Privacy Enhancing Technologies

By the Radical Tech Collective
http://groups.fsf.org/wiki/RTC

Why Anarchists Need Privacy Technology

In the history of humanity, there have been events which have proved impossible to revert, points in time that have drawn a bright, clear line between then and now. The first of those events was the dawn of agriculture and the adoption of sedentary, urban lifestyles. The second of those events was the Industrial Revolution and the dawn of automation of labor. These events are described as Revolutions for a reason: they permanently and decisively displaced a prior method of operations in favor of a new and radically better one. We stand now at the dawn of the 21st century, able to look back all those millennia and see how far we have truly come.

We stand now at the dawn of the 21st century, and at the dawn of another such revolution. This revolution, like all those before it, has not happened overnight, but has been brewing in relative obscurity for nearly half a century. Only now is it beginning to take its place in the minds of its contemporaries, only now are we beginning to see what it is truly capable of. This is the Digital Revolution: The displacement of old means of communications, information storage, archive, distribution, and creation.

Like all revolutions, the chief area that will feel the alterations of progress is that of the infrastructure. Just like irrigation become important at the dawn of agriculture and trains and telegraphs became vital at the dawn of industry, the rise of a digital infrastructure will alter the face of the Earth and of humanity.

The anarchist movement, in its modern form, is, for the most part, a product of the Industrial Revolution. Like other struggles of its time, the anarchist movement used the new infrastructures to carry the Revolution out of the realm of theory and into the streets. The Industrial Age saw a wave of revolutions, crashing like waves on the status quo of Europe and America, showing the castles of the ruling class to truly be just so much sand. These revolutions were possible only because of the neutrality of infrastructure -- a train would carry, with the same force and speed, the troops of the State and of the people.

However, the Digital Revolution differs from the Industrial Revolution in the most vital way possible -- where the Industrial Revolution built dumb technology, impartial and non biased, the Digital Revolution has constructed technology able to reason and implement logic. Thus, the infrastructure of the Digital Age is not dumb or impartial, but controlled by the smart technology that runs it.

Combined with the remotely-controlled nature of technology in the Digital Age, this raises rather pressing questions for anarchists. How can we use this new technology to reach our goals when it is capable of rejecting our use of it? How can we share information over it when it can eavesdrop or interfere?

The answer to the first question is, use software that is free -- as in, allowing the user freedom. That way, no external entity can revoke your right to access a computer. This is not the subject of this essay, but ample information on this particular answer can be found in other RTC materials, or on the Internet at <http://www.fsf.org>.

However, the second question still stands. How can we use the Internet as an infrastructure, akin to roads or railways, if it is controlled by those hostile to our movement, the ruling class?

The answer is, make the smart technology dumb. Force it to treat all users of the infrastructure as equal and equivalent by rendering it unable to discrese between any one user of the network and any other. Furthermore, render the ruling class's control over the network moot by employing technology that allows us to use it without identifying ourselves to the network.

The answer is, in short, privacy enhancing technologies.

PET's allow us to use the Internet anonymously, without revealing our identities or locations to the ruling class, even though they control the Internet. PET's allow us to transmit data over the Internet encrypted, ciphered so that only the intended reader can make sense of the entropic collection of bits that actually traversed the net. PET's allow us to use the Internet as a dumb tool, one that does not have the possibility of betraying us.

Already, some have started awakening to the promise the Internet provides. Operating with the assumption the state or ruling entities will not act against them, and exposing themselves to great personal risk, they have proved the potential inherent in the new resources available to us.

One example is the psuedo-group, the mononym, Anonymous. Anonymous is an Internet-only entity -- it has no concrete, single manifestation or headquarters in the physical world. Anonymous is completely decentralized -- it has no organization, control or command structure, no hierarchies, and no leaders. Anonymous has no political platform or even concrete goals, but because of its structure, it could be said to be anarchist. Using only the resources given by component parts, without command of any treasury or war chest, Anonymous was able to rally support for and launch an assault against the Church of Scientology -- a powerful, centralized, capitalistic enterprise, with many a headquarters in the physical world, with a great deal of organization and command structure, with leaders and hierarchies. The Church of Scientology is nominally apolitical (insofar as any capitalist enterprise is), but because of its structure, it could be said to be an estimation of the ruling class.

Anonymous launched its first strikes with the very essence of diversity of tactics. Mainstream Internet news channels were inundated with propaganda,

propaganda produced by nameless members of the mononym. The Internet manifestations of the Church of Scientology were immediately made the target of various Denial of Service (DoS) attacks, rendering them inoperable. Within the first weak, the all-out offensive made Anonymous a clear victor in the initial sorties.

While these direct actions against the Church of Scientology continued, individuals of Anonymous began organizing, networking with other opponents of the Church of Scientology and gaining strength. Within a month, Anonymous launched coordinated protests across the entire world, and as the sun rose over New Zealand to when it set over California, it shone on Anonymous.

Without any offline presence save the decentralized and often solitary action of its sympathizers, Anonymous was able to summon hundreds or thousands in international cities to give voice to its cause. As a result, the Church of Scientology lost innumerable amounts of resources defending itself against the broad array of tactics employed by Anonymous, and risked losing its status as a religion in various regions. Remarkably, Anonymous targeted the Church largely randomly, in response to a copyright claim that deleted a YouTube video. It is left to the imagination what could have happened if the individuals comprising Anonymous had acted in the interests of what they thought was a worthy cause, not merely a source of amusement.

Drifting back towards the mainstream, we find an assortment of democracy movements and small-scale rebellions given voice and a world-wide view merely through the Internet. When Buddhist monks protested in Burma, they were only cut silent when the ruling junta cut off Internet access entirely. And when Iranians felt their democratic system had been compromised, they turned to the Internet to vent their anger. Even though they relied on a proprietary microblogging website, Twitter, to be their infrastructure, their goals and the goals of Twitter's ruling class were fortunately convergent, and thus they were able to broadcast their struggles to a world-wide audience. Another group that relied on Twitter -- the 2008 RNC protesters -- might not be so lucky, and certainly will not be in the future.

Make no mistake, the making of smart technology dumb is an arms race. But it is not one of steel and guns -- it is not one where the ownership of the means of production has any meaning at all. It is a race of ideas. As long as we can out-think the state, we can beat the state. And we, as dynamic agents of change, will always be able to out-think the static, close-minded, bureaucratic state.

Cryptography and anonymity. Full-disk-encryption and TLS. dm-crypt and LUKS, encfs, ecryptfs, Off-the-record messaging, PGP, GPG, i2p, GNUnet, Freenet, and Tor. Learn those words and study them, for they are the privacy-enhancing technologies of this iteration of the arms race. One day they will be obsolete, replaced by the next generation of technologies that out-think the state, but until then, make use of them. Anonymize your network traffic,

your web browsing, your instant messaging, your emailing. Encrypt your data, your hard drive, your folders, your network traffic, your emails, your instant messages. Encrypt everything.

Because the smart technology can only spy on what it can see, and to the all-seeing eye of the state, these PET's raise a black fog that is blinding. If we work together, if we remain vigilant and guarded, we can construct structures and communities in cyberspace that will be like a shadow to the ruling class -- far more impenetrable than any organization or group operating inside the fragile, "real", physical world, a world in which brute force and strength can prevail.

But the Digital Age is here, and the basic logic and mathematics of our PET's cannot be strong-armed into submission. The state and its tactics are obsolete. We can and will evolve faster and better than it can. We must, if we are to survive.

The future is ours, if we make it so.

Anarchy and Anonymity

The advent of the networked age offers up innumerable possibilities for the future. While the advent of the internet and the computer definitely benefit anarchists, the state can just as easily leverage the same resource against us. We, as anarchists, need to utilize these resources to the maximum of their capacity in order to make our movement relevant and effective in the 21st century.

Many anarchists are iniitally critical of anonymity. Anonymity looks, on the surface, to be akin to the dehumanization that the state rampantly spreads. To the contrary, anonymity is the best way of ensuring the fair spread of ideas, the fair treatment of ideas, and the efficiency of the movement.

We must begin with a discussion of anonymity. Anonymity comes in two forms: perceived anonymity and technical anonymity. Perceived anonymity is exactly what it seems -- a percieved sense of anonymity. Technical anonymity is the actual achivevement of a state in which it is unable to trace the message to its speaker. A good example of purely perceived anonymity is a forum on the public web (www from here on) that does not store names -- while participants cannot readily identify one another, a controller of the network is easily able to do so. An example of purely technical anonymity is a forum on the Tor in-proxy .onion network ("onionland", "torland", or ".onion" from here out) that does store names. While participants are capable of readily identifying themselves, no one is capable of finding the actual location of any speaker. The combination of these two states is "pure" or "true" anonymity -- a good example of this is an anonymous bulletin board that does not store any names. Note that to be truly anonymous, no names can be involved -- otherwise, the space is merely a technically anonymous space with psuedonymity, or mononymity (having one name, akin to the 'Anonymous' of *chan websites).

Percieved and technical anonymity exist for different reasons and fulfill different needs. In brief, perceived anonymity facilitates the transmission and judgement of ideas, while technical anonymity protects the sources of those ideas.

Technical anonymity is advantageous to anarchists for obvious reasons. We are currently in the midst of the green scare, and while that will eventually subside, state persecution of anarchists will always be a problem (until the state is dissolved). In such an environment it is clearly beneficial to avoid detection by participating anonymously within internet communities, and to avoid the easy mapping of anarchist networks. Even in the abscence of a state, anonymity is a facet of the right to personal privacy, and will still be advantageous.

The transition from a public system to an anonymous system for anarchists will no doubt be a difficult one. While www sites are

beneficial to have in order to allow non-anarchists a glimpse into the movement, the planning of any action of risk should be completely anonymous. To reflect this, maintainers of networks (such as infoshop.org, indymedia, hackbloc, etc.) should move towards a dual-gated system in which information can be accessed in both www and anonymous ways. It's trivial to configure a Tor hidden server to move an indymedia node to Onionland -- but more than just protect the sources of information, this will allow those with sensitive information to be able to publicize it without fear, such as videos of police brutality or state repression. A good example of such a gateway is Wikileaks, a project to anonymously publish sensitive state or corporate documents. Wikileaks was actually censored in the USA, with the effect that its domain name (the human-readable form of a site address, in this case wikileaks.org) was pulled from American DNS (Domain Name System) servers. Wikileaks remained accessible to some, but not many. However, Wikileaks is also accessible as a Tor Hidden Service, at the address <http://gaddbiwdftapglkq.onion/>. This allows Wikileaks to remain reachable anonymously, and for it to publish documents obtained via extremly secure cryptographic links. Regardless of the legal status of Wikileaks in any country, that gateway will always be reachable, because it does not betray the location of the Wikileaks server. For more details on the Tor .onion network, refer to RTC's documentation of Tor.

Percieved anonymity is often maligned by those against the "depersonalizing" effects of technology, but it is perceived anonymity that shows exactly when depersonalization is a positive thing. Percieved anonymity is the perception of being anonymous; this perception, like any other, depends on the perciever, and as such perceived anonymity differs between people. To a person who is ignorant of the traceability of network connections, a site like anontalk.com might be anonymous, but to someone who is more security-concious, perceived anonymity and technical anonymity can be indistinguishable.

The benefit in perceived anonymity comes with the lack of ownership of ideas. When an idea is connected to a given person, human bias always comes into play, at concious or unconcious levels. This bias can stem from any number of sources: race, gender, sexual preference, past statements, lifestyle, social class, etc.. The goal of perceived anonymity is to make all ideas initially equal to one another, allowing the ideas to compete on the basis of their own merit.

This pre-judgement of ideas is a major problem in activism. The activist anarchist community faces constant criticism for being alienating or unavailable, a haven for the priviledged with under-representation of people of color, women, non-punks, or any other group. Above that is the potential for authoritarian cores to develop in anarchist organizations, or for notable individuals to

become targets of hero-worship, or for simple groupthink to set in and cause stagnation, or for cliques to form that alienate all others.

Anonymity allows for this dynamic to be swept away. It is impossible to assert privilege if your idea is not connected to you. It is impossible to exclude based on ethnicity if the ethnicity of participants in a dialogue is unknown. It is impossible to reject ideas that do not have a source inside the clique if the source of the idea is unknown. And is it impossible for an authoritarian core to shoot down ideas they disagree with if they have no real argument against the idea. Additionally, anonymity makes it easier for new blood to insert itself into processes, allowing for new viewpoints and more flexible organizations. This increased insight compounds the benefit of ideas lacking owners and creators who would tend to sheperd them "their" way, claiming their source-ness as legitimacy to rule over "their" idea. Perceived anonymity allows ideas to break free from masters and re-insert themselves into an evolutionary environment where the truly best ideas survive. Even mob rule is impossible if you don't know how many are in the mob.

Not many people have experience with truly anonymous systems. A good example of a mononymous system working quite well, however, is Anonymous (big-A). Beginning in March 2008, Anonymous was able to organize a worldwide resistance to the arbitrarily-chosen Church of Scientology. "Members" of Anonymous who joined into the mononym acted independently, congregating anywhere possible on the internet and using a diversity of tactics in order to damage the Church. Some of the mononym launched cyber-warfare attacks on the CoS, necessatating greater expenditure in cyber-defence, while others successfully rallied the general public to their cause, while others attempted to court fringe groups such as fundamentalist Christians, while others attempted to gain media exposure.

Even with this massive initial success, the Anonymous mononym failed to accomplish its objective of dismantling the CoS. The primary reason seems to be the very fact of Anonymous's mononymity. By identifying and assimilating into Anonymous, a mononym-adopter adopted not just the name, but the characteristics that a mononym was to represent. And the mononym is capricious. When lulz were no longer to be had in the fight against Scientology, the general population of the mononym dropped the assault, leaving only a dedicated, radicalized core remaining. Were Anonymous truly anonymous, and the anti-scientology movement without any names, the momentum gained early on would not have been lost as soon as it was, due mostly to the lack of purist factions within the mononym counter-attacking the movement due to perceived un-mononym-ness. These appeals to mononym personality would obviously be unavailable in an anonymous system, and additionally, the more grating facets of the mononym would not be presented to the general public. These personality aspects

characterized the movement as more of a joking lark than a serious assault against the CoS, losing the mononym and the movement a great deal of credibility.

While numerous anonymous networks exist, such as Tor, I2P, Freenet, and GNUnet, there is as of the time of this document's writing a lacked absence of anarchist sub-networks. As the world becomes increasingly digitized, and as adoption of digital life increases, anarchists cannot afford to remain in the past. Disruptive technologies (such as txtmob) have proven their worth in the past, but adoption has been slow. If anarchists are to transfer their message into the new world, they must adopt anonymity in order to not only safeguard their movement, but to improve it.

Introduction to Cryptography

Put simply, cryptography is the practice of creating secret messages that can only be understood by someone in possession of a given secret -- in cryptography, this secret is called a "key". There are two basic types of cryptography -- symmetric and asymmetric.

Symmetric cryptography uses one key to both encrypt and decrypt. It's the simplest form of cryptography to understand, and is useful for encrypting static things, like data, that only need to be accessed by one person. But in order to use symmetric cryptography for communication, both the sender and receiver of a message would have to know the key- and communicating the key without it being intercepted would be a logistical nightmare. To avoid this, we use asymmetric crypto for communications.

In asymmetric crypto, one key encrypts while another decrypts. In PGP, for example, everyone has a public key, which encrypts messages, and a private key, which decrypts them. You can tell your public key to everyone, so that they can use it to encrypt messages to you, knowing that only you can decrypt them. This provides privacy in communications. You can also use your private key to encrypt messages, allowing everyone who has your public key to decrypt them and be absolutely sure that you were the sender. That way, asymmetric cryptography can also provide verification that you are who you say you are.

There is also a combination of the above two forms, called a "hybrid" cryptosystem. In this system, asymmetric cryptography is used to secure the key for symmetric cryptography. This is what the program GnuPG uses for communications, for instance. Hybrid cryptosystems are beneficial because they combine the key-distribution simplicity of asymmetric crypto with the efficiency of symmetric crypto.

In cryptography, an "algorithm" is a way of doing encryption and decryption. Algorithms can be symmetric or asymmetric.

The most notable asymmetric crypto algorithm is RSA, named for the initials of its inventors. RSA is the backbone of most communications crypto like SSL and TLS (the backbone of https, amoung other things) and also of GnuPG. RSA was unavailable to the free software community until recently, when its patents expired.

The most notable symmetric crypto algorithms are AES (or Rijndael), Twofish, and Serpent. These three were the top finalists (and one of them, the winner) of the 2001 AES competition, the goal of which was to select an algorithm as the US Government's Advanced Encryption Standard. Rijndael won, claiming the title of AES. However, each of the three is a powerful algorithm. AES is the fastest (meaning it is the easiest to brute-force, while that is still very, very unlikely to happen) and the simplest of the three, which has allowed some attacks to be launched against it successfully -- however, due to the other two's relative lack of popularity, successful attacks against them

could go unnoticed by the open crypto community for a longer period of time than AES. All three have 256-bit keys, meaning there are 2^256 possible permutations of keys.

While cryptography can be off-putting initially, it's easy to learn, and the freedom-loving spirit of cryptographic researchers tends to guarantee that information about crypto is freely available. Anarchists must embrace crypto in order to maintain security and privacy in a digitized world, and while it is not necessary, it is best for us to understand our tools, for if we do not, they may end up using us instead of the alternative.

Overview of Crypto Technology

Crypto, overall, fulfills two purposes -- securing data at rest, and securing data in motion, or put simpler, storage and communications. Cryptographic software can be separated along this line.

Communications Crypto

GnuPG

GnuPG, or GNU Privacy Guard, is a free software replacement of the original public-key (asymmetric) crypto program, PGP (Pretty Good Privacy). While it has numerous functions, being a sort of crypto jackknife, its main purpose is encrypting and signing emails.

GnuPG implements the OpenPGP standard -- a document specifying how programs should interact when transmitting encrypted messages. This means that if you use GnuPG, but your friend uses the original PGP software, the two of you can communicate. However, like most GNU programs, GnuPG extends the standard -- so if possible, try to get everyone you communicate with to use it instead of other OpenPGP implementations.

GnuPG doesn't limit itself to one sort of data -- it can operate on arbitrary data. This means that while you are fully able to encrypt and sign emails with it, you're also able to encrypt individual files, or sign an archive to ensure it isn't tampered with. The latter feature is widely used in the free software world to secure software repositories.

Off-the-Record messaging

Instant messaging carries a slightly modified threat model than email, so GnuPG is less useful for securing it. While some IM programs can use GnuPG, most use the OTR protocol, which is uniquely adapted to securing IM.

Initially a plugin for the free software IM client Gaim, OTR has expanded into a library that numerous free software IM clients use, including Pidgin (which was once Gaim), Adium, and Trillion. OTR provides authentication, allowing two people to prove they are who they say they are, encryption, allowing them to talk privately, and deniability, allowing them to assert that any given message is not actually theirs.

While all anarchists should use the free, decentralized XMPP protocol (for more on that, see jabber.org), many still use MSN or AIM, and thus implicitly rely on corporate ethics to limit spying. Microsoft in particular has shown that it logs information about who is talking to whom in its system -- but there's no way of knowing if that's all they do. But, with OTR, these networks can be used with a modicum of security.

Communication Crypto Note

One thing to remember when using the above crypto technologies is the necessity of verifying authenticity of a key. Both GnuPG and OTR use asymmetric crypto, meaning that for Alice to encrypt a message to Bob, she must have his public key. It is possible for an adversary to trick you into using a public key that does not belong to its advertised owner. Always verify key fingerprints and signatures with the person you plan on communicating with.

Data Crypto

Data Crypto Overview

The common thread of all of the following technologies is that they provide a GNU/Linux filesystem with transparent encryption -- programs can access files inside the encrypted filesystem without knowing that it is encrypted. However, the following programs each provide that filesystem in a different way, or with different features.

GNU/Linux Full Disk Encryption

The "nuclear option" of data crypto is FDE -- full disk encryption. In an FDE environment, the entire filesystem hierarchy is encrypted, allowing no opportunity for cleartext data to be written to disk. This minimizes the chance of information leaks at the price of some overall performance.

dm-crypt is the kernel Linux's built-in crypto workhorse. Since the crypto operations occur in the kernel, it's faster than userspace crypto programs (all things being equal), and it supports all the crypto algorithms and modes that the kernel does.

On Ubuntu (specifically, Ubuntu's alternate install CD) you have the option of setting up full-disk encryption with dm-crypt in the installer.

encfs and ecryptfs: directory-level crypto

If you don't want to encrypt your full system, or have encrypted your full system and want to more crypto, directory-level crypto allows you to easily set up encrypted directories (obviously) and let them grow as they need to.

The only difference between encfs and ecryptfs is that while encfs is a user-space filesystem written with the FUSE (filesystem-in-userspace) framework, ecryptfs is a true, in-kernel filesystem. As a result, encfs has to rely on itself or userspace libraries to support cryptographic operations, and is slower than ecryptfs, which, since it is in kernel-space, can access all the algorithms the kernel implements, and is marginally faster.

However, encfs -- being just another program -- does not require any

special permissions to use. This means you could set up an encfs directory on a system you don't have root access to. ecryptfs is faster, but, since it's a filesystem like any other, an ordinary user of GNU/Linux can't mount it -- only the administrative user can. On most home systems, the person controlling the ordinary user account also controls the administrator account, but of course the software doesn't know that.

ecryptfs in particular is being heavily integrated into Ubuntu in later releases -- all users have the option of creating a private directory encrypted with ecryptfs, and users installing from the alternate install CD can encrypt their entire home directory with ecryptfs.

Truecrypt: Deniable encrypted containers

Truecrypt differs substantially from the other crypto technologies discussed here. For one, it is not free software, but "semi-free" software -- users have some freedoms, but not enough to render it truly free (one of these is access to the source code, so Truecrypt can be verified as secure). Also, Truecrypt creates "containers" -- single files, that cannot grow in size, that it mounts onto the filesystem.

However, Truecrypt has one feature in particular that renders it worth mentioning and even using -- deniable crypto. Truecrypt can create "hidden" Truecrypt containers inside another Truecrypt container. Nobody can tell the hidden container is there just by looking at the outer container -- except, of course, anyone with the key for the hidden container.

This is obviously a powerful feature. The state can't interrogate you for your encryption key if they don't know where your encrypted documents are. If you are coerced into handing over your keys, simply hand over the key for the outer Truecrypt volume -- it will appear to be a completely ordinary container. This works because encrypted data is identical to random data, so even if space is unused inside a Truecrypt container, it won't appear as unused. That space can be overwritten with the second, hidden, container.

If you have a need for truly vital data to remain secure, use Truecrypt and hidden containers. But for most other things, Truecrypt is sub-optimal due to its license and its lack of on-demand growth.

Overview of Anonymity Technology

Privacy enhancing technologies have two purposes, of which they may fulfill one or both: preventing data from being read, and preventing data from being uniquely identified as belonging to a particular people or group of people. Anonymity technology attempts to solve the latter problem.

Anonymity systems are designed so that it will be possible for a source to communicate with a destination with one or more of the following assurances:

- The destination is unable to identify the source (source anonymity)
- The source is unable to identify the destination (destination anonymity)
- An observer is unable to tell that the source and the destination are communicating (communication anonymity)

Not all of these attributes need to be present in the system for it to be secure.

There are other attributes that may be present in the system, but are not needed for basic anonymous communications:

- Censorship resistance -- in the context of anonymous filesharing, the ability of the system to shield individual files from censorship.
- Circumvention resistance / deniability -- the ability for users of the system to plausibly deny using the system, or to use the system without being detected.
- Friend-to-friend operation -- in peer-to-peer systems, the ability for a user to connect only to a certain set of peers which they have selected.

The Centralization Problem

The earliest "anonymity systems" were exceedingly simple. The most notable were remailers and, later, proxy servers. Remailers did exactly what you'd expect -- they re-sent email messages sent to them (this was in the days of the Internet before spam became widespread, so abuse was less of an issue). Proxy servers, initially used by some ISP and corporate networks, were often misconfigured, and left as "open proxies". This allowed anyone who could connect to the proxy to use it as a means to use the Internet.

Ultimately, however, these single, centralized hubs of information did not anonymize anything. While some proxy servers or remailers didn't keep detailed logs of accesses, their Internet providers could have, or their lines of communication could be tapped. The lesson was clear after the Church of Scientology demanded that a remailer hand over logs exposing critics of Scientology, and was successful. Centralized systems could not successfully safeguard anonymity.

The first step to safeguard remailers was twofold. First, remailers stopped being pseudonymous, and started being anonymous -- instead of keeping a record as to who a particular user was, the field was stripped off the email and sent without any logging. Second, remailers began to be "chained" -- a remailer would be used to send mail to a remailer, which would send mail to a remailer, etc., until the recipient was reached. This tactic was later applied to proxies.

Mix Networks and the Future of Anonymity Systems

However, when David Chaum published his paper on the hypothetical "MIX-net", the anonymity world changed. While never being literally implemented, Chaum inspired researchers around the world to create anonymity networks -- systems of nodes, collaborating so that any one of them could anonymously communicate with any other. Combined with a veritable revolution of cryptographic technology, specifically the decriminalization of "strong" cryptography, researchers were free to design anonymity networks that gave the people back the rights that the digital age had initially allocated to the ruling class.

Some of the first networks created were Mixmaster, an anonymous email forwarder, devoid of the problems of the single-use remailers, and Onion Routing, a system of forwarding communications in successively encrypted containers, or "onions" -- the precursor to today's Mixminion and Tor networks, respectively.

Anonymity network research continues to this day, as part of the overall cryptographic arms race. Like all arms races between the state and freedom, it is inevitably biased towards one side.

Notable Anonymity Technology

There are vast numbers of anonymity systems. They range from file shares to publication points to internet overlays to entirely new paradigms of networking. This chapter will go over some of the more well-known and tested ones that actually provide some guarantee of anonymity. All of the anonymity systems mentioned here are Free Software.

Tor: The Onion Router

Overview

Onion routing is an anonymity protocol that was one of the first to be implemented on the new internet. It works quite simply: traffic is sent from the origin to a node, which sends it to another node, which sends it to another node, and after N nodes have been routed through, the traffic is forwarded to the destination. Tor works by encrypting traffic with AES-128 and a Public Key system, then forwarding traffic through three nodes before the traffic is decrypted and leaves the network to go to its destination. Tor was originally funded by the US Navy and is still used by the US Government for forwarding sensitive information out of potentially hostile networks.

New Terminology

Entry Node: The node at which traffic enters the network. This can be thought of as the first node in the three-node chain, or the SOCKS interface listening on your computer. While traffic is encrypted on your machine, but an Entry Node could determine your IP address (if it wasn't for Tor's protocol, but we'll get to that).

Exit Node: The node at which traffic leaves the network. The last layer of public key encryption is peeled off, the AES encryption is removed, and traffic is forwarded to the destination. The exit node can read any of your traffic that is not encrypted from you to the destination. This includes http (web), ftp, telnet, Instant Messaging, and IRC, and of course passwords to any of these protocols. Do not send plaintext passwords over Tor!

Perfect Forward Secrecy: Tor has a property called "Perfect forward secrecy" for it's forwarded communication. Perfect Forward Secrecy (or PKS) means that only the last link in the chain can read any of the data. You can think of this like an encryption onion. You build up the onion on your end, wrapping an inner core of data with several layers of public key encryption. When the first node gets the onion, it peels one layer, and the next node does the same, until the exit node is reached and the data is forwarded.

Bad Node or Bad Exit: An exit node that takes advantage of its

link in the chain to sniff data. A famous bad exit was able to sniff email passwords for thousands of government embassy logins. Bad exits might sniff your data, or they might modify it to insert advertisements or hostile data to break your anonymity.

Strengths

Tor's strength comes from its uniformity. At any point in the chain besides the exit, no node knows where in the chain it is. This means that encrypted traffic from you into the entry node and from the entry node into a circuit node is just traffic, and its origin can't be determined. Tor protects against forms of Traffic Analysis, an attack on anonymity that involves watching connections. If an adversary could see all the connections of all the Tor nodes in the world, they could break Tor. But since there are Tor nodes all over the world, in various countries with various diplomatic status between them, that won't happen. Previously it was thought that Tor would be trivial to break due to the low number of nodes, but since then Tor has grown from having 400 nodes to 5000 nodes, with an average of 1000 online at any given time.To strengthen your anonymity and everyone else's, run a Tor node. Not only will this help the network, it'll make your anonymity stronger, as traffic coming from you could be originated from you OR forwarded by you.

Tor is also low-latency. While it might not be low-latency compared to your normal net connection, it is certainly low latency compared to other anonymity systems, like Freenet or GNUnet.

Possibly the biggest advantage of using Tor is this feature: Hidden Services. Hidden Services are just like any other service on the net: IRC servers, websites, shell servers, chat servers, anything that runs on TCP (and most of the net runs on TCP), but with one important difference. Hidden Services are anonymous. With normal websites, you can always find the owner, and possibly persecute/prosecute him for his speech, but with a Hidden Service, she's hidden behind Tor. Plus, even plaintext content is safe, because ALL traffic is encrypted end-to-end with a Hidden Service.

Hidden Services are like a whole new internet. There's a culture on the ones that are open to the public of anonymity and free information. The author of this article was inspired many times by events or statements on the Hidden Service scene, and without the environment it provided, might not have written this article. That being said, hidden services allow many that are persecuted to engage in behavior that many in society find utterly disgusting. True

freedom isn't for the faint of heart.

Weaknesses

While this isn't a weakness of Tor (there is no way to implement this in any system ever), the biggest drawback of Tor is the lack of trust in the node operators. While this won't compromise anonymity, it can compromise data. While using Tor, make sure to take the same precautions as you would on any other untrusted network. Encrypt everything. Passwords should be sent in SSL or secure hashed form, messages should be encrypted. While bad nodes on Tor aren't nearly as prevalent as good ones, there is no way to know if an exit node is sniffing your traffic.

Tor is also vulnerable to a few classic attacks on anonymity networks, including the "Giant Overseer" attack and timing/correlation attacks. The Giant Overseer attack is simple: If the adversary can see all traffic on all nodes of the Tor network, the game is over. But this attack isn't really feasible unless the Illuminati (exists and) wants to break Tor, or if one government took over the entire world. A more potent attack is a timing attack: If I watch Bob sending a request for a file, and then observe Alice getting a request of equal size, followed by Alice sending a 300MB file, if Bob gets a 300MB file, there is a good chance it might be Bob talking to Alice. This could be defeated with padding (making all data distributed on the network use a certain amount of data all the time), but that would be impractical and severely impact Tor's speed. This attack would be very useful in discovering the location of a Hidden Service, but it would take a very large amount of resources to successfully complete.

Although this attack is impractical due to the US Navy's endorsement and the US Government's (and other governments) widespread use of Tor, Tor is extremely vulnerable to attacks on centralized resources. Tor nodes look up hidden service and node addresses via a centralized directory, and while the directory is mirrored, only a few servers are "authoritative" and have supreme say over the network.

Closing Thoughts

Tor has taken a lot of flak from people who are pissed about the ability of exit node opers to sniff data, but it should be kept in mind that sniffing data or the potential to sniff data does not compromise anonymity. Encrypted traffic is the only truly safe content when using Tor to access public-web servers. The Hidden Service feature, a main focus of Tor development, is a great boon to radicals, and in fact, Tor has become home to ALF communiques from all

around the world and Chinese dissident speech. Certain Indymedias also run a Tor portal, allowing users to have hidden-service level anonymity, but communicate with those that don't. Another site that allows this is masked.name, a blogging/publishing site hosted by a prominent Tor Citizen, or Torizen.

Tor, like any system, should not be trusted 100%. However, it can be safely used for any variety of things that would be impossible otherwise. It cannot be stressed enough that a Tor user is only as good as their configuration: Tor can be broken via client-side holes in a variety of ways. But with a safe configuration and a cautious end-user, Tor can not only be safe, but possibly the safest means of anonymous TCP traffic.

i2p

Overview

In a lot of ways, i2p is the opposite of Tor. i2p is written in Java, Tor is written in C. Tor uses TCP for its transport (and can only transmit TCP streams), i2p uses UDP for transport (and can transport UDP and TCP streams), i2p was designed originally for two-way anonymity (in the style of hidden services), Tor was designed as an outproxy system. The differences between the two networks offer an intriguing opportunity to compare implementation of the same general goal.

New Terminology

Eepsite: The analogue of a Hidden Service in the i2p world, an Eepsite is a website that is only accessible on i2p.

Eeproxy: An Eeproxy is part of the middleman system that allows i2p to communicate with TCP-using protocols. An Eeproxy specifically deals with HTTP, and is used by a web browser to access .i2p sites.

Garlic Routing: Similar to Onion Routing, the major difference of Garlic Routing is the inclusion of other data between layers of encryption. This partially defends i2p against timing attacks, as data within the encrypted payload is not necessarily just the data received.

Tunnel: In i2p, every node has set of inbound and outbound tunnels. These tunnels are the tendrils through which i2p is able to communicate with the outside world anonymously. The design of i2p, with each node having inbound and outbound tunnels, also means that every i2p node is anonymous.

Strengths

i2p's design is more of a replacement for the IP layer or an IP

overlay than a TCP overlay such as Tor. In this way, i2p is more diverse and possibly more resilient than Tor, as UDP applications are able to utilize it nativity, and TCP applications can be coerced through a TCP stream layer. Another focus of i2p is decentralization. There are few central points of weakness in the i2p system: unlike Tor, which bootstraps nodes from a central directory, i2p has a distributed database which it uses for lookups, bootstraps itself off of a distributed system, and even holds the source code in a distributed framework.

Another design strength of i2p is the fact that all participants are fully anonymous. i2p lacks mass outproxy support, and in effect the network functions as a fully anonymous internet, running on an anonymous IP implementation. This, combined with i2p's variable-length chains, allows for a large amount of diversity in usage. Modified clients or other projects exist to provide distributed forums, jabber servers, IRC servers, email, and even high-bandwidth p2p such as BitTorrent.

Weaknesses

i2p is vulnerable to the some of the same attacks as Tor, with the exception of timing attacks, due to garlic routing. i2p is powerless against an observer who can watch every node in the network, and i2p is also weak against brute-force denial of service attacks more so than Tor, due to it's Java implementation. i2p has not received the peer review or attention as Tor, so developer error could be a possible factor.

Closing Thought

i2p is a great anonymity system for those that are willing to make certain sacrifices, mostly in speed. Java is not a fast platform, and i2p knows this. However, i2p has many benefits that make it possibly more resilient to attack than Tor. Anyone who needs anonymity should not play systems bigot, but instead familiarize themselves with everything that might help them, and i2p certainly will.

i2p's use as an IP overlay is especially important. Currently to the author's knowledge, i2p is the only system that will enable anonymizing UDP applications.

Freenet

Overview

One similarity in Tor and i2p is that both are low-latency forwarding

systems. Tor is a TCP overlay, and i2p is an IP overlay. Freenet is completely different.

Freenet could be appropriately called a publishing system. It is possibly the most resilient publishing system ever created. Freenet is designed to stave off censorship, by providing distributed storage and anonymity. Once a file is uploaded to Freenet, it is nearly impossible to remove. Freenet is, of course, anonymous, and is capable of operating in an even-more-anonymous "Darknet" mode. While Tor or i2p are good for IM, IRC or Email, Freenet is hands down the best for communiques, information on opponents, or anything that must not go down.

New Terminology

Opennet: This is the mode that Freenet operated under in version 0.5 and is an optional mode of operation in freenet 0.7. This is a method for a Freenet node to discover other freenet nodes, and does so openly - hence the name. In Opennet mode, a node will connect first to "seed" nodes, which then offer the node connections to other nodes, and so on. This is vulnerable to attack more so than

Darknet: This is the opposite of Opennet. Instead of connecting to any possible peer, a node is configured to only connect to trusted peers. Don't be too stingy or liberal with your definition of trust - peers you connect to still don't know if connections originate from or are forwarded by you, and if you have enough people as peers that also have enough people as peers, "small-world routing" can be used to create a highly efficient network.

Strengths

Freenet is possibly the most reliable way to publish data. Once put on the network, data cannot be manually deleted by any single party, and will only be removed after very long periods of disuse and want for space on the part of other, more highly used files. This means that unless your content isn't downloaded in many years, it won't disappear. No other system can claim this to the extent which Freenet can.

In addition to its reliability, with enough people in Darknet mode, a small-world network will be formed allowing for easy routability. Small world networking refers to the principle that there is a small number of hops between any two given acquaintances: a cyberpunk version of "six degrees of Kevin Bacon". While seeming fantastic, this works quite well in practice. The main barrier to its implementation is that people who use freenet quite often don't know many others who also use it.

Freenet is one of the longer-running anonymity systems, and has seen a lot of development and use over the years. As a result, many possible holes have been covered, and countermeasures have been devised to a number of attacks.

Freenet is highly fault-tolerant. If a hostile user tried to DoS a node by requesting lots of data, the data would eventually be cached by the node immediately next to them in the chain. Freenet is highly distributed, and anyone using freenet also operates a freenet node, forwarding traffic and storing data for the rest of the network. It is impossible to determine what content is hosted on your node. The Freenet cloud is able to move data around to where it's needed most, so if Bob and Alice both lived in the same area, and downloaded a file multiple times, eventually a node in that area would cache the content, allowing for lower latency and further decentralization.

Weaknesses

A major weakness in Freenet is the discrepancy in security between the Darknet and Opennet modes. While Darknet is far more secure, it is harder to implement in practice. Opennet provides a far easier solution, but allows the network to be attacked by hostile ISPs or governments. There are a large amount of possible attacks on Opennet, almost as much as there are on the rest of the Freenet structure.

Freenet, like i2p, is also implemented in Java. Up until recently, there was no free software version of the Java Virtual Machine, so java code could be potentially untrusted. Currently, Freenet is not compatible with the OpenJDK, so this problem remains.

Final Thought

Freenet, while often overlooked due to high latencies and demands on the end-user, is a thoroughly reliant system of disseminating information, especially information that is disliked by powerful entities. Freenet's network structure and design are resilient, and the developers have experience dealing with anonymity attacks. Freenet was so good that the Chinese government blocked version 0.5. That should say more than I can about its potency.

GNUnet

Overview

GNUnet, the Gnu Project's filesharing protocol, is a relative newcomer on the anonymity scene. Unlike Tor or i2p or Freenet, GNUnet's goal is to be a peer-to-peer protocol for sharing information freely. It draws on the Gnu Project's standards of modularity and portability to

produce a powerful application, but it is still in its infancy, both in terms of code maturity and network growth.

New Terminology

Transport: A transport is a means by which GNUnet uses its network. Currently, GNUnet has four transports: TCP, UDP, HTTP, and SMTP. Transports are fully modular, and have various strengths and weaknesses. For instance, the SMTP transport, while high latency and abuse-vulnerable, is able to get around just about any firewall or NAT (since everywhere allows email).

Strengths

GNUnet's biggest strength is its modularity. The GNUnet application operates as a client and server, and a GNUnet server can serve to multiple clients. Currently, clients exist for basic command-lines, the GTK toolkit, and the QT toolkit. GNUnet is also modular in it's transport layer, allowing users behind restrictive firewalls to still have access to the network.

Weaknesses

As great as the GNUnet codebase is, it's a project that just needs more love. GNUnet is relatively immature, and has a large number of bugs. Any anonymous network needs to grow a bit before becoming fully usable, and GNUnet isn't quite there yet. Hopefully, it will be soon.

Final Thought

GNUnet is a great application with a moderately powerful network that isn't used or developed anywhere near as much as it should be. That's just about all that needs saying.