Implement Dynamic Access Control

DAC relies on file classifications, on user and device attributes called claims, and on rules and policies built from all of these elements. DAC, admittedly, can be very complex.

Introduction to DAC

DAC is a new way to control access to files. It doesn't replace NTFS and share permissions but is sometimes combined with them. When DAC permissions are combined with the NTFS and share permissions, the most restrictive permissions always apply to the account requesting access.

You can think of DAC as being based on *access rules*. These rules are if-then statements built on the attributes of files, users, and devices. An example expression to serve as the basis for an access rule could be "If a user is a member of the finance department with an office on Floor 10 and is connecting from a device that is located in the company HQ, then that user can access finance files and folder designated as having a high business impact." Before you can even create such an access rule, you need to create and assign the needed attributes to all the objects mentioned in that rule. The user and device attributes are called *claims*. The file attributes are called *classifications* (or *resource properties*).

The way these three attribute types relate to an access rule is illustrated in Figure 11-1.

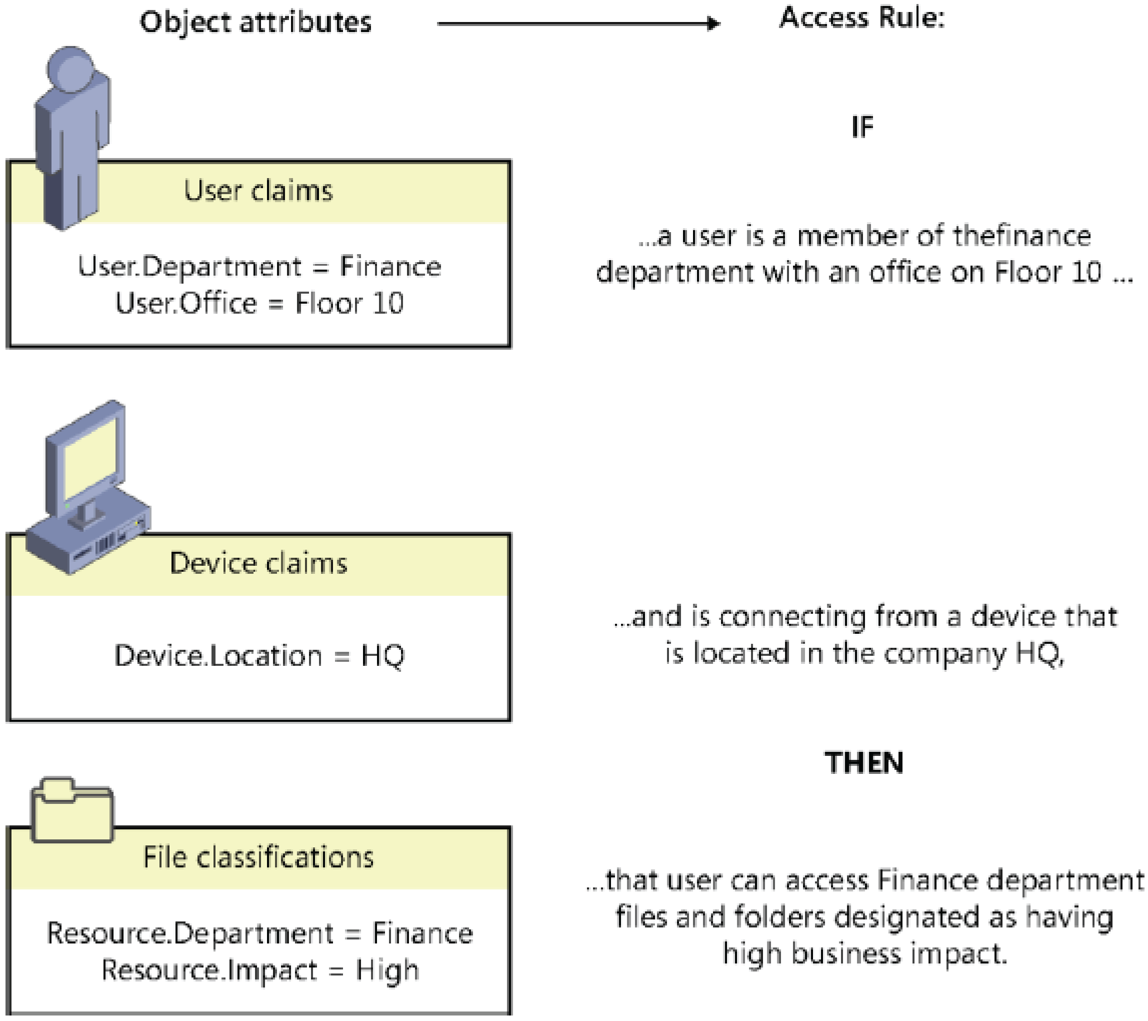

Object attributes
Access Rule:
IF
User claims
User.Department = Finance
User.Office = Floor 10
...a user is a member of thefinance department with an office on Floor 10 ...
Device claims
Device.Location = HQ
...and is connecting from a device that is located in the company HQ,
THEN
File classifications
Resource.Department = Finance
Resource.Impact = High
...that user can access Finance department files and folders designated as having high business impact.

FIGURE 11-1 Access rules refer to the attributes of users, devices, and files.

FIGURE 11-1

Access rules refer to the attributes of users, devices, and files.

DAC is advantageous for a number of reasons. First, it allows administrators to manage file access centrally, in a way that impacts all file servers in the organization. (It should be noted, however, that you cannot enforce access rules centrally through DAC; you can only make access rules available for enforcement.) Another advantage of DAC is that it allows you to dramatically reduce the number of user groups you would otherwise need to create and manage to implement a particular access policy. A third advantage of DAC is that it allows you to construct access rules in a way that is much more flexible and much more likely to correspond to the needs of your organization. Instead of access control lists (ACLs) based only on user and group accounts, you can create rules based on location, office, country, telephone number, or any other parameter that is most useful to you.

To implement DAC, you need at least one Windows Server 2012 file server, at least one Windows Server 2012 domain controller (one recommended at each site), and Windows 7 clients or higher. In addition, specific features such as access-denied assistance require Windows 8. The domain functional level must also be set to Windows Server 2012.

Even more than with most new features that you will be tested on for the 70-412 exam, DAC is best understood by working with it hands-on. True, it's not the easiest of the new Windows Server 2012 features to master, but without implementing it on a test network, it can seem more complicated than it really is. To prepare for the exam, then, use the following sections as a general walk-through for hands-on configuration, if at all possible. Plan to walk through these steps at least twice, and it will all begin to make sense.

Configuring claims-based authentication

DAC relies on an expanded Kerberos token. This Kerberos token includes more than the usual data, which is (as you remember) the user's ID and group memberships. Besides this information, the expanded Kerberos token used in DAC includes certain attribute values called claims about the user, additional claims about the device to which the user is signed on, and the same device's own group memberships. The expanded Kerberos token used in DAC is illustrated in Figure 11-2.

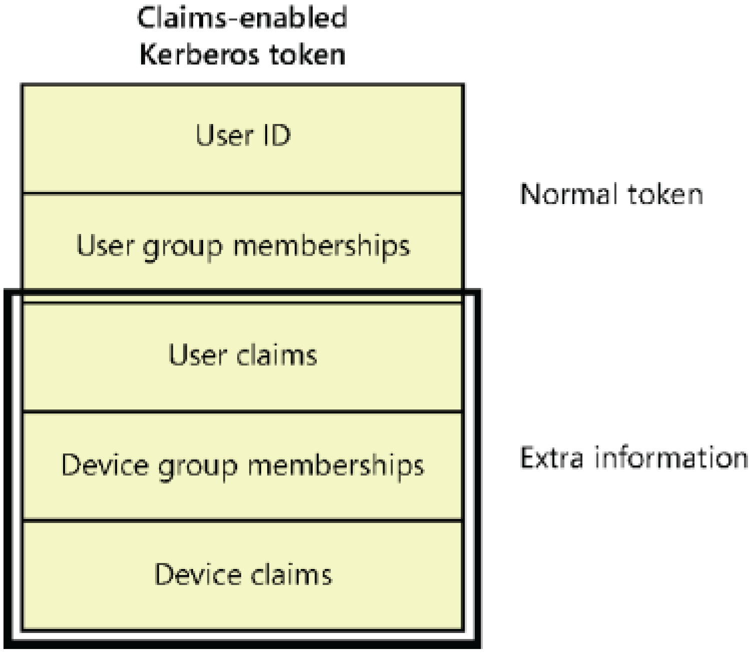


FIGURE 11-2 The Kerberos token used in DAC.

To configure a DAC policy, you need to perform the following steps:

1. Define the types of claims about users and devices you want to include in Kerberos tokens.

2. Configure Active Directory Domain Services to use the expanded Kerberos tokens that include these claims.

Step 1: Define user and device claims types

In this step, you choose the specific user and device properties that will be presented as claims in the Kerberos token whenever access permissions are evaluated. User and device claim types correspond to names of Active Directory attributes (such as “Department” or “City”) for user and computer account objects. The actual claim values included in a token are copied from the corresponding Active Directory attribute values. Because access rules refer to these claim types in their specifications about who is allowed or denied access to a given resource, you want to define claims types you will need later when you create access control rules.

You can use Active Directory Administrative Center to configure the user and device claim types. In the console tree, select tree view and then navigate to Dynamic Access Control\Claim Types. Right-click Claim Types, click New, and then select Claim Type, as shown in Figure 11-3.

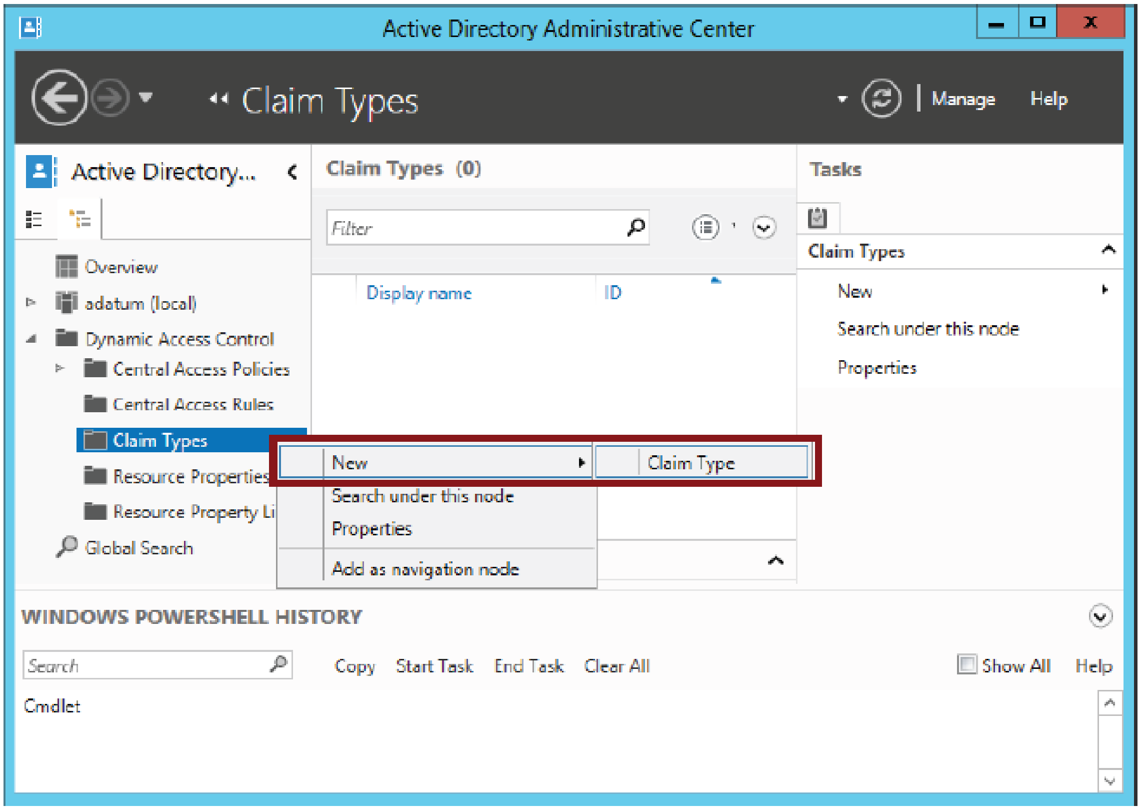
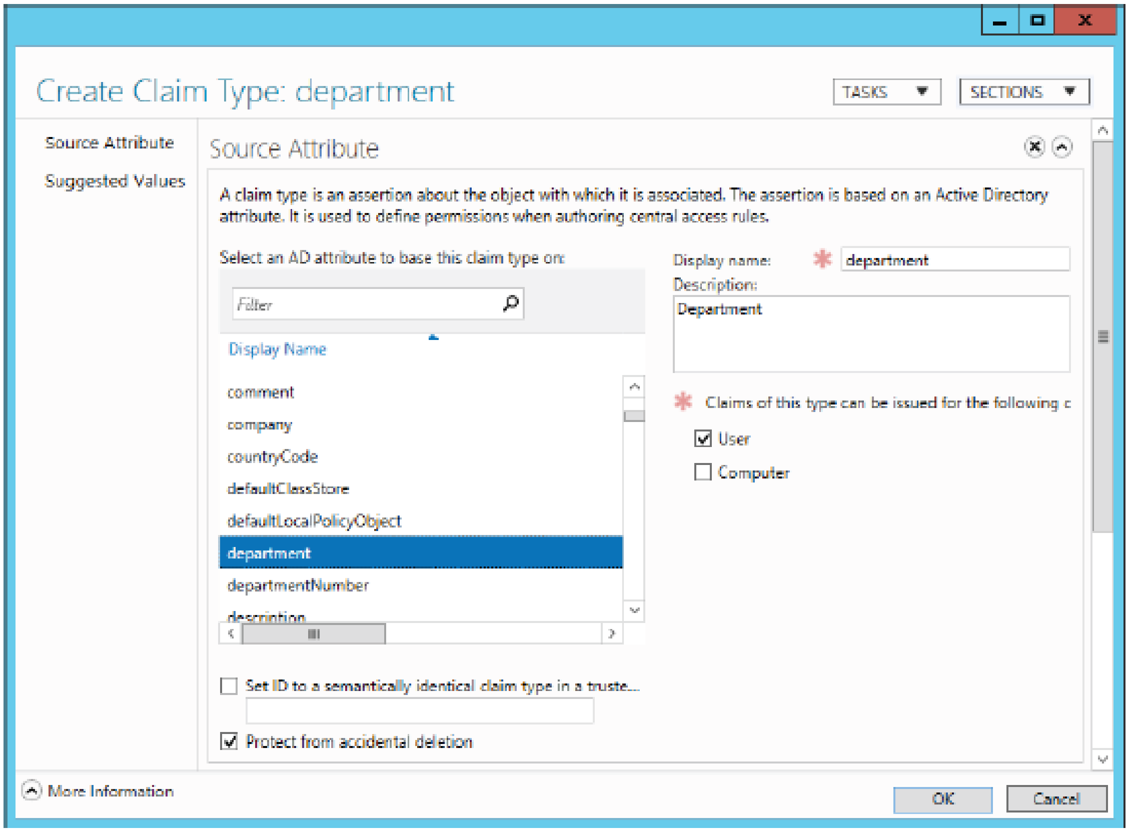


FIGURE 11-3 Creating a new claim type for a user or device.

The Create Claim Type page that opens is shown in Figure 11-4.



Create Claim Type: department

Source Attribute

A claim type is an assertion about the object with which it is associated. The assertion is based on an Active Directory attribute. It is used to define permissions when authoring central access rules.

Select an AD attribute to base this claim type on:

- comment
- company
- countryCode
- defaultClassStore
- defaultLocalPolicyObject
- department
- departmentNumber
- description

Display name: * department

Description: Department

* Claims of this type can be issued for the following c

☑ User

□ Computer

□ Set ID to a semantically identical claim type in a truste...

☑ Protect from accidental deletion

FIGURE 11-04 Creating a claim type for a user object.

In the Source Attribute section, click the Active Directory object attribute name that you want to use as the basis of the claim type. You can also specify whether you want this claim type to be issued for users, for computers (devices), or both. For example, if you plan to define rules that include references to the department of either the user or the device to which a user is signed on, you should select both User and Computer when you create the Department claim type.

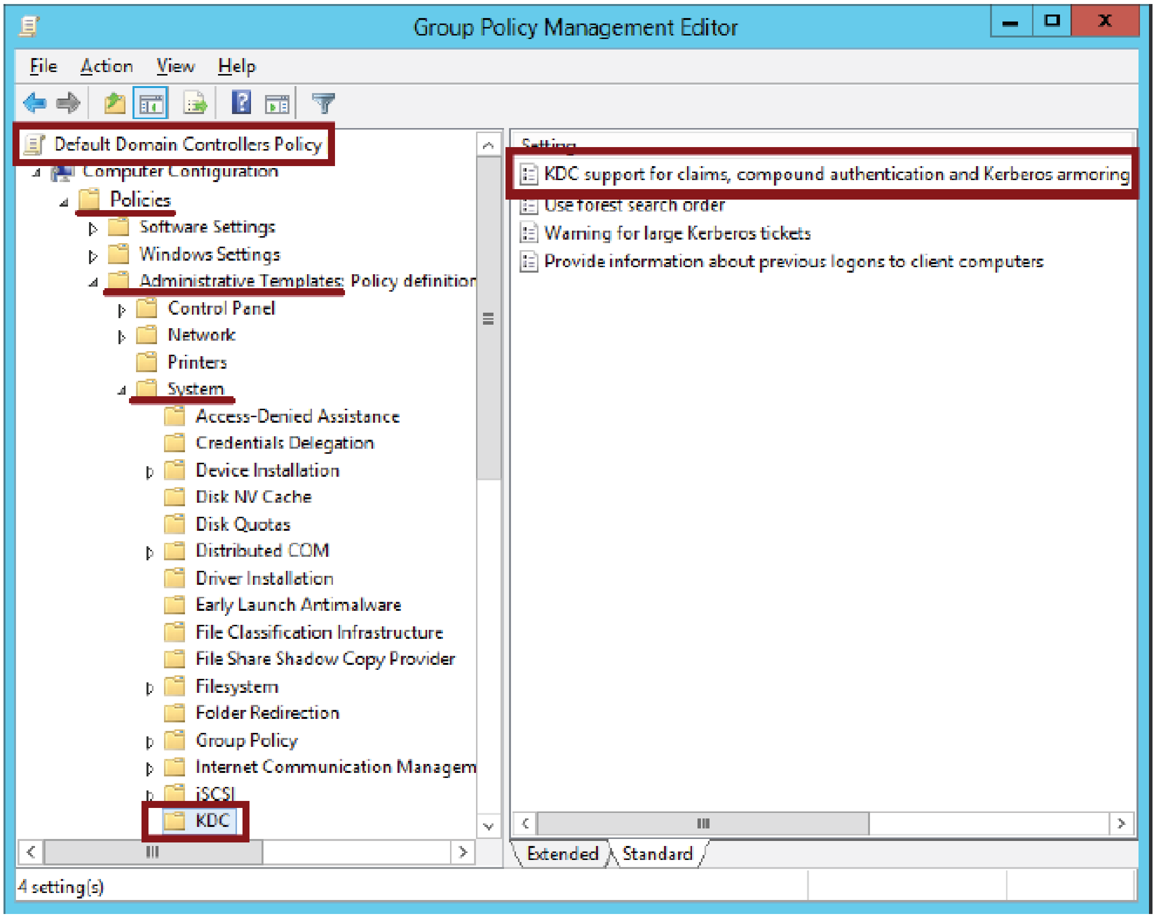
In the Suggested Values section, you can provide a list of suggested matching values that you will later be able to supply in access rules. For example, if you plan to create access rules that specify user or device Department values such as “Finance,” “Engineering,” “Operations,” “Marketing,” and “Sales,” you can precreate those same strings as suggested values now when you are creating the claim type. Note that if you define any suggested values, those values you supply will be the only ones available to select when you create rules that refer to the claim type.

Step 2: Enable Kerberos support for claims-based access control

In this step, you use Group Policy to enable Kerberos support for claims on domain controllers. This step ensures that Kerberos tokens include claims information and that this information can then be evaluated by domain controllers for access authorization.

In the Group Policy Management Console, create or edit a group policy object (GPO) linked to the Domain Controllers organizational unit (OU), and then enable the following setting: Computer Configuration/Policies/Administrative Templates/System/KDC/KDC Support For Claims, Compound Authentication, And Kerberos Armoring. (Within the Policy Setting dialog box, leave selected the default option of Supported.)

The requirement that you set this policy for claims-based authorization, and that you should do so at the Domain Controllers OU level, is one of the most likely aspects about DAC that you’ll be tested on during the 70-412 exam. Learn to recognize not only the full name of this policy, but also the possible ways its name might be shortened. (For example, an answer choice might say simply “Use Group Policy to enable Kerberos armoring on the Domain Controllers OU.”) The location of this policy setting within a GPO is shown in Figure 11-5.



KDC support for claims, compound authentication and Kerberos armoring
Use forest search order
Warning for large Kerberos tickets
Provide information about previous logons to client computers

FIGURE 11-5 Enabling Kerberos support for claims.

Configuring file classification

File classification refers to the process of adding attributes to the properties of files and folders. These attributes allow you to construct access rules that apply to these resources. Configuring file classification can be broken down into the following four steps:

1. Enable or create selected resource properties.

2. Add resource properties to a resource property list.

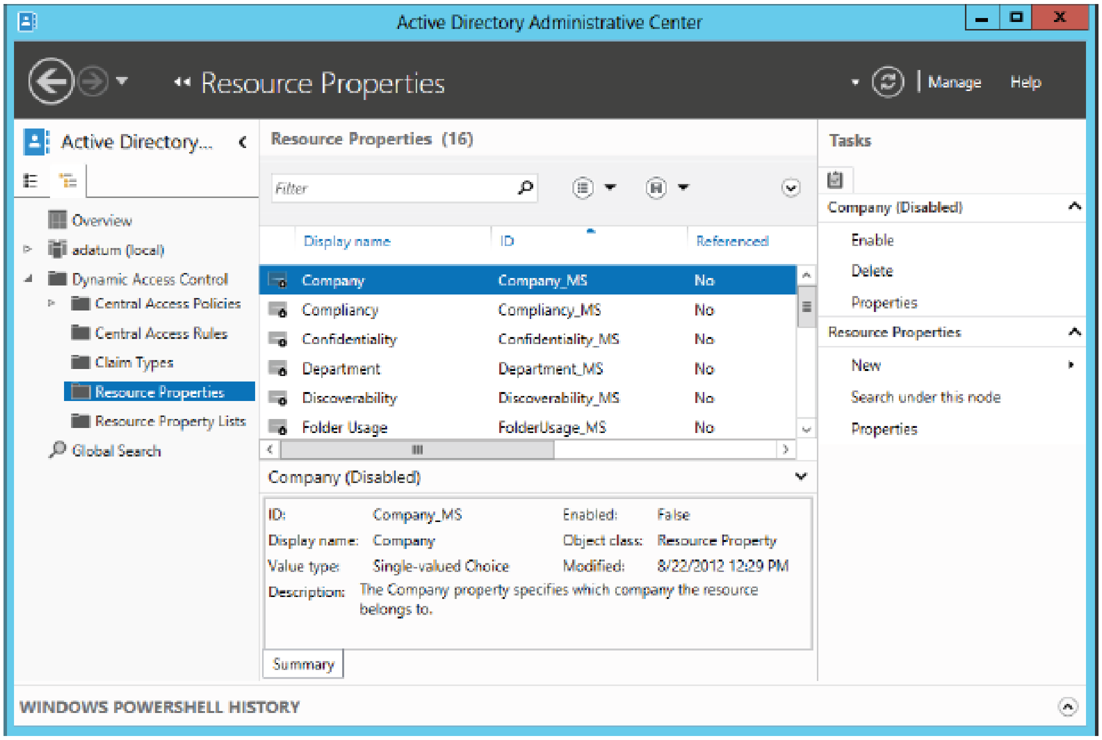
3. Update Active Directory file and folder objects.

4. Classify files and folders.

Step 1: Enable or create selected resource properties

You perform this step on a domain controller running Windows Server 2012, in Active Directory Administrative Center. In the console tree, select tree view (the right tab in the navigation pane) and then the Resource Properties container, as shown in Figure 11-6.

Resource properties correspond to attribute categories, such as Department, that you can make appear on the Classification tab of the Properties dialog box of files and folders. You make a resource property appear on this Classification tab by first enabling the property and then performing steps 2 and 3 described later in this section. Generally, you should enable only the resource properties you plan to use later in access rules. For example, if your eventual goal is to create and apply the access rule shown in Figure 11-1, you should enable the Department and Impact resource properties.



Resource Properties (16)

Display name	ID	Referenced
Company	Company_MS	No
Compliancy	Compliancy_MS	No
Confidentiality	Confidentiality_MS	No
Department	Department_MS	No
Discoverability	Discoverability_MS	No
Folder Usage	FolderUsage_MS	No

Company (Disabled)

ID:	Company_MS	Enabled:	False
Display name:	Company	Object class:	Resource Property
Value type:	Single-valued Choice	Modified:	8/22/2012 12:29 PM

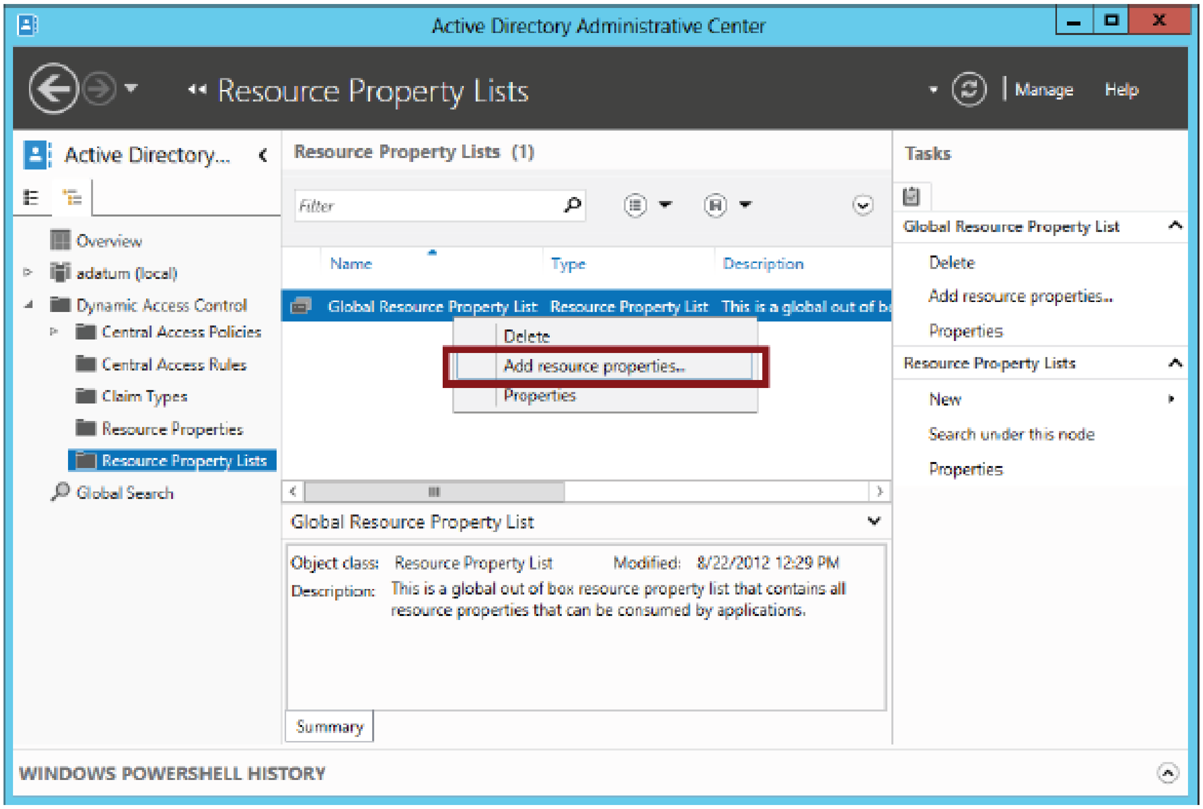
Description: The Company property specifies which company the resource belongs to.

FIGURE 11-6 Resource properties.

Windows Server 2012 includes 16 predefined resource properties, including Department, Impact, Compliancy, Intellectual Property, and Confidentiality. These resource properties include predefined suggested values you can eventually assign to objects, values such as the specific names of departments; High, Medium, or Low; and Yes or No. However, if a resource property you need isn’t predefined (such as City or Country), you can create it and define suggested values you need, such as London, New York, UK, US, and so on. Any new resource properties you create are automatically enabled.

Step 2: Add resource properties to a resource property list

After you enable your desired resource properties, you have to add them to a resource property list before they can be applied to objects. Begin by selecting the Resource Property Lists container in Active Directory Administrative Center. One predefined list is available, named Global Resource Property List. If you want the same classifications to be available for all objects, use this list. To add the resource properties you have enabled, right-click the list and select Add Resource Properties, as shown in Figure 11-7. In the Select Resource Properties dialog box that opens, add the desired resource properties that you have enabled, and click OK.



Object class: Resource Property List
Modified: 8/22/2012 12:29 PM
Description: This is a global out of box resource property list that contains all resource properties that can be consumed by applications.

FIGURE 11-7 Adding resource properties to a resource property list.

EXAM TIP

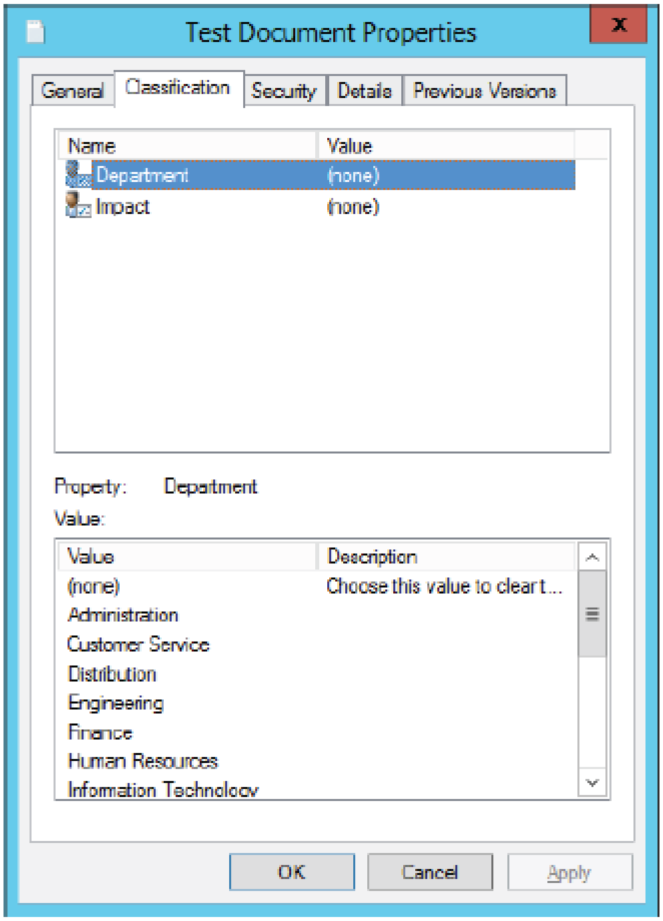
Beware of trick answer choices that suggest you need to *create* a resource property list when you configure file classification. You don't need to create a resource property list. You just need to add the resource properties to a list (usually the built-in Global Resource Property List).

Step 3: Update Active Directory file and folder objects

To update Active Directory Domain Services with the new classifiable properties, you now need to run the following cmdlet on a file server on which the File Server Resource Manager (FSRM) component of the File Server role has been installed:

After you perform this step, the resource properties you chose in step 1 appear on the Classification tab of every file and folder on that file server. The Classification tab is shown in Figure 11-8.

Note that this cmdlet is one of the most likely items related to DAC to appear on the 70-412 exam. Make sure you understand its function.



Name	Value
Department	(none)
Impact	(none)

Property: Department

Value:

Value	Description
(none)	Choose this value to clear t...
Administration	
Customer Service	
Distribution	
Engineering	
Finance	
Human Resources	
Information Technology	

FIGURE 11-8 Resource properties on the Classification tab.

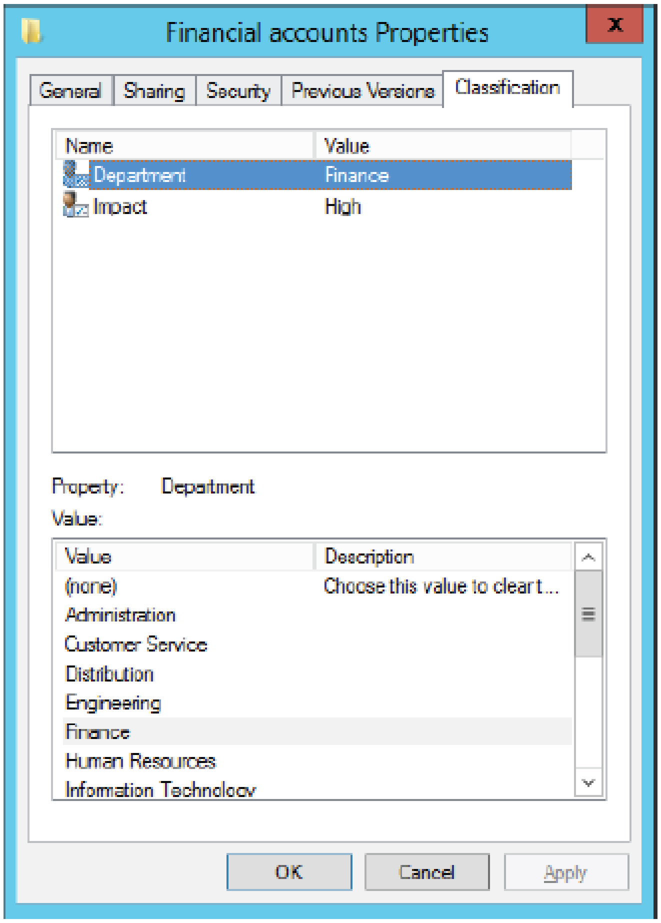
Step 4: Classify files and folders

Objects in the file structure can be classified manually or automatically. The following sections provide instructions about how to classify files by using both of these strategies.

MANUAL CLASSIFICATION

To classify file objects manually, you can select and apply a resource property value on the Classification tab directly on selected files, or on their parent folder. For example, for the folder shown in Figure 11-9, the Finance and High values have been selected for the Department and Impact properties, respectively. When you click Apply, these classifications will automatically be applied to all child objects within the folder.

Note that child objects keep these classification settings until they are reapplied. Files do not automatically inherit the values of other parent folders if they are moved into those other folders. In fact, the classifications remain applied to those objects even when you copy them from computer to computer. However, you can only see and read these classifications that have been applied to objects after you install FSRM and run the Update-FSRMClassificationPropertyDefinition cmdlet.



Name	Value
Department	Finance
Impact	High

Property: Department

Value:

Value	Description
(none)	Choose this value to clear t...
Administration	
Customer Service	
Distribution	
Engineering	
Finance	
Human Resources	
Information Technology	

FIGURE 11-9 Classification values set on a parent folder.