ANOCAST: Rethinking Broadcast Anonymity in the Case of Wireless Communication

André Adelsbach*
andre.adelsbach@telindus.lu

Ulrich Greveler†
greveler@fh-muenster.de

Stephan Groß‡
Stephan.Gross@tu-dresden.de

Sandra Steinbrecher‡
Sandra.Steinbrecher@tu-dresden.de

Abstract: In this work we present the ANOCAST environment being an anonymous communication framework based on wireless technology having reached prototype status. By exploiting the availability of current broadcast communication systems (e. g. digital satellites and Wi-Fi) we achieve recipient anonymity and unobservability. We examine our approach by a simulation approach using a prototype implementation, raising several questions concerning its scalability, efficiency and possible application scenarios as well as a discussion on its security accomplishments.

The key finding of our work shows that practicability of anonymous communication for the masses can be realised today and the community should rethink the common consensus that enormous traffic overhead jeopardises any practical anonymous communication system.

1 Introduction

Broadcast as anonymous communication system seems to be an old hat and was already studied more than 30 years ago [FL75]. Over the last decades, there was a broad consensus that broadcast-based anonymous communication was impractical, mainly because of the traffic overhead introduced by transmitting every message not only to the dedicated recipient but its whole anonymity set, ideally all parties participating in the system.

However, this argument of traffic overhead only holds for the traditional cable-based unicast network technology, where traffic can only be broadcast by initiating a unicast transmission to every receiver in the anonymity set. Today, new *wireless* communication technologies and infrastructures are established and in widespread use, replacing an increasing number of cable-based networks. On the data link layer most wireless communication media are broadcast media by nature and are transformed to unicast media by using hardware destination addresses and applying encryption on higher layers of the network stack.

*Telindus S.A, Security, Audit and Governance Services (SAGS), L-4244 Esch/Alzette, Luxembourg
†Fachhochschule Münster, Fachbereich Elektrotechnik und Informatik, Steinfurt, Germany
‡Technische Universität Dresden, Institute for System Architecture, Dresden, Germany

Given the widespread availability of broadcast communication systems, it is time to rethink the practicality of broadcast for anonymous communication, especially, because broadcasts not only achieve recipient anonymity but also recipient unobservability.

Today, wireless communication has become an indispensable part of the mobile computing world we are living in. We are used to an all-embracing network access, all the time and wherever we are. For example, within the last five years the number of public accessible wireless LAN hotspots has been constantly growing. Even in trains and planes such a service is provided and carriers like Fon[1] are developing new community-based business models to increase the availability of WLAN Internet access while simultaneously lowering the costs for the end user.

All radio communication technologies share the broadcast nature of their data link layer. However, in practice this feature is often shattered by cryptographic means on the network layers above. For example, GSM as well as UMTS networks both use an individual encryption between base station and mobile station to prevent eavesdropping of someone's communication [MP92, Mit94]. To restore the broadcast feature one would have to extract the applied keys from the lower network layers and subsequently reverse the encryption on the application layer. Although possible, this method is rather clumsy.

WLAN networks, on the other hand, are more promising as they leave the control of encryption mechanisms to the user. Most commercial WLAN hotspots are even run without any encryption enabled. The currently evolving WiMAX standard will most likely overcome the major drawback of wireless LAN concerning our application scenario, i.e. its small footprint, as it introduces a range of coverage up to 50km, thus, allowing huge recipient anonymity sets.

Alternatively, one can use the large recipient sets provided by satellite communications. A satellite is a specialised wireless transmitter placed in terrestrial orbit for diverse purposes such as weather forecasting, television broadcast, radio communications, Internet access and GPS positioning. Satellites can receive and re-transmit thousands of signals simultaneously, from simple digital data to television programmes. Especially, in low-infrastructure areas they provide an interesting alternative, e.g., for high-speed access to the Internet, because they provide high data rates and cover very large areas with comparably low efforts.

The data packets in the downstream are broadcast which makes it easy to receive the data of all users, not only the data packets addressing a specific user. Every person owning a DVB-S card and a digital enabled satellite dish is able to intercept all the data packets sent by the satellite. There are publicly available tools to watch data stream information in human readable form [Lic]. Moreover, interception of unsecured satellite signals for intelligence purposes is on the public agenda since the nineteen-nineties [Cam99].

Due to the fact that satellite signals can be received in a huge footprint, the recipient anonymity set contains millions of users.[2] Furthermore, receivers of satellite signals are completely passive and, in contrast to cable-based Internet access, they cannot be identified and traced by a central authority (the ISP).

[1] http://www.fon.com

[2] As an example consider the Astra 19.2 satellites, whose footprints cover central Europe. Alone in Germany, there are several millions of households with a satellite dish and, therefore, receivers within the anonymity set.

In this paper we present the ANOCAST environment being an anonymous communication framework based on wireless technology. The project has reached prototype status and we can demonstrate practical ways of implementing the "old, impractical" paradigm of broadcast-based recipient anonymity/unobservability on top of existing communication technologies. We show that broadcast-based anonymous communication can be efficiently implemented and easily used. We provide details of our findings with the concrete examples of wireless LAN and satellite-based broadband ISPs, both being part of the ANOCAST prototype implementation. Furthermore, we examine our approach by drawing a comparison to alternative techniques, raising several questions concerning its scalability, efficiency and possible application scenarios.

The remainder of our paper is organised as follows. After summarising the state of the art in anonymous communication (section 2) we describe how to utilise the broadcast nature of wireless communication to implement unconditionally strong recipient anonymity in section 3. Concretely, we analyse the case for satellite based ISPs and Wi-Fi-based Internet communication in sections 3.1 and 3.2, respectively. A detailed overview of our prototype implementation is given in section 3.3. In section 4 we scrutinise our results by discussing open issues as well as the achieved anonymity. We conclude our work by summarising the achieved results and our future plans.

2 Anonymous Communication

2.1 Anonymity, unlinkability and unobservability

Being anonymous regarding a certain action means being 'not identifiable within a set of subjects, the anonymity set' who might have caused the action as well [KP04]. In communication systems these actions are sending and receiving of messages. Sender and/or recipient anonymity are 'the properties that a particular message is not linkable to any sender/recipient and that to a particular sender/recipient, no message is linkable' [KP04]. Both the size of the respective anonymity set as the probability distribution within the set determine the respective sender/recipient's anonymity. In open environments like the Internet a user is a member of the anonymity set if the probability that he initiated the action is non-zero [KEB98], but this set is often difficult to determine.

Being unobservable regarding a certain action means 'being indistinguishable from any item of interest (of the same type) at all'. In communication systems this means messages 'are not discernible from e.g. 'random noise' [KP04]. Sender/recipient unobservability then means that it is not noticeable whether any sender/recipient within the unobservability set sends/receives a message.

Users cannot have the same anonymity resp. unobservability against every possible participant and outsider who might be seen as attacker. One has to distinguish between passive attackers who only observe (parts of) the system and active attackers who are able to control parts of the network to achieve his goal of breaking anonymity resp. unobservability of arbitrary or even chosen users.

Depending on the attacker's knowledge gained during his attack the above set of possible subjects and the likelihood with which they have sent/received a message varies. His success regarding a sender/recipient's identifiability/observability with respect to a particular message can be measured with information theoretical measures. It usually is by measuring the difference between an attacker's a-priori and his a-posteriori knowledge after the attack (e.g. anonymity with the respecting entropies [SD02, DSCP02, THV04, CS06]).

In contrast to anonymity and unobservability (un)linkability is not restricted to actors and their actions, actions also might be linkable to each other or not. In communication systems the linkability of messages might endanger a recipient's or sender's anonymity. One specific message might be unlinkable to him but a set of linkable messages only might be his and so each single message becomes linkable to him. An attacker's success regarding the unlinkability of messages 'means that within this system, these [messages] are no more and no less related than they are related concerning the a priori knowledge' [KP04].

A communication system provides its senders and recipients information theoretical unobservability if it is not noticeable during sending/receiving whether any recipient within the unobservability set sends/receives a message even with unlimited computing power.

Communication systems can be categorised in systems that offer *one-way* or *two-way communication*, that means if the roles of sender and recipient are fixed or not; Especially this means if it is only possible to send or receive or if it is possible to get an answer to a message sent or to answer a message received by the communication system. This has an effect on the anonymity properties, whether both sender and recipient anonymity are achievable by two-way communication if the roles are changed.

2.2 Anonymising services

Numerous systems have been proposed that offer services for anonymous communication on the Internet providing different degrees of security and protection against several attacker types.

Simple anonymising proxies (e.g., Anonymizer[3]) achieve anonymity against outsiders only, but not against their providers. This is, because these proxies simply collect messages from senders, relabel them with their sender address and send them to the respective recipient.

More secure services like Crowds [RR98] involve a set of users, the Crowd, that guarantee a user's anonymity by forwarding a user's message through a path within this Crowd. If a member of the Crowd wants to send a message he chooses another member and encrypts and redirects his message to him instead of sending it directly to the recipient. The member who receives a redirected message for which he is not the recipient with the same probability either encrypts and redirects this message again to a randomly chosen member of the Crowd, or directly sends it to the final, intended recipient. The system thereby tries to achieve sender anonymity, but correlation regarding the times at which messages are

[3] http://www.anonymizer.com/

sent or received is still possible even if the messages themselves are encrypted between two members of the Crowds. This system does not protect against an outside attacker who monitors the whole network and does not protect a member whose in- and outgoing links are all observed.

By using mixes anonymisation of messages can be reached by sending them not directly from the sender to the recipient but through a chain of intermediate mix servers. If every single mix is run by an independent provider this system guarantees security against a stronger attacker who might control all but one of the mixes a particular message and some other messages are routed through. A user has to prepare messages for the mix servers by encrypting them multiple times like an onion and then sending them to the chosen path of mixes. Each mix collects the messages sent during a certain time frame by several senders, changes their appearance and sequence by decrypting them once, and outputs them for the next mix or the web server requested. Thereby the message will be decrypted while travelling through the mixes. The output message that the last mix sends to the recipient has the original format the sender intended to send if he would not have used the mixes. The mix principle was theoretically invented for anonymous e-mail 25 years ago [Cha81]. The concept of anonymisation by multiple encryption like an onion was re-invented as Onion Routing [RSG98]. There exist some implementations for WWW usage, e.g. Web Mixes [BFK01] or Tor [DMS04].

2.3 Broadcast

Perfect information-theoretically secure recipient anonymity and even unobservability can be achieved if broadcast is used. This means that every message is sent to any possible receiver and, thereby, the recipient's anonymity set comprises any possible receiver [FL75]. Due to its inherent traffic overhead, when implemented in traditional, cable-based unicast networks, broadcasts were only of minor practical relevance in anonymising services for the last 30 years.

However, in the meantime, new wireless communication technologies have emerged and are already in widespread use, which, on the data link layer, are broadcast networks by nature. As these broadcast technologies are commonly transformed to unicast networks by applying encryption on higher layers of the network stack, their value for implementing anonymity services has not been recognised so far. But we will argue in the following that such broadcast media can be adapted quite easily to achieve information-theoretic recipient unobservability without an additional anonymising service. This even holds, if encryption is applied to enforce unicast communication on these networks, because the employed cryptographic protocols have not been developed with security against insiders in mind [AG07]. Therefore, it is quite easy for an insider to bypass the encryption, e.g., by publishing session keys on electronic blackboards or by applying an inverse operation, which annihilates encryption.

Addressing a specific recipient within the set of possible receivers can be done with an implicit address. In contrast to explicit addresses, needed for routing, an implicit address does

not identify the technical location of the recipient but only is an identifier for his device. Addressing with implicit addresses can be done either open or covered: For open addressing the message sent only needs to contain an address field which content every receiver can compare with the addresses he holds. Unfortunately this makes all messages addressing him with the same address linkable to each other and might endanger his anonymity. He either needs to offer one-time addresses to possible senders or covered addressing has to be used that makes the addresses used only interpretable by him.

The usual implementation of covered implicit addressing is done by embedding some redundancy in the message and then encrypting the message completely or partially with the encryption key of the recipient who should be addressed. Every receiver has to decrypt every message he receives. By checking the redundancy he is able to decide if he is the recipient of this message.

Usually a symmetric encryption scheme is needed because asymmetric ones take too much effort especially because every receiver has to decrypt every message he receives [Ken81].

But if a recipient uses more than one covered implicit address he has to decrypt all messages with all his decryption keys. Usually the number of addresses when using symmetric encryption will be higher than for asymmetric encryption because for every sender a separate key is needed. This makes asymmetric encryption sensible at least for the establishment of communication relationships.

Address management is an important issue for recipient anonymity in broadcast technology. To offer people the possibility to contact a recipient for the first time he might offer a public address (like an entry in the 'white pages'). Offering a public address does not exclude to be anonymous, because this address is not necessarily linkable to any identifying attribute like the recipient's real name. Typically, public encryption keys will be associated to public addresses.

But one might establish private addresses in the communication with single communication partners. If addressing with these addresses is done covered implicitly this guarantees the unlinkability of messages sent to the same recipient.

3 Utilising the Broadcast Property of Wireless Communication

In the previous section we have already explained why one should reconsider using broadcasts to implement recipient anonymity. We are now going to put our idea in more concrete terms by analysing two common wireless communication techniques: satellite-based and Wi-Fi-based Internet communication, respectively. We describe how a simple broadcast-based anonymising service can be implemented for both services utilising familiar tools, namely a Web-Server, some Web-Client and a standard protocol analyser. However, to simplify the whole process we have implemented a prototype application named ANOCAST that encapsulates all the necessary functionality and will be explained in section 3.3.

3.1 Satellite-based Internet Communication

How it works Satellite based ISPs provide a wireless alternative for high-speed Internet access. This alternative is especially relevant for low-infrastructure areas since it provides high data rates and covers very large footprints with low effort compared to cable-based communication. Satellite based ISPs come in two flavours:

- **One-Way:** In this lower cost variant, the satellite only handles the data downstream to the user with outbound data travelling through a telephone modem taking care of the low-bandwidth traffic from the user to the ISP. Most private customers only desire a high download bandwidth while they accept a rather small uplink capacity so this hybrid solution satisfies their needs (e g. surfing the web).
- **Two-Way:** The more expensive two-way option lets the user have a satellite transmitter entity at their site that enables two-way communication with high bandwidth for up-link and down-link.[4] This option is more suitable for companies connecting their remote branches to a data network than for private households.

In our prototype realisation for the lab test environment we focus on the one-way variant, because it is more common for today's standard users – and the tool could easily be adopted by all users being provided with a digital LNB satellite dish.[5] To illustrate how one-way satellite-based ISPs operate, consider the setting, where a user fetches a file from a web server as depicted in figure 1.

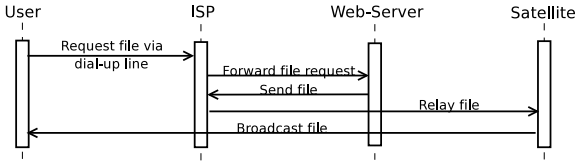

Figure 1: One-way satellite-based Internet communication

The user establishes a (usually small bandwidth) dial-up Internet connection, e. g., an ISDN line, to the ISP. [6] In order to initiate a download, a request is sent through the dial-up line to an ISP proxy server, which relays the request to the desired destination server. The reply coming from this server (e.g. the requested file) is re-routed by the satellite ISP so that it is sent wireless to the user – as well as to all other users in the satellite footprint. The requested file is encapsulated together with the user's specific IP into DVB packets. A piece of software on the user's PC completes the TCP communication transparently to the application or operating system.

Let us stress the pivotal property we will exploit for our purposes: Due to the broadcast character of satellite communication, all signals dedicated for one user can be received by

[4] Note, that the up-link bandwidth is commonly still smaller than the down-link bandwidth.

[5] However, we want to stress that our proposal is even more suitable for two-way satellite communication, as the ISP has to broadcast all packets via the satellite down-link.

[6] A TCP-connection to the ISP can be used alternatively if available

all users in the footprint of the satellite while there is no way to monitor who actually does listen to the signal.

Making it anonymous The general idea is to exploit the fact that the satellite downstream, containing the data requested by the user, can be received in the whole footprint of the satellite. Figure 2 visualizes the general structure of our approach. To broadcast certain data, e.g. a file, the sender first puts it on a dedicated server, which is connected to the Internet (*step 1*). Then the sender requests this data over the satellite ISP (*step 2*), which results in the data being broadcast by the satellite. All receivers simply listen to the satellite broadcast and filter the data from the DVB network interface, e.g. by implicit addresses. This can be done by any standard packet capture tool such as *libpcap*. The captured packets are temporarily stored until the transmission ends to reconstruct the original file afterwards. Obviously, this system achieves unconditionally strong recipient anonymity due to the nature of a broadcast channel and the fact that the recipients are completely passive. Furthermore, the anonymity set is very large, providing strong anonymity: every household with a satellite dish may be a potential receiver.

Our approach works immediately if the satellite ISPs do not encrypt the data broadcast. If the satellite ISP encrypts the satellite downstream using individual keys for each user, the approach needs to be modified: in this case the sender has to *publish* his session key, such that it is accessible anonymously by all receivers and enables receivers to decrypt the user's part of the satellite downstream. Fortunately, this effort is not necessary, as long as there are satellite ISPs which do not enforce encryption of the satellite downstream (see e.g., [AG05]).

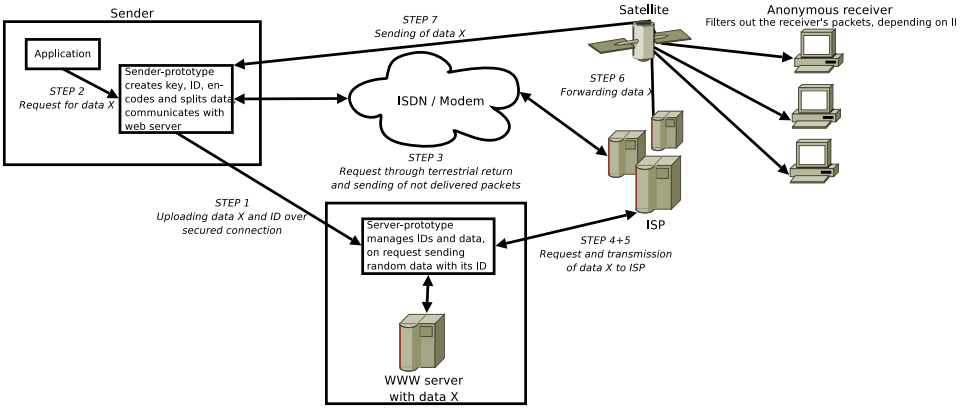

Figure 2: Anonymous data broadcasts via satellite

3.2 Wi-Fi-based Internet Communication

How it works To validate our approach in a two-way communication variant we will now describe its application in a wireless LAN setup. Today, so-called hotspots at public places like bars, restaurants or hotels are a common alternative for easy high-speed Internet access as almost every mobile device supports the deployed IEEE 802.11 standard. A WLAN hotspot operator provides shared access to a fast backbone connection via a WLAN access point. The major drawback compared to the just described satellite-based communication is the smaller footprint of a WLAN hotspot, typically between 30 and 300 metres. However, new arising technologies like IEEE 802.16 also known as WiMAX [IEE04] will offer a broader range of up to 50 kilometres. Although security mechanisms are an integrated part of the wireless LAN standards they have to be deactivated at public hotspots to provide service for everyone . Thus, every signal can be received by anyone in the footprint of the access point and confidentiality must be gained by additional measures on higher layers.

Making it anonymous The general idea of anonymization stays the same as with satellite-based Internet communication. However, we are using a two-way communication between sender and WLAN access point. Our system again works directly if the access point does not use encryption, e.g. like public hotspots do. If encryption is enabled the sender has to extract his individual session key and publish it, such that it is anonymously accessible by potential receivers to enable them to decrypt the transmission. The customized setup for WLAN is depicted in figure 3. Please note, that requesting (*step 3*) and sending (*step 7*) specific data is now handled using the same data link. Furthermore, due to the higher upload bandwith of two-way WLAN communication there is no need for an external web service provider. Instead, the sender runs a web server himself.

3.3 ANOCAST Implementation Details

Having discussed the feasibility of anonymous broadcasts for satellite and WLAN communication in principle we are now going to present some more details of our prototype implementation. In our system we basically distinguish two roles: the *sender* who publishes some data and the *receiver* who wants to receive this data anonymously. The two of them have to interact to achieve this goal. Please note that our prototype does not support the exchange of an encryption key yet. For more details on this please refer to section 4.

First of all, the data to be received anonymously by the receiver has to be made available by the *sender* by publishing it on a public web server. A standard GUI interface enables the user to upload a file to an appropriate web server. Optionally, it is possible to protect the file against unwanted eavesdropping by encrypting it using AES. In the next step, the sender has to initiate the anonymous download for the receiver side by sending a standard HTTP request for the just uploaded file. In turn the file will be broadcast by the ISP and all stations in the satellite's/hotspot's footprint are able to receive it.

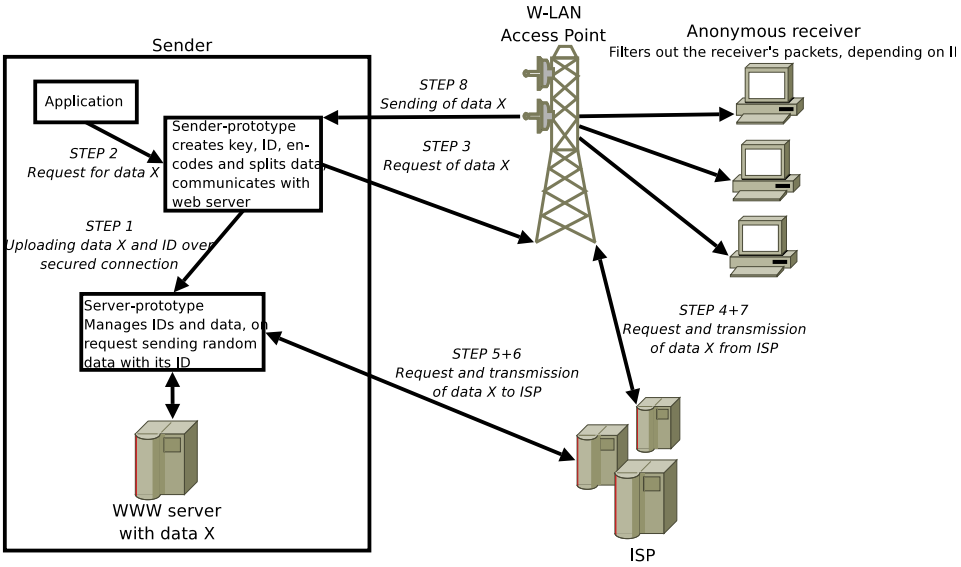

Figure 3: Anonymous data broadcasts via WLAN

The *receiver* part records all traffic transmitted. By some specific implicit address scheme only known to him and the sender he is able to filter the relevant parts out of the whole bunch of network traffic and can finally reconstruct the original data packets.

System Requirements Our prototype implementation is realized using Java version 1.5 and the following additional tools and libraries:

- The *Bouncy Castle API* [The] is a JCE[7] compatible library. It offers TLS-protection for the data publishing by the sender. Additionally, the also provided AES implementation can be used to protect the published data against eavesdropping.
- The packet capturing functionality at the receiver's side is based on the well-known *libpcap* library [tcp]. It utilizes the promiscuous mode of common ethernet devices and the so called monitor mode on WLAN NICs to record any network traffic within range. The C-library is embedded into Java via the *Jpcap* packet [Fuj] utilizing the Java Native Interface (JNI).
- Last but not least, we use Jigsaw [W3C], the W3C's official web server implementation. Its features include TLS secured transmissions and the ability to easily modify HTTP headers.

In our laboratory test environment we used several standard off-the-shelf hardware. The web server was implemented on a standard PC with Pentium III processor and 1 GB RAM

[7] Java Cryptographic Extension

running a Debian Linux. Both *sender* and *receiver* part were realized on standard PCs with a Pentium III and 512 MB RAM running Debian Linux. For network connectivity we use a Hauppage Win-TV-Nova DVB-S card and an IEEE 802.11b/g compatible NIC based on a Prism chipset (Allnet ALL0281A), respectively. Furtheron, we used a standard SAT-ISP access by Deutsche Telekom and a Netgear FWAG114 WLAN router to build up an exemplary WLAN hotspot.

4 Discussion

Shortcomings of our Implementation Although we have demonstrated, that our approach can easily be applied to both one-way and two-way communication, there are several open issues to be addressed in future work. A major challenge one has to deal with is the extraction of used encryption keys from lower network layers and their anonymous publication for potential recipients. This becomes especially problematic when you consider for example the dynamic key exchange protocols introduced in recent wireless LAN security protocols like WPA or WPA2. However, most of the public WLAN hotspots do not use any security mechanisms at all and by patching the NIC's hardware driver on the sender side it should be possible to log all used keys for later exchange with the recipient. Alternatively, a sender could deploy its own hotspot with no encryption enabled which is already supported by several WLAN NICs.

Furthermore, our prototype currently neglects potential problems caused by lost or flawed packages. During "regular" communications the recipient initiates the retransmission of a package in case of such failures. This is of course impossible for an anonymous recipient who neither maintains a direct connection with the ISP nor is the true recipient of the data transmitted. To overcome this issue the sender could initiate multiple broadcasts of the same data chunk, thus lowering the possibility of loosing it. A more sophisticated solution would be to utilize some error correcting code like Reed-Solomon codes [RS60]. However, we postponed a more detailed insight in this aspect for future research.

Another possible direction for further developments might be the deployment of self-organizing wireless mesh networks [AWW05] as proposed by the upcoming IEEE 802.11s standard integrating conventional communication with peer-to-peer techniques. For example, a packet is received by a WLAN hotspot via its DSL backbone access and broadcast to all stations in range. They in turn pass it to other stations within an existing mesh network from where the packet might also drain away through another backbone gateway. As the communication links in a mesh network are much more dynamic the surveillance of the resulting data flows will become extremely difficult if not impossible.

Efficiency and Security Analysis While unconditionally strong receiver anonymity follows trivially by the nature of a broadcast channel, achieving *sender anonymity* is more involved and requires a more advanced system design: one idea is to run a common server, where potential senders upload their encrypted data packets via some traditional point-to-point anonymizer. Now, instead of requesting its own packets, each sender requests

random packets from the server, i. e., the party requesting a certain packet and, thereby initiating its broadcast, is different from the originator of this packet. This guarantees that nobody – not even the server – can tell who is the originator of a specific packet.[8] We consider this issue to be an important and challenging strand of future work.

Furtheron, we are currently planning intensive field tests to estimate both scalability and practicability of our approach in a real-life scenario. We expect this to help us with a sound evaluation of our approach's performance. We are especially interested in the question how well our approach performs in specific application scenarios compared to competitive anonymizers. For instance we assume our approach to be less suitable for web surfing than for anonymous file sharing and hope to prove or contradict this assumption by the planned field tests.

5 Conclusion

We have presented ANOCAST, the prototype of an anonymous communication framework based on wireless technology. It utilizes current broadcast communication systems used by satellite ISPs and Wi-Fi hotspot providers to realize both recipient anonymity and communication unobservability. We have demonstrated the feasibility and practicability of our approach by a prototype implementation based on open source tools. Thus, we believe it is time to think over the common consensus that broadcast anonymity is of no practical use. In future work we plan to substantiate our demand by comparing our approach to competitive anonymizers based on a comprehensive field test.

References

[AG05] André Adelsbach and Ulrich Greveler. Satellite Communication without Privacy – Attacker's Paradise. In Hannes Federrath, editor, *Sicherheit*, volume 62 of *LNI*. GI, 2005.

[AG07] André Adelsbach and Ulrich Greveler. Insider Attacks Enabling Data Broadcasting on Crypto-Enforced Unicast Links. volume 4734 of *LNCS*, pages 469–484, Dresden, Germany, September 2007. Springer.

[AWW05] Ian F. Akyildiz, Xudong Wang, and Weilin Wang. Wireless mesh networks: a survey. *Computer Networks*, 47:445–487, 2005.

[BFK01] Oliver Berthold, Hannes Federrath, and Stefan Köpsell. Web MIXes: A system for anonymous and unobservable Internet access. In *Designing Privacy Enhancing Technologies. Proc. Workshop on Design Issues in Anonymity and Unobservability*, volume 2009 of *LNCS*, pages 115–129. Springer Verlag, 2001.

[Cam99] Duncan Campbell. Interception Capabilities 2000. Report to the Director General for Research PE 168.184, European Parliament, 1999.

[8] Obviously, the sender anonymity set only consists of those ISP customers requesting packets from this server and is significantly smaller than the receiver anonymity set.

[Cha81] David Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Communications of the ACM*, 4(2):84–88, February 1981.

[CS06] Sebastian Clauß and Stefan Schiffner. Structuring Anonymity Metrics. In Atsuhiro Goto, editor, *DIM '06, Proceedings of the 2006 ACM Workshop on Digital Identity Management*, pages 55–62, Fairfax, Virgina, USA, November 2006. ACM.

[DMS04] Roger Dingledine, Nick Mathewson, and Paul Syverson. Tor: The Second-Generation Onion Router. In *Proceedings of the 13th USENIX Security Symposium*, August 2004.

[DSCP02] Claudia Díaz, Stefaan Seys, Joris Claessens, and Bart Preneel. Towards measuring anonymity. In Roger Dingledine and Paul Syverson, editors, *Proceedings of Privacy Enhancing Technologies Workshop (PET 2002)*. Springer-Verlag, LNCS 2482, April 2002.

[FL75] David J. Farber and Kenneth C. Larson. Network Security Via Dynamic Process Renaming. In *Fourth Data Communications Symposium*, pages 8–18, Quebec City, Canada, October 1975.

[Fuj] Keita Fujii. Jpcap: Java package for packet capture. Project web site. http://netresearch.ics.uci.edu/kfujii/jpcap/doc/.

[IEE04] IEEE. Std 802.16, Part 16: Air Interface for Fixed Broadband Wireless Access Systems. Technical report, Institute of Electrical and Electronics Engineers, Inc, New York, USA, 2004.

[KEB98] Dogan Kesdogan, Jan Egner, and Roland Büschkes. Stop-and-Go-MIXes Providing Probabilistic Anonymity in an Open System. In *Information Hiding*, volume 1525 of *LNCS*, pages 83–98. Springer Verlag, 1998.

[Ken81] Stephen T. Kent. Security Requirements and Protocols for a Broadcast Scenario. In *IEEE Transactions on Communications 29/6*, pages 778–786, 1981.

[KP04] Marit Köhntopp and Andreas Pfitzmann. Anonymity, Unobservability, and Pseudonymity - A Proposal for Terminology. Draft v0.18., July 2004, http://dud.inf.tu-dresden.de/Literatur_V1.shtml, 2004.

[Lic] GNU General Public License. DVBSNOOP: a DVB / MPEG stream analyzer program. http://dvbsnoop.sourceforge.net.

[Mit94] Hakan Mitts. *Broadland Islands '94, Connecting with the End-User*, chapter Universal Mobile Telecommunication Systems – Mobile access to Broadband ISDN, pages 203–209. 1994.

[MP92] Michel Mouly and Marie-Bernadette Pautet. *The GSM System for Mobile Communications. A comprehensive overview of the European Digital Cellular Systems*. Published by the authors, 1992. ISBN 2-9507190-0-7.

[RR98] Michael Reiter and Aviel Rubin. Crowds: Anonymity for Web Transactions. *ACM Transactions on Information and System Security (TISSEC)*, 1(1):66–92, 1998.

[RS60] I. Reed and G. Solomon. Polynomial codes over certain finite fields. *Journal of the Society for Industrial and Applied Mathematics*, 8:300–304, 1960.

[RSG98] Michael G. Reed, Paul F. Syverson, and David M. Goldschlag. Anonymous Connections and Onion Routing. *IEEE Journal on Selected Areas in Communications*, 16(4), May 1998. Special Issue on Copyright and Privacy Protection.

[SD02] Andrei Serjantov and George Danezis. Towards an Information-Theoretic Metric for Anonymity. In *Privacy Enhancing Technologies 2002*, volume 2482 of *LNCS*. Springer-Verlag, 2002.

[tcp] tcpdump.org. tcpdump/libpcap project web site. http://www.tcpdump.org/.

[The] The Legion of the Bouncy Castle. Bouncy Castle Crypto APIs. http://www.bouncycastle.org/.

[THV04] Gergely Tóth, Zoltán Hornák, and Ferenc Vajda. Measuring Anonymity Revisited. In Sanna Liimatainen and Teemupekka Virtanen, editors, *Proceedings of the Ninth Nordic Workshop on Secure IT Systems*, pages 85–90, Espoo, Finland, November 2004.

[W3C] W3C. Jigsaw - W3C's Server. http://www.w3.org/Jigsaw/.