The Wireless Application Protocol

Dave Singelée and Bart Preneel
(Corresponding author: Dave Singelée)

ESAT-COSIC, K.U. Leuven, Belgium
Kasteelpark Arenberg 10, B-3001 Leuven-Heverlee, Belgium (Email: Dave.Singelee@esat.kuleuven.ac.be)

(Received June 29, 2005; revised and accepted July 31, 2005)

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. In this paper, we address the security weaknesses of WAP and WTLS and propose some countermeasures and good practices when using WAP. We conclude with advising when to use WAP and when not.

Keywords: Cryptography, mobile networks, security, WAP, WTLS

1 Introduction

In the modern society, information and access to information is getting more and more important. During the last couple of years, there is a strong tendency towards mobility. This implies an increasing need for being online and having access to information all the time. Browsing on the Internet is not restricted anymore to desktop computers, people can now also use their mobile phones or PDA. This is done by WAP, the Wireless Application Protocol. WAP is a protocol stack for wireless communication networks, specified by the WAP forum. The WAP forum is currently part of the Open Mobile Alliance [4]. WAP is essentially a wireless equivalent to the Internet protocol stack (TCP/IP). A big advantage of WAP is that it is bearer independent. The most common bearer is currently GSM, but also a PDA or a third generation mobile phone can be used. In the rest of the paper, we will assume that a mobile phone is used to browse on the Internet.

1.1 Wireless Mark–up Language

Just as for the WWW, the user interface to WAP is via a mini browser in the mobile phone. WAP has its own Mark-up Language WML (*Wireless Mark–up Language*). WML is the WAP equivalent of HTML. WML also includes scripting (WMLScript, which is roughly equivalent to JavaScript). It also provides digital signature functionality through the WMLScript Crypto Library [8], which is similar to Netscape's Javascript signing.

1.2 Wireless Transport Layer Security

As will be discussed in Section 2.1, all the communication from the mobile phone to the Internet passes through the WAP gateway. The communication between the mobile phone and this WAP gateway has to be secured. The SSL/TLS protocol [2] can not be used for this purpose because of the constraints of the mobile phone. A mobile phone has very limited bandwidth, memory, computational power and battery power and can not perform heavy (cryptographic) computations (e.g., public key cryptography with a 2048–bit key). That is why the WAP forum has adapted TLS to make it suitable for a wireless environment with small mobile devices. The result is WTLS (*Wireless Transport Layer Security*), the wireless variant of SSL/TLS. WTLS includes the usage of elliptic curve cryptography (*ECC*) by default. The advantage of elliptic curve cryptography is that is uses keys with a much smaller size (ECC with a key size of 170–180 bits is estimated to achieve a 1024-bit RSA level of security [6]). WTLS does also work on top of a datagram-based instead of a connection-based communication layer. Finally, WTLS defines its own certificate format optimized for size (limited bandwidth), but supports the ordinary X.509 certificate too [1].

1.3 Organization of the Paper

This paper consists of five sections. In this section, we have given a very short introduction to WAP and WTLS. Section 2 describes the WAP security architecture and the

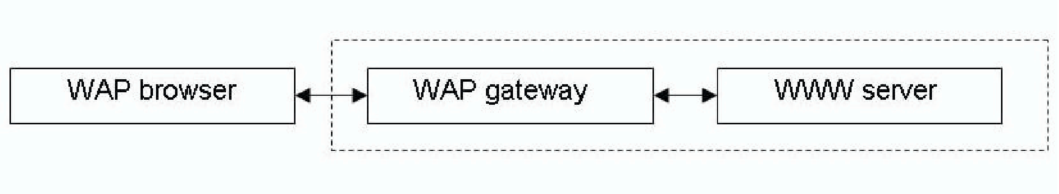


Figure 1: WAP architecture

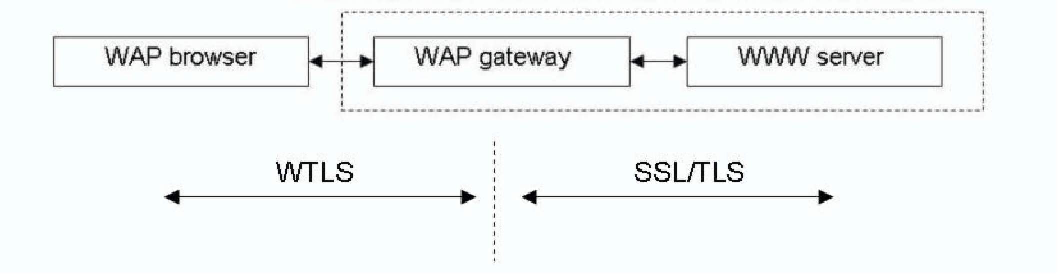


Figure 2: WTLS-traffic gets translated to SSL/TLS–traffic

WAP protocol stack. It also explains some basis issues of the WTLS protocol and the Wireless Identity Module (*WIM*). There are however some security problems with WAP and WTLS. The most important ones are discussed in Section 3. Some solutions and good practices how to use WAP are proposed in Section 4. Finally, Section 5 concludes the paper.

2 How Does WAP Work?

WAP has a very specific architecture which has some security consequences. This architecture will be discussed first. Then, the WAP protocol stack, the WTLS protocol and the Wireless Identity Module will be explained.

2.1 WAP Architecture

Figure 1 shows the basic WAP architecture [1]. There are three participating entities: the WAP browser, the WAP gateway (also called *WAP proxy*) and a server on the Internet.

When the mobile device wants to connect to the Internet, all the communication passes through the WAP gateway. This WAP gateway translates all the protocols used in WAP to the protocols used on the Internet. For example, the WAP proxy encodes (and decodes) the content to reduce the size of the data that has been sent over the wireless link. Another example is the WTLS protocol. The communication between the mobile device and the WAP gateway is secured with WTLS. WTLS is only used between the mobile device and the WAP gateway, while SSL/TLS can be used between the gateway and the Internet. This means that the WAP gateway first has to decrypt the encrypted WTLS–traffic and then has to encrypt it again (using SSL/TLS), as shown in Figure 2. This has some security consequences, which will be discussed in Section 3.

2.2 WAP Protocol Stack

Many years ago, a theoretical protocol stack was developed by the OSI (*Open Systems Initiative*). This was done to facilitate a common understanding of the functionality provided by a protocol stack and to facilitate comparisons between different vendor's implementations. The mapping of the WAP protocol stack to the OSI model is shown in Figure 3.

The WAP protocol stack contains the following elements [3]:

- **Physical and Data Link Layer:** In WAP, Point to Point Protocols (*PPP*) are used over one or more Over–The–Air (*OTA*) bearer protocols.

- **Network Layer:** IP is the network layer of choice. However, not all wireless networks are capable of transmitting IP. That is why SMS or some other non-packet network protocol can be used.

- **Transport Layer:** The protocol used in the transport layer is UDP. However, this may not be feasible over non–IP networks. That is why (there are also other reasons) that WAP defines an additional transport layer protocol, WDP, which can be used when UDP can not.

- **Session Layer:** The functionality of the session layer is partially included in WTP. Other aspects of the functionality are implemented in WSP.

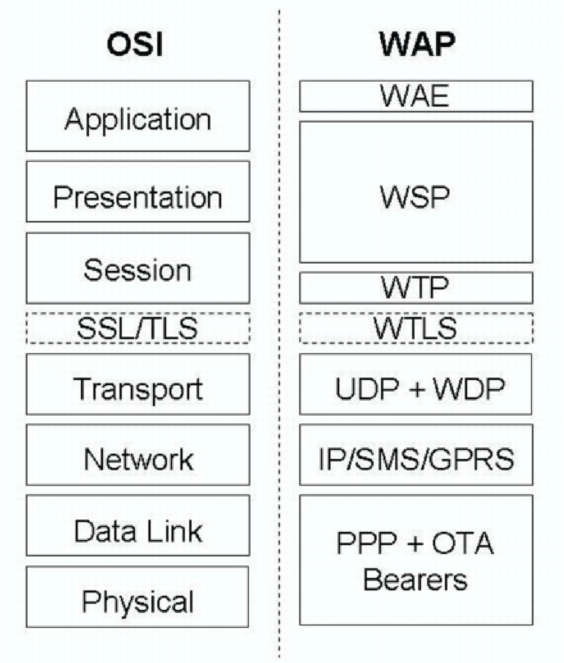


Figure 3: WAP protocol stack

- **Presentation Layer:** The functionality of the presentation layer is included in WSP.
- **Application Layer:** Some aspects of the functionality of the application layer are included in WSP, the others are implemented in WAE.

2.3 WTLS: Basic Issues

WTLS, the Wireless Transport Layer Security protocol, operates just above the transport layer in the OSI protocol stack, as can be seen in Figure 3. Explaining all the details of WTLS would take hundreds of pages. That is why we will only discuss some basic issues of WTLS. More details can be found in [7].

WTLS establishes a session between a client (the mobile phone) and a server (the WAP gateway). This phase is called the *handshake phase*. During this handshake phase, security parameters used to protect the session are negotiated. These include the encryption protocols, signature algorithms, public keys, pre-master secrets, ... WTLS includes support for both a full handshake, with negotiation of all security parameters, and for a lightweight handshake in which the security parameters of another session are reused.

Once a session has been established, all the communication between the client and the server is encrypted. WTLS also support the feature to suspend a session and resume it later. This way, sessions can last for days. The longer the session remains valid, the longer the secret keys remain valid, and thus, the higher the probability for an attacker to find a secret key. That is why WTLS allows keys to be renegotiated during a session.

WTLS also uses certificates. Because certificates were not really designed to be used by mobile devices, WAP defines a new format of certificate that is optimized for storage on mobile devices and transmission over wireless networks. These certificates have the same functionality as ordinary X.509 certificates, but rely on the server to perform more of the processing under some circumstances.

2.4 Wireless Identity Module (WIM)

WAP devices use a Wireless Identity Module (*WIM*) which contains the necessary private and public keys to perform digital signatures and certificate verification respectively. It is a tamper-proof device, which means that it is very difficult for an attacker to obtain the keys which are stored in this device. The WIM can be compared to the SIM of the GSM.

3 Security Problems with WAP

There are some security problems with WAP. The most important threat associated with WAP is the use of the WAP gateway. There are however also some security weaknesses in the WTLS protocol and some possible threats by using mobile devices. The most important security problems will now be discussed.

3.1 WAP Gateway

WAP does not offer end–to–end security. As explained in Section 2.1, WAP devices communicate with web servers through an intermediate WAP gateway. WTLS is only used between the device and the gateway, while SSL/TLS can be used between the gateway and the web server on the Internet. This means that the WAP gateway contains, at least for some period of time, unencrypted data (which can be highly confidential). The gateway vendors have to take steps to ensure that the decryption and re–encryption takes place in memory, that keys and unencrypted data are never saved to disk, and that all memory used as part of the encryption and decryption process is cleared before handed back to the operating system. But how sure can you be that this happens, there are no standards or guarantees about these precautions? How do you know that the WAP gateway prevents the operating system from swapping memory pages out to swap space ...?

The problem is even worse! The WAP architecture implicitly assumes that the user of the mobile phone (and the web server) trust the WAP gateway. All the (sensitive) data gets unencrypted by the WAP gateway. This means that in sensitive services, such as for example electronic banking, the bank should not rely on the client's default (and untrusted) WAP gateway! A solution for this problem is proposed in Section 4.

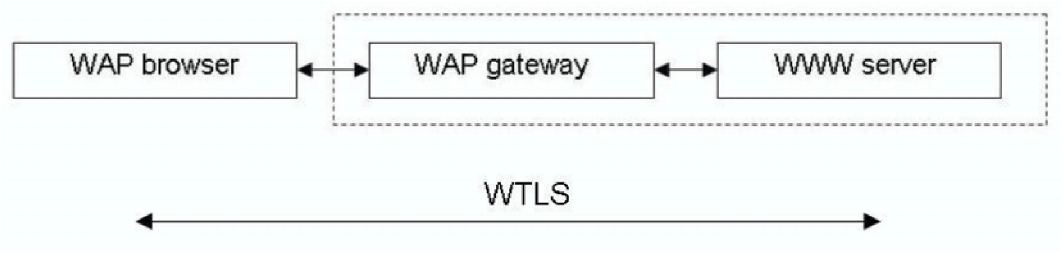


Figure 4: WAP in passthrough mode

3.2 WTLS Allows for Weak Encryption Algorithms

The encryption protocol used to encrypt data during a WTLS session is negotiated in the handshake phase. There is the possibility to choose the 40-bit DES encryption method. In this method, a 5 byte key is used which contains 5 parity bits. This means that there are only 35 effective key bits in the DES key. It is very easy to find this DES key by a brute force attack. A 40-bit DES encryption is a very weak encryption algorithm!

3.3 Predictable IVs

The WTLS protocol should be able to operate above an unreliable transport layer, so datagrams may be lost, duplicated or reordered. If CBC–encryption mode is used, this means that it is necessary for the IV (*Initial Value*) to be contained in the packet itself or that the IV for that block can be derived from data that is already available to the recipient. WTLS always uses a linear IV computation. When a block cipher is used in CBC mode, the IV for encrypting each packet is computed as follows:

$$IV_s = IV_0 \oplus (s|s|s|s)$$

In this formula, s is a 16-bit sequence number of the packet and IV_0 is the original IV, derived during key generation.

When CBC mode is used in combination with a terminal application where each keypress is sent as an individual packet (such as telnet), this can give problems when low–entropy secrets (such as passwords) are entered in the application. An attacker can guess every character of the password and can immediately check if his guess was correct. This makes it very easy to perform a brute force attacks. Details of this attack can be found in [5].

3.4 Potential for Viruses

Mobile phones are getting more and more advanced and have a sophisticated operating system. Furthermore, WAP contains a scripting language (*WMLScript*). This makes it easier for viruses to affect a mobile phone. What makes it even more dangerous is that it is not possible to run sophisticated anti–virus software on a mobile phone. Viruses for mobile phones have already appeared, but are not yet widespread. But experts agree that it is just a matter of time before they will strike massively.

3.5 Physical Security

The weakest link of the system will be the mobile phone itself. It easily gets lost[1] or stolen and it is likely to be used more and more for the storage of sensitive data. The PIN code offers some protection, but it only consists of 4 digits and most users choose weak PINs (e.g., 1234 or 0000). If one makes a risk analysis of WAP, then the physical security of the mobile phone certainly has to be considered too!

4 Solutions and Good Practices

Nevertheless the security problems of WAP, there are some easy solutions and good practices to use WAP more securely.

The first solution is to switch to a **trusted and secure gateway** instead of using the default WAP gateway. This is important in sensitive services like electronic banking applications. The problem with this solution is that it is not always very easy for a (non–technical) user to switch to another gateway. Note that if WAP is deployed over GSM, switching from one gateway to another can be done by sending a SMS message. Another possibility would be to change the gateway automatically on request of the target web server.

Another solution is to **upgrade all WAP gateways** such that they can work in *passthrough mode*. When a WAP gateway works in this mode, it just lets pass all the traffic untouched (see Figure 4). In this way, the WTLS encrypted data stream travels from the mobile phone to the server without being decrypted and the gateway would just be a relay for the data stream. A WAP gateway would have two modes. When it is in normal mode, it just works like a WAP gateway works today. When the WAP gateway detects a WTLS stream, it changes to passthrough mode and simply lets the data stream pass through to the web server. Upgrading all WAP gateways

[1] British Railways publish every year a list of the most popular object to get lost, and the mobile phone occupies the first place in this list (some years ago, it was the umbrella which was the most popular object to get lost).

and WAP servers (they have to "understand WTLS" in this solution) is much easier than upgrading all WAP devices!

There are also some **good practices** when using WAP. It is a good idea not to use WAP in very sensitive services and to make sure that your system does not support weak encryption algorithms. Also use a secure PIN number (so not 1234) to protect your mobile phone from being misused in case it has got lost.

5 Conclusion

WAP enables mobile phones to browse on the internet. It is the wireless equivalent to TCP/IP and has the big advantage of being bearer independent. The security architecture of WAP consists of three parts: the mobile phone, the WAP gateway and the Internet. The communication between the mobile phone and the gateway is protected by WTLS, a wireless version of SSL/TLS, while the traffic from the gateway to the Internet can be protected by SSL/TLS. The WAP gateway decrypts all the WTLS traffic and encrypts all the SSL/TLS traffic. From a security point of view, this means that the gateway should be considered as an entity–in–the–middle. It is due to this fact that both the user and the web server on the Internet have to trust the WAP gateway. As this is not always the case, solutions have been searched for to avoid this entity–in–the–middle. All these solutions have some disadvantages: the user has to configure his own system (choose the WAP gateway) or all the WAP gateways and servers have to be upgraded. There is a need for easier solutions!

Until better solutions are found, it is a good idea to be cautious when using WAP. When you want to execute some sensitive application (like electronic banking), it is maybe a good idea not to use WAP. For other applications, WAP is a nice and ingenious technology!!

References

[1] J. Claessens, *Analysis and Design of an Advanced Infrastructure for Secure and Anonymous Electronic Payment Systems on the Internet*, PhD thesis, Katholieke Universiteit Leuven, Dec. 2002. (220 pages)

[2] T. Dierks and E. Rescorla, *The TLS Protocol Version 1.1*, IETF Internet Draft, Mar. 2003.

[3] R. Howell (Concise Group Ltd), *WAP Security*, `http://www.vbxml.com/wap/articles/wap-security/default.asp`.

[4] Open Mobile Alliance, `http://www.openmobilealliance.org/`.

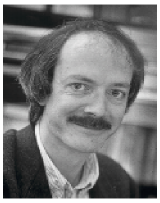
[5] M. Saarinen, *Attacks Against The WAP WTLS Protocol*, `http://www.cc.jyu.fi/~mjos/wtls.pdf`.

[6] M. Wiener, Performance Comparison of Public-Key Cryptosystems, *RSA Laboratories' CryptoBytes*, vol. 4, no. 1, pp. 1–5, Summer 1998.

[7] Wireless Application Protocol Forum, *WAP Wireless Transport Layer Security*, `http://www1.wapforum.org/tech/documents/WAP-261-WTLS-20010406-a.pdf`.

[8] Wireless Application Protocol Forum, *WAP WMLScript Crypto Library*, `http://www1.wapforum.org/tech/documents/WAP-161-WMLScriptCrypto-20010%620-a.pdf`.

Dave Singelée was born on April 20, 1979. He received the Master's degree of Science in Electrical Engineering from the Katholieke Universiteit Leuven (Belgium) in July 2002. He is currently a phd student at COSIC (Computer Security and Industrial Cryptography, Katholieke Universiteit Leuven, Belgium). His main research interests are cryptography, the security architecture of Wireless Personal Area Networks (WPAN) and wireless network security in general. `http://www.esat.kuleuven.ac.be/cosic/`



Bart Preneel received the Master degree in Electrical Engineering and the Doctorate in Applied Sciences in 1987 and 1993 respectively, both from the Katholieke Universiteit Leuven (Belgium). He is a professor at the Electrical Engineering Department of the Katholieke Universiteit Leuven and visiting professor at the T.U.Graz in Austria. Together with Prof. J. Vandewalle, he is heading the research group COSIC at the K.U. Leuven, which currently has 35 members. He has held visiting professor positions at the Ruhr-Univ. Bochum (Germany), at the Univ. of Bergen (Norway) and the Univ. of Ghent (Belgium). He was also a research fellow at the EECS Department of the University of California at Berkeley. His main research interests are cryptology and information security. He has authored and co-authored more than 180 articles in international journals and conference proceedings. He is Vice President of the International Association of Cryptologic Research (http://www.iacr.org) and Chairman of the Leuven Security Excellence Consortium (http://www.l-sec.be). Currently he is project manager of ECRYPT (http://www.ecrypt.eu.org) the EU-funded European Network of Excellence on Cryptology and Watermarking. In 2003, he has received the European Information Security Award in the area of academic research. He is a member of the Editorial Board of the Journal of Cryptology and of the ACM Transactions on Information Security.