Security of Wireless Application Protocol

Parminder Kaur
Lecturer, Electronics and Telecommunication Dept.,
Smt. Indira Gandhi College of Engg., Navi Mumbai

Abstract

The wireless application protocol (WAP) is a protocol stack for wireless communication networks. WAP uses WTLS, a wireless variant of the SSL/TLS protocol, to secure the communication between the mobile phone and other parts of the WAP architecture. This paper describes the security architecture of WAP and some important properties of the WTLS protocol. There are however some security problems with WAP and the WTLS protocol. Privacy, data protection and integrity are not always provided. Users and developers of WAP-applications should be aware of this.

Introduction

The primary means of communicating information of these days are voice and Internet. The unlimited accesses to Internet and sheer number of people connected to the Internet have made industry captain realize its potential. The industry now plans its marketing and communication strategies around the Internet. The wireless industry initially struggled within a number of issues like low bandwidth and low connection stability, to bring Internet to its users. They came together to form a common forum to tackle these issues. This forum is called the WAP. Wireless Application Protocol (WAP) is a worldwide standard for providing Internet communications and advanced services on digital mobile devices, such as handheld phones, pagers, and other wireless devices. This protocol is an open and global specification that allows users of the referenced digital devices to securely access and interacts with Internet, intranet, and extranet applications and services. The WAP Architecture Specification is intended to present the system and protocol architectures essential to achieving the objectives of the WAP Forum. The WAP Architecture Specification acts as the starting point for understanding the WAP technologies and the resulting specifications.

Architectural Goals

The goals of the WAP Forum architecture are as follows.

Provide a web-centric application model for wireless data services that utilizes the telephony, mobility, and other unique functions of wireless devices and networks and allows maximum flexibility and ability for vendors to enhance the user experience.

1. Enable the personalization and customization of the device, the content delivered to it, and the presentation of the content.
2. Provide support for secure and private applications and communication in a manner that is consistent and interoperable with Internet security models.
3. Enable wireless devices and networks that are currently or in the near future being deployed, including a wide variety of bearers from narrow-band to wide-band.
4. Provide secure access to local handset functionality.
5. Facilitate network-operator and third party service provisioning.
6. Define a layered, scaleable
7. Extensible architecture.
8. Leverage existing standards where possible, especially existing and evolving Internet standards.

Architecture Overview

The World-Wide Web Model

The Internet World-Wide Web (WWW) architecture provides a very flexible and powerful programming model (Fig. 1). Applications and content are presented in standard data formats, and are browsed by applications known as web browsers.

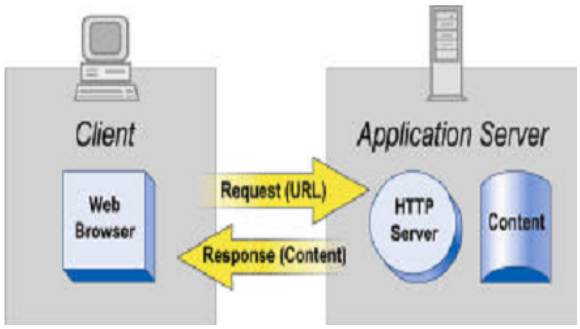

Fig. 1: World Wide Web Programming Model

The web browser is a networked application, i.e., it sends requests for named data objects to a network server and the network server responds with the data encoded using the standard formats.

The WAP Model

The WAP programming model (Fig. 2) is the WWW programming model with a few enhancements. Adopting the WWW programming model provides several benefits to the application developer community, including a familiar programming model, a proven architecture, and the ability to leverage existing tools. Optimisations and extensions have been made in order to match the characteristics of the wireless environment. Wherever possible, existing standards have been adopted or have been used as the starting point for the WAP technology.

The most significant enhancements WAP has added to the programming model are:

1. Push
2. Telephony Support (WTA)

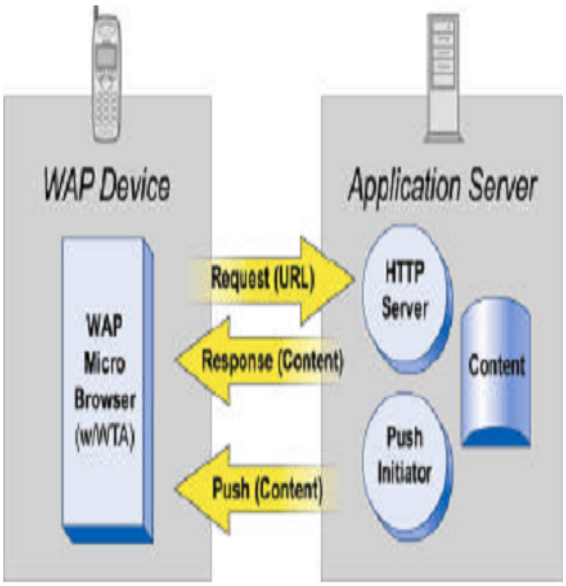

Fig. 2: WAP programming Model

Feature/Performance-Enhancing Proxies

WAP utilizes proxy technology to optimize and enhance the connection between the wireless domain and the WWW. The WAP proxy may provide a variety of functions, including:

1. Protocol Gateway – The protocol gateway translates requests from a wireless protocol stack (e.g., the WAP 1.x stack—WSP, WTP, WTLS, and WDP) to the WWW protocols (HTTP

and TCP/IP). The gateway also performs DNS lookups of the servers named by the client in the request URLs.

2. Content Encoders and Decoders – The content encoders can be used to translate WAP content into a compact format that allows for better utilisation of the underlying link due to its reduced size.
3. Caching Proxy – A caching proxy can improve perceived performance and network utilization by maintaining a cache of frequently accessed resources.
4. User Agent Profile Management – User agent profiles describing client capabilities and personal preferences [UAProf] are composed and presented to the applications.

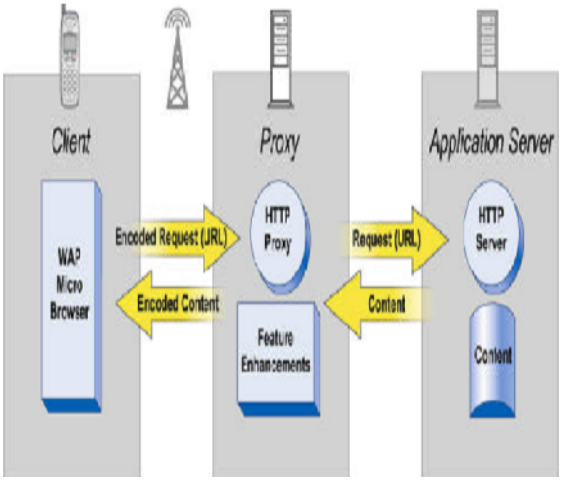

Fig. 3: Feature/Performance Enhancing Proxies

Security Model

WAP enables a flexible security infrastructure that focuses on providing connection security between a WAP client and server.

WAP can provide end-to-end security between protocol endpoints. If a browser and origin server desire end-to-end security, they can communicate directly using the security protocols. Moreover, the WAP specifications include support for application-level security, such as signed text.

(i) WAP 1.1 security

The main security initiative in WAP 1.1 is the Wireless Transport Layer Security protocol ('WTLS'). WTLS provides similar functionality to that of the Internet's transport layer security

v1.0 protocol ('TLS'), the IETF's standard for securing Internet browsing and this, in turn, is based on Secure Sockets Layer ('SSL') v3.0 -Internet Protocol.

However, compared to traditional TLS/SSL, WTLS provides faster algorithm processing (by minimizing protocol overhead), enables

more data compression and provides the added functionality of datagram support[1], Optimized handshake[2] and dynamic key refreshing.

1- TLS/SSl cannot operate over UDP, whereas WTLS can.

2- For optimized handshakes in contrast to the full handshakes, the server obtain the client certificate from a certificate distribution service or from its own service, without requesting it over the air from the client.

WTLS provides a robust, efficient basis for secure transaction and support data integrity, authentication and privacy services between communicating applications (WAPF-D). A summary of each security service and corresponding WTLS security mechanism is outlined in the table below.

Security requirement	**Security Mechanism**
Confidentially privacy.	Secret key cryptography using bulk ciphers, such as RC5_CBC, DES_CBC, 3DES_CBC, IDEA
Authentication/ authorization and non-repudiation	Public key cryptography using key exchange suites, such as RSA, Diffie-Hellman, Elliptic Curve Diffie-Hellman
Integrity	MACs-HMAC based (e.g. SHA-1,MD5) or XOR based (for example SHA-1)

(ii) WAP 1.2 security

To address the lack of both non-repudiation services and real end-user authentication in WAP 1.1, the WAP Forum introduced two new initiatives in WAP 1.2:

1. The WMLScript Crypto Library- provides application layer security by the use of WMLScript applets to enable cryptography signing of WML content. These applets run on the client and are stored within a WMLScript crypto library.
2. The WAP library Module ('WIM')- 'a temper resistant[3] device which is used I performing WTLS and application level security functions, and especially, to store and process information needed for user identification and authentication. These two initiatives are supported by a wireless Public Key Infrastructure (PKI), which provides the functions that store and process information needed for user identification and authentication (WAPF-B).

(iii) WAP 2.0 Security

A much publicized criticism of WAP 1.x, and a primary cause of reluctance to adopt the protocol, is the lack of end-to-end security and end-to-end authentication. This is due to the presence of a WAP gateway, which effectively acts as a bridge between the WAP and standard Internet Protocols (WTLS and TLS/SSL) and markup languages (WML[4]/WMLScript[5] and HTML[6]/JavaScript). As part of the translation process, data is momentarily present in plaintext and it is this 'gap' in security that can, potentially, pose a serious risk.

Alternatively, vendors with strenuous security requirements could host their own gateway. This would be within the vendor's own network environment and would be under their control and security measures. Data, encrypted by WTLS, would pass directly between the client and the vendor's gateway and would then pass through the vendor's network to their web server. This would, effectively, provide a form of end-to-end security. However, in practice this is a very large overhead for vendors. Also, the majority of WAP phones are sold with the mobile operator's gateway setting pre-loaded. It can be complicated and frustrating exercise to change that setting and some operators prevent users from accessing any gateway other than their own.

WAP 2.0 Architecture

In the previous versions of the WAP specifications, a new set of protocols, collectively known as WAP 1.x stack, were created to facilitate the transfer of data along low-bandwidth mobile networks to constrained devices. With the emergence of high-speed wireless networks (e.g.

2.5G, 3G) and improvement in device technology, appropriate IP connectivity can now be achieved between the device technology, appropriate IP connectivity can now be achieved between the device and Wireless network. WAP 2.0 takes advantage of this by introducing IP directly into the WAP environment.

The previous WAP 1.x stack and proposed WAP 2.0 stacks are detailed in figure 4

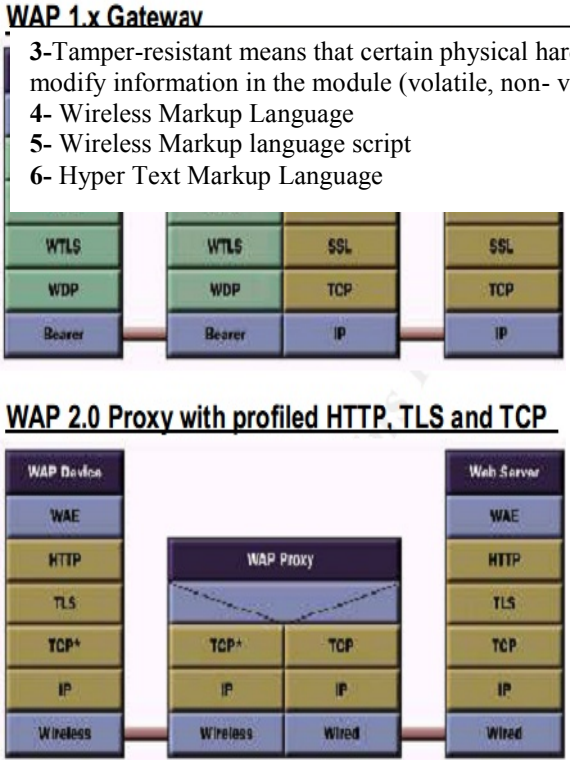

Fig. 4: WAP 1.x stack versus WAP 2.0 stack

WAP 1.x uses the lightweight markup language WML (which is subset of XML and is similar to the internet's HTML, but optimized for use in handheld mobile devices). WAP 2.0 introduces WML2, which is based profile of XHTML, developed by W3C to replace and enhance the use of HTML language commonly used today.

Conclusion

WAP 2.0 continues to support the original WAP 1.x stack and by encompassing both stacks, provides backwards compatibility. Although this has the advantage of enabling connectivity over a broader range of networks and wireless bearers, the wireless environment is now exposed to security issues related to the internet protocols as well as those related to WAP 1.xandthe WAP 1.x architecture. Some of the weaKness in WTLS are also present in the current TLS/SL used over the internet, and will therefore also apply to the profiled TLS used by WAP 2.0. These include: no obligation to exchange certificates; no obligation to verify certificates and authenticate owners; and the permitting of anonymous diffie-Hellman mode (where exchanges are not supported by public key certificates) and may allow man-in-the-middle attacks.

References

1. N. Sklavos, A. P. Fournaris and O. Koufopavlou, "WAP Security: implementation Cost and Performance Evaluation of a Scalable Architecture for RC5 Parameterized Block Cipher"
2. "Wireless Application Protocol 2.0 Security" by SANS Institute.
3. Joris Claessens. Analysis and design of an advanced infrastructure forsecure and anonymous electronic payment systems on the Internet. PhD thesis, Katholieke Universiteit Leuven, December 2002. 220 pages.
4. Tim Dierks and Eric Rescorla. The TLS Protocol Version 1.1. IETF Internet Draft, March 2003.
5. Ric Howell (Concise Group Ltd). WAP security. http://www.vbxml.com/wap/articles/wap-security/default.asp.
6. Open Mobile Alliance. http://www.openmobilealliance.org/
7. Markku Saarinen. Attacks Against The WAP WTLS Protocol. http://www.cc.jyu.fi/~mjos/wtls.pdf.
8. Michael Wiener. Performance Comparison of Public-Key Cryptosystems. RSA Laboratories' CryptoBytes, 4(1):1–5, Summer 1998.
9. Wireless Application Protocol Forum. WAP Wireless Transport Layer Security. http://www1.wapforum.org/tech/docu-ments/WAP-261-WTLS-20010406-.pdf.

10. Wireless Application Protocol Forum. WAP WMLScript Crypto Library. http://www1.wapforum.org/tech/documents/WAP-161-WMLScriptCrypto-20010620-a.pdf.
11. Durham-Vichr, Deborah & Getgen, Kimberly (2001), 'WAP 2.0 Securing the Internet without Wires', IBM, August 2001. http://www.106.ibm.com/developerworks/wireless/library/wisectrends/