A social embedding of network security
Trust, constraint, power and control

Version 3.0, not for wide dissemination

David Clark, ddc@csail.mit.edu

1 Introduction

Over a period of less than 30 years, the Internet has moved from research speculation to academic curiosity to an integral part of our daily lives. For many of us, email, the Web, Instant Messages and the like have become something that we count on.

There are many benefits to the Internet, and many concerns about it. One of the most persistent concerns is that the Internet suffers from a bewildering range of security problems, from nuisances such as spam to much more serious problems such as identity theft and network-based attacks. The Internet designers have known for at least 20 years that the Internet has security problems, and there have been numerous studies that document the poor state of security (both in the Internet narrowly and in the computing world more broadly). Indeed, over the last decade there have been many improvements in the technology and practice of security. Many threats that concerned us 10 years ago have been contained. But as the level of attack rises, there is a sense that overall, our level of protection has not improved, and indeed may have gotten worse, even as we more and more depend on Internet and the computers that it connects.

Given our long history of attempts to improve the situation, and the persistent sense that the situation is not getting better, it seems useful to go back to fundamentals: do we agree on what it would mean to have a secure Internet, and can we identify basic principles of computer and communications security to derive a set of building blocks for good security on the Internet.

Traditional texts on security often parse the goal of security into several sub-goals, of which the most important are usually disclosure control, preservation of integrity, and availability. Secondary goals might include non-repudiation. The important point about these sub-goals is that they are functional. They tend to suggest that these are sub-problems that can be separately solved, and that the objective of “disclosure control” for example, is to do the best job possible. This framing equates “good security” with a good solution to these functional objectives.

The mapping of “good security” to “best disclosure control” made lots of sense in the era when the problem was protection of classified information. Today, too much “disclosure control” (in the form of strong encryption) has been the bane of the national security community. The definition of “good security” as “supporting the goals of national security" might suggest that in fact, too much disclosure control is too much of a good thing.[1]

So a basic issue, which we often skate right over, is that we don't have an agreed definition of what "good security" means. If it is not best practice in functional goals such as disclosure control and integrity control, what is it? I will argue that it is a good balance in a multi-dimensional space of sometimes contradictory goals. This point is perhaps obvious, but is often not clearly stated as a starting point for discussion. This way of defining security certainly maps well to the real world, where we know that there is no such thing as perfect security, and what we seek is a reasonable balance of safety and freedom of action, and as well an ability to assess that balance so we don't make mistakes.

Instead of looking at security via its functional subgoals, I want to propose that we analyze security alone an alternate set of axes that more directly reflect today's reality: *trust* and *control*. I will argue that these dimensions give us a means to understand the issues of securing the open Internet.

1.1 Trust:

Communication across the Internet is a shared experience, defined by the goals and wishes of the various participants. The experience can be mapped along a spectrum of trust. At one end we have a set of parties that fully trust each other. They benefit from (and would seek) a network that is transparent, so that these folks can communicate at will, and they would use traditional tools such as encryption to stabilize their communication. This is the world where users would prefer a communication path that is "open" or "transparent", and classic mechanisms for disclosure control make sense. At the other end of the spectrum, there are folks who want nothing to do with each other, and want barriers to keep out the bad guys. This is where "closed" is desirable. But most activity is in the middle, where we accept that we are going to communicate with some other party that we do not fully trust. At this point on the spectrum, we replace trust with constraint. We want checking, validation, witnesses (trusted third parties), and so on. We want viruses removed from incoming mail, spam stripped out, and so on. This situation maps well to the real world, where we daily deal with people we have no reason to trust fully. But if this context is familiar, it is less well studied and framed, in terms of the security problems it raises and how they are dealt with in cyberspace. Disclosure control has little to do with this. Nor does preservation of integrity, if it is defined in a simple form where the receiver gets exactly what the sender sent. Finally, along this spectrum (or perhaps off to the side), is the point where a set of parties fully trust each other, but we deem them "bad guys", and we want to observe or intervene on what they do. This, of course, is what policing is all about, but it is totally contrary to the goal of facilitating the goals of trusting parties.

[1] In the section below on stakeholder analysis, we will observe that disclosure control is not just a simple tool to protect data from the "bad guys", but today a sophisticated tool to shift the balance of power among stakeholders.

1.2 Control

The concept of control relates to the concept of power, which is a topic of study in several discipline. One manifestation of power is to look at the user (the user as "individual"), and ask about the condition of that actor. The values of the Internet are historically centered on the idea of end-user empowerment. This is a valuable technical goal, as captured in the original end-to-end argument . It is also a social goal, if sometimes only implicit. In the early days, user empowerment was expressed as the ability to write and run any code that you wanted to. But users today do not have the skills to do that. They are dependent on others, and in that context we need to re-examine the mechanisms and dimensions of power. One obvious metric is *choice,* and the control of choice.

User empowerment today may be best represented as the ability of the user to choose among providers, services, servers, software and hardware. When we see insufficient competition in the ISP market, which may be the case in today's consumer-oriented broadband, we fear the rise of *market power*, and contemplate regulatory intervention. We see actors (for example, consumer ISPs) seeming to exercise their power to restrict user choice in other areas, for example what applications they can run, or what servers they can use for their services.

What does this have to do with security? If a user has no choice, he may be coerced into using services that he does not trust, or which do not act in his best interests. For example, if an ISP insists that a user only use the mail server of that ISP, that ISP has gained the power to look at all of the user's mail. If a user is forced to use a particular Instant Message service, that service will be able to see all of the users messages. And so on.

Power and control are a way to re-interpret the goal of "disclosure control". Instead of viewing disclosure control as a simple functional objective, we can consider it from the perspective of control, asking questions such as: do users have the ability to encrypt a communication if they want, and who has the power to force content to be revealed? From the point of view of services that store content, who can compel stored content to be remembered, or forgotten, or revealed?

1.3 A rough and ready definition of security

The discussion above suggests that while we may not have a precise definition of security, we in fact have a rough concept that can perhaps get us a ways into this discussion. Security is not the optimization of any single functional dimension, such as disclosure control, but a balance of concerns in a multi-dimensional space, and a balance among the stakeholders. A system such as the Internet is "secure" when the balance is such that the users feel safe enough with the behavior of the system that they are prepared to use it for the desired purposes, and where there is enough assurance and predictability that the user has confidence that the expected balance will actually hold.

This is in fact how security in the real world is designed. We know that there is no such thing as perfect security. To live life, we take risks. But we make judgments as to whether some context or another is secure enough—safe enough and predictable enough—that we are prepared to enter the context and interact there. The park should be "safe enough" that we are willing to go, but not so secure—so constrained or so monitored or so limited—that the experience is of no pleasure or benefit. It is that balance that defines acceptable security, and it is that sort of balance for which the Internet must strive.

An additional point, which we will explore later, is that the balance will need to be different at different times. Some users with a very high need for security and assurance may be willing to accept a very constrained context with limited options for action. In other circumstances, the users may be less risk-averse and more interested in lack of constraints. So there is not going to be any single answer to what balance constitutes "good security". The Internet must be able to provide a range of answers, and part of the mechanism must be to allow the users to control that balance, and to confirm that it is what they want at the moment.

1.4 What follows

In the next two sections, I explore in more depth the dimensions of trust, and of control and power. The discussion of control and power leads to a discussion of stakeholders, and stakeholder analysis. I then use this model to consider the problem of the insecure end-node, which is a fundamental problem in achieving better security in the Internet. Finally, I attempt to derive some specific design principles and approaches for the design of a set of security mechanisms that derive from this basic framework.

2 Trust

A network is a shared medium, and part of communication involves the balancing and reconciling the needs and objectives of the communicating parties. We have proposed that one fundamental dimension of security is the degree of trust among the participants in a communication.

2.1 Defining the spectrum of trust

Above I proposed a spectrum of trust, where at one end of the spectrum are communicating parties that fully trust each other. They have one set of security problems, classically equated to privacy and assurance that they can communicate reliably at will. At the other end of the spectrum are parties that want nothing to do with each other, but are connected to the same network. This situation raises a different set of security problems, related to isolation and protection. Both of these ends are well understood. But in the middle is perhaps the most common situation: parties that do not fully trust each other but who want none-the-less to communicate. Finally, at a point on the spectrum or perhaps off to the side, is the situation that is the inverse of the first, where there are parties that fully trust each other, but they are deemed "bad guys", and the good guys want to intercede. This section further explores the relation of trust and security.

2.2 The social nature of trust and constraint

Trust, as defined by non-technical thinkers, is a relationship between trustor and trustee in which the trustor is willing to assume that the trustee will act in the best interest of the trustor. This does not mean that the trustor can predict exactly what the behavior of the trustee will be, but that the trustee will use judgment and intelligence to restrict the range of actions undertaken.

One who is not trustworthy may be malicious, or simply inattentive, incompetent, or in an unsuited role. Trust is usually accepted with respect to a particular role. We, as humans, use a mix of means to assess how trustworthy a party is: past experience, explicit information, the nature of the relationship (blood is thicker than water, etc.), the role in which the party is to be trusted, and so on. Among humans, trust is a matter of judgment and emotional reaction for all the parties.

When we are in a situation where we lack trust, we use the tools of society to impose constraints on the interactions. We don't meet strangers in a dark alley, we use third parties to verify and assure major financial transactions, we go with a friend to a club, and so on. But constraint on behavior is not a basis for trust. Constraint is in some sense the opposite of trust. When one person trusts another, the trustee is expected to "do the right thing", even though not externally constrained to. A police state may greatly constrain what the citizens do, and this may provide certain sorts of predictability and assurance of behavior, but it does not induce trust. For real trust to develop, there must be freedom for the trust to be tested; there must be the potential for that trust to be violated. It is this risk, and the freedom that accompanies it, that is the essence of human trust. Society gives us the tools to constrain interaction, and also the means to bypass them. The nature of the Internet, which has been called "open" or "transparent", is the constraint-free context where trusting parties interact. And it is among trusting parties, where the overhead of interaction is lowest, but we most easily find innovation, novelty and originality. I believe that the fundamental challenge as we "secure" the Internet is to preserve the freedom that we associate with the open Internet, while giving assurance to parties that don't fully trust each other that there are enough constrains available so that an interaction can be made safe enough to be undertaken.

2.3 Studying the spectrum of trust

If we accept this framework of trust as a place to start, then we can pose a "trust" question about each region of the spectrum:

1) How do we develop trust, and what tools are needed to support the shared basis of trust and the resulting modes of communication?

2) In what circumstances are we willing to communicate with others whom we do not fully trust, and what tools are needed to make this objective as safe as possible?
3) How can we prevent people from carrying out misdeeds and attacks directed at each other, and to hold others accountable for misbehavior, and what tools are needed for this task?
4) To what extent should the system allow for the observation of and intervention in the ongoing communication among parties, without their agreement, in order to carry out tasks of policing and enforcement?

I suggest that if we could answer these questions, we would have a basis to discuss the security and safety of the shared experience in the Internet. And I suggest that in fact, we know a lot about these questions and how we solve them. Finally, I note that this framework does not render irrelevant the traditional security objectives of disclosure control, preservation of integrity, and availability; rather, these objectives will take on different forms along this spectrum.

2.3.1 Question 1: developing trust as a basis for action.

One building block of trust is identity. If I cannot know for sure with whom I am communicating, then it is very hard to make any trust assumptions. The identity need not be that of a specific person—it can be a known role, or an institution as a whole. I may be talking to a policeman, or to my bank. But if I cannot be sure that the policeman is really a policeman, or whether my bank is actually my bank, there is no basis for any sort of confidence in anything, and no basis for any sort of trust. So one reason why the issues of identity and role come up is that they are a building block of trust.

How do we come to develop a sense of trust? This is not a technical problem, but a social one. Part of it is a process of "getting to know one another". The process of "getting to know one another" depends on an important aspect of identity, continuity. The necessary building block of this process is not a certification of attributes or absolute identity, but the assurance that the person today is the same as the person yesterday. The richness of the characterization can be built over time, but only on the basis of continuity. Out of continuity can be built reputation, and reputation is central to the creation of trust.

Another source of trust is the embedding of an experience in a larger context. I trust a policeman if I know the reputation of the police force of which he is a part. I trust a bank because I have a sense of the trustworthy nature of banks in my culture. There might be parts of the world where I would not choose to trust a policeman or a bank, and people moving from one culture to another—from one context to another, may make the wrong starting assumptions about trust. There are cities where one can walk alone at midnight with no fear of being mugged, and cities where this action would be total folly. So we as individuals depend on our social context to help us make efficient, sufficient decisions about the degree of trust to assign to specific situations. We mutually depend on others to build up a shared framework of trust assumptions.

In some cases, this requires the ability to share cues for trust, such as identity, with others. It is much harder to develop robust assessment of trust in isolation. And one of the problems of the Internet is that there are few tools to create that ability to share. Reputation management systems and collaborative filtering systems are examples of explicit efforts to build communities for the purpose of assessing (and enhancing) trust in useful ways. But taking a cue from society, we should expect to move away from a model where every user acts in isolation and toward a model where decisions about identity and trust are made in a more collective way.

A specific and formalized aspect of collective trust is the use of credentials and other sorts of third-party players to vouch for an actor. The use of a "government ID" when you fly, a passport when you enter a country, and so on, are examples of formalized credentials, and we have on-line mirrors of these sort of elements, in the form of so-called "merchant certificates", which are used as part of secure Web communication. In fact, most users never see these (although it is possible to do so); the Web browser performs the task of checking them. It is interesting that most users do not even know what is happening here, and how much decision-making and trust management they have delegated to this software running on their behalf. To a very considerable extent, the ultimate control over trust in the Internet today is in the hands of the creators of the browsers: Microsoft, Apple, and Mozilla (the creators of the Firefox browser.)

One specific question about identity is to what extent, and along what lines, users should be able to take on different identities. In the Internet, tools for managing identity have mostly been created within specific applications. Email names, EBay identities, Instant Message identities and so on have nothing to do with each other. One could argue that this is a good thing—if every action I take on the Internet is associated with the same signal of identity, I may reveal a lot about me in total to someone who can observe and aggregate my actions. On the other hand, there is no reason to believe that the application (email, IM or the like) is the natural partitioning if I have multiple persona. It might equally well be the case that I would like to use different persona in different contexts, but in any one context to use multiple applications. I might want to have one persona I use at work, one I use among my friends and one for interaction with strangers, but in each case I might want to use both email and IM. So this suggests some questions:

- Should the Internet include a system for creation and use of identities (is *persona* a useful word here?) that function cross-application?

- How can we move to a more collective basis for assessment of trustworthyness?

- What are the right tools to help a user manage multiple identities, and use them in the intended contexts? How can this situation be made tractable?

2.3.2 Question 2: how can communication be made safe in the absence of trust?

When two parties want to interact but do not totally trust each other, they often turn to outside agents to help them: trusted third parties and intermediates, brokers, registries, and so on. These can be private sector (credit card companies) and public (registries of deeds), and are often quite nuanced in terms of the service they provide and the protection they offer. The simple, two party model of Internet communication does not directly capture this richness, but in fact it can be built in, and many applications have this sort of structure. The most obvious example is e-commerce on the Internet, which works because of credit card companies that track the identity of buyer and seller (and take on the role of insurance company as they shoulder the risk of fraud). Ebay (and its escrow service) plays a similar role. (Ebay also importantly provides a reputation management service to allow a communal development of trust among the players.)

These sorts of schemes also depend on identity. A buyer and a seller on the Internet may not know much about each other, but will transact because they can trust that the credit card company knows a lot about each of them.

One important form of constraint is the inspection of network traffic to detect undesirable or unwelcome communication. For example, much email today is inspected in transit to remove spam and viruses. This can be seen as just another role for a trusted third party, but it is important to note that if this inspection is truly "in the middle" of the communication path, then this third party has to be able to see, and in some cases to modify, the content being sent. This objective is directly the opposite of the traditional goals of disclosure control and preservation of integrity, if integrity is defined simply as making sure that what the receiver gets is exactly what the sender sent.

Today, most tools to constrain communication are at the application level, since the Internet itself, as originally designed, was completely open. The only network-level mechanism we see today is the firewall, which is a rather crude tool in this space. It blocks certain ports (but for all senders), but normally it has no way to distinguish different senders and adapt its behavior. In general, we see constraints today based only crudely on identity (e.g. the firewall separates the world into "inside" and "outside", which cannot deal with the insider attack nor the trusted person at a distance). The more common form of constraint today is based on what you are doing, not who you are. That sort of constraint is hard to craft so that it cleanly divides good from bad behavior, which is why we are arguing that it needs to be modulated by knowledge of who you are.

Another mechanism to deal with low assurance of trust is to pick randomly. One advice given to a child is that if you get lost, find someone in uniform and ask that person for help. In fact, it is good odds that if a child asks *anyone* at random that the person will be caring and helpful, and very low that they are a sexual predator. The odds change totally if they initiate the contact. It is often safe to depend on the kindness of strangers, if you pick the stranger.

We see examples of this behavior in the Internet in mechanisms such as onion routing , in which a message is sent sequentially through a random sequence of anonymizers, any one of which may not be trustworthy, with the expectation that the sequence of actions will in total be sufficient.

In fact, choice, which is often seen primarily as an economic tool to impose the discipline of competition, is also a part of trust, because it prevents someone in power from forcing a user to make use of a component they do not wish to trust.

A final means to deal with lack of trust is to require that two or more unrelated actors must agree in order for some important action to occur. In business, where there is concern about dishonest employees, there is a well-understood principle called *separation of duties*, in which two separate people must concur for a check to be cut, for example. Outside of business support systems, we have not incorporated this technique into many technical systems, and it has an interesting consideration, which is that it must not be possible for one person to play both roles by taking on two identities, so it imposed an interesting constraint on the design of the identity scheme that supports it.

Some design questions in this space:

- What aspects of constraint and trust-modulated transparency can be implemented in the Internet itself, as opposed to the applications?
- Can we find a variant of the firewall with more discrimination in what it allows under different circumstances?
- What sort of identity information will have to be provided to the "next generation firewall" to allow it to make reasonable decisions?

2.3.3 Question 3: Prevention and accountability

The situations in the first two questions involve a known set of actors—the end points or (in the case of question 2) the end points and the secondary outside agents. Depending on the situation, the community of actors can be small or quite large, but it can be seen to some extent as closed. The situation of prevention and accountability is quite different. I can be attacked by anyone who chooses to do so. Prevention in this case must take the form of blocking by default—anyone unknown must be totally blocked, or else subjected to enough constraint and inspection that their actions cannot cause any harm. Unconditional rejection of the outsider is relatively easy to implement (as we do today with virtual private networks and corporate intranets), but creates gated communities. Inspection of actions to prove them harmless is quite hard in general (think of airport screening), and this point on the spectrum is perhaps the cause of the most obvious security problems in the Internet.

The patterns of behavior that are most difficult to secure are those that are designed to be "open"—to permit any parties to communicate by default. The richness of the space can be seen by looking at patterns of behavior that are embedded in different applications. The two obvious examples of "open" applications on the Internet are email and the Web.

The design of the email system assumes that email addresses are public (or can be guessed) and that anyone can send email to anyone. This sort of open pattern of communication has very strong positive values, but leads to spam, email that contains viruses and so on. Lots of attention has been directed to trying to control these attacks, using various combinations of the two approaches that this discussion suggests. One is to inspect mail and detect all mail that can have bad consequences, which leads to an arms race between the attackers and the spam detection and virus detection tools. The other is to sort incoming mail based on whether the receiver knows the sender, and treat the two differently, which leads to identity theft as a means of attack, and a rather cumbersome process of establishing identity, since strong identity was not built into the original email. The other example is the public Web server, which wants to offer its contents to anyone without demanding a known identity from them. This goal leaves the Web server open to attack from unknown agents, both denial of service (DOS) attacks and direct system penetrations that exploit system vulnerabilities. We will always be living with buggy code, and we can always expect attackers to discover these vulnerabilities, so the uncertainty and insecurity of this situation seems fundamental. Just like all-night convenience stores, web servers that offer to serve any unknown person at any time sometimes get robbed.

In the real world, when we are attacked, we fall back on accountability and deterrence. We call the police, we bring lawsuits and so on. And here the nature of identity changes. Here we need to prove the identity of a party that does not want to be identified. This might be called adversarial identity, in contrast to the willing (if partial) construction of identity that supports questions one and two. Adversarial identity is much harder to arrange, and (given the easy and undistinguished border-crossing communication of the Internet,) accountability and deterrence seems somewhat uncertain, unpredictable, and not very reassuring. It is this area where both users and designers of the Internet struggle the most.

One question is whether there is an online analog of the security camera in the convenience store? What should an observer be able to capture about an exchange of packets that can be used to deter an attack or hold the attacker accountable. One answer is that we might design a system where any packet (or any packet that is the first in an exchange), might be required to carry some indication of identity. But this begs the question about jurisdiction, as well as validity of the identity in a court of law.

At a minimum, an open receiver might well refuse to receive an encrypted message from an unknown sender. Having an encrypted conversation with a stranger is like meeting them in a dark alley—whatever happens there are no witnesses. Witnesses, like security cameras, are useful in deterrence, and only providers of certain particular sorts of services may with to offer the option of encrypted conversation with strangers.

(Note that there is a significant difference between the security camera “in the convenience store” and one “on the street”. The one in the store is installed and used by the owner of the store, and set up knowing the goals of the store owner. The one on the street, perhaps installed by the police, is in a much more public place, and looks for a much more general range of activities. Different agencies may be trusted in different ways.)

When we consider deterrence and accountability, we must consider how this can be implemented. The most basic deterrence is shunning. Shunning does not require the intervention of a police element—it is (to use imagery from the Internet), an end-to-end form of deterrence. The problem with shunning is that it does not work if the offender can just abandon his identity, create a new one, and return in this new guise. Credit bureaus implement the possibility of shunning in the real world, because they maintain not only a record of your credit but a strong idea of who you are, linked to where you live, where you work, and other attributes that are hard to escape. But the Internet today has few such attributes that are “hard to escape”. Perhaps the most persistent form of identity (the one that we will least want to walk away from casually), is the identity we construct in a social network such as Facebook. It is interesting to contemplate whether one could use one’s Facebook identity as a persistent identity for trust and accountability.

When we go beyond shunning, the issue of jurisdiction and boundaries arises. In what jurisdiction can you be held accountable: prosecuted, sued, and do on. Credit card companies, as private sector actors, are trans-national but they can only shun, not arrest. Part of the problem with spam, phishing, and the like is that it seems to originate in foreign countries where, even if we have treaties of various sorts, it will be difficult to trigger the instigation of investigation. So it seems as if the victim has little recourse. There is the additional issue that each such action by an attacker may seem *de minimis,* while the sum of the actions may be quite material. The Internet may allow the possibility of a million dollar fraud, one penny at a time.

If we wanted to add some manifestation of jurisdictional boundaries to the Internet, we can see a range of ways to do it. One would be to add that knowledge “into” the net, so that it is possible to tell, based on some signal such as packet address or route, whether the various parties are in the same jurisdiction. The original designers went to great pains to avoid this capability, arguing that it would bring more harm than good. The alternative would be to add the capability for end-nodes to obtain and exchange robust (hard to forge) certificates of home jurisdiction that they can exchange as demanded by the other parties. These, again, would probably work only if they were tied back to a base identity which is hard or impossible to abandon, but they might preserve anonymity unless a third party (e.g. a court of the jurisdiction) found that there was cause to reveal the identity. This scheme is one where the responsibility for correct operation would be shared among technical and social institutions.

- Could we invent a form of identity that would permit shunning? An “end to end” form of deterrence?

- Can we invent a form of identity that is “hard to escape”, but still allows for efficient operation and some degree of anonymity?

2.3.4 Question 4: Policing the conspiracy

This question of deterrence and accountability presented so far is in terms of the victim and the attacker. The other form of the story is the conspirators and the police. In this version of the story, there are fully willing (and perhaps fully trusting) communicants who want to carry out some action that another party wants to prevent. The other party might be a private sector actor (the RIAA trying to control music sharing), or a public sector actor (the police) trying to prevent distribution of child pornography or detect terrorist plotting. In this case, it can be assume that the conspirators will take every possible step to avoid observation, including encryption of their communication, and the problem of deterrence and accountability becomes most challenging if not impossible.

Law enforcement agents will acknowledge, if only privately, that if two willing parties want to have a private exchange on the Internet, they can probably figure out how to do it, and we should not imagine that we can build in mechanisms that prevent this. The more interesting case is where one of the actors needs to take on a more public role as a part of the activity. Music sharing servers have to advertise themselves, however discretely, to be of use to potential recipients, and this may open them up to detection. This raises images of “gentlemen’s clubs” and other such groups that form by mutual consent, scrutinize their membership, and try to carry out marginal activities within a closed group.

2.4 The landscape of identity

As we have mapped the spectrum of trust, we have in passing started to lay out the range of requirements for identity as well.

Between parties that have somehow come to trust each other, there is a strong requirement to be able to verify who the others are, but this can be done in ways that are private among them. There is no reason that the identities used among trusting parties be revealed to outside parties.

However, users may want to have communication from trusted and from untrusted parties distinguished in the network, so that it can be subjected to different levels of constraint before it reaches the intended recipient. This means that there has to be some indication of identity that can be understood by trusted third parties in the network. Two points are worth making. First, this visible signal of identity need not be the same as the one used by the end-points, although in many cases it may be convenient to share the indication. Second, this externally visible indication of identity does not have to be meaningful “everywhere” in the network. From the perspective of a receiver, it is only useful if the signal is meaningful to agents that the recipient trusts, since there seems little use in invoking an “untrusted third party” to impose constraints. One way to characterize this sort of limited scope of identity is that an end-node can “out-source” into the network (or into a server in the network) some aspects of checking and constraint, and since the end-points can (in general) pick the agents to which the functions are out-sourced, this pattern can still be seen as an “edge-driven” mode of managing identity and trust.

When users want to hold attackers accountable for their actions, then there is a need for some sort of identity that is meaningful to a broader set of players—identity that will hold up in a court of law, identity that cannot be abandoned at will, and so on. This implies the need for some authority that can issue and manage these identities. If all communication had to be associated with this sort of identity, there would be no opportunity for anonymous communication or action where the identity is private to the communicating end-nodes. To allow for private and anonymous communication in cases where it is desired, what is needed is strong enough constraints and other mechanisms so that the risk of attack can be mitigated to the point where accountability after the fact is not a necessity. These mechanisms can include actual inspection of the message contents, and the tools by which trust can be developed and maintained over time so that users are willing to accept the absence of external accountability in their communication.

2.5 Opening a conversation

If we imagine that communication between parties will range from open to constrained, based on the degree of trust among parties, this implies that every communication will begin with a negotiation in which each end determines the degree of trust (and thus openness) he is prepared to accept during the communication. An intrinsic part of interaction is the process of negotiating the concerns and objectives of both parties so that the parties feel sufficiently safe about the interaction, and so that the objectives can be met in a way consistent with the concerns. Computer scientists tend to consider the efficiency of an interaction, which can be measured as number of round-trips in the interaction and size of messages. Protocols designed for efficiency will minimize round-trips and message sizes. But this approach may have risks, in that it requires the sender to put “too much” in the first message, and the receiver to accept “too much” in the first message. The characterization of interaction as a social negotiation implies that there may be real value in an interaction that proceeds in stages, where each stage is perhaps less constrained as confidence grows. Protocols will need to be designed so that each stage of the interaction permits a balance between “degree of risk” and the “degree of trust assessment”.

It may be helpful to outsource the first stage of this negotiation to “blockers”, which can be replicated to a degree sufficient to diffuse an attack. This is another example where collective action is required rather than a mode where each end-node is responsible for its own defense. Diffusion, like shared trust assessment, is of necessity a collaborative effort.

3 Control and power

Power is a concept well studied by political scientists and sociologists. It should not be a surprise that power is an important issue in the design and operation of the Internet, but it has not been well studied and categorized. As a practical matter, issues of power come up every day, as we worry about the market power of broadband providers of consumer Internet access, or we worry about the balance of power in rights of privacy. But there has been no methodical discussion that attempts to catalog the tools of power in the Internet, or catalogs the stakeholders in the power struggles.

3.1 The traditional view—user empowerment

A much-quoted design principle of the Internet is the "end to end argument", which states a preference for placement of function outside the communications substrate, and in the end node. This design approach can be contrasted with that of the telephone system, where the intelligence is in the switches, and the telephone equipment (the "end node") has very little function. This distinction has been summarized as the "smart network" (the phone system) and the "stupid network" (the Internet).

One of the benefits of the Internet's design is that the user can run the code of his choice on his end-node without requiring permission or modification of the network itself (the switches or routers). When a new application is invented, like the Web, instant messaging or a multi-player game, users can simply download new code and start running it. This feature has contributed to the explosion of innovation that has occurred on the Internet, the (perhaps over-)investment and experimentation in new applications, and the rapid creation of new value for the consumer.

It can be argued that all the Internet stakeholders benefit from this innovation and the creation of new value, but the power in this story lies with the end-user and the innovator, not with the Internet service provider or the regulator. The ISP just carries packets, and while he may (or may not) benefit financially as traffic goes up, he does not control which traffic is sent, nor have many opportunities to set prices based on value. Needless to say, the real story is more complex than this.

3.2 The role of topology

There is a myth about the Internet that since there is no such thing as a "call" or "call setup", and the traffic just flows as it is directed dynamically by the underlying routing and forwarding protocols, it is impossible to watch what the sender is sending in any reliable way. In the "center" of the net that is true. But it is not true at all at the edges, where the consumer attaches to the network, or at "constriction points", where traffic is funneled into a restricted path (such as where a corporation connects to the rest of the Internet over a small number of paths). For residential users, who usually connect over a single path (e.g. a DSL connection or a cable connection), all of their traffic flows over a path that is highly predictable and stable. This gives the owner of this path the power to observe all of the user’s traffic, and to exercise whatever sorts of controls or discrimination that can be devised from what can be seen in the traffic.

Corporation often use their points of connection to the public Internet as a point where they can observe and police what their employees do, looking for forbidden activities such as downloading pornography or on-line gambling. These points can also be used to log activity, such as capture and retention of email and instant messaging. While security folks use these points to filter what comes *in* (using devices such as firewalls), it is also common to control what goes *out.*

These constriction points also make a useful target for third parties (such as the government) who want to observe and control. So these points become favored options for wiretap exercises.

End users have a range of means to resist these impositions, as outlined in the next section. Among these are VPNs, and the option of “multi-homing”, in which a user obtains multiple paths into the network. Multi-homing is usually justified as a means to improve reliability in cases when connections fail, but it can also allow a user at least some control over the path his data takes.

3.3 Control of DNS and naming

End-points on the Internet are identified by addresses: 32 bit numbers (often written as four decimal number separated by dots). But users almost never identify the machine they want to contact by giving its number. Instead, they give a *domain name*, which is a sequence of character strings separated by dots. Email addresses contain domain names, as do URLs. A system called the Domain Name System, or the DNS, converts names to addresses, which are then used to send packets to the desired end-point.

The DNS is a globally distributed set of servers, with each server controlled by its owner/operator. While the control is globally distributed, the ISP that runs the DNS server that provides the first point of contact for query has special control. The design of the Internet is such that users contact a local, nearby server with every query, and it may forward the query on, or it may provide an answer based on cached information from some other recent query. The end-node essentially always accepts and acts on the answer it gets from the local DNS server. So if the server gives an incorrect or malicious answer, the originating end-node will start out trying to reach a named destination, and end up quite somewhere else.

The DNS is not a very secure system, and DNS servers can be corrupted—this is the basis of the so-called *pharming* attacks. Attackers have modified local DNS servers to give the wrong answers to queries, and have modified routers so that when a new end-node connects, it is configured to use a malicious DNS server provided by the attacker, instead of a trustworthy one.

But the ISP itself may also choose to modify or override the “correct” name-to-address binding, for its own purposes. So the DNS is a focus of security concerns today both because a third party can attack it, and because it provides a locus of control for any access ISP.

3.4 Control over routing and forwarding

Another point of control arises in the actual forwarding of packets by an ISP. The sender puts the Internet address of the destination in the packet and sends it, but an ISP may choose to deliver it to any destination it selects. Normally, this sort of thing happens because of a transient error in the computation of routes to a destination: the packet goes astray or to a useless destination, and the result is that nothing happens. But if the “false” destination answers, and does an imitation of the legitimate destination, and there is no mutual exchange of validating identity information, the sender may never know that it is talking to the wrong end point.

The combination of control over the DNS and control over routing means that in fact, no sender should ever be sure that it is talking to the intended destination unless there is some end-to-end confirmation. But most applications assume that the DNS and routing work correctly. There are many occasions today where these controls are used to manipulate the destination address, and since they are (usually) benign, users do not challenge them. When a user goes to a hotel or hot-spot where payment is required for service, the instruction is to “go to your browser and connect to any web page”. The ISP then uses exactly these sorts of controls to deflect the connection to the server that accepts payment. Later, if the user sends mail, the ISP may deflect the connection for the mail to a local mail server. This is actually viewed as the preferred practice, since it solves a problem related to spam. But we should be concerned and alerted by the fact that today, we often see intended connections across the Internet end up at a location quite different from the one specified.

3.5 What is concealed, what is revealed?

To the extent that a service provider (or third party) can observe what the user is sending, that observation can be translated into restrictions on action (blocking certain traffic), differential pricing, selective logging, and so on. It can also form the basis of selective service enhancement (such as enhanced quality of service for Internet telephone calls). This reality begs the question of what can be observed, and why.

At one extreme, the ISP can see “everything”—every byte in every packet. At the other extreme, the ISP can see “nothing”—all the traffic is encrypted and the destination address reveals little or nothing about whom the user is actually communicating with? So an important question about power is who can control what is visible and what is hidden.

To some extent, the answer to this question today is a matter of historical accident. In the beginning, there was not much concern about power, and a presumption that the interests of the stakeholders were aligned. Encryption was expensive (and discouraged by governmental policy—a clear exercise of power in a non-technical sphere), and all traffic was sent in the clear, with little concern as to who was watching.

In fact, the design of most applications made it especially easy to track what the user was doing. There is a technical mechanism in the Internet called “ports”, and “port numbers”. A packet contains a destination address that identifies the end-node that is to receive the message. Once the message has reached the end-node, it has to be directed to the proper service on that machine—a machine can host many services. So the packet contains a further piece of information called the port that directs the packet once it reaches the destination machine. By convention, different applications, such as the Web or email, use “well known ports”, and thus an observer can tell what application a user is running just by looking at one field in the packet. There is no reason to examine the packet in detail.

There is no reason why applications need to use well known ports, and some applications today have been designed to select port numbers at random to make it harder to track them. In particular, some music-sharing services, which are being tracked by the rights-holders, have taken this approach to disguise themselves.

Users can employ encryption to hide (to some extent) what they are doing. Encryption is now becoming more widespread across the Internet, with many different patterns of usage. Secure Web sites use a protocol called Secure Sockets Layer, or SSL, to provide protection against observation for ecommerce and other financial transactions, or for transactions where private information is exchanged. Employees on the road often use a mechanism called Virtual Private Networking, or VPNs, to make an encrypted connection from wherever they are (a wireless hotspot or hotel room) back to their employer’s network. If a user has used an encrypted VPN, there is essentially nothing the hot-spot or hotel can see, because all the messages are encrypted, and they all go to the same destination inside the employer’s network. SSL does not normally disguise the port number. So today we see a range of encryption options, which greatly influence how much the ISP (or third party) can observe, and how much power each player has to regulate what is happening.

One way to control this power directly is to restrict the use of encryption. It has always been the policy of the United States that its citizens have the right to use encryption, but other countries have had different policies. (France, for example, outlawed the use of encryption by its citizens until around 1999). The US did use a variety of means to discourage the use of encryption, not to give the ISP more control but to give law enforcement more opportunity to monitor. ISPs, as private actors, can forbid the use of certain forms of encryption if their customers will accept this limitation, or if the market has insufficient competition to give the user any choice. It would not be practical today to block SSL, since so much of Internet activity depends on it, but ISPs occasionally block VPNs, usually in attempt to sell a higher-priced variant of the service that permits it.

3.6 An example--email

An example may help clarify how these factors play out. When a user sends an email, it normally does not go directly from the sender’s computer to the recipient’s computer.

Instead, it is normally sent from the sender's computer to the sender's SMTP server, and from there to the receiver's server, where finally the receiver retrieves it using one of two protocols: POP or IMAP. The original reason for this design was to permit users (and their computers) to exchange email even if their computers were connected to the network only occasionally. This structure is of considerable value today, with the prevalence of lap-tops that are often off the net. This feature also allows a user to read his mail (using IMAP, for example), from one of several end-nodes, and have actions taken at that end-node (e.g. deleting a message) visible to the same user from any other of the end-nodes.

The question about power and choice is very simple: can the user select the server that he uses to send and/or receive mail, or does some other actor constrain that choice and impose the answer on the user. In the original conception of email, this question was not an obvious one to ask, since the designers were not thinking about actors with adverse interests, and the designers assumed that the user would pick a service provider based on convenience and reliability. The design of the email protocols does not constrain the ability of the user to choose, and most email software allows the user to configure it with the servers to use for sending and receiving. So it would seem that the user has the power to choose.

However, some access ISPs, by virtue of their control over topology (discussed above), have imposed traffic blocking and rerouting that attempts to constrain that choice. There are several reasons why an ISP might want to do this. One is to create customer "stickiness". When a user picks an email server to use, he picks his email address. If a user obtains mail service from the "xyz.com" company, then his email address will be something like user@xyz.com. Once the user has given this address to all of his friends, he is not likely to change it casually. To avoid this "capture", some users take advantage of third-party providers that offer services that give users an email address independent of their ISP. Many universities will give their alumni an address (for free, in order to make connections to their alumni community), and many professional societies will provide email addresses to their members. This removes the element of customer stickiness that an ISP would like to create. So one response is to try to block such addresses.

For another class of users, corporate users working from home, the motivation may shift from stickiness to additional revenue. For such users, by blocking their ability to use their corporate email address from home, it might be possible to shift them to a higher-price access service.

There have been many complaints from users about these restrictions. The ISPs have in some cases relented from this blocking, and in other cases justified them (with some if not compelling justification) with arguments about preventing spam. Corporate users respond by creating encrypted VPN connections back to their corporate network, and then sending and receiving mail over that connection, so that the access ISP cannot see or block their activities. Some ISPs have responded by blocking encrypted VPN connections unless a higher-price access service is purchased. Users respond with intense complaint, and some balance is struck. Most ISPs today do not attempt to block VPNS, so for savvy users that can master the mechanics of making encrypted connections, the ISPs have little ability to inspect or block.

4 Stakeholder analysis

In order to study question of power, and the balance and tussle over power, it is important to understand the set of stakeholders and their motivations. The previous section has hinted at some of the important actors, the end-users, their access ISPs, the designers and providers of application level services. In general, a stakeholder analysis will reveal the same general sets of actors as we might find elsewhere, for example governments and other agents of the state, as they try to impose regulation and order on their society, and organizations that represent the interests of the individual (with concerns such as civil liberties or privacy) who advocate to the government for the collective interests of the citizens.

What is most distinctive about the Internet is that since its goal is communication, all the parties to the communication have to been seen as stakeholders with interests that are not necessarily aligned. That is, there are *shared* concerns that a set of users must negotiate or harmonize before they can communicate successfully. In the original conception of the Internet, the necessity of this negotiation was (once again) not very obvious, because the idea that users would have adverse interests and still want to communicate was not obvious. But today, we see many examples of tussle and balance of power in the resolution of shared concerns.

For example, a Web site may offer free content if the user will agree to reveal some demographic information, and the user will have to decide whether to accept or reject this proposition. Protocols (P3P) have been designed to capture this negotiation. One user may wish to have an encrypted (private) communication, and the other party may not choose to use encryption. Any use of encryption reflects a shared agreement—one end cannot do it alone. So, to return to the example of email, some email servers support the use of the encryption mechanism called Secure Socket Layer, some do no. Some web servers will provide SSL to protect transactions such as ecommerce that benefit from protection, other web servers will not offer this option.

Most of the example above do not actually include any sort of "negotiation"; they are more of the "take it or leave it" form. And this reflects a general concern that in the resolution of these *shared* concerns, the power is with the large actors, not the individual.

With this introduction, we can list the set of stakeholders that we have so far identified in this discussion. We have organized these into distinct *contexts*, as a way to organize the brief analysis we provide of each category.

The individual context: The individual user, who had concerns about privacy, safety, freedom from attack, stability of information, and so on. In this context, the "individual" can indeed be a person, but also an organization or larger collective entity. Within the individual context we also find groups that advocate for the rights of the individual, such as civil liberties groups, consumer advocate groups, human rights groups, and so on.

The shared context: The set of users who are communicating at any moment, who must negotiate a shared set of concerns in order for the communication to proceed. As noted above, this context is a distinctive consequence of the fact that the Internet is a communications medium, and all communication involves a set of actors who choose to communicate. Also, as noted above, different actors may have different power in the negotiation of the resolution of shared concerns. Additionally, the shared context can be used to characterize activities beyond actual data transfer, including collaborative filtering and reputation systems, where users make shared decisions modulated by a state of agreed trust.

The communal context: In this context we find the state, as represented by government at all levels, which is responsible for communal concerns, such as policing, prevention of unacceptable behavior, stability of commerce, and so on.

The global context: The Internet is a global system, and crosses many different societies with different communal concerns. All the governments of the world, each representing their communal contexts, define the global context. In this context we also find stakeholders that span different jurisdictions and diverse sets of values and cultures, such as the United Nations, the ITU, and current (as of 2006) activities such as the World Summit on the Information Society.

The provider context: The service providers and system providers, who provide the Internet service, the facilities on which it runs, the end-user hardware and software, and so on. This context includes both Internet Service Providers and providers of higher level services, such as email, web, and so on.

The designer/standardization context: Many of the tussles over control seem to arise at “run-time”, when different stakeholders are involved as the users of the Internet actually interact. But it is important to remember that the playing field for these run-time interactions is defined by the protocols and standards of the Internet, which are created by yet another set of stakeholders, including corporate players, academic researchers, and advocates for various sorts of designs that favor one or another stake-holder at run-time. Designers can have an important influence of the subsequent tussle for control by the definition of system modularity and critical interfaces, including the intentional omission of specific interfaces to make the “transaction cost” of interacting at that point in the design much higher.

The third-party context: Private sector third parties, who wish to involve themselves in the actions of other stakeholders, perhaps to protect their interests (e.g. the music industry protecting copyright), or to gather marketing information. This context is distinguished from the provider context in that these stakeholders have not been explicitly selected as service providers by the individual users. They may seek help with their agenda by appealing to stakeholders in the provider context, or by appeal to governments (e.g. by lobbying for laws that protect rights-holders.

These various stakeholders and contexts interact in complex ways. For example, the Department of Justice (a stakeholder in the communal context of the United States), tried to address its needs for wiretap by approaching the IETF, a stakeholder in the designer/standardization context. They were rebuffed, for a number of reasons including the desire of the IETF to remain credible in the global context and a lack of priority for the concerns of law enforcement. So the Department of Justice instead turned to other actors within their own context (the FCC, in particular), and obtained a legal/regulatory mandate to demand implementation of tools that support wiretapping. In doing so, they have caused the Internet to be modified with a new point of control that may in the future be used by a wide range of stake-holders for a wide range of purposes in different context.

4.1 Stakeholder analysis and security

It is important to loop back, after this extended discussion of stakeholders, and link this discussion to the starting point of security. In fact, we can use our original axes of trust and control for this purpose.

The original analysis of trust was presented in a very simple context, the **shared** context. It was framed as the question of whether a set of communicating users chose to trust each other, or to invoke constraints to bring a level of safety where trust does not hold. But in fact, we should perform this trust analysis for any stakeholder in a position to influence the outcome of our use of the Internet. For any stakeholder, we can ask whether we trust them to act in our best interests, or whether their behavior is sufficiently constrained that they cannot disrupt our interactions in unacceptable ways. If we have not identified a stakeholder, then we will not perform this trust analysis, and our trust assumptions can only be implicit, not articulated. So, for example, most protocols today do not confirm that they are talking to the end point they intended to connect to. Since the ISPs can alter this relationship, we are implicitly trusting the ISPs to map names to addresses and to map addresses to destinations in ways that are consistent with our interests. Perhaps ISPs are trustworthy, perhaps not—we implicitly trust them unless we catalog them as stakeholders and ask the question explicitly.

The second reason to catalog the stakeholders is that if we identify a situation where there is an actor that is not fully trustworthy, and we seek to identify constraints that will make the involvement of that actor more safe or predictable, we should remember to look in all the contexts we have described, and at all the stakeholders we have described, to fine possible constraints. Most computer scientists are most comfortable with technical constraints, which arise in the **design/standardization** context. We know how to reason about them, and they sometimes have a comfortable sense of predictability and certainty. But one may equally well turn to the **communal** context, with its laws and regulations (within a jurisdiction), or to the **provider** context, where it may be possible to use the discipline of competition to constrain a certain actor.

In a complex system such as the Internet, the network of the stakeholders will contain many connections and cross-dependencies that function as constraints on untrustworthy action. A technically focused computer scientist may be uncomfortable with the degree of uncertainty associated with some of these constraints, but these sorts of constraints are the glue of real society.

Next, the design of mechanism can have the effect of adjusting the relative basis for control and power among the stakeholders, and to reason about this carefully, it helps to have a catalog of who those stakeholders are. A very simple example of such a mechanism is end-to-end encryption. The use of encryption can be associated with the goal of disclosure control, but this not the simple goal of keeping the "bad guys" at bay, but the much more sophisticated goal of shifting the balance of control and power away from the providers, the third parties, and the state, and toward the individual and the shared set of willing communicants.

Finally, we noted in the discussion of trust that a set of actors may have a different degree of mutual trust in different circumstances, and we may want to have a set of mechanisms that can adapt to this range of trust. In the case of communicating end-users, we identified the feature we called *openness* or *transparency* as the behavior that we wanted to vary as the degree of trust changed. It is not too hard to see that openness relates to security, in that being open to people you don't trust is an invitation to compromise. But in any interaction among stakeholders, when we do a trust analysis, we should ask what the functional behaviors are that we would like to adjust as we move along that spectrum, in order to improve security. So, for example, regions that trust each other might just exchange routing information, while regions with less trust might do more consistency-checking or validation. DNS servers (or providers) that agree to trust each other may just transfer data in bulk (called a "zone transfer"), while other providers may choose only to accept limited data, or to check it in different ways.

All of these examples map in some way onto the dimension of *openness*; with actors we trust we are prepared to "accept what they say", with actors we don't trust as much, we spend more effort to verify, cross-check and confirm.

4.2 Mechanism evaluation via stakeholder analysis

The discussion above hints at an approach to a comparative evaluation of security schemes. Given a specific security problem, for example spam, it is possible to conceive of solutions that are positioned in the different contexts, and an analysis of these contexts may provide some hints about the relative merits of the different approaches. One could try to solve spam in the **global** context, and impose some common requirement on all email such as the sender must pay a penny. This raises all sorts of issues of getting agreement, enforcement, assurance and fraud control. One could try to solve spam in the **communal** context and pass laws that prohibit certain behavior. This approach raises issues of limited jurisdiction. One could try to solve spam in the **provider** context, and empower the email providers to control spam by licensing bulk mail senders (an interesting solution, in that it requires a careful analysis of trust among the email providers). One could try to solve spam in the **shared** context, and create mail clubs, collaborative spam detection schemes and so on. Finally, one can try to solve spam in the **individual** context, and just install spam filters.

Each of these schemes has advantages and disadvantages. **Global** schemes may require an unachievable degree of agreement and consistent commitment. Disagreement and tussle for control and power may doom this approach to interminable debate. **Communal** schemes limit this debate to one state—one government or one society. The tradeoff for limiting the stakeholders in the debate is exactly the exclusion of those actors from the solution. One may fall back on more general cross-state mechanisms such as extradition to mitigate this limitation. **Provider** schemes may shift more power and control to the service providers than is desirable. **Individual** schemes are the easiest to deploy, since they do not depend on any sort of stakeholder negotiation (and indeed this sort of scheme, together with simple sorts of outsourcing, are the most common for spam control today) but they may suffer from the lack of cooperation in solving the problem. **Shared** schemes may be a very fruitful approach, but they are hard to design, and require careful analysis of trust and stakeholder motivation.