McAfee®

Protect what you value.

# Mobile Malware: Threats and Prevention

By Zhu Cheng

# Mobile Malware: Threats and Prevention

Carrying a "smart" mobile phone is almost like having a powerful computer in your pocket. An increasing number of phones sold today—especially in Europe—include not only a camera but also extensive online access, keyboards, and other typical computer functions.[1] However, with power and convenience comes a cost. Just as our desktop and laptop computers face security threats, so do these smart phones. The unfortunate irony of this evolution is that the greater the functionality gains, the more vulnerable smart phones become to the same types of threats that plague our laptops and desktops.

The most common operating systems used by smart phones and personal digital assistants (PDAs) are Microsoft Windows CE and the Symbian OS. Windows Mobile 2003 and Windows Mobile 6 are based on the Windows CE, which has an open-source kernel strategy. The operating system S60 is based on Symbian OS, a closed-source operating system that is developed and maintained by Nokia. (Many other products use S60, including phones from Samsung, Panasonic, Motorola, and Lenovo.) Because of Microsoft's open-source policy with Windows CE, more smart phone device manufacturers have begun to adopt it. At the same time, the open-source code is attracting more and more malware writers, so the growing security problems of smart phones are becoming a real concern to users.

In this paper, I discuss vulnerabilities in smart phones based on Windows Mobile. Most of these vulnerabilities are also found on the Windows Mobile PDAs. Because the Symbian OS is similar in many ways to Windows Mobile, Symbian phones may also suffer from these problems.

Based on detection data from McAfee Avert® Labs, we've seen rapid growth in mobile malware, and we foresee a continuation of this trend for the rest of the year.[2] What are the primary reasons for increased threats?

- The price of smart phones continues to drop, and more vendors are involved in smart phone production.
- Windows CE's open-source kernel policy allows virus writers to gain a deep understanding of the operating system.
- Smart phone users tend to input a great deal of private data into their devices; this appeals to virus writers because of the potential financial gains from identity theft or misappropriation of credit card information.
- As smart phone hardware capabilities increase, the operating system functionality also increases, which means malware authors always have new opportunities for exploitation.
- Developing software under Windows Mobile and Win32 is very similar, so it's easy for authors of Win32 malware to transition to mobile malware.

[1] http://telephia.com/html/Smartphonepress_release_template.html

[2] http://www.mcafee.com/us/about/press/corporate/2006/20061129_080000_f.html

## Which Features Are Most at Risk?

**We see the greatest threats to mobile phones in these seven areas:**

- Text messages
- Contacts
- Video
- Phone transcriptions
- Call records
- Documentation
- Buffer overflows

### *Text messages*

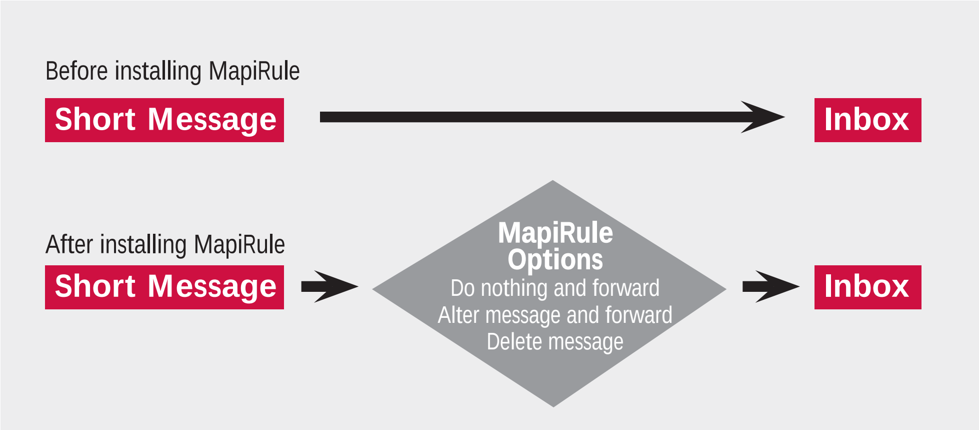
Windows Mobile provides a development API that mainly provides functions for sending and blocking messages. These functions can be used by viruses or other malware to steal your private information and potentially wreak havoc on your life and finances.

Researchers at McAfee Avert Labs have observed examples of SMS (short message service) phishing (also known as SMiShing), which seems to be on the rise.[3] One example is malware that uses the text-messaging APIs to send fake messages to people on your contact list. This is similar to email spoofing, but this type of phishing has an even higher likelihood of success because of the victims' lack of awareness of this type of threat. If we trust an incoming message based solely on its telephone number, then we are vulnerable to anyone in our contact list who has been infected by a virus, which can easily send spoofed messages. Users will find it hard to tell if the SMS is malicious.

A virus can also use text message APIs to charge cell phone fees through the SMS payment gateway. Although we haven't yet seen such a virus on Windows Mobile devices, Nokia's S60 operating system has already been victimized by this threat. This virus sends special text messages to a Russian service provider and deducts the users' prepaid cell phone fees. It's reasonable to assume similar attacks will occur against Windows Mobile devices as these devices become more popular.

It wouldn't be difficult for a malware writer to create a new threat. According to the Windows Mobile Software Development Kit, an application developer could write code using the sample code MapiRule and load it to implement text message blocking. Because Microsoft already provides a MapiRule framework in the SDK, all that a developer has to do is modify it a bit for use as a DLL. The figure below shows the short message handling process before and after MapiRule has been installed. After installation, MapiRule becomes a filter between short messages and the tmail (text mail) mail program. So, a programmer could interrupt the short message handling process by deleting or forwarding messages, or by performing other operations while acting as the man in the middle. Malware could use this feature to install a DLL in the user's smart phone to block the short message and disturb normal communication, give responses to messages, or forward messages. If SMS was used for corporate communications, it would create an avenue for intercepting corporate data.

[3] http://www.avertlabs.com/research/blog/?p=75
http://www.f-secure.com/weblog/archives/archive-042007.html#00001173



*Using MapiRule code with SMS allows hackers to create a “man in the middle” threat.*

In spite of this danger, there’s no reason to panic! MapiRule’s short message blocking technology uses a Microsoft-provided fixed port, so users can easily determine whether their Windows Mobile systems have picked up a DLL. To install to this port, the malware must have registered the filtering DLL module and then added the DLL’s CLSID key under:

```
[HKEY_LOCAL_MACHINE\Software\Microsoft\Inbox\Svc\SMS\Rules]
```

and must have set the key assignment as 1, for example:

```
"{3AB4C10E-673C-494c-98A2-CC2E91A48115}"=dword:1
```

This key assignment indicates that the system had a filter DLL installed for MapiRule. Readers should, however, not delete any content under that item—because DLLs found here may not have been installed by malware. If the user removes the wrong keys, an important program could fail. If you find such a key, rely on your smart phone antivirus software for a fix.

## *Contacts*

In a corporate environment, the contact list is one of the most important features of a smart phone. Theft of corporate contact data could have dire consequences for the employee and the company. We already mentioned that mobile viruses can “steal” a contact list and send out short messages containing a virus. It would be even worse if the malware packed your contact information and sent it to a malicious third-party. Many smart phone users take advantage of the phones’ contact backup tools, which typically use programming calls such as IPOutlook, ItemCollection, IFolder, and IContact from the Pocket Outlook Object Model APIs in the Windows Mobile SDK API. Malware developers could easily use these calls to get and modify contact information and send the results to someone else.

## *Video*

Most smart phones now have a video and still camera. Through Microsoft’s APIs, mobile malware could take over the phone and use the camera to snap photos, though it would probably be difficult to get a good angle when the virus wants to use the camera. So, the possibility of automated exploitation using the camera is pretty small. However, the security of the photos and video already on the device is much easier to exploit. A virus could search for all JPG files through the file API and send those files to a malicious third-party via the wireless network. Although the images can be large, there’s generally plenty of bandwidth available in any country with a widespread 3G mobile network.

### Phone transcriptions

What if your mobile phone were to suddenly turn into a tape recorder? Using the mobile voice-recording API, a virus could indeed change a mobile into a tape recorder. Microsoft applies the Waveform Audio Functions to record and play Wav files, according to the Windows Mobile SDK. Because of the comparability between Windows Mobile and Windows, many recording APIs and codecs used by Windows can be applied to Windows Mobile—and serve as a reference for mobile malware authors. When we tested the Dopod smart phone, for example, we found that the recording quality was very high—even when the mobile was in a user's pocket. Smart phones have limited storage space, however, so malware cannot record indefinitely. But, it could send the recorded file to an attacker via email or via the Multimedia Message Service (similar to SMS). If the attack were combined with the SMS interception technology we've already discussed, the malware could use SMS to activate the recording function, turning a mobile into a tape recorder that could be turned on and off remotely.

### Call records

Call records are not particularly valuable, so there are only a few viruses that exploit this function; but malicious programs can still read this information. Users should pay attention to their call records and perhaps periodically delete unnecessary records to lessen the severity of an infection.

### Documentation

Many smart phone users read and store Word, Excel, or PDF files on their mobile phones—especially PocketPCs. But malware can also steal these files, again using the file API function. Files with the extensions *.doc, *.xls, and *.pdf are likely to become popular targets for mobile malware thieves. Smart phone users are advised to exercise caution when saving sensitive corporate data or private files on their phones.

### Buffer overflows

Buffer overflows also plague mobile devices. Way back at Xcon 2005, we saw a presentation on hacking Windows CE.[4] The talk included shell code development advice as well as sample code.

## How to Prevent Mobile Malware Attacks

To protect the Windows Mobile API, Microsoft by default employs a certificate system. Only programs with signed certificates can call mobile APIs. This system works well until a user wants to add an unsigned program. One way around the default security is to use a tool such as Novosec's SDA_ApplicationUnlock, which completely disables certificate security on a mobile phone. The danger of unlocking, however, is that it reduces the security of the device. Once you unlock, any program—including malware—could use the API calls. Our advice is simple. Don't use programs like this if you want to keep your smart phone secure. (For more information, take a look at Microsoft's overview article on Windows Mobile 5.0 Application Security.[5])

[4] http://www.phrack.org/issues.html?issue=63&id=6#article

[5] http://msdn2.microsoft.com/en-us/library/ms839681.aspx

## An ounce of prevention

The best way to protect your mobile device is to keep malware off in the first place. Use the same precautions for your smart phone as you would for your Windows laptop or desktop computer. Anti-virus and anti-malware tools to prevent infection are more effective solutions than products that only detect or clean viruses. After your mobile has been infected by a virus, removal can be complicated. It's best to use a combination of both PC-based anti-virus software (with on-access scanning enabled) and mobile anti-virus software. Mobile users also should follow the same safe browsing practices they observe at their computers. We also recommend that users accept only programs that bear digital signatures—programs that have passed the certificate test and are developed by legitimate commercial software vendors.

## Install process management

Using process-management software, you can search for suspicious processes on your mobile phone and stop them. Windows Mobile cannot run too many processes because of hardware limitations. So, log all the running processes when you're sure the mobile is not infected. Any time thereafter, it should be easy to spot a malicious process and stop it by following the advice of the mobile anti-virus software.

## Be careful with Wi-Fi and Bluetooth

Disable Wi-Fi and Bluetooth when you're outdoors. These functions are easy to exploit for sending malicious code or viruses. It's also possible that sensitive information could be intercepted by a sniffer when these functions are enabled. The safest place to use these functions is at home or at trusted locations.

## Watch for unauthorized GPRS connections

If you find your phone is auto-connected to GPRS (General Packet Radio Service), then your mobile might be infected with a virus that is sending your data to other parties. If you discover this problem, disconnect the device immediately and install anti-virus software to remove the malware.

## Back up frequently

Contact lists are vitally important to the company you work for and to you personally. If the list is lost or stolen, the consequences can be disastrous. It's a good practice to make frequent backups of data stored on mobile devices. Then, even if your mobile device has been infected, you can recover the default phone settings to clean the system.

## Install mobile anti-virus software

The majority of large security software vendors now have a mobile version of their anti-virus solutions. It's time to give your smart phone the same protection you give your desktop system.

## Do not save business data on your mobile

Save confidential files or photos on removable disks. Don't save them on a mobile device. Smart phones and PDAs are simply not very secure. The profit motive is driving malware writers in increasing numbers to create mobile viruses. The bottom line is that the danger is not going away.

## Suggestions for developers

If you are a developer of smart phone applications on Windows CE devices, we recommend using Microsoft Visual Studio .NET 2005, which adds the security_cookie function to prevent buffer overflows. Here's a code snippet:

```
STMFD   SP!, {R4,LR}
LDR     R12, =0xFFFFFA5C
ADD     SP, SP, R12
LDR     R3, =__security_cookie
LDR     R3, [R3]
STR     R3, [SP,#0x5AC+var_C]    ; security_cookie
...
LDR     R12, =__security_cookie
LDR     R12, [R12]
CMP     R0, R12
MOVEQS  R12, R0, LSR#16
MOVEQ   PC, LR
```

Using this method, Visual Studio strengthens the program's security. But programmers should not depend solely on this protection method at the edit level. The most important thing is to check the length of the character string that transmits to the buffer—to make sure that it will not cause an overflow.

## Conclusion

Smart phones that use the Windows Mobile operating system are convenient and are growing in popularity. But because of the many APIs, users lack of security awareness, and the powerful promise of financial gain, malware writers continue to create viruses. The majority of today's mobile malware do not present a significant risk to users, but we can't let our guard down. Right now, we're in the early stages of what is likely to become a longstanding trend. It will probably only get worse, so it's essential to exercise caution when using your smart phone.

*Zhu Cheng is a research scientist with McAfee Avert Labs in Beijing. When he's not writing code for McAfee's Network Access Control product or digging for mobile malware, he enjoys playing ping pong.*

McAfee, Inc.
3965 Freedom Circle
Santa Clara, CA 95054,
888.847.8766
www.mcafee.com