THE IMPACT OF IMMEDIATE DISCLOSURE ON ATTACK DIFFUSION AND VOLUME

Sam Ransbotham
Carroll School of Management
Boston College
Chestnut Hill, Massachusetts 02467
sam.ransbotham@bc.edu

Sabyasachi Mitra
College of Management
Georgia Institute of Technology
Atlanta, Georgia 30332
saby.mitra@mgt.gatech.edu

May 2011

INTRODUCTION

Most common types of attacks on computer systems exploit vulnerabilities present in the software running on these systems (Cavusoglu et al. 2007; Cavusoglu et al. 2008). These errors in software can be eliminated through corrective patches released by the software vendor, or their effects can often be contained through other protective measures initiated by security professionals. Thus, the impact of a software vulnerability depends on whether the software vendor and security professionals have the opportunity to eliminate the vulnerability (or otherwise protect systems) before the vulnerability is exploited by attackers. Consequently, the discovery and disclosure process for vulnerabilities plays a vital role in securing computer systems. The key question is how to design effective disclosure processes that advantage security professionals and disadvantage attackers.

There are two primary methods for disclosing vulnerabilities discovered by security professionals. First, security professionals can disclose the vulnerability immediately after discovery through security mailing lists such as BugTraq. We refer to this pathway as *Immediate Disclosure*. When disclosed through immediate disclosure, the vulnerability information is immediately disseminated to security professionals who can install countermeasures, to vendors who can develop patches, and to potential attackers who can also exploit the information to their advantage. Second, security professionals may report the vulnerability to CERT (Computer Emergency Response Team) or other similar agencies (e.g. the private vulnerability markets operated by iDefense and Tipping Point). We refer to this pathway as *Non-public Disclosure* (Ransbotham et al. 2011). These agencies immediately notify the software vendor and disclose the vulnerability to the public when a patch is available from the vendor, or after a specific period (typically 45-180 days after notifying the vendor). In non-public disclosure, security service providers and potential attackers receive notification at the time of public disclosure, while vendors are notified in advance so that they can develop patches. When a vulnerability is discovered by attackers, it is exploited first before it is discovered by security professionals (after an attack is detected) and finally reported to agencies like CERT.

A significant debate in the security industry revolves around the benefits and drawbacks of immediate disclosure. The dominant viewpoint, termed as *Responsible Disclosure,* encourages disclosure through CERT and other similar mechanisms that provide a reasonable time for the vendor to develop patches before the vulnerability is disclosed to the public. The basic motivation behind responsible disclosure, which is supported by many software vendors and security professionals, is that the alternative immediate disclosure creates an unsafe period when the vulnerability may be exploited before the patch is developed and deployed. Proponents of responsible disclosure therefore argue that responsible disclosure will lead to lower risk of attack, more protected systems, and a safer security environment. On the other hand, immediate disclosure is often motivated by the need to force unresponsive vendors to address a vulnerability and to create incentives for developing secure software (Arora et al. 2006; Arora et al. 2008). Proponents argue that immediate disclosure will lead to more responsive software vendors and more alert security service providers, and consequently a safer information security environment.

In this paper, we shed light on this overall debate through an empirical study that compares vulnerabilities disclosed through the *immediate disclosure* and *non-public disclosure* mechanisms. Specifically, we evaluate the *impact* of immediate disclosure by analyzing over 2.4 billion information security alerts for 960 clients of an US based security service provider. We examine four measures of impact: (a) attack delay —does immediate disclosure speed the diffusion of attacks corresponding to the vulnerability through the population of target systems, (b) attack penetration – does immediate disclosure increase the number of systems affected by the vulnerability within the population of target systems, (c) attack risk – does immediate disclosure increase the risk that a computer system is attacked for the first time on any specific day after the vulnerability is reported, and (d) attack volume—does immediate disclosure increase the volume of attacks based on the vulnerability? Attack delay, attack penetration and risk of first attack are important because they affect the time that vendors have to release a patch and security professionals have to protect systems before they are attacked. Likewise, attack volume measures the overall amount of malicious attack activity (Park et al. 2007).

There are two primary contributions of this research to the information security literature. First, while several analytical models have examined optimal vulnerability disclosure and patching policies (Arora et al. 2006; August and Tunca 2006; Cavusoglu et al. 2007; Arora et al. 2008; August and Tunca 2008), this research is one of a few that empirically evaluate the effect of disclosure policies through the examination of intrusion detection system (IDS) data. Second, we empirically evaluate a research question that is of significant practical importance for policy formulation— whether immediate disclosure has a detrimental effect on information security. We believe that our findings are of practical interest to policy makers and vendors.

The rest of the paper is organized as follows. In the next section, we summarize the hypotheses examined in this research. In the following section, we describe the data and empirical methods used to evaluate our hypotheses. We then describe the results of our empirical analysis, and the final section summarizes the implications of our analysis.

HYPOTHESES DEVELOPMENT

Attack Delay and Risk of First Attack

The dominant view in the information security community is that immediate disclosure will lead to a less secure environment because public disclosure of the vulnerability can lead to systems being attacked before the vendor provides a patch or before security professionals can protect systems. In contrast, when a vulnerability is reported through CERT and other similar agencies, there is a lag between the discovery of the vulnerability and subsequent public disclosure. Consequently, responsible disclosure introduces a delay in the start of the diffusion process for attacks because attackers, on average, become aware of the vulnerability at a later date. Further, on any specific day after the vulnerability is discovered, the delay associated with responsible disclosure also reduces the risk of first attack corresponding to the vulnerability. The risk of first attack measures the probability that a target system is attacked on any specific day after the vulnerability is discovered, given that the target has not been attacked until that

time. Both the attack delay and the risk of first attack are important metrics because they affect the time that the vendor has to correct the vulnerability and that security professionals have to otherwise protect systems. This discussion leads to the following two hypotheses.

H1: The diffusion of attacks through the population of target systems will have less delay for vulnerabilities reported through immediate disclosure.

H2: The risk of first attack for a target system on any specific day after the vulnerability is discovered will be higher for vulnerabilities reported through immediate disclosure.

Attack Penetration and Volume of Attacks

When a patch corresponding to a vulnerability is not available, specific countermeasures can provide partial protection against attacks through three types of countermeasures that limit the impact of a vulnerability (Ransbotham and Mitra 2009): (a) access control methods that limit access to the affected software, (b) feature control methods that disable functionality and features in the affected software and devices, and (c) traffic control methods that filter suspicious traffic based on the attack signature. Similar descriptions of countermeasures also appear in (Ransbotham et al. 2011). Countermeasures are easier to develop and deploy than patches, but they provide imperfect protection until the vulnerability is corrected through patches.

We argue that immediate disclosure induces a race between attackers who attack systems and security service providers who develop and install countermeasures to protect systems. This race, which is similar in concept to a patent race in the economics literature (Denicolo 2000), raises urgency among security service providers and accelerates the development and deployment of countermeasures. Consequently, the time window for successful exploitation by attackers is small until countermeasures are installed, and the vulnerability has a short life span. The shorter life span leads to a lower penetration level of attacks among the population of target systems since many target systems have countermeasures installed and the population of vulnerable systems rapidly decreases. The short life span of the vulnerability and its lower penetration levels among target systems reduces the overall volume of attacks

as attackers divert their attention to more profitable opportunities. This forms the basis of the following two hypotheses:

H3: The diffusion of attacks through the population of target systems will have reduced penetration for vulnerabilities reported through immediate disclosure.

H4: The volume of attacks will be lower for vulnerabilities reported through immediate disclosure.

DATA AND METHODS

We utilize two main data sources for the study. First, we use a database of alerts generated from intrusion detection systems (IDS) installed in client firms of a security service provider. The dataset contains real alert data (as opposed to data from a research setting) from a large number of clients with varied infrastructure across many industries. The alert database contained over four hundred million alerts generated during 2006 and 2007 for over 900 clients of the security service provider. We created a panel dataset of the number of alerts generated every day during the two-year period of our analysis, for each target firm and specific vulnerability. That is, each data point in our dataset is for a specific target firm – vulnerability combination, and it contains a count of the number of alerts generated for each day in the two year period (2006-2007).

We combine the above data set with information in the National Vulnerabilities Database (NVD 2008) to obtain several characteristics of the vulnerabilities we study. The NVD obtains data from several other public vulnerability data sources such as CERT, BugTraq, XForce and Secunia. We match the records in our alert database with the data in the NVD through a CERT assigned unique ID for each vulnerability. We use the following variables from the NVD data as controls in our empirical analysis to ensure that the results we observe are due to immediate disclosure and not because of the characteristics of the vulnerability itself. The control variables are described below and shown in italics.

Once the attacker has access, vulnerabilities require varying degrees of complexity to exploit and are categorized by experts as *Low, Medium* or *High Complexity* and we include control variables for medium and high complexity, with low complexity as the base type. We also include an indicator

variable (*Sig*) that is set to 1 if a signature was available at the time that the vulnerability was disclosed, 0 otherwise. The *Impact* of a vulnerability is categorized by experts into one or more categories, and we use an indicator variable for each impact category that is set to 1 if the potential for the specific impact is present, 0 otherwise. The NVD classifies vulnerabilities into several different *Types* based on the software defect that the vulnerability represents, and we used indicator variables to control for each vulnerability type. We also include an indicator variable (*Patch*) that is set to 1 if a patch was available on the focal day of analysis, 0 otherwise. We also include the *Age* of the vulnerability (log transformed) at the time of our analysis (measured by the number of days since the vulnerability was reported) to control for any age related effects. An additional variable (*Server)* indicates whether the software corresponding to vulnerability is desktop (0) or server (1) based.

Our focal variable *(Immediate)* indicates if a disclosure was made through a public forum (e.g. BugTraq). An important caveat is that we classify a vulnerability as *immediate* if it is ever reported on a public forum, even if it may also have been reported through other reporting agencies. Thus, some vulnerabilities may be misclassified as immediate, making it more difficult to obtain significant results. Consequently, our results will be stronger if we could better identify immediately disclosed vulnerabilities. (Our research is ongoing to further clarify the *first* disclosure mechanism.)

Table 1 shows selected descriptive statistics for the vulnerabilities in our sample, divided into immediate and non-immediate disclosure vulnerabilities. The two types of vulnerabilities are similar in terms of the reported characteristics.

Modeling the Diffusion of Attacks

We model the diffusion of attacks through the population of target systems through a s-curve that has been extensively used to model the diffusion of innovations (Rogers 2003). Let *N(t)* be the cumulative number of target systems affected at time *t* where *t* is measured from the time the vulnerability is disclosed. Let *P* be the height of the s-curve, or the maximum number of target systems in the population affected by the vulnerability (referred to as penetration of the diffusion process). *D* is the time

when *P/2* systems are affected by the vulnerability (i.e. the s-curve reaches half of its ultimate penetration level) and captures the delay associated with the diffusion process. *R* is the slope of the s-curve and it is dependent on various factors such as the type of vulnerability and the complexity of developing exploits.

$$N(t) = \frac{\mathrm{P}}{1+e^{-(\mathrm{R}t-\mathrm{D})}} \quad (1)$$

We use non-linear least squares to estimate (1) with *P*, *R* and *D* as linear functions of our focal (*Immediate*) and other control variables described earlier.

Analyzing the Risk of First Attack

We use the Cox proportional hazard model to examine the risk of first attack from a vulnerability. A hazard model explains the first exploitation attempt of a vulnerability for a specific target firm. We constructed a data set that contains for each target firm and vulnerability combination, the day of first attempt to exploit the vulnerability (960 firms and 1201 vulnerabilities for a total of 1,152,406 observations). All vulnerabilities were aligned so that day 0 represented the date the vulnerability was reported to the reporting agencies or publicly disclosed. We incorporate our focal (*Immediate*) and control variables described earlier as explanatory covariates in the hazard model.

Volume of Attacks

We use a two-stage Heckman model to analyze the number of alerts generated by a vulnerability for a specific firm. Recall that our data set has for each firm (960 firms) and each vulnerability (1201 vulnerabilities), the number of alerts generated on each day of our research period. All vulnerabilities are aligned so that day 0 represents the day the vulnerability was first reported to the reporting agencies or disclosed publicly. Many vulnerabilities are never exploited in our alert data and ordinary least squares estimation will ignore the selection bias. The two-stage Heckman model allows us to incorporate selection bias in the volume of attacks. In the first stage, we use a selection model to investigate vulnerability attributes that affect overall likelihood of exploitation. In the second stage, we examine the number of alerts per day (with a natural log transformation). In this analysis, we control for all

vulnerability covariates and we include monthly fixed effects based on attack date to control for changes in attack behavior over time. We also include 960 firm fixed effect indicators to control for potential differences in a firm's inherent risk of attack.

RESULTS

Table 2 shows the results of the non-linear least squares estimation of (1). Based on the estimated parameters, we find that immediate disclosure reduces delay (*D*) of diffusion (accelerates the diffusion process) and slightly increases penetration (*P*) of attacks based on the vulnerability. To ease the interpretation of the estimated parameters, Figure 1 plots the s-curve for immediate and non-immediate disclosure vulnerabilities. The figure shows that while immediate disclosure significantly reduces delay of the diffusion process by approximately 12 days, it has a small effect on the penetration level. Thus, we find support for H1 and our results slightly disagree with H3.

Table 3 shows the results of the Cox proportional hazard model to analyze the risk of first attack from a vulnerability for a specific target firm. Model 0 provides the results with only the control variables included, while Model 1 includes our focal variable (*Immediate*). The results in Table 3 show that immediate disclosure significantly increases the risk of first attack by an estimated 49.7 %. Thus, our results support H2.

The results from our evaluation of H4 are reported in Table 4. The dependent variable is the number of attacks (log transformed) on a specific date for a specific client and for a specific vulnerability. Table 4 reports results from a two-stage Heckman selection model. The coefficient of the *Immediate* variable is negative and significant, indicating that immediate disclosure reduces the volume of attacks. However, based on the estimated parameter, immediate disclosure reduces volume of attacks by approximately 3.6%. Thus, we find only limited support for H4.

Although the effect size was small, our results indicate that immediate disclosure paradoxically increases the number of distinct firms attacked (increased penetration), but decreases the total number of

attack attempts. This may indicate a unique search pattern shaped by the exploitation race. Attackers may attempt a broad search to rapidly determine if countermeasures are in place. If countermeasures are found, then there is no utility for continued attempts within a firm and overall attack volume does not correspondingly increase with the increased penetration. This supports the conversion from broad untargeted reconnaissance activity to targeted attacks previously theorized (Ransbotham and Mitra 2009).

Interestingly, we also find that public availability of an attack signature accelerates the diffusion process, increases penetration of attacks, increases risk of first attack, and increases the volume of attacks, indicating that the signature contains information that the attacker can utilize to build tools and exploit the vulnerability. Some of the other variables in the models also provide interesting insights. For example, vulnerabilities that require complex execution methods (e.g. social engineering) have delayed diffusion processes and lower attack volumes.

SUMMARY AND IMPLICATIONS

Contrary to the dominant view in the security industry and the practitioner literature, we find that immediate disclosure of vulnerabilities reduces delay in the attack diffusion process (as expected), but also slightly increases penetration of attacks in the population of target systems and the volume of attacks. Our results can be explained by viewing the attack process as a race between attackers who attack systems and security service providers who develop countermeasures, similar to a patent race that has been examined in the economics literature (Denicolo 2000). This race accelerates the attack diffusion process, but also increases awareness, forces security service providers to be more vigilant, accelerates the deployment of countermeasures, and reduces the window of opportunity for attackers before countermeasures are installed.

Our results have two important implications for policy makers, security organizations such as CERT, and software vendors. First, limited public disclosure of vulnerability information may combine the benefits of non-public and immediate disclosure to skew the race towards securing systems. For example, organizations such as CERT can immediately disclose the vulnerability to trusted security

service providers (as well as the software vendor) so that they can develop countermeasures to protect systems for their clients until a patch is made available by the software vendor. This may provide an advantage to security service providers in the attack and countermeasures race without publicly disclosing the signature and other attack details. This limited disclosure to trusted security service providers is particularly important since our results indicate that public disclosure of signatures increases attack penetration and attack volume. Unfortunately, limiting disclosure is inherently difficult and, in the end, relies on obscurity to provide advantage to defenders.

Second, while immediate disclosure causes security service providers to be more vigilant and limits the volume of attacks based on the vulnerability, it is possible (and perhaps even likely) that the effect on those who are not protected through such services is in the opposite direction as attackers focus their attention on such targets in the absence of others. Also, a similar diversion-based argument applies to vulnerabilities not disclosed through immediate disclosure. In general, the attack and countermeasures race for immediate disclosure vulnerabilities may cause security service providers to adjust priorities and focus less on other (perhaps more critical) vulnerabilities.

It is important to note that our analysis focuses on exploitation attempts and we do not observe the costs associated with immediate or non-public disclosure. Immediate disclosure is likely to significantly increase costs to defenders because it requires urgent handling instead of routine processes. If all vulnerabilities were immediately disclosed, benefits from prioritization would likely diminish while defensive costs may increase. Overall, our analysis and results indicate that the effects of different disclosure methods are complex and nuanced, and represent a fruitful area of further research.

REFERENCES

Arora , A., J. P. Caulkins, et al. (2006). "Sell First, Fix Later: Impact of Patching on Software Quality." Management Science **52**(3): 465-471.

Arora , A., R. Telang, et al. (2008). "Optimal Policy for Software Vulnerability Disclosure." Management Science **54**(4): 642-656.

August, T. and T. I. Tunca (2006). "Network Software Security and User Incentives." Management Science **52**(11): 1703-1720.

August, T. and T. I. Tunca (2008). "Let the Pirates Patch? An Economic Analysis of Software Security Patch Restrictions." Information Systems Research **19**(1): 48-70.

Cavusoglu, H., H. Cavusoglu, et al. (2007). "Efficiency of Vulnerability Disclosure Mechanisms to Disseminate Vulnerability Knowledge." IEEE Transactions on Software Engineering **33**(3): 171-185.

Cavusoglu, H., H. Cavusoglu, et al. (2008). "Security Patch Management: Share the Burden or Share the Damage?" Management Science **54**(4): 657-670.

Denicolo, V. (2000). "Two-stage patent races and patent policy." RAND Journal of Economics **31**(3): 488-501.

NVD (2008). National Vulnerability Database.

Park, I., R. Sharman, et al. (2007). "Short Term and Total Life Impact analysis of email worms in computer systems " Decision Support Systems **43**: 827-841.

Ransbotham, S. and S. Mitra (2009). "Choice and Chance: A Conceptual Model of Paths to Information Security Compromise." Information Systems Research **20**(1): 121-139.

Ransbotham, S., S. Mitra, et al. (2011). "Are Markets for Vulnerabilities Effective?" MIS Quarterly **forthcoming**.

Rogers, E. M. (2003). Diffusion of Innovations. New York, NY, The Free Press.

Table 1: Sample Descriptive Statistics

		Immediate Disclosure Vulnerabilities		Non-Immediate Vulnerabilities	
Variable	**Value**	**Count**	**%**	**Count**	**%**
Complexity	Low	270	61.04%	347	51.87%
	Medium	194	23.26%	263	39.31%
	High	68	15.70%	59	8.82%
Confidentiality Impact	No	121	23.47%	157	23.47%
	Yes	411	76.53%	512	76.53%
Integrity Impact	No	104	13.95%	156	23.32%
	Yes	428	76.68%	513	76.68%
Availability Impact	No	106	19.77%	97	14.50%
	Yes	426	80.23%	572	85.50%
Vulnerability	Input	184	37.21%	206	30.79%
	Design	76	11.63%	111	16.59%
	Exception	44	6.40%	72	10.76%
Market Disclosure	No	441	82.89%	600	89.69%
	Yes	91	17.11%	69	10.31%
Server Application	No	513	96.43%	651	97.31%
	Yes	19	3.57%	18	2.69%
Contains Signature	No	466	87.59%	576	86.10%
	Yes	66	12.41%	93	13.90%
Patch Available	No	224	42.11%	320	47.83%
	Yes	308	57.89%	349	52.17%

Table 2: Diffusion of Vulnerability Exploit Attempts

	Model 0			Model 1		
Variable	**P**	**R**	**D**	**P**	**R**	**D**
Constant	72.921***	-0.045***	-23.822***	58.711***	-1.122***	76.100***
	(2.400)	(0.007)	(3.186)	(2.038)	(0.258)	(17.587)
Confidentiality Impact	-35.980***	-0.091***	71.715***	-32.475***	0.191***	135.880***
	(1.807)	(0.011)	(8.640)	(1.526)	(0.045)	(31.256)
Integrity Impact	-0.354	-0.015***	40.826***	11.739***	0.394***	91.899***
	(1.936)	(0.004)	(5.149)	(1.660)	(0.089)	(21.953)
Availability Impact	-10.909***	-0.147***	-40.211***	-11.125***	-0.776***	-156.507***
	(1.612)	(0.018)	(4.714)	(1.430)	(0.178)	(36.045)
Input Type	61.636***	-0.102***	89.354***	51.834***	0.504***	121.676***
	(1.303)	(0.012)	(10.635)	(1.169)	(0.115)	(28.107)
Design Type	-25.785***	-0.047***	-1.596***	-24.477***	-0.339***	9.165***
	(1.947)	(0.006)	(0.228)	(1.714)	(0.078)	(2.507)
Exception Type	22.260***	-0.608***	189.362***	-43.074***	-1.567***	27.602***
	(3.442)	(0.073)	(23.084)	(2.246)	(0.359)	(6.871)
Medium Complexity (*Med*)	207.046***	-0.060***	72.532***	174.273***	0.573***	136.684***
	(5.404)	(0.008)	(8.523)	(4.497)	(0.132)	(31.015)
High Complexity (*High*)	45.598***	-0.002	10.702***	42.092***	0.573***	20.652***
	(1.503)	(0.002)	(1.266)	(1.456)	(0.022)	(4.683)
Market Disclosure (*Market)*	-78.618***	-0.740***	240.813***	-57.462***	-1.151***	278.744***
	(2.371)	(0.087)	(28.363)	(1.683)	(0.263)	(63.943)
Server Application (Server*)*	13.605***	-1.311***	466.265***	-3.054*	-0.104***	27.296***
	(2.373)	(0.154)	(54.696)	(1.345)	(0.024)	(6.349)
Signature Available (*Sig)*	124.750***	0.300***	-47.806***	123.242***	1.415***	-141.577***
	(2.272)	(0.036)	(5.998)	(2.126)	(0.324)	(32.944)
Patch Available (*Patch)*	-22.575***	0.104***	-98.445***	-19.941***	-0.597***	-140.865***
	(1.063)	(0.013)	(11.822)	(0.936)	(0.136)	(32.694)
Immediate Disclosure (*ImmDisc*)				3.686***	-0.094***	-5.765**
				(1.040)	(0.021)	(1.830)
R^2			31.66			29.47

132,768 daily observations of 333 vulnerabilities from 2006-2007

Robust (HC3) standard errors in parentheses; significance: * p<.05; ** p<.01; *** p<.001

Nonlinear regression on number of firms affected, $N(t) = \frac{P}{1+e^{-(Rt-D)}}$ where the cumulative penetration (P), the rate of diffusion (R) and delay (D) are linear functions of the variables shown in the table.

Table 3: Risk of Exploitation of Vulnerabilities

Variable	Model 0	Model 1
Confidentiality Impact	-0.135*** (0.024)	-0.165*** (0.024)
Integrity Impact	0.288*** (0.026)	0.298*** (0.026)
Availability Impact	0.296*** (0.024)	0.339*** (0.024)
Input Type*)*	0.302*** (0.018)	0.289*** (0.018)
Design Type	-0.388*** (0.028)	-0.359*** (0.028)
Exception Type	-0.093** (0.030)	-0.108*** (0.030)
Medium Complexity (*Med*)	-0.215*** (0.021)	-0.188*** (0.021)
High Complexity (*High*)	0.227*** (0.020)	0.227*** (0.020)
Market Disclosure (*Market*)	-1.508*** (0.043)	-1.594*** (0.043)
Server Application (*Server)*	-0.620*** (0.073)	-0.658*** (0.074)
Signature Available (*Sig*)	1.034*** (0.018)	1.075*** (0.018)
Patch Available (*Patch)*	0.009 (0.016)	-0.001 (0.016)
Immediate Disclosure (*ImmDisc)*		0.497*** (0.016)
Log likelihood	-111736.2	-111225.21
Wald χ^2	8436.90***	8504.00***

Table 4: Volume of Alerts per Client Firm per Vulnerability

Variable	Model 0		Model 1	
Constant	0.430***	(0.082)	0.465***	(0.082)
Confidentiality Impact	0.037***	(0.003)	0.031***	(0.003)
Integrity Impact	-0.076***	(0.004)	-0.083***	(0.004)
Availability Impact	-0.003	(0.003)	-0.005	(0.003)
Input Type	0.145***	(0.002)	0.136***	(0.002)
Design Type	-0.089***	(0.003)	-0.089***	(0.003)
Exception Type	-0.132***	(0.004)	-0.128***	(0.004)
Age (ln)	-0.210***	(0.002)	-0.210***	(0.002)
Medium Complexity (*Med*)	-0.042***	(0.003)	-0.050***	(0.003)
High Complexity (*High*)	-0.036***	(0.003)	-0.037***	(0.003)
Market Disclosure (*Market*)	-0.101***	(0.003)	-0.098***	(0.003)
Server Application (*Server*)	0.132***	(0.007)	0.130***	(0.007)
Signature Available (*Sig*)	0.170***	(0.003)	0.166***	(0.003)
Patch Available *(Patch)*	-0.024***	(0.002)	-0.019***	(0.002)
Attack Month fixed effects	Included		Included	
Firm fixed effects	Included		Included	
Immediate Disclosure *(Immediate)*			-0.034***	(0.002)
Inverse Mills	-0.0812***	(0.004)	-0.095***	(0.004)
Constant	0.263***	(0.008)	0.329***	(0.008)
Confidentiality Impact (*I_conf*)	0.024***	(0.004)	0.015***	(0.004)
Integrity Impact (*I_integ*)	0.503***	(0.004)	0.501***	(0.004)
Availability Impact (*I_avail*)	-0.246***	(0.004)	-0.253***	(0.004)
Input Type (*T_input)*	0.146***	(0.003)	0.138***	(0.003)
Design Type (*T_design*)	-0.195***	(0.004)	-0.197***	(0.004)
Exception Type (*T_exception*)	0.569***	(0.006)	0.572***	(0.006)
Medium Complexity (*Med*)	0.111***	(0.003)	0.100***	(0.003)
High Complexity (*High*)	0.278***	(0.004)	0.280***	(0.004)
Market Disclosure (*Market*)	-0.062***	(0.004)	-0.050***	(0.004)
Server application *(Server)*	-0.331***	(0.008)	-0.325***	(0.008)
Signature Available (*Sig*)	0.739***	(0.004)	0.738***	(0.004)
Patch Available (*Patch)*	-0.438***	(0.003)	-0.432**	(0.003)
Immediate Disclosure (*Immediate)*			-0.067***	(0.003)
Publication Month fixed effects	Included		Included	
Wald χ^2	2.16e+06***		2.16e+06***	

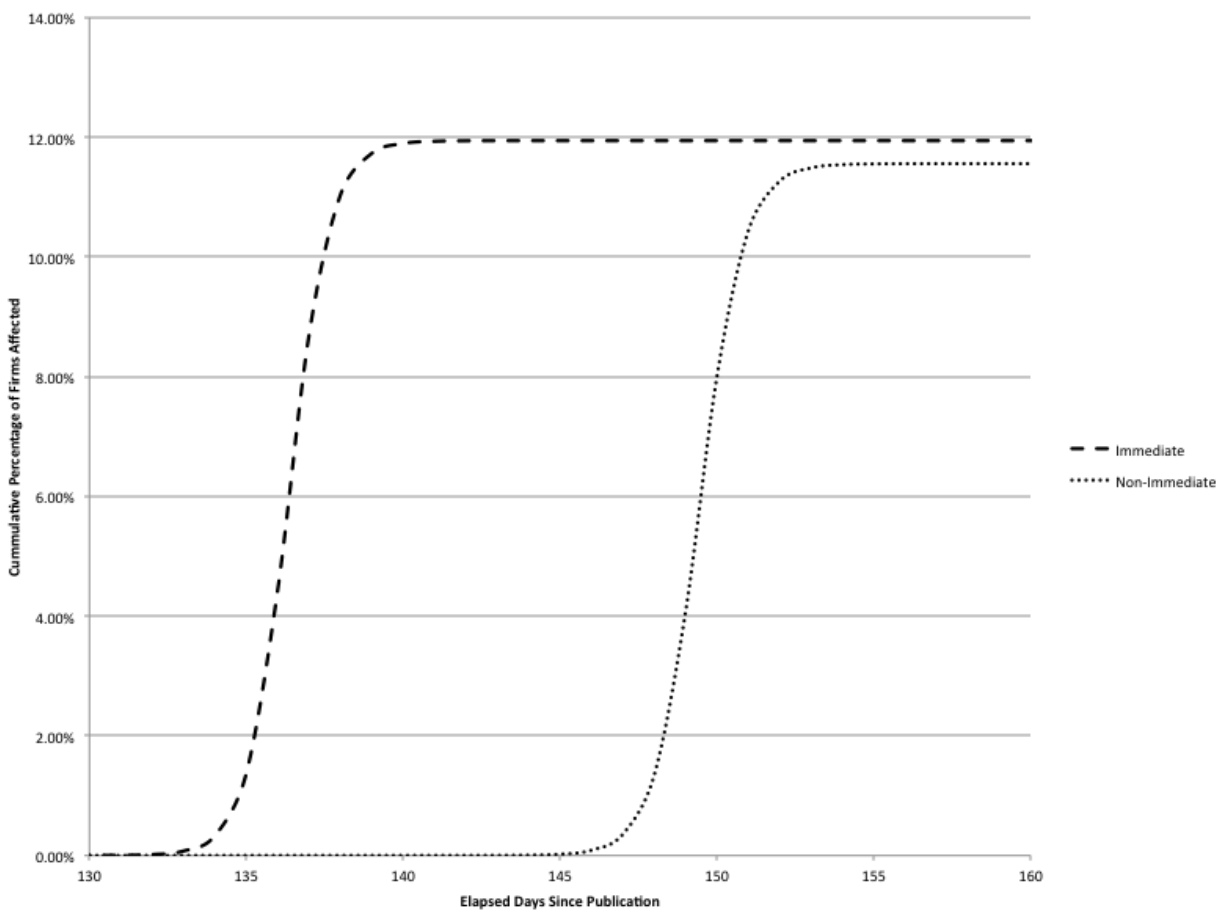

Figure 1: The Diffusion of Immediate and Non-Immediate Vulnerabilities