Omnivore:
Risk Management through Bidirectional Transparency

Scott Flinn
National Research Council of Canada
Institute for Information Technology
Fredericton, NB, Canada
Scott.Flinn@nrc.gc.ca

Steve Stoyles
Dalhousie University
Halifax, NS, Canada
stoyles@cs.dal.ca

ABSTRACT

Internet users face a variety of risks as they conduct their business on-line, but they are often ill-equipped to recognize the risks and deal with them effectively. As a result, many users take the approach of limiting their on-line activities so as to reduce their exposure. This paper describes a *risk management* approach to building confidence and trust for Internet users. The underlying philosophy is not to make the Internet inherently safer, but to help users build an awareness of the risks they might encounter and to supply them with timely guidance. We also report on experience with a prototype system built to evaluate some of these ideas.

Keywords

Human factors, privacy, risk, risk management, safe staging, security, transparency, trust, usability, Web, WWW.

1. INTRODUCTION

At the CRA Conference on Grand Challenges in Information Security and Assurance held in November 2003, a risk management approach to security was identified as one of the important challenges facing the field [2]. Specifically, the challenge was stated as follows:

> *Within 10 years, quantitative information-systems risk management is at least as good as quantitative financial risk management.*

As stated, the challenge is concerned primarily with quantification and measurement of risk. It is based on the premise that it is difficult to reason about security scientifically if you have no way of measuring it. Implicit in the challenge is the need to formulate metrics that decision makers can actually understand and use, and to communicate them effectively.

This paper describes one possible approach to risk management in information systems. It outlines an approach that is broadly applicable in many situations, then demonstrates how the approach may be applied to the specific security issues surrounding Web browsing and Internet use.

The paper is based upon the combination of three major ideas, which are introduced here and elaborated on in later sections.

People are an intrinsic part of information and security systems. There is a growing community that in recent years has devoted considerable attention to the human factors of system security. There have been many research results, all of them interesting, many of them alarming. A common theme is that information systems almost always include people, and the security of those systems often depends as much on the behaviour of those people as on any technical consideration about the system. This idea is expanded in Section 2.

Security measures are complemented by a risk management perspective. There is certainly a pressing need for continued research into security mechanisms, measures, protocols and systems. At the same time, a risk management perspective can complement the effort. A useful analogy emerged from the CRA conference cited earlier and relates to the distinction between the *function* and the *purpose* of the brakes on a car. The function of your brakes is to slow or stop your car. Their purpose, in contrast, is to allow you to go fast. By analogy, security technologies perform a variety of functions relating to identity, information flow, etc. Their purpose, however, is to allow us to seek rewards from use of information systems by shielding us from potentially harmful consequences. Very often we can pursue greater rewards if we are willing to assume greater risks. The risk management perspective of security focuses on how to find a sensible balance and is the subject of Section 3.

Omnivore: building trust through bidirectional transparency. Section 4 introduces a system we call *Omnivore* that represents a specific application of the risk management approach in the context of typical Internet users browsing the World Wide Web. It does not seek to better protect users from the risks they may encounter on-line, but rather to help them develop an awareness of the risks, to encourage them to take

NSPW 2004 Nova Scotia Canada
© 2005 ACM 1-59593-076-0/05/05…$5.00
Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full cit ation on the first page. To copy otherwise, to republish, to post on servers, or to redistribute to lists, requires prior specific permission and/or a fee.

actions to mitigate the risks, and to provide timely information to effectively support their risk management decisions. The section also reports on early experience with a prototype system.

2. THE SYSTEM INCLUDES PEOPLE

This paper is primarily about the people in information systems and how they affect system security. The traditional view of information systems is one of machines, networks, storage and software. Human users are external to the system and interact with it through a variety of interfaces ranging from programmatic to interactive. When people use a system, however, they become part of the system in a practical sense. They make decisions affecting the information that flows in and out of the system, and they take actions that alter its state. From a security standpoint, attackers who analyze a system for weakness generally include people in their analysis, and often find them to be the most vulnerable part.

Consequently, people cause harm to the information systems they use. Malicious insiders can damage systems and data directly, illegitimately divulge sensitive information to which they have legitimate access, or deploy trojans to create avenues for illegitimate access for themselves or others. In a world where password authentication is ubiquitous, users engage in a wide range of insecure password practices [1]. Social engineering is a technique that preys upon the natural tendency of people to be helpful and open, freely divulging not only passwords and other credentials, but also the information ostensibly protected by them. It is a difficult phenomenon to control and was the chief weapon of one of the most notorious and successful hackers of our time [21].

A good example is provided by a 1996 study of fraudulent telephone calls in the United Kingdom's National Health Service (NHS). Anderson has argued that the greatest threat to NHS information systems is from private investigators and other outsiders posing as doctors to obtain patient information. Calls are typically of the following form:

> *Hello, this is Dr. Burnett of the cardiology department at the Conquest Hospital in Hastings. Your patient Sam Simmonds has just been admitted here in a coma, and he has a funny-looking ventricular arrythmia. Can you tell me if there's anything relevant in his record?* (Quoted from [5], p.167)

In a pilot study, it was found that one particular health authority received roughly 30 such fraudulent calls per week ([5], p.167-8). Loosely extrapolated to the entire NHS, this suggests that some 200,000 such fraudulent calls are placed every year. It is reasonable to expect that most of them result in inappropriate disclosure of sensitive or confidential patient information. A simple solution is to institute a call-back protocol: if you receive such a call, you inform the caller that you have the phone number for their location and will call them right back with the information.

This example illustrates three important points. (1) The vulnerability is created by people and exploited by people. Technology is involved only indirectly (massive aggregation of data within the NHS makes the attack more effective). (2) There is a simple and effective solution that can be achieved without any technological change. The difference between a damaging vulnerability and an effective protective measure is strictly one of user behaviour. (3) The problem, though epidemic in proportion, is almost invisible. It is likely that these fraudulent calls have continued, but we have no mechanism to link visible consequences back to the cause.

People can also be exploited indirectly to attack other targets. The MyDoom.A worm released in February of 2004 relied on relatively simple social engineering to mount a massive (and highly effective) DDoS attack against a specific target (www.sco.com).

At the same time, people are harmed through their use of information systems. Malicious and damaging viruses are an obvious source of harm. Spyware is also commonplace, and can be equally damaging. For example, a spyware program might record everything you type into your Web browser and send it off to a remote server. In both cases, users are frequently unaware that their machines have been infected. An interesting example is furnished by the KaZaA Media Desktop. A recent study found that KaZaA users were generally unable to recognize that the entire `C:` drive of the host machine was being shared, believing instead that only an isolated sub-folder was accessible. Furthermore, they were generally unable to restrict access to the desired folder once the situation was revealed to them [17]. Once again, it is an example of something going wrong invisibly.

People also fall prey to so-called *phishing* scams and similar social engineering attacks that dupe them into revealing access credentials and other personal information. Techniques such as these can be combined to perpetrate identity theft on a large scale.

On the other hand, people can also contribute strength to a security system. Authentication remains a stubbornly difficult problem, yet authentication of familiar people is something we can all do almost effortlessly and with high assurance. Spam filters and intrusion detection systems alike still struggle to sort the wheat from the chaff, yet most people can detect spam with high precision (because it is people who create the ever-changing definition of what it is), and confirmation of intrusion often requires human input. In both cases, we can adapt to new variants relatively easily while automated systems largely fail.

It is as important to exploit human strengths as it is to address human weaknesses. That is one of the major themes of this paper. Trust can be a complex, messy human affair. Users may be better served if we put as much effort into helping them make informed trust decisions for themselves as we do into teaching our machines about what trust means to people.

3. RISK MANAGEMENT

The conventional reaction to unsafe user behaviour is to suggest that users be educated to behave differently. The best advice currently available for end-users is to install virus filters, spyware detectors and personal firewalls, to browse anonymously, refuse cookies, lie when filling out forms, delete e-mail attachments without opening them, to view every e-mail communication with suspicion, and to avoid installing non-essential software. The philosophy is to minimize exposure by shutting down the flow of accurate information. There are some who appear to believe that our problems will end once this message has been widely communicated.

Sometimes, however, there are rewards to be gained by taking risks. You may get better service at a Web site if you accept their cookies. You may find on-line banking so convenient that you are willing to assume the associated risks. In general you may find greater value in on-line services if you are comfortable with the exposure and risk they present.

The problem is that we lack metrics for risk in information systems. Without the ability to measure risk, it is difficult to evaluate the trade-offs and to manage it rationally. This is the central idea of the grand challenge summarized in the introduction.

The challenge is to supply users with information they can use to minimize the impact of the risks they face and the risks they pose. To be effective it must be information they are capable of understanding and acting upon, and therein lies a substantial part of the challenge: the most appropriate information and the most effective way of communicating it will depend strongly on the target user population. Metrics suitable for system administrators may not be useful for more typical Internet users.

We have found that many of the issues surrounding this kind of approach can be divided rather neatly into five categories, each one typified by a question a user might ask. Further, we have found it to be a useful framework for analyzing a wide variety of situations with respect to end-user risk management. The questions are first listed below and then described in more detail with reference to a number of common examples. The remainder of the paper focuses on a specific user population in a specific context: typical Internet users interacting with the Web through a contemporary Web browser. The examples are therefore drawn from that context, but the questions and principles are more broadly applicable.

Risk management questions for end-users:

1. What could go wrong?
2. How likely is it, and what damage would it cause to me or to others if it did?
3. How would I know if something went wrong?
4. What reason do I have to believe that it won't?
5. Who is responsible to ensure that it doesn't, and what recourse do I have if it does?

What could go wrong? This is nothing new to a security specialist. You always begin by enumerating the significant threats, then consider how you might counter them. For many users of information systems, however, the question is not routine, and they are less practiced at answering it well. The challenge is to have a *comprehensive* awareness of potential harms. Incomplete lists can lead to both weakness and complacency. If you are implementing authorization controls, for example, have you considered the impact of social engineering attacks (recall the NHS example)? When Internet users consider paying for something on-line with a credit card, they may be aware of the importance of a having a secure connection (though they may not recognize it [9]), but are they also aware of the risks posed by the vendor's subsequent storage and handling of their number?

How likely is it, and what damage would it cause? The essence of a risk management decision is to find a suitable balance between likely cost and likely reward. The cost depends in part on the likelihood of harm and in part on its magnitude. Both are difficult to assess.

If you follow security related news and disclosure lists such as BugTraq, you will be aware that hundreds of new vulnerabilities are discovered every year in widely deployed systems. On the other hand, we know of comparatively few that are actually exploited. Why is that? Is it because there are too few people with the right mix of skill, interest and malicious intent to exploit them? Or is it because we aren't aware when exploits occur (see question 3)? What really is the likelihood of harm from a given threat? Without effective metrics, it is difficult to assess the potential harms, which in turn makes it difficult to determine how to allocate resources to guard against them. For example, the odds of someone cracking the encryption in a secure HTTPS connection are insignificant compared with the likelihood of the vendor storing sensitive information insecurely. This knowledge can significantly affect how you might evaluate the risks involved in making an on-line purchase and what actions you might take to protect yourself from them.

It may be even more difficult to accurately assess the potential damage if something goes wrong. The damage may be to yourself, to people you know, to people you don't know, to your company or organization, and so on. It may be to your finances (your on-line bank account is compromised), your reputation (someone unfairly gives you a negative rating following a transaction on eBay), or both (your identity is stolen and used for illicit activity). And the damage may be indirect or difficult to detect (again, see question 3).

Web browser cookie management provides a good example. Cookies can be used by providers of Internet services for a variety of behaviour tracking purposes and are therefore viewed by many as a privacy threat. In particular, they can contribute to aggregation processes through which scattered bits of information about an individual can be relinked to build a detailed profile. If this is the threat, then how significant is the potential damage?

When cookies first appeared, they were completely invisible to most users. Web browsers have since added a number of features for the purpose of helping users understand what is happening and providing them with an opportunity to give informed consent [20]. For instance, depending on your browser, you might now have the opportunity to review (and delete) the cookies your browser has stored, to selectively block cookies from designated sites, and to inspect cookies as Web sites makes requests to store them, choosing in each case whether to accept or reject them. But how much practical protection does this really afford? When asked whether you would like to accept a cookie from a particular site, on what information can you base your response?

A common strategy is to accept cookies from sites associated with a trusted brand, and reject them from all others. How often does this strategy result in unanticipated harms from accepted cookies? How often does it result in diminished service or value because of reluctance to accept a cookie that is “unbranded” but nevertheless trustworthy? Why do we believe that it leads to better outcomes than simply accepting all cookies? Ideally, decisions would be based on factual statements of the following form. “If I accept this cookie, my e-mail address will become available to a marketing firm which may then sell it to clients.” Or, “If I accept cookies from this site, certain on-line vendors may learn that I have visited it.”

How would I know? Information systems have been carefully designed to hide their inner workings. Security measures in particular are often intentionally hidden in an effort to minimize their negative impact on usability, or to eliminate the temptation for users to circumvent them. The nature of digital information also tends to mask malicious activity. When you steal digital information you take only a copy, leaving the original undisturbed. Evidence that a copy has been made, if it exists, will typically be buried in voluminous logs.

Consequently, when security measures fail they often do so invisibly. Fraudulent phone calls to the British NHS apparently occur at an alarming rate, yet there is almost no direct and visible evidence of it. In a widely reported incident, an intruder had access to Microsoft’s internal network for months before being detected.

At home, KaZaA users unwittingly share sensitive personal information because they have incorrectly configured their software. Spyware programs (some of which are installed along with the KaZaA Media Desktop) monitor the flow of information within their computers and report to external sites. Viruses deliver trojans and other payloads that are used for DDoS attacks and open spam relays. The only evidence many users are aware of is sluggish performance at times.

What reason do I have to believe that things won’t go wrong? One answer is to trust your own experience. Suppose you had good reason to be confident that you would in fact be aware if something was going wrong, and over time you observe that things have not gone wrong in certain situations. This may be sufficient reason to engage in the same activities with continued confidence. In other words, you can develop your own body of reputation information.

For many people, however, continuous monitoring and reliance on one’s own vigilance may not be an acceptable solution. The amount of reputation information you can develop this way may simply not be large enough. An obvious alternative is to share reputation information through some trusted network, selectively choosing to trust the opinion of knowledgeable and trustworthy third parties.

There are a variety of other possibilities. For example, we may rely on legislation for protection, which would normally imply the need for some kind of policing to enforce it. We may rely on system administrators for protection against intruders, or on friends or family members for assistance protecting home PCs and for guidance on safe on-line practices. Ideally, there would be a clearly identified individual directly responsible for our safety and protection in any given situation, which leads to the final question.

Who is responsible, and what recourse do I have? No matter how careful you are and what assurance you have, things will always go wrong. Your decision whether to engage in an activity may well depend on the likelihood that you can limit your exposure or obtain compensation for any harms you may suffer. For example, limits of liability for credit card transactions make it possible for people to use them in confidence that a fraudulent charge in their name will not ruin them financially.

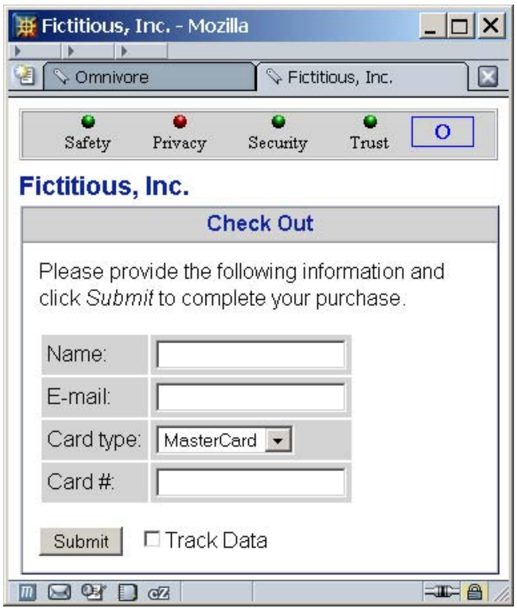
The Internet masks geographic boundaries and creates many opportunities for anonymity. These characteristics may be seen as virtues in many respects, but they limit opportunities for recourse and limit our ability to hold people accountable. For example, you may be comfortable engaging in a transaction with someone in your own town because you understand the local laws and customs and how they can be used to seek compensation or retribution. You may be considerably less comfortable engaging in the same transaction with someone in a distant country whose laws and customs you do not know and whose government is not committed to your well being. The anonymous, borderless nature of the Internet makes it difficult to make these kinds of determinations.

4. OMNIVORE

The underlying hypothesis of this paper is that timely answers to the risk management questions outlined above will allow users to engage in activities with greater confidence and with better outcomes with respect to privacy, safety and security. *Omnivore* is an experimental system we have developed to help evaluate the hypothesis. It addresses the risk management questions in the specific context of typical Internet users accessing the Web with a browser and represents several preliminary steps toward a solution in that domain.

We also use the name Omnivore to refer to the underlying philosophy of our approach. It does not seek to make the Internet an inherently safer or more secure place, but rather to assist individual users to understand the risks and potential for exposure and provide them with tools to help manage the risks. The primary strategy is to provide timely answers to the risk management questions, based on the belief that users will make more informed decisions if they have more (relevant) information on which to base them – information they can understand and utilize without a high cognitive burden.

The US Federal Bureau of Investigation (FBI) has a sys-


Fictitious, Inc. - Mozilla

Omnivore | Fictitious, Inc.

● Safety ● Privacy ● Security ● Trust O

Fictitious, Inc.

Check Out

Please provide the following information and click *Submit* to complete your purchase.

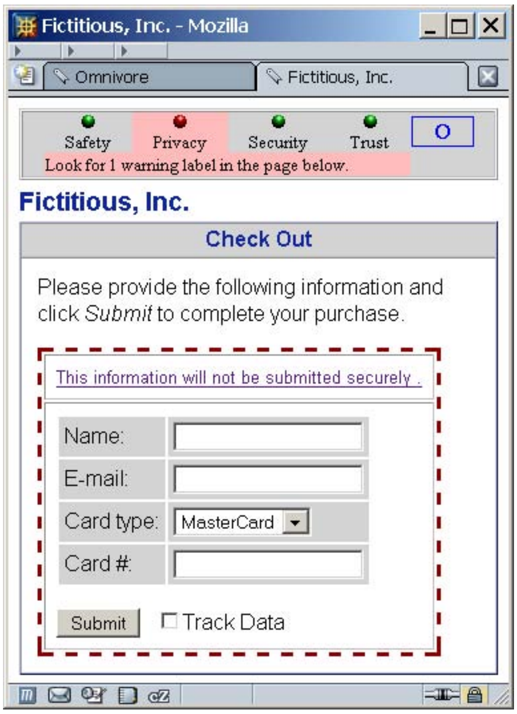
Name:	
E-mail:	
Card type:	MasterCard
Card #:	

Submit □ Track Data

Figure 1: A Web page instrumented by Omnivore. The Privacy light is red, indicating that there may be a privacy concern associated with this page. The other indicators are all green, and initially no other information is presented.

tem known as *Carnivore* that is deployed at the Internet Service Provider (ISP) level and assists with court ordered monitoring of e-mail. It is widely (though inaccurately) believed to be a data harvesting system that monitors the flow of all information through the ISP in a way that is largely invisible to end-users [14]. A number of systems have been developed under the name *Herbivore*, a name that may be a reaction to the Carnivore system. One is a spam detection network [19], another a system for distributed anonymous communication [16]. Certainly there is value in both capabilities, though the underlying philosophy is still one of restricting information flow. The name *Omnivore* is intended to reflect an intermediate position: bidirectional transparency through a free and visible flow of information. It is a philosophy that is similar in spirit to Brin's Transparent Society [11].

The design of the Omnivore system is inspired by a combination of ideas from the AT&T Privacy Bird developed by Cranor et al. [3] and the *safe staging* approach of Whitten and Tygar [7]. The Privacy Bird is a plug-in for the Internet Explorer browser that uses a traffic light metaphor to communicate privacy related information. For each site visited by the browser, the plug-in attempts to obtain the P3P privacy policy statement for the site and compare it with privacy preferences the user has configured. If the policy respects the user's preferences, a small green singing bird icon is displayed in the title bar. If a conflict with a user's preferences is detected, the bird turns red (and swears). You can obtain a concise description of the situation by clicking on the red bird icon. If the plug-in is unable to make a definite determination for any reason, an orange bird is displayed.


Fictitious, Inc. - Mozilla

Omnivore | Fictitious, Inc.

● Safety ● Privacy ● Security ● Trust O

Look for 1 warning label in the page below.

Fictitious, Inc.

Check Out

Please provide the following information and click *Submit* to complete your purchase.

This information will not be submitted securely .

Name:	
E-mail:	
Card type:	MasterCard
Card #:	

Submit □ Track Data

Figure 2: Clicking on a red indicator (the Privacy indicator in this case) toggles the display of additional information.

Ideally, the bird would remain green while a user travels the Internet, giving a reasonable measure of assurance that the sites being visited have reasonable policies. Infrequently, the bird would turn red (a visual cue that is known to be effective in drawing attention) and the user would have the choice of avoiding the site or proceeding with the understanding that the site may not respect the user's privacy preferences. If the user chooses to invest the time, their understanding can be further informed through interaction with the tool. Most of the time, the cognitive load created by the tool is extremely low and the users' attention is engaged at precisely the time that privacy concerns arise.

Omnivore takes exactly this idea and simply multiplies it across categories. Where the Privacy Bird is concerned solely with privacy policies, the Omnivore architecture allows evidence to be sought from arbitrarily many sources. Figure 1 shows a Web page instrumented by Omnivore. The *dashboard* at the top of the page displays a set of indicator lights that monitor a number of categories. Ideally, as you travel the Internet, the lights will generally remain green. When one turns red, clicking on it will display more information regarding the cause.

Consider, for example, a situation in which an input form is displayed on a secure Web page (*i.e.*, a page retrieved using the HTTPS protocol), but where the *action* attribute of the form – the destination to which the data will be sent – is specified by an insecure URL (one beginning with `http://`). This situation does occur in practice, though it is more likely the result of programming error than malicious intent. Nevertheless, it is a potential risk. In contemporary browsers, there is no way of knowing in advance that data will be submitted in clear text short of inspecting the HTML source for the page, which is clearly beyond typical Internet users. Contemporary browsers do detect this condition and generate a warning, but the timing and execution are poor. An alert appears in a modal dialog only after the user attempts to submit the form, and it describes an unexpected situation that is difficult to understand. The overwhelming majority of users will simply click OK in this circumstance and proceed without understanding. Is there a more effective way to provide guidance?

The *safe staging* approach developed by Whitten and Tygar may be ideal for this situation. Their technique draws from a substantial body of research into product warning labels. They argue that a disciplined approach to the delivery of small packages of information at appropriate times will lead to more effective communication of information related to one's security context. In our HTML form example, this idea can be applied by detecting at the outset that a form in a page retrieved from an `https:` location is to be submitted to an `http:` destination and literally drawing a warning label around the form, as shown in Figure 2. The label precisely highlights the area of concern and displays a succinct summary of the potential risk formatted as a hyperlink that will display more detailed information in a separate window.

The Omnivore prototype functions primarily as an HTTP(S) proxy. It inspects requests as they are sent from the browser, and it inspects responses returned from servers, instrumenting them as appropriate with dashboards, warning labels and additional supporting information.

The system creates a persistent record of each user's browsing history and uses it to help detect and react to suspicious conditions, as discussed below. It uses basic proxy authentication to create an association between users and their corresponding records, which makes it possible for multiple users to share a browser on the same machine, for a single user to maintain separate profiles, for multiple users to use a single proxy server concurrently, and it enables roaming.

Filter modules are then added to the framework to provide detection and notification of suspicious conditions in particular categories. Each module provides all of the functionality behind a single dashboard light. Hereafter, the term *filter module* (or simply *module*) will refer to all of the logic associated with detection and notification in a category such as these.

In normal usage when a user requests a Web page, the proxy inserts the Omnivore dashboard and supporting data into the page and returns it to the browser. It then continues to process the request in the background, collecting relevant information to be brought to the user's attention if appropriate. The dashboard lights remain orange during this period, indicating that the status is unknown. If and when a determination has been made in a particular category, the corresponding light is turned green or red, depending on the outcome. Some categories may take considerably longer than others to complete. When a light turns red, a user may click on it to obtain further information. If the condition relates to items displayed in the page, then warning labels are dynamically inserted in response to clicking a red indicator. Clicking the light again removes the warnings. This helps reduce clutter when many categories report problems for the same page. (If risk could be accurately estimated, then clutter might also be managed by displaying only the most risky conditions.)

For the purpose of the prototype, the dashboard and warning labels have been implemented using JavaScript. Markup for the dashboard is inserted into the top level page, and one `<script>` tag is inserted for each active module (*i.e.*, one for each dashboard light). These tags cause the browser to spawn asynchronous requests for the results from each filter module. Data is delivered in JavaScript data structures in the responses to these requests, and the final line in each response is a JavaScript function call that triggers processing of the results. Dynamic page updates, such as warning label display, is achieved through JavaScript using the level 2 DOM API. Although this arrangement has proven to be an adequate platform for experimentation, there are significant drawbacks. These are discussed further in Section 6, along with possible alternatives.

In the remainder of this section we consider the potential Omnivore has to address each of the risk management questions.

1. *What could go wrong?* One of the primary functions of Omnivore is to draw attention to areas of potential concern. This is done visually in the current design, using the safe staging approach to group succinct guidance together with the source of concern. Summary information is displayed below the indicator lights in the dashboard when information must be communicated about elements that have no visual representation. Non-speech audio cues may also prove effective for this purpose and could be integrated into future versions.

2. *How likely is it, and what damage would it cause to me or to others?* To a first approximation, the risk posed by a possible harm is proportional to the product of the likelihood that it will occur and the damage it would cause. Lightening can strike you in a storm, but you may still choose to go out because it is so unlikely that the risk is acceptable. It is much more likely that you will get wet in the storm, but you may still choose to go out because the consequences of wetness are (usually) minor.

   Likelihood is an empirical matter. In principle, if it were possible to know whenever something goes wrong (see question 3), one could estimate the likelihood of something going wrong based on historical knowledge of how frequently it has done so in similar situations. In its current form, Omnivore provides an effective way to communicate this information but does not address the problem of obtaining it.

   Some kinds of damage are easily measured, such as financial loss; others, such as damage to reputation, are more subjective. Quantifying the potential damage is a difficult challenge, and a core part of the CRA

Grand Challenge. Again, Omnivore can assist in delivering this information, but does not contribute to a solution for obtaining it.

3. *How would I know if it went wrong?* Detection of dangerous conditions and anomalous or malicious activity is the focus of considerable research effort covering a wide variety of threats such as network borne attacks and intrusions, malicious insiders, or accidental breaches of privacy policy within a complex organization. Omnivore can utilize results from any of these efforts to help detect and communicate potentially unsafe conditions to users.

   Its architecture also creates new opportunities for monitoring and detection. For example, we have been experimenting with mechanisms for detecting privacy policy violations through data tracking. When an input form is detected in an incoming page, an extra checkbox labeled "Track Data" is added to the form (Figure 2). If the user activates this feature, then outgoing data is perturbed sufficiently to be able to distinguish it from all previous tracked submissions (adding additional characters to a name, for example) and permanently recorded along with the destination and time of the transaction.

   If a user is subsequently contacted using this information (which will be obvious from the misspelled name), they can consult Omnivore through a direct Web application interface to the proxy to determine where the data came from. Because the Omnivore proxy is able to inspect all communication with the browser, it is in a position to perform the same detailed tracking performed by the Web sites it contacts. When a user consults Omnivore to determine where contact information may have come from, Omnivore is able to utilize the relationship graph it builds from its own tracking data to suggest how information may have found its way to unexpected places. If it appears to have originated from a site that promised not to divulge it, then a policy violation has been detected.

   Our experiments in this area have been intriguing, but we are still a long way from a usable solution. Ideally, detection would be completely automated and would not rely on a reporting step; the process would be driven by empirical data regarding the likelihood of various risks; and users would not have to work so hard to understand complex webs of relationships in order to benefit from this approach. Still, it is a promising approach towards helping users build, over time, a more sophisticated understanding of the nature of the space they are traveling.

4. *What reason do I have to believe that it won't go wrong?* The answer, in a word, is *trust*. Consider the following definition of trust from Jøsang and Presti, from their paper analyzing the relationship between risk and trust [4]:

   > *Trust is the extent to which one party is willing to depend on somebody, or something, in a given situation with a feeling of relative security, even though negative consequences are possible.*

   The notion that one might feel relatively secure even though negative consequences are possible is just an alternative way to characterize what we have been calling the risk management approach. Our objective is not so much to make negative consequences impossible as to give users good reason to feel relatively secure. In this context, where good outcomes are not guaranteed by internal constraints, *reputation* becomes central. Evidence that trust has not been betrayed in the past is a reasonable basis for believing that it will not be betrayed in the future.

   One of the primary functions of Omnivore is to gather and communicate empirical information regarding potential risks. In some cases, such as the example described for question 3 above, Omnivore can build its own repository of evidence. In most cases, however, the evidence available locally will be insufficient. A critical aspect of Omnivore's operation, therefore, is to seek information from external sources. Reputation systems and networks are an obvious choice. It may also be useful for Omnivore to participate directly in a reputation network, sharing the empirical data it has developed over time.

5. *Who is responsible to ensure that it doesn't go wrong, and what recourse do I have if it does?* This question is the domain of Alternative Dispute Resolution systems (ADRs) [13]. Omnivore could be used to make users aware when such mechanisms exist, how they can be used and what their limitations are. It is not obvious how else it might contribute to a solution, though there are possibilities relating to evidence it might provide as input to a dispute resolution process.

5. RELATED WORK

The factors that influence the perceptions of typical Internet users regarding security, privacy and trustworthiness have been well documented. For example, factors strongly influencing the perceived credibility of a Web site include visual appeal and professionalism, effective navigation and ease of use, brand reputation, and recommendation or affiliation with reputable third parties [22, 15]. Similar factors influence perceptions of security even though they have little bearing on actual security. Detailed information about site security is usually available to those with the motivation to seek it out and the technical training to understand it, but typical users generally do not have the technical background necessary to utilize it effectively [23].

Several studies have provided strong evidence that the abilities of typical users fall far short of the level needed to use security features in the way their designers intended. The most widely known example is a usability study of the PGP 5.0 e-mail encryption product [6] in which experienced e-mail users struggled to send a signed and encrypted message (7 of 12 subjects ultimately failed). The study of KaZaA file sharing ([17]) suggests that most users of this software are exposing far more of their personal computer files than they realize, and that they have difficulty correcting the situation even with substantial guidance. Friedman et al. found that users commonly misjudge whether a Web browser connection is secure or not [9].

A number of different approaches have been combined to address the situation. Considerable work has been done to understand how users perceive their on-line world and what skills they can bring to bear on it. In addition to the Web credibility work outlined earlier, Dourish et al. have studied typical user environments and how users perceive them [8]. A broad study by Friedman et al. focused on the risks and harms users perceive on-line [10].

Some projects have focused on addressing specific problems that manifest themselves in the user interface. Ye and Smith have demonstrated the ability for Web sites to spoof the visual cues browsers display to communicate a state of relative security. They have proposed a technique called Synchronized Random Dynamic (SRD) borders as a way of clearly delineating browser generated visuals from server generated material [24].

Millett et al. have worked on techniques for rendering browser cookies more visible and involving users in their management [20]. It is questionable, however, whether cookie management tools and increased visibility will improve the privacy situation for users. Even if you are presented with complete information when asked by the browser whether you would like to accept a given cookie, on what basis can you decide? Ideally you would choose to accept the cookie if the potential benefits were likely to outweigh the potential harms, or some similar criterion. But how can you estimate the harms that may result from cookie based tracking and profiling?

Several other projects seek to render suspicious activity more visible, an idea that is central to Omnivore. The Bugnosis software, for example, monitors incoming Web pages for Web bugs (images, frames or other objects that generate new outgoing browser requests that carry contextual information to third party destinations), modifying the pages to render the bugs visible [12].

Others have advocated a more comprehensive approach. For example, Grinter and Smetters have called for significant rethinking of how security is embedded into applications [18]. They propose that the security architecture of an application be driven by a user-centered threat model, and that it be able to base security related actions on user intent and do so in a visible way. The safe staging technique of Whitten and Tygar may be a good step in this direction [7].

6. FUTURE WORK

Preliminary experience with our prototype system has been encouraging, but Omnivore currently represents only an initial step towards effective risk management for end users. Additional work is needed on several fronts.

First, the design process must remain user-centered and be based on empirical validation of the central ideas. Although user interface design has been derived from previous human factors work [24, 7, 3], specific design elements must be tested. Will users understand the connection between Omnivore dashboard signals and warning labels displayed in a page? Warning labels are displayed only in response to a user query, and not by default, but will they still disrupt page content to an unacceptable degree?

The risk categories we have chosen for the prototype represent only an initial guess. The intention is that Privacy will be used for disclosure of personal information; Security for attacks or intrusions that could compromise the local host; Trust for reputation and other trustworthiness issues; and Safety will be reserved for critical conditions that could result in physical exposure to risk. However, it is fairly certain that these labels will not be widely understood. Much work is needed to develop an effective taxonomy.

It is clear from our experience that much of the functionality currently implemented in the Omnivore proxy and through JavaScript would be more effective if added as a core capability to the browser client. For example, placing the dashboard directly in a Web page creates several problems. It scrolls out of view, it can conflict with the dynamic content generated by some pages, and it is relatively easy to spoof. JavaScript provides no security model and is poorly suited for implementing security features. Many difficulties could be easily addressed if the dashboard was implemented as a browser toolbar.

The risk management approach can be applied to other contexts. Consider for example the e-mail attachments that are currently being exploited to carry viruses. It is commonly claimed that educating users to not open unknown attachments would significantly reduce the problem, but education has not yet proven effective. Would users be educated more effectively if e-mail clients drew warning labels around potentially unsafe attachments? It would be interesting to explore such applications of the idea.

Much of Omnivore's usefulness is predicated on the existence of a body of empirical data to be used as a source of reputation evidence and as a basis for quantitative estimates of likelihood. A detailed record of a user's browsing activity over time would be valuable in this regard. It could be used, for example, to estimate what tracking and profiling is possible on the basis of past browsing activity. Such a detailed record would clearly pose privacy concerns and measures are necessary to safeguard private data.

Finally, it would be extremely useful to have access to a larger body of reputation evidence through a trusted network of friends and associates. A reputation network capable of scaling to a useful size is a significant challenge.

7. CONCLUSION

To summarize, we have described a general approach to the problem of dealing with the risks and potential harms that users face as they interact with complex information systems. We have explored this approach in the specific context of typical Internet users accessing the World Wide Web through a browser interface. Our preliminary experience has been sufficiently encouraging that we intend to continue with empirical evaluation of the ideas using a refinement of the existing prototype implementation. Success of this approach depends critically on the availability of a rich source of empirical data that can be used as a basis for assessing risk. Further work is required to identify or create suitable resources and to secure trusted paths from the repositories to applications such as Omnivore that would use it.

8. REFERENCES

[1] Anne Adams and Martina Angela Sasse. Users Are Not The Enemy: Why users compromise security mechanisms and how to take remedial measures. *Communications of the ACM*, 42(12):40–46, December 1999.

[2] Computing Research Association. CRA Conference on Grand Research Challenges in Information Security & Assurance. http://www.cra.org/Activities/grand.challenges/security/, November 16-19 2003.

[3] Lorrie Faith Cranor, Manjula Arjula, and Praveen Guduru. Use of a P3P user agent by early adopters. In *Proceeding of the ACM workshop on Privacy in the Electronic Society*, pages 1–10. ACM Press, 2002. See also http://www.privacybird.com/.

[4] Audun Jøsang and S. Lo Presti. Analysing the Relationship Between Risk and Trust. In T. Dimitrakos, editor, *Proceedings of the Second International Conference on Trust Management*, April 2004.

[5] Ross J. Anderson. *Security Engineering: A Guide to Building Dependable Distributed Systems*. John Wiley & Sons, Inc., New York, 2001.

[6] Alma Whitten and J. D. Tygar. Why Johnny can't encrypt: A usability evaluation of PGP 5.0. In *Proceedings of the Eighth USENIX Security Symposium (Security'99)*, pages 169–183, Washington, DC, USA, 23–26 August 1999. USENIX Association. Available as http://www.cs.cmu.edu/~alma/johnny.pdf.

[7] Alma Whitten and J. D. Tygar. Safe Staging for Computer Security. Presented at the CHI'03 workshop on HCI and Security Systems, April 6 2003. Available as http://www.andrewpatrick.ca/CHI2003/HCISEC/hcisec-workshop-whitten.pdf.

[8] Paul Dourish, Rebecca E. Grinter, Brinda Dalal, Jessica Delgado de la Flor, and Melissa Joseph. Security Day-to-Day: User Strategies for Managing Security as an Everyday, Practical Problem. Technical Report UCI-ISR-03-5, Institute for Software Research, University of California, Irvine, June 2003.

[9] Batya Friedman, David Hurley, Daniel C. Howe, Edward Felten, and Helen Nissenbaum. Users' Conceptions of Web Security: A Comparative Study. In *Conference Extended Abstracts on Human Factors in Computer Systems*, pages 746–747, Minneapolis, Minnesota, USA, April 20-25 2002. ACM Press.

[10] Batya Friedman, Helen Nissenbaum, David Hurley, Daniel C. Howe, and Edward Felten. Users' Conceptions of Risks and Harms on the Web: A Comparative Study. In *Conference Extended Abstracts on Human Factors in Computer Systems*, pages 614–615, Minneapolis, Minnesota, USA, April 20-25 2002. ACM Press.

[11] David Brin. *The Transparent Society: Will Technology Force Us to Choose Between Privacy and Freedom?* Perseus Publishing, May 1998.

[12] Bugnosis Web Bug Detector. http://www.bugnosis.org/.

[13] Anne Carblanc. Privacy protection and redress in the online environment: Fostering effective alternative dispute resolution. In *Proceedings of the 22nd International Conference on Privacy and Personal Data Protection*, Venice, September 28-30 2000.

[14] Electronic Privacy Information Center. The Carnivore FOIA Litigation. http://www.epic.org/privacy/carnivore/, May 2002.

[15] B.J. Fogg, J. Marshall, O. Laraki, A. Osipovich, C. Varma, N. Fang, P. Jyoti, A. Rangnekar, J. Shon, P. Swani, and M. Treinen. What makes Web sites credible? A report on a large quantitative study. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 61–68, Seattle, Washington, 31 March – 5 April 2001. ACM Press.

[16] Sharad Goel, Mark Robson, Milo Polte, and Emin Gün Sirer. Herbivore: A Scalable and Efficient Protocol for Anonymous Communication. Technical Report TR2003-1890, Cornell University Computing and Information Science, February 2003. See also http://www.cam.cornell.edu/~sharad/herbivore/.

[17] Nathaniel S. Good and Aaron Krekelberg. Usability and privacy: a study of KaZaA P2P file-sharing. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 137–144, Fort Lauderdale, Florida, April 5-10 2003. ACM Press.

[18] Rebecca E. Grinter and D. K. Smetters. Three Challenges for Embedding Security into Applications. Presented at the CHI'03 workshop on HCI and Security Systems, April 6 2003. Available as http://www.andrewpatrick.ca/CHI2003/HCISEC/hcisec-workshop-grinter.pdf.

[19] Herbivore Distributed Anti-Spam Filter. http://www.herbivore.us/, 2004.

[20] Lynette I. Millett, Batya Friedman, and Edward Felten. Cookies and web browser design: toward realizing informed consent online. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 46–52. ACM Press, 2001.

[21] Kevin D. Mitnick, William L. Simon, and Steve Wozniak. *The Art of Deception: Controlling the Human Element of Security*. John Wiley & Sons, first edition, October 4 2002.

[22] Cheskin Research. Trust in the wired americas. Available from http://www.cheskin.com/, July 2000.

[23] Carl W. Turner. Investigating consumers' perceptions of security and privacy of e-commerce web sites. In *Proceedings of the Usability Professionals Association Conference*, Orlando, Florida, 2002.

[24] Zishuang (Eileen) Ye and Sean Smith. Trusted Paths for Browsers. In *Proceedings of the 11th USENIX Security Symposium (Security'02)*, pages 263–279, San Francisco, August 5-9 2002. USENIX Association.