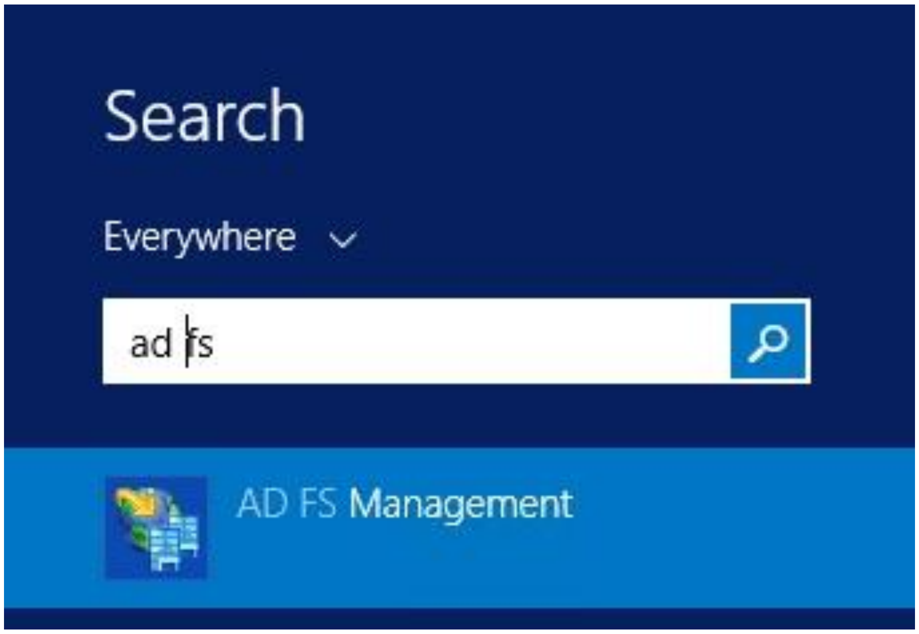
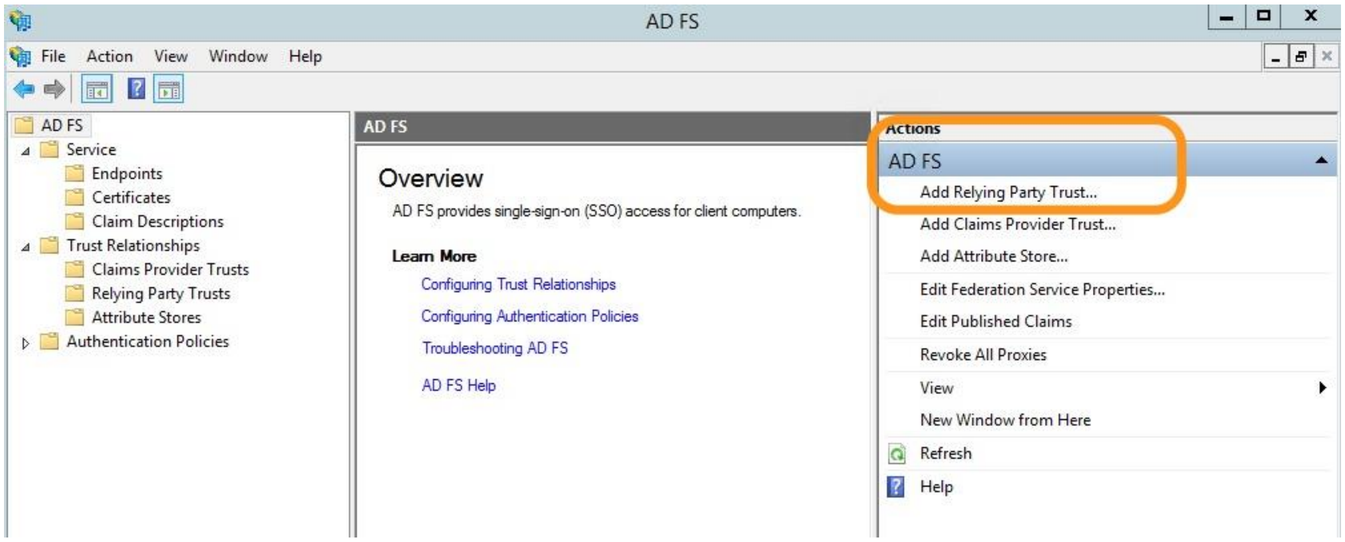
Configure AWS as a Trusted Relying Party

Next, we need to tell AD FS that it should offer authentication services for AWS. The SAML configuration for doing so is known as a *relying party*.

To start, choose the **Start menu**, type **ad fs**, and choose **AD FS Management**.

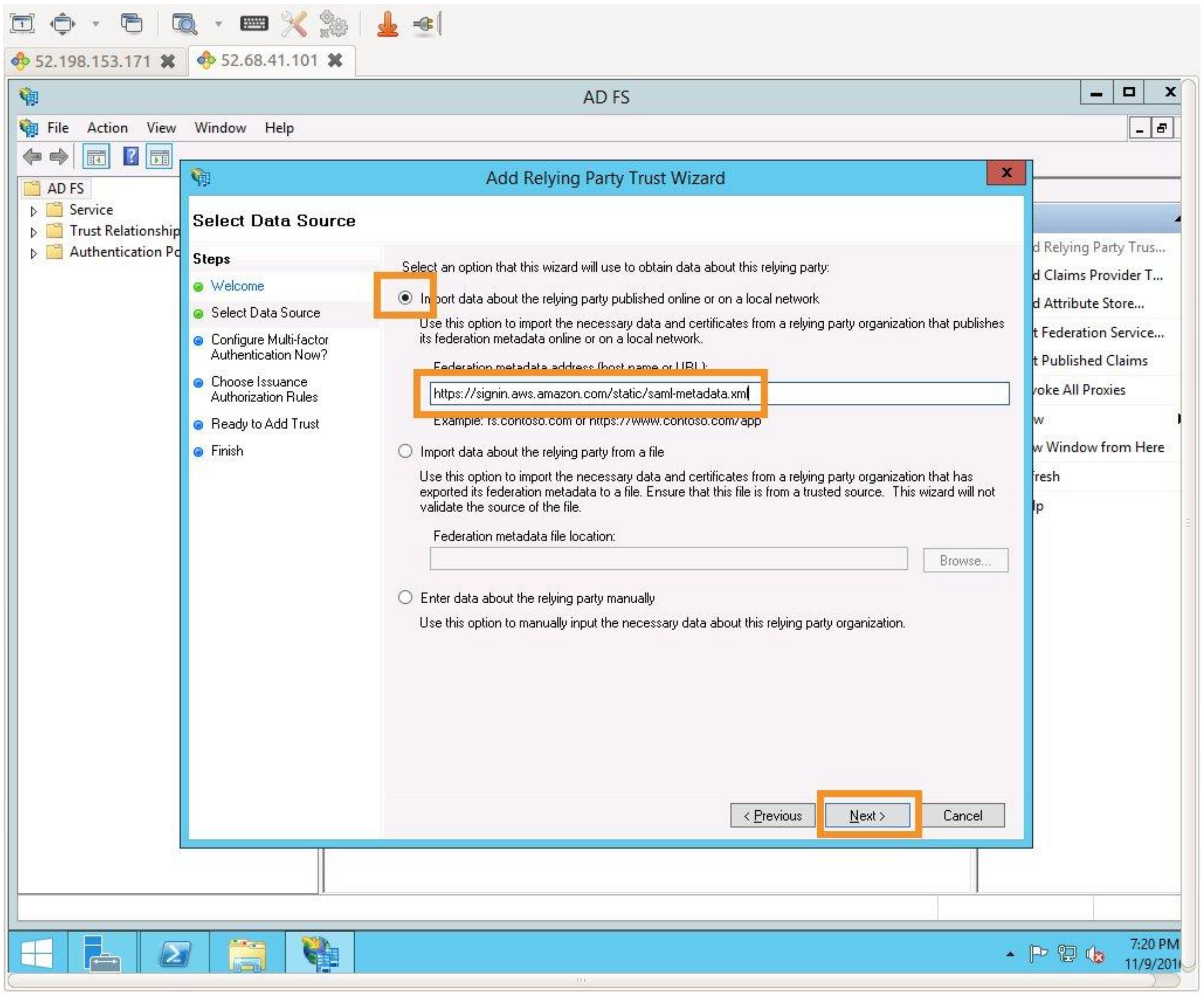


Choose **Add Relying Party Trust** to start the **Add Relying Party Trust Wizard**.



Choose **Start** to begin. Then, on **Select Data Source**, type the following URL for the AWS SAML metadata, and then choose **Next**.

```
https://signin.aws.amazon.com/static/saml-metadata.xml
```


Add Relying Party Trust Wizard

Select Data Source

Steps
- Welcome
- Select Data Source
- Configure Multi-factor Authentication Now?
- Choose Issuance Authorization Rules
- Ready to Add Trust
- Finish

Select an option that this wizard will use to obtain data about this relying party:

● Import data about the relying party published online or on a local network

Use this option to import the necessary data and certificates from a relying party organization that publishes its federation metadata online or on a local network.

Federation metadata address (host name or URL):

https://signin.aws.amazon.com/static/saml-metadata.xml

Example: fs.contoso.com or https://www.contoso.com/app

○ Import data about the relying party from a file

Use this option to import the necessary data and certificates from a relying party organization that has exported its federation metadata to a file. Ensure that this file is from a trusted source. This wizard will not validate the source of the file.

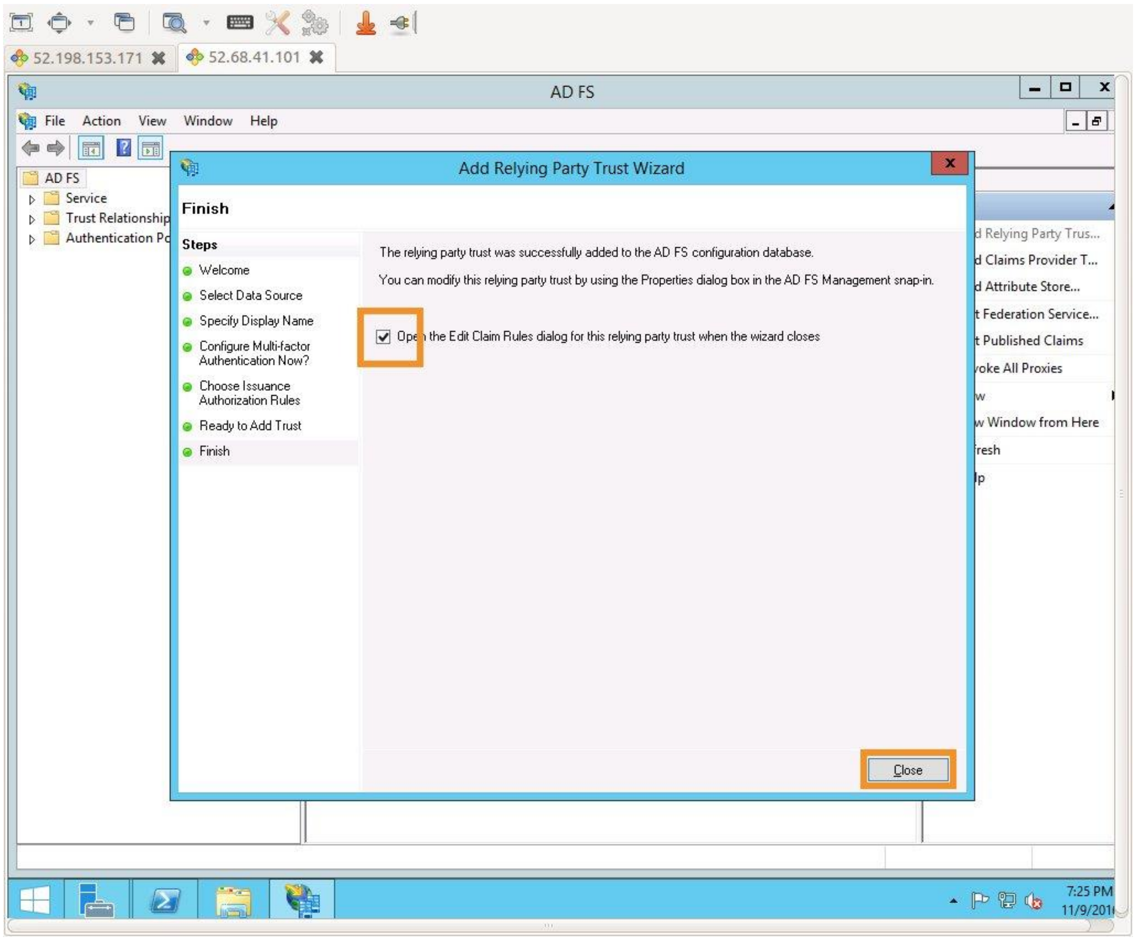
Federation metadata file location:

○ Enter data about the relying party manually

Use this option to manually input the necessary data about this relying party organization.

Note: SAML federations use metadata documents to maintain information about the public keys and certificates that each party utilizes. At run time, each member of the federation can then use this information to validate that the cryptographic elements of the distributed transactions come from the expected actors and haven't been tampered with. Since these metadata documents do not contain any sensitive cryptographic material, AWS publishes federation metadata at https://signin.aws.amazon.com/static/saml-metadata.xml.

Accept the defaults for **Specify Display Name**, **Configure Multi-factor Authentication Now?**, and **Choose Issuance Authorization Rules** by choosing **Next** three times. Choose **Close** on the **Finish** page to complete the **Add Relying Party Trust Wizard**.



The relying party trust was successfully added to the AD FS configuration database.

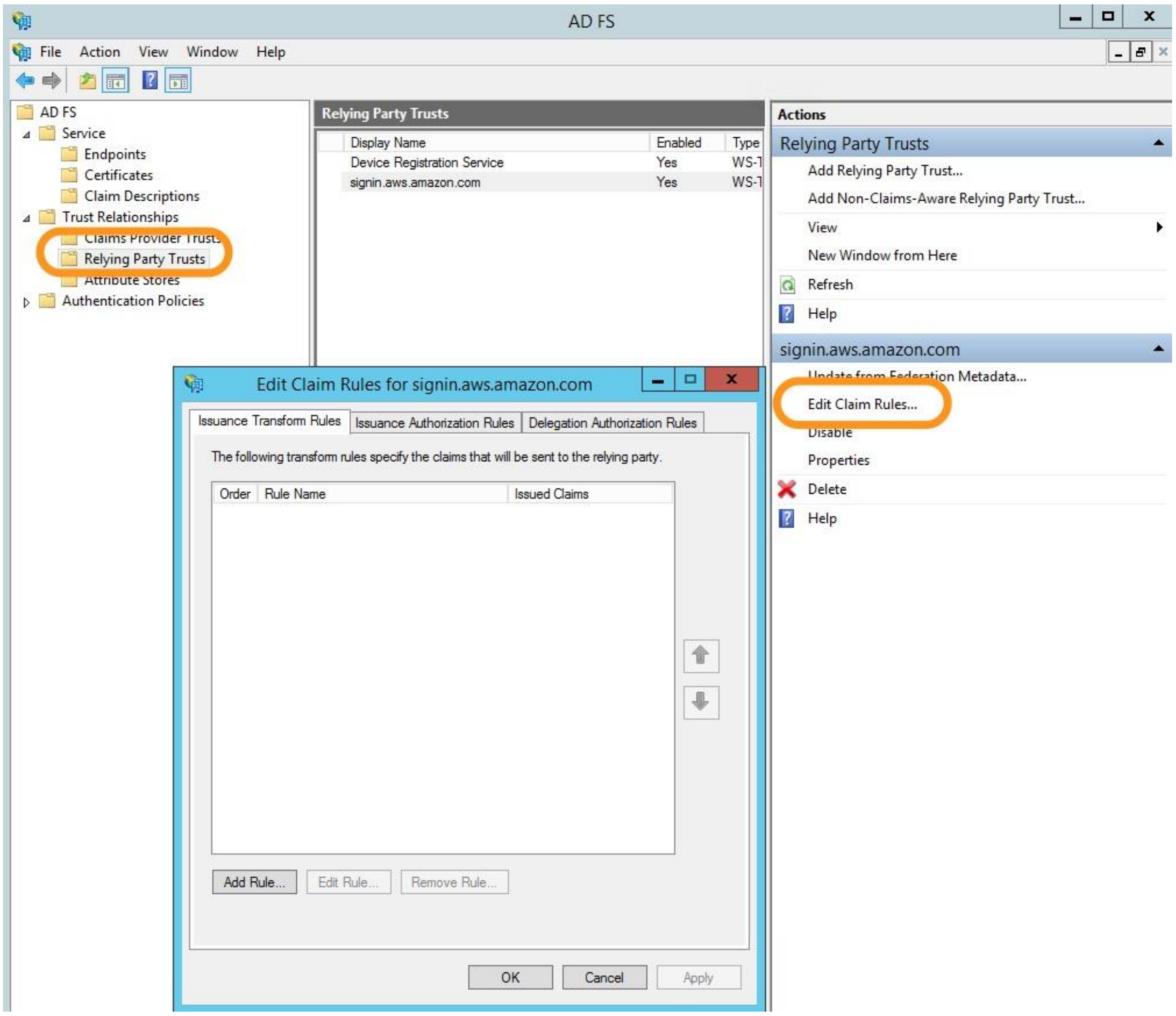
You can modify this relying party trust by using the Properties dialog box in the AD FS Management snap-in.

☑ Open the Edit Claim Rules dialog for this relying party trust when the wizard closes

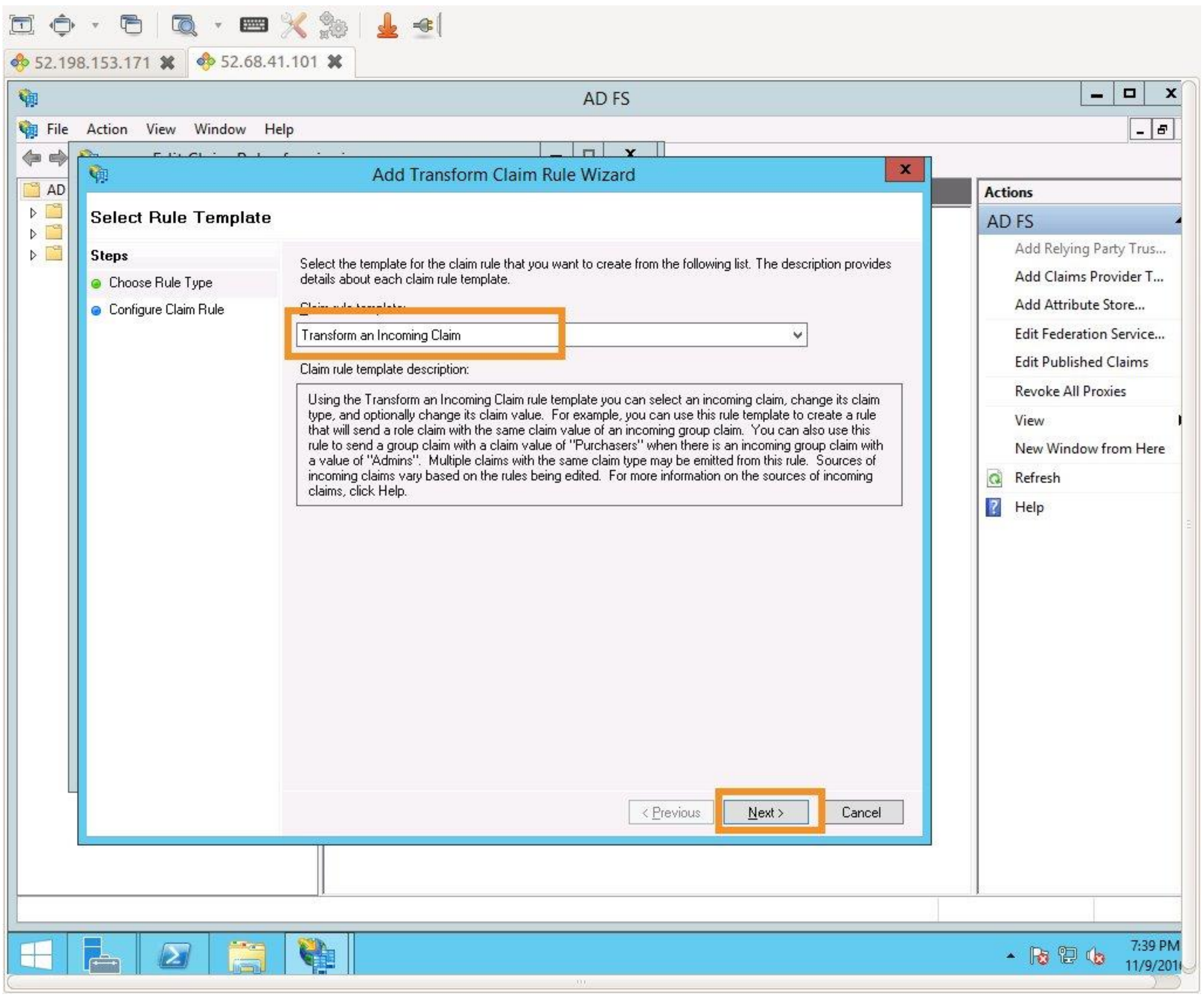
Configure claim rules for the AWS relying party

In a SAML federation, the IdP can pass various attributes about the user, the authentication method, or other points of context to the service provider (in this case AWS) in the form of SAML attributes. In order for SAML federation for AWS to function properly, several attributes are required. In AD FS, **Claim Rules** are used to assemble these required attributes using a combination of Active Directory lookups, simple transformations, and regular expression based custom rules. You will configure a total of **four claim rules**.

The **Edit Claim Rules** window should be already open, if not, you can re-open it from **Relying Party Trusts**, by choosing signin.aws.amazon.com, and then choosing **Edit Claim rules** as shown in the following screenshot.


AD FS
File Action View Window Help
AD FS
Service
Endpoints
Certificates
Claim Descriptions
Trust Relationships
Claims Provider Trusts
Relying Party Trusts
Attribute Stores
Authentication Policies
Relying Party Trusts
Display Name Enabled Type
Device Registration Service Yes WS-T
signin.aws.amazon.com Yes WS-T
Actions
Relying Party Trusts
Add Relying Party Trust...
Add Non-Claims-Aware Relying Party Trust...
View
New Window from Here
Refresh
Help
signin.aws.amazon.com
Update from Federation Metadata...
Edit Claim Rules...
Disable
Properties
Delete
Help
Edit Claim Rules for signin.aws.amazon.com
Issuance Transform Rules
Issuance Authorization Rules
Delegation Authorization Rules
The following transform rules specify the claims that will be sent to the relying party.
Order Rule Name Issued Claims
Add Rule...
Edit Rule...
Remove Rule...
OK
Cancel
Apply

Choose **Add Rule...** to configure the first rule, and then choose **Transform an incoming claim**. Finally, choose **Next**.



Select Rule Template

Select the template for the claim rule that you want to create from the following list. The description provides details about each claim rule template.

Claim rule template:

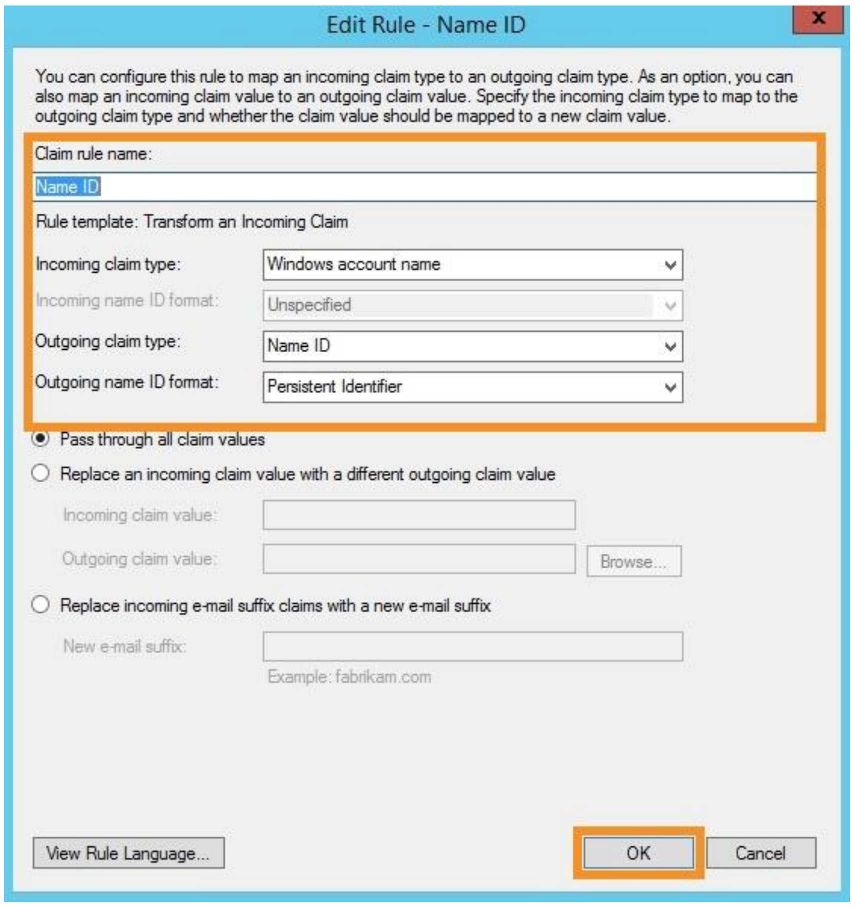
Transform an Incoming Claim

Claim rule template description:

Using the Transform an Incoming Claim rule template you can select an incoming claim, change its claim type, and optionally change its claim value. For example, you can use this rule template to create a rule that will send a role claim with the same claim value of an incoming group claim. You can also use this rule to send a group claim with a claim value of "Purchasers" when there is an incoming group claim with a value of "Admins". Multiple claims with the same claim type may be emitted from this rule. Sources of incoming claims vary based on the rules being edited. For more information on the sources of incoming claims, click Help.

On the **Configure Claim Rule** page, type the following settings, and choose **OK**, as shown in the following screenshot.

Configuration Element	**Value**
Claim rule name	Name ID
Incoming claim type	Windows account name
Outgoing claim type	Name ID
Outgoing name ID format	Persistent Identifier


Edit Rule - Name ID

You can configure this rule to map an incoming claim type to an outgoing claim type. As an option, you can also map an incoming claim value to an outgoing claim value. Specify the incoming claim type to map to the outgoing claim type and whether the claim value should be mapped to a new claim value.

Claim rule name: Name ID

Rule template: Transform an Incoming Claim

Incoming claim type: Windows account name

Incoming name ID format: Unspecified

Outgoing claim type: Name ID

Outgoing name ID format: Persistent Identifier

[✓ Pass through all claim values]

Replace an incoming claim value with a different outgoing claim value

Incoming claim value:

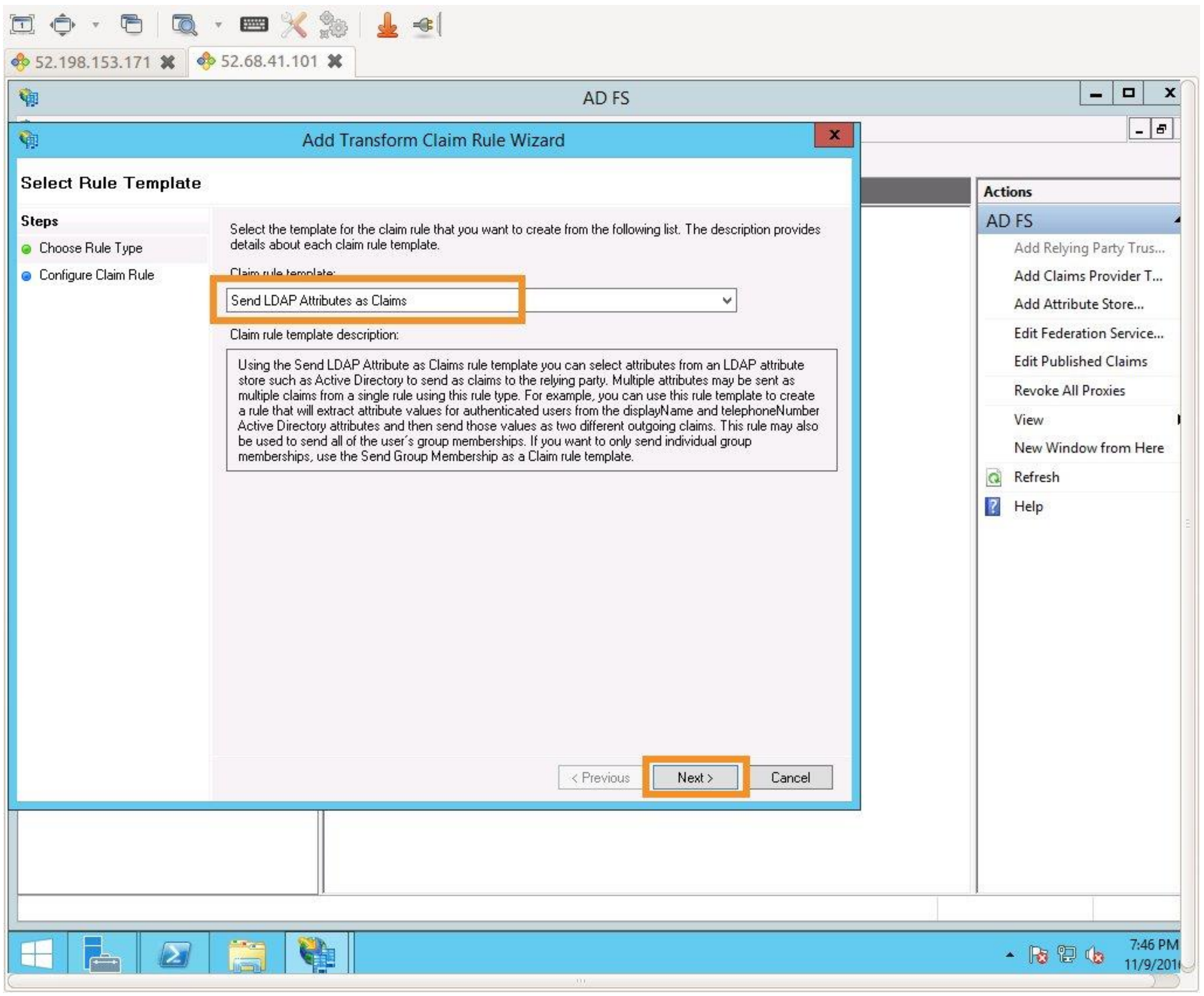
Outgoing claim value:

Replace incoming e-mail suffix claims with a new e-mail suffix

New e-mail suffix:

Example: fabrikam.com

Choose **Add Rule** to configure the second rule, and then choose **Send LDAP Attributes as Claims**. Finally, choose **Next**.



Select the template for the claim rule that you want to create from the following list. The description provides details about each claim rule template.

Claim rule template:

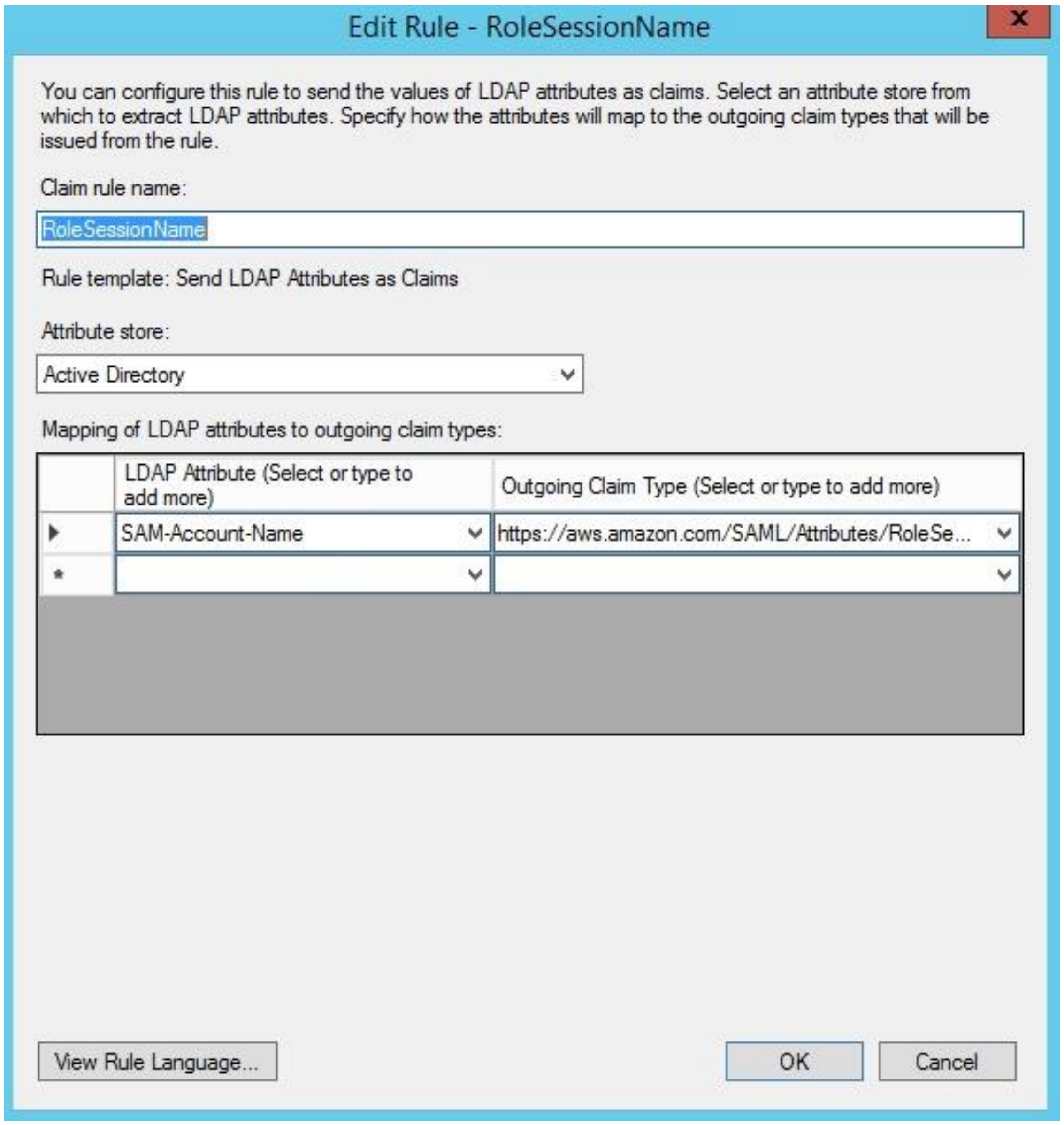
Send LDAP Attributes as Claims

Claim rule template description:

Using the Send LDAP Attribute as Claims rule template you can select attributes from an LDAP attribute store such as Active Directory to send as claims to the relying party. Multiple attributes may be sent as multiple claims from a single rule using this rule type. For example, you can use this rule template to create a rule that will extract attribute values for authenticated users from the displayName and telephoneNumber Active Directory attributes and then send those values as two different outgoing claims. This rule may also be used to send all of the user's group memberships. If you want to only send individual group memberships, use the Send Group Membership as a Claim rule template.

On the **Configure Claim Rule** page, type the following settings, and choose **OK**, as shown in the following screenshot.

Configuration Element	**Value**
Claim rule name	RoleSessionName
Attribute store	Active Directory
LDAP Attribute	SAM-Account-Name
Outgoing Claim Type	https://aws.amazon.com/SAML/Attributes/RoleSessionName



You can configure this rule to send the values of LDAP attributes as claims. Select an attribute store from which to extract LDAP attributes. Specify how the attributes will map to the outgoing claim types that will be issued from the rule.

Claim rule name:

RoleSessionName

Rule template: Send LDAP Attributes as Claims

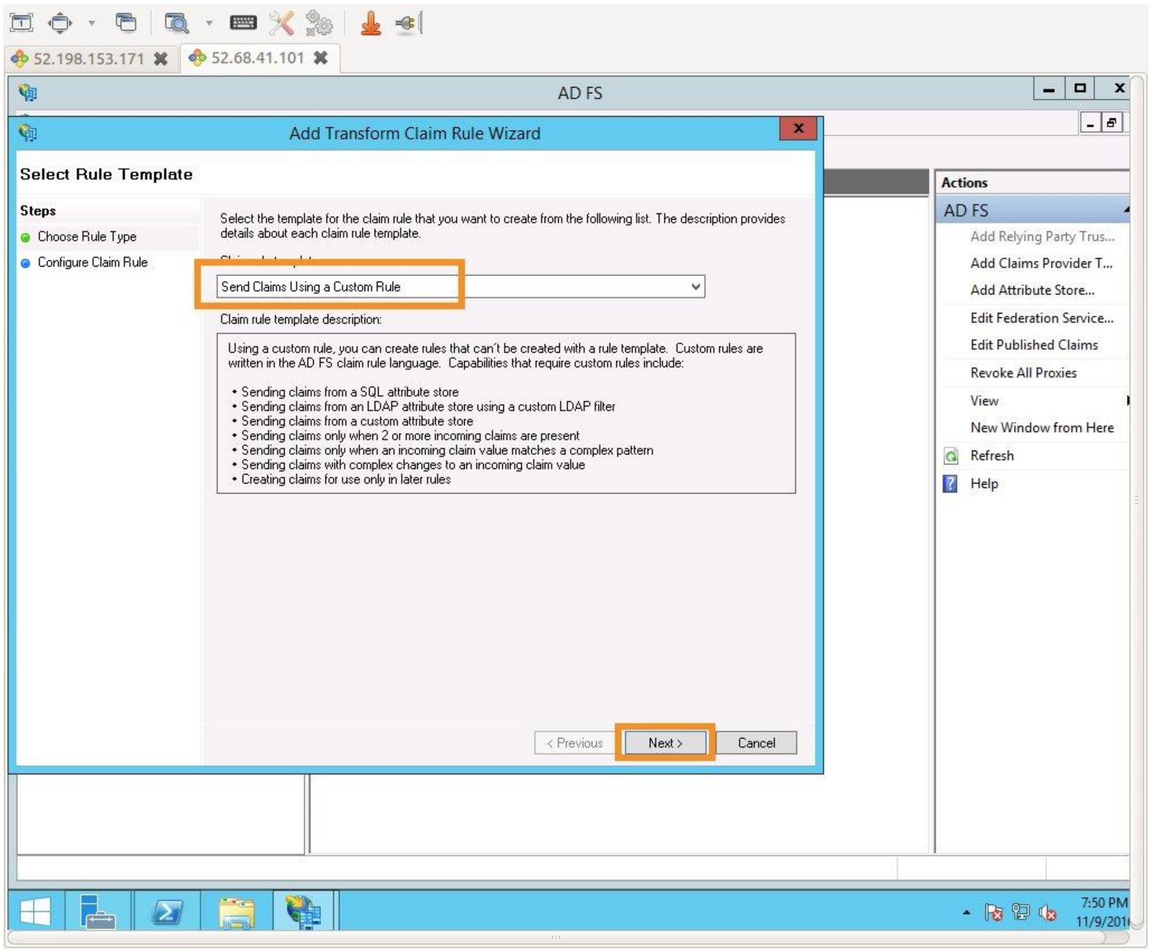
Attribute store:

Active Directory

Mapping of LDAP attributes to outgoing claim types:

	LDAP Attribute (Select or type to add more)	Outgoing Claim Type (Select or type to add more)
▶	SAM-Account-Name	https://aws.amazon.com/SAML/Attributes/RoleSe...
*		

Choose **Add Rule** to configure the third rule, and then choose **Send claims using a custom rule**. Finally, choose **Next**.



Select the template for the claim rule that you want to create from the following list. The description provides details about each claim rule template.

Send Claims Using a Custom Rule

Claim rule template description:

Using a custom rule, you can create rules that can't be created with a rule template. Custom rules are written in the AD FS claim rule language. Capabilities that require custom rules include:

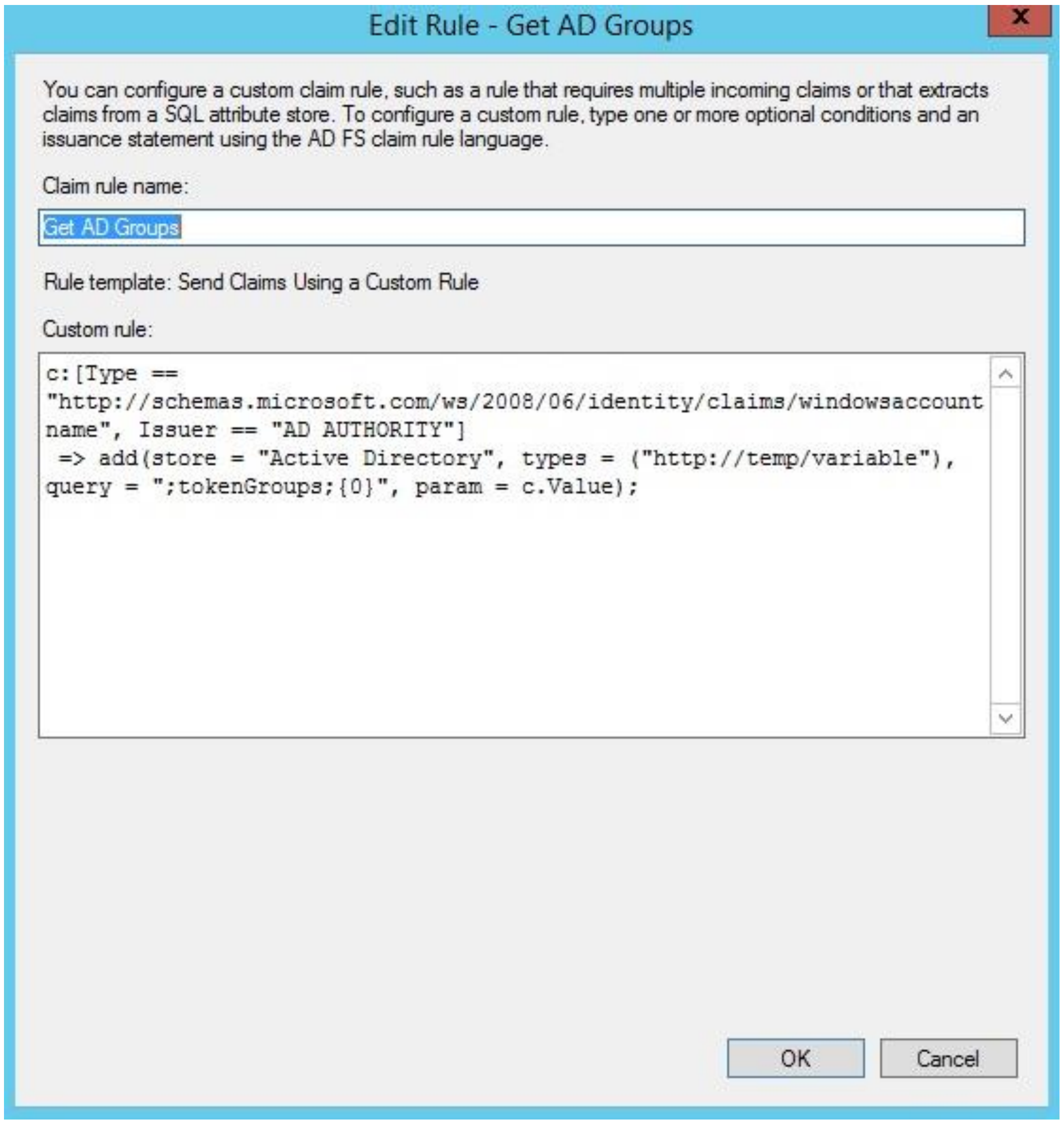
- Sending claims from a SQL attribute store
- Sending claims from an LDAP attribute store using a custom LDAP filter
- Sending claims from a custom attribute store
- Sending claims only when 2 or more incoming claims are present
- Sending claims only when an incoming claim value matches a complex pattern
- Sending claims with complex changes to an incoming claim value
- Creating claims for use only in later rules

In the **Claim rule name** box, type **Get AD Groups**. In the **Custom rule** box, enter the following, and then choose **OK**.

```
c:[Type ==
"http://schemas.microsoft.com/ws/2008/06/identity/claims/windowsaccountname",
Issuer == "AD AUTHORITY"] => add(store = "Active Directory", types =
("http://temp/variable"), query = ";tokenGroups;{0}", param = c.Value);
```

This custom rule uses a script in the claim rule language that retrieves all the groups the authenticated user is a member of and places them into a temporary claim named http://temp/variable. Think of this as a variable you can access later.

Note: Ensure that there isn't any trailing whitespace, as this may cause unexpected results.


Edit Rule - Get AD Groups

You can configure a custom claim rule, such as a rule that requires multiple incoming claims or that extracts claims from a SQL attribute store. To configure a custom rule, type one or more optional conditions and an issuance statement using the AD FS claim rule language.

Claim rule name:

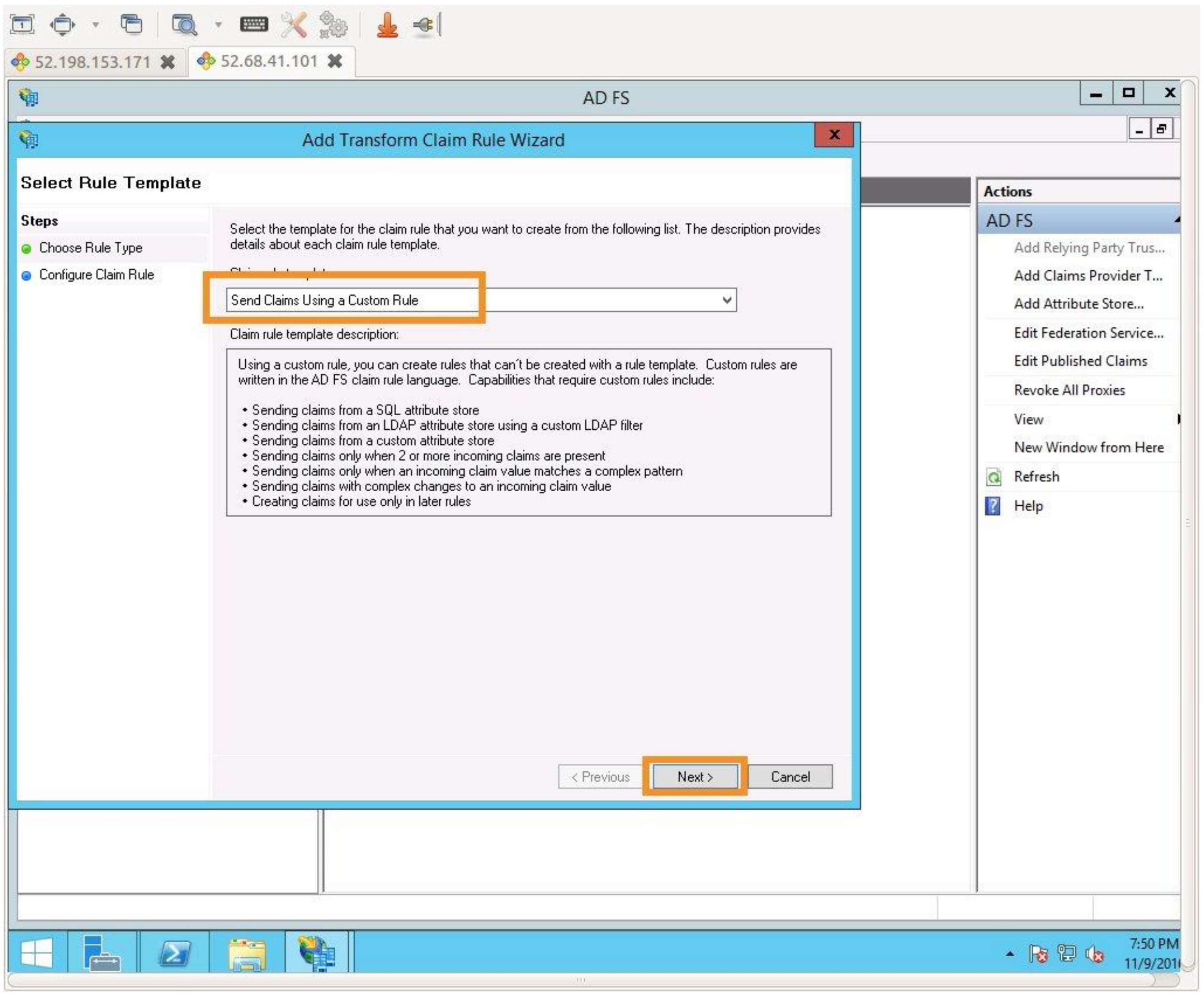
Get AD Groups

Rule template: Send Claims Using a Custom Rule

Custom rule:

```
c:[Type ==
"http://schemas.microsoft.com/ws/2008/06/identity/claims/windowsaccount
name", Issuer == "AD AUTHORITY"]
 => add(store = "Active Directory", types = ("http://temp/variable"),
query = ";tokenGroups;{0}", param = c.Value);
```

Choose **Add Rule** to configure the fourth and final rule. Then choose **Send claims using a custom rule**, and choose **Next**.



Select Rule Template

Select the template for the claim rule that you want to create from the following list. The description provides details about each claim rule template.

Send Claims Using a Custom Rule

Claim rule template description:

Using a custom rule, you can create rules that can't be created with a rule template. Custom rules are written in the AD FS claim rule language. Capabilities that require custom rules include:

- Sending claims from a SQL attribute store
- Sending claims from an LDAP attribute store using a custom LDAP filter
- Sending claims from a custom attribute store
- Sending claims only when 2 or more incoming claims are present
- Sending claims only when an incoming claim value matches a complex pattern
- Sending claims with complex changes to an incoming claim value
- Creating claims for use only in later rules

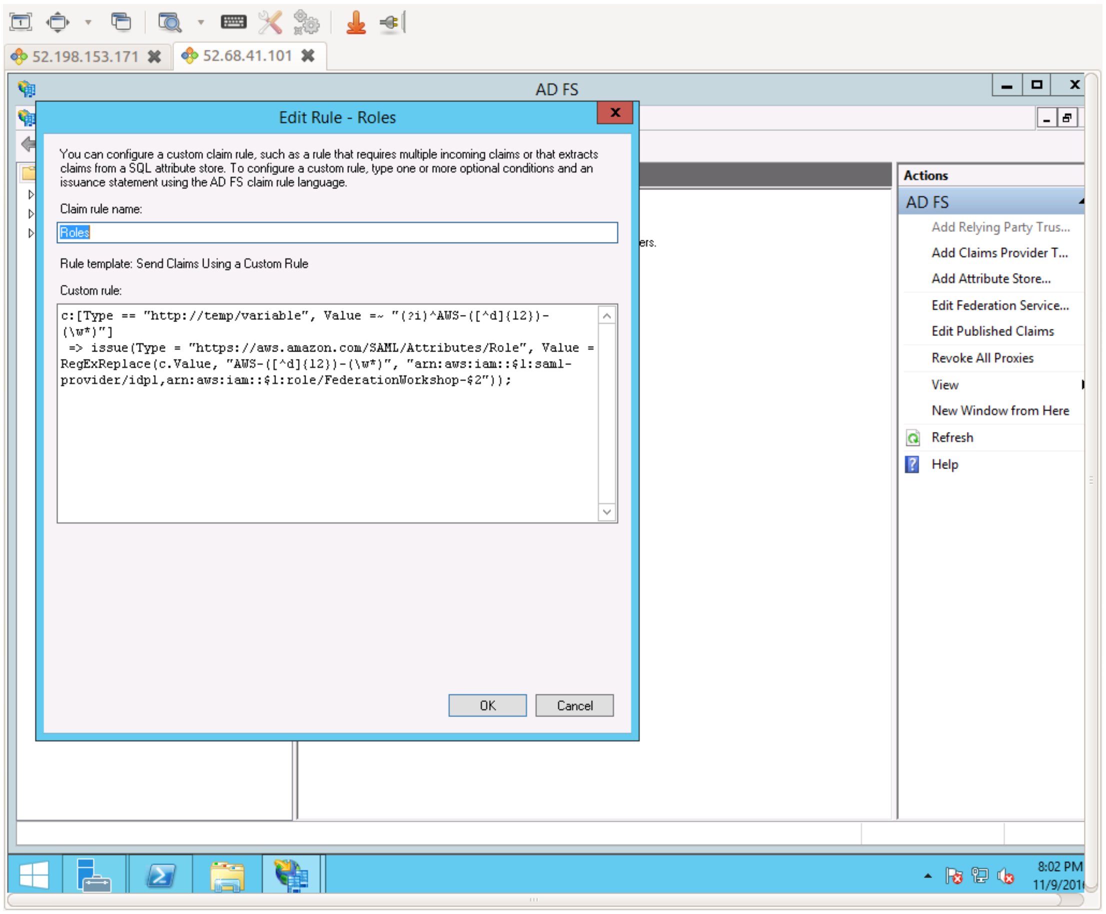
In the **Claim rule name** box, type **Roles**. In the **Custom rule** box, enter the following, and then choose **OK**.

```
c:[Type == "http://temp/variable", Value =~ "(?i)^AWS-([^d]{12})-(\w*)"] =>
issue(Type = "https://aws.amazon.com/SAML/Attributes/Role", Value =
RegExReplace(c.Value, "AWS-([^d]{12})-(\w*)", "arn:aws:iam::$1:saml-
provider/idp1,arn:aws:iam::$1:role/FederationWorkshop-$2"));
```

This custom rule uses regular expressions to transform each of the group memberships of the form **AWS-<Account Number>-<Role Name>** into in the IAM role ARN, IAM federation provider ARN form AWS expects. It does so by matching the first pattern match (\d{12}) to $1 and the second pattern match (\w*) to $2 for each entry in the temp variable.

This attribute makes better sense later in the exercise when you see it in action. For now, the key takeaway is that you are defining the resulting value for the AWS Role attribute in a dynamic way (by mapping group memberships) instead of a static way (by explicitly defining ARNs). This dynamic resolution allows the IdP configuration to scale to support virtually **any number** of AWS accounts and any number of IAM roles without further configuration.

Note: Ensure that there isn't any trailing whitespace, as this may cause unexpected results.



You can configure a custom claim rule, such as a rule that requires multiple incoming claims or that extracts claims from a SQL attribute store. To configure a custom rule, type one or more optional conditions and an issuance statement using the AD FS claim rule language.

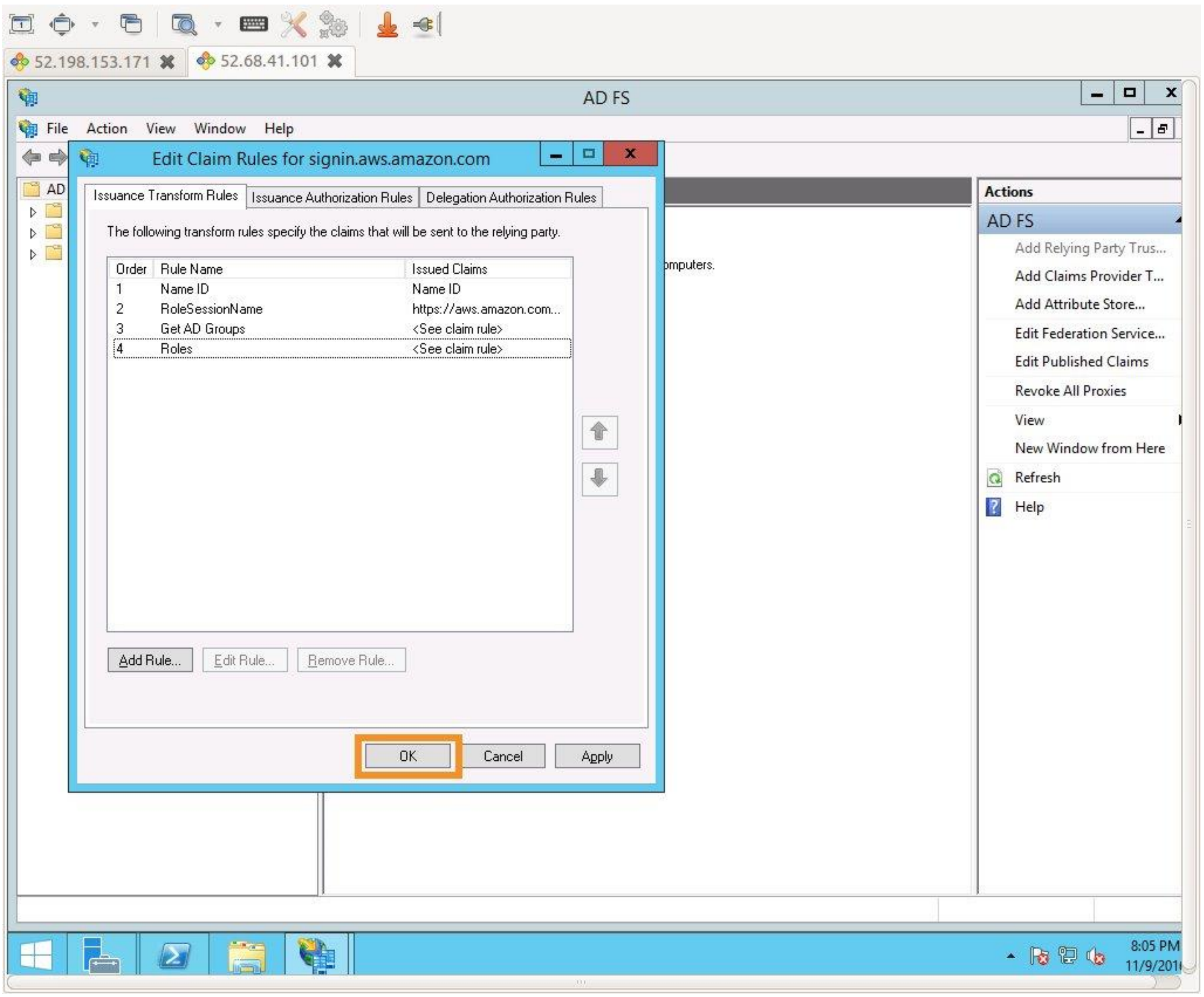
Claim rule name: Roles

Rule template: Send Claims Using a Custom Rule

Custom rule:

```
c:[Type == "http://temp/variable", Value =~ "(?i)^AWS-([^d]{12})-
(\w*)"]
 => issue(Type = "https://aws.amazon.com/SAML/Attributes/Role", Value =
RegExReplace(c.Value, "AWS-([^d]{12})-(\w*)", "arn:aws:iam::$1:saml-
provider/idp1,arn:aws:iam::$1:role/FederationWorkshop-$2"));
```

Choose **OK** to complete the **Edit Claim Rules** wizard.



The following transform rules specify the claims that will be sent to the relying party.

Order	Rule Name	Issued Claims
1	Name ID	Name ID
2	RoleSessionName	https://aws.amazon.com...
3	Get AD Groups	<See claim rule>
4	Roles	<See claim rule>