



**UNIVERSIDAD AUTÓNOMA
DEL ESTADO DE HIDALGO**

**INSTITUTO DE CIENCIAS BÁSICAS E
INGENIERÍAS**

**PROTOCOLO DE APLICACIONES
INALÁMBRICAS (WAP)**

TESIS COLECTIVA PARA
OBTENER EL TITULO DE
LIC. EN SISTEMAS COMPUTACIONALES

PRESENTAN: JUANA HERNÁNDEZ HERNÁNDEZ
ELVIS SANDOVAL CASTELAN

ASESOR: MAESTRO: LUIS ISLAS HERNÁNDEZ

PACHUCA, HGO. DICIEMBRE 2007.

AGRADECIMIENTOS

A MI MAMÁ:

Gracias por todo lo que has hecho por mí durante todo este tiempo, sin esperar nada a cambio. Reconozco todo el esfuerzo que has tenido que realizar para que yo pudiera lograr esto, gracias por inculcarme grandes valores y principios que me han hecho crecer como persona, gracias por haberme dado la herencia más grande que una madre puede darle a un hijo, mi carrera.

iiii Gracias Mamá !!!!!

A. mis Hermanos:

Gracias por los momentos hermosos que hemos pasado, mismos que me han dado grandes satisfacciones como esta y espero ser un Buen ejemplo para ustedes

iiii (Gracias Aby y Hugo !!!!!

A mi Esposa e Hijos:

Gracias doy al creador por haberme enviado una esposa cariñosa, amorosa y comprensible, que con su apoyo he podido Lograr esto y me ha dado dos seres maravillosos mis hijos Vana Yohirí y Alan, gracias por ser el motor para salir adelante. Los Amo

iiii Gracias Viri!!!!

ÍNDICE

| | |
|---|-----------------|
| <i>Prólogo</i> | <i>i</i> |
| Objetivo | iii |
| Justificación | iv |
| Introducción | vii |
| | |
| Capítulo I: Generalidades | |
| 1.1 Protocolo de aplicaciones inalámbricas (WAP) | 1 |
| | |
| Capítulo II Componentes de la arquitectura WAP | |
| 2.1 El entorno inalámbrico de aplicaciones | 10 |
| 2.2 El entorno inalámbrico de sesión | 12 |
| 2.3 El entorno inalámbrico de transacción | 14 |
| 2.4 La capa inalámbrica de seguridad de transporte | 17 |
| 2.5 El protocolo inalámbrico de datagramas | 20 |
| | |
| Capítulo III Lenguajes | |
| 3.1 WML | 25 |
| 3.2 WMLSCRIPT | 27 |
| | |
| Capítulo IV Herramientas | |
| 4.1 Editores WML | 49 |
| 4.2 Pasarelas | 50 |
| 4.3 Emuladores | 53 |
| 4.4 Editores WBMP | 54 |
| 4.5 Kit | 55 |
| | |
| Capítulo V Servicios | |
| 5.1 Artículos | 58 |
| 5.2 Directorio WAP | 59 |
| 5.3 Beneficios para el usuario | 59 |
| | |
| Conclusiones | 60 |
| Apéndice A: Comparación entre los protocolos WAP y WEB | 61 |
| Apéndice B: Glosario de términos | 62 |
| Notas al pie | 71 |
| Bibliografía | 72 |
| Cibergrafía | 73 |

ÍNDICE DE FIGURAS

| | |
|--|----|
| Figura 1: Modelo de funcionamiento del WAP | 2 |
| Figura 2: Ejemplo de una red WAP | 3 |
| Figura 3: Componentes WAP | 5 |
| Figura 4: Arquitectura de WAP | 7 |
| Figura 5: Ejemplo de capas en WAP | 9 |
| Figura 6: Componentes del cliente de WAE | 11 |
| Figura 7: Ejemplo intercambio de primitivas entre capa Sesión y Transacción | 15 |
| Figura 8: Secuencia de primitivas para el establecimiento de una Sesión segura | 18 |
| Figura 9: Modelo de seguridad en WAP | 19 |
| Figura 10: Arquitectura del protocolo inalámbrico de datagramas | 21 |
| Figura 11: WDP sobre GSM SMS | 22 |
| Figura 12: WDP sobre GSM canal de datos de circuitos conmutados | 22 |
| Figura 13: WDP sobre servicios portadores CDMA | 23 |
| Figura 14: Documento WML y esquema de navegación | 23 |

ÍNDICE DE TABLAS

| | |
|--|----|
| Tabla 1: Primitivas de servicio de Sesión | 12 |
| Tabla 2: Tipos de primitivas de Servicio | 13 |
| Tabla 3: Primitivas de Servicio de Transacción | 14 |
| Tabla 4: Primitivas de Servicio de capa de Seguridad | 17 |
| Tabla 5: Primitivas de Servicio de la capa de Datagramas | 21 |

PROLOGO

Prólogo

Dos de los avances técnicos que han tenido un mayor impacto en la vida de millones de personas en los últimos años han sido la telefonía móvil e Internet. Internet nos permite acceder de una manera fácil, rápida y organizada a una cantidad ingente de información. Por otro lado, la telefonía móvil ha roto las ataduras entre la información y la ubicación de terminales.

Juntas estas dos tecnologías, permiten el acceso a la información sin importar la ubicación de la fuente ni la de quien la solicita.

Bien es cierto que algunos dispositivos móviles tales como radios, o sistemas propietarios, ya podían acceder a redes TCP/IP, pero no se sometían a ningún estándar y por tanto eran sistemas cerrados y con muy poca capacidad de crecimiento.

En muy poco tiempo, WAP se ha implantado como el único estándar para acceder a Internet desde cualquier sitio, usando red inalámbrica. Este estándar ha sido respaldado por miles de fabricantes de teléfonos y sistemas de telefonía, operadores, compañías de software y desarrolladores. Fruto de esta unión surgió el denominado WAP Forum que hoy en día agrupa a los fabricantes de terminales móviles.

WAP mejorará muchas de las aplicaciones que están disponibles actualmente. Las aplicaciones posibles están limitadas sólo por la imaginación.

Algunas de las que se beneficiarán de WAP son:

- Gestiones de clientes
- Notificaciones de mensajes y gestión de llamadas
- E-mail
- Servicios de valor añadido en telefonía
- Comercio electrónico, operaciones financieras y servicios de banca
- Agendas, recordatorios
- Aplicaciones corporativas de intranet

La presente se enfoca a un nivel introductorio del WAP (Protocolo de Aplicaciones Inalámbricas) y en cada uno de los capítulos se involucran los conceptos más esenciales para cada tema que tratan en particular. También se ha dispuesto de figuras con la finalidad de reforzar los conceptos.

El trabajo se desarrolla en los capítulos:

1. GENERALIDADES
2. COMPONENTES DE LA ARQUITECTURA WAP
3. LENGUAJES
4. HERRAMIENTAS
5. SERVICIOS

El primer capítulo trata acerca de los antecedentes, así como los conceptos básicos del protocolo WAP.

En el segundo capítulo se explica las capas que integran la arquitectura WAP.

En el tercer capítulo se da una breve descripción de los lenguajes que utiliza WAP.

En el cuarto capítulo se describen brevemente algunas herramientas utilizadas para WAP.

Objetivo

Generar un compendio de información que incluya:

- Conceptos básicos relacionados dentro del Protocolo WAP.
- Lenguajes utilizados en el protocolo WAP.
- Diferentes tipos de herramientas que se utilizan para el protocolo WAP.
- Facilitar una documentación desde un punto de vista general para aquellos usuarios que quieran involucrarse en el ambiente inalámbrico.

Justificación

En los últimos años el Protocolo de Aplicaciones Inalámbricas a sido de gran importancia por el uso de la telefonía móvil en Internet, y sin embargo es difícil encontrar información o manuales que nos permitan conocer este concepto, a la vez que la bibliografía que se encuentra está en inglés y por pequeñas introducciones; es por ello el afán de que la presente monografía sirva como marco de referencia o bien como un apoyo didáctico para todos aquellos que requieran algunas nociones o profundizar acerca de protocolos inalámbricos .

Introducción

Hasta hace muy poco, la computadora era la única forma de acceder a Internet, actualmente el número de host conectados supera los 300 millones y continúa en crecimiento.

Con la aparición de tecnologías para aplicaciones inalámbricas como WAP, la computadora deja de ser el único medio para acceder la información contenida en Internet y se adicionan gradualmente dispositivos móviles como teléfonos móviles, asistentes personales PDA, etc.

Actualmente existe un gran número de empresas que ofrecen contenido WAP, observándose una tendencia marcada en Europa donde el número de sitios se ha incrementado drásticamente; en América apenas hace poco ha entrado Brasil, Argentina, Estados Unidos y próximamente en Colombia.

Los servicios que ofrece la tecnología WAP son similares a los que ofrece Internet, pero actualmente con serias limitaciones. Estos servicios pueden ser: navegar, realizar transacciones bancarias, comprar o reservar entradas para espectáculos, reservar billetes de avión o tren, consultar el tiempo y las noticias o mandar un E-mail.

Además WAP habilita un nuevo tipo de servicios que no se encuentran en Internet, los servicios telefónicos. Algunos de estos ya están incorporados desde hace tiempo en los teléfonos móviles convencionales. Estos servicios dependen de las operadoras que pueden ofrecerlos gratuitamente como estrategia de marketing o bajo determinadas condiciones de contrato.

Entre los principales servicios cabe destacar la administración de llamadas (selección de llamadas, llamada en espera, etc.), buzón de voz y mensajería unificada (e-mail, faxes, mensajes de voz, etc.), servicios SMS, agenda y servicios de atención al cliente.

El diseño de avanzados micro-navegadores, menús lógicos y pantallas de mayor tamaño, seguramente permitirán un manejo ágil de la web a través del móvil. De momento surgen de manera constante páginas y sitios web con contenidos y servicios específicos para los usuarios de estos pequeños teléfonos. También cabe destacar el aumento de las alianzas entre compañías de distintos ámbitos (bancos, cadenas de televisión, agencias de viajes, etc.) con empresas de telecomunicaciones para mostrar sus contenidos en este nuevo mercado que está surgiendo. Esto permite augurar que en poco tiempo la limitación del uso de estos servicios no será debida a la falta de contenidos, sino a carencias técnicas, de manejabilidad o de precio.

Los grandes beneficiarios de esta nueva tecnología en principio no van a ser los usuarios particulares, que pueden encontrar actualmente poco práctico en comparación con la versatilidad de un PC, sino las empresas dado que se podrá acceder a información de *stocks*, realizar pedidos o acceder a la Intranet desde cualquier sitio del mundo y en cualquier circunstancia.

Antecedentes del WAP

Aunque existe una patente sobre tecnología inalámbrica de Geoworks Corp de Alameda, California en 1994, se considera que la primera compañía en iniciar un proyecto con el propósito de crear un protocolo genérico fue Ericsson en 1995. Este protocolo, denominado ITTP (Intelligent Terminal Transfer Protocol), pretendía ofrecer nuevos servicios mediante telefonía móvil estableciendo comunicaciones entre un nodo (donde resida la aplicación) y un teléfono móvil inteligente. Este protocolo no tuvo éxito.

Posteriormente Nokia y Unwired Planet (ahora Phone.com) empezaron a crear sus propios productos con sus propios protocolos con la intención de que alguno de ellos se estandarizase y obtener ventajas competitivas. La propuesta de Unwired Planet fue el HDML (Handheld Device Markup Language) y el HDTP (Handheld Device Transport Protocol), con una estructura bastante similar al HTML adaptado a pequeñas pantallas y capacidad de comunicación mediante radiofrecuencia. Por su parte, Nokia presentó el Smart Messaging que procuraba una conexión entre Internet y dispositivos GSM mediante SMS (Short Message Service) y el lenguaje TTML (Tagged Text Markup Language).

WAP (Wireless Application Protocol) o Protocolo de Aplicaciones Inalámbricas , nace como producto de un proceso de convergencia entre Internet y los sistemas inalámbricos que comenzó a gestionarse en 1997, cuando Nokia, Motorola, Ericsson y Unwired Planet, ahora Phone.com, crearon el WAPForum.

El objetivo de este Foro era proponer unas especificaciones comunes que se convirtieran en un estándar "de facto" para asegurar el crecimiento de la telefonía móvil y compatibilidad de todos los componentes WAP independientemente de la compañía que lo haya desarrollado. En abril de 1998 apareció la primera versión (WAP 1.0). Este protocolo ya ha tenido dos actualizaciones (WAP 1.1 y WAP 1.2). Actualmente el número de miembros asciende a más de 100.

Según cifras publicados en la revista Norteamericana Wired, (<http://www.wired.com>) 470 millones de personas usan teléfonos móviles y se espera que para el 2001, el número se incremente a 530 millones y en el 2004 ascienda a un billón de usuarios móviles. De esta cantidad, se estima que para finales del 2000, 10 millones sean móviles habilitados para WAP.

CAPITULO I

GENERALIDADES

OBJETIVOS:

- Proporcionar elementos necesarios que permitan conocer los conceptos básicos del protocolo WAP.
- Describir los componentes que integran el protocolo WAP.

1.1 PROTOCOLO DE APLICACIONES INALÁMBRICAS (WAP)

El **Protocolo de Aplicaciones Inalámbricas** surge como la combinación de dos tecnologías de amplio crecimiento y difusión durante los últimos años: *Las Comunicaciones Inalámbricas e Internet*.

WAP, es un estándar de fábrica para la presentación y envío de información y utilización de servicios adicionales de telefonía sobre dispositivos móviles y otros terminales inalámbricos.

A diferencia de las tecnologías de Internet para computadoras, WAP esta pensado para dispositivos que tienen algunas limitaciones técnicas inherentes a la tecnología actual como son:

- Menor potencia de procesamiento
- Menor capacidad en Memoria (ROM-RAM)
- Restricciones de suministro de potencia
- Despliegues pequeños
- Dispositivos de entrada diferentes

De igual forma, las redes inalámbricas (Wireless Network), imponen limitaciones como:

- Menor ancho de banda
- Mayor latencia
- Menos estabilidad de la conexión
- Disponibilidad menos predecible.

La especificación de WAP hace uso de los mejores estándares existentes a nivel de protocolos, proxies, seguridad, etc. Y está desarrollando nuevas extensiones donde sea necesario.

Algunos de los objetivos del WAPForum fueron ofrecer una especificación independiente de la interfaz de aire (SMS, PPP, CDMA, etc.), y de los dispositivos (teléfonos móviles, PDA...), interoperabilidad, seguridad y que los sistemas fueran escalables y eficientes

Mas allá de la posibilidad de acceder a los servicios de información contenidos en Internet, el protocolo pretende proveer de servicios avanzados adicionales como, por ejemplo: El desvío de llamadas inteligente, el cual proporciona una interfaz al usuario y se le pregunta la acción que desea realizar (aceptar la llamada, desviarla a otra persona, desviarla a un buzón vocal, etc.)

Para ello, se parte de una estructura basada en la arquitectura definida para el *World Wide Web (WWW)*, pero adaptada a los nuevos requisitos del sistema. En la Figura 1 se muestra el esquema de la arquitectura WAP.

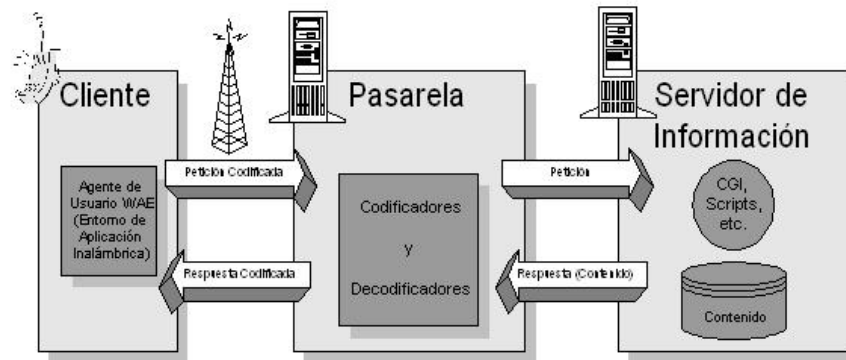


Figura 1: Modelo de funcionamiento del WAP

De esta forma, en el terminal inalámbrico existirá un “*micro navegador*¹” encargado de la coordinación con la pasarela, el cual realiza peticiones de información que son tratadas y redirigidas al servidor de información adecuado. Una vez procesada la petición de información en el servidor, se envía esta información a la pasarela que de nuevo procesa para enviarlo al terminal inalámbrico.

Para conseguir consistencia en la comunicación entre el terminal móvil y los servidores de red que proporcionan la información, WAP define un conjunto de componentes estándar:

- Un modelo de nombres estándar. Se utilizan las URIs² definidas en WWW para identificar los recursos locales del dispositivo (tales como funciones de control de llamada) y las URLs³ (también definidas en el WWW) para identificar el contenido WAP en los servidores de información.
- Un formato de contenido estándar, basado en la tecnología WWW.
- Protocolos de comunicación estándares, que permitan la comunicación del *micro navegador* del terminal móvil con el servidor Web en red.

Veamos ahora un modelo global de funcionamiento de este sistema en la Figura 2.

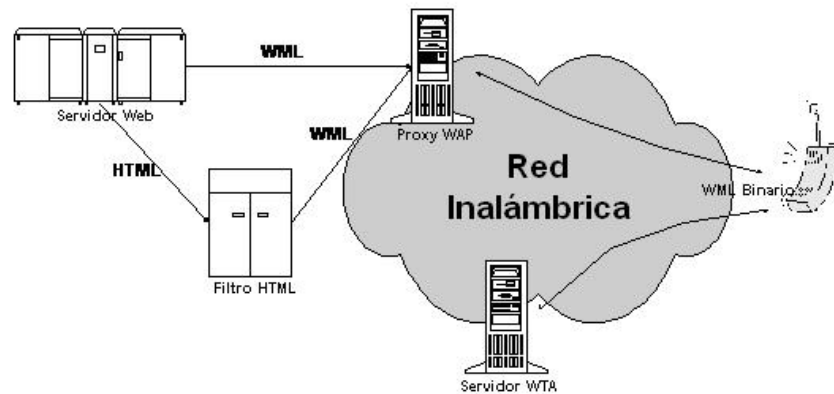


Figura 2: Ejemplo de una red WAP

En el ejemplo de la figura, el terminal móvil tiene dos posibilidades de conexión: a un proxy WAP, o a un servidor WTA. El proxy WAP traduce las peticiones WAP a peticiones Web, de forma que el cliente WAP (el terminal inalámbrico) pueda realizar peticiones de información al servidor Web. Adicionalmente, este proxy codifica las respuestas del servidor Web en un formato binario compacto, que es interpretable por el cliente. Por otra parte, el segundo de ellos, el Servidor WTA⁴ está pensado para proporcionar acceso WAP a las facilidades proporcionadas por la infraestructura de telecomunicaciones del proveedor de conexiones de red.

CAPITULO II

COMPONENTES DE LA ARQUITECTURA WAP

OBJETIVOS:

- Describir los componentes que integran el protocolo WAP.
- Describir las capas que se utilizan en las aplicaciones inalámbricas

COMPONENTES DE LA ARQUITECTURA WAP

Los componentes involucrados en aplicaciones WAP son muy similares a los de WWW como se ilustra en la Figura 3, excepto por la incorporación de una pasarela utilizada para servir de intermediario entre el mundo inalámbrico e Internet.

Ciente: Esta constituido por un terminal inalámbrico habilitado para usar el protocolo WAP, contiene un micro navegador que sirve de interfaz de usuario de manera análoga a los navegadores web.

La especificación del micronavegador ha sido diseñada de tal forma que el código resultante sea compacto y eficiente y para que provea una interfaz flexible y potente a los usuarios.

El cliente puede hacer uso de funciones WTA (Wireless Telephony Application) para acceder a funcionalidades de control de llamada, libreta de direcciones, mensajes etc. a través de guiones escritos en WMLScript.

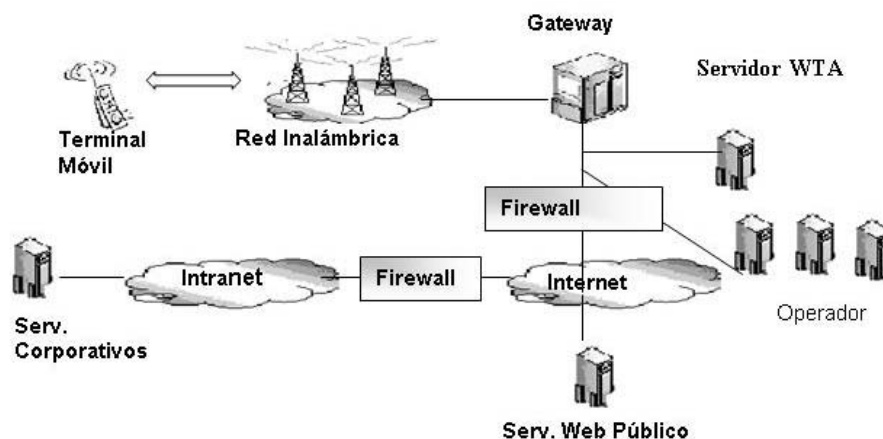


Figura 3: Componentes WAP

El lenguaje usado para crear las interfaces de usuario en WAP es denominado WML, el cual es una especialización de XML para sistemas móviles y provee un conjunto de etiquetas similar a HTML, pero muy reducido y adecuado al entorno inalámbrico.

Los documentos WML son denominados Deck y están estructurados en unidades bien definidas llamadas Cards; también se dispone de WMLScript, un lenguaje estandarizado para realizar guiones que corren en el cliente de manera similar a JavaScript y que permite realizar labores como validación de datos, realización de cálculos, etc.

Pasarela: Es la encargada de realizar el traslado de requerimientos WAP a requerimientos HTTP, hace uso de tecnología proxy estándar para web, para permitir la conexión de terminales inalámbricos al dominio web; típicamente tiene dos funciones:

- Pasarela de Protocolo: Permitiendo la traducción de WAP a HTTP
- Codificador/Decodificador: compactar el contenido WML para ser enviado por el medio inalámbrico.

Además, esta pasarela puede prestar servicios de seguridad para aplicaciones de comercio electrónico, realizar traducción de páginas HTML a WML, ofrecer servicios de DNS, caché para optimizar la respuesta a los móviles, y servir de interfaz hacia el sistema del operador para prestar servicios de localización, personalización, configuración de servicios etc.

Servidor de origen: Típicamente es un servidor web al que se ha modificado para servir contenidos WAP, recibe peticiones de páginas WML, imágenes WBMP, páginas compiladas. Haciendo uso de tecnologías para aplicaciones web como CGI, Servlets, ASP, PHP, etc. se puede generar contenido dinámico WML a partir de Bases de Datos, Servidores de Correo, etc.

Servidor WTA: Es un servidor que responde directamente a los requerimientos de un cliente y es usado para proveer acceso WAP a las características de la infraestructura de red inalámbrica, como por ejemplo, configurar servicios, utilizar servicios de valor agregado etc.

En la Figura 3 también se puede apreciar el alcance de la misma a nivel de Internet e intranet.

Es importante además tener en cuenta que respecto a la seguridad de los datos transportados se debe integrar la seguridad de las redes inalámbricas basadas en WAP (WTSL) y las de redes TCP/IP (SSL)

Una vez introducido al sistema, vamos a ver la arquitectura que le da consistencia. La arquitectura WAP está pensada para proporcionar un *“entorno escalable y extensible para el desarrollo de aplicaciones para dispositivos de comunicación móvil”*. Para ello, se

define una estructura en capas, en la cual cada capa es accesible por la capa superior así como por otros servicios y aplicaciones a través de un conjunto de interfaces muy bien definidos y especificados. Este esquema de capas de la arquitectura WAP la podemos ver en la Figura 4.

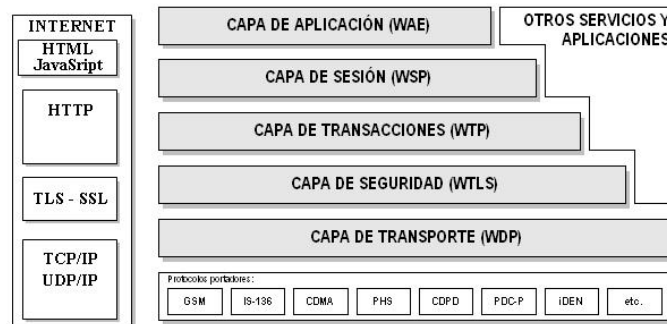


Figura 4: Arquitectura de WAP

Hagamos un recorrido por estas capas de forma breve, antes de pasar a analizarlas con más profundidad.

Capa de Aplicación (WAE⁵)

El *Entorno Inalámbrico de Aplicación (WAE)* es un entorno de aplicación de propósito general basado en la combinación del *World Wide Web* y tecnologías de Comunicaciones Móviles.

Este entorno incluye un *micro navegador*, del cual ya hemos hablado anteriormente, que posee las siguientes funcionalidades:

- Un lenguaje denominado WML⁶ similar al HTML, pero optimizado para su uso en terminales móviles.
- Un lenguaje denominado WMLScript, similar al JavaScript (esto es, un lenguaje para su uso en forma de Script)
- Un conjunto de formatos de contenido, que son un conjunto de formatos de datos bien definidos entre los que se encuentran imágenes, entradas en la agenda de teléfonos e información de calendario.

Capa de Sesión (WSP⁷)

El *Protocolo Inalámbrico de Sesión (WSP)* proporciona a la Capa de Aplicación de WAP interfaz con dos servicios de sesión: Un servicio orientado a conexión que funciona por encima de la Capa de Transacciones y un servicio no orientado a conexión que funciona por encima de la Capa de Transporte (y que proporciona servicio de datagramas seguro o servicio de datagramas no seguro).

Actualmente, esta capa consiste en servicios adaptados a aplicaciones basadas en la navegación Web, proporcionando las siguientes funcionalidades:

- Semántica y funcionalidades del HTTP/1.1 en una codificación compacta.
- Negociación de las características del Protocolo.
- Suspensión de la Sesión y reanudación de la misma con cambio de sesión.

Capa de Transacciones (WTP⁸)

El *Protocolo Inalámbrico de Transacción (WTP)* funciona por encima de un servicio de datagramas, tanto seguros como no seguros, proporcionando las siguientes funcionalidades:

- Tres clases de servicio de transacciones:
 - Peticiones inseguras de un solo camino (Permite enviar mensajes sin contestación por parte del receptor).
 - Peticiones seguras de un solo camino (Permite que el receptor notifique si ha recibido el mensaje).
 - Transacciones seguras de dos caminos (Petición-respuesta. Consiste en que cuando el emisor recibe el resultado enviado por el receptor, se vuelve a hacer una confirmación al receptor).
- Seguridad usuario-a-usuario opcional.
- Transacciones asíncronas.

Capa de Seguridad (WTLS⁹)

La *Capa Inalámbrica de Seguridad de Transporte (WTLS)* es un protocolo basado en el estándar SSL, el cual se basa a su vez en la especificación SSL3.0 publicada por Netscape, y utilizado en el entorno Web para la proporción de seguridad en la realización de transferencias de datos. Este protocolo ha sido especialmente diseñado para los protocolos de transporte de WAP y optimizado para ser utilizado en canales de comunicación de banda estrecha.

Para este protocolo se han definido las siguientes características:

- Integridad de los datos. Este protocolo asegura que los datos intercambiados entre el terminal y un servidor de aplicaciones no ha sido modificada y no es información corrupta.
- Privacidad de los datos. Este protocolo asegura que la información intercambiada entre el terminal y un servidor de aplicaciones no puede ser entendida por terceras partes que puedan interceptar el flujo de datos.
- Autenticación. Este protocolo contiene servicios para establecer la autenticidad del terminal y del servidor de aplicaciones.
- Servicio de protección – Repudio de Servicio: WTLS tiene facilidades para detectar y rechazar datos que son reenviados o no verificados satisfactoriamente.

Capa de Transporte (WDP¹⁰)

El *Protocolo Inalámbrico de Datagramas (WDP)* proporciona un servicio fiable a los protocolos de las capas superiores de WAP y permite la comunicación de forma transparente sobre los protocolos portadores válidos.

Debido a que este protocolo proporciona un interfaz común a los protocolos de las capas superiores, las capas de Seguridad, Sesión y Aplicación pueden trabajar independientemente de la red inalámbrica que dé soporte al sistema.

Antes de pasar a estudiar con más profundidad cada uno de estos protocolos, veamos tres ejemplos de interconexión de estas capas en la Figura 5:

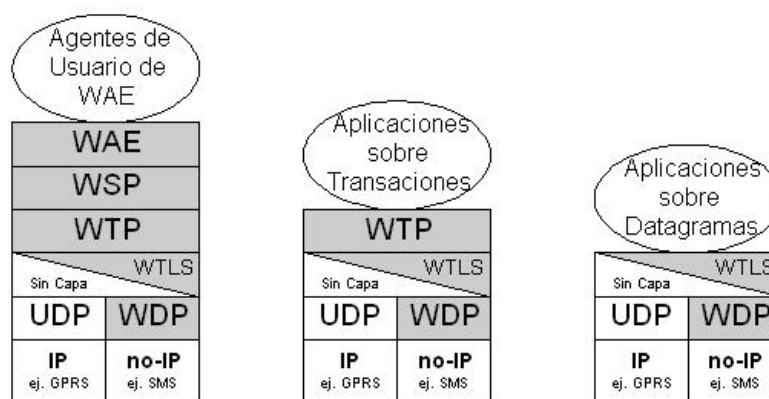


Figura 5: Ejemplo de capas en WAP

Así pues, dependiendo de la aplicación en cuestión, la comunicación se realizará con una determinada capa de la estructura de WAP.

Servicios Portadores (Bearers)

WAP esta diseñado para operar sobre una variedad de diferentes servicios portadores, incluyendo SMS, Circuit Switched Data (CSD) y Packet Data entre otros. Los portadores ofrecen diferentes niveles de calidad de servicio respecto a throughput, rata de errores y retardos, WAP esta diseñado para compensar o tolerar variaciones en el nivel de servicio.

Otros servicios y aplicaciones

WAP posibilita que otras aplicaciones y servicios utilicen el stack WAP a través de interfaces bien definidas, esto permite usar a WAP para aplicaciones que no están especificadas por WAP.

2.1 EL ENTORNO INALÁMBRICO DE APLICACIONES

El objetivo del *Entorno Inalámbrico de Aplicaciones* es construir un entorno de aplicación de propósito general, basado fundamentalmente en la filosofía y tecnología del World Wide Web (WWW). Principalmente, se pretende establecer un entorno que permita a los operadores y proveedores de servicios construir aplicaciones y servicios que puedan utilizarse en una amplia variedad de plataformas inalámbricas de forma útil y eficiente.

De esta forma, la arquitectura del Entorno Inalámbrico de Aplicaciones (WAE) está enfocado principalmente sobre los aspectos del cliente de la arquitectura del sistema de WAP, esto es, de los puntos relacionados con los agentes de usuario¹¹. Esto es debido a que la parte que más interesa de la arquitectura es aquella que afecta principalmente a los terminales móviles, esto es, a aquellos puntos en los cuales van a estar ejecutándose los diversos agentes de usuario.

Si volvemos sobre la *Figura 1*, vemos que entre los agentes de usuario localizados en el cliente (en el terminal móvil) y los servidores de información se define un nuevo elemento: *Las Pasarelas*. Su función es codificar y decodificar la información intercambiada con el cliente, para así minimizar la cantidad de datos radiados, así como minimizar el proceso de la información por parte del cliente.

Basándonos en esta arquitectura, vamos a profundizar un poco más en los componentes de este Entorno Inalámbrico de Aplicación. Tal y como podemos observar en la *Figura 6*, se divide en dos partes, dos capas lógicas:

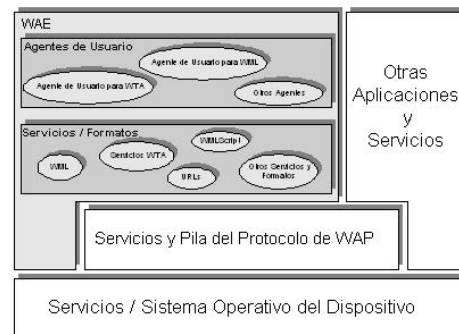


Figura 6: Componentes del Cliente de WAE

- Los Agentes de Usuario, que incluye aquellos elementos como navegadores, agendas telefónicas, editores de mensajes, etc.
- Los Servicios y Formatos, que incluyen todos aquellos elementos y formatos comunes, accesibles a los Agentes de Usuario, tales como WML, WMLScript, formatos de imagen, etc.

Como se puede ver en la Figura, dentro de WAE se separan Servicios de Agentes de Usuario, lo que proporciona flexibilidad para combinar varios Servicios dentro de un único Agente de Usuario, o para distribuir los Servicios entre varios Agentes de Usuario.

Los dos Agentes de Usuario más importantes son el Agente de Usuario para WML y el Agente de Usuario para WTA.

El Agente de Usuario para WML es el Agente de Usuario fundamental en la arquitectura del Entorno Inalámbrico de Aplicación. A pesar de su importancia, este Agente de Usuario no está definido formalmente dentro de esta arquitectura, ya que sus características y capacidades se dejan en manos de los encargados de su implementación. El único requisito de funcionalidad que debe cumplir este Agente de Usuario, es el de, proporcionar un sistema intérprete a los lenguajes WML y WMLScript, de forma que se permita la navegación desde el terminal móvil.

Por otra parte, el Agente de Usuario para WTA permite a los autores acceder e interactuar con las características de los teléfonos móviles (p.e. Control de Llamada), así como otras aplicaciones supuestas en los teléfonos, tales como agendas de teléfono y aplicaciones de calendario.

2.2 EL PROTOCOLO INALÁMBRICO DE SESIÓN

El *Protocolo Inalámbrico de Sesión* constituye la capa que se sitúa por debajo de la capa de Aplicación, proporcionando la capacidad necesaria para:

- Establecer una conexión fiable entre el cliente y el servidor, y liberar esta conexión de una forma ordenada.
- Ponerse de acuerdo en un nivel común de funcionalidades del protocolo, a través de la negociación de las posibilidades.
- Intercambiar contenido entre el cliente y el servidor utilizando codificación compacta.
- Suspender y recuperar la sesión.

Actualmente, este protocolo ha sido definido únicamente para el caso de la navegación, definiéndose como WSP/B¹². Esta implementación está realizada para el establecimiento de una conexión sobre la base de un protocolo compatible con HTTP1.1.

De esta forma, se han definido un conjunto de primitivas de servicio¹³ para permitir la comunicación entre la capa de sesión integrada dentro del equipo cliente y la capa de sesión integrada en el equipo servidor. Estas primitivas, junto con una pequeña descripción de las mismas, pueden verse en la Tabla 1:

| Nombre de la primitiva | Descripción |
|------------------------|---|
| S-Connect | Esta primitiva se utiliza para iniciar el establecimiento de la conexión, y para la notificación de su éxito. |
| S-Disconnect | Esta primitiva se utiliza para desconectar una sesión, y para notificar al usuario de una sesión que esa sesión no se puede establecer, que ha sido desconectada. |
| S-Suspend | Esta primitiva se utiliza para solicitar la suspensión de la sesión. |
| S-Resume | Esta primitiva se utiliza para solicitar que se recupere la sesión utilizando para las direcciones el nuevo identificador de punto de acceso de servicio. |
| S-Exception | Esta primitiva se utiliza para notificar aquellos eventos que no están asignados a una transacción en particular, ni provocan la desconexión o suspensión de la sesión. |
| S-MethodInvoke | Esta primitiva se utiliza para solicitar una operación que deba ser ejecutada en el servidor. |

| | |
|-----------------|---|
| S-MethodResult | Esta primitiva se utiliza para devolver una respuesta a una petición de operación. |
| S-MethodAbort | Esta primitiva se utiliza para abortar una solicitud de ejecución de operación, que no haya sido aún completada. |
| S-Push | Esta primitiva se utiliza para enviar información no solicitada desde el servidor, dentro del contexto de una sesión de forma y sin confirmación. |
| S-ConfirmedPush | Esta primitiva realiza las mismas funciones que la anterior, pero con confirmación. |
| S-PushAbort | Esta primitiva se utiliza para anular una primitiva anterior del tipo <i>S-Push</i> o <i>S-ConfirmedPush</i> . |

Tabla 1: Primitivas de Servicio de Sesión

Adicionalmente, existen cuatro tipos de cada una de estas primitivas, tal y como puede verse en la Tabla 2:

| Tipo | Abreviación | Descripción |
|------------|-------------|---|
| Request | Req | Se utiliza cuando una capa superior solicita un servicio de la capa inmediatamente inferior. |
| Indication | Ind | Una capa que solicita un servicio utiliza este tipo de primitiva para notificar a la capa inmediatamente superior de las actividades relacionadas con su par, o con el proveedor del servicio |
| Response | Res | Este tipo de primitiva se utiliza para reconocer la recepción de la primitiva de tipo <i>Indication</i> de la capa inmediatamente inferior |
| Confirm | Cnf | La capa que proporciona el servicio requerido utiliza este tipo de primitiva para notificar que la actividad ha sido completada satisfactoriamente. |

Tabla 2: Tipos de Primitivas de Servicio.

Por último, reseñar que cada una de estas primitivas está perfectamente definida dentro de la especificación, tanto desde el punto de vista del diagrama de tiempos en el que se tienen que invocar las primitivas, como desde el punto de vista de los parámetros intercambiados.

2.3 EL PROTOCOLO INALÁMBRICO DE TRANSACCIÓN

El Protocolo Inalámbrico de Transacción corre sobre un servicio de datagrama y conforma una delgada capa orientada a transacción, opera eficientemente sobre redes seguras y no seguras por lo que se establece para proporcionar los servicios necesarios que soporten aplicaciones de “navegación” (del tipo petición/respuesta). Es a este dúo petición/respuesta, lo que vamos a denominar como transacción. Este protocolo se sitúa por encima del Protocolo Inalámbrico de Datagramas y, de forma opcional, de la Capa Inalámbrica de Seguridad de Transporte, que serán estudiados posteriormente.

Las características de este protocolo son

- Proporciona tres clases de servicios de transacción:
 - Clase 0: mensaje de solicitud no seguro, sin mensaje de resultado.
 - Clase 1: mensaje de solicitud seguro, sin mensaje de resultado.
 - Clase 2: mensaje de solicitud seguro, con, exactamente, un mensaje de resultado seguro.
- La seguridad se consigue a través del uso de identificadores únicos de transacción, asentimientos, eliminación de duplicados y retransmisiones.
- Seguridad opcional usuario a usuario.
- De forma opcional, el último asentimiento de la transacción puede contener algún tipo de información adicional relacionada con la transacción, como medidas de prestaciones, etc.
- Se proporcionan mecanismos para minimizar el número de transacciones que se reenvían como resultado de paquetes duplicados.
- Se permiten las transacciones asíncronas.

Al igual que en el protocolo anterior (el protocolo inalámbrico de sesión), en la Tabla 3 vamos a ver las primitivas de servicio¹⁴ que sustentan la comunicación entre dos capas de transacciones situadas en dos equipos distintos:

| Nombre de la primitiva | Descripción |
|------------------------|---|
| TR-Invoke | Esta primitiva se utiliza para iniciar una nueva transacción. |
| TR-Result | Esta primitiva se utiliza para devolver el resultado de transacción iniciada anteriormente. |
| TR-Abort | Esta primitiva se utiliza para abortar una transacción existente. |

Tabla 3: Primitivas de Servicio de Transacción

A modo de ejemplo, vamos a ver en la Figura 7 la concatenación de Primitivas de Servicio de Sesión y de Transacción para el caso de una petición-respuesta:

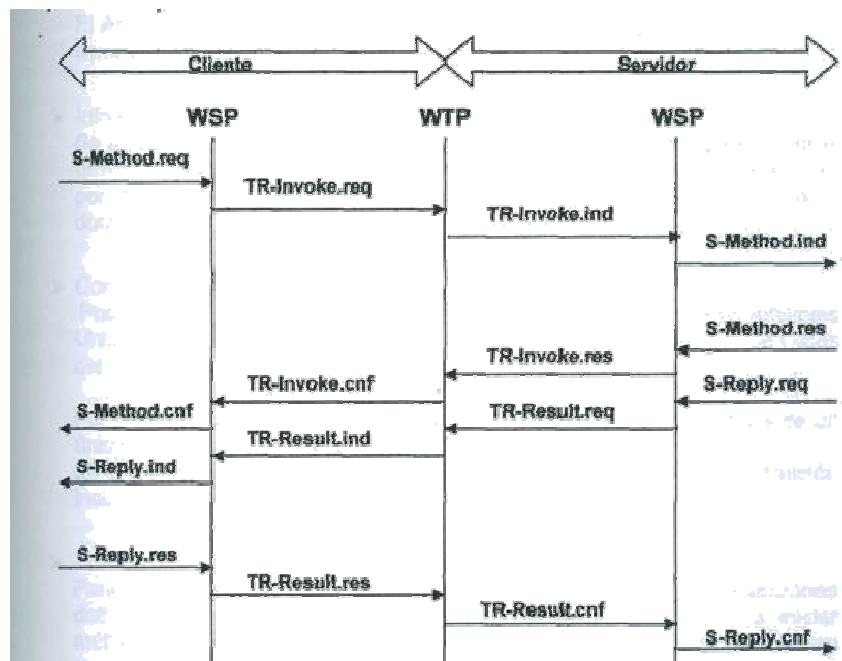


Figura 7: Ejemplo intercambio de primitivas entre capa Sesión y Transacción

Para finalizar, vamos a detallar un poco más las principales características de este protocolo:

➤ **Transferencia de Mensajes.**

Dentro de este protocolo se distinguen dos tipos de mensajes: mensajes de datos y mensajes de control. Los mensajes de datos transportan únicamente datos de usuario, mientras que los mensajes de control se utilizan para los asentimientos, informes de error, etc. pero sin transportar datos de usuario.

➤ **Retransmisión hasta el asentimiento.**

Esta característica se utiliza para la transferencia fiable de datos desde un proveedor WTP a otro, en caso que haya pérdida de paquetes. A modo de comentario, dejar claro que para reducir lo máximo posible el número de paquetes que se transmiten, este protocolo utiliza asentimiento explícito siempre que sea posible.

- Asentimiento de usuario.
El Asentimiento de Usuario permite al usuario de este protocolo, confirmar cada mensaje recibido por el proveedor WTP.
- Información en el Último Asentimiento.
Se permite, así pues, enviar información en el último, y únicamente en el último, asentimiento de una transacción. De esta forma, se puede enviar, por ejemplo, información del rendimiento proporcionado por el sistema durante la transacción realizada, etc.
- Concatenación y Separación.
Podemos definir concatenación como el proceso de transmitir múltiples Unidades de Datos del Protocolo (PDU¹⁵) de WTP en una Unidad de Datos del Servicio (SDU¹⁶) de la red portadora.

Por el contrario, separación es el proceso de separar múltiples PDUs de un único SDU (esto es, el proceso inverso al anterior).
El objetivo de estos sistemas es proveer eficiencia en la transmisión inalámbrica, al requerirse un menor número de transmisiones.
- Transacciones Asíncronas.
Para un correcto funcionamiento del protocolo, múltiples transacciones deben ser procesadas de forma asíncrona, debe ser capaz de iniciar múltiples transacciones antes que reciba la respuesta a la primera transacción.
- Identificador de la Transacción
Cada transacción está identificada de forma única por los pares de direcciones de los sockets (Dirección fuente, puerto fuente, dirección destino y puerto destino) y por el Identificador de Transacción (TID¹⁷), el cual se incrementa para cada una de las transacciones iniciadas. Este número es de 16 bits, utilizándose el bit de mayor orden para indicar la dirección.
- Segmentación y re-ensamblado. (opcional)
Si la longitud del mensaje supera la Unidad Máxima de Transferencia (MTU¹⁸), el mensaje puede ser segmentado por el WTP y enviado en múltiples paquetes. Cuando esta operación se realiza, estos paquetes pueden ser enviados y asentidos en grupos. De esta forma, el emisor puede realizar control de flujo cambiando el tamaño de los grupos de mensajes dependiendo de las características de la red.

2.4 LA CAPA INALÁMBRICA DE SEGURIDAD DE TRANSPORTE

La *Capa Inalámbrica de Seguridad de Transporte (WTLS)*, constituye una capa modular, que depende del nivel de seguridad requerido por una determinada aplicación. Esta capa proporciona a las capas de nivel superior de WAP de una interfaz de servicio de transporte seguro, que lo resguarde de una interfaz de transporte inferior.

El principal objetivo de esta capa es proporcionar privacidad, integridad de datos y autenticación entre dos aplicaciones que se comuniquen. Adicionalmente, la WTLS proporciona una interfaz para el manejo de conexiones seguras.

Al igual que hemos hecho en los protocolos anteriores, en la Tabla 4 vamos a ver las primitivas de servicio¹⁹ que sustentan la comunicación entre dos capas situadas en dos equipos distintos:

| Nombre de la primitiva | Descripción |
|------------------------|---|
| SEC-Unitdata | Esta primitiva se utiliza para intercambiar datos de usuario entre los dos participantes. Sólo puede ser invocada cuando existe previamente una conexión segura entre las direcciones de transporte de los dos participantes. |
| SEC-Create | Esta primitiva se utiliza para iniciar el establecimiento de una conexión segura. |
| SEC-Exchange | Esta primitiva se utiliza en la creación de una conexión segura si el servidor desea utilizar autenticación de clave pública o intercambio de claves con el cliente. |
| SEC-Commit | Esta primitiva se inicia cuando el <i>handshake</i> ²⁰ se completa y cualquiera de los equipos participantes solicita cambiar a un nuevo estado de conexión negociado. |
| SEC-Terminate | Esta primitiva se utiliza para finalizar la conexión. |
| SEC-Exception | Esta primitiva se utiliza para informar al otro extremo sobre las alertas de nivel de aviso. |
| SEC-Create-Request | Esta primitiva se utiliza por el servidor para solicitar al cliente que inicie un nuevo <i>handshake</i> . |

Tabla 4: Primitivas de Servicio de Capa de Seguridad

Hemos hablado anteriormente del proceso de establecimiento de una sesión segura o *handshake*. En la Figura 8 podemos ver este intercambio de primitivas:

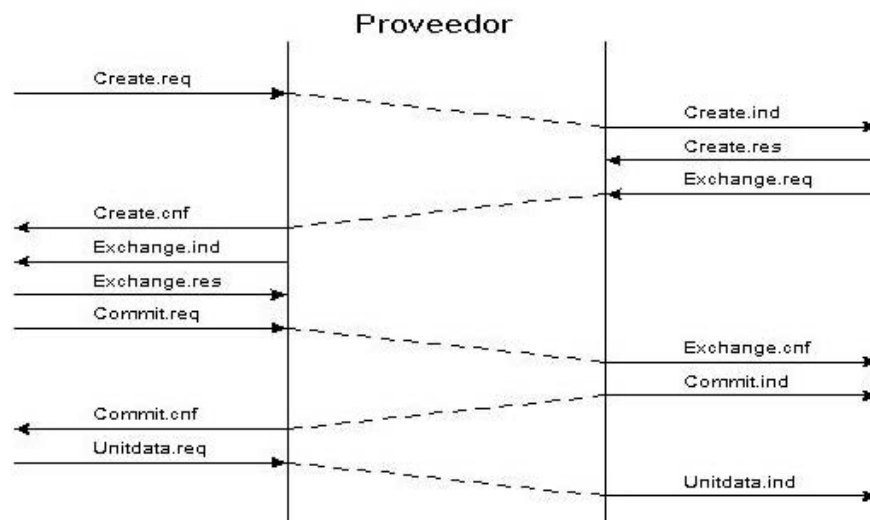


Figura 8: Secuencia de Primitivas para el establecimiento de una sesión segura

Adicionalmente, el WTLS puede ser utilizado para la realización de comunicación segura entre terminales. Las aplicaciones pueden habilitar o deshabilitar WTLS dependiendo de los requerimientos de seguridad y características de la red, por ejemplo en el caso de operaciones de comercio electrónico entre terminales móviles.

Aspectos de Seguridad en WAP

La pila de protocolos WAP tiene una capa "opcional" denominada WTSL que adapta el trabajo de TSL al entorno inalámbrico y está presente por encima de la capa WDP para proveer mecanismos para cifrar y descifrar transparentemente los mensajes.

WTSL usa RSA (basada en certificados y anónimos), Diffie - Hellman (DH), ECDH anónimo y ECDH-ECDSA como sus principales algoritmos de intercambio de claves. Una pasarela necesita soportar solo uno de estos de acuerdo con la especificación, pero entre más soporte, mejor. También hace uso de RC5, DES, 3DES, e IDEA como algoritmos fuertes de cifrado y MD5 y SHA - 1 para autenticación de mensajes o algoritmo MAC.

Una sesión WTLS puede intercambiar hasta 2^{16} mensajes antes que la sesión sea cerrada y hace uso del concepto de refresco de llave - durante el proceso de negociación, el cliente y la pasarela deciden una rata de refresco - que indica cada cuantos mensajes se cambiara la clave usada para cifrar o descifrar, esto hace a WTLS menos susceptible a ataques.

La Figura 9 ilustra el modelo de seguridad de WAP, donde se aprecia que para ofrecer seguridad extremo a extremo se hace necesaria una funcionalidad en la pasarela que realice el traslado de WTLS a SSL. La necesidad de este traspaso obedece a la adaptación que sufrió el protocolo SSL para poder ser usado en el ambiente móvil minimizando overhead del protocolo SSL, algoritmos rápidos y permitiendo una comprensión de datos mayor que en las soluciones SSL tradicionales, esta optimización es la que permite que dispositivos portables, de bajo consumo y potencia se puedan comunicar seguramente sobre Internet.

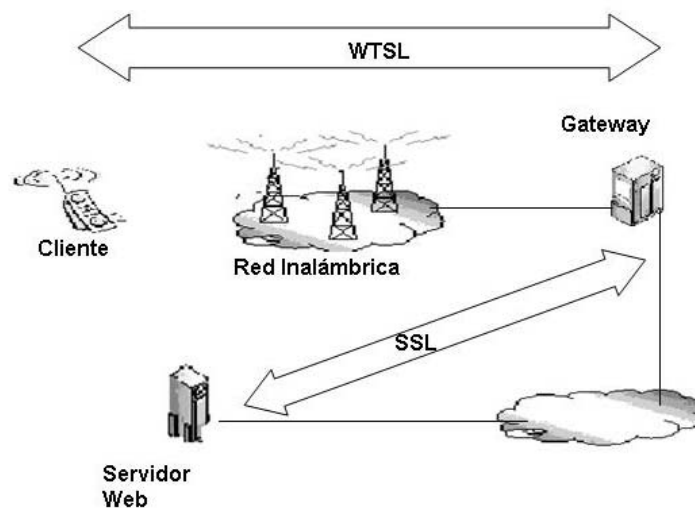


Figura 9: Modelo de Seguridad en WAP

Este mecanismo supone un "*problema en la pasarela*" donde para realizar el traslado de un protocolo a otro, los datos deben ser descifrados en memoria haciendo susceptible que éstos sean recuperados por terceros; para evitarlo, se deben seguir ciertas normas de seguridad en las gateways tanto a nivel físico como de acceso por terceros que no sean los administradores.

Los desarrolladores de gateways también deben garantizar que los algoritmos sean lo suficientemente rápidos, que los datos descifrados nunca se almacenan en otro medio diferente a la memoria y sólo durante el proceso de traslado.

La versión 1.3 de WAP que se espera que muy pronto salga a la luz por el WAP Forum supone resolver este problema, pues muchas compañías preferirían tener sus propias pasarelas sabiendo que sus datos son susceptibles de ser sacados en un punto intermedio de la comunicación.

Otro problema que se presenta actualmente es que muchas pasarelas no ofrecen un soporte de traslado WTLS/SSL permitiendo que a pesar de que del lado del servidor se ofrezca seguridad, esto no se haga del lado del cliente.

Por otro lado, este modelo exige que exista una fuerte relación entre el operador de la red y los proveedores de contenido para implementar la mejor seguridad posible

2.5 EL PROTOCOLO INALÁMBRICO DE DATAGRAMAS

El *Protocolo Inalámbrico de Datagramas* (en adelante WDP¹⁰) ofrece un servicio consistente al protocolo (Seguridad, Transacción y Sesión) de la capa superior de WAP, comunicándose de forma transparente sobre uno de los servicios portadores disponibles.

Este protocolo ofrece servicios a los protocolos superiores del estilo a direccionamiento por número de puerto, segmentación y re-ensamblado opcional y detección de errores opcional, de forma que se permite a las aplicaciones de usuario funcionar de forma transparente sobre distintos servicios portadores disponibles. Para ello, el autor plantea una arquitectura de protocolo como el que se muestra en la Figura 10:

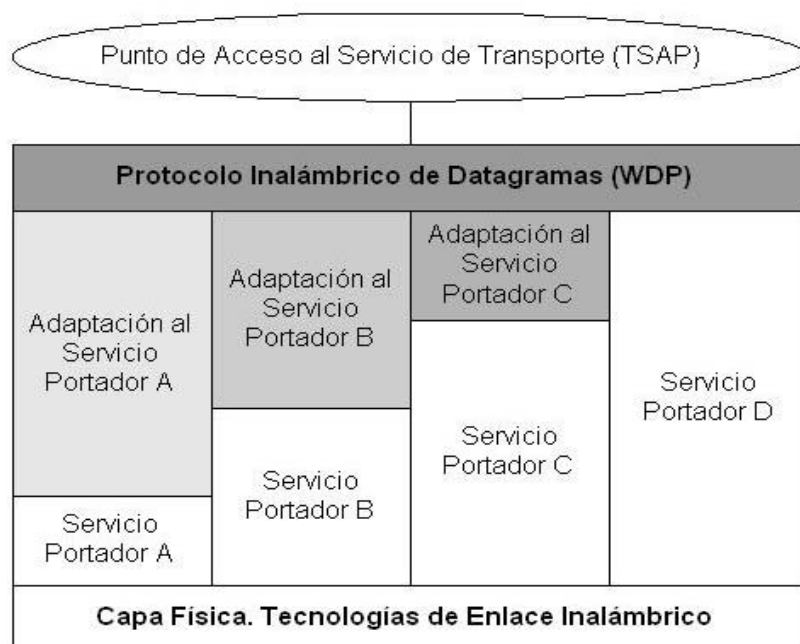


Figura 10: Arquitectura del Protocolo Inalámbrico de Datagramas

Al igual que hemos hecho en los protocolos anteriores, en la Tabla 5 vamos a ver las primitivas de servicio²¹ que se utilizan en este protocolo:

| Nombre de la primitiva | Descripción |
|------------------------|---|
| T-Dunitdata | Esta primitiva es la utilizada para transmitir datos como datagramas. No requiere que exista una conexión para establecerse. |
| T-Derror | Esta primitiva se utiliza para proporcionar información a la capa superior cuando ocurre un error que pueda influenciar en el servicio requerido. |

Tabla 5: Primitivas de Servicio de la Capa de Datagramas

Por último, revisaremos la arquitectura de este protocolo dentro de la arquitectura global de WAP, para el caso de utilizarse GSM como servicio portador, que es el protocolo que más nos puede interesar por su amplia implantación en los sistemas de comunicaciones móviles telefónicas existentes hoy en día.

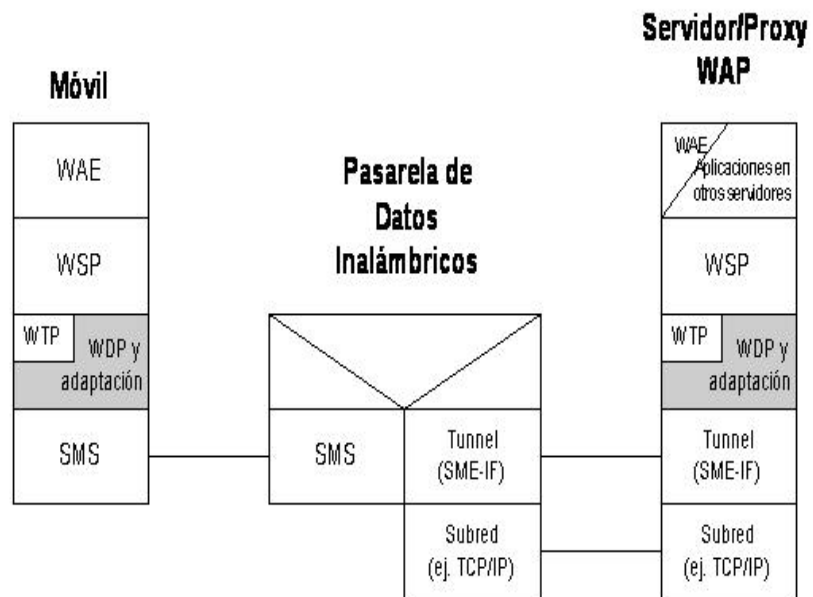


Figura 11: WDP sobre GSM SMS

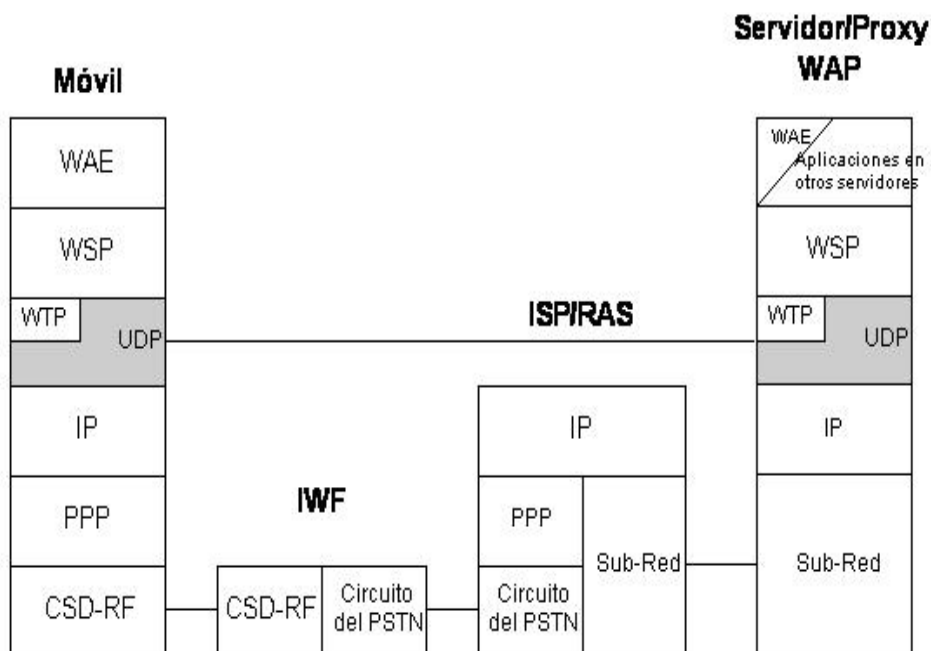


Figura 12: WDP sobre GSM Canal de Datos de Circuitos Conmutados

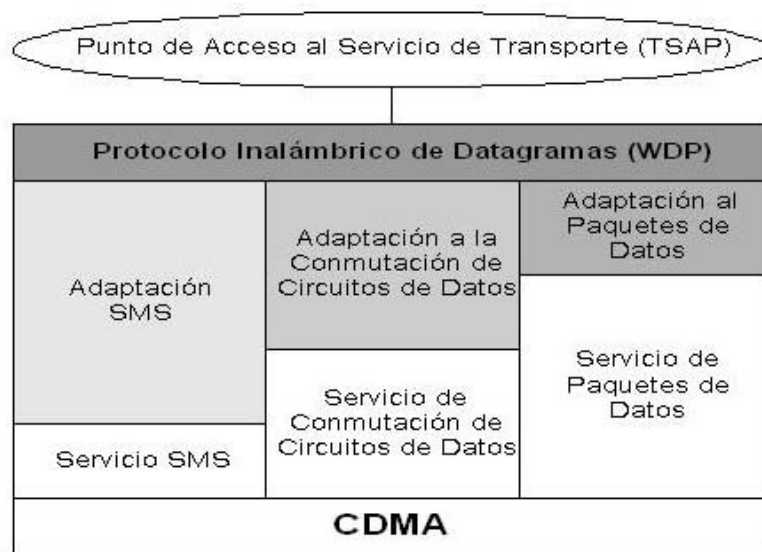


Figura 13: WDP sobre Servicios Portadores CDMA

CAPITULO III

LENGUAJES

OBJETIVO:

- Describir los lenguajes que se utilizan para las aplicaciones inalámbricas.

3.1 Wml

3.2 WmlScript

3.1 WML

WML (Wireless Markup Language - Lenguaje de Marcación Inalámbrico) es un lenguaje de marcado como lo es HTML (Lenguaje de Marcación de Hipertexto) pero a diferencia de éste, WML es desde su nacimiento un lenguaje conforme XML 1.0 (Lenguaje Extensible de Marcación) definido por la [W3](#) y por ese motivo, se ajusta a éste estándar.

Al igual que todo WAP, WML está diseñado para ser usado en dispositivos con ciertas restricciones, como lo son:

- Despliegues pequeños
- Ancho de banda limitada
- Poca capacidad de memoria
- Poca capacidad de procesamiento

Las características principales ofrecidas por WML son:

- Soporte para texto e imágenes
- Los documentos WML está conformado por una estructura denominada DECK (Conjunto de cartas - En nuestra monografía haremos uso del término en Inglés), y cada DECK agrupa un conjunto de elementos denominados Card (Carta) como se muestra en la Figura. Cada Card agrupa un conjunto de elementos WML que permiten generar el contenido visual del usuario.
- Cada Card representa una pantalla de visualización para el usuario, para ver los otros Card que conforman un documento, WML ofrece un mecanismo de navegación entre Cards.
- WML ofrece además un mecanismo de navegación similar a HTML.
- Ofrece un mecanismo de manejo de eventos del dispositivo que permite realizar operaciones de navegación o ejecución de scripts.
- Ofrece un mecanismo de manejo de variables que permiten la parametrización de un documento WML mediante un modelo de estado.

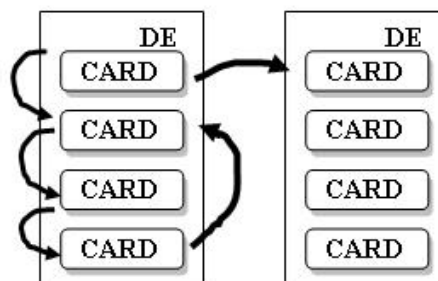


Figura14: Documento WML y Esquema de navegación

El contenido WML puede ser elaborado con cualquier editor de texto, pero existen algunos editores especiales para WML que permiten desarrollar las páginas WML. Adicionalmente, puede ser generado a partir de la ejecución de un programa en el servidor web a través de mecanismos como CGI, Servlet, ASP, PHP, etc.

Las páginas hechas en WML tienen extensión *.WML*, los Scripts *.WMLS* y los dibujos *.WBMP* además, tienen un prólogo obligatorio en los documentos y es el siguiente:

```
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"
"http://www.wapforum.org/DTD/wml_1.1.xml">
```

El código del documento se encuentra dentro de las etiquetas `<wml>` y `</wml>` y para los Card tenemos la etiqueta `<card></card>` donde se coloca lo que se va a desplegar al usuario, de tal forma que si se hace un esqueleto de un documento sería así:

```
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"
"http://www.wapforum.org/DTD/wml_1.1.xml">
<wml>      <!-- Inicio del documento WML -->
  <card id="primera" > <!-- Inicio del primer Card-->
  -----
  </card> <!-- Fin del primer Card-->
  -----

  <card id="segunda">
  </card>
</wml>      <!-- Fin del documento WML-->
```

3.2 WMLSCRIPT

Introducción

WMLScript es parte de la capa de aplicación del protocolo WAP y esta basado en ECMAScript; modificado para soportar anchos de banda pequeños. El lenguaje es compilado en bytecode y es procedural por naturaleza; además soporta librerías locales instaladas. El WMLScript(Wireless Markup Language Script) y sus librerías estándares son documentadas a continuación.

3.2.1 ByteCode

WMLScript es compilado en binario antes de ser procesado por el intérprete del bytecode. El compilador de WMLScript toma una unidad de compilación como entrada y genera bytecode como salida. Las funciones pueden ser llamadas en un bytecode codificado. Una llamada a una función WMLScript debe especificar la función, los argumentos y la unidad de compilación a la que pertenece la función.

3.2.2 Pragmas

Los pragmas especifican información en el ámbito de la unidad de compilación. Los pragmas se detallan al comienzo de los archivos de WMLScript, antes de cualquier declaración de función. Todos los pragmas empiezan con una palabra clave seguidos de sus atributos.

Archivos externos

El pragma url se utiliza para llamar una función en un archivo externo.

```
url OtherScript "http://www.host.com/app/script";  
  
function test (par1, par2)  
{  
    return OtherScript#check (par1-par2);  
}
```

El comportamiento de la función llamada en el ejemplo demuestra los siguientes principios:

- El pragma especifica el URL al archivo WMLScript
- La función llamada carga el archivo utilizando el URL especificado
- La función verifica el contenido del archivo y ejecuta la función especifica(check)

El pragma url tiene su propio espacio para sus nombres locales. Sin embargo, los nombres locales deben ser únicos en un archivo dado (unidad de compilación) WMLScript soporta URLs y URLs relativos.

Acces Control

El pragma acces control puede ser usado para proteger los contenidos de los archivos. Se deben llamar los pragma acces control antes de llamar a las funciones externas.

Nota: Usar más de un pragma acces control en un archivo genera errores de compilación.

Cada vez que una función externa es llamada, el compilador realiza un chequeo basado en este pragma para determinar si el archivo de destino permite acceso al que realiza el llamado. El pragma acces control especifica los atributos de dominio y path con los cuales el compilador realiza el chequeo para controlar el acceso. Si un archivo tiene un dominio o un path, el URL que referencia el archivo debe coincidir con los valores de estos atributos.

El compilador hace la comparación como sigue:

- El sufijo de dominio debe coincidir con el elemento del nombre de dominio de la URL referida.
- El prefijo Path debe coincidir con el elemento del nombre de path de la URL referida.

Para simplificar el desarrollo de aplicaciones que pueden no conocer el path absoluto al archivo actual, el atributo path acepta URL relativos.

Ejemplo: Los elementos acces control para un archivo son:

```
use access domain "wapforum.org" path "/finance";
```

Las siguientes URLs pueden llamar a las funciones externas especificadas en los archivos:

```
http://wapforum.org/finance/money.cgi  
http://www.wapforum.org/finance/markets.cgi  
http://www.wapforum.org/finance/demos/packages.cgi?x+123&y+456
```

Sin embargo, las siguientes URLs no pueden llamar a las funciones que se encuentren en:

http://www.test.net/finance
http://www.wapforum.org/internal/foo.wml

Por defecto, el control de acceso esta deshabilitado.

Meta Information

El pragma meta-information especifica las propiedades nombre y contenido del archivo. También puede especificar el esquema del archivo, que especifica una forma o estructura que puede ser usada para interpretar los valores de las propiedades dependiendo del tipo de meta-data. El atributo valor es un literal de cadena. Los Meta-pragma no definen ninguna propiedad, ni tampoco definen como el agente de usuario debe interpretar los meta-datos. Los agentes de usuario no requieren actuar sobre los meta-data. Los Meta-pragma pueden tener los siguientes atributos: Name, HTTP Equiv, User Agent

Name

El pragma name especifica la meta-information usada por los servidores de origen. El agente de usuario puede ignorar cualquier meta dato nombrado mediante este atributo. Los servidores de red pueden no emitir contenido que lleve meta-pragma name.

use meta name "Created" "18-June-1999";

HTTP Equiv

El pragma http equiv especifica meta-information indicando que la propiedad puede ser interpretada como cabecera HTTP. Los meta datos nombrados con este atributo pueden ser convertidos al protocolo WSP o cabeceras de respuesta HTTP si el archivo es compilado antes de llegar al agente de usuario.

use meta http equiv "Keywords" "Script, Language";

User Agent

El pragma user agent especifica la meta-information para los agentes de usuario. Estos meta datos deben ser entregados al agente de usuario; no deben ser removidos por algún intermediario de red.

use meta user agent "Type" "Test";

3.2.3 Estructura

WMLScript es sensible a mayúsculas y minúsculas e ignora los espacios, tabs, y saltos de línea; a menos que se usen en una cadena.

WMLScript define dos estructuras de comentarios: // ... //, /* */

3.2.4 Variables Literales y Tipos

Las variables tienen que ser declaradas para que puedan usarse. Son soportadas solo dentro de las funciones o como parámetros de estas.

Ejemplos:

```
var sum1 = x;
```

```
var sum2 = y;
```

```
var result = sum1 + sum2;
```

Literales:

Valores codificados en el script:

Boolean: --> Verdadero/Falso

Integer: --> Decimal (0,1,2,3,) --> Hex (0xB3) --> Oct (023)

String: --> Cualquier secuencia de caracteres encerrados en comillas dobles o simples

Floating Point: --> Lugares decimales (1.2, 0.02, 1.3E1)

Tipos:

Define el tipo de datos que una variable puede contener:

Boolean: --> var name = truth;

Integer: --> Lang.maxInt (); //Entero Máximo, Lang.minInt (); /Entero Mínimo

String: --> var msg = "Hola";

Floating Point: --> Float.maxFloat (); //Punto Flotante Máximo, Float.minFloat (); //Punto Flotante Mínimo

3.2.5 Operadores y Expresiones

Operadores de Asignación

| | |
|-----------|---|
| = | Asignar |
| += | Sumar (números) / concatenar (cadenas) y asignar |
| -= | Restar y asignar |
| *= | Multiplicar y asignar |
| /= | Dividir y asignar |
| Div= | Dividir (división de enteros) y asignar |
| %= | Residuo o módulo (el signo del resultado es igual al signo del dividendo) y asignar |
| <<= | Corrimiento a la derecha y asignar |
| >>= | Corrimiento a la izquierda y asignar |
| >>>= = | Corrimiento a la derecha archivo cero y asignar |
| &= | AND y asignar |
| ^= | XOR y asignar |
| = | OR and assign |

Operadores de Aritméticos

| | |
|-----|--|
| + | Sumar (números) /concatenar (cadenas) |
| - | Restar |
| * | Multiplicar |
| / | Dividir |
| Div | División de enteros |
| % | Residuo o módulo, el signo del resultado es igual al signo del dividendo |
| << | Corrimiento a la derecha |
| >> | Corrimiento a la izquierda |
| >>> | Corrimiento a la derecha llenado con cero |
| & | AND |
| | OR |
| ^ | XOR |
| -- | Decremento |
| ++ | Incremento |
| ~ | NOT |

Operadores Lógicos

| | |
|----|--------------|
| && | Y ó AND |
| | O ó OR |
| ! | NOT (unario) |

El operador lógico AND evalúa el primer operando y prueba el resultado. Si el resultado es falso o invalido, el resultado de la operación es falso o invalida, y el segundo operador no es evaluado.

Si el primer operando es TRUE, el resultado de la operación es el resultado del segundo operando. El operador lógico OR evalúa el primer operando y prueba el resultado.

Si el resultado es verdadero o invalido, el segundo operador no es evaluado. Si el primer operando es falso, el resultado de la operación es el resultado del segundo operando.

```
weAgree = (iAmRight && youAreRight) || (!iAmRight && !youAreRight);
```

WMLScript requiere un valor booleano para las operaciones lógicas y soporta conversión automática de otros tipos a tipos booleanos.

Nota: Si el valor del primer operando para el operador AND u OR es invalido, el segundo operando no es evaluado, y el resultado es invalido.

```
var a = (1/0) || foo (); // resultado: invalido, no llama a foo ()
var b = true || (1/0); // verdadero
var c = false || (1/0); // invalido
```

Operadores de Cadena

Los operadores + y += concatenan cadenas. Otros operadores de cadena son soportados por la librería estándar String.

```
var str = "Beginning" + "End";
var chr = String.charAt (str,10) // chr = "E"
```


Operadores de Comparación

| | |
|----|-------------------|
| < | Menor que |
| <= | Menor que o igual |
| == | Igual |
| >= | Mayor que o igual |
| > | Mayor que |
| != | Diferente |

Los operadores de comparación siguen las siguientes reglas:

Boolean: true es mayor que false.

Integer: La comparación se basa en los valores enteros dados.

Floating-point: La comparación se basa en los valores de punto flotante dados.

String: La comparación se basa en el orden de los códigos de carácter de las cadenas dadas. Los códigos de carácter son definidos por el conjunto de caracteres soportados por el interprete de WMLScript.

Invalid: Si al menos uno de los operandos es invalido, el resultado es invalido.

Ejemplo:

```
var res = (myAmount > yourAmount);

var val = (( 1/0) == invalid); // val = invalid
```

Operadores de Arreglos

WMLScript no soporta arreglos convencionales. Sin embargo, la librería estándar String soporta funciones en las que el comportamiento de un arreglo puede ser implementado utilizando cadenas.

Ejemplo:

```
function dummy() { var str = "Lets learn WMLScript";

var word = String.elementAt (str,4,""); }
```

Operadores Coma

Los operadores coma combinan múltiples evaluaciones en una expresión. El resultado del operador coma es el valor del segundo operador.

Ejemplo:

```
for (a=1, b=100; a < 10; a++,b++) { //... do something ... }
```

Las comas usadas en una llamada a función para separar parámetros o en una declaración de variables para separar múltiples declaraciones de variables no son operadores coma. En estos casos, el operador coma debe ir entre paréntesis.

Ejemplo:

```
var a = 2;
```

```
var b = 3, c = 3;
```

```
myFunction("Name", 3*(b* a, c)); // Two parameters: "Name", 9
```

Operadores Condicionales

Los operadores condicionales (?:) asignan un valor a una expresión, basada en el resultado booleano de una sentencia inicial. Se encuentra esencialmente en sentencias if-then-else que toman tres operandos. Si el valor del primer operando es true entonces el resultado es el resultado del operando 2. Si la condición es falsa o inválida, entonces el resultado es el valor o resultado del operador 3.

Ejemplo:

```
myResult = flag ? "Off" : "On (value=" + level + ")";
```

Operador typeof

El operador typeof regresa un valor entero que describe el tipo de una expresión dada. El operador typeof no convierte el resultado de un tipo a otro.

| Code | Type |
|------|----------------|
| 0 | Entero |
| 1 | Punto Flotante |
| 2 | Cadena |
| 3 | Booleano |
| 4 | Inválido |

Ejemplo:

```
var str = "123";  
  
var mytype = typeof str; // myType = 2
```

Operador isvalid

Evalúa la validez de una variable o de una expresión. Devuelve un valor booleano de falso si el resultado de la expresión es invalido o si la variable no tiene un tipo valido. Si el resultado de la expresión no es invalido o si la variable es de un tipo valido, isvalid devuelve true.

Ejemplo:

```
var str = "123"; var ok = isvalid str;  
  
// true var tst = isvalid (1/0); // false
```

Expresiones

Es una combinación de operadores, constantes, valores literales, funciones, o nombres de variables que se combinan para realizar una operación específica del programa. Las expresiones son el valor de una constante o una variable.

```
567  
66.77  
"This is too simple"  
`This works too`  
  
true  
myAccount
```

Se pueden definir expresiones más complejas usando expresiones simples con operadores, y llamadas a funciones.

```
myAccount + 3; (a + b) / 3; initialValue + nextValue(myValues);
```

3.2.6 Funciones

Declaración de funciones

La declaración de funciones incluye un nombre de función, con los parámetros opcionales, y un bloque que es ejecutado cuando la función es llamada.

Todas las funciones tienen las siguientes características:

- La declaración de funciones no puede ser anidada.
- Los nombres de funciones deben ser únicos en cada archivo de script.
- Todos los parámetros de la función son pasados por valor.
- Las llamadas a las funciones deben pasar exactamente el mismo número de argumentos a la función llamada, como se especifico en su declaración.

Una función siempre retorna un valor. Por defecto es una cadena vacía (""). Se usa `return` para especificar otros valores de retorno. Se pueden usar palabras claves externas para especificar que la función esta disponible a una facilidad externa.

Ejemplo:

```
function currencyConverter(currency, exchangeRate)
{
  return currency*exchangeRate;
}
extern function testIt()
{
  var USD = 10;
  var FIM = currencyConverter (USD, 5.3);
}
```

Llamadas a Funciones Locales

Son definidas y llamadas en el mismo archivo. Pueden ser llamadas con el nombre de la función y una lista de argumentos separados por comas. El numero de argumentos debe coincidir con el numero de parámetros aceptados por la función. Una función local no puede ser llamada por otro archivo pero puede ser llamada antes de su declaración.

Ejemplo:

```
function test2 (param)
{
  return test1 (param+1);
}
```

```
function test1 (val)
{
  return val*val;
}
```

Funciones Externas

Son accesibles a otros archivos además de aquel en que la función externa fue declarada. Las funciones externas pueden aceptar parámetros y variables cuando son llamadas por otros archivos. La palabra clave externa es usada antes del nombre de la función para declarar la función externa. EL pragma url es usado para especificar el nombre y localización de la unidad de compilación externa. El mapea el archivo de script externo al nombre que puede ser usado en la declaración de la función. Este nombre y el símbolo numero(#) son usados como prefijo de la sintaxis estándar de la llamada a la función.

Ejemplo:

```
use url OtherScript "http://www.WapColombia.com/script"
```

```
function test3(param)
{
  return OtherScript#test2(param+1);
}
```

Funciones de Librería

Una llamada a una función de librería debe usarse si la función llamada es una función de la librería estándar de WMLScript. Se puede llamar a una función de librería antecediendo el nombre de la librería y el operador punto(.) al nombre de la función.

Ejemplo:

```
function test4(param)
{
    return Float.sqrt(Lang.abs(param)+1);
}
```

3.2.7 Sentencias

Sentencias vacías

Usadas cuando una sentencia es requerida pero no se requiere que se realice una operación.

```
while (!poll(device)); // Wait until poll() is true
```

Sentencias de Expresiones

Las sentencias de expresión son usadas para asignar valores a variables, calcular expresiones matemáticas, hacer llamadas a funciones, etc.

```
str = "Hello" + yourName;
val3 = prevVal + 7;
counter++;
myValue1 = counter, myValue 2 = val13;
alert("Error");
```

Sentencias de Bloque

Una colección de sentencias que están entre corchetes ({}), que son tratadas como una sentencia simple o sentencias compuestas.

```
{
    var i = 0;
    var x = Lang.abs(b);
    popUp("Remember!");
}
```

Sentencias If

Define una condición y sentencias que son ejecutadas dependiendo del valor booleano de la condición. Las sentencias pueden ser cualquier expresión de WMLScript, incluyendo otras sentencias If anidadas.

```
if (sunShines)
{
myDay = "Good" goodDays++;
} else
myDay = "Oh well...";
```

Sentencias de variables

Declaran variables con un valor inicial. El alcance de la variable declarada es el resto de la función actual; la expresión es evaluada cada vez que la sentencia de la variable es ejecutada. Las variables pueden ser inicializadas con un valor específico o por defecto con una cadena vacía.

```
function count(str)
{
var result = 0; // Initialized once
while(str != "") { var ind = 0; // Initialized every time };
return result
};
function example(param)
{
var a = 0;
if (param > a)
{
var b = a+1; // Variables a and b can be used
}
else
{
var c = a+2; // Variables a, b, and c can be used
}
return a; // Variables a, b, and c are accessible };
```

Sentencias While

Crea un bucle para evaluar una expresión: Si la expresión es true ejecuta la sentencia, sino se repite hasta que la condición sea verdadera.

```
var counter = 0;

var total = 0;

while (counter < 3) { counter++; total += c; }
```

Sentencias For

Crea un lazo que se ejecuta mientras que la condición declarada sea verdadera. Esta constituida por tres expresiones opcionales entre paréntesis y separadas por punto y coma seguidas por el código que la sentencia for ejecuta. La primera expresión se utiliza para iniciar una variable contadora. Esta expresión puede opcionalmente declarar una nueva variable con la palabra clave var. El alcance de la variable declarada es el resto de la función. La segunda expresión puede ser cualquier expresión de WMLScript para que evalúe un valor booleano valido o invalido. Esta condición se evalúa en cada paso a través del bucle. Si la condición es valida, la sentencia es ejecutada. Este test condicional es opcional. Si se omite, la condición siempre se evalúa a true y las sentencias en el bucle for son siempre ejecutadas. La tercera expresión es generalmente usada para actualizar o incrementar la variable contador. Esta sentencia es ejecutada mientras que la condición es verdadera.

```
for (var index = 0; index < 100; index++) { count += index; myFunc(count); }
```

Sentencias Break

Terminan el actual bucle for o while y continua la ejecución del programa en la sentencia siguiente al bucle terminado.

Nota: Si se usa afuera de un bucle while o for se genera un error.

```
function testBreak(x)
{
  var index = 0;
  while (index < 6) { if (index == 3) break; index ++ }
  return index*x;
}
```


Sentencia Continue

Termina la ejecución de un bloque de sentencias en un bucle for o while y continua la ejecución del bucle en la próxima iteración.

Nota: La sentencia continue no termina la ejecución del bucle.

- En el bucle while, salta la condición
- En el bucle for salta la expresión de actualización.

Nota: Si se usa fuera de un bucle while o for genera un error

```
var index = 0;
var count = 0;
while (index < 5)
{
    index++;
    if (index == 3) continue;
    count += index;
}
```

Sentencia Return

Se usa la sentencia return dentro del cuerpo de una función para especificar el valor devuelto por esta. Si no se especifica la función devuelve una cadena vacía.

```
function square (x)
{
    if (!(Lang.isFloat(x))) return invalid;
    return x * x;
}
```

3.2.8 Librerías

Librería Lang

La librería Lang contiene un conjunto de funciones que están estrechamente relacionadas con el núcleo de WMLScript1.1.

| | | | | |
|--------------|---------|--------|------------|----------|
| Abort | exit | IsInt | Min | ParseInt |
| Abs | float | max | MinInt | Random |
| CharacterSet | IsFloat | MaxInt | parseFloat | Seed |

abort

Función: abort(errorDescription)

Descripción: Abandona la interpretación del bytecode WMLScript 1.1 y devuelve el control al interprete devolviendo errorDescription. Esta función puede ser usada para realizar una salida anormal en casos donde la ejecución de WMLScript1.1 debe ser descontinuada porque errores serios son detectados por la función llamada. Si el tipo de errorDescription es invalido, la cadena "invalid" es usada instantáneamente.

Parámetros: errorDescription=String

Valor devuelto: ninguno(esta función abandona la interpretación).

Ejemplo:

```
Lang.abort("Error: " + errVal); //Error value is a string
```

abs

Función: abs(valor)

Descripción: Devuelve el valor absoluto de un numero dado. Si el numero dado es de tipo entero, un entero es devuelto. Si el numero es de punto flotante, es devuelto un numero de punto flotante.

Parámetros: valor=numero

Valor devuelto: numero o invalido

Ejemplo:

```
var a = -3;
```

```
var b = Lang.abs(a); //b = 3
```

characterSet**Función:** lang.characterSet()**Descripción:** Retorna el conjunto de caracteres soportados por el interprete WMLScript. el valor retornado es un entero que denota un valor MIBEnum asignado por la IANA(Internet Assigned Numbers Authority, para todos los conjuntos de caracteres.**Valor devuelto:** Entero**Ejemplo:**

```
var charset = Lang.characterSet(); // charset = 4 for latin1
```

exit**Función:** Exit(valor)**Descripción:** Termina la interpretación del bytecode WMLScript1.1 y devuelve el control al interprete que realizo la llamada con un valor de retorno dado. Esta función puede ser usada para realizar una salida normal de una función en casos donde la ejecución del bytecode WMLScript1.1 debe ser suspendido.**Parámetros:** ninguno**Ejemplos:**

```
Lang.exit("Value: " + myVal); // Returns a string Lang.
```

```
exit(invalid); // Returns invalid
```

float**Función:** float()**Descripción:** Devuelve true si los puntos flotantes son soportados, y false sino.**Valor devuelto:** Booleano**Ejemplo:**

```
var floatsSupported = Lang.float();
```

isFloat**Función:** isFloat(valor)**Descripción:** Devuelve un valor booleano que es verdadero si el valor dado puede ser convertido a un numero de punto flotante usando parseFloat(value). De lo contrario devuelve false.**Parámetros:** valor = cualquiera**Valor devuelto:** Booleano o invalido**Excepciones:** Si el sistema no soporta punto flotante un valor invalid es devuelto.

Ejemplos:

```
var a = Lang.isFloat (" -123"); // true
var b = Lang.isFloat (" 123.33"); // true
var c = Lang.isFloat ("string"); // false
var d = Lang.isFloat ("#123.33"); // false
var e = Lang.isFloat (invalid); // invalid
```

isInt**Función:** isInt(value)**Descripción:** Devuelve un valor booleano que es true si el valor dado puede convertirse a entero utilizando parseInt(value). De lo contrario devuelve false.**Parámetros:** valor = cualquiera**Valor devuelto:** Booleano o invalido**Ejemplos:**

```
var a = Lang.isInt(" -123"); // true
var b = Lang.isInt(" 123.33"); // true
var c = Lang.isInt("string"); // false
var d = Lang.isInt("#123"); // false
var e = Lang.isInt(invalid); // invalid
```

max**Función:** max(value1,value2)**Descripción:** devuelve el valor máximo de dos números dados. El valor y tipo devuelto son los mismos que el valor y el tipo del numero seleccionado. La selección se hace de la siguiente forma: El operador WMLScript1.1 de las reglas de conversión de tipos de datos para enteros y punto flotante debe ser usado para especificar el tipo de datos para la comparación(entero o punto flotante) Compara los números para seleccionar el mayor Si los valores son iguales el primer valor es seleccionado.**Parámetros:** Value1=number Value2=number**Valor devuelto:** Numero o invalido**Ejemplos:**

```
var a = -3;
var b = Lang.abs(a);
var c = Lang.max(a, b); // c = -3
var d = Lang.max(45.5, 76); // d = 76(integer)
var e = Lang.max(45.0, 45); // e = 45.0
```

maxInt**Función:** maxInt()**Descripción:** Devuelve el máximo valor entero.**Valor devuelto:** Entero 2147483647.**Ejemplo:**

var a = Lang.maxInt();

min**Función:** min(value1, value2)**Descripción:** Devuelve el valor mínimo entre dos números dados. El valor y el tipo de datos devuelto es igual al tipo y valor del número seleccionado. La selección sigue reglas similares a las de la función max(valor1, valor2)**Parámetros:** Value1=number Value2=number**Valor devuelto:** Número o inválido**Ejemplos:**

var a = -3;

var b = Lang.abs(a);

var c = Lang.min(a, b); // c = -3

var d = Lang.min(45, 76.3); // d = 45 (integer)

var e = Lang.min(45, 45.0); // e = 45 (integer)

minInt**Función:** minInt()**Descripción:** devuelve el mínimo valor de un entero.**Valor devuelto:** Entero -2147483648.**Ejemplo:**

var a = Lang.minInt();

parseFloat**Función:** parseFloat(valor)**Descripción:** Devuelve un valor de punto flotante definido por el valor de la cadena. La sintaxis legal para punto flotante es definida por la gramática para cadenas numéricas de literales de punto flotante de WMLScript1.1 con las siguientes reglas:

- El proceso termina cuando el primer carácter no puede ser traducido como el comienzo de una representación de punto flotante.
- El resultado es la cadena convertida a un valor de punto flotante Parámetros: valor = string

Valor devuelto: Punto flotante o invalido

Excepciones: En caso de error, un valor invalid es devuelto Si el sistema no soporta operaciones de punto flotante un valor invalid es devuelto.

Ejemplos:

```
var a = Lang.parseFloat("123.4");      // a = 123.4
var b = Lang.parseFloat(" +7.34e2 Hz") // b = 7.34e2
var c = Lang.parseFloat(" 70e-2 F");   // c = 70.0e-2
var d = Lang.parseFloat("-.1 C");      // d = -0.1
var e = Lang.parseFloat(" 100 ");      // e = 100.0
var f = Lang.parseFloat("Number: 5.5"); // f = invalid
var g = Lang.parseFloat("7.3e meters"); // g = invalid
var h = Lang.parseFloat("7.3e- m/s");  // h = invalid
```

parseInt

Función: Lang.parseInt(valor)

Descripción: Devuelve un valor entero definido por el valor de cadena. La sintaxis legal para los enteros es especificada por la gramática de cadenas numéricas para literales de enteros decimales del WMLScript1.1 con las siguientes reglas:

- El proceso termina cuando el primer carácter encontrado no es mayor '+' o '-' o un dígito decimal
- El resultado es la cadena convertida a un valor entero

Parámetros: valor = String

Valor devuelto: Entero o invalid.

Excepciones: En caso de error un valor invalid es devuelto.

Ejemplos:

```
var i = Lang.parseInt ("1234");      // i = 1234
var j = Lang.parseInt (" 100 m/s"); // j = 100
var k = Lang.parseInt("The larch") // k = invalid
```

random**Función:** random(valor)**Descripción:** Devuelve un entero positivo que es mayor o igual a 0 pero es menor o igual al numero dado. Si el valor es de punto flotante Float.int() es usada primero para calcular el valor entero actual**Parámetros:** valor = Entero**Valor devuelto:** Entero o invalid.**Excepciones:**

- Si el valor es igual a cero, la función devuelve cero.
- Si el valor es menor que cero, la función devuelve invalid.
-

Ejemplos:

```
var a = 10;  
var b = Lang.random(5.1)*a; // b = 0..50  
var c = Lang.random("string"); // c = invalid
```

seed**Función:** seed(valor)**Descripción:** Inicia una secuencia de números pseudo aleatorios y devuelve una cadena vacía. Si el valor es cero o un entero positivo entonces el valor dado es usado para la inicialización, de otra manera, el valor dependiente del sistema es utilizado.**Parámetros:** valor = Entero**Valor Devuelto:** Cadena o Invalid.**Ejemplos:**

```
var a = Lang.seed(123); // a = ""  
var b = Lang.random(20); // b = 0..20  
var c = Lang.seed("seed"); // c = invalid (random seed left unchanged)
```

CAPITULO IV

HERRAMIENTAS

OBJETIVO:

- Describir las herramientas que se utilizan para desarrollar aplicaciones inalámbricas

4.1 Editores WML

4.2 Pasarelas

4.3 Emuladores

4.4 Editores WBMP

4.5 KIT

4.1 EDITORES WML

| WML | Características |
|---------------------------------|--|
| WML Express | Procesador de palabra WAP disponible en versiones online y descargable. |
| WAPTor | El WAPTor es un editor WML para Windows 95/98, Windows NT y Windows 2000. Permite abrir y editar archivos WML existentes o crear nuevos archivos a partir de una simple plantilla. |
| WMLEdit | Tiene dos tipos de descarga diferentes. Uno requiere que usted tenga VBRUN 6.0 y el otro lo instala por usted. |
| Textpad | TextPad is a simply, powerful replacement for Notepad. It is a tool for editing your web pages, or a programming IDE. TextPad does what you want, the way you would expect. Now there is a WML clip library add-on available for TextPad. Windows 9x,NT,2000 |
| Nokia WML Studio | Nokia WML Studio para Dreamweaver es una extensión descargable al Dreamweaver, el cual habilita a los usuarios para crear contenido para dispositivos WAP. La extensión se activa desde el entorno de edición HTML y provee un ambiente de desarrollo “Lo que ve es lo que obtiene” para el Lenguaje de Marcas Inalámbrico (WML) La extensión incluye un completo analizador gramatical WML 1.1 con realimentación visual de errores, y función de previsualización que puede despegar contenido WML en múltiples interfaces de dispositivos móviles WAP Nokia, corriendo en un navegador convencional. Funciona en plataformas Windows. |
| ScriptBuilder 3.0 WML Extensión | Esta extensión provee soporte WML para el ScriptBuilder 3.0. Funciona en plataformas Windows. |
| Dot WAP 2.0 | DotWap es una herramienta visual para construcción de sitios WAP. Permite la construcción rápida de sitios WAP sin necesidad de tener conocimientos en WAP o WML. Corre en plataformas Win32 y su versión es libre |
| XML Writer | Provee a los usuarios un extenso rango de funcionalidad XML tal como: validación de documentos XML contra un esquema DTD o XML, y la habilidad de convertir XML a HTML utilizando hojas de estilo XSL. Los usuarios también pueden combinar hojas de estilo en cascada (CSS) con XML para un diseño directo de datos XML. |
| Mobile JAG | MobileJAG es de versión libre y no es necesario descargarlo ni instalarlo. Todo se encuentra online |

4.2 PASARELAS

| Pasarelas | Características |
|---|---|
| Anam.com WirelessWindow | Gateway de Anam.com. |
| Anny Way Carrier WAP Gateway | La Gateway Anny Way Carrier WAP es totalmente compatible con WAP y soporta los últimos estándares de WAP. |
| Anny Way Corporate WAP Gateway | Este Gateway posibilita a las compañías, integradores y desarrolladores de software dar acceso a los servicios basados en su intranet o internet a clientes, socios de negocios y empleados vía dispositivos WAP. |
| Audicode WAP Server 2.0 | Para usuarios de Microsoft Windows 95/98/NT4/2000. Se incluye guía de usuario. |
| Ericsson's WAP Gateway | Conecta el mundo móvil con el mundo de Internet a manera de nodo de una red GSM. Adicionalmente a los servicios de internet la gateway soporta la integración con servicios de telefonía. Esta cumple con la especificación WAP 1.1. |
| iNWAVE WAP Gateway | Pasarela apropiada para ser usada por Operadores, Portales y Empresas. Esta trabaja con todos los estándares industriales de servidores web, haciendo uso de HTTP 1.0. Es totalmente compatible con WAP 1.1. y ofrece WTLS1.2 como una opción. Soporta portadores CSD y SMS. Esta disponible para Windows NT/2000 y plataformas Solaris. SAS es miembro de WAP-Forum. |
| Jataayu Carrier WAP NServer | Creado para Proveedores de Servicios Wireless, es una plataforma de alto desempeño, creada para entregar servicios ininterrumpidos a los suscriptores. Esta puede trabajar como una Gateway tanto WAP como HTTP. Igualmente puede almacenar contenido y aplicaciones localmente. |
| Jataayu Enterprise NWAP Server | Permite a las organizaciones acceder a información crítica del negocio tal como estados de las ordenes, nivel de stocks, información de daños y bases de conocimiento corporativas. |
| Jataayu Personal WAP NGateway | Permite marcar a su caso u oficina y obtener información personal en su dispositivo móvil. Todo lo que necesita es instalar Jataayu Personal WAP Server en su PC Windows, y configurarlo para trabajar con las aplicaciones email de su preferencia. |
| Kannel Open Source NWAP y SMS Gateway | Gateway de fuente abierta. Pretende entregar la infraestructura básica de WAP de manera gratuita. Kannel trabaja igualmente como una gateway SMS para redes GSM. La mayor parte de Teléfonos GSM pueden enviar y recibir |

| | |
|------------------------------|--|
| | mensajes SMS, de esta manera es posible proveer servicios a un rango más amplio de clientes. |
| Kuulalaakeri WAP Gateway 1.0 | Esta construida de acuerdo con las especificaciones WAP 1.1. Este servidor WAP se acomoda tanto a operadores como a compañías. La Gateway puede se usada para almacenar diferentes servicios WAP y prestarlos a clientes con dispositivos celulares GSM de la nueva Generación. |
| Motorola's WAP Server | Servidor WAP de Motorola para desarrolladores. |
| Nokia WAP Server 1.0 | Gateway de Nokia para operadores Celulares. |
| OnnClick.com | Provee acceso global para cualquier dispositivo móvil que soporte los estándares para Internet Inalámbrico WAP/WML o HDML. |
| ONphelia WAP GaNteway | Gateway que provee fácil acceso a información interactiva, servicios y aplicaciones basadas en el Web, igualmente permite transacciones seguras desde dispositivos móviles. Esta es interoperable con una variedad de dispositivos WAP, tiene soporte para GPRS y otras tecnologías futuras. Incluye tanto la funcionalidad de una gateway de protocolos como la de un codificador y decodificador de contenidos Web para traducir contenido web a formato compacto con el fin de para optimizar el tamaño y numero de paquetes que viajan sobre la red inalámbrica. |
| Peramon WAP Gateway | Soporta todos los tipos de dispositivos WAP y tiene un alto desempeño en la conversión de contenidos Web a WAP. Es compatible tanto con servidores Windows NT como UNIX. |
| Real GEWI beta | Actualmente se puede obtener una versión gratuita de Realgewi. Hasta que el periodo de prueba se de por culminado, luego, existiran dos versiones diferentes que son: Edición limitada Edicion comercial Completa. |
| RTS Wireless WAP Gateway | Gateway para operadores de red. |
| ThinAir Server 1.1 | Soporta un amplio rango de dispositivos inalámbricos. Palm, Pocket PCs, dispositivos inalámbricos RIM, y todas las variaciones de teléfonos WAP. Este servidor esta orientado a : Empresas con profesionales móviles quienes necesitan acceso instantáneo a datos corporativos. ASPs, ISPs, portadores u otros proveedores de servicios de Web que deseen ofrecer servicios inalámbricos a sus clientes. Compañías de software o integradores de sistemas que deseen agregar opciones inalámbricas a sus soluciones. |
| ThunderWAP® para IIS | Permite la incorporación del stack de protocolos WAP al Internet Information Server de Microsoft version 4.0 o 5.0. Luego de la instalación su servidor IIS es capas de |

| | |
|------------------------|---|
| | comunicarse directamente con dispositivos WAP. No se requiere mas de una Gateway WAP. |
| UP LINK Aerver Suite | Creada para operadores moviles. |
| WAP Lite | Infinite Technologies. Conecta dispositivos móviles compatibles con WAP a aplicaciones de contenido basada en una Intranet o Internet. Invoca archivos y datos desde cualquier servidor web, via HTTP, soporta tanto métodos GET como POST. Soporta modos orientadas y no orientada a la conexión sobre UDP. Soporta Nokia 7110, Ericsson MC218 y Motorola Timport con mas dispositivos por venir. |
| WapWorld Linux | Gateway basada en linux. |
| WAP Service Broker | Conjunto de soluciones WAP y CMG's. |
| Ztango.com WAP Gateway | Soportan estándares abiertos y formatos que son encontrados en los actuales SMS y en el futuro WAP. La arquitectura de la plataforma contiene un servidor de mensajes que son interfaces con SMSC para correr servicios basados en el tradicional SMS, así como estándares WAP. Esta arquitectura hereda sistemas que son capaces de correr aplicaciones tanto SMS como WAP. |

4.3 EMULADORES

| Emuladores | Características |
|--|---|
| Emulador Ericsson R380 | <p>El emulador WAP R380 tiene como propósito probar aplicaciones WML sesarrolladas por el Navegador WAP en el Ericsson Smartphone R380. El emulador contiene el navegador WAP y las opciones de funcionalidad que pueden ser cargadas en el R380.</p> <p>Nota: se requiere ser miembro en Ericsson Developer Zone (gratis)</p> |
| Klondika WAP Browser | <p>El Browser WAP para Windows Klondike es un navegador WAP que corre sobre cualquier plataforma de Windows de 32 bits y le permite ver páginas WML en internet o en un stio local. Klondike es muy similar a su navegador HTML favorito. Sin embargo, los navegadores estándar (Netscape Navigator, Microsoft IE, etc.) no están diseñados para soportar WML</p> |
| Wapalizer | <p>Emulador WAP basado en Web que muestra una página WML en un entorno de Nokia 7110 o Ericsson R380 El script llama páginas WML desde sitios WAP y las convierte en HTML “en el vuelo”. Esto quiere decir que usted será capaz de ver la mayoría de páginas WAP, pero algunas páginas especialmente aquellas con una gran cantidad de entradas de formulario serán muy difíciles de convertir a HTML.</p> |
| Wapsilon | <p>Visualizador WAP basado en Web que despliega sitios en la ventana del navegador (mayor espacio) o en un equipo WAP futurista. Wapsilon convierte sitios WAP en HTML que pueden ser visualizados en su sitio o en su “equipo”. Wapsilon puede ser integrado dentro de su website o agregado a su navegador.</p> |
| Wapjag | <p>Emulador WAP basado en Web que despliega sitios en una ventana separada similar a un teléfono móvil.</p> |
| Yospace | <p>Edición Web Site 1.0 Beta Disponible: El emulador SmartPhone versión Web Site es un apropiado, liviano y altamente versátil emulador capaz de soportar diferentes “personalidades” de teléfono. Edición 1.0 Beta para Desarrolladores Disponible: Los desarrolladores WAP pueden utilizar la edición de escritorio del emulador para visualizar aplicaciones WAP desde su escritorio con la certeza de que el emulador provee una fiel reproducción de los actuales terminales. Usted puede agregar soporte para microteléfonos adicionales tan pronto como estén disponibles. Acompañado de una colección de herramientas de línea de comandos (codificador WML, compilador WMLScript) para fácil</p> |

| | |
|--|--|
| | integración con su entorno de desarrollo existente. |
| <u>EzWAP 1.0</u> | El primer navegador WAP independiente de la plataforma, habilitando toda clase de sistemas computacionales para acceder el internet móvil: equipos móviles (PDA, PC de bolsillo, etc), equipos de computación móviles y sistemas embebidos, PCs corriendo Microsoft Windows NT, "00, CE... |
| <u>WinWAP</u> | WinWAP es un navegador WML que trabaja en cualquier computador con Windows de 32 bits instalado (Win95, Win98, WinNT). Usted puede navegar en archivos WML localmente desde su disco duro, o desde internet con el protocolo http (tal como su navegador web normal). |
| <u>WAPMan for Windows 95/98/NT</u> | El WAPman es un equipo de navegación portable, combinando el acceso a internet con las propiedades de un teléfono móvil. Con su estructura de compuerta WAP única, el WAPman tiene capacidades de rápida descarga y es altamente compacto y portable. |
| <u>WAPsody</u> | Wapsody simula la mayor parte de los aspectos de WAP. Está diseñado para utilizarlo como un ambiente de desarrollo de aplicaciones WAP. El ambiente de simulación de Wapsody puede ser ejecutado por sí solo, o puede ejecutar demos que están hospedados en la infraestructura WAP en el laboratorio de investigaciones de IBM en Zurich. |
| Wireless Companion | Con el Wireless Companion usted puede acceder algún contenido WAP y WEB en internet incluyendo el los servicios personales inalámbricos en YOURWAP.com. |
| Opera | Opera es un potente navegador HTML que ahora soporta WML. |

4.4 EDITORES WBMP

| WBMP | Características |
|------------|--|
| WAP Pictus | WBMPgen es una herramienta para crear mapas de bits inalámbricos (WBMP) "en el vuelo". Puede por ejemplo ser utilizado para generar diagramas desde valores en una base de datos, desplegar cierta clase de estadísticas en su sitio WAP. La versión actual soporta el trazado de pixels individuales, líneas, rectángulos y círculos. El programa está actualmente disponible en el formulario de una clase Java y una librería C++ para Windows. |
| BMP/WBMP | Convierte imágenes entre formatos bmp y wbmp |

| | |
|---------------------------|--|
| WAPTiger BMP Converter | Con esta herramienta, usted puede producir archivos WMPs. Estos son mapas de bits que corresponden al estándar WAP. Los WBMPs producidos pueden ser utilizados con el código en una página WML. Cada navegador web inalámbrico puede mostrar este dibujo en la pantalla. |
| WAPDraw | WAPDraw es un programa para construir imágenes WBMP. Los usuarios pueden hacer imágenes de hasta 96x80 pixels. |
| Dissectimage | Soporta los formatos de imágenes más populares: BMP, GIF, JPEG, ICO, WMF, RAW y WBMP. |
| Pic2wbmp | Esta pequeña herramienta de versión libre importa archivos gráficos y los guarda en formato WBMP. WBMP es un formato de gráficos utilizado para teléfonos móviles que soportan el protocolo WAP. |
| ImageMagick | ImageMagick TM, version 5.1.1, es una robusta colección de herramientas y librerías para leer, escribir y manipular una imagen en alguno de los formatos más populares, incluyendo WBMP. |

4.5 KIT

| Kit | Características |
|------------------|---|
| SDK de Mobiledey | El SDK de Mobiledev permite una rápida creación de aplicaciones inalámbricas con mínimos conocimientos de WAP, WML o HDML. En Windows 2000, Windows NT V4.0, Windows 98 o Windows 95 |
| Kit de Ericsson | El SDK (Service Development Kit) para el Ericsson R380. Hay versiones de este kit de Ericsson para varios sistemas operativos. Para bajárselo hay que registrarse, pero es gratuito |
| Nokia WAP | El kit de desarrollo del Nokia, sólo funciona para Windows NT. Para bajárselo hay que registrarse, pero es gratuito. |
| Phone.com (UpSD) | El SDK de Phone.com se llama Unwired Planet (UP) y es gratuito, aunque hay que registrarse. Es el simulador del browser que llevan muchos de los teléfonos móviles WAP que conocemos. Toolkit compatible con WAP 1.1, con la ventaja de |

| | |
|---|---|
| | <p>poder descargar plugins para el emulador en esta misma página. Permite a los desarrolladores de red crear servicios y aplicaciones HDML y WML fácilmente. Es gratis.</p> |
| <p>SDK de Motorola (o Mobile Applications Development Toolkit)</p> | <p>Simplifica el desarrollo de las aplicaciones de voz y datos por terceros desarrolladores incorporando el lenguaje de marca VoxML y el WML. Incluye también su Plataforma de comunicaciones MIX basada en la aproximación cliente/servidor en una filosofía API abierta. Motorola, a través de su kit de desarrollo de aplicaciones móviles permite el desarrollar rápidamente aplicaciones inalámbricas que integren voz y datos para que sean operativos a través de esta plataforma.</p> |
| <p>SDK de Dynamical Systems Research</p> | <p>Incluye versión 1.1. un browser WAR (Wireless Applications Reader) y un conjunto de ejemplos de código fuente de WML y WMLScript. Pronto saldrá la nueva</p> |
| <p>Z-Y-G-O</p> | <p>Incluye versión 1.1. un browser WAR (Wireless Applications Reader) y un conjunto de ejemplos de código fuente de WML y WMLScript. Pronto saldrá la nueva</p> |

CAPITULO V

SERVICIOS

OBJETIVO:

- Mencionar los servicios que proporciona el protocolo WAP

- 5.1 Artículos
- 5.2 Directorio WAP
- 5.3 Servicios

5.1 ARTÍCULOS

Sistemas de pago y banca virtual

Algunas compañías de telecomunicaciones, proveedores de tarjetas inteligentes, bancos y suscriptores están trabajando en el desarrollo de sistemas para realizar transacciones seguras en Internet móvil basadas en tarjetas Inteligentes. (Smart Cards).

Comercio móvil, una nueva frontera – Parte Uno

Al crecimiento de las redes inalámbricas se suma ahora, el desarrollo de una nueva forma de negocios en el WWW: El comercio móvil.

Configurando APACHE para entregar contenido WAP

WAP define sus propios tipos de datos MIME, esto exige que los servidores web deban ser reconfigurados para entregar este tipo de contenido.

WAP 2.0, una nueva visión al mundo de Internet Móvil

Hoy en día, el WAPForum (<http://www.wapforum.com>) está decidido a darle el empujón definitivo a WAP con la versión 2.0 del estándar, empujón que lo llevará a competir mano a mano con iMode, su contendor más fuerte y que lo pondrá a las puertas de los sistemas de tercera generación como la alternativa más viable para la implementación de servicios de Internet Móvil.

5.2 DIRECTORIO WAP

| Directorio colombiano de sitios WAP | |
|---|---|
| Arte y Entretenimiento: Portales, Juegos | Autos |
| Economía: Bancos, Corporaciones | Medios de comunicación: Periódicos, Revistas |
| Tecnología WAP | Telecomunicaciones: Operadores de Celular |
| Viajes / Información: Agencias | |

5.3 BENEFICIOS PARA EL USUARIO

WAP ha sido desarrollado para proporcionar servicios de valor añadido al usuario. Debido a su diseño e implementación WAP proporciona:

Un interface de usuario estándar: El interface es independiente del terminal usado. Al igual que las páginas WEB, se pueden ver en cualquier plataforma con cualquier sistema operativo (Mac, Windows, Unix, etc.), las paginas WML son independientes del dispositivo (PDA, teléfono móvil, terminales portátiles, etc.).

Ubicuidad del servicio: Al ser un servicio inalámbrico e independiente del transporte, un usuario puede solicitar la información en cualquier momento y lugar.

Gran disponibilidad de terminales: Al ser un estándar abierto, cualquier fabricante puede incorporarlo en sus teléfonos (actualmente más de 15). Además conforme se introduzca WAP en la sociedad, se espera que los fabricantes de PDA's, lo incorporen en sus terminales. Multitud de aplicaciones: En los últimos años, el modelo Internet ha demostrado ser la forma más barata y más efectiva de distribuir aplicaciones al mayor número de personas. WAP al adoptar este modelo, garantiza, que los usuarios finales dispondrán de un gran abanico de aplicaciones.

CONCLUSIONES

CONCLUSIONES

El área de la telefonía a tenido grandes avances en lo que a Internet se refiere, con los nuevos teléfonos celulares y el servicio móvil a Internet que están ofreciendo las compañías celulares podremos hacer muchas de las operaciones que hacíamos desde nuestras computadoras, como leer correos electrónicos, consulta de saldos en bancos, transacciones e inclusive comprar algún producto desde nuestro teléfono. Y aún hay más..., con la introducción de la tercera y cuarta generación de telefonía móvil, vamos a tener la posibilidad de recibir y transferir grandes cantidades de información en el rango de Mbps, por lo que podremos recibir video o música en tiempo real desde nuestros teléfonos y otros dispositivos inalámbricos.

El uso del WAP se ha incrementado como un área de interés en la investigación, ya que se están creando nuevos sistemas de computadoras, además, debido a su diseño e implementación proporciona un interface independiente del terminal usado, al igual que las páginas web, se pueden ver en cualquier plataforma con cualquier sistema operativo (Mac, Windows, Unix, etc.).

Al ser un servicio inalámbrico e independiente del transporte, un usuario puede solicitar la información en cualquier momento y lugar.

Al ser un estándar abierto cuenta con gran disponibilidad de terminales, ya que cualquier fabricante puede incorporarlo en sus teléfonos, además conforme se introduzca WAP en la sociedad, se espera que los fabricantes de PDA's, lo incorporen en sus terminales.

El modelo Internet ha demostrado ser la forma más barata y más efectiva de distribuir aplicaciones al mayor número de personas y WAP al adoptar este modelo, garantiza que los usuarios finales dispondrán de un gran cantidad de aplicaciones.

De acuerdo a las experiencias obtenidas durante el desarrollo de este trabajo se infiere lo siguiente:

- Con la creación de este trabajo se estará brindado una guía clara, objetiva y concisa, con lo que se logrará una rápida introducción de cómo trabaja este protocolo.
- Esta recopilación de información nos muestran como utilizar los elementos que aquí se describen de una manera ordenada.
- Los conceptos plasmados tienen la ventaja de poderse emplear como tema de apunte, referencia o guía para introducirnos a un nuevo protocolo.

APÉNDICES

APÉNDICE A: COMPARACION ENTRE LOS PROTOCOLOS WAP Y WEB.

| WEB | WAP |
|---|--|
| Es necesario tener conocimientos previos de computación , para navegar por la Internet. | No es necesario tener conocimientos anteriores de computación. |
| Ingreso a través desde computadores ,con un alto costo de inversión para los usuarios. | El acceso es desde móviles o celulares Wap de bajo costo ,en comparación al precio de un computador. |
| El acceso se realiza desde un lugar específico , restringiendo su libertad de uso. | Puedes acceder desde cualquier lugar. |
| Para obtener la información que deseas buscar desde Internet , deberás organizarte y decidir premeditadamente el momento y lugar desde donde te conectarás. | El acceso a la información es de forma inmediata ,sin importar el sitio ,ni desde donde te conectes. |
| En estos momentos la gran masa de usuarios de Internet son principalmente los jóvenes ,ya que este segmento de la población, es capaz de entender y asimilar con más rapidez el uso de las nuevas tecnologías . | Su Uso Simple e Intuitivo permite llegar a un amplio espectro de personas , aumentando el radio de acción y efecto de la internet, en la población . |

GLOSARIO

APÉNDICE B: GLOSARIO DE TÉRMINOS

Agente de Usuario

Se refiere al software capaz de interpretar códigos como el WML o WMLScript, que está instalado en nuestro dispositivo WAP.

Algoritmos simétricos

Los algoritmos simétricos, o de clave secreta, se caracterizan por ser altamente eficientes (en relación al tamaño de su clave) y robustos. Se les llama así porque se emplea la misma clave para cifrar y para descifrar; (RC5, DES, 3DES, IDEA, RSA, DH).

Ancho de banda

Técnicamente es la diferencia en hertzios (Hz) entre la frecuencia más alta y la más baja de un canal de transmisión. Este término se usa muy a menudo para referirse a la velocidad de transmisión. *En Telefonía Móvil*- Margen de frecuencias capaz de transmitirse por una red de telecomunicación y de interpretarse en sus terminales.

Asentimiento de Usuario

Esta característica permite al usuario confirmar cada mensaje recibido por el proveedor WTP

Autenticación

El proceso de identificar y confirmar la identidad de cualquier teléfono desde el que se llame o que reciba una llamada. Se trata de un requisito previo al acceso a los recursos de un sistema informático.

Autenticación de Mensajes

Un resumen de mensajes se llama *message digest*, *one-way function* o *hash function*, dependiendo de la longitud del mensaje. Los resúmenes de mensajes se utilizan para crear mensajes cortos (*message digest*), representaciones con una longitud fija más largas (*one-way function*), o mensajes de longitud variable (*hash function*). Los resúmenes de mensajes están diseñados para que sea muy difícil determinar el contenido del mensaje a partir de su resumen, y también para que sea imposible encontrar un mismo resumen, que haya sido creado por dos mensajes diferentes, así se elimina la posibilidad de sustituir un mensaje por otro mientras se mantiene el mismo resumen.

Una función *hash* , resumen o compendio toma una cadena de entrada, usualmente de longitud no definida, y produce un número de salida que es función de la entrada. Las funciones *hash* producen una única salida para la misma entrada, es decir son determinísticas. (MD5, SHA-1)

Bearer

Se encarga de transmitir los datos desde el teléfono móvil a la operadora de telefonía. Los principales portadores son SMS, CSD y GPRS.

Card

Estructura básica definida por el lenguaje WML , que representa una interacción con el usuario, y sintacticamente contiene una primera parte no visible, con contenidos ejecutables, a la que le sigue una segunda parte de elementos visibles.

CDMA (Code Division Multiple Access)

Tecnología celular que no asigna una frecuencia específica a cada usuario, sino que cada canal utiliza un espectro completamente disponible. Las conversaciones individuales se codifican con una secuencia digital tomada al azar. Se utiliza de acuerdo con el estándar US (IS 95) y con un intervalo de frecuencias entre 800 y 1900 MHz.

Celda

Área geográfica que contiene una antena y los dispositivos necesarios para recibir señales desde otra celda. Abarcan diámetros desde unos pocos kilómetros hasta unos 32.

Célula

Dispositivo receptor o transmisor al cual se conecta un teléfono GSM.

Celular

Las comunicaciones celulares es una tecnología de comunicaciones inalámbricas en la que las áreas de comunicación se dividen en pequeñas áreas llamadas celdas y en las que las transmisiones pasan de celda a celda hasta que llegan a los destinatarios. Cada celda contiene una antena y dispositivos que permiten recoger información y pasarla de una celda o de un emisor a otro.

Cobertura

Extensión del haz de la señal emitida por antenas de telefonía móvil que habilita la realización de llamadas sin interrupción pasando de una célula a otra.

CODEC (Compressor/DECompressor)

Chip que permite una utilización más óptima de la anchura de banda al transmitir de una manera rápida la voz a través de GSM, desechando información repetitiva o inútil.

Codificación

Es una operación de la cual un mensaje se ve modificado de manera tal que su contenido no puede ser descifrado a menos que se posea una determinada «clave». Esta actúa como una llave que me permite «abrir» el mensaje y leer su contenido.

CSD (Circuit Switched Data)

Conmutación de Circuitos de Datos.

Datagrama

Datos agrupados en paquetes que viajan por una red.

Deck

Conjunto de cartas. Por ser ese conjunto una representación de las sucesivas interacciones con el usuario, un deck sería el equivalente a un programa o documento WAP, y por ello tiene asociado un único URL (Uniform Resource Locator). En la práctica, y a efectos de desarrollo, se puede comparar un deck, o documento WML con un documento HTML que contiene numerosas secciones identificables sin ambigüedad (de hecho, en WML se usa para ello la sintaxis #etiqueta, lo que recuerda enormemente a las anclas nominales de HTML).

DECT (Digital Enhanced Cordless Telecommunications)

Norma armonizadora de comunicaciones por medio de terminales sin cables y que mejora la calidad de transmisión de voz.

DES (Data Encryption Standard)

Algoritmo diseñado por IBM y utilizado desde los 70. Es un método de cifrado altamente resistente frente a ataques criptoanalíticos diferenciales. Su tamaño de clave (56 bits) la hace vulnerable a ataques de fuerza bruta. En la actualidad ofrece protección contra el pirata informático habitual, pero no contra un esfuerzo masivo por parte de un usuario con grandes recursos.

DES (Data Encryption Standard)

Se creó para evitar el problema de la clave corta. El triple DES está basado en tres iteraciones del algoritmo y utiliza una clave de 128 bits, siendo compatible con el DES simple.

Se utiliza una bits clave de 128 (16 de paridad y 112 de clave), se aplican 64 bits a los dos DES y otros 64 bits al DES inverso (ANTIDES) que se realiza entre los otros dos.

Digital

Conversión de la información en bits de datos para que se transmitan a través de cables, fibra óptica, cable de fibra óptica, o de manera inalámbrica. Este sistema permite la transmisión simultánea de voz, datos y video. La tecnología de comunicaciones digitales permite una velocidad de transmisión de datos y menos errores que la tecnología analógica. Debido a que las señales analógicas se amplifica en cada estación de repetición, cualquier sonido se amplifica, mientras que las señales digitales se detectan y se regeneran (no se amplifican) y cualquier ruido se pierde a menos que corresponda a un valor que el regenerador interprete como una señal digital. Son sistemas digitales: GSM 900, GSM 1800, GSM 1900, D-AMPS y DECT.

Dual Band

Se refiere a la capacidad de la infraestructura y teléfonos GSM para operar a través de dos bandas de frecuencia. Los primeros dispositivos de banda dual los ofreció Motorola en 1997. Ese mismo año estaban ya preparadas las primeras redes europeas de banda dual de 900/1800 MHZ.

Dual Mode

Aquellos dispositivos que están adaptados a más de un sistema, con la ventaja de utilización digital en zonas cubiertas por el sistema analógico.

Elemento

Aquel que especifica el Markup y la información estructural en una baraja WML.

Emulador

Programa que hace una replica de su Palm en un computador; esto es usado frecuentemente para probar programas antes de instalarlos en la Palm real. También es muy usado por desarrolladores de programas para Palm.

Gateway (Pasarela)

Dispositivo que permite conectar entre sí dos redes normalmente de distinto protocolo o un host a una red.

GSM (Global System for Mobile Communications)

Sistema Global para Comunicaciones Móviles. Sistema digital de telecomunicaciones principalmente usado para telefonía móvil. Definido inicialmente como un estándar paneuropeo para una red telefónica celular digital que soportará la itinerancia entre

países, GSM es ahora el principal estándar móvil digital del mundo. Actualmente está implantado en las bandas de frecuencia de 900 Mhz, 1800 Mhz o 1900 MHz.

Hertz

Unidad de frecuencia estándar, referido a ciclos por segundo.

HDML (Handheld Markup Language)

Interoperabilidad: Posibilidad de los componentes de hardware y software de funcionar juntos efectivamente.

IDEA (International Data Encryption Algorithm)

Desarrollado por Xuejia Lay y James Massey. A pesar de pocos años en uso, es probablemente el mejor algoritmo de bloques existente. Utiliza clave de 128 bits y se cree que es resistente al criptoanálisis. Se encuentra bajo patente de Ascom-Tech, aunque se permite su uso gratuito para aplicaciones no comerciales.

IP (Internet Protocol)

Protocolo de Internet

ISO (International Standards Organisation)

Organización con sede en Ginebra responsable de la definición de normas técnicas internacionales.

Latencia

Es el tiempo que tarda un dato a estar disponible desde que se realiza su petición. Por ejemplo, si estamos comiendo y te digo pásame la sal, el tiempo que tardas en dármela sería tu latencia. Se puede comparar con el tiempo de reacción. Se mide en nanosegundos (ns) o en milisegundos (ms). Cuanta menos latencia, mejor.

MAC (Medium Access Control)

Control de Acceso al Medio

MD5

El algoritmo de *hash* más utilizado en estos momentos es el MD5. Este algoritmo fue desarrollado por Ronald Rivest en 1995 y está basado en dos algoritmos anteriores MD2 y MD4.

Todos estos protocolos producen un número de 128 bits a partir de un texto de cualquier longitud.

MD5 comienza rellenando el mensaje a una longitud congruente en módulo 448 mod 512. Es decir la longitud del mensaje es 64 bits menos que un entero múltiplo de 512. El relleno consiste en un bit en 1 seguido por cuantos bits en 0 sean necesarios. La longitud original del mensaje es almacenada en los últimos 64 bits del relleno.

Durante varias rondas de procesamiento el algoritmo toma bloques de 512 bits de la entrada y los mezcla con los 128 bits del buffer. Este proceso es repetido hasta que todos los bloques de entrada han sido consumidos. El valor resultante en el buffer es el *hash* del mensaje.

Micro navegador

El micro navegador de los aparatos para WAP solamente permiten el acceso a sitios en formato WML (lenguaje de marcado inalámbrico). Si entras a un sitio en HTML o HDML, recibirás un mensaje de error en la pantalla de tu teléfono.

Packet Data

El servicio de packet data significa que los clientes están siempre conectados, de modo que los servicios deberían tener más rápido y fácil acceso.

Palm

Nombre de la compañía más importante en el mercado de los PDAs, y nombre genérico que se les da comúnmente a todos estos portátiles. Además es importante que también lleven este nombre el sistema operativo "Palm OS". Corriendo sobre esta plataforma los equipos fabricados por Handspring, Sony, IBM y obviamente Palm Computing.

PDA (Personal Digital Assistant)

Asistente Personal Digital. Se refiere a aquellos dispositivos de mano que pueden combinar ordenador, teléfono, fax y conexión a Internet, como por ejemplo, una agenda electrónica tipo Palm V.

PPP (Point-to-Point Protocol)

Protocolo Punto-a-Punto. Paquete de software que permite al usuario tener conexión directa a Internet a través de la línea telefónica.

Pragma

Los pragmas son sentencias que permiten añadir ciertas características más a los ficheros WMLScript. Hay tres tipos diferentes de pragmas: url, access y meta.

Primitiva de servicio

Al intercambio lógico de información entre la capa de sesión y capas adyacentes es conocido como "Primitiva. Estas primitivas están definidas tanto en el diagrama de tiempos en el que se tienen que invocar, como en los parámetros de intercambios.

Protocolo

El protocolo se refiere a un conjunto de reglas para enviar y recibir información en una red. Las reglas determinan el formato de los datos que se transmiten y otros aspectos de la conexión a la red, tales como la manera de detectar y corregir errores. El controlador de protocolo en cada equipo es un software que respeta dichas reglas cuando envía y recibe información. Estos controladores también se llaman frecuentemente protocolos.

PROXY (Servidor Caché)

El servidor proxy acepta solicitudes hechas por un cliente (por ejemplo, su navegador) y las direcciona al servidor respectivo (WWW, ftp o gopher) "en nombre" del cliente; espera el resultado del servidor y se lo hace llegar al cliente. El cliente y el servidor no se comunican directamente (no se "ven").

RC2

Código protegido bajo secreto comercial (aunque no patentado) por RSA Data Security Inc. Existen ataques criptoanalíticos que, aunque requieren de gran cantidad de texto cifrado, muestran las vulnerabilidades de RC-2. Actualmente RC2 tiende a utilizarse cada vez menos en beneficio de RC4.

RC4

Es un intento de reemplazar RC2 por un algoritmo más sólido. También es un secreto comercial, aunque (al igual que RC2) su código fuente ha sido publicado en grupos de discusión. No se conocen ataques contra él. Forma una parte vital del sistema de cifrado en capas SSL, ampliamente utilizado en navegadores de Internet tales como Netscape Navigator y Microsoft Internet Explorer. La versión exportable de RC4 tiene una clave de solamente 40 bits, lo que lo hace altamente vulnerable a ataques de fuerza bruta. La versión EEUU, con clave de 128 bits, es segura.

RC5

Diseñado por Ron Rivest y se encuentra bajo patente de RSA Data Security Inc. Es relativamente nuevo, y se conocen ciertos tipos de ataques contra él. Asimismo existe un cierto número (pequeño) de claves débiles que no deben utilizarse. A pesar de ello, se le considera un sistema seguro.

RSA (Rivest, Shamir, Adleman)

Algoritmo de clave pública más utilizado, y uno de los más populares. En principio utiliza claves de cualquier longitud; en la actualidad se emplean claves de 1024 bits, consideradas lo bastante largas como para resistir ataques de fuerza bruta. Su seguridad se basa en la dificultad de factorizar números primos de gran tamaño. En principio se puede deducir la clave secreta conocida la clave pública, pero solamente por medio de la factorización de números de gran longitud.

Diffie-Hellman

Algoritmo de intercambio de claves. Una variante conocida como ElGamal funciona como algoritmo de clave pública; por abuso del lenguaje, se suele conocer dicho algoritmo como Diffie-Hellman (o DH, variante ElGamal). Se basa en llamado problema de los logaritmos discretos, que se cree es computacionalmente tan complejo como el de la factorización de números primos (y, al igual que su primo RSA, no está demostrado que el problema de logaritmos discretos no se pueda resolver mediante herramientas matemáticas más poderosas en el futuro). Está siendo utilizado cada vez con más frecuencia, entre otras cosas por cuestiones de patentes (la patente D-H ha expirado). Aunque el algoritmo DH es más antiguo que el RSA, es más reciente en su utilización.

SHA

Algoritmo desarrollado por NSA. A diferencia de MD5, SHA genera números *hash* de 160 bits.

SMS (Short Message Service)

Servicio de Mensajes Cortos. Es la tecnología que permite mandar mensajes de texto y recibirlos a través de teléfonos móviles, máquina de fax y/o direcciones IP. Los mensajes no deben superar los 160 caracteres alfanuméricos y no contener imágenes gráficos.

SSL (Secure Socket Layer)

Capa de Enlace Seguro. Uno de los estándares más reconocidos para la encriptación de datos, desarrollado por Netscape, que proporciona integridad y certificación en la transmisión. Es un protocolo para proporcionar comunicación cifrada entre un cliente (navegador) y un servidor HTTP, que utiliza conjuntamente el cifrado simétrico y el cifrado asimétrico. Protocolo que ofrece funciones de seguridad a nivel de la capa de transporte para TCP.

Teléfono Móvil

Teléfono portátil sin hilos conectado a una red celular y que permite al usuario su empleo en cualquier lugar cubierto por la red, normalmente dispersa por todo el territorio en el que opera la compañía, que puede ser todo el país. Una red celular, y los teléfonos a ellos conectados, pueden ser digitales o analógicas.

W3C (World Wide Web Consortium)

Consortio World Wide Web. Organización que desarrolla especificaciones interoperables, software y herramientas para la www.

WAE (Wireless Application Environment)

Entorno de Aplicaciones Inalámbricas. Parte integrada del Protocolo de Aplicaciones Inalámbricas cuyo fin, basado en tecnología para WWW, es especificar todas aquellas materias que permitan a los operadores y proveedores crear aplicaciones y servicios accesibles desde la mayor gama posible de plataformas.

WAPForum

El Foro WAP (Wireless Application Protocol Forum) es un grupo industrial dedicado a habilitar servicios de información y telefonía en dispositivos inalámbricos móviles. Estos dispositivos incluyen teléfonos móviles, pagers, PDAs y otros terminales inalámbricos. Basándose en la arquitectura del World Wide Web, el Foro WAP alinea su tecnología de manera muy próxima a Internet y la Web. La especificación WAP se extiende e influye en las tecnologías existentes (como los estándares de tratamiento digital de señal) y tecnologías de Internet (como IP, HTTP, XML, SSL, URL, etc.). Se fundó en junio de 1997 por Nokia, Ericsson, Motorola y Phone.com (formalmente Unwired Planet).

WAP Gateway

Dispositivo de dos direcciones a través del cual se convierte el contenido que hay en el WAP Server al formato WML que pueda entender nuestro teléfono WAP. Igualmente, a través de él y desde el punto de vista del servidor de HTTP, el WAP Gateway envía información acerca del dispositivo WAP por medio de los encabezamientos (headers) HTTP.

WAP Server

Básicamente es un servidor HTTP, es decir un servidor de la red Internet como otro cualquiera. En cambio, Nokia llama WAP Server a un conjunto de dos productos: un gateway WAP y un servidor HTTP (servidor de contenidos). Pero un WAP Server es un simple servidor HTTP.

WCDMA (Wideband CDMA)

La llamada telefonía sin cables de tercera generación (también referida con los servicios 3G) alargará significativamente la cantidad de opciones disponibles a los utilizadores, como por ejemplo: para permitir la comunicación, información y servicios de entretenimiento a través de los terminales sin cable.

WML Card (Carta WML)

Bloque de navegación WML, dentro de una baraja WML. Dentro de una baraja WML debe existir una o más cartas WML. Cuando el usuario accede a una baraja WML, se le presentará en la pantalla únicamente la primera de sus cartas.

WML Deck (Baraja WML)

Conjunto de cartas WML. Cuando se requiere un URL se carga automáticamente todo el conjunto de cartas, la carta que se especifique en el URL será a la que se acceda. Si no se especifica ninguna en particular, se accederá a la primera.

WMLScript

Lenguaje de programación con base en JavaScript,, hasta este momento, es limitada, pero permite al teléfono realizar cálculos, personalizar páginas WML o validar los datos introducidos por el usuario antes de enviarlos al servidor WAP. Es un lenguaje débilmente tipificado y que NO acepta objetos. Se compila en el servidor a un código intermedio denominado *bytecode* que es lo que se envía al teléfono cuando este lo requiere.

World Phone

Teléfonos con funcionalidad mundial ya que operan tanto en modos digitales GSM 900 y 1800 o 1900 MHz; D-AMPS 800 y 1900 MHz, como en el modo analógicos AMPS de 800 MHz.

WSP (Wireless Session Protocol)

Hace referencia a la aplicación de más alto nivel que ofrece WAP a través de un interface para servicios de dos sesiones. La primera consistiría en un servicio con conexión que operaría sobre el protocolo del nivel de transacción y el segundo sería sin conexión y operaría sobre el servicio de transporte de información.

WSP - B (Wireless Session Protocol - Browsing)

Este protocolo ha sido definido solo para la navegación.

WTLS (Wireless Transport Layer Security)

Capa de Seguridad de Transporte Inalámbrico. Está basado en el estándar SSL y sirve para establecer conexiones seguras que permiten una privacidad de datos esencial sobre todo en algunos tipos de operaciones como transacciones bancarias. Debido a que en muchas aplicaciones no es un factor importante la seguridad de los datos existen mecanismos de activación y desactivación de esta capa.

WTA (Wireless Telephony Applications)

Un entorno para aplicaciones de telefonía que permite a los operadores la integración de funciones de telefonía del propio dispositivo móvil con el micronavegador incorporado.

WTAI (Wireless Telephony Application Interface)

Es una interfaz utilizada en los terminales móviles para operaciones locales de control de llamadas (recepción, iniciación y terminación) y de acceso a líneas telefónicas.

WTP (Wireless Transaction Protocol)

Protocolo Inalámbrico de Transacciones. Proporciona un protocolo simplificado y diseñado para optimizar el ancho de banda que debe ocupar la información con la que trabaja. De esta manera es mayor el número de redes inalámbricas que pueden transmitir estos datos y de alguna manera independiza tanto el dispositivo como el mensaje de la red usada. WTP permite el envío de tres tipos de transacciones:

XML

Extensible Markup Language. El WML tiene su origen en este lenguaje.

NOTAS AL PIE

NOTAS AL PIE

1. El micro navegador actúa de interfaz con el usuario de la misma forma que lo hacen los navegadores estándar.
2. *Universal/Uniform Resource Identifier* ó Identificador Uniforme/Universal de Recurso
3. *Universal/Uniform Resource Location* ó Localización Universal/Uniforme de Recurso
4. *Wireless Telephony Application* ó Aplicación de Telefonía Inalámbrica
5. *Wireless Application Environment* ó Entorno Inalámbrico de Aplicación
6. *Wireless Markup Language*
7. *Wireless Session Protocol* ó Protocolo Inalámbrico de Sesión
8. *Wireless Transaction Protocol* ó Protocolo Inalámbrico de Transacción.
9. *Wireless Transport Layer Security* ó Capa Inalámbrica de Seguridad de Transporte
10. *Wireless Datagram Protocol* ó Protocolo Inalámbrico de Datagramas
11. Agentes de usuario: Un agente de usuario es todo aquel software o dispositivo que interpreta un contenido, p. e. WML. Esto incluye navegadores de texto, navegadores de voz, sistemas de búsqueda, etc.
12. *Wireless Session Protocol – Browsing*
13. Una primitiva de servicio representa el intercambio lógico de información entre la capa de Sesión y capas adyacentes.
14. Estas primitivas pueden ser de cuatro tipos, como se puede observar en la Tabla 2.
15. Protocol Data Unit
16. Service Data Unit
17. Transaction Identifier
18. Maximum Transfer Unit
19. Estas primitivas pueden ser de cuatro tipos, tal y como se puede ver en la Tabla 2.
20. Término utilizado para denominar el intercambio de primitivas entre cliente y servidor con el objetivo de establecer una sesión segura. Posteriormente veremos este intercambio de primitivas.
21. Estas primitivas pueden ser de cuatro tipos, como se puede observar en la Tabla 2.

BIBLIOGRAFÍA

BIBLIOGRAFÍA

1. Desarrollo Wap con WML y WMLScript

Be Forta, Keith.D.Lawver.

Ed.Anaya

2. Inside WAP:programming applications with WML and WMLScript

Pekka Niskanen

Ed.Prentice Hall

3. Professional Wap

Charles Arehart

Ed.Wrox

4. Programming Applications with the Wireless Application Protocol

Steve Mann

Ed.Wiley

CIBERGRAFÍA

1. <http://wapcolombia.ucauca.edu.co/index.php?opcion=2>
2. <http://www.wapforum.com>
3. <http://www.infowap.cl/web/serviwap/queeswap.htm>