
CNCCS
Consejo Nacional Consultivo de Cyber-Seguridad
Smartphone Malware

About CNCCS:

The National Cyber-Security Advisory Council (CNCCS) is a private organization with founding members including AEDEL, Amper, Bdigital, EIIEO, Eside Deusto, Hispasec, Indra, Informática 64, Kinamik, Optenet, Panda Security, S2grupo, Secuware, TB security, Isec Auditors and S21sec. The Council's mission is to provide organizations that operate in Spain (governmental or not) the knowledge and experience of the council members regarding national and global cyber-security issues in order to make the Internet and information networks safer, and boost innovation and financial growth.

CONTENTS

1. INTRODUCTION

Cell phones have evolved into small personal computers with many of the same functions and features of desktops and laptops. Frequently designed with a disregard for security, their popularity poses new risks. This report summarizes the dangers posed by malware for mobile devices, with a particular focus on smartphones.

Our approach provides a global view of the problem and creates a solid basis to forecast future trends and confirm existing ones. We have taken into account technical aspects (platform-specific peculiarities), financial aspects (market evolution will undoubtedly determine potential targets of cyber-crime) and historical aspects (from early proof-of-concept prototypes to modern-day fraud applications).

We also provide historical background on the most important milestones in the development of smartphones starting with their origins at the end of the 1990s.

2. HISTORY

Smartphones were originally created as mere cell phones but have since evolved into true computers. Today, it's normal for these devices to include schedulers, enhanced video and music players, advanced connectivity options and a myriad of other functions that might have seemed inconceivable just a few years ago.

The first smartphone was the IBM Simon. It was designed in 1992 and shown as a concept product that year at COMDEX, the computer industry trade show held in Las Vegas, Nevada. It was released to the public in 1993 and sold by BellSouth. Outside of its phoning capabilities, it also contained a calendar, address book, world clock, calculator, note pad, games, email, ability to send and receive faxes, and even a PCMCIA card reader.

In 1996, Nokia launched the Nokia 9000, which combined the features of a personal digital assistant (PDA) with a wireless phone. This vendor's models were the first to feature color screens and WiFi connectivity. In the next few years these features would become standard on high-end smartphones. The Nokia 9210 Communicator was the first model to use the Symbian operating system.

In 1997, Ericsson released the concept GS88 phone, the first device actually labeled as a 'smartphone'. Later models would include a touchscreen.

The number of smartphones in the marketplace increased significantly after 2000. The primary landmarks were the release of the Windows CE Pocket PC OS and particularly the Research in Motion (RIM) Blackberry smartphone in 2002. This was the first smartphone optimized for wireless email use.

During these years, all vendors evolved their products into multi-function devices with features that would become standard equipment in smartphones (connectivity, touchscreen, multimedia and management applications, etc.).

In 2007, Apple Inc. launched the first generation iPhone device. This was the first cell phone managed through a touchscreen display, which revolutionized the mobile phone market. Over the past few years Apple has released new versions of its iPhone with 3G support and its App Store, which allows users to browse and download applications for their devices.

In August 2008 Android was released. Android is an open-source Linux-based platform that became the flagship software of the Open Handset Alliance, an organization created by Google in 2007 and made up of different hardware, software and telecom companies (Intel, HTC, Dell, ARM, Motorola, among others) devoted to advancing open standards for mobile devices.

The HTC Dream (also marketed as T-Mobile G1) was the first phone to use the Android platform. Its software included Google services like Maps, Calendar, Gmail, and the Chrome browser. It also incorporated third-party free and paid applications, which were made available in the Android Market.

Following the popularity of the App Store, competitors developed their own software stores: RIM launched its application store BlackBerry App World, Nokia released Ovi Store (May 2009), Palm published Palm App Catalog (June 2009) and Microsoft released Windows Marketplace for Mobile (October 2009).

Google launched Nexus One in January 2010. This was introduced as a ”superphone” with Android 2.2, and sold via Google’s website in the United States and through carriers such as Vodafone in other countries.

Now that we have taken a look at the major milestones in the history of smartphones, we will provide an overview of each platform's and model's market share, with the aim of understanding what smartphone models are most likely to become the next big target for cyber-crooks.

3. SMARTPHONE MARKET EVOLUTION

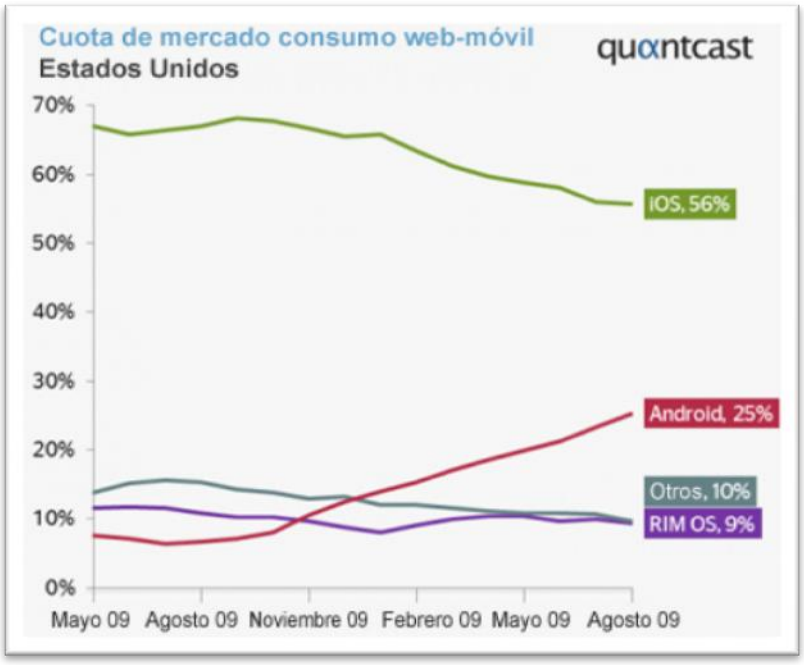
Smartphone penetration is very high among cell phone users and sales growth has been rapid, although different than what some expected; Gartner predicted in 2006 that Windows Mobile would be the leading mobile platform by 2010 and overtake Nokia. This is actually far from reality. Nokia, despite its market penetration, seems to be decreasing in popularity and its models look old-fashioned compared to some competitors.

The smartphone "revolution" began with the introduction of Apple's iPhone back in 2007, and today, the battle is between iPhone and Android-based devices.

The introduction of touchscreen devices changed the market. Users could enjoy an easier to use, more intuitive interface. The wider LCD display allowed them to experience PC-like Web browsing and enjoy all forms of high-quality multimedia content.

Hardware has also evolved, and it is now a regular occurrence to find devices coming with 1 GHz processors, 512 MB of RAM, and all kinds of features such as GPS, Bluetooth and a compass or accelerometer, introducing new possibilities for developers.

The creation of flat-rate data plans from operators has played a major role in popularizing this type of device, and subsequently also increased use of its associated services – from email to online banking.



As you can see from the graph above, Android-based devices are on the rise and gaining market share on its rivals. iOS is on the decline despite maintaining market leadershipBlackBerry has dropped slightly in popularity over the last few years.

OS	2009	2010	2011	2014
Symbian	80,876.3	107,662.4	141,278.6	264,351.8
Market Share (%)	46.9	40.1	34.2	30.2
Android	6,798.4	47,462.1	91,937.7	259,306.4
Market Share (%)	3.9	17.7	22.2	29.6
Research In Motion	34,346.8	46,922.9	62,198.2	102,579.5
Market Share (%)	19.9	17.5	15.0	11.7
iOS	24,889.8	41,461.8	70,740.0	130,393.0
Market Share (%)	14.4	15.4	17.1	14.9
Windows Phone	15,031.1	12,686.5	21,308.8	34,490.2
Market Share (%)	8.7	4.7	5.2	3.9
Other Operating Systems	1,.431.9	12,588.1	26,017.3	84,452.9
Market Share (%)	6.1	4.7	6.3	9.6
Total Market Sales	**172,374.3**	**268,783.7**	**413,480.5**	**875,573.8**

It's important to consider the future of the smartphone, as as malware creation will most surely be determined by its evolution. Even though cyber-criminals will try to exploit every platform available to them, logic dictates that the most popular ones will suffer most attacks.

Today, iOS and Android dominate mobile Web consumption, despite the fact that Symbian still leads the worldwide smartphone market with just over 40 percent of the market, followed by Android with 17.7 percent and BlackBerry with 17.5 percent. Gartner predicts that by 2014 Android will become the most popular platform, and smartphones will outsell PCs by 2013. This should give you a clear idea of the relative importance of security for each of these devices.

4. MOBILE DEVICE SECURITY

Mobile devices present an attractive target to cyber-criminals. They are used everywhere, contain a vast amount of personal and confidential information and can be used to perform all kinds of online transactions.

A common smartphone security concern is their communication channels. In this sense, they are more vulnerable than traditional PCs and can be subjected to various attack vectors – SMS, Bluetooth, WiFi, Web browsers, applications and email --an aspect that can result in the proliferation of malicious code targeting these platforms.

Cell phones are really personal devices and it is precisely this personalization that makes them even more unsafe. You might have one computer for a family but every family member has a personal device and it is with them all the time. Contributing to their security risks is that many users are unaware they can be a security hazard and that battery limitations of the devices themselves prevent running complex applications like antivirus solutions.

4.1. False Sense of Security

From a purely physical point of view, smartphones feel like very personal devices. You carry them around with you and control their operation, which can make you believe they are less accessible to intruders. This false sense of security, combined with phones often linking to personal email applications, social networks and multimedia content, can lead to private and confidential information being stored, sometimes inadvertently.

This false sense of security can sometimes make users overlook basic precautions such as changing default device security settings.

At present, the number of smartphone attacks is quite small compared to attacks on PCs. There are more than 60 million known malicious programs for PCs as opposed to 600 for smartphones, although we expect an increase in the amount of malicious code targeting the latter.

With regard to this, on September 25, 2010, S21sec reported the first malware strain capable of bypassing the two-factor (PC client and cell phone) authentication systems used to enhance online banking security. These systems are discussed at a later point in this report.

There are many factors that will contribute to the proliferation of new threats. The only security mechanism implemented in most smartphones is a password, making the security and reliability of downloaded applications integral to smartphone safety.

It is advisable to have encryption mechanisms to block access to lost or stolen devices. This is particularly important in organizations and companies where there is a thin line between corporate security policies and personal use due to the popularity of these devices among the general public.

Smartphone security must not only be considered from the end user's point of view. It involves many other aspects like the smartphone's operating system kernel, application deployment and platform development environments.

4.2. Application Development Kits (SDKs)

The top mobile platforms (Blackberry, iPhone, and Android) offer software development tools (SDK) to facilitate development of applications. Each SDK has its own characteristics, many of which aim to improve security, like encryption, hardware access restrictions and memory

administration. Without going into too much detail, these are the main development environments for each platform:

- **Android SDK**: Supplied by Google. Applications for the Android platform are developed using the Java programming language. Therefore, the Android apps run using a special version of the Java Virtual Machine called Dalvik. Even though there are other development environments available for Android, Eclipse is the most popular one.Unlike its competitors, Android is an open platform.

- **Blackberry SDK**: Supplied by RIM, the company behind the Blackberry. The operating system used by RIM is a proprietary multitasking environment and Blackberry relies on JavaME for applications. The developed applications need to be packaged to keep the security of the RIM operating system. All applications must be digitally signed in order to associate them with a developer account.

- **Nokia SDK**: In the past, you needed a Symbian SDK to develop Nokia applications. Now Nokia uses the Web Runtime (WRT) platform, which is more accessible to developers.

- **iOS**: To keep the the iPhone tightly protected to ensure its security and stability, Apple offers its own software developer's kit for iOS. The latest iOS4 version has eliminated some restrictions, allowing use of intermediate development environments. This has opened the door for applications such as Flash, Java, Silverlight or Mono.

A recent BBC article explains how they downloaded an application development kit, learned some basic concepts about how to program in Java and compiled some code fragments available on the Web. With all these ingredients, in a few weeks they managed to create a game which, inadvertently to users, collected contacts, copied text messages and traced the cell phone location. All this information was then sent to a predetermined email address. The entire game was only 1,500 lines of code, with spyware taking up about 250 lines. The application was tested on a cell phone, but was not uploaded to any app store.

Developers must be responsible and always informed about the data their applications can access. However, on many occasions not even they know all the functionalities of their code as it combines with third-party applications. There is no doubt that controlling the applications available for these devices will be a key factor to safeguard end users' security.

4.3. App Markets

The App Store model created by Apple has been copied by every other cell phone vendor. This model is based on shifting security responsibilities to a central distribution point, where each developer sets their own application distribution rules. All apps are tested before being released to the market, to ensure that they work and are free of any malicious code.

- **Apple App Store**: For an application to be published on Apple's Store it must first be approved by Apple. Also, to get approved, developers must create a developer account and pay an annual fee. Apple ensures that the application works as expected and does not have any detrimental effects on the iPhone stability.

- **Android Market**: Google, on the contrary, never vetoes applications uploaded to Android Market. Google has its own rules, but delegates all software responsibilities to the user. Android's protection against malicious applications is a security model based on "capacities". Each Android app that is installed must tell the OS what capacities it requires. When an application is installed, the Android OS lists all the "capacities" that is required in order for the application to run. However, it is the user who decides if these capacities are consistent with the application's functionality.

  In any event, Google will remotely disable apps found to be malicious. Like Apple, it demands that developers register and indicate the permissions that their applications need to interact with the phone.

- **Nokia OVI Store**: Like Apple, they can disapprove applications uploaded to their market.

- **BlackBerry App Store**: Like Apple, they can disapprove applications uploaded to their market.

To summarize, Apple's software model is more 'closed' compared to Android's open model. Blackberry and Nokia are similar to Apple. These three vendors take responsibility for the applications available on their App Stores, whereas Android delegates security to developers. Each model has pros and cons, and none of them has been able to completely eradicate infected apps.

Finally, we would like to point out that trusting app stores as the first security barrier for mobile devices is simply not enough. It is extremely important to keep good programming and data storage practices.

It is worth mentioning that the diversity of mobile phone technologies, compared to Windows' PC dominance, can work against the proliferation of malicious code in this environment, as malware creators must write the most appropriate malicious code for each platform.

5. MALWARE ON MOBILE DEVICES

The smartphone malware phenomenon is no longer in its early stages in terms of development and maturity. Even though the last decade has seen the appearance of several smartphone malware families, so far very few of them were designed to generate revenue for the malware writer (mainly by sending SMS messages to premium numbers controlled by the hacker). Even less could steal banking credentials or attack platforms to target online transactions.

As you will see in this section, long gone are the days when only proof-of-concept malware was created. Nowadays, mobile malware is much more often designed for financial gain.

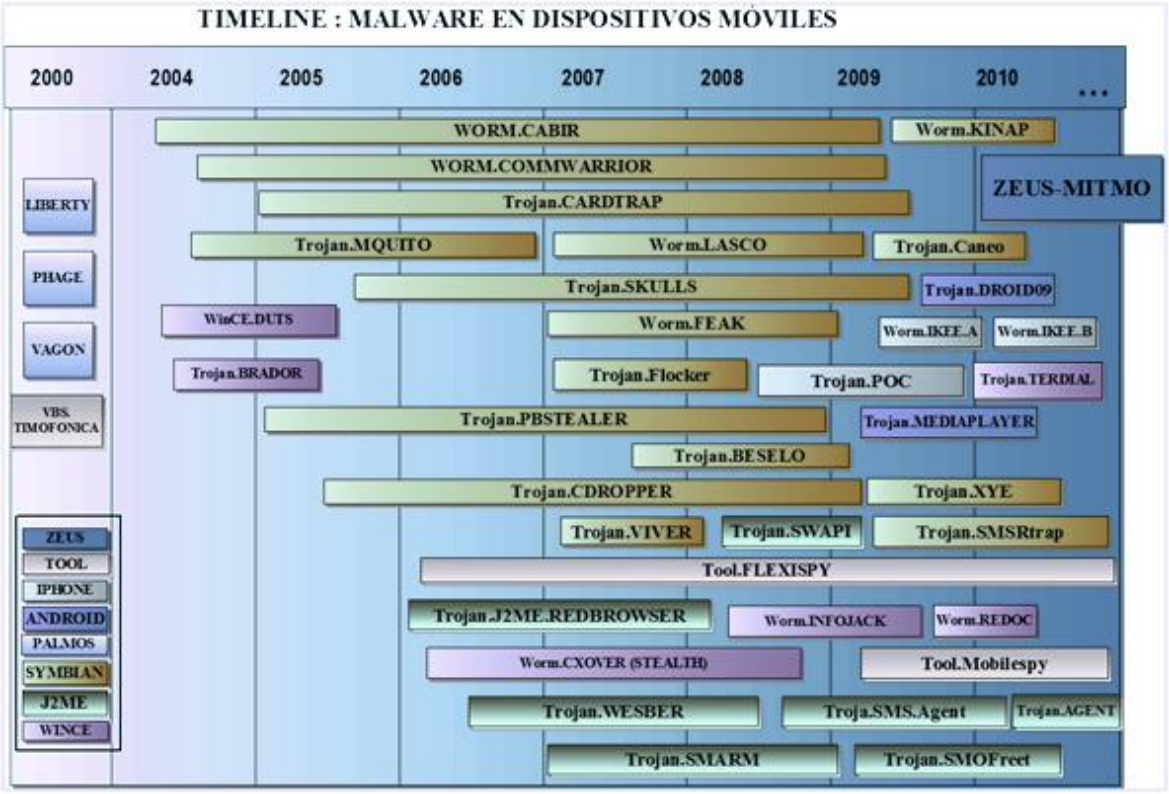
As predicted by many forecasts made over the past few years, cyber-crime is expected to extend to other platforms, either to look for additional propagation and monetization channels, reinforce its infrastructure or simply attack the second authentication factor (cell phones) used by online banking services. As recently seen (Zeus-Mitmo Trojan explained later on in the report) the 'modus operandi' adopted by cyber-criminals has changed. The objective is now to multiply the effects of their infections and attacks, affecting as many devices as possible.

The graph below shows the timeline of the most significant malware strains appeared since 2000, including their category and the target operating system/platform. We have also tried to show each malware family's persistence over time, according to the following criteria:

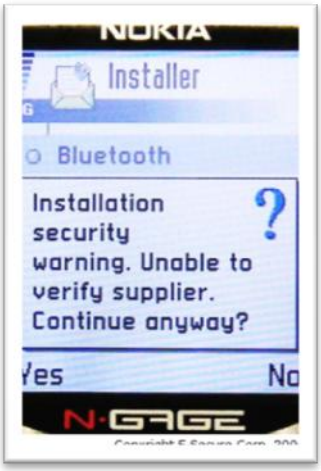
- Number of documented incidents during the malware lifecycle.
- Number of malware samples and variants over time.
- Statistics from several antivirus firms.
- Lifecycle of the device the strain was designed for.
- Malware peculiarities (obsolete code, infection viability, distribution, objective, etc.).

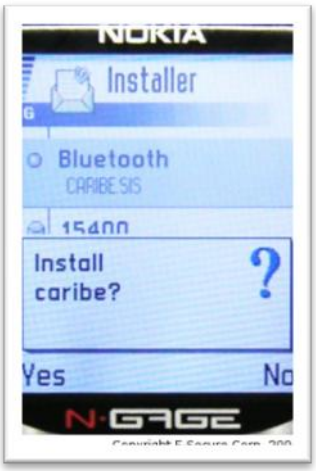
The graph starts in 2000 with four major malware strains. Three of them targeted Palm OS platforms and could be considered the first viruses aimed at a smartphone ancestor, the PDA.

The fourth one is **VBS.Timofonica**. Even though it was initially designed to attack Microsoft platforms, **VBS.Timofonica** actually became the first virus to spread by email. It sent a copy of itself not only to all contacts it found on the infected system, but also to subscribers of the movistar.net service (which allowed users to receive email messages on their cell phones). The binary sent messages to random numbers within the Movistar carrier range: MOVISTAR_PREFFIX+random_6_figure_number@movistar.net.



The first malicious code for smarphones appeared in 2004. More precisely, the first sample was received on June 14 by the Kaspersky antivirus company from a well-known Spanish virus collector known as VirusBuster. **Cabir** was written as a proof-of-concept virus by someone going under the name “Vallez”, a member of a group of virus writers called 29A. The group’s aim was to demonstrate that it was possible to infect non-standard operating systems and applications, like Symbian OS in this case. It used a .sis file and was capable of spreading via Bluetooth.





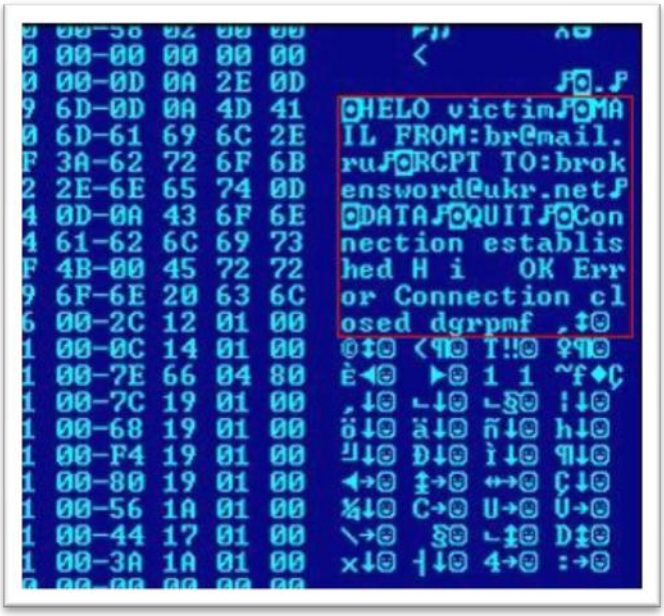
Towards the end of 2004, several versions of Cabir appeared, far more damaging than the original one. Unlike Cabir.A, which could only replicate to one device at a time, these variants could spread massively to any phone via Bluetooth.

Even though this was not considered malicious code per se, a similar specimen appeared a few months after Cabir: **Mosquito**. Mosquito or Mquito was designed specifically to attack the Symbian platform. Mosquito was a copy-prevention system implemented by the company Ojum Software, which included a hidden feature in its games to allow the detection of illegal software. So if a "cracked" or illegal version of the game was developed or Mosquito was played on an unregistered smartphone, the Trojan dialed a specific number silently in the background—sending an SMS message notifying the company.

A month after Cabir appeared, antivirus companies were startled by another technological innovation: "**Duts**" (Win.CE4.Duts.a). This was the first known virus for the Windows CE platform, and it was once again created by a member of the 29A group. It was actually a proof-of-concept virus and had no malicious payload. Neither was it released "in the wild" as the first detected sample was sent directly by the malware writer to various antivirus security firms.

Once infected, the device displayed a message with the question "Am i allowed to spread?" Then, if the user clicked Yes, Dust infected all executable files found on the directory where it was installed.

A month after Dust was born, **WinCE.Brador.A** made its appearance. This was a binary file that acted as a backdoor Trojan and was developed in assembly language for PocketPC. The malicious program opened port 2989/TCP in the victim device, waiting for the attacker to establish a connection and take control of the compromised device. Brador could also copy files from/to the device, run executable files, display messages and even send folders to the remote user. Finally, it sent an email message to brokensword@ukr.net (supposedly the attacker's address) with the IP address of the compromised computer. This code was not designed to spread on its own.



In November 2004, after a three-month break, a new Trojan appeared: **Skulls** (Skuller.A), one of the most prolific families of malware targeting cell phones. Skulls acted as a Trojan and delivered a more dangerous payload than previous malware. Skulls would overwrite original applications which would cease to function. It was the first program to exploit a vulnerability in the Symbian architecture to write files without the necessary privileges. Launching and installing the program on the system led to the standard application icons being replaced by a single icon, a skull and crossbones. It used several means of propagation, particularly email and P2P file sharing networks under the name Extended-Theme-Manager. Some Skulls variations dropped a Cabir variant on affected computers. This aimed at ensuring propagation of the latter Trojan.

The source code for the original Cabir appeared on the Web in late December 2004, and caused an avalanche of new variants which were more virulent than the original,which simply acted as a worm spreading via Bluetooth.

Cardtrap had a similar payload to Skulls (it overwrote files) but is considered the first cross-platform mobile worm as it affected Symbian OS and Windows. It copied a worm targeting Microsoft's platform (more precisely a Padobot variant), together with the autorun.inf file in an attempt to start automatically if the phone memory card was inserted in a PC. Cardtrap was one of the first malware strains capable of infecting a mobile device and a standard computer.

It spread by email or through P2P networks normally under the name Black_Symbianv0.10. The second Cardtrap variant was especially dangerous as it rendered the phone useless on reboot.

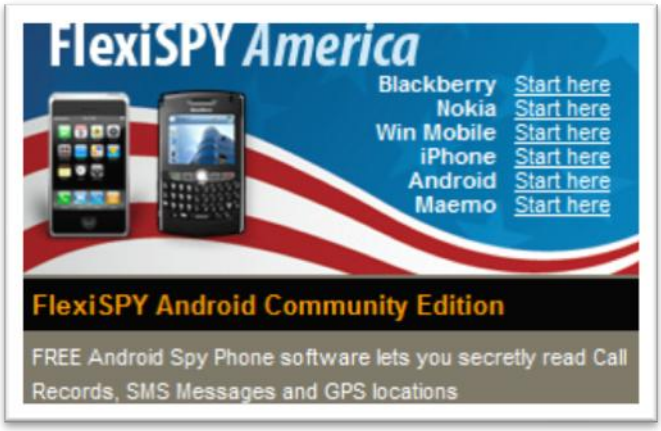
The end of 2005 also saw the emergence of **Pbstealer**, one of the first binary files that could steal confidential information from cell phones. Pbsetaler copied the user's phonebook to a text file, and transmitted it to Bluetooh-enabled devices that were within range. Pbstealer did not spread automatically using its own means.

From 2006, the development and implementation of new technologies and systems led to a new wave of smartphone malware. This coincided with an increased sense of danger and the development of monitoring solutions for cell phones.

The appearance of the **REDbrowser** Trojan in February 2006 was an unpleasant surprise. This malware specimen sent a flood of SMS messages to several premium services in Russia, charging the user for each message. This was the first J2ME mobile phone Trojan, which allowed it to work on most phones with J2ME support (i.e. hundreds of different cell phones). The user was continually prompted to allow the SMS messages to be sent. REDbrowser pretented to be a WAP browser offering free WAP browsing and SMS messages.

The first version of the **CxOver** worm was released in March 2008. This was the first malicious code to use stealth techniques and infect devices without user intervention. It was programmed by Dr. Julius Storm in a language known as Microsoft .NET Intermediate Language (MSIL). It infected Windows operating systems with .NET Framework Support as well as mobile devices. It spread via the ActiveSync protocol. Once it infected a computer, Cxover copied itself to any cell phone that used the protocol for synchronizing.

Flexispy appeared in March 2006. Flexispy was a mobile phone monitoring application that secretly recorded all activity on the phone. It sent a copy of all email and MMS messages to a password-protected website. The **Mobler** worm emerged just a few days later and copied itself to any device with write permissions.



Viver.A appeared almost a year later (in May 2007). This Trojan's malicious payload was to send SMS messages to a premium-rate number, which the hacker was paid commission for.

Almost a year passed without any new significant developments until the appearance of iPhone.A in January 2008. iPhone.A pretended to be a firmware update but once installed overwrote some system utilities.

A month later, in February 2008, a new epidemic broke out. **Infojack** run on WINCE platforms, stole information from affected devices and sent it to an Internet server.

2009 was dominated by new variants of known malware specimens, as well as some developments with regard to potential targets. One of the most notable malware strains was **Ikee.A**, the first ever iPhone worm. Once it infected a device, it changed the wallpaper to an image of Rick Astley. Some 21,000 users fell victim to this malware in just two weeks.

Nearly two weeks after the Ikee.A incident, a new and more malicious iPhone malware was spotted. This new malware, named **Ikee.B** or Duh (the name of the primary binary) was designed to conduct financial fraud and included command and control logic to render all infected iPhones under the control of a bot master. Also, it stole the victim's private data and transmitted it to its command center in Lithuania. Ikee.B was programmed to send users connecting to a certain online banking site to a compromised Japanese Web page controlled by cyber-criminals. That is, Ikee.B conducted a typical pharming attack on cell phones.

In November 2009, a malicious application called **Droid09** infiltrated the Android Market. The application, posed as a useful utility for managing online bank accounts, turned out to be the work of a fraudster who used it to harvest online banking details. From the user's perspective it was not unusual that a mobile banking application requested their online banking credentials, however they did not know that this data would end in the hands of a cyber-criminal gang. This application was immediately removed from Android Market. Android.Fakeplayer was another piece of Android malware detected at the end of 2010. It pretended to be a legit video player but started spouting off SMS text messages to premium numbers. So far, three variants of Android.Fakeplayer have been unveiled.

So far, nearly all of the malware for smartphones has targeted the Symbian operating system. However, the relatively short lifecycle of Symbian devices seems to indicate that iOs (iPhone) and Android platforms will become the next number one target for cyber-crooks. The proliferation of malware targeting J2ME platforms over the last few years is also noteworthy.

6. ZEUS Man In The Mobile

Now that we have examined the top milestones of the past decade regarding malicious code for cell phones, it is time to turn our attention to today's malware situation. This section deals with a recent phenomenon that can mark a turning point in the history of malware evolution: the combination of PC malware and mobile malware into a single attack known as MITMO, incorporated into one of the most popular banking Trojans today: Zeus.

Even though Zeus has not yet targeted smartphones, it is important to note that it has been one of the first malware families to realize the potential of exploiting phones used as a second authentication factor in the online banking sector. In this case, it has used a technique to bypass SMS-based authentication in certain phone models.

Zeus is a Trojan designed to steal online banking credentials. The S21sec security company has studied the evolution of this malware threat throughout its lifecycle not only with regard to its infrastructure but also in the way it steals user data.

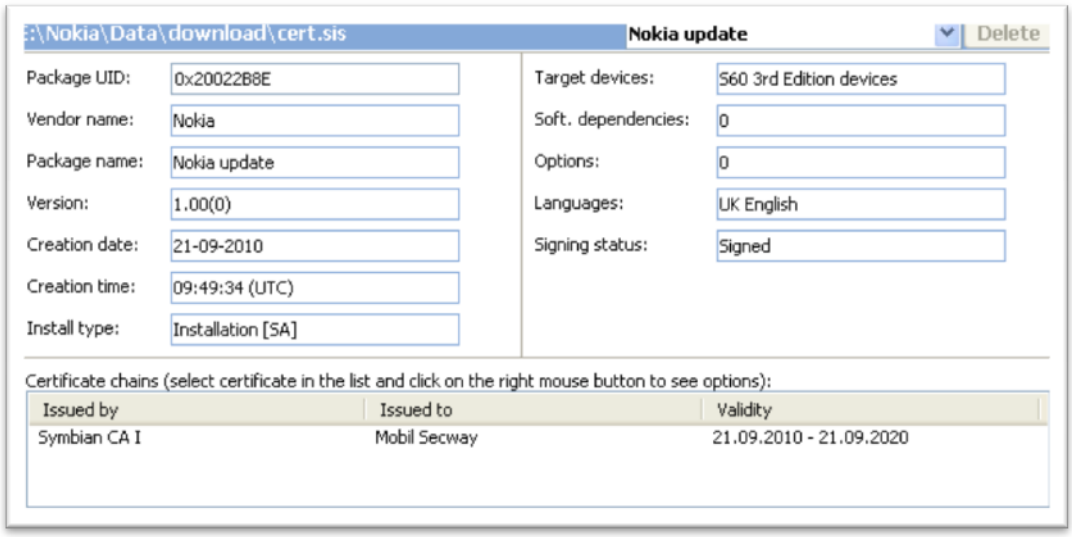
Mainly, Zeus uses three different attack methods to steal information:

- Redirecting to counterfeit sites: Users are redirected to a fake online banking site, and the account information inputted is transferred to the hacker.
- Capturing personal data input: Hackers capture user data through keylogging techniques or screenshots of the device screen (depending on the malware configuration file).
- Injecting HTML code: The malware injects HTML code into the user's browser and asks for data not usually requested by bank entities (normally the full security code). This is usually combined with the aforementioned methods.

Today, the most frequent and successful technique is the code injection attack. One of the latest developments consists of injecting code so that, once users are logged in to their online banking service as usual, they are prompted to enter their cell phone number and model.

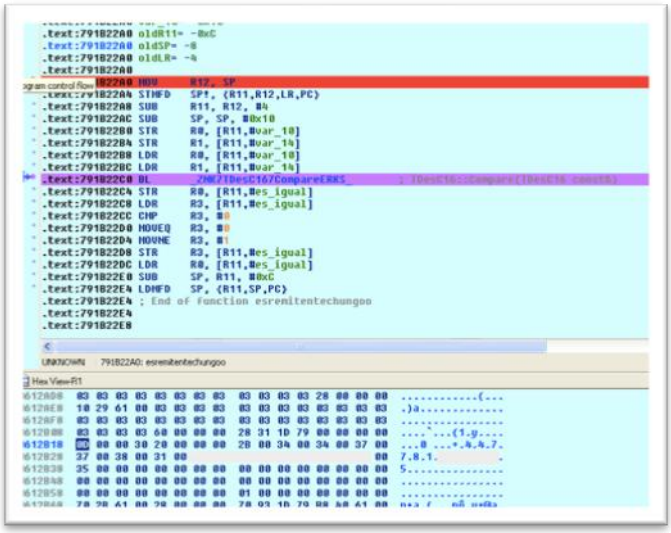
Once the user has entered this data, they receive a message to download an additional application. Researchers have detected malicious code targeting Blackberry and Symbian-based devices. In the example below, we take a look at a piece of maware targeting the latter platform.

The downloaded file is called cert.sis (MD5: b1ce81affa43bf0e51637e702d908d55) and is digitally signed by MobileSecWay (www.mobilesecway.org), an institution we don't have any more information about. The package includes a file called NokiaUpdate.exe (MD5 05c97a2f749f6a2cb92e813a48e54253) which is the actual malicious application.



Once the user installs the application on their phone, the malware triggers its malicious payload. First, it confirms execution of the application by sending an SMS message to the +447781481725 number (supposedly the hacker's number) with the text "App installed ok".

Then it monitors all inbound SMS messages. Every message coming from +447781481725 is scanned for commands to run.



The malware can receive the following commands:

- BLOCK ON: Ignore SMS requests.
- BLOCK OFF: Accept SMS requests.
- SET ADMIN: Change the phone number from which commands are received.
- ADD SENDER: Add a new contact whose messages must also be intercepted.

- ADD SENDER ALL: Intercept messages from any sender.
- REM SENDER: Delete contact.
- REM SENDER ALL: Delete all contacts.
- SET SENDER: Update contact.

Once the worm carries out the commands it has received, the original SMS is deleted from the device. Messages are intercepted before they are show on screen, so infected users are unaware of the entire process. The application includes a feature to forward SMS messages. This allows it to send out the TAN (Transaction Authentication Numbers) used as a second authentication factor in certain online banking operations. This way, the hacker bypasses the two-factor authentication system, and can use the stolen credentials to access the victim's account.

The fact that the malicious code can receive commands via SMS can lead to far more serious infections. With traditional malware there is always a central point that you can shut down to stop the fraud (a C&C server, pharming server, etc). The only times when this is not possible is when hackers use P2P as a C&C or the Trojan is a traditional backdoor Trojan that listens on a specific port. In the latter cases the attacker must continue receiving the victim's IP address in one way or another (if the IP address is dynamic) in order to be able to access the target system. These methods provide a point (email account, Web server, etc) that you can intercept to stop the attack. Also, as most networks use some kind of NAT or firewall (including 'domestic' ADSL routers), these attacks are not as effective as they used to be.

However, cell phones can receive commands via SMS, so there is no central point to shut down in order to neutralize the attack. In the case of Zeus:Mitmo, once security firms managed to neutralize the number that intercepted tokens were sent to, the attacker could have just sent a message to the infected cell phone to forward this data to a new number. Once a cell phone has been infected there is no way of preventing tokens from ending up in hackers' hands. The only way would be for the operator to inspect all messages sent across its network and block every message with expressions commonly used by Trojans (^SET ADMIN: .*$, etc.). Having said this, there are endless ways of camouflaging these commands with certain syntax that prevents pattern-based blocking.

The phone number that received the messages was in the UK and is no longer available. Scotland Yard launched an investigation and tracked a second phone involved in the attack. It seems that the attackers had been making a series of tests from the summer, sending out SMSs with the text "App installed ok". A series of identical attacks have been detected in Germany.

It is important to bear in mind that the infection triggered when the target user clicked a link to install the malicious application. This is because the user failed to perceive the threat, as this is no more than a phishing attack in a different environment.

The solution consists of raising awareness of this new type of scam, as well as changing the perception that two-factor authentication is invulnerable to hacking and fraud. Conceptually speaking, it is extremely safe, but as phone devices and computers converge, this is becoming quite different. Some researchers are looking to use other items as second authentication factors. However, it would still be possible to forward calls on a compromised phone, for example. Obviously, this would all depend on the model.

7. THE FUTURE

We have to be ready to fight future infections. Malware creators are constantly developing new techniques and it is certain that smartphone malware will continue to evolve.

- Cell phones as a new method of payment: A lot of progress is being made with the NFC (Near Field Communication) chip. NFC is a wireless communication technology which enables the exchange of data between devices over a distance of about 5-10 centimetres. NFC can be used to make payments, transfer information, etc. The benefits of NFC are fast synchronization and associated low energy consumption. It is also compatible with RFID. Nokia has already put an NFC chip in their C7 model, whereas Google devices or the future iPhone 6 will also include it.

- An increasing number of banks are now offering online banking applications for cell phones (browser-independent applications like any other application available in an app store). It is possible that the new malware for smartphones starts using more advanced techniques like syscall hooking for intercepting API calls from banking applications and capturing sensitive information.

- User tracking: Now that cell phones incorporate GPS technology it would be trivial to create a program that checked the GPS system periodically and sent the phone location to an attackers' Web server to track the user's movements.

- Now that cell phones can connect to WiFi networks, criminals could create mobile worms that scanned all devices connected to a WiFi network and exploited their vulnerabilities in order to transmit malicious code to other systems such as PCs, for example.

- Advanced social-engineering attacks. Some of the malware strains mentioned in this report can manipulate the targeted user's phone book, which is extremely useful to launch targeted social engineering attacks, change a contact's information for impersonation purposes, etc.

8. CONCLUSIONS

The lack of security awareness among cell phone users and their carelessness are the two most important risk factors for smartphones in the short term. It is extremely important to understand that a smartphone is far more than just a phone and cannot be treated like one. Unlike the previous generations of cell phones, that were at their worst susceptible to local Bluetooth hijacking, modern smartphones are today susceptible to the same risks as PCs. New attack vectors will increasingly be exploited by fraudsters as online banking services use these devices as second authentication factors given the current convergence between PCs and cell phones.

Many of the security measures that apply to PCs also apply to smartphones. These are some best practices we recommend to help you protect your cell phone:

- Enable access protection measures such as a PIN or password (if possible).
- Configure the smartphone to automatically lock after a minute or so being idle.
- Before installing or using new smartphone apps or services, check their reputation. Only install applications from trusted sources.
- Pay attention to the security permissions requested by every application and service you install.
- Keep your operating system and software applications up to date.
- Disable features not in use: Bluetooth, infrared or Wi-Fi.
- If you have Bluetooth enabled, set your device to be hidden and password-protect it.
- Make regular backup copies of your important files.
- Encrypt sensitive information whenever possible.
- Use call and SMS encryption software.
- Whenever possible, do not store sensitive information on the smartphone. Make sure it is not cached locally.
- Erase all information from the smartphone once you get rid of it.
- In the event your phone is lost or stolen, inform your service provider and give them your device's IMEI number to block it.
- You can also use remote or automatic deletion of data (after several failed login attempts).
- Monitor the smartphone for anomaly detection.
- Check your account activity frequently to detect fraud.
- Be aware of the risks associated with these devices and use them correctly.

- Take all necessary precautions when opening email messages, SMS attachments or clicking links. (Remember that this was one of the entry points of Zeus-Mitmo).
- Be wary of any files, links or numbers received from unsolicited email or SMS messages.
- Avoid using untrusted WiFi networks.
- Take smartphones into account when establishing your corporate security policy.

9. BIBLIOGRAPHY

- S21sec ecrime Report: Malware for Smartphones.
- Gartner Report
- Statistical data from www.quancast.com
- http://www.hispasec.com/laboratorio/troyano_android.pdf
- http://www.hispasec.com/laboratorio/Troyano_android_tab_snake.pdf
- http://developer.android.com/sdk/ndk/index.html
- http://csrc.nist.gov/publications/nistpubs/800-124/SP800-124.pdf
- http://www.enisa.europa.eu/act/it/oar/smartphones-information-security-risks-opportunities-and-recommendations-for-users
- http://nakedsecurity.sophos.com/2010/12/31/geinimi-android-trojan-horse-discovered/
- http://itknowledgeexchange.techtarget.com/security-bytes/google-android-trojan-surfaces-in-china/
- http://blog.mylookout.com/_media/Geinimi_Trojan_Teardown.pdf