

Network Working Group
Internet-Draft
Intended status: Informational
Expires: January 17, 2013

A. Cooper
CDT
H. Tschofenig
Nokia Siemens Networks
B. Aboba
Microsoft Corporation
J. Peterson
NeuStar, Inc.
J. Morris

M. Hansen
ULD Kiel
R. Smith
JANET(UK)
July 16, 2012

Privacy Considerations for Internet Protocols
draft-iab-privacy-considerations-03.txt

Abstract

This document offers guidance for developing privacy considerations for inclusion in IETF documents and aims to make protocol designers aware of privacy-related design choices.

Discussion of this document is taking place on the IETF Privacy Discussion mailing list (see <https://www.ietf.org/mailman/listinfo/ietf-privacy>).

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on January 17, 2013.

Copyright Notice

Internet-Draft

Privacy Considerations

July 2012

Copyright (c) 2012 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1.	Introduction	3
2.	Scope	4
3.	Terminology	6
3.1.	Entities	6
3.2.	Data and Analysis	6
3.3.	Identifiability	7
4.	Internet Privacy Threat Model	9
4.1.	Communications Model	9
4.2.	Privacy Threats	10
4.2.1.	Combined Security-Privacy Threats	11
4.2.2.	Privacy-Specific Threats	12
5.	Threat Mitigations	16
5.1.	Data Minimization	16
5.1.1.	Anonymity	16
5.1.2.	Pseudonymity	17
5.1.3.	Identity Confidentiality	18
5.1.4.	Data Minimization within Identity Management	18
5.2.	User Participation	19
5.3.	Security	19
6.	Guidelines	21
6.1.	General	21
6.2.	Data Minimization	21
6.3.	User Participation	22
6.4.	Security	23
7.	Example	24
8.	Security Considerations	29
9.	IANA Considerations	30

10.	Acknowledgements	31
11.	Informative References	32
	Authors' Addresses	35

[1.](#) Introduction

[RFC3552] provides detailed guidance to protocol designers about both how to consider security as part of protocol design and how to inform readers of IETF documents about security issues. This document intends to provide a similar set of guidance for considering privacy in protocol design.

Privacy is a complicated concept with a rich history that spans many disciplines. With regard to data, often it is a concept applied to "personal data," information relating to an identified or identifiable individual. Many sets of privacy principles and privacy design frameworks have been developed in different forums over the years. These include the Fair Information Practices (FIPs), a baseline set of privacy protections pertaining to the collection and use of personal data (often based on the principles established in [OECD], for example), and the Privacy by Design concept, which provides high-level privacy guidance for systems design (see [PbD] for one example). The guidance provided in this document is inspired by this prior work, but it aims to be more concrete, pointing protocol designers to specific engineering choices that can impact the privacy of the individuals that make use of Internet protocols.

Privacy as a legal concept is understood differently in different jurisdictions. The guidance provided in this document is generic and can be used to inform the design of any protocol to be used anywhere in the world, without reference to specific legal frameworks.

Whether any individual document will require a specific privacy considerations section will depend on the document's content. Documents whose entire focus is privacy may not merit a separate section (for example, [RFC3325]). For certain specifications, privacy considerations are a subset of security considerations and can be discussed explicitly in the security considerations section. The guidance provided here can and should be used to assess the privacy considerations of protocol, architectural, and operational

specifications and to decide whether those considerations are to be documented in a stand-alone section, within the security considerations section, or throughout the document.

This document is organized as follows. [Section 2](#) describes the extent to which the guidance offered is applicable within the IETF. [Section 3](#) explains the terminology used in this document. [Section 4](#) discusses threats to privacy as they apply to Internet protocols. [Section 5](#) outlines privacy goals. [Section 6](#) provides the guidelines for analyzing and documenting privacy considerations within IETF specifications. [Section 7](#) examines the privacy characteristics of an IETF protocol to demonstrate the use of the guidance framework.

[2.](#) Scope

The core function of IETF activity is building protocols. Internet protocols are often built flexibly, making them useful in a variety of architectures, contexts, and deployment scenarios without requiring significant interdependency between disparately designed components. Although protocol designers often have a particular target architecture or set of architectures in mind at design time, it is not uncommon for architectural frameworks to develop later, after implementations exist and have been deployed in combination with other protocols or components to form complete systems.

As a consequence, the extent to which protocol designers can foresee all of the privacy implications of a particular protocol at design time is significantly limited. An individual protocol may be relatively benign on its own, but when deployed within a larger system or used in a way not envisioned at design time, its use may create new privacy risks. Protocols are often implemented and deployed long after design time by different people than those who did the protocol design. The guidelines in [Section 6](#) ask protocol designers to consider how their protocols are expected to interact with systems and information that exist outside the protocol bounds, but not to imagine every possible deployment scenario.

Furthermore, in many cases the privacy properties of a system are dependent upon the complete system design where various protocols are combined together to form a product solution; the implementation, which includes the user interface design; and operational deployment practices, including default privacy settings and security processes

within the company doing the deployment. These details are specific to particular instantiations and generally outside the scope of the work conducted in the IETF. The guidance provided here may be useful in making choices about these details, but its primary aim is to assist with the design, implementation, and operation of protocols.

Transparency of data collection and use -- often effectuated through user interface design -- is normally a key factor in determining the privacy impact of a system. Although most IETF activities do not involve standardizing user interfaces or user-facing communications, in some cases understanding expected user interactions can be important for protocol design. Unexpected user behavior may have an adverse impact on security and/or privacy.

In sum, privacy issues, even those related to protocol development, go beyond the technical guidance discussed herein. As an example, consider HTTP [[RFC2616](#)], which was designed to allow the exchange of arbitrary data. A complete analysis of the privacy considerations for uses of HTTP might include what type of data is exchanged, how

this data is stored, and how it is processed. Hence the analysis for an individual's static personal web page would be different than the use of HTTP for exchanging health records. A protocol designer working on HTTP extensions (such as WebDAV [[RFC4918](#)]) is not expected to describe the privacy risks derived from all possible usage scenarios, but rather the privacy properties specific to the extensions and any particular uses of the extensions that are expected and foreseen at design time.

[3.](#) Terminology

This section defines basic terms used in this document, with references to pre-existing definitions as appropriate.

[3.1.](#) Entities

Several of these terms are further elaborated in [Section 4.1](#).

\$ Attacker: An entity that intentionally works against some protection goal.

\$ Eavesdropper: An entity that passively observes an initiator's communications without the initiator's knowledge or authorization. See [[RFC4949](#)].

\$ Enabler: A protocol entity that facilitates communication between an initiator and a recipient without being directly in the communications path.

\$ Individual: A natural person.

\$ Initiator: A protocol entity that initiates communications with a recipient.

\$ Intermediary: A protocol entity that sits between the initiator and the recipient and is necessary for the initiator and recipient to communicate. Unlike an eavesdropper, an intermediary is an entity that is part of the communication architecture. For example, a SIP proxy is an intermediary in the SIP architecture.

\$ Observer: An entity that is authorized to receive and handle data from an initiator and thereby is able to observe and collect information, potentially posing privacy threats depending on the context. As defined in this document, recipients, intermediaries, and enablers can all be observers.

\$ Recipient: A protocol entity that receives communications from an initiator.

[3.2.](#) Data and Analysis

\$ Correlation: The combination of various pieces of information relating to an individual.

\$ Fingerprint: A set of information elements that identifies a device or application instance.

\$ Fingerprinting: The process of an observer or attacker uniquely identifying (with a sufficiently high probability) a device or application instance based on multiple information elements communicated to the observer or attacker. See [\[EFF\]](#).

- \$ Item of Interest (IOI): Any data item that an observer or attacker might be interested in. This includes attributes, identifiers, identities, or communications interactions (such as the sending or receiving of a communication).
- \$ Personal Data: Any information relating to an identified individual or an individual who can be identified, directly or indirectly.
- \$ (Protocol) Interaction: A unit of communication within a particular protocol. A single interaction may be comprised of a single message between an initiator and recipient or multiple messages, depending on the protocol.
- \$ Traffic Analysis: The inference of information from observation of traffic flows (presence, absence, amount, direction, and frequency). See [[RFC4949](#)].
- \$ Undetectability: The inability of an observer or attacker to sufficiently distinguish whether an item of interest exists or not.
- \$ Unlinkability: Within a particular set of information, the inability of an observer or attacker to distinguish whether two items of interest are related or not (with a high enough degree of probability to be useful to the observer or attacker).

[3.3](#). Identifiability

- \$ Anonymity: The state of being anonymous.
- \$ Anonymous: A property of an individual in which an observer or attacker cannot identify the individual within a set of other individuals (the anonymity set).
- \$ Attribute: A property of an individual or initiator.

- \$ Identifiable: A state in which a individual's identity is capable

of being known.

- \$ **Identifiability:** The extent to which an individual is identifiable.
- \$ **Identified:** A state in which an individual's identity is known.
- \$ **Identifier:** A data object that represents a specific identity of a protocol entity or individual. See [[RFC4949](#)].
- \$ **Identification:** The linking of information to a particular individual to infer the individual's identity or that allows the inference of the individual's identity.
- \$ **Identity:** Any subset of an individual's attributes that identifies the individual within a given context. Individuals usually have multiple identities for use in different contexts.
- \$ **Identity confidentiality:** A property of an individual wherein any party other than the recipient cannot sufficiently identify the individual within a set of other individuals (the anonymity set).
- \$ **Identity provider:** An entity (usually an organization) that is responsible for establishing, maintaining, and securing the identity associated with individuals.
- \$ **Pseudonym:** An identifier of an individual other than the individual's real name.
- \$ **Pseudonymity:** The state of being pseudonymous.
- \$ **Pseudonymous:** A property of an individual in which the individual is identified by a pseudonym.
- \$ **Real name:** The opposite of a pseudonym. An individual's real name typically comprises his or her given names and a family name. An individual may have multiple real names over a lifetime, including legal names. From a technological perspective it cannot always be determined whether an identifier of an individual is a pseudonym or a real name.
- \$ **Relying party:** An entity that manages access to some resource. Security mechanisms allow the relying party to delegate aspects of identity management to an identity provider. This delegation requires protocol exchanges, trust, and a common understanding of semantics of information exchanged between the relying party and the identity provider.

[4.](#) Internet Privacy Threat Model

Privacy harms come in a number of forms, including harms to financial standing, reputation, solitude, autonomy, and safety. A victim of identity theft or blackmail, for example, may suffer a financial loss as a result. Reputational harm can occur when disclosure of information about an individual, whether true or false, subjects that individual to stigma, embarrassment, or loss of personal dignity. Intrusion or interruption of an individual's life or activities can harm the individual's ability to be left alone. When individuals or their activities are monitored, exposed, or at risk of exposure, those individuals may be stifled from expressing themselves, associating with others, and generally conducting their lives freely. They may also feel a general sense of unease, in that it is "creepy" to be monitored or to have data collected about them. In cases where such monitoring is for the purpose of stalking or violence, it can put individuals in physical danger.

This section lists common privacy threats (drawing liberally from [\[Solove\]](#), as well as [\[CoE\]](#)), showing how each of them may cause individuals to incur privacy harms and providing examples of how these threats can exist on the Internet.

[4.1.](#) Communications Model

To understand attacks in the privacy-harm sense, it is helpful to consider the overall communication architecture and different actors' roles within it. Consider a protocol element that initiates communication with some recipient (an "initiator"). Privacy analysis is most relevant for protocols with use cases in which the initiator acts on behalf of a natural person (or different people at different times). It is this individual whose privacy is potentially threatened.

Communications may be direct between the initiator and the recipient, or they may involve an application-layer intermediary (such as a proxy or cache) that is necessary for the two parties to communicate. In some cases this intermediary stays in the communication path for the entire duration of the communication and sometimes it is only used for communication establishment, for either inbound or outbound communication. In rare cases there may be a series of intermediaries that are traversed. At lower layers, additional entities are involved in packet forwarding that may interfere with privacy protection goals as well.

Some communications tasks require multiple protocol interactions with

different entities. For example, a request to an HTTP server may be preceded by an interaction between the initiator and an

Authentication, Authorization, and Accounting (AAA) server for network access and to a DNS server for name resolution. In this case, the HTTP server is the recipient and the other entities are enablers of the initiator-to-recipient communication. Similarly, a single communication with the recipient may generate further protocol interactions between either the initiator or the recipient and other entities. For example, an HTTP request might trigger interactions with an authentication server or with other resource servers.

As a general matter, recipients, intermediaries, and enablers are usually assumed to be authorized to receive and handle data from initiators. As [\[RFC3552\]](#) explains, "we assume that the end-systems engaging in a protocol exchange have not themselves been compromised."

Although recipients, intermediaries, and enablers may not generally be considered as attackers, they may all pose privacy threats (depending on the context) because they are able to observe and collect privacy-relevant data. These entities are collectively described below as "observers" to distinguish them from traditional attackers. From a privacy perspective, one important type of attacker is an eavesdropper: an entity that passively observes the initiator's communications without the initiator's knowledge or authorization.

The threat descriptions in the next section explain how observers and attackers might act to harm individuals' privacy. Different kinds of attacks may be feasible at different points in the communications path. For example, an observer could mount surveillance or identification attacks between the initiator and intermediary, or instead could surveil an enabler (e.g., by observing DNS queries from the initiator).

[4.2.](#) Privacy Threats

Some privacy threats are already considered in IETF protocols as a matter of routine security analysis. Others are more pure privacy threats that existing security considerations do not usually address. The threats described here are divided into those that may also be

considered security threats and those that are primarily privacy threats.

Note that an individual's awareness of and consent to the practices described below can greatly affect the extent to which they threaten privacy. If an individual authorizes surveillance of his own activities, for example, the harms associated with it may be mitigated, or the individual may accept the risk of harm.

Cooper, et al.

Expires January 17, 2013

[Page 10]

Internet-Draft

Privacy Considerations

July 2012

[4.2.1.](#) Combined Security-Privacy Threats

[4.2.1.1.](#) Surveillance

Surveillance is the observation or monitoring of an individual's communications or activities. The effects of surveillance on the individual can range from anxiety and discomfort to behavioral changes such as inhibition and self-censorship to the perpetration of violence against the individual. The individual need not be aware of the surveillance for it to impact privacy -- the possibility of surveillance may be enough to harm individual autonomy.

Surveillance can be conducted by observers or eavesdroppers at any point along the communications path. Confidentiality protections (as discussed in [\[RFC3552\] Section 3](#)) are necessary to prevent surveillance of the content of communications. To prevent traffic analysis or other surveillance of communications patterns, other measures may be necessary, such as [\[Tor\]](#).

[4.2.1.2.](#) Stored Data Compromise

End systems that do not take adequate measures to secure stored data from unauthorized or inappropriate access expose individuals to potential financial, reputational, or physical harm.

Protecting against stored data compromise is typically outside the scope of IETF protocols. However, a number of common protocol functions -- key management, access control, or operational logging, for example -- require the storage of data about initiators of communications. When requiring or recommending that information about initiators or their communications be stored or logged by end systems (see, e.g., [RFC 6302](#)), it is important to recognize the

potential for that information to be compromised and for that potential to be weighed against the benefits of data storage. Any recipient, intermediary, or enabler that stores data may be vulnerable to compromise.

[4.2.1.3.](#) Intrusion

Intrusion consists of invasive acts that disturb or interrupt one's life or activities. Intrusion can thwart individuals' desires to be let alone, sap their time or attention, or interrupt their activities.

Unsolicited messages and denial-of-service attacks are the most common types of intrusion on the Internet. Intrusion can be perpetrated by any attacker that is capable of sending unwanted traffic to the initiator.

Cooper, et al.

Expires January 17, 2013

[Page 11]

Internet-Draft

Privacy Considerations

July 2012

[4.2.1.4.](#) Misattribution

Misattribution occurs when data or communications related to one individual are attributed to another. Misattribution can result in adverse reputational, financial, or other consequences for individuals that are misidentified.

Misattribution in the protocol context comes as a result of using inadequate or insecure forms of identity or authentication. For example, as [[RFC6269](#)] notes, abuse mitigation is often conducted on the basis of source IP address, such that connections from individual IP addresses may be prevented or temporarily blacklisted if abusive activity is determined to be sourced from those addresses. However, in the case where a single IP address is shared by multiple individuals, those penalties may be suffered by all individuals sharing the address, even if they were not involved in the abuse. This threat can be mitigated by using identity management mechanisms with proper forms of authentication (ideally with cryptographic properties) so that actions can be attributed uniquely to an individual to provide the basis for accountability without generating false-positives.

[4.2.2.](#) Privacy-Specific Threats

[4.2.2.1.](#) Correlation

Correlation is the combination of various pieces of information related to an individual. Correlation can defy people's expectations of the limits of what others know about them. It can increase the power that those doing the correlating have over individuals as well as correlators' ability to pass judgment, threatening individual autonomy and reputation.

Correlation is closely related to identification. Internet protocols can facilitate correlation by allowing individuals' activities to be tracked and combined over time. The use of persistent or infrequently replaced identifiers at any layer of the stack can facilitate correlation. For example, an initiator's persistent use of the same device ID, certificate, or email address across multiple interactions could allow recipients to correlate all of the initiator's communications over time.

As an example, consider Transport Layer Security (TLS) session resumption [[RFC5246](#)] or TLS session resumption without server side state [[RFC5077](#)]. In [RFC 5246](#) [[RFC5246](#)] a server provides the client with a session_id in the ServerHello message and caches the master_secret for later exchanges. When the client initiates a new connection with the server it re-uses the previously obtained

session_id in its ClientHello message. The server agrees to resume the session by using the same session_id and the previously stored master_secret for the generation of the TLS Record Layer security association. [RFC 5077](#) [[RFC5077](#)] borrows from the session resumption design idea but the server encapsulates all state information into a ticket instead of caching it. An attacker who is able to observe the protocol exchanges between the TLS client and the TLS server is able to link the initial exchange to subsequently resumed TLS sessions when the session_id and the ticket are exchanged in the clear (which is the case with data exchanged in the initial handshake messages).

In theory any observer or attacker that receives an initiator's communications can engage in correlation. The extent of the potential for correlation will depend on what data the entity receives from the initiator and has access to otherwise. Often, intermediaries only require a small amount of information for message routing and/or security. In theory, protocol mechanisms could ensure that end-to-end information is not made accessible to these entities,

but in practice the difficulty of deploying end-to-end security procedures, additional messaging or computational overhead, and other business or legal requirements often slow or prevent the deployment of end-to-end security mechanisms, giving intermediaries greater exposure to initiators' data than is strictly necessary from a technical point of view.

[4.2.2.2.](#) Identification

Identification is the linking of information to a particular individual. In some contexts it is perfectly legitimate to identify individuals, whereas in others identification may potentially stifle individuals' activities or expression by inhibiting their ability to be anonymous or pseudonymous. Identification also makes it easier for individuals to be explicitly controlled by others (e.g., governments) and to be treated differentially compared to other individuals.

Many protocols provide functionality to convey the idea that some means has been provided to guarantee that entities are who they claim to be. Often, this is accomplished with cryptographic authentication. Furthermore, many protocol identifiers, such as those used in SIP or XMPP, may allow for the direct identification of individuals. Protocol identifiers may also contribute indirectly to identification via correlation. For example, a web site that does not directly authenticate users may be able to match its HTTP header logs with logs from another site that does authenticate users, rendering users on the first site identifiable.

As with correlation, any observer or attacker may be able to engage

in identification depending on the information about the initiator that is available via the protocol mechanism or other channels.

[4.2.2.3.](#) Secondary Use

Secondary use is the use of collected information without the individual's consent for a purpose different from that for which the information was collected. Secondary use may violate people's expectations or desires. The potential for secondary use can generate uncertainty over how one's information will be used in the future, potentially discouraging information exchange in the first

place.

One example of secondary use would be a network access server that uses an initiator's access requests to track the initiator's location. Any observer or attacker could potentially make unwanted secondary uses of initiators' data. Protecting against secondary use is typically outside the scope of IETF protocols.

4.2.2.4. Disclosure

Disclosure is the revelation of information about an individual that affects the way others judge the individual. Disclosure can violate individuals' expectations of the confidentiality of the data they share. The threat of disclosure may deter people from engaging in certain activities for fear of reputational harm, or simply because they do not wish to be observed.

Any observer or attacker that receives data about an initiator may engage in disclosure. Sometimes disclosure is unintentional because system designers do not realize that information being exchanged relates to individuals. The most common way for protocols to limit disclosure is by providing access control mechanisms (discussed in the next section). A further example is provided by the IETF geolocation privacy architecture [[RFC6280](#)], which supports a way for users to express a preference that their location information not be disclosed beyond the intended recipient.

4.2.2.5. Exclusion

Exclusion is the failure to allow individuals to know about the data that others have about them and to participate in its handling and use. Exclusion reduces accountability on the part of entities that maintain information about people and creates a sense of vulnerability about individuals' ability to control how information about them is collected and used.

The most common way for Internet protocols to be involved in limiting

exclusion is through access control mechanisms. The presence architecture developed in the IETF is a good example where individuals are included in the control of information about them. Using a rules expression language (e.g., Presence Authorization Rules

[[RFC5025](#)]), presence clients can authorize the specific conditions under which their presence information may be shared.

Exclusion is primarily considered problematic when the recipient fails to involve the initiator in decisions about data collection, handling, and use. Eavesdroppers engage in exclusion by their very nature since their data collection and handling practices are covert.

[5.](#) Threat Mitigations

Privacy is notoriously difficult to measure and quantify. The extent to which a particular protocol, system, or architecture "protects" or "enhances" privacy is dependent on a large number of factors relating to its design, use, and potential misuse. However, there are certain widely recognized classes of mitigations against the threats discussed in [Section 4.2](#). This section describes three categories of relevant mitigations: (1) data minimization, (2) user participation, and (3) security.

[5.1.](#) Data Minimization

Data minimization refers to collecting, using, disclosing, and storing the minimal data necessary to perform a task. The less data about individuals that gets exchanged in the first place, the lower the chances of that data being misused or leaked.

Data minimization can be effectuated in a number of different ways, including by limiting collection, use, disclosure, retention, identifiability, sensitivity, and access to personal data. Limiting the data collected by protocol elements only to what is necessary (collection limitation) is the most straightforward way to ensure that use of the data does not incur privacy harm. In some cases, protocol designers may also be able to recommend limits to the use or retention of data, although protocols themselves are not often capable of controlling these properties.

However, the most direct application of data minimization to protocol design is limiting identifiability. Reducing the identifiability of data by using pseudonymous or anonymous identifiers helps to weaken the link between an individual and his or her communications. Allowing for the periodic creation of new identifiers reduces the possibility that multiple protocol interactions or communications can be correlated back to the same individual. The following sections explore a number of different properties related to identifiability that protocol designers may seek to achieve.

(Threats mitigated: surveillance, stored data compromise, correlation, identification, secondary use, disclosure)

[5.1.1.](#) Anonymity

To enable anonymity of an individual, there must exist a set of individuals with potentially the same attributes. To the attacker or the observer these individuals must appear indistinguishable from each other. The set of all such individuals is known as the

anonymity set and membership of this set may vary over time.

The composition of the anonymity set depends on the knowledge of the observer or attacker. Thus anonymity is relative with respect to the observer or attacker. An initiator may be anonymous only within a set of potential initiators -- its initiator anonymity set -- which itself may be a subset of all individuals that may initiate communications. Conversely, a recipient may be anonymous only within a set of potential recipients -- its recipient anonymity set. Both anonymity sets may be disjoint, may overlap, or may be the same.

As an example, consider [RFC 3325](#) (P-Asserted-Identity, PAI) [[RFC3325](#)], an extension for the Session Initiation Protocol (SIP), that allows an individual, such as a VoIP caller, to instruct an intermediary that he or she trusts not to populate the SIP From header field with the individual's authenticated and verified identity. The recipient of the call, as well as any other entity outside of the individual's trust domain, would therefore only learn that the SIP message (typically a SIP INVITE) was sent with a header field 'From: "Anonymous" <sip:anonymous@anonymous.invalid>' rather than the individual's address-of-record, which is typically thought of as the "public address" of the user. When PAI is used, the individual becomes anonymous within the initiator anonymity set that is populated by every individual making use of that specific intermediary.

Note that this example ignores the fact that other personal data may be inferred from the other SIP protocol payloads. This caveat makes the analysis of the specific protocol extension easier but cannot be assumed when conducting analysis of an entire architecture.

[5.1.2.](#) Pseudonymity

In the context of IETF protocols, almost all identifiers are pseudonyms since there is typically no requirement to use real names in protocols. However, in certain scenarios it is reasonable to assume that real names will be used (with vCard [[RFC6350](#)], for example).

Pseudonymity is strengthened when less personal data can be linked to the pseudonym; when the same pseudonym is used less often and across fewer contexts; and when independently chosen pseudonyms are more

frequently used for new actions (making them, from an observer's or attacker's perspective, unlinkable).

For Internet protocols it is important whether protocols allow pseudonyms to be changed without human interaction, the default length of pseudonym lifetimes, to whom pseudonyms are exposed, how individuals are able to control disclosure, how often pseudonyms can be changed, and the consequences of changing them.

Cooper, et al.

Expires January 17, 2013

[Page 17]

Internet-Draft

Privacy Considerations

July 2012

[5.1.3.](#) Identity Confidentiality

An initiator has identity confidentiality when any party other than the recipient cannot sufficiently identify the initiator within the anonymity set. In comparison to anonymity and pseudonymity, identity confidentiality is concerned with eavesdroppers and intermediaries.

As an example, consider the network access authentication procedures utilizing the Extensible Authentication Protocol (EAP) [[RFC3748](#)]. EAP includes an identity exchange where the Identity Response is primarily used for routing purposes and selecting which EAP method to use. Since EAP Identity Requests and Responses are sent in cleartext, eavesdroppers and intermediaries along the communication path between the EAP peer and the EAP server can snoop on the identity. To address this treat, as discussed in [RFC 4282](#) [[RFC4282](#)], the user's identity can be hidden against these eavesdroppers and intermediaries with the cryptography support by EAP methods. Identity confidentiality has become a recommended design criteria for EAP (see [[RFC4017](#)]). EAP-AKA [[RFC4187](#)], for example, protects the EAP peer's identity against passive adversaries by utilizing temporal identities. EAP-IKEv2 [[RFC5106](#)] is an example of an EAP method that offers protection against active attackers with regard to the individual's identity.

[5.1.4.](#) Data Minimization within Identity Management

Modern systems are increasingly relying on multi-party transactions to authenticate individuals. Many of these systems make use of an identity provider that is responsible for providing authentication and authorization information to entities (relying parties) that require authentication or authorization of individuals in order to process transactions or grant access. To facilitate the provision of authentication and authorization, an identity provider will usually

go through a process of verifying the individual's identity and issuing the individual a set of credentials. When an individual seeks to make use of a service provided by the relying party, the relying party relies on the authentication and authorization assertions provided by its identity provider.

Such systems have the ability to support a number of properties that minimize data collection in different ways:

Relying parties can be prevented from knowing the real or pseudonymous identity of an individual, since the identity provider is the only entity involved in verifying identity.

Cooper, et al.

Expires January 17, 2013

[Page 18]

Internet-Draft

Privacy Considerations

July 2012

Relying parties that collude can be prevented from using an individual's credentials to track the individual. That is, two different relying parties can be prevented from determining that the same individual has authenticated to both of them. This requires that each relying party use a different means of identifying individuals.

The identity provider can be prevented from knowing which relying parties an individual interacted with. This requires avoiding direct communication between the identity provider and the relying party at the time when access to a resource by the initiator is made.

[5.2.](#) User Participation

As explained in [Section 4.2.2.5](#), data collection and use that happens "in secret," without the individual's knowledge, is apt to violate the individual's expectation of privacy and may create incentives for misuse of data. As a result, privacy regimes tend to include provisions to support informing individuals about data collection and use and involving them in decisions about the treatment of their data. In an engineering context, supporting the goal of user participation usually means providing ways for users to control the data that is shared about them. It may also mean providing ways for users to signal how they expect their data to be used and shared. (Threats mitigated: surveillance, secondary use, disclosure,

exclusion)

[5.3.](#) Security

Keeping data secure at rest and in transit is another important component of privacy protection. As they are described in [\[RFC3552\]](#) [Section 2](#), a number of security goals also serve to enhance privacy:

- o Confidentiality: Keeping data secret from unintended listeners.
- o Peer entity authentication: Ensuring that the endpoint of a communication is the one that is intended (in support of maintaining confidentiality).
- o Unauthorized usage: Limiting data access to only those users who are authorized. (Note that this goal also falls within data minimization.)
- o Inappropriate usage: Limiting how authorized users can use data. (Note that this goal also falls within data minimization.)

Note that even when these goals are achieved, the existence of items

of interest -- attributes, identifiers, identities, communications, actions (such as the sending or receiving of a communication), or anything else an attacker or observer might be interested in -- may still be detectable, even if they are not readable. Thus undetectability, in which an observer or attacker cannot sufficiently distinguish whether an item of interest exists or not, may be considered as a further security goal (albeit one that can be extremely difficult to accomplish).

(Threats mitigated: surveillance, stored data compromise, misattribution, secondary use, disclosure, intrusion)

[6.](#) Guidelines

This section provides guidance for document authors in the form of a questionnaire about a protocol being designed. The questionnaire may be useful at any point in the design process, particularly after document authors have developed a high-level protocol model as described in [[RFC4101](#)].

Note that the guidance does not recommend specific practices. The range of protocols developed in the IETF is too broad to make recommendations about particular uses of data or how privacy might be balanced against other design goals. However, by carefully

considering the answers to each question, document authors should be able to produce a comprehensive analysis that can serve as the basis for discussion of whether the protocol adequately protects against privacy threats.

The framework is divided into four sections that address each of the mitigation classes from [Section 5](#), plus a general section. Security is not fully elaborated since substantial guidance already exists in [\[RFC3552\]](#).

[6.1.](#) General

a. Trade-offs. Does the protocol make trade-offs between privacy and usability, privacy and efficiency, privacy and implementability, or privacy and other design goals? Describe the trade-offs and the rationale for the design chosen.

[6.2.](#) Data Minimization

a. Identifiers. What identifiers does the protocol use for distinguishing initiators of communications? Does the protocol use identifiers that allow different protocol interactions to be correlated?

b. Data. What information does the protocol expose about individuals, their devices, and/or their device usage (other than the identifiers discussed in (a))? To what extent is this information linked to the identities of the individuals? How does the protocol combine personal data with the identifiers discussed in (a)?

c. Observers. Which information discussed in (a) and (b) is exposed to each other protocol entity (i.e., recipients, intermediaries, and enablers)? Are there ways for protocol implementers to choose to limit the information shared with each entity? Are there operational controls available to limit the

information shared with each entity?

d. Fingerprinting. In many cases the specific ordering and/or occurrences of information elements in a protocol allow users, devices, or software using the protocol to be fingerprinted. Is

this protocol vulnerable to fingerprinting? If so, how?

e. Persistence of identifiers. What assumptions are made in the protocol design about the lifetime of the identifiers discussed in (a)? Does the protocol allow implementers or users to delete or replace identifiers? How often does the specification recommend to delete or replace identifiers by default?

f. Correlation. Does the protocol allow for correlation of identifiers? Are there expected ways that information exposed by the protocol will be combined or correlated with information obtained outside the protocol? How will such combination or correlation facilitate fingerprinting of a user, device, or application? Are there expected combinations or correlations with outside data that will make users of the protocol more identifiable?

g. Retention. Do the protocol or its anticipated uses require that the information discussed in (a) or (b) be retained by recipients, intermediaries, or enablers? Is the retention expected to be persistent or temporary?

6.3. User Participation

a. User control. What controls or consent mechanisms does the protocol define or require before personal data or identifiers are shared or exposed via the protocol? If no such mechanisms are specified, is it expected that control and consent will be handled outside of the protocol?

b. Control over sharing with individual recipients. Does the protocol provide ways for initiators to share different information with different recipients? If not, are there mechanisms that exist outside of the protocol to provide initiators with such control?

c. Control over sharing with intermediaries. Does the protocol provide ways for initiators to limit which information is shared with intermediaries? If not, are there mechanisms that exist outside of the protocol to provide users with such control? Is it expected that users will have relationships (contractual or otherwise) with intermediaries that govern the use of the information?

d. Preference expression. Does the protocol provide ways for initiators to express individuals' preferences to recipients or intermediaries with regard to the collection, use, or disclosure of their personal data?

6.4. Security

a. Surveillance. How do the protocol's security considerations prevent surveillance, including eavesdropping and traffic analysis?

b. Stored data compromise. How do the protocol's security considerations prevent or mitigate stored data compromise?

c. Intrusion. How do the protocol's security considerations prevent or mitigate intrusion, including denial-of-service attacks and unsolicited communications more generally?

d. Misattribution. How do the protocol's mechanisms for identifying and/or authenticating individuals prevent misattribution?

7. Example

The following section gives an example of the threat analysis and threat mitigation recommended by this document. It covers a particularly difficult application protocol, presence, to try to demonstrate these principles on an architecture that is vulnerable to many of the threats described above. This text is not intended as an example of a Privacy Considerations section that might appear in an IETF specification, but rather as an example of the thinking that should go into the design of a protocol when considering privacy as a first principle.

A presence service, as defined in the abstract in [\[RFC2778\]](#), allows users of a communications service to monitor one another's availability and disposition in order to make decisions about communicating. Presence information is highly dynamic, and generally characterizes whether a user is online or offline, busy or idle, away from communications devices or nearby, and the like. Necessarily, this information has certain privacy implications, and from the start the IETF approached this work with the aim to provide users with the controls to determine how their presence information would be shared. The Common Profile for Presence (CPP) [\[RFC3859\]](#) defines a set of logical operations for delivery of presence information. This abstract model is applicable to multiple presence systems. The SIP-based SIMPLE presence system [\[RFC3261\]](#) uses CPP as its baseline architecture, and the presence operations in the Extensible Messaging and Presence Protocol (XMPP) have also been mapped to CPP [\[RFC3922\]](#).

The fundamental architecture defined in [RFC 2778](#) and [RFC 3859](#) is a mediated one. Clients (presentities in [RFC 2778](#) terms) publish their presence information to presence servers, which in turn distribute information to authorized watchers. Presence servers thus retain presence information for an interval of time, until it either changes or expires, so that it can be revealed to authorized watchers upon request. This architecture mirrors existing pre-standard deployment models. The integration of an explicit authorization mechanism into the presence architecture has been widely successful in involving the end users in the decision making process before sharing information. Nearly all presence systems deployed today provide such a mechanism, typically through a reciprocal authorization system by which a pair of users, when they agree to be "buddies," consent to divulge their presence information to one another. Buddylists are managed by

servers but controlled by end users. Users can also explicitly block one another through a similar interface, and in some deployments it is desirable to provide "polite blocking" of various kinds.

From a perspective of privacy design, however, the classical presence architecture represents nearly a worst-case scenario. In terms of

data minimization, presences share their sensitive information with presence services, and while services only share this presence information with watchers authorized by the user, no technical mechanism constrains those watchers from relaying presence to further third parties. Any of these entities could conceivably log or retain presence information indefinitely. The sensitivity cannot be mitigated by rendering the user anonymous, as it is indeed the purpose of the system to facilitate communications between users who know one another. The identifiers employed by users are long-lived and often contain personal information, including real names and the domains of service providers. While users do participate in the construction of buddylists and blacklists, they do so with little prospect for accountability: the user effectively throws their presence information over the wall to a presence server that in turn distributes the information to watchers. Users typically have no way to verify that presence is being distributed only to authorized watchers, especially as it is the server that authenticates watchers, not the end user. Connections between the server and all publishers and consumers of presence data are moreover an attractive target for eavesdroppers, and require strong confidentiality mechanisms, though again the end user has no way to verify what mechanisms are in place between the presence server and a watcher.

Moreover, the sensitivity of presence information is not limited to the disposition and capability to communicate. Capability can reveal the type of device that a user employs, for example, and since multiple devices can publish the same user's presence, there are significant risks of allowing attackers to correlate user devices. An important extension to presence was developed to enable the support for location sharing. The effort to standardize protocols for systems sharing geolocation was started in the GEOPRIV working group. During the initial requirements and privacy threat analysis in the process of chartering the working group, it became clear that the system would require an underlying communication mechanism supporting user consent to share location information. The

resemblance of these requirements to the presence framework was quickly recognized, and this design decision was documented in [\[RFC4079\]](#). Location information thus mingles with other presence information available through the system to intermediaries and to authorized watchers.

Privacy concerns about presence information largely arise due to the built-in mediation of the presence architecture. The need for a presence server is motivated by two primary design requirements of presence: in the first place, the server can respond with an "offline" indication when the user is not online; in the second place, the server can compose presence information published by different devices under the user's control. Additionally, to

preserve the use of URIs as identifiers for entities, some service must operate a host with the domain name appearing in a presence URI, and in practical terms no commercial presence architecture would force end users to own and operate their own domain names. Many end users of applications like presence are behind NATs or firewalls, and effectively cannot receive direct connections from the Internet - the persistent bidirectional channel these clients open and maintain with a presence server is essential to the operation of the protocol.

One must first ask if the trade-off of mediation is worth it, for presence. Does a server need to be in the middle of all publications of presence information? It might seem that end-to-end encryption of the presence information could solve many of these problems. A presentity could encrypt the presence information with the public key of a watcher, and only then send the presence information through the server. The IETF defined an object format for presence information called the Presence Information Data Format (PIDF), which for the purposes of conveying location information was extended to the PIDF Location Object (PIDF-LO) - these XML objects were designed to accommodate an encrypted wrapper. Encrypting this data would have the added benefit of preventing stored cleartext presence information from being seized by an attacker who manages to compromise a presence server. This proposal, however, quickly runs into usability problems. Discovering the public keys of watchers is the first difficulty, one that few Internet protocols have addressed successfully. This solution would then require the presentity to publish one encrypted copy of its presence information per authorized watcher to the presence service, regardless of whether or not a

watcher is actively seeking presence information - for a presentity with many watchers, this may place an unacceptable burden on the presence server, especially given the dynamism of presence information. Finally, it prevents the server from composing presence information reported by multiple devices under the same user's control. On the whole, these difficulties render object encryption of presence information a doubtful prospect.

Some protocols that provide presence information, such as SIP, can operate intermediaries in a redirecting mode, rather than a publishing or proxying mode. Instead of sending presence information through the server, in other words, these protocols can merely redirect watchers to the presentity, and then presence information could pass directly and securely from the presentity to the watcher. In that case, the presentity can decide exactly what information it would like to share with the watcher in question, it can authenticate the watcher itself with whatever strength of credential it chooses, and with end-to-end encryption it can reduce the likelihood of any eavesdropping. In a redirection architecture, a presence server could still provide the necessary "offline" indication, without

requiring the presence server to observe and forward all information itself. This mechanism is more promising than encryption, but also suffers from significant difficulties. It too does not provide for composition of presence information from multiple devices - it in fact forces the watcher to perform this composition itself, which may lead to unexpected results. The largest single impediment to this approach is however the difficulty of creating end-to-end connections between the presentity's device(s) and a watcher, as some or all of these endpoints may be behind NATs or firewalls that prevent peer-to-peer connections. While there are potential solutions for this problem, like STUN and TURN, they add complexity to the overall system.

Consequently, mediation is a difficult feature of the presence architecture to remove, and due especially to the requirement for composition it is hard to minimize the data shared with intermediaries. Control over sharing with intermediaries must therefore come from some other explicit component of the architecture. As such, the presence work in the IETF focused on improving the user participation over the activities of the presence server. This work began in the GEOPRIV working group, with controls

on location privacy, as location of users is perceived as having especially sensitive properties. With the aim to meet the privacy requirements defined in [[RFC2779](#)] a set of usage indications, such as whether retransmission is allowed or when the retention period expires, have been added to PIDF-LO that always travel with location information itself. These privacy preferences apply not only to the intermediaries that store and forward presence information, but also to the watchers who consume it.

This approach very much follows the spirit of Creative Commons [[1](#)], namely the usage of a limited number of conditions (such as 'Share Alike' [[2](#)]). Unlike Creative Commons, the GEOPRIV working group did not, however, initiate work to produce legal language nor to design graphical icons since this would fall outside the scope of the IETF. In particular, the GEOPRIV rules state a preference on the retention and retransmission of location information; while GEOPRIV cannot force any entity receiving a PIDF-LO object to abide by those preferences, if users lack the ability to express them at all, we can guarantee their preferences will not be honored.

The retention and retransmission elements were envisioned as the only first and most essential examples of preference expression in sharing presence. The PIDF object was designed for extensibility, and the rulesets created for PIDF-LO can also be extended to provide new expressions of user preference. Not all user preference information should be bound into a particular PIDF object, however - many forms of access control policy assumed by the presence architecture need to

be provisioned in the presence server by some interface with the user. This requirement eventually triggered the standardization of a general access control policy language called the Common Policy (defined in [[RFC4745](#)]) framework. This language allows one to express ways to control the distribution of information as simple conditions, actions, and transformations rules expressed in an XML format. Common Policy itself is an abstract format which needs to be instantiated: two examples can be found with the Presence Authorization Rules [[RFC5025](#)] and the Geolocation Policy [[I-D.ietf-geopriv-policy](#)]. The former provides additional expressiveness for presence based systems, while the latter defines syntax and semantic for location based conditions and transformations.

Ultimately, the privacy work on presence represents a compromise between privacy principles and the needs of the architecture and marketplace. While it was not feasible to remove intermediaries from the architecture entirely, nor to prevent their access to presence information, the IETF did provide a way for users to express their preferences and provision their controls at the presence service. By documenting and acknowledging the limitations of these mechanisms, the designers were able to provide implementers, and end users, with an informed perspective on the privacy properties of the IETF's presence protocols.

[8.](#) Security Considerations

This document describes privacy aspects that protocol designers should consider in addition to regular security analysis.

9. IANA Considerations

This document does not require actions by IANA.

[10](#). Acknowledgements

We would like to thank Christine Runnegar for her extensive helpful review comments.

We would like to thank Scott Brim, Kasey Chappelle, Marc Linsner, Bryan McLaughlin, Nick Mathewson, Eric Rescorla, Scott Bradner, Nat Sakimura, Bjoern Hoehrmann, David Singer, Dean Willis, Christine Runnegar, Lucy Lynch, Trent Adams, Mark Lizar, Martin Thomson, Josh Howlett, Mischa Tuffield, S. Moonesamy, Ted Hardie, Zhou Sujing, Claudia Diaz, Leif Johansson, and Klaas Wierenga.

Finally, we would like to thank the participants for the feedback they provided during the December 2010 Internet Privacy workshop co-organized by MIT, ISOC, W3C and the IAB.

11. Informative References

- [Chaum] Chaum, D., "Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms", Communications of the ACM , 24/2, 84-88, 1981.
- [CoE] Council of Europe, "Recommendation CM/Rec(2010)13 of the Committee of Ministers to member states on the protection of individuals with regard to automatic processing of personal data in the context of profiling", available at (November 2010) , <https://wcd.coe.int/ViewDoc.jsp?Ref=CM/Rec%282010%2913>, 2010.
- [EFF] Electronic Frontier Foundation, "Panopticklick", 2011.
- [I-D.iab-identifier-comparison] Thaler, D., "Issues in Identifier Comparison for Security Purposes", [draft-iab-identifier-comparison-02](#) (work in progress), May 2012.
- [I-D.ietf-geopriv-policy] Schulzrinne, H., Tschofenig, H., Cuellar, J., Polk, J., Morris, J., and M. Thomson, "Geolocation Policy: A Document Format for Expressing Privacy Preferences for Location Information", [draft-ietf-geopriv-policy-26](#) (work in progress), June 2012.
- [OECD] Organization for Economic Co-operation and Development, "OECD Guidelines on the Protection of Privacy and Transborder Flows of Personal Data", available at (September 2010) , <http://www.oecd.org/EN/document/0,EN-document-0-nodirectorate-no-24-10255-0,00.html>, 1980.
- [PbD] Office of the Information and Privacy Commissioner,

Ontario, Canada, "Privacy by Design", 2011.

- [RFC2616] Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., and T. Berners-Lee, "Hypertext Transfer Protocol -- HTTP/1.1", [RFC 2616](#), June 1999.
- [RFC2778] Day, M., Rosenberg, J., and H. Sugano, "A Model for Presence and Instant Messaging", [RFC 2778](#), February 2000.
- [RFC2779] Day, M., Aggarwal, S., Mohr, G., and J. Vincent, "Instant Messaging / Presence Protocol Requirements", [RFC 2779](#), February 2000.

Cooper, et al.

Expires January 17, 2013

[Page 32]

Internet-Draft

Privacy Considerations

July 2012

- [RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", [RFC 3261](#), June 2002.
- [RFC3325] Jennings, C., Peterson, J., and M. Watson, "Private Extensions to the Session Initiation Protocol (SIP) for Asserted Identity within Trusted Networks", [RFC 3325](#), November 2002.
- [RFC3552] Rescorla, E. and B. Korver, "Guidelines for Writing RFC Text on Security Considerations", [BCP 72](#), [RFC 3552](#), July 2003.
- [RFC3748] Aboba, B., Blunk, L., Vollbrecht, J., Carlson, J., and H. Levkowetz, "Extensible Authentication Protocol (EAP)", [RFC 3748](#), June 2004.
- [RFC3859] Peterson, J., "Common Profile for Presence (CPP)", [RFC 3859](#), August 2004.
- [RFC3922] Saint-Andre, P., "Mapping the Extensible Messaging and Presence Protocol (XMPP) to Common Presence and Instant Messaging (CPIM)", [RFC 3922](#), October 2004.
- [RFC4017] Stanley, D., Walker, J., and B. Aboba, "Extensible Authentication Protocol (EAP) Method Requirements for Wireless LANs", [RFC 4017](#), March 2005.

- [RFC4079] Peterson, J., "A Presence Architecture for the Distribution of GEOPRIV Location Objects", [RFC 4079](#), July 2005.
- [RFC4101] Rescorla, E. and IAB, "Writing Protocol Models", [RFC 4101](#), June 2005.
- [RFC4187] Arkko, J. and H. Haverinen, "Extensible Authentication Protocol Method for 3rd Generation Authentication and Key Agreement (EAP-AKA)", [RFC 4187](#), January 2006.
- [RFC4282] Aboba, B., Beadles, M., Arkko, J., and P. Eronen, "The Network Access Identifier", [RFC 4282](#), December 2005.
- [RFC4745] Schulzrinne, H., Tschofenig, H., Morris, J., Cuellar, J., Polk, J., and J. Rosenberg, "Common Policy: A Document Format for Expressing Privacy Preferences", [RFC 4745](#), February 2007.

Cooper, et al.

Expires January 17, 2013

[Page 33]

Internet-Draft

Privacy Considerations

July 2012

- [RFC4918] Dusseault, L., "HTTP Extensions for Web Distributed Authoring and Versioning (WebDAV)", [RFC 4918](#), June 2007.
- [RFC4949] Shirey, R., "Internet Security Glossary, Version 2", [RFC 4949](#), August 2007.
- [RFC5025] Rosenberg, J., "Presence Authorization Rules", [RFC 5025](#), December 2007.
- [RFC5077] Salowey, J., Zhou, H., Eronen, P., and H. Tschofenig, "Transport Layer Security (TLS) Session Resumption without Server-Side State", [RFC 5077](#), January 2008.
- [RFC5106] Tschofenig, H., Kroeselberg, D., Pashalidis, A., Ohba, Y., and F. Bersani, "The Extensible Authentication Protocol-Internet Key Exchange Protocol version 2 (EAP-IKEv2) Method", [RFC 5106](#), February 2008.
- [RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", [RFC 5246](#), August 2008.
- [RFC6269] Ford, M., Boucadair, M., Durand, A., Levis, P., and P.

Roberts, "Issues with IP Address Sharing", [RFC 6269](#),
June 2011.

[RFC6280] Barnes, R., Lepinski, M., Cooper, A., Morris, J.,
Tschofenig, H., and H. Schulzrinne, "An Architecture for
Location and Location Privacy in Internet Applications",
[BCP 160](#), [RFC 6280](#), July 2011.

[RFC6350] Perreault, S., "vCard Format Specification", [RFC 6350](#),
August 2011.

[Solove] Solove, D., "Understanding Privacy", 2010.

[Tor] The Tor Project, Inc., "Tor", 2011.

[1] <<http://creativecommons.org/>>

[2] <http://wiki.creativecommons.org/Share_Alike>

Cooper, et al.

Expires January 17, 2013

[Page 34]

Internet-Draft

Privacy Considerations

July 2012

Authors' Addresses

Alissa Cooper
CDT
1634 Eye St. NW, Suite 1100
Washington, DC 20006
US

Phone: +1-202-637-9800
Email: acooper@cdt.org
URI: <http://www.cdt.org/>

Hannes Tschofenig
Nokia Siemens Networks

Linnoitustie 6
Espoo 02600
Finland

Phone: +358 (50) 4871445
Email: Hannes.Tschofenig@gmx.net
URI: <http://www.tschofenig.priv.at>

Bernard Aboba
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052
US

Email: bernarda@microsoft.com

Jon Peterson
NeuStar, Inc.
1800 Sutter St Suite 570
Concord, CA 94520
US

Email: jon.peterson@neustar.biz

John B. Morris, Jr.

Email: ietf@jmorris.org

Cooper, et al.

Expires January 17, 2013

[Page 35]

Internet-Draft

Privacy Considerations

July 2012

Marit Hansen
ULD Kiel

Email: marit.hansen@datenschutzzentrum.de

Rhys Smith
JANET(UK)

Email: rhys.smith@ja.net