ACS 5.x Policy Model

ACS 5.x is a policy-based access control system. The term *policy model* in ACS 5.x refers to the presentation of policy elements, objects, and rules to the policy administrator. ACS 5.x uses a rule-based policy model instead of the group-based model used in the 4.x versions.

This section contains the following topics:

Note See Functionality Mapping from ACS 4.x to ACS 5.8, page 2-4 for a mapping of ACS 4.x concepts to ACS 5.8.

Overview of the ACS 5.x Policy Model

The ACS 5.x rule-based policy model provides more powerful and flexible access control than is possible with the older group-based approach.

In the older group-based model, a *group* defines policy because it contains and ties together three types of information:

- Identity information—This information can be based on membership in AD or LDAP groups or a static assignment for internal ACS users.
- Other restrictions or conditions—Time restrictions, device restrictions, and so on.
- Permissions—VLANs or Cisco IOS privilege levels.

The ACS 5.x policy model is based on *rules* of the form:

If *condition* then *result*

For example, we use the information described for the group-based model:

If *identity-condition, restriction-condition* then *authorization-profile*

In ACS 5.8, you define conditions and results as global, shared objects. You define them once and then reference them when you create rules. ACS 5.8 uses the term *policy elements* for these shared objects, and they are the building blocks for creating rules.

Table 3-1 shows how the various policy elements define all the information that the old group contained.

Table 3-1 Information in Policy Elements

Information in ACS 4.x Group	Information in ACS 5.8 Policy Element
Identity information	• AD group membership and attributes • LDAP group membership and attributes • ACS internal identity groups and attributes
Other policy conditions	• Time and date conditions • Custom conditions
Permissions	Authorization profiles

A *policy* is a set of rules that ACS 5.x uses to evaluate an access request and return a decision. For example, the set of rules in an:

- *Authorization policy* return the authorization decision for a given access request.
- *Identity policy* decide how to authenticate and acquire identity attributes for a given access request.

ACS 5.x organizes the sequence of independent policies (a policy work flow) into an *access service*, which it uses to process an access request. You can create multiple access services to process different kinds of access requests; for example, for device administration or network access. For more information, see Access Services, page 3-5.

You can define simple policies and rule-based policies. Rule-based policies are complex policies that test various conditions. Simple policies apply a single result to all requests without any conditions.

There are various types of policies:

For more information on the different types of policies, see Types of Policies, page 3-4.

For more information about policy model terminology, see Policy Terminology, page 3-2.

Related Topics

- Policies and Identity Attributes, page 3-17
- Flows for Configuring Services and Policies, page 3-19

Policy Terminology

Table 3-2 describes the rule-based policy terminology.

Table 3-2 Rule-Based Policy Terminology

Term	Description
Access service	Sequential set of policies used to process access requests. ACS 5.x allows you to define multiple access services to support multiple, independent, and isolated sets of policies on a single ACS system. There are two default access services: one for device administration (TACACS+ based access to the device shell or CLI) and one for network access (RADIUS-based access to network connectivity).
Policy element	Global, shared object that defines policy conditions (for example, time and date, or custom conditions based on user-selected attributes) and permissions (for example, authorization profiles). The policy elements are referenced when you create policy rules.
Authorization profile	Basic permissions container for a RADIUS-based network access service, which is where you define all permissions to be granted for a network access request. VLANs, ACLs, URL redirects, session timeout or reauthorization timers, or any other RADIUS attributes to be returned in a response, are defined in the authorization profile.
Shell profile	Basic permissions container for TACACS+ based device administration policy. This is where you define permissions to be granted for a shell access request. IOS privilege level, session timeout, and so on are defined in the shell profile.
Command set	Contains the set of permitted commands for TACACS+ based, per-command authorization.
Policy	Set of rules that are used to reach a specific policy decision. For example, how to authenticate and what authorization to grant. For any policies that have a default rule, a policy is a first-match rules table with a default rule for any request which does not match any user-created rules.
Identity policy	ACS 5.8 policy for choosing how to authenticate and acquire identity attributes for a given request. ACS 5.8 allows two types of identity policies: a simple, static policy, or a rules-based policy for more complex situations.
Identity group mapping policy	Optional policy for mapping identity information collected from identity stores (for example, group memberships and user attributes) to a single ACS identity group. This can help you normalize identity information and map requests to a single identity group, which is just a tag or an identity classification. The identity group can be used as a condition in authorization policy, if desired.
Authorization policy	ACS 5.8 policy for assigning authorization attributes for access requests. Authorization policy selects a single rule and populates the response with the contents of the authorization profiles referenced as the result of the rule.
Exception policy	Special option for authorization policy, which allows you to define separately the set of conditions and authorization results for authorization policy exceptions and waivers. If defined, the exception policy is checked before the main (standard) authorization policy.
Default rule	Catchall rule in ACS 5.8 policies. You can edit this rule to specify a default result or authorization action, and it serves as the policy decision in cases where a given request fails to match the conditions specified in any user-created rule.

Simple Policies

You can configure all of your ACS policies as rule-based policies. However, in some cases, you can choose to configure a simple policy, which selects a single result to apply to all requests without conditions.

For example, you can define a rule-based authentication policy with a set of rules for different conditions; or, if you want to use the internal database for all authentications, you can define a simple policy.

Table 3-3 helps you determine whether each policy type can be configured as a simple policy.

- If you create and save a simple policy, and then change to a rule-based policy, the simple policy becomes the default rule of the rule-based policy.
- If you have saved a rule-based policy and then change to a simple policy, ACS automatically uses the default rule as the simple policy.

Related Topic

- Types of Policies, page 3-4

Rule-Based Policies

Rule-based policies have been introduced to overcome the challenges of identity-based policies. In earlier versions of ACS, although membership in a user group gives members access permissions, it also places certain restrictions on them.

When a user requests access, the user's credentials are authenticated using an identity store, and the user is associated with the appropriate user group. Because authorization is tied to user group, all members of a user group have the same access restrictions and permissions at all times.

With this type of policy (the simple policy), permissions are granted based on a user's association with a particular user group. This is useful if the user's identity is the only dominant condition. However, for users who need different permissions under different conditions, this policy does not work.

In ACS 5.x, you can create rules based on various conditions apart from identity. The user group no longer contains all of the information.

For example, if you want to grant an employee full access while working on campus, and restricted access while working remotely, you can do so using the rule-based policies in ACS 5.8.

You can base permissions on various conditions besides identity, and permissions are no longer associated with user groups. You can use session and environment attributes, such as access location, access type, health of the end station, date, time, and so on, to determine the type of access to be granted.

Authorization is now based on a set of rules:

If *conditions* then *apply the respective permissions*

With rule-based policies, conditions can consist of any combination of available session attributes, and permissions are defined in authorization profiles. You define these authorization profiles to include VLAN, downloadable ACLs, QoS settings, and RADIUS attributes.

Types of Policies

Table 3-3 describes the types of policies that you can configure in ACS.

The policies are listed in the order of their evaluation; any attributes that a policy retrieves can be used in any policy listed subsequently. The only exception is the Identity group mapping policy, which uses only attributes from identity stores.

Table 3-3 ACS Policy Types

Policy	Can Contain Exception Policy?	Simple[1] and Rule-Based?	Available Dictionaries for Conditions	Available Result Types	Attributes Retrieved
Service Selection Determines the access service to apply to an incoming request.	No	Yes	All except identity store related	Access Service	—
Identity Determines the identity source for authentication.	No	Yes	All except identity store related	Identity Source, Failure options	Identity Attributes; Identity Group for internal ID stores
Identity Group Mapping Defines mapping attributes and groups from external identity stores to ACS identity groups.	No	Yes	Only identity store dictionaries	Identity Group	Identity Group for external ID stores
Network Access Authorization Determines authorization and permissions for network access.	Yes	Rule-based only	All dictionaries	Authorization Profile, Security Group Access	—
Device Administration Authorization Determines authorization and permissions for device administration.	Yes	Rule-based only	All dictionaries	Shell Profile, Command Set	—

1. A simple policy specifies a single set of results that ACS applies to all requests; it is in effect a one-rule policy.

Access Services

Access services are fundamental constructs in ACS 5.x that allow you to configure access policies for users and devices that connect to the network and for network administrators who administer network devices.

In ACS 5.x, authentication and authorization requests are processed by access services. An access service consists of the following elements:

- Identity Policy—Specifies how the user should be authenticated and includes the allowed authentication protocols and the user repository to use for password validation.
- Group Mapping Policy—Specifies if the user's ACS identity group should be dynamically established based on user attributes or group membership in external identity stores. The user's identity group can be used as part of their authorization.
- Authorization Policy—Specifies the authorization rules for the user.

The access service is an independent set of policies used to process an access request.

The ACS administrator might choose to create multiple access services to allow clean separation and isolation for processing different kinds of access requests. ACS provides two default access services:

- Default Device Admin—Used for TACACS+ based access to device CLI
- Default Network Access—Used for RADIUS-based access to network connectivity

You can use the access services as is, modify them, or delete them as needed. You can also create additional access services.

The TACACS+ protocol separates authentication from authorization; ACS processes TACACS+ authentication and authorization requests separately. Table 3-4 describes additional differences between RADIUS and TACACS+ access services.

Table 3-4 Differences Between RADIUS and TACACS+ Access Services

Policy Type	TACACS+	RADIUS
Identity	Optional[1]	Required
Group Mapping	Optional	Optional
Authorization	Optional[1]	Required

1. For TACACS+, you must select either Identity or Authorization.

For TACACS+, all policy types are optional; however, you must choose at least one policy type in a service. If you do not define an identity policy for TACACS+, ACS returns authentication failed for an authentication request.

Similarly, if you do not define an authorization policy and if ACS receives a session or command authorization request, it fails. For both RADIUS and TACACS+ access services, you can modify the service to add policies after creation.

Note Access services do not contain the service selection policy. Service selection rules are defined independently.

You can maintain and manage multiple access services; for example, for different use cases, networks, regions, or administrative domains. You configure a service selection policy, which is a set of service selection rules to direct each new access request to the appropriate access service.

Table 3-5 describes an example of a set of access services.

Table 3-5 Access Service List

Access Service A for Device Administration	Access Service B for Access to 802.1X Agentless Hosts	Access Service C for Access from 802.1X Wired and Wireless Devices
Identity Policy A	Identity Policy B	Identity Policy C
Shell/Command Authorization Policy A	Session Authorization Policy B	Session Authorization Policy C

Table 3-6 describes a service selection policy.

Table 3-6 Service Selection Policy

Rule Name	Condition	Result
DevAdmin	protocol = TACACS+	Access Service A
Agentless	Host Lookup = True	Access Service C
Default	—	Access Service B

If ACS 5.8 receives a TACACS+ access request, it applies Access Service A, which authenticates the request according to Identity Policy A. It then applies authorizations and permissions according to the shell/command authorization policy. This service handles all TACACS+ requests.

If ACS 5.8 receives a RADIUS request that it determines is a host lookup (for example, the RADIUS service-type attribute is equal to *call-check*), it applies Access Service C, which authenticates according to Identity Policy C. It then applies a session authorization profile according to Session Authorization Policy C. This service handles all host lookup requests (also known as MAC Auth Bypass requests).

Access Service B handles other RADIUS requests. This access service authenticates according to Identity Policy B and applies Session Authorization Policy B. This service handles all RADIUS requests except for host lookups, which are handled by the previous rule.

Access Service Templates

ACS contains predefined access services that you can use as a template when creating a new service. When you choose an access service template, ACS creates an access service that contains a set of policies, each with a customized set of conditions.

You can change the structure of the access service by adding or removing a policy from the service, and you can change the structure of a policy by modifying the set of policy conditions. See Configuring Access Services Templates, page 10-21, for a list of the access service templates and descriptions.

RADIUS and TACACS+ Proxy Services

ACS 5.8 can function as a RADIUS, RADIUS proxy or TACACS+ proxy server.

- As a RADIUS proxy server, ACS receives authentication and accounting requests from the NAS and forwards the requests to the external RADIUS server.
- As a TACACS+ proxy server, ACS receives authentication, authorization and accounting requests from the NAS and forwards the requests to the external TACACS+ server.

ACS accepts the results of the requests and returns them to the NAS. You must configure the external RADIUS and TACACS+ servers in ACS for ACS to forward requests to them. You can define the timeout period and the number of connection attempts.

The ACS proxy remote target is a list of remote RADIUS and TACACS+ servers that contain the following parameters:

- IP
- Authentication port
- Accounting port
- Shared secret
- Reply timeout
- Number of retries
- Connection port

- Network timeout

The following information is available in the proxy service:

- Remote RADIUS or TACACS+ servers list
- Accounting proxy local/remote/both
- Strip username prefix/suffix

When a RADIUS proxy server receives a request, it forwards it to the first remote RADIUS or TACACS+ server in the list. If the proxy server does not receive a response within the specified timeout interval and the specified number of retries, it forwards the request to the next RADIUS or TACACS+ server in the list.

When the first response arrives from any of the remote RADIUS or TACACS+ servers in the list, the proxy service processes it. If the response is valid, ACS sends the response back to the NAS.

Table 3-7 lists the differences in RADIUS proxy service between ACS 4.2 and 5.8 releases.

Table 3-7 Differences in RADIUS and TACACS+ Proxy Service Between ACS 4.2 and 5.8

Feature	ACS 5.8	ACS 4.2
Configurable timeout (RADIUS)	Yes	No
Configurable retry count (RADIUS)	Yes	No
Network timeout (TACACS+)	Yes	No
Authentication and accounting ports (RADIUS)	Yes	Yes
Connection port (TACACS+)	Yes	No
Proxy cycles detection	Yes (For RADIUS only)	No
Username stripping	Yes	Yes
Accounting proxy (local, remote, or both)	Yes	Yes
Account delay timeout support (RADIUS)	No	No

ACS can simultaneously act as a proxy server to multiple external RADIUS and TACACS+ servers. For ACS to act as a proxy server, you must configure a RADIUS or TACACS+ proxy service in ACS. See Configuring General Access Service Properties, page 10-13 for information on how to configure a RADIUS proxy service.

For more information on proxying RADIUS and TACACS+ requests, see RADIUS and TACACS+ Proxy Requests, page 4-27.

Related Topics

- Policy Terminology, page 3-2
- Types of Policies, page 3-4
- Flows for Configuring Services and Policies, page 3-19

Identity Policy

Two primary mechanisms define the mechanism and source used to authenticate requests:

- Password-based—Authentication is performed against databases after the user enters a username and password. Hosts can bypass this authentication by specifying a MAC address. However, for identity policy authentication, host lookup is also considered to be password-based.
- Certificate-based—A client presents a certificate for authentication of the session. In ACS 5.8, certificate-based authentication occurs when the PEAP-TLS or EAP-TLS protocol is selected.

In addition, databases can be used to retrieve attributes for the principal in the request.

The identity source is one result of the identity policy and can be one of the following types:

- Deny Access—Access to the user is denied and no authentication is performed.
- Identity Database—Single identity database. When a single identity database is selected as the result of the identity policy, either an external database (LDAP or AD) or an internal database (users or hosts) is selected as the result.

  The database selected is used to authenticate the user/host and to retrieve any defined attributes stored for the user/host in the database.
- Certificate Authentication Profile—Contains information about the structure and content of the certificate, and specifically maps certificate attribute to internal username. For certificate-based authentication, you must select a certificate authentication profile.

  For certificate based requests, the entity which identifies itself with a certificate holds the private key that correlates to the public key stored in the certificate. The certificate authentication profile extends the basic PKI processing by defining the following:

  - The certificate attribute used to define the username. You can select a subset of the certificate attributes to populate the *username* field for the context of the request. The username is then used to identify the user for the remainder of the request, including the identification used in the logs.
  - The LDAP or AD database to use to verify the revocation status of the certificate. When you select an LDAP or AD database, the certificate data is retrieved from the LDAP or AD database and compared against the data entered by the client in order to provide additional verification of the client certificate.
- Identity Sequence—Sequences of the identity databases. The sequence is used for authentication and, if specified, an additional sequence is used to retrieve only attributes. You can select multiple identity methods as the result of the identity policy. You define the identity methods in an identity sequence object, and the methods included within the sequence may be of any type.

  There are two components to an identity sequence: one for authentication, and one for attribute retrieval. The administrator can select to perform authentication based on a certificate or an identity database or both.

  - If you choose to perform authentication based on a certificate, ACS selects a single certificate authentication profile.
  - If you choose to perform authentication based on an identity database, you must define a list of databases to be accessed in sequence until authentication succeeds. When authentication succeeds, any defined attributes within the database are retrieved.

  In addition, you can define an optional list of databases from which additional attributes are retrieved. These additional databases can be accessed irrespective of whether password- or certificate-based authentication was used.

  When certificate-based authentication is used, the username field is populated from a certificate attribute and is used to retrieve attributes. All databases defined in the list are accessed and, in cases where a matching record for the user is found, the corresponding attributes, are retrieved.

Attributes can be retrieved for a user even if the user's password is marked that it needs to be changed or if the user account is disabled. Even when you disable a user's account, the user's attributes are still available as a source of attributes, but not for authentication.

Failure Options

If a failure occurs while processing the identity policy, the failure can be one of three main types:

- Authentication failed—ACS received an explicit response that the authentication failed. For example, the wrong username or password was entered, or the user was disabled.
- User/host not found—No such user/host was found in any of the authentication databases.
- Process failed—There was a failure while accessing the defined databases.

All failures returned from an identity database are placed into one of the types above. For each type of failure, you can configure the following options:

- Reject—ACS sends a reject reply.
- Drop—No reply is returned.
- Continue—ACS continues processing to the next defined policy in the service.

The *Authentication Status* system attribute retains the result of the identity policy processing. If you select to continue policy processing in the case of a failure, this attribute can be referred to as a condition in subsequent policy processing to distinguish cases in which identity policy processing did not succeed.

Because of restrictions on the underlying protocol being used, there are cases in which it is not possible to continue processing even if you select the Continue option. This is the case for PEAP, LEAP, and EAP-FAST; even if you select the Continue option, the request is rejected.

The following default values are used for the failure options when you create rules:

- Authentication failed—The default is *reject*.
- User/host not found—The default is *reject*.
- Process failure—The default is *drop*.

Group Mapping Policy

The identity group mapping policy is a standard policy. Conditions can be based on attributes or groups retrieved from the external attribute stores only, or from certificates, and the result is an identity group within the identity group hierarchy.

If the identity policy accesses the internal user or host identity store, then the identity group is set directly from the corresponding user or host record. This processing is an implicit part of the group mapping policy.

Therefore, as part of processing in the group mapping policy, the default rule is only applied if both of the following conditions are true:

- None of the rules in the group mapping table match.
- The identity group is not set from the internal user or host record.

The results of the group mapping policy are stored in the **IdentityGroup** attribute in the System Dictionary and you can include this attribute in policies by selecting the Identity Group condition.

Authorization Policy for Device Administration

Shell profiles determine access to the device CLI; command sets determine TACACS+ per command authorization. The authorization policy for a device administration access service can contain a single shell profile and multiple command sets.

Processing Rules with Multiple Command Sets

It is important to understand how ACS processes the command in the access request when the authorization policy includes rules with multiple command sets. When a rule result contains multiple command sets, and the rule conditions match the access request, ACS processes the command in the access request against each command set in the rule:

Step 1 If a command set contains a match for the command and its arguments, and the match has *Deny Always*, ACS designates the command set as *Commandset-DenyAlways*.

Step 2 If there is no *Deny Always* for a command match in a command set, ACS checks all the commands in the command set sequentially for the first match.

- If the first match has *Permit*, ACS designates the command set as *Commandset-Permit*.
- If the first match has *Deny*, ACS designates the command set as *Commandset-Deny*.

Step 3 After ACS has analyzed all the command sets, it authorizes the command:

a. If ACS designated any command set as *Commandset-DenyAlways*, ACS denies the command.

b. If there is no *Commandset-DenyAlways*, ACS permits the command if any command set is *Commandset-Permit*; otherwise, ACS denies the command.

Related Topics

- Policy Terminology, page 3-2
- Authorization Profiles for Network Access, page 3-16

Exception Authorization Policy Rules

A common real-world problem is that, in day-to-day operations, you often need to grant policy waivers or policy exceptions. A specific user might need special access for a short period of time; or, a user might require some additional user permissions to cover for someone else who is on vacation.

In ACS, you can define an exception policy for an authorization policy. The exception policy contains a separate set of rules for policy exception and waivers, which are typically ad hoc and temporary. The exception rules override the rules in the main rule table.

The exception rules can use a different set of conditions and results from those in the main policy. For example, the main policy might use Identity Group and Location as its conditions, while its related exception policy might use different conditions

By default, exception policies use a compound condition and a time and date condition. The time and date condition is particularly valuable if you want to make sure your exception rules have a definite starting and ending time.

An exception policy takes priority over the main policy. The exception policy does not require its own default rule; if there is no match in the exception policy, the main policy applies, which has its own default rule.

You can use an exception to address a temporary change to a standard policy. For example, if an administrator, *John*, in one group is on vacation, and an administrator, *Bob*, from another group is covering for him, you can create an exception rule that will give *Bob* the same access permissions as *John* for the vacation period.

Related Topics

- Policy Terminology, page 3-2
- Policy Conditions, page 3-15
- Policy Results, page 3-16
- Policies and Identity Attributes, page 3-17

Service Selection Policy

When ACS receives various access requests, it uses a service selection policy to process the request. ACS provides you two modes of service selection:

- Simple Service Selection, page 3-12
- Rules-Based Service Selection, page 3-12

Simple Service Selection

In the simple service selection mode, ACS processes all AAA requests with just one access service and does not actually select a service.

Rules-Based Service Selection

In the rules-based service selection mode, ACS decides which access service to use based on various configurable options. Some of them are:

- AAA Protocol—The protocol used for the request, TACACS+ or RADIUS.
- Request Attributes—RADIUS or TACACS+ attributes in the request.
- Date and Time—The date and time ACS receives the request.
- Network Device Group—The network device group that the AAA client belongs to.
- ACS Server—The ACS server that receives this request.
- AAA Client—The AAA client that sent the request.
- Network condition objects—The network conditions can be based on
  - End Station—End stations that initiate and terminate connections.
  - Device—The AAA client that processes the request.
  - Device Port—In addition to the device, this condition also checks for the port to which the end station is associated with.

For more information on policy conditions, see Managing Policy Conditions, page 9-1.

ACS comes preconfigured with two default access services: Default Device Admin and Default Network Access. The rules-based service selection mode is configured to use the AAA protocol as the selection criterion and hence when a TACACS+ request comes in, the Default Device Admin service is used and when a RADIUS request comes in, the Default Network Access service is used.

Access Services and Service Selection Scenarios

ACS allows an organization to manage its identity and access control requirements for multiple scenarios, such as wired, wireless, remote VPN, and device administration. The access services play a major role in supporting these different scenarios.

Access services allow the creation of distinct and separate network access policies to address the unique policy requirements of different network access scenarios. With distinct policies for different scenarios, you can better manage your organization's network.

For example, the default access services for device administration and network access reflect the typical distinction in policy that is required for network administrators accessing network devices and an organization's staff accessing the company’s network.

However, you can create multiple access services to distinguish the different administrative domains. For example, wireless access in the Asia Pacific regions can be administered by a different team than the one that manages wireless access for European users. This situation calls for the following access services:

- APAC-wireless—Access service for wireless users in the Asia Pacific region.
- Europe-wireless—Access service for wireless users in the European countries.

You can create additional access services to reduce complexity in policies within a single access service by creating the complex policy among multiple access services. For example, if a large organization wishes to deploy 802.1x network access, it can have the following access services:

- 802.1x—For machine, user password, and certificate-based authentication for permanent staff.
- Agentless Devices—For devices that do not have an EAP supplicant, such as phones and printers.
- Guest Access—For users accessing guest wireless networks.

In this example, instead of creating the network access policy for 802.1x, agentless devices, and guest access in one access service, the policy is divided into three access services.

First-Match Rule Tables

ACS 5.8 provides policy decisions by using first-match rule tables to evaluate a set of rules. Rule tables contain conditions and results. Conditions can be either simple or compound. Simple conditions consist of attribute operator value and are either True or False. Compound conditions contain more complex conditions combined with AND or OR operators. See Policy Conditions, page 3-15 for more information.

The administrator selects simple conditions to be included in a policy. The conditions are displayed as columns in a rule table where the column headings are the condition name, which is usually the name of the attribute.

The rules are displayed under the column headings, and each cell indicates the operator and value that are combined with the attribute to form the condition. If *ANY* Figure 3-1 on page 14 shows a column-based rule table with defined condition types.

Figure 3-1 Example Policy Rule Table

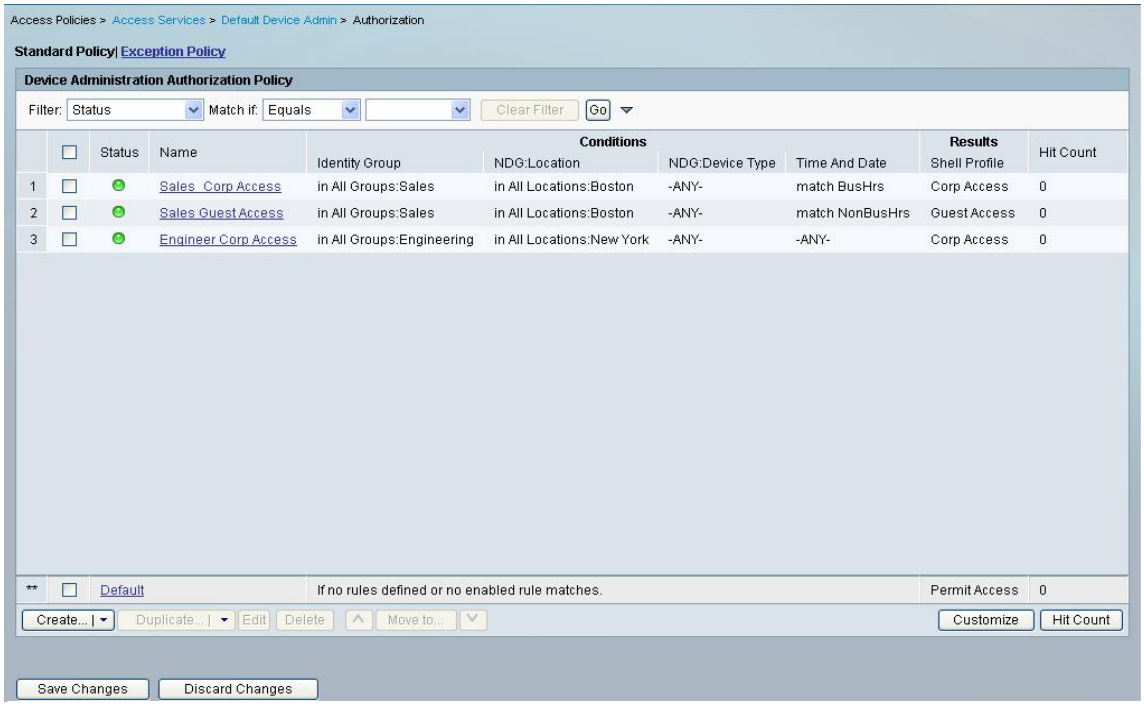


Table 3-8 Example Policy Rule

Column	Description
Status	You can define the status of a rule as enabled, disabled, or monitored: • Enabled—ACS evaluates an enabled rule, and when the rule conditions match the access request, ACS applies the rule result. • Disabled—The rule appears in the rule table, but ACS skips this rule and does not evaluate it. • Monitor Only—ACS evaluates a monitored rule. If the rule conditions match the access request, ACS creates a log record with information relating to the match. ACS does not apply the result, and the processing continues to the following rules. Use this status during a running-in period for a rule to see whether it is needed.
Name	Descriptive name. You can specify any name that describes the rule's purpose. By default, ACS generates rule name strings *rule-number*.
Conditions	
Identity Group	In this example, this is matching against one of the internal identity groups.
NDG: Location	Location network device group. The two predefined NDGs are Location and Device Type.
Results	

Table 3-8 Example Policy Rule

Shell Profile	Used for device administration-type policies and contains permissions for TACACS+ shell access request, such as Cisco IOS privilege level.
Hit Counts	Displays the number of times a rule matched an incoming request since the last reset of the policy's hit counters. ACS counts hits for any monitored or enabled rule whose conditions all matched an incoming request. Hit counts for: • Enabled rules reflect the matches that occur when ACS processes requests. • Monitored rules reflect the counts that would result for these rules if they were enabled when ACS processed the requests. The primary server in an ACS deployment displays the hit counts, which represent the total matches for each rule across all servers in the deployment. On a secondary server, all hit counts in policy tables appear as zeroes.

The default rule specifies the policy result that ACS uses when no other rules exist, or when the attribute values in the access request do not match any rules.

ACS evaluates a set of rules in the first-match rule table by comparing the values of the attributes associated with the current access request with a set of conditions expressed in a rule.

- If the attribute values do not match the conditions, ACS proceeds to the next rule in the rule table.
- If the attribute values match the conditions, ACS applies the result that is specified for that rule, and ignores all remaining rules.
- If the attribute values do not match any of the conditions, ACS applies the result that is specified for the policy default rule.

Related Topics

- Policy Terminology, page 3-2
- Policy Conditions, page 3-15
- Policy Results, page 3-16
- Exception Authorization Policy Rules, page 3-11

Policy Conditions

You can define simple conditions in rule tables based on attributes in:

- Customizable conditions—You can create custom conditions based on protocol dictionaries and identity dictionaries that ACS knows about. You define custom conditions in a policy rule page; you cannot define them as separate condition objects.
- Standard conditions—You can use standard conditions, which are based on attributes that are always available, such as device IP address, protocol, and username-related fields.

Related Topics

- Policy Terminology, page 3-2
- Policy Results, page 3-16
- Exception Authorization Policy Rules, page 3-11
- Policies and Identity Attributes, page 3-17

Policy Results

Policy rules include result information depending on the type of policy. You define policy results as independent shared objects; they are not related to user or user group definitions.

For example, the policy elements that define authorization and permission results for authorization policies include:

- Identity source and failure options as results for identity policies. See Authorization Profiles for Network Access, page 3-16.
- Identity groups for group mapping. See Group Mapping Policy, page 3-10.
- Authorization Profiles for Network Access, page 3-16.
- Authorization Policy for Device Administration, page 3-11.
- Security groups and security group access control lists (ACLs) for Cisco Security Group Access. See ACS and Cisco Security Group Access, page 4-22.

For additional policy results, see Managing Authorizations and Permissions, page 9-17.

Related Topics

- Policy Terminology, page 3-2
- Policy Conditions, page 3-15
- Exception Authorization Policy Rules, page 3-11
- Policies and Identity Attributes, page 3-17

Authorization Profiles for Network Access

Authorization profiles define the set of RADIUS attributes that ACS returns to a user after successful authorization. The access authorization information includes authorization privileges and permissions, and other information such as downloadable ACLs.

You can define multiple authorization profiles as a network access policy result. In this way, you maintain a smaller number of authorization profiles, because you can use the authorization profiles in combination as rule results, rather than maintaining all the combinations themselves in individual profiles.

Processing Rules with Multiple Authorization Profiles

A session authorization policy can contain rules with multiple authorization profiles. The authorization profile contains general information (name and description) and RADIUS attributes only. When you use multiple authorization profiles, ACS merges these profiles into a single set of attributes. If a specific attribute appears:

- In only one of the resulting authorization profiles, it is included in the authorization result.
- Multiple times in the result profiles, ACS determines the attribute value for the authorization result based on the attribute value in the profile that appears first in the result set.

  For example, if a VLAN appears in the first profile, that takes precedence over a VLAN that appears in a 2nd or 3rd profile in the list.

Note If you are using multiple authorization profiles, make sure you order them in priority order.

The RADIUS attribute definitions in the protocol dictionary specify whether the attribute can appear only once in the response, or multiple times. In either case, ACS takes the values for any attribute from only one profile, irrespective of the number of times the values appear in the response. The only exception is the Cisco attribute value (AV) pair, which ACS takes from all profiles included in the result.

Related Topics

- Policy Terminology, page 3-2
- Authorization Policy for Device Administration, page 3-11

Policies and Identity Attributes

The identity stores contain identity attributes that you can use as part of policy conditions and in authorization results. When you create a policy, you can reference the identity attributes and user attributes.

This gives you more flexibility in mapping groups directly to permissions in authorization rules. When ACS processes a request for a user or host, the identity attributes are retrieved and can then be used in authorization policy conditions.

For example, if you are using the ACS internal users identity store, you can reference the identity group of the internal user or you can reference attributes of the internal user. (Note that ACS allows you to create additional custom attributes for the internal identity store records.)

If you are using an external Active Directory (AD), you can reference AD groups directly in authorization rules, and you can also reference AD user attributes directly in authorization rules. User attributes might include a user's department or manager attribute.

Related Topics

- Managing Users and Identity Stores, page 8-1
- Policy Terminology, page 3-2
- Types of Policies, page 3-4

Policies and Network Device Groups

You can reference Network device groups (NDGs) as policy conditions. When the ACS receives a request for a device, the NDGs associated with that device are retrieved and compared against those in the policy table. With this method, you can group multiple devices and assign them the same policies. For example, you can group all devices in a specific location together and assign to them the same policy.

When ACS receives a request from a network device to access the network, it searches the network device repository to find an entry with a matching IP address. When a request arrives from a device that ACS identified using the IP address, ACS retrieves all NDGs associated with the device.

Related Topics

- Managing Users and Identity Stores, page 8-1
- Policy Terminology, page 3-2

- Types of Policies, page 3-4

Example of a Rule-Based Policy

The following example illustrates how you can use policy elements to create policy rules.

A company divides its network into two regions, East and West, with network operations engineers at each site. They want to create an access policy that allows engineers:

- Full access to the network devices in their region.
- Read-only access to devices outside their region.

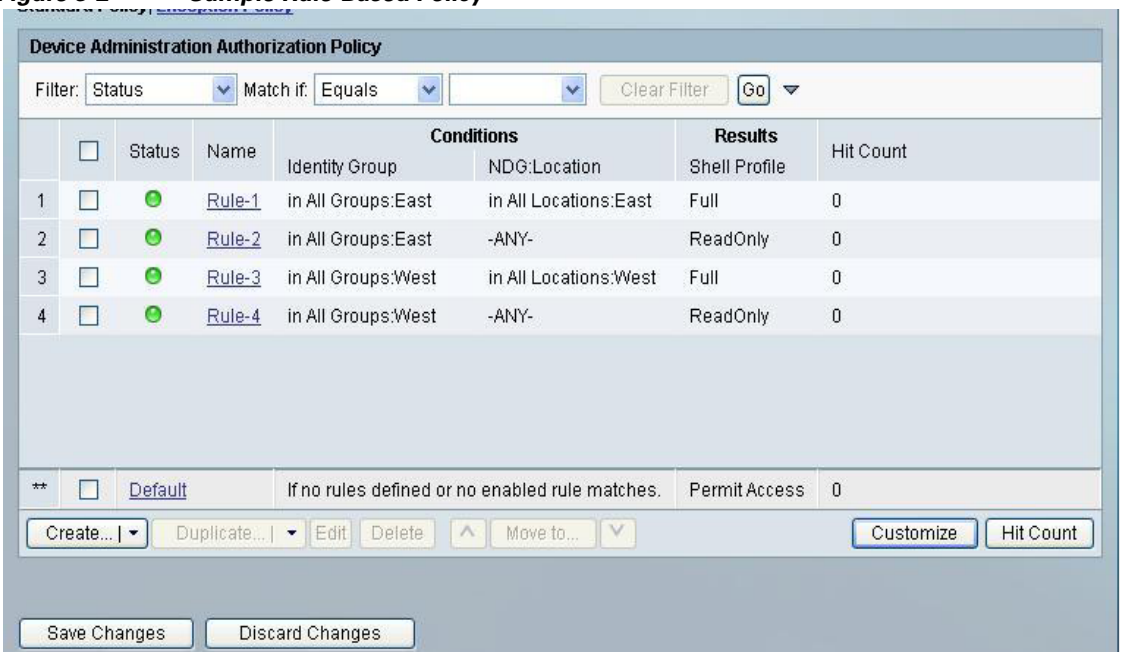
You can use the ACS 5.8 policy model to:

- Define East and West network device groups, and map network devices to the appropriate group.
- Define East and West identity groups, and map users (network engineers) to the appropriate group.
- Define Full Access and Read Only authorization profiles.
- Define Rules that allow each identity group full access or read-only access, depending on the network device group location.

Previously, you had to create two user groups, one for each location of engineers, each with separate definitions for permissions, and so on. This definition would not provide the same amount of flexibility and granularity as in the rule-based model.

Figure 3-2 illustrates what this policy rule table could look like.

Figure 3-2 Sample Rule-Based Policy



Each row in the policy table represents a single rule.

Each rule, except for the last Default rule, contains two conditions, ID Group and Location, and a result, Authorization Profile. ID Group is an identity-based classification and Location is a nonidentity condition. The authorization profiles contain permissions for a session.

The ID Group, Location, and Authorization Profile are the policy elements.

Related Topics

- Policy Terminology, page 3-2
- Types of Policies, page 3-4
- Access Services, page 3-5
- Flows for Configuring Services and Policies, page 3-19

Flows for Configuring Services and Policies

Table 3-9 describes the recommended basic flow for configuring services and policies; this flow does not include user-defined conditions and attribute configurations. With this flow, you can use NDGs, identity groups, and compound conditions in rules.

Prerequisites

Before you configure services and policies, it is assumed you have done the following:

- Added network resources to ACS and create network device groups. See Creating, Duplicating, and Editing Network Device Groups, page 7-2 and Network Devices and AAA Clients, page 7-5.
- Added users to the internal ACS identity store or add external identity stores. See Creating Internal Users, page 8-13, Managing Identity Attributes, page 8-7, or Creating External LDAP Identity Stores, page 8-34.

Table 3-9 Steps to Configure Services and Policies

Step	Action	Drawer in Web Interface
Step 1	Define policy results: • Authorizations and permissions for device administration—Shell profiles or command sets. • Authorizations and permissions for network access—Authorization profile. See: • Creating, Duplicating, and Editing a Shell Profile for Device Administration, page 9-23 • Creating, Duplicating, and Editing Command Sets for Device Administration, page 9-28 • Creating, Duplicating, and Editing Authorization Profiles for Network Access, page 9-18	Policy Elements
Step 2	(Optional) Define custom conditions to policy rules. You can complete this step before defining policy rules in Step 6, or you can define custom conditions while in the process of creating a rule. SeeCreating, Duplicating, and Editing a Custom Session Condition, page 9-5.	—
Step 3	Create Access Services—Define only the structure and allowed protocols; you do not need to define the policies yet. See Creating, Duplicating, and Editing Access Services, page 10-12.	Access Policies

Table 3-9 Steps to Configure Services and Policies (continued)

Step	Action	Drawer in Web Interface
Step 5	Add rules to Service Selection Policy to determine which access service to use for requests. See: • Customizing a Policy, page 10-4 • Creating, Duplicating, and Editing Service Selection Rules, page 10-8	Access Policies
Step 6	Define identity policy. Select the identity store or sequence you want to use to authenticate requests and obtain identity attributes. See Managing Users and Identity Stores, page 8-1.	Users and Identity Stores
Step 7	Create authorization rules: • Device administration—Shell/command authorization policy. • Network access—Session authorization policy. See: • Customizing a Policy, page 10-4 • Configuring Access Service Policies, page 10-23	Access Policies

Related Topics

- Policy Terminology, page 3-2
- Policy Conditions, page 3-15
- Policy Results, page 3-16
- Policies and Identity Attributes, page 3-17