MODIFIED WAP FOR SECURE VOICE AND VIDEO COMMUNICATION

N. Komninos, B. Honary

Lancaster University (UK)

ABSTRACT

It is essential for future mobile communication systems to support different media types such as audio, video. There is also an increasing need to preserve the confidentiality, data integrity, and authenticity of these media to prevent fraud. In this paper, modifications to the wireless application protocol (WAP) are proposed to enable real time voice and video transmission in future mobile communication systems through a secure communication protocol.

INTRODUCTION

Future mobile systems will integrate satellite, radio, infrared and other means of communication to provide a whole host of upgraded services to those available today (i.e. voice, text and data), by utilizing the mobile network capability for high-speed information transfer. Thus, WAP will provide services like PC2call, net2phone, which support audio and video communication. Nowadays, GPRS data calls support high data rates, which provide an increase in quality of service for audio and video. Therefore, in future mobile systems audio and video conference can be established through data calls.

Wireless Application Protocol (WAP) provides a universal open standard for bringing Internet content and advanced services to mobile phones and other wireless devices. A mobile phone and gateway are the main components of WAP. Nowadays, whenever a mobile phone uses WAP a connection is created via wireless session protocol (WSP) between the mobile and the gateway. When the user enters the address of the WAP site the gateway is sent a request for the device's micro browser using WSP. The gateway translates the WSP request into a hypertext transfer protocol (HTTP) request and sends it to the appropriate origin server (or web server). The web server then sends back the requested information to the gateway via HTTP. Finally, the gateway translates and compresses the information, which can then be sent back to the micro browser in the mobile phone.

Communication between different networks and devices increases security issues. It is essential to preserve the confidentiality, data integrity and authenticity of data to prevent fraud. Nowadays, a WAP gateway handles secure sessions according to figure 1.

A secure socket layer (SSL) session is opened between the web server and the WAP gateway and a wireless transport layer security (WTLS) session is initialised between the gateway and the mobile device. The encrypted content is sent from the server to the gateway, which translates it and sends it to the mobile phone.

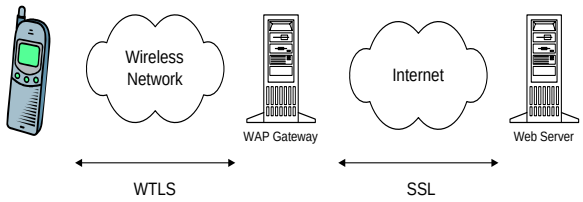

Figure 1 – Present Security Structure

SECURITY IN WAP

WTLS is an optional layer based on Transport Layer Security (TLS) that operates over the transport layer (7). WTLS supports either reliable or unreliable secure connections. Mainly, WTLS attempts to reduce the overheads associated with establishing a secure connection between two applications. It provides the same grade of security that is supplied by the Secure Socket Layer (SSL) while reducing the transaction times. It provides services that ensure privacy, server authentication, client authentication and data integrity. Mobile phone and WAP gateway terms will be referred as client and server respectively from now on.

According to the WAP specifications, WTLS is composed by the record protocol, which is a layered protocol. The WTLS record protocol takes messages to be transmitted, optionally compresses the data, applies MAC, encrypts, and transmits the data. Received data is decrypted, verified and decompressed, then delivered to higher-level layers. Four record protocols exist according to the specifications: the change cipher spec protocol, the handshake protocol, the alert protocol, and the application data protocol.

The cipher spec protocol deals with ciphering strategies and consists of a single message [5]. The cryptographic parameters of a secure session are

produced by the handshake protocol. The alert protocol is used to knowledge to the client and server that the secure connection is ending. It also determines the level of error in the secure connection and provides a description of the error to the client and server in the termination of the connection. Finally, the client and server exchange user data using the application data protocol.

The client sends a hello message to which the server must respond with a hello message, or else a fatal error will occur and the secure connection will fail. The client hello and server hello are used to establish security enhancement capabilities between client and server such as, protocol version, key exchange suite, cipher suite, compression method, key refresh and sequence number mode. Additionally, two random values are generated and exchanged to the client and server.

Following the hello messages the server will send its certificate, if it is to be authenticated. Additionally, a server key exchange message may be sent, if it is required. The server may request a certificate from the client if that is appropriate to the key exchange suite selected. Then, the client responds with the requested information to the server. At this point, the client and server may begin to exchange application data.

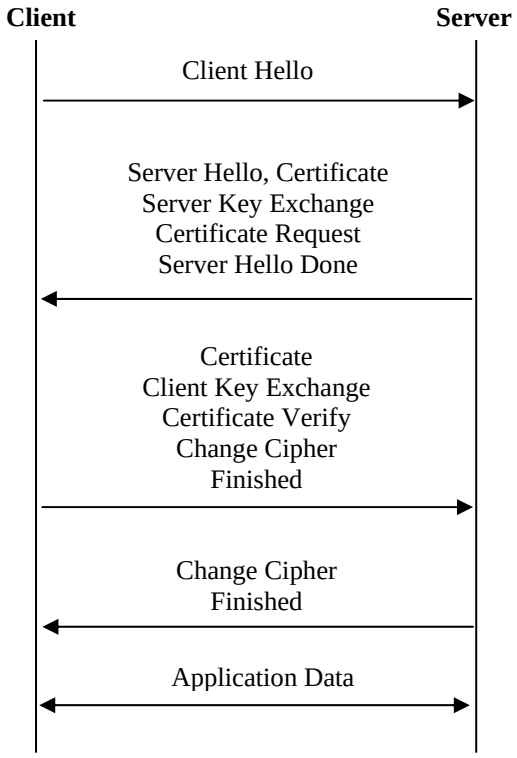

Figure 2 – WTLS Secure Session

MODIFIED WAP FOR SECURE AUDIO/VIDEO

When a WAP session is created between the server and the client, it is agreed the session will be either connectionless or connection oriented since both services are suitable for data transmission. However, when different data types are supported by WAP such as audio and video, it is necessary that the contents of the data should be known to the server. In this case, the gateway can enable or disable some functionality to the different media types. For example, when audio or video is transmitted, translation of HTML to WML is not required. It is essential, to provide the least amount of latency possible for continuous media types.

In a session establishment a media type message (MTM) is also exchanged between the client and server. MTM contains source and destination addresses and ports, data type, network performance, and group id parameters as shown in Table 1.

Parameters				
	Req	Ind	Res	Cnf
Source Address	M	M	-	-
Source Port	M	M	-	-
Destination Address	M	O	-	-
Destination Port	M	O	-	-
Network Performance	M	O	M	M
Data Type	M	O	-	-
Group ID	M	O	-	-

Table 1 - Media Type Message

The source and destination addresses identify the peers with which the session is to be established. Similar, source and destination ports identify the port to which the message is sent/received. The data type parameter as the name implies, identifies continuous and non-continuous data types. The network performance parameter is used to evaluate the quality of service the network requires to support different data types. For example, when the available bandwidth is not enough for the quality of service required in a real time audio or video transmission then a session is not established. Finally, the group id identifies group calls for each session.

When a session is opened it will be either connection-oriented or connectionless. A connection-oriented session provides confirmed services used to manage a session and to transmit reliable data where clients can then send request messages and receive confirmation. A connectionless session provides only non-confirmed services where clients can only use request messages and servers are only able to use indication messages.

In a session request the network performance and data type parameters can be used to determine which service is suitable for the session. For example, when text or image is transmitted then a session can be either connection oriented or connectionless but when audio or video is transmitted a session can only be

connectionless. The type of service also enables if a session can be suspended or not.
When a session is established the group id parameter is also assigned. The group id, as already mentioned, identifies group calls that can be established later on in the session. When a third party wants to join a secure session, the media type message is exchanged between the client and server. The group call is initiated when the group id requested is the same with the one used in the session. Thus, the joint session is established when the network performance and the data type parameters agree with those used in the two party session. For example, if there is available bandwidth for a group call, a joint session can be established. Then, the cryptographic parameters of the secure joint session can be produced by the handshake protocol. Finally, a secure joint session can be suspended, when for example a data circuit in the underlying bearer network is closed. Similarly, a secure joint session can be resumed using the media type message when the data circuit in the bearer network is opened again.

When the mode of encryption (i.e. OFB, CBC, CFB etc.) is decided in a session of two or more parties encryption synchronisation is required. For example, if used as stream ciphers, output feedback mode (OFB) and cipher feedback mode (CFB) require synchronisation when encrypted data is transmitted over a noisy channel. Initially, the receiver is synchronised to an incoming encrypted data stream. During transmission, the receiver must maintain synchronisation to the incoming encrypted data stream after the initial synchronisation. Encryption synchronization is achieved by periodically sending the synchronization message to the receiver. The synchronisation message includes the following parameters: source and destination addresses and ports, group id, algorithm number, and an initialisation vector as shown in Table 2.

Parameters		
	Req	Ind
Source Address	M	M
Source Port	M	M
Destination Address	M	O
Destination Port	M	O
Group ID	O	O
Algorithm Number	M	M
Initialisation Vector	M	O

Table 2 - Synchronisation Message

The source and destination addresses identify the peers to which the message is sent/received. Similarly, source and destination ports identify the port to which the message is sent/received. Group id is used to indicate the secure session(s), which will be synchronised. Furthermore, the algorithm number indicates the encryption algorithm, if multiples are supported, per session. Finally, the initialisation vector is used to update the IV of the receiver to the incoming encrypted data stream.

Encryption synchronisation is particularly required in continuous media such as audio or video. For non-continuous media such as image and text files encryption synchronisation is not necessary because files can be encrypted before transmitted since ARQ schemes are used and no unrecoverable errors are assumed.

SIMULATION

The simulation is a two-step process; firstly a WAP model is built, which consists of a mobile client, a WAP gateway, and a web server. Secondly real time audio and video in WAP is enabled using a secure communication protocol (SCP) (10).

The mobile station, the WAP gateway, and the web server were built using JAVA applets for the graphical user interface as shown in Figure 3. The communication between the WAP components was established using JAVA sockets. The mobile client contains the keypad of a mobile phone and acts as a client/server model. The WAP gateway also acts as a client/server model, which translates WSP requests into HTTP requests and sends them to the appropriate web server. Finally, the web server also acts as a client/server model, which responds to and requests data from the WAP gateway.

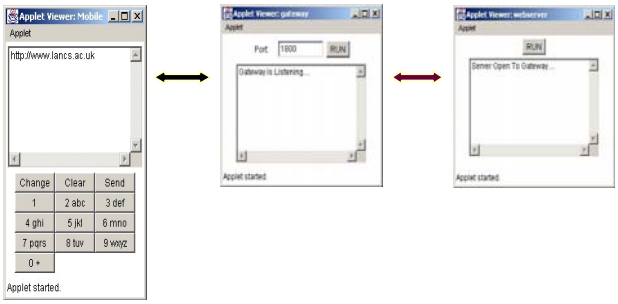

Figure 3 – WAP Model

In the WAP model of figure 3, a session is opened from the mobile client to the gateway and a WML URL is requested from the web server. Then, the requested URL is returned to the mobile station and its contents displayed on the micro-browser. The data packets from the mobile station pass along a wireless network in WML format to a WAP gateway. This then reconfigures the essential data and then passes it again to a WML format. Conversely, when WML data packets need to reach the mobile client, they must first pass through a WAP gateway.

Moreover, when the web server acts as a client then it opens a secure session with the WAP gateway for

audio or video using the media type message. When the session is opened, the gateway does not perform encoding/decoding but just forwards the audio or voice packets which are then transmitted to the mobile client. The secure session is opened by the SCP, which interconnects the web server with the gateway and the mobile client. SCP follows the same principle as the WTLS but with fewer overheads. It provides services that ensure privacy, server authentication, client authentication and data integrity. When a client and a server first start communicating, they select cryptographic algorithms, authenticate each other, and use public key encryption techniques to distribute the keys.

In the simulation, the web server and the gateway first start communicating, they select cryptographic algorithms, authenticate each other, and use public key encryption techniques to distribute keys. Then, the gateway and the mobile client extend the already opened secure session, between the gateway and the web server. When gateway and mobile clients open a new secure session and agree upon the same group id to be used in the gateway-web server session which combines the two sessions as shown in figure 4.

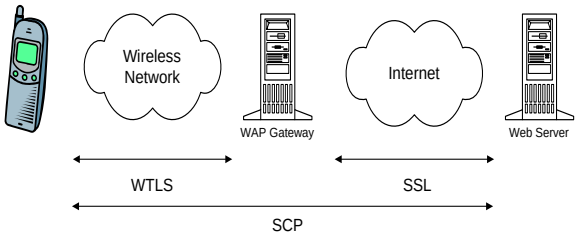

Figure 4 – Future Security Structure

The protocol is composed by three phases: **open session**, **data session**, **close session**. These processes can be summarised as follows: In the open session phase, the web server sends a session request including the media type message to both gateway and mobile client. Then they respond with a session response message. Following the session response, the web server sends a session indication message to finalise the opening of the session. The session request, response, and indication establish the following attributes: authentication of web server, gateway, and mobile station, agreement of cipher suite, and generation of session key.

In the data session phase the encrypted data is transferred. Encryption synchronization is also supported using the synchronisation message to overcome synchronization problems that arise from dropped data frames due to channel errors. The synchronization message is included in the data session phase.

Finally, the application program that receives the first close session request sends a close session response to acknowledge the end of the session. The application program uses the close session phase to shut down a connection when it has finished using it. The same operation is used to abort a session, in which case data transfer in both directions ceases immediately, and resources such as buffers are released.

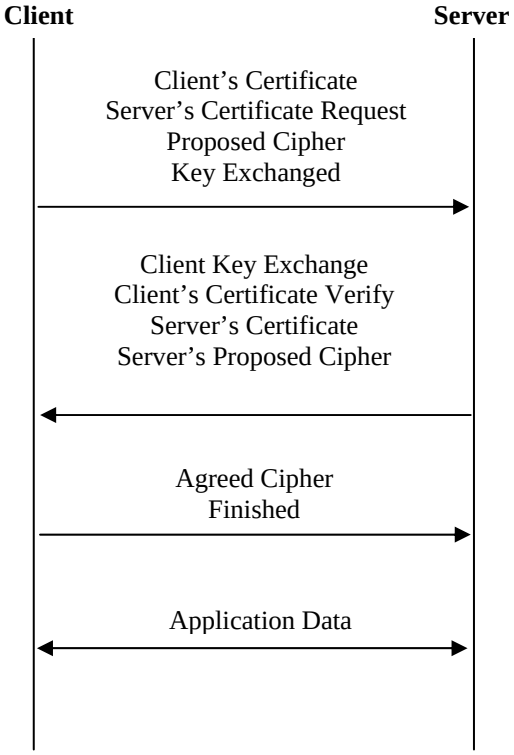

Figure 5 – SCP Secure Session

CONCLUSION

In this paper modifications to the WAP protocol were presented. WAP has been modified to support secure real time audio and video. When a WAP session is opened a media type message (MTM) is transmitted to enable audio and video communication. Once received the WAP gateway is enabled to perform according to the media requirements. The media type message is transmitted when a session is initiated either from a mobile client to a web server or from a web server to a mobile client. Furthermore, different media types and modes of encryption require encryption synchronisation for continuous media types. Encryption synchronisation is achieved by a synchronisation message, which is also added to the WAP protocol. The synchronisation message is transmitted either from a mobile client to a web server or from a web server to a mobile client. Moreover, group calls are also supported and can be established with the group id parameter, which is included in the media type message. Group calls can be established using data calls in future mobile communication systems where high data rates will be supported. Finally, in the simulation a secure communication protocol (SCP) was implemented to replace the

security layers WTLS and TLS. Through SCP audio, video and group calls are enabled and encryption synchronisation is supported between two applications.

ACKNOWLEDGEMENTS

Special thanks to HW Communications Limited for their financial support. We also like to thank K.Kotsokalis and A.Samaras for their help in the implementation, which is in progress.

REFERENCES

(1) WAP Forum, 2000, "Wireless Application Protocol Architecture Specification", http://www.wapforum.org/what/technical.htm

(2) WAP Forum, 2000, "Wireless Application Environment Specification", http://www.wapforum.org/what/technical.htm

(3) WAP Forum, 2000, "Wireless Session Protocol Specification", http://www.wapforum.org/what/technical.htm

(4) WAP Forum, 2000, "Wireless Transaction Protocol Specification", http://www.wapforum.org/what/technical.htm

(5) WAP Forum, 2000, "Wireless Transport Layer Security Specification", http://www.wapforum.org/what/technical.htm

(6) WAP Forum, 2000, "Wireless Datagram Protocol Specification", http://www.wapforum.org/what/technical.htm

(7) RFC 2646, 1999, "The TLS Protocol", http://www.rfc-editor.org/rfc/rfc2246.txt

(8) B. Schneier, 1996, "Applied Cryptography", Published by John Wiley & Sons Inc.

(9) Alfred J.Menezes, P.C. van Oorschot, S.A. Vanstone, 1996, "Handbook of Applied Cryptography", CRC Press.

(10) N. Komninos, B. Honary, 2000 "Secure Communication Protocol for Mobile Multimedia Applications", Third International Symposium on Wireless Personal Multimedia Communications, 2, 1011-1014