName =
`"Foo`" c:[Type ==
`"http://schemas.microsoft.com/ws/20
08/06/identity/claims/groupsid`",
Value =~ `"^(?i)<group_SID>$`"]
=>issue(Type =
`"http://schemas.microsoft.com/autho
rization/claims/deny`", Value =
`"DenyUsersWithClaim`");"
Set-AdfsRelyingPartyTrust -
TargetRelyingParty $rp -
IssuanceAuthorizationRules
$GroupAuthzRule
```

Note

Ensure to replace *<group_SID>* with the value of the SID of your AD group.

To grant access to an application that is secured by AD FS only if this user's identity was validated with MFA

---

1. On your federation server, open the Windows PowerShell command window and run the following command.

   Copy

   ```
   $rp = Get-AdfsRelyingPartyTrust -
   Name relying_party_trust
   ```

Note

Ensure to replace *<relying_party_trust>* with the value of your relying party trust.

2. In the same Windows PowerShell command window, run the following command.

   Copy

   ```
   $GroupAuthzRule = "@RuleTemplate =
   `"Authorization`"
   @RuleName = `"PermitAccessWithMFA`"
   c:[Type ==
   `"http://schemas.microsoft.com/claim
   s/authnmethodsreferences`", Value =~
   `"^(?i)http://schemas\.microsoft\.co
   m/claims/multipleauthn$`"] =>
   issue(Type =
   `"http://schemas.microsoft.com/autho
   ```

```
rization/claims/permit`", Value =
'`"PermitUsersWithClaim'");"
```

To grant access to an application that is secured by AD FS only if the access request comes from a workplace-joined device that is registered to the user

---

1. On your federation server, open the Windows PowerShell command window and run the following command.

   Copy

   ```
   $rp = Get-AdfsRelyingPartyTrust -
   Name relying_party_trust
   ```

Note

Ensure to replace *<relying_party_trust>* with the value of your relying party trust.

2. In the same Windows PowerShell command window, run the following command.

   Copy

   ```
   $GroupAuthzRule = "@RuleTemplate =
   `"Authorization`"
   @RuleName =
   `"PermitAccessFromRegisteredWorkplac
   eJoinedDevice`"
   c:[Type ==
   `"http://schemas.microsoft.com/2012/
   01/devicecontext/claims/isregistered
   user`", Value =~ `"^(?i)true$`"] =>
   issue(Type =
   `"http://schemas.microsoft.com/autho
   rization/claims/permit`", Value =
   `"PermitUsersWithClaim`");
   ```

To grant access to an application that is secured by AD FS only if the access request comes from a workplace-joined device that is registered to a user whose identity has been validated with MFA

---

1. On your federation server, open the Windows PowerShell command window and run the following command.

   Copy

   ```
   $rp = Get-AdfsRelyingPartyTrust -
   Name relying_party_trust
   ```

Note

Ensure to replace *<relying_party_trust>* with the value of your relying party trust.

2. In the same Windows PowerShell command window, run the following command.

   Copy

   ```
   $GroupAuthzRule = '@RuleTemplate =
   "Authorization"
   @RuleName =
   "RequireMFAOnRegisteredWorkplaceJoin
   edDevice"
   c1:[Type ==
   `"http://schemas.microsoft.com/claim
   s/authnmethodsreferences`", Value =~
   `"^(?i)http://schemas\.microsoft\.co
   m/claims/multipleauthn$`"] &&
   c2:[Type ==
   `"http://schemas.microsoft.com/2012/
   01/devicecontext/claims/isregistered
   user`", Value =~ `"^(?i)true$"] =>
   issue(Type =
   "http://schemas.microsoft.com/author
   ization/claims/permit`", Value =
   `"PermitUsersWithClaim`");"
   ```

To grant extranet access to an application secured by AD FS only if the access request comes from a user whose identity has been validated with MFA

---

1. On your federation server, open the Windows PowerShell command window and run the following command.

   Copy

   ```
   $rp = Get-AdfsRelyingPartyTrust -
   Name relying_party_trust
   ```

Note

Ensure to replace *<relying_party_trust>* with the value of your relying party trust.

2. In the same Windows PowerShell command window, run the following command.

   Copy

```
$GroupAuthzRule = "@RuleTemplate =
`"Authorization`"
@RuleName =
`"RequireMFAForExtranetAccess`"
c1:[Type ==
`"http://schemas.microsoft.com/claim
s/authnmethodsreferences`", Value =~
`"^(?i)http://schemas\.microsoft\.co
m/claims/multipleauthn$`"] &&
c2:[Type ==
`"http://schemas.microsoft.com/ws/20
12/01/insidecorporatenetwork`",
Value =~ `"^(?i)false$`"] =>
issue(Type =
`"http://schemas.microsoft.com/autho
rization/claims/permit`", Value =
`"PermitUsersWithClaim`");"
```