Using Identity Caching

Cisco Prime Access Registrar (Prime Access Registrar) software includes the identity caching feature. Prime Access Registrar runs as application layer software and can be used standalone or in conjunction with other workstations running Prime Access Registrar.

Note The identity caching feature is available on Prime Access Registrar releases 3.5.2 and above.

Identity caching provides subscriber identity resolution services with fast access to associated subscriber identity data for service providers, enabling them to offer new services to their customers based on identity caching and context information management.

This chapter contains the following sections:

Overview

Identity caching enables Cisco equipment to gain context information about the operator's subscribers to support network functions or to enhance subscriber's experience on the operator's network. Figure 7-1 on page 7-2, Prime Access Registrar System Overview, shows the network environment where Prime Access Registrar identity caching might be used.

For example, Client Services Gateway (CSG) uses IP mapping information provided by identity caching to support post-paid content billing. Identity caching acquires subscriber information from other devices and information sources in the operator's network. The type of information provided is limited by the available information sources and is configurable by the operator, but might include information such as IP address, MSISDN, and IMSI. Identity caching does not duplicate the operator's persistent data stores. Identity caching provides a protocol-based interface through which Cisco network elements (Prime Access Registrar identity caching clients) can access subscriber information.

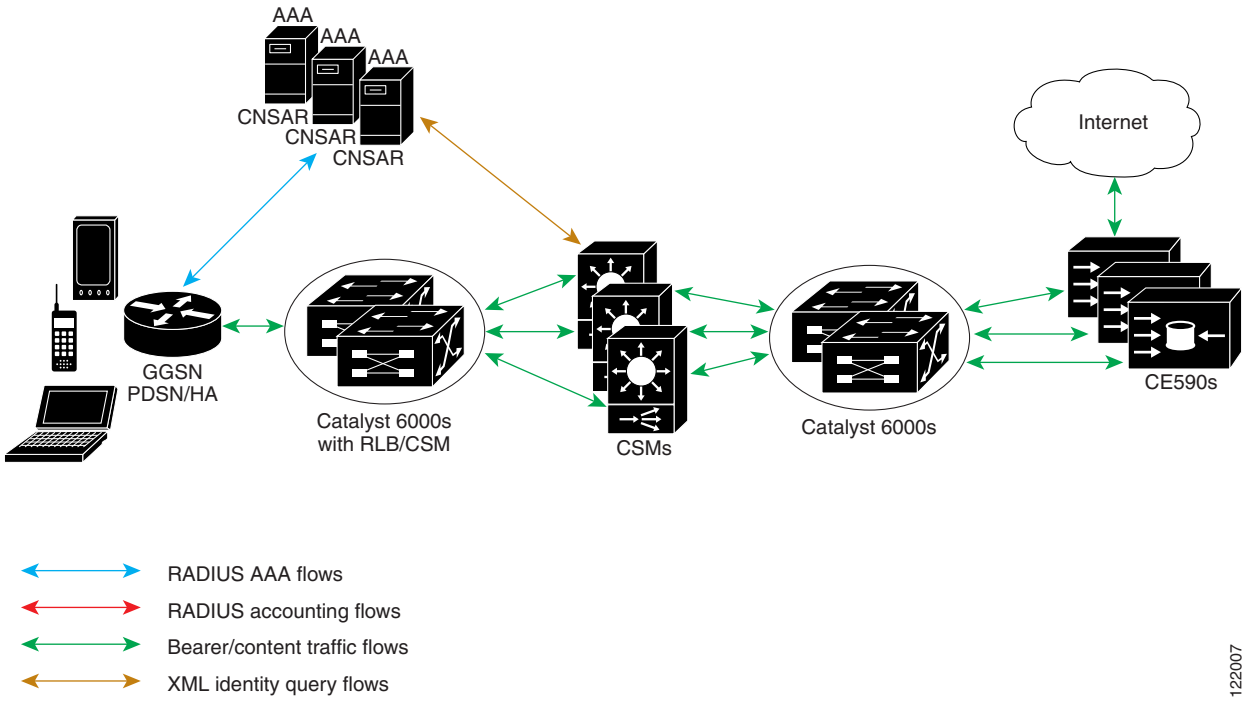
The Prime Access Registrar servers receive RADIUS flows from the Gateway GPRS support Node (GGSN) which acts as a type of network access station (NAS). These flows perform full AAA (authentication, authorization, and accounting). You can configure the Prime Access Registrar servers to redirect the accounting information (only) to an identity caching server to be cached. The GGSN can also be configured to direct only the RADIUS accounting information directly to the Prime Access Registrar server.

Prime Access Registrar also receives XML identity query flows from the CSM which acts as a NAS. In the event that a CSM should fail or lose its information, the information can be refreshed from the information cached in the Prime Access Registrar server.

Prime Access Registrar acquires subscriber information such as the IP address, the mobile Subscriber ISDN number (MSISDN), and the International Mobile System Identifier (IMSI) from AAA requests the Prime Access Registrar server receives, typically from the GGSN. The types of information provided is limited by the available information sources and is configurable by the operator.

Prime Access Registrar includes an XML Query Identity enhancement. Prime Access Registrar previously supported User-Name lookup based on the Framed IP address of an existing session. The XML Query Identity enhancement enables Framed IP address lookup based on the User-Name in an existing session.

Figure 7-1 Prime Access Registrar System Overview



Identity Caching Features

Prime Access Registrar identity caching provides the following features:

- Supports GGSN subscriber data attributes from RADIUS authentication sequences
- Provides basic identity mapping services from IP address or username/APN to Mobile DN for one network presence at a time.
- Provide session management support for Content Switch Module (CSM)

  Prime Access Registrar enables the CSM to keep the data session and content correlated to the same subscriber reconnecting, perhaps after an attach/detach sequence for a GPRS subscriber connecting again. This is done through the MSISDN identity to IP mapping in the identity caching function.

- Enhance redundancy with stateful fail-over support for applications by finding the right connection between subscriber identity and IP address using the Identity Cache function.
- Uses an XML interface to make it easier for any network function or application to use without having to have detailed internal knowledge about the execution environment or programming methods.
- Provides user identity resolution with fast access to associated subscriber data
- Establishes an identity and Access Management solution that can be used in and across multiple network domains
- Provides a way to use identity resolution to manage the growth of 2.5G mobile data access services (GSM/GPRS) and to provide always-on mobile data access including the following:
  - Ties various IP addresses to a unique subscriber identifier
  - Dynamically assigning and reusing IP addresses and controlling services with consistent identification
  - Correlates previous content activity when a mobile subscriber reconnects
  - Correlates IP addresses, mobile numbers, username, and identifiers to support customer billing
  - Correlates and identifies subscribers using both 2.5G and WLAN services and provides a way to control and manage operator network services
  - Provides subscriber privacy control
  - Provides a way to cache content with various customers and their networks

Configuring Cisco Prime Access Registrar for Identity Caching

Use the command line interface **aregcmd** to configure Prime Access Registrar to perform identity caching.

Configuring the Identity Caching

To configure identity caching:

Step 1 Launch **aregcmd**.

Step 2 Define a client object for each client that will send either RADIUS or XML packets to the Prime Access Registrar server performing identity caching.

There should be one client object for each GGSN, one for each CSM and one for each packet simulator (if used in a test environment).

For example, if a packet simulator will be used on the same server where you perform identity caching, add a client object as in the following:

cd /Radius/Clients

add xml-client

cd xml-client

```
[ //localhost/Radius/Clients/xml-client ]
    Name = xml-client
    Description =
    IPAddress =
    SharedSecret =
    Type = NAS
    Vendor =
    IncomingScript~ =
    OutgoingScript~ =
    EnablePOD = FALSE
```

This client object is very similar to the localhost object defined in the example configuration. The **SharedSecret** property will be ignored if the client is an XML client, but still must be set to a non-null value. The **Type** property is also ignored for XML clients.

Step 3 Define a port object for each RADIUS port and each XML port to be used. Two RADIUS ports, the second immediately following the first in numeric value, must be defined even if only one is needed. A typical identity caching installation requires the following port configuration:

cd /Radius/Advanced/Ports

add 1812

add 1813

add 8080

Note Although ports 1812 and 1813 are the default ports for Prime Access Registrar, you must add them to **/Radius/Advanced/Ports** to also add port 8080.

Step 4 Change directory to the 1812 port and set its type to Radius-Access.

cd /Radius/Advanced/Ports/1812

set Type Radius-Access

Step 5 Change directory to the 1813 port and set its type to Radius-Accounting.

cd /Radius/Advanced/Ports/1813

set Type Radius-Accounting

Step 6 Change directory to the 8080 port and set its type to XML.

-cd /Radius/Advanced/Ports/8080

set Type XML

Step 7 Define and configure an accounting service of type file and set it as the DefaultAccountingService.

An accounting service is required for Prime Access Registrar to cache identity information, even if no accounting service is needed otherwise. If you added the example configuration during installation, a local-file accounting service is already configured.

If you did not add the example configuration during software installation, see the Setting Up Accounting section in Chapter 3, "RADIUS Accounting."

Step 8 Define and configure a ResourceManager for identity caching.

cd /Radius/ResourceManagers

add cache

Step 9 Set the ResourceManager to type session-cache for identity caching.

cd cache

set type session-cache

The following shows the default properties of a session-cache ResourceManager:

```
[ //localhost/Radius/ResourceManagers/cache ]
    Name = cache
    Description =
    Type = session-cache
    OverwriteAttributes = FALSE
    QueryKey =
    PendingRemovalDelay = 10
    AttributesToBeCached/
    QueryMappings/
```

Step 10 Set the QueryKey to a RADIUS attribute you want to key on.

For example, use the following command to set the QueryKey to User-Name:

set QueryKey User-Name

The QueryKey must match the string on the right-hand side of one of the pairs you list in QueryMappings. It is not necessary for the QueryKey to be configured under **AttributesToBeCached** because the QueryKey will always be cached by default.

Note The QueryKey property must always be a RADIUS attribute. The Prime Access Registrar server forces a NULL IP address (0.0.0.0) if it detects an incorrectly configured QueryKey.

Step 11 Change directory to **AttributesToBeCached** and use the **set** command to provide a list of RADIUS attributes you want to store in cache.

cd AttributesToBeCached

set 1 Calling-Station-ID

Set 2 User-Name

Set 3 Framed-IP-Address

The attributes a session-cache resource manager caches can be queried through both RADIUS Query and XML Query packets. When you cache attributes Framed-IP-Address or User-Name, or when you use XML-Address-format-IPv4 or XML-UserId-id_type-subscriber_id as the QueryKey, you must map the XML attributes to RADIUS attributes in the **QueryMappings** subdirectory.

Step 12 Change directory to **QueryMappings** and use the **set** command to list the attribute pairs, mapping the XML attributes on the left-hand side to the RADIUS attribute on the right-hand side.

set XML-Address-format-IPv4 Framed-IP-Address

set XML-UserId-id_type-subscriber_id User-Name

Step 13 Change directory to **/Radius/SessionManagers** and add a SessionManager for identity caching.

cd /Radius/SessionManagers

add IDcache

Step 14 Change directory to the new identity caching SessionManager, then change directory to the **ResourceManager** list.

cd IDcache/ResourceManagers

Step 15 Use the **set** command to associate the identity caching ResourceManager with this SessionManager.

set 1 cache

Step 16 Change directory to **/Radius** and set the DefaultSessionManager to the identity caching SessionManager.

cd /Radius

set DefaultSessionManager IDcache

Step 17 Run the **save**, **reload**, and **exit** commands:

save

reload

exit

Starting Identity Caching

To start identity caching, you must send an Accounting-Request to the specified accounting port (The default accounting port is 1813.) A minimal Accounting-Request will contain the following attributes:

- NAS-Identifier or NAS-IP-Address
- NAS-Port
- Framed-IP-Address
- User-Name
- Acct-Status-Type
- Acct-Session-Id

Starting Identity Caching

To start identity caching:

Step 1 Launch **radclient**:

cd /opt/CSCOar/bin

radclient -C localhost -N admin -P aicuser

Step 2 Enter the following **radclient** commands:

set p [acct_request Start joeuser@cisco.com]

$p set attrib [attrib Framed-IP-Address 123.123.123.123]

$p send

This assumes that you are running **radclient** on the same server and using 1813 as the accounting port.

Step 3 Send XML requests to the specified XML port (Cisco suggests port 8080 as shown above). A typical XML packet will look like the following:

```
<?xml version="1.0"?>
<Request>
    <UserIdRequest>
        <UserId id_type="subscriber_id">bob</UserId>
    </UserIdRequest>
</Request>
```

To do this using **xmlclient**, put the XML text into a file, then enter the following command:

cd /opt/CSCOar/bin

./xmlclient -srd <file>

Note This assumes that xmlclient is running on the same server as identity caching and that 8080 is the XML port. Use the command **xmlclient -H** for information about how to use a different port or how to run **xmlclient** from a different server.

Note For a successful query, xml response will have the IPAddress associated with the requested user-name and for failure query it returns 0.0.0.0 as the IPAddress.

XML Interface

The XML interface is used for subscriber context information queries and responses to those queries. The XML interface is on a UDP port (8080) and is configurable. Identity caching supports the XML data-type definition (DTD) supported by the CSG.

The mapping from queries to replies can be one to many. For example, a UDP datagram might contain several queries but each reply will be returned in a separate datagram. No single query or reply can exceed the configured MTU of a datagram. Any that does results in an error.

If a query result is negative, the reply will consist of a null subscriber ID. All other error conditions cause Prime Access Registrar to drop the request. Errors are logged locally using the Prime Access Registrar logging mechanism.