Economic Analysis of Incentives to Disclose Software Vulnerabilities

Dmitri Nizovtsev[♣] Marie Thursby[♠]

Abstract

This paper addresses the ongoing debate about the practice of disclosing information about software vulnerabilities through an open public forum. Using game-theoretic approach, we show that such practice may be an equilibrium strategy in a game played by rational loss-minimizing agents.

We find that under certain parameters public disclosure of vulnerabilities is desirable from the social welfare standpoint. The presence of an opportunity to disclose allows individual software users to reduce their expected loss from attacks and by doing so improves social welfare. We analyze the effect of several product characteristics and the composition of the pool of software users on the decisions to disclose and on social welfare and compare several public policy alternatives in terms of their efficacy in reducing the overall social welfare loss from attacks. Our results suggest that designing an incentive system that would induce vendors to release fixes sooner and improve the quality of their products should be among the priorities for any policymaking agency concerned with information security. Doing so would reduce individual incentives to disclose vulnerabilities, thus further reducing the potential damage from any given vulnerability.

Our preliminary analysis of information-sharing coalitions suggests that such entities have a positive effect only under a fairly restrictive set of conditions.

[♣] Corresponding author; Washburn University, dmitri.nizovtsev@washburn.edu

[♠] Georgia Institute of Technology, marie.thursby@gatech.edu

Introduction

The security of information systems in general and computer systems in particular is an important component of national security. This idea is widely recognized and has been reflected in numerous government documents.[1] Despite this fact, the state of information security is far from perfect.[2] Security flaws in software products remain widespread, making users of those products vulnerable to attacks. This was true in the early nineties (National Research Council, 1991) and remains true nowadays (Power, 2001). While not every vulnerability represents a major security risk, the damage from some of them may be devastating.[3]

In this paper, we are interested in practices related to managing the information about security flaws, or vulnerabilities, discovered in software. When a discovery is made by a malicious hacker, it usually leads to attacks on vulnerable computer systems with the intention of getting access to data, stealing sensitive information, or taking complete control over the system. Benign discoverers, on the other hand, try to handle information in a way that would benefit the community. However, opinions on what is the best for the community differ. Some prefer to simply inform the vendor about the discovered flaw and wait until the vendor chooses to fix it. The other extreme is the practice of making all the information about a vulnerability, including the tools for exploiting it, publicly available through an open forum. The former practice is sometimes referred to as "friendly disclosure" while the latter became known as "full disclosure". Naturally, practices combining elements of the two extremes may exist as well.

The practice of full disclosure has been subject to a heated debate within the computer community.[4] Parties opposing it call it irresponsible and claim it further aggravates the already

---

[1] For instance, *The National Strategy to Secure Cyberspace* (2003) states the following: "Our economy and national security are fully dependent upon information technology and the information infrastructure" (p.viii).

[2] As stated in *The Critical Foundations* (1997), "…we are convinced that our vulnerabilities are increasing steadily, that the means to exploit those weaknesses are readily available and that the costs associated with an effective attack continue to drop".

[3] Here are a few examples. Several major Web sites, including Amazon and Yahoo, that fell victim to the infamous February 2000 attacks claimed several million dollars in lost revenue and advertising income. (Hopper and Thomas, 2000). In a more recent incident, a Russian programmer was found responsible for an aggregate loss of approximately $25 million (U.S. Department of Justice, 2003).

According to the *2003 Computer Crime and Security Survey* performed by the Computer Security Institute (available from www.gosci.com), the average estimated loss from theft of proprietary information resulting from attacks equals $2.7 million per occurrence.

[4] See Pond (2000), Raikow (2000), Lasser (2002), and Bank (2004), to name a few.

poor state of information security by making systems more vulnerable to malicious attacks.[5] Proponents of full disclosure use two major arguments. One is that in order to protect themselves against an attack, end users of software need to be informed about potential threats. Since all systems are different in terms of composition and configuration, "being informed" takes more than receiving a simple announcement that the problem exists. Only giving users complete information about the vulnerability as soon as it becomes available can ensure they will be able to protect themselves in one way or another. The other argument is based on the belief that the responsibility for the poor quality of software products and in particular the existence of security holes lies with software vendors. Trying to beat their competitors to the market, vendors tend to reduce the time spent on testing their products, which results in the low quality of products released. One way to create incentives for product quality improvement by vendors is to increase the negative effect a discovery of a vulnerability has on them, and making information about such discoveries public does just that.

In this paper, our goal is to understand the economic incentives to disclose sensitive security-related information publicly. We model the process of making disclosure decisions as a game played by benevolent loss-minimizing agents and show that the current situation in regard to information disclosure practices represents a mixed strategy equilibrium of such a game. We analyze the effect of several product characteristics and the composition of the pool of software users on the decisions to disclose as well as public welfare. Finally, we compare several public policy alternatives in terms of their efficacy in reducing the overall social welfare loss from computer attacks.

The rest of the paper is organized as follows. The next section discusses existing research related to the issues of our interest. Section III discusses the assumptions of the model. The analysis of the basic version of the model is presented in section IV. Sections V through VII expand our analysis by considering extensions of the basic model. Policy implications of our findings are discussed in Section VIII. Section IX outlines possible directions for further research and concludes.

[5] Certain steps were recently made by lawmakers to correct the situation. A fairly recent example in that regard is the Digital Copyright Millenium Act (DCMA), which, among other things, makes it illegal to disclose any flaws that may exist in commercial software. DCMA was warmly welcomed by software vendors but caused mixed reaction among the rest of computer community, including some security experts (Pavlicek, 2002).

II. Related research

Sharing information about software vulnerabilities is a form of informational spillover. The existing literature on cooperation and information sharing in research and development (R&D)[6] suggests that such spillovers can be internalized by the means of cooperative R&D agreements, which reduce the firms' costs and lead to higher efficiency. Of that literature, papers focusing on "technology sharing cartels" (Baumol (1992), Petit and Tolwinski (1996)) are the most relevant to us and provide a few insights of interest. One is that small firms are likely to benefit more from such an agreement than large ones because of a difference in the amount of information acquired. This difference gets smaller as the cartel gets larger. Overall, cartel members are expected to get a competitive advantage over non-members as a result of cost reduction and risk diversification. At the same time, the majority of the papers cited above point at the free-riding problem in any joint research venture and the need to design a mechanism that would enforce the contributions.

Gordon et al. (2003) extends this approach to information security issues by discussing the viability and effects of Information Sharing and Analysis Centers created under a presidential directive. Gal-Or and Ghose (2004) provide a formal analysis of ISAC's and find that the benefit of such alliances increases with a firm's size and is greater in more competitive industries. This is somewhat different from earlier work on joint research ventures due to different model specifications.

Arora et al. (2003) studies vendors' decisions to invest in the security of their products and the timing of issuing patches for known vulnerabilities. It is shown that quality of software products and vendor's investment in patching technology are strategic substitutes. The presence of patching technology induces vendors to enter the market sooner with buggier software. Incentives to do so get stronger with an increase in the market size and the degree of competition. Arora et al. (2004a) shows that vendors always choose to issue patch later than is socially optimal. Beattie et al. (2002) focuses on the administrators' decision to apply a known patch.

Schneier (2000) used the concept of "window of exposure" to express the fact that the overall damage from attacks depends not only on their intensity but also on how long a security hole

[6] See Bhattacharya et al. (1990), d'Aspremont and Jacquemin (1988), Kamien et al. (1992), and Katz and Ordover (1990).

remains open. An empirical study by Arora et al. (2004b) uses the statistics on attacks to show that full instant disclosure of a discovered vulnerability has a twofold effect on the window exposure as it increases the number of attacks on one hand and induces vendors to respond to issue patches faster on the other hand. Wattal and Telang (2004) partially supports the view that full and immediate disclosure creates incentives for vendors to improve the quality of their products by showing that vulnerability disclosure results in some, albeit small, loss in the market value of the software vendor.

Several researchers have been interested in optimal policy for software vulnerability disclosure. Arora et al. (2004a) shows using a theoretical model that neither immediate disclosure nor non-disclosure is socially optimal. Kannan and Telang (2004) and Ozment (2004) analyze the viability of a market solution to the problem of information security that consists in offering monetary rewards for finding vulnerabilities.

While all the aforementioned research provides important insights into the behavior of various parties involved with information security issues, none of them directly analyzes factors affecting the users' decision to disclose. The goal of our paper is to fill this void by studying the foundations of the individual decisions to disclose security related information and factors affecting those decisions. This is especially important since no discussion of public policy in that regard is complete without proper understanding of decisions made by agents coming across such information.

Lerner and Tirole (2000) interpret full disclosure as an instrument used by security researchers to signal their abilities. It is known, however, that signaling incentive is stronger when the impact of effort on performance is greater (Holmstrom, 1999). Since anecdotal evidence from the field suggests that the process of discovery is largely random, our paper is looking for an alternative explanation for the practice of disclosing vulnerabilities.

III. Description of the model

A. Composition of the pool.

In our model, the computer community consists of "white hats" and "black hats". All agents of the same type are identical in their preferences and computer skills. The number of agents of

each type is denoted *W* and *B*, respectively. There are also an unspecified number of software vendors, one for each software product.[7]

All agents have access to software products that contain flaws, or "bugs", not known at the time the product is released. Agents can obtain information about bugs in a software product either through a discovery process or from a public forum if such is available.

B. Timeline.

We assume that the probability of an individual discovery, π, is a linear function of time, $\pi = r \cdot t$. Higher values of *r* correspond to more transparent bugs and a higher probability of a discovery.[8] The value of *r* is the same for all agents who know about a bug.

We allow for the possibility of independent discoveries of the same bug and treat discovery as a Poisson process. If the probability of a discovery made by an agent in a unit of time is equal to *r,* then in a pool of *(B+W)* identical agents the expected number of discoveries within the same unit of time will be equal to *(B+W) r*.[9]

The problem of interest is presented as a multi-period game. The length of a period is set equal to the expected time between two successive discoveries of the same bug, $\tau = \frac{1}{(B+W) \cdot r}$. All agents discount the future at the same continuous rate ρ. Hence an event that is to occur one period in the future is discounted by a factor of $\delta = e^{-\rho \cdot \tau} = e^{-\rho / r(B+W)}$.

C. Individual preferences and utility specifications

C.1. Black hats derive utility from attacking other users' systems. Thus, the expected utility of a black hat is

[7] The exact number of vendors is not important since we consider one product at a time, therefore only one vendor is involved.

[8] Parameter *r* can also be thought of as one of the metrics of product quality. The more time and effort the vendor puts into testing and debugging the product before its release, the less likely is a discovery of a bug in that product within a given time period.

[9] An alternative specification is possible, which would address the widespread belief that black hats may have more interest in finding bugs and therefore put more effort into finding them. In that case the expected number of discoveries within a unit of time becomes *(KB+W) r,* where $K > 1$ accounts for a greater effort on the black hats' part. We chose to preserve a simpler notation and make *B* account not only for the number of black hats in the community but for their effort as well.

$$E(U_B) = \sum_{i=1}^{\infty} \delta^i (U_{ai} - \sigma \cdot C_{Bi}), \quad (1)$$

The positive term, U_{ai}, represents the utility from performing an attack. U_{ai} = 1 if an attack[10] is performed in period i and 0 otherwise. The negative term represents the expected cost of performing an attack in period i and equals the product of the probability of being caught, σ, and the punishment imposed in that case, C_{Bi}. The decision process of a black hat is therefore trivial – once he obtains information about a security bug, he randomly chooses a victim[11] and performs an attack, unless the cost of doing so is prohibitively high.

C.2. Vendors.

Once a security flaw is found in a product, the vendor of the product needs to decide whether and when to issue a fix. Two factors are important here. First, there is a fixed cost, F, of developing a fix and distributing it to all product users. Second, every attack that successfully exploits a security hole negatively affects the product's and therefore the vendor's reputation, causing a decrease in future profits. The size of that reputation loss is assumed to be linear in the number of attacks exploiting a bug in the product.

Vendors are minimizing their expected loss,

$$E(L_V) = \min\left\{F, \sum_{i=1}^{\infty} \delta^i E(n_{ai}) \cdot L_{rep}\right\}, \quad (2)$$

where L_{rep} is the size of the reputation loss resulting from each attack, and $E(n_{ai})$ is the expected number of attacks in period i. This implies that a vendor decides to issue a fix in a given period only if the discounted reputation loss from all future attacks is greater than the cost of developing a fix. Once the decision to issue and distribute a fix is made, it is carried out within the same period, and the game ends with no further damage to anyone.[12]

[10] For simplicity we assume that a black hat can attack no more than one system each period. This has no effect on our results.

[11] This is consistent with the evidence from the software industry, which shows that the majority of attacks are opportunistic – the victims are usually the systems with the lowest level of protection.

[12] Although applying a patch to all systems takes some time (Beattie, 2002), we assume the hole closes immediately after a patch is released. We do that because the focus of our paper is on decisions to disclose and not on decisions to patch. Arora et al. (2004a) considers two specifications, one where patch is applied immediately and the other with patching taking some time. The results for the two specifications of the model are not qualitatively different.

C.3. White hats. The expected utility of a white hat is given by

$$E(U_W) = \sum_{i=1}^{\infty} \delta^i [U_0 - \sigma_i C], \quad (3)$$

where U_W is the utility an agent obtains every period if his system works properly and σ_i is the agent's perceived probability of being attacked in period i. This probability depends on several factors, exogenous as well as endogenous, which are to be discussed later. C represents the loss suffered as a result of being attacked, which is assumed to be the same across all agents.

Without loss of generality, we set U_0 to zero. The white hat's problem therefore becomes to minimize the expected loss from attacks, $E(L_W) = \sum_{i=1}^{\infty} \delta^i \sigma_i C$.

White hats never attack other users' systems. Upon discovering a bug, a white hat immediately informs the vendor. If this is his only action, then the game proceeds to the next period, when someone else discovers the same bug. The other possibility is to disclose the information about the discovered vulnerability via an open public forum[13] along with informing the vendor. Once the information is made public, all black hats use that information to attack. We assume there is some delay, t_a, between the moment the information is disclosed and the time of the attack. The length of the delay depends, among other factors, on the amount of information that is disclosed. It is well known that the discoverer of a bug does not necessarily have complete information about it. That information may range from vague awareness of a security problem to the ability to write a fully functional exploit. This will in turn affect the amount of information that is disclosed. Naturally, the more information is disclosed publicly, the easier it is for black hats to perform attacks. A white hat discounts the future loss that may result from his decision to disclose by a factor of $\varepsilon = e^{-\rho \cdot t_a}$. We assume that the loss from an attack performed on a mass scale is substantial enough to force the vendor to issue the patch immediately, after which the game ends.

[13] In the basic version of the model, the only venue for disclosure is an open public forum. Later, we discuss another option in the form of a closed coalition of users.

IV. Analysis of the model – the benchmark case

A. Expected losses

In this section, we analyze the behavior of white hats seeking to minimize the discounted stream of expected future losses. In the basic version of the model discussed in this section, we consider an infinite game,[14] the only potential venue for disclosure is an open public forum, and agents do not attempt to develop a fix by themselves. Each of these three assumptions will be relaxed in the later sections.

Note that, under the current model specification, black hats never choose full public disclosure. This follows trivially from the notion that black hats obtain positive utility from attacks they perform, whereas full disclosure shortens the time period within which such attacks are possible.

In the basic version of the model, the only two pure strategies available to a white hat are "disclose" and "not disclose". There is also an infinite number of mixed strategies, in each of which "disclose" is played by each agent with some probability α, $0<\alpha<1$.[15]

When a white hat informs the world about the discovered vulnerability, then he faces the risk of being one of the victims of a mass-scale attack, after which the game ends. His expected loss from playing "disclose" is therefore

$$E(L_{WD}) = \frac{\varepsilon BC}{W} \qquad (4)$$

In the case when a white hat discoverer refrains from public disclosure, the game proceeds to period two, when another independent discovery of the same bug occurs. The process of discovery is random. The next discoverer may therefore be a white hat who will later choose to disclose (the probability of such an event is $\gamma_D = \frac{\alpha W}{B+W}$), a white hat who will choose not to disclose (with probability $\gamma_N = \frac{(1-\alpha)W}{B+W}$), or a black hat (with probability $\beta = 1-\gamma_D-\gamma_N = \frac{B}{B+W}$). In the first case the game ends after a massive attack. In the second case, the game proceeds to period three with no attacks and therefore no loss. In the last case, a

[14] Such a setup may be justified by the F/L_{rep} ratio being large so that the vendor does not fix the problem until full public disclosure actually occurs.

[15] We assume that the choice of strategy in regard to disclosure is a one-time decision and is never reversed at later stages of the game.

single attack occurs, and the game ends only if the agent in question happens to be a victim of that attack, otherwise it goes on. If the game lasts beyond the second period, then the game tree forks again, and the same logic can be applied over and over. Naturally, any losses that occur after the first period are discounted. The extensive form of the game under the listed assumptions is presented in Figure 1.

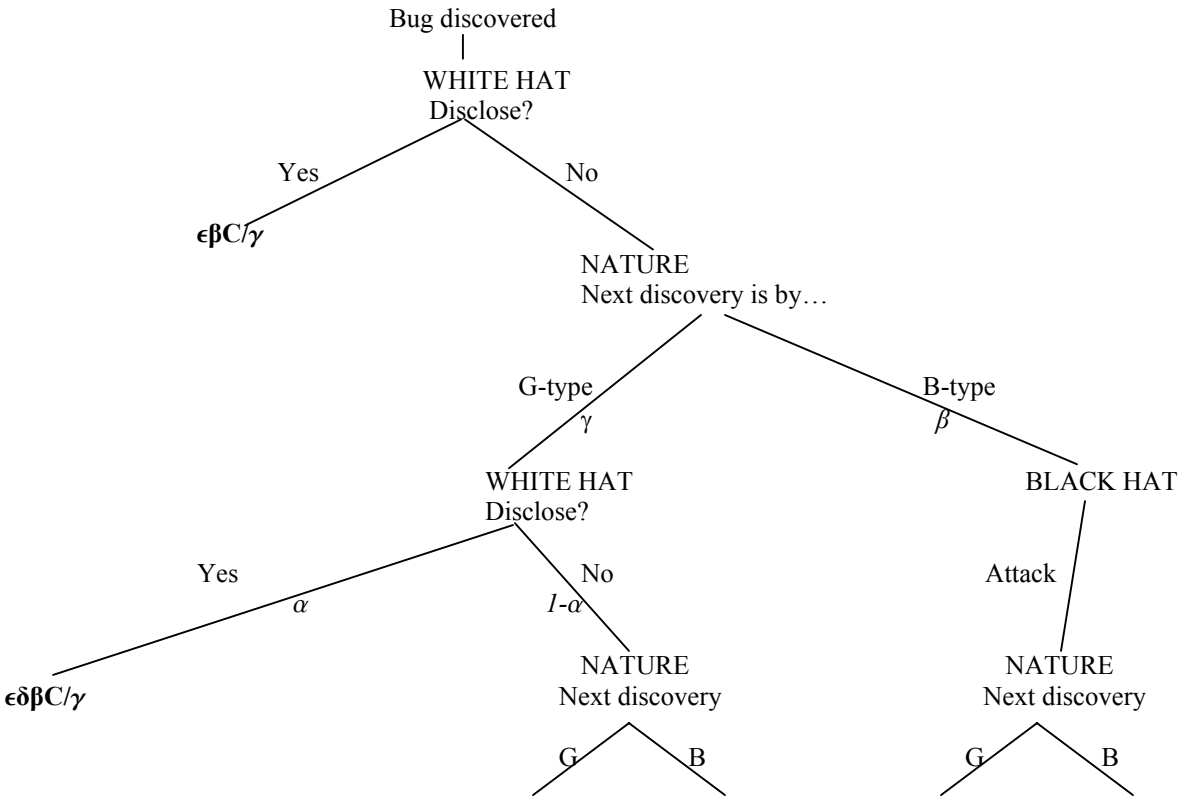

Figure 1. The game in the absence of effort.
The player that makes a decision at each node is shown in capital letters. NATURE stands for probabilistic events. Bold characters denote the payoffs of the initial discoverer at terminal nodes. Italicized characters next to a particular path show the probabilities that the game follows that path.

The expected loss of a white hat who refrains from disclosure can be expressed as

$$E(L_N) = \delta\beta C\left[\frac{\varepsilon\alpha}{1-\delta(\beta+\gamma_N)} + \frac{1}{W(1-\delta(\beta+\gamma_N))^2}\right] \quad (5)$$

(for the derivation of (5), see the Appendix).

B. Types and stability of equilibria

We find it useful to start with the notion that, when a white hat discovers a bug that has not been disclosed publicly, he does not know for how many periods the game has been played or how many other agents have access to the same information he has. Therefore every white hat plays the same game, shown in Figure 1.

Since all white hats are assumed identical and there are only two pure strategies available to them, the game has no more than two pure strategy equilibria. In one of them, all white hats disclose. We further refer to it as the full disclosure equilibrium, or FD-equilibrium. The other pure strategy equilibrium is when none of the white hats discloses, further referred to as the no disclosure, or ND-equilibrium. There may also exist mixed strategy equilibria (or M-equilibria), in each of which all white hats play the same mixed strategy by choosing disclosure with some probability α. A mixed strategy equilibrium exists when each player is indifferent between the pure strategies available to him, given the actions of other players. In the context of our model the existence of a mixed strategy equilibrium requires the existence of $0<\alpha^*<1$ for which $E(L_{WD}) = E(L_{WN})$.

Lemma 1. $E(L_{WD}) = E(L_{WN})$ *has only one real root,*

$$\alpha^* = \frac{1}{\varepsilon W(1-\delta)} - \frac{(1-\delta)(B+W)}{\delta W}. \tag{6}$$

All proofs are in the Appendix.

Proposition 1. In the game specified above, an FD-equilibrium exists if and only if

$$(1-\delta)(W + (1-\delta)B) \le \frac{\delta}{\varepsilon}. \tag{7}$$

Proposition 2. In the game specified above, an ND-equilibrium exists if and only if

$$(B+W)(1-\delta)^2 \ge \frac{\delta}{\varepsilon}. \tag{8}$$

Proposition 3. There is no more than one mixed strategy equilibrium in the game specified above. The necessary and sufficient condition for the existence of such an equilibrium is

$$0 < \delta - \varepsilon(1-\delta)^2(B+W) < \delta\varepsilon W(1-\delta). \quad (9)$$

Whenever a mixed strategy equilibrium exists, it is perfect.

Corollary 1: No more than one equilibrium exists in the specified game.

All three possible cases regarding the type of equilibria are presented in Figure 2 below.

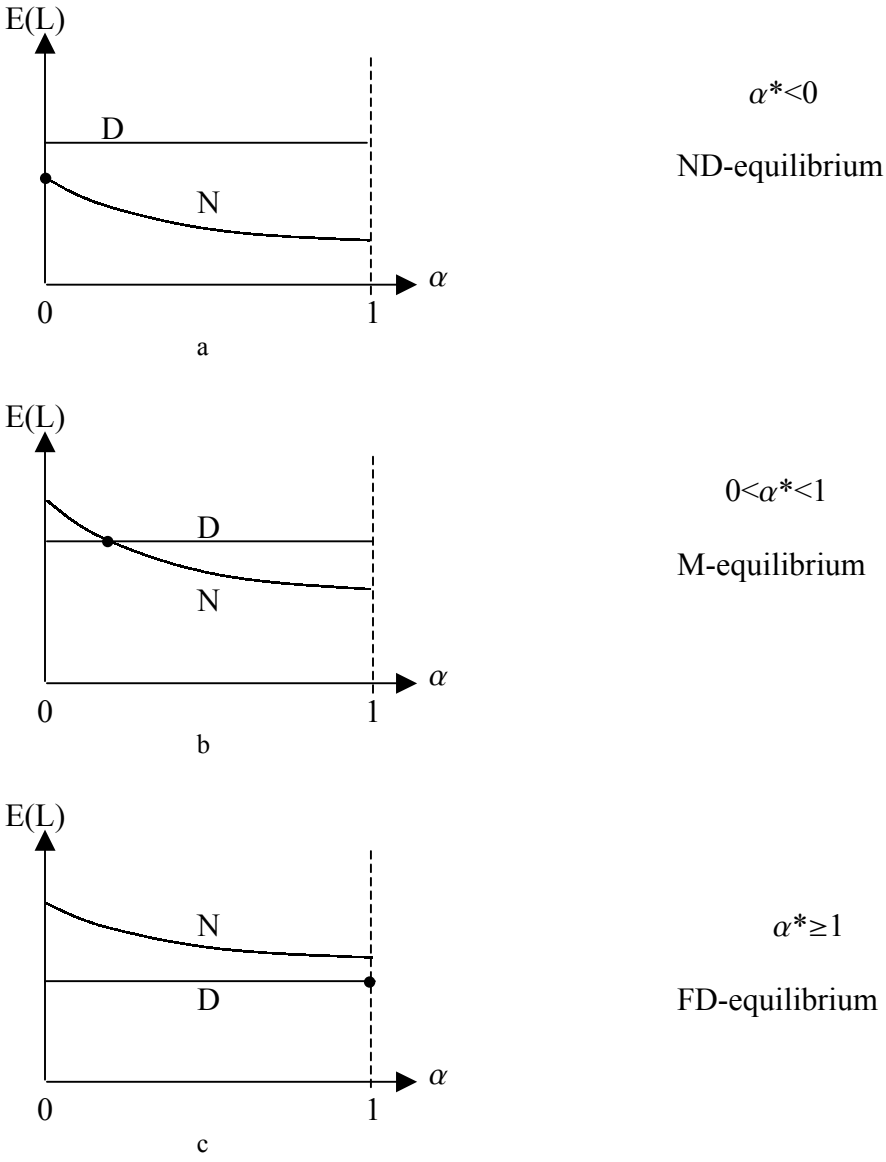

Figure 2. Equilibria of the game in the absence of effort. 'D' stands for "disclose", 'N' – for "not disclose". Dots denote the equilibria.

C. Comparative static results.

In this section we discuss the effect of various model parameters on the proportion of white hats choosing full disclosure, α^*.

Proposition 4: Full disclosure is more likely to occur when the number of black hats in the population increases ($\frac{\partial \alpha^*}{\partial B} > 0$)*, bugs in software get more transparent* ($\frac{\partial \alpha^*}{\partial r} > 0$)*, users get more patient* ($\frac{\partial \alpha^*}{\partial \rho} < 0$)*, and more details of the bug become known* ($\frac{\partial \alpha^*}{\partial \varepsilon} < 0$)*. The effect of the number of white hats on disclosure is ambiguous.*

The intuition behind each result in this proposition goes as follows:

Since black hats never choose full disclosure, an increase in the number of black hats in the population increases the chances that more discoveries are made by black hats. This will make the game last for more periods. On the other hand, periods themselves get shorter, plus the number of black hats aware of the bug accumulates faster, increasing the expected loss from individual attacks. Overall, the last two effects dominate the first one, and an increase in the number of black hats in the population leads to a larger equilibrium proportion of white hats who choose to disclose.

When the number of users of a software product increases, the expected number of periods the game will last before full disclosure occurs gets smaller, and so does the length of each period. Keeping the information secret makes less sense if someone else is going to disclose it soon anyway. Thus, playing "not disclose" becomes relatively less attractive. At the same time, a larger number of white hats slows down the growth in the number of black hats aware of the security hole. Thus, fewer covert attacks occur each period, making "not disclose" relatively more attractive. The overall effect of W on α^* is ambiguous.

An increase in the rate of intertemporal preferences, ρ, makes future losses matter less. This may be due to a lack of patience or stronger myopia on white hats' part. Ceteris paribus, this increases the incentive for white hats to refrain from disclosure in order to delay the timing of a mass-scale attack.

Larger values of ε correspond to more information about a bug being disclosed. This makes it easier for black hats to develop an attack tool and thus decreases the delay between the

moment of disclosure and the attack. In this case, playing "not disclose" makes less sense, and white hats place a larger weight on "disclose" in the equilibrium.

An increase in r represents a greater transparency of bugs, which makes them easier to discover, thus increasing the chances of new independent discoveries of the same bug. As a result, the period length gets shorter. The fact that full disclosure by someone else may be very near in the future reduces the benefit of playing "disclose". Hence lower product quality increases the likelihood of disclosure.

Note that our results so far indicate that the choice of strategy does not depend on the size of the economic loss imposed on the victim of an attack, C. It is easy to see from comparing (4) and (5) that any change in the value of C causes proportional changes in both $E(L_{WN})$ and $E(L_{WD})$. The effect of C is addressed in the next section.

V. Modification #1: Presence of effort

So far in our analysis, white hats did not attempt to work on a fix by themselves since doing so was assumed to be prohibitively difficult. We now modify our model to allow such a possibility. The set of choices facing a white hat who discovers a bug therefore expands. Now such an agent has to decide whether to disclose the information or not, and in the latter case also how much effort, x, he wants to put into finding a fix for it.[16] Agents that choose to disclose the vulnerability do not work on a fix. Effort is costly and leads to success only with some probability, $p = 1 - e^{-\kappa \cdot x}$, where κ is a characteristic of the software product. Greater values of κ indicate higher probability that a fix will be developed for a given level of effort. Thus, a better familiarity of product users with the source code or a higher skill level on their part increase κ. On the other hand, the more complex the product is, the harder it is for users to understand how it works, which makes κ smaller. Note that $\kappa = 0$ effectively results in the version of the model considered in the previous section, when no one works on a fix. When a fix is found, the agent

[16] Chances to produce a fully functional fix are higher when the developers have access to what is known as source code. Source code can be thought of as software product blueprints, written in a language understood by many in the computer community. Naturally, the software vendor has access to such code while users usually do not. Therefore it is more common for end users to resort to what is known as workarounds. Common shortcomings of a workaround are that it usually requires disabling certain features normally available in the product, and rarely eliminates the security problem completely. In this paper, we do not distinguish between these two solutions and use the term "fix" to denote some form of protection, which may be of some value to everyone.

protects his own system and publishes the fix (but not the vulnerability itself), which allows the rest of the user community patch their systems. We assume that the community applies the fix one period after the person who developed it. After all the systems are patched, the game ends with no further losses to anyone.

The game tree modified to include the choice of effort is presented in Figure 3. Note the game may now end not only when a fix is issued by the vendor but also when a user's attempt to develop a fix is successful.

Lemma 2. The optimal choice of effort by a white hat is the same across periods. All white hats playing the game choose the same effort level.

We find this result straightforward and therefore provide only an intuitive proof. A white hat discovering a bug does not know how many prior discoveries of the same bug were made. Still, the game facing him will always be the one in Figure 3. The fact that *(B+W)* is large implies that any prior discoveries change the composition of the pool and the odds of each outcome only marginally. The homogeneity of the white hats pool and the fact that the probability of not finding a patch within a given period, $q = 1 - p = e^{-\kappa \cdot x}$, is exponential in effort imply that the marginal benefit of effort stays the same no matter how long the game has been played. Therefore, every agent playing the game solves the same optimization problem and makes the same decision every period.[17]

Introducing effort into the game does not affect $E(L_{WD})$. $E(L_{WN})$, on the other hand, changes to include an extra effort term and account for the possibility that the game ends without a massive loss. Obtaining a closed form expression for $E(L_{WN})$ for this version of the model appeared impossible. However, we were able to obtain the expressions that bind it from above and from below, respectively, $\underline{E(L_{WN})} \le E(L_{WN}) \le \overline{E(L_{WN})}$.

$$\underline{E(L_{WN})} = \varepsilon\delta q\beta C\gamma_D\left[1+\sum_{j=1}^{\infty}\delta^j q^j\prod_{i=1}^{j}(\beta a^i + q^{i-1}\gamma_N)\right] + \delta q\beta C\left[\frac{1}{N}+\sum_{j=1}^{\infty}\delta^j q^j(1-a^{j+1})\prod_{i=1}^{j}(\beta a^i + q^{i-1}\gamma_N)\right] + \\ + x\left[1+\delta q(\beta a+\gamma_N)+\sum_{j=2}^{\infty}\delta^j q^j\prod_{i=2}^{j}(\beta a^i + q^{i-2}\gamma_N)\right] \qquad (10a)$$

[17] Clearly, the assumptions about the form of the probability function and the population size are necessary for this result to hold.

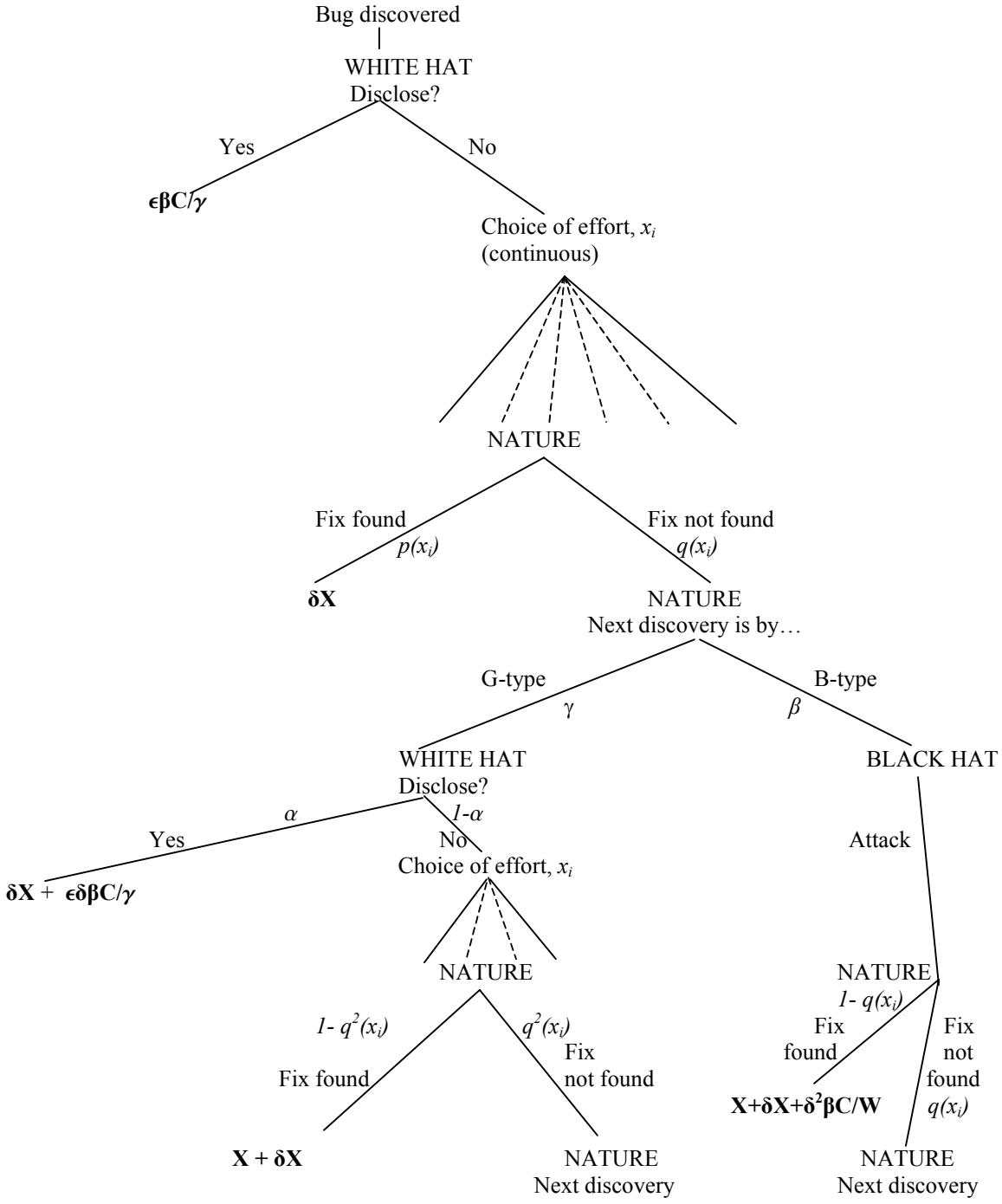

Figure 3. The game in the presence of effort.
The player that makes a decision at each node is shown in capital letters. NATURE stands for probabilistic events. Bold characters denote the payoffs of the initial discoverer at terminal nodes. Italicized characters next to a particular path indicate the probability that the game follows that path.

$$\overline{E(L_{WN})} = \frac{\delta q \beta C}{N}\left[1+\sum_{j=1}^{\infty}\left[\delta^j q^j \prod_{i=1}^{j}(\beta a^i + q^{i-1}\gamma_N)\right] + (\beta a+\gamma_N)\sum_{j=1}^{\infty}\left[\delta^j q^j \prod_{i=1}^{j}(\beta a^i + q^{i-1}\gamma_N)\cdot\sum_{i=1}^{j}\frac{a^i}{\beta a^i + q^{i-1}\gamma_N}\right]\right]$$
$$+\varepsilon\delta q\beta C\gamma_D\left[1+\sum_{j=1}^{\infty}\delta^j q^j \prod_{i=1}^{j}(\beta a^i + q^{i-1}\gamma_N)\right] + x\left[1+\delta q(\beta a+\gamma_N)+\sum_{j=2}^{\infty}\delta^j q^j \prod_{i=2}^{j}(\beta a^i + q^{i-2}\gamma_N)\right] \qquad (10b)$$

The derivation of (10a) and (10b) is omitted for brevity and available upon request.

This version of the model allows us to analyze the effect of two more model parameters, κ and *C*, on equilibrium outcomes. Due to the unavailability of a closed form analytical solution, in this section we resort to numerical simulations.

Proposition 5. An increase in κ *induces white hats to put more effort into finding a fix* ($\frac{\partial x^*}{\partial \kappa} \geq 0$) *and decreases the expected loss from playing "not disclose"* ($\frac{\partial E(L_{WN})}{\partial \kappa} < 0$). *This effect is stronger when the equilibrium proportion of white hats choosing "disclose" is smaller* ($\frac{\partial^2 E(L_{WN})}{\partial \kappa \partial \alpha} > 0$).

This result implies that the opportunity to work on a fix reduces the occurrences of public disclosure. More specifically:

- if an ND-equilibrium existed in the κ = 0 case, it will still exist in the κ > 0 case, but the expected equilibrium loss will get smaller;
- as κ gets larger, an M-equilibrium, if existent, will occur at smaller values of α and eventually be replaced with an ND-equilibrium;
- an FD-equilibrium that may have existed in the κ = 0 case will disappear if κ is sufficiently large.

The intuition behind this result is straightforward. Working on a fix is an additional option which did not exist in the κ = 0 case. A greater number of options available to non-disclosing agents cannot make them worse off. Hence $E(L_{WN})$ can either decrease or stay the same. The effect of κ on $E(L_{WN})$ gets smaller as α gets larger because beliefs that public disclosure will occur soon

anyway reduces incentives to work on a fix. Note the earlier result regarding the uniqueness of an equilibrium no longer holds for this version of the game.

Figure 4 below confirms the result stated in Proposition 5.

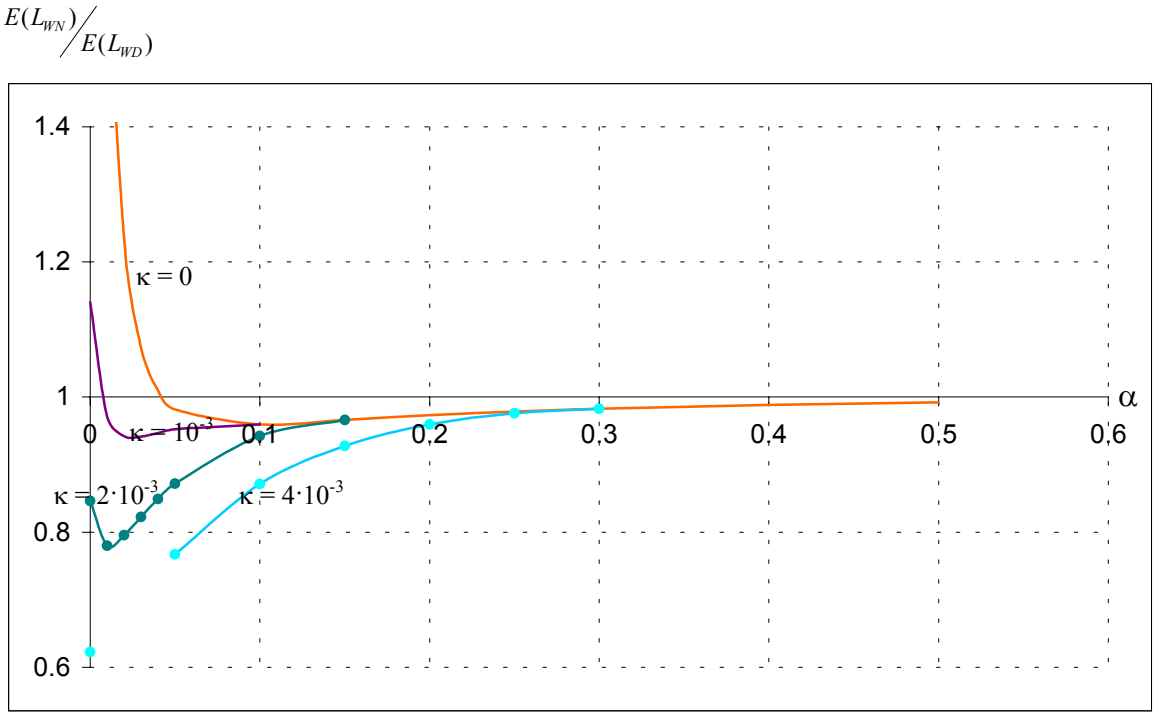

Figure 4. The effect of κ on the shape of the $E(L_{WN})$ curve.

The diagram is based on the results of a numerical simulation for $B+W = 2000$, $C = 2000$, $\delta = 0.99$.
The four curves correspond to $\kappa = 0$ (hence no effort), $\kappa = 10^{-3}$, $\kappa = 2\cdot10^{-3}$, and $\kappa = 4\cdot10^{-3}$.
In the cases when effort is feasible, the effort level is chosen endogenously to minimize $E(L_{WN})$.

The effect of an increase in the size of the loss caused by each attack, C, is qualitatively similar to that of an increase in κ since an increase in C also increases the expected benefit from working on a fix relative to the cost of doing so. As a result, larger C will lead to a greater equilibrium effort exerted by white hats and a lower equilibrium proportion of white hats that choose to disclose.

VI. Modification #2 – Finite number of periods

In this section, we consider the version of the model in which the game is played for a finite number of periods. This may represent a situation when vendors are committed to releasing the

fix no later than *T* periods after the moment when they first received the information from a discoverer.[18] As a result, the game ends after public disclosure or after *T* periods, whichever comes first.

Proposition 5. A decrease in the number of periods, T, decreases the expected loss from playing "not disclose" ($\frac{\partial E(L_{WN})}{\partial T} > 0$). *This effect is stronger when the equilibrium proportion of white hats choosing full disclosure is smaller* ($\frac{\partial^2 E(L_{WN})}{\partial T \partial \alpha} > 0$).

To see the intuition behind the last result, note that the probability that full disclosure will occur at some point in the game equals $1 - \left(1 - \frac{\alpha W}{(B+W)}\right)^T$. When $T = \infty$, the game always ends as a result of full disclosure. When $T = 30$ and $\alpha W = 0.5(B+W)$, the game ends with public disclosure and a mass-scale attack in 99.99999% cases. If, however, $T = 30$ and $\alpha W = 0.01(B+W)$, then in 26% of cases the game would end with a massive attack and in the remaining 74% cases - with no such attack but because vendors release a fix. Clearly, in the latter case a decrease in *T* matters more and incentives to disclose are reduced. To summarize, if vendors are more "conscientious" (or are forced to be that way), we expect to see less public disclosure.[19]

Figure 5 below illustrates the effect of the game length on the relative size of $E(L_{WD})$ and $E(L_{WN})$ and the weight put on each strategy in a mixed equilibrium. As in previous section, this modification makes multiple equilibria possible.

[18] Such a commitment may arise if vendors are facing a credible threat that the bug will be disclosed publicly after a certain 'grace period'.

[19] Coincidentally, this aligns with one of the arguments commonly used to justify full disclosure. As Raikow (2000) put it, "…the threat of widespread disclosure is the only thing that gets most security bugs fixed. I've never found a vendor that wouldn't deny, ignore or mischaracterize a security problem if it could. Bug fixes have no profit margin – the threat of a PR disaster is all that keeps vendors honest."

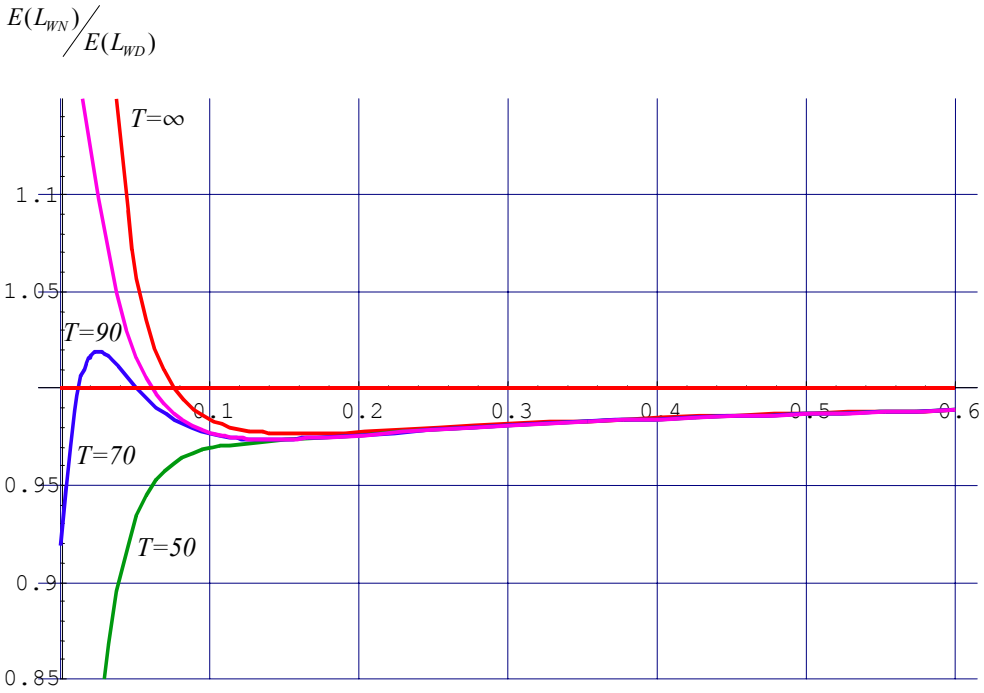

Figure 5. The effect of the number of periods, T, on the shape of the $E(L_{WN})$ curve.
The diagram is based on the results of a numerical simulation for $B+W = 2000$, $\delta = 0.99$.

VII. Modification #3 –Limited information sharing

In this section we add another option regarding the form of disclosure. In addition to the previous two strategies, "disclose" (to the world) and "not disclose", white hats now have the opportunity to share information about discovered vulnerabilities with a closed coalition of agents.[20]

The type of a coalition we consider invites inputs from everyone. The two types of agents are represented within the coalition in the same proportion as in the entire population.[21] Once coalition members get access to information about a bug, each of them acts in his best interest. Black hat members use that information to perform attacks. White hats who belong to the

[20] Limited sharing of information with trusted sources has been encouraged by the government for some time. As one of the government documents on information security policy puts it, "The quickest and most effective way to achieve a much higher level of protection from cyber threats is a strategy of cooperation and information sharing based on partnerships among the infrastructure owners and operators and appropriate government agencies" (Critical Foundations, 1997, p.xi). The most recent government document on the issues, *The National Strategy to Secure Cyberspace* (2003), devotes a section to the importance of information sharing about cyberattacks, threats, and vulnerabilities. Similar calls also come from industry leaders (Johnston, 2000).

[21] We could not rule out the existence of black hats within the coalition. Several studies show that a substantial share of security threats come from people who take advantage of security sensitive information they were trusted with. See Ohlson (1999), Glader (2001), Schlesinger (2001).

coalition may work on a fix along with the discoverer of the vulnerability. Everyone works on a fix separately, independently and simultaneously. Once someone's effort leads to success, the fix is shared with the rest of the community (not just the coalition members).[22] None of the coalition members discloses the information about the vulnerability he has acquired through membership.

A simplified game tree is provided in Figure 6.

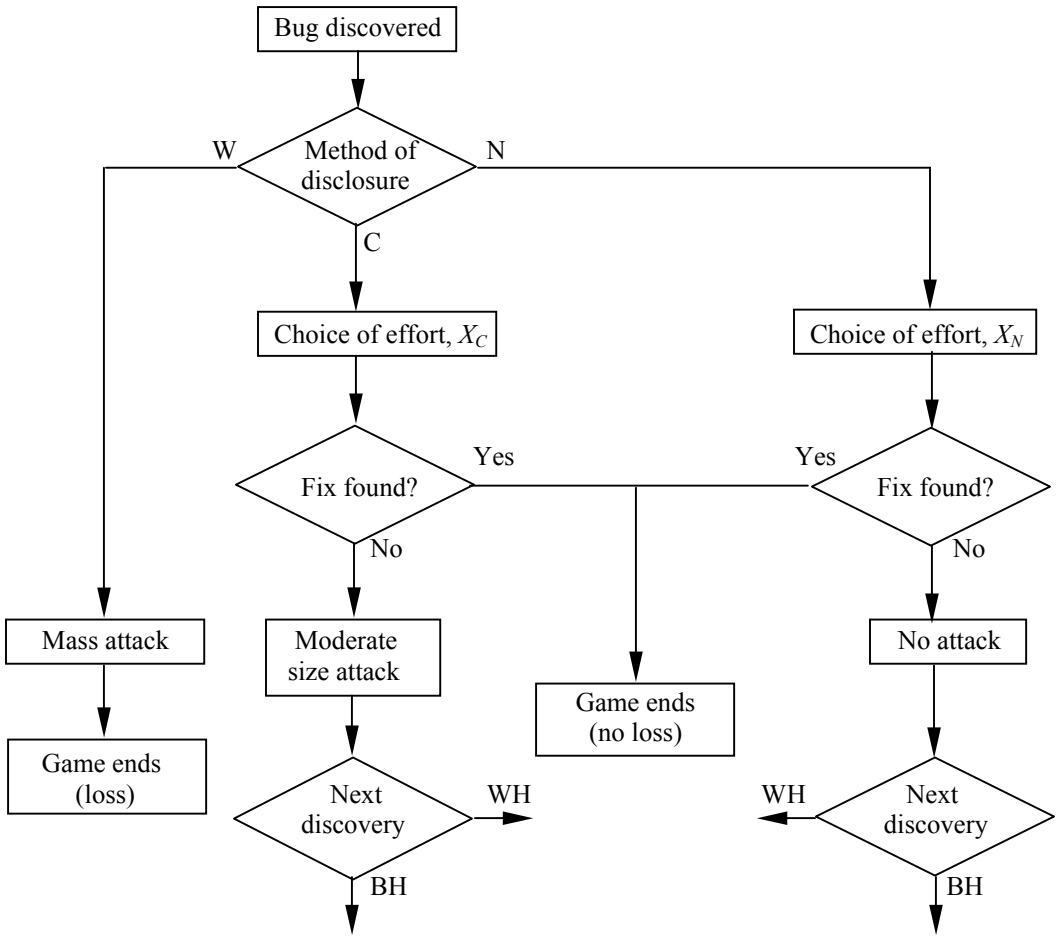

Figure 6. Schematic tree of the game in the presence of a coalition. W, C, and N stand for disclosure to the World, disclosure to the Coalition, and No disclosure, respectively. WH and BH indicate whether the next discoverer is a white hat or a black hat.

[22] One example of an entity operating under such or similar rules is CERT/CC, a federally funded agency founded in 1988 whose mission is to "serve as a major reporting center for Internet security problems" (www.cert.org). CERT/CC plays an important role in compiling statistics, issuing advisories, analyzing trends in the types of attacks information systems are subjected to, and disseminating security-related information to the community.

Another form of a coalition is represented by Information Sharing and Analysis Centers, or ISACs. In the case of ISACs, information is shared and disseminated only among coalition members. See Gal-Or and Ghose (2003) for the analysis of issues related to such a mechanism.

To better demonstrate the effect the presence of a coalition has, we compare disclosure to a coalition to no disclosure. On one hand, disclosure to a coalition makes the number of black hats informed about the bug accumulate faster than in the case of no disclosure, which increases the intensity of covert attacks compared to no disclosure at all. Note that such limited disclosure does not necessarily accelerate the release of a fix by the vendor. From this standpoint, disclosure to a coalition is dominated by no disclosure. On the other hand, disclosure to a coalition leads to a larger number of white hats informed about a bug and working on a fix at any point in time. Thus, the probability that a fix will be found increases, which benefits the original discoverer along with the rest of the community. This increases the incentive to disclose information to a coalition. Note, however, that the latter effect is necessary to allow the "disclose to a coalition" strategy to compete for the best. In the cases when the cost of working on a fix is prohibitively high or if white hat discoverers do not believe (for any reason) that coalition members will work on a fix, only the first of the two aforementioned effects is present. As a result, disclosure to the coalition is suboptimal since it is dominated by at least one of the two original strategies, and no information is shared with the coalition.[23]

The result regarding suboptimality of sharing information with or within a group is not entirely new. Existing literature on cooperative R&D also points out the reluctance of individual members to contribute to such a databank and the incentives to free ride. A thorough analysis of various possible incentive schemes and their applicability to information security issues would require a separate research paper. In this section, we provided an intuitive analysis of just one such scheme. Still, our finding in this section may serve as a reminder that any attempt to induce limited information sharing requires a cleverly and carefully designed mechanism of monitoring and/or rewarding contributions and inducing truth-telling. It may also provide a possible explanation for the limited success information-sharing schemes had so far.

[23] This intuition is supported by the results of numerical simulations, which are not presented here but will be included in the final version of the paper.

VIII. Policy implications

We start with the notion that the ultimate goal of government agencies concerned with information security is, or at least should be, to reduce not the occurrences of full public disclosure per se, but rather the overall cost imposed on the community by malicious attacks. The expected social loss equals the sum of the expected losses of individual white hats. Since in our model all white hats were assumed identical,

$$E(L_{soc}) = E(L_{WD}) \cdot W \tag{11a}$$

if the game results in an FD- or M-equilibrium, and

$$E(L_{soc}) = E(L_{WN})\big|_{\alpha=0} \cdot W \tag{11b}$$

if it results in an ND-equilibrium. Thus, the minimization of the expected loss by individual white hats effectively results in the maximization of social welfare.[24] Whatever is best for the individual is best for the community. The conflict between private and public interests common for the public goods literature does not exist in our model.

This implies that whenever loss-minimizing agents choose full disclosure, they do so to reduce their loss and therefore the loss of the entire white hat population. In that sense, it is desirable to have a venue for public disclosure. To see this, consider cases when the game results in a FD- or M- equilibrium (panels a, b of Figure 2). Elimination of the opportunity to disclose will result in the individual loss being equal to $E(L_{WN})\big|_{\alpha=0} > E(L_{W,eq}) = E(L_{WD})$. Therefore our model suggests that prohibiting full disclosure and prosecuting individuals who disclose security-related information in a public forum[25] ("shooting the messenger" in the terminology of Rauch (1999)) is a bad idea. In the absence of a venue for full disclosure, attacks may last longer and draw less public attention but cause more damage on the aggregate.

The following discussion of policy alternatives is based on the results stated in propositions 4-6.

---

[24] Naturally, any recommendations provided in this section are contingent on our conjecture regarding the form of the social welfare function. We can think of many reasons why the form of that function the may be different. Whenever that happens, the socially optimal outcome will likely be different from the equilibrium, which may serve as a justification for government regulation of information security practices.

[25] There is substantial evidence of such attempts. The Digital Millenium Copyright Act (Pavlicek, 2002; Poulsen, 2003) is one example. Also see Lee (2001a, 2001b) for a story about a programmer prosecuted for demonstration of flaws in e-book security.

A reduction in the proportion of black hats in the computer community, *B*, reduces the expected social loss. It may also shift the equilibrium of the game towards smaller α, thus reducing the chances that full disclosure will occur. A decrease in *B* may be achieved by increasing the probability of intrusion detection, the punishment for performing attacks, or by using a combination of the two approaches.

While the idea itself is very straightforward, implementing any policies in that regard faces several obstacles. One is the difficulty of identifying the source of an attack. Another is that a large portion of attacks on U.S. computer systems originates abroad. Therefore, such a policy would require a code of international laws on information security. While some progress in those directions is being made, it still remains more of a long-term policy goal.

A decrease in the size of the loss from each individual attack, *C*, will trivially lead to a reduction in the overall social loss. The most straightforward way to decrease *C* is to better protect individual systems.[26] However, one has to keep in mind that a reduction in *C* reduces incentives for white hats to work on a fix, which, as discussed in Section V, may lead to an increase in the equilibrium proportion of white hats who choose full disclosure, α^*. Therefore, unless looking for a fix is prohibitively costly for users, a reduction in *C*, ceteris paribus, will be accompanied by more frequent public disclosure.

The rate of repeat independent discoveries of the same bug, *r*, is greater when bugs in a software product are easier to come across. Our analysis suggests that *r* being larger makes white hats more likely to choose public disclosure. This result underscores the extreme importance of software quality in providing information security. Lack of effort put by vendors into debugging a product increases the number and transparency of bugs contained in it. This makes discoverers more likely to disclose vulnerabilities publicly, thus further aggravating the state of information security. This makes us believe that software quality improvement should be one of the priorities while attempting to improve the state of information security.

How to achieve this, however, remains a big question. Software producers do what they believe is in their best interest, reacting to strong demand for new, yet not necessarily properly

[26] Better patching is also expected to reduce the gain black hats get from performing attacks and thus further improve security through a reduction in the proportion of black hats, *B*. However, decisions on whether and when to patch a system are made by system administrators, whose decisions our model treats as exogenous.

tested, products. It is a common notion that in the software market, speed is often valued over quality. Further research is needed before the causes of such situation are fully understood.[27]

Parameter *r* can also be affected by the increasing complexity of software products. The sign of this effect on *r* is unclear, however. Will more complex software make discoveries less likely because fewer and fewer people will be able to understand how this software works, or will the number of holes in the finished product get larger due to a more difficult task of debugging the product?[28] While not a policy issue in itself, this factor needs to be taken into account while devising any policies since, as our model suggests, it may affect disclosure practices.

Parameter κ accounts for the ease with which a fix for a security hole can be found by users. One factor affecting κ is once again the complexity of software. Since, as we just pointed out, software is getting increasingly complex, chances to find a fix get smaller. We show that, ceteris paribus, this increases the incentives to disclose.

Another factor that affects κ is the extent to which users are familiar with the details of a particular piece of software. Normally, users do not have access to the source code of the product they are using, which makes developing a fix more difficult for them. One exception is the "open source" software, which lets every user see the source code and modify it as needed. One of the issues widely discussed nowadays is whether open source software is more secure than closed source one. Our findings suggest that providing users with the source code for software they are using would increase their ability to develop fixes, reduce the loss to social welfare, and may also substantially reduce or completely eliminate public disclosure occurrences. [29]

However, we understand that such a scenario is purely hypothetical, not only because it is unfeasible but also because it would undermine the existing system of intellectual property rights and reduce the incentives to develop new products. Taking into account the fact that advances in information technology have accounted for a large part of overall growth in the last few decades, such a move would be very undesirable.

---

[27] For some useful insights on this issue, see Arora et al. (2003).

[28] According to a "rule of thumb" used in software economics, with an increase in the size of a software product (often measured in lines of code) the time that need to be spent on debugging the product increases more than proportionally.

[29] It is interesting to note here that public disclosure of security flaws in open source products is much less common and causes a much more negative reaction even among the proponents of full disclosure. As an example, see Lasser (2002) for a discussion of an attempt to publish a bug discovered in Apache server, an open source product. This is consistent with the notion following from our research that disclosing vulnerabilities in open source products without offering a fix is simply irrational.

We are also able to show that, as the amount of information that is disclosed publicly increases, attacks follow sooner, thus increasing the incentives to disclose. However, we are not ready to recommend limited scope disclosure based on this fact alone. Any public disclosure is justified only if its threat forces vendors to issue fixes or better improve the quality of their products. Therefore there is a limit to how small the amount of disclosed information can be for it to achieve this goal. We are unable to elaborate further on this issue since in our model the choice of the scope of disclosure is not endogenous.

Finally, our analysis shows that the choice of a disclosure strategy strongly depends on how promptly vendors issue fixes. The effect of this factor on social welfare is twofold. First, it makes the window of exposure shorter. Second, the expectations that the fix will be issued soon reduces incentives for public disclosure. As a result, a decrease in the time it takes the vendor to issue a fix causes a more than proportional decrease in the social loss.

Several attempts have been made lately to develop a uniform procedure governing the disclosure practices.[30] Each of the proposed procedures involves a choice of the length of a 'grace period'. According to our findings, a shorter grace period is better since it reduces the incentives for full public disclosure while more liberal policies in that regard will lead to more public disclosure.

IX. Conclusion

In this paper, we use game-theoretic approach to analyze existing practices related to disclosure of software vulnerabilities. We are able to show that the existing practice of full public disclosure of security-sensitive information is not necessarily malicious or irresponsible. Instead, it may be an equilibrium of a game played by benevolent agents minimizing their expected loss.

We find that under certain parameters public disclosure of vulnerabilities is desirable from the social welfare standpoint. The presence of an opportunity to disclose allows individual users to reduce their expected loss from attacks and therefore improves social welfare. We discuss several alternative policies aimed at improving the overall state of computer security. Of all the

[30] Examples include the "Responsible Disclosure Forum" (available at http://www.ntbugtraq.com/RDForum.asp) and a more recent "Security Vulnerability Reporting and Response Process" proposed by the Organization for Internet Safety (http://www.oisafety.org/process.html).

factors we consider, two appear to be the most important. According to our results, a reduction in the quality of software products causes more public disclosure and therefore more than proportional reduction in social welfare due to attacks. Another factor that has a similarly strong negative effect is vendors' delay in releasing fixes for known vulnerabilities. Therefore our results suggest that designing an incentive system that would induce vendors to release fixes sooner and improve the quality of their products should be among the priorities for any policymaking agency concerned with information security.

We find that coalitions inviting limited information sharing can have a positive effect, but only under a restrictive set of conditions. This may serve as a possible explanation for the limited success such entities had so far.

Our model also contributes to the discussion of the virtues and flaws of open source and closed source software. According to our model, among the two products with the same number of security flaws, an open-source product leads to a smaller number of attacks (and hence smaller aggregate losses) than a closed-source product due to higher chances that a fix is developed by the discoverer himself. However, making any policy recommendations based on this fact alone seems premature.

As always, there are several ways in which our model could be improved. The biggest shortcoming is that it models the behavior of such important parties as software vendors and system administrators in a rather schematic way. This is due to the fact that our main focus is on the decisions made by users who discover security flaws. Up to now, disclosure decisions did not receive much attention in economic research, and our paper makes a contribution by providing several important insights in that regard. It complements the existing body of research in the area of information security and makes a step towards a comprehensive model of information security that would endogenize the choices of all parties involved.

References

Arora, A., J. P. Caulkins, and R. Telang, 2003, "Provision of Software Quality in the Presence of Patching Technology," Working paper, Carnegie Mellon University.

Arora, A., R. Telang, and H. Xu, 2004a, "Optimal Policy for Software Vulnerability Disclosure", Working paper, Carnegie Mellon University.

Arora, A., R. Krishnan, A. Nandkumar, R. Telang, and Y. Yang, 2004b, "Impact of Vulnerability Disclosure and Patch Availability – an Empirical Analysis," *Third Workshop on the Economics of Information Security*.

Bank, D., 2004, "MyDoom Worm Renews Debate on Cyber-Ethics," *The Wall Street Journal*, Nov 11, 2004.
Baumol, W., 1992, "Horizontal Collusion and Innovation", *Economic Journal*, 102, pp.129-137.
Beattie, S., S. Arnold, C. Cowan, P. Wagle, C. Wright, and A. Shostack, 2002, "Timing the Application of Security Patches for Optimal Uptime," *Proceedings of LISA 2002: Sixteenth Systems Administration Conference*.
Bhattacharya, S., J. Glazer, and D. Sappington, 1990, "Sharing Productive Knowledge in Internally Financed R&D Contests", *Journal of Industrial Economics*, 39, pp.187-208.
Bridis, T., 2001,"Tech Alliance to Share Data about Hackers", *The Wall Street Journal*, Jan 16, 2001.
Bridis, T., and G. Simpson, "CERT Plans to Sell Early Warnings on Web Threats", *The Wall Street Journal*, April 19, 2001.
"Critical Foundations: Protecting America's infrastructures", 1997, The Report of the President's Commission on Critical Infrastructures Protection.
d'Aspremont, C. and A. Jacquemin, 1988, "Cooperative and Noncooperative R&D in Duopoly with Spillovers", *American Economic Review*, 78, pp.1133-1137.
Gal-Or, E., and A. Ghose, 2004, "The Economic Incentives for Sharing Security Information," SSRN electronic paper.
Glader, P., 2001, "Mission: Protect Computers from Insiders," *The Wall Street Journal*, Dec 13, 2001.
Gordon, A.L., M. Loeb and W. Lucyshyn, 2003, "Sharing information on computer system security: an economic analysis," *Journal of Accounting and Public Policy*, vol. 22(6), pp. 461-485.
Holmstrom, B., 1999, "Managerial Incentive Problems: A Dynamic Perspective," *Review of Economic Studies*, 66, pp.169-182.
Hopper, D.I, and P. Thomas, 2000, "FBI smokes out "Coolio" in computer attacks probe", *CNN News*, available at http://www.cnn.com/2000/TECH/computing/03/02/dos.coolio/
Johnston, M., 2000, "Info Sharing Crucial to Security, Says EDS CEO", *IDG News Service\Washington Bureau*, Oct 16, 2000.
Kamien, M.I., E. Muller, and I. Zang, 1992, "Research Joint Ventures and R&D Cartels", *American Economic Review*, 82, pp. 1293-1307.
Kannan, K., and R. Telang, 2004, "An Economic Analysis of Market for Software Vulnerabilities," *Third Workshop on the Economics of Information Security.*
Katz, M.L. and J.A. Ordover, 1990, "R&D Cooperation and Competition", *Brookings Papers on Microeconomics*, pp. 137-203.
Lasser, J., 2002, "Irresponsible Disclosure", *Security Focus*, June 26, 2002.
Lee, J., 2001a, "Man Denies Digital Piracy in First Case under '98 Act", *The New York Times*, Aug 31, 2001.
Lee, J., 2001b, "In Digital Copyright Case, Programmer Can Go Home", *The New York Times*, Dec 14, 2001.
Lerner, D., 2000, "US Groups Target Hackers", *Financial Times*, May 29, 2000.
Lerner, J., and J. Tirole, 2000, "The Simple Economics of Open Source", *NBER Working Paper 7600*.
National Research Council, System Security Study Committee, CSTB, *Computers at Risk*, National Academy Press, 1991. Chapter 6, "Why the Security Market Has Not Worked Well", pp.143-178. Also available at http://www.nap.edu/books/0309043883/html/index.html
National Strategy to Secure Cyberspace, 2003, *Office of the President of the United States*.
Ohlson, K., 1999, "Disgruntled Employees: The Newest Kind of Hacker", *ComputerWorld* Mar 11, 1999, also available at http://www.computerworld.com/news/1999/story/0,11280,27470,00.html
Ozment, A., 2004, "Bug Auctions: Vulnerability Markets Reconsidered," *Third Workshop on the Economics of Information Security*.
Pavlicek, R., 2002, "DMCA Horror Show", *InfoWorld*, Oct 19, 2002.
Petit, M.L. and B. Tolvinski, 1996, "Technology Sharing Cartels and Industrial Structure", *International Journal of Industrial Organization* **15**, pp.77-101.
Pond, W., 2000, "Do Security Holes Demand Full Disclosure?", *eWeek*, Aug 16, 2000.

Poulsen, K., 2003, "'Super-DMCA' Fears Suppress Security Research," *SecurityFocus*, Apr 14, 2003, available at: http://www.securityfocus.com/news/3912
Power, R. "2001 CSI/FBI Computer Crime and Security Survey." *Computer Security Journal* **XVII**, pp. 29-51.
Raikow, D. "Bug Fixes Have No Profit Margin", *eWeek,* Oct 19 2000.
Rauch, J., 1999, "Full Disclosure: The Future of Vulnerability Disclosure?", *;login:- The Magazine of USENIX and SAGE*, *Special Issue on Security*, Nov 1999, available at http://www.usenix.org/publications/login/1999-11/features/disclosure.html
Reuters, 2001, "KPMG: Biggest Threat to Data from Insiders", available at http://news.zdnet.co.uk/story/0,,t269-s2085405,00.html
Schlezinger, L., 2002, "Threat from Within", available at http://www.zdnet.co.uk/help/tips/story/0,2802,e7109952,00.html
Schneier, B., 2000, "Full Disclosure and the Window of Exposure," *CRYPTO-GRAM*, Sep 15, 2000, Counterpane Internet Security, Inc., available at http://www.schneier.com/crypto-gram-0009.html#1
U.S. Department of Justice, 2003, "Russian Man Sentenced for Hacking into Computers in the United States", U.S. Department of Justice Press Release, July 25, 2003, available at http://www.usdoj.gov/criminal/cybercrime/ivanovSent.htm
Wattal, S. and R.Telang, 2004, "Effect of Vulnerability Disclosures on Market Value of Software Vendors – An Event Study Analysis," Working Paper, September 2004, Carnegie Mellon University.

APPENDIX

A.1 - Derivation of (5).

When period 1 discoverer plays "not disclose", he incurs no losses in period 1. In the periods that follow, he can become a victim of an attack in two cases. One is an attack performed on a mass scale following full public disclosure by one of the future white hat discoverers. In the case such an event occurs, the expected loss to the agent in question is $\frac{BC}{W}$. The probability that full public disclosure will occur in a certain period is $\frac{\alpha W}{(B+W)} = \gamma_D$ for period 2, $\frac{(Ba+(1-\alpha)W)\alpha W}{(B+W)^2} = (\beta a + \gamma_N)\gamma_D$ for period 3, and so on, where $a = 1 - \frac{1}{W}$ is the probability that the agent in question is not hit by one of the attacks in an earlier period.

Combining the terms together and adding discounting yields the total discounted expected loss resulting from mass-scale attacks following disclosure

$$\varepsilon\delta\alpha\beta C\left[1+\sum_{j=1}^{T}\delta^{j}\prod_{i=1}^{j}(\beta a^{i}+\gamma_N)\right] \qquad \text{(A.1)}$$

where T is the maximum number of periods the game may last.

A G-type agent can also get hit by an individual attack performed by a B-type agent who discovered the bug independently, in which case the expected loss to the agent in question is $\frac{C}{W} = (1-a)C$.. The chance that a black hat will make a discovery in period 2 is $\frac{B}{B+G} = \beta$,

in period 3 it is $\beta a\beta(1-a^2)+\beta a\gamma_N(1-a)+\gamma_N\beta(1-a)=\beta(1-a^2)(\beta a+\gamma_N)$, and so on. Note that the number of such attacks in each period gradually increases due to a potential increase in the number of black hats who learn about the bug through individual discoveries.

Proceeding in the same fashion for an arbitrary number of periods, T, and adding discounting yields

$$\delta\beta C\left[(1-a)+\sum_{j=1}^{T}\delta^{j}(1-a^{j+1})\prod_{i=1}^{j}(\beta a^{i}+\gamma_N)\right] \tag{A.2}$$

The present value of the expected loss from both types of attacks is therefore

$$E(L_{WN})=\delta\beta C\left[\varepsilon\alpha+\varepsilon\alpha\sum_{j=1}^{T}\delta^{j}\prod_{i=1}^{j}(\beta a^{i}+\gamma_N)+(1-a)+\sum_{j=1}^{T}\delta^{j}(1-a^{j+1})\prod_{i=1}^{j}(\beta a^{i}+\gamma_N)\right] \tag{A.3}$$

When W is large, we can simplify (A.3) by setting $a=1-\frac{1}{W}\to 1$ and $1-a^{i}\to i/W$ so that $E(L_{WN})$ becomes

$$E(L_{WN})=\delta\beta C\left[\varepsilon\alpha\sum_{j=0}^{T}\delta^{j}(\beta+\gamma_N)^{j}+\frac{1}{W}\sum_{j=0}^{T}\delta^{j}(j+1)(\beta+\gamma_N)^{j}\right] \tag{A.4}$$

And when $T=\infty$,

$$E(L_{WN})=\delta\beta C\left[\varepsilon\alpha\sum_{j=0}^{\infty}\delta^{j}(\beta+\gamma_N)^{j}+\frac{1}{W}\sum_{j=0}^{\infty}\delta^{j}(j+1)(\beta+\gamma_N)^{j}\right]=$$

$$\delta\beta C\left[\frac{\varepsilon\alpha}{1-\delta(\beta+\gamma_N)}+\frac{1}{W(1-\delta(\beta+\gamma_N))^{2}}\right]$$

A.2 – Proof of Lemma 1, derivation of (6)

$E(L_{WD})=E(L_{WN})$ implies $\delta\beta C\left[\frac{\varepsilon\alpha}{1-\delta(\beta+\gamma_N)}+\frac{1}{W(1-\delta(\beta+\gamma_N))^{2}}\right]=\varepsilon\beta C$

$$\delta\left[\frac{\varepsilon\alpha}{(1-\delta+\delta\alpha\gamma)}+\frac{1}{W(1-\delta+\delta\alpha\gamma)^{2}}\right]=\varepsilon$$

$$\frac{\delta[(B+W)^{-1}+\varepsilon\alpha\gamma((1-\delta)+\delta\alpha\gamma)]}{\varepsilon((1-\delta)+\delta\alpha\gamma)^{2}}=1$$

$$\frac{\delta}{B+W}=\varepsilon((1-\delta)+\delta\alpha\gamma)^{2}-\delta\varepsilon\alpha\gamma((1-\delta)+\delta\alpha\gamma)=\varepsilon((1-\delta)+\delta\alpha\gamma)(1-\delta)$$

$$\frac{\delta}{\varepsilon(B+W)(1-\delta)}=(1-\delta)+\delta\alpha\gamma$$

$$\delta\alpha\gamma=\frac{\delta}{\varepsilon(B+W)(1-\delta)}-(1-\delta)$$

$$\alpha=\frac{1}{\varepsilon\gamma(B+W)(1-\delta)}-\frac{(1-\delta)}{\delta\gamma}=\frac{1}{\varepsilon W(1-\delta)}-\frac{(1-\delta)(B+W)}{\delta W}$$

A.3. Proof of Proposition 1.

The existence of an FD-equilibrium requires $E(L_{WD})\big|_{\alpha=1} \le E(L_{WN})\big|_{\alpha=1}$, which implies $(B+W)(1-\delta)(1-\delta\beta) \le \frac{\delta}{\varepsilon}$, or $(1-\delta)(W+(1-\delta)B) \le \frac{\delta}{\varepsilon}$.

A.4. Proof of Proposition 2.

The existence of an ND-equilibrium requires $E(L_{WN})\big|_{\alpha=0} \le E(L_{WD})\big|_{\alpha=0}$, which trivially yields $(B+W)(1-\delta)^2 \ge \frac{\delta}{\varepsilon}$.

A.5. Proof of Proposition 3.

The existence of an M-equilibrium requires $E(L_{WN}) = E(L_{WD})$ for some 0<α<1. By Lemma 1, $E(L_{WN}) = E(L_{WD})$ has only one root, $\alpha^* = \frac{1}{\varepsilon W(1-\delta)} - \frac{(1-\delta)(B+W)}{\delta W}$. The 0<$\alpha$<1 condition is satisfied if and only if $0 < \delta - \varepsilon(1-\delta)^2(B+W) < \delta\varepsilon W(1-\delta)$.

Note an M-equilibrium is stable only if at the equilibrium $\frac{dE(L_{WN})}{d\alpha} < 0$. To see this is so, note that any deviation from the equilibrium mixed strategy violates $E(L_{WN}) = E(L_{WD})$. When $\frac{dE(L_{WN})}{d\alpha} < 0$ is satisfied, deviations in the direction of more disclosure make disclosure less attractive, and vice versa, so that the player is better off going back to the equilibrium strategy. When $\frac{dE(L_{WN})}{d\alpha} > 0$, any deviation from the equilibrium mixed strategy changes the expected payoffs in a way that induces further departures from the equilibrium.

$$E(L_{WN}) = \delta\beta C\left[\frac{\varepsilon\alpha}{1-\delta(1-\alpha\gamma)} + \frac{1}{W(1-\delta(1-\alpha\gamma))^2}\right],$$

$$\frac{dE(L_{WN})}{d\alpha} = \delta\beta C\left[\frac{\varepsilon(1-\delta)}{(1-\delta+\alpha\delta\gamma)^2} - \frac{2\delta\gamma}{W(1-\delta+\alpha\delta\gamma)^3}\right] = \delta\beta C\left[\frac{\varepsilon W(1-\delta)(1-\delta+\alpha\delta\gamma) - 2\delta\gamma}{W(1-\delta+\alpha\delta\gamma)^3}\right]$$

The denominator of the expression is positive. Substituting $\alpha^* = \frac{1}{\varepsilon W(1-\delta)} - \frac{(1-\delta)}{\delta\gamma}$ in the numerator yields $\varepsilon W(1-\delta)^2 + \varepsilon W(1-\delta)\delta\gamma\alpha^* - 2\delta\gamma = \varepsilon W(1-\delta)^2 + \delta\gamma - \varepsilon W(1-\delta)^2 - 2\delta\gamma = -\delta\gamma < 0$

Therefore $\frac{dE(L_{WN})}{d\alpha} < 0$.

A.6. Proof of Corollary 1.

Both $E(L_{WN})$ and $E(L_{WD})$ are continuous in the α-space. By Lemma 1, $E(L_{WN}) = E(L_{WD})$ has only one real root in the α-space. By Proposition 3, $\frac{dE(L_{WN})}{d\alpha} < 0$.

Existence of an M-equilibrium requires 0<α*<1, which in turn implies $E(L_{WD})\big|_{\alpha=0} \le E(L_{WN})\big|_{\alpha=0}$ and $E(L_{WN})\big|_{\alpha=1} \le E(L_{WD})\big|_{\alpha=1}$, so neither ND- nor FD-equilibrium exists.

$\alpha^* \geq 1$ violates (9) and implies $E(L_{WD})\big|_{\alpha=1} \leq E(L_{WN})\big|_{\alpha=1}$ and $E(L_{WD})\big|_{\alpha=o} \leq E(L_{WN})\big|_{\alpha=0}$. Therefore, the only equilibrium is the FD-equilibrium.

$\alpha^* \leq 0$ violates (9) and implies $E(L_{WN})\big|_{\alpha=o} \leq E(L_{WD})\big|_{\alpha=0}$ and $E(L_{WN})\big|_{\alpha=1} \leq E(L_{WD})\big|_{\alpha=1}$. Therefore, the only equilibrium in this case is the ND-equilibrium.

A.7. Proof of Proposition 4.

$\alpha^* = \frac{1}{\varepsilon W(1-\delta)} - \frac{(1-\delta)(B+W)}{\delta W}$, where $\delta = e^{-\frac{\rho}{r(B+W)}}$. Therefore,

$\frac{\partial \alpha^*}{\partial \rho} = \frac{-\delta}{\varepsilon W(1-\delta)^2 r(B+W)} - \frac{1}{\delta W r} < 0$;

$\frac{\partial \alpha^*}{\partial r} = \frac{\rho\delta}{\varepsilon W(1-\delta)^2 r^2 (B+W)} + \frac{\rho}{\delta W r^2} > 0$;

$\frac{\partial \alpha^*}{\partial \varepsilon} = \frac{-1}{\varepsilon^2 W(1-\delta)} < 0$;

$\frac{\partial \alpha^*}{\partial B} = \frac{1}{W}\left(1 - \frac{1}{\delta} + \frac{\rho}{\delta r(B+W)} + \frac{\rho\delta}{\varepsilon(1-\delta)^2 r(B+W)^2}\right) = \frac{\delta - 1 - \ln\delta}{\delta W} + \frac{\rho\delta}{\varepsilon(1-\delta)^2 rW(B+W)^2} > 0$;

$\frac{\partial \alpha^*}{\partial W} = \frac{B(1-\delta) - W\ln\delta}{\delta W^2} - \frac{\delta \ln\delta}{\varepsilon(1-\delta)^2 W(B+W)} - \frac{1}{\varepsilon(1-\delta)W^2}$.

A.8 – Proof of Proposition 5.

By using a monotonic transformation, the expected loss of a non-disclosing white hat can be expressed as $E(L_{WN}) = C \cdot F(q,\cdot) + x$, therefore

$$dE(L_{WN}) = F(\cdot)dC + (1 + C \cdot \frac{\partial F}{\partial q} \cdot \frac{\partial q}{\partial x^*})dx^* + (C \cdot \frac{\partial F}{\partial q} \cdot \frac{\partial q}{\partial \kappa} + (C \cdot \frac{\partial F}{\partial q} \cdot \frac{\partial q}{\partial x^*} + 1) \cdot \frac{\partial x^*}{\partial \kappa})d\kappa ,$$

By the envelope theorem, $\frac{dE(L_{WN})}{dx^*} = 1 + C \cdot \frac{\partial F}{\partial q} \cdot \frac{\partial q}{\partial x^*} = 0$, hence

$$dE(L_{WN}) = F(\cdot)dC + C \cdot \frac{\partial F}{\partial q} \cdot \frac{\partial q}{\partial \kappa} d\kappa .$$

$$\frac{\partial q}{\partial \kappa} = -x^* \cdot e^{-\kappa x^*} \leq 0 .$$

From the above envelope theorem conjecture, and given the fact that $\frac{\partial q}{\partial x^*} = -\kappa \cdot e^{-\kappa x^*} \leq 0$, $\frac{\partial F}{\partial q}$ has to be positive. This implies $\frac{dE(L_{WN})}{d\kappa} < 0$.