Internet Security, Vulnerability Disclosure, and Software Provision

by

Jay Pil Choi*, Chaim Fershtman**, and Neil Gandal***

September 2005
Revised, July 2006

Abstract

This paper presents a simple model to investigate issues related to the economics of Internet security. Large networks are typically more vulnerable to security breaches because the success of the network provides hackers with a greater incentive to exploit potential vulnerabilities. Our model incorporates this "negative network effect" as a central feature of the model and approaches the problem from the perspective of economic incentives facing software vendors, hackers, and users. In particular, we analyze software vendors' private incentives to make an upfront investment in the quality of the software to reduce potential vulnerabilities and to announce vulnerabilities if they are found. We compare the market outcome to the socially optimal one. We also discuss implications of our model for formulating appropriate public policies to enhance information security.

JEL Classification: L100, L630.
Keywords: Internet security, software vulnerabilities, disclosure policy.

* Department of Economics, Michigan State University, 101 Marshall Hall, East Lansing, Michigan 48824-1038, Tel: 517-353-7281, E-mail: choijay@msu.edu

** The Eitan Berglas School of Economics, Tel Aviv University, Tel Aviv 69978, Israel, Tel: 972-3-640-7167, E-mail: fersht@post.tau.ac.il

*** Department of Public Policy, Tel Aviv University, Tel Aviv 69978, Israel, Tel: 972-3-640-6742, E-mail: gandal@post.tau.ac.il

We thank Jacques Lawarree, Shlomit Wagman, several anonymous reviewers from WEIS 2005, and seminar participants from WEIS 2005 for their helpful comments. We are grateful to Ity Shurtz for research assistance. A research grant from Microsoft is gratefully acknowledged. Any opinions expressed are those of the authors.

1. Introduction

Despite all the access and convenience provided by the Internet, the new information technology also poses serious security problems. According to a recent study conducted by America Online and the National Cyber Security Alliance (2004), 80 percent of the computers in the US are infected with spyware and almost 20 percent of the machines have viruses. Although some of the so-called killer viruses turned about to be hoaxes, several real viruses have done significant damage. According to *The Economist* magazine, the Blaster worm and SoBig.F viruses from the summer of 2003 resulted in $35 Billion in damages.[1]

In addition, it appears that the time between the announcement of a software vulnerability and the time in which that vulnerability can be exploited has declined significantly. According to *The Economist*, the time between disclosure to attack was six months for the Slammer worm (January 2003) which infected 90% of all unprotected computers within 10 minutes, while the time for the Blaster worm (August 2003) was only three weeks.[2] The high cost of these viruses emphasizes the increasing importance of cyber security.

The solutions to the problem, however, cannot rely on purely technical measures. The management of information security requires a much deeper understanding of various actors who play distinct roles in the system. For instance, the Slammer, Blaster, and Sobig.F worms exploited vulnerabilities even though security patches had been released by Microsoft. That is, although the patches were widely available, relatively few users applied them.[3] We thus present a model to approach the problem from the perspective of economic incentives facing software vendors, hackers, and users.

Our focus in this paper is how software vulnerabilities affect the firms that license[4] the software and the consumers that purchase software. In particular, we model three decisions of the firm:

[1] http://www.economist.co.uk/science/displayStory.cfm?story_id=2246018.
[2] Ibid.
[3] In practice, patches are typically released only when several bugs have been fixed. If a patch is released for each bug, the vulnerability can easily be "reverse engineered" and exploited by hackers. But when a cumulative patch is released, it is more difficult to reverse engineer and find the individual vulnerabilities.
[4] Like other products based on intellectual property, the intellectual property in software is typically "licensed" for use, not sold outright. Someone who purchases a music CD buys the physical CD and the right to play the music

(i) An upfront investment in the quality of the software to reduce potential vulnerabilities, (ii) a policy decision whether to announce vulnerabilities, and (iii) a price for the software. We also model two decisions of the consumer: (i) whether to purchase the software and (ii) whether to apply a patch. To analyze these decisions, we consider a profit maximizing software vendor in a three-stage game. In the first stage, the firm chooses the level of investment that determines the quality of software, i.e., the number of vulnerabilities. In the second stage, the firm sets the software price and announcement policy. In the third stage, consumers make purchasing decisions. As usual, we apply backward induction to solve the subgame perfect equilibrium of this dynamic game. More specifically, given a fixed level of software quality (security), we first examine whether the software vendor will announce software vulnerabilities and the price that the vendor will charge for the software. We then examine the level of investment in software security. We compare private incentives to announce vulnerabilities and invest in software security with those of a (second-best) social planner. We show that, given the level of investment and the set of buyers, the firm has less incentives to announce vulnerabilities than the social planner. Given the incentives to announce vulnerabilities and consumers' patching decisions, we also show that the firm's incentives to invest in security are suboptimally low. Our analysis will provide new insight on the issues of Internet security and have implications for formulating appropriate public policies to enhance information security.

Network effects are prevalent in the computer software industry. Network effects are typically thought to benefit consumers and firms that have coalesced around a standard. However, large networks are also more vulnerable to security breaches, because the success of the network provides hackers with a greater incentive to exploit potential vulnerabilities. Our model incorporates this "negative network effect" feature.

Our paper builds on the budding literature at the "intersection" of computer science/engineering and economics on cyber security.[5] Much of the early work in this area has focused on the lack of incentives for individuals or network operators to take adequate security precautions. This is

---

under specific circumstances (which do not include the right to play it on the radio, etc). Software is similarly "licensed for use."

[5] See Anderson (2001) for an introduction to the topic. Another helpful source is Anderson's "Economics and Security Resource Page" page: http://www.cl.cam.ac.uk/users/rja14/econsec.html. For a wealth of articles on computer security, see Bruce Schneir's web page at http://www.schneier.com/essays-comp.html.

because there is a security externality; individuals (or network operators) will not adequately protect against viruses on their computer (networks), since a large portion of the cost of the spread of the virus is incurred by others.[6]

Varian (2000) argues that assigning liability to network operators would likely lead to a market for insurance.[7] Since insurance firms typically will insure only those who engage in preventive measures,[8] he argues that the incentive for providing security would be increased, that is, the security externality would be internalized.

Computer Emergency Response Team/Coordination Center (CERT/CC) is a public agency that acts as an intermediary between users who report vulnerabilities to CERT/CC and vendors who produce the software and the patches. When informed by a user about a vulnerability, CERT/CC conducts research into the matter. If the user has indeed uncovered a security vulnerability, CERT/CC then informs the software vendor and gives it a 45 day "vulnerability window." This allows the firm time to develop a patch. After the 45 day period, CERT/CC will disclose the vulnerability even if a patch has not been made available. Recently, a private market for vulnerabilities has developed where firms such as iDefense act as intermediaries, paying those who report vulnerabilities and providing the information to software users who have subscribed to the service.

Several papers in the literature examine the effects of creating a market for vulnerabilities.[9] Camp and Wolfram (2004) heuristically discuss this issue. Kannan and Telang (2004) employ a formal model to examine whether a market based mechanism is better than the setting in which a public agency (CERT/CC) acts as an intermediary. Schechter (2004) formally models the market for vulnerabilities and Ozment (2004) formally shows how such a market can function as

[6] An interesting question is whether the disclosure of vulnerabilities hurts the market value of the software vendors. If so, this would suggest that software firms have incentives to improve the quality of the software prior to releasing it. Using a data set with 114 vulnerability announcements, Wattal and Telang (2004) show that software firms lose on average 0.76% of their market value when a vulnerability is discovered.
[7] For a formal analysis of system reliability and free riding, see Varian (2002).
[8] Think about the automobile industry – insurers will typically not insure a car against theft or will charge a higher premium unless protective devices such as an alarm and/or an immobilizer has been installed.
[9] Many of these papers discussed in this section have been presented in workshops on the economics of information security (WEIS). See the references for the web pages of these workshops.

an auction. In these settings, there is no strategic role for a software vendor, which is a main feature of our analysis.

Arora, Telang, and Xu (2004) examine the optimal policy for software vulnerability disclosure. Although they indeed have a strategic software vendor, the vendor strategy is limited to whether it will release a patch and if so when to release the patch. August and Tunca (forthcoming) have a strategic software vendor as well, but the vendor strategy is limited to pricing the software. In our paper, we examine incentives for vendors to invest in quality in order to reduce vulnerabilities, how vendors will price the software, and whether vendors will announce vulnerabilities and release a patch.

Png, Tang, and Wang (2006) develop a model of information security that is close to ours. They analyze the strategic interactions among end-users of software and between users and hackers. Their focus is mainly on comparative statics results that analyze the direct and indirect effects of changes in the user cost of precaution and the rate of enforcement against hackers. Our focus, in contrast, is on software vendors' optimal decisions concerning voluntary disclosure and investment in security.

Finally, Garcia and Horowitz (2006) construct a game-theoretic model that analyzes the incentives to invest in Internet security by ISPs. They show that if ISPs cannot completely appropriate the social surplus created by investment in security (i.e., the social value derived from successful operation of Internet exceeds the revenue received by ISPs), there is potential for underinvestment. Their result, however, does not rely on the competitive nature of the market. Given their assumption about the nonappropriability of consumer surplus, it is not surprising that they derive an underinvestment result. Our model introduces heterogeneity of consumers and explicitly analyzes consumers' optimal purchase and patching decisions, which provides a micro-foundation for the non-appropriability of consumer surplus and identifies the bias in the firm's investment in security.

The remainder of the paper is organized in the following way. Section 2 sets up the basic model. In Section 3, we derive the equilibrium disclosure behavior of the firm and patch

decisions of consumers and compare the private incentives to disclose security vulnerabilities with the social incentives. Section 4 derives the optimal investment level selling strategies for the monopolist and compares the market investment level with the socially optimal one. Section 5 contains concluding remarks in which we discuss the robustness of the main results and consider potential extensions.

2. The Model

There is a profit-maximizing firm that decides on investment, price, and disclosure policy. Consumers maximize utility and can either purchase one unit of the software or not purchase at all. If consumers purchase the software and the firm discloses a patch, consumers have to decide whether to install the patch, which is costly. Hackers do not have a formal objective function, but there are parameters that describe the outcome of their behavior.[10]

Consumers

There is a continuum of consumers whose number is normalized to 1. They are heterogeneous in terms of their valuation of software and the cost of being attacked in case of security breach. We represent consumer heterogeneity by parameter $\theta \in [0,1]$. It is distributed according to cdf $F(.)$, where F is differentiable and strictly increasing with continuously differentiable derivative f. More specifically, we assume that the value of software to consumer type θ is given by $v_1 + \theta v_2$ in the *absence* of any security problem, where $v_1, v_2 > 0$. Damage from each security problem to consumer type θ is assumed to be θD. Hence, both the consumer value and damage are linear functions of consumer type. This assumption reflects the fact that high valuation consumers tend to suffer more damage from an attack. Our qualitative results do not depend on this feature; we made these assumptions for tractability. The cost of downloading/installing a patch for consumers, when one is available, is given by c. We assume that c is a constant and that $c<D$. If $c>D$, no one will ever install a patch and the analysis is completely uninteresting. The cost of applying patches includes shutting the system down and restarting it, which can be quite expensive for a critical system. Moreover, a patch to an operating system or other low-lying

[10] See Png, Tang, and Wang (2006) for an analysis that explicitly models hackers as a strategic player. They assume that hackers derive enjoyment from an attack on a user provided that they are not discovered by an enforcement agency.

code may create a conflict with key third-party applications. As a result, users may need to conduct extensive testing before implementing patches, which takes time and monetary resources.[11]

Software Vendor

The software vendor maximizes profits and needs to make decisions regarding the level of investment that determines the security of the software, price, and disclosure policy. We assume that there is a negative relationship between the level of investment and the number of security bugs, that is, $n'(I)<0$, where n is the number of security problems and I is the level of investment.[12] We also assume that the marginal investment required to reduce additional vulnerabilities increases as the software gets more secure. That is, each additional vulnerability is harder to find and hence requires more investment with $n''(I)>0$.[13] The firm also makes a disclosure policy on whether it will announce a security problem and simultaneously release a patch when a security bug is discovered after sales of software. [14]

Hackers and Technology

Hackers exert effort, which is costly. We assume that either (i) they receive monetary rewards for causing damage, or (ii) that they have an intrinsic motivation that comes from causing damage. In both cases, hackers work harder when they expect to create more damage, i.e., with a larger unprotected network.[15] The following parameters describe technology and hacker behavior.

[11] See Meta Group Staff (2002).
[12] For simplicity, we treat n as a continuous variable. Alternatively, we can interpret n as a parameter representing the severity of the security problem.
[13] The model assumes a fixed number of vulnerabilities at release time. That is not always the case; for example, new technologies can create new vulnerabilities. The results are robust to the number of vulnerabilities at release time being a multiple of n(I).
[14] In our setting, an announcement is only made when the patch is available; this is the typical practice in the industry.
[15] Of course, hacker incentives are more complicated. Some hackers, for example, are likely motivated by the possibility of recognition within the hacker community that comes from finding and exploiting vulnerabilities. See Nissenbaum (2004) and the references cited within.

- η-- The probability that the firm will find the problem before the hackers, that is, the percentage of problems that the firm finds first (or the probability that the problem is reported to a firm by a benevolent user). We assume that η is exogenous.

- γ -- The probability that each security bug will be discovered by hackers on their own. Thus, γ is the probability of attack if the problem is not announced. If the firm announces the vulnerability and releases a patch, the probability of attack equals one because the problem is revealed to hackers.[16] The assumption captures the fact that the release of a patch makes reverse engineering feasible for the hacker and increases the likelihood of attack. The increased probability of attack due to information revelation is the downside of announcing the vulnerability with a patch.

Expected Damage

It is a well-known fact that large networks are more vulnerable to security breaches. There are two reasons for this: first, the success of the network provides hackers with a greater incentive to exploit potential vulnerabilities; second, the presence of a large number of unprotected computers facilitates the spread of harmful viruses and worms and can be used for denial of service attacks by hackers. To incorporate this "negative network effect" feature, we assume that the expected damage to a consumer increases with the number of consumers who are exposed. If hackers find a problem first or no patch was released, all consumers who purchased the software are exposed to security vulnerability. More precisely, we assume:

- $\gamma(1-\eta)B\theta D$ -- The expected damage for consumer type θ from an attack on a problem first found by hackers, where B is the number of consumers (all of whom do not have a patch since the problem was first found by hackers). The expected damage is a function of the number of consumers who do not have patch, because this provides the incentive for hackers to exert more effort and make a more sophisticated attack.[17]

[16] Arora, Krishnan, Nandkumar, Telang, and Yang (2004) find empirical evidence that vulnerability disclosure increases the number of attacks per host and patching decreases the number of attacks per host.

[17] The formal model is identical under the following interpretation: γN = the probability of attack if the problem is not announced. Hence, a larger network (of consumers without patches) increases the probability of attack by hackers. In this interpretation, $\gamma<1$ represents the difficulty of exploiting a vulnerability when reverse engineering

If a patch is released before hackers find the problem, those who are exposed are consumers who purchase the software but do not install the patch.

- $\eta N\theta D$ – the expected damage for consumer type θ who do *not* patch, where N is the number of consumers who purchase the software but do not install a patch.

Timing

The timing of the game is as follows

- Stage 1: Firms choose the level of investment (I) that determines the number of vulnerabilities, n(I).
- Stage 2: Firms set price (p) and announcement policy (A) and consumers make purchasing decision.
- Stage 3: If there is an announcement of security vulnerability and a patch is released, consumers make patching decisions.

As usual, we apply backward induction to derive the subgame perfect equilibrium of the game between the firm and consumers. In the next section, we derive the equilibrium behavior of the firm and consumers in vulnerability disclosure and patching decisions.

3. Vulnerability Disclosure and Patching Decisions

Given the investment level and the number of potential security bugs $n(I)$, we analyze consumers' optimal purchasing and patching decisions. We first need to determine the optimal consumer choice under all possible subgames. First consider the case in which the firm commits to announcing vulnerabilities.

(via a patch) is not possible, and N^e is the probability of attack if the problem is announced (γ=1 when the problem is announced). The assumption here is that the release of a patch makes reverse engineering feasible for the hacker and increases the likelihood of attack. Arora, Krishnan, Nandkumar, Telang, and Yang, (2004) find empirical evidence that not disclosing vulnerabilities may result in fewer attacks.

The Firm Announces Vulnerabilities

Let $W_p(\theta, B)$ be the net consumer value from buying the software and installing the patch when the consumer type is θ and the number of software buyers is B:

(1) $W_p(\theta, B) = [v_1 + \theta v_2] - \gamma(1-\eta)B\theta Dn(I) - \eta n(I)c.$

The first term, $[v_1 + \theta v_2]$, is the consumer valuation which is increasing in type θ; the second term, $\gamma(1-\eta)B\theta Dn(I)$, is the expected damage, should the hackers find the vulnerabilities before the firm. The expected damage increases in γ, where higher values of γ mean that there is a higher probability of attack when vulnerabilities are not announced. The expected damage also increases in $(1-\eta)$ which is the probability that the hacker discovers the vulnerabilities before the firm. The expect damage also increases in N, the size of the unprotected consumer network that is equal to B in case of no announcement, consumer type θ, and n(I), the number of software vulnerabilities. The third term is the overall expected cost of the patches (to consumers who patch) if the firm finds the problems first.

Similarly, let $W_{np}(\theta, B, N)$ be the net consumer value from buying the software, but not installing the patch, where B is the total number of buyers, and N is the number of buyers who do *not* install the patch.

(2) $W_{np}(\theta, B, N) = [v_1 + \theta v_2] - \gamma(1-\eta)B\theta Dn(I) - \eta N\theta Dn(I)$

The second term is again the damage when the hackers find the vulnerabilities before the firm, while the third term is the expected damage when the firm finds the vulnerabilities before the hackers. There is potential damage in the latter case to consumers who do not employ a patch because the release of the patch facilitates reverse engineering.

Since high type consumers also suffer more from security vulnerabilities, $W_p(\theta, N)$ and $W_{np}(\theta, B, N)$ can be either increasing or decreasing in θ, depending on parameter values. In this paper, we limit our attention to the case where both $W_p(\theta, N)$ and $W_{np}(\theta, B, N)$ are increasing in θ, that

is, the software value net of the damage is increasing in consumer type. To ensure this, we make the following assumption.[18]

A1. $v_2 > \gamma n(0)D$

When both $W_p(\theta, N)$ and $W_{np}(\theta, B, N)$ are increasing in θ, it is immediate that the consumers' purchase decision can be characterized by a threshold type θ^*, that is, the set of buyers $\mathcal{B} = \{\theta | \theta \in [\theta^*, 1]\}$ and the number of buyers $B = 1 - F(\theta^*)$.

If we compare equations (1) and (2), the only difference between $W_p(\theta, N)$ and $W_{np}(\theta, B, N)$ is the last term in each of the equations. Consider an equilibrium in which every buyer patches with N = 0. In such a case, however, we have $W_p(\theta, 0) > W_{np}(\theta, B, 0)$. That is, "not patching" is a better option for any individual consumer, and there is a unilateral incentive to deviate. Hence, there cannot be an equilibrium in which all consumers patch. This illustrates that problems with vulnerabilities cannot be solved (exclusively) "ex post" by having everyone patch; because of the incentive to be a free rider, such an equilibrium cannot exist.[19]

From equations (1) and (2) above, given D, c and N, there is a marginal consumer in patch decision– denoted $\hat{\theta}$ – such that for $\theta \geq \hat{\theta}$, a consumers installs the patch and for $\theta < \hat{\theta}$, a consumer does not install the patch. From these equations, the marginal consumer type $\hat{\theta}$ given N is characterized by $W_p(\hat{\theta}, B) = W_{np}(\hat{\theta}, B, N)$, which implies $\hat{\theta} = \frac{c}{ND}$. See Figure 1.[20]

[18] If assumption A1 is violated, we cannot rule out the case in which, without any investment in security, the software has too many security vulnerabilities, and thus higher type consumers may have lower reservation values than lowerer type consumer who have less concern with security. See Choi, Fershtman, and Gandal (2005) for an analysis dealing with such a possibility.

[19] Png, Tang, and Wang (2006) and August and Tunca (forthcoming) also identify the same type of free-rider problem in user security. See also Ayres and Levitt (1998).

[20] If $\hat{\theta} > 1$, no consumers patch.

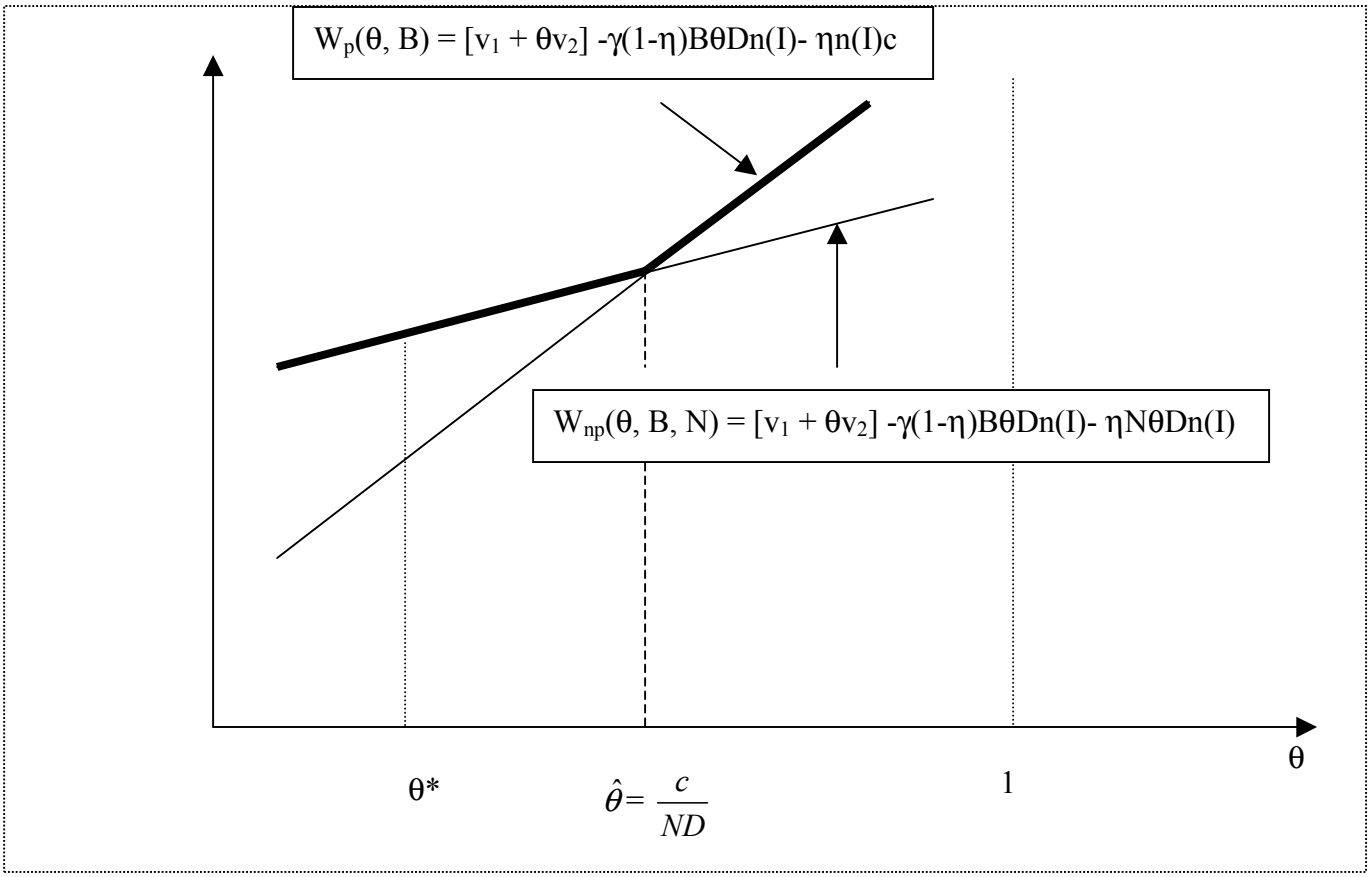

Figure 1: Optimal Patching Decision for Consumers

In equilibrium, we need to have a consistency condition that $N = F(\hat{\theta}) - F(\theta^*)$. Thus, given the set of buyers $\mathbb{B} = \{\theta | \theta \in [\theta^*, 1]\}$, the patch decision is characterized by a threshold type $\hat{\theta}$ ($>\theta^*$), which is implicitly defined by:

$$(3) \qquad \hat{\theta} = \frac{c}{[F(\hat{\theta}) - F(\theta^*)]D}$$

Lemma 1. There is no equilibrium in which every consumer patches. Suppose that the set of software buyers is given by $\mathbb{B} = \{\theta | \theta \in [\theta^*, 1]\}$. Given an announcement, the consumers' patch decision is characterized by a cutoff rule in which consumers do not patch if $\theta \in [\theta^*, \hat{\theta})$ and patch

if $\theta \in [\hat{\theta}, 1]$, where $\hat{\theta} > \theta^*$ and is implicitly defined by (3). This implies that the marginal consumers in the purchasing decision (type θ^* consumers) do not patch, given announcements.

Lemma 1 is consistent with the real world observation that there are consumers who do not patch even though patches are available.

Given θ^*, we can define $\hat{\theta}(\theta^*)$ as the value of $\hat{\theta}$ that satisfies (3). Then, we have:

$\mathbf{N}$ = the set of those who do not patch = $\{\theta | \ \theta \in [\theta^*, \hat{\theta}(\theta^*))\}$.

$N = F(\hat{\theta}(\theta^*)) - F(\theta^*)$.

Since we know that the marginal type for purchase decision θ^* do not patch when the firm announces security vulnerabilities, the software price of selling to the set of buyers $\mathbf{B} = \{\theta | \ \theta \in [\theta^*, 1]\}$ is given by

(4) $p^A(\theta^*) = [v_1 + \theta^* v_2] - \gamma(1-\eta)[1 - F(\theta^*)]\theta^* Dn(I) - \eta[F(\hat{\theta}(\theta^*)) - F(\theta^*)]\theta^* Dn(I)$

Note that the equilibrium price is decreasing in the number of vulnerabilities. This will give software firms incentive to invest in security, since this will increase the equilibrium price of the software. We analyze the incentives to invest in security in Section 4.

The following two lemmas will be used later.

Lemma 2. $\frac{d\hat{\theta}}{d\theta^*} > 0$

Proof. Given θ^*, $\hat{\theta}$ is implicitly defined by

(3)′ $\hat{\theta}[F(\hat{\theta}) - F(\theta^*)]D = c$.

By totally differentiating (3)′, we have

$$\{[F(\hat{\theta}) - F(\theta^*)] + \hat{\theta}F'(\hat{\theta})\}Dd\hat{\theta} - \hat{\theta}F'(\theta^*)Dd\theta^* = 0$$

Therefore, we have

$$\frac{d\hat{\theta}}{d\theta^*} = \frac{\hat{\theta}F'(\theta^*)}{[F(\hat{\theta}) - F(\theta^*)] + \hat{\theta}F'(\hat{\theta})} > 0$$

Lemma 3. $\frac{dN}{d\theta^*}<0$

Proof. $\frac{dN}{d\theta^*} = F'(\hat{\theta})\frac{d\hat{\theta}}{d\theta^*} - F'(\theta^*) = F'(\hat{\theta})\frac{\hat{\theta}F'(\theta^*)}{[F(\hat{\theta})-F(\theta^*)]+\hat{\theta}F'(\hat{\theta})} - F'(\theta^*)$

$$= -\frac{F'(\theta^*)[F(\hat{\theta})-F(\theta^*)]}{[F(\hat{\theta})-F(\theta^*)]+\hat{\theta}F'(\hat{\theta})} < 0.$$

The Firm Does Not Announce Vulnerabilities

Now we need to analyze what happens if the firm does not announce vulnerabilities. For a given network size of unprotected consumers, this is better for consumers who do not patch, since it reduces the probability that they will suffer damage. But this is not necessarily more profitable for the firm or better for consumers, since the number of unprotected consumers will be higher under this strategy, and hence the expected hacker damage will be higher as well. This, of course, lowers profits and consumer willingness to pay.

The value to the consumer of type θ from no announcement (W_{na}) is given by

$W_{na}(\theta, B) = v_1 + \theta v_2 - \gamma B\theta Dn(I)$

Once again, our assumption A1 ensures that $W_{na}(\theta, B)$ is increasing in θ. When $W_{na}(\theta, N^e)$ is increasing in θ, the consumers' purchase decision can be characterized by a threshold type θ* as before, that is, the set of buyers $\mathbf{B} = \{\theta | \theta \in [\theta^*, 1]\}$ and the number of buyers $B = 1 - F(\theta^*)$. Without announcement, $B = N = 1 - F(\theta^*)$.

Given that the marginal type for purchase decision is θ*, the software price is given by

(5) $p^{NA}(\theta^*) = [v_1 + \theta^* v_2] - \gamma[1 - F(\theta^*)]\theta^* Dn(I)$

Private Incentives to Announce Vulnerabilities

Suppose that the monopolist considers selling to the set of consumers whose types belong to [θ*, 1]. Then, the monopolist will announce vulnerabilities if and only if $p^{A}(\theta^*) \geq p^{NA}(\theta^*)$.

Proposition 1. The monopolist announces security vulnerabilities with patches if and only if $\gamma \geq \Psi(\theta^*) \equiv \frac{F(\hat{\theta}(\theta^*)) - F(\theta^*)}{1 - F(\theta^*)} = 1 - \frac{1 - F(\hat{\theta}(\theta^*))}{1 - F(\theta^*)}$.

Proof. We can rewrite (5) as

(5)' $p^{NA}(\theta^*) = [v_1 + \theta^* v_2] - \gamma(1-\eta)[1 - F(\theta^*)]\theta^* Dn(I) - \gamma\eta[1 - F(\theta^*)]\theta^* Dn(I)$

Therefore, $p^{A}(\theta^*) \geq p^{NA}(\theta^*)$ if and only if

$\gamma\eta[1 - F(\theta^*)]\theta^* Dn(I) \geq \eta[F(\hat{\theta}(\theta^*)) - F(\theta^*)]\theta^* Dn(I)$, which yields the desired condition.

The intuition for Proposition 1 can be explained in the following way. The private incentives to make an announcement depend on the effect of announcement on the marginal consumer who we know does not install a patch. When an announcement is made, it aids reverse engineering of hackers and increases the probability of an attack. If γ is low and the probability of independent finding of vulnerability is low, the cost of announcement is relatively more important and thus no announcement is the optimal policy. However, when γ is high the attack is more likely to occur even without any announcement. In such a case, it is better to make an announcement and reduce the size of the unprotected network to minimize the damage from an attack.

Socially Optimal Disclosure Policy

Given that the monopolist sells to the set of consumers $\mathbb{B} = \{\theta | \theta \in [\theta^*, 1]\}$ and consumers makes their patching decision in a privately optimal way, we derive conditions under which announcements should be made from a social planner's viewpoint.

The expected social harm from the disclosure policy is given by:

$$SH^A =$$

$$\int_{\theta^*}^{1} \gamma(1-\eta)[1-F(\theta^*)]\theta Dn(I)dF + [\int_{\theta^*}^{\hat{\theta}(\theta^*)} \eta[F(\hat{\theta}(\theta^*)) - F(\theta^*)]\theta Dn(I)dF + \int_{\hat{\theta}(\theta^*)}^{1} \eta n(I)cdF]$$

The expected social harm from the nondisclosure policy is given by:

$$SH^{NA} = \int_{\theta^*}^{1} \gamma(1-\eta)[1-F(\theta^*)]\theta Dn(I)dF + \int_{\theta^*}^{1} \gamma\eta[1-F(\theta^*)]\theta Dn(I)dF$$

Needless to say, the socially optimal disclosure policy is to announce vulnerabilities if and only if $SH^A \leq SH^{NA}$.

Proposition 3. The monopolist's incentive to announce vulnerabilities is less than that of the social planner.

Proof. The social planner will announce vulnerabilities if and only if $SH^A \leq SH^{NA}$, that is,

$$(6) \qquad \underbrace{\eta[F(\hat{\theta}(\theta^*)) - F(\theta^*)]Dn(I)\int_{\theta^*}^{\hat{\theta}(\theta^*)} \theta dF}_{A} + \underbrace{\eta n(I)c[1-F(\hat{\theta}(\theta^*))]}_{B} \leq$$

$$\underbrace{\gamma\eta[1-F(\theta^*)]Dn(I)\int_{\theta^*}^{\hat{\theta}(\theta^*)} \theta dF}_{C} + \underbrace{\gamma\eta[1-F(\theta^*)]Dn(I)\int_{\hat{\theta}(\theta^*)}^{1} \theta dF}_{D}$$

I prove the proposition by showing that condition (6) is less stringent than the condition for the monopolist to make announcements, that is, condition (6) is always satisfied whenever $\gamma \geq \Psi(\theta^*) \equiv \frac{F(\hat{\theta}(\theta^*)) - F(\theta^*)}{1-F(\theta^*)}$.

If we compare terms A and C in condition (6), it can be easily verified that A≤C if and only if $\gamma \geq \Psi(\theta^*) \equiv \frac{F(\hat{\theta}(\theta^*)) - F(\theta^*)}{1-F(\theta^*)}$. Thus, as far as terms A and C are concerned, the private and social incentives to disclose coincide. However,

$$\text{D} = \gamma\eta[1-F(\theta^*)]Dn(I)\int_{\hat{\theta}(\theta^*)}^{1} \theta dF$$

$$> \gamma\eta[1-F(\theta^*)]Dn(I)\hat{\theta}(\theta^*)[1-F(\hat{\theta}(\theta^*))]$$

$$= \gamma\eta[1-F(\theta^*)]Dn(I)\frac{c}{[F(\hat{\theta}(\theta^*)) - F(\theta^*)]D}[1-F(\hat{\theta}(\theta^*))] \text{ (see equation (3))}$$

If $\gamma \geq \Psi(\theta^*) \equiv \frac{F(\hat{\theta}(\theta^*)) - F(\theta^*)}{1 - F(\theta^*)}$,

$$\gamma\eta[1 - F(\theta^*)]Dn(I)\frac{c}{[F(\hat{\theta}(\theta^*)) - F(\theta^*)]D}[1 - F(\hat{\theta}(\theta^*))] \geq \eta n(I)c[1 - F(\hat{\theta}(\theta^*))] = \text{B}.$$

Therefore, $SH^A < SH^{NA}$ whenever $\gamma \geq \Psi(\theta^*) \equiv \frac{F(\hat{\theta}(\theta^*)) - F(\theta^*)}{1 - F(\theta^*)}$. *QED.*

Alternatively, we can prove Proposition 3 by showing that the threshold value of γ for a social planner's announcement is lower than that for the monopolist. More precisely, let γ^o and γ* be the threshold values for the social planner and the monopolist in the vulnerability disclosure decision, where $\gamma^o = \frac{[F(\hat{\theta}(\theta^*)) - F(\theta^*)]Dn(I)\int_{\theta^*}^{\hat{\theta}(\theta^*)} \theta dF + n(I)c[1 - F(\hat{\theta}(\theta^*))]}{[1 - F(\theta^*)]Dn(I)\int_{\theta^*}^{\hat{\theta}(\theta^*)} \theta dF + [1 - F(\theta^*)]Dn(I)\int_{\hat{\theta}(\theta^*)}^{1} \theta dF}$ and γ*= $\frac{F(\hat{\theta}(\theta^*)) - F(\theta^*)}{1 - F(\theta^*)}$. The logic in the proof of Proposition 3 indicates that $\gamma^o < \gamma^*$. Therefore, when $\gamma \in (\gamma^o, \gamma^*)$, we have a discrepancy between the private and social incentives to disclose security vulnerabilities; the social planner will announce them with patches, whereas the monopolist will not make an announcement.

The reason for the divergence in the incentives between the social planner and the monopolist can be explained by fact that the socially optimal disclosure policy depends on the effect of announcement on the *average* consumer, whereas the vendor's profit-maximizing disclosure policy depends on the impact on the *marginal* consumer. Notice that the average consumer type is higher than the marginal type in our model, which implies that the average consumer cares more about security and is more willing to apply patches when they are available.

Some security experts recommend mandatory public disclosure of the discovery of potential security vulnerabilities, both to warn system administrators and users and to spur the vendor involved to develop a patch as quickly as possible. However, our analysis suggests that such a policy is not always optimal, given that there are users who do not apply patches even if they are

available. Such a policy would be welfare improving when $\gamma \in (\gamma^o, \gamma^*)$. In contrast, a policy that mandates public disclosure can be welfare reducing when $\gamma < \gamma^o$.[21]

4. Investment Decision in Software Security

So far, we have analyzed incentives to announce vulnerabilities given the investment level (I) and thus the (expected) number of security bugs. Now we analyze the incentives to invest in security. Reducing the number of vulnerabilities increases the profitability of both strategies in the second stage (announcing and not announcing vulnerabilities), because if hackers indeed find the vulnerabilities, there will be less damage. This raises the willingness of consumers to pay for the software.

Let $p^{NA}(\theta^*, I)$ and $p^{A}(\theta^*, I)$ be respectively the market price of software when the monopolist makes no announcements and announcements, given the investment level of I and the marginal consumer of type θ* Then, given I, the monopolist solves the following problem:

(7) $\underset{\theta^*}{Max}\, p(\theta^*, I)[1 - F(\theta^*)]$,

where $p\ (\theta^*, I) = p^{NA}(\theta^*, I)$ or $p^{A}(\theta^*, I)$ depending on the firm's announcement policy.

Let $\theta^*(I)$ be the solution to (7). Then, the monopolist's investment decision can be written as:

(8) $\underset{I}{Max}\, \pi(I) = p(\theta^*(I), I)[1 - F(\theta^*(I))] - I$

By the envelope theorem, the privately optimal investment level I* is characterized by:

(9) $$\frac{d\pi(I)}{dI} = \frac{\partial p(\theta^*(I), I)}{\partial I}[1 - F(\theta^*(I))] - 1 = 0$$

Any investment level made by the monopolist will implicitly determine the set of buyers whose types belong to [θ*, 1]. For this set of consumers, we ask the question of whether the investment level made by the monopolist is optimal, *given* the monopolist's announcement policy and

[21] Mandated public disclosure of security vulnerabilities also induces the monopolist to invest more on security in the development stage. As will be seen in the next section, the monopolist's investment level is lower than the socially optimal level. Therefore, the policy of mandating public disclosure has potential to mitigate the investment problem.

consumers' patching decisions. In both cases where the monopolist makes announcements and no announcements, we show that the investment level is sub-optimally low.

The Firm Announces Vulnerabilities

In this case, $p(\theta^*, I) = p^A(\theta^*, I) = [v_1 + \theta^* v_2] - \gamma(1-\eta)[1 - F(\theta^*)]\theta^* Dn(I) - \eta[F(\hat{\theta}(\theta^*)) - F(\theta^*)]\theta^* Dn(I)$. Therefore, the optimal investment level is characterized by:

$$(10)\quad \underbrace{-\left[\gamma(1-\eta)[1-F(\theta^*)]D\frac{dn(I)}{dI}\theta^* + \eta[F(\hat{\theta}(\theta^*)-F(\theta^*)]D\frac{dn(I)}{dI}\theta^*\right][1-F(\theta^*)]}_{\text{Private Incentives to Invest (PI)}} = 1$$

Now we derive a second best investment level, given the sales strategy and announcement policy of the monopolist, and compare it with that chosen by the monopolist. Given the set of consumers [θ*, 1], social welfare as a function of the firm's investment level can be written as:

$$SW^A(I) =$$

$$\int_{\theta^*}^{1}[v_1 + \theta v_2 - \gamma(1-\eta)[1-F(\theta^*)]\theta Dn(I)dF - [\int_{\theta^*}^{\hat{\theta}(\theta^*)}\eta[F(\hat{\theta}(\theta^*)) - F(\theta^*)]\theta Dn(I)dF + \int_{\hat{\theta}(\theta^*)}^{1}\eta n(I)cdF] - I$$

Therefore, the first order condtion for the socially optimal level of investment is given by:

(11)

$$\underbrace{-\left[\gamma(1-\eta)[1-F(\theta^*)]D\frac{dn(I)}{dI}\int_{\theta^*}^{1}\theta dF + \eta[F(\hat{\theta}(\theta^*)-F(\theta^*)]D\frac{dn(I)}{dI}\int_{\theta^*}^{\hat{\theta}(\theta^*)}\theta dF + \int_{\hat{\theta}(\theta^*)}^{1}\eta c\frac{dn(I)}{dI}dF\right]}_{\text{Social Incentives to Invest (SI)}} = 1$$

We can show that SI > PI in the following way.

We first notice that $c = \hat{\theta}[F(\hat{\theta}) - F(\theta^*)]D$ (see equation (3)). Therefore, $c > \theta^*[F(\hat{\theta}(\theta^*) - F(\theta^*)]D$. As a result,

$$SI > -[\gamma(1-\eta)[1-F(\theta^*)]D\frac{dn(I)}{dI}\theta^*[1-F(\theta^*)]$$

$$+\int_{\theta^*}^{\hat{\theta}(\theta^*)}\eta[F(\hat{\theta}(\theta^*)-F(\theta^*)]D\frac{dn(I)}{dI}\theta^* dF + \int_{\hat{\theta}(\theta^*)}^{1}\eta\theta^*[F(\hat{\theta}(\theta^*)-F(\theta^*)]D\frac{dn(I)}{dI}dF]$$

$$= -[\gamma(1-\eta)[1-F(\theta^*)]D\frac{dn(I)}{dI}\theta^*[1-F(\theta^*)] + \int_{\theta^*}^{1}\eta[F(\hat{\theta}(\theta^*)-F(\theta^*)]D\frac{dn(I)}{dI}\theta^* dF]$$

$= PI.$

The firm does not announce vulnerabilities

In this case, $p\ (\theta^*, I) = p^{NA}(\theta^*, I) = [v_1 + \theta^* v_2] - \gamma[1 - F(\theta^*)]\theta^* Dn(I)$

Therefore, the optimal investment level is characterized by:

$$(12)\ \underbrace{-\left[\gamma[1-F(\theta^*)]D\frac{dn(I)}{dI}\theta^*\right]}_{\text{Private Incentives to Invest (PI)}}[1-F(\theta^*)] = 1$$

Given the set of consumers $[\theta^*, 1]$, social welfare as a function of the firm's investment level can be written as:

$$SW^{NA}(I) = \int_{\theta^*}^{1}[v_1 + \theta v_2 - \gamma[1-F(\theta^*)]\theta D n(I) dF - I$$

Therefore, the first order condition for the socially optimal level of investment, in the case of no announcement, is given by:

$$(13)\ \underbrace{-\left[\gamma[1-F(\theta^*)]D\frac{dn(I)}{dI}\int_{\theta^*}^{1}\theta dF\right]}_{\text{Social Incentives to Invest (SI)}} = 1$$

Once again, we can show that SI > PI.

$$\text{SI} > -\left[\gamma[1-F(\theta^*)]D\frac{dn(I)}{dI}\int_{\theta^*}^{1}\theta^* dF\right] = -\left[\gamma[1-F(\theta^*)]D\frac{dn(I)}{dI}\theta^*[1-F(\theta^*)]\right]$$

$$= \text{PI}$$

Thus, we have the following proposition.

Proposition 4. Regardless of the firm's announcement policy, the private incentive to invest in security is sub-optimally low.

Once again, the sub-optimality of the private investment decision can be explained by Spence's (1975) intuition for the monopolistic provision of quality. The monopolist's investment decision will depend on its impact on the *marginal* consumer, whereas the social planner's incentive depends on its effect on the *average* consumer.

With our assumption A1 that $v_2 > \gamma n(0)D$, the software value net of the damage is increasing in consumer type θ, regardless of the monopolist's investment level I and announcement policy. This implies that the average consumer type values the monopolist's investment more than the marginal consumer, which is responsible for the sub-optimally low level of investment by the monopolist. However, if assumption A1 is violated, the relationship between the average and marginal consumers depends on the level of investment. More specifically, if $v_2 < \gamma n(0)D$, there exists an investment level $\tilde{I}$ such that $v_2 = \gamma n(\tilde{I})D$. In such a case, if the investment level is less than $\tilde{I}$, the reservation values of consumers are decreasing in θ. Therefore, the set of buyers will be $\{\theta | \theta \in [0,\theta^*]\}$ rather than $\{\theta | \theta \in [\theta^*, 1]\}$ when the marginal type is θ^*. As a result, the marginal consumer values the investment more than the average consumer. It is possible to have a case of overinvestment when A1 is violated.[22]

Arora, Caulkins, and Telang (forthcoming) also investigate the impact of patching on software quality. In their model, the tradeoff facing a monopolist is between an early release with more security holes and a late release with a more secure product. They show that a software monopolist releases a product with fewer bugs but later than what is socially optimal, which can be considered as an overinvestment in our framework. Even if we have a case in which A1 is violated and we have an instance of overinvestment, the reason behind their result would be completely different from ours. They compare the market outcome with the investment level in the *first best* outcome, and the driving force in their result is the fact that a monopolist restricts output relative to socially efficient levels. In our model, we compare the investment levels *fixing* the output level, and the negative externality from consumers without patch is the main reason for the suboptimal level of investment.

Our model also shed light on some of the policy issues related to Internet security. One recent issue that attracted significant attention is the so-called "bug bounty" program. For instance, in 2004 the Mozilla Foundation announced the Mozilla Security Bug Bounty program that rewards users who identify and report security vulnerabilities in the open source project's software. Under the program, users who report security bugs that are judged as critical by the Mozilla

[22] See Choi, Fershtman, and Gandal (2005) for more detail.

Foundation staff can collect a $500 cash prize.[23] Independent security intelligence companies also offer a bounty for security bugs. TippingPoint, for instance, solicits hackers to report vulnerabilities in exchange for money under its "Zero Day Initiative" program. If a valid bug is found, TippingPoint notifies the maker of the flawed product and update its security products to protect users against exploitation of the flaw until an official patch is released. There is also an underground market for information on vulnerabilities. It has also been known that cybercriminals pay top dollar for previously undisclosed flaws that they can then exploit to break into computer systems.[24] TippingPoint is not alone in offering such a bounty. IDefense, another security firm, recently offered $10,000 to anyone who discovers a Windows flaw that leads to a critical fix under its controversial "Vulnerability Contributor Program." In response, Microsoft criticized the IDefense program, saying that it "does not believe that offering compensation for vulnerability information is the best way they can help protect their customers."[25]

In our model, the bounty program can have a bite only when $\gamma \geq \gamma^* = \frac{F(\hat{\theta}(\theta^*)) - F(\theta^*)}{1 - F(\theta^*)} = \frac{N}{B}$ and the firm acts on the reported security vulnerabilities with patches. Otherwise, the bounty program is irrelevant since the firm does not make an announcement, and an attack will take place when hackers find out the problem independently.[26] Therefore, we limit our attention to the parameter space in which $\gamma \geq \gamma^*$. The effect of a bounty program can be represented by an increase in η, thereby increasing the probability that the firm identifies the problem before the hackers. In the region in which $\gamma \geq \gamma^*$, it can be easily verified that both $W_p(\theta, B)$ and $W_{np}(\theta, B, N)$ increase (see equations (1) and (2)). As a result, all users can benefit from a bounty program. The use of a bounty program can also increase the firm's profits since it can charge a higher

[23] http://www.mozilla.org/security/bug-bounty.html
[24] See Evers (2005).
[25] See Gonsalves (2006).
[26] If hackers' incentives change due to the bounty program, (that is, hackers who find the vulnerabilities decide to sell them through the bounty program), it can have the beneficial effects of reducing attacks by reducing γ in our model. However, such a possibility is discounted by security experts. To quote Neel Mehta, the team leader of X-Force Research at Ineternet Security Systems, a security company, "I'd be surprised if the people who are finding these vulnerabilities in the hacker underground are motivated to sell them for a few thousand dollars to a security company, when they might make a lot more by holding onto them and using them for economically motivated hacking." See Evers (2005).

price.[27] However, we should also note unintended side effects of the bounty program on ex ante investment in security. Equation (10), which characterizes the incentives for the firm's ex ante investment, also reveals that an increase in η reduces the incentives to invest I when $\gamma \geq \gamma^*$. Thus, ex post optimal bounty program can undermine ex ante investment incentives, which has been shown to be sub-optimally low. Thus, the effect of the bounty program can be more subtle than it appears.[28]

5. Concluding Remarks

In this paper we develop a model that endogenizes three decisions of the firm: (i) An upfront investment in the quality of the software to reduce potential vulnerabilities, (ii) a policy decision whether to announce vulnerabilities, and (iii) a price for the software. We also modeled two decisions of the consumer: (i) whether to purchase the software and (ii) whether to apply a patch.

Our paper, of course, leaves some research questions unanswered. In this paper, we did not allow for intermediaries, like CERT/CC, who obtain vulnerability information from end users and encourage firms to develop patches and eventually disclose this information. Such an intermediary or mandatory disclosure requirement could improve welfare in the intermediate values of γ (more precisely, $\gamma \in (\gamma^o, \gamma^*)$) as we discussed above. However, for vulnerabilities for which $\gamma < \gamma^o$, an intermediary of voluntary disclosure can be counterproductive since the potential risk of attack is relatively small compared to the cost of revealing information to hackers who may reverse engineer. In addition, they can also have unintended consequences for the ex ante investment level in security. Hence, this suggests that one should be cautious before regulating this market.

In addition, we assumed a single software vendor and did not examine the time at which software is released. With competition in software provision and a dynamic setting with new

[27] It should be emphasized that we assume the bounty program, if offered by independent security companies, is implemented with "responsible disclosure." That is, the vulnerability will be disclosed only when a patch is available form software vendors.

[28] The bounty programs also raise an ethical question of potentially dealing with criminal hackers (known as black hats) or illegal groups.

consumers over time, there would potentially be two additional effects: (I) there would likely be increased investment in reducing software vulnerabilities due to competition and (II) If consumer valuations depended on network size, software firms might have an incentive to release products earlier to build up an installed base.

References

American Online and the National Cyber Security Alliance, *AOL/NCSA Online Safety Study*, October 2004.

Anderson, R., (2001), "Why Information Security is Hard," available at http://www.ftp.cl.cam.ac.uk/ftp/users/rja14/econ.pdf.

Arora, A., Caulkins, J.P., and R. Telang, "Sell First, Fix Later: Impact of Patching on Software Quality," *Management Science*, forthcoming.

Arora, A., Krishnan, R., Nandkumar, A., Telang, R., and Y. Yang, "Impact of Vulnerability Discolsure and Patch Availabilitry – An Empirical Analysis, mimeo 2004, available at http://www.dtc.umn.edu/weis2004/telang.pdf.

Arora, A., Telang, R., and X., Hao, "Optimal Policy for Software Vulnerability Disclosure," Carnegie Mellon Working Paper, 2004

August, T., and T. Tunca, "Network Software Security and User Incentives," *Management Science*, forthcoming.

Ayres, I., and S. D. Levitt, "Measuring the Positive Externalities from Unobservable Victim Precaution: An Empirical Analysis of Lojack," *Quarterly Journal of Economics*, 1998, 43-77.

Camp, L.J., and C. Wolfram, "Pricing Security," in L.J. Camp and S. Lewis, eds., Economics of Information Security, vol. 12, Advances in Information Security. Springer-Kluwer, 2004.

Cavusoglu, Huseyin, Hasan Cavusoglu, and Jun Zhang, "Economics of Security Patch Management," unpublished manuscript, 2006, available at http://weis2006.econinfosec.org/docs/5.pdf

Choi, Jay Pil, Chaim Fershtman, and Neil Gandal, "The Economics of Internet Security," December 2005, unpublished manuscript.

Evers, Joris, "Offering a Bounty for Security Bugs," CNET News.com, July 25, 2005.

Garcia, Alfredo and Barry Horowitz, "The Potential for Underinvestment in Internet Security" Implications for Regulatory Policy," 2006, unpublished manuscript.

Gonsalves, Antone, "Microsoft Slams Security Firm's Bounty for Windows Flaws," *Information Week*, February 21, 2006.

Kannan, K., and R. Telang, "Market for Software Vulnerabilities? Think Again," Carnegie Mellon Working Paper, 2004.

Kawamoto, D., *Study: Few Corporations Use Anti-Spyware Tools*, CNET News, October 27, 2004.

Helen Nissenbaum, "Hackers and the Contested Ontology of Cyberspace, New Media and Society, 6:195-217, 2004, available at www.nyu.edu/projects/nissenbaum/papers/hackers.pdf

Meta Group Staff, "META Report: Security Vulnerability Disclosures," January 2002, available at http://itmanagement.earthweb.com/it_res/article.php/947271

Ozment, A., "Bug Auctions: Vulnerability Markets Reconsidered," mimeo, available at http://www.dtc.umn.edu/weis2004/ozment.pdf

Png, Ivan, Tang, Qian, and Wang, Qiuhong, "Information Security: Use Precautions and Hacker Targeting," 2006, available at http://papers.ssrn.com/sol3/papers.cfm?abstract_id=912161

Schechter, S., "Computer Security, Strength and Risk: A Quantitative Approach," 2004, available at http://www.eecs.harvard.edu/~stuart/papers/thesis.pdf

Spence, A.M., "Monopoly, Quality, and Regulation," *Bell Journal of Economics*, 1975, 6, pp. 417-429.

Varian, H., "Managing Online Security Risks," New York Times; New York, N.Y.; Jun 1, 2000, available at http://www.sims.berkeley.edu/~hal/people/hal/NYTimes/2000-06-01.html.

Varian, H., 2002, "System Reliability and Free Riding," available at http://www.sims.berkeley.edu/resources/affiliates/workshops/econsecurity/econws/49.pdf.

Wattal, S., and R. Telang, "Effect of Vulnerability Disclosure on Market Value of Software Vendors -- An Event Study, CMU mimeo, 2004.