Infrastructural Requirements for a Privacy Preserving Internet

Brad Rosen

Fall 2003

Professor Feigenbaum
Sensitive Information in the Wired World

Abstract

With much gusto, firms routinely sell "privacy enhancing technology" to enrich the web experience of typical consumers. Standards bodies have thrown in their hats, and even large organizations such as AT&T and IBM have gotten involved. Still, it seems no one has asked the question, "Are we trying to save a sinking ship?" "Are our ultimate goals actually achievable given the current framework?" This paper tries to examine the necessary infrastructure to support the goals of privacy enhancing technologies and the reasoning behind them.

Contents

1 Introduction

It has become readily apparent to even the most casual of observers that within the status quo, every internet-capable organization and entity is hemorrhaging information – "sensitive" or otherwise. More and more often, consumer advocacy groups, forward-thinking commercial firms, and tech-savvy users keep referring to "privacy enhancing technologies" and hawking the necessity to protect "sensitive" and "private" data. The overwhelming majority of these policies merely offer duct tape to cover the gaping cracks in a flawed infrastructure... the internet as it exists in its current form was not designed to protect information; it was designed to facilitate its dissemination. By examining the stringent requirements

necessary for a hypothetical internet experience that preserves and/or protects "privacy" and "sensitive" data, we hope to show how near to impossible that pipe dream truly is.

2 Definition of Terms

To properly frame discussion, the vague and sloppy definitions typically used for nebulous words such as "privacy" and "sensitive" must be well defined, if for no other reason than to succinctly specify the problem and task at hand.

2.1 User-Centric Terms

Privacy Enhancing Technology Policies, Practices and Technology to further increase the user's [or data originator's] control of data and/or control over the "internet experience."

Privacy Preserving Policies, Practices and Technology that ensure that all users are explicitly assured of having, a priori and irrevocably, the types of control that Privacy Enhancing Technologies try to empower them with.

Reasonable Expectations Users should reasonably expect to have control over to whom they give their data and with whom that data is shared, and should expect full control over their "internet experience." It is a reasonable expectation to not get pop ups – unfortunately it is not realistic.

Realistic Expectations Users can expect, in the absence of invasions of privacy by spyware and its ilk, to be in in control of to whom they give their data. They can expect some limited form of control over their experience, although very little with regards to web "annoyances" like pop ups, cookies and e-mail SPAM.

Desirable Expectations It would be ideal to provide users with complete granular control over to whom their data is given, under what circumstances it is used, stored, shared and looked at. Furthermore, it would be nice to give users complete and total control over their "internet experience", including their exposure to pop ups, cookies, e-mail SPAM et cetera.

2.2 Technical Terms

Proxy Server A computer that re-directs internet requests, masking their true source. Large scale NAT[1], dedicated proxies and web-accelerating servers all count as proxies.

Spyware Software that installs itself with the express intent of monitoring user behavior/performing user profiling, often to report back to a marketing or data-mining agency that will resell that information. Oftentimes, this software is installed surreptitiously by popular peer-to-peer filesharing programs, but has been known to be installed by malicious exploits. Otherwise known as Malware, one of the most detested features on the current privacy landscape. Often viewed as an act of trespassing or invasion of privacy as it consumes processor cycles, power, and in some cases, massive amounts of hard drive space[1].

Cookie Client-side information storage. Designed as a workaround for HTTP being a stateless protocol.

Cookie-Cutter Program which blocks websites from storing/reading cookies. *May be integrated into browser.*

Pop-Up Either a window which will appear ("pop") up on a windows based machine via the Microsoft Messenger Service, stealing focus from a user application, or an additional browser window, opened by many sites to display advertisements, sometimes opened by spyware and its ilk. As Microsoft Messgenger Popups are caused by a system service (*And a poor design choice on Microsoft's part*) [6] and do not effect users with sane filters at an ISP level, or on non-Microsoft platform, or with a sufficiently hardened system, we will consider ONLY the latter definition.

Pop-Up Blocker Program which prevents websites or third-party programs [c.f. spyware] from opening pop-up windows. *May be integrated into browser.*

Web-Bug Typically, one-by-one pixel "shims" (placeholders) that leverage common web hosting packages logging abilities to track users. Commonly, a single tracking site will host many web-bugs on various websites to keep a log of their browsing habits.

[1]Network Address Translation

Trusted Platform To be put most simply, a computing device on which it is possible to know the given output for all inputs by virtue of validation that the device is running the code it claims to be running.[2] The authenticity of its software/firmware can be verified.

2.3 Hypothetical Terms

These terms are presented here in brief, used extensively in 6.

Signing Bodies Disinvolved third parties willing to certify that a webserver [or internet-company] is running the software it claims it is running - by checking their currently running code sporadically against known signatures.

Verifying Bodies Disinvolved third parties willing to certify by exhaustive proof, induction or code inspection that a piece of software complies with the "P4P" policy that a company espouses for it.

Trusted Proxies Proxy servers that a user trusts not to save log files, statistically multiplex data, et cetera. Possible to enable trusted proxies via signing/verification bodies. Simply: A user trusts this proxy to protect his/her identity from the outside world.

P4P A "Perfect" version of P3P. This is a perfect policy language that has a total ordering and encompasses all aspects of data usage, including IP harvesting, profiling, click-tracking, statistical analysis of aggregate data, data sale, use, transfer, data protection and employee safeguards, protections against letting data fall into the kitchen sink, et cetera ad naseum.

3 Privacy and Annoyances

There has been a substantial blurring of what is considered "privacy" and what data is considered "sensitive." Many websites which attempt to sell such privacy enhancing technologies to consumers mostly discuss cookies, pop ups, and browser histories. [13] By and large, it is possible to divide the current space of protecting "privacy" and "sensitive" data into two halves; the prevention of unwanted *inflows* and *outflows*.

[2] Remote Attestation is the NGTCB/Palladium term.

3.1 Outflows – Encroachment

Outflows embody the true spirit of "invasion of privacy" and "violation of privacy." Outflows include practices like using customer databases in a manner that violates the privacy policy or the spirit thereof. One of the most well known, and current, examples is that of JetBlue, who shared customer data despite having a posted privacy policy. [16] This class involves tracking of user-actions and browsing habits[3], as well as the improper collection and/or use of "personal data."[4] Typically, outflows occur when a user submits data to a foreign web-server, thereby releasing that data into the wild. Typically, this data is shared with "valued partners" or "business affiliates" oftentimes without the assurances that their data-handling and data-security policies are as stringent as the firm to whom the data was trusted. Furthermore, the "user-profiling" and click-data tracking performed by Amazon.com and other well known sites can be considered outflows because this may be done without user consent. Some users even go so far as to consider website logs that collect IP addresses of visitors to be unwanted outflows.

Another type of outflow is the data that is reported back by spyware such as Gator, Alexa, and many "Media Plugins" and "System Accelerators." Much of the data collected by these programs is relatively benign, as they mostly report back browsing habits so that they can better target advertisements in the form of pop ups. [17] While the threat exists that more malicious spyware might attempt to usurp credit card numbers and other such data, by and large, outflows caused by spyware tend only to stimulate *inflows.*

3.2 Inflows – Annoyances

Inflows are the class of things that collectively can be called "annoyances." This includes pop-up windows, SPAM, telemarketers, physical junk mail, site redirects, and aggressive marketing[5]. There are a plethora of adjectives that have been used to describe inflows, "obnoxious marketing"[11], "relentless pop ups"[14], and plain old junk mail. While these may sap our time, frustrate us, and leave us feeling somewhat helpless, they are not the root of the problem, merely the symptoms. In the case of email SPAM and pop up ads, they are the result of poor architectural choices being exploited by unscrupulous business folk. Again,

[3]We do not distinguish by methods used. Cookies and Server-Side state can be used equally effectively to track user habits.

[4]The concept of "personally identifiable information" is fairly well understood: it includes such commonplace datums as SSNs, phone numbers, shipping and billing addresses, email address, and so forth.

[5]Typically, when you purchase something from a company and they send a flood of faux-legitimate correspondence under the guise of your existing patronage. This could also be arguably a misuse of data, and then viewed as an outflow of data, but this is more of a social concern than a technological one.

with email addresses, telephone and fax numbers and physical mailboxes, the unwanted communication is merely a result of the unwanted outflow of data. Had the user never lost control over his or her telephone number, email address or home address in the first place, there never would have been an opportunity to send spam, telemarketing calls or junk mail. For the scope of this paper, while we will touch on inflows briefly where appropriate to the infrastructural discussion, we will not hinge on them as they are either easily solved technologically[6] or a social problem resulting from a prior outflow[7].

3.3 Relevance

The overwhelming majority of cookies, pop ups, and other annoyances can easily be blocked by current technologies. The Opera, Mozilla, Netscape, Galeon, Safari, Konqueror, Omniweb, and other browsers based on Gecko and KHTML offer pop up blocking and cookie crushing. There are numerous toolbars and plugins available for Microsoft Internet Explorer that add the same functionality. Although many of the same tracking functions that cookies allow can be done server-side using web-bugs or tracking of session click data, even these referrer tags and remote loaded images can be stopped by these browsers. The only real remaining way for passive data harvesting is via user-login, and once a user has logged into a site, it follows logically that the site can link their actions to their account. As to what they do with that data, the site is only bound by their privacy policies[8] — and JetBlue has shown us how much those are worth. Therefore, our main focus will be on preventing data outflows that happen via some other mechanism. The focus on pop ups and cookies loses sight of the more pressing technical issues.

4 Privacy Preserving vs. Privacy Enhancing

The most important distinction we can make is between the enhancement of privacy that P3P and "best-of-breed" data practices strive to achieve and actual user ability to prevent outflows of data and the concept of unilateral protection and respect for the user or data subject's wishes. The very fact that we use the term "Privacy Enhancing" reflects the mental state we take when we approach the problem: privacy is viewed as something that must be tacked on and not something that is inherent in the design of the system. With the current internet, this is the case. The internetwork-network[9] was designed for efficient dissemination of information. As it grew, the focus was on reliable, scalable and fast distribu-

[6] In the case of SPAM and PopUps
[7] Telemarketing, junk mail, SPAM
[8] HIPAA et cetera notwithstanding
[9] The internet...

tion of data. Security and authenticity verification were afterthoughts in most common internet technologies[10]. In order to have the protected experience that makers of privacy enhancing technologies indicate that consumers want, security and verification need to be built into the infrastructure from the ground up, in lieu of being tacked on as afterthoughts. The internet, like most areas of computer science, is plagued by the "Law of Leaky Abstractions"[15]. The layered nature of the status quo internet allows many leaks to percolate up, leaving fundamental flaws that cannot be glossed over with anything but a paradigm shift. Instead of trying to bolt privacy enhancements onto an infrastructure that was not designed for it, if privacy and security are tantamount, we need to begin looking toward a privacy preserving infrastructure; a shift from the mantra that *Information is ubiquitous* to the idea that *Information is ubiquitously controlled by the entity that it concerns or the entity that generated it*[11]. While this paradigm shift may sound simple at a high level, it is multi-faceted and involves many aspects of computing that are poorly understood, if not completely taken for granted, by even significantly web-savvy users. An examination of the current infrastructure will provide the framework for one possible set of requirements for a next-generation *privacy preserving* infrastructure.

5 Current Infrastructure

5.1 Overview

We must reiterate a common theme at this point; the status quo works very well. Many, if not all users, are happy with the way the internet works. By and large, a few less-than-scrupulous firms and individuals have the potential to ruin it for the masses[12]. The internet grew out of a few physics researchers attempting to share data – at its most primitive level, there was an implicit understanding that no one would do anything "naughty." The commercialization and loss of privacy on the web took its creators by surprise.

5.2 DNS Request

When a user types a URL into a browser or other application, the first thing that must happen is that the sequence of characters must be translated into the IP address of the server that hosts that specific domain. This is fairly trivial when stated, but actually involves one of the largest and most complex distributed databases in the world [12]. Furthermore, the process is almost unilaterally taken

[10]DNSSEC and DNS, HTTPS and HTTP, SSH and TELNET....

[11]As applicable

[12]Feigenbaum's corollary to Metcalf's law: A network becomes more useful as it gains users, until even the scum of the Earth is on the network, and at that point it has no value to anyone.

for granted by end users, and a great deal of potentially sensitive information is revealed by outgoing DNS requests. First and foremost, DNS requests are sent *in cleartext* to local DNS servers. For many home users that share cable local loops, this means that any one of their neighbors can see what sites they are visiting by monitoring outgoing DNS requests. More sinister is the threat of DNS hi-jacking, where a hacker could make a clone of Amazon.com and harvest thousands of credit card numbers and other sensitive information[3]. [8]

5.3 Routing

Subsequent to DNS resolution, user requests are routed to the target website via a number of intermediate hops. These routers operate via *store and forward*, meaning they save a copy of packets being sent until they know they have been received by the next hop. All these packets contain both source and destination IP addresses, and there is no assurance that these packets will not be purged from memory. Any human with access to one of these routers could save packets that met certain criteria, and glean significant information from just the raw TCP streams.

5.4 Website Navigation

As a user navigates through the target site, their actions may be traced if they have logged on, have cookies enabled, or if they are not using a browser capable of blocking web-bugs[2]. The de-facto standards in web servers[13] have integrated logging functions that make tracking via IP address easy – and the types of users that are most likely to spoof their IP addresses are not the users that are overly concerned about the illicit behaviors of others.

5.5 Sensitive Data-Handling

Most often, transfer of "sensitive" data[14] is done via HTTPS – HTTP with secure sockets layer. Encryption of all streams is not possible[15] and so all data not sent via HTTPS can be compromised. HTTPS simply ensures safe transit of data to the recipient. The overwhelming problem is that once the user has transferred that data elsewhere, they have absolutely no control over it. No matter what privacy a company has posted, no matter what statements they may have made, the information is no longer under the direct control of the original user, and may be copied, modified, redistributed or sold without the user's permission.

[13]Microsoft Internet Information Services or IIS and APACHE
[14]Credit Card number, shipping address, phone number, email address
[15]Ibid

5.6 Infrastructural Details

We will briefly outline some infrastructural details that have created the space for many of today's so-called privacy enhancing technologies — most of which compensate for flaws in the design of the current internet.

5.6.1 IPv4

IPv4 Is the low-level transfer protocol that drives the internet [4]. Due to the limited size of IPv4, NAT has become frequent in the United States, preventing more common usage of encryption tools such as Kerberos and IPSEC[10]. The end-to-end security measures available for IPv4 are not native, and spoofing of IPv4 addresses is both common and uncomplicated. This means that many common tools that would otherwise be able to verify authenticity by IP address cannot rely on source IP's – this has a large impact on email validation and the SPAM inflow problem.

5.6.2 Java/ECMA Script

Javascript, formally standardized as ECMA script, was intended to do away with a number of perceived shortcomings in HTTP and HTML. The decision to allow for opening of new windows via Java/ECMA script is the single source of one of the greatest frustrations in the internet: pop ups.

5.6.3 Applets/ActiveX

Java Applets and ActiveX controls[16] have a much greater impact than simple pop-ups, web bugs, cookies or Java/ECMA script. These plugins may access the hard-drive, re-write files, and even be capable of rebooting a system. Moreso, they are used to install spyware. Although tools already exist for the management of these permissions, users often blindly click "yes". This is a social problem and a problem of insufficient knowledge, and therefore cannot simply be solved with technology.

5.6.4 (E)SMTP

The Extended Simple Mail Transfer Protocol started out as a simple and efficient way of ensuring the reliable delivery of email. Due to the simple nature of the early internet – no sender validation is required. Servers[17] will blindly accept any incoming message and endeavor to deliver it, despite not having checked that the sending address is even valid, much less that the sender is authentic. The

[16] As a majority of the web-using public is on a machine running a Microsoft Operating System and running MSIE, ActiveX controls are relevant.

[17] open relays

problem is compounded by IPv4, because one of the simplest checks that could have been implemented, source IP checking, is defeated by simple IP spoofing – a technique spammers already employ.

6 Next-Generation Infrastructure

The "Next-Generation" Infrastructure is differs markedly in a number of areas from the current system. It requires tremendous overhead in terms of encryption, authenticity validation and processing. Furthermore, it requires ubiquitous "trusted computing" and independent bodies that are willing to certify that organizations are running the software that complies with their posted privacy policies[18] and other bodies that can certify that the software being run actually *does* obey the enumerated policy.

6.1 Overview

Many of the tools required are "on the horizon" or "have been discussed." The biggest issue facing these tools are significant barriers to entry[5]. Also, thanks to the law of leaky abstractions, without securing all phases of the pipeline, the problems in the insecure stages will continue to leak through. It must be kept in mind that for this paradigm shift to be espoused, the mantra of efficiency must take a backseat to security and authenticity. We require that users, a priori, mathematically express their privacy desires via P4P, so that their computer can carry out their wishes.

6.2 DNS Request

The easiest way to ensure privacy of DNS requests is to encrypt all DNS requests and replies – via SSL. However, this fails to prevent any user with access to the DNS server itself from seeing the requests. There have been some ways of handling increased privacy within IPv6[9], but by and large, the only way to prevent your own ISP from knowing your DNS requests is to tunnel all DNS requests through a trusted proxy[19]. Lastly, DNSSEC must be used to prevent DNS-hijacking[7]. This two (or three) pronged attack protects the base of any further internet transactions.

[18]The concept of P4P alone: An all-encompassing, exact language for enumerating the methods of collection, storage, use and transfer of private data that has both a total ordering and is easily represented is simply preposterous. However, a close approximation of it is simply necessary for the desired scenario. This is laughably impossible at worst, and horrendously impractical at best, and sheds some more light on the unreasonable demands of many of today's privacy zealots...

[19]For the truly zealous...

6.3 Routing

After resolution, the user's computer must test each router along the path to ensure that each router attests to the following:

1. Running a known routing algorithm which will destroy packets after their acknowledgment by the next hop.
2. Will not permit access to those packets by a local accessing user
3. Will not store the packets in an unencrypted form
4. Will not forward the packets to any router which does not meet these same requirements.

This adds the requirements that all routers along the path be running a trusted platform and be willing to undergo attestation.

6.4 Website Navigation

Typically, a browser would now contact the server, download the P4P profile, and either allow the user to continue normally, abort, or continue with a warning that the P4P profile was not met. Before discussing website navigation, what if the truly privacy-obsessed user doesn't want the website in question to know their IP at all, but the P4P policy is on the website. There is a chance that the act of downloading the P4P policy will reveal "sensitive" information. Thus, the only solution is to piggy-back all P4P profiles for websites onto DNS lookup requests. The user must have at least a modicum of trust in a DNS server – or must manually type thousands of IP addresses into their .host files[20]. Also, it might be possible for users to get many P4P profiles distributed on CD or other media, but with a greater chance of them being accurate and out of date[21]. Furthermore, as many modern browsers already prevent downloading of remote-loaded images, next-generation browsers will simply prevent the downloading of any objects from servers which do not meet the P4P requirements of the user. Again, tracking of click data, if not specified in the P4P profile as verboten, could be prevented by a trusted proxy.

6.5 Sensitive Data-Handling

Once data has reached the intended server, via conventional encryption, the user can rest safe in the knowledge that the verifying and signing bodies have provided him/her with — that the server in question is fully obiding by their P4P policy, and they cannot transfer data outside the scope that is enumerated in that policy.

[20] A heck of a chore with IPv6...
[21] Attributed to Joan Feigenbaum

We have now fully reversed control of the data, as the firm is now completely bound by the P4P Policy. This does *not* allow for fine grained control of matching user/website preferences: the number of P4P policy negotiations would be on the order of number-users*number-websites! However, assuming that employee data-access control is also specified in the P4P policy, then users also need not fear errant browsing of the customer database[22].

6.6 Infrastructural Details

Despite better planning, a significant amount of encryption is still needed. At the first level of transport of data, users still need encryption to prevent local users from being able to see incoming packets. Multi-user machines might have unencrypted packets coming in, in which another user can see packets that do not belong to them. Also, packets must be secure when moving between routers – no "wiretaps" can be tolerated or the scheme breaks down. Also, any changes in the path must re-initiate the attestation check. Since internet routing is not always assured to be symmetrical, occasionally a server may have to perform additional attestation checks to ensure that its path to the client, which may differ from the client's original path, meets the client's routing specifications.

6.6.1 IPv6

Since spoofing is much more difficult under IPv6, a number of problems in the traditional infrastructure go away. Mail filtering becomes much easier, and with the additions in 6.6.4, SPAM would vanish entirely. The added benefits of no fragmentation would prevent DDOS attacks that rely on IPv4 fragmentation, and the abolishment of NAT would allow for IPSEC and Kerberos authentication as needed.

6.6.2 Java/ECMA Script

Remove the "new window" command. Pop-ups problem solved. Failing that, have companies put their pop-up usage in their P4P profiles and let users decide if they wish to accept pop-ups.

6.6.3 Applets/ActiveX

Applets and ActiveX controls can be screened at loading time by the webserver, and appropriate P4P profiles produced for their usage. With trusted computing, we assume that even more refined tools would be available for end users to stop

[22]This whole scheme relies on the perfection of P4P. Part of the overarching problem of this whole arena is that it is simply ridiculous to expect to enumerate all the ways in which data can be collected and manipulated in a concise, machine readable format.

them from hosing their systems – even the deadly *yes of death* could be prevented by a well-meaning system administrator.

6.6.4 NSSSMTP

The not-so-simple simple-mail-transfer-protocol follows:

- All users must authenticate themselves to mail servers.
- All mail servers must be willing to take responsibility for any mail they forward.
- Mail servers will only accept trusted mail from other servers that are willing to remotely attest to abiding by the same rules.

NSSSMTP provides full accountability for email sent. It does not provide a complete technological solution: legitimate users may still send spam. By the same token, anyone can pay to have a phone installed in his or her house. They may choose to call you at the stroke of midnight every evening. You may choose to block their number or call the police. No technological advance can fully prevent this type of social problem, but full accountability[23] is sufficient to solve the problem.

7 Failed Materialization of PETs

We will refrain from going into too much detail on any one specific proposed privacy enhancing technology. The truth of the matter is that aside from the ones which seek to stem the inflow of unwanted communication, no true privacy enhancing technologies have materialized. The original intent of this research was to closely examine PETs and see where they could be broken; all sufficiently high profile PETs are obviously broken enough as to need no such prodding. Unilaterally, they require massive proprietary server farms, with near one hundred percent uptime, and massive vendor, credit card company, and traditional business support. PETs are a step in the wrong direction. If the ideals espoused by these companies offering PETs are truly that important to the consumer, than a massive shift in the way the internet operates is necessary. The simple fact of the matter is that despite the occasional outcry, recent events[24] indicate that consumers don't care enough to sacrifice convenience.

[23] 555-1212 keeps calling me and breathing heavily! Please block this number.

[24] JetBlue's lack of a fall in operating profits...

8 Are We Asking Too Much?

8.1 Short Answer

Yes.

8.2 Long Answer

These suggested measures are not the only possible measures that could bring about a privacy preserving experience, yet they are not substantially less onerous than any that could. The utopian desires of total privacy are an anathema to the very operating principals on which the internet was formed. Perhaps a lesson is in order from NSSSMPT and ESMPT; we should be far more concerned with accountability than absolute prevention. If we are truly concerned about company usage of private data, let us pass legislation to require some P3P or P4P like system to which all companies must adhere or face stiff financial penalties. This is far easier to enable without bringing a working systems to its knees. By and large, internet users are accustomed to the benefits of having a robust, scalable and reliable network. The higher costs, wait times and inconveniences of a secure, stable, and provably authenticated network will simply be unacceptable to most users. By and large, most consumer outrage is over spam and telemarketing – and not little picayune things like cookie tracking of browsing habits.

9 Conclusion

As P4P is impossible[25], only an approximation of the secure Next Generation Infrastructure could ever be realized. However, from the overwhelmingly large overhead required to support this infrastructure, it is clearly apparent that such a system would never be practical without the support of virtually every human being involved. The barriers to entry are so great, the refinements to business practices and methods are so drastic, and the computational and platform requirements are so heavy that this type of system will simply never come to be. We can, however, learn a few important lessons from this type of system. Notably that little things like SPAM and cookies and Pop-Ups receive an undue amount of attention, drawing away focus from larger issues. However, it is these larger issues which require greater and greater lengths of infrastructural change to prevent. Before we go off installing trusted-computing routers in every backbone link, how about we ask users if they really care if someone knows what they bought at Amazon.com, or if they just want to stop getting ads for cheap Viagra — and then maybe we'd hold off a bit on the infrastructural overkill.

[25] Or impractical...

References

[1] Borland, J. P2p network hidden in kazaa downloads. http://zdnet.com.com/2251-1110-875169.html.

[2] Callahan, K. Hacked by a corporation? http://www.pbs.org/wgbh/pages/frontline/shows/hackers/risks/corphack.html.

[3] Coombs, J. The large scale threat of bad data in dns. http://www.linuxsecurity.net/articles/network_security_article-5514.html.

[4] Council, N. R. The internet's coming of age. http://www.nap.edu/html/coming_of_age/ch1.html.

[5] Force, E. C. I. T. Barriers to ipv6 deployment. http://www.ec.ipv6tf.org/PublicDocuments/Barriers_to_Deployment_v1.1.pdf.

[6] Gibson, S. Shoot the messenger. http://www.grc.com/stm/shootthemessenger.htm.

[7] Group, N. W. Collection of rfcs: Dns security extensions. http://www.dnssec.net/rfc.php.

[8] Group, N. W. Domain names - implementation and specification. http://www.ietf.org/rfc/rfc1035.txt.

[9] Group, N. W. Privacy extensions for stateless address autoconfiguration in ipv6. http://www.faqs.org/rfcs/rfc3041.html.

[10] Group, N. W. Protocol complications with the ip network address translator. http://www.faqs.org/rfcs/rfc3027.html.

[11] Hall, S. Bonzi loses case for obnoxious fake error ads. http://www.marketingwonk.com/archives/2003/05/28/bonzi_loses_case_for_obnoxious_fake_erro

[12] Knowles, B. Introduction to dns. http://www.nluug.nl/events/sane2002/day/abstracts/knowles.html.

[13] Marketing. Historysweep. http://www.historysweep.com/download.jsp.

[14] NA. Does advertising on the web really work. http://www.tek-tips.com/gpviewthread.cfm/qid/550892/pid/717/lev2/83/lev3/86.

[15] Sapolsky, J. The law of leaky abstractions. http://www.joelonsoftware.com/articles/LeakyAbstractions.html.

[16] Singel, R. Jetblue shared customer data. http://www.wired.com/news/privacy/0,1848,60489,00.html.

[17] SPYWARE.CO.UK. Spyware, adware, stealware, stay away! http://www.spyware.co.uk/.