Directory and Resource Administrator and Exchange Administrator

Administrator Guide

June 2017

NetIQ

Legal Notice

NetIQ Directory and Resource Administrator and Exchange Administrator are protected by United States Patent No. 6,792,462.

THIS DOCUMENT AND THE SOFTWARE DESCRIBED IN THIS DOCUMENT ARE FURNISHED UNDER AND ARE SUBJECT TO THE TERMS OF A LICENSE AGREEMENT OR A NON-DISCLOSURE AGREEMENT. EXCEPT AS EXPRESSLY SET FORTH IN SUCH LICENSE AGREEMENT OR NON-DISCLOSURE AGREEMENT, NETIQ CORPORATION PROVIDES THIS DOCUMENT AND THE SOFTWARE DESCRIBED IN THIS DOCUMENT "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. SOME STATES DO NOT ALLOW DISCLAIMERS OF EXPRESS OR IMPLIED WARRANTIES IN CERTAIN TRANSACTIONS; THEREFORE, THIS STATEMENT MAY NOT APPLY TO YOU.

This document and the software described in this document may not be lent, sold, or given away without the prior written permission of NetIQ Corporation, except as otherwise permitted by law. Except as expressly set forth in such license agreement or non-disclosure agreement, no part of this document or the software described in this document may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, or otherwise, without the prior written consent of NetIQ Corporation. Some companies, names, and data in this document are used for illustration purposes and may not represent real companies, individuals, or data.

This document could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein. These changes may be incorporated in new editions of this document. NetIQ Corporation may make improvements in or changes to the software described in this document at any time.

© 2017 NetIQ Corporation. All rights reserved.

U.S. Government Restricted Rights: If the software and documentation are being acquired by or on behalf of the U.S. Government or by a U.S. Government prime contractor or subcontractor (at any tier), in accordance with 48 C.F.R. 227.7202-4 (for Department of Defense (DOD) acquisitions) and 48 C.F.R. 2.101 and 12.212 (for non-DOD acquisitions), the government’s rights in the software and documentation, including its rights to use, modify, reproduce, release, perform, display or disclose the software or documentation, will be subject in all respects to the commercial license rights and restrictions provided in the license agreement.

Check Point, FireWall-1, VPN-1, Provider-1, and SiteManager-1 are trademarks or registered trademarks of Check Point Software Technologies Ltd.

ActiveAudit, ActiveView, Aegis, AppManager, Change Administrator, Change Guardian, Compliance Suite, the cube logo design, Directory and Resource Administrator, Directory Security Administrator, Domain Migration Administrator, Exchange Administrator, File Security Administrator, Group Policy Administrator, Group Policy Guardian, Group Policy Suite, IntelliPolicy, Knowledge Scripts, NetConnect, NetIQ, the NetIQ logo, PSAudit, PSDetect, PSPasswordManager, PSSecure, Secure Configuration Manager, Security Administration Suite, Security Manager, Server Consolidator, VigilEnt, and Vivinet are trademarks or registered trademarks of NetIQ Corporation or its subsidiaries in the USA. All other company and product names mentioned are used only for identification purposes and may be trademarks or registered trademarks of their respective companies.

For purposes of clarity, any module, adapter or other similar material ("Module") is licensed under the terms and conditions of the End User License Agreement for the applicable version of the NetIQ product or software to which it relates or interoperates with, and by accessing, copying or using a Module you agree to be bound by such terms. If you do not agree to the terms of the End User License Agreement you are not authorized to use, access or copy a Module and you must destroy all copies of the Module and contact NetIQ for further instructions.

Contents

About this Book and the Library

The *Administrator Guide* provides conceptual information about the Directory and Resource Administrator (DRA) and Exchange Administrator (ExA) products. This book defines terminology and includes implementation scenarios.

Intended Audience

This book provides information for individuals responsible for understanding administration concepts and implementing a secure, distributed administration model.

Other Information in the Library

The library provides the following information resources:

Installation Guide

Provides detailed planning and installation information.

User Guide

Provides conceptual information about DRA and ExA. This book also provides an overview of the user interfaces and step-by-step guidance for many administration tasks.

Trial Guide

Provides product trial and evaluation instructions and a product tour.

Help

Provides context-sensitive information and step-by-step guidance for common tasks, as well as definitions for each field on each window.

About NetIQ Corporation

We are a global, enterprise software company, with a focus on the three persistent challenges in your environment: Change, complexity and risk—and how we can help you control them.

Our Viewpoint

Adapting to change and managing complexity and risk are nothing new

In fact, of all the challenges you face, these are perhaps the most prominent variables that deny you the control you need to securely measure, monitor, and manage your physical, virtual, and cloud computing environments.

Enabling critical business services, better and faster

We believe that providing as much control as possible to IT organizations is the only way to enable timelier and cost effective delivery of services. Persistent pressures like change and complexity will only continue to increase as organizations continue to change and the technologies needed to manage them become inherently more complex.

Our Philosophy

Selling intelligent solutions, not just software

In order to provide reliable control, we first make sure we understand the real-world scenarios in which IT organizations like yours operate — day in and day out. That's the only way we can develop practical, intelligent IT solutions that successfully yield proven, measurable results. And that's so much more rewarding than simply selling software.

Driving your success is our passion

We place your success at the heart of how we do business. From product inception to deployment, we understand that you need IT solutions that work well and integrate seamlessly with your existing investments; you need ongoing support and training post-deployment; and you need someone that is truly easy to work with — for a change. Ultimately, when you succeed, we all succeed.

Our Solutions

- Identity & Access Governance
- Access Management
- Security Management
- Systems & Application Management
- Workload Management
- Service Management

Contacting Sales Support

For questions about products, pricing, and capabilities, contact your local partner. If you cannot contact your partner, contact our Sales Support team.

Worldwide:	www.netiq.com/about_netiq/officelocations.asp
United States and Canada:	1-888-323-6768
Email:	info@netiq.com
Web Site:	www.netiq.com

Contacting Technical Support

For specific product issues, contact our Technical Support team.

Worldwide:	www.netiq.com/support/contactinfo.asp
North and South America:	1-713-418-5555
Europe, Middle East, and Africa:	+353 (0) 91-782 677
Email:	support@netiq.com
Web Site:	www.netiq.com/support

Contacting Documentation Support

Our goal is to provide documentation that meets your needs. The documentation for this product is available on the NetIQ web site in HTML and PDF formats on a page that does not require you to log in. If you have suggestions for documentation improvements, click **comment on this topic** at the bottom of any page in the HTML version of the documentation posted at www.netiq.com/documentation. You can also email Documentation-Feedback@netiq.com. We value your input and look forward to hearing from you.

Contacting the Online User Community

NetIQ Communities, the NetIQ online community, is a collaborative network connecting you to your peers and NetIQ experts. By providing more immediate information, useful links to helpful resources, and access to NetIQ experts, NetIQ Communities helps ensure you are mastering the knowledge you need to realize the full potential of IT investments upon which you rely. For more information, visit http://community.netiq.com.

1 Introducing DRA and ExA

NetIQ Enterprise Administration solutions provide enterprise customers with the ability to safely and securely delegate administrative privileges across their Windows server, Active Directory, Group Policy, Office 365, Skype, and Exchange server environments. Combined with detailed auditing of and reporting on administrative activities, NetIQ Enterprise Administration solutions provide organizations with unprecedented levels of accountability while reducing the costs associated with daily operations, internal policy, and regulatory compliance activities.

Organizations have increasingly relied upon Active Directory for the central management of identities and for the authentication and authorization of those identities to the network and IT services. With the introduction of Office 365, organizations have come to depend more and more on maintaining a single point of control over on-premises and cloud directory and messaging administration. However, assuring the security, availability and integrity of Active Directory and Office 365 requires more than just delegating permissions or changing group memberships. IT Governance and auditors also require proof that policies and procedures are enforced, that changes are tracked, and that administrators are not able to manage beyond the scope of their responsibilities.

NetIQ Directory and Resource Administrator (DRA) delivers an unparalleled ability to control who can manage what within Active Directory and Office 365 while protecting the consistency and integrity of its information by validating all administrative changes. Through granular delegation of permissions, robust change management policies, and automation that simplifies workflows, DRA reduces down time and operational risks to Active Directory and Office 365 that are posed by the consequences of malicious or accidental changes.

NetIQ Exchange Administrator (ExA) extends the powerful features of DRA to provide seamless management of Microsoft Exchange. Through a single, common user interface, ExA delivers policy-based administration for the management of directories, mailboxes and distribution lists across your Microsoft Exchange environment.

Together, DRA and ExA provide the solutions you need to control and manage your Active Directory, Microsoft Windows, and Microsoft Exchange environments.

Key benefits of DRA include:

Policy and regulation compliance

Involves the assessment, operation, and control of systems and resources in accordance with security standards, best practices, and regulatory requirements and provides logging and auditing capabilities that help demonstrate compliance.

Operational integrity

Prevents malicious or incorrect changes that affect the performance and availability of systems and services by providing granular access control for administrators and managing access to systems and resources.

Process enforcement

Maintains the integrity of key change management processes that help you improve productivity, reduce errors, save time, and increase administration efficiency.

What are DRA and ExA?

DRA and ExA are comprehensive account and resource management products for the key Microsoft identity and messaging platforms, Active Directory, Office 365, and Exchange. Using a flexible, rules-based management model, both DRA and ExA deliver capabilities that streamline administration, increase security, assure operational integrity, and ease the challenges of regulatory compliance for your Active Directory, Office 365, and Microsoft Exchange messaging environments.

An enterprise-scale directory and resource management product, DRA controls and manages Active Directory and Office 365 administration. Its powerful policy-based management, coupled with its safe, distributed administration, dramatically reduces administration efforts and costs. DRA provides increased data security while protecting the integrity of your Active Directory and Office 365 content.

ExA extends the power and flexibility of DRA to include Microsoft Exchange management. Within the context of account administration, you can manage mailboxes, Microsoft Exchange permissions, contacts, and distribution lists. DRA and ExA provide a single, integrated solution for controlling and managing complex IT environments.

What DRA and ExA Provide

DRA and ExA allow you to manage your enterprise within the context of a dynamic security model. This model ensures that your enterprise management and security remains current as your enterprise changes and evolves.

DRA and ExA provide advanced delegation and robust, policy-based administration features that improve the security and efficiency of your Microsoft Windows environment. They provide a secure, integrated administration solution for the following environments:

Environment	Supported Versions		
Microsoft Windows Server Active Directory	2012	2012 R2	2016
Microsoft Exchange Server	2013	2016	Microsoft Exchange Online

DRA and ExA offer significant flexibility using patented ActiveView technology and granular delegation. An ActiveView is a dynamic set of objects, such as user accounts or computers, that you want an administrator to collectively manage. ActiveViews can include or exclude objects from multiple domains, OUs, and groups into virtual containers for easy administration. With ActiveViews, administrators only sees the objects they can manage, without exposing them to the other objects present across the managed environment.

Granular delegation lets you securely distribute specific tasks, such as resetting a user password or modifying Microsoft Exchange mailbox rights. The flexibility of ActiveViews helps eliminate many of the problems associated with managing data in difficult-to-change, hierarchical structures.

DRA and ExA also help you assure compliance with internal policies and with regulatory requirements. For example, DRA offers dual-key security, so you can require two people to independently confirm portions of the same workflow. You can delegate one administrator to send a user account to the Recycle Bin, and another administrator to review the action and either approve the decision or revoke the change. DRA provides additional reports, logging, and auditing capabilities to help you demonstrate compliance with policies and with regulatory requirements.

With the Web Console, DRA and ExA provide out-of-the-box relief where you want to delegate administrative tasks, but do not want to deploy the product console. For example, you may want employees to manage their personal information, or provide limited privileges to a Help Desk organization. This easy-to-use, task-based interface significantly reduces administration time and lets you securely delegate specific tasks without additional training. You can quickly and easily customize the scope of the administration tasks you want to make available from the Web Console.

These technologies seamlessly join and manage data from multiple sources across your enterprise, including Active Directory, Office 365, Microsoft Exchange, and computer resources. To further expand these benefits, DRA and ExA let you apply policies to directory updates that can extend beyond the directory itself to other applications and databases, making the task of enterprise management easy.

DRA lets you define administration policies that it then automatically propagates and enforces for all DRA users, increasing security and reducing administration costs. This model is dynamic, so as your enterprise changes, objects inherit the appropriate level of security.

DRA and ExA help you automate and streamline many routine administration tasks, such as creating a user account and home share for a new employee. While many automated Active Directory administration tasks are provided out-of-the-box, you can also extend DRA and ExA using well-known standard interfaces such as the Active Directory Service Interfaces (ADSI) and Windows Terminal Server (WTS). DRA and ExA also provide tools, such as DRA PowerShell modules, a DRA REST API, automation triggers, and the DRA Software Development Kit (SDK), so you can integrate enterprise administration with your current business systems.

DRA supports the 64-bit platform, which provides you with increased scalability, increased performance, reduced query time, and more effective use of memory.

Using state-of-the-art technology, these products provide the features you need to create a more secure, productive, and manageable Active Directory and Microsoft Exchange environment.

2 Understanding and Implementing Your Security Model

Understanding the Dynamic Security Model

DRA and ExA allow you to manage your enterprise within the context of a dynamic security model. This model ensures that your enterprise management and security remains current as your enterprise changes and evolves. Dynamic security allows you to control the issues you have today while providing a stable foundation for future solutions.

What Is Distributed Administration?

Distributed administration allows you to delegate specific authority to one or more people. With distributed administration, you can easily and safely grant powers over a set of objects regardless of your domain or organizational unit (OU) structure.

When you use distributed administration, you create rules that define *who* can do *what* to *whom* or *what*.

who

Assistant Admins (AAs) represent one or more user accounts, groups, or AA groups. AAs manage the user accounts, groups, contacts, OUs, and resources in an ActiveView.

what

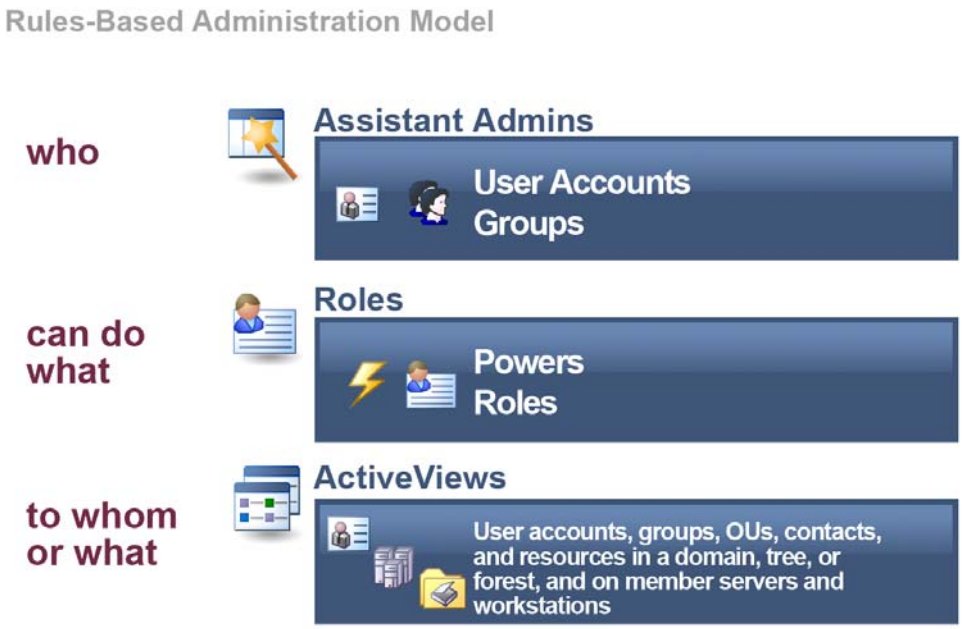
Roles and powers represent subsets of administration authority that you want to grant to AAs.

whom or what

ActiveViews represent a set of objects your AAs can manage collectively.

These rules establish and control your security model. When using a dynamic security model, these rules provide the flexibility you need to automatically maintain security while your enterprise changes.

The following figure illustrates this rules based model.



What Is the Dynamic Security Model?

The dynamic security model provides powerful and flexible rules based management for your enterprise. This model enhances distributed administration by accommodating change. Rather than delegating a power to a person, you are delegating roles to people. In this way, the dynamic security model more closely represents how your company works, letting you design your enterprise security to support workflows across organizations. You can expand this flexibility by configuring rules that match established naming conventions or group memberships.

In the dynamic security model, roles can be reused in different ActiveViews and assigned to different AAs, user accounts can be moved from one AA group to another, and powers can be moved from one role to another. As these changes occur, DRA and ExA automatically update security settings across your enterprise.

When you develop your administration and security plan, first identify the workflows your company has. Each workflow determines *who* can do *what* to *whom* or *what*. Use your workflow definitions to create the appropriate AA groups, assign the required roles, and configure the corresponding ActiveViews.

Roles

Roles are the *what* in the dynamic security model. A role is a set of powers that provide the permissions required to perform a specific administration task, such as creating a user account or moving shared directories. To create a role, first define the job description. The job description provides the list of powers an Assistant Admin needs to perform a task or complete a workflow.

A role can contain any set of powers you specify. Because you can choose from hundreds of powers, you have the flexibility to create roles that best fit your organization. You can also use the pre-defined built-in roles that are installed with the product. For more information about built-in roles, see “Understanding the Default Security Model” on page 21.

When you add a power to a role, any Assistant Admin associated with that role automatically receives the new power. You can add roles to other roles, creating a hierarchical model based on increasing power. You can also associate the same role to different Assistant Admin groups.

Roles by themselves do not grant power. To delegate power, you must associate the role with an Assistant Admin in an ActiveView.

Powers

A power defines the properties of an object an Assistant Admin can view, modify, or create in your managed domain or subtree. A group of powers forms a role. DRA allows you to group powers into roles and delegate the roles and powers to users, group accounts, and Assistant Admin groups.

The Delegation and Management taskpad allows you to list built-in powers, clone and create new powers, assign powers to ActiveViews and Assistant Admin groups, change properties of custom powers, and view all powers. Roles are created from the inclusion of multiple powers.

ActiveViews

ActiveViews are the *whom or what* in the dynamic security model. An ActiveView represents a set of objects. When you create or modify an ActiveView, you specify rules that define which objects you want to manage as a collection. The ActiveView rule also associates AAs with roles and powers to manage this collection of objects.

What ActiveViews Provide

ActiveViews allow you to implement a security model that has the following features:

- Is independent from your Active Directory structure
- Allows you to assign powers and define policies that correlate to your existing workflows
- Provides automation to help you further integrate and customize your enterprise
- Dynamically responds to change

An ActiveView represents a set of objects within one or more managed domains. You can include an object in more than one ActiveView. You can also include many objects from multiple domains or OUs.

ActiveViews Include Dynamic Sets of Objects

An ActiveView provides real time access to specific objects within one or more domains or OUs. You can add or remove objects from an ActiveView without changing the underlying domain or OU structure.

You may think of an ActiveView as a virtual domain or OU, or the results of a select statement or database view for a relational database. ActiveViews can include or exclude any set of objects, contain other ActiveViews, and have overlapping contents. ActiveViews can contain objects from different domains, trees, and forests. You can configure ActiveViews to meet any enterprise management need.

ActiveViews can include the following types of objects:

- User accounts
- Direct reports

- Groups
- Managed groups
- Organizational units
- Contacts
- Computers
- Resources, such as:
  - printers
  - print jobs
  - published printers
  - published printer print jobs
  - open files
  - connected users
  - event logs
  - shares
  - devices
  - services
- Domains
- Other ActiveViews
- Resource mailboxes
- Shared mailboxes
- Exchange Dynamic Groups
- Member servers

You specify which objects DRA will include in an ActiveView by querying the object attributes, much as you would make a query to select items from a database. As your enterprise changes or grows, ActiveViews change to include or exclude the new objects. Thus, you can use ActiveViews to reduce the complexity of your model, provide the security you need, and give you far more flexibility than other enterprise organizing tools.

ActiveViews Include Flexible Rules

An ActiveView can consist of rules that include or exclude objects such as user accounts, groups, OUs, contacts, resources, computers, resource mailboxes, shared mailboxes, dynamic distribution groups, and ActiveViews. In addition, ActiveView rules can specify security and policy objects. When you specify a wildcard character in a rule specification, the rule includes all objects that match the specified value. This flexibility makes ActiveViews dynamic.

These matches are called **wildcards**. When you add a rule to an AA group using wildcards, the rule includes all computer accounts that match the specified pattern. For example, you can define a rule to include all computers with names matching `DOM*`. This wildcard specification will search for any computer account whose name begins with the character string DOM. Wildcard matching makes administration dynamic because accounts are automatically included when they match the rule. Thus, when you use wildcards, you do not need to reconfigure the ActiveViews as your organization changes.

Another example is defining ActiveViews based on group membership. You can define a rule that includes all members of groups that begin with the letters NYC. Then, as members are added to any group matching this rule, these members are automatically included in this ActiveView. As your enterprise changes or grows, DRA reapplies the rules to include or exclude the new objects in the proper ActiveViews.

ActiveViews and Distributed Administration

ActiveViews help you distribute administration tasks to specific people. For example, to allow members of the Atlanta Help Desk group to reset passwords and unlock accounts for users in Atlanta, an administrator at the Houston headquarters defines the following rules:

Atlanta Help Desk Assistant Admins

Rule for this AA group includes all the employees in the Atlanta Help Desk group.

Atlanta User Accounts ActiveView

Rules for this ActiveView include all the user accounts in Atlanta and associate the Reset User Passwords role with the Atlanta Help Desk AAs. By distributing administration through the Atlanta User Accounts ActiveView, the Houston administrator achieves the following benefits:

Improved service to users in Atlanta

Users in Atlanta no longer need to call Houston if they forget their password. Any time a new user account is added in Atlanta, the user account is included automatically in the Atlanta User Accounts ActiveView. Atlanta Help Desk Assistant Admins can now reset passwords for this new user account.

Reduced workload for the central administrators in Houston

Atlanta Help Desk Assistant Admins can reset passwords for users in the Atlanta User Accounts ActiveView while the Houston administrators focus their attention on other issues.

Enhanced security for the corporation

Atlanta Help Desk Assistant Admins are in a better position to determine whether a request to reset a password from a user in Atlanta is valid. Therefore, distributing this portion of the administration workload permits the corporation to run with fewer Microsoft Windows administrators, maintaining a higher level of security.

Reset User Passwords Role

Rules for this role include all the powers needed to unlock user accounts and reset passwords.

In this way, DRA allows you to manage your organization the way you think and work. Once you establish your own security model using ActiveViews, the model can grow and change as your organization does. For more information about implementing help desk administration, see the *Product Overview Guide*.

How the Administration Server Processes Requests

When the Administration server receives a request for an action, such as changing a user password, it uses the following process:

1. Search all ActiveViews for the object of this action.
2. Validate the powers assigned to the account that is requesting the action.
3. ***If the account has the correct power***, the Administration server allows the action to be performed.

If the account does not have the correct power, the Administration server returns an error.

4 Update the Active Directory.

For example, when you attempt to set a new password for the JSmith user account, the Administration server finds all ActiveViews that include JSmith. This search looks for any ActiveView that specifies JSmith directly, through a wildcard rule, or through group membership. If an ActiveView includes other ActiveViews, the Administration server also searches these additional ActiveViews. The Administration server determines whether you have the Reset User Account Password power in any of these ActiveViews. If you have the Reset User Account Password power, the Administration server resets the password for JSmith. If you do not have this power, the Administration server denies your request.

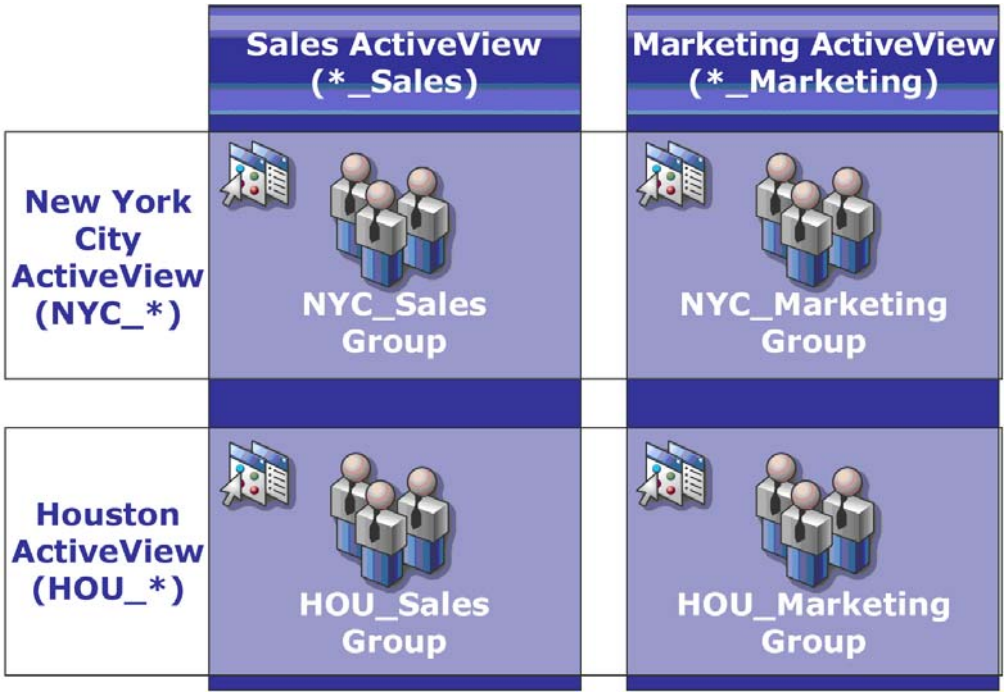
How Powers Can Increase

A power defines the properties of an object an Assistant Admin can view, modify, or create in your managed domain or subtree. More than one ActiveView can include the same object. This configuration is called **overlapping ActiveViews**.

When ActiveViews overlap, you can accumulate a set of different powers over the same objects. For example, if one ActiveView allows you to add a user account to a domain and another ActiveView allows you to delete a user account from the same domain, you can add or delete user accounts in that domain. In this way, the powers you have over a given object are cumulative.

Understanding How Powers Increase

It is important to understand how ActiveViews can overlap and you can have increased powers over objects included in these ActiveViews. Consider the ActiveView configuration illustrated in the following figure.



The white tabs identify ActiveViews by location, *New York City* and *Houston*. The black tabs identify ActiveViews by their organizational function, *Sales* and *Marketing*. The cells show the groups included in each ActiveView.

The NYC_Sales group and the HOU_Sales group are both represented in the Sales ActiveView. If you have power in the Sales ActiveView, then you can manage any member of the NYC_Sales and HOU_Sales groups. If you also have power in the New York City ActiveView, then these additional powers apply to the NYC_Marketing group. In this way, powers accumulate as the ActiveViews overlap.

Overlapping ActiveViews can provide a powerful, flexible security model. However, this feature can also have unintended consequences. Carefully plan your ActiveViews to ensure each AA has only the powers you intend over each user account, group, OU, contact, or resource.

Groups in Multiple ActiveViews

In this example, the NYC_Sales group is represented in more than one ActiveView. The members of the NYC_Sales group are represented in the New York City ActiveView because the group name matches the `NYC_*` ActiveView rule. The group is also in the Sales ActiveView because the group name matches the `*_Sales` ActiveView rule. By including the same group in multiple ActiveViews, you can allow different AAs to manage the same objects differently.

Using Powers in Multiple ActiveViews

Assume there is an AA, JSmith, who has the Modify General User Properties power in the New York City ActiveView. This first power allows JSmith to edit all the properties on the General tab of a user properties window. JSmith has the Modify User Profile Properties power in the Sales ActiveView. This second power allows JSmith to edit all the properties on the Profile tab of a user properties window.

The following figure indicates the powers JSmith has for each group.

JSmith has the following powers:

- General Properties in the NYC_* ActiveView
- Profile Properties in the *_Sales ActiveView

The power delegation in these overlapping ActiveViews allows JSmith to modify the General and Profile properties of the NYC_Sales group. Thus, JSmith has all the powers granted in all the ActiveViews that represent the NYC_Sales group.

Extending Powers

You can add permissions or functionality to a power by extending that power.

For example, to allow an AA to create a user account, you can assign either the Create User and Modify All Properties power or the Create User and Modify Limited Properties power. If you also assign the Add New User to Group power, the AA can add this new user account to a group while using the Create User wizard. In this case, the Add New User to Group power provides an additional wizard feature. The Add New User to Group power is the **extension power**.

Extension powers cannot add permissions or functionality by themselves. To successfully delegate a task that includes an extension power, you must assign the extension power along with the power you want to extend.

NOTE

- To successfully create a group and include the new group in an ActiveView, you must have the Add New Group to ActiveView power in the specified ActiveView. The specified ActiveView must also include the OU or built-in container that will contain the new group.
- To successfully clone a group and include the new group in an ActiveView, you must have the Add Cloned Group to ActiveView power in the specified ActiveView. The specified ActiveView must also include the source group as well as the OU or built-in container that will contain the new group.

The following table lists the administration tasks that include extension powers.

To Delegate This Task	Assign This Power	And This Extension Power
Clone a group and include the new group in a specified ActiveView	Clone Group and Modify All Properties	Add Cloned Group to ActiveView
Create a group and include the new group in a specified ActiveView	Create Group and Modify All Properties	Add New Group to ActiveView
Create a mail enabled contact	Create Contact and Modify All Properties Create Contact and Modify Limited Properties	Enable Email for New Contact
Create a mail enabled group	Create Group and Modify All Properties	Enable Email for New Group
Create a mail enabled user account	Create User and Modify All Properties Create User and Modify Limited Properties	Enable Email for New User
Create a user account and add the new account to specific groups	Create User and Modify All Properties Create User and Modify Limited Properties	Add New User to Group

Understanding the Default Security Model

DRA and ExA extend your existing native Microsoft Windows security model. For example, DRA uses your existing group memberships to define default permissions. You can meet your specific needs by customizing and extending these permissions across the organization.

The default DRA and ExA security model provides built-in roles, AA groups, policy, and ActiveViews so that you can quickly incorporate DRA and ExA into your current security model. Through the default security model, your team can start using DRA out of the box with little or no additional configuration.

This section describes the key concepts about the default security model and illustrates these concepts with examples and scenarios. These examples and scenarios assume you have the DRA Admin role or the corresponding powers. Review these sections to learn about the following concepts:

- How to extend your security model with DRA and ExA
- How to use the built-in ActiveViews, AA groups, and roles

What Is the Default Security Model?

The default security model consists of several built-in ActiveViews, AA groups, and roles. These built-in components let you immediately manage domain objects and customize the Administration server. With these default definitions and rules, you can quickly start planning and implementing your enterprise management model.

You can use the available built-in components as the basis for your own security model. You can easily define ActiveViews and associate the built-in roles with AA groups you create to delegate administration powers within your enterprise. For information about implementing a security model, see “Implementing Your Dynamic Security Model” on page 34.

What Built-in Security Provides

Built-in security provides immediate and secure access to your domains, objects, and policies. The built-in ActiveViews, AA groups, and roles allow you to extend your existing security model. You can start using DRA and ExA to manage your enterprise without redesigning your existing security.

These built-in components provide a starting point. Before you define an ActiveView, decide if you can use one of the built-in roles or if you need to define a new role. When you specify which objects the ActiveView includes, you can quickly associate AAs with roles or powers for this ActiveView. These built-in roles and ActiveViews allow you to begin work immediately, using the full capabilities of DRA and ExA.

For example, members of the Administrators group in the managed domain are automatically empowered with the DRA Administration role in the Objects Current User Manages as Windows Administrator ActiveView. This ensures that Microsoft Windows administrators can start DRA and ExA with the same permissions they have using native tools.

All Powers for DRA Admins

You can grant any group or user account all powers across the enterprise by delegating the following built-in security objects:

- DRA Admins AA group
- DRA Administration role
- All Objects ActiveView

The following table describes the relationship between these security objects.

Object Name	Object Type	Description
DRA Admins	AA group	Includes the user account or group you specified during setup
DRA Administration	Role	Includes all powers
All Objects	ActiveView	Includes all user accounts, groups, resources, contacts, OUs, and Microsoft Exchange mailboxes from all managed domains

With this association, all members of the built-in DRA Admins AA group have all powers for all directory objects across the enterprise.

Domain Powers for Administrators

To grant members of the native Administrators group all powers in domains where they are Administrators, DRA provides the following built-in security objects:

- Administrators from Managed Domains AA group
- DRA Administration role
- Objects Current User Manages as Windows Administrator ActiveView

The following table describes the relationship between these security objects.

Object Name	Object Type	Description
Administrators from Managed Domains	AA group	All members of the native Administrators group for the managed domain
DRA Administration	Role	Includes all powers
Objects Current User Manages as Windows Administrator	ActiveView	Includes all user accounts, groups, resources, contacts, OUs, and Microsoft Exchange mailboxes in managed domains where the AA is an Administrator

With this association, all members of the native Administrators group have all powers for accounts, resources, and mailboxes in the managed domain.

Built-in Delegations

By default, DRA delegates the built-in ActiveViews and roles to specific AA groups. The following table lists the built-in ActiveViews and identifies the built-in AA groups and roles associated with each ActiveView.

Built-in ActiveView	Built-in AA group	Assigned role
All Objects	DRA Admins	DRA Administration
Objects Current User Manages as Windows Administrator	Administrators from Managed Domains	DRA Administration
Administration Servers and Managed Domains	DRA Configuration Admins	Configure Servers and Domains
DRA Policies and Automation Triggers	DRA Policy Admins	Manage Policies and Automation Triggers
DRA Security Objects	DRA Security Model Admins	Manage Security Model
SPA Users from All Managed and Trusted Domains	SPA Admins	Reset Password and Unlock Using SPA

Understanding Built-in ActiveViews

Built-in ActiveViews are the default ActiveViews provided by DRA and ExA. These ActiveViews represent all current objects and security settings. Thus, built-in ActiveViews provide immediate access to all your objects and settings as well as the default security model. You can use these ActiveViews to manage objects, such as user accounts and resources, or apply the default security model to your current enterprise configuration.

Built-in ActiveViews

DRA and ExA provide several built-in ActiveViews that can represent your security model. The built-in ActiveView node contains the following ActiveViews:

All Objects

Includes all objects in all managed domains. Through this ActiveView, you can manage any aspect of your enterprise. Assign this ActiveView to the administrator or to an AA who needs auditing powers across the enterprise.

Objects Current User Manages as Windows Administrator

Includes objects from the current managed domain. Through this ActiveView, you can manage user accounts, groups, contacts, OUs, and resources. Assign this ActiveView to native Administrators who are responsible for account and resource objects in the managed domain.

Administration Servers and Managed Domains

Includes Administration server computers and managed domains. Through this ActiveView, you can manage the daily maintenance of your Administration servers. Assign this ActiveView to AAs whose duties include monitoring the synchronization status or performing cache refreshes.

DRA Policies and Automation Triggers

Includes all policy and automation trigger objects in all managed domains. Through this ActiveView, you can manage policy properties and scope, as well as automation trigger properties. Assign this ActiveView to AAs responsible for creating and maintaining your company policies.

DRA Security Objects

Includes all security objects. Through this ActiveView, you can manage ActiveViews, AA groups, and roles. Assign this ActiveView to AAs responsible for creating and maintaining your security model.

SPA Users from All Managed and Trusted Domains

Includes all user accounts from managed and trusted domains. Through this ActiveView, you can manage password of the users.

Accessing Built-in ActiveViews

Access built-in ActiveViews to audit the default security model or manage your own security settings.

To access built-in ActiveViews:

1. In the left pane, select **Delegation Management**.
2. Under Common Tasks in the right pane, click **Manage ActiveViews**.
3. Select the appropriate ActiveView.

Using Built-in ActiveViews

You cannot delete, clone, or modify built-in ActiveViews. However, you can incorporate these ActiveViews into your existing security model or use these ActiveViews to design your own model.

You can use built-in ActiveViews in the following ways:

- Assign the individual built-in ActiveViews to the appropriate AA groups. This association allows the AA group members to manage the corresponding set of objects with the appropriate powers.
- Refer to the built-in ActiveView rules and associations as guidelines towards designing and implementing your security model.

For more information about designing a dynamic security model, see “Implementing Your Dynamic Security Model” on page 34.

Understanding Built-in Assistant Admin Groups

Built-in AA groups provide immediate access to a set of commonly used roles. You can extend your current security configuration by using these default groups to delegate power to specific user accounts or other groups.

Most built-in AA groups do not include any members. Use these groups to quickly let the appropriate people manage objects in the built-in ActiveViews. For example, if you add the AtlantaAdmins group to the DRA Security Model Admins AA group, members of the AtlantaAdmins group can create and modify all the rules that define the administration model. If you add the HoustonAdmins group to the built-in DRA Policy Admins AA group, members of the HoustonAdmins group can create and modify all policies, such as user account naming conventions. A member of the built-in DRA Policy Admins AA group can also create and modify automation triggers.

These groups are already associated with the corresponding built-in role so their members can perform common administration tasks. For example, members of the Administrators from Managed Domains AA group can manage objects in domains where they are administrators. Because built-in AA groups are part of the default security model, you can use the built-in AA groups to quickly delegate power and implement security.

Built-in Assistant Admin Groups

DRA and ExA provide several built-in AA groups that you can use in your security model. The following list describes each built-in AA group and discusses the AAs typically associated with that group. For more information, see “Understanding Built-in ActiveViews” on page 24 and “Understanding Built-in Roles” on page 27.

DRA Admins

Allows AAs to manage all objects in your managed domain, including the Administration servers, and maintain your security model. By default, the DRA Admins group includes the account or group you specified during setup. Add AAs to this group if they are responsible for managing all aspects of your enterprise.

Administrators from Managed Domains

Allows AAs to manage all user accounts, groups, contacts, OUs, and resources for the domains in which they are administrators. By default, the Administrators from Managed Domains group includes the native Administrators group.

DRA Configuration Admins

Allows AAs to configure the Administration servers and managed domains. This group also allows AAs to create custom user interface extensions and custom tools, manage file replication between Administration servers and DRA client computers, and specify clone exceptions to use when cloning user accounts. By default, the DRA Configuration Admins group includes the Administration server service account. Add AAs to this group if they are responsible for configuring and maintaining your Administration servers, such as performing accounts cache refreshes or server synchronization.

DRA Policy Admins

Allows AAs to manage policies and automation triggers for all managed domains. By default, the DRA Policy Admins group does not have members. Add AAs to this group if they are responsible for establishing and maintaining policies and automating workflows.

DRA Security Model Admins

Allows AAs to manage security objects such as other AA groups, roles, and ActiveViews. By default, the DRA Security Model Admins group does not have members. Add AAs to this group if they are responsible for establishing and maintaining your security model.

SPA Admins

Allows AAs to manage password of the users. It also allows AAs to reset passwords and unlock user accounts. By default, the SPA Admins group does not have members. Add AAs to this group if they are responsible for managing passwords.

Accessing Built-in Assistant Admin Groups

Access built-in AA groups to audit the default security model or manage your own security settings.

To access built-in AA groups:

1. In the left pane, select **Delegation Management**.
2. Under Common Tasks in the right pane, click **Manage Assistant Admins**.
3. Select the appropriate AA group.

Using Built-in Assistant Admin Groups

You cannot delete or clone built-in AA groups. However, you can incorporate the built-in AA groups into your existing security model or use these groups to design and implement your own model.

You can use built-in AA groups in the following ways:

- Add user accounts or other groups to a built-in AA group. These new members are then empowered with built-in roles in the ActiveViews associated with the AA group.
- Associate a built-in AA group with an ActiveView. This association allows the AA group members to manage a specific set of objects.

For more information about designing a dynamic security model, see “Implementing Your Dynamic Security Model” on page 34.

Understanding Built-in Roles

Built-in AA roles provide immediate access to a set of commonly used powers. You can extend your current security configuration by using these default roles to delegate power to specific user accounts or other groups.

These roles contain the powers required to perform common administration tasks. For example, the DRA Administration role contains all the powers required to manage objects. To use these powers, however, the role must be associated with a user account or an AA group and the managed ActiveView.

Because built-in roles are part of the default security model, you can use the built-in roles to quickly delegate power and implement security.

Built-in Roles

These built-in roles address common tasks you can perform through the DRA and ExA user interfaces. The following list describes each built-in role and summarizes the powers associated with that role.

Audit All Objects

Provides all the powers required to view properties of objects, policies, and configurations across your enterprise. This role does not allow an AA to modify properties. Assign this role to AAs responsible for auditing actions across your enterprise. Allows AAs to view all nodes except the Custom Tools node.

Audit Limited Account and Resource Properties

Audit Resources

Provides all the powers required to view properties of managed resources. Assign this role to AAs responsible for auditing resource objects.

Audit Users and Groups

Provides all the powers needed to view user account and group properties, but no powers to modify these properties. Assign this role to AAs responsible for auditing account properties.

Built-in Scheduler - Internal Use Only

Clone User with Mailbox

Provides all the powers required to clone an existing user account along with the account mailbox. Assign this role to AAs responsible for managing user accounts.

NOTE: To allow the AA to add the new user account to a group during the clone task, also assign the Manage Group Memberships role.

Computer Administration

Provides all the powers required to modify computer properties. This role allows AAs to add, delete, and shut down computers, as well as synchronize domain controllers. Assign this role to AAs responsible for managing computers in the ActiveView.

Configure Servers and Domains

Provides all the powers required to modify Administration server options and managed domains. Also provides powers necessary to configure and manage Office 365 tenants. Assign this role to AAs responsible for monitoring and maintaining the Administration servers.

Contact Administration

Provides all the powers required to create a new contact, modify contact properties, or delete a contact. Assign this role to AAs responsible for managing contacts.

Create and Delete Computer Accounts

Provides all the powers required to create and delete a computer account. Assign this role to AAs responsible for managing computers.

Create and Delete Groups

Provides all the powers required to create and delete a group. Assign this role to AAs responsible for managing groups.

Create and Delete Resources

Provides all the powers required to create and delete shares and computer accounts, and clear event logs. Assign this role to AAs responsible for managing resource objects and event logs.

Create and Delete Resource Mailbox

Provides all the powers required to create and delete a a mailbox. Assign this role to AAs responsible for managing mailboxes.

Create and Delete User Accounts

Provides all the powers required to create and delete a user account. Assign this role to AAs responsible for managing user accounts.

Dynamic Group Administration

Provides all the powers required to manage Active Directory dynamic groups.

DRA Administration

Provides all powers to an AA. This role gives a user the permissions to perform all administration tasks within DRA and ExA. This role is equivalent to the permissions of an administrator. An AA associated with the DRA Administration role can access all Directory and Resource Administrator nodes.

Execute Advanced Queries

Provides all the powers required to execute saved advanced queries. Assign this role to AAs responsible for executing advanced queries.

Group Administration

Provides all the powers required to manage groups and group memberships, and view corresponding user properties. Assign this role to AAs responsible for managing groups or account and resource objects that are managed through groups.

Help Desk Administration

Provides all the powers required to view user account properties, and to change passwords and password related properties. This role also allows AAs to disable, enable, and unlock user accounts. Assign this role to AAs responsible for Help Desk duties associated with ensuring users have proper access to their accounts.

Mailbox Administration

Provides all the powers required to manage Microsoft Exchange mailbox properties. If you use Microsoft Exchange, assign this role to AAs responsible for managing Microsoft Exchange mailboxes.

Manage Active Directory Collectors, DRA Collectors, and Management Reporting Collectors

Provides all the powers required to manage Active Directory Collectors, DRA Collectors, Office 365 Tenant Collectors, and Management Reporting Collectors for data collection. Assign this role to AAs responsible for managing reporting configuration.

Manage Active Directory Collectors, DRA Collectors, Management Reporting Collectors, and Database Configuration

Provides all the powers required to manage Active Directory Collectors, DRA Collectors, Management Reporting Collectors, and database configuration for data collection. Assign this role to AAs responsible for managing reporting and database configuration.

Manage Advanced Queries

Provides all the powers required to create, manage, and execute advanced queries. Assign this role to AAs responsible for managing advanced queries.

Manage and Execute Custom Tools

Provides all the powers required to create, manage, and excecute custom tools. Assign this role to AAs responsible for managing custom tools.

Manage Clone Exceptions

Provides all the powers required to create and manage clone exceptions.

Manage Computer Properties

Provides all the powers required to manage all properties for a computer account. Assign this role to AAs responsible for managing computers.

Manage Database Configuration

Provides all the powers required to manage database configuration for Management reports. Assign this role to AAs responsible for managing reporting database configuration.

Manage Dynamic Distribution Groups

Provides all the powers required to manage Microsoft Exchange dynamic distribution groups.

Manage Exchange Mailbox Rights

Provides all the powers required to manage security and rights for Microsoft Exchange mailboxes. If you use Microsoft Exchange, assign this role to AAs responsible for managing Microsoft Exchange mailbox permissions.

Manage Group Email

Provides all the powers required to view, enable, or disable the email address for a group. Assign this role to AAs responsible for managing groups or email addresses for account objects.

Manage Group Membership Security

Provides all the powers required to designate who can view and modify Microsoft Windows group memberships through Microsoft Outlook

Manage Group Memberships

Provides all the powers required to add and remove user accounts or groups from an existing group, and view the primary group of a user or computer account. Assign this role to AAs responsible for managing groups or user accounts.

Manage Group Properties

Provides all the powers required to manage all properties for a group. Assign this role to AAs responsible for managing groups.

Manage Mailbox Move Requests

Provides all the powers required to manage mailbox move requests.

Manage Policies and Automation Triggers

Provides all the powers required to define policies and automation triggers. Assign this role to AAs responsible for maintaining company policies and automating workflows.

Manage Printers and Print Jobs

Provides all the powers required to manage printers, print queues, and print jobs. To manage print jobs associated with a user account, the print job and the user account must be included in the same ActiveView. Assign this role to AAs responsible for maintaining printers and managing print jobs.

Manage Resources for Managed Users

Provides all the powers required to manage resources associated with specific user accounts. The AA and the user accounts must be included in the same ActiveView. Assign this role to AAs responsible for managing resource objects.

Manage Resource Mailbox Properties

Provides all the powers required to manage all properties for a mailbox. Assign this role to AAs responsible for managing mailboxes.

Manage Security Model

Provides all the powers required to define the Administration rules, including ActiveViews, AAs, and roles. Assign this role to AAs responsible for implementing and maintaining your security model.

Manage Services

Provides all the powers required to manage services. Assign this role to AAs responsible for managing services.

Manage Shared Folders

Provides all the powers required to manage shared folders. Assign this role to AAs responsible for managing shared folders.

Manage Temporary Group Assignments

Provides all the powers required to create and manage temporary group assignments. Assign this role to AAs responsible for managing groups.

Manage UI Reporting

Provides all the powers required to generate and export Activity Detail reports for users, groups, contacts, computers, organizational units, powers, roles, ActiveViews, containers, published printers, and Assistant Admins. Assign this role to AAs responsible for generating reports.

Manage User Dial in Properties

Provides all the powers required to modify the dial in properties of user accounts. Assign this role to AAs responsible for managing user accounts that have remote access to the enterprise.

Manage User Email

Provides all the powers required to view, enable, or disable the email address for a user account. Assign this role to AAs responsible for managing user accounts or email addresses for account objects.

Manage User Password and Unlock Account

Provides all the powers required to reset the password, specify password settings, and unlock a user account. Assign this role to AAs responsible for maintaining user account access.

Manage User Properties

Provides all the powers required to manage all properties for a user account, including Microsoft Exchange mailbox properties. Assign this role to AAs responsible for managing user accounts.

Manage Virtual Attributes

Provides all the powers required to create and manage virtual attributes. Assign this role to AAs responsible for managing virtual attributes.

Manage WTS Environment Properties

Provides all the powers required to change the WTS environment properties for a user account. Assign this role to AAs responsible for maintaining the WTS environment or managing user accounts.

Manage WTS Remote Control Properties

Provides all the powers required to change the WTS remote control properties for a user account. Assign this role to AAs responsible for maintaining WTS access or managing user accounts.

Manage WTS Session Properties

Provides all the powers required to change the WTS session properties for a user account. Assign this role to AAs responsible for maintaining WTS sessions or managing user accounts.

Manage WTS Terminal Properties

Provides all the powers required to change the WTS terminal properties for a user account. Assign this role to AAs responsible for maintaining WTS terminal properties or managing user accounts.

OU Administration

Provides all the powers required to manage organizational units. Assign this role to AAs responsible for managing the Active Directory structure.

Rename Group and Modify Description

Provides all the powers required to modify the name and description of a group. Assign this role to AAs responsible for managing groups.

Rename User and Modify Description

Provides all the powers required to modify the name and description of a user account. Assign this role to AAs responsible for managing user accounts.

Replicate Files

Provides all the powers required to upload, delete and modify file information. Assign this role to AAs responsible for replicating files from the primary Administration server to other Administration servers in the MMS and the DRA client computers.

Reset Local Administrator Password

Provides all the powers to reset the local administrator account password and view the name of the computer administrator. Assign this role to AAs responsible for managing the administrator accounts.

Reset Password

Provides all the powers required to reset and modify passwords. Assign this role to AAs responsible for password management.

Reset Password and Unlock Account Using SPA

Provides all the powers required to use Secure Password Administrator to reset passwords and unlock user accounts.

Reset Unified Messaging PIN Properties

Provides all the powers required to reset Unified Messaging PIN properties for user accounts.

Resource Administration

Provides all the powers required to modify properties of managed resources, including resources associated with any user account. Assign this role to AAs responsible for managing resource objects.

Resource Mailbox Administration

Provides all the powers required to manage resource mailboxes.

Self Administration

Provides all the powers required to modify basic properties, such as telephone numbers, of your own user account. Assign this role to AAs to allow them to manage their own personal information.

Start and Stop Resources

Provides all the powers required to pause, start, resume, or stop a service, start or stop a device or printer, shut down a computer, or synchronize your domain controllers. Also provides all the powers required to pause, resume, and start services, stop devices or print queues, and shut down computers. Assign this role to AAs responsible for managing resource objects.

Transform a User

Provides all the powers required to add a user to or remove a user from groups found in a template account, including the ability to modify the user's properties while transforming the user.

User Administration

Provides all the powers required to manage user accounts, associated Microsoft Exchange mailboxes, and group memberships. Assign this role to AAs responsible for managing user accounts.

View Active Directory Collectors, DRA Collectors, Management Reporting Collectors, and Database Configuration Information

Provides all the powers required to view AD collectors, DRA collectors, management reporting collectors, and database configuration information.

View All Computer Properties

Provides all the powers required to view properties of a computer account. Assign this role to AAs responsible for auditing computers.

View All Group Properties

Provides all the powers required to view properties for a group. Assign this role to AAs responsible for auditing groups.

View All Resource Mailbox Properties

Provides all the powers required to view properties for a resource mailbox. Assign this role to AAs responsible for auditing resource mailboxes.

View All User Properties

Provides all the powers required to view properties for a user account. Assign this role to AAs responsible for auditing user accounts.

WTS Administration

Provides all the powers required to manage Windows Terminal Server (WTS) properties for user accounts in the ActiveView. If you use WTS, assign this role to AAs responsible for maintaining the WTS properties of user accounts.

If you have licensed the File Security Administrator product, additional built-in roles are available. For more information about File Security Administrator, see the *User Guide for File Security Administrator*.

Accessing Built-in Roles

Access built-in roles to audit the default security model or manage your own security settings.

To access built-in AA groups:

1. In the left pane, select **Delegation Management**.
2. Under Common Tasks in the right pane, click **Manage Roles**.
3. Select the appropriate role.

Using Built-in Roles

You cannot delete or modify built-in roles. However, you can incorporate the built-in roles into your existing security model or use these roles to design and implement your own model.

You can use built-in roles in the following ways:

- Associate a built-in role with a user account or AA group. This association provides the user or AA group members with the appropriate powers for the task.
- Clone a built-in role and use that clone as the basis for a custom role. You can add other roles or powers to this new role and remove powers originally included in the built-in role.

For more information about designing a dynamic security model, see “Implementing Your Dynamic Security Model” on page 34.

Implementing Your Dynamic Security Model

By designing and implementing an environment that is both secure and easy to manage, you can maximize the power and flexibility DRA and ExA offer. Through a dynamic security model, you can distribute and automate many administration tasks and duties.

This section describes key concepts about dynamic, distributed administration and illustrates these concepts with examples and scenarios. These examples and scenarios assume you have the DRA Administration role or the corresponding powers. Review these sections to learn about the following concepts:

- How to implement your security model with DRA and ExA
- How to build dynamic, self maintaining ActiveViews
- How to harness the power of overlapping and hierarchical ActiveViews

DRA and ExA provide powerful tools for distributing specific administration permissions. In addition, these products provide policy and automation features to help you streamline your security model.

How to Create a Security Model

When you design and implement your security model, consider the following questions:

- Which objects need to be managed?
- Which actions need to be performed to complete a specific task?
- Which people need to perform these tasks and manage these objects?

How you answer these questions determines what types of ActiveViews, roles, and AA groups you need. Your answers also determine what type of security model you should create.

NOTE: If you are managing a subtree of a domain, you can implement the DRA security model on the Administration server in the corresponding domain or in another domain managed by DRA.

For more information about these security model components, see “Understanding the Dynamic Security Model” on page 13 and “Understanding the Default Security Model” on page 21.

Delegating Administration through a Static Model

When you distribute administration through a static security model, you define rules that include specific objects, delegate specific powers, and assign specific AAs to manage these objects. You define a unique rule for each object, power, and AA.

A static model can be appropriate for situations in which the enterprise scope is unlikely to change. However, this approach has limitations. It can prevent your security model from automatically responding to change, and it can require more maintenance.

For example, suppose JSmith, the Executive Assistant for Engineering, needs to change the home addresses and phone numbers for the five engineering managers. You can create an ActiveView that includes these five user accounts by defining a rule for each account. You can assign the individual powers to JSmith. However, when the Executive Assistant for Marketing needs to change personal information for the marketing managers, you must duplicate your original efforts. If you need to limit powers later, you must manually remove the unwanted powers from each ActiveView in which the Executive Assistants have power.

Delegating Administration through a Dynamic Model

When you distribute administration through a dynamic security model, you define rules that include multiple objects, delegate reusable sets of powers, and assign multiple AAs to manage these objects. You define rules that specify these objects, powers, and AAs through naming conventions, wildcard matching, group memberships, and roles. In this way, a dynamic model allows you to more effectively respond to change and decrease maintenance.

Using groups can help simplify your security model while providing a more dynamic solution. Instead of assigning individual powers to individual user accounts, you assign roles to a group. Each group member inherits the powers assigned to the group. In this model, when you add a user account to the group, the user automatically gains the set of powers associated with this group. When you add a power to the role, each group member automatically gains that power. You do not need to redefine your security model to accommodate a changing enterprise environment.

For example, suppose JSmith, the Executive Assistant for Engineering, needs to change the home addresses and phone numbers for all engineering managers.

You can create the following objects:

- A group called Engineering Managers
- A group called Executive Assistants With Power
- A role called Modifying Home Information

You can assign the Modifying Home Information role to the Execute Assistants With Power group, delegating the same set of powers to all members of that group. You can also create an ActiveView that includes user accounts from the Engineering Managers group. Thus, when a new manager is hired, you can immediately allow JSmith to access the account properties by adding this new user to the Engineering Managers group. This dynamic delegation occurs because the ActiveView rule uses group membership to automatically include the new account.

Understanding Power Creation

A power defines the properties of an object an Assistant Admin can view, modify, or create in your managed domain or subtree. You can create custom powers. Custom powers allow you to delegate power over specific object properties.

You can create custom powers for many different scenarios. You can create or clone specific powers you need to include in roles for common administration tasks. For example, you may need a power to control some properties that have been added to your schema or to grant power over an Active Directory property that is exposed in the DRA consoles with a UI extension. Custom powers can include access to multiple powers, such as the View All User Properties power, so a custom power should contain all the necessary properties to control the object you want to manage or modify. To create a power, you must have the appropriate powers, such as those included in the Manage Security Model role. For more information about custom powers, see Appendix C, “Custom Powers,” on page 87.

Understanding Role Creation

A role should contain all the necessary powers to complete a particular job or workflow. In this way, a role presents a job description. You can create new roles that group together specific powers you need or use the provided built-in roles for common administration tasks. For example, you may need a role that includes the powers to only reset passwords of user accounts.

When implementing roles in a dynamic security model, you can take advantage of this flexibility by assigning roles to AA groups. This delegation helps your model ensure that the proper people have the required permissions.

For more information about built-in roles, see “Understanding Built-in Roles” on page 27.

Understanding Assistant Admin Group Creation

An AA group contains all the user accounts and groups you want to grant powers. AA groups typically have roles within your security model. When you create AA groups, you can include user accounts, groups, and other AA groups. You can specify accounts and groups by name or use a wildcard specification that matches multiple accounts and groups.

If you create a static AA group that includes specific user accounts, you must maintain the group definition each time you want to add or remove someone from that AA group. For example, each time you add a user to a static AA group, you must define a rule that specifies the user account.

An easier way to implement your model is to create dynamic AA groups based on naming conventions or group memberships. Dynamic AA groups reduce and simplify your enterprise maintenance.

For example, wildcard specifications allow you to define dynamic AA groups that include user accounts and groups based on criteria, such as naming conventions. These definitions are self maintained. When you define AA group membership through a wildcard specification, DRA automatically updates the AA group membership whenever a new account matches the wildcard specification.

Another way to incorporate flexibility into your model is to define groups based on group membership. For example, you could create an AA group that includes all Help Desk personnel in the New York City office. If you have a group, such as NYC_HelpDesk, that includes these user accounts, you can include that group in an AA group. Then, when you update the membership of the NYC_HelpDesk group, DRA automatically updates the AA group membership.

NOTE: To fully grant powers to an AA group, you must create an ActiveView and associate the AA group with a role in that ActiveView.

Understanding ActiveView Creation

ActiveViews provide access to defined sets of objects, such as contacts or print jobs. When you create an ActiveView, you are creating an ActiveView object that has basic properties, such as a name and a description. To use this ActiveView, you must add objects, assign AAs, and assign roles or powers. The result is an ActiveView containing objects that particular AAs can view and manage.

For example, you can create a NYC_Sales People ActiveView that represents a set of user accounts, such as all the user accounts in the NYC_Sales group. Then, you can assign the NYC Help Desk AA group to the Update User Addresses role and the NYC_Sales People ActiveView. Through this delegation, you are giving the NYC_HelpDesk group members the ability to modify the address fields of all user accounts in the NYC_Sales group.

DRA provides a Delegation Wizard that allows you to easily and quickly define and assign ActiveViews. For more information, see the Getting Started Guide.

You also can use ActiveView rule restrictions to limit how an AA manages a set of objects. When defining which objects an ActiveView will include, you can select a restriction that designates these objects as **source objects** or **target objects**. For example, when the AA adds a user account to a group, the user account is the source object and the group is the target object. If you select the **Do not allow the users to be added to groups or moved to OUs** restriction, the AA cannot manage these user accounts as source objects. If you select this restriction, DRA will not allow the AA to add a user account in this ActiveView to any group in the enterprise.

Optimizing Your ActiveView Rules

You can configure your ActiveView rules to optimize performance for your enterprise. The following optimization tips can significantly increase performance when managing domains that contain over 250,000 objects.

Specific Matches

Specific matches let you identify the exact objects to include in an ActiveView. For example, you can create an ActiveView that includes user accounts from a specific OU. If your Active Directory structure allows you to specify objects by OU or domain, define rules that include objects from the specific OU or domain. If you need to specify objects from several OUs, and these OUs are unlikely to change, define a rule for each OU. For example, if you want to create an ActiveView that includes computers from the Sales and Marketing OUs, define a rule for each OU.

All rules that define a specific object, such as an OU, group, user, computer, resource type, contact, or domain, can optimize your model. You can also optimize a wildcard rule, such as including all user accounts whose description matches Sales, by specifying the domain of these accounts. Rules that specify a user principal name or logon name may not optimize your model.

Wildcards

If your security model uses a naming convention, wildcards offer tremendous power and flexibility. For example, you can create an ActiveView that includes computers whose pre-Windows 2000 names match ATL*. When using a naming convention, keep in mind that wildcard matches that look for prefixes (`ATL*`), groups, or pre-Windows 2000 names provide better performance.

You can use wildcards instead of regular expressions to narrow or broaden the scope of the rule. Wildcard matching is not case-sensitive. You can also use the question mark (?), asterisk (*), or number sign (#) wildcard characters as normal characters by prefixing a backslash (\) to the particular wildcard character. For example, to search for `abc*`, type the search text `abc\*`.

DRA and ExA support the following wildcard characters. You cannot use wildcard characters in names.

Match Item	Character	Definition
Any character	Question mark ?	Matches exactly one character
Any digit	Number sign #	Matches one digit
Any character, 0 or more matches	Asterisk *	Matches zero or more characters

The following table provides examples of wildcard character specifications and what they match and do not match.

Example	Matches	Does Not Match
Den???	Denton and Dennis	Denison
El ????o	El Campo and El Indio	El Paso
Houston, TX #####	Houston, TX 77024	Houston, TX USOFA

DRA and ExA do not support wildcard specifications that contain logical operations.

Groups

Groups can help you implement a dynamic security model while optimizing performance. For example, if you need to configure an ActiveView that includes many objects from multiple OUs or domains, you can create a group that contains these objects and then create an ActiveView that includes members of this group. In this case, the ActiveView has one rule that acts on one specific object (the group), even though it includes multiple objects (the group members).

If the Active Directory structure and group set are unlikely to change, you can define a rule for each group, specifying the group and its members.

To make your security model dynamic, the ActiveView can be maintained through a wildcard specification that acts on established group naming conventions. For example, if the pre-Windows 2000 names of your groups have a common prefix, you can define a single rule that matches this prefix.

Delegation Management Node

Use the Delegation Management node to define and maintain your security model. Delegation Management consists of the following nodes:

ActiveViews

Allows you to list current ActiveViews, create and clone ActiveViews, delegate administration, define ActiveView rules, and view managed objects. You can also view assigned AA groups and roles.

Roles

Allows you to list current roles, create roles, delegate administration, add powers to roles, nest roles within a role, clone individual roles or nested roles, delete roles, and view the properties of a role. You can also view assigned AA groups and ActiveViews.

Powers

Allows you to list built-in and custom powers, create and clone powers, and view and change power properties. Directory and Resource Administrator offers you the ability to quickly and efficiently manage and create custom powers. New powers allow you to delegate power over specific object properties. A power defines the properties of an object an Assistant Admin can view, modify, or create in your managed domain or subtree. Custom powers can include access to multiple properties, such as the View All User Properties power.

Assistant Admins

Allows you to list current AAs and AA groups, create and clone AA groups, delegate administration, define AA group rules, and view AA properties. You can also view AA group members and assigned roles and ActiveViews.

Allowing Users to Change Personal Information

You can allow users to manage personal information for their own accounts. This type of administration is called **self administration**. AAs with self administration permissions can change their basic account properties, such as telephone numbers and street addresses.

To delegate self administration:

1. In the left pane, select **Delegation Management**.
2. Under Common Tasks in the right pane, click **Delegate Administration**.
3. On the Welcome window, click **Next**.
4. Add the user accounts or groups to whom you want to delegate self-administration, and then click **Next**.
5. Add the Self Administration role, and then click **Next**.

   For more information, see “Understanding Built-in Roles” on page 27.
6. On the Add menu, click **Objects that match a rule**.
7. Click **Self Administration**, and then click **OK**. This rule automatically includes the AAs you assigned.
8. Click **Next**.
9. Specify the name, description, and comment for this new ActiveView.
10. Click **Finish**.

A The Command-Line Interface

These sections describe the syntax and operation of the command-line interface (CLI). You can use the CLI to create the delegation model and perform account and server administration tasks for multiple objects at one time. The CLI supports basic administration commands for DRA and ExA.

Understanding the CLI

You can install the CLI through the user interface part of the Setup program. By default, the Setup program places the `EA.EXE` file in the `Program Files (x86)\NetIQ\DRA` folder on the DRA client computer. This file allows you to run DRA and ExA commands from the command prompt on Microsoft Windows computers.

The CLI processes the commands from the command prompt. You can run the CLI commands from the `\Program Files(x86)\NetIQ\DRA\` directory. By default, this location is not added to your path statement. For more information about adding this location to your path statement, see your Microsoft Windows documentation.

The CLI provides a way to quickly create AVs with rules matching the OUs.

The CLI uses several conventions to help you use the available commands. The following sections define these conventions. These sections also describe several specific characteristics of the CLI. Making mass changes through the CLI can cause other user interfaces, such as the Account and Resource Management console, to wait while the Administration server applies these changes.

CLI Syntax

This section uses a specific syntax for documenting CLI commands. For example, the syntax for using the `GROUP` command to update group properties is:

`EA [/DOMAIN:`*`domain`* `[/SERVER:`*`computername`*`|/MASTER]] GROUP target UPDATE {NAME:group|CN:`*`"commonname"`*`|COMMENT:`*`"comment"`*`}`

The following table lists the conventions and how they apply to this `GROUP` command.

Convention	Represents	Example
`CAPITAL LETTERS`	Commands and options	The command is `GROUP`. This command allows you to manage group accounts. The `/DOMAIN` and `UPDATE` parameters provide additional controls you can use with this command.
Italics	Variable names and values	Specify the appropriate domain name and computer name in place of the domain and computername variables.
Brackets, such as [*value*]	Optional parameters.	You are not required to specify the `/DOMAIN` or `/SERVER` parameter.

Convention	Represents	Example
Braces, such as {*value*}	Required parameters.	You must specify which properties you want to update.
Logical OR, such as *val1* | *val2*	Exclusive parameters. Choose one parameter.	You can update only one of the following properties: `NAME, CN, COMMENT`.

TIP: If you specify an incomplete command, the CLI displays syntax and option descriptions. For example, to display the syntax and help information for the `GROUP` command, enter:

```
EA GROUP
```

Embedded Spaces and Quotes

The CLI uses spaces to separate keywords and arguments. If you want to specify a value that contains one or more embedded spaces, enclose the value in quotation marks (`"."`). For example, to set the fullname property of the `JaneSmith` user account to `Jane Smith`, enter:

```
EA USER JaneSmith UPDATE FULLNAME:"Jane Smith"
```

The quotation marks ensure the CLI treats `Jane Smith` as a single value. Without the quotation marks, the CLI treats `Jane` and `Smith` as separate terms and generates an error message.

Date and Time Format

The CLI uses a consistent date and time format for input and output. Use the following format with the CLI:

```
YYYY MM DD,hh:mm:ss
```

For example, to set the expiration date of the `MWest` user account to May 1, 2002 at 5:00 PM, enter:

```
EA USER MWest UPDATE EXPIRES:2002 05 01,17:00:00
```

Wildcard Characters and Naming Restrictions

DRA and ExA support wildcard characters in AA group names and many other CLI options. Names of some objects, such as user accounts, groups, resources, ActiveViews, AAs, and Administrators, cannot contain specific characters. These restrictions apply throughout the DRA and ExA user interfaces.

Special Terms

The following terms provide a consistent way to refer to types of command parameters:

name

Indicates a single name that includes no wildcard characters. For example, to indicate the `TomB` user account, specify `TomB`.

wildcard

Indicates a name that includes wildcard characters. The CLI expands wildcard characters in context, similar to the DOS and Microsoft Windows command line file name wildcard specifications. For example, to indicate all groups that begin with `Sales_`, specify `Sales_*`.

Special Functions and Variables

The CLI provides tremendous expressive power. Use the following special functions and variables to reference common sets of objects. These functions and variables enable you to perform a single command on multiple objects.

`@GroupMembers(`*`"wildcard"`*`)`

Returns a list of contacts, computers, groups, and user accounts that are members of any group matching the specified *wildcard.* You can use this function with the `ACCOUNT` and `GROUP` commands.

`@GroupMembersR(`*`"wildcard"`*`)`

Returns a list of contacts, computers, groups, and user accounts that are members of any group matching the specified *wildcard.* This function is recursive, which means it will enumerate group memberships for groups that are members of the *wildcard* group. You can use this function only with the `ACCOUNT` and `GROUP` command.

`@GroupUsers(`*`"wildcard"`*`)`

Returns a list of user accounts that are members of any group matching the specified *wildcard.*

`@GroupUsersR(`*`"wildcard"`*`)`

Returns a list of user accounts that are members of any group matching the specified *wildcard*. This function is recursive, which means it will enumerate group memberships for groups that are members of the *wildcard* group.

`@Target()`

Represents the current target of the command. This function allows you to include the target name in the specified command. For example, to set the home directory path of each user account in the `Atlanta Users` group to `\\ATLHOME\USERS\`*username*, enter:

```
EA USER @GroupUsers("Atlanta Users") UPDATE HOMEDIR:\\ATLHOME\USERS\@Target()
```

NOTE: The console allows you to use `%username%` to represent the current target. However, when the `%username%` variable is used in the CLI, Microsoft Windows defines the `%username%` variable as the currently logged on user.

`@TerritoryAccounts(`*`"wildcard"`*`)`

Returns a list of all groups and user accounts included in any ActiveView matching the specified *wildcard.*

`@TerritoryGroups(`*`"wildcard"`*`)`

Returns a list of all groups included in any ActiveView matching the specified *wildcard*.

`@TerritoryMembers(`*`"wildcard"`*`)`

Returns a list of all groups and user accounts included in any ActiveView matching the specified *wildcard*. This function is the same as the `@TerritoryAccounts(`*wildcard*`)` special function.

`@TerritoryUsers(`*`"wildcard"`*`)`

Returns a list of all user included in any ActiveView matching the specified *wildcard.*

Special Operators and Prefixes

When you specify some options, you may also need to specify a prefix. The CLI supports the following prefixes:

domain	Allows you to identify a user account or group in a different domain. For example, if you manage multiple domains, members of local groups can be user accounts or groups in any managed or trusted domain. If the user accounts and groups are in a domain other than the connected domain, the user account and group specifications must contain the `domain\` prefix, where domain identifies the name of this other domain. For example, to add the TomB user account from the Houston trusted domain to the Sales group, enter: `EA GROUP Sales MEMBERADD Houston\TomB` If you do not specify a domain, the CLI defaults to the connected domain.
/UNICODE	Allows you to output unicode text to the console or a file. If you want unicode output from a batch file to be redirected to a file, you should include both the "/unicode" flag as well as the redirected filename within the batch file.
/NOCR	Removes the extra CR character sequence from the CLI command output when you direct the output of the CLI command to a text file. If you use the `/UNICODE` option, type the `/NOCR` option before the `/UNICODE` option. For example, if you use the `USER` command and want to direct the output to the `temp.txt` file, enter: `EA /DOMAIN:acct04 /MASTER /NOCR /UNICODE USER DISPLAY > temp.txt`
g:	Identifies a new rule or AA as a *group.* When you specify a new rule or AA that includes a wildcard character, you need to indicate whether the rule or AA is a user account, group, or ActiveView specification. Specify `xg:` for an exclude group rule.
t:	Identifies a new rule as an *ActiveView.* When you specify a new rule that includes a wildcard character, you need to indicate whether that new rule is a user account, group, or ActiveView specification. Specify `xt:` for an exclude ActiveView rule.
u:	Identifies a new rule or AA as a *user*. When you specify a new rule or AA that includes a wildcard character, you need to indicate whether that new rule or AA is a user account, group, or ActiveView specification. Specify `xu:` for an exclude user rule.

Return Codes

You can create batch files with CLI commands. The CLI commands return codes depending on the success or failure of the commands. You can use these return codes to write conditional statements. The return codes are:

0	Information
1	Warning
2	Error
3	Severe error
4	Very severe error
5	Unrecoverable error
6	Extremely unrecoverable error

CLI Commands

The following sections provide details about each of the CLI commands, including the following:

- Required powers and permissions
- Syntax statements
- Supported options
- Several usage examples

AA Command

The AA command creates a custom Assistant Admin Group with the name, comment, and description you provide. The AA command allows you to create user and group rules with the `ADD` verb.

You must associate AAs with roles and ActiveViews to ensure AAs manage objects included in ActiveViews. This association is called delegation. Through delegation, you specify the tasks AAs can perform on the managed objects. To assign roles to AAs and ActiveViews, you must have the appropriate powers, such as those included in the Manage Security Model role.

Required Powers and Permissions

To manage assigned roles and ActiveViews, you must have the appropriate powers, such as those included in the DRA Administration and Manage Security Model roles.

Syntax

```
EA [/DOMAIN:domain [/SERVER:computername|/MASTER]] AA target CREATE [fields]
EA [/DOMAIN:domain [/SERVER:computername|/MASTER]] AA target DELETE [mode:{I|B}]
EA [/DOMAIN:domain [/SERVER:computername|/MASTER]] AA target UPDATE [NAME:newname]
[mode:{I|B}] [fields]
EA [/DOMAIN:domain [/SERVER:computername|/MASTER]] AA target DISPLAY [fields]
EA [/DOMAIN:domain [/SERVER:computername|/MASTER]] AA target ADD ruleName [OU:]
{TYPE:ruleType} {MATCH:matchString} [MEMBERS:] [RECURSIVE:] [SELECTBASE:]
[ACTION:{include|exclude}] [RESTRICTION:] [GROUPSCOPE:] [GROUPTYPE:][MODE:{I|B}]
EA [/DOMAIN:domain [/SERVER:computername|/MASTER]] AA target REMOVE rulename
[MODE:{I|B}]
EA [/DOMAIN:domain [/SERVER:computername|/MASTER]] AA target DISPLAYRULES
ruleTarget [ruleFields]
EA [/DOMAIN:domain [/SERVER:computername|/MASTER]] AA target UPDATERULES
ruleTarget [ruleFields] [NAME:newname] [MODE:{I|B}]
```

NOTE: It is not possible to create an Assistant Admin Group of type "principal".

Verbs:

CREATE Creates a custom Assistant Admin Group with the given name, comment, and description. It is not possible to create an Assistant Admin Group of the type principal. An Assistant Admin Group becomes *principal* when assigning a user or group to an ActiveView.

DELETE	Deletes all custom Assistant Admin Groups with name matching target. When you delete Assistant Admin groups, you do not delete the Assistant Admin group members. However, Assistant Admin group members can no longer act on objects in the previously associated ActiveViews. Deleting an Assistant Admin group also deletes the Security Identifier (SID) associated with the Assistant Admin group assignment. You do not need to delete an Assistant Admin group to disassociate it from an ActiveView or role.
UPDATE	Updates all custom Assistant Admin Groups with name matching *target*.
DISPLAY	Displays any custom or built-in Assistant Admin Groups with a name matching *target*. The "name" parameter is always returned. However, by specifying only "name" on the command line, only the name field will be returned. "Assigned" corresponds to the "In Use" column of the Delegation and Configuration user interface. "Type" indicates built-in or custom.
ADD	Creates a user or group rule and associates it with an Assistant Admin. You cannot create rules matching all domains or all OUs matching wildcard in all managed domains. You cannot create a wildcard match where only a single object matches.
REMOVE	Removes the association between an Assistant Admin and an ActiveView, preventing this Assistant Admin from managing objects specified by the specified ActiveView.
DISPLAYRULES	Displays rules matching the match parameter in the target Assistant Admin. Target Assistant Admins can include both custom and built-in types.
UPDATERULES	Updates rules matching the match parameter in the target Assistant Admin. Only custom Assistant Admins will be considered for a match.

Options

/DOMAIN:*domain*	Specifies the name of the managed domain. If you do not specify this option, the CLI provides information about the domain to which the consoles last connected. If you have not used a console on this computer and you do not specify this option, the CLI connects to the domain where your user account resides. You can use wildcard characters to specify multiple domains.
/SERVER:*computername*	Specifies the name of an Administration server managing the specified domain. If you specify a domain without a server, the CLI automatically locates the closest Administration server in the specified domain. You can prefix the specified computer name with two backslashes (\\)
/MASTER	Specifies the primary Administration server managing the specified domain.
/DELI:{TAB|x}	Specifies the delimiter character you want the CLI to use to separate displayed field values. You can use this option to format output redirected to a file, easing the import of the file into a database or spreadsheet program for further analysis and reporting. You can specify any delimiter character. To specify a tab as the delimiter character, type TAB.
fields	Specifies the fields or options you want to modify or display for an Assist Admin account. When you specify one or more fields with the DISPLAY verb, type the field name without a value. For example, to display the user account comment, type COMMENT. You can specify the following field values: ALL Displays all fields.

	ASSIGNED Specifies whether the rule is in use. COMMENT:*text* Specifies the comment for the Assistant Admin group. To display comments, type COMMENT with the DISPLAY verb. DESCRIPTION:*text* Specifies the description for the Assistant Admin group. To display comments, type Description with the DISPLAY verb. NAME Specifies the new Assistant Admin name. TYPE Specifies the Assistant Admin type.
ruleFields	Specifies the rule fields or options you want to modify or display that pertain to the specified AA account. COMMENT Specifies the rule comment. DESCRIPTION Provides the rule description. The description is read-only. NAME Specifies the rule name.
TYPE:*ruleType*	[G|GROUP] Specifies a group rule. [U|USER] Specifies a user rule.
MATCH:*accountname*	Specifies the user account characters on which to match. You can use domain\AccountName format or wildcards. Uses the NetBIOS name of the current CLI focus domain.
ACTION:{include|exclude}	Specifies whether to include or exclude objects. Excludes overrule includes. For example, specifically excluding Marketing\Bob overrules an include of Marketing\B*.
RESTRICTION: {S|T|ST}	Specifies whether the rule is source or target. Default is source and target.
RECURSIVE:{Y|N}	Specifies whether to match objects in sub-containers or not. Default is yes.
SELECTBASE:{YES|NO}	Specifies whether the rule matches the group itself. Default is yes. For rules matching containers or groups, specifies whether to match base object. Default is yes.
GROUPSCOPE[[U|Universal] [G|Global] [L|Local]]	Specifies any combination of [[U|Universal] [G|Global] [L|Local]] or [All]. Multiple entries should be separated by commas.
GROUPTYPE:{S|D|ALL}	Specifies Security, Distribution, or All. Default is “All”.
MEMBERS:*memberTypes*	Specifies which type of member objects to manage. Group rules can specify group member types. Container and domain rules can specify managed object types. OU Specifies OU objects. U|USER Specifies user account objects. C|COMPUTER Specifies the computer member object. G|GROUP Specifies group objects. CT|CONTACT Specifies contact objects. ALL|NONE Specifies the all or none parameter.
MATCHNESTED:{YES|NO}	Specifies whether group rules match objects in nested groups. Adding one ActiveView to another is called **nesting**. By using nested ActiveViews in your security model, you can divide administration power and scope into smaller pieces and then assemble these pieces to meet different needs.

MODE:{I|B}	B|BATCH Specifies batch mode. Batch mode runs silently and processes without confirmation. I:Interactive Specifies interactive mode. Interactive mode provides confirmation and allows you to see the rule sentence. This is the default mode.

AA Example 1

To create an AA, enter:

```
EA AA SeattleAdmins CREATE comment:testComment description:"Admins in Seattle"
```

AA Example 2

To delete an AA, enter:

```
EA AA SeattleAdmins Delete
```

AA Example 3

To update an AA, enter:

```
EA AA "SeattleAdmins" UPDATE comment:"Seattle Printer Admins"
```

AA Example 4

To rename an AA, enter:

```
EA AA "SeattleAdmins" RENAME
```

AA Example 5

To display an AA, enter:

```
EA AA b* DISPLAY
```

AA Example 6

To display the Assistant Admin group properties, enter

```
EA AA "SeattleAdmins" DISPLAY type
```

AA Example 7

To create and associate a group rule with an AA, enter:

```
EA AA us-los* ADD groupRule type:g match:a* ou:testou*
```

ACCOUNT Command

The ACCOUNT command displays a list of all user accounts and groups in the specified domain.

Required Powers and Permissions

To run this command, you must have the appropriate powers, such as those included in the built-in User Administration and Group Administration roles.

Syntax

`EA [/DOMAIN:`*domain* `[/SERVER:`*computername*`|/MASTER]] ACCOUNT` *targetdomain*`\"`*accountname*`" DISPLAY`

NOTE: If the value for an option contains spaces, such as a user account name of `Jane Smith`, you must surround the option value with quotation marks. In this case, to specify a value for the *accountname* option, type *"Jane Smith".*

Verbs

DISPLAY	Displays the list of user accounts and groups in the specified domain.

Options

/DOMAIN:*domain*	Specifies the name of the managed domain. If you do not specify this option, the CLI provides information about the domain to which the consoles last connected. If you have not used a console on this computer and you do not specify this option, the CLI connects to the domain where your user account resides. You can use wildcard characters to specify multiple domains.
/SERVER:*computername*	Specifies the name of an Administration server managing the specified domain. If you specify a domain without a server, the CLI automatically locates the closest Administration server in the specified domain. You can prefix the specified computer name with two backslashes (`\\`)
/MASTER	Specifies the primary Administration server managing the specified domain.
targetdomain	Specifies the name of the domain that contains the accounts you want to display. If the target domain is the same as the domain specified by `/DOMAIN`, then you do not need to specify this option. You can use wildcard characters to specify multiple domains.
"*accountname*"	Specifies the accounts the CLI displays. The specified account name can contain wildcard characters.

ACCOUNT Example 1

To display all user accounts in the managed domains, enter:

```
EA ACCOUNT * DISPLAY
```

ACCOUNT Example 2

To display all user accounts in managed domains with names that start with HOU, enter:

```
EA ACCOUNT HOU*\* DISPLAY
```

ACCOUNT Example 3

To display all user accounts managed by the HOU_ADMIN02 secondary Administration server in the CITIES domain, enter:

```
EA /DOMAIN:CITIES /SERVER:\\HOU_ADMIN02 ACCOUNT * DISPLAY
```

ACCOUNT Example 4

To display all user accounts managed by the Primary Administration server in the SPACE domain, enter:

```
EA /DOMAIN:SPACE /MASTER ACCOUNT * DISPLAY
```

AV Command

The ActiveView command can create, delete, update, and rename ActiveViews. You can also use the ActiveView command to display the properties of an ActiveView, including the name, comment, description and type fields.

The ActiveView command allows for the creation of rules in conjunction with the ADD verb.

An ActiveView creates a virtual domain containing only those objects you want. You can then associate Assistant Admins with these ActiveViews and grant extremely granular control over the included objects. For more information, see “What ActiveViews Provide” on page 15.

Required Powers and Permissions

To create or delete an ActiveView or assign rules to ActiveViews, you must have the appropriate powers, such as those included in the built-in DRA Administration or Manage Security Model roles.

Syntax

```
EA [/DOMAIN:domain [/SERVER:computername|/MASTER]] AV target CREATE [fields]

EA [/DOMAIN:domain [/SERVER:computername|/MASTER]] AV target DELETE [mode:{I|B}]

EA [/DOMAIN:domain [/SERVER:computername|/MASTER]] AV target UPDATE [mode:{I|B}]
[fields] [NAME:target]

EA [/DOMAIN:domain [/SERVER:computername|/MASTER]] AV target DISPLAY [fields]

EA [/DOMAIN:domain [/SERVER:computername|/MASTER]] AV target ADD ruleName
[ruleFields] {TYPE:ruleType} {MATCH:matchString} [OU:ouString] [ACTION:]
[RESTRICTION:] [RECURSIVE:] [SELECTBASE:] [MEMBERS:memberType,...] [MODE:]
[MATCHWILDCARD]
```

EA [/DOMAIN:*domain* [/SERVER:*computername*|/MASTER]] AV *target* ADD ruleName [ruleFields] {TYPE:ruleType} {MATCH:matchString} [OU:ouString] [ACTION:] [RESTRICTION:] [RECURSIVE:] [SELECTBASE:] [MEMBERS:memberType,...] [MODE:] [MATCHWILDCARD] Resource rules only, {RESOURCES:resourceType,...} required parameter

EA [/DOMAIN:*domain* [/SERVER:*computername*|/MASTER]] AV *target* ADD ruleName [ruleFields] {TYPE:ruleType} {MATCH:matchString} [OU:ouString] [ACTION:] [RESTRICTION:] [RECURSIVE:] [SELECTBASE:] [MEMBERS:memberType,...] [MODE:] [MATCHWILDCARD] Group rules only, [matchNested], [groupScope], [groupType] optional parameters may be specified

EA [/DOMAIN:*domain* [/SERVER:*computername*|/MASTER]] AV *target* REMOVE ruleName [MODE:{I|B}]

EA [/DOMAIN:*domain* [/SERVER:*computername*|/MASTER]] AV *target* DISPLAYRULES ruleTarget [ruleFields]

EA [/DOMAIN:*domain* [/SERVER:*computername*|/MASTER]] AV *target* UPDATERULES ruleTarget [ruleFields] [NAME:newname] [MODE:{I|B}]

Verbs

CREATE Creates an ActiveView and specifies the name and the properties for the ActiveView including the comment and description.

DELETE Deletes a custom AV that matches the *target*. You can automate large deletes using a matching target in MODE:*batch*.

UPDATE Updates any custom AV that matches *target.*

RENAME Renames any custom AV that matches *target*.

DISPLAY Displays any custom or built-in AVs matching *target*. The name parameter is always returned. However, by specifying only "name" on the command line, then only the name field will be returned (by default other fields are displayed). The display command can be used to enumerate ActiveViews matching a certain name, including wildcards.

ADD Creates and assigns rules in conjunction with the AV command to cover the most frequent types of delegation. You use the ADD command with the following options and parameters: ruleName, [ruleFields], {TYPE:ruleType}, {MATCH:matchString}, [OU:ouString], [ACTION:], [RESTRICTION:], [RECURSIVE:], [SELECTBASE:], [MEMBERS:memberType,...], [MODE:], {RESOURCES:resourceType,...}, [matchNested], [groupScope], and [groupType]

REMOVE Removes the associated rule from the AV.

DISPLAYRULES Displays rules and rule properties for a given AV. Properties for a rule include the name, comment, and description.

UPDATERULES Updates rules and rule properties for a given AV. Properties for a rule include the name, comment, and description.

Options

/DOMAIN:*domain* — Specifies the name of the managed domain. If you do not specify this option, the CLI provides information about the domain to which the consoles last connected. If you have not used a console on this computer and you do not specify this option, the CLI connects to the domain where your user account resides. You can use wildcard characters to specify multiple domains.

/SERVER:*computername* — Specifies the name of an Administration server managing the specified domain. If you specify a domain without a server, the CLI automatically locates the closest Administration server in the specified domain. You can prefix the specified computer name with two backslashes (\\).

/MASTER — Specifies the primary Administration server managing the specified domain.

target — Specifies the name of the AV you want to manage. The AV name can contain wildcard characters. You can also precede the AV name with a trusted domain or wildcard character, such as **, which targets all groups in the managed domains and trusted domains. When using the CREATE verb, the specified AV must not already exist and you cannot specify wildcard characters. When using the DISPLAY verb, specify the target as * to list all AVs in the specified OU.

commonfields — Specifies the fields or options that you want to specify, modify, or display for the ActiveView group. When you specify one or more of fields with the DISPLAY verb, specify the field name without any value. For example, to display the ActiveView comment, specify COMMENT. You can specify the following field values: ALL Displays all fields. COMMENT:"*text*" Specifies the comment for the ActiveView group. To display comments, type COMMENT with the DISPLAY verb. DESCRIPTION:"*text*" Specifies the description for the ActiveView group. To display comments, type DESCRIPTION with the DISPLAY verb. NAME:"*name*" Specifies the new name of the ActiveView. TYPE: Specifies the type of ActiveView.

RuleName: — Specifies the name of the rule you want to create or manage.

ruleFields — Specifies the rule properties that you want to modify or display for the ActiveView. These are universal rule parameters. COMMENT Specifies the comment for the specified rule. DESCRIPTION Provides the rule description. The description is read-only. NAME Specifies the name for the rule.

TYPE:{*ruleType*} — G|GROUP Specifies a group rule. OU Specifies an OU rule. DOMAIN Specifies a domain rule. U|USER Specifies a user rule. COMPUTER Specifies a computer rule. RESOURCE Specifies a resource rule.

MATCH:*matchString* — Specifies domain or wildcard name match. Must be at least one character, though that character can be a *. For ActiveViews it matches by name.

OU:*ouname* — Specifies the name of either a DN path to an OU, container, built-in, or a wildcard matching at most one OU by name. If this is specified, then the match will be evaluated only within this OU. In the case of resource rules, note that the "match" that this will apply to is the "compMatch" parameter. Default is any OU. Note that if an OU is specified for an NT4 domain, the CLI should return an error message to the client indicating that this is an invalid argument.

ACTION:{include|exclude} — Specifies whether the rule is include or exclude. This is a universal rule parameter. The default is include.

RESTRICTION:{S|T|ST}	Specifies whether the rule is source or target only (or both). This is a universal rule parameter. The default is both.
RECURSIVE:{Y|N}	Specifies whether the container rule manages groups from children OUs and containers. Default is yes. Specifies whether the rule manages nested OUs and members. Default is yes. This parameter applies to either groups or OU rules.
SELECTBASE:{YES|NO}	Specifies whether the rule matches the group itself. Default is yes. For rules matching containers or groups, specifies whether to match base object. Default is yes.
MEMBERS:memberType	Specifies which type of member objects is managed. There are group member types for group rules or managed object types for container or domain rules. Default is `ALL`. `OU` Specifies OU objects. `U|USER` Specifies account objects. `C|COMPUTER` Specifies computer objects. `G|GROUP` Specifies group objects. `CT|CONTACT` Specifies contact objects. `ALL|NONE` Specifies the all or none parameter.
MATCHNESTED:{Y|N}	Specifies whether the rule manages nested groups and members. Default is yes.
GROUPSCOPE	Specifies any combination of `[[U|Universal] [G|Global] [L|Local]]` (multiple entries must be separated by commas) or `[All]`.
GROUPTYPE:{S|D}	Specifies Security, Distribution, or All. Default is "All".
MODE: [I|B]	B specifies batch mode. The default is interactive. This mode will allow the client to see the rule sentence. By setting mode to batch, the command is processed without confirmation.
MATCHWILDCARD	Specifies whether to include all groups that do not exactly match the string specified in the `MATCH:`*matchString* option.

AV Example 1

To create a custom AV with the given name, comment, and description, enter:

```
EA AV ouComputers CREATE comment:testComment description:"Contents of computers OU"
```

AV Example 2

To delete all custom AVs matching "g*", enter:

```
EA AV g* DELETE
```

AV Example 3

To update or rename any custom AVs matching "h*", enter:

```
EA AV h* UPDATE comment:"This AV starts with letter h"
```

AV Example 4

To add a group rule, enter:

```
EA AV us-los* add groupRule type:g match:a* ou:testou*
```

AV Example 5

To create an exclude rule for ou1, which is the only OU matching ou* within testou2, enter:

```
EA AV kt* add ouRule type:ou match:ou* ou:testou2 action:exclude
```

AV Example 6

To create a rule that manages the domain and only OUs and Groups in the domain, enter:

```
EA AV d* ADD domainRule type:d match:schwamx-dom members:OU,G
```

AV Example 7

To create a rule that excludes the CEO from management in any custom AV, enter:

```
EA AV * ADD ceoExcludeRule type:u match:netiqus\boesenb* action:exclude
```

AV Example 8

To exclude objects from the K* AVs in the L* AVs, enter:

```
EA AV L* Add ExcludeKAvs type:av match:K* action:exclude
```

AV Example 9

To add a rule to the "Domain Controllers" AV to match computers named "dc*" in the OU "domain controllers" in domain netiq.local, enter:

```
EA AV "Domain Controllers" Add DCRule1 type:c match:dc* ou:"ou=domain
controllers,dc=netiq,dc=local"
```

AV Example 10

To add a rule to the "Resources" AV to match services on computer "HOULAGOS" in the current CLI focus domain, enter:

```
EA AV Resources ADD type:resource resources:services match:houlagos
```

NOTE: Through ActiveViews, you can display and change the settings of many resource properties, create and clone resources, delete resources, as well as stop and start resources. Wildcard specifications allow you to include objects from several domains or OUs while making your security model more dynamic.

CACHE Command

The `CACHE` command refreshes the accounts cache and the resource cache on the Administration server.

The accounts cache contains information about user accounts, groups, computer accounts, and contacts. The Administration server builds and maintains the accounts cache, which contains portions of the Microsoft Windows 2008 or higher Active Directory. The Administration server uses the

accounts cache to improve performance when validating requests. The Administration server maintains the coherency of this cache for all account administration performed through DRA and ExA.

The resource cache contains computer information. The Administration server uses the resource cache to improve performance when managing computers.

NOTE: When you use the `CACHE` command to refresh the accounts cache, the Administration server performs an incremental cache refresh by default. An incremental accounts cache refresh updates only the data that changed since the previous refresh.

Required Powers and Permissions

To perform a manual refresh of the accounts and resource caches, you must have the appropriate powers, such as those included in the built-in Configure Servers and Domains role. Other AAs can only view cache refresh information.

Syntax

```
EA [/DOMAIN:domain [/SERVER:computername|/MASTER]] CACHE {targetdomain} [/FULL|/
SYSTEM]
EA [/DOMAIN:domain [/SERVER:computername|/MASTER]] CACHE {targetdomain} [DISPLAY]
```

Verbs

DISPLAY	Displays the time of the last refresh and the time of the next refresh.

Options

/DOMAIN:*domain*	Specifies the name of the managed domain for which you want to refresh or display the accounts and resource caches. If you do not specify this option, the CLI refreshes the cache for the domain to which the consoles most recently connected. If you have not used the CLI before and you do not specify this option, the CLI connects to the domain where your user account resides. You can use wildcard characters to specify multiple domains.
/SERVER:*computername*	Specifies the name of an Administration server that manages the specified domain. If you specify a domain and do not specify a server, the CLI automatically locates the best available Administration server in the specified domain. You can prefix the specified computer name with two backslashes (\\).
/MASTER	Specifies the primary Administration server that manages the specified domain.
targetdomain	Refreshes the accounts or resource cache for the specified domain, domain member, or computer.
/FULL	Performs a full accounts cache refresh. By default, the CLI performs an incremental accounts cache refresh.
/SYSTEM	Performs a resource cache refresh.

CACHE Example 1

To perform an incremental accounts cache refresh for the NORTHEAST domain, enter:

```
EA /DOMAIN:NORTHEAST CACHE NORTHEAST
```

CACHE Example 2

To perform a full accounts cache refresh on the LAB01 server in the NORTHEAST domain, enter:

```
EA /DOMAIN:NORTHEAST /SERVER:\\LAB01 CACHE NORTHEAST /FULL
```

CACHE Example 3

To refresh the resource cache for the PITTSBURGH child domain in the NORTHEAST domain, enter:

```
EA /DOMAIN:NORTHEAST CACHE PITTSBURGH /SYSTEM
```

CACHE Example 4

To display the last and next cache refresh times for the primary Administration server in the NORTHEAST domain, enter:

```
EA /DOMAIN:NORTHEAST /MASTER CACHE NORTHEAST DISPLAY
```

DOMAIN Command

The DOMAIN command displays an alphabetical list of all managed and trusted domains. The domain list includes the workstation, the domain to which the workstation belongs, and all the domains the workstation trusts.

Required Powers and Permissions

You must have the appropriate powers, such as those included in the built-in Computer Administration role, to run this command.

Syntax

```
EA [/DOMAIN:domain [/SERVER:computername|/MASTER]] DOMAIN namespec DISPLAY
[CONFIG]
```

Verbs

DISPLAY Displays the information about the specified domain.

Options

/DOMAIN:*domain*	Specifies the name of the managed domain. If you do not specify this option, the CLI displays the information for the domain to which the consoles most recently connected. If you have not used the CLI before and you do not specify this option, the CLI connects to the domain where your user account resides. You can use wildcard characters to specify multiple domains.
/SERVER:*computername*	Specifies the name of an Administration server that manages the specified domain. If you specify a domain and do not specify a server, the CLI automatically locates the best available Administration server in the specified domain. You can prefix the specified computer name with two backslashes (\\).
/MASTER	Specifies the primary Administration server that manages the specified domain.
namespec	Specifies the domains you want to include in the displayed list. The *namespec* variable can include wildcard characters. To display all managed and trusted domains, specify an asterisk (*).
CONFIG	Displays domain information, such as the number of user accounts and groups, and the scheduled accounts cache refresh time.

DOMAIN Example 1

To display information for all managed and trusted domains, as well as the PDC computer names, enter:

```
EA DOMAIN * DISPLAY
```

The CLI displays the domain names and the PDC computer names:

```
EA 7.50.00 (c) Copyright 2011 NetIQ Corporation; all rights reserved.
LAB_HOULAB_HOU              LAB_HOULAB_HOU.COM
HOUSTON_LABHOUSTON_LAB     HOUSTON_LABHOUSTON_LAB.LOCAL
NORTH_HOUNORTH_HOU         NORTH_HOUNORTH_HOU.COM
```

DOMAIN Example 2

To display the configuration information for the HOUTX domain, enter:

```
EA DOMAIN HOUTX DISPLAY CONFIG
```

EXEC Command

The EXEC command allows you to apply actions to large numbers of objects. This command differs from other CLI commands because it does not inherently perform specific administrative tasks. Use the EXEC command to run a command against a result set built through CLI wildcard characters and special functions. For more information about special functions, see “Special Functions and Variables” on page 43. The EXEC command runs a specified command at the CLI client where you enter the EXEC command. The EXEC command also allows you to run commands other than CLI commands.

NOTE: Use caution when using this command. Before using this command to make major system changes, back up your complete system. You can create a user account that has specific permissions and then sign on with this user account to restrict the use of this command to a limited number of objects.

Required Powers and Permissions

You must have the appropriate powers to run this command. The command you choose to run with the `EXEC` command may require specific powers for the objects affected by that command. If you do not have the required powers, the command fails and the CLI displays an error message.

Syntax

`EA [/DOMAIN:`*`domain`* `[/SERVER:`*`computername`*`|/MASTER]] EXEC [`*`"userspec" command [command_options`*`]]`

NOTE

- If the value for an option contains spaces, such as a user account name of `Jane Smith,` you must surround the option value with quotation marks. In this case, to specify a value for the userspec option, type userspec:`"Jane Smith"`.
- The *userspec, command,* and *command_options* parameters can include up to a total of 256 characters. If these parameters total more than 256 characters, the Administration server processes only the first 256 characters that you specify.

Options

`/DOMAIN:`*`domain`*	Specifies the name of the managed domain. If you do not specify this option, the CLI executes the specified command in the domain to which the consoles most recently connected. If you have not used the CLI before and you do not specify this option, the CLI connects to the domain where your user account resides. You can use wildcard characters to specify multiple domains.
`/SERVER:`*`computername`*	Specifies the name of an Administration server that manages the specified domain. If you specify a domain and do not specify a server, the CLI automatically locates the best available Administration server in the specified domain. You can prefix the specified computer name with two backslashes (`\\`).
`/MASTER`	Specifies the primary Administration server that manages the specified domain.
`targetdomain`	Refreshes the accounts or resource cache for the specified domain, domain member, or computer.
"userspec"	Specifies an explicit user account or list of user accounts on which to execute the command. This option is often a list of user account specifications that you generate using wildcard characters or the `@GroupUsers` filter.
command	Specifies a command to run against the users you specified for the *userspec* variable.

command_options	Specifies an option for the command you specified. This option often references the `@Target()` set operation function. For example, you can use the `@Target()` set operation to create a home directory using the user account name as part of the directory path.

EXEC Example 1

To move the home directories for all user accounts in the `SALES` group from the `\\TREK1` server to the `\\TREK2` server, enter the following commands:

```
EA EXEC @GroupUsers(SALES) XCOPY /O \\TREK1\USERS\@Target()
\\TREK2\USERS\@Target()
EA USER @GroupUsers(SALES) UPDATE HOMEDIR:\\TREK2\USERS\@Target()
EA EXEC @GroupUsers(SALES) DELTREE \\TREK1\USERS\@Target()
```

Administrators often need to reconfigure systems and relocate files and user account directories as system capacities and usages change. In this example, the first command runs the `XCOPY /O` command for each user account in the `SALES` group. The `XCOPY` command copies the home directory data for each user from the `\\TREK1` server to the `\\TREK2` server.

The second command (the `USER` command) changes the user account information to point the home directory to this new location.

The third command runs the `DELTREE` command for each user account in the `SALES` group. The `DELTREE` command deletes all the previous home directory data for these users on the `\\TREK1` server.

NOTE: The `EXEC` command allows you to run commands other than CLI commands. The `XCOPY` and `DELTREE` commands in this example are not CLI commands. This example outlines how the `EXEC` command allows you to run the `XCOPY` and `DELTREE` commands.

EXEC Example 2

Rather than separately specifying each of these commands, combine the three commands from the previous example into a single script file (`.bat` or `.cmd`) and use the `EXEC` command to run the script file.

To perform the actions in the previous example using a single script file, enter:

```
EA EXEC @GroupUsers(SALES) MOVEHD @Target()
```

In this example, the `MOVEHD.cmd` file contains the following lines:

```
XCOPY /O \\TREK1\USERS\%1 \\TREK2\USERS\%1
EA USER %1 UPDATE HOMEDIR:\\TREK2\USERS\%1
DELTREE \\TREK1\USERS\%1
```

GROUP Command

The `GROUP` command allows you to create, clone, modify, display, and delete groups.

Required Powers and Permissions

The different tasks you can perform with the GROUP command require different powers. The following table identifies the powers required for each task.

Tasks	Required Powers
Creating a new group	Create Group and Modify All Properties In order to create a group in an ActiveView, the AA must be associated with the ActiveView.
Cloning a group	Clone Group and Modify All Properties
Adding a member	Add a Member Modify Group Memberships Both the new member and the group must exist in the same ActiveView.
Removing a member	Add Object to Group
Updating the group description	Modify General Group Properties
Renaming a group	Modify Group Name
Displaying group information	View All Group Properties
Deleting a Group	Delete Group

Syntax

```
EA [/DOMAIN:domain [/SERVER:computername|/MASTER]] GROUP target CREATE {OU:ouname}
{CN:commonname} [GLOBAL|LOCAL|UNIVERSAL|LOCALDIST|GLOBALDIST|UNIVERSALDIST]
[CLONE:{"group"}] [COMMENT:"comment"] [TERRITORIES:"activeview"]
EA [/DOMAIN:domain [/SERVER:computername|/MASTER]] GROUP target MEMBERADD "member"
EA [/DOMAIN:domain [/SERVER:computername|/MASTER]] GROUP target MEMBERREMOVE
"member"
EA [/DOMAIN:domain [/SERVER:computername|/MASTER]] GROUP target DISPLAY
[OU:ouname] ["member"] [ALLMEMBERS]
EA [/DOMAIN:domain [/SERVER:computername|/MASTER]] GROUP target UPDATE
{[NAME:newname]|[COMMENT:"comment"]|[CN:commonname]}EA [/DOMAIN:domain [/
SERVER:computername|/MASTER]] GROUP target DELETE
```

NOTE: If the value for an option contains spaces, such as an OU name of `Sales and Marketing Consultants`, you must surround the option value with quotation marks. In this case, to specify a value for the `OU` option, type `OU:OU="Sales and Marketing Consultants",DC=Houston,DC=local`.

Verbs

CREATE Creates a new global or local group. The required `GLOBAL` or `LOCAL` keyword specifies whether the group is global or local. You can also specify `UNIVERSAL`, `LOCALDIST`, `GLOBALDIST`, or `UNIVERSALDIST` instead. There is no default group type.

MEMBERADD Adds the specified members to the specified group. You can add multiple accounts to a group. To specify multiple accounts, use the following syntax: `MEMBERADD` *accountA*,*accountB*

MEMBERREMOVE Removes the specified members from the specified group. This verb does not delete the group member or group itself. You can remove multiple accounts from a group. To specify multiple accounts, use the following syntax: `MEMBERADD` *accountA*`,`*accountB*

DISPLAY Displays the specified group names, as well as the group comments. If you specify an OU of a Microsoft Windows domain, the CLI lists all groups contained in the specified OU.

UPDATE Updates the specified group information for the specified group. The group name specification can be a list of *wildcards*. If you rename a group, the Administration server does not rename the non wildcard rules that identify this group. However, the rule will match the renamed group because the Administration server uses the group SID to identify the group. If you rename a group that is included in ActiveViews through wildcard specifications, that group may no longer be included in the same ActiveViews. The Administration server ensures that a renamed group remains in at least one ActiveView in which the AA has one or more powers. The Administration server also ensures that the ActiveView that includes the renamed group does not give the AA more powers over the renamed group.

DELETE Deletes the specified group. When you delete a group, the Administration server also deletes all group rules that exactly match the deleted group in all ActiveViews. The Administration server does not delete the group members. If the Recycle Bin is disabled for the specified domain, the Administration server permanently deletes the group when you delete a group. If the Recycle Bin is enabled for the specified domain, the deleted group is transferred to the Recycle Bin and can be restored or permanently deleted later. If you permanently delete a group, you cannot return access capabilities for that group simply by creating a new group with the same name. Microsoft Windows uses an internal Security Identifier (SID) to refer to a group. When you create a group, Microsoft Windows assigns a unique SID to that group, rather than generating the SID from the group name.

Options

/DOMAIN:*domain* Specifies the name of the managed domain. If you do not specify this option, the CLI displays the information for the domain to which the consoles most recently connected. If you have not used the CLI before and you do not specify this option, the CLI connects to the domain where your user account resides. You can use wildcard characters to specify multiple domains.

/SERVER:*computername* Specifies the name of an Administration server that manages the specified domain. If you specify a domain and do not specify a server, the CLI automatically locates the best available Administration server in the specified domain. You can prefix the specified computer name with two backslashes (`\\`).

/MASTER Specifies the primary Administration server that manages the specified domain.

target Specifies the name of the group you want to manage. The group name can contain wildcard characters. You can also precede the group name with a trusted domain or wildcard character, such as `*\*`, which targets all groups in the managed domains and trusted domains. When using the `CREATE` verb, the specified group must not already exist and you cannot specify wildcard characters. When using the `DISPLAY` verb, specify the target as `*` to list all groups in the specified OU.

CLONE:"*group*"	Specifies the group you want to clone. The Administration server uses the specified group as a template to create a new group. The Administration server then adds all members from the cloned group to the new group. If applicable, the Administration server also adds the new group to the ActiveViews of the original group. In a Microsoft Windows 2000 domain, you can specify the group type.
COMMENT:"*comment*"	Specifies the comment for the specified group. This comment is usually a description of the group.
OU:*ouname*	Specifies the name of a Microsoft Windows OU. ***If you want to specify the name of an enterprise OU***, use the following format: `OU=`*ou*`,DC=`*domain*`,DC=`*toplevel* For example, to specify the `SALES OU` in the `HOUSTON.LOCAL` domain, type: `OU:OU=SALES,DC=HOUSTON,DC=LOCAL` *If you want to specify the name of a built-in OU,* use the following format: `CN=`*ou*`,DC=`*domain*`,DC=`*toplevel* For example, to specify the `Users OU` in the `HOUSTON.LOCAL` domain, type: `OU:CN=Users,DC=HOUSTON,DC=LOCAL` When you create or clone a group, you must specify a Microsoft Windows OU. You do not need to specify an OU for a Microsoft Windows 2000 member server.
CN:"*commonname*"	Specifies the common name (display name) of the group.
NAME:"*group*"	Specifies the new account name of the group. This option allows you to rename an existing group. The powers you have in the selected ActiveView determine whether you can rename the group and the name you can assign to the group.
"*member*"	Specifies the name of the group member. When you use the `MEMBERADD` or `MEMBERREMOVE` verbs, use this option to specify which members should be added or removed from the group. If the Administration server does not find the specified member in the group, the CLI displays an error. If you specify the `DISPLAY` verb, this option specifies which group members the CLI displays. For local groups, the member specification can include a *domain/* prefix. If you do not specify this option, the CLI does not display any member information. You can use wildcard characters to specify the list of members you want to display. You can also use the `@GroupUsers` set operation to specify all members of a group. To add or remove a computer from group, enter the computer name in the following format: `[`*domainname*`\]`*computername*`[$]` If you are managing Microsoft Windows domains and want to add computer accounts created by either the DRA user interfaces or the native Windows 2000 Active Directory Users and Computers, append the computer name with a $. If you are managing Microsoft Windows domains and want to add computer accounts created with the NetIQ and LDAP ADSI providers, do not append the computer name with a $.
ALLMEMBERS	Displays group members the group contains, and includes group members in domains that DRA does not manage. To display all group members from the `EasternRegion` group, enter: `EA GROUP EasternRegion DISPLAY ALLMEMBERS`

GROUP Example 1

To create the EasternRegion global group in the SalesRegions OU of the USRegion domain and populate that group with the members of the Boston, Phila, and NYC groups, enter the following commands:

```
EA GROUP EasternRegion CREATE OU:OU=SalesRegions,DC=USRegion,DC=ACME,DC=COM GLOBAL
EA GROUP EasternRegion MEMBERADD
@GroupUsers(Boston),@GroupUsers(Phila),@GroupUsers(NYC)
```

GROUP Example 2

To create the WesternRegion group on the primary Administration server by cloning the EasternRegion group, enter:

```
EA /DOMAIN:USRegion /MASTER GROUP WesternRegion CREATE
OU:OU=SalesRegions,DC=USRegion,DC=ACME,DC=COM CLONE:EasternRegion
```

GROUP Example 3

To populate the EasternRegion group with members of the Boston group, enter:

```
EA GROUP EasternRegion MEMBERADD @GroupUsers(Boston)
```

GROUP Example 4

To add all members of the TXPoliticians global group to the Friends local group, enter:

```
EA GROUP Friends MEMBERADD @GroupUsers(TXPoliticians)
```

GROUP Example 5

To remove all members of the Players group from the Cleveland group, enter:

```
EA GROUP Cleveland MEMBERREMOVE @GroupUsers(Players)
```

GROUP Example 6

To update the comment for the Programmers local group, enter:

```
EA GROUP Programmers UPDATE COMMENT:"very unique individuals"
```

GROUP Example 7

To display all instances of a SmithJL user account in any domain in a group beginning with NYC_, enter:

```
EA GROUP NYC_* DISPLAY *\SmithJL
```

GROUP Example 8

To display all groups that contain members' names beginning with the letter m, enter:

```
EA GROUP * DISPLAY m*
```

GROUP Example 9

To display a list of all groups in the `Sales` OU of the SW domain, enter:

```
EA /DOMAIN:SW GROUP * DISPLAY OU:OU=Sales,DC=Houston,DC=SW,DC=US
```

GROUP Example 10

To delete the `Mammoth` group, enter:

```
EA GROUP Mammoth DELETE
```

INFO Command

The `INFO` command displays information about the Administration server to which you are connected, the DRA client computer, and your current user account.

Required Powers and Permissions

The Administration server does not require any powers or permissions to run this command.

Syntax

```
EA [/DOMAIN:domain [/SERVER:computername]] INFO
```

Options

**`/DOMAIN:`*`v`*	Specifies the name of the managed domain for which you want to refresh or display the accounts and resource caches. If you do not specify this option, the CLI refreshes the cache for the domain to which the consoles most recently connected. If you have not used the CLI before and you do not specify this option, the CLI connects to the domain where your user account resides. You can use wildcard characters to specify multiple domains.
**`/SERVER:`*`computername`*	Specifies the name of an Administration server that manages the specified domain. If you specify a domain and do not specify a server, the CLI automatically locates the best available Administration server in the specified domain. You can prefix the specified computer name with two backslashes (`\\`).

INFO Example 1

To display information about the `NORTHEAST` domain, enter:

```
EA /DOMAIN:NORTHEAST INFO
```

INFO Example 2

To display information about the `NORTHEAST` domain by connecting to the `PIT SERVER01` computer, enter:

```
EA /DOMAIN:NORTHEAST /SERVER:PIT SERVER01 INFO
```

ROLE Command

The ROLE command displays roles and role properties, enumerates roles matching a certain name, delegates and revokes roles to AVs. Properties for a role include the name, comment, description, type, and whether the role is assigned.

A role and role properties can be displayed. You can enumerate the roles with the DISPLAY command. The command is interpreted as a wildcard match of roles.

A role is a set of powers that provide the permissions required to perform a specific administration task, such as creating a user account or moving shared directories. To create a role, first define the job description. The job description provides the list of powers an AA needs to perform a task or complete a workflow.

A role can contain any set of powers you specify. Because you can choose from hundreds of powers, you have the flexibility to create roles that best fit your organization.

Required Powers and Permissions

To run these commands, you must have the appropriate powers, such as those included in the built-in DRA Administration and Manage Security Model roles.

Syntax

```
EA [/DOMAIN:domain [/SERVER:computername|/MASTER]] AV target DELEGATE {role:}
{admin:} [mode:]
EA [/DOMAIN:domain [/SERVER:computername|/MASTER]] AA target DISPLAY
EA [/DOMAIN:domain [/SERVER:computername|/MASTER]] AV target REVOKE {role:}
{admin:} [mode:]
```

Verbs

DELEGATE	Delegates a specific role to a specific admin in a given AV or AV wildcard match. The DELEGATE command can clearly distinguish the AA as a user, group, or AA Group. Specify users or groups by their Account Name or a wildcard that matches at most one user or group. Roles can be delegated or revoked, but cannot be defined.
DISPLAY	Displays any custom or built-in roles and role properties with name matching target. Properties for a role include the name, comment, description, type, and whether the role is assigned. The name parameter is always returned. By specifying only "name" on the command line, then only the name field will be returned. By default other fields are displayed
REVOKE	Revokes a specific role from a specific admin in a given AV or AV wildcard match. An error message is returned, if the delegation is not found on the Administration server.

Options

/DOMAIN:*domain*	Specifies the name of the managed domain. If you do not specify this option, the CLI provides information about the domain to which the consoles last connected. If you have not used a console on this computer and you do not specify this option, the CLI connects to the domain where your user account resides. You can use wildcard characters to specify multiple domains.
/SERVER:*computername*	Specifies the name of an Administration server managing the specified domain. If you specify a domain without a server, the CLI automatically locates the closest Administration server in the specified domain. You can prefix the specified computer name with two backslashes (\\)
/MASTER	Specifies the primary Administration server managing the specified domain.
{ADMIN:}	Specifies the *domain**Account Name* match, or the AA name. This must match only one object or the command will not be processed. This required option is used with the `DELEGATE` and `REVOKE` command.
commonfields	`ALL` Displays all fields. `ASSIGNED` Specifies whether the rule assigned or in use. `COMMENT:"`*text*`"` Specifies the comment for the ActiveView group. To display comments, specify `COMMENT` with the `DISPLAY` verb. `DESCRIPTION:"`*text*`"` Specifies the description for the ActiveView group. To display comments, specify `COMMENT` with the `DISPLAY` verb. `NAME` Specifies the new AV name of the ActiveView. `TYPE` Specifies the type of AV.
{ROLE:}	Specifies the role alias or role name that the client wants to assign to the target AV match. This is interpreted as a role name match. This must match only one object or the command will not be processed. This required option is used with the `DELEGATE` and `REVOKE` command.
MODE:{B|BATCH}	Allows you to display GUID information for the specified Administration servers.

ROLE Example 1

To delegate a role to the target AV match, enter:

```
EA AV kt* DELEGATE role:helpdesk* admin:dep1
```

ROLE Example 2

To enumerate roles matching a certain name, enter:

```
EA ROLE "a*" DISPLAY type
```

SERVER Command

The `SERVER` command displays Administration server information.

Required Powers and Permissions

The Administration server does not require any powers or permissions to run this command.

Syntax

```
EA SERVER {BEST|MASTER|*} DISPLAY [ADVANCED]
```

Verbs

DISPLAY	Displays a list of Administration servers for the user's domain as well as information for each server.

Options

BEST	Specifies the closest Administration server for the user's domain. Specify an asterisk (*) to display information for all Administration servers managing the domain.
MASTER	Specifies the primary Administration server for the user's domain. Specify an asterisk (*) to display information for all Administration servers managing the domain.
ADVANCED	Allows you to display GUID information for the specified Administration servers.

SERVER Example 1

To display information about the primary Administration server for the managed domain, enter:

```
EA SERVER MASTER DISPLAY
```

SERVER Example 2

To display information, including GUID information, about all Administration servers for the managed domain, enter:

```
EA SERVER * DISPLAY ADVANCED
```

USER Command

The `USER` command allows you to create, clone, modify, display, and delete user accounts on an Administration server.

Required Powers and Permissions

The different tasks you can perform with the USER command require different powers. The following table identifies the powers you need for each task.

Tasks	Required Powers
Creating or cloning a user account	◆ Add New User to Group ◆ Clone User and Modify All Properties ◆ Create User and Modify All Properties
Adding a mailbox for an existing user account	Create Exchange mailbox and modify all properties.
Updating user account or mailbox properties	The powers required to update user account properties depends on what properties you want to update.
Displaying user account properties	View All User Properties
Deleting a user account	Delete User Account

Syntax

```
EA [/DOMAIN:domain [/SERVER:computername|/MASTER]] USER target CREATE  {OU:ouname}
{PASSWORD:password} [fields] [CLONE:"username"] [mailboxfields] [MBDIRNAME]
[GROUPS:"groupname"]
EA [/DOMAIN:domain [/SERVER:computername|/MASTER]] USER target MBCLONE
[CLONE:"username"] [Code1Variable] [MBDIRNAME]
EA [/DOMAIN:domain [/SERVER: computername |/MASTER]] USER target UPDATE [fields]
[mailboxfields] [wtsfields] [NAME:newname] [PASSWORD:password]
EA [/DOMAIN:domain [/SERVER: computername |/MASTER]] USER target DELETE
EA [/DOMAIN:domain [/SERVER: computername |/MASTER]] USER target GROUPS
EA [/DOMAIN:domain [/SERVER: computername |/MASTER]] [/DELI:{TAB|x}] USER target
DISPLAY {OU:ouname} [fields] [wtsfields] [displayfields]
```

NOTE: If the value for an option contains spaces, such as an OU name of Sales and Marketing Consultants, you must surround the option value with quotation marks. In this case, to specify a value for the OU option, type `OU:OU="Sales and Marketing Consultants",DC=Houston,DC=local`.

Verbs

CREATE	Creates the specified user account.
MBCLONE	Creates a mailbox for the specified user account by cloning the existing mailbox for the user account identified by the CLONE:*username* option.
UPDATE	Updates the specified attributes of an existing user account. You can update only the properties for which you have the required powers to modify.
GROUPS	Displays the groups to which the specified user account belongs.

DISPLAY

Displays the existing user account information. If you do not identify specific field names, the CLI displays the values for the user name, comment, and user comment fields. The CLI always displays the user name field. If you specify an OU of a Microsoft Windows 2000 domain, the CLI lists all user accounts contained in the specified OU.

DELETE

Deletes the specified user accounts.

When you delete a user, the Administration server automatically deletes all user rules that exactly match (not through a wildcard rule specification) the deleted user in all ActiveViews.

If the Recycle Bin is disabled for the specified domain, the Administration server permanently deletes the user account when you delete a user account.

If the Recycle Bin is enabled for the specified domain, the deleted user account is transferred to the Recycle Bin and can be restored or permanently deleted later.

If you permanently delete a user account, you cannot return access capabilities for that account simply by creating a new user account with the same name. Microsoft Windows uses an internal Security Identifier (SID) to refer to a user account. When you create a user account, Microsoft Windows assigns a uniue SID to that account, rather than generating the SID from the user account name.

You can use policy to configure the Administration server to also delete the associated home directory or mailbox.

Options

/DOMAIN:*domain*

Specifies the name of the managed domain. If you do not specify this option, the CLI uses the domain to which the consoles most recently connected. If you have not used the CLI before and you do not specify this option, the CLI connects to the domain where your user account resides. You can use wildcard characters to specify multiple domains.

/SERVER:*computername*

Specifies the name of an Administration server that manages the specified domain. If you specify a domain and do not specify a server, the CLI automatically locates the best available Administration server in the specified domain. You can prefix the specified computer name with two backslashes (\\).

/MASTER

Specifies the primary Administration server that manages the specified domain.

/DELI:{TAB|x}

Specifies the delimiter character that the CLI uses to separate the displayed field values. This option allows you to format the output when you redirect the results to a file. You can then import the file into a database or spreadsheet program for further analysis and reporting. You can specify any delimiter character. To specify a tab as the delimiter character, type TAB.

OU:*ouname*

Specifies the name of a Microsoft Windows 2000 OU. ***If you want to specify the name of an enterprise OU***, use the following format: OU=*ou*,DC=*domain*,DC=*toplevel* For example, to specify the SALES OU in the HOUSTON.LOCAL domain, type: OU:OU=SALES,DC=HOUSTON,DC=LOCAL ***If you want to specify the name of a built-in OU***, use the following format: CN=*ou*,DC=*domain*,DC=*toplevel* For example, to specify the Users OU in the HOUSTON.LOCAL domain, type: OU:CN=Users,DC=HOUSTON,DC=LOCAL When you create or clone a user, you must specify a Microsoft Windows 2000 OU. You do not need to specify an OU for a Microsoft Windows 2000 member server.

target

Specifies the user account name or logon name for the user account you want to create or manage. If you are creating a new user account, you must specify a single user account name. You can specify wildcard characters with all verbs except the CREATE verb. When using the DISPLAY verb, specify the target as * to list all users in the specified OU.

CLONE:"*username*"

Specifies the user account to use as a template for the new user account. The Administration server copies the field values and group memberships from the specified user account and uses them as defaults for the new user account. The Administration server sets any fields not specified in this USER command, except the password field, to the value of the specified user account you want to clone.

GROUPS:"*groupname*"

Specifies the groups to which you want to add the new user account as a member.

NAME:"*name*"

Specifies a new common name for the user account. This option allows you to rename an existing user account.

fields

Specifies the fields or options that you want to specify, modify, or display for the specified user account. When you specify one or more of fields with the DISPLAY verb, specify the field name without any value. For example, to display the user account comment, specify COMMENT. You can specify the following field values:

ACTIVE:{Y|:N} Specifies whether the account is enabled (:Y) or disabled (:N). The default is enabled (Y).

CODEPAGE:*nnn* Specifies the code page you want to use to display characters. The default is 0, which specifies the code page configured for the local computer.

COMMENT:"*text*" Specifies the comment for the user account. To display comments, specify COMMENT with the DISPLAY verb.

COUNTRYCODE:*nnn* Specifies the country code number. The default is 0.

DIALCALLBACK:*telephonenumber* Specifies the telephone number for the AdminSetCallBack value of the DIALFLAGS field.

DIALFLAGS:[DialinPrivilege,]callbacksetting Specifies the dial-in privileges for the user account. If you do not specify the DialinPrivilege option, the Administration server disables the dial-in privileges. You can use the following values for the *callbacksetting* value:

AdminSetCallBack Directs the server to call the user at the telephone number specified by the DIALCALLBACK field. The server calls the user back only at the specified number. CallerSetCallBack Directs the server to prompt the user for a telephone number. NoCallBack Disables the call back function for the user account. This is the default setting.

DISPName:"*displayname*" Specifies the display name of a Microsoft Windows user account.

EXPIRES:{*date*|NEVER} Specifies an expiration date and time for the user account. Specify dates in the following format: YYYY MM DD,hh:mm:ss You can truncate the date at any point, after which the Administration server completes the specification with the lowest allowable value. For example, if you specify 2002-1, the Administration server sets the expiration date to 2002-1 01,00:00:00. If you specify NEVER, the Administration server sets no expiration date for the user account.

FIRSTNAME:"*givenname*" Specifies the first name of the user account.

FULLNAME:"*name*" Specifies the full name of a Microsoft Windows user account.

HOMEDIR:"*path*" Specifies the UNC path of the home directory. If you want to map a drive letter to a location, you must specify the HOMEDIRDRIVE and the HOMEDIR options to identify the mapping. DRA allows you to use %username% to represent the current target. However, when you use the %username% variable in the CLI, Microsoft Window 2000 defines the %username% variable as the currently logged on user.

HOMEDIRDRIVE:"*x*:" Specifies the drive letter you want to map to the home directory (HOMEDIR:) when the user logs on. To clear the mapped home directory for a user account, specify a space instead of a drive letter. HOMEDIRREQ:{Y|N} Specifies whether a home directory is required for a user account. The default is Yes (Y).

INITIALS:"*initials*" Specifies the initials of the user account.

LASTNAME:"*sn*" Specifies the last name of the user account.

MIDDLENAME:"*middlename*" Specifies the middle name of the user account.

PASSWORDCHG:{Y|N} Specifies whether the user can change the user account password. The default is Yes (Y).

PASSWORDEXPIRED:{Y|N} Specifies whether the user must change the password the next time the user logs on. The default is No (N). If you specify Yes (Y), the user must change the password. If you specify PASSWORDNOEXPIRE:Y for a user account, you cannot specify PASSWORDEXPIRED:Y for the same user account.

PASSWORDNOEXPIRE:{Y|N} Specifies whether the user account password never expires (Y). The default is No (N).

PASSWORDREQ:{Y|N} Specifies whether the user account must have a password. The default is Yes (Y).

PRIMARYGROUP:"*group*" Specifies the primary group for the user account. The primary group must be a global group. In addition, the user account must be a member of the group before you can specify the group as the primary group for the user account. You cannot remove a user account from the primary group. Use primary groups mainly for POSIX compatibility

PROFILEPATH:"path" Specifies the path of the logon profile for the user account. To specify a path that ends with a backslash (\), such as "C:\PROFILE\", you must specify two backslashes: C:\PROFILE\\. A specified share cannot end with a backslash.

SCRIPTPATH:"*path*" Specifies the location of the logon script for the user account relative to the %SYSTEMROOT%\SYSTEM32\REPL\IMPORT\SCRIPTS directory. This script is run when the identified user logs on. Do not specify a file name with a UNC or drive letter.

TIMES:{times|ALL} Specifies the times during which a user account can log on. Specify the times value in 1 hour increments and in the following format: *day[day][,day[day]], time[time][,time[time]]* Abbreviate the name of the day. Specify hour values from 0 to 24, based on a 24 hour clock. If you specify 4-8, the user can log on from 4:00 AM until 7:59 AM. If you specify ALL, the user can log on at any time of the day. If you do not specify any value, the user can never log on. Separate day and time entries with a comma (,), and separate multiple day and time entries with a semicolon (;). For example, to allow a user to log on any time except from 4:00 PM to 8:00 PM on Sundays, specify: sun,0 16;sun,20 24;mon sat,0 24

UNLOCK:Y Unlocks a locked user account.

USERCOMMENT:"*comment*" Sets the user account comment. If a comment contains a space, such as "Sales and Marketing", you must surround the comment with quotation marks.

WORKSTATIONS:*computername* Lists as many as eight computers from which a user account can log on to the network. Separate each workstation name with a comma (,). If you do not specify any computer names, the user account can log on from any computer.

WTSProfilePath:"*path*" Specifies the location of the Microsoft Windows terminal services profile path for the specified user account.

mailboxfields Specifies the properties of the mailbox for the specified user account. You can use the following values to specify the mailbox properties you want to create or change.

MBALIAS:"*alias*" Specifies the alias for the mailbox.

MBDIRNAME:"*directory*" Specifies the directory where you want to store the mailbox. You can specify this field only when cloning a user account or mailbox. You cannot specify this field when using the UPDATE verb.

MBFIRSTNAME:"*firstname*" Specifies the first name for the mailbox.

MBLASTNAME:"*lastname*" Specifies the last name for the mailbox.

MBINITIALS:*initials* Specifies the initials for the mailbox.

wtsfields Specifies the Windows Terminal Server (WTS) properties for the specified user account.

WTSALLOW:{Yes|No} Specifies whether the user can log on to the Terminal Server.

WTSHOMEDIR:"*path*" Specifies the UNC path of the home directory for the user when that user logs on to the Terminal Server. If you want to map a drive letter to the location, you must specify the WTSHOMEDIR and the WTSHOMEDIRDRIVE options. If you do not specify this option, the Administration server assigns the user account home directory to the WTSHOMEDIR option.

`WTSHOMEDIRDRIVE:"`*x:*`"` Specifies the drive letter you want to map to the WTS home directory (`WTSHOMEDIR`) when the user logs on to the Terminal Server. To clear the mapped WTS home directory for a user account, specify a space instead of a drive letter.

`WTSPROFILEPATH:"`*path*`"` Specifies the path and name of the user profile to use when the user logs on to Terminal Services.

`WTSCLIENTDRIVES:{Yes|No}` Specifies whether Windows Terminal Services reconnects mapped drives after the user logs on, for Citrix ICA clients. This setting does not apply to RDP clients.

`WTSCLIENTPRINTERS:{Yes|No}` Specifies the Terminal Services server to download and install the printer driver for the local client printer when the user logs on to a Terminal Services session.

`WTSPRINTERDEFAULT:{Yes|No}` Specifies that the Terminal Services session default to the main client printer. Selecting this option prevents Terminal Server from downloading and installing multiple printer drivers when users log on to a Terminal Services session.

displayfields Specifies the fields for which you want the CLI to display information. You cannot specify these fields when you create or update a user account. Use the following values to specify the information you want to display:

`ALL` Displays all user account fields, except the password field.
`BADPWCOUNT` Displays the number of invalid passwords currently outstanding for this user account. This count is cleared once the user enters a valid password.

`DISPNAME` Displays the display name, or friendly name, of this user account.

`LASTLOGON` Displays the last time the user logged on and the domain controller that validated the log on. The Administration server periodically consolidates this information from all domain controllers.

`NAME` Displays the name of the user account. Specify this field if you want to display only the user account name. If you specify any other fields, the CLI automatically displays the user account name.

`NETWARE` Displays the NetWare compatibility information of this user account. Specifying this field displays all the NetWare compatibility fields.

`NUMLOGONS` Displays the number of times the PDC has validated a logon attempt for that user account. The BDC also validates logon attempts, so this number is not an indicator of how many times a user actually logged on.

`PASSWORDAGE` Displays the time interval since the user, Administrator, or AA last set the password.

`USERID` Displays the RID of the user account. The RID is an internal numeric identifier for the user account.

USER Example 1

To create the `JASmith` user account on the primary Administration server for the `SPACE` domain, and add the account to the `Sales Personnel` group, enter:

```
EA /DOMAIN:SPACE /MASTER USER JASmith CREATE OU:OU=Jupiter,DC=Space,DC=com
FULLNAME:"Jane A. Smith" COMMENT:President GROUPS:"Sales Personnel"
```

NOTE: The Administration server sets all fields (*commonfields* and *mailboxfields*) that you did not specify to the default values.

USER Example 2

To create the JohnDoe user account and mailbox on the primary Administration server for the SPACE domain by cloning the JaneDoe user account, enter:

```
EA /DOMAIN:SPACE /MASTER USER JohnDoe CREATE OU:OU=Jupiter,DC=Space,DC=com
FULLNAME:"John Q. Doe" CLONE:JaneDoe MBALIAS:"John Doe" MBFIRSTNAME:John
MBLASTNAME:Doe PASSWORD:1234 GROUPS:"Western Region"
```

NOTE: If you defined proxy generation rules for this domain, use the FIRSTNAME, LASTNAME, MIDDLENAME, and INITIALS fields to specify a unique email address for the target account.

USER Example 3

To unlock the JASmith user account enter:

```
EA USER JASmith UPDATE UNLOCK:Y
```

NOTE: This example does not update any other properties.

USER Example 4

To change the LBond logon name to LDoe, and change the mailbox last name to Doe, enter:

```
EA USER LBond UPDATE NAME:LDoe MBLASTNAME:Doe
```

USER Example 5

To clone the JSmith user account mailbox to create a mailbox for the LBond user account, enter:

```
EA USER LBond CREATE OU:OU=agents,DC=london,DC=uk CLONE:JSmith PASSWORD:Phooey
MBFIRSTNAME:Lisa MBLASTNAME:Bond
```

To add more information about the LBond user account, use the UPDATE verb once the Administration server completes creating the LBond user account.

USER Example 6

To display a list of all user accounts in the Sales OU of the SW domain, enter:

```
EA /DOMAIN:SW USER * DISPLAY OU:OU=Sales,DC=Houston,DC=SW,DC=US
```

USER Example 7

To save a tab delimited list of all user accounts, along with the last logon timestamp for each user account, in the D:\TEMP\USERS.TXT file, enter:

```
EA /DELI:TAB USER * DISPLAY LASTLOGON > D:\TEMP\USERS.TXT
```

Directory and Resource Reporting provides last logon statistic reports to help you view this important last logon information.

USER Example 8

To delete the `JASmith` user account, enter:

```
EA USER JASmith DELETE
```

WARNING: If you delete a user account, you cannot return access capabilities for that user simply by creating a new user account with the same name. Microsoft Windows 2008 or later uses an internal Security Identifier (SID) to refer to a user account. When you create a user account, Microsoft Windows 2008 or later assigns a SID to that user account. Microsoft Windows 2008 or later does not generate the SID from the user account name.

WHOAMI Command

The `WHOAMI` command displays information, such as the total number of managed domains and user accounts, about both the DRA client computer and the Administration server. This information is often important when diagnosing problems or when reporting any problems to NetIQ Technical Support.

Required Powers and Permissions

You do not need any special powers or permissions to run this command.

Syntax

`EA [/DOMAIN:`*`domain`* `[/SERVER:`*`computername`*`|/MASTER]] [/POWERS] WHOAMI`

Options

`/DOMAIN:`*`domain`*	Specifies the name of the managed domain. If you do not specify this option, the CLI displays the information for the domain to which the consoles most recently connected. If you have not used the CLI before and you do not specify this option, the CLI connects to the domain where your user account resides. You can use wildcard characters to specify multiple domains.
`/SERVER:`*`computername`*	Specifies the name of an Administration server that manages the specified domain. If you specify a domain and do not specify a server, the CLI automatically locates the best available Administration server in the specified domain. You can prefix the specified computer name with two backslashes (\\).
`/MASTER`	Specifies the primary Administration server that manages the specified domain.
`/POWERS`	Displays the powers of the user account logged on to the DRA client computer. This option displays all the powers that an AA has in a domain. It does not display the powers for each ActiveView. Therefore, the AA may not have all the displayed powers in each ActiveView.

WHOAMI Example 1

To retrieve information for the MARS Administration server in the SPACE domain, enter:

```
EA /DOMAIN:SPACE /SERVER:MARS WHOAMI
```

WHOAMI Example 2

To retrieve information for the primary Administration server in the SPACE domain, enter:

```
EA /DOMAIN:SPACE /MASTER WHOAMI
```

WHOAMI Example 3

To display all the powers of the user account logged on to the Administration CLI computer, enter:

```
EA /POWERS WHOAMI
```

B Available Utilities

These sections discuss the Diagnostic Utility, Deleted Object Utility, and Recycle Bin Utility provided with DRA and ExA.

Diagnostic Utility

The Diagnostic Utility gathers information from your Administration server to help diagnose issues with DRA and ExA. Use this utility to provide log files to your NetIQ Technical Support representative. The Diagnostic Utility provides a wizard interface that guides you through setting log levels and collecting diagnostic information.

Accessing the Diagnostic Utility

You can access the Diagnostic Utility from any Administration server computer. However, you should run the Diagnostic Utility on the Administration server where you are experiencing the issue.

By default, the setup program installs the Diagnostic Utility with the Administration server component. You cannot copy the utility to another folder and run the utility from the new folder.

To access the Diagnostic Utility:

1. Log on to the Administration server computer using the DRA Admin account.
2. Run `DRADiagnosticUtil.exe` from the `Program Files (x86)\NetIQ\DRA` folder.

Understanding the Diagnostic Information

You can select which diagnostic information you want to collect or let the Diagnostic Utility collect the recommended data for a specific failure type. Based on the failure type you choose, the Diagnostic Utility gathers the following information from your Administration server:

ADSI logs

Collects diagnostic information about the DRA ADSI provider from Administration registry entries under `HKEY_Local_Machine\Software\WOW6432Node\Mission Critical Software\OnePoint\ADSI`.

APJS diagnostics

Collects diagnostic information about the Accounts Provider Job Scheduler and the tasks this provider controls.

Application Event log

Collects diagnostic information from the Windows server hosting the DRA server application Windows Application Event Log.

Automation scripts

Collects diagnostic information about scripts running for automation triggers.

Domain cache files

Collects the domain cache files on this Administration server.

Domain cache logs

Collects diagnostic information about the domain cache on this Administration server.

DRA registry settings

Collects diagnostic information from the entire Administration registry entry under `HKEY_Local_Machine\Software`.

Dr. Watson logs

Collects debugging information about applications you run on this computer. To enable this option, install and set up Dr. Watson as your default debugging tool. For more information about using Dr. Watson as your default debugging tool, see the *Admnistration Installation Guide*.

File times and sizes

Collects the time stamp and size for each file in the `Program Files (x86)\NetIQ\DRA` folder on your Administration server.

IIS service logs

Collects diagnostic information about IIS application settings.

Install information

Collects diagnostic information from a subset of the entire registry.

Licensing information

Collects information about the current effective license.

Server state information

Collects internal Administration server information. If the Administration server is not currently running, the utility is unable to collect any server state information. In addition the data collection process fails. If the collection process fails, deselect **Server State Information** and re-run the utility.

Server logs

Collects diagnostic information from the Administration server logs. The `McsAdminSvc_Startup.nq1` file logs service initialization data and the `McsAdminSvc.nq1` file logs general Administration server data.

Services configurations

Collects a list of the services running on the Administration server including the service type, status, and logon parameters.

System Event log

Collects diagnostic information from the Windows server hosting the DRA Server Application Windows System Event Log.

Web Console logs

Collects diagnostic information about event log entries, performance metrics, and the virtual directory settings for the Web Console.

Win32 console log files

Collects diagnostic information about the Account and Resource Management console and the Delegation and Configuration console.

Win32 console settings

Collects the client options and other settings for the Account and Resource Management console and the Delegation and Configuration console.

Configuring Log Settings

You can configure different log settings, such as logging levels, for individual DRA components on each Administration server computer. The logging level determines the quantity and detail of the diagnostic information you can collect. For example, if you set a high logging level, DRA logs additional information that can be used to help diagnose a more complex issue. Before collecting diagnostic information, ensure the log level is appropriate for the issue you are experiencing.

To configure log settings:

1. Start the Diagnostic Utility. For more information, see “Accessing the Diagnostic Utility” on page 77.
2. Click **Enable or change logging for a DRA component**, and then click **Next**.
3. Specify the appropriate log level for the DRA component about which you want to collect diagnostic information.
4. To specify additional settings, such as the log file location, complete the following steps:
   - **4a** Click **Settings** for the appropriate DRA component.
   - **4b** Specify which settings you want to the Diagnostic Utility to use when logging information about this component.
   - **4c** Click **OK**.
5. To apply these settings, click **Finish**.

Collecting Diagnostic Information

You can select which diagnostic information you want to collect or let the Diagnostic Utility collect the recommended data for a specific failure type. Based on the failure type you choose, the Diagnostic Utility gathers the appropriate information from your Administration server.

By default, the Diagnostic Utility outputs information as a `.zip` file. When you call NetIQ Technical Support, you may need to provide this file to your Technical Support representative. By default, DRA puts log files in the following location on the Administration server computer: `C:\Documents and Settings\`*`username`*`\Local Settings\Application Data\NetIQ\DRA\Logs`. For more information, see “Finding Specific Log Files” on page 80.

For more information about which diagnostics to select, see “Understanding the Diagnostic Information” on page 77 or consult your Technical Support representative.

TIP: If Windows UAC is enabled, to run as Administrator you must select the option from the Shift + Right-click menu.

To collect diagnostic information:

1. Start the Diagnostic Utility. For more information, see “Accessing the Diagnostic Utility” on page 77.
2. Click **Collect diagnostics information about a DRA failure**, and then select the failure type that best represents your issue.
3. Click **Next**.

4 Review the diagnostic settings for the selected failure type, and then click **Next**. You can select or clear a setting, or choose another failure type.

5 Specify the name and location of the `zip` file.

6 To begin collecting diagnostics, click **Finish**.

Viewing APJS Diagnostics

The Accounts Provider Job Scheduler (APJS) is part of the Administration server component. APJS controls schedules and records the status of various domain and server tasks, such as accounts cache refreshes and synchronization across multiple Administration servers. APJS diagnostics provide detailed information about these server and domain activities.

To view Accounts Provider diagnostics:

1 Start the Diagnostic Utility. For more information, see "Accessing the Diagnostic Utility" on page 77.

2 Click **View Accounts Provider Job Scheduler diagnostics**.

3 To update the diagnostic information, click **Refresh**.

4 Click **Finish**.

Viewing Lock Diagnostics

You can use the Diagnostic Utility to view the status of read and write locks on this Administration server. Locks occur during server synchronizations and some cache refreshes. During a lock, an Assistant Admin may not be able to view (read) or modify (write) managed objects. Lock diagnostics provide a count of pending read and write operations, as well as the last time the Administration server successfully released a lock.

To view lock diagnostics:

1 Start the Diagnostic Utility. For more information, see "Accessing the Diagnostic Utility" on page 77.

2 Click **View lock diagnostics for Read/Write and Replication locks**.

3 To update the diagnostic information, click **Refresh**.

4 Click **Finish**.

Finding Specific Log Files

You can use the Diagnostic Utility to find specific log files for a DRA component. By default, DRA puts log files in the following location on the Administration server computer: `C:\Documents and Settings\`*`username`*`\Local Settings\Application Data\NetIQ\DRA\Logs`. You can configure the log file location for individual DRA components. For more information, see "Configuring Log Settings" on page 79.

To find a specific log file:

1 Start the Diagnostic Utility. For more information, see "Accessing the Diagnostic Utility" on page 77.

2 Click **Enable or change logging for a DRA component**, and then click **Next**.

3 Click **Find Logs** for the appropriate DRA component.

4. To view a log file, select the file, and then click **File > Open**.
5. To print a log file, select the file, and then click **File > Print**.
6. Close the file view window, and then click **Finish**.

Deleted Objects Utility

This utility allows you to enable incremental accounts cache refresh support for a specific domain when the domain access account, such as the access account, is not an administrator. If the domain access account does not have read permissions on the Deleted Objects container in the domain, DRA cannot perform an incremental accounts cache refresh.

You can use this utility to perform the following tasks:

- Verify that the specified user account or group has read permissions on the Deleted Objects container in the specified domain
- Delegate or remove read permissions to a specified user account or group
- Delegate or remove the Synchronize directory service data user right to a user account
- Display security settings for the Deleted Objects container

By default, you can run the Deleted Objects Utility from the `Program Files (x86)\NetIQ\DRA` folder on your Administration server. You can install and run the Deleted Objects Utility on a computer that is not an Administration server. To install this utility, choose custom installation in the setup program. For more information about performing a custom installation, see the *Installation Guide*.

Required Permissions for Deleted Objects Utility

To use this utility, you must have the following permissions:

If you want to …	You need this permission …
Verify account permissions	Read Permissions access to the Deleted Objects container
Delegate read permissions on the Deleted Objects container	Administrator permissions in the domain where the Deleted Objects container is located
Delegate the Synchronize directory service data user right	Administrator permissions in the domain where the Deleted Objects container is located
Remove previously delegated permissions	Administrator permissions in the domain where the Deleted Objects container is located
Display security settings for the Deleted Objects container	Read Permissions access to the Deleted Objects container

Syntax for Deleted Objects Utility

`DRADELOBJSUTIL /DOMAIN:`*`DOMAINNAME`* `[/DC:`*`COMPUTERNAME`*`] {/DELEGATE:`*`ACCOUNTNAME`* `| /VERIFY:`*`ACCOUNTNAME`* `| /REMOVE:`*`ACCOUNTNAME`* `| /DISPLAY [/RIGHT]}`

Options for Deleted Objects Utility

You can specify the following options:

/DOMAIN:*domain*	Specifies the NETBIOS or DNS name of the domain where the Deleted Objects container is located.
/SERVER:*computername*	Specifies the name or IP address of the domain controller for the specified domain.
/DELEGATE:*accountname*	Delegates permissions to the specified user account or group.
/REMOVE:*accountname*	Removes permissions previously delegated to the specified user account or group
/VERIFY:*accountname*	Verifies permissions of the specified user account or group.
/DISPLAY	Displays security settings for the Deleted Objects container in the specified domain
/RIGHT	Ensures the specified user account or group has the Synchronize directory service data user right. You can use this option to delegate or verify this right. The Synchronize directory service data user right allows the account to read all objects and properties in the Active Directory.

NOTE

- If the name of the user account or group you want to specify contains a space, enclose the account name in quotation marks. For example, if you want to specify the Houston IT group, type "Houston IT".
- When specifying a group, use the pre-Windows 2000 name for that group.

Examples for Deleted Objects Utility

The following examples demonstrate sample commands for common scenarios.

Example 1

To verify that the `MYCOMPANY\JSmith` user account has read permissions on the Deleted Objects container in the `hou.mycompany.com` domain, enter:

```
DRADELOBJSUTIL /DOMAIN:HOU.MYCOMPANY.COM /VERIFY:MYCOMPANY\JSMITH
```

Example 2

To delegate read permissions on the Deleted Objects container in the `MYCOMPANY` domain to the `MYCOMPANY\DraAdmins` group, enter:

```
DRADELOBJSUTIL /DOMAIN:MYCOMPANY /DELEGATE:MYCOMPANY\DRAADMINS
```

Example 3

To delegate read permissions on the Deleted Objects container and the Synchronize directory service data user right in the `MYCOMPANY` domain to the `MYCOMPANY\JSmith` user account, enter:

```
DRADELOBJSUTIL /DOMAIN:MYCOMPANY /DELEGATE:MYCOMPANY\JSMITH /RIGHT
```

Example 4

To display security settings for the Deleted Objects container in the `hou.mycompany.com` domain using the `HQDC` domain controller, enter:

```
DRADELOBJSUTIL /DOMAIN:HOU.MYCOMPANY.COM /DC:HQDC /DISPLAY
```

Example 5

To remove read permissions on the Deleted Objects container in the `MYCOMPANY` domain from the `MYCOMPANY\DraAdmins` group, enter:

```
DRADELOBJSUTIL /DOMAIN:MYCOMPANY /REMOVE:MYCOMPANY\DRAADMINS
```

Recycle Bin Utility

This utility allows you to enable Recycle Bin support when you are managing a subtree of a domain. If the domain access account does not have permissions on the hidden NetIQRecycleBin container in the specified domain, DRA cannot move deleted accounts to the Recycle Bin.

NOTE: After using this utility to enable the Recycle Bin, perform a full accounts cache refresh to ensure the Administration server applies this change.

You can use this utility to perform the following tasks:

- Verify that the specified account has read permissions on the NetIQRecycleBin container in the specified domain
- Delegate read permissions to a specified account
- Display security settings for the NetIQRecycleBin container

By default, you can run the Recycle Bin Utility from the `Program Files (x86)\NetIQ\DRA` folder on your Administration server. You can install and run the Recycle Bin Utility on a computer that is not an Administration server. To install this utility, choose custom installation in the setup program. For more information about performing a custom installation, see the *Installation Guide*.

Required Permissions for the Recycle Bin Utility

To use this utility, you must have the following permissions:

If you want to …	You need this permission …
Verify account permissions	Read Permissions access to the NetIQRecycleBin container
Delegate read permissions on the NetIQRecycleBin container	Administrator permissions in the specified domain
Display security settings for the NetIQRecycleBin container	Read Permissions access to the NetIQRecycleBin container

Syntax for Recycle Bin Utility

`DRARECYCLEBINUTIL /DOMAIN:`*`DOMAINNAME`* `[/DC:`*`COMPUTERNAME`*`] {/DELEGATE:`*`ACCOUNTNAME`* `| /VERIFY:`*`ACCOUNTNAME`* `| /DISPLAY}`

Options for Recycle Bin Utility

The following options enable you to configure the Recycle Bin Utility:

`/DOMAIN:domain`	Specifies the NETBIOS or DNS name of the domain where the Recycle Bin is located.
`/SERVER:`*`computername`*	Specifies the name or IP address of the domain controller for the specified domain.
`/DELEGATE:`*`accountname`*	Delegates permissions to the specified account.
`/VERIFY:`*`accountnam`*`e`	Verifies permissions of the specified account.
`/DISPLAY`	Displays security settings for the NetIQRecycleBin container in the specified domain.

Examples for Recycle Bin Utility

The following examples demonstrate sample commands for common scenarios.

Example 1

To verify that the `MYCOMPANY\JSmith` user account has read permissions on the NetIQRecycleBin container in the `hou.mycompany.com` domain, enter:

```
DRARECYCLEBINUTIL /DOMAIN:HOU.MYCOMPANY.COM /VERIFY:MYCOMPANY\JSMITH
```

Example 2

To delegate read permissions on the NetIQRecycleBin container in the `MYCOMPANY` domain to the `MYCOMPANY\DraAdmins` group, enter:

```
DRARECYCLEBINUTIL /DOMAIN:MYCOMPANY /DELEGATE:MYCOMPANY\DRAADMINS
```

Example 3

To display security settings for the NetIQRecycleBin container in the `hou.mycompany.com` domain using the `HQDC` domain controller, enter:

```
DRARECYCLEBINUTIL /DOMAIN:HOU.MYCOMPANY.COM /DC:HQDC /DISPLAY
```

C Custom Powers

You can use the Custom Power Wizard to granularize the delegation model. The Custom Power Wizard supports building specific powers in DRA and ExA.

Understanding the Custom Powers

The Custom Power Wizard provides a way to quickly create Powers with specific properties matching the actions you want to delegate to roles.

The Custom Powers Wizard uses several conventions to help you build a custom power. The following sections define these conventions and several specific characteristics of the custom power.

During the creation of a new Custom Power in DRA, you have the ability to specify object types, actions, and properties over which you can grant access with the new power. Furthermore, you have the option to select all properties, specific properties, on no properties for the object type the power will be associated with. By default, if you selected Include all object properties you Custom Power will work without incident. If you select Include only listed properties, you should review the list of required, minimum property fields below to make sure the Custom Power will work.

Custom Power Properties

The following sections list the object types and associated actions that require a minimum set of properties to successfully create a new Custom Power. These are the minimum required properties for each object type in a new Custom Power.

User

The following table lists the action, additional permissions, and required properties of the User object type to show the required properties for the power to work without incident.

Action	Additional permissions	Required Properties
Sets the properties of a user	Disables the user account	`AccountDisabled`
Sets the properties of a user	Enables the user account	`AccountDisabled`

Creates a user	None selected	`givenName`
		`fullName`
		`displayName`
		`sn`
		`samAccountName`
		`userPrincipalName`
		`userPassword`

Action	Additional permissions	Required Properties

Creates a user	Enable email during user creation	`givenName`
		`fullName`
		`displayName`
		`sn`
		`samAccountName`
		`userPrincipalName`
		userPassword
		`homeMDB`
		`mailNickName`
		`legacyExchangeDN`

Creates a user	Add object to groups during user creation	`givenName`
		`fullName`
		`displayName`
		`sn`
		`samAccountName`
		`userPrincipalName`
		`userPassword`

Creates a user	Create Exchange 2008 or newer mailbox for created user account	homeMDB mailNickName givenName fullName displayName sn samAccountName userPrincipalName userPassword

Action	Additional permissions	Required Properties

Clones a user	None selected	`cn`
		`description`
		`displayName`
		`FullName`
		`givenName`
		`name`
		`samAccountName`
		`sn`
		`userPassword`
		`userPrincipalName`

Clones a user	Enable email for the cloned user	cn description displayName EmailAddress FullName givenName homeMDB legacyExchangeDN mailNickName name samAccountName sn userPassword userPrincipalName

Action	Additional permissions	Required Properties
Clone a user	Create Exchange 2008 or newer mailbox for the cloned user	cn description displayName EmailAddress FullName givenName homeMDB legacyExchangeDN mailNickName name samAccountName sn userPassword userPrincipalName

Group

The following table lists the action, additional permissions, and required properties of the Group object type to show the required properties for the power to work without incident.

Action	Additional permissions	Required Properties

Creates a group	None selected	`displayName`
		`groupType`
		`name`
		`samAccountName`

Creates a group	Enable email for the group you create	`displayName`
		`groupType`
		`name`
		`samAccountName`
		`legacyExchangeDN`
		`mailNickName`
		`msExchHideFromAddressLists`

Action	Additional permissions	Required Properties

Creates a group	Add the group you create to an ActiveView	`displayName`
		`groupType`
		`name`
		`samAccountName`

Clones a group	None selected	`displayName`
		`name`
		`samAccountName`

Clones a group	Add the group to an ActiveView during group clone	`displayName`
		`name`
		`samAccountName`

Dynamic Distribution Group

The following table lists the action, additional permissions, and required properties of the Dynamic Distribution Group object type to show the required properties for the power to work without incident.

Action	Additional permissions	Required Properties
Sets the properties of an Exchange Dynamic Distribution Group	No additional permissions available	
Gets the properties of an Exchange Dynamic Distribution Group	No additional permissions available	
Creates an Exchange Dynamic Distribution Group	No additional permissions available	
Clones an Exchange Dynamic Distribution Group	No additional permissions available	

Computer

The following table lists the action, additional permissions, and required properties of the Computer object type to show the required properties for the power to work without incident.

Action	Additional permissions	Required Properties

Creates a computer in the specified domain	No additional permissions available	`AccountDisabled`
		`samAccountName`
		`$McsAllowPreW2K`
		`$McsCanbeJoinedBy`

Contact

The following table lists the action, additional permissions, and required properties of the Contact object type to show the required properties for the power to work without incident..

Action	Additional permissions	Required Properties
Creates a contact	No additional permissions available	`givenName` `sn`

Creates a contact	Enable email for the contact you create	`givenName`
		`sn`
		`legacyExchangeDN`
		`mailNickName`

Creates a contact	Add the contact you create to groups	`givenName` `sn`
Clones a contact	No additional permissions available	`givenName` `sn`

Organizational Unit

The following table lists the action, additional permissions, and required properties of the Organizational Unit object type to show the required properties for the power to work without incident.

Action	Additional permissions	Required Properties
Creates an organizational unit	No additional permissions available	`name`
Clones an organizational unit	No additional permissions available	`name`

Published Printer

The following table lists the action, additional permissions, and required properties of the Published Printer object type to show the required properties for the power to work without incident.

Action	Additional permissions	Required Properties
Sets the properties for an ADprinter	No additional permissions available	
Retrieves information about an ADprinter	No additional permissions available	

Resource Mailbox

The following table lists the action, additional permissions, and required properties of the resource mailbox object type to show the required properties for the power to work without incident.

Action	Additional permissions	Required Properties
Updates a resource mailbox	No additional permissions available	
Gets the properties for a resource mailbox	No additional permissions available	
Creates a resource mailbox	Create resource mailbox for created user account	
Copy a resource mailbox	No additional permissions available	

D The Legacy Web Console

The legacy Web Console, which was superseded by a newer Web Console with the release of DRA 9.0.1, is still available for use. Consult the *NetIQ Directory and Resource Administrator and Exchange Administrator Installation Guide* for information about installing this version of the Web Console.

The legacy Web Console is a Web-based user interface that provides quick and easy access to many user account, group, computer, resource, and Microsoft Exchange mailbox tasks. You can also manage general properties of your own user account, such as the street address or cell phone number.

The legacy Web Console is easy to learn and simple to use, which makes it a great tool for occasional or beginning administrators. The Web Console provides step-by-step help as it guides you through each task. When you complete a task, it displays links to other related tasks, so you can quickly address an entire workflow. The Web Console displays a task only if you have the power to perform that task.

Starting the Legacy Web Console

You can start the Web Console from any computer running Internet Explorer. To start the Web Console, specify the appropriate URL in your Web browser address field or use the link provided in the Account and Resource Management console. For example, if you installed the Web component on the HOUserver computer, type `http://HOUserver/dra` in the address field of your Web browser.

NOTE: To display the most current account and Microsoft Exchange information in the Web Console, set your Web browser to check for newer versions of cached pages at every visit.

You can also start the Web Console from the DRA program group, and from the File menu in the Account and Resource Management console and the Delegation and Configuration console.

Using Quick Start to Solve Issues

Quick Start allows you to quickly and easily resolve account issues. You can view vital statistics and properties for a specific user account, computer, or group. You can then link to the appropriate task, such as resetting the password for a user account, which addresses your problem.

Customizing the Legacy Web Console

You can quickly and easily customize the Web Console in the following ways:

Modify provided tasks

For example, you can modify the update user's properties task to include a new field that manages a proprietary setting. You can hide specific tasks you do not want Assistant Admins (AAs) to use regardless of their delegated powers. You can also publish reports generated from Directory and Resource Reporting.

Develop new tasks

For example, you can develop a new update user's properties task that meets your unique administration needs. You can replace provided tasks with custom tasks without losing built-in functionality.

Modify workflows

For example, you can modify the Web Console framework and navigation, changing how AAs step through a given task. This flexibility allows you to add, remove, or move steps to create the exact solution you require.

Deploy multiple Web Console applications

You can install and configure multiple Web Console applications. For example, you can deploy one custom Web Console application for your Houston facility and another custom Web Console application for your Atlanta facility. Each application can support a unique set of tasks that meet the specific needs of your facility. For more information, see the Deploying DRA in Unique Environments Technical Reference. For more information about customizing the Web Console, see the Directory and Resource Administrator Software Development Kit.