Using WiMAX in Cisco Prime Access Registrar

Cisco Prime Access Registrar (Prime Access Registrar) supports Worldwide Interoperability for Microwave Access (WiMAX) technology. This feature support in Prime Access Registrar complies with the WiMAX forum NWG_R1_V1.3.1-Stage-3 specifications.

This chapter contains the following sections:

WiMAX - An Overview

WiMAX is a standards-based wireless technology that offers high throughput broadband connections over long distances. WiMAX can be used for a number of applications, including “last mile” broadband connections, fixed and mobile cellular service, hotspots and cellular backhaul, and high-speed enterprise connectivity for business. WiMAX is based on the IEEE 802.16d standard for fixed wireless, and the 802.16e standard for mobile wireless. This standard is appealing to customers because it allows mass production of chipsets that reduce CPE costs, ensures multi-vendor interoperability, and reduces investment risk for operators.

The architectural framework of a WiMAX network consists of the Access Service Network (ASN), the Connectivity Service Network (CSN), and a AAA Server. An Access Service Network is a set of network functions that provide radio access to a WiMAX subscriber. The ASN typically provides functions such as network discovery and selection, connectivity service between the MSS and CSN, Radio Resource Management, Multicast and Broadcast Control, Intra-ASN mobility, Paging, and Location Management. The WiMAX architecture consists of both mobile and fixed subscribers, as well as the ASN and CSN.

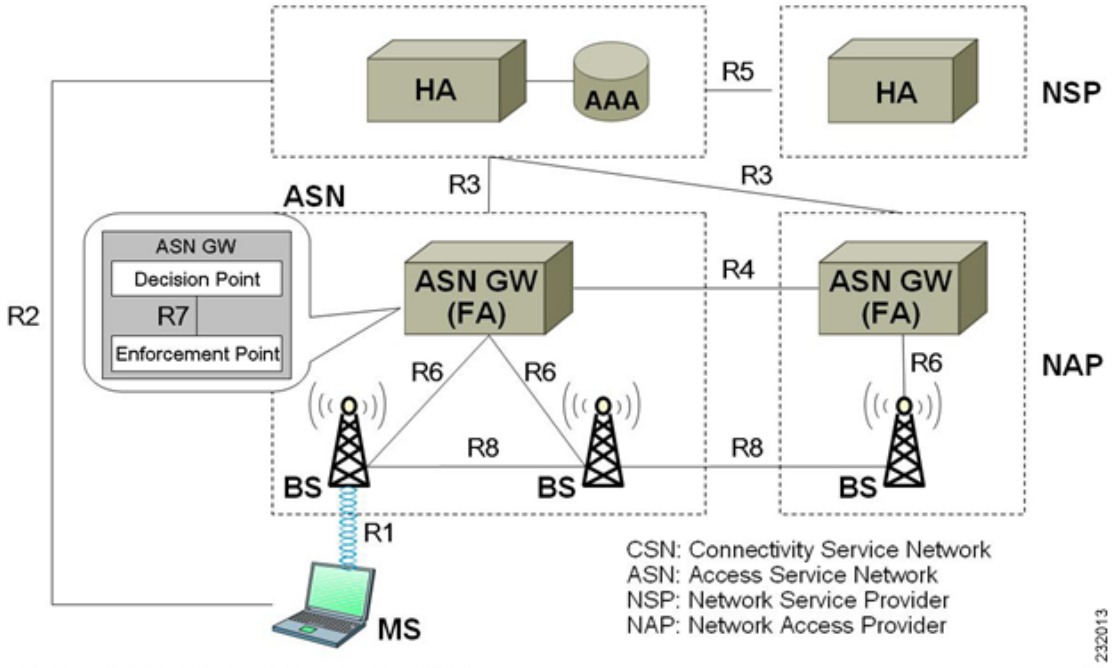
A CSN is defined as a set of network functions that provide IP connectivity services to the WiMAX subscribers. CSN might comprise network elements such as Routers, Home Agent, AAA proxy/servers, user databases, Policy Servers, Content Service Gateways, Service Selection Gateways, and interworking gateway devices.

The Access Service Network is connected to a home network HCSN (Home Connectivity Service Network) via at least one visited network (Visited Connectivity Service Network VCSN) or intermediate network.

The Visited CSN plays the role of a AAA proxy. During all AAA interaction the VCSN AAA server acts as a RADIUS proxy transporting RADIUS packets between the ASN and the HCSN.

Figure 3-1 describes the network reference model of a typical WiMAX scenario.

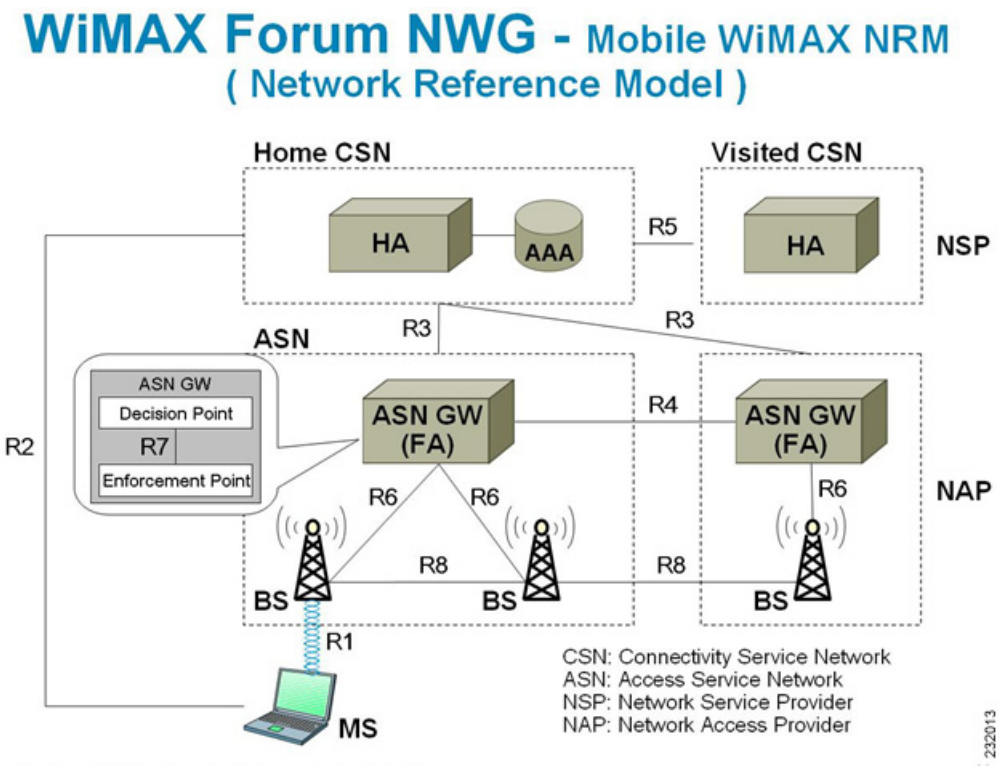
Figure 3-1 WiMAX Network Reference Model



WiMAX in Cisco Prime Access Registrar

Prime Access Registrar uses the Extensible Authentication Protocol (EAP) to enable the WiMAX feature. It also caches the IP attributes and Mobility Keys that are generated during network access authentication. To enable caching of the WiMAX attributes, you must configure the respective resource managers. See Configuring the Resource Manager for WiMAX, page 3-8, for information on configuring resource manager. Figure 3-2 shows the WiMAX workflow in Prime Access Registrar.

Figure 3-2 WiMAX Workflow



The WiMAX workflow in Prime Access Registrar includes:

- Direct interaction between the ASN GW and Prime Access Registrar
- Interaction between the ASN GW and Prime Access Registrar through the HA

This section contains the following topics:

Direct Interaction Between the ASN GW and Cisco Prime Access Registrar

When the mobile node (MN) sends a RADIUS request to the ASN GW, it forwards this request to the CSN. If it is VCSN, the VAAA proxies the request with Visited HA address in the Access Request to HAAA. The HAAA initiates an authentication using the EAP service, **for example, eap-ttls**. The initial Access-Request containing the WiMAX capability and NAS-Port-Type (Type:61) attributes indicate that the specified flow is for a WiMAX request from ASN GW. Prime Access Registrar redirects this request to the WiMAX service that you configure. The WiMAX service redirects the request to the EAP-based Wimax-Authentication-Service for authentication. Upon successful authentication, the WiMAX service redirects the request to Wimax-Session-Manager to allocate the home agent. Subsequently, Prime Access Registrar generates the appropriate keys based on the Extended Master Session Key (EMSK) and records the generated keys in the session cache resource manager as configured, before sending Access-Accept to the ASN GW.

If there is no VCSN, then the HAAA will send the Access-Accept to ASNGW. Otherwise, the HAAA sends the Access-Accept to VAAA. The VAAA then generates the visited HA-RK Key with SPI and Lifetime and sends the access-accept to ASNGW.

The authentication methods followed by Prime Access Registrar are:

- User-only
- Device-only
- Single-EAP Device or User authentication

Note Prime Access Registrar 4.2 does not support Double-EAP authentication.

Prime Access Registrar uses the following values to identify the service-type:

- Framed—for initial authentication
- Authenticate-Only—for reauthentication
- Authorize-Only—for prepaid request

Note Prepaid attributes can also be sent in the initial authentication.

The attributes contained in this flow are listed in Table 3-1. For detailed information on the attributes refer to the WiMAX forum NWG_R1_V1.3.1-Stage-3 specifications document.

Table 3-1 Attributes: ASN GW-Prime Access Registrar Flow

Attribute	Description
User-Name	Must be present. This attributes gets the NAI from the EAP-Response/Identity.
Service-Type	Must be present and the value is Framed, Authenticate-Only or Authorize-Only.
WiMAX Capability	This attribute is chosen by the ASN GW. The request to the Prime Access Registrar is provided through the WiMAX-Capability attribute. The server might respond with the chosen WiMAX Capability.
NAS-Port-Type	The request must contain this attribute with the value 27. This indicates Wireless IEEE 802.16 port when coming from a WiMAX ASN.
Calling-Station-ID	The request must contain this attribute with the value set to the MAC address of the device in binary format.
Device-Authentication-Indicator	The request might contain this attribute to indicate whether the device authentication was performed or not and the result of the action.
CUI	The NAS might intimate the support for CUI by sending the CUI attribute with the value ‘null’.
GMT-Time-Zone-Offset	The request must contain the offsets in seconds from the GMT at the NAS.

Table 3-1 Attributes: ASN GW-Prime Access Registrar Flow (continued)

Attribute	Description
Framed-IP-Address	This is the CMIPv4 Home address to be assigned to the MN. If this attribute is not present then the Home address is derived by the ASN from MIP procedures or through DHCP.
WiMax-Session-ID	This attribute shall not be present in the initial authentication. The value is a unique identifier in the home realm for this session as set by the HAAA(Prime Access Registrar) in the Access-Accept, when the authentication is successful and it will be included in all subsequent requests from the NAS, such as online accounting.
MSK	The MSK shall be provided by the AAA Server as a result of successful EAP-authentication. MSK can be transmitted using either the MS-MPPE-Keys or the MSK attribute.
Packet-Flow-Descriptor	The pre-provisioned service flow which might be present in the Access-Accept packet.
QoS-Descriptor	The pre-provisioned service flow which might be present in the Access-Accept packet, if configured in Prime Access Registrar.
BS-ID	Might be present in the Access-Request packet which will identify NAP-ID base station. If both NAP-ID and BS-ID are present, the NAP-ID will be ignored.
Acct-Interim-Interval	Sent in the Access-Accept packet. It indicates the accounting update intervals.

Prime Access Registrar generates a few more attributes upon successful authentication. These attributes are described in Table 3-2.

Table 3-2 Additional Attributes: ASN-GW Prime Access Registrar Flow

Attribute	Description
hHA-IP-MIP4	The IP address of the home HA allocated for the incoming request.
vHA-IP-MIP4	The IP address of the visited HA. To be used by the PMIP4 client.
MN-hHA-MIP4-KEY	The MN-hHA key is used for MIP4 procedures.
MN-hHA-MIP4-SPI	The SPI associated with the MN-hHA-MIP4-KEY.
MN-vHA-MIP4-KEY	The MN-vHA key is used for MIP4 procedures.

Table 3-2 Additional Attributes: ASN-GW Prime Access Registrar Flow (continued)

Attribute	Description
MN-vHA-MIP4-SPI	The SPI associated with the MN-vHA-MIP4-KEY.
FA-RK-KEY	The FA-RK key will be used at ASN GW to derive MN-FA for MIP4 procedures.

Note A policy engine can parse the NAI decoration and conclude the type of authentication method for the incoming access-request for passing on to WiMAX service.

Interaction Between ASN GW and Cisco Prime Access Registrar Through HA

After Prime Access Registrar returns the Access-Accept to the ASN GW, the mobile node, which initially sent the request, sends a registration request to the ASN GW. The ASN GW receives this request and sends an Access-Request to the HA. A Query-Request will be sent to the Prime Access Registrar by HA to receive the security context for authenticating the FA.

Prime Access Registrar identifies the request as HA query request, if:

- the WiMAX mobility attribute is present
- the NAS-Port-Type attribute is absent

Prime Access Registrar checks for a valid session in the session cache based on NAI and sends an Access-Accept to the HA.

Table 3-3 HAAA Cached Attributes

Attribute	Description
Pseudo Identity	As received from the MS in the NAI in the EAP-Response/Identity. The HAAA is required to correlate this to the true identity of the user.
NAS-ID/NAS-IP address	One or both of these parameters are cached by the HAAA. This is required to locate the serving NAS.
Framed-IP Address	The IP address allocated to the user session. This information is useful in identifying the session during AAA dynamic procedures.
MIP-RK, hHA-RK, FA-RK, MN-hHA	Mobility keys generated during network access authentication. These keys are cached and used by the network for mobility authentication.
hHA-IP address	The IP address of the home HA assigned to the MS.

Table 3-4 VAAA Cached Attributes

Attribute	Description
vHA-RK, vHA-RK-SPI, vHA-RK Lifetime, MN-vHA	Mobility keys generated during network access authentication. These keys are cached and used by the network for mobility authentication.
vHA-IP address	The IP address of the visited HA assigned to the MS.

Note Prime Access Registrar responds with the correct keys back to the HA based on the NAI in **User-Name** attribute. Prime Access Registrar returns an Access-Reject if it does not find a valid session for the NAI during the user authentication and authorization or if there are other errors.

Prepaid and Hot-Lining

Prime Access Registrar supports prepaid and hot-lining flows for WiMAX. These are supported by the existing mechanisms.

Configuring WiMAX in Cisco Prime Access Registrar

A new service type named **wimax** will be used for the WiMAX feature in Prime Access Registrar. **aregcmd** command is used to configure WiMAX in Prime Access Registrar. WiMAX service contains—Session Manager (with a session-cache resource manager and HA resource manager), Query Service that is connected to the session manager configured for this service, and Prepaid Service, which are required to connect all the flows appearing in Prime Access Registrar for WiMAX. This service will be used as a container for the new key generation modules and the existing modules such as EAP services.

Configuring WiMAX in Prime Access Registrar involves configuration of:

- Resource Manager for WiMAX
- Session Manager for WiMAX
- Query Service for WiMAX
- WiMAX properties

This section contains the following topics:

Configuring the Resource Manager for WiMAX

You must configure the following two Resource Managers:

- HA (home-agent or home-agent-ipv6)
- HA Cache (session-cache)

The HA Resource Manager must contain the IP ranges covering all the HA IP addresses that are to be assigned in round-robin. You must configure the HA Cache Resource Manager to cache the mobility keys (Table 3-3).

Note The HA Resource Manager allocates the IP addresses to the HA. If you do not configure the HA Resource Manager properly, Prime Access Registrar will not generate some of the keys, which result in an Access-Reject by the NAS.

The following shows the sample configuration for HA:

```
[ /Radius/ResourceManagers/HA ]
Name = HA
Description =
Type = home-agent
Home-Agent-IPAddresses/
Entries 1 to 1 from 1 total entries
Current filter: <all>
209.165.200.225-209.165.200.254/
```

The following shows the sample configuration for HA Cache in HAAA:

```
[ /Radius/ResourceManagers/HA-Cache ]
Name = HA-Cache
Description =
Type = session-cache
OverwriteAttributes = TRUE
QueryKey = User-Name
PendingRemovalDelay = 10
AttributesToBeCached/
        1. WiMax-Session-ID
        2. hHA-RK-Key
        3. hHA-RK-SPI
        4. MN-hHA-MIP4-Key
        5. hHA-RK-Lifetime
        6. MIP-RK
```

The following shows the sample configuration for HA Cache in VAAA:

```
[ /Radius/ResourceManagers/HA-Cache ]
Name = HA-Cache
Description =
Type = session-cache
OverwriteAttributes = TRUE
QueryKey = User-Name
PendingRemovalDelay = 10
AttributesToBeCached/
        1. vHA-RK-Key
        2. vHA-RK-SPI
        3. MN-vHA-MIP4-Key
        4. vHA-RK-Lifetime
```

When the OverwriteAttributes value is set as TRUE, the newly generated mobility keys will be cached with the session record. By default, the value is FALSE.

The HA-RK-Lifetime attribute type must be of type STRING instead of UINT32 under **/Radius//advanced/attribute\ dictionary/vendor-Specific/vendors/wimAX/subAttribute\ Dictionary.**

Note For generating RRQ-MN-HA key, we must configure MIP-RK in the AttributesToBeCached list.

Configuring the Session Manager for WiMAX

Before configuring WiMAX service, you must configure a session manager for WiMAX with a HA and session cache resource manager. The following shows an example configuration of a session manager with HA and session cache resource managers.

```
[ /Radius/SessionManagers/session-mgr-2 ]
    Name = session-mgr-2
    Description =
    IncomingScript =
    OutgoingScript =
    AllowAccountingStartToCreateSession = FALSE
    SessionTimeOut =
    PhantomSessionTimeOut =
    SessionKey =
    ResourceManagers/
        1. HA-Cache
        2. HA
```

Note If a default session manager is configured with the same key as that of the WiMAX session manager, the incoming WiMAX request will fail.

Configuring the Query Service for WiMAX

When you configure a query service for the WiMAX service in Prime Access Registrar, you must refer it to the WiMAX Session Manager that you created. While configuring WiMAX, you must refer the **WiMAX-Query-Service** parameter to a valid Query Service.

You must configure the Query key as the **User-Name** attribute, which contains the NAI. You must also configure the query service to return all the relevant mobility keys as described in Table 3-5.

Table 3-5 Mobility Keys

Key	Generated By	Used At
MN-HA-CMIP4	MN and HAAA	HA and MN
MN-HA-PMIP4	MN and HAAA	HA and PMIP4 client
MN-HA-CMIP6	MN and HAAA	MN and HA
FA-RK	MN and HAAA	MN and Authenticator
MN-FA	MN and Authenticator	FA and MN or PMIP4 client
HA-RK	HAAA or VAAA	HA and Authenticator
FA-HA	HA and Authenticator	HA and FA

The following shows a sample configuration for a WiMAX Query Service:

```
[../haQueryService ]
Name = haQueryService
Description =
Type = radius-query
IncomingScript~ =
OutgoingScript~ =
SessionManagersToBeQueried/
1. session-mgr-2
AttributesToBeReturned/
1. WiMax-Session-ID
2. HA-RK-Key
```

Note If AttributesToBeReturned is not configured, all the cached attributes will be returned.

Configuring WiMAX

When you configure the WiMAX service under **/Radius/Services**, you must set its type to **wimax** and provide the following configuration options:

```
[ //localhost/Radius/Services/wimax ]
Name = WiMAX
Description =
Type = WiMAX
IncomingScript~ =
OutgoingScript~ =
OutagePolicy~ = RejectAll
OutageScript~ =
HA-RK-Key = cisco112
HA-RK-LifeTime = 60
WiMAX-Authentication-Service = None
WiMAX-Session-Manager = None
WiMAX-Query-Service = None
WiMAX-Prepaid-Service = None
Allow-HAAA-To-Include-Keys = TRUE
Require-MSK = False
```

The syntax to generate the a WiMAX request from radclient is

simple_wimax_asn_test bob(username) bob(password)

Table 3-6 WiMAX Service Parameters

Parameter	Description
Name	Required; inherited from the upper directory.
Description	An optional description of the service.
Type	Must be set to **wimax** for WiMAX service.
IncomingScript	Optional.
OutgoingScript	Optional.
OutagePolicy	Required; must be set to **AcceptAll**, **DropPacket**, or **RejectAll**. Default is **DropPacket.**

Table 3-6 WiMAX Service Parameters (continued)

Parameter	Description
OutageScript	Optional. if you set this property to the name of a script, Prime Access Registrar runs it when an outage occurs. This property allows you to create a script that notifies you when the RADIUS server detects a failure.
HA-RK-Key	Used as the base key to generate random HA-RK-Key for all the HAs that are configured in Prime Access Registrar. By default, the value is `cisco112`. You can change this value.
HA-RK-LifeTime	Used as time (in minutes) to regenerate the HA-RK-Keys based on its lifetime.
WiMAX-Authentication-Service	A valid eap service which can be used for WiMAX authentication. By default, this value is `none`. For VAAA, it should be configured with valid radius proxy service.
WiMAX-Session-Manager	A valid session manager which has HA and HA Cache as resource managers. By default, this value is `none`.
WiMAX-Query-Service	A valid RADIUS query service configured with WiMAX session manager. By default, this value is `none`.
WiMAX-Prepaid-Service	A valid prepaid service can be given to carry out the prepaid functionality of WiMAX. Otherwise this value is set to `none`.
Allow-HAAA-To-Include-Keys	If this is set, the HAAA will include the hHA-RK-Key, hHA-RK-SPI and hHA-RK-Lifetime in the Access-Accept. Otherwise, those attributes will not be in the Access-Accept. By default this value is True.
Require-MSK	If this is set, the MSK will be provided by the AAA server as a result of successful EAP-Authentication. By default, this value is False.

WiMAX - OMA-DM Provisioning Support with BEK Key

In addition to WiMax subscriber authentication, the Prime Access Registrar generates and caches the Bootstrap Encryption Key (BEK) when it receives the authentication request from the unprovisioned WiMax subscriber/device. Prime Access Registrar can identify the unprovisioned device either by looking the special pattern in Access-Request or by performing explicit database lookup.

The BEK key derived from EMSK is calculated as follows:

BEK = the 16 most significant (leftmost) octets of HMAC-SHA256(EMSK, “bek@wimaxforum.org”).

When Prime Access Registrar receives the accounting start packet for the unprovisioned device,

1. IP, MAC address, and BEK of the unprovisioned device notifies the OMA-DM server to initiate the provisioning.
2. Prime Access Registrar maintains the IP address to MAC address association using web-service until it receives the provisioning complete message from the OMA-DM server.

The Backend Portal queries the Prime Access Registrar web-service for this unprovisioned device MAC address by giving the device IP address and also the OMA-DM server request the Prime Access Registrar web-service to validate the MAC to IP address association

The communication between Prime Access Registrar and OMA-DM/Portal server is through web-service by using SOAP over HTTPS. It is assumed that the OMA-DM server (or a mediation function) will have a web-service using which AR can communicate.

Configuring the WiMax-Provisioning

To configure WiMax provisioning:

Step 1 Configure a script object, such as wimax-provision.

```
[ //localhost/Radius/Scripts/wimax-provision ]
    Name = wimax-provision
    Description =
    Language = rex
```

--> set FileName to 'libProvisioning.so'

```
set FileName /cisco-ar/scripts/radius/rex/libProvisioning.so
```

--> set EntryPoint 'ProvisionedDeviceLookup'

```
set EntryPoint ProvisionedDeviceLookup
```

--> set InitEntryPoint 'InitializeProvisioning'

```
set InitEntryPoint InitializeProvisioning
```

--> set InitEntryPointArgs to 'ldap:wimax'

```
set InitEntryPointArgs ldap:wimax
```

ls

```
[ //localhost/Radius/Scripts/wimax-provision ]
    Name = wimax-provision
    Description =
    Language = rex
    Filename = /cisco-ar/scripts/radius/rex/libProvisioning.so
    EntryPoint = ProvisionedDeviceLookup
    InitEntryPoint = InitializeProvisioning
    InitEntryPointArgs = ldap:wimax
```

The file libProvisioning.so is come up with Prime Access Registrar kit. You have to copy it into **/cisco-ar/scripts/radius/rex** path. Entrypoint ProvisionedDeviceLookup literally looks up a datastore to check if the user is provisioned. InitEntryPoint 'InitializeProvisioning' takes care of all initialization work for entry point. InitEntryPointArgs 'ldap-wimax' says the user look up to be performed against ldap datastore. Oracle datastore can also be used wherein you have to give this property to 'oracle:wimax'.

Step 2 Configure the configured script object to the server's incoming scripting point.

set IncomingScript wimax-provsion

ls

```
[ //localhost/Radius ]
    Name = Radius
    Description =
    Version = 7.2.0.0
    IncomingScript~ = provision
    OutgoingScript~ =
```

Step 3 Webclient setup

Create a script object which calls the Prime Access Registrar's wimax-provisioning webservice.

```
[ //localhost/Radius/Scripts/WebServicecall ]
    Name = WebServicecall
    Description =
    Language = rex
    Filename = libProvisioning.so
    EntryPoint = WebServiceCall
    InitEntryPoint =
    InitEntryPointArgs =
```

Entry point should be set to WebServiceCall.

Step 4 Save the configuration:

save

Step 5 Reload the configuration:

reload