TKK Dissertations 168
Espoo 2009

PRACTICAL PRIVACY ENHANCING TECHNOLOGIES FOR MOBILE SYSTEMS

Doctoral Dissertation

Janne Lindqvist

Helsinki University of Technology
Faculty of Information and Natural Sciences
Department of Computer Science and Engineering

TKK Dissertations 168
Espoo 2009

PRACTICAL PRIVACY ENHANCING TECHNOLOGIES FOR MOBILE SYSTEMS

Doctoral Dissertation

Janne Lindqvist

Dissertation for the degree of Doctor of Science in Technology to be presented with due permission of the Faculty of Information and Natural Sciences for public examination and debate in Auditorium T2 at Helsinki University of Technology (Espoo, Finland) on the 5th of June, 2009, at 12 noon.

Helsinki University of Technology
Faculty of Information and Natural Sciences
Department of Computer Science and Engineering

Teknillinen korkeakoulu
Informaatio- ja luonnontieteiden tiedekunta
Tietotekniikan laitos

Distribution:
Helsinki University of Technology
Faculty of Information and Natural Sciences
Department of Computer Science and Engineering
P.O. Box 5400
FI - 02015 TKK
FINLAND
URL: http://www.cse.tkk.fi/
Tel. +358-9-4511
E-mail: janne.lindqvist@iki.fi

ISBN 978-951-22-9902-7
ISBN 978-951-22-9903-4 (PDF)
ISSN 1795-2239
ISSN 1795-4584 (PDF)
URL: http://lib.tkk.fi/Diss/2009/isbn9789512299034/

TKK-DISS-2608

Picaset Oy
Helsinki 2009

ABSTRACT OF DOCTORAL DISSERTATION	HELSINKI UNIVERSITY OF TECHNOLOGY P. O. BOX 1000, FI-02015 TKK http://www.tkk.fi

Author Janne Lindqvist

Name of the dissertation
Practical Privacy Enhancing Technologies for Mobile Systems

Manuscript submitted 16.12.2008	Manuscript revised 17.05.2009
Date of the defence 05.06.2009	
☐ Monograph	☒ Article dissertation (summary + original articles)

Faculty	Information and Natural Sciences
Department	Computer Science and Engineering
Field of research	Systems security and privacy
Opponent(s)	Professor Marco Gruteser
Supervisor	Professor Antti Ylä-Jääski
Instructor(s)	Professor Tuomas Aura

Abstract

Mobile computers and handheld devices can be used today to connect to services available on the Internet. One of the predominant technologies in this respect for wireless Internet connection is the IEEE 802.11 family of WLAN standards. In many countries, WLAN access can be considered ubiquitous; there is a hotspot available almost anywhere. Unfortunately, the convenience provided by wireless Internet access has many privacy tradeoffs that are not obvious to mobile computer users. In this thesis, we investigate the lack of privacy of mobile computer users, and propose practical enhancements to increase the privacy of these users.

We show how explicit information related to the users' identity leaks on all layers of the protocol stack. Even before an IP address is configured, the mobile computer may have already leaked their affiliation and other details to the local network as the WLAN interface openly broadcasts the networks that the user has visited. Free services that require authentication or provide personalization, such as online social networks, instant messengers, or web stores, all leak the user's identity. All this information, and much more, is available to a local passive observer using a mobile computer.

In addition to a systematic analysis of privacy leaks, we have proposed four complementary privacy protection mechanisms. The main design guidelines for the mechanisms have been deployability and the introduction of minimal changes to user experience. More specifically, we mitigate privacy problems introduced by the standard WLAN access point discovery by designing a privacy-preserving access-point discovery protocol, show how a mobility management protocol can be used to protect privacy, and how leaks on all layers of the stack can be reduced by network location awareness and protocol stack virtualization. These practical technologies can be used in designing a privacy-preserving mobile system or can be retrofitted to current systems.

Keywords privacy, mobile systems, IEEE 802.11, anonymity, wireless networks

ISBN (printed) 978-951-22-9902-7	ISSN (printed) 1795-2239
ISBN (pdf) 978-951-22-9903-4	ISSN (pdf) 1795-4584
Language English	Number of pages 56 p. + app. 55 p.

Publisher Department of Computer Science and Engineering

Print distribution Department of Computer Science and Engineering

☒ The dissertation can be read at http://lib.tkk.fi/Diss/2009/isbn9789512299034/

VÄITÖSKIRJAN TIIVISTELMÄ	TEKNILLINEN KORKEAKOULU PL 1000, 02015 TKK http://www.tkk.fi

Tekijä Janne Lindqvist

Väitöskirjan nimi
Practical Privacy Enhancing Technologies for Mobile Systems

Käsikirjoituksen päivämäärä 16.12.2008	Korjatun käsikirjoituksen päivämäärä 17.05.2009
Väitöstilaisuuden ajankohta 05.06.2009	
☐ Monografia	☒ Yhdistelmäväitöskirja (yhteenveto + erillisartikkelit)

Tiedekunta	Informaatio- ja luonnontieteet
Laitos	Tietotekniikka
Tutkimusala	Tietoturvallisuus
Vastaväittäjä(t)	Professori Marco Gruteser
Työn valvoja	Professori Antti Ylä-Jääski
Työn ohjaaja(t)	Professori Tuomas Aura

Tiivistelmä

Internet-yhteyksiä otetaan yhä useammin mukana kulkevista laitteista, kuten matkapuhelimista tai kannettavista tietokoneista. IEEE 802.11 langattomien lähiverkkojen (WLAN) standardi on yksi vallitseva tekniikka langattomien Internet-yhteyksien muodostamiseen. Monissa maissa WLAN-tekniikkaa voidaan jopa pitää kaikkialla läsnä olevana, käyttäjä voi löytää paikasta riippuen useitakin tukiasemia. Valitettavasti langattoman Internetin käytön vaivattomuus tuo mukanaan monia yksityisyyden suojan uhkia, jotka eivät ole ilmiselviä käyttäjille.

Väitöskirjassa tutkimme kannettavien tietokoneiden käyttäjien yksityisyyden suojaa ja sen vähäisyyttä, ja esitämme ratkaisuja, joilla uhkia yksityisyyden suojalle voidaan vähentää. Osoitamme kuinka selkeitä käyttäjien identiteettiin liittyviä tunnisteita vuotaa lähiverkkoon jokaisesta protokollapinon kerroksesta. Ennen kuin käyttäjän laite on saanut edes IP-osoitteen, kannettava tietokone on saattanut vuotaa informaatiota käyttäjän edustamasta organisaatiosta. WLAN-yhteys taasen lähettää kaikkien näkyville tiedon missä verkoissa käyttäjä on vieraillut. Käyttäjän identiteetin vuotavat myös monet tunnistamista vaativat ilmaispalvelut. Näitä palveluita ovat esimerkiksi pikaviestimet sekä web-pohjaiset verkostoitumispalvelut, sähköpostipalvelut tai kirjakaupat. Kaikki tämä informaatio on saatavilla lähiverkkoa tarkkailevalle hyökkääjälle.

Järjestelmällisen informaatiovuotojen analysoinnin lisäksi esitimme neljä toistaan tukevaa yksityisyyttä suojaavaa tekniikkaa. Tekniikoiden suunnittelun periaatteeksi otimme helpon käyttöönoton, sekä käyttäjäkokemuksen säilyttämisen olemassa olevissa laitteissa. Erityisesti, lievitämme uhkia yksityisyydelle WLAN-tukiasemien etsinnässä, osoitamme kuinka liikkuvuudenhallintakäytäntöä voidaan hyödyntää yksityisyyden suojaamiseksi, ja kuinka vuotoja protokollapinon jokaisesta kerroksesta voidaan vähentää tunnistamalla käytössä oleva verkko sekä näennäistämällä protokollapino. Esitetyt käytännönläheiset tekniikat ovat avuksi suunnitellessa yksityisyyttä suojaavia mobiilijärjestelmiä ja ne voidaan ottaa myös käyttöön olemassaolevissa järjestelmissä.

Asiasanat yksityisyys, mobiilijärjestelmät, IEEE 802.11, anonymiteetti, langattomat verkot

ISBN (painettu) 978-951-22-9902-7	ISSN (painettu) 1795-2239
ISBN (pdf) 978-951-22-9903-4	ISSN (pdf) 1795-4584
Kieli Englanti	Sivumäärä 56 s. + liit. 55 s.

Julkaisija Tietotekniikan laitos

Painetun väitöskirjan jakelu Tietotekniikan laitos

☒ Luettavissa verkossa osoitteessa http://lib.tkk.fi/Diss/2009/isbn9789512299034/

To Blerta, lepurushja ime

Preface

"Too many secrets", Sneakers, 1992.

What can I say? I have always been interested in privacy.

During the course of my research, I was funded by the Graduate School of Computer Science and Engineering of TKK and various projects. I am also grateful for the scholarships awarded by HPY Research Foundation (Elisa), Nokia Foundation, Research and Development Foundation of Telia-Sonera, and Oy Strömberg Ab Foundation.

I am grateful for my supervisor Antti Ylä-Jääski for his generous support and giving me the time to work on this thesis independently. I thank my instructor Tuomas Aura for the many times we among other colleagues went to a meeting room at Microsoft Research and broke the protocols we had designed earlier. Some of the *secure* protocols we designed in those meetings are now part of the systems presented in this thesis. I thank also the pre-examiners Matthew Wright and Alf Zugenmaier for their thorough review and insightful suggestions.

I was fortunate to have the opportunity to visit International Computer Science Institute (ICSI), close to UCB in Berkeley, California. During my 6 month visit, I shared the office with Teemu Koponen. Teemu really introduced me into systems thinking, and I am happy he had the time to co-author one of the papers presented in this thesis, while advising me on so many other research leads, too. Nobody else than Teemu has given me so much sound advice related to research and on many other topics as well.

I am in debt to my co-authors. I learned so much from Mike Roe and George Danezis in many ways during my internships with Microsoft Research. I had the privilege of specifying Master's thesis topics for Laura Takkinen and Annu Myllyniemi and instructing them. Jussi Mäki and Juha-Matti Tapio were essential in order to realize the architectures I had designed.

I thank my colleagues in the former TML and now in CSE, especially Sanna Suoranta and Miika Komu for all their support. During his term as a professor in the former TML, Asokan gave me lots of good advice, which most of I did not listen, then. Jukka Manner's viewpoints and especially office sofa was very useful after a long night.

I cannot thank enough my family and friends. From my father I learned to follow my own path and question authorities, while my mother taught me compassion and not to trample others on my path. My *little* brother gave me always good advice on life, universe and everything, because he managed to do all the mistakes I should have done before him while I was aspiring to be a scholar.

I have dedicated this thesis to my fiancée Blerta. Even writing her name makes me smile.

Janne Lindqvist
Helsinki, May 2009

Contents

Original Publications

This thesis consists of an overview followed by four published articles:

I. Janne Lindqvist and Laura Takkinen. *Privacy Management for Secure Mobility*, in Proceedings of the 5th ACM CCS Workshop on Privacy in Electronic Society (WPES), Alexandria, Virginia, USA, October 30th, 2006.

II. Tuomas Aura, Janne Lindqvist, Michael Roe, and Anish Mohammed. *Chattering Laptops*, in Proceedings of the 8th Privacy Enhancing Technologies Symposium (PETS), Leuven, Belgium, July 23th-25th, 2008.

III. Janne Lindqvist and Juha-Matti Tapio. *Protecting Privacy with Protocol Stack Virtualization*, in Proceedings of the 7th ACM CCS Workshop on Privacy in Electronic Society (WPES), Alexandria, Virginia, USA, October 27th, 2008.

IV. Janne Lindqvist, Tuomas Aura, George Danezis, Teemu Koponen, Annu Myllyniemi, Jussi Mäki and Michael Roe. *Privacy-Preserving 802.11 Access-Point Discovery*, Microsoft Research Technical Report, Cambridge, January 2009. *An abridged version of this article is available in Proceedings of the Second ACM Conference on Wireless Network Security (WiSec'09), Zurich, Switzerland, March 16-18, 2009.*

Other Published Work

During my studies, I published the following works that are not included in this thesis:

Peer-reviewed articles

1. Miika Komu and Janne Lindqvist, *Leap-of-Faith Security is Enough for IP Mobility*, in Proceedings of the 6th Annual IEEE Consumer Communications & Networking Conference - IEEE CCNC 2009, Las Vegas, Nevada, January 2009.

2. Anu Markkola and Janne Lindqvist, *Accessible Voice CAPTCHAs for Internet Telephony* in The Symposium on Accessible Privacy and Security (SOAPS '08) part of 2008 Symposium on Usable Privacy and Security (SOUPS), July 23, 2008, Pittsburgh, PA, USA.

3. Janne Lindqvist, Essi Vehmersalo, Miika Komu and Jukka Manner, *Enterprise Network Packet Filtering for Mobile Cryptographic Identities*, two-page abstract presented in the posters session at the Usenix 2007 Annual Technical Conference, Santa Clara, CA, June 20, 2007.

4. Janne Lindqvist, *Privacy in Networked Mobile Device Configuration*, in Doctoral Colloquium at the 9th International Conference on Ubiquitous Computing (UbiComp 2007-DC), September 16, 2007, Innsbruck, Austria.

5. Anton Alstes and Janne Lindqvist, *Learning Routing and Network Programming Online*, in Proceedings of 12th Annual Conference on Innovation and Technology in Computer Science Education - ITiCSE 2007, 25-27th June 2007, Dundee, Scotland, UK.

6. Kristiina Karvonen and Janne Lindqvist, *Usability Improvements for WLAN Access*, in Proceedings of 12th International Conference on Human-Computer Interaction (HCI International 2007), 22-27.7.2007, Beijing, China, (Human-Computer Interaction, Part I, HCII 2007, LNCS 4550-0549, pp. 549-558).

7. Janne Lindqvist and Miika Komu, *Cure for Spam over Internet Telephony*, in Proceedings of the 4th IEEE Consumer Communications and Networking Conference - IEEE CCNC 2007, Las Vegas, USA, 11-13 January 2007.

8. Janne Lindqvist. *IPv6 Stateless Address Autoconfiguration Considered Harmful*, in Proceedings of the Military Communications Conference (MILCOM), Washington, D.C., USA, October 23th-25th, 2006.

9. Janne Lindqvist, Antti Ylä-Jääski and Jukka Manner, *Resilient IPv6 Multicast Address Allocation in Ad Hoc Networks*, in Proceedings of the 2006 IEEE International Workshop on Wireless Ad-hoc and Sensor Networks - IWWAN 2006, New York, 28-30 June 2006.

10. Janne Lindqvist and Sanna Liimatainen, *VERKKOKE: Online Teaching Environment for Telecommunications Software and Routing*, One page abstract in Proceedings of The Eleventh Annual Conference on Innovation and Technology in Computer Science Education - ITiCSE'06, University of Bologna, Italy, 26-28 June 2006.

11. Janne Lindqvist, Sanna Liimatainen and Tuomo Katajamäki, *Secure Pairing Architecture for Wireless Mobile Devices*, in Proceedings of IEEE 63rd Vehicular Technology Conference - IEEE VTC 2006 Spring, 7 - 10 May 2006, Melbourne, Australia.

12. Janne Lindqvist and Sasu Tarkoma, *Protecting Internet Connectivity of Hybrid Ad Hoc Network Gateways*, in MiNEMA Workshop, February 7-8 2006, Leuven, Belgium.

13. Janne Lindqvist, *Micro Mobility Routing and Multicasting in Hybrid Ad Hoc Networks*, in Proceedings of IEE Mobility Conference 2005, Guangzhou, China, 15.11-17.11.2005.

14. Mikko Martsola, Timo Kiravuo and Janne Lindqvist, *Machine to Machine Communication in Cellular Networks*, in Proceedings of IEE Mobility Conference 2005, Guangzhou, China, 15.11-17.11.2005.

15. Janne Lindqvist, *Suspend Mode Mobility Management Problem*, in Proceedings of the 6th IEE International Conference on 3G and Beyond - IEE 3G 2005, London, 7.11. - 9.11.2005, p. 245-248.

Book Chapters

1. Janne Lindqvist, *Yksityisyyden suoja verkotetussa yhteiskunnassa*, in Silmät auki! Tietoyhteiskunnan uhat ja mahdollisuudet. Toim. Ville Eloranta. Eduskunnan tulevaisuusvaliokunta. ISBN 978-951-53-3043-7 (nid.) ISBN 978-951-53-3044-4 (PDF)

Magazine Articles

1. Janne Lindqvist, Pravin Pawar and Erich Stuntebeck *HotMobile 2008: Postconference Report*, in IEEE Pervasive Computing, October-December 2008, Volume 7, Issue 4.

Technical Reports

1. Petri Kärhä, Jaakko Hollmen, Satu Karling, Jaakko Kujala, Petri Kuosmanen and Janne Lindqvist, *Opinnäytetyön ohjaus Teknillisessä korkeakoulussa*, Helsinki University of Technology Metrology Research Institute Report 28/2005.

2. Teemu Koponen, Janne Lindqvist, Niklas Karlsson, Essi Vehmersalo, Miika Komu, Mika Kousa, Dmitri Korzun and Andrei Gurtov, *Overview and Comparison Criteria for the Host Identity Protocol and Related Technologies*, Technical Report, February 2005.

Internet-Drafts

1. Janne Lindqvist, *Establishing Host Identity Protocol Opportunistic Mode with TCP Option (draft-lindqvist-hip-opportunistic-01.txt)*, March 6, 2006, work in progress, Expired: September 7, 2006.

2. Janne Lindqvist, *Piggybacking TCP to Host Identity Protocol (draft-lindqvist-hip-tcp-piggybacking-00.txt)*, July 11, 2006, work in progress, Expired: January 12, 2007.

Chapter 1

Introduction

> "I always feel like somebody's watching me... and I have no privacy" – Rockwell, *Somebody's Watching Me*, Motown, 1984.

Mobile computing is pervasive today. There is a multitude of mobile devices available for users: to help with their work when they travel and to entertain them everywhere, etc. The mobile computers range from full-fledged laptops with the processing power of tabletops, to mobile phones (or smartphones), and even to mp3 players. A press release [101] of a survey reported that 53 % of US-based adults using computers, use either a notebook or a tablet PC, thus, making mobile PC users now a majority in the US.

Unfortunately, the privacy tradeoffs that these devices introduce are also pervasive. These devices have many helpful features that improve the usability of network access. For example, they automatically search for available printer services and access points. However, many features that users have learned to expect have implications for the privacy of these users.

In this thesis, we focus on the privacy of mobile computer users. We present a systematic analysis of information leaks from mobile computer systems, and show how the privacy of the user is violated by leaks from all layers of the protocol stack. We present practical privacy enhancing technologies to mitigate these risks to privacy. First of all, however, we should consider briefly how we understand privacy.

The focus of privacy in this thesis is from the viewpoint of *information flow control* and *the right to be left alone*. Two famous quotes will serve as a starting point here:

> "[Privacy is] the claim of individuals, groups, or institutions to determine themselves when, how, and to what extent information about them is communicated to others" [139]

And when speaking of humankind in general

> "modern enterprise and invention have, through invasions upon his privacy, subjected him to mental pain and distress, far greater than could be inflicted by mere bodily injury" [137]

To put the quotes in their right context, the authors of these quotes are discussing legal mechanisms needed to protect privacy. The second quote explains what can happen when the right to be left alone is violated. To further clarify what we mean by the concepts, we give an example of the life of the author during the writing of this thesis.

"The author uses two major credit cards, which he usually keeps in his wallet in the left pocket of his pants."

Now, the above sentence is an exercise in information flow control where the author voluntarily divulges personal information. Now that this information is public, the author can no longer control who uses this information and for what purposes. Neverthless the author did not divulge the identification numbers, the expiration dates or the security codes of the credit cards. Hence, information flow control was exercised.

Credit card details offer a very convenient way to, for example, shop for books on the Internet. The online bookstore needs only some more personal information, such as, the name of the owner of the credit card and the billing and shipping addresses. Giving these details to the online store introduces privacy risks, and is therefore a tradeoff between convenience and privacy. First of all, if we limit ourselves to what is exposed legally, the credit card company gets to know that the user is buying books online. The online bookstore also receives explicit information who is buying and what, and can use this information for direct marketing or recommendations. This might already violate the right to be left alone, since some users might not like the exposure to directed marketing based on their buying history.

Having said this it should be pointed out that the above discussion of privacy is extremely narrow, but sufficient for the purposes of this thesis. For a more detailed coverage of privacy violations in the context of US legalization and court cases, we refer the reader to Daniel J. Solove's "Taxonomy of Privacy" [127]. The concepts already presented, information flow control and the right to be left alone, are all important for understanding why this research was done. The concepts to be introduced next, however, are essential for understanding how privacy can be protected using privacy enhancing technologies.

Confidentiality of Communication Content The concept of *confidentiality of communication content* is rather self-explanatory; the problem is to protect communication data, for example, the voice in a phone conversation or the body of an email message. Confidentiality of communication content and the right to privacy are commonly associated concepts, and in the context of computer communication are often considered to merely mean data encryption. Good examples of this line of thinking are the Wired Equivalent Privacy (WEP) protocol [73], and a classic privacy book "Privacy on the Line: The Politics of Wiretapping and Encryption" [41]. WEP is the name of the first WLAN authentication and encryption protocol, while the book mostly discusses the importance of using encryption technologies to protect privacy online. Next, we introduce an equally important concept related to confidentiality of communication.

Confidentiality of Communication Participants Confidentiality of communication participants means concealing the identities of the involved parties. We refer to this concept also as *identity privacy*. In practice, this kind of confidentiality is achieved with *pseudonymity* or *anonymity*. In short, pseudonymity means that, for example, a person has an identifier that cannot be directly attributed to that person. Whereas anonymity means that the person cannot be identified within an anonymity set. For

example, consider Alcoholics Anonymous (AA); it is common knowledge[1] that in AA meetings people introduce themselves by stating, for example, ”I’m James, and I’m an alcoholic.” James may or may not be a real name, but it is in fact a *pseudonym* for the person, and thus, the person can be identified in the context of AA meetings, at least. Thus, in exact terms, the person is not anonymous. For a more complete treatment of related terminology, we refer the reader to “Anonymity, Unlinkability, Undetectability, Unobservability, Pseudonymity, and Identity Management – A Consolidated Proposal for Terminology” by Pfitzmann and Hansen [114].

The reason why both confidentiality of communication content and communication participants are equally important in privacy protection is neatly summarized in the following quote from a US Supreme Court Justice:

> “The evil incident to invasion of privacy of the telephone is far greater than that involved in tampering with the mails. Whenever a telephone line is tapped, the privacy of persons at both ends of the line is invaded, and all conversations between them upon any subject, and although proper, confidential and privileged, may be overheard. Moreover, the tapping of one man’s telephone line involves the tapping of the telephone of every other person whom he may call or who may call him. As a means of espionage, writs of assistance and general warrants are but puny instruments of tyranny and oppression when compared to wire-tapping.” – Justice Louis Brandeis, dissenting opinion in *Olmstead vs. United States* (277 US 438, 1928, pp. 475–476), cited from a secondary source [41].

Confidentiality of Location The last privacy concept to be introduced is about protecting or hiding the location of the communicating party. Providing location privacy does not necessarily require protection of the confidentiality of communication content or the participants. This means that we may know in detail who is talking to whom and the content of their communication, but not their physical or actual network location. However, many anonymity systems provide also location privacy while providing sender and receiver anonymity. Location privacy can be provided with different granularities, for example, locally, on the level of access point usage, on the level of network access (network prefix), or by the country from which the user currently is contacting the network. Naturally, finer or coarser-grained granularities can also be defined.

As a concrete example of location privacy protection let us consider GSM networks. Roughly speaking, the GSM number consists of the country code, operator code and finally the number of the particular subscription. Thus, an outside observer by looking at the subscriber’s number can identify where the subscription was bought, but not much more. The GSM network operator, however, knows all the time where the user resides because of the access points near the mobile phone. GSM networks fail to provide complete location privacy against outside observers, too. The dial tone is different in many countries, which already gives a clue where the user might reside. Furthermore,

[1] as depicted in popular tv series and movies

when the mobile phone is switched off and somebody tries to call the user, the caller hears recorded message from the operator along the lines of "the number you have tried to reach.." This message is usually given in different languages depending on the country, and thus, gives a good hint about the country the user resides in at the moment. Maybe somebody wants to visit a person's house when the person is not present. Maybe the other person wants to be reachable from the same number, but not want to reveal the fact that he or she has left the country.

1.1 Problem Statement

The research problem was two-fold:

1. What are the privacy tradeoffs that modern mobile computer usage introduces?
2. How can we mitigate or prevent the adverse privacy effects of using mobile computers?

One of the chosen key limitations or requirements was to design mechanisms that do not alter the *user experience* or otherwise alter it minimally, in order to provide acceptability for the chosen approach. Another consideration in respect of that is *deployability*. In addition to the above requirement for ease of deployment, other considerations were needed. We wanted to minimize the changes to existing standards or protocols and minimize the involvement of other hosts or infrastructure. In other words, the user does not need to trust the network or any infrastructure other than the operating system and the user's device. In particular, we investigated how privacy can be improved by only modifying either a single host, or two hosts, or a host and an access point.

The method for the research was constructive. Given a research problem, we investigated possible solutions to solve a particular problem while keeping in mind the key limitations and requirements. When we conceived a solution candidate, we started implementing it in software. During the implementation, we found limitations or different views on the problem, and refined the original solution or redesigned it from scratch. In the latter case, we then started to implement the software again. Finally, the research prototypes were tested in wireless research platforms.

1.2 Contributions

This thesis belongs to the field of *systems security and privacy*, and to the subfield of *privacy enhancing technologies*, and this section succinctly summarizes the contributions of the publications included in this thesis. In Chapter 3, we discuss the contributions and their limitations further, and contrast them to the most significant and recent related work.

- Publication I shows how a mobility management protocol integrated with IPsec can be used to provide privacy for the mobile hosts. The approach is generalizable, and can therefore be integrated with other protocols, such as SSH or TLS.

- Publication II presents a systematic analysis of privacy leaks from mobile computers. We propose a system based on network location awareness to prevent these leaks.
- Publication III presents how to mitigate privacy leaks by traffic isolation in a networked mobile computer system.
- Publication IV presents how to modify access-point discovery to preserve privacy in IEEE 802.11 with minimal changes to the standard protocol. We show that the modifications do not affect the user experience or security of IEEE 802.11.

1.3 Author's Contributions

- Publication I. The author was the lead architect of the project, and the original idea for the problem setting was his. The second author implemented the system under the author's close guidance.
- Publication II. The author of this thesis shared similar ideas for a project with the first author of Publication II. The author implemented parts of the system presented and did all the experiments for the results presented in the paper.
- Publication III. The author was the lead architect of the project, and the original idea for the problem setting was his. Based on the author's initial draft of the publication and rudimentary scripts, the second author of Publication III implemented the system under the author's close guidance.
- Publication IV. The author of this thesis was the lead architect of the project, and the original idea for the problem setting was his.

The only exception being Publication II, the author of this thesis was the only person responsible for formulating the research problems, and for the outcome of the projects. However, it is obvious that the author of this thesis owes a great debt to the group of excellent seniors and students who are joint authors of the publications. The seniors were crucial for staying on the right track in the work, whereas the Master's students did most of the concrete implementation work with the help and guidance of the author.

1.4 Structure of this Thesis

The rest of the thesis is organized as follows. Background on relevant related work on privacy enhancing technologies is given in Chapter 2. The contributions of the thesis are discussed in Chapter 3, and the conclusions are in Chapter 4. The published contributions of this thesis are presented in the four individual chapters: "Privacy Management for Secure Mobility" in Publication I, "Chattering Laptops" in Publication II, "Protecting Privacy with Protocol Stack Virtualization" in Publication III and "Privacy-Preserving 802.11 Access-Point Discovery" in Publication IV.

Chapter 2

Background

This chapter presents the background on privacy enhancing technologies. We have limited ourselves to work that is essential for understanding the field of the thesis: how it came about and where it is today. More specific comparison with other related domains is given in the respective related work sections of the published articles. We also discuss some related work that was omitted in the publications because of space limitations, and new related work published after the author's publications.

The chapter is organized as follows. We start with the basics of cryptography and move from them to the privacy-preserving and anonymity systems that use them. We give the background on cryptographic authentication and key exchange protocols, their identity privacy issues, and privacy protection mechanisms in Section 2.1. What follows after that in Section 2.2 is a discussion on anonymity systems and privacy on the Internet: infrastructure-based solutions for providing privacy and research on their limitations, and how real-world privacy-preserving systems work. Finally, we discuss privacy in wireless networks and mobile systems in Section 2.3, which provides the fundamental background for the contributions presented in this thesis.

Many privacy enhancing technologies, however, such as anonymous payment systems [27, 30], privacy-preserving data-mining [136], private information retrieval [110] are considered to be beyond the scope of this thesis.

2.1 Cryptographic Protocols and Privacy Protection

Protecting the confidentiality of content can be done with cryptographic protocols. In this section, therefore, we review some basic cryptographic building blocks for protocols, introduce elementary authentication and key establishment protocols, and then proceed with practical protocols used on the Internet. The emphasis in this section is on protocols and mechanisms that provide identity privacy with authentication and key establishment. These protocols are mostly related to the contributions presented in Publications I and IV, and provide background on identity leaks in the cryptographic protocols presented in Publication II.

2.1.1 Cryptographic Building Blocks and Elementary Protocols

Secret Key Cryptography Cryptographic algorithms that use a shared key belong to the family of secret key or symmetric key cryptography. The crucial point in symmetric key cryptography is that the secrecy of the protected data should be dependent on the key. If the attacker knows the algorithm and receives plaintext and encrypted data, the encrypted data still remains protected provided the key is kept secret. This also highlights

one of the fundamental problems of symmetric key cryptography called key management: how to share the secret? Today, symmetric key cryptography is an important building block for many systems that may also use public key cryptography, as is explained later.

The simplest and most secure form of a symmetric key encryption is the *one time pad*. With a one time pad, the key must be used only once to encrypt the plaintext. The key also needs to be at least the length of the plaintext. This kind of construct has been proven to have perfect secrecy, but is unfortunately highly impractical for most applications. More practical symmetric key algorithms are divided into *block ciphers* and *stream ciphers*.

Today, the most important block cipher is the Advanced Encryption Standard (AES) [109]. It can use variable key lengths of 128, 192 and 256 bits, and takes input and produces output in blocks of 128 bits. As of today, there are no known practical attacks or even theoretically feasible attacks against AES.

Another important building block is the cryptographic hash function. A cryptographic hash function can take an arbitrary length of input and produce a fixed length digest or hash as a result. The important part is that the function must be one-way; an attacker should not be able to deduce the input from the result. An important building block for practical protocols that use both symmetric key cryptography and cryptographic hash-functions is the *keyed-hash based message authentication code* (HMAC) [94]. The purpose of the HMAC is to provide integrity for a given message, and it can only be verified if the authenticator has the shared secret key.

Public Key Cryptography We mentioned above that symmetric key cryptography has a fundamental problem related to key management: the distribution of keys. To address this problem, Merkle [100] invented the public key distribution system, today also known as asymmetric key cryptography. Due to the review process of *Communications of the ACM*, Merkle's earlier submission was published later than Whitfield Diffie's and Mark Hellman's article titled "New directions in cryptography" [40]. Although later the invention of public key distribution systems has been attributed to Merkle. Despite this, the world now knows the process of exchanging keying material as the Diffie-Hellman key exchange. In the abstract form, the idea is simple: each peer owns a public and private key pair that have a mathematical relation. The private key is kept secret and the public key is published. The public key is used to encrypt data that can only be decrypted with the corresponding private key.

The first practical proposal for public key cryptography was the Rivest, Shamir and Adleman algorithm, today known simply as RSA [121]. The algorithm uses modular arithmetic and relies on the assumption that factoring prime numbers is a hard problem. The private key is a combination of positive integers (d, n) and the public key is a combination of positive integers (e, n), where n is the product of two primes p and q, and d is chosen relative to p and q. Finally, e is computed from p, q and d to be the "multiplicative inverse" of d. The security of RSA and public key encryption is computational: the public key is always mathematically related to the private key. The RSA algorithm also presents a method for *digital signatures*. In contrast to using the

public key for encrypting the data, a user uses the private key. The resulted encryption is the digital signature, which can be verified using the public key.

We have now briefly introduced some of the most important building blocks of modern cryptographic protocols. In practice, real-world protocols are usually hybrid cryptosystems: they use both symmetric and asymmetric cryptography. We give examples of these hybrid cryptosystems and their relation to privacy protection below.

Elementary Protocols In this section, we discuss two important elementary protocols related to the topic of this thesis. The protocol specifications are abstract. To implement these protocols in real systems, many details that are not discussed in the specifications, need to be considered. We discuss the two-pass entity authentication protocol specified in the ISO/IEC 9798-4 standard [77] and the STS [42] protocol for authentication and authenticated key exchanges. These protocols are used between two hypothetical users, Alice and Bob, as follows.

The ISO two-pass protocol uses only a cryptographic hash function and a pseudorandom number called *nonce*. The pseudorandom number needs to be chosen carefully, and is never to be used again in order to maintain the liveness of the protocol, that is, prevent *replay attacks*. We now assume that Alice wants to prove her identity, that is, authenticate to Bob. A successful run of the protocol goes as follows:

1. Bob generates the nonce R_{Bob} and sends it to Alice.

2. Alice generates a keyed cryptographic hash of [R_{Bob} and concatenation of the name Bob] and sends it to Bob.

3. When Bob receives Alice's message, he calculates the function and if it matches with what Alice has sent, Alice has successfully proven that she knows the shared key and has thus authenticated herself to Bob.

The protocol can be extended for mutual authentication by adding a third pass. We also note that, by executing the protocol, Bob knows that the authenticated user is Alice if, and only if, the key is not shared with other users. Thus, the protocol can be used as a group authentication protocol by sharing the key with multiple users, but then it does not identify individual users.

A public key based authentication and key exchange protocol called STS [42] was published in 1992. The publication mentions many desirable characteristics for a protocol, such as perfect forward secrecy, direct authentication and no timestamps. STS achieves all these properties with the following construction:

1. Alice generates a random number x and sends the exponential α^x to Bob.

2. Bob generates a random number y and uses it with Alice's exponential to compute the key $K = \alpha^{xy}$

3. Bob computes α^y, signs both α^x and α^y and encrypts the signature using the key K, and sends all these to Alice.

4. Alice computes K, decrypts Bob's ciphertext and verifies Bob's signature using Bob's public key.

5. Alice sends the corresponding ciphertext encrypted with the key K to Bob. The ciphertext includes the exponentials signed with her key.

6. Bob decrypts the ciphertext Alice sent and verifies Alice's signature.

Alice and Bob have now mutually authenticated themselves and share a secret K, which can be used to encrypt further communication between the peers. For the above protocol to be secure, Alice needs to know Bob's public key beforehand and vice versa. The public key needs not be known, if the protocol is modified to use public key certificates, which consequently assumes a trusted third party as a certificate authority. The STS protocol can be used to build hybrid cryptosystems as discussed above, and many modern implemented cryptographic protocols have design choices similar to STS, but add more features, for example, denial of service resistance.

2.1.2 Cryptographic Protocols and Identity Protection

It is prudent engineering practice [3] for cryptographic protocols that a name for the authenticating principal is given during the process of authentication:

> "If the identity of a principal is essential to the meaning of a message, it is prudent to mention the principal's name explicitly in the message." [3]

This practice is heeded in many protocols, but unfortunately, by sending the names of the principals and usually the certificates in plaintext, the protocols reveal the identity of the principals to even passive observers. This section presents protocols that are designed to protect the identity of the authentication participants, in addition to providing confidentiality of content.

The Security Architecture for Internet Protocol (IPsec) [86] working group in the IETF has inspired many authentication and key exchange protocols that provide identity protection. The earliest is the SKEME [93] protocol. SKEME provides identity protection against observers not participating in the protocol by encrypting the identity of the initiator with the responders public key. Thus, the initiator needs to reveal its identity to potential responders first. The IETF designed the IKE protocol [64], which provides identity protection in the main mode similar to SKEME, but unfortunately uses many roundtrips. The designers of Just Fast Keying (JFK) [6, 7] designed two protocols, JFKi and JFKr, that take two round-trips and provide the same level of denial of service protection. JFKi provides identity protection for the initiator against active attacks, while the JFKr provides identity protection for the responder and protects both parties' identities against passive observers. Today IKE specification has been made obsolete by IKEv2 [83], but still the initiator has to prove its identity first. Many protocols, however, do not provide identity protection at all.

As a side note, the IPsec architecture has always been surrounded with controversy. Encrypting the packet payload at the IP layer was seen as too drastic, and therefore the

IETF needed to specify the IP Authentication Header (AH) [84], which does not provide confidentiality for the content, but only integrity. This could have also been established using the null encryption mode with Encapsulating Security Payload (ESP) [85], but the IETF opted to specify a header for the sole purpose of authentication. On the other hand, AH does protect also the headers of the IP packet.

Independent of IPsec related work, Abadi proposed two protocols for private authentication that protect the identities of both parties, and later formally proved the privacy properties [1, 2]. One of the important notes Abadi makes is that the used public key encryption protocol used needs to be "which-key concealing" or "key-private" [18]. This property guarantees that, if the attacker sees the ciphertext, he or she cannot tell which specific key, out of a set of known public keys, was used to create the specific ciphertext.

There are also special protocols designed to limit the exposure of identity in specific contexts. Secret handshake protocols [9, 17] provide a way to authenticate membership in a group and role in it, while a third-party observing the exchange does not learn anything new (including whether the users doing the handshake belong to the same group, the identities of the groups or the roles of the users). However, these protocols have been asymmetric so far, in the sense that it is possible for a user belonging to a group to learn whether another user belongs to the same group and not reveal anything of herself by aborting the protocol at the appropriate time. Other schemes of interest are anonymous credentials [25, 28] for authorizing pseudonyms to access a system, short group signatures [21] for hiding who in a group signed a message, and secret sets [102] for providing sets where anybody can test for their own membership, but only the creator of the set can test another party's membership in the group she created.

2.2 Anonymity Systems and Privacy on the Internet

In this section, we review influential literature on anonymity systems and privacy on the Internet, and after introducing the fundamental concepts, we will focus on real-world deployed systems. Some of the cryptographic building blocks presented earlier in Section 2.1 are used to build anonymity systems and are also used to protect privacy on the Internet. We briefly discuss the limitations of anonymity systems and end the discussion with how privacy protection mechanisms fail in practice.

Anonymity Systems Privacy enhancing technology research can be said to have started with Chaum's seminal paper on network mixes [26]. The article published in *Communications of the ACM* has been inspiring researchers for over two decades. Chaum's work falls in the realm of infrastructure-based approaches for privacy, or so called anonymity systems. Anonymity systems can generally be categorized as *low-latency* or *high-latency*. The systems can be further categorized by their type, for example, as *mixes* [26], *onion routing* [56, 133], *dc-nets* [29] or by the infrastructure support they require: a single server or remailer, multiple servers or *peer-to-peer* systems.

We start our review of anonymity systems with the simplest form of infrastructure support: a single server in the hands of a trusted third party. The single server acts, for

example, as an email remailer. Users send emails to the server including information on the final recipient, and the remailer assigns pseudonyms for both parties to allow subsequent communication. To enhance the system, the traffic to the server can be protected using encryption, such as PGP. However, in addition to the problem that the server needs to be trusted, the architecture is brittle, as the story of the famous Finnish anonymous remailer *anon.penet.fi* demonstrates [68]. In 1995 and 1996, the Church of Scientology wanted to find out who were behind some pseudonyms relayed by the anon.penet.fi server. These issues finally led to the shutdown of the system, since it could not guarantee the privacy of their user because of explicit real address and pseudonym mappings.

A more advanced design using multiple servers as a cascade was proposed by Chaum already in 1981 [26] as noted above. The mix network provides sender and receiver anonymity using public key cryptography, where the users have digital pseudonyms denoted by their public key. The mix(es) provide anonymity by introducing bitwise unlinkability: the input to the mix cannot be correlated to the output of the mix. Chaum also proposed a way to arrange the mixes into a network and the use of dummy traffic to further complicate traffic analysis. As the mix network relies on public key cryptography, it is computationally secure. In 1988, Chaum proposed the dining cryptographers network (dc-net) that, in contrast to mixes, provides information theoretic security, but is unfortunately not very practical (and never deployed in any form) due to the amount of messages it needs in every round [29]. Furthermore, despite its information theoretic security, dc-net is also vulnerable to many traffic analysis attacks, as discussed later.

Perhaps the most important anonymity system that has been developed from Chaum's mix nets is *onion routing* [55, 56, 132, 133]. Whereas the original mix design can be understood as a packet anonymizer, the onion routing system provides the equivalent of circuit switching for mixes. The anonymity of onion routing is established by distributed trust: the clients contact the onion router network that is build as an overlay on the Internet, and hide their path by layers of encryption. As a result the onion router was a success and in 1999 the network served more than one million Web connections per month in twenty countries [55]. The onion routing project has been developed further, and the current de facto way of achieving sender and receiver anonymity on the Internet is the second-generation onion router succinctly known as Tor [44]. The Tor network is currently estimated to have several hundred thousand users and over a thousand active volunteer servers relaying the traffic [45]. Although the low-latency network is primarily used for Web browsing, it can be used for accessing a number of different connection-oriented services, such as IRC or SSH. One interesting feature that onion routing (and Tor) provide is the hidden server [133]. A user can set up a server (SSH, web, etc.) that has a name as used in the DNS system to resolve IP addresses. The difference is that this hidden server name is resolvable only in the onion routing system, and the IP address is thus hidden from its users. A hidden server establishes an anonymous connection to a rendezvous point in the onion routing network, and a client that wishes to contact the server establishes another anonymous connection via the rendezvous point.

Conceptually, this rendezvous mechanism could be seen as the anonymous equivalent of a Mobile IP home agent or a SIP relay server.

There are also many other designs for providing anonymity on the Internet. Among the most succesful ones were the Crowds [118, 119], designed for solely providing privacy for Web browsing, and the Freenet peer-to-peer network [31, 32] for anonymous file storage.

Limits of Anonymity Systems and Privacy The privacy that anonymity systems can provide is limited at least by the anonymity set [131]. In other words, anonymity is about hiding in the crowds. Therefore, usability is also important for an anonymity system. Unless the system can attract users there will be no crowds to hide in [43]. The anonymity a system provides can also be measured with entropy [39, 125]. There are a number of attacks, countermeasures and their analysis [15, 97, 105, 111, 117] documented in the literature on anonymity systems. However, in the end, it seems that long-term communication cannot be anonymous. This is shown by the results of passive logging attacks called *predecessor*, *intersection* and *disclosure*.

The predecessor attack was first introduced against Crowds [118], and later a related attack against onion routing was analyzed [132]. However, Wright et al. generalized the attack to work against all known anonymity protocols (some introduced above) and showed upper bounds for how long these protocols can maintain anonymity [144]. Later, Wright et al. improved their analysis by removing some of their previous simplifying assumptions and in simulations, they also showed that, in practice, the anonymity of real systems is worse than the theoretical upper bounds [145–147]. The predecessor attack can be established when the attacker controls some nodes in the anonymous system. The passive attacker observes path reformations where identifiable streams recur from possible initiators, and uses this information to deduce the identity of the users.

Another attack that sets limits for the anonymity provided by anonymity systems is the disclosure attack. The attacker sits on the edges of the anonymity system viewing it as a black box and tries to correlate traffic in order to identify all the peers of a particular user. The attack was first proposed by Kesdogan et al. [87] and further refined by Agrawal et al. [4, 5]. The disclosure is a NP-complete problem and therefore computationally exhausting, which led to a proposal for a statistical disclosure attack [35]. The statistical disclosure attack is further refined by Danezis and Serjantov [36], as well as by Mathewson and Dingledine [99]. The results of the analysis of the predecessor and disclosure attacks showed that long-term communications cannot be protected by known anonymity systems.

Finally, we note that in practice, anonymity and privacy-preserving systems can be compromised in many ways. Thus, even though the users would be using the anonymity systems presented above to protect their identity (IP address), many other attacks are possible. Authorship analysis [108, 116] can be used to find the author from writings done under a pseudonym. Cookies [95] can also be used to subvert anonymous systems as well as bringing with them privacy issues of their own. A server can plant a cookie to track the user, and therefore an anonymity system needs also to filter the Web traffic.

2.3 Privacy in Wireless Networks and Mobile Systems

The previous sections reviewed fundamental building blocks for the confidentiality of communication content and anonymity systems on the Internet. In this section, however, we give the background on privacy enhancing technologies in wireless networks and mobile systems. Privacy protection in mobile and wireless systems can be investigated from the viewpoint of confidentiality of communication content, confidentiality of communication participants and location privacy as discussed already in the Introduction chapter. Below, we emphasize WLAN and related technologies because of the focus of this thesis.

2.3.1 Confidentiality of Communication Content

In this section, we review the link-layer security development for wireless local area networks and other common mechanisms for protecting WLANs. This is the most common challenge to privacy in wireless networks. We discuss another common problem: the privacy of location later in Section 2.3.2. To start with, however, we focus on the problems with the first hop: the wireless medium between a client and an access point. End-to-end security is also an important problem for wireless networks although it is orthogonal to the solutions presented below. End-to-end encryption is usually provided by the solutions presented in Section 2.1.2.

Link-layer security The earliest work known to the author on security of wireless local area networks dates back to 1994. Aziz and Diffie propose a link-layer protocol for "privacy and authentication" [14] that uses public key certificates for authentication and also shared key primitives in subsequent encrypted communication. The authors also provide proofs of the protocol based on BAN logic for authentication [24].

In 1997, the first version of IEEE 802.11 WLAN was standardized [73]. The physical and MAC layer specifications also contained a security protocol called Wired Equivalence Privacy (WEP), which later became infamous for its lack of security. WEP, however, provided a convenient way to configure the network security with the "Shared Key". The other option was to use "Open System" meaning no encryption on the link-layer. WEP later was found to be flawed [23, 48, 130], and finally in 2006 the numerous patches (e.g. re-keying) for it were also proven to be completely insufficient [20].

The flaws of WEP did not go unnoticed by the wireless industry and the Wi-Fi Alliance [140] specified and implemented Wi-Fi Protected Access (WPA). WPA was designed to be compatible with deployed hardware that supported WEP while the IEEE standardized the replacement of WEP with IEEE 802.11i [74]. WPA could be introduced to WLAN devices with firmware upgrades, while the IEEE 802.11i was being standardized. An analysis of 802.11i [65, 66] found flaws that made the protocol susceptible to denial of service attacks, and a corrected version was incorporated into the final published protocol. Later, a formal correctness proof of 802.11i with Protocol Composition Logic (PCL) was presented [67]. The 2007 [76] version of 802.11 incorporated 802.11i [74], which was published separately in 2004.

One of the features that was introduced in the protocols succeeding WEP was the

possibility to integrate the authentication with AAA systems and use the 802.1X port-based access control [75]. These features enabled so-called *managed networks* that can use considerable key lengths because users do not need to configure the WPA or 802.11i keys manually. These enterprise versions of the security protocols use Extensible Authentication Protocol (EAP) [128] implementations, such as EAP-TLS [126].

Despite the efforts to secure WLAN, the usability of WLAN security was questioned [50]. Users could not enable the link-layer encryption for their access points or were using short keys that could be broken easily with dictionary-based attacks. Even though WPA and 802.11i use PBKDF2 [82] to enhance the security of passwords, very simple passwords could still be cracked. (The usefulness of PBKDF2 implementation was also questioned [148].) To address the usability problems of WLAN security setup, the Wi-Fi alliance published Wi-Fi protected setup (WPS) [141] specification. Unfortunately, the usability of WPS specifications was found to be poor [96], and even the certified implementations have therefore varied in functionality. It is worth noting that, independent of the Wi-Fi alliance, setting up WLAN security using location limited channels was proposed in "Network-in-a-Box" [16]. Because enabling link-layer security was problematic for users, or the hardware was insufficient to support enhanced protocols, or some networks do not deploy link-layer security, a number of other security practices have emerged. We describe these practices next.

Other WLAN protection mechanisms We now describe security mechanisms that are not specified by the IEEE 802.11 standards, but are deployed widely.

One widely recommended [53, 120, 129, 142] security measure is to disable SSID broadcasting in WLAN. Disabling the broadcasting means that instead of a user-readable name, the WLAN beacons contain an empty string. Thus, the access point's presence can still be spotted from the beacons. To gain access to the network, the client must actively send probe requests for the hidden name, and when the access point is present, it will reply with the probe response. This mechanism introduces a problem for client privacy, first recognized by Greenstein et al. [57], and is further discussed below.

Another common mechanism for securing an access point is MAC address filtering. The access point is configured with an access control list, and only clients with preconfigured MAC addresses are allowed to connect. The non-broadcast SSID and MAC address filtering mechanisms only offer protection from possible casual network intruders. Determined attackers merely need to be present and passively observing when the hidden network with MAC address filtering is accessed, and they can then easily configure appropriate parameters for their attacking clients.

The final non-standard mechanism, which is widely deployed, is HTTP authentication or web browser based authentication, also known as Universal Access Method (UAM). To gain access to the WLAN network, the users launch their web browser. The web browser is redirected by the DNS server to an authentication page, where the users can give their credentials, that is, username and password. This mechanism is convenient for service providers since it provides the possibility of roaming by using AAA servers, such as RADIUS. Further, if used with pre-paid credentials, UAM also provides for

anonymous access. The downside of the method is that usually no link-layer encryption is used in combination. Thus, if users' applications do not encrypt traffic, everything is freely observable by anybody in the proximity of the access point. Some organizations have deployed non-universal methods, such as the use of TLS or IPsec VPN clients for protecting the traffic at the access link.

2.3.2 Location Privacy

Anonymity and location privacy are overlapping concepts, but we can distinguish them as follows: we mean with location privacy that the location of an entity is hidden and with anonymity that the identity of an entity in a location is hidden. The overlap comes from the fact that many anonymity systems discussed in Section 2.2 provide both properties.

The location privacy problem was mentioned already in 1993 in a CACM review article: "a key problem with ubiquitous computing is preserving privacy of location" [138]. The author of the article briefly discusses their transition from centralized location databases with unrestricted access to individual access control in ubiquitous location based applications.

Roaming Networks and Location Based Services Early work on location privacy of mobile computers was theoretical in nature. Cooper and Birman [33] proposed a system model where the mobile computers would communicate via a message server. The mobile computers communicate by assigning labels on the message server. Anonymity by "blinded read" operations: attackers cannot determine the location of the mobile node, because they do not know what position of the shared memory of the message server is currently being read. The blind read resembles what is today known as Private Information Retrieval [110].

Other early theoretical work [10, 123] focused on protecting the privacy of the authentication in a network model similar to GSM: how to protect the privacy of the mobile user when the mobile client needs to be authenticated in the visiting network?

The location privacy problem in cellular networks is related to recent interest in privacy protection in Location Based Services (LBS) [19, 34, 47, 52, 60, 70, 149]. These systems and protocols are important, but beyond the scope of this thesis, since our focus is on privacy problems that occur without the availability of any explicit location based service.

Identifiers in Networking Protocols Network protocols on all layers of the protocol stack use identifiers for establishing and multiplexing connections. Unfortunately, these identifiers introduce privacy problems. The identifiers might not directly reveal the identity of the communication participants, but persistent identifiers can be used to (re)locate the users. The attacker tries to find answers to questions such as: "Did this traffic sample come from device D?"

The first deployed wireless network location privacy mechanism is in the GSM networks according to Samfat et al. [123]. The mechanism is, however, flawed. The mobile phones used pseudonyms called Temporary Mobile Subscriber Identifiers (TMSI) in communication with the access points. However, when the users turn on the mobile

phone, their identities are revealed in the form of International Mobile Subscriber Identifier (IMSI), and TMSI is allocated after that. Thus, an attacker could correlate IMSIs to TMSIs. The mobile phone is also forced to send the IMSI if the synchronization of TMSI between the phone and the home network is lost.

In WLAN networks it was first noted that the 48-bit 802 MAC address [72] can be used for tracking wireless devices [61, 62]. Similar issues were found in Bluetooth [78, 143], in RFID [80], in early versions of the upcoming Bluetooth replacement Wibree [46, 71], and even in special appliances, such as Nike and IPod Sports kit [124]. The main differences between the technologies were that in some of them the addresses might be changed periodically. However, the change of the identifiers was usually done in a predictable manner, before proposals to change the mechanisms were introduced [46, 143].

Unfortunately, the trivially observable hardware address is not the only problem at the WLAN MAC layer. WLAN networks are discovered either by periodic *beacons* broadcast by the access point or active probing by clients. The beacons and probes contain human-readable network names called SSIDs. The probing of the SSIDs presents a problem for user privacy [57, 112]. First of all, the broadcast SSIDs are highly likely [112] to present a unique fingerprint of the user. Second, they also tell in plaintext the history of network service usage of the user, that is, the networks the user has been visiting. Interestingly, the timings of the probe requests can be used to identify the device driver of the 802.11 interface [49] or even to differentiate clients using the same operating system and device drivers [38].

Similar problems with identifier usage also exist when we go up in the protocol stack. The IPv6 [37] addressing architecture [69] allows hosts to configure their addresses automatically without involving a server. The original specifications [134] of the IPv6 stateless address autoconfiguration created a lot of controversy. The host could choose the interface identifier portion of the address by using the EUI-64 encoding of the MAC address. Creating the address this way meant that there would be a persistent unique identifier for the host wherever it might travel. This would have made the privacy of the mobile user much worse compared to the situation with the current Internet Protocol (IPv4) [115]. The privacy problem was fixed five years later in privacy extensions for the IPv6 stateless address autoconfiguration [106] while the autoconfiguration mechanism specification was also updated. The privacy extensions provided a way to choose the interface identifier in random. However, Escudero-Pascual [47] argues that the privacy extensions present another kind of problem for privacy, because an attacker can observe that the privacy extensions are used: it might be interesting in some situations to see which of the clients are to keep hidden from observation. It should be also noted that even the current specifications [135] of the IPv6 stateless address autoconfiguration today allow it to be implemented without the corresponding updated [107] privacy extensions. Interestingly, the use of IP addresses as Home Address or Care of Address in Mobile IP and Mobile IPv6 can also introduce location privacy problems [47, 91] and, for example, the security design of IPv6 did not even consider the location privacy [11]. Finally, even

the IPsec architecture ESP [85] uses identifiers called SPIs that can be used to identify the mobile host, although not persistently [8].

Even application layer identifiers (or names) can be harmful for the location privacy of the user. The user names used in the Session Initiation Protocol [122], which is used for VoIP signaling, for example, can be used to identify the mobile host and the user. Further, it has been shown that dynamic DNS is a very effective way to track the user remotely [63]. Dynamic DNS servers allow the user to register their current IP address for a particular domain name. This enables the users to be reachable with a human-readable name, even though they would be changing their places, and thus, frequently networks and IP addresses. Unfortunately, the IP addresses can be effectively mapped with geolocation services and researchers have even developed an automatic tool for dynamic DNS-based tracking [63].

Fingerprinting Techniques Above, we presented location privacy problems related to the use of identifiers at different layers of the protocol stack. We now examine ways to identify a mobile client in some other, usually non-obvious, ways. These techniques are generally known as fingerprinting, and can be passive or active.

Starting again from the lowest layers of the protocol stack, RF fingerprinting for Ethernet [54], and as noted above, probe-request-based device driver fingerprinting for 802.11 [49] can be used. The probe requests can also be used to fingerprint unique clients that use the same operating system and device drivers [38]. The time of transmission and the signal strength can also be used to identify the mobile node [79], and so can broadcast packet sizes and MAC protocol fields, such as 'more fragments', 'retry' and the supported authentication algorithms [112].

There are many approaches available that can fingerprint the operating system with active scanning. A recent proposal [59] uses TCP SYN packets to open ports and tries to remain undetected. In "remote physical device fingerprinting" [88, 89], the authors use the TCP timestamps to passively observe the clock skew of remote clients and consequently identify the exact client device.

We also observed during our work that censorship-resistance techniques are related to location privacy in a way that perhaps is not obvious. For example, Tor [44] provides in practice good sender and receiver anonymity, given the limitations discussed before. However, it is somewhat easy to spot the use of Tor in the local network. For example, today, Tor clients usually connect to a small number of guard nodes, which already makes the user identifiable because of the persistent destination IP addresses. This is, in fact, similar to the problem of broadcast SSIDs above. Therefore, even though Tor provides identity and location privacy on the Internet, it may harm the location privacy in the local wireless network where the client resides.

Privacy Protection Mechanisms Above, we have explained how the location privacy of a mobile client can be compromised with the use of identifiers. This section presents some techniques that mitigate possible attack vectors.

Gruteser and Grunwald proposed MAC address randomization and also quantified how well the user can be tracked despite the MAC address changes [61, 62]. Their

approach assumed that the client itself would change the address. Later, Jiang et al. [79] proposed that the access point could assign the pseudonymous MAC addresses. However, these proposals themselves neither cover the question of client mobility nor take into consideration the MAC address changes regarding network connectivity.

Concerning the issue of the WLAN MAC layer, Tryst implements confidential discovery of access points [58, 113]. The approach is clean-slate and is combined with SlyFi [58] - an identifier-free link layer, which solves most of the identifier related problems at the MAC layer. However, the approach has some practical implementation issues such as the requirement of synchronized clocks, and does not follow the IEEE 802.11 standard.

To mitigate tracking based on radio signals Li et al. [98] proposed Swing & Swap. They provide analytical and simulation details for the applicability of their approach, but no implementation details are given. In Swing, a wireless node broadcasts an identifier update, and is silent for a random period. In Swap, local mobile nodes are solicited to exchange identifiers with the mobile node. The authors do not consider how connectivity with this approach is maintained. Jiang et al. [79] opportunistically attenuated the transmission power of the wireless interface to make it more difficult for the attacker to track the device by measuring the received signal strength of the radio interface, and they provided experimental data on the applicability of their approach.

Moving up the stack, we could use the anonymity systems presented in Section 2.2 to protect the privacy of the user. There are also other kinds of practical, but limited, mechanisms that provide additional privacy for mobile users without the need for infrastructure support. For example, Aura & Zugenmaier [13] proposed that the mobile host should acquire a new IPv6 address for every new TCP flow. This introduces the equivalent privacy for IPv6 that the Network Address Translation provides for IPv4.

2.4 Summary

So far we have presented related work on privacy enhancing technologies while focusing on the scope of this thesis. We started our discussion with cryptographic building blocks and introduced some elementary protocols that employ similar techniques that are used in modern authentication and key exchange protocols on the Internet. We also looked into the identity protection mechanisms in these protocols. Anonymity systems and practical privacy protection mechanisms on the Internet and their limitations followed. Finally, we discussed security and location privacy mechanisms in mobile systems, the topic of this thesis. The next chapter deals with the practical privacy enhancing technologies we propose in this thesis.

Chapter 3

Practical Privacy Enhancing Technologies

In this chapter, we further discuss the contributions of this thesis, and highlight the benefits and the known limitations of the proposed privacy enhancing technologies. We also contrast the contributions with the main related work and also with recent results. The chapter is organized as follows. We first discuss the systematic analysis of privacy leaks presented in Publication II, and then the proposed privacy preserving mechanisms starting from the first publication and continue with a discussion how the subsequent publications complement our earlier work. Together, the separate contributions help to realize the aim of a privacy-preserving mobile computer.

Publication I discussed in Section 3.2 presents how to leverage a secure mobility management protocol for changing identifiers in the protocol stack to provide privacy and seamless connectivity. Lessons learned from the analysis of privacy leaks presented in Publication II led us to propose a policy for mitigating these leaks, also presented in the same publication and summarized in Section 3.3. The lessons learned also lead us to adopt a pessimistic view of mobile user privacy, since we realized just how vast the amount of leaks were not prevented in the system proposed in Publication I. The approach of Publication I was further refined with the concept of protocol stack virtualization in Publication III discussed in Section 3.4. These approaches unfortunately do not help to mitigate the leaks present in the MAC layer of IEEE 802.11 in access-point discovery. This lead to the work presented in Publication IV and discussed in Section 3.5.

The presented contributions are summarized in the conclusion of this chapter. At the same time the engineering principles applied in this work and the lessons learned are discussed. The chapter ends with considerations for further work.

3.1 Analysis of Privacy Leaks

In Publication II, we uncovered a number of leaks from modern mobile computers: Windows XP and Windows Vista laptops. As already discussed in the Background chapter, many leaks have been separately reported before, whereas our main contribution was the systematic analysis of the leaks at all layers of the protocol stack. To accomplish the analysis, we implemented software that:

1. gathered potentially personally identifiable information from the user's computer,
2. used the gathered information for searching captured traffic logs,

Protocol	Leaks	Layer
Application Metadata	various leaks, e.g. instant messaging	application
DNS	DNS queries, resolving private names and default suffixes	application
NetBIOS	NetBIOS lookup and WINS registration	application
LLMNR	LLMNR name resolution	application
DHCP	host identification and DNS registration	application
Domain Controller	LDAP queries in addition to DNS	application
File shares and printers	mounted network drives, shortcuts to network shares and printer discovery	application
IKE	IKE with GSSAPI	IP
TLS	plaintext certificates in VPN	transport
EAP-TLS	client and server certificates	transport

Table 3.1: Overview of the uncovered privacy leaks

3. and integrated the tools with Microsoft Netmon 3 for manual analysis.

Due to space limitations, we did not report full details of all the leaks uncovered during the analysis in Publication II. Next, we briefly summarize the uncovered leaks in the following order: service or resource discovery, cryptographic authentication and key exchange protocols and application metadata. We end this section with information we discovered on the recommended use of default suffixes.

Many of the leaks were related to unsuccessful service or resource discovery attempts, such as the use of DNS, Netbios, DHCP, mounted network drives and the discovery of printers. An interesting open problem we discovered is ad hoc service discovery, exemplified by the iTunes media player. When two or more previously unknown devices rendezvous, there is no easy way to discover them and preserve privacy, while giving the users meaningful names for the devices. The approach adopted by Apple has interesting consequences: iTunes broadcasts the username and the host name to the network. In the case of the author of this thesis, iTunes broadcasts *"janne @ Janne Lindqvist's computer"*, because the author was asked the full name when he first started to use the computer.

Other interesting leaks were related to the use of names in cryptographic authentication protocols: the client and domain name are leaked from IKE with GSSAPI authentication, and plaintext certificates are leaked from TLS-based virtual private

networks and from EAP-TLS used in *managed* 802.11 based wireless networks. Fortunately, a new specification for the EAP-TLS authentication protocol published in March 2008 adds client privacy protection [126].

We did not elaborate much on application layer leaks in Publication II. We would like to note in this connection that the availability of many personalized services renders privacy protection against passive observers on the local link a demanding task. The problem lies in the fact that the service providers usually have cryptographic authentication for the service, but for the remainder of the session, the data is transmitted in plaintext. This is the case for many instant messengers and web-based services such as online social networks, email, blogs and web searching. Also, browser toolbars conveniently transmit plaintext information about the user. Interestingly, some personalized services that do not require (strong) authentication until subsequent purchases reveal the user identity, because they authenticate the user with cookies stored in the client operating system. The use of cookies is a known privacy problem [95] that has been studied before, and it additionally makes the user vulnerable to passive observers because of greetings along the lines of "Hello Janne, we have recommendations for you...". This privacy problem could be prevented by also encrypting the session traffic data. Unfortunately, many service providers do not have the economic incentives to do that. Some providers do, however, encrypt the session data, too. For example, with Google Mail, the user can *optionally* choose to encrypt the whole Mail session with TLS. However, since the same account is used for different services, which are not encrypted with TLS, the user identity (e.g. firstname.lastname@gmail.com) is revealed as plaintext when using Google Search while logged in to the Mail service. We would like to emphasize that we use Google Mail and Search merely to exemplify why solving the information leakage is a difficult problem.

The final remark we make on the information leak analysis is the use of default suffixes. For example, when the user accesses http://www.tkk.fi, the www.tkk.fi domain name can be appended with a default suffix (e.g. sales.contoso.com) to form the name www.tkk.fi.sales.contoso.com. Recently, we discovered that there is already an Informational RFC [51] published in 1993 that recommends *not* using search path or at least restricting it to local domains. Careless appending of domain names introduces a security risk that can be used to capture connections.

Limitations of the Privacy Leak Analysis The completeness of the analysis can be questioned for two reasons. First, the used method is at best heuristic: we cannot know how many of the possible leaks we were able to capture. We did not capture encrypted or obfuscated traffic (Skype is an excellent example of encrypted and obfuscated traffic [22]). We supported only ASCII, Unicode UTF-8, UTF16 in big- and little-endian byte order, UTF32, and NetBIOS encoding. We did not implement support for other encodings, such as Base64 and uuencode.

The second obvious limitation of the analysis is that we used only Windows XP and Windows Vista operating systems in the analysis. Common operating systems, such as Mac OS X and Linux, were not included in the analysis. As an anonymous

reviewer of the publication noted, it would be also interesting to apply the tool to various WLAN-enabled devices and entertainment devices, such as mp3 players. In fact, some related leaks from entertainment devices have been reported, for example, from the Nike+iPod sports kit [124].

Recently, after our work was published, work with similar goals was published. Privacy Oracle [81] detects leaks using "black-box differential fuzz testing": applications are given variable inputs (e.g. for username bob and alice), and network traces are compared for the differences. In contrast to our work, Privacy Oracle can even detect hashed usernames, while we detect identifiers that were explicitly available on the user's computer. However, Privacy Oracle requires setting up the test separately for every application to be tested, whereas we can gather the identifiers in one run and analyze network traces from ordinary computer usage.

The systematic analysis of the leaks was crucial in understanding the limitations of the following privacy-preserving mechanisms that follow. We hope that the work will have an impact on the overall awareness on mobile computer user privacy.

3.2 Privacy Management for Secure Mobility

The privacy management for secure mobility marked the launch of the author's privacy research. The work for Publication I started from the observation that the use of Host Identifiers in the Host Identity Protocol (HIP) [104] can introduce privacy violations for both identity and location. In this context, the identity means the mobile system, which could be used further to identify the mobile user. The location means the physical location, e.g. the access point location, since the network location will in any case be revealed because of the network-prefix in IPv6 address. The HIP architecture [103] basically proposes two kinds of identifiers, long-lasting public and temporary "anonymous". The identifiers are public/private key pairs and hashes of the public keys are used in the protocol messages. However, neither the Host Identity Protocol nor the architecture specifies how to manage the use of these identifiers; neither does the HIP "native API" [90]. Thus, anonymous and public identifiers might be used at the same time, and only one of the identifiers needs to be public for the anonymity of all other identifiers to be immediately compromised. Similar problems exist with other protocols and privacy architectures (e.g. SIP [122]). Additionally, the privacy of the identifiers can be compromised if any inbound connections use public identifiers. One of the obvious ways how all these identifiers can be linked together is the hardware MAC address. Thus, the operating system needs to make sure that these kind of identifiers are not used at the same time.

We did not limit ourselves to the management of public and anonymous identifiers, although it was one of the more important aspects of the work. We also wanted to provide privacy-preserving seamless mobility meaning that even though a mobile system needs to use some (implicit) identifiers, such as MAC and IP addresses, the connections should not be severed when using changing pseudorandom identifiers. Previous work had proposed using pseudorandom MAC addresses [61, 62], which would have resulted

in severed connections. Instead of focusing on a single layer, however, we decided to consider all layers of the protocol stack. Independently, the use of shared pseudorandom sequences for all layers of protocol stack had been proposed [8]. But in contrast to this proposal, it was only necessary to change the MAC and IP addresses (and possibly the used keys) from time to time, for example, when changing the access point. The transport and application layer traffic is protected with IPsec ESP provided by HIP. Furthermore, with implementation experience, we showed that our approach works, whereas the previous independent proposal [8] did not contain implementation experience.

In hindsight, the privacy management system should have been implemented on SSH and TLS, and for mobility we could have used resilient SSH and TLS connections [92]. This would have made the approach much more deployable, since HIP itself is not essential for the approach to work, only the ability to encrypt transport and application layer traffic. However, since the authors were involved in a HIP project, their thinking was naturally directed towards improving the HIP implementation.

Limitations of the Approach The first unfortunate limitation of the approach is related to deployability. There are practically no incentives for server operators to deploy the proposed mechanism. A similar problem exists with the use of end-to-end encryption (even for TLS) for protecting session data. It is more cost-effective to only use cryptographic protection in the authentication process, compared to encrypting the whole session. Service providers lack an economic incentive for protecting the session, because they are not responsible for the potential loss of privacy. If service providers started to provide TLS protection for whole sessions, then they might get interested in providing support for privacy management too. Reimplementing the approach in a way that would not affect server performance substantially might motivate service providers to implement the approach as a sign of good will. The author of this thesis currently believes that there are not enough incentives to deploy HIP on client or server side.

The possibility to observe the services the users contact limits the privacy protection provided by the approach. For example, the users might contact the web or email servers of the organization they represent. A study [112] using 802.11 network traces from SIGCOMM 2004, and one day of all traffic from U.C. San Diego's CS building reported that, on average, 60 % of the users (with false positive rate of 0.01) could be identified by their use of destination IP addresses alone. A further limitation of the approach is that many of the leaks covered by our analysis in Publication II cannot be protected with the use of end-to-end encryption only. Thus, even though the MAC addresses and IP addresses could be changed, the DHCP and WLAN interface might still broadcast enough information for identifying the mobile node.

3.3 Network Location Awareness Based Privacy Policy

To prevent the various leaks we analyzed in Publication II, we also proposed a new privacy policy for (mobile) operating systems:

> "When client software stores information about an online service for the purpose of connecting to it later, it must also store the NLA network

> identifiers of the access links where the service is known to be accessible. Automatic connection attempts to the service are only allowed on those networks."

Or more succinctly:

> Automatically connect to a service only in networks where the service is known to exists.

The idea behind the policy is simply to identify previously visited networks and control the use of networked applications based on the settings for that particular network. The key observation behind this policy proposal was that many of the leaks were the results of unsuccessful service discovery attempts. In Publication II, we briefly discuss what needs to be considered when implementing the policy using, for example, network location awareness services in Windows Vista. However, we did not implement the mechanism. Thus, it remains to be seen how good the proposed policy would be in practice. We further note that the policy does not prevent application layer leaks.

There are other alternatives to the proposed policy that were not discussed in Publication II. In the next paragraphs, we summarize these alternatives. To begin with, we could think of a statistical protection mechanism, that is, using the network connection only when it is needed. Unfortunately, this kind of approach is at best heuristic, and thus, brittle and can have adverse effects on user experience. Similarly, we could think of disabling all automatic service discovery. This approach would not only make the user experience bad, but would only solve the problems related to the service discovery. For similar reasons, manual configuration for each network would not work.

It might also be thought that the users could tunnel everything through a VPN. This is feasible for most protocols, but not for the basic service discovery protocols such as DHCP or printer discovery. Further, it is highly likely that the VPN server would introduce a new very distinct identifier for the users and their affiliations: the domain name and the IP address of the VPN server. These identifiers would persist for the user network to network, similar to the default IPv6 stateless address autoconfiguration interface identifier based on a hardware address. Instead of a fixed VPN server (or server farm), the mobile host could be configured to use deployed anonymity networks, such as Tor by default, but this would harm the user experience because of the latency these networks introduce, and might also produce a distinct identifier as will be discussed later in Section 3.8.

The above work is closely related to a proposal for secure NLA [12]. Secure NLA adds public key authentication to DHCP messages for providing a simple and efficient way to detect previously visited networks. The authors of the secure NLA designed the protocol to preserve client privacy. However, implementing the proposed privacy policy should not require secure NLA. To succeed in falsifying the location, the attacker needs a considerable amount of information from the networks that the potential victims visits. Therefore, the privacy policy should be able to be implemented by modifying only the client operating system.

Limitations The proposed policy has not been implemented, and thus our analysis of it is at best an educated guess. The policy should not impact user experience, if we assume that the user is familiar with the NLA mechanism already. Implementation of the policy is not trivial since we need to ensure that user experience is not affected. However, at this point, we cannot know the tradeoffs that the implementation would introduce to the mobile system. Therefore, implementing the policy would be worth further work. The implementation should be designed not to assume anything about the deployed networks, and not to impact in any way corporate network design. Otherwise, it is unlikely that it will be taken into use. There are also some subtleties that we did not discuss in Publication II. For example, leaking the corporate default home page can be as harmful as leaking the local DNS suffixes. Fortunately, both can be controlled and set accordingly.

3.4 Protocol Stack Virtualization

The privacy protection mechanisms discussed above do not completely prevent application layer leakage by protocol metadata or Web page access, for example. Although encrypting payload data as suggested above would help, it is not feasible to assume that this would happen in all possible cases. Thus, in Publication III, we adopted a very pessimistic view of user privacy: *leaks will happen*. We then asked the question how these leaks, especially the application layer leaks, could be mitigated by modifying only the client and without changing the user experience at all. This work continues the line of thinking presented in Publication I; we again use pseudorandom identifiers in the protocol stack, but this time we modify only a single host. In essence, the system provides traffic isolation. For example, different applications using the network can seem to be different hosts because they have different MAC addresses and IP addresses. Protocol stack virtualization was designed not to rely on the NLA privacy policy, and is independent of it. Still, protocol stack virtualization is complementary with NLA and the privacy management approach presented above. Moreover, the virtualization approach addresses some leaks those approaches cannot cover.

Independently, a similar approach called FLASCHE [150] had been proposed for mobile user privacy. FLASCHE proposed to use "location addresses". In other words, the address of the mobile host would be tied to a location and clients in the same location would be distinguished by a random part in the address. The implementation of FLASCHE was limited to HTTP at the application layer, and to IEEE 802.11b at the link layer. In contrast to this, we decided from the beginning to support all possible applications and legacy operating systems, and thus implemented the approach akin to a host-based network address translator. We showed that our approach worked with all of the varying legacy applications we tried, and further, we showed that user experience was not affected by the approach, even in IPv4 networks.

One question that remains to be answered is the amount of improvement in privacy provided by the approach. We did not try to quantify it, and exact quantification might not even be feasible.

3.5 Privacy-Preserving 802.11 Access-Point Discovery

Discussions of privacy problems in personal area networks, such as Wibree, were the inspiration for this work. However, we moved our focus to IEEE 802.11 WLAN since it has been successfully deployed all around the world, and the Wibree is still waiting to be launched. Our initial goal was to solve the privacy problems created by the disabling of the SSID broadcast without re-enabling it. This section highlights a problem that can not be solved by any of the other privacy protection mechanisms discussed above alone. In the current IEEE 802.11 [76] protocol, enabling the SSID broadcast is unacceptable for some network owners, yet disabling the broadcast creates much more serious privacy issues for the clients. We present a system that preserves all the advantages of hidden networks, that is, the SSID broadcast can be disabled, while the system provides stronger privacy protection for the clients than the existing alternatives. In other words, our protocol is a win-win solution that does not require the network administrators to choose between protecting the network or the clients. In fact, we end up with slightly better privacy and security for the network, and also a more efficient hidden network discovery than in the current legacy implementations. The contributions of the work in Publication IV include the following: analysis of the privacy issues in WLAN that had been presented in previous work, analysis of previous work on private identification, design requirements, a system-centred protocol design, implementation, overhead measurements, formal verification of the security and privacy properties of the protocol.

The SSID broadcast problem was addressed by Tryst [58, 113] and SlyFi [58]. However, the authors of Tryst and SlyFi opted for a clean-slate design that replaced the whole protocol stack, whereas we desired deployability requiring integration with the standard IEEE 802.11 from the beginning.

Limitations The major limitation we see in our approach is that it requires a shared key between the network and the client. As a positive side of the limitation, the user experience and configuration does not change for networks protecting the link-layer with, for example, WPA-PSK. In contrast, for managed networks using, for example EAP-TLS authentication, a shared key is needed in addition to the other parameters EAP-TLS provides. We have left the bootstrapping in managed networks as an open problem. Further, the authentication used in managed networks at least reveals the server certificates. However, this does not introduce a problem for client privacy. We could prevent the leakage of certificates during the EAP-TLS authentication by encrypting it with the shared key, but this would mean more radical changes to the standard IEEE 802.11 protocol.

Finally, as with any security technology it is not useful if there are no users for the technology. For example, if nobody else other than you uses encryption software, it is not useful to encrypt your email. Similarly, a privacy-preserving access-point discovery is not useful, if access-point owners do not deploy it.

3.6 Summary of Contributions

The contributions presented above complement each other in the quest of implementing a privacy-preserving mobile computer. We first discussed the information leaks from modern mobile operating systems, and how these leaks affect the identity and organizational privacy of the user. The analysis of the leaks was crucial in understanding the limitations of our privacy protection proposals. Then, we discussed different but complementing approaches starting from the earliest publication of the author. The management of identifiers on all layers of the protocol stack, a privacy policy using network location awareness and protocol stack virtualization all provide privacy without affecting the user experience, but with different deployability obstacles. Finally, we presented privacy-preserving 802.11 access-point discovery, which is crucial for preventing trivial user profiling in WLAN networks.

3.7 Engineering Principles and Lessons Learned

We started this work advocating the position that privacy needs to be considered from the system viewpoint, especially when enhancing the privacy of legacy systems. We did not know from the beginning how right we were in taking this viewpoint, and how many lessons we would need to learn in the process.

In designing the privacy-preserving mechanisms, we have followed an approach that is more common in systems research than in systems security research. Instead of proposing new systems from a clean-slate, we retrofitted security and privacy to real-world systems used today.

In the security community, there lives a meme that security cannot be introduced to a system after the design and implementation of the system. Instead, security design should be integrated into the system design from the beginning. In practice, this kind of integrated security design does not happen for real-world systems, the sole exception being perhaps military-grade systems. Instead, when attacker models change or new flaws are recovered, security must be designed after the system has been deployed.

In this thesis, we have followed the approach of viewing the mobile host as a system that needs to be secured from information leaks that could be used to identity and locate the user of the mobile system. We first thought that it was adequate to consider the mobile system from the viewpoint of "all layers in the protocol stack". After the systematic analysis of privacy leaks, however, we understood that the approach alone was not at all adequate.

System design is always about tradeoffs. Complete network security can be achieved only by not connecting the computer to a network. And even then, the human factor needs to be considered in order to ensure that nobody accidentally connects the computer to the network. The tradeoffs we considered in this thesis were between user experience, deployability, and security of the proposed approaches.

One of the key lessons learned from this work was from the design of the privacy-preserving 802.11 access-point discovery. It was obvious that to cover the whole mobile system, something needed to be done about the IEEE 802.11 access-point discovery,

in addition of course to the well-known problem of MAC address based identification and tracking. When trying to ensure that the user experience does not change and that our design would be deployable and would provide real security and privacy properties, we discovered that protocol engineering is quite different from developing an abstract secure protocol.

At some point in the work, we thought that changing the user experience would facilitate the deployability of the protocol. The privacy-preserving discovery protocol could work without using any name in configuring the network. However, during discussions with usability and networking experts, we abandoned this track. The user experience should not be changed in that way, since the networks in any case need to be called something, that is, they need a name. User interface integration issues also came up in the later stages of the implementation. When hidden networks are configured in the client user interface, the user needs to specify the type of link-layer encryption used, e.g., WEP, WPA Personal, or WPA2 Enterprise. The used names may even vary somewhat depending on the used operating system. Since the client operating system needs to know during the configuration process whether it is configuring a legacy hidden network or a secure hidden network established by our protocol, we need to add that information to the configuration interface. The approach which seemed the most viable, but not confirmed with usability tests, was to annotate the link-layer encryption descriptions. For example, the user interface could say "WPA Personal + privacy discovery". Another option could have been to have the user click a button in the configuration interface "this network is securely hidden", but we thought that users might easily forget to click that additional new button.

The other key lesson learnt during the work was that it is not enough to redesign the access-point discovery to be correctly privacy-preserving. It is also relevant what happens after the discovery procedure. Thus, we needed to look at all the protocol messages that may be sent to the access network during the attachment procedure. Our initial design might otherwise have been of little use, since after the discovery process, the SSID would have been revealed. We had a formal proof of the initial design, which showed that it was correct. Of course, we had not modeled the properties of the system that we did not know about at that point.

Similarly to the 802.11 access-point discovery, the system aspects were a priority when designing and implementing the protocol stack virtualization approach. It did not suffice that we merely provided different address spaces for every flow. We needed to consider how a mobile system is used in practice, since it involves DNS requests and ARP queries as well, for example. This introduced complexity since we decided we needed a transparent proxy for DNS to be able to filter the requests appropriately. Similarly, for ARP, we needed a protocol helper to be able to map the requests correctly. If these and other protocols had not been considered during the design, the protocol stack virtualization would not have been of much use.

In summary, retrofitting privacy enhancements to already deployed systems is possible, but takes considerable effort, even when the additions are simple and small. In

that sense, the security community is right that security should be integrated to systems from the beginning. The engineering work for privacy enhancement is analogous to the advancement of cryptography. First, a flaw is discovered in a system or protocol. If the flaw, the system, or the protocol is significant enough, solutions are proposed by the community. The published solutions may contain further flaws and subsequently be revised. However, in contrast to the systems community, formally proofing the design is usually a requirement in the cryptographic community. We found this is also beneficial in reducing obvious flaws in our protocol design. Unfortunately, not all aspects of the system are amenable to proofs in practice. Thus, the iterative process of engineering is needed when redesigning real-life systems.

3.8 Further Considerations

In this thesis, we set out to find the tradeoffs of mobile computer usage in relation to privacy and to find out how to mitigate or prevent privacy leaks. We think we have answered these research questions appropriately. As shown, even though our analysis of privacy tradeoffs cannot be considered as complete, it was fundamental in evolving our thinking on the problem space. Given the constraints we set for the work, user experience and deployability, we believe that our design is practical enough to be deployed. Nevertheless, it remains to be seen whether our designs will really be deployed in real-networks, or whether the alternative clean-slate solutions are preferred, for example, by standardization bodies. Next, we discuss some further work that we did not answer in this thesis.

Most of the privacy enhancing technologies presented in this thesis make it actually easier to notice the user in today's networks. Especially in the case when the user is the only one in the network using a particular mechanism; the mechanism is an anomaly in the network traffic, and the user is therefore easier to spot. Thus, the location privacy of the user is much easier to compromise. However, this is the situation for most novel and practical privacy enhancing technologies. As discussed in the background chapter, even Tor [44] can be harmful for location privacy despite the fact it provides sender-receiver anonymity. In a wireless network, the users that are using Tor can be more easily distinguished from other users. Clearly, further work is warranted in analyzing techniques that could allow the user to blend into crowds in current networks, even though nobody else might be using similar mechanisms. The implementation of the NLA-based privacy policy could be one example. Maybe only MAC address randomization at some intervals would in the final analysis serve to be indistinguishable from other network traffic.

It is interesting that this work highlights the differences between using open networks and closed networks. An attacker in GPRS networks, for example, needs much more effort to detect leaks similar to the ones we uncovered in Publication II, compared to the situation of using WLAN. The disadvantage is that the location privacy in relation to the service provider is automatically lost in closed networks, such as GPRS.

As further work, it would also be interesting to quantify how large the network has to be to gain the benefits of many of the approaches when the client is mobile. For example, if protocol stack virtualization is used, and the client switches the access point, all the identifiers appear simultaneously in the new access point. This information could be used to reveal clients that are using the protocol stack virtualization.

Chapter 4

Conclusions

Privacy is at stake today. Many western societies could almost be called surveillance societies. Surveillance cameras are becoming more and more common, and an increasing number of countries grant some of their governmental agencies the right to observe and retain all communication traffic traversing their country. Even some governments, which have democratically elected representatives, spy on their own citizens in "the war against terrorism", and anti-piracy organizations are given access to log files of popular services in the quest to reveal illegal distribution of copyrighted material.

In this thesis, we have focused on protecting the users of wireless mobile networks from much more resource-constrained attackers than governmental agencies. We were not interested in a potentially omnipresent attacker; instead we were concerned about the passive observer in the local link or a few hops away. We have shown that even this kind of attacker with very limited resources, perhaps equipped only with a laptop, can observe a considerable amount of personally identifiable information and implicit information for identifying and tracking interesting targets. What the attacker can do with the information varies, and can range from cyberstalking to targeting employees of interesting organizations for physical attacks, such as stealing a laptop.

Many of the privacy problems we have discussed in this thesis are caused by unsuccessful service discovery attempts and the use of explicit and implicit identifiers on all layers of the protocol stack. We have presented mechanisms that prevent or mitigate these leaks. None of the solutions presented in this thesis can accomplish much alone to preserve the privacy of the user. However, the combination of these mechanisms is essential for building a privacy-preserving mobile computer. When all of the mechanisms are implemented in a mobile computer, the privacy of the user is clearly improved. In fact everything we have proposed has been implemented in software and in a way that does not affect the user experience at all.

Unfortunately, protecting privacy still requires the involvement of users, operating systems and device manufacturers. For example, even though we can do a decent job of protecting accidental leaks, we cannot prevent the users from explicitly identifying themselves when they access a personalized web-based service. If service providers have no incentives to provide end-to-end encryption, mobile user privacy will remain insufficient due to the application layer plaintext leaks. This work shows that privacy-preserving mobile computers need to consider the whole protocol stack, including network attachment methods. Additionally, a cultural change in the way network-enabled applications are developed might be needed in order to provide more complete privacy protection mechanisms.

Finally, this thesis highlights the fact that preserving the privacy of the user remains a difficult problem. We have not built a perfect solution. Nevertheless, we have shown that modifying a mobile host to be more privacy-preserving can be achieved with simple and efficient methods. We think that the solutions presented in this thesis provide a baseline for designing future wireless network standards and also provide much needed insight into implementing privacy-preserving network-enabled applications and protocols.

References

[1] Martín Abadi. Private authentication. In *Second International Workshop on Privacy Enhancing Technologies (PET 2002)*, April 2002.

[2] Martín Abadi and Cédric Fournet. Private authentication. *Theor. Comput. Sci.*, 322(3):427–476, September 2004.

[3] Martín Abadi and Roger Needham. Prudent engineering practice for cryptographic protocols. *IEEE Transactions on Software Engineering*, January 1996.

[4] Dakshi Agrawal and Dogan Kesdogan. Measuring anonymity: The disclosure attack. *IEEE Security & Privacy*, 1, November–December 2003.

[5] Dakshi Agrawal, Dogan Kesdogan, and Stefan Penz. Probabilistic Treatment of MIXes to Hamper Traffic Analysis. In *IEEE Symposium on Security and Privacy*, May 2003.

[6] William Aiello, Steven M. Bellovin, Matt Blaze, Ran Canetti, John Ioannidis, Angelos D. Keromytis, and Omer Reingold. Just fast keying: Key agreement in a hostile internet. *ACM Transactions on Information and System Security (TISSEC)*, 7, May 2004.

[7] William Aiello, Steven M. Bellovin, Matt Blaze, John Ioannidis, Omer Reingold, Ran Canetti, and Angelos D. Keromytis. Efficient, DoS-resistant, secure key exchange for internet protocols. In *9th ACM conference on Computer and Communications Security*, November 2002.

[8] Jari Arkko, Pekka Nikander, and Mats Näslund. Enhancing Privacy with Shared Pseudo Random Sequences. In *Proc. of Security Protocols*, April 2005.

[9] Giuseppe Ateniese, Marina Blanton, and Jonathan Kirsch. Secret handshakes with dynamic and fuzzy matching. In *Proc. of NDSS '07*, February 2007.

[10] Giusseppe Ateniese, Amir Herzberg, Hugo Krawczyk, and Gene Tsudik. Untraceable mobility or how to travel incognito. *Comput. Netw.*, 31(9):871–884, 1999.

[11] Tuomas Aura and Michael Roe. Designing the Mobile IPv6 Security Protocol. *Annales des télécommunications / Annals of telecommunications, special issue on Network and information systems security*, 61(3-4), March-April 2006.

[12] Tuomas Aura, Michael Roe, and Steven J. Murdoch. Securing Network Location Awareness with Authenticated DHCP. In *SecureComm*, September 2007.

[13] Tuomas Aura and Alf Zugenmaier. Privacy, Control and Internet Mobility. In *Security Protocols, 12th International Workshop*, April 2004.

[14] Ashar Aziz and Whitfield Diffie. Privacy and Authentication for Wireless Local Area Networks. *IEEE Personal Communications*, 1, First Quarter 1994.

[15] Adam Back, Ulf Möller, and Anton Stiglic. Traffic analysis attacks and trade-offs in anonymity providing systems. In *Information Hiding Workshop (IH 2001) LNCS (2137)*, April 2001.

[16] Dirk Balfanz, Glenn Durfee, Rebecca E. Grinter, D.K. Smetter, and Paul Stewart. Network-in-a-Box: How to Set Up a Secure Wireless Network in Under a Minute. In *Proc. of USENIX Security*, May 2004.

[17] Dirk Balfanz, Glenn Durfee, Narendar Shankar, Diana Smetters, Jessica Staddon, and Hao-Chi Wong. Secret handshakes from pairing-based key agreements. In *Proc. of IEEE Security and Privacy*, May 2003.

[18] Mihir Bellare, Alexandra Boldyreva, Anand Desai, and David Pointcheval. Key-privacy in public-key encryption. In *ASIACRYPT*, December 2001.

[19] Alastair R. Beresford and Frank Stajano. Location privacy in pervasive computing. *IEEE Pervasive Computing*, 2, January–March 2003.

[20] Andrea Bittau, Mark Handley, and Joshua Lackey. The Final Nail in WEP's Coffin. In *IEEE Symposium on Security and Privacy*, May 2006.

[21] Dan Boneh, Xavier Boyen, and Hovav Shacham. Short group signatures. In *Proc. of Crypto*, August 2004.

[22] Dario Bonfiglio, Marco Mellia, Michela Meo, Dario Rossi, and Paolo Tofanelli. Revealing skype traffic: When randomness plays with you. In *SIGCOMM*, August 2007.

[23] Nikita Borisov, Ian Goldberg, and David Wagner. Intercepting Mobile Communications: The Insecurity of 802.11. In *ACM SIGMOBILE Annual International Conference on Mobile Computing and Networking (MobiCom)*, July 2001.

[24] Michael Burrows, Martín Abadi, and Roger Needham. A logic of authentication. *ACM Trans. Comput. Syst.*, 8(1):18–36, 1990.

[25] Jan Camenisch and Anna Lysyanskaya. Dynamic Accumulators and Application to Efficient Revocation of Anonymous Credentials. In *CRYPTO*, August 2002.

[26] David Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Commun. ACM*, 24(2):84–90, February 1981.

[27] David Chaum. Blind signatures for untraceable payments. In *CRYPTO'82*, pages 199–203. Springer-Verlag (1983), 1983.

[28] David Chaum. Security without identification: transaction systems to make big brother obsolete. *Commun. ACM*, 28(10):1030–1044, 1985.

[29] David Chaum. The dining cryptographers problem: Unconditional sender and recipient untraceability. *Journal of Cryptology*, 1, 1988.

[30] David Chaum, Amos Fiat, and Moni Naor. Untraceable electronic cash. In *CRYPTO*, August 1998.

[31] Ian Clarke, Scott G. Miller, Theodore W. Hong, Oskar Sandberg, and Brandon Wiley. Protecting free expression online with freenet. *IEEE Internet Computing*, 6(1):40–49, 2002.

[32] Ian Clarke, Oskar Sandberg, Brandon Wiley, and Theodore W. Hong. Freenet: A Distributed Anonymous Information Storage and Retrieval System. In *Designing Privacy Enhancing Technologies: International Workshop on Design Issues in Anonymity and Unobservability, LNCS 2009*, July 2001.

[33] David A. Cooper and Kenneth P. Birman. Preserving privacy in a network of mobile computers. In *IEEE Symposium on Security and Privacy*, May 1995.

[34] Landon P. Cox, Angela Dalton, and Varun Marupadi. SmokeScreen: Flexible Privacy Controls for Presence-Sharing. In *Proc. of MobiSys '07*, June 2007.

[35] George Danezis. Statistical disclosure attacks: Traffic confirmation in open environments. In *Proceedings of Security and Privacy in the Age of Uncertainty, (SEC2003)*, May 2003.

[36] George Danezis and Andrei Serjantov. Statistical disclosure or intersection attacks on anonymity systems. In *Proceedings of 6th Information Hiding Workshop (IH 2004)*, May 2004.

[37] S. Deering and R. Hinden. RFC 2460: Internet Protocol, Version 6 (IPv6) Specification, December 1998. Status: Draft Standard.

[38] Loh Chin Choong Desmond, Cho Chia Yuan, Tan Chung Pheng, and Ri Seng Lee. Identifying Unique Devices through Wireless Fingerprinting. In *ACM Conference on Wireless Network Security (WiSec)*, March–April 2008.

[39] Claudia Díaz, Stefaan Seys, Joris Claessens, and Bart Preneel. Towards measuring anonymity. In *Proceedings of Privacy Enhancing Technologies Workshop (PET 2002), LNCS 2482*, April 2002.

[40] Whitfield Diffie and Martin E. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, 22, November 1976.

[41] Whitfield Diffie and Susan Landau. *Privacy on the Line: The Politics of Wiretapping and Encryption – Updated and Expanded Edition*. The MIT Press, May 2007. ISBN 978-0-262-04240-6.

[42] Whitfield Diffie, Paul C. van Oorschot, and Michael J. Wiener. Authentication and authenticated key exchanges. *Designs, Codes and Cryptography*, 2, 1992.

[43] Roger Dingledine and Nick Mathewson. Anonymity Loves Company: Usability and the Network Effect. In *Workshop on the Economics of Information Security*, June 2006.

[44] Roger Dingledine, Nick Mathewson, and Paul Syverson. Tor: The Second-Generation Onion Router. In *Proceedings of the 13th USENIX Security Symposium*, August 2004.

[45] Roger Dingledine, Nick Mathewson, and Paul Syverson. Deploying low-latency anonymity: Design challenges and social factors. *IEEE Security and Privacy*, 5(5):83–87, 2007.

[46] Jan-Erik Ekberg. Implementing Wibree Address Privacy. 1st International Workshop on Security for Spontaneous Interaction, September 2007.

[47] Alberto Escudero-Pascual. *Privacy in the next generation Internet: Data protection in the context of European Union policy*. PhD thesis, Royal Institute of Technology, 2002.

[48] Scott R. Fluhrer, Itsik Mantin, and Adi Shamir. Weaknesses in the Key Scheduling Algorithm of RC4. In *Revised Papers from the 8th Annual International Workshop on Selected Areas in Cryptography (SAC)*, August 2001.

[49] Jason Franklin, Damon McCoy, Parisa Tabriz, Vicentiu Neagoe, Jamie Van Randwyk, and Douglas Sicker. Passive Data Link Layer 802.11 Wireless Device Driver Fingerprinting. In *Proc. of USENIX Security*, pages 167–178, July/August 2006.

[50] Steven Furnell and Bogdan Ghita. Usability pitfalls in Wireless LAN security. *Network Security*, 2006(3), March 2006.

[51] Ehud Gavron. RFC 1535: A Security Problem and Proposed Correction With Widely Deployed DNS Software, October 1993.

[52] Bugra Gedik and Ling Liu. Location Privacy in Mobile Systems: A Personalized Anonymization Model. In *25th IEEE International Conference on Distributed Computing Systems (ICDCS)*, June 2005.

[53] Jim Geier. *Wireless Networks first-step*. Cisco Press, August 2004. ISBN 1-58720-111-9.

[54] Ryan Gerdes, Thomas Daniels, Mani Mina, and Steve Russell. Device identification via analog signal fingerprinting: A matched filter approach. In *The 13th Annual Network and Distributed System Security Symposium (NDSS)*, February 2006.

[55] David Goldschlag, Michael Reed, and Paul Syverson. Onion routing. *Commun. ACM*, 42(2):39–41, 1999.

[56] David M. Goldschlag, Michael G. Reed, and Paul F. Syverson. Hiding Routing Information. In *Workshop on Information (LNCS 1174)*, 1996.

[57] Ben Greenstein, Ramakrishna Gummadi, Jeffrey Pang, Mike Y. Chen, Tadayoshi Kohno, Srinivasan Seshan, and David Wetherall. Can Ferris Bueller Still Have His Day Off? Protecting Privacy in an Era of Wireless Devices. In *Proc. of HotOS XI*, May 2007.

[58] Ben Greenstein, Damon McCoy, Jeffrey Pang, Tadayoshi Kohno, Srinivasan Seshan, and David Wetherall. Improving wireless privacy with an identifier-free link layer protocol. In *Proc. of MobiSys '08*, June 2008.

[59] Lloyd G. Greenwald and Tavaris J. Thomas. Toward undetected operating system fingerprinting. In *First USENIX workshop on Offensive Technologies (WOOT)*, August 2007.

[60] Marco Gruteser and Dirk Grunwald. Anonymous Usage of Location-Based Services through Spatial and Temporal Cloaking. In *Proceedings of First ACM/USENIX International Conference on Mobile Systems, Applications, and Services (MobiSys)*, May 2003.

[61] Marco Gruteser and Dirk Grunwald. Enhancing location privacy in wireless LAN through disposable interface identifiers: A quantitative analysis. In *Proc. of ACM WMASH*, September 2003.

[62] Marco Gruteser and Dirk Grunwald. Enhancing location privacy in wireless LAN through disposable interface identifiers: a quantitative analysis. *Mob. Netw. Appl.*, 10(3):315–325, 2005.

[63] Saikat Guha and Paul Francis. Identity Trail: Covert Surveillance Using DNS. In

Workshop on Privacy Enhancing Technologies (PET), June 2007.

[64] Dan Harkins and Dave Carrell. The Internet Key Exchange (IKE), November 1998.

[65] Changhua He and John C. Mitchell. Analysis of the 802.11i 4-way handshake. In *3rd ACM Workshop on Wireless Security (WiSe)*, October 2004.

[66] Changhua He and John C. Mitchell. Security analysis and improvements for ieee 802.11i. In *Symposium on Network and Distributed System Security (NDSS)*, February 2005.

[67] Changhua He, Mukund Sundararajan, Anupam Datta, Ante Derek, and John C. Mitchell. A modular correctness proof of IEEE 802.11i and TLS. In *12th ACM Conference on Computer and Communications Security (CCS)*, November 2005.

[68] Sabine Helmers. A brief history of anon.penet.fi - the legendary anonymous remailer. *Computer-Mediated Communication Magazine (CMC)*, September 1997.

[69] Robert Hinden and Stephen Deering. RFC 4291: IP Version 6 Addressing Architecture, February 2006. Status: Draft Standard.

[70] Jason I. Hong and James A. Landay. An architecture for privacy-sensitive ubiquitous computing. In *Proc. of MobiSys*, June 2004.

[71] Mauri Honkanen, Antti Lappeteläinen, and Kalle Kivekäs. Low end extension for Bluetooth. In *IEEE Radio and Wireless Conference*, September 2004.

[72] IEEE. *IEEE Std 802-1990, IEEE Standards for Local and Metropolitan Area Networks: Overview and Architecture*. The Institute of Electrical and Electronics Engineers, Inc., 1990.

[73] IEEE Standard for Information technology – Telecommunications and information exchange between systems – Local and metropolitan area networks – Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. IEEE Std 802.11-1997, November 1997.

[74] IEEE Standard for Information technology – Telecommunications and information exchange between systems – Local and metropolitan area networks – Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications – Amendment 6: Medium Access Control (MAC) Security Enhancements. IEEE Std 802.11i-2004, July 2004.

[75] IEEE Standard for Local and Metropolitan Area Networks – Port-Based Network Access Control. IEEE Std 802.1X-2004, December 2004.

[76] IEEE Standard for Information technology – Telecommunications and information exchange between systems – Local and metropolitan area networks – Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. IEEE Std 802.11-2007 (Revision of IEEE Std 802.11-1999), June 2007.

[77] ISO/IEC. Information technology – Security techniques – Entity authentication – Part 4: Mechanisms using a cryptographic check function, 1999. Reference number ISO/IEC 9798-4:1999(E).

[78] Markus Jakobsson and Susanne Wetzel. Security Weaknesses in Bluetooth. In *CT-RSA 2001: Proceedings of the 2001 Conference on Topics in Cryptology*, pages 176–191, London, UK, 2001. Springer-Verlag. LNCS 2020.

[79] Tao Jiang, Helen J. Wang, and Yi-Chun Hu. Location privacy in wireless networks. In *Proc. of MobiSys '07*, June 2007.

[80] Ari Juels. RFID security and privacy: a research survey. *IEEE JSAC*, February 2006.

[81] Jaeyeon Jung, Anmol Sheth, Ben Greenstein, David Wetherall, Gabriel Maganis, and Tadayoshi Kohno. Privacy Oracle: a system for finding application leaks with black box differential testing. In *15th ACM conference on Computer and communications security*, October 2008.

[82] Burt Kalinski. RFC 2898: PKCS #5: Password-Based Cryptography Specification Version 2.0, September 2000.

[83] Charlie Kaufman. RFC 4306: Internet Key Exchange (IKEv2) Protocol, December 2005.

[84] Stephen Kent. RFC 4302: IP Authentication Header (AH), December 2005.

[85] Stephen Kent. RFC 4303: IP Encapsulating Security Payload (ESP), December 2005.

[86] Stephen Kent and Karen Seo. RFC 4301: Security Architecture for the Internet Protocol, December 2005.

[87] Dogan Kesdogan, Dakshi Agrawal, and Stefan Penz. Limits of anonymity in open environments. In *Information Hiding Workshop (IH 2002) LNCS (2578)*, October 2002.

[88] Tadayoshi Kohno, Andre Broido, and K.C. Claffy. Remote physical device fingerprinting. In *IEEE Symposium on Security and Privacy*, May 2005.

[89] Tadayoshi Kohno, Andre Broido, and K.C. Claffy. Remote physical device fingerprinting. *IEEE Transactions on Dependable and Secure Computing*, 2(2), April-June 2005.

[90] Miika Komu and Thomas Henderson. Basic socket interface extensions for host identity protocol (hip) (draft-ietf-hip-native-api-05), November 2007. Work in progress. Expires May 22, 2008.

[91] Rajeev Koodli. RFC 4882: IP Address Location Privacy and Mobile IPv6: Problem Statement, May 2007.

[92] Teemu Koponen, Pasi Eronen, and Mikko Särelä. Resilient Connections for SSH and TLS. In *USENIX Annual Technical Conference*, May 2006.

[93] Hugo Krawczyk. SKEME: a versatile secure key exchange mechanism for Internet. In *Symposium on Network and Distributed System Security (NDSS)*, February 1996.

[94] Hugo Krawczyk, Mihir Bellare, and Ran Canetti. RFC 2104: HMAC: Keyed-Hashing for Message Authentication, February 1997.

[95] David M. Kristol. HTTP Cookies: Standards, privacy, and politics. *ACM Trans. Inter. Tech.*, 1(2):151–198, November 2001.

[96] Cynthia Kuo, Jesse Walker, and Adrian Perrig. Low-cost Manufacturing, Usability, and Security: An Analysis of Blue tooth Simple Pairing and Wi-Fi Protected Setup. In *Usable Security workshop (USEC'07)*, February 2007.

[97] Brian N. Levine, Michael K. Reiter, Chenxi Wang, and Matthew K. Wright. Timing attacks in low-latency mix-based systems. In *Proceedings of Financial Cryptography (FC '04)*, February 2004.

[98] Mingyan Li, Krishna Sampigethaya, Leping Huang, and Radha Poovendran. Swing & Swap: User-Centric Approaches Towards Maximizing Location Privacy. In *5th ACM workshop on Privacy in Electronic Society (WPES)*, October 2006.

[99] Nick Mathewson and Roger Dingledine. Practical traffic analysis: Extending and resisting statistical disclosure. In *Proceedings of Privacy Enhancing Technologies workshop (PET 2004)*, May 2004.

[100] Ralph C. Merkle. Secure communications over insecure channels. *Commun. ACM*, 21(4):294–299, 1978.

[101] MetaFacts. Busy mobile profile report (press release), September 2008. http://metafacts.wordpress.com/2008/09/23/mobile-computer-users-are-now-in-the-majority/.

[102] Refik Molva and Gene Tsudik. Secret sets and applications. *Information Processing Letters*, 65, 1998.

[103] Robert Moskowitz and Pekka Nikander. RFC 4423: Host Identity Protocol (HIP) Architecture, May 2006.

[104] Robert Moskowitz, Pekka Nikander, Petri Jokela, and Tom Henderson. RFC 5201: Host identity protocol, April 2008.

[105] Steven J. Murdoch and George Danezis. Low-Cost Traffic Analysis of Tor. In *IEEE Symposium on Security and Privacy*, May 2005.

[106] Thomas Narten and Richard Draves. RFC 3041: Privacy Extensions for Stateless Address Autoconfiguration in IPv6, January 2001.

[107] Thomas Narten, Richard Draves, and Suresh Krishnan. RFC 4941: Privacy Extensions for Stateless Address Autoconfiguration in IPv6, September 2007. Status: Draft Standard.

[108] Jasmine Novak, Prabhakar Raghavan, and Andrew Tomkins. Anti-aliasing on the Web. In *13th international conference on World Wide Web*, May 2004.

[109] National Institute of Standards and Technology (NIST). Advanced Encryption Standard (AES) (FIPS PUB 197), November 2001.

[110] Rafail Ostrovsky and William E. Skeith III. A survey of single database pir: Techniques and applications. Cryptology ePrint Archive, Report 2007/059, 2007. `http://eprint.iacr.org/`.

[111] Lasse Overlier and Paul Syverson. Locating hidden servers. In *IEEE Symposium on Security and Privacy*, May 2006.

[112] Jeffrey Pang, Ben Greenstein, Ramakrishna Gummadi, Srinivasan Seshan, and David Wetherall. 802.11 user fingerprinting. In *MobiCom'07*, September 2007.

[113] Jeffrey Pang, Ben Greenstein, Damon McCoy, Srinivasan Seshan, and David

Wetherall. Tryst: The Case for Confidential Service Discovery. In *Proc. of HotNets-VI*, November 2007.

[114] Andreas Pfitzmann and Marit Hansen. Anonymity, unlinkability, unobservability, pseudonymity, and identity management – a consolidated proposal for terminology, February 2008. Version v0.31 http://dud.inf.tu-dresden.de/Anon_Terminology.shtml.

[115] Jon Postel. RFC 791: Internet Protocol, September 1981. Status: Standard.

[116] Josyula R. Rao and Pankaj Rohatgi. Can Pseudonymity Really Guarantee Privacy? In *Proceedings of the 9th USENIX Security Symposium*, August 2000.

[117] Jean-François Raymond. Traffic analysis: protocols, attacks, design issues, and open problems. In *International workshop on Designing privacy enhancing technologies*, pages 10–29, New York, NY, USA, 2001. Springer-Verlag New York, Inc.

[118] Michael K. Reiter and Aviel D. Rubin. Crowds: Anonymity for Web Transactions. *ACM Transactions on Information and System Security*, 1(1):66–92, November 1998.

[119] Michael K. Reiter and Aviel D. Rubin. Anonymous Web Transactions with Crowds. *Communications of the ACM*, 42, February 1999.

[120] John W. Rittinghouse and James F. Ransome. *Wireless Operational Security*. Digital Press, March 2004. ISBN 155558-317-2.

[121] R. L. Rivest, A. Shamir, and L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Commun. ACM*, 21(2):120–126, 1978.

[122] Jonathan Rosenberg, Henning Schulzrinne, Gonzalo Camarillo, Alan Johnston, Jon Peterson, Robert Sparks, Mark Handley, and Eve Schooler. RFC 3261: SIP: Session Initiation Protocol, June 2002.

[123] Didier Samfat, Refik Molva, and N. Asokan. Untraceability in mobile networks. In *1st annual international conference on Mobile Computing and Networking (MobiCom)*, November 1995.

[124] T. Scott Saponas, Jonathan Lester, Carl Hartung, Sameer Agarwal, and Tadayoshi Kohno. Devices That Tell On You: Privacy Trends in Consumer Ubiquitous Computing. In *Proc. of USENIX Security*, August 2007.

[125] Andrei Serjantov and George Danezis. Towards an information theoretic metric for anonymity. In *Privacy Enhancing Technologies Workshop (PET 2002)*, April 2002.

[126] Dan Simon, Bernard Aboba, and Ryan Hurst. RFC 5216: The EAP-TLS Authentication Protocol, March 2008.

[127] Daniel J. Solove. A taxonomy of privacy. *University of Pennsylvania Law Review*, 154(3), January 2006.

[128] Dorothy Stanley, Jesse R. Walker, and Bernard Aboba. RFC 4017: Extensible Authentication Protocol (EAP) Method Requirements for Wireless LANs, March 2005.

[129] Richard Stanley. *Managing Risk in a Wireless Environment: Security, Audit and*

Control Issues. Information Systems Audit and Control Association, 2005. ISBN 1-893209-68-7.

[130] Adam Stubblefield, John Ioannidis, and Aviel D. Rubin. A key recovery attack on the 802.11b wired equivalent privacy protocol (WEP). *ACM Trans. Inf. Syst. Secur.*, 7(2):319–332, 2004.

[131] Latanya Sweeney. k-Anonymity: a model for protecting privacy. *International Journal on Uncertainty, Fuzziness and Knowledge-based Systems*, 10, 2002.

[132] Paul Syverson, Gene Tsudik, Michael Reed, and Carl Landwehr. Towards an analysis of onion routing security. In *International workshop on Designing privacy enhancing technologies*, pages 96–114, New York, NY, USA, 2001. Springer-Verlag New York, Inc.

[133] Paul F. Syverson, David M. Goldschlag, , and Michael G. Reed. Anonymous connections and onion routing. In *IEEE Symposium on Security and Privacy*, May 1997.

[134] Susan Thomson and Thomas Narten. RFC 1971: IPv6 Stateless Address Autoconfiguration, August 1996. Status: Obsolete.

[135] Susan Thomson, Thomas Narten, and Tatuya Jinmei. RFC 4862: IPv6 Stateless Address Autoconfiguration, September 2007. Status: Draft Standard.

[136] Vassilios S. Verykios, Elisa Bertino, Igor Nai Fovino, Loredana Parasiliti Provenza, Yucel Saygin, and Yannis Theodoridis. State-of-the-art in privacy preserving data mining. *SIGMOD Rec.*, 33(1):50–57, 2004.

[137] Samuel D. Warren and Louis D. Brandeis. The right to privacy. *Harvard Law Review*, 4, December 1890.

[138] Mark Weiser. Some computer science issues in ubiquitous computing. *Commun. ACM*, 36(7):75–84, 1993.

[139] Alan F. Westin. *Privacy and Freedom*. Atheneum, New York, 1967.

[140] Wi-Fi Alliance. http://www.wi-fi.org/.

[141] Wi-Fi Alliance. Wi-Fi Protected Setup Specification, Version 1.0h, December 2006.

[142] Edward Wilding. *Information Risk And Security: Preventing And Investigating Workplace Computer Crime*. Gower Publishing, 2006. ISBN 0-566-08685-9.

[143] Ford-Long Wong and Frank Stajano. Location Privacy in Bluetooth. In *Proc. of ESAS '05*, July 2005.

[144] Matthew Wright, Micah Adler, Brian N. Levine, and Clay Shields. An analysis of the degradation of anonymous protocols. In *Network and Distributed Security Symposium (NDSS)*, February 2002.

[145] Matthew Wright, Micah Adler, Brian N. Levine, and Clay Shields. Defending anonymous communication against passive logging attacks. In *IEEE Symposium on Security and Privacy*, May 2003.

[146] Matthew K. Wright, Micah Adler, Brian Neil Levine, and Clay Shields. The predecessor attack: An analysis of a threat to anonymous communications systems. *ACM Trans. Inf. Syst. Secur.*, 7(4):489–522, November 2004.

[147] Matthew K. Wright, Micah Adler, Brian Neil Levine, and Clay Shields. Passive-logging attacks against anonymous communications systems. *ACM Trans. Inf. Syst. Secur.*, 11(2):1–34, March 2008.

[148] Frances F. Yao and Yiqun Lisa Yin. Design and Analysis of Password-Based Key Derivation Functions. *IEEE Transactions on Information Theory*, 51:3292 – 3297, September 2005.

[149] Ge Zhong, Ian Goldberg, and Urs Hengartner. Louis, Lester and Pierre: Three Protocols for Location Privacy. In *7th Privacy Enhancing Technologies Symposium (PETS)*, June 2007.

[150] Alf Zugenmaier. FLASCHE - A Mechanism Providing Anonymity for Mobile Users. In *4th International Workshop on Privacy Enhancing Technologies (PET 2004)*, May 2004.

ISBN 978-951-22-9902-7
ISBN 978-951-22-9903-4 (PDF)
ISSN 1795-2239
ISSN 1795-4584 (PDF)