An Economic Analysis of Software Market with Risk-Sharing Contract

Byung Cho Kim
Tepper School of Business
Carnegie Mellon University
Pittsburgh, PA 15213
bckim@andrew.cmu.edu

Pei-Yu Chen
Tepper School of Business
Carnegie Mellon University
Pittsburgh, PA 15213
pychen@andrew.cmu.edu

Tridas Mukhopadhyay
Tepper School of Business
Carnegie Mellon University
Pittsburgh, PA 15213
tridas@andrew.cmu.edu

Author addresses:

Byung Cho Kim (contact author)
Tepper School of Business,
Carnegie Mellon University,
5000 Forbes Ave., Pittsburgh, PA 15213, USA
Phone: 412-268-3681, Email: bckim@andrew.cmu.edu

Pei-Yu Chen
Tepper School of Business,
Carnegie Mellon University,
5000 Forbes Ave., Pittsburgh, PA 15213, USA
Phone: 412-268-1291, Email: pychen@andrew.cmu.edu

Tridas Mukhopadhyay
Tepper School of Business,
Carnegie Mellon University,
5000 Forbes Ave., Pittsburgh, PA 15213, USA
Phone: 412-268-2307, Email: tridas@andrew.cmu.edu

An Economic Analysis of Software Market with Risk-Sharing Contract

Byung Cho Kim, Pei-Yu Chen, and Tridas Mukhopadhyay

ABSTRACT: Poor quality of software has been blamed for poor security of our computer networks in the sense that major viruses and worms exploit the vulnerabilities of such software. However, software vendors have no incentive to improve the quality of their products since they are not directly liable for any loss due to poor quality. Software liability has been intensely discussed among computer scientists and jurists for decades as a possible solution for software quality improvement. This paper proposes a risk-sharing mechanism between software vendors and customers as a market-driven way to impose software liability. We consider two dimensions of software quality, which are functionality and security quality. We present an economic model of software market with a risk-sharing mechanism, which takes into account the strategic interplay of risk-sharing and security quality of the software given a certain level of functionality. We then apply this model in different scenarios, and examine the implications of risk-sharing mechanism in the context of cyber security. Our model provides evidence of under-provided security quality of software under the monopoly case, as has been observed in the market. We consider feasibility and effectiveness of the risk-sharing mechanism under various scenarios, and our results suggest that this mechanism is promising.

KEY WORDS AND PHRASES: Cyber Security, Software Quality, Risk-Sharing

1. Introduction

As the Internet has revolutionized the way individuals, industry and the government communicate and conduct their daily business, the intensive interconnectivity has increased the vulnerability of computer systems. Consequently, network security becomes a major issue for e-business and corporate communications. The first report of the Joint Security Commission to the United States Central Intelligence Agency and the Department of Defense (Smith, 1994) stated, "The security of information systems and networks is the major security challenge of this decade and possibly the next century."

Network security includes protecting data against accident or malicious intent, verifying identity of users who have access to data, preserving privacy and confidentiality, insuring validity of transactions, detecting network viruses, and preventing system crashes. Failure to protect computer systems can cause enormous losses such as physical destruction or theft of tangible assets, loss of data or program files, theft of information, and delayed processing. To cope with the new risk, the computer industry has tried to develop new weapons such as firewalls, encryption techniques, access control mechanisms, and intrusion detection systems. The federal government has formed the Department of Homeland Security and is developing a National Strategy to Secure Cyber Space. Despite these efforts, the security level of computer networks is still very low, and the potential loss is enormous. Ernst & Young's Global Information Security Survey in 2002 shows that only 40% of respondents are confident that they would detect a system attack and 75% had experienced unexpected unavailability (Ernst & Young, 2002). In another 2002 survey, the Federal Bureau of Investigation and the Computer Security Institute reported that 90% of respondents had been victimized by a cyberattack or security breach in the preceding 12 months, and the average estimated loss was about $2 million

per organization (Power, 2002).

Fisk (2002) argues that there are well known technical and procedural techniques for preventing computer system vulnerability. However, applying these techniques can be resource intensive and will not be done without sufficient incentive. One major reason for this low security level is that the software industry is at a sub-optimal, but self-supporting equilibrium that does not support the effort required for software improvements. Customers do not have good enough safeguards, both because available options on the security market seem to be ineffective but too expensive, and because the value of running safe operation is often not fully appreciated. They have learned to tolerate low-quality software, enabling vendors to be successful without improving the quality of their products. On the vendor side, both a perceived small market and high development costs have made developing high-quality software a significant risk to the vendors.

The slow growth of security market and the low quality of software have been identified as main causes of the poor state of network security (Yurcik and Doss, 2002). However, software vendors have no incentive to improve the quality of their products since they are not directly liable for any loss due to poor quality. To solve this problem, security experts suggest legal liability and cyber insurance mechanism as possible solutions. Unfortunately, not much research has been done on this problem from the economic perspective. In this paper, with the goal to improve software quality, we propose a risk-sharing mechanism between software vendors and customers as an alternative solution. According to Fisher (2002), some companies are already demanding liability clauses in contracts with vendors, holding the vendors responsible for any security breach connected to their software. Karl Keller, president of IS Power Inc., says, “Contractual liability is a great motivator. I’m encouraged that liability for vulnerabilities is

entering into contracts." We present an economic model of software market, which takes into account the strategic interplay of risk-sharing and software quality.

We consider two dimensions of software quality, which are functionality and security quality. We first examine the implications of risk-sharing mechanism both in monopoly and socially optimal cases. Our model provides evidence of under-provided quality of software under monopoly, as has been observed in the market. We find that the social planner who maximizes social surplus offers higher-quality product than the monopolist and that the risk-sharing factor and the quality of product are not strategic complements. This intuition provides insights for the otherwise unexpected result that neither the social planner nor the monopolist has any incentive to bear the risk. This is interesting in the sense that even for the social planner, sharing risk with the customers is not optimal at equilibrium, although risk-sharing mechanisms such as warranties are widely used in other industries.

We extend the model to duopoly competition. We start with examining the case where the entrant brings a product with the same quality level as the incumbent who does not want to share any risk. Unlike the monopoly case, we find that the entrant has an incentive to introduce positive risk-sharing to alleviate competition and that the risk-sharing level increases as the quality level increases. Then we extend this scenario to the case where vendors with products of same quality differentiate their products by offering different levels of risk-sharing. We find that in the presence of competition where two vendors differentiate their product not by quality but by risk-sharing, the high-value vendor is willing to share the risk whereas sharing no risk is the optimal choice for the low-value vendor. The high-value vendor's optimal level of risk-sharing is the same as the risk-sharing level of the entrant in the first duopoly scenario, which increases as the quality increases.

It was proposed by the policy makers and computer security experts that the government should offer tax incentives to businesses for spending on security (Harmon, 2003). We examine how government subsidy affects the quality level of the software under monopoly. We find that a subsidizing policy such as offering tax incentives creates no incentive for the vendor to bear the risk and worse, it reduces the quality level. Although the subsidizing policy may lead to higher level of customer security awareness, it may make the situation worse in terms of quality of the software. We investigate whether regulation on risk-sharing level gives an incentive for quality improvement to the monopolist. Interestingly, we find that the monopolist has an incentive to increase the quality when a certain level of risk-sharing is imposed by the government. Moreover, our results show that when the proportion of the expected loss to the cost of security development increases, the range of regulation which leads to higher quality also increases.

The rest of the paper is organized as follows. We provide a model and examine monopolist's and social planner's cases in section 2. We extend our model to various scenarios of duopoly competition in section 3. In section 4, we examine policy implication of the risk-sharing mechanism. We discuss existing mechanisms suggested by practitioners and prerequisites for the proposed risk-sharing mechanism in section 5. Finally, section 6 concludes the paper.

2. Model

We analyze a software vendor's decision on the quality and the risk-sharing levels, using a model built on the models of vertical quality differentiation (Mussa and Rosen, 1978). There are two types of players in the market: a software vendor and customers. It is shown that the best strategy for the software vendors is to introduce their products as early as possible and then to

patch them later (Arora et al., 2003). Consequently, the initial quality of software products is lower than expected. Security experts argue that the defects of such software are exploited by the malicious hackers to attack computer systems and that the quality of the general software in terms of security should be improved (CSTB/NRC, 1991). Customers in our model are considered to be firms that are likely to have higher incentive to adopt security solutions than do individuals, whose awareness of security in general is still very low in reality. We assume that increasing the quality of software reduces the expected loss from cyberattack in the life-span of the product. This is reasonable in the sense that attacks on computers or systems with more secure software are less likely to succeed.

2.1. Customer's Utility Function

We consider two dimensions of software quality, which are functionality and security quality. In early 2002, Microsoft stopped all Windows feature development and focused only on analysis of design, code, test plans and documentation. Our model reflects the current phenomenon by considering a vendor that emphasizes on security development given a certain level of functionality, V. Let q be the security quality of the software product where $q \in [0, 1]$. Security quality measures vulnerability of the software to attacks at the product launch. Bug-free software can be considered to be of perfect security quality. Following security experts' argument that the initial quality of product launch matters and that the availability of patching mechanism may worsen the situation, we focus on the initial quality in our model. $P(q)$ is the probability that an attack succeeds when q-quality software is installed and L is the loss caused by a successful attack. Under the proposed risk-sharing mechanism, the vendor takes some proportion of the risk, denoted by r. If an attack on the customer's network or system is

successful and incurs loss, then the vendor shares the responsibility with its damaged customer. Thus, the expected utility of a customer who purchases software with price p is

$$\begin{aligned} E(U) &= P(q)[\theta(V-(1-r)L)-p]+(1-P(q))[\theta V-p] \\ &= \theta[V-(1-r)K(q)]-p. \end{aligned} \quad (1)$$

The expected loss when q-quality software is installed is denoted by $K(q)$, which is $P(q)L$. $K(q)$ in our model is based on a certain period of time. It is reasonable since most software products are licensed to the corporate customers. Thus, the life-span of the software is considered to be the licensing period. We assume that $K'(q) < 0$ and $K''(q) > 0$, so that the expected loss decreases as the quality level increases at diminishing rate. θ captures customer heterogeneity indicating how much utility a customer derives from the software's functionality. Losses arise out of business activities. Thus, the same attack may cause more severe damage to some firms than others. If θ is high, the customer is more sensitive to security features of the product, in that she enjoys more utility from the product, but also suffers more disutility from a successful attack. It holds in reality that some firms are more sensitive to security than others. For example, banks may be such customers with high θ. We assume that θ is uniformly distributed on [0, 1]. Customers who have expected utility greater than zero buy the software whereas others do not.

2.2. Vendor's Profit Function

A software vendor's expected profit is

$$E(\pi(p,q,r)) = D(p,q,r)(p-rK(q))-C(q) \quad (2)$$

where $D(p, q, r)$ is the demand for the product, p is the price and $C(q)$ represents fixed cost for producing a product with quality level q. Production of information good such as software involves high fixed costs but low variable costs. In other words, the cost of producing

the original copy is substantial whereas the cost of producing additional copies is negligible. As a result, given the context of software product, the cost does not depend on quantity, that is, the variable cost of production is zero. We assume convex cost function, that is, $C'(q) > 0$ and $C''(q) > 0$, so that the cost increases as the quality level rises at a growing rate. $rK(q)$ is the expected loss, for which the vendor is responsible per unit of the product. Although variable cost of production is assumed to be zero, $rK(q)$ plays a role of variable cost in our model.

2.3. Market Equilibrium under Monopoly

The monopoly case is quite relevant to software industry. Consider the case of Microsoft that dominates the PC operating systems market. According to the "2001 Security Software Market Share" report, Symantec continued to lead the antivirus market, the largest segment of security software industry, with a 50 percent share of the combined enterprise and consumer business, more than double the nearest competitor. Thus, analyzing the monopoly case still has a significant implication for software industry.

We analyze a three-stage game. At the first stage, the monopolistic vendor decides the quality level *q* and the risk-sharing level *r* simultaneously and at the second, the vendor sets up the price *p*. Then the customers decide whether or not to buy the product at the last stage. Demand for the software product offered by a monopolist is derived from the equation (1), which is the customers' expected utility and the uniform distribution of θ. The demand is

$$D(p,q,r) = 1 - \frac{p}{V - (1-r)K(q)}.$$

Then the expected profit for the monopolist becomes

$$E(\pi(p,q,r)) = D(p,q,r)(p - rK(q)) - C(q)$$
$$= \left(1 - \frac{p}{V - (1-r)K(q)}\right)(p - rK(q)) - C(q). \quad (3)$$

The first order condition for p is

$$\frac{\partial E(\pi(p,q,r))}{\partial p} = 1 - \frac{2p}{V - (1-r)K(q)} + \frac{rK(q)}{V - (1-r)K(q)} = 0. \quad (4)$$

Thus,

$$p^* = \frac{V - (1-2r)K(q)}{2}. \quad (5)$$

Substituting p^* in (3) leads to

$$E(\pi(q,r)) = \frac{(V - K(q))^2}{4(V - (1-r)K(q))} - C(q). \quad (6)$$

Note that $r^* = \arg\max_r \pi(q,r) = 0$ since $(V - K(q))^2 > 0$ and $K(q) > 0$.

The first-order condition for q is

$$\frac{\partial E(\pi(q,r))}{\partial q} = -\frac{K'(q)(V - K(q))}{2(V - (1-r)K(q))} + \frac{(1-r)K'(q)(V - K(q))^2}{4(V - (1-r)K(q))^2} - C'(q) = 0. \quad (7)$$

At equilibrium, we have $r^* = 0$ and $\frac{K'(q^*)}{C'(q^*)} = -4$.

2.4. Social Planner's Solution

A social planner will offer the product at marginal cost which is $rK(q)$. Hence, the social surplus can be written as

$$S(q,r) = \int_{\theta^*}^{1} \theta[V - (1-r)K(q)]d\theta - (1-\theta^*)rK(q) - C(q) \quad \text{where} \quad \theta^* = \frac{rK(q)}{V - (1-r)K(q)}.$$

It simplifies to

$$S(q,r) = \frac{(V - K(q))^2}{2(V - (1-r)K(q))} - C(q). \tag{8}$$

Note that $r^* = \arg\max_r \pi(q,r) = 0$ since $(V - K(q))^2 > 0$ and $K(q) > 0$.

The first-order condition for q is

$$\frac{\partial S(q,r)}{\partial q} = -\frac{K'(q)(V - K(q))}{(V - (1-r)K(q))} + \frac{(1-r)K'(q)(V - K(q))^2}{2(V - (1-r)K(q))^2} - C'(q) = 0. \tag{9}$$

At equilibrium, we have $r^* = 0$ and $\frac{K'(q^*)}{C'(q^*)} = -2$.

Proposition 1: *In a software market, neither the monopolist nor the social planner is willing to share the risk. At equilibrium, the social planner offers a higher-quality product than the monopolist.*

Proof. Please refer to the Appendix.

This is interesting in the sense that neither social planner nor the monopolist has any incentive to bear the risk. Note that the risk-sharing factor does not affect the fixed cost and that sharing no risk allows the social planner to face zero marginal cost and to cover the entire market. In other words, the social planner is left with no resource to share the customer's risk when it serves the entire market by offering price at marginal cost. Thus, it turns out that even the social planner does not want to share any risk. Interestingly, the risk-sharing factor and the quality in our model are not strategic complements. In other words, it is not always true that the factors that increase the risk-sharing level also result in higher quality. Proposition 1 provides evidence of under-provided quality of software under monopoly, as what has been observed in the market. Figure 1 illustrates the relationship between the quality of a monopolist and a social planner.

[INSERT FIGURE 1 HERE]

3. Competition

3.1. Incumbent and Entrant with Same Quality but Different Risk-Sharing

We first study the case of a duopoly market with an incumbent and an entrant offering software products of same quality. This scenario captures the market where there are a monopolistic incumbent that has no incentive to share the risk and an entrant that enters the market bringing a product with the same quality level as the incumbent's product. Our focus is on whether the entrant has an incentive to share the risk and if so, how much it will be at equilibrium. In this game, the entrant chooses its optimal risk-sharing level first. Then both the incumbent and the entrant set up the price simultaneously. At the last stage, customers decide whether to buy from the incumbent or the entrant or neither.

The expected utility offered by the incumbent is

$$E(U_I) = \theta[V - K(q_I)] - p_I. \tag{10}$$

Similarly, the expected utility offered by the entrant is

$$E(U_E) = \theta[V - (1-r)K(q_E)] - p_E. \tag{11}$$

Note that $q_E = q_I = \bar{q}$. In order to derive the demand function for both vendors, we presume that customers can choose buying from the incumbent, the entrant or neither. For computational convenience, let *v* denote the total value offered to the customer. Thus,

$$v_I = V - K(\bar{q}) \quad \text{and} \quad v_E = V - (1-r)K(\bar{q}).$$

The customers will buy from the entrant when $\theta v_E - p_E \geq \theta v_I - p_I$ and buy from the incumbent when $\theta v_I - p_I > \theta v_E - p_E$ and $\theta v_I - p_I > 0$.

The demands for both firms can be derived from the above conditions.

$$D_E = 1 - \frac{p_E - p_I}{v_E - v_I} \text{ and } D_I = \frac{p_E - p_I}{v_E - v_I} - \frac{p_I}{v_I}.$$

The expected profits are

$$E(\pi_E) = \left(1 - \frac{p_E - p_I}{v_E - v_I}\right)(p_E - rK(\bar{q})) - C(\bar{q}) \tag{12}$$

$$E(\pi_I) = \left(\frac{p_E - p_I}{v_E - v_I} - \frac{p_I}{v_I}\right)p_I - C(\bar{q}). \tag{13}$$

By analyzing conditions that maximizes (12) and (13), we derive the following results.

Proposition 2: *In the presence of competition, the entrant offering the same quality product as the incumbent sharing no risk has an incentive to introduce positive risk-sharing to alleviate competition. Moreover, as the quality level increases, the risk-sharing level also increases.*

Proof. Please refer to the Appendix.

In contrast to the monopolist and the social planner, the entrant in this scenario has an incentive to share the risk. It follows that the risk-sharing level increases as the quality level increases. This result is quite interesting in the sense that without the risk-sharing mechanism, the entrant may have less incentive to enter the market because its entry may trigger Bertrand-like price competition.

3.2. Duopoly Competition with Same Quality but Different Risk-Sharing

This scenario serves as an extended case. In this scenario, two vendors compete against each other with the product of same quality. However, vendors differentiate their products by offering different levels of risk-sharing. At the first stage, both vendors decide the risk-sharing level given quality and at the second, optimal prices are chosen. At the third stage, customers

decide whether to buy from the incumbent or from the entrant or neither. We label the vendor sharing high risk, hence offering high value to customers as H vendor and denote the other vendor sharing low risk, hence offering low value to customers as L vendor. Then the total value offered to the customer is

$$v_H = V - (1 - r_H)K(\bar{q}) \quad \text{and} \quad v_L = V - (1 - r_L)K(\bar{q}) \quad \text{where} \quad r_H > r_L.$$

Following the same logic as we applied in the previous section yields

$$E(\pi_H) = \left(1 - \frac{p_H - p_L}{v_H - v_L}\right)(p_H - r_H K(\bar{q})) - C(\bar{q}) \tag{14}$$

$$E(\pi_L) = \left(\frac{p_H - p_L}{v_H - v_L} - \frac{p_L}{v_L}\right)(p_L - r_L K(\bar{q})) - C(\bar{q}). \tag{15}$$

It can be verified that the solutions for $\frac{\partial E(\pi_L)}{\partial r_L}$=0 at equilibrium prices are complex numbers. In other words, $r_L^* = \arg\max E(\pi_L)$ is not an interior solution but a boundary one. Since $r_H > r_L$ for any value of r_H,

$$r_L^* = \arg\max E(\pi_L) = 0. \tag{16}$$

Substituting r_L^* in $\frac{\partial E(\pi_H)}{\partial r_H}$ yields

$$r_H^* = \frac{3(V - K(\bar{q}))}{4K(\bar{q})}. \tag{17}$$

Proposition 3: *In the presence of competition where two vendors offer same-quality products, in equilibrium, risk-sharing acts as a differentiator that one firm will share positive risk,* $r_H^* = \frac{3(V - K(\bar{q}))}{4K(\bar{q})}$ *and thus offer higher value to customers, while sharing no risk is the optimal choice for the other firm.*

Proof. The results directly come from (16) and (17).

Proposition 3 shows the optimal risk-sharing level for each of the high-value and the low-value vendors. When they differentiate their products by offering different levels of risk-sharing, the high-value vendor has an incentive to share positive risk whereas the low-value vendor is not willing to bear any risk. This is interesting in the sense that for the low-value vendor, risk-sharing may seem to be a risky business when it perceives that its rival will share higher risk than itself. Also note that high-value vendor's optimal risk-sharing level increases as the quality level increases. Thus, customers may use the high-value vendor's risk-sharing level as a proxy of its quality level.

4. Policy Implication

4.1. Subsidy for Customers

Proposals for government action being discussed by policy makers and computer security experts include offering tax incentives to businesses for spending on security (Harmon, 2003). In this section, we examine how government subsidy for customers affects the quality level of the software under monopoly. Let s be the subsidy for each customer who makes a purchase of software. Then the expected utility of the customer is

$$E(U) = \theta[V - (1-r)K(q)] - p + s. \tag{18}$$

The demand derived from the above expected utility is

$$D(p,q,r) = 1 - \frac{p-s}{V-(1-r)K(q)}.$$

A monopolist vendor's expected profit is then

$$E(\pi(p,q,r)) = D(p,q,r)(p - rK(q)) - C(q)$$
$$= \left(1 - \frac{p-s}{V-(1-r)K(q)}\right)(p - rK(q)) - C(q). \quad (19)$$

At equilibrium, we have $r^* = 0$ and

$$\frac{C'(q^*)}{K'(q^*)} = -\frac{1}{4} + \frac{s^2}{4(V - K(q^*))^2}. \quad (20)$$

Proposition 4: *For a software product, the monopolist reduces the quality of its product when government subsidizes the customers. In terms of quality improvement, government's subsidizing policy makes the problem worse in monopoly case.*

Proof. Please refer to the Appendix.

Interestingly, we find that the government subsidizing policy creates no incentive for the vendor to bear the risk and worse, it reduces the quality level. Poor quality of available options in the software market has been identified as the main cause of the poor state of security. Although the subsidizing policy may motivate the customers to get the software, it may make the situation worse because the problem is on the vendor side. Our findings imply that cheering up the customers may not work effectively. Rather, finding a way to penalize the vendor that produces bad-quality products may be a more effective policy to make our cyber space secure.

4.2. Regulation on Risk-Sharing

We investigate whether government regulation on risk-sharing creates an incentive for the monopolist to increase quality. This is a policy that directly regulates the vendor unlike subsidizing the customers. We assume the expected loss and the cost to be quadratic functions of quality as follows:

$$K(q) = V(1-q)^2, \quad V > 0$$
$$C(q) = cq^2, \quad c > 0.$$

Note that V is functionality of the software as defined earlier. Suppose that the government imposes regulation on risk-sharing. Let r be the risk-sharing level that the software vendor is responsible for. The expected profit for the monopolist from (6) is

$$E(\pi(q,r)) = \frac{(V - K(q))^2}{4(V - (1-r)K(q))} - C(q).$$

Given r, the first-order condition for q from (7) is

$$\frac{\partial E(\pi(q,r))}{\partial q} = -\frac{K'(q)(V - K(q))}{2(V - (1-r)K(q))} + \frac{(1-r)K'(q)(V - K(q))^2}{4(V - (1-r)K(q))^2} - C'(q) = 0.$$

Thus, we have

$$K'(q)(V - K(q))\{(V - K(q)) + r(V + K(q))\} + 4C'(q)(V - (1-r)K(q))^2 = 0$$
$$\Leftrightarrow V(q-1)(q-2)\{q(q-2) - r(1 + (1-q)^2)\} + 4c\{1 - (1-r)(1-q)^2\}^2 = 0. \quad (21)$$

Further derivation leads to the following Proposition.

Proposition 5: *When the government imposes risk-sharing between 0 and* $\bar{r}$ *where* $\bar{r} = \frac{1}{256}\left(\frac{V}{c}\right)^2\left(\frac{V}{c} + 8\right)^2$*, the monopolist increases the quality of the product. As the proportion of V to c increases, the range of regulation which leads to higher quality also increases.*

Proof. Please refer to the Appendix.

We find that regulation on risk-sharing level works better than the subsidizing policy in terms of quality improvement. It implies that the government should adopt the policy to directly penalize the monopolist that produces bad-quality product rather than motivating customers. Imposing a certain level of risk-sharing on the monopolist side can be one example. The results show the desirable range of such regulation that creates incentive for the monopolist to increase

quality. Intuitively, as the proportion of functionality to the cost for quality development becomes higher, the admissible range of risk-sharing level that leads to higher quality becomes wider. However, imposing too much risk-sharing on the monopolist may make the situation worse by making the monopolist reduce security quality. Our findings may give policy makers a guideline when they want to regulate the monopolized software market.

[INSERT FIGURE 2 HERE]

Figure 2 illustrates how quality changes as the government imposes different level of risk-sharing on the monopolist. It shows that there exists a desirable range of regulation on risk-sharing that creates an incentive for the monopolist to improve the quality and that the range increases as the proportion of the expected loss to security development cost increases. It has a policy implication that regulation on risk-sharing may be a good way for quality improvement but imposing too much risk-sharing on the monopolist may make situation even worse.

5. Discussion

Policy makers and security experts suggest two solutions to the software market problem: legal liability and cyber insurance. With a liability mechanism, software and systems vendors are legally liable for safety- or security-relevant flaws that involve negligence or misrepresentation. Legal enforcement is expected to increase the quality of the software and systems. However, the computer industry is uncomfortable with regulation, fearing that it may inhibit innovation by linking legal risks and the development of new products, and discourage production. Also, there is a delicate issue of open source software. The writers of open source software can be considered to be volunteers, so imposing legal liability on them may not be fair. Another problem is that while liability mechanisms may indirectly increase the customers' utility by

improving software quality, they do not directly increase customers' utility and, therefore, may not give sufficient incentives for the customers to have more secure systems. Considering the characteristics of the software industry, a voluntary and market-driven regulation is likely to be more effective than a mandatory one.

Some researchers argue that insurance companies can be a market lever to encourage sound security by setting up standards for best practices, applying pressure on firms to reduce insurance premiums, and providing incentives for software companies to offer secure products. However, the insurance companies argue that the victims of computer mishaps are reluctant to make their information public by reporting to a third party – the insurance company. Another problem with cyber insurance mechanisms is that vendors of software and systems are not responsible for the losses caused by the low quality of their products. Although the cyber insurance mechanism may indirectly create incentives for vendors to produce higher-quality software, it cannot directly impose liability on the software vendors. Moreover, insurance companies are reluctant to get involved in the software industry due to uncertainty, lack of adequate statistics and technological changes, although some companies (such as Safeware, American Insurance Group, and Zurich) are offering policies ranging from hardware replacement to full information-asset protection.

Unlike the existing solutions, a risk-sharing mechanism can directly affect the customer's utility and the vendor's profit. Under this mechanism, software vendors are responsible for a certain proportion of the loss caused by security breaches of their products. Thus, risk-sharing can create incentives for the software vendors to develop reliable products. On the customer side, since the vendor's risk-sharing reduces the burden on the customer, the demand for the secure software products is expected to increase. In addition, a risk-sharing mechanism requires an

endeavor to predict loss, which can increase the customer's awareness of the importance of cyber security. Therefore, we suggest a risk-sharing mechanism as a form of voluntary regulation. It imposes liability on the software vendor and allows the customers to share their risk without revealing their information to a third party. By improving both the quality of the product and the customer's awareness, the risk-sharing mechanism is expected to improve the level of network security of our society.

6. Conclusion

To enhance the poor state of network security, one needs to solve the fundamental problem – giving software vendors an incentive to increase the quality of their products. This paper proposes a risk-sharing mechanism between software vendors and customers as a potential solution, and analyzes this approach under various scenarios. We present an economic model of the software market, which takes into account the strategic interplay of risk-sharing and quality of product. We first compare the monopolist's and the social planner's solutions. Our results give evidence of under-provided quality of software under monopoly, as has been observed in the market. We find that the social planner who maximizes social surplus offers higher-quality product than the monopolist and that neither the social planner nor the monopolist has any to bear the risk. This is interesting in the sense that even for the social planner, sharing risk with the customers is not optimal at equilibrium although risk-sharing mechanisms such as warranties are widely used in other industries. This can be explained by intuition that risk-sharing and quality are not strategic complements.

We extend the model to duopoly competition. In the case where the entrant brings a product with the same quality level as the incumbent who does not want to share any risk, we

find that the entrant has an incentive to introduce positive risk-sharing to alleviate competition and that the risk-sharing level increases as the quality level increases. We also find that in the presence of competition where two vendors differentiate their product not by quality but by risk-sharing, the high-value vendor is willing to share the risk whereas sharing no risk is the optimal choice for the low-value vendor and that the high-value vendor's optimal level of risk-sharing increases as the quality increases.

We examine how different forms of government policy affect the quality level of the software under monopoly. First, we analyze the software market where the government subsidizes the customers who spend on security. Unlike the prediction of the practitioners, we find that the government subsidizing policy creates no incentive for the monopolistic vendor to bear the risk and even reduces the quality level. Then we investigate whether regulation on risk-sharing creates an incentive for the monopolist to increase quality. Our findings show that a certain level of regulation on risk-sharing leads to higher quality and that it becomes more effective as the expected loss gets more severe compared to the cost for quality improvement. This implies that the government should adopt the policy to directly penalize the monopolist that produces bad-quality software rather than motivating customers. Regulation on risk-sharing can be one example.

Software liability has been an important issue among computer scientists and practitioners. Nevertheless, no effective liability-imposing mechanism has been found yet. We propose a risk-sharing mechanism as a possible solution. Our research contributes to the literature in the following ways. First, we provide an economic framework to a security issue where only a little previous research has dealt with the problem of the software market from an economic perspective although the solution to this problem is economic rather than technical.

Second, our results suggest that a risk-sharing mechanism as a form of market-driven regulation that imposes liability on the software vendor is promising. Finally, we illustrate how a policy maker can establish an effective way to increase the quality of software.

While significant, this study can be improved in several ways. First, in our duopoly competition scenarios, we assume that vendor compete with same quality but different risk-sharing. Examining a scenario with different quality and different risk-sharing will be interesting. Second, comparison of the risk-sharing mechanism with other existing mechanisms such as legal liability and cyber insurance based on the effectiveness can be a way of extending this paper. Third, developing a way to deploy the risk-sharing mechanism may form a separate research area. For example, loss measurement and risk analysis are prerequisites of the proposed risk-sharing mechanism.

Appendix of Mathematical Proofs

Proof of Social Surplus

Let $X = V - (1-r)K(q)$ and $Y = rK(q)$. Then

$$S(q,r) = \int_{\frac{Y}{X}}^{1} \theta X d\theta - \left(1 - \frac{Y}{X}\right)Y - C(q) = \frac{(X-Y)^2}{2X} - C(q) = \frac{(V-K(q))^2}{2(V-(1-r)K(q))} - C(q).$$

This completes proof. ***QED.***

Proof of Proposition 1

Denote $F = \frac{K'(q)}{C'(q)}$. Showing that *F* is an increasing function of *q* completes the proof.

$$\frac{\partial F}{\partial q} = \frac{K''(q)}{C'(q)} - \frac{K'(q)}{C'(q)^2}C''(q) = \frac{K'(q)}{C'(q)}\left(\frac{K''(q)}{K'(q)} - \frac{C''(q)}{C'(q)}\right).$$

Note that $C'(q) > 0, C''(q) > 0, K'(q) < 0$ and $K''(q) > 0$ by assumption. Thus,

$$\frac{K'(q)}{C'(q)} < 0 \text{ and } \frac{K''(q)}{K'(q)} < 0 < \frac{C''(q)}{C'(q)}.$$

Therefore, we have $\frac{\partial F}{\partial q} > 0$. ***QED.***

Proof of Proposition 2

The first order conditions for p_E and p_I are

$$\frac{\partial E(\pi_E)}{\partial p_E} = 2 - \frac{2p_E}{v_E - v_I} + \frac{p_I}{v_E - v_I} = 0 \text{ and } \frac{\partial E(\pi_I)}{\partial p_I} = \frac{p_E}{v_E - v_I} - \frac{2p_I}{v_E - v_I} - \frac{2p_I}{v_I} = 0.$$

Solving the first order conditions p_E and p_I gives the optimal prices charged by both vendors:

$$p_E^{\ *} = \frac{4v_E(v_E - v_I)}{4v_E - v_I} \text{ and } p_I^{\ *} = \frac{2v_I(v_E - v_I)}{4v_E - v_I}.$$

Substituting $p_E^{\ *}$ and $p_I^{\ *}$ in equation (12) yields

$$E(\pi_E(r)) = \left(1 - \frac{p_E^* - p_I^*}{v_E - v_I}\right)(p_E^* - rK(\bar{q})) - C(\bar{q})$$
$$= (V - K(\bar{q}))^2 \frac{rK(\bar{q})}{(3V - 3K(\bar{q}) + 4rK(\bar{q}))^2} - C(\bar{q}).$$

The first order condition for r is

$$\frac{\partial E(\pi_E(r))}{\partial r} = (V - K(\bar{q}))^2 \frac{K(\bar{q})}{(3V - 3K(\bar{q}) + 4rK(\bar{q}))^2} - (V - K(\bar{q}))^2 \frac{8rK(\bar{q})^2}{(3V - 3K(\bar{q}) + 4rK(\bar{q}))^3} = 0.$$

Thus, the optimal risk-sharing level for the entrant is

$$r^* = \frac{3(V - K(\bar{q}))}{4K(\bar{q})}.$$

Then we have

$$\frac{\partial r^*}{\partial \bar{q}} = -\frac{3K'(\bar{q})}{4K(\bar{q})} - \frac{3(V - K(\bar{q}))}{4K(\bar{q})^2}K'(\bar{q}) = -\frac{3K'(\bar{q})}{4K(\bar{q})^2}V.$$

Note that $V > 0, K(\bar{q})^2 > 0,$ and $K'(\bar{q}) < 0.$ Therefore, $\frac{\partial r^*}{\partial \bar{q}} > 0.$ ***QED.***

Proof of Proposition 4

Let q_1 be the optimal quality of the monopolist without subsidy and q_2 be the optimal quality of the monopolist with subsidy. Then we have

$$\frac{C'(q_1)}{K'(q_1)} = -\frac{1}{4} \quad \text{and} \quad \frac{C'(q_2)}{K'(q_2)} = -\frac{1}{4} + \frac{s^2}{4(V - K(q_2))^2}.$$

Thus,

$$\frac{C'(q_2)}{K'(q_2)} - \frac{C'(q_1)}{K'(q_1)} = \frac{s^2}{4(V - K(q_2))^2} > 0.$$

In the proof of proposition 1, we have shown that $\frac{K'(q)}{C'(q)}$ is increasing in q, that is, $\frac{C'(q)}{K'(q)}$ is decreasing in q. Therefore, $q_1 > q_2$. ***QED.***

Proof of Proposition 5

Solving (21) for *r* yields

$$r_1 = \frac{(q-2)}{8c(q-1)^3}\left\{q^2(V+8c) - 2q(V+4c) + 2V - \sqrt{V^2q^4 - 4V^2q^3 + 8V(V+2c)q^2 - 8V(V+2c)q + 4V^2}\right\}$$

$$r_2 = \frac{(q-2)}{8c(q-1)^3}\left\{q^2(V+8c) - 2q(V+4c) + 2V + \sqrt{V^2q^4 - 4V^2q^3 + 8V(V+2c)q^2 - 8V(V+2c)q + 4V^2}\right\}.$$

Denote q_m be the monopolist's equilibrium quality when there is no regulation. Then

$$r_1(q_m) = 0$$

$$\Leftrightarrow q_m^2(V+8c) - 2q_m(V+4c) + 2V = \sqrt{V^2q_m^4 - 4V^2q_m^3 + 8V(V+2c)q_m^2 - 8V(V+2c)q_m + 4V^2}.$$

Thus,

$$r_2(q_m) = \frac{(q_m-2)}{4c(q_m-1)^3}\left\{q_m^2(V+8c) - 2q_m(V+4c) + 2V\right\}.$$

Solving for the optimal quality of the monopolist yields

$$\frac{K'(q_m)}{C'(q_m)} = -4 \Leftrightarrow q_m = \frac{V}{V+4c}.$$

Thus,

$$r_2(q_m) = \frac{\left(\frac{V}{V+4c}-1\right)}{4c\left(\frac{V}{V+4c}-1\right)^3}\left\{\left(\frac{V}{V+4c}\right)^2(V+8c) - 2\left(\frac{V}{V+4c}\right)(V+4c) + 2V\right\}$$

$$= \frac{1}{256}\left(\frac{V}{c}\right)^2\left(\frac{V}{c}+8\right)^2.$$

Let $x = \frac{V}{c}$, then $\bar{r} = \frac{x^2(x+8)^2}{256}$. Thus we have

$$\frac{\partial \bar{r}}{\partial x} = \frac{1}{64}x(x+4)(x+8) > 0.$$

Thus, $\bar{r}$ increases as $\frac{V}{c}$ increases. ***QED.***

Appendix of Figures

Figure 1

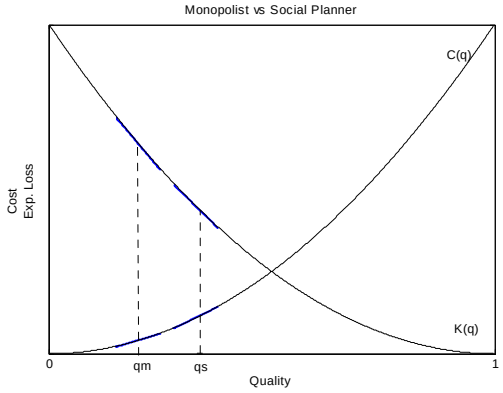

Figure 1. Equilibrium quality of monopolist and social planner

Figure 2

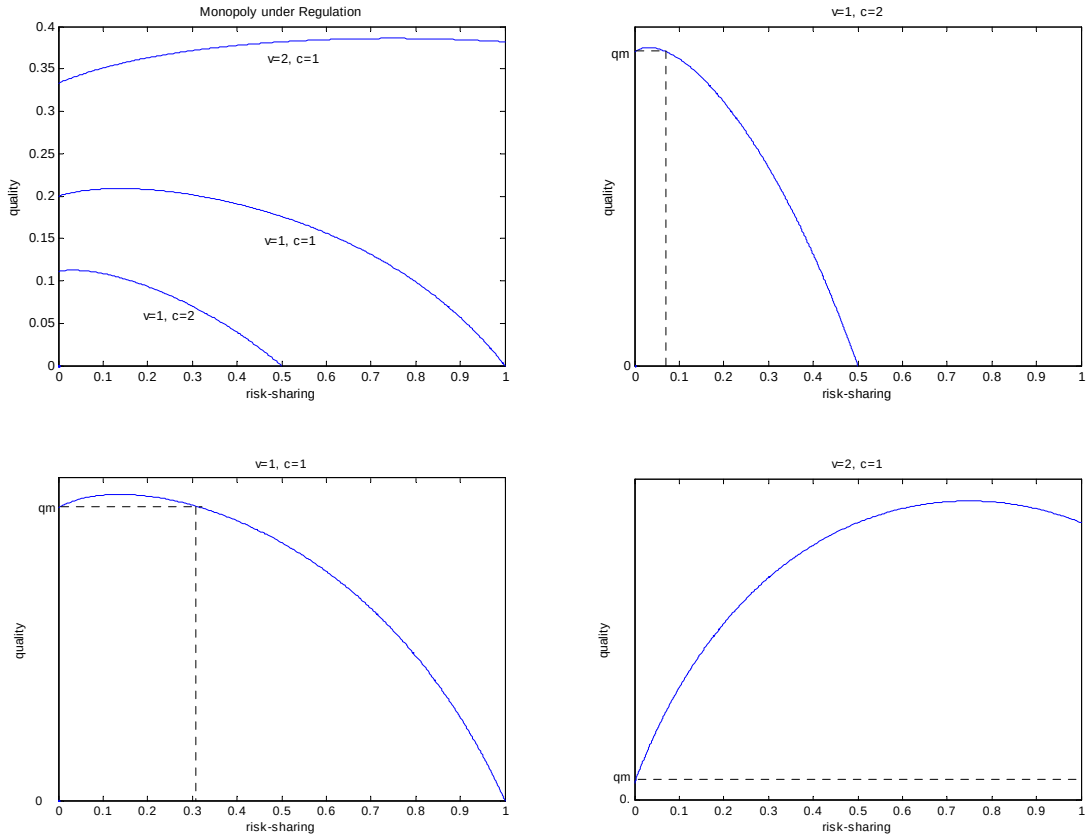

Figure 2. Monopoly under regulation

References

1. Arora A., J. Caulkins, and R. Telang, "Sell First, Fix Later: Impact of Patching on Software Quality", *Working Paper*, Carnegie Mellon University, 2003.
2. Cavusoglu H., B. Mishra, and S. Raghunathan, "A Model for Evaulating IT Security Investments", *Communications of the ACM* 47(7): 87-92, 2003.
3. Cavusoglu H., B. Mishra, and S. Raghunathan, "The Effect of Internet Security Breach Announcements on Market Value: Capital Market Reaction for Breached Firms and Internet Security Developers", *International Journal of Electronic Commerce* 9(4): 69-105, 2004.
4. Computer Science and Telecommunications Board (CSTB) and National Research Council (NRC), *Computers at Risk: Safe Computing in the Information Age*, National Academy Press, 1991.
5. Ernst & Young, "Global Information Security Survey 2002", http://www.ey.com.pl /gcrdownload_/GISS_2002.pdf, 2002.
6. Fisher D., "Contracts Getting Tough on Security", *eWeek*, April 15, 2002.
7. Fisk M., "Causes & Remedies for Social Acceptance of Network Insecurity", *in Proceeding of the Workshop on Economics and Information Security*, University of California, Berkeley, May 16-17, 2002.
8. Harmon A., "Digital Vandalism Spurs a Call for Oversight", *New York Times,* September 1, 2003.
9. Krishnan M. S., C. H. Kriebel, S. Kekre, and T. Mukhopadhyay, "An Empirical Analysis of Productivity and Quality in Software Products", *Management Science* 46(6): 745 – 759, 2000.

10. Mussa M. and S. Rosen, “Monopoly and Product Quality”, *Journal of Economic Theory* 18: 301-317, 1978.
11. Power R., “2002 CSI/FBI Computer Crime and Security Survey”, *Computer Security Issues and Trends* 8:1-22, 2002.
12. Ronnen U., “Minimum Quality Standards, Fixed Costs, and Competition”, *The RAND Journal of Economics* 22: 490-504, 1991.
13. Schneier B., “Information Security: How liable should vendors be?”, *Computer World*, October 28, 2004.
14. Smith J., “Redefining Security”, *A Report of the Joint Security Commission*, 1994.
15. Spence M., “Monopoly, Quality and Regulation”, *Bell Journal of Economics* 6: 417-429, 1975.
16. Spence M., “Product Differentiation and Welfare”, *American Economic Review*, 66:407 – 414, 1976.
17. Varian H. R., “Managing Online Security Risks”, *New York Times,* June 1, 2000.
18. Yurcik W and D. Doss, “Cyberinsurance: A Market Solution to the Internet Security Market Failure”, *in Proceeding of the Workshop on Economics and Information Security*, University of California, Berkeley, May 16-17, 2002.