Provisioning Architecture Overview Version 1.1

Version 12-Nov-2002

Open Mobile Alliance
OMA-WAP-ProvArch-v1_1-20021112-C

Continues the Technical Activities
Originated in the WAP Forum

A list of errata and updates to this document is available from the Open Mobile Alliance™ Web site, http://www.openmobilealliance.org/, in the form of SIN documents, which are subject to revision or removal without notice.

This document is available online in PDF format at http://www.openmobilealliance.org/.

Known problems associated with this document are published at http://www.openmobilealliance.org/.

Comments regarding this document can be submitted to the Open Mobile Alliance™ in the manner published at http://www.openmobilealliance.org/technical.htm

Document History	
OMA-WAP-ProvArch-v1_1-20021112-C	Current
WAP-182-ProvArch-20010314-a	Approved
OMA-WAP-ProvArch-v1_1-20021112-C	Draft

Contents

1. Scope

The Wireless Application Protocol (WAP) is a result of continuous work to define an industry-wide specification for developing applications that operate over wireless communication networks. The Open Mobile Alliance continues the work of the WAP Forum to define a set of specifications to be used by service applications. For information on the WAP architecture, please refer to "*Wireless Application Protocol Architecture Specification*" [WAPARCH].

Provisioning is the process by which a WAP client is configured with a minimum of user interaction. The term covers both over the air (OTA) provisioning and provisioning by means of, e.g., SIM cards. This specification defines the architecture of the provisioning process. The specification is an informative document.

2. References

2.1. Normative References

Not applicable.

2.2. Informative References

[E2ESEC] "Transport Layer End to End Security Specification", WAP Forum™, WAP-187-E2ESEC, URL: http://www.openmobilealliance.org/

[PROVBOOT] "Provisioning Bootstrap 1.1". Open Mobile Alliance™. OMA-WAP-ProvBoot-v1_1, URL: http://www.openmobilealliance.org/

[PROVCONT] "Provisioning Content 1.1", Open Mobile Alliance™, OMA-WAP-ProvCont-v1_1, URL: http://www.openmobilealliance.org/

[PROVSC] "Provisioning Smart Card 1.1", Open Mobile Alliance™, OMA-WAP-ProvSC-v1_1, URL: http://www.openmobilealliance.org/

[PROVUAB] "Provisioning User Agent Behaviour 1.1", Open Mobile Alliance™, OMA-WAP-ProvUAB-v1_1, URL: http://www.openmobilealliance.org/

[WAPARCH] "WAP Architecture". WAP Forum™. WAP-210-WAPArch. URL: http://www.openmobilealliance.org/

[WAPPUSH] "Push Architectural Overview", WAP Forum™, WAP-250-PushArchitecture, URL: http://www.openmobilealliance.org/

3. Terminology and Conventions

3.1. Conventions

All sections and appendixes are informative.

3.2. Definitions

Application Access Information
Information provisioned into the phone that relate to identity and applications rather than to plain connectivity.

Bootstrap Document
A connectivity or application access document with information of relevance to the bootstrap process only.

Bootstrap process (bootstrapping)
The process by which the unconfigured ME is taken from the initial state to or through the TPS Access State. This process can be system specific.

Bootstrap Server
Bootstrap Server is the sender of the bootstrap message. It may physically be co-located with a TPS but that is irrelevant from an architecture point of view. The address of the Bootstrap Server is not relevant.

Configuration Context
A Configuration Context is a set of connectivity and application configurations typically associated with a single TPS. However, the Configuration Context can also be independent of any TPS. A TPS can be associated with several Configuration Contexts, but a TPS cannot provision a device outside the scope of the Configuration Contexts associated with that particular TPS. In fact, all transactions related to provisioning are restricted to the Configuration Contexts associated with the TPS.

Connectivity Information
This connectivity information relates to the parameters and means needed to access WAP infrastructure. This includes network bearers, protocols, access point addresses as well as proxy, DNS, and application access addresses and Trusted Provisioning Server URLs.

Continuous provisioning
The process by which the ME is provisioned with further infrastructure information at or after the TPS Access state. The information received during the bootstrap may be modified. This process is generic and optional. Continuous implies that the process can be repeated multiple times, but not that it is an ongoing activity.

Logical Proxy
A logical proxy is a set of physical proxies that may share the same WSP and WTLS context (shared session id value space). This implies that physical proxies within a logical proxy share the same WSP and WTLS session cache. For example, the device does not have to create a new WTLS session when switching from CSD to SMS if the target is the same logical proxy.

MMS Proxy-Relay
A server that provides access to various messaging systems. It may operate as a WAP origin server in which case it may be able to utilize features of the WAP system.

Network Access Point

A physical access point is an interface point between the wireless network and the fixed network. It is often a Remote Access Server, an SMSC, a USSDC, or something similar. It has an address (often a telephone number) and an access bearer.

Origin Server

The server on which a given resource resides or is to be created. Often referred to as a web server or an HTTP server.

Physical Proxy

A physical proxy is a specific address with proxy functionality. It can be the IP address plus port for an IP accessible proxy, or the SME-address plus port for an SMS accessible proxy.

Privileged Configuration Context

A privileged configuration context is a special context in which it is possible to define the number of additional configuration contexts allowed. Not all WAP service providers are, however, allowed to bootstrap the privileged context.

Provisioned state

The state in which the ME has obtained connectivity information extending its access capabilities for content, applications or continuous provisioning. This state is reached when the bootstrap process has provided access to generic proxies, or the continuous provisioning process has been performed.

Provisioning document

A particular instance of an XML document encoded according to the provisioning content specification [PROVCONT].

Proxy Navigation

An in-band mechanism to provision the device in real time as defined in [E2ESEC].

Push Proxy

A WAP Push Proxy is a gateway intended to provide push connectivity between wired and wireless networks.

Trusted Provisioning Server

A Trusted Provisioning Server, is a source of provisioning information that can be trusted by a Configuration Context. They are the only entities that are allowed to provision the device with static configurations. In some cases, however, a single TPS is the only server allowed to configure the phone. Provisioning related to a specific TPS is restricted to Configuration Contexts that are associated with this TPS.

Trusted Provisioning Server Access State

The state in which the ME has obtained a minimum set of infrastructure components that enable the ME to establish the first communication channel(s) to WAP infrastructure, i.e. a trusted WAP proxy. This allows continuous provisioning, but may also provide sufficient information to the ME to access any other WAP content or application.

Trusted Proxy

The trusted (provisioning) proxy has a special position as it acts as a front-end to a trusted provisioning server. The trusted proxy is responsible to protect the end user from malicious configuration information.

3.3. Abbreviations

DNS	Domain Name System
IP	Internet Protocol
ME	Mobile Equipment
MMS	Multimedia Messaging Service
MSC	Mobile Switching Centre
NAP	Network Access Point
OTA	Over The Air
PX	Proxy
SIM	Subscriber Identification Module
SIM ATK	SIM Application Toolkit
SMSC	Short Message Service Centre
TPS	Trusted Provisioning Server
URL	Uniform Resource Locator
USSDC	Unstructured Supplementary Service Data Centre
WAP	Wireless Application Protocol
WIM	WAP Identification Module
WSP	WAP Session Protocol
WTA	Wireless Telephony Application
WTLS	Wireless Transport Layer Security
WWW	World Wide Web

4. Introduction

The purpose of this specification is to serve as the starting point for anyone who wants to know, at a high level, what is WAP Provisioning all about. This specification shall introduce the reader to the concepts and high-level architecture used to implement WAP provisioning on wireless devices.

The WAP provisioning mechanism leverages the WAP technology whenever possible [WAPARCH]. This includes the use of the WAP stack as well as mechanisms such as WAP Push [WAPPUSH]. The provisioning architecture attempts to generalise the mechanisms used by different network types so that the network specific part is isolated to the bootstrap phase.
The WAP provisioning framework specifies mechanisms to provision devices with connectivity and application access information. This provisioning framework allows one or more trusted points of configuration management to tune their respective Configuration Contexts within an ME.
The WAP infrastructure includes access points between the wireless and wire-line networks, as well as proxies for various purposes (WSP proxy, WTA proxy, etc.) and DNS servers. The device has to know about some of these elements in order to use the service they provide.

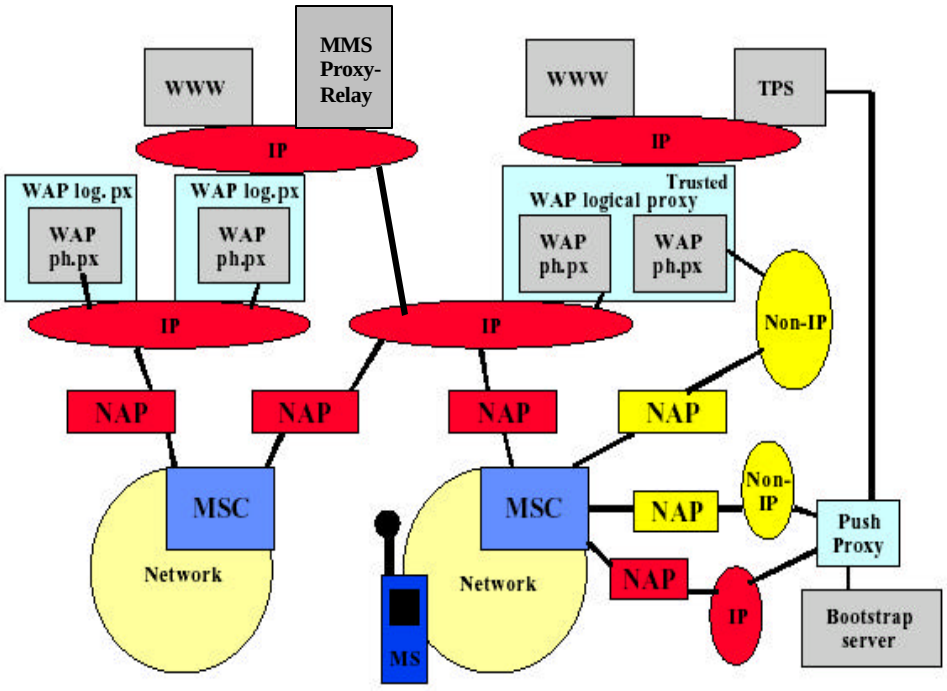


Fig. 1 - Network topology, and the addresses and methods to access particular resources. The picture shows a typical structure with / without WAP proxies and Network Access Points (often a Remote Access Server) needed to reach a particular proxy.

A non-bootstrapped WAP device is by itself not able to contact any kind of service or content through WAP.
WAP devices must thus be loaded with connectivity information, which is done during the bootstrap process. In order for the infrastructure to perform the download and/or addition of connectivity and/or application access information after the bootstrap process has been performed (continuous provisioning), WAP devices need to have a trusted relationship with the infrastructure, i.e. with one or more trusted provisioning servers, and that is in that case established during the bootstrap process.

Very few end users in a mass-market environment will be able (or interested) to perform proper set up of the various configuration contexts needed by the user. The user is seldom able to validate the correctness and reliability of a configuration (access point, proxy, provisioned DNS server, MMS Proxy-Relay). A trusted provisioning server is thus

responsible for continuous provisioning of a particular configuration context in several user devices, i.e. for the correctness and validity of connectivity and application access information, in order to protect the user from malicious service information.

Each bearer network has unique mechanisms, i.e. network specific procedures to initiate the phone or in the case of deploying the Multimedia Messaging (MMS) application it would be necessary to configure the MMS access specific parameters. In some cases, SIM cards can be used to pre-configure devices and/or application access information, but this is only a special case. Typically a bearer specific over the air provisioning mechanism is used.

5. Provisioning Framework

The architecture is based on a separation between a bearer specific bootstrap and a generic continuous provisioning mechanism. The bootstrapping is done to create a trusted relationship between the device and the infrastructure. The generic continuous provisioning leverages the bootstrapped information to load/manage one or more means to access Remote Access Servers, and/or generic WAP proxies and/or application servers (e.g. MMS Proxy-Relay). The protocols used for continuous provisioning should also be generally applicable for all provisioning needs.

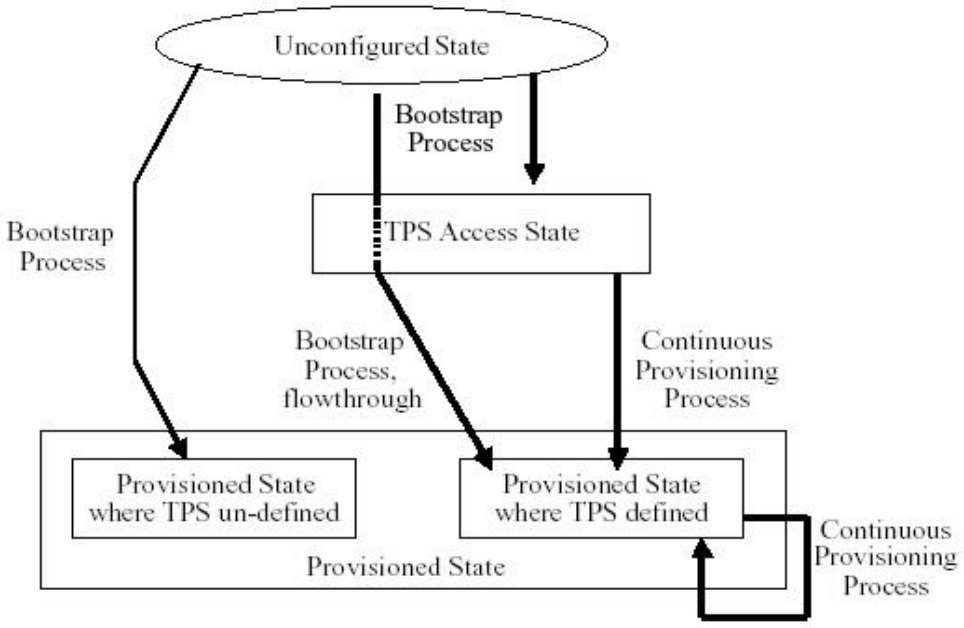


Fig. 2 - The configuration context normally is in the provisioned state. The two boxes for the provisioned state show that it is possible to do continuous provisioning only if a Provisioning URL is defined, i.e. the TPS access state has been visited in the bootstrap.

5.1. Bootstrapping and Continuous Provisioning

A device may contain one or more configuration contexts of which one must be reserved for the privileged context. The privileged context controls whether other configuration contexts are available. Hence, arbitrary parties cannot store/alter information in the privileged context. The user can normally not modify the information in the privileged context, however the user may make additions to the privileged context (for example userID and password). Furthermore, the user can modify the information that has been defined by the user [PROVUAB].

In order to initialise a configuration context and establish a basic relationship between the device and a WAP infrastructure (e.g. a WAP proxy) in this context an initial set of connectivity information must be loaded into the device. This information, usually a network access point, and/or a proxy, and a content location (the TPS), is designed to specify an access method to a Trusted Provisioning Server (TPS) [PROVCONT]. This phase has been named "bootstrap process". The bootstrap process may provide sufficient information to the devices for accessing any other WAP service or application via generic Access Points and/or WAP proxies beyond the access to a TPS [PROVBOOT].

After the bootstrap process the device configuration context may contain a trusted point of configuration (i.e. the TPS), in which case the device can use a process defined as "continuous provisioning" to update configurations in the configuration contexts associated with the TPS. The configuration context might also have connectivity or application access information already after the bootstrap process.

By separating the bootstrap and the continuous provisioning the former can be made network and bearer specific while the latter can be generic.

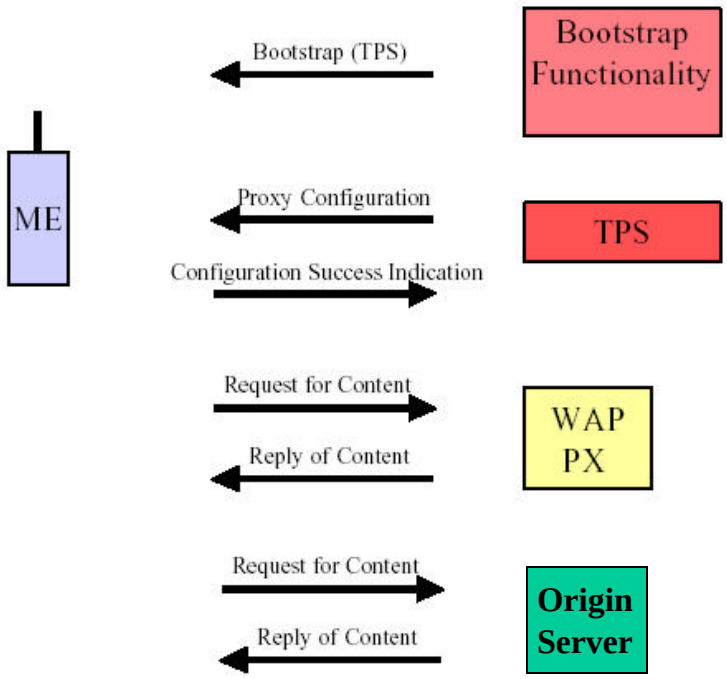


Fig. 3 - The separation between the bearer specific bootstrapping and the generic provisioning (a connectivity provisioning example). The bootstrapping process can be adapted to the bearer network type, while the continuous provisioning (updates) is based on mostly generic concepts.

5.2. Bootstrapping

The separation of the bootstrap from the continuous provisioning has several advantages.

- The bootstrap can be done in a system dependant way, leveraging the underlying system
  - can be pre-provisioned in device hardware or in SIM/WIM [PROVSC]
  - can leverage bearer and network specific provisioning mechanisms
  - can leverage voice provisioning mechanisms
  - can be based on restrictive filters (both automatic and based on user interaction) using an over the air mechanism
- The bootstrap can be based on a generic trust relationship, and the bootstrapped phone will have a specific relationship of trust established afterwards.

This allows the continuous provisioning to be defined in a generic way, providing advantages especially in a multibearer environment. For example, the identities of one or more TPS, potentially including authentication features, do not have to be known at manufacturing as they are defined in the Bootstrap process.

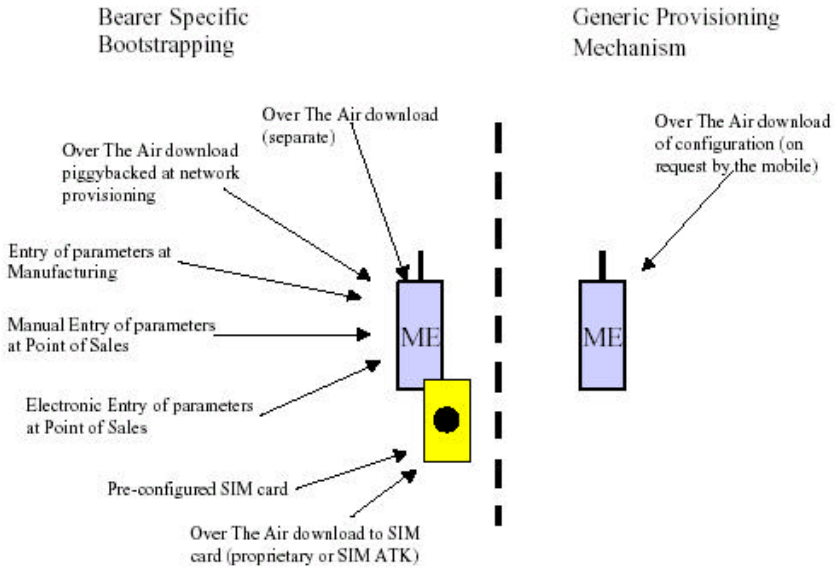
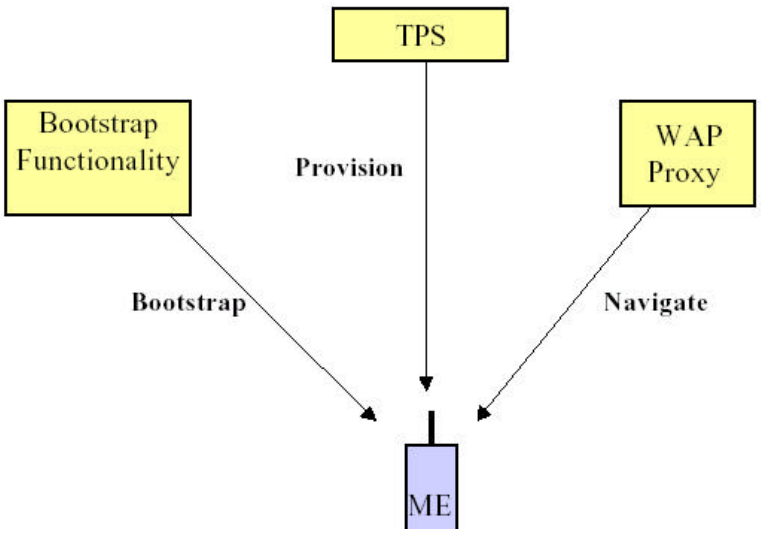


Fig. 4 - The separation between the bearer specific bootstrapping and the generic provisioning. The bootstrapping process can be adapted to the bearer network type, while the continuous provisioning (updates) is based on mostly generic concepts.

The picture above suggests a number of means to execute the bootstrap. However, in a particular bearer network only one or two of the methods would typically be used. In order to make device manufacturing, and administration of the live network manageable, it is important to select a subset for each environment. For example, all devices in a particular bearer network could be bootstrapped as an effect of the voice provisioning, with no alternative method available.

The bootstrap information defines a fixed relationship between a single configuration context and a single TPS entity. It is conceivable that a single TPS entity may allow access to a number of physical TPS's. Normally the bootstrap information is not modified. However, the bootstrap information may be modified during the continuous provisioning process and it may be possible to reset it in some cases, e.g. when it is stored inside the phone. This is required to change (reset) the trust relationship between the mobile device and the TPS. It is necessary, for example when the user changes carrier, or WAP service provider, but keeps his original mobile device. An out of band mechanism is used to reset the bootstrap information of the configuration context.

5.3. Continuous Provisioning



ɔns in

for new

nts, or
ıtexts

The provisioning includes both the content formats that express the provisioning information, as well as the protocols by which the content formats are transferred to the device. The content formats should be able to express at a minimum

- connectivity information
- bearer selection
- proxy navigation
- provisioned DNS addresses
- application access information

5.4. Navigation

Proxy navigation using navigation documents is an in-band mechanism to provision the device in real time with the path to a particular resource in a browsing environment where the usage of a proxy has been configured. It is a dynamic mechanism, not changing the static configurations. The dynamically provided documents have a limited validity time and may temporarily overwrite the static configurations. Continuous provisioning and boots trap information that is stored in a configuration context of the ME cannot be modified by navigation documents [E2ESEC].

5.5. Trust Management

The provisioning concept is built around a concept of trust between the device, i.e. a configuration context of the device, and a server side entity (the Trusted Provisioning Server).
The server side entity of the trust relationship is defined in the bootstrap of a device, but can be changed later through updates of the provisioned information. The trust relationship is thus transient, i.e. the trusted entities can define new trusted entities or even replace itself.

The device assumes (trusts) that information downloaded from the trusted entities is in the best interest of the end-user. However, the device may still allow the end user to make the final decision on the usefulness of the information.

The key components of the trust relationships for connectivity information are the:

- Optionally a Trusted Proxy, i.e. a WAP proxy that is trusted to be used between the client and the Trusted Provisioning Server for transmission of connectivity configuration related data. However, the trusted proxy does not guarantee that all resources accessed through it are non-malicious.
- Trusted Provisioning Server, a content server that is able to provide the configuration context with updates of its current configuration (connectivity and application access information). The device (configuration context) can assume that information (configurations) received from the TPS is non-malicious.
- Master Proxy, a WAP proxy that is trusted by the configuration context to provide non-malicious temporary connectivity configurations.

If used, the trusted proxy can be used to protect the end user from access to malicious connectivity configurations during the continuous provisioning process.
The verification of whether an entity that is declared to be trusted in the bootstrap process actually is worthy of end-user trust is outside the scope of the specification.

6. The Trusted Provisioning Server

The trusted provisioning server is the key element of the provisioning infrastructure. It serves the devices and applications residing on the devices with configuration information. The identity of the trusted provisioning server is established in the bootstrap of a configuration context in the device.

The server has a Provisioning Manager that controls the continuous provisioning process. The same physical server might also provide the device with OTA bootstrap information, but it has not yet been established as the trusted point, and is thus not yet the TPS.

7. The Client-Side Infrastructure

The client device has a Provisioning User Agent that manages the configuration storage on the device and executes the provisioning mechanisms. This can be a potential OTA (Over The Air) bootstrap protocol as well as the continuous provisioning process.

8. The Provisioning Content Type

The provisioning content type provides the device with information that enables it to do

- the selection of the appropriate proxy
- the selection of the network access point
- the selection of the appropriate bearer
- the selection of the provisioned DNS server
- the selection of the application resources

The provisioning content type defines mechanisms to support multiple bearers and geographically distributed access points and application resources.

9. Security Considerations

When implementing WAP Provisioning security considerations is an important piece of the concept. For the OTA bootstrap process security is built around the usage of a shared secret between the client and the bootstrap initiator. For continuous provisioning the security is built around a trust relationship between a TPS of a configuration context and the client.

A TPS is an application addressed by a URL, and is accessed either through a Proxy or directly from the ME using a Network Access Point (NAP). There might be multiple proxy access points, for example using multiple bearers, and multiple NAPs.

The TPS of a configuration context, and the means to access it, can be established in the bootstrap process of that context. This process can also initiate additional security parameters such as shared secrets and certificates. These can be used to authenticate the TPS as well as the proxy providing access to the TPS.

When the TPS has been established the continuous provisioning process handles the management of the configuration contexts associated with that TPS. This process leverages parameters provided in the bootstrap for security: address of proxy, address of DNS server, unique Network Access Points, Server Certificates for authentication, shared secret for authentication and application access information.

Security can often be enhanced significantly by leveraging the authentication and confidentiality mechanisms of WTLS and TLS.

10. Scope of different Provisioning Specifications

- Provisioning Architecture Overview

This document. The starting point for anyone who wants to know more, at a high level, about WAP Provisioning.

- Provisioning Content

This document specifies the content type used to transport connectivity information between the provisioning infrastructure (Provisioning Server, Bootstrap server) and the mobile device.

- Provisioning Bootstrap

This document specifies the mechanisms available for bootstrap of the device in different network technologies.

- Provisioning User Agent Behaviour

This document defines some of the basic behaviour of the provisioning agent in the device.

- Provisioning Smart Card

This document defines the files on a WIM card or on a SIM card that have to be used to store WAP provisioning data.

Appendix A. Change History (Informative)

Type of Change	Date	Section	Description
Class 0	12-Nov-2002		The initial version of this document.