Network Working Group
Request for Comments: 2636
Obsoletes: 2604
Category: Informational

R. Gellens
Qualcomm
July 1999

Wireless Device Configuration (OTASP/OTAPA) via ACAP

Status of this Memo

This memo provides information for the Internet community. It does not specify an Internet standard of any kind. Distribution of this memo is unlimited.

Copyright Notice

Abstract

Wireless carriers today are faced with creating more efficient distribution channels, increasing customer satisfaction, while also improving margin and profitability. Industry trends are pushing the sale of handsets further into the retail channel. The cost and effort of provisioning handsets, activating users, and updating handset parameters can be greatly reduced by using over-the-air activation mechanisms. A comprehensive and extensible means for over-the-air provisioning and handset parameter updating is required.

One approach is to purchase EIA/TIA/IS-683A (Over-the-air Service Provisioning of Mobile Stations in Spread Spectrum Systems) equipment. The cost of this has led carriers to seek alternative solutions. A very viable means for providing over-the-air (OTA) provisioning is to leverage the rollout of IS-707 data services equipment, which most carriers are in the process of deploying. This paper presents an approach to OTA provisioning that utilizes the deployment of IS-707 to deliver OTA provisioning and parameter upgrading.

IS-707 data services makes available several methods of providing over-the-air provisioning and parameter updating. A well thought-out approach utilizing Internet-based open standard mechanisms can provide an extensible platform for further carrier service offerings, enhanced interoperability among back-end services, and vendor independence.

This paper describes a viable and attractive means to provide OTASP/OTAPA via IS-707, using the ACAP [ACAP] protocol.

Table of Contents

1. Terms

Application Configuration Access Protocol (ACAP) – An Internet protocol (RFC-2244) that provides remote storage and access of configuration and preference information.

Activation – A process in which a mobile station and network become programmed so that a mobile station becomes operable and can be used for cellular service once authorized by the service provider.

Authentication – A procedure used to validate a mobile station's identity.

Authentication Center – An entity that manages the authentication information related to the mobile station.

Authentication Key (A-key) – A secret 64-bit pattern stored in the mobile station. It is used to generate and update the mobile station's shared secret data. The A-key is used in the authentication process.

Authorization – An action by a service provider to make cellular service available to a subscriber.

Call – A temporary communication between telecommunications users for the purpose of exchanging information. A call includes the sequence of events that allocates and assigns resources and signaling channels required to establish a communications connection.

Cellular Service Provider – A licensee of the responsible government agency (in the U.S. a licensee of the Federal Communications Commission) authorized to provide Cellular Radiotelephone Service.

Challenge/Response Authentication Mechanism using Message Digest 5 (CRAM-MD5) – An authentication mechanism which is easy to implement, and provides reasonable security against various attacks, including replay. Supported in a variety of Internet protocols. Specified as baseline mechanism in ACAP. CRAM-MD5 is published as RFC 2195.

Code Division Multiple Access – A technique for spread-spectrum multiple-access digital communications that creates channels through the use of unique code sequences.

Customer Service Center – An entity of a service provider that provides user support and assistance to subscribers.

Customer Service Representative – A person that operates from a customer service center and provides user support and assistance to subscribers.

Diffie-Hellman Algorithm – A public-key cryptography algorithm for exchanging secret keys. Uses the equation $k = g^{xy} \bmod p$, where k is the secret key. The equation is executed by each party of the session based on the exchange of independently generated public values.

Digits – Digits consist of the decimal integers 0,1,2,3,4,5,6,7,8, and 9.

Dual-mode Mobile Station – A mobile station capable of both analog and digital operation.

Electronic Serial Number (ESN) – A 32-bit number assigned by the mobile station manufacturer used to identify a mobile station. The ESN is unique for each legitimate mobile station.

Home Location Registry (HLR) – The location register or database to which a MIN is assigned for record purposes such as subscriber information.

Message Digest 5 (MD5) – A one-way cryptographic hash function. Widely deployed in Internet protocols. Published as RFC 1321.

Mobile Identification Number (MIN) – The 10-digit number that represents a mobile station's directory number.

Mobile Station (MS) – A station, fixed or mobile, which serves as the end user's wireless communications link with the base station. Mobile stations include portable units (e.g., hand-held personal units) and units installed in vehicles.

Mobile Switching Center (MSC) – A configuration of equipment that provides cellular radiotelephone service.

Mobile Terminal Authorizing System (MTAS) – A control system that provides the capability to load the CDMA network HLR with mobile station profile information.

Number Assignment Module (NAM) – The mobile station's electronic memory module where the MIN and other subscriber-specific parameters are stored. Mobile stations that have multi-NAM features offer users the option of using their units in several different markets by registering with a local number in each location.

Over-the-air Service Provisioning Function (OTAF) – A configuration of network equipment that controls OTASP functionality and messaging protocol.

Over-the-air Parameter Administration (OTAPA) – Network initiated OTASP process of provisioning mobile station operational parameters over the air interface.

Over-the-air Service Provisioning (OTASP) – A process of provisioning mobile station operational parameters over the air interface.

Quick-Net-Connect (QNC) – An IS-707 data service capability that utilizes the Async Data Service Option number but bypasses the modem connection for a direct connection to an IP-based internet.

Roamer – A mobile station operating in a cellular system or network other than the one from which service was subscribed.

Simple Authentication and Security Layer (SASL) – An Internet protocol (RFC-2222) that provides a framework for negotiating authentication and encryption mechanisms.

Service Provider – A company, organization, business, etc. which sells, administers, maintains, and charges for the service. The service provider may or may not be the provider of the network.

Shared Secret Data (SSD) – A 128-bit pattern stored in the mobile station (in semi-permanent memory) and known by the network. The A-key is used to generate the SSD at the network and in the mobile station for comparison.

Wireless Application Protocol (WAP) – A set of network and application protocols including a datagram protocol (WDP), Transport Layer Security (WTLS), Transaction Protocol (WTP), Session Protocol (WSP), and Application Environment (WAE), which use carrier-based gateways to enable wireless devices to access Web resources. See <http://www.wapforum.org> for specifications and details.

2. Feature Descriptions

2.1 OTASP Feature Description

The Over the Air Service Provisioning (OTASP) feature allows a potential wireless service subscriber to activate new wireless services, and allows an existing wireless subscriber to make services changes without the intervention of a third party. OTASP includes the following:

- A way to establish a user profile.
- "Over-The-Air" programming of a Number Assignment Module (NAM), IMSI and Roaming Lists, including Data option parameters, and optionally, service provider or manufacturer specific parameters (e.g., lock code, call timer).
- An Authentication Key (A-key) Generation procedure.
- A-key storage

2.2 OTAPA Feature Description

The Over-the-Air Parameter Administration (OTAPA) feature allows wireless service providers to update a NAM, IMSI, and Roaming List information in the mobile station remotely without the intervention of a third party. This capability increases flexibility and reduces costs for carriers involved with mass changes that affect every handset, such as area-code splits.

OTAPA includes the following:

- Update a user's Number Assignment Module (NAM)
- Update Data option parameters
- Update service provider or manufacturer specific parameters (e.g., Server address(es), lock code, call timer).
- Update roaming lists

3. Operation

3.1 Initial Provisioning Activity

A new subscriber needs to give the intended service provider sufficient information (e.g., name, address, etc.) to prove credit-worthiness and establish a record within the service provider's billing system. In addition, the ESN of the mobile station needs to be given to the provider. This may occur in three ways:

Voice scenario — A customer care representative collects credit information during a voice conversation. This call is made from a different phone (e.g., wired service) or is initiated using the IS-683A OTASP dialing scheme (i.e., *228*xx*).

Once the user has been authorized, the customer care representative creates a record in the CDMA network HLR, thus allowing use of the CDMA network. In addition, a limited-time N-digit password is created which is tied to the ESN. The choice of N (how many digits) is up to the carrier (as a trade-off between security and user inconvenience). All required provisioning information (including the limited-time password) is loaded into the provisioning server.

The user is then told to hang up and call a special number, of the form *228 XX *<N-digit password>* SEND (the XX code is the same as used in the initial voice call). This causes the mobile station to initiate a provisioning session.

The mobile station and the provisioning server authenticate, and all required provisioning information is downloaded into the mobile station. The user receives some form of notification once the activity is complete. This notification can be an audible tone or a text message on the mobile station display. (The form and content of this notification can be part of the provisioning data downloaded by the mobile station.) Once this initial provisioning activity is complete the user has a fully authorized mobile station ready for use.

Forms scenario — An interactive user interface is presented via a browser on the mobile station. The subscriber fills in the requested information. (Note that entering non-numeric data presents some user interface challenges on many mobile devices.)

A back-end server validates the information, and if possible, the customer is authorized, a limited-time password is generated, HLR and provisioning server records are created and the actual OTASP operation begins. Otherwise, a voice call is made to a customer care representative.

Desktop scenario — The subscriber uses a desktop (or in-store kiosk) web browser to contact the carrier, and enters the usual personal information.

The carrier's server validates the information, and if possible, the customer is authorized, a limited-time password is generated, HLR and provisioning server records are created, and the subscriber is told to dial a special number on the handset. Once this code is entered, the actual OTASP operation begins. Otherwise, the user is asked to make a voice call to a customer care representative.

3.2 OTASP for Authorized Users

Users already authorized for use of the CDMA network can also initiate provisioning activity. This could happen after being directed to do so by a Customer Care representative, either from a phone conversation or via mail notification. This type of OTASP activity is needed in cases where the carrier desires to upgrade CDMA parameters in the mobile stations or in cases where mobile station troubleshooting is needed.

This type of OTASP occurs in similar fashion to the initial OTASP activity. The mobile station downloads the new provisioning information in the same way.

3.3 OTAPA Activity

Typical OTAPA capability involves upgrading a large number of mobile stations. OTAPA activity needs to be performed in a manner that does not impose on revenue bearing CDMA network activity. OTAPA operations are initiated at the customer care center. This can be accomplished by queuing a notification to the mobile station (via 1-way SMS or special caller-ID) after the provisioning server has the updated configuration data. OTAPA activity will not occur until the mobile station has acquired CDMA service on the carrier's network and the notification has reached the mobile station.

Alternatively, OTAPA can be handled by including a recheck interval in the set of data used to provision the mobile station. When using a low-overhead protocol, such as ACAP [ACAP], it is reasonable to have a mobile station check in periodically to see if anything has changed. The time of day and/or day of week that such rechecks should occur could be included in the provisioning data.

OTAPA activity can be terminated at any time due to user call activity.

4. Requirements

4.1 General Requirements

IS-683A OTASP operations occur between a mobile station and an over-the-air service provisioning function (OTAF) using IS-95A traffic channel data burst messages. OTASP/OTAPA via data services require that the CDMA carrier have an IS-707 data services capable network. The IS-707 service must be either Packet Data Service (IS-707.5) or Quick-Net-Connect (QNC).

The mobile station must support:

- IS-707 Data Service capability
- Packet/QNC RLP protocol
- PPP protocol to peer to the IS-707 IWF
- IP protocol to provide the network layer for routing to the provisioning server
- A transport layer for end-to-end communication (such as TCP)
- Authentication and optionally encryption
- Application software to handle the provisioning protocol and memory access.
- Domain Name System (DNS) query capabilities sufficient to obtain the (IP) address of the provisioning server (or the provisioning server's address could be provided during PPP negotiation).

Lastly, the ability must exist for the mobile to make a data call and (optionally) a voice call without a MIN.

4.2 OTASP Requirements

The OTASP function requires the mobile station to originate an IS-707 data call and (optionally) a voice call using a completely unprovisioned mobile station. The CDMA network must support this capability.

OTASP via data services uses a provisioning server that contains the parameter information for mobile stations. The authorizing agent (or software) at the customer care center must be able to add user and mobile station information into both the CDMA network HLR and the provisioning server during the initial authorizing process. The provisioning server must be capable of servicing a mobile as soon as its record is created.

4.3 OTAPA Requirements

IS-683A OTAPA is performed by a mobile-terminated call that downloads parameters to the mobile station. OTAPA calls occur without user interaction.

In order to perform OTAPA via data services the network needs to direct the mobile station to initiate a special-purpose data call. Several existing methods can be used to implement this capability, for example, a mobile-terminated one-way SMS message. The SMS message content can contain any information required by the mobile station. The mobile station would need a simple parser of SMS messages in order to know when to originate an OTAPA call, as opposed to normal SMS message processing. The OTAPA call would be performed in similar fashion to a registration call. More specifically, the user would not be informed of the call activity.

There are alternative means that can be employed to initiate OTAPA activity; for example, a mobile-terminated voice call with caller-ID using a specialized telephone number. Another alternative is for mobile stations to periodically check in with the provisioning server to check for updated information. ACAP, for example, is designed for such a model. The recheck interval, as well as the time of day and/or day of week that such checks should be used, can be part of the provisioning data sent to the mobile stations.

4.4 Provisioning Server Requirements

IS-683A utilizes an over-the-air service provisioning function (OTAF) to perform the network-side provisioning activity. OTASP/OTAPA via data services replaces the OTAF with a provisioning server. The provisioning server resides on an IP network within the controlled confines of the carrier. The provisioning server must perform all the service provisioning and parameter administration functions that the OTAF provides. The provisioning server must also have an interface to the carrier's Mobile Terminal Authorizing System (MTAS). This interface serves to synchronize the provisioning server with the information in the MTAS. The specific requirements of this interface depend on the capabilities and interfaces of the carrier's customer care center system(s). The provisioning server must be capable of receiving dynamic updates from the MTAS and have the provisioning information immediately available for downloading into the chosen mobile station. A standard ACAP server provides an excellent means to meet these requirements.

The provisioning server must be capable of performing an authentication procedure with the mobile station. This authentication mechanism must be capable of authenticating both the mobile station and the provisioning server.

4.5 Security Requirements

OTASP requires that an authentication procedure be performed to validate the mobile station to the provisioning server, while OTAPA requires a mechanism where the mobile validates the server.

The provisioning server must be capable of either:

- OTAF A-key generation using a Diffie-Hellman mechanism

Or:

- Receiving A-keys from the carrier network.

Since data OTASP/OTAPA operates over IP within the carrier's network, end-to-end encryption between the mobile station and the provisioning server should be considered as a future enhancement. End-to-end encryption protects against attacks from within a carrier's network, and safeguards the provisioning data (for example, roaming lists).

5. Architecture

5.1 ACAP over TCP/IP

Figure 1 shows a provisioning server in the carrier's intranet which supports the Application Configuration Access Protocol (ACAP, RFC 2244). An administrative client in the customer care domain updates this server using ACAP. Handsets are provisioned and configured using a small ACAP client.

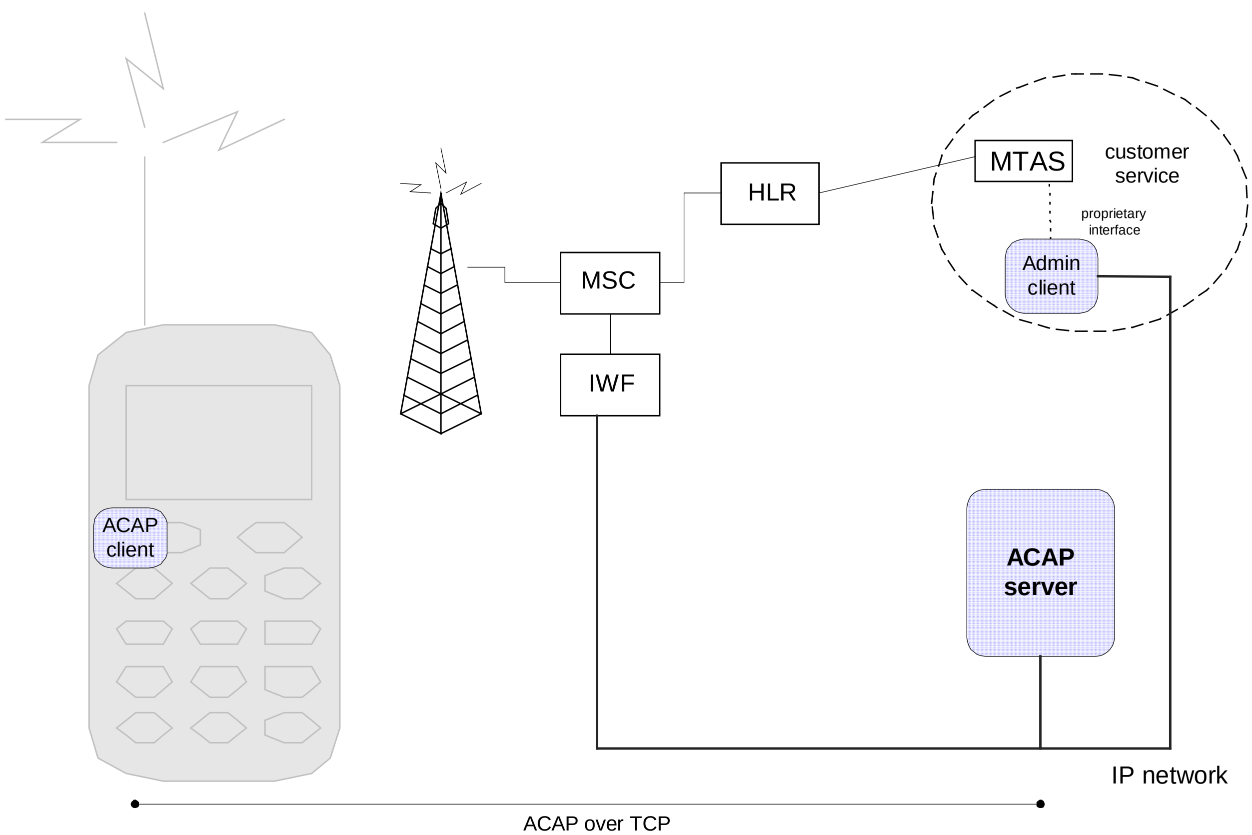

Figure 1

ACAP is an open Internet protocol designed to solve the problem of client access to configuration and related data. Among its primary goals are protocol simplicity, support for thin clients, and ease of operation over limited bandwidth. ACAP provides a high degree of extensibility, especially in authentication mechanisms, specialized attribute handling, and data management.

5.1.1 Mobile Authentication and A-Key Generation

The mobile client authenticates with the ACAP server prior to performing any activities. Authentication uses the mobile's ESN and a shared secret. Provisioned mobiles derive the shared secret from the A-Key; unprovisioned mobiles use a limited-time password as the secret.

The limited-time password is provided to the user by the Customer Care representative during the initial call (as instructions to dial a specific number). This code is N digits long. The carrier selects the number of digits, as a trade-off between security and user convenience.

The baseline ACAP authentication mechanism uses the shared secret plus a random challenge from the server as input to a one-way cryptographic hash function (specifically, keyed-MD5). This is analogous to the existing IS-683A authentication mechanism which uses a random challenge and the CAVE algorithm.

An A-Key is generated using a Diffie-Hellman exchange, as is done in IS-683A.

5.1.2 Mobile Identification

Provisioning records are identified using the ESN and the current NAM in use.

5.1.3 ACAP Server

As a standard ACAP server, the provisioning server includes configurable datasets and dataset inheritance for the management of the data stores.

The administrative client can use the same simple ACAP protocol to load and modify the ACAP server as the mobile stations uses for provisioning. While any implementation-specific mechanisms available from the server vendor could instead be used for this purpose, the ability to use ACAP can greatly simplify the administrative client, as well as make it independent of the server.

ACAP includes an authentication framework (Simple Authentication and Security Layer, SASL, RFC 2222)[SASL]. SASL allows any standard or custom authentication and encryption mechanism to be used. One of the most important features of SASL is that it is designed for a world in which what is “good enough” security today isn't good enough tomorrow. As the threat model changes, SASL allows higher-strength mechanisms to be easily added while supporting already deployed clients and servers. SASL is achieving widespread deployment in a number of Internet protocols.

Strongpoints: Since the ACAP protocol was designed for precisely this type of provisioning activity, its adoption can greatly reduce the cost, time to market, and support required for the provisioning server. Additionally, the ACAP protocol provides an open standard method for mobile stations and other systems to access the provisioning server. Commercial ACAP servers are being developed by numerous companies. The ACAP client code is very small and simple, and thus can be incorporated into virtually any mobile device at minimal cost. As an open standard, the ACAP protocol has benefited from years of review and experience.

5.1.4 Overview of ACAP Structure

ACAP organizes data by *datasets*. The structure of a dataset is defined by the *dataset class*. Generally, ACAP servers do not have knowledge of dataset classes. This allows the dataset to be expanded without modifying the server in any way. A dataset is an instantiation of the dataset class, which is a logical definition of what is in a dataset, and how it is used.

Datasets contain *entries*. Entries contain *attributes* and *values*. Attributes and values are actually *metadata*, such as *name*, *length*, and *value*. Any entry can also be a dataset (datasets are hierarchical).

For example, consider the ACAP addressbook dataset class, designed to hold names, email addresses, phone numbers, and related information for a person's contacts. A given user may have one or more addressbook datasets. Each entry holds information about one person or entity. Attributes in the entry hold specific items of information, such as the given name, surname, various email addresses, phone numbers, and so forth. If an entry is a list of people (such as a mailing list or specific group of people), it is a subdataset, containing its own entries.

Some clients may look at only a subset of the attributes. For example, a mobile handset ACAP client may download only the alias (nickname), name, primary phone number and email address of each entry, while a desktop client may access all attributes.

5.1.5 Data Organization and Capabilities

ACAP provides custom hierarchical datasets. Server data can be organized to fit the needs of the applications using the dataset.

In OTASP/OTAPA over ACAP, data on the server is organized to both take advantage of ACAP capabilities and to use items that are identical to IS-683A, allowing for reuse of IS-683A handset engines.

ACAP servers also support data inheritance. All data items which are physically in the inherited dataset and not in the inheriting dataset logically also exist in the inheriting dataset. This is recursive, as the inherited dataset can itself inherit from another dataset. This powerful concept allows potentially large groups of mobile stations to inherit items from a single common entity. For example, preferred roaming lists can be stored in datasets based on geographic areas, and automatically inherited by an individual mobile station in that area. The roaming lists could be further subdivided, for example based on tiers of free NVRAM in the mobile. The mobile client need not be aware of this; it happens entirely on the server.

ACAP uses trees to provide the data hierarchy. These data trees can be viewed as similar to the directory/file structure used with all common operating systems. The built-in inheritance mechanism, together with the hierarchical structure, makes it extremely easy to update general data without disturbing specific data.

Datasets exist within the *user*, *group*, and *host* hierarchies. The user hierarchy holds datasets which belong to individual users. The group hierarchy holds datasets which belong to groups (for example, the “Region.” groups in section 5.1.5.6). The host hierarchy holds datasets which are for specific machines or systems.

In addition to providing customizable data trees, ACAP also provides several standard datasets for all clients. There is a *capabilities* dataset that contains information on custom functionality and enhanced features available to a specific client or at the site generally. This allows a server to advertise any protocol extensions, specialized attribute handling, or other enhanced functionality it supports. A client that needs to use these features can thus easily determine what is available before trying to use them.

5.1.5.1 Structure

We divide the data accessed by the client into provisioning items, group items, and client state items. Provisioning data contains NAM items and requested-data items. Group items (such as preferred roaming lists), are not specific to any mobile device. Group items physically exist in their own datasets, but through inheritance logically appear in client datasets.

The mobile stations always read data from provisioning entries and write data to client state entries. This structure makes both mobile clients and server configuration easy and simple, while allowing for extensive custom and diagnostic capabilities.

5.1.5.2 Conventions

“” This signifies the empty string (used here for ACAP entries).

~ This is shorthand for “user/<userid>”. It is part of the ACAP protocol.

5.1.5.3 Dataset

Provisioning information is located in the “OTAP” dataset class. (The full specification of this dataset will be published in a subsequent document.) The prefix “Provision.” is used for items that are to be downloaded to the mobile, and the prefix “Client.” is used for items which the client stores on the server.

Provisioning data within the OTAP dataset is organized as a series of items, each of which is stored in its own entry. The entry name is the item name, and the “OTAP.VALUE” attribute contains the item value. This structure permits change notification on a per-item basis.

We chose the “Provision” and “Client” names to simplify various operations. For example, the mobile client can easily download all changed provisioning items by performing a search which returns the “OTAP.VALUE” attribute of all entries whose name begins with “Provision” and whose modtime is greater than the last time the client retrieved data. An administrative client can easily generate a report of all clients which have not received the most recent update by searching for all entries named “Client” whose “OTAP.modtime” attribute is less than the desired value.

A partial list of items follows.

5.1.5.4 Entries and Attributes

dataset.inherit

This is a standard ACAP attribute that identifies the location of inherited data. It exists in the “” entry (the entry with the empty name) within each dataset.

5.1.5.5 NAM Records

The OTAP dataset class contains an entry for each provisioned mobile. The standard location for provisioning records is:

/OTAP/USER/<esn>/<nam>/

This tree format allows multiple NAMs per ESN. The specific entries contain data in IS-683A parameter block types.

For example, the CDMA NAM would be stored in an entry called:

/OTAP/USER/<esn>/<nam>/Provision.CDMA-NAM/

The entries below show how NAM records would be organized on the ACAP server:

CDMA/Analog NAM

Entry-Path: /OTAP/USER/<esn>/<nam>/Provision.CDMA-AMPS-NAM/

OTAP.Value: {17} xx xx xx … xx

The CDMA/Analog NAM entry from IS-683A (section 4.5.2.1) consists of at least 129 information bits, depending on the number of SID NID list entries. This is stored as 17 (or more) octets of binary data (padding is used to ensure an integral number of octets).

Mobile Directory Number

Entry-Path:

/OTAP/USER/<esn>/<nam>/Provision.MOBILE-DN/

OTAP.Value: {10} xxxxxxx

The Mobile Directory Number from IS-683A contains BCD-encoded digits representing the phone number. This is stored as a string of 10 or more ASCII digits, e.g., “6195551212”.

CDMA NAM

Entry-Path:

/OTAP/USER/<esn>/<nam>/ Provision.CDMA-NAM/

OTAP.Value: {13} xx xx xx … xx

The CDMA-NAM entry from IS-683A (section 4.5.2.3) consists of at least 100 information bits, depending on the number of SID-NID list entries. This is stored as 13 (or more) octets of binary data (padding is used to ensure an integral number of octets).

IMSI_T

Entry-Path: /OTAP/USER/<esn>/<nam>/ Provision.IMSI_T/

OTAP.Value: {7} xx xx xx xx xx xx xx

The IMSI_T entry from IS-683A (section 4.5.2.4) consists of 55 bits of information in five fields. This is stored left-justified in 7 octets of binary data.

5.1.5.6 Server Roaming Lists

The ACAP Server will have an entry for each different roaming list configuration for a carrier. The example below assumes that the desired differentiation for the roaming list is geographic, with subdivisions for tiers of mobile free NVRAM It shows that for each region there exists a set of roaming lists per free NVRAM range. Note that a carrier can easily implement different or further differentiation (e.g., by phone vendor or product type) by simply changing the dataset tree and assigned the appropriate value to the “dataset.inherit” attribute in the provisioning records.

```
/OTAP/GROUP/region.NorthEast/free-nv.128-512/preferred.roaming.list/OTAP.Value
/OTAP/GROUP/region.NorthEast/free-nv.512-1024/preferred.roaming.list/OTAP.Value
/OTAP/GROUP/region.SouthEast/free-nv.128-512/preferred.roaming.list/OTAP.Value
/OTAP/GROUP/region.SouthEast/free-nv.512-1024/preferred.roaming.list/OTAP.Value
/OTAP/GROUP/region.NorthWest/free-nv.128-512/preferred.roaming.list/OTAP.Value
/OTAP/GROUP/region.NorthWest/free-nv.512-1024/preferred.roaming.list/OTAP.Value
/OTAP/GROUP/region.SouthWest/free-nv.128-512/preferred.roaming.list/OTAP.Value
/OTAP/GROUP/region.SouthWest/free-nv.512-1024/preferred.roaming.list/OTAP.Value
```

5.1.5.7 Requested-Data Record

Inside the OTAP dataset is an entry with the name “Provision.Requested-Data”, which contains one attribute called “OTAP.Requested-Data”. This attribute is multi-valued. It is either NIL or contains a list of strings. Each string is the name of one element of data that the server requests the client to supply.

After authenticating, the ACAP client issues a SEARCH command to retrieve the values of the “OTAP.Requested-Data” attribute of the “Provision.Requested-Data” entry. The client processes the returned values (if any) by issuing a STORE command to set one or more attributes in the “Client” entry. The value of each attribute is either the corresponding mobile value (which may be an empty string if the item has no value), or the special value “[N/A]” if the item does not exist or is unknown on the mobile.

This mechanism is quite general, and can be used in the normal OTASP case to modify the mobile’s dataset as appropriate for the condition of the mobile. For example, the inheritance could be set based on the amount of NVRAM available, to cause one set of preferred roaming list data or another to be used. This mechanism can also be used in other situations, such as to retrieve a complete set of mobile configuration parameters for diagnostic reasons.

5.1.5.8 Sample Server Entry

The entry below is an excerpt of a sample ACAP server dataset entry for a single mobile station, with an ESN of FB9876E and using NAM 1:

/OTAP/USER/FB9876E/1/

entry	=	“”
dataset.inherit	=	“/OTAP/GROUP/region.NorthEast/ free-nv.128-512/preferred.roaming.list/ OTAP.Value/”
entry	=	“Provision.Requested-Data”
OTAP.Requested-Data	=	(“Phone-Make” “Phone-Model” “SW-Rev” “Free-NVRAM”)
entry	=	“Client”
OTAP.Phone-Make	=	“Qualcomm”
OTAP.Phone-Model	=	“pdQ1900”
OTAP.SW-Rev	=	“001.030.0908”
OTAP.Free-NVRAM	=	“65536”
OTAP.Last-Modtime	=	“199812181703”
entry	=	“Provision.Mobile-DN”
OTAP.Value	=	{10} 619 555 1234
entry	=	“Provision.CDMA-NAM”
OTAP.Value	=	{13} xx xx xx xx xx xx xx xx xx xx xx xx xx

This dataset shows not only provisioning data which was downloaded into the mobile station, but also the items of client data requested by the server (the Requested-Data attribute) and the values of those items (the “Client” entry). It also indicates that the mobile client successfully stored the values associated with the modtime “199812181703”. In addition, it shows that this client inherits data (i.e., roaming lists) from the “NorthEast” region.

5.1.6 Administrative Client

The administrative client loads initial provisioning information into the server, including specifying the roaming list to inherit. The administrative client also updates provisioning server records as needed, and retrieves data for reports (such as a list of clients which have not yet been updated).

Data is loaded into provisioning records by using the ACAP STORE command. The administrative client authenticates to the ACAP server using credentials that permit access to datasets for mobiles.

When a new mobile is authorized for service, the administrative client creates the dataset by storing into it values that are specific for the device. It also sets the “dataset.inherit” attribute so that values which are not tied to the specific mobile are inherited as appropriate.

- Updates to user records

Existing user records may need updating from time to time. For example, a user may change service plans or purchase an additional or replacement mobile device, or the carrier may need to modify some aspect of provisioned data.

- Perusal and editing of provisioning records

  The administrative client can provide general browse and edit capability for user records.

- Report generation

  The administrative client can extract data from the ACAP server in order to generate reports. For example, after OTAPA activity, a carrier may wish to identify those mobiles which have not yet been updated.

- Queuing of OTAPA sessions

  Depending on the OTAPA update procedures chosen (e.g., SMS, CLID, periodic recheck), the administrative client may be involved in initiating the activity. This may or may not use an interface to the provisioning server.

5.1.7 Mobile Client

The ACAP mobile client is implemented as a state machine that performs the equivalent of IS-683A provisioning parameter information exchange and non-volatile memory storage. The ACAP Client state machine diagram (Figure 2) and descriptions are below.

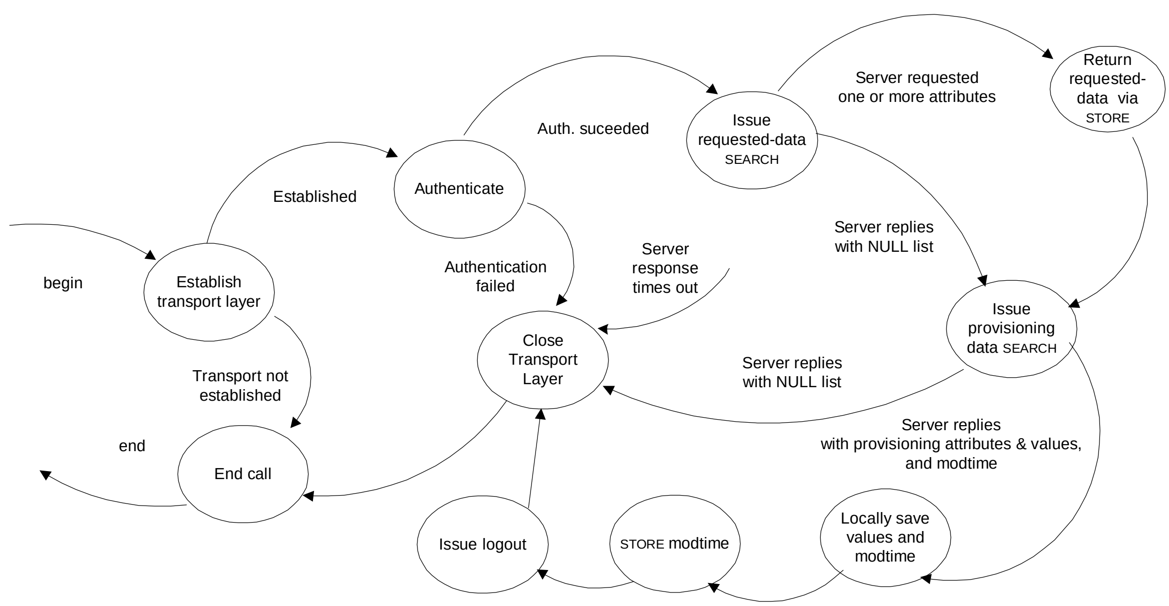

Figure 2

Establish Transport Layer/Authenticate

Authentication and/or encryption can occur at the application layer and/or at the network/transport layer.

Basic ACAP authentication occurs in the application layer (i.e., within the ACAP session), and in its baseline form uses the CRAM-MD5[CRAM-MD5] mechanism. If desired, other mechanisms can be used which provide more protection and encryption. This occurs after the transport layer is established, as shown in the client state machine diagram above

Figure 3 (below) shows the CRAM-MD5 authentication mechanism for an unprovisioned mobile. In the case of provisioned mobiles, the shared secret is derived from the A-Key, instead of the limited-time N-digit code used for unprovisioned devices.

Use of basic ACAP authentication is preferred for initial implementations of data-OTASP because it is simple, easy to implement, and all procedures and methods are in place. Stronger SASL mechanisms and/or IPSec can be rolled out in the future without disrupting the deployed base.

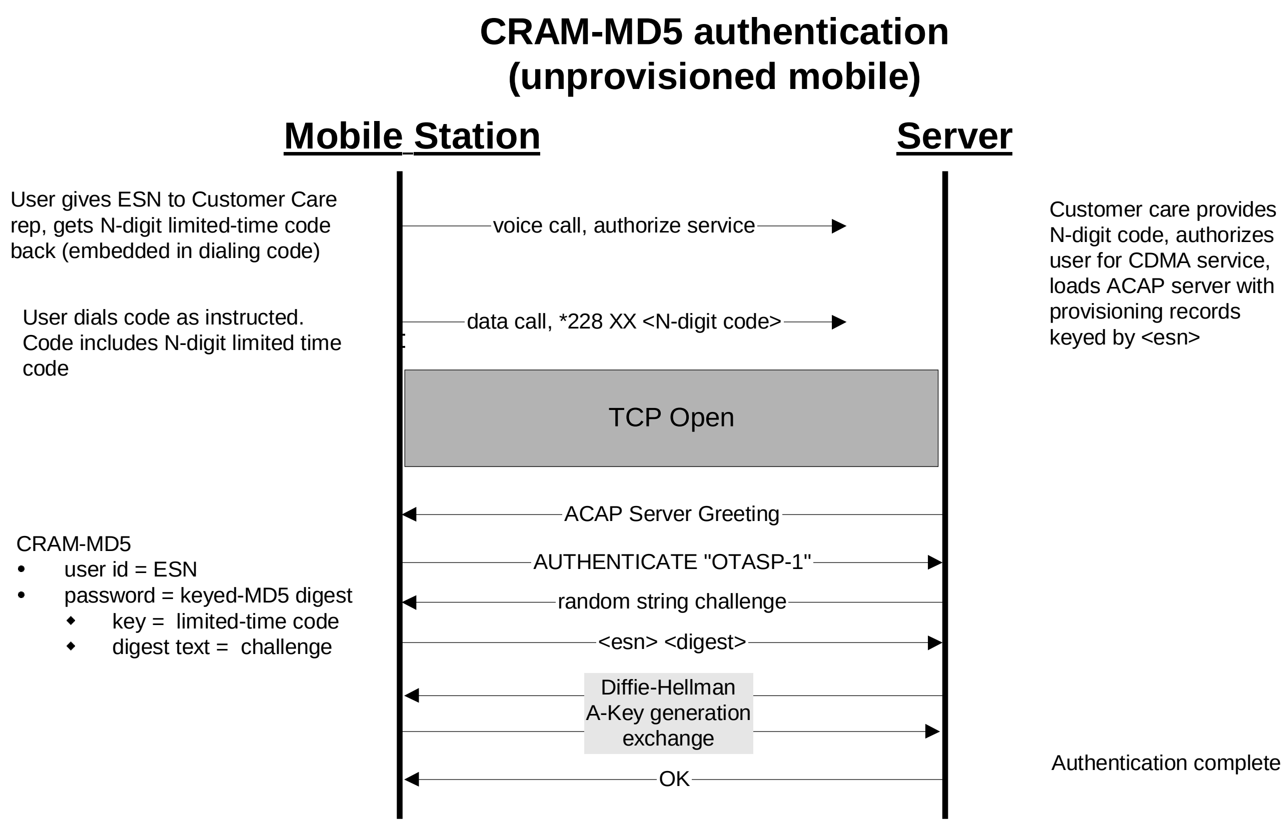

Figure 3

Requested-data SEARCH

The mobile ACAP client issues a search command asking the server to return the attribute "OTAP.Requested-Data" in the entry "Requested-Data".

Receive requested-data values

The server instructs the client to store attributes by returning one or more values of requested-data in response to the Requested-Data SEARCH.

For example, the attribute "OTAP.Requested-Data" in the entry "Requested-Data" might contain four values: "phone-make", "phone-model", "SW-Rev", and "Free-NVRAM".

STORE attribute list

If the response to the requested-data SEARCH returns any values, the client issues a STORE command. Each attribute in the STORE command corresponds to one item of requested-data. If the client does not recognize an item, it stores the string "[n/a]".

Continuing with our example, the client uses this STORE command to write four attributes into the "Client" entry. Each attribute name is identical to one value of the OTAP.Requested-Data" attribute (with the prefix "OTAP." added). Each attribute value is determined by the respective mobile value.

In our example, this STORE command sets the following attributes and values:

- "OTAP.Phone-Make" = "Qualcomm
- "OTAP.Phone-Model" = "pdQ1900
- "OTAP.SW-Rev" = "001.030.0908",
- "OTAP.Free-NVRAM" = "65536".

Provisioning data SEARCH

The mobile ACAP client issues a search command to retrieve any items of provisioning data that have changed since it last checked in (which in the initial session retrieves all provisioning data).

This SEARCH command asks the server to return the "OTAP.Value" attribute of any entries whose name starts with "provision." (case-insensitive) and whose modtime is greater than the supplied value (which is zero for an unprovisioned mobile).

Receive provisioning data and modtime

The server returns the provisioning items, each as one entry name and one attribute value. The server response to the SEARCH command includes the modtime of the latest entry returned.

Save values

The mobile writes the returned values into NVRAM.

STORE modtime

The ACAP client stores the returned modtime on the server as an acknowledgement that the data was received and NVRAM updated.

LOGOUT

The client issues the LOGOUT command.

Close transport layer

The client closes the TCP connection.

End call

The data call is terminated.

5.2 WAP with ACAP

An advantage of the ACAP solution is that is can easily coexist with a WAP-based mechanism, giving carriers more options.

A carrier can deploy handsets into its service area which use WAP-based provisioning, if desired, alongside those which use ACAP provisioning. All that is required is that the WAP server contain a small ACAP client (or an interface to an ACAP server).

Figure 4 shows how mobile stations can be configured using a WAP browser. By using an ACAP server for provisioning, carriers are free to simultaneously deploy mobile stations that use either WAP or ACAP, as desired. In either case, the ACAP server is the source for provisioning data.

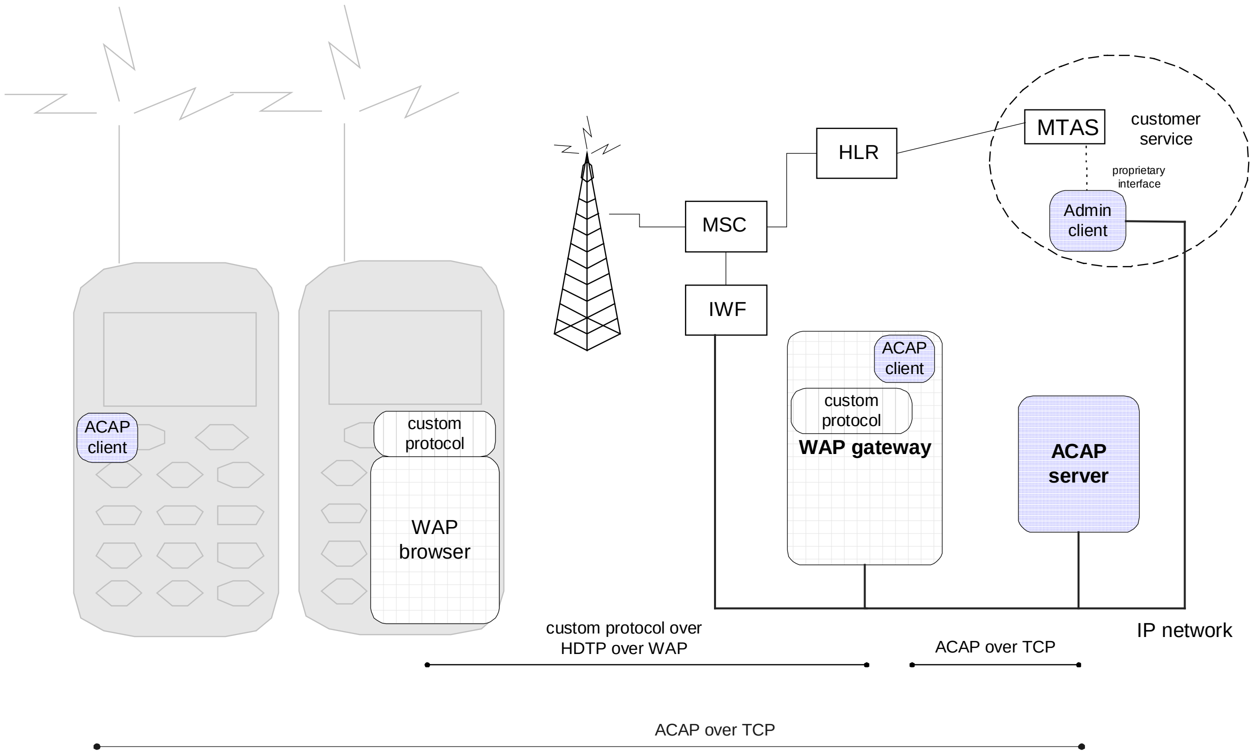

Figure 4

5.3 Network-Resident vs. Configuration Data

It is useful to recognize that wireless devices access two different types of carrier-provided data: network-resident and configuration. Network-resident data exists primarily within the carrier's network. Examples include account status, billing detail, service plan options, etc. While mobiles may access this information for user display, it resides in the network. Configuration data, in contrast, affects the operation of the handset, is usually not shown to the user, and must persist in the device.

For network-resident data access, the obvious choice is the web. The data is highly interactive and time-variant, making web browsers a reasonable solution. Any appropriate web browser can be used. There are many good reasons for having a web browser in a wireless device which contains a display and is capable of user interaction.

For configuration data, the best solution is to use ACAP. ACAP is optimized for the job, can be implemented quickly, requires a very small amount of memory, and does not depend on a display or any user interaction capability.

Trying to use the same access method for both types of data unnecessarily complicates the solution, leading to increased design, development, and debug time and expense. It makes it more difficult to offer low-cost devices. Since the two types of data fundamentally differ, it is good engineering practice to select optimal code and protocols for each.

5.4 Intellectual Property Issues

There are no known intellectual property issues with the ACAP solution. The ACAP specification was developed within the IETF, and no ownership, patent, or other IP claims have been asserted. Multiple independent vendors are developing ACAP clients and servers, in addition to the existing usage in deployed products.

6. Handset Protocol Suites

6.1 ACAP over TCP/IP

Figure 5 depicts the mobile station protocol suite for the ACAP over TCP/IP solution. The mobile station is capable of supporting both IS-683A OTASP and OTASP over ACAP.

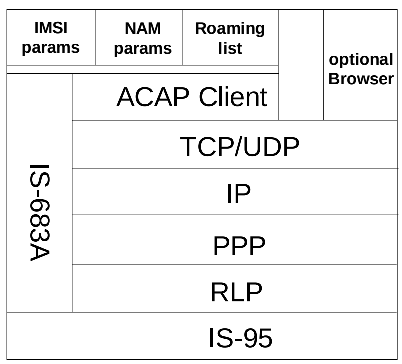

Figure 5

7. IS-683A Compatibility

7.1 OTASP Operations

To maximize compatibility and allow for reuse of IS-683A handset code, the data formats used in OTASP over ACAP are identical to those used in IS-683A. Section 5.1.5 *Data Organization and Capabilities* discusses this in more detail.

OTASP via IS-683A requires custom design and development for the specific CDMA infrastructure used by a carrier. This can greatly limit the data management capabilities and significantly reduces the extensibility of the solution. Conversely, OTASP over data can be implemented on a generic IP network using an Internet standards-based capability that provides extensible provisioning activities for carriers.

OTASP over data uses a traffic channel whereas IS-683A OTASP runs over the limited-bandwidth signaling channel.

IS-683A OTASP operations are inherently simultaneous voice and data. This allows the customer care representative to extract information from the mobile station while conversing with the user. OTASP over data services is a data-only solution (at least for now). This makes OTASP operations slightly more sequential and potentially problematic. Simultaneous voice and data will alleviate this issue.

7.2 OTASP Call Flow

The call flow diagram below (Figure 6) depicts the message sequence and operations for a typical IS-683A OTASP (provisioning) call. Any data-OTASP solution must perform all the functions of the IS-683A OTASP call. The proposed solution meets these requirements.

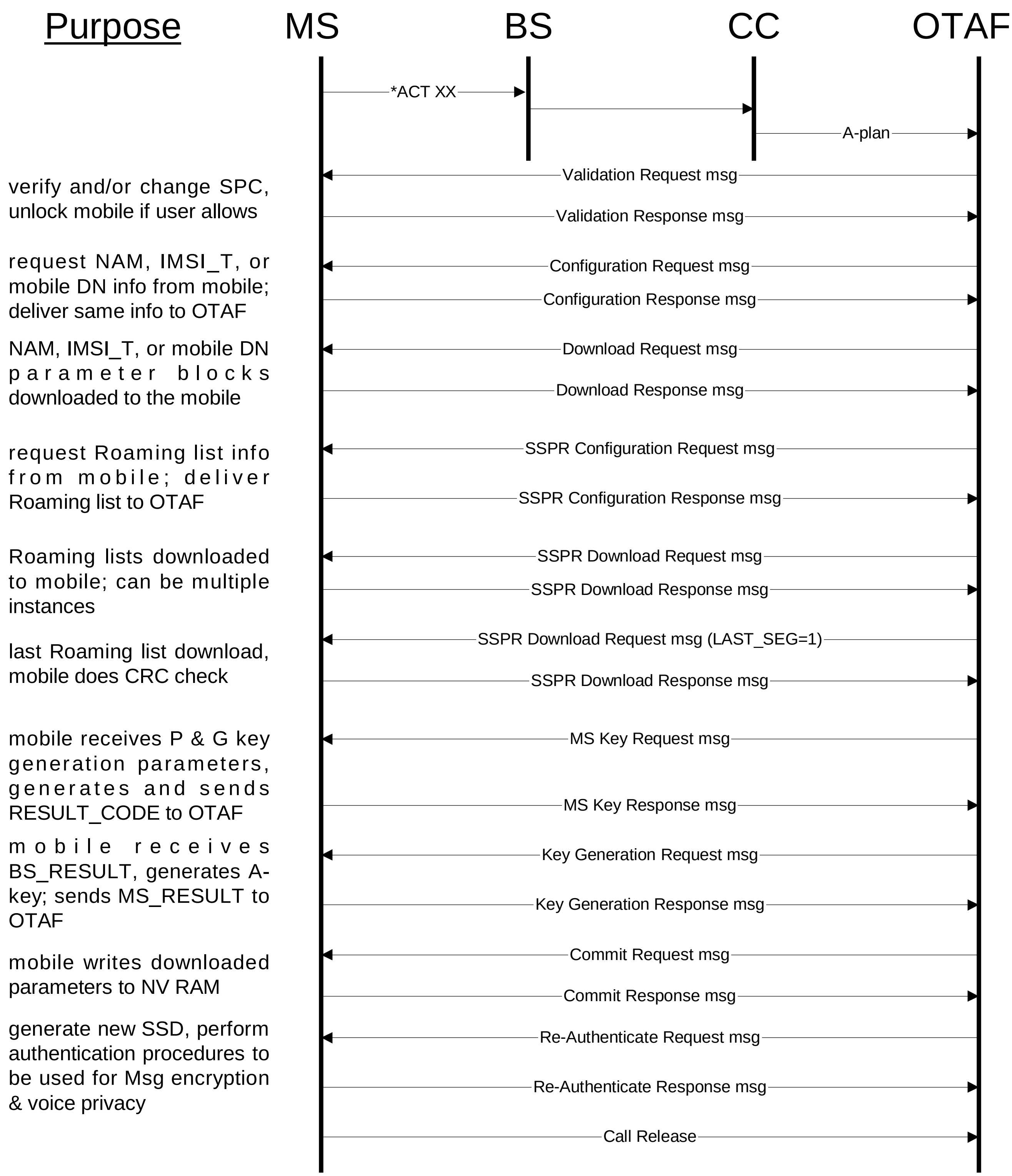
Purpose
MS
BS
CC
OTAF
*ACT XX
A-plan
verify and/or change SPC, unlock mobile if user allows
Validation Request msg
Validation Response msg
request NAM, IMSI_T, or mobile DN info from mobile; deliver same info to OTAF
Configuration Request msg
Configuration Response msg
NAM, IMSI_T, or mobile DN parameter blocks downloaded to the mobile
Download Request msg
Download Response msg
request Roaming list info from mobile; deliver Roaming list to OTAF
SSPR Configuration Request msg
SSPR Configuration Response msg
Roaming lists downloaded to mobile; can be multiple instances
SSPR Download Request msg
SSPR Download Response msg
last Roaming list download, mobile does CRC check
SSPR Download Request msg (LAST_SEG=1)
SSPR Download Response msg
mobile receives P & G key generation parameters, generates and sends RESULT_CODE to OTAF
MS Key Request msg
MS Key Response msg
mobile receives BS_RESULT, generates A-key; sends MS_RESULT to OTAF
Key Generation Request msg
Key Generation Response msg
mobile writes downloaded parameters to NV RAM
Commit Request msg
Commit Response msg
generate new SSD, perform authentication procedures to be used for Msg encryption & voice privacy
Re-Authenticate Request msg
Re-Authenticate Response msg
Call Release

Figure 6

7.3 OTAPA Operations

Data-OTAPA requires the ability to instruct mobiles to originate a data call to the provisioning server. Several viable approaches are discussed in sections 3.3 and 4.3.

OTAPA over data has the potential to require far less channel resources to download new information to mobile stations. The ACAP server inherently only communicates changes to the clients, thus only changed information needs to be downloaded to the mobile stations using OTAPA over data via ACAP.

7.4 OTAPA Call Flow

The call flow diagram below (Figure 7) depicts the message sequence for a typical IS-683A OTAPA operation. Any data-OTAPA solution must perform all the functions of the IS-683A OTAPA call. The proposed solution meets these requirements.

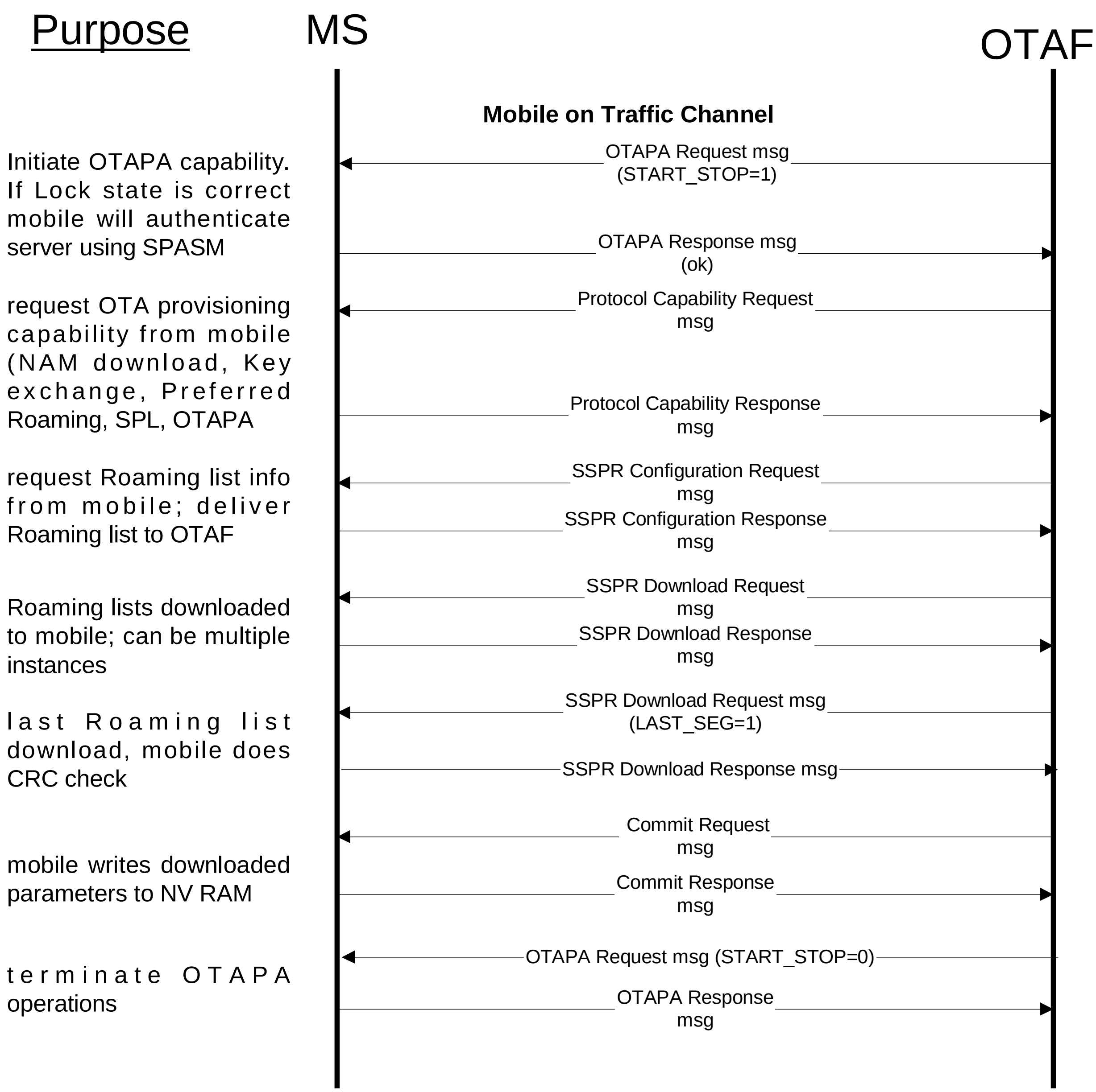
Purpose
MS
OTAF
Mobile on Traffic Channel
Initiate OTAPA capability. If Lock state is correct mobile will authenticate server using SPASM
OTAPA Request msg (START_STOP=1)
OTAPA Response msg (ok)
request OTA provisioning capability from mobile (NAM download, Key exchange, Preferred Roaming, SPL, OTAPA
Protocol Capability Request msg
Protocol Capability Response msg
request Roaming list info from mobile; deliver Roaming list to OTAF
SSPR Configuration Request msg
SSPR Configuration Response msg
Roaming lists downloaded to mobile; can be multiple instances
SSPR Download Request msg
SSPR Download Response msg
last Roaming list download, mobile does CRC check
SSPR Download Request msg (LAST_SEG=1)
SSPR Download Response msg
mobile writes downloaded parameters to NV RAM
Commit Request msg
Commit Response msg
terminate OTAPA operations
OTAPA Request msg (START_STOP=0)
OTAPA Response msg

Figure 7

8. Alternative Methods

8.1 IS-683A over TCP/IP

One alternative is to port IS-683A to TCP, remaining as close as possible to the IS-683A protocol exchange.

Figure 8 (below) depicts the architecture and communications backbone to support OTASP/OTAPA via IS-707 data services with a provisioning server based on the IS-683A OTAF function.

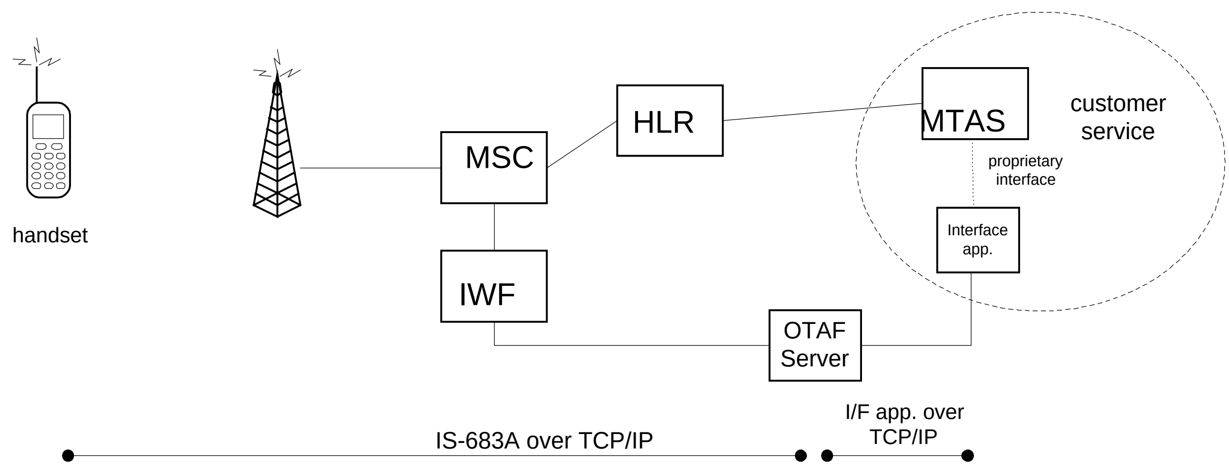

Figure 8

8.1.1 OTAF Server

This provisioning server is modeled after the IS-683A OTAF. The OTAF server performs the specific operations and messaging of IS-683A OTAF. This includes A-key reauthentication procedures.

Strongpoints:

1. OTAF and mobile station behavior mirrors IS-683A (reduced duplicate software in mobile station). Nearly all procedures fully defined.

Drawbacks:

1. The OTAF server would need to be custom-designed and built.
2. No inherent data manipulation capabilities in the OTAF server. All required or desired data management activities would have to be built from scratch.

3. Interface application would require a non-standard interface (and therefore proprietary) to OTAF server.

4. End-to-end encryption scheme still needed for full security.

8.1.2 Interface Application

This function loads all required provisioning-related information from the CDMA network information system to the OTAF server. This includes the queuing of provisioning transactions and data.

8.1.3 Protocol Handset Suite

Figure 9 below depicts the mobile station protocol suite for the IS-683A over TCP/IP solution. The OTASP client is capable of supporting both IS-683A OTASP activities or OTASP activities over the data transport.

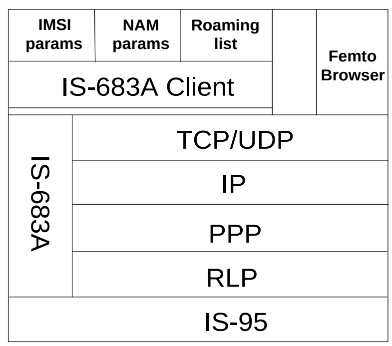

Figure 9

8.2 Browser-Based Forms

Another alternative is to use forms embedded in web pages.

Encapsulating the provisioning data into custom tags embedded in a web form is an idea that at first seems attractive. There are a lot of advantages in having a browser in the handset, web servers are very widely deployed, and everyone is familiar on some level with the web.

However, a meta-protocol for this would need to be designed, and a fully detailed specification produced. This solution requires custom software on the provider side to handle the meta-protocol. It additionally requires handset vendors to add custom software in the handset browser to handle this protocol.

This solution would require a provisioning-capable browser in every phone. While it may be desirable to have a browser, the decision to require it needs to be considered carefully, especially in light of the memory requirements it would impose on *all* devices.

This solution would complicate the handset browser, by requiring it to handle provisioning as well as browsing. As provisioning and browsing are functionally dissimilar, this code is not a natural fit within the browser. Implementing this solution would require a significant increase in development and debug resources, and thus negatively impact time-to-market and cost.

Also because the web is functionally dissimilar, a high level of carrier-side customization would be needed, leading to reduced vendor choice and increased deployment costs.

This approach would layer custom data on top of a standard protocol. This would require design work, and would not have much time for open review before deployment, greatly increasing the risk. By contrast, ACAP has had years of open review and refinement.

This approach also limits the extensibility of the solution. ACAP, conversely, is very extensible. Because ACAP is such a simple protocol, it can be added to a wide variety of applications at low cost. This allows increasing numbers of applications on the mobile device to share information with servers as well as desktop applications.

9. Conclusion

ACAP provides a high degree of extensibility, especially in authentication mechanisms, custom attribute handling, and data management. By using an Internet standard protocol, interoperability and integration with a variety of equipment is possible, and carriers are not locked into any vendor. It is also easier to add new levels of service and capabilities, especially integration with future subscriber devices and applications (e.g., email).

Since an ACAP client is so small, it can be incorporated into virtually any device, even low-end ones without displays, and can be added without sacrificing other features. The simplicity of the client and protocol directly translate to shorter development cycles and faster time-to-market.

Because the ACAP protocol was designed for precisely this type of provisioning activity, its adoption can greatly reduce the cost, time to market, and support required for the provisioning server as well as the handsets. As an open standard, the ACAP protocol has benefited from years of review and experience.

Another advantage of the ACAP solution is that is can easily coexist with a WAP-based mechanism, giving carriers more options and reducing the minimal requirement burden on mobile devices.

A carrier can deploy handsets into its service area which use WAP-based provisioning, if desired, alongside those which use ACAP provisioning. By using an ACAP server for provisioning, carriers are free to simultaneously deploy mobile stations that use either WAP or ACAP, as desired.

The lack of intellectual-property issues further adds to ACAP’s appeal.

10. References

[ACAP] Newman, C., and J. Myers, "ACAP -- Application Configuration Access Protocol", RFC 2244, November 1997.

[CRAM-MD5] Klensin, J., Catoe, R., and P. Krumviede, "IMAP/POP AUTHorize Extension for Simple Challenge/Response", RFC 2195, September 1997.

[SASL] Myers, J., "Simple Authentication and Security Layer (SASL)", RFC 2222, October 1997.

11. Security Considerations

Security is discussed in many sections of this document. In particular, the need and methods to guard against unauthorized updating of handsets, usurpation of newly-created accounts, compromise of handset security values, and disclosure of carrier proprietary data and handset parameters is covered.

12. Acknowledgments

Jim Willkie and Marc Phillips contributed greatly to this document. Their help is very much appreciated.

13. Author's Address

Randall Gellens
QUALCOMM Incorporated
6455 Lusk Boulevard
San Diego, CA 92121-2779

+1 619 651 5115
randy@qualcomm.com

14. Full Copyright Statement

Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.