

---

---

# Understanding WAP Security

---

---

*On the face of it, the much-hyped WAP appears quite secure, but there are several issues you should be aware of.*

**By David Norfolk  
IT Journalist**

**W**AP (Wireless Application Protocol) is being much hyped as part of the “mobile computing revolution” - e-business and e-commerce computing over mobile phones. There is certainly going to be a lot of WAP about (IDC has forecast that all “personal communications service” phones will be Internet-enabled using WAP by mid-2001), and as WAP is intended to cope with commercial transactions, security will be a requirement.

On the face of it, WAP security appears to be pretty good. Digital phones are trusted and the WAP model includes security in the form of WTLS (Wireless Transaction Level Security). WTLS is a form of SSL (Secure Sockets Layer) optimised for mobile phones. Nevertheless, things aren't as simple as this. For a start, some of the threats associated with WAP are different to those associated with ordinary Internet computing (and some are the same), and this must be taken into account when designing WAP applications. However, security firms such as ISS think that there are serious and fundamental problems with the WAP security model and, even if these don't affect you, there are certainly implementation issues with WAP (which isn't a mature technology) that may compromise security.

The fundamental problem with WAP security is that it is optional. This leads to the risk that someone can be persuaded to think that a transmission is safe when it isn't, and this may lead to a further compromise of security. Some of the people involved in WAP want to change this, but it seems unlikely that this will be addressed even in the upcoming next release. Perhaps security is seen as a usability barrier to adoption of a new technology, and many first-time apps are simple and don't need security, but even a perception of a lack of security can kill confidence in a new technology.

## **Cryptography**

However, the cryptography used for transmission across the Web probably isn't a problem. No doubt it can be broken, but not easily enough to be generally useful (that is, not in anything like real time). Where a serious problem does arise is with the WAP Gateway, which converts from the WAP security protocol WTLS, used between phone and Gateway, and SSL, used over the Internet. This implies that the message is available for a short time unencrypted on the Gateway and, if the Gateway is compromised, so are any WAP communications. Work is under way to address this (it seems to have no real advantages for anyone), but it isn't known when, or even if, the WAP specs will change.

A second fundamental problem at the moment, but one that will presumably go away in time, is the essential immaturity of the technology. Modern mobile phones are pushing the boundaries of what you can squeeze into a phone (both physically, and in terms of processor and memory), and this is a classic recipe for the production of hardware-specific solutions, code that is hard to understand, and obscure bugs introduced in maintenance releases. Tom DeMarco pointed out years ago (in *Controlling Software Projects*, Yourdon Press, 1982, ISBN 0-13-171711-1, reporting work by Weinberg and Schulman in 1974) that giving a programming team a goal of optimising one project metric (completion time or program size, for instance) severely impacted performance on other project metrics (program clarity or user-friendly output, perhaps). It is unlikely that the situation will be much different for the code in a WAP phone (although at least it isn't running Windows applications).

The net result of this technology immaturity is that phones tend to be subtly different

at the hardware level, even when they nominally conform to a standard, and may contain hard-to-test and buggy code. At the same time, the mobile phone industry is driven by fashion, and time-to-market is king, so there is a real incentive not to “waste” time on testing. And mobile phones are expected to interoperate freely regardless of vendor. This reads like a recipe for disaster: how can you claim that a system is secure if inadequate testing means that you can’t be sure that it is working as its designer intended, especially when it is working in conjunction with software and hardware from another vendor?

Of course, WAP security issues don’t end with the phone. A WAP-enabled Web site is a key part of a WAP solution, and all the security issues associated with Web sites still apply (probably more so, in practice, because of the fashion hype and time-to-market issues around WAP). Badly written CGI programs, for example, provide a classic way in for Internet hackers - and will probably be behind some exposures for WAP Internet access too.

### ***The Weakest Link***

Nevertheless, the phone is, and will probably always be, the weakest link. It is easily lost or stolen, and the capabilities of WAP (and broadband third-generation mobile phone technologies) mean that it is increasingly likely to be used for the storage of valuable data of varying degrees of importance - such as a list of customers, their phone numbers and even what you expect to sell them. Phone PIN numbers are some protection - but these only consist of four numerics and many users don’t bother to use a PIN anyway. There is also the risk of industrial sabotage - if your competitor depends on a WAP-enabled sales force, how effective will its salesperson in your area be once you’ve nicked his phone? Don’t get too hung up on WAP security until you’ve done a proper risk/threat assessment for the system as a whole, and, as usual, made sure that your security policy (and security awareness classes) are WAP-aware.

Nevertheless, physical WAP security is a factor in the threats facing you and you have to prioritise these threats. The real issues with WAP security today are bad enough, but experts like Chris Rouland, who is head of the ISS X-Force security team (see [www.iss.net](http://www.iss.net)), think the potential for security exposure is far worse. He worries about the intrinsic trust model in WAP, which means that if you compromise one part of the distributed system then security as a whole is compromised, and he isn’t happy with the design of the WML language. As is so often the case, security appears not to have been considered fully enough or early enough in the design - but at least security was considered in the design of WAP, which is a considerable advance on the state-of-the-art a few years ago.

### ***Potential For Viruses***

However, many of the threats associated with WAP remain potential threats for now. Phone viruses are feasible, but space and processing power in a phone is limited, and it will probably be hard to get them to reproduce efficiently - there isn’t room in phones for bloatware with macro autorun facilities. Nevertheless, we mustn’t be too complacent - there isn’t room in a phone for sophisticated anti-virus scanners either. Companies are already talking about anti-virus products for mobile phones - F-Secure’s Anti-Virus for WAP Gateways Release 5.0 (see [www.fsecure.com/news/2000/20000215.html](http://www.fsecure.com/news/2000/20000215.html)) apparently shipped early in 2001 - but many experts think this is a bit premature. F-Secure itself (on the page referenced) admits that “there are no known instances of in-the-wild malicious code for WAP-based devices”, but claims it is important to get the infrastructure prepared for when they appear. F-Secure’s product sits on the WAP Gateway and monitors material being downloaded to the WAP devices.

Rouland also thinks that WML (Wireless Mark-up Language), which is the WAP equivalent of HTML, may be a problem, as it works differently to HTML and programmers may not allow for this. For instance, he points out that it handles the user interface by assigning variables in the telephone and then putting them in the right place on the screen. The phones themselves, in current implementations, cannot differentiate between public and private variables, and rely on the server to clear variables in memory; if they don’t, passwords etc can stick around in memory where they can potentially be compromised.

---

*“Phone viruses are feasible, but space and processing power in a phone is limited, and it will probably be hard to get them to reproduce efficiently.”*

---

## Scripting

Later implementations of WAP will introduce scripting (WMLScript is roughly equivalent to JavaScript). The rise of macro viruses for MS Office and Outlook on the PC platform shows the potential for security exposures in advanced scripting environments. In the worst-case scenario, a malicious WAP-enabled Web page could suck the entire contents of your WAP phone (or, more worryingly, your PDA), including your address/phone book and any passwords, off your phone, for transmission to a third party. You can envisage wonderful possibilities for industrial espionage from this, but it is important to remember that these are, for the time being at least, only fantasies. Similarly, the rise of wireless technologies such as Bluetooth for connecting appliances at a distance has security implications (eavesdropping, for example), but this is far from being a serious issue today. The security model built into the Bluetooth specification looks adequate as it provides for verifying device identity, authorising the services you can access, and encryption to prevent eavesdropping - but the WAP security model looks OK until you look at it in detail.

## Good Practices

However, the potential for security problems with WAP shouldn't cause you to panic. Adding complex bolt-on security infrastructures to WAP will make systems more complex and may introduce additional points of management - a classic recipe for introducing bad procedures and human error which may, paradoxically, reduce security. Attention to security "good practices", such as the integration of WAP security with a directory-based enterprise security management environment, will allow WAP to be controlled adequately. It is vital that the "new technology" aspects of WAP don't cause you to ignore simple common-sense security practice:

- Hold security awareness seminars with WAP users. Point out the security measures that are available (changing default codes and so on), and the importance of using them.
- Use internal firewalls and so on to segment your network. Limit the parts of your network open to authorised users of Internet-enabled phones (this will also reduce traffic usefully).
- Embrace WAP in your official network strategy and use the availability of support as a carrot. This will discourage departments from building their own, insecure WAP applications and help you to keep WAP under control.
- Look at installing intrusion monitors to track whoever is using your network (if you haven't already). Make sure that they're aware of the possibility of WAP users and, ideally, that they can disable stolen WAP devices.
- Monitor your access logs, using suitable software assistance, if you're not doing this already.

Good system design, which considers security as part of the business requirements for a system, and uses WAP as an alternative channel where it is appropriate and where its security risks are adequately controlled, will allow you to exploit the potential of WAP (always supposing that it has one) without risk. Never forget that WAP is being over-hyped, which automatically hypes the security risk associated with it, and that FUD (Fear Uncertainty Doubt) is used by unscrupulous vendors to sell expensive security technology. It is quite possible to conceive of "killer applications" for WAP - locating the nearest restaurant in a strange town, for example - which have minimal security implications.

## Solutions

There are several obvious solutions to WAP security issues. For instance, you can keep the WAP Gateway under your control rather than place it with an external ISP. Your Gateway is still exposed if intruders breach your security, of course, but presumably you've set up a proper demilitarised zone and so on as appropriate to your needs and exposures. External testing of WAP applications, by specialists who are aware of the idiosyncrasies of the various phones and servers out there, is probably appropriate, although this doesn't mean that you can avoid your own testing, from early in the lifecycle.

Another risk control is to place value limits on transactions through the WAP

---

*"The mobile phone industry is driven by fashion, and time-to-market is king, so there is a real incentive not to "waste" time on testing."*

---

channel, or even limit the business functions it is available to. This may not be an issue in practice - your customers may appreciate getting near real-time account balances on their mobile phone, but have very little desire to complete complex, high-value financial transactions on an inch-square screen. User interface design and consideration of user comfort is vital to designing killer applications for WAP - applications that are so attractive users overlook the limitations of the technology - and applications must be designed as a whole, with WAP as just one possible channel.

Use of proper identity technology - digital signatures based on dual-key encryption - offers interesting possibilities for enabling secure WAP applications by identifying the phone holder, but the implementation of this is far from trivial. It is obviously vital that you do authenticate the person using the phone, not just the phone itself, and this probably implies the use of biometrics. However, there are worrying possibilities for "spoofing" biometrics - for arranging for a fake fingerprint reader, for example, to always return "person checks out" when queried - without even bothering to compromise the biometric itself. In addition, you'll need coercion procedures - processes that allow transactions to continue when they are being made under duress, but which alert security invisibly - to protect the users of WAP-enabled phones for high-value transactions.

### **Conclusion**

As the use of WAP increases - if it does - no doubt its security issues will be addressed. Interoperability certification will be commonplace, so differences between phone interfaces and differences in basic technical security functionality will disappear; the testing of WAP will become easier. The power of phones will increase, so that security overheads will become unimportant, and zeroising memory (for example) after every call will be more feasible. Once the initial hype dies down, people will have time to design more robust phone security architectures. Nevertheless, there will be consequences from the current security shortcomings of WAP. Take-up for serious business applications, even those for which it is suitable, will be slower than anticipated. Supporters of rival systems, such as iMode from NTT/DoCoMo in Japan (see the unofficial iMode FAQ at [www.eurotechnology.com/imode/faq-gen.html](http://www.eurotechnology.com/imode/faq-gen.html)) may well see a window of opportunity in the European marketplace, although when looked at closely these rivals will doubtless have security issues of their own.

As far as the network is concerned, you'll need to be aware of WAP and its implications for security (such as running your WAP Gateway on your own network, and hardening it against intrusion). However, in practice, it is unlikely to be a major issue for some time.

---

*“Security firms such as ISS think that there are serious and fundamental problems with the WAP security model, and implementation issues that may compromise security.”*

---

**PCNA**

Copyright ITP, 2001

## New Reviews from [Tech Support Alert](#)

### [Anti-Trojan Software Reviews](#)

A detailed review of six of the best anti trojan software programs. Two products were impressive with a clear gap between these and other contenders in their ability to detect and remove dangerous modern trojans.

### [Inkjet Printer Cartridge Suppliers](#)

Everyone gets inundated by hundreds of ads for inkjet printer cartridges, all claiming to be the cheapest or best. But which vendor do you believe? Our editors decided to put them to the test by anonymously buying printer cartridges and testing them in our office inkjet printers. Many suppliers disappointed but we came up with several web sites that offer good quality cheap inkjet cartridges with impressive customer service.

### [Windows Backup Software](#)

In this review we looked at 18 different backup software products for home or SOHO use. In the end we could only recommend six though only two were good enough to get our "Editor's Choice" award

### [The 46 Best Freeware Programs](#)

There are many free utilities that perform as well or better than expensive commercial products. Our Editor Ian Richards picks out his selection of the very best freeware programs and he comes up with some real gems.