

PRIVACY AND ANONYMITY ON THE INTERNET

Dr. Ian Goldberg
ian@cypherpunks.ca

1. Introduction

The Internet is a ubiquitous tool of communication, entertainment, and commerce. While its decades-old design has held up admirably in the face of applications not then imagined, it offers very little in the way of intrinsic protection for privacy and anonymity.

In this paper, we will examine some of the threats to privacy and anonymity on the Internet. We will also look at a brief overview of technologies that Internet users can employ to help protect themselves.

2. Threats to Privacy and Anonymity on the Internet

2.1. Network-related threats

Let us first look at a high-level view of how information is passed around the Internet.

Every computer on the Internet has an **Internet Protocol address** (usually called an **IP address**). It is a set of four numbers, separated by dots. For example, 199.222.69.109 is the IP address of the computer that hosts the website for CFP 2005.

Traditionally, discussions of network communications refer to the two communicating parties as **Alice** and **Bob**, and we will use that terminology here. **Charlie** will be the name of a third party, not directly involved in the communication between Alice and Bob.

For the sake of example, let's say Alice's computer has an IP address of 206.207.85.33 and Bob's computer has an IP address of 66.151.148.168.

Now suppose Alice wants to send some information to Bob. This could be email, an instant message, a request to Bob's web server, etc.

1. Alice's computer breaks up the information into small chunks, called **packets**. Each packet is labelled with a **source address** and a **destination address**: these are like the "From:" and "To:" on an email message. [The packets also contain other things, such as information about what chunk number it contains, so that Bob's computer can reassemble the packets in the right order, information to distinguish different messages sent by Alice to Bob, etc.]
2. Alice's computer then sends each packet (labelled with 206.207.85.33 as the source address and 66.151.148.168 as the destination address) to her Internet Service Provider (ISP).
3. Alice's ISP looks at the destination address of each packet it receives, and sends the packet to some other machine closer to its destination (this machine is probably a router in the middle of the Internet somewhere).
4. That machine similarly looks at the destination address of the packet, and sends it on to another machine, and so on. The packet makes its way closer and closer to Bob's machine.¹
5. Eventually (after a fraction of a second), Bob's ISP will receive the packet, look at its destination address, and send it on to Bob.

¹ At least, that's the plan. Sometimes there are problems, and the packet gets lost, or goes in circles. But for now, we'll assume things go according to plan.

6. Bob's machine will collect all the packets it receives from his ISP, and reassemble the chunks into the information Alice originally intended to convey.

What threats to privacy can we identify just from this, completely independent of the type of information Alice is sending to Bob? Alice's packets pass through a number of machines on their way to Bob. These include Alice's ISP, Bob's ISP, and all of the machines in step 4. The operators of these machines (and the connections between them), as well as third parties watching those machines and connections, all have access to the contents of the packet. This includes the source address, the destination address, and the chunks of information in the packet.

Given the source and destination IP addresses, how hard is it to figure out that Alice and Bob are the ones communicating? Well, it depends on exactly how Alice and Bob's ISPs have set up their networks. With some ISPs, Alice's machine keeps the IP address 206.207.85.33 always. With other ISPs, Bob's machine may get different IP addresses each time he logs in. Sometimes he's 66.151.148.168, but sometimes he's something else, and someone else entirely (a different customer of Bob's ISP) will get the address 66.151.148.168.

In either case, of course, the ISP definitely knows which of its customers has which address at any given time. They may or may not be willing to give this information to others. [The recent RIAA lawsuits were about this very issue: trying to get ISPs to reveal the name of the customer who had a certain IP address at a certain time.] Parties other than the ISP can also gather a lot of information about Alice and Bob: they can build a dossier (see: Choicepoint) of information they know about "the person with IP address 206.207.85.33". It may not have Alice's full name in it, until Alice uses her computer to register at some website that asks her for it. It may be a little bit harder to compile such a dossier on Bob, since his computer's address keeps changing, but technologies such as cookies, or a new technique called "remote fingerprinting", can be used to identify Bob's machine, even if you don't know its IP address.

In addition to these privacy exposures to third parties, using the Internet also requires that Alice and Bob reveal their IP addresses to each other (since those addresses are in the source and destination address parts of the packets). Web servers, for example, almost always keep logs of the IP addresses of people viewing them. (See Thursday's session "EFF's Best Practices for OSP Logging" for a discussion of how server operators can best configure their machines.)

Those of you paying close attention may note that Alice could simply lie about her IP address when she sends packets to Bob; she could put any value at all in the source address part of the packet, and it will still be delivered. Indeed, this is called **forging**, and can be done successfully, but only in limited circumstances. The problem is that almost all Internet communication is bi-directional; Bob needs to send information back to Alice, and so he needs to have her real address. [Even most Internet communication that, at first glance, would seem to be one-way (like sending an email message) turns out to require two-way communication, since Alice and Bob's computers perform a **protocol** in order to send the message: it actually takes several messages back and forth between them to cause the email message to be sent.]

Alice's ISP may also detect the forging; if it receives a packet from Alice that doesn't have Alice's IP address as the source address, it knows something is fishy, and will likely discard the packet instead of forwarding it. [In practice, only some ISPs actually perform such checks; this is called **egress filtering**.² Doing it can help prevent some kinds of so-called "distributed denial of service attacks".] Forging packets is considered bad form on the Internet; Alice cannot rely on it working in order to protect her privacy.

² Confusingly, this is sometimes called "ingress filtering," even though that would seem to be the opposite.

2.2. Application-related threats

So far, we have only looked at threats that arise directly from the fact that Alice and Bob are using the Internet to communicate, and not anything to do with the kind, or the content, of the communication itself. Services (like email, web, and instant messaging) which use the Internet for their communications are called **Internet applications**, or commonly, just **applications**. In this section, we will look briefly at the additional threats that arise from using certain common applications. Some of these are closely related to the network-related threats discussed earlier, and some are over and above that set of threats.

2.2.1. Email

Electronic mail is one of the oldest Internet applications, and has changed very little in many decades. The most obvious privacy threats with email involve two things:

Email is insecure: As the cryptography community has been fond of pointing out for a decade or more, sending email is like writing your message on a postcard. Anyone in the chain of delivery can not only see who you're talking to (which could be a privacy issue in itself), but also exactly what you're saying.

Email is persistent: As was evidenced at the Microsoft anti-trust trial, email, once sent, tends to live somewhere (on some ISP's backup tapes, for example) forever. *Unlike* a postcard, the protocol for sending email usually causes the message to be copied onto the hard drives of at least Alice's ISP and Bob's ISP, in addition to Alice's and Bob's computers themselves; Alice and Bob have no control over what happens to these copies.

2.2.2. Instant Messaging

Instant Messaging (or IM) networks (such as AIM, ICQ, MSN, YIM, and Jabber) are the communications method of choice among a growing segment of the Internet community. If Alice and Bob are each members of the same IM network, Alice can send Bob an IM as follows:

- Alice sends the message over the Internet to the operator of the IM service (for example, AOL, Microsoft, or Yahoo), with instructions to send it on to Bob.
- The operator of the IM service checks if Bob is currently logged in, and if so, sends the message over the Internet to Bob.
- If Bob is not currently logged in to the IM service, some networks will reject Alice's message, and others will store it, delivering it to Bob the next time he does log in.

The extra privacy risks in Instant Messaging (above the network-related ones) involve the operator: whoever runs the IM network has direct access to all of the instant messages. In March, AOL announced a change to its Terms of Service which included:

"by posting Content on an AIM Product, you grant AOL, its parent, affiliates, subsidiaries, assigns, agents and licensees the irrevocable, perpetual, worldwide right to reproduce, display, perform, distribute, adapt and promote this Content in any medium. You waive any right to privacy. You waive any right to inspect or approve uses of the Content or to be compensated for any such uses."

AOL claimed they never meant for these clauses to apply to personal messages sent over AIM, but the uproar over this change caused them to update their Terms of Service yet again, four days later, including the removal of the line "You waive any right to privacy." But AOL is of course free to change its Terms of Service whenever it likes, as are all the

other IM network operators. It is clear that these operators are *technically* able to read and use your instant messages however they like; if they choose not to look at them, it is only by their good graces.

2.2.3. Filesharing

First, a nomenclature note: filesharing is often confused with the more general term **peer-to-peer** (or **p2p**). A peer-to-peer application is one in which users of the Internet communicate with other users of the Internet. This not only includes filesharing, but also email, instant messaging, and many other protocols. This is because the Internet was designed, by its very nature, to be a network for enabling peer-to-peer communications. All that means is that there's no "master server" which runs the Internet: no one organization is necessary for the Internet to function. So while filesharing is certainly a p2p application, it is only one such, along with many others.

Filesharing networks, as the name implies, allow users to have a **shared folder** on their computer, which other members of the network can search and use in a read-only manner (you yourself can write new files to it, but others can only read files from it). These networks can be used to share any kinds of files, but the biggest uproar is over the sharing of music (and more recently, video) files.

One of the biggest threats to privacy on filesharing networks is exactly the fact that anyone on the network can search your shared folder, and associate your IP address with the list of files contained therein.

2.2.4. World-Wide Web

The world-wide web (or, "the web") is probably where, in practice, most privacy violations take place. This is because for a great many people, the web *is* the Internet.

Websites often explicitly collect personal data from users, and also collect other information about users' visits, including IP address, what pages they viewed and when they viewed them, how long they viewed a given page, etc. For example, Amazon.com was recently granted a patent on inferring information such as age, gender, and birthday from the gifts users buy for people. By correlating information between sites, even larger dossiers can be compiled.

Websites can also provide the user with short files called **cookies**; the user's browser will generally store these cookies, and present them back to that site (and, often, related sites) whenever he visits it. In that way, the site can more easily correlate the information from different visits of the same user, even if the user's IP address changes. For example, all of Google's services, including search, images, maps, news, etc., use a single shared cookie. Any action you perform on any one of those services is correlated with all of the actions you perform on all of the services. As another example, companies that sell advertisements on a variety of web sites can use their own cookies (which will be presented any time the user visits any site with one of their ad banners, or even web bugs) to track users' movements among sites.

3. What can we do?

We have seen a number of threats to privacy and anonymity that arise from using Internet applications, and indeed merely from using the Internet at all. But all is not lost: tools exist to help protect your privacy. These tools go by the collective name of **privacy-enhancing technologies** (or **PETs**), and are in active development, as new threats to privacy emerge.

3.1. Application-related solutions

In this section, we will take brief looks at one sample privacy-enhancing technology for each of the Internet applications we have examined.

3.1.1. Email

Privacy in email can be achieved by using **Mixminion**³. Mixminion is the fourth generation of **anonymous remailer**. It provides the following privacy protections:

Anonymous sending: Alice (who has Mixminion installed) can send anonymous email to Bob (who doesn't have to have it installed). Bob will receive the email, but not know who sent it.

Anonymous receiving: Alice can arrange to have a **pseudonym**, or **nym**. Mail Bob sends to that nym will be delivered to her, but Bob will not know Alice's identity.

If Alice and Bob are both running Mixminion, of course, they can combine these features, and are able to have conversations where neither one knows the other's identity.

Mixminion improves over previous anonymous remailers by expanding the protection Alice gets against people trying to foil her anonymity.

Mixminion relies on there being a number of servers around the Internet running the Mixminion software; Alice can't get her anonymity without their cooperation. (Though they do not themselves end up learning her identity.)

3.1.2. Instant Messaging

People expect instant messages to behave like private person-to-person chats (in the "Real World"). **Off-the-Record Messaging** (or **OTR**)⁴ provides the following privacy protections for instant messaging:

³ <http://www.mixminion.net/>

⁴ <http://www.cypherpunks.ca/otr/>, a project by the author and Nikita Borisov.

Encryption: No third party can listen in on Alice and Bob's conversation.

Authentication: Alice and Bob are assured that they're actually talking to each other, and not an imposter.

Deniability: Bob could report to Charlie, everything that Alice said to him. But he can't *prove* it. Charlie will have to take his word for it.

Forward secrecy: If Bob or his computer is compromised today, the conversations he had with Alice yesterday are not revealed.

Both Alice and Bob need to have OTR installed in order to communicate privately, but there is no infrastructural requirement, as with Mixminion. If Alice and Bob are the only two people in the world running OTR software, they still get the full benefit.

3.1.3. Filesharing

WASTE⁵ is a filesharing network offering the following protections:

- A small set of trusted friends all run the WASTE software, and can search and read each other's shared folders.
- No one outside that small set has any access to the folders at all; communication between WASTE users is encrypted, so eavesdroppers cannot see the contents of the folders.

Users of WASTE form small, separate filesharing networks among groups of friends. One group of WASTE users has no access to the shared folders of another group.

⁵ <http://waste.sourceforge.net/>

3.1.4. World-Wide Web

In order to ameliorate some of the privacy problems encountered on the web, users can run **Privoxy**⁶. This is a privacy-enhancing web proxy which runs on your computer, and filters your web content (much like "parental control" software does, to filter out porn). But instead of porn, Privoxy filters out privacy-endangering aspects of web pages instead, including cookies, web bugs, ads, and more.

Privoxy offers powerful and flexible control over what gets filtered and modified, surpassing the rudimentary image and cookie management mechanisms offered in most modern web browsers.

Privoxy is run only on the user's computer; there is no infrastructural requirement, nor do the web sites with which you are communicating need any special software.

3.2. Network-related solutions

In order to protect your privacy on the Internet, it does not suffice to use only privacy-protecting applications; you also need to address the network-related threats. OTR, for example, allows Alice to send instant messages no one except Bob can read, but it still reveals the fact that Alice is talking to Bob.

For full protection against threats to privacy and anonymity on the Internet, you need to use *both* application-related *and* network-related privacy-enhancing technologies.

One such network-related PET is **Tor**⁷. Tor will be more fully described elsewhere in this Workshop, but the extremely-high-level view of it is this:

⁶ <http://www.privoxy.org/>

⁷ <http://tor.eff.org/>

- There are a number of servers on the Internet running the Tor software. These servers are called **onion routers (ORs)**.
- Suppose Alice wants to send information over the Internet to Bob. She first constructs a **circuit** of ORs. This is a list of ORs with the property that each OR in the list learns which OR is before it, and which is after it, but none of the others.
- Alice gives the information to the first OR in the circuit. That one passes it to the next one, which passes it to the next one, and so on.
- Eventually, the last OR in the circuit gets the information, and passes it on to Bob.
- Bob only sees the IP address of the last OR in the circuit, not that of Alice. He sends his replies there, and they travel backwards through the chain until they reach Alice.

In this way, Alice (running Tor software) can communicate with Bob (not running any special software) over the Internet, without revealing her IP address to him.

4. Conclusion

Alice's use of the Internet is fraught with threats to her privacy and anonymity. Data about whom she's communicating with, what she's saying to them, what web pages she's perusing, what weblogs she reads, what she buys online, what comic strips she enjoys, and many other things, as well as a plethora of personally identifying information, is whizzing around the world, often at the speed of light.

Alice should be rightly concerned about the availability of her information to parties out of her control, and about the purposes for which that information is used. But by using

network-related PETs like Tor as well as application-related PETs like OTR or Privoxy, Alice can go a long way to protecting her privacy and anonymity on the Internet.