A Model-based Analysis of Tunability in Privacy Services

Reine Lundin[1], Stefan Lindskog[1,2], and Anna Brunstrom[1]

[1] Department of Computer Science
Karlstad University
Karlstad, Sweden
reine.lundin|anna.brunstrom@kau.se
[2] Centre for Quantifiable Quality of Service in Communication Systems
Norwegian University of Science and Technology
Trondheim, Norway
stefan.lindskog@q2s.ntnu.no

Abstract. In this paper, we investigate the tunable privacy features provided by Internet Explorer version 6 (IE6), Mix Net and Crowds, by using a conceptual model for tunable security services. A tunable security service is defined as a service that has been explicitly designed to offer various security configurations that can be selected at run-time. Normally, Mix Net and Crowds are considered to be static anonymity services, since they were not explicitly designed to provide tunability. However, as discussed in this paper, they both contain dynamic elements that can be used to utilize the trade-off between anonymity and performance. IE6, on the other hand, was indeed designed to allow end users to tune the level of privacy when browsing the Internet.

1 Introduction

Many security services today only provide one security configuration at run-time, making it impossible to utilize the trade-off between performance and security, when user demands and/or the environment changes. Furthermore, the security configuration is often set by default, during setup or installation, i.e., before run-time, to achieve a high level of security, which may affect the performance of the system negatively. According to, for example, Pfleeger and Pfleeger [14], security services should be operating according to the principle of adequate security, which states that computer items must be protected only until they lose their value, and they must be protected to a degree consistent with their value. Hence, in situations where we want to make use of the trade-off between security and performance, tunable security services are needed. In this paper, a tunable security service is defined as a security service that has been explicitly designed to offer various security configurations that can be selected at run-time.

One important component of security is privacy, which by Westin [17] is defined as:

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

Note, however, that the European view of privacy is slightly different than Westin's definition. In most European contries privacy protection is regarded as a basic right, not as a claim, and only individuals can have the right to privacy, not groups and institutions. A well-known aspect of privacy is anonymity. In [13] anonymity is defined as:

> "Anonymity of a subject from an attacker's perspective means that the attacker cannot sufficiently identify the subject within a set of subjects, the anonymity set."

Hence, anonymity ensures that a user may use a resource without disclosing his or her identity.

A service that considers tunable privacy aspects is the web browser Internet Explorer version 6 (IE6) that is equipped with two mechanisms, cookie blocker and pop-up blocker, to safeguard the end users' spatial privacy [3] when browsing the web. Furthermore, to provide anonymity when browsing the web, the two services Mix Net, a network of mixes [1], and Crowds [16] could be used. The major difference between these two is that Mix Net provides anonymity by hiding the relation between incoming and outgoing messages for each mix, while Crowds provides anonymity by hiding one user's actions within the actions of many others. Even though Mix Net and Crowds were not explicitly designed as tunable anonymity services, they both contain dynamic elements that can be used to utilize the trade-off between anonymity and performance.

In this paper, IE6, Mix Net, and Crowds are analyzed using a conceptual model for tunable security services. The model was first proposed in [10], and it describes in a formal way the requirements for tunable security services. Thus, it provides a suitable tool to examine the tunability of privacy services.

The rest of the paper is organized as follows. In Section 2, the conceptual model for tunable security services used in the analysis is presented. Using this model, Sections 3–5 investigate IE6, Mix Net, and Crowds, respectively. Section 6 provides a discussion on tunable privacy and future work. Finally, Section 7 concludes the paper.

2 Conceptual Model

The conceptual model for tunable security services is described by the three sets:

- $T = \{\text{Tuner preferences}\}$
- $E = \{\text{Environmental descriptors}\}$
- $S = \{\text{Security configurations}\}$

and the function

$$TS : T \times E \rightarrow S \tag{1}$$

The TS function represents the mapping from tuner preferences, T, and environmental descriptors, E, to a particular security configuration, S. Hence, the TS mapping gives under which conditions the security configuration should be changed for the service. For example, when a device reaches a threshold in battery level the TS function could give that the security configuration of the device should be changed to increase the remaining time of the battery. Note that, for a security service to be a tunable security service S must contain at least two security configurations, otherwise the service will be static. The same will happen if both T and E are singular sets, since then $T \times E$ is a singular set as well.

Through the elements in T, the tuner preferences, a tuner entity can affect the security configurations in order to achieve desired trade-offs between security and performance. The tuner entities that set the tuner preferences of the security services typically exist on several layers, or phases of the system life cycle, such as system owner and/or end user. For example, a system owner might assign some tuner preferences for the provided service so that it fulfills the security policy of the company, while the end users in the same company are free to affect the rest of the preferences. The elements in T can be expressed at various abstraction levels, for example as low, medium, or high security, or by specifying frames or layers to encrypt in MPEG video streams [6, 11]. T might also be constructed from several parameters, each representing a different security objective such as confidentiality and integrity. In E, the environment and application descriptors that may influence the selection of security configurations are described. Possible elements in E include characteristics of equipment, threat model, energy consumption, and network load [4, 5, 9, 15]. The elements in S represent the possible security configurations of the tunable security service, such as encryption algorithm, MAC algorithm, key length(s), and key establishment algorithm.

In previous work, the above described conceptual model has successfully been used to examine the tunable features provided by seven different security services. Four services were analyzed in [10], the paper that introduced the model, and three additional services were evaluated in [8]. In this paper, we first apply the conceptual model when investigating the tunable privacy features provided within IE6. Then, we also apply the conceptual model when analyzing the two anonymity services, Mix Net and Crowds. Furthermore, since we only consider the privacy aspects of security, the term privacy configuration is used instead of security configuration in the rest of the paper.

3 Analyzing IE6

The first service to analyze is IE6, which is a web browser that is equipped with mechanisms to safeguard end users' privacy on the web[3]. The reason for describing IE6 as the first service is motivated by the fact that it is so commonly used today, and is therefore also quite well-known.

[3] See `http://msdn2.microsoft.com/en-us/library/ms537343.aspx` for a detailed discussion on privacy in IE6.

In IE6, two different blockers are provided to preserve end users' spatial privacy. One mechanism is referred to as the cookie blocker, and the other is referred to as the pop-up blocker. A cookie is a small file stored on the local computer and it is used as an identifier. Cookies are created by web sites to store information gathered about your visits to sites including, where you went, what you did, and any personal information you provided. Web sites may also have embedded links to other domains which set cookies. The latter type of cookies is known as third-party cookies. Cookies in relation with privacy is further described in [7]. The cookie blocker controlls how the browser handles cookies, and the pop-up blocker is aimed to prevent pop-up windows from appearing on the end users' screen.

3.1 Privacy Configurations (S)

The available privacy configurations in IE6 are based on the set of possible configurations of the cookie blocker, denoted CB, and the set of possible configurations of the pop-up blocker, denoted PB. Thus, the privacy configurations of IE6 can be expressed as: $S = CB \times PB$. With respect to cookie handling, six different privacy configurations are provided, ranging from "Block All Cookies" to "Accept All Cookies". The full set of cookie blocker options are as follows: $CB = \{BAC, Hi, MH, Me, Lo, AAC\}$, where the different options are abbreviations of "Block All Cookies", "High", "Medium High", "Medium", "Low", and "Accept All Cookies", respectively. The pop-up blocker feature, on the other hand, is simpler and could either be turned on or off. Thus, $PB = \{yes, no\}$.

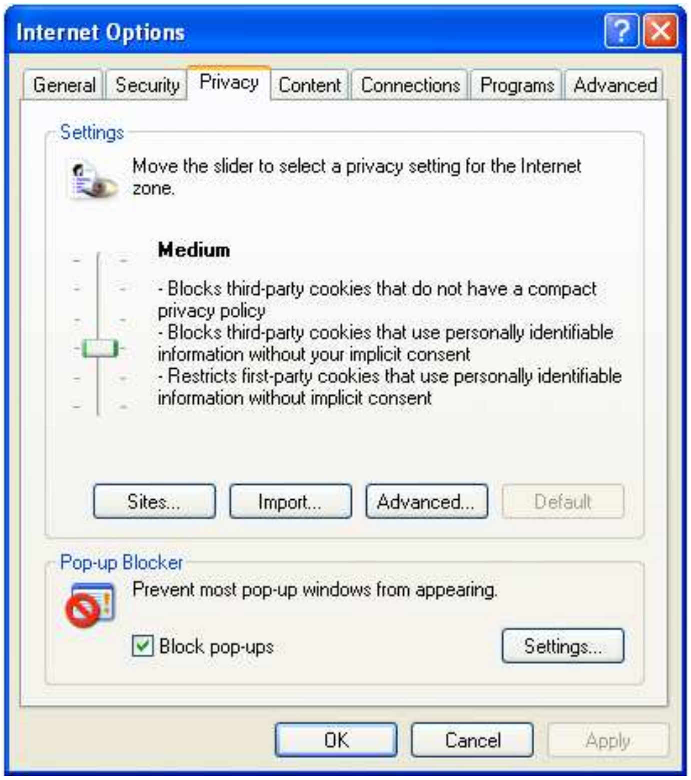
3.2 Tuner Preferences (T)

The user interface provided in IE6 is illustrated in Fig. 1. In the figure, the cookie blocker is set to "Medium" and the pop-up blocker is turned on. As is shown in the figure, when selecting a cookie blocker option additional information about that particular choice is displayed. Furthermore, the IE6 service offers no abstraction of the privacy configurations. Hence, the privacy configurations are directly controlled by the tuners and, thus, $T = CB \times PB$.

Note that some web sites require that cookies can be stored and later retrieved from the client to work properly. This implies that web sites that are built on cookies may not be accessible as expected when the high privacy protection policies are used in IE6.

3.3 Environmental Descriptors (E)

There is no explicit use of environmental descriptors, since the privacy configuration is directly controlled by the tuner. Hence, the set of environmental descriptors contains the empty set in this case, i.e., $E = \{\emptyset\}$. However, the tuner can take the environment into account when selecting a privacy configuration.



Settings

Move the slider to select a privacy setting for the Internet zone.

Medium

- Blocks third-party cookies that do not have a compact privacy policy
- Blocks third-party cookies that use personally identifiable information without your implicit consent
- Restricts first-party cookies that use personally identifiable information without implicit consent

Pop-up Blocker

Prevent most pop-up windows from appearing.

☑ Block pop-ups

Fig. 1. User interface for privacy settings provided to end users in IE6.

3.4 The TS Mapping

The TS function is in this case the identity mapping

$$TS(t, \emptyset) = t \tag{2}$$

where $t \in T$. The simplicity of the TS function is an effect of the direct tuner control of the privacy configurations. Thus, it is up to the tuner to select an appropriate privacy configuration.

4 Analyzing Mix Net

To achieve untraceable electronic mail David Chaum introduced the idea of mixes [1]. A mix is a special network station that has the basic task of hiding the relation between incoming and outgoing messages. Hence, a mix basically attains sender anonymity and unlinkability between sender and receiver. A network of mixes is called a Mix Net. In Fig. 2 a Mix Net chain, an ordered sequence of mixes, is illustrated.

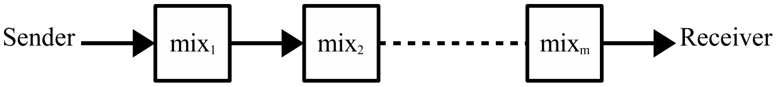


Fig. 2. A Mix Net chain.

The major work for a single mix is to collect messages in a pool, decide when a subset of messages should be flushed from the pool, and decide which subset of the messages in the pool to flush. The flushing conditions divide the mixes into two types, timed mixes and threshold mixes [2]. Timed mixes flush on certain predefined time intervals and threshold mixes flush when they have collected a certain amount of messages. A combination of the two types also exists [12]. The subset of messages to flush is determined by the pool flushing algorithm. Below we will analyze mixes that have a deterministic pool flushing algorithm [2].

4.1 Privacy Configurations (*S*)

A deterministic pool flushing algorithm uses the number of messages in the pool, n, to determine the number of messages to send out, s. For such mixes, we can write $s = nP$, where P is the fraction of sent messages, obviously $1 \leq s \leq n$. Note, however, that the subset of sent messages are still randomly chosen from the pool, even if the number of sent messages is deterministic. See Fig. 3 for an illustration of a mix with a deterministic flushing algorithm.

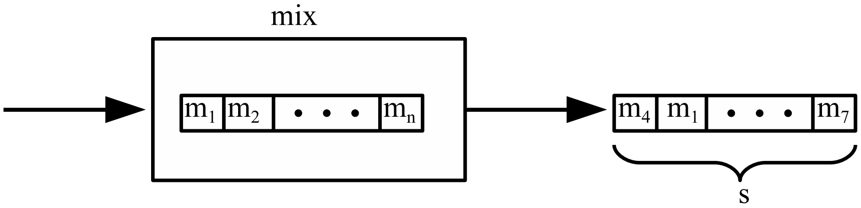


Fig. 3. A mix having n messages in the pool and flushing s messages.

The cycle of collecting and flushing messages is called one round. Furthermore, since a Mix Net consists of several mixes, we write s_{ij} to denote the number of sent messages in round i at mix j, $i \geq 1$ and $1 \leq j \leq m$. In a similar way, n_{ij} denotes the number of messages in the pool in round i at mix j. Now, since s_{ij} and n_{ij} are the only parameters that affect the privacy configurations, and also the performance, of a Mix Net with a deterministic pool flushing algorithm, we get the following privacy configurations $S = \prod_{j=1}^{m}(s_{ij} \times n_{ij})$.

4.2 Tuner Preferences (*T*)

For deterministic mixes the privacy configurations are directly controlled by the tuner, just as for IE6, since the system offers no abstraction of the privacy configurations. Furthermore, in a Mix Net there might be a tuner for each mix, which is the system owner of the mix. Hence, the set of tuner preferences is in this case equal to the set of possible privacy configurations, $T = \prod_{j=1}^{m}(s_{ij} \times n_{ij})$. Although not explicitly expressed in T, the selection of a privacy configuration represents a trade-off between the level of anonymity and the resulting overhead in terms of message delay in the system. Note, however, that it is the system owner for each mix that controls the anonymity level. Hence, the system owners are the tuners, not the end users.

4.3 Environmental Descriptors (*E*)

Exactly as for IE6, there is no explicit use of environmental descriptors when considering deterministic mixes, since the privacy configuration is directly controlled by the tuner. Hence, $E = \{\emptyset\}$.

4.4 The *TS* Mapping

The TS function is, hence, as for IE6, the identity mapping

$$TS(t, \emptyset) = t \qquad (3)$$

where $t \in T$. Thus, it is up to the tuner to select an appropriate privacy configuration and to investigate the trade-off between anonymity and performance.

5 Analyzing Crowds

The basic idea of Crowds [16] is to provide anonymous web browsing by hiding one end user's web actions within the web actions of many others. The Crowds system consists of two main components. The Jondo proxy application, which the browser requests must be set to go through, and the Blender server for managing memberships. See Fig. 4 for an illustration of the Crowds system.

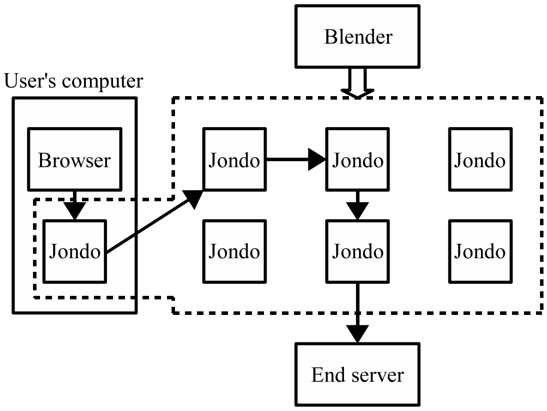


Fig. 4. The Crowds system.

5.1 Privacy Configurations (S)

One important parameter in Crowds is p_f, $0 \leq p_f < 1$. It gives the probability of forwarding a message, in the path creation process. When the first request arrives at a local Jondo, it forwards the request to another randomly selected Jondo in Crowds, possibly itself. The next Jondo, on the path, chooses to forward the request to another randomly selected Jondo in the Crowds system with probability p_f, or to submit the request to the end server with probability $1 - p_f$. This decision process, forward or submit, continues until a Jondo submits the request to the end server. Since the only parameter that gives the privacy configuration in Crowds is p_f, we get that $S = p_f$.

5.2 Tuner Preferences (T)

In [16], the authors defined six different anonymity levels (AL) for sender/receiver anonymity. Note that in the definitions below, the sender can be exchanged to receiver.

- A sender has absolute privacy (AP), if an observation gives the attacker no additional information.
- A sender is beyond suspicion (BS), if though the attacker can see evidence of a sent message the sender appears no more likely to be the originator of that message than any other potential sender in the system.
- A sender has probable innocence (PrI), if from the attacker's point of view the sender appears no more likely to be the originator than to not be the originator.
- A sender has possible innocence (PoI), if from the attacker's point of view there is a non trivial probability, $\delta > 0$, that the real sender is someone else.

- A sender is exposed (Ex), if the attacker can identify the sender but not necessarily prove it to others.
- A sender is provably exposed (PE), if the attacker can identify the sender and also prove the identity to others.

In this paper, we treat the ALs Ex and PE as equal. Furthermore, as will be discussed later, it is only possible to tune sender anonymity against attackers that are collaborating members. For this case, using the anonymity measure given in [16], the Crowds system can not be configured to guarantee BS and AP. Thus, $T = \{PrI, PoI, Ex\}$. However, as we will see, the Crowds system offers from the sender's perspective and under probabilistic assumption both AP and BS.

5.3 Environmental Descriptors (E)

Except from the p_f parameter, two further parameters, n and a, are needed to describe the Crowds system. The first parameter, n, represents the total number of Jondos in the Crowds system, hence, $n > 1$. The second parameter, a, on the other hand, represents one of the following three attacker types in the Crowds system.

1. A local eavesdropper (LE) is an attacker who can observe all communication to and from the computer of a specific Crowds member.
2. Collaborating members (CM) are attackers in the form of Crowds members that can pool their information and even deviate from the prescribed protocol.
3. The end servers (ES) are attackers to which web requests are directed.

Thus, $a \in A = \{LE, CM, ES\}$, and $E = n \times A$. When $a = CM$ we set $a = c$, the number of collaborating Jondos.

5.4 The TS Mapping

The sender/receiver ALs that are achieved by Crowds [16] are given in Table 1.

Table 1. Levels of anonymity offered by Crowds against different types of attackers.

Attacker	Sender Anonymity	Receiver Anonymity
LE	Ex	$\lim_{n\to\infty} P(BS) = 1$
CM	PrI if $n \geqslant \frac{p_f(c+1)}{p_f - \frac{1}{2}}$	$\lim_{n\to\infty} P(AP) = 1$
ES	BS	N/A

From Table 1 we see that for receiver anonymity against LE the probability of BS tends to one as the number of Jondos tends to infinity. However, receiver anonymity against LE can be both Ex and BS, depending on if the initiating Jondo finally sends the request by itself to the end server or not. If the initiating Jondo sends the request it will be unencrypted to the LE and receiver anonymity is Ex, otherwise the receiver has BS. The probability of the receiver, R, to be Ex against LE are

$$\begin{aligned} P_{R,LE}(Ex) &= \frac{1}{n}\sum_{i=0}^{\infty} p_f^i (1-p_f) \\ &= \frac{1}{n} \end{aligned} \tag{4}$$

Hence, $\lim_{n\to\infty} P_{R,LE}(Ex) = 0$ or $\lim_{n\to\infty} P_{R,LE}(BS) = 1$, as stated in Table 1.

For receiver anonymity against CM we have a similar situation. From Table 1 we see that the probability of AP tends to one as the number of Jondos tends to infinity, for a fixed number of CM. However, the receiver has AP if the path does not contain a CM and Ex otherwise, since all requests on a path are unencrypted to all crowds members on that path. The probability of the receiver, R, to have AP against CM are

$$\begin{aligned} P_{R,CM}(AP) &= \sum_{i=0}^{\infty} \left(\frac{n-c}{n}\right)^{i+1} p_f^i (1-p_f) \\ &= 1 - \frac{c}{n - p_f(n-c)} \end{aligned} \tag{5}$$

Hence, $\lim_{n\to\infty} P_{R,CM}(AP) = 1$, if c is fixed, as stated in Table 1.

For sender anonymity against CM, Table 1 only gives PrI when inequality (6) holds.

$$n \geqslant \frac{p_f(c+1)}{p_f - \frac{1}{2}} \tag{6}$$

However, in general for sender anonymity against CM three important cases can occur.

1. The sender, S, has AP if the path does not contain any CM. This happens with the same probability as $P_{R,CM}(AP)$, see equation (5). Thus,

$$P_{S,CM}(AP) = 1 - \frac{c}{n - p_f(n-c)} \tag{7}$$

2. The sender, S, is BS if he or she does not have a CM as an immediate successor, since the CMs suspect the immediate preceding Jondo more than the other Jondos. This situation occurs with probability

$$\begin{aligned} P_{S,CM}(BS) &= \frac{c(n-c-1)}{n^2}(1-p_f)\sum_{i=0}^{\infty} p_f^{i+1} \sum_{j=0}^{i} \left(\frac{n-c}{n}\right)^j \\ &= \frac{p_f c(n-c-1)}{n(n-p_f(n-c))} \end{aligned} \tag{8}$$

3. The sender, S, has X anonymity, where $X = PrI|PoI|Ex$ depending on the value of $P_{S,CM}(X)$, if he or she has a CM as an immediate successor. This situation occurs with probability

$$\begin{aligned} P_{S,CM}(X) &= \frac{c}{n}(1-p_f)\sum_{i=0}^{\infty} p_f^i + \frac{c}{n^2}(1-p_f)\sum_{i=0}^{\infty} p_f^{i+1} \sum_{j=0}^{i} \left(\frac{n-c}{n}\right)^j \\ &= \frac{c(n-p_f(n-c-1))}{n(n-p_f(n-c))} \end{aligned} \tag{9}$$

From the discussion above, Table 2 extends Table 1. In Table 2 all the possible ALs are included.

Table 2. Extended table of levels of anonymity offered by Crowds against different types of attackers.

Attacker	Sender Anonymity	Receiver Anonymity
LE	Ex	BS|Ex
CM	AP|BS|PrI|PoI|Ex	AP|Ex
ES	BS	N/A

It is only in situation three for sender anonymity against CM where it is possible with the value of p_f to guarantee the achieved anonymity. Thus, it is only possible to tune sender anonymity against CM. In [16], the authors derived and used an anonymity measure, $P(I|H_{1+})$, for sender anonymity against CM, where $P(I|H_{1+})$ is the probability that the path initiator is the first collaborator's immediate predecessor, given that there is at least one CM on the path. This is the same as

$$\begin{aligned} P(I|H_{1+}) &= \frac{p(X)}{1-p(AP)} \\ &= 1 - p_f\frac{n-c-1}{n} \\ &= 1 - p_f N(n,c) \end{aligned} \tag{10}$$

where $N(n,c)$ is the fraction of non-CMs in Crowds excluding your own Jondo. Note that in [16], the authors used another approach to calculate $P(I|H_{1+})$.

Using this measure, it is only possible to have $T = \{PrI|PoI|Ex\}$, since $AL = AP$ is excluded due to the fact that CMs are on the path, and $AL = BS$ is not possible, since

$$1 - p_f N(n,c) > \frac{1}{n} \tag{11}$$

Furthermore, by setting $P(I|H_{1+}) \leq \frac{1}{2}$, Reiter and Rubin [16] showed that Crowds offers PrI as long as inequality (6) holds, which can be rewritten to

$$p_f \geq \frac{1}{2N(n,c)} \tag{12}$$

This implies that we must have $N(n,c) \geq \frac{1}{2}$. Similarly, by setting $P(I|H_{1+}) \leq 1-\delta$, Crowds offers PoI as long as

$$p_f \geq \frac{\delta}{N(n,c)} \tag{13}$$

Finally, by setting $P(I|H_{1+}) = 1$ the AL is Ex. Thus $p_f = 0$.

Now, assume that we would like to minimize the delay in Crowds, under a given privacy constraint. Then, since the expected path length [16] is, $L = \frac{2-p_f}{1-p_f}$, the smallest value of p_f minimizes the delay. Thus, we get the following TS function.

$$\begin{aligned} TS(PrI,n,c) &= \tfrac{1}{2N(n,c)} \\ TS(PoI,n,c) &= \tfrac{\delta}{N(n,c)} \\ TS(Ex,n,c) &= 0 \end{aligned} \tag{14}$$

We have in this case assumed that $N(n,c) \neq 0$, otherwise Crowds will only offer Ex. In Fig. 5, we have plotted X with respect to p_f for $N(n,c) = 1$ ($n \to \infty$, c fixed) and $N(n,c) = 1/2$ when $\delta = \frac{1}{6}$. Note, however, that it is not possible for the system to achieve PrI as $N(n,c)$ becomes one half.

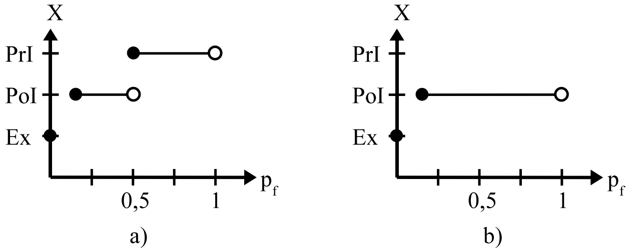


Fig. 5. X as a function of p_f for a) $N(n,c) = 1$ and b) $N(n,c) = 1/2$.

6 Discussion and Future Work

In this paper, three different services that provide privacy protection have been studied by using a previously proposed conceptual model for tunable security services. All three offer some form of privacy tunability. In the case of IE6, the degree of privacy protection is directly specified by the end user. The same holds for deterministic mixes, where the degree of anonymity is directly controlled by the mix owners. In Crowds, both end user preferences and environmental characteristics are taken into consideration when selecting a privacy configuration. A continuation of this work would be to investigate the tunable features in other privacy services.

The basic idea with tunable privacy services is to have the opportunity to utilize the trade-off between performance and privacy at run-time. However, to be able to select the most appropriate privacy configuration in a particular situation, both performance and privacy metrics must be studied further. A simple ordering of privacy configurations with respect to privacy and performance is sometimes sufficient, while finer grained metrics that also specify the difference between available configurations are needed in other situations. We believe that developing such metrics is a very challenging research task in this field.

From the three investigated privacy services, it is not trivial for the tuner to specify the most suitable configuration in a given situation. Hence, further research is clearly needed on end user privacy configurability in relation with usability, since very few end users are today able to make correct decisions on how to configure privacy services on their own computer. This is partly due to poor user interfaces and partly due to a lack of privacy awareness. More emphasis on the design of user interfaces for end user controlled privacy and more efforts on spreading knowledge about privacy will therefore considerably reduce the risk for the end users.

7 Concluding Remarks

In this paper, the tunable privacy features of IE6, Mix Net and Crowds have been analyzed. Both tuner preferences (T) and environmental characteristics (E) that influence the choice of a specific privacy configuration (S) have been identified. In addition, the mapping to a particular privacy configuration has been described through a mapping function, which is referred to as the TS function. This implies that some dynamic elements of each service have been identified and analyzed.

Acknowledgment

The work performed at Karlstad University was supported by grants from the Knowledge Foundations of Sweden with TietoEnator and Ericsson as industrial partners.

References

1. D. Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Communications of the ACM*, 4(2), February 1981. `http://www.eskimo.com/~weidai/mix-net.txt`.
2. C. Díaz. *Anonymity and Privacy in Electronic Services.* PhD thesis, Katholieke Universiteit Leuven, Leuven, Belgium, December 2005.
3. S. Fischer-Hbner and C. Andersson. Privacy risks and challenges for the mobile internet. In *Proceedings of the IEE Summit on Law and Computing*, London, November 2 2004.
4. C. T. R. Hager. *Context Aware and Adaptive Security for Wireless Networks.* PhD thesis, Virginia Polytechnic Institute and State University, Blacksburg, VA, USA, November 2004.

5. H. Johnson, L. Isaksson, M. Fiedler, and S. F. Wu. A decision system for adequate authentication. In *Proceedings of the International Conference on Networking, International Conference on Systems and International Conference on Mobile Communications and Learning Technologies (ICNICONSMCL'06)*, Washington, DC, USA, April 23–29, 2006. IEEE Computer Society.
6. Y. Li, Z. Chen, S. M. Tan, and R. H. Campbell. Security enhanced MPEG player. In *Proceedings of the 1996 International Workshop on Multimedia Software Development (MMSD'96)*, pages 169–176, Berlin, Germany, March 25–26 1996.
7. H. Lindskog and S. Lindskog. *Web Site Privacy with P3P.* Wiley Publishing, Indianapolis, IN, USA, 2003.
8. S. Lindskog, A. Brunstrom, and Z. Faigl. Analyzing tunable security services. In *Proceedings of the 3rd Swedish National Computer Networking Workshop (SNCNW 2006)*, Luleå, Sweden, October 26–27, 2006.
9. S. Lindskog, A. Brunstrom, Z. Faigl, and K. Tóth. Providing tunable security services: An IEEE 802.11i example. In *Proceedings of the first Workshop on Enterprise Network Security (WENS 2006)*, Baltimore, MD, USA, August 28, 2006.
10. S. Lindskog, A. Brunstrom, R. Lundin, and Z. Faigl. A conceptual model of tunable security services. In *Proceedings of the 3rd International Symposium on Wireless Communication Systems (ISWCS 2006)*, pages 531–535, Valencia, Spain, September 5–8, 2006.
11. J. Meyer and F. Gadegast. Security mechanisms for multimedia data with the example MPEG-I video, 1995. http://www.gadegast.de/frank/doc/secmeng.pdf.
12. U. Möller, L. Cottrell, P. Palfrader, and L. Sassaman. Mixmaster Protocol — Version 2, July 2003.
13. A. Pfitzmann and M. Hansen. Anonymity, unlinkability, undetectability, unobservability, pseudonymity, and identity management – a consolidated proposal for terminology. Draft, July 2007.
14. C. P. Pfleeger and S. L. Pfleeger. *Security in Computing.* Prentice Hall, Upper Saddle River, NJ, USA, 3rd edition, 2003.
15. P. Prasithsangaree and P. Krishnamurthy. On a framework for energy-efficient security protocols in wireless networks. *Computer Communications*, 27(17):1716–1729, 2004.
16. M. K. Reiter and A. D. Rubin. Crowds: anonymity for Web transactions. *ACM Transactions on Information and System Security*, 1(1):66–92, 1998.
17. A. Westin. *Privacy and Freedom.* Atheneum, New York, NY, USA, 1967.