The Security Hole in WAP: An Analysis of the Network and Business Rationales Underlying a Failure

Niels Christian Juul and Niels Jørgensen

ABSTRACT: To succeed commercially, the Wireless Application Protocol (WAP), a protocol for the delivery of Internet-like services for mobile phones, had to dominate the market for mobile electronic commerce, but a security hole made it unsuitable for e-commerce transactions. The security hole was a byproduct of the so-called WAP-gateway. Mobile service providers offering WAP to their subscribers were supposed to deploy the gateway as a converter between the fixed and mobile networks. The early versions of WAP could not solve the security problem in a way that was satisfactory from both a technical and a business perspective. Although the security weakness was not the main reason that WAP failed, it would probably have caused the protocol suite to fail even if every other problem had been solved.

KEY WORDS AND PHRASES: Gateway, Internet, mobile commerce, protocols, security, standards, WAP.

A customer using a mobile device to place an order with an e-merchant is an example of a simple security-sensitive mobile transaction. A transaction of this kind entails an exchange of sensitive information with the merchant, typically including such items as credit card number and delivery address. Customers are not likely to engage in such transactions if there is a risk that the confidentiality of their data will be violated anywhere in between the parties. As a consequence, e-merchants are not likely to invest in the technology. The same consideration applies to authentication, message integrity, and nonrepudiation, but for simplicity, this paper focuses on confidentiality (e.g., of credit card numbers).

Confidentiality is not guaranteed when a customer uses a mobile phone and the Wireless Application Protocol (WAP) [12, 13]. This holds even when encryption is used in accordance with the security protocols of WAP version 1.2.1 from June 2000 [18, 20, 21]. The WAP gateway constitutes a security hole because data are transmitted inside it in their original unencrypted form. WAP fails to provide end-to-end security, that is, a secure communication channel between the two parties buffering a potentially insecure network. The gateway constitutes a point in WAP between the two end points (the customer and the merchant) where data may be compromised.

The WAP gateway is software. Typically, it runs on a computer in a building controlled by the mobile service provider (MSP). Figure 1 illustrates the role of the mobile service provider as an intermediary in a WAP-based e-commerce transaction. Because of the WAP security weakness discussed in this paper, all the data exchanged may be available to anyone with privileged access to the WAP gateway—for example, a system administrator of the machine the WAP gateway is running on. Thus, the privacy of the data depends

International Journal of Electronic Commerce / Summer 2003, Vol. 7, No. 4, pp. 73–92.

1086-4415/2002 $9.50 + 0.00.

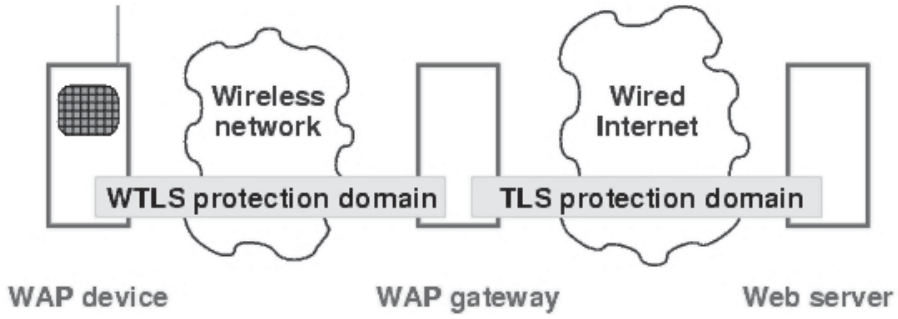

Figure 1. Security Zones and Parties Involved in a Generic m-Commerce Transaction Using WAP
Middle: Mobile telephone company (mobile service provider, MSP) uses WAP gateway to connect between wired and wireless Internet. *Left:* Customer uses WAP-enabled mobile phone. *Right:* E-merchant's Web site is connected to fixed Internet. The networks illustrated are simplifications. The first is a mostly wireless, mobile access network secured by WTLS; the second is a fixed, mostly wired Internet secured by TLS.

on the internal security policy of the mobile service provider and the methods of giving computer access to technical personnel.

The existence of the security hole sketched above is commonly recognized [3, 10, 14]. Since the early version of WAP did not profess to offer end-to-end security, the hole did not violate explicit claims or specifications about the protocol by the WAP Forum. Companies behind the WAP standard have argued that the security hole is negligible [7], but the WAP Forum has not attempted to justify the protocol's weak security or shed any light on the design discussions that led to it. Therefore, it has been necessary to reconstruct—one might say, to postmortem—a design discussion of the WAP gateway, taking into account both technical and business goals.

So far as is known, no independent overall evaluation of the security features of WAP has been published. The present paper does not constitute such an evaluation, in that it focuses on the security hole at the gateway. A rigorous analysis of the so-called Wireless Transaction Protocol (WTP) part of WAP by Gordon, Kristensen, and Billington revealed some errors but strongly indicated that the basic design of this part is sound [4]. The security problem related to the WAP gateway can be avoided if the most recent version of WAP is used, version 2.0 from July 2001. This version incorporates an alternative model into the standard that does not use a gateway at all, and instead utilizes Internet protocols end-to-end. However, version 2.0 was not widely adopted as of November 2002 **<<update?>>**, and it is questionable whether it ever will be, because WAP apparently does not have sufficient momentum to attract the investments required for the implementation of the new version.

An overall evaluation of the failure of WAP to gain widespread acceptance despite the enthusiastic reception of the WAP standard by mobile service providers and producers of mobile phones and network equipment must take

account of several other contributory factors. In particular, early users were rightly disappointed with the lack of available WAP content and the network latency experienced when requesting such content.

The need for specific WAP services (Web servers providing WML-documents) became more apparent during the implementation of the initial version of WAP, when there was no automatic translation available between HTML and VML. Thus, the original expectation of making the entire Internet or at least a large part of the World Wide Web available via conversions in the WAP gateway failed, and Web servers were needed that could provide content readable by WAP phones. The lack of WAP services may, in turn, have been related to at least the following two factors: (1) the lack of a business model, such as the one in the Japanese iMode system, in which content providers can charge the mobile phone user via the ordinary billing system of the mobile service provider [1, 6], and (2), the difficulty of developing WAP services because WAP was implemented differently on different mobile phones.

The original bearer used by WAP on GSM provided a dial-up connection, similar in performance to modem connections on the wired phone system. The high mobile tariffs on these connections made it economically feasible to shut down after a short idle time. Thus, latencies measured in minutes were apparent when surfing using WAP. With the introduction of a packet switching data network into GSM, that is, the "2.5 G" provided by GPRS, network latency was greatly reduced, but this took place at a time when WAP's initial momentum had already vanished.

However, even if solutions had been found to these and other challenges, the security weakness studied in this paper would have remained because it was intrinsic to the WAP protocol's definition. In itself it would probably have severely limited the acceptance of WAP. The assertion that the security weakness associated with the gateway was intrinsic to WAP is substantiated by the analysis in this paper of two proposals for achieving better security despite the gateway hole. The first proposal recommended placing the gateway inside the "vault," that is, in the same security zone as an e-merchant's Web server (rather than at the mobile service provider). The second proposal called for building so-called application-level security on top of WAP (where WAP itself is considered insecure). Also studied below in some detail is a third approach, enabling the Internet directly on the mobile phone—the alternative introduced in WAP version 2.0. All three remedies to WAP's security weakness have serious drawbacks from both a technical and business perspective.

The Gateway-Based Design of WAP

This section introduces the WAP standard with the objective of identifying the technical and business rationale behind the WAP gateway.

The Wireless Application Protocol (WAP) [23, 12, 13][1] is a suite of standards[2] for browsing the Web with a thin client browser, such as a mobile phone. The standard describes a full suite, sometimes referred to as a "stack" of protocols, in compliance with the ISO-OSI model for network protocols [11]. The WAP protocol stack is compared with the Internet protocol stack in Figure 2.

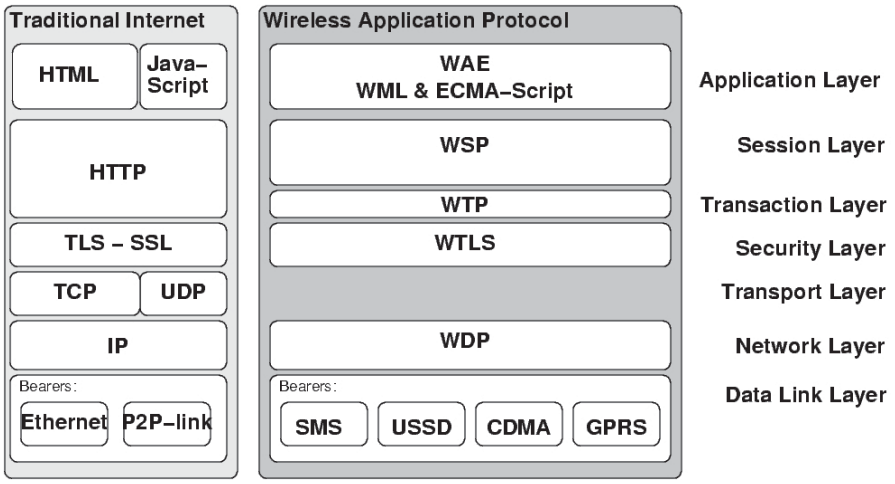

Figure 2. WAP vs. Internet Protocol Stack

Although illustrated side by side, neither protocol layer communicates across the two stacks. The obligation of the gateway is to utilize the two stacks and convert between the two protocols HTTP and WSP at the top.

Technical Rationale for the WAP Gateway

In addition to mobile phones, the prototype example of a WAP device for the purposes of this paper, the WAP standard aims at other hand-held digital devices, such as pagers, two-way radios, and smart-phones. These devices are characterized by limited processing power, storage capacity, input/output (key-pad and display), and power (battery capacity). WAP was designed to work not only with GSM but with most other digital wireless telephone networks. Some bearers are illustrated at the bottom of Figure 2. Compared to the well-known Internet, mobile wireless networks are characterized by limited communication capacity (bandwidth), higher latencies, higher variation in packet-loss (jitter), and variation in long-term connectivity/availability (on/off). By the launch of WAP on GSM in Europe, the bottom level WAP protocol, WDP, provided a packet-oriented transport service on top of a PPP service based on a dial-up modem service (GSM data). WAP's high latency was due to the dial-up latency. It had vanished as early as the 2.5 G mobile networks, like GSM-based GPRS,.

Despite early plans to have the WAP gateway convert between HTML and WML (Wireless Markup Language), the initial WAP standard required that WML documents be delivered from the Web server to the mobile phone. WML is WAP's replacement for HTML, designed specifically for the small and diverging displays of the mobile phone. A WML document is ordered and sent in a request-reply cycle in which the roles of the WML browser, the gateway, and the Web server are as follows (*see Figure 1*):

- **The request:** The user of the mobile phone orders information by clicking on a link, which is shown on the display by the WML browser. This action spawns a so-called WSP (Wireless Session Protocol) request sent from the phone to the WAP gateway. The WSP request parallels the HTTP request sent when an ordinary browser requests a document from a Web server. In the gateway, the WSP request is converted to a standard HTTP request, which is sent to the Web server.
- **The reply:** The Web server's response to the request is the action of sending the WML document to the WAP gateway. After some processing in the gateway (see below), the modified document is sent to the mobile phone.

In order for the WAP gateway to deal with the low-capacity wireless *network connection*, it has the following functions:

1. Switching between transport protocols

In its communication with the mobile phone, the gateway uses a transport protocol (WDP, for Wireless Datagram Protocol),[3] which is of a connectionless and unreliable nature. In contrast, TCP, the transport protocol used by HTTP on the ordinary Internet, is connection-oriented and reliable. In TCP, there is additional communication to ensure that all data actually gets through, and if not is retransmitted. Omitting this in the wireless part of the connection incurs a risk of losing data but saves significantly on latency WML and bandwidth.

2. Compression and decompression

The gateway compresses the document before sending it to the mobile phone. The gateway applies so-called lossless data compression, which means that the same information can be sent in a format that occupies fewer bites. Symmetrically, the gateway reads and interprets the original WSP request, which is in a compact form that cannot be understood by the Web server. The Web server understands lengthy HTTP requests in which various commands are given as ordinary text rather than by numerical codes. In comparison, compression is not used by default on the ordinary Internet.

3. DNS look-up

The gateway translates the symbolic Internet address specified in the mobile phone's WSP request into an IP number. The gateway does this in the usual way by communicating with a DNS (Domain Name Service) server. Relieving the mobile phone from doing DNS look-ups reduces consumption of the scarce network bandwidth of the wireless connection. More important, though, the total latency between the user's pressing the button and the result's being shown on the mobile phone as two request/reply cycles over the wireless link is reduced to one by this scheme.

In addition, to accommodate the limited capacity of the mobile phone, the gateway also has the following function:

Task of gateway	Summary of benefits
1. Switching between transport protocols	Accommodates different characteristics of wired vs. wireless networks
2. Compression and decompression	Compact communication over wireless link saves bandwidth and time
3. DNS look-up	Avoiding DNS-lookup over wireless link saves bandwidth and time
4. Compilation	Efficient storage and interpretation of compact WML documents in WAP device

Table 1. Technical Rationale for WAP Gateway.

4. Compilation

If a WML document contains embedded source code, the gateway compiles it into a byte code format. This relieves the mobile phone of the task of parsing the code. WML documents may contain code written in the scripting language WML-script, which is similar to the JavaScript language that may be embedded in HTML documents. Such code is executed inside the browser. For example, it may be used for validation, say, checking that a credit card number contains a certain number of digits and allowing the user to correct it before sending the information to the Web server.

While the gateway is essential for most of the techniques employed by WAP to accommodate the limitations of the mobile phone and the wireless network, other means, such as various features of WML, are independent of the gateway-based design. This includes the splitting of a WML document into "cards" sent as one document. The user browses from one card to the next by clicking on links, thus navigating through all the cards in the same document (or "deck") without needing further communication with the server. This eliminates latencies originating from the time it takes to retrieve the next card in a request/reply cycle across the high-latency wireless network.

In summary, the technical rationale for the WAP gateway is listed in Table 1.

Business Rationale for the WAP Gateway

Supplementing the technical rationale, there is a business rationale underlying WAP's gateway-based design. The gateway may open new business opportunities for the mobile service provider. Its impact on business, and on usability in a wider sense, is summarized below from the point of view of the mobile service provider, the mobile customer, and the e-merchant.

The *mobile service provider* (MSP) may use the gateway to tie its mobile customers to a portal-like access point to the Internet. Such portals are well known on the fixed Internet, where Internet Service Providers like America Online and Tiscali provide them to private/home-based users of the Internet. There is, however, an important aspect where an MSP driven mobile portal may differ from an ordinary Internet portal and, indeed, provide much more pow-

erful services. The MSP can provide added services to a mobile portal by combining the WAP service with its basic phone service or location information. WAP integrates with the basic phone services by using a dedicated protocol, Wireless Telephony Application Interface (WTAI) and the mobile phone system provide location information, such as the exact cell in which the mobile phone is located [17]. By running the WAP gateway, the MSP may combine the gateway service with personal and privileged information of this kind.

The *mobile WAP* user has a dual interest. On the one hand, WAP users would presumably like to have access to whichever of their MSP's services integrate WAP and ordinary mobile telephone service, as made possible by accessing the Internet via the MSP's gateway. On the other hand, they are clearly interested in the cheapest possible, universal access to all WAP-based services on the Internet.

The *typical e-merchant* has one obvious interest in relation to WAP: the interest of minimizing the cost, in terms of hardware, software, maintenance, and so forth, of extending e-commerce to WAP. For most e-merchants, WAP-based e-commerce is just an additional channel to be added to e-commerce via the Web. The chief advantage of the gateway is that the e-merchant can open this channel merely by hosting a documents of a new type, WML documents, because the gateway takes care of any adaptations to special requirements of the wireless network and devices.

The MSP's interest in controlling a portal-like access to the Internet is illustrated by the success of iMode, launched in Japan by NTT DoCoMo in February 1999. In December 2001, the number of iMode subscriptions exceeded 29.5 million, far outnumbering the total number of WAP users throughout the world.

iMode is a proprietary standard controlled by NTT DoCoMo, and users must purchase an iMode mobile phone and connect to DoCoMo's iMode service, which resembles a portal. Via the portal, the user can access two kinds of sites run by other companies: so-called official and unofficial sites. For a site to become official requires approval by DoCoMo and entails the privilege of charging for the site's services via DoCoMo's billing system, in return for a 9 percent fee to DoCoMo. The protocols and other standards that iMode is based on have not been fully published or subjected to as much public analysis as those of WAP. The account here is based on the discussions by Ashley, Hinton, and Vandenwauver, and by Megler [1, 6].

It must be emphasize once again that the business rationale is reconstructed, for iMode had not been deployed or conceived when WAP was designed. It should also be noted that the development of the WAP standard was not initiated by representatives of any of the three parties listed above, but by producers of hardware and software for mobile devices and networks, including Nokia, Motorola, Ericsson, and Phone.com (formerly Unwired Planet). The main interests of these producers may be identified as selling hardware and software to mobile service providers (and companies that own mobile networks), and mobile phones and other devices to users. The prospects for this, of course, are highly dependent on successful integration with the Internet, including e-commerce.

Table 2 summarizes the business rationale of the WAP gateway.

Party	Main benefit
A. Mobile service provider	Revenue potential from tying customer to access via provider-controlled portal
B. User of mobile device	Access to services that rely on gateway's access to sensitive data available only to provider
C. E-merchant	Availability of WAP as extra channel that uses merchant's existing Web server infrastructure, merely requiring hosting of a new type of document (WML documents)

Table 2. Business Rationale of WAP Gateway.

The Security Hole at the WAP Gateway

For security, WAP provides a secure protocol for data transport: WTLS (Wireless Transport Layer Security) [18]. WTLS also contains features for authentication of both parties, and also for nonrepudiation using message digests and digital signatures. Authentication of the user of a GSM phone, as was pointed out above in the discussion of the technical rationale for the WAP gateway, may utilize the phone's SIM card [20]. The definition of WML-Script includes a specification of a function library called a "crypto package" [21]. This library contains a signing function with which a user can digitally sign a message in the conventional manner in a Public Key Infrastructure. Thus, WAP supports nonrepudiation of messages (e.g., placement of an order) sent by a mobile WAP-user. (The cryptographic library and its possible usefulness in implementing application-level security will be discussed further on.)

WAP's Two-Stage Security Model

Consider a WAP-based m-commerce transaction in which a document, say www.weSellIt.com/orderNow.wml, must be transferred to the mobile phone in a secure manner.

The basic security feature of WAP 1.2.1 provides secrecy in the two halves of the path that connects the WAP client and the Web server: the (presumed) wireless path between the WAP client and the WAP gateway, and the (presumed) wired path between the gateway and the Web server (*see Figure 1*). In the middle point constituted by the gateway, incoming data are decrypted. Then, while the data are in their original unencrypted form, they are subjected to some processing. Finally, they are encrypted again and are sent off along the other path. The processing done on the unencrypted data corresponds to the protocol layers above the security layers in Figure 2 in the gateway.

For encryption over the wired path, WAP relies on the Transport Layer Security (TLS) protocol, a widely used Internet standard [2]. TLS is standardized by the Internet Engineering Task Force and is known under the name Secure Socket Layer (SSL) given to it by Netscape, the company that originally developed the standard. For encryption over the wireless path, WAP uses WTLS, which is, in essence, an adoption for wireless communication of

the TLS protocol [18]. The changes to TLS embodied in WTLS do not weaken security.

WAP's topmost protocol, the Wireless Session Protocol (WSP), initiates and links the two secure halves of the connection [16] The major steps in this are as follows: The user of the WAP client selects the URL https://www.weSellIt.com/orderNow.wml. This tells the WSP layer at the WAP client to set up a WTLS connection to the WAP gateway and then pass a WSP request (the equivalent of an HTTP request) over the connection for the file. Thus prompted by the WAP client, the WSP layer at the gateway will initiate a TLS connection to the Web server, in exactly the same way as it sets up a connection between an ordinary Web client and the server.

This combination of WTLS and TLS provides secrecy (and, indeed, integrity as well) over both halves of the WAP client/Web server connection. The crucial weakness is, of course, that the data transferred between the WAP client and the Web server are all decrypted at the WAP gateway—in other words, all the data (e.g., credit card numbers) exist as free text in the memory of the gateway. Technical solutions, such as various programming techniques applied to the software that implements the WAP gateway, can make it somewhat difficult to get access to the data, but not impossible. Organizational solutions, such as tightening the security policy of the organization that hosts the WAP gateway, may limit access to the gateway and its data. These methods do not eliminate the security hole, however—they merely make it more difficult to exploit.

Encryption vs. the Functions of the Gateway

The breach of end-to-end security at the WAP gateway is not an accidental error. It is a disadvantage of the gateway-based design that the designers felt was outweighed by its advantages. To argue this, it is necessary to reconsider the technical rationale underlying the gate way design. The question is: Which of the four functions discussed earlier would the gateway be unable to fulfill if the WAP standard prescribed a different approach to confidentiality, one that attained end-to-end security as provided by TLS on the ordinary Web?

The answer is that the gateway would be unable to perform compression/decompression and compilation. However, it could be used as a point of switching between transport protocols and for DNS look-up. The gateway may function as a caching DNS server or provide dynamic address translation between domain names and IP addresses for each message, assuming that the addresses are exchanged outside the encrypted channel. Code written in WML-Script is encrypted and cannot be compiled unless the gateway can first decrypt the script, thus breaking the end-to-end encryption. For the same reason, it would not be possible for the gateway to understand (decompress) encrypted WSP requests. Finally, the gateway's compression of ordinary WML documents is ruled out. If the gateway cannot recognize, for example, the WML tags card and /card, it cannot compress them (by replacing them with numbers that consume fewer bytes). In general, the characters of (well-)encrypted text are randomly distributed, that is, with no apparent patterns, and therefore such

Task of gateway	Realizable assuming end-to-end encryption	
1. Switching between transport protocols	√	Still possible to switch between transport protocols on wired and wireless network below encryption layer of protocol stack
2. Compression and decompression	–	Switching between WSP and HTTP thus providing compression/decompression cannot be achieved when message is unreadable at gateway
3. DNS look-up	(√)	Gateway may act as caching DNS server, still requiring additional request/reply cycle from mobile phone, and address information must be exchanged off encrypted channel
4. Compilation	(-)	Compilation of WML and WML-script cannot be achieved at gateway but can be done at Web server, e.g., by providing WMLc.

Table 3. Realizable Tasks of WAP Gateway When End-to-End Encryption Is Employed on WTLS/TLS/SSL Layer.

data cannot be compressed. (Encryption versus compression is discussed at length by Klein, Bookstein, and Deerwester [5].)

In summary, if end-to-end encryption were employed on the WTLS/TLS/SSL layer, a modified gateway would be able to perform only a subset of the gateway tasks, as shown in Table 3.

Thus, modifying the WAP standard to provide full end-to-end encryption, as in TLS, requires substantial protocol redesign and conflicts with several of the key functions of the gateway that serve to limit the amount of data that has to be transmitted over the wireless network. In short, the security hole at the gateway is intrinsic to the gateway's overall technical goal.

First Suggestion: Putting the Gateway Inside the "Vault"

One of the founding partners of the WAP Forum, Phone.Com, suggested putting the WAP gateway at the Web server end of the connection, that is, inside the same security zone [7]. Residing inside the local network of the merchant, the gateway is protected against the outside world in the same way as the Web server (*see Figure 3*). This remedy to the security hole can be implemented using WAP 1.2.1, and thus uses the original gateway-based variant of WAP.

Most financial institutions providing WAP-access to their customers adopt this solution. Secure WAP banking is possible when the WAP gateway is kept inside the vaults of the banks. One may ask why so many first-generation WAP banking solutions restricted the functionality to low-risk transactions, only allowing transfers between customers' own accounts, not to other accounts. The main reason was that WAP 1.1 did not utilize full identification of customers. Since this restriction has since been solved in WAP 1.2.1, the lack of security in WAP transactions discussed in this article is not due to lack of customer identification but to security breaches in the WAP gateway.

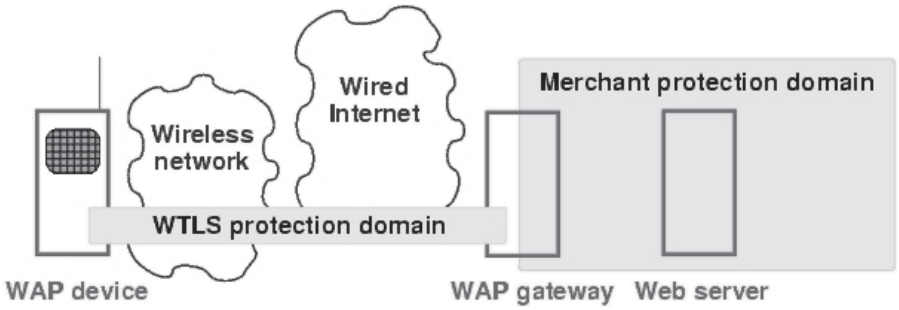

Figure 3. Security Zones by Replacement of WAP Gateway
The customer utilizes either a dial-up point at the MSP and continues across the Internet or the merchant's own dial-up point bypassing the Internet. The secure channel is protected by WTLS in both cases.

Technical Implications

The above-described solution to the security problem conflicts with one of the fundamental purposes of the WAP gateway: converting between two distinct protocol suites, one for the wireless network (WAP) and one for the traditional wired network (the Internet protocol suite, including HTTP and TCP). The protocols used for transport constitute a crucial point of difference between these two stacks. If the wired and wireless networks are different enough to justify the use of the different transport protocols, the gateway should be placed at the intersection of the two networks, so that transportation over each networks uses the appropriate protocol.

The WAP Forum has proposed a variety of gateways and proxies, including information enabling the mobile phone to switch between gateways [14]. So far, this added functionality has not been provided to users at large, and there are unsolved questions regarding recovery inside the mobile phone after network failures. While it enables end-to-end security between the transport layer in the handset and the merchant gateway utilizing WTLS, an additional burden is also placed on the merchant to maintain the entire WAP stack of protocols [18, 19, 15].

It the gateway is put inside the same security domain as the Web server, most of the benefits due to the gateway are still achievable (*see Table 4*).

Although many banks took up the burden, and provided WAP banking using their own WAP gateways, the solution never reached widespread customer acceptance and use. It is fair to say that this was not entirely the fault of the WAP security solution. The main hindrances in using first-generation WAP for banking were not the cumbersome handset reconfiguration but the high latency and the lack of business applications needing WAP for banking.

Business Implications

A closer look at the business potential for the three stakeholders reveals further problems for this solution.

Task of gateway	Achievable benefits assuming gateway co-located with server
1. Switching between transport protocols	– Does not provide transport protocols dedicated to wired and wireless networks respectively
2. Compression and decompression	√ Switching between WSP and HTTP and provision of compression/decompression is achieved
3. DNS look-up	√ Function is even more simplified, as only local Web server addresses is needed
4. Compilation	√ Compilation of WML and WML-script is easily achievable

Table 4. Achievable Tasks of WAP Gateway When Co-located with Web Server.

The *mobile service provider* (MSP) looses control in several ways when it does not control the gateway in the middle of all communication. As in the case of NTT DoCoMo's iMode, being the middle-man through whom customers and merchants must communicate makes a strong business case. This control is lost when the gateway is moved to the merchant. The same holds for a number of possible WAP-services that the MSP can offer only if it hosts the gateway, since this is its only way to ensure that the data used in those services stay within the mobile network it controls. For instance, location-dependent services utilize the MSP's access to information from its own network about the exact location (cell) of the mobile phone. There may be strong business and security reasons for the MSP to not want such data to be available outside the network. In the case of location information, the data are clearly sensitive, and at the same time they may have a high business value because they are a prerequisite for certain services. Moreover, this solution challenges the MSP to open its network to outsiders, such as merchants running their own gateways.

Each limitation on the MSP's full control over the usage of its networks may introduce further traffic management problems [7].

The *user* of the WAP client will witness degraded performance. The WAP protocols are tailored for wireless networks but are now used for the entire transport of Web content to the WAP client, instead of merely for the wireless part, as intended. This may incur increased delays if there is congestion in the wired network.

The end-user interface becomes less friendly, because the user of the WAP client will be forced to swap between gateways during Internet browsing. For example, the user that wants to buy from two distinct Web sites needs to use the WAP gateways of those sites (given that both transactions are secured by the method discussed in this section).

Swapping gateways raises two problems when the gateway profile of the phone is changed. The gateway profile includes a dial-up phone number, the IP number of the gateway, and other information. First, in many current implementations of WAP, the user of a WAP device has to go through a cumbersome procedure that involves clicking through several menus and typing an

Internet address in the form of an IP number. Future WAP implementations may provide features for easily switching gateways, but the user should at least be asked to confirm a gateway switch. Second, different gateways may have slightly different properties. Although standardized by the WAP Forum, current gateways perform differently on the same task. Furthermore, users who have problems (e.g., quality of service) may find it difficult to resolve whether the problem is with the MSP, the merchant, or the mobile phone manufacturer [7].

The option by which the mobile service provider runs a kind of default gateway—one that the user switches back to, after using a particular e-merchant's gateway—is not as simple as may first appear. A default gateway requires either that the client explicitly "log out" or, alternatively, that the WAP device include a "time out" mechanism letting it switch back automatically.

The *merchant* is burdened with a complex piece of additional software (the gateway) that must be acquired and maintained. For a WAP gateway, maintenance includes ensuring that security-related software in the gateway is up to date, and updating the gateway to new versions of the WAP protocols. Furthermore, location-based services, micro-payment, and similar functionality may not be available to the m-commerce application either from the MSP-detached gateway or from the short-cut MSP.

It is not clear, however, whether the merchant will be further "burdened with handset provisioning and activation issues," as suggested in a Phone.com White Paper [7]. In any event, because of the huge differences in how WML documents are interpreted in the various WML browsers and mobile phones, these issues must be addressed by the Web server servicing the WML.

The goal of WAP banking was to provide customers with the ability to do financial transactions over yet another medium. Mobile banking was already available over the phone using voice, and so was Internet banking on a larger display. Thus, the business case for mobile banking on a small handheld screen is mostly to the benefit of the bank (reduced labor cost in call center function), but the customer sees no incentives.

In summary, placing the WAP gateway with the Web server may achieve security for customer and merchant at the cost of an additional burden on all the players (*see Table 5*).

Application-Level Security on Top of WAP

This amounts to introducing security in a software layer above WAP and considering WAP to be a potentially insecure communication means (*see Figure 4*). Instead of using WAP's protocol for secure transport (WTLS), security is taken care of by means of dedicated software running at the two "ends," the mobile phone and the e-merchant's Web server. Such software could perform encryption in a way that eliminates the security hole at the gateway. Thus, as with the first solution, the idea here is to retain the gateway-based variant of WAP. In general, the approach would be in line with Saltzer, Reed, and Clark's conjecture that protocol design should place an end-to-end function at a level where end-to-end control is available, that is, at the application level [9].

Party	Benefits and burden assuming gateway co-located with server
A. Mobile service provider	– Lost revenue potential when customer and merchants are no longer tied to provider-controlled portal; lost control over traffic on MSP's own network
B. User of mobile device	– Burden with handset reconfiguration and lost access to MSP's specific services like location information
C. E-merchant	– Burden with WAP gateway maintenance and lost access to MSP's specific services like location information

Table 5. Benefits and Burdens on Business Players When WAP Gateway Is Co-located with Web Server.

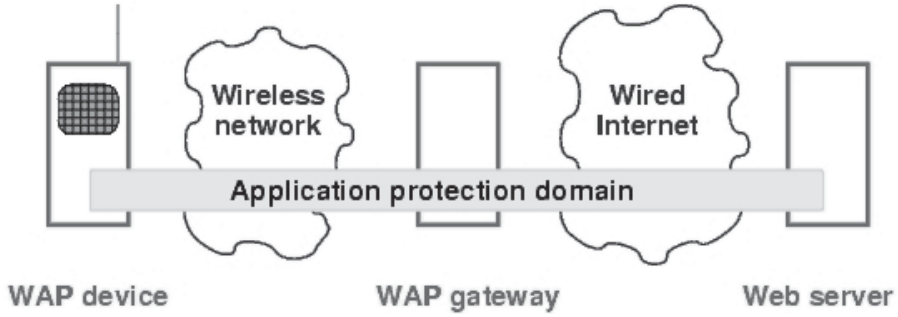

Figure 4. Security Zones Using Application-Level Encryption

While technically possible, this solution would add a burden of extra complexity in the mobile phone. The implementation of application-level security on top of WAP requires that sophisticated cryptographic functionality be made available to applications on the mobile WAP phone, either as future enhancements of the WML Script Crypto Library [21, 22] or in other ways. The single function for digitally signing a message, currently the only function contained in the library (as of WAP 2.0), is insufficient. The Crypto Library's signing function is not useful for encryption of data because everyone in possession of a WAP user's public key can, in principle, read a message that the user has signed digitally.

Moreover, this approach would neutralize most of the optimization provided by the WAP gateway. It would lose the benefits of the data conversion and compression that take place in the gateway to accommodate the limited bandwidth of the wireless network (since encrypted data cannot be compiled or compressed/decompressed, as mentioned earlier). This loss can be minimized by a balanced approach that encrypts only small fragments of data, not the data as a whole. The WAP gateway would then be able to perform compression, decompression, and compilation by providing access to the WML tags and WSP/HTTP commands in their original, unencrypted form.

Finally, the application-level security approach requires infrastructural remedies to persuasively familiarize consumers with the level of security avail-

able. The solutions used by Internet banking with signed or certified apples seem to satisfy bank customers. If certificates could be easily verified on WAP, then a similar solution could be adapted to persuade WAP users. Ponce and Soriano propose further infrastructure to enable end-to-end security between mobile phone and application-server [8]. Current solutions to the end-to-end security problem on the application layer still lack standardization and support from the WAP Forum.

If encryption were available only for sensitive parts of messages, then the gateway would be able to run almost without changes (*see Table 6*). Unfortunately, the application-level solution is not standardized and places an additional burden on both merchant and customer in each relationship.

Enabling Internet on the Mobile Device

The third and last approach is to redesign the WAP protocol to not use a gateway and instead employ existing Internet standards, including the transport protocol (TCP), for the entire wired and wireless part of a connection. By definition, this solves the security problem introduced by the gateway.

The WAP Forum has proposed this change of design for version 2.0 of the WAP protocol [13]. The standard now allows a range of different gateways, corresponding to having the conversion between the two protocol stacks anywhere from the top to the bottom of the stack. At the top layers, the gateway works like the traditional WAP gateway, taking care of the functions mentioned in the discussion of the technical rationale for the WAP gateway. At the bottom layer, it works like a traditional bridge/router. Alternatively, at the TCP-level the gateway allows for two versions of TCP, one for the wired network and the other for the wireless network, on top of which a TLS channel can be established all the way from the mobile device to the server. The move away from top-level conversion constitutes a fundamental change of design, which removes not only the security problem, but also the optimization for the wireless network, made possible by the gateway. Some optimization has been achieved with wireless profiled versions of the Internet protocols. As long as the existing implementation of the protocols on the Internet can accommodate the new wireless profiled variants, they can be deployed between the gateway and the WAP device.

Discarding the WAP gateway makes it possible to attain the same high level of security for an m-commerce transaction as for an e-commerce transaction on the ordinary Web using full end-to-end encryption. Indeed, if WAP were to discard the WAP gateway, this would turn the (fully) WAP-enabled mobile phone into an ordinary Internet device.

Besides the cost of deploying a full Internet protocol stack in the mobile phone, this solution requires additional messages to be sent between the device and the Internet. A simple request for a WML document must, in contrast to the gateway-based WAP solution, wait for a DNS-lookup to derive the appropriate IP-number from the symbolic name in the original request. The gateway-based solution only has this cost over the wired Internet, when the gateway converts the WAP-request to an HTTP-request.

Task of gateway	Achievable benefits assuming application-level security
1. Switching between transport protocols	√ Selection of transport protocols is not affected
2. Compression and decompression	√ Switching between WSP and HTTP and provision of compression/decompression is achieved; only encrypted parts must be passed unaltered
3. DNS look-up	√ As long as destination address is not part of secret, DSN look-up is unaffected
4. Compilation	(√) Compilation of WML and WML-script must pass encrypted parts unaltered (an encryption tag can be utilized)

Table 6. Achievable Tasks of WAP Gateway If Security Is Applied at Application Layer.

In summary, the omission of the WAP gateway wrt.WSP-HTTP translation provides a solution close to the one mentioned above in the discussion of the security hole. The achieved benefits are also similar (*see Table 7*).

Conclusion

The Wireless Application Protocol provides technical and business benefits for mobile users, mobile service providers, and content providers, but all three solutions to the security hole at the WAP gateway involve the loss of some of these benefits. The major disadvantages of the three ways to achieve end-to-end security based on WAP are summarized in Table 8.

The solution that redesigns the WAP protocol to employ existing Internet standards instead of a gateway is an option in version 2.0 of the WAP protocol, introduced in July 2001. The introduction and subsequent derouting of the WAP gateway has increased the complexity of the WAP standard, now comprising variants both inside and outside the gateway. It is now more difficult for providers of software for mobile phones, e-merchants, and other content providers to implement the WAP standard. As of January 2003 **<<update?>>**, the market share of the new version remained invisible, indicating that the protocol as a whole had lost momentum. The manufacturers of WAP phones that formerly were associated in the WAP Forum have now moved to a new organization, the Open Mobile Alliance, in an effort to revitalize WAP and other mobile standards.

The standardization process that took place in the WAP Forum can be characterized as semi-open. On the one hand, the Forum was open to any company or organization (if it was able to pay the high entrance fee), and the standards agreed on were made publicly available. On the other hand, the discussions were not public. The WAP Forum never revealed the considerations that led to the redesign of version 2.0, which made the WAP gateway (largely) useless. Similarly, although the security hole associated with the gateway was obvious and well known, the WAP Forum did not reveal the delib-

Task of gateway	Achievable benefits assuming end-to-end encryption
1. Switching between transport protocols	√ Still possible to switch between transport protocols on wired and wireless network below encryption layer of protocol stack
2. Compression and decompression	– Switching between WSP and HTTP, providing compression/decompression cannot be achieved when message is unreadable at gateway
3. DNS look-up	(-) Gateway may act as caching DNS server, but mobile phone must look up addresses to be able to address final destination
4. Compilation	(√) Compilation of WML and WML-script cannot be achieved at gateway; but can be done at Web server, e.g., by providing WMLc.

Table 7. Achievable Tasks of Simplified WAP Gateway (Without WSP-HTTP Translation).

Remedy	Main technical disadvantage	Main business disadvantage
1. Putting gateway inside "vault"	Does not provide transport protocols dedicated to wired and wireless networks.	Customer is burdened with handset reconfiguration and lost access to MSP's specific services like location information
2. Application-level security on top of WAP	Complex approach not supported by WAP (even most recent version).	Approach is not standardized; parties must consider security of instances of approach on case-by-case basis.
3. Enable Internet on mobile device	Loss of three out of four optimizations performed by gateway.	Fundamental change of design that questions raison-d'être of WAP because gateway becomes (largely) useless.

Table 8. Major Technical and Business Disadvantages of the Three Solutions.

erations whereby it concluded that the gateway's advantages would outweigh its (security-related) disadvantages. In contrast, Internet and Web standards such as TCP, HTML, and the secure transport protocol TSL [2] were all subjected to public scrutiny prior to acceptance in the Internet Engineering Task Force and the World Wide Web Consortium, respectively. If the WAP Forum had released its considerations about the gateway and invited public debate about the security issues, perhaps it could have avoided the zigzag course whereby the WAP standard was initially based on an insecure gateway and subsequently redefined to not need the gateway.

Acknowledgment

An equipment grant from Siemens and a WAP/GSM-air-time grant from the National Danish Telecommunication Company, TDC, are gratefully acknowledged. We are in debt to Torben Lauritzen of Siemens for many fruitful discussions of WAP. A private e-mail correspondence with Scott Goldman, former CEO of the WAP Forum, and the discussion at the SSGRR conference on Advances in Infrastructure for Electronic Business, Science, and Education on the Internet in L'Aquila, Italy, in August 2001 and at the Bled conference on e-commerce in June 2002 further helped in shaping the paper. The constructive comments of several anonymous referees further improved the presentation.

This research was also funded by the Development Center for Electronic Business and by the IT-University, Copenhagen.

NOTES

1. The reader is referred to the Web site of the WAP Forum (www.wapforum.org) for the full list of standard documents—both approved and proposed standards.
2. The standards discussed here are the two latest approved sets of WAP standards, versions 1.2.1 (June 2000) and 2.0 (July 2001).
3. For the purposes of this discussion, WDP is identical to the Internet's UDP transport protocol

REFERENCES

1. Ashley, P.; Hinton, H.; and Vandenwauver, M. Wired versus wireless security: The Internet, WAP and iMode for e-commerce. In *Proceedings of the Seventeenth Annual Computer Security Applications Conference (ACSAC)*, December 2001. **<<ed.? place? pp.?>>**
2. Dierks, T., and Allen, C. The TLS protocol version 1.0. RFC 2246. Available at www.rfc-editor.org/rfc/rfc2246.txt.
3. Durham-Vichr, D., and Getgen, K. Securing the Internet without wires. Available at www-106.ibm.com/developerworks/library/wi-sectrends (August 2001).
4. Gordon, S.; Kristensen, L.M.; and Billington, J. Verification of a revised WAP wireless transaction protocol. In *Proceedings of* <<**the?**>> *International Conference on Application and Theory of Petri Nets (ICATPN) 2002*, *Lecture Notes in* **<<on?>>** *Computer Science 2360*. **<<place?>>**: Springer-Verlag, pp. 182–202. **<<ed. by?>>**
5. Klein, S.T.; Bookstein, A.; and Deerwester, S. Storing text retrieval systems on CD-ROM: Compression and encryption considerations. *ACM Transactions on Information Systems, 7*, 3 (July 1989), 230–245.
6. Megler, V. iMode. From bandwidth problem into Internet phenomenon. Available at www-106.ibm.com/developerworks/library/wi-imode/?dwzone=wireless (February 2002).

7. Phone.Com. Understanding security on the wireless Internet. White paper WPSE-R4–001, 2000. **<<available where?>>**
8. Ponce, D., and Soriano, M. Secure intelligent broker for m-commerce. In Veljko Milutinovic (ed.), *Proceedings of the SSGRR 2001*: *International Conference on Advances in Infrastructure for Electronic Business*, *Science*, *and Education on the Internet*. L'Aquila, Italy: Scuola Superiore G. Reiss Romoli (SSGRR), August 2001, CD-ROM.
9. Saltzer, J.H.; Reed, D.P.; and Clark, D.D. End-to-end arguments in system design. *ACM Transactions on Computer Systems, 2*, 4 (November 1984), 277–288.
10. Singhal, S.; Bridgman, T.; Suryanarayana, L.; Mauney, D.; Alvinen, J.; Bevis, D.; Chan, J.; and Hild, S. *WAP: The Wireless Application Protocol Writing Applications for the Mobile Internet*. New York: ACM Press, 2001.
11. Tanenbaum, A.S. *Computer Networks*, 3d ed. Upper Saddle River, NJ: Prentice-Hall, 1996.
12. WAP Forum. *WAP Architecture: Wireless Application Protocol Architecture Specification. Approved Specification*. WAP 100, WAP Forum Ltd., www.wapforum.org (part of the June 2000 [WAP 1.2.1] conformance release).
13. WAP Forum. *WAP Architecture: Wireless Application Protocol Architecture Specification. Approved Specification*. WAP 210, WAP Forum Ltd., www.wapforum.org (part of the July 2001 [WAP 2.0] conformance release).
14. WAP Forum. *WAP Transport Layer End-to-End Security: Wireless Application Protocol. Approved Specification*. WAP 187, WAP Forum Ltd., www.wapforum.org (part of the July 2001 [WAP 2.0] conformance release).
15. WAP Forum. *WAP TLS Profile and Tunneling WAP-219–TLS: Wireless Application Protocol TLS Profile and Tunneling Specification. Approved Specification*. WAP 219, WAP Forum Ltd., www.wapforum.org (part of the July 2001 [WAP 2.0] conformance release).
16. WAP Forum. *WAP WSP: Wireless Application Protocol Wireless Session Protocol Specification. Approved Specification*. WAP 203, WAP Forum Ltd., www.wapforum.org Last accessed on Jan. 24, 2003 (part of the June 2000 [WAP 1.2.1] conformance release).
17. WAP Forum. *WAP WTAI: Wireless Application Protocol Wireless Telephony Application Interface Specification. Approved Specification*. WAP 170, WAP Forum Ltd., www.wapforum.org (part of the June 2000 [WAP 1.2.1] conformance release).
18. WAP Forum. *WAP WTLS: Wireless Application Protocol Wireless Transport Layer Security Specification. Approved Specification*. WAP 199, WAP Forum Ltd., www.wapforum.org (part of the June 2000 [WAP 1.2.1] conformance release).
19. WAP Forum. *WAP WTLS: Wireless Application Protocol Wireless Transport Layer Security Specification. Approved Specification*. WAP 261, WAP Forum Ltd., www.wapforum.org (part of the July 2001 [WAP 2.0] conformance release).
20. WAP Forum. *Wireless Application Protocol Identity Module Specification (WIM) Part: Security. Approved Specification*. WAP 198, WAP Forum Ltd.,

www.wapforum.org (part of the June 2000 [WAP 1.2.1] conformance release).
21. WAP Forum. *WML Script Crypto Library: Wireless Application Protocol WMLScript Crypto Library Specification. Approved Specification*. WAP 161, WAP Forum Ltd., www.wapforum.org (part of the June 2000 [WAP 1.2.1] conformance release).
22. WAP Forum. *WML Script Crypto Library: Wireless Application Protocol WMLScript Crypto Library Specification. Approved Specification.* WAP 161, WAP Forum Ltd., www.wapforum.org (part of the July 2001 [WAP 2.0] conformance release).
23. WAP Forum. *Official Wireless Application Protocol: The Complete Standard with Searchable CD-ROM*. **<<place?>>**: Wiley Computer Publishing, 2000.

NIELS CHRISTIAN JUUL (ncjuul@acm.org) is an associate professor and head of the Computer Science Department at Roskilde University, Denmark. He has a Ph.D. in computer science (1993) from the University of Copenhagen and has researched distributed systems at the University of Copenhagen, the University of California, Riverside, and the Copenhagen Business School. He is a member of the ACM, IEEE Computer Society, USENIX, DISP, CPSR, and IFIP.

NIELS JØRGENSEN (nielsj@ruc.dk) is an associate professor in the Computer Science Department at Roskilde University, Denmark. He first became interested in mobile systems while working for Nokia as a software developer. Besides distributed systems and security, his research interests include optimized compilation and open-source software development.