Toward Econometric Models of the Security Risk from Remote Attacks

Security risk models have successfully estimated the likelihood of attack for simple security threats such as burglary and auto theft. Before we can forecast the risks to computer systems, we must first learn to measure the strength of their security.

STUART E.
SCHECHTER
*MIT Lincoln
Laboratory*

Those working in the computer security field work to both reduce the chance that our systems will be breached and to reduce the damage that would occur should such a breach occur. In the risk management field, risk is generally defined as a measure of the expected loss due to a potential security risk event.

> risk = (likelihood of loss event) × (cost of loss event)

We can rewrite the equation more specifically to the study of security risks by noting that the loss events of interest are security breaches that result when an adversary succeeds in violating a desired security property of our systems.

> security risk = (likelihood of security breach) × (cost of security breach).

Because multiple security breaches can occur, it's sensible to define security risk in terms of the frequency, or expected rate, of breaches.

> security risk = (security breach rate) × (average cost per breach).

While firms already have processes in place to estimate the costs of loss events, forecasting future breach rates poses a far greater challenge. In this article, I'll show where regression models have succeeded at predicting breach rates for other security threats and describe the obstacles that must be overcome before we can apply these models to computer systems.

Forecasting breach rates

As we can't know the actual breach rate in advance, this rate must be forecast using a predictive model. We can do this by modeling the breach rate as a dependent variable that varies as a function of certain independent variables that, unlike the breach rate, can be measured at the time of forecast. We must thus find variables that not only have historically affected the rate at which breaches occur, but also for which we can reasonably expect will have the same effect on the breach rate in the future. For example, given a dependent variable Y (breach rate) that appears to change linearly with the value of independent variables X_1 and X_2, we could create a model to estimate Y with a value $\hat{Y}$, which is defined via a function f:

$$\hat{Y} = f(\mathbf{X}) = f(X_1, X_2) = \beta_0 + \beta_1 X_1 + \beta_2 X_2. \qquad (1)$$

For a given data point i, u_i is an error term that represents the difference between the estimate $\hat{Y}_i$ and the actual measured value of Y_i:

$$Y_i = \hat{Y}_i + u_i = f(\mathbf{X}) + u_i. \qquad (2)$$

A *regression* is the process of setting the constants' values, represented by the b values in Equation 1, to minimize the error terms calculated from historical observations. In other words, a regression fits the function so that it would best match observed historical values (other texts provide a more detailed introduction to regression analysis[1]). Once those constants are computed and fixed within $f(\mathbf{X})$, we can use the function to model the effect,

on Y, of changes to or more of the independent variables (X). However, the relationships between these independent and dependent variables aren't guaranteed to remain stationary over time. Because an adversary's best strategy is often to examine our assumptions and actively work to defy our predictions, finding an $f(\mathbf{X})$ that remains an accurate forecast of Y despite these efforts makes security models significantly more difficult to create than models of natural phenomena.

Factors affecting attack and breach rates

Keeping in mind the complications caused by an adversary's presence, we can classify into four categories independent variables that are likely to affect the rate at which a system is attacked—and the rate at which security will be successfully breached.

- *Adversaries' ranks.* The number of potential adversaries is likely to be positively correlated with the rate of security breaches and the resulting security risk. Certain facts and variables influence the number of individuals with the means, motive, and opportunity to attack a system. For example, if the cooperation of an insider is required for the class of breach being studied, the set of potential adversaries is likely to be smaller than it would be otherwise.
- *Adversaries' incentives.* Other variables indicate how valuable an attack appears to be to potential adversaries. The greater the incentive to attack, the more likely it is that a potential adversary will choose to do so. Thus, the attack incentive is also positively correlated with attack frequency, breach rate, and security risk.
- *Adversaries' attack risk.* The presence of factors that discourage attacks are expected to be negatively correlated with the frequency of attempted breaches and thus also negatively correlated with successful breaches and overall security risk. The risk of undesirable consequences to the adversary, as a result of attacking a system, could include such factors as capture and incarceration, disclosure of attack tools or techniques, damage to reputation, or even physical harm.
- *Adversaries' cost of attack.* The other disincentive to attack is the collective cost of equipment, effort, and other resources required to stage a successful attack. Like attack risk, it is negatively correlated with attack frequency and security risk. The stronger the system, the greater the attacker's expected cost. Measuring how strong a system is can thus help us better estimate security risks it faces.

From this adversary-centric perspective, the introduction of security measures is a process that reduces the number of potential adversaries, reduces the value that can be obtained through successful attacks, increases the risks incurred by adversaries during attack, and raises other attack costs.

Modeling security breach rates

Although regression models have yet to be successfully applied to forecasting risks in computer security, they've proved successful in other security domains. For example, analysts have collected and regressed statistical data to construct models that predict the likelihood of burglary for homes with different traits and safeguards.[2–4]

Yochanan Shachmurove and colleagues' survey, which used data collected from the town of Greenwich, Connecticut, measured factors from four categories of independent variables.[2] One such variable represented the home's value, which would affect the amount of loot a thief could expect to find inside. Because Greenwich is close to the cities of New York and Stamford, the distance between a home and the nearest highway entrance could also affect the number of potential adversaries who might have discovered the house when scoping out a neighborhood. Highway proximity simultaneously provided an indicator of the number of potential adversaries and the attack risk, as a close highway entrance eases escape and reduces risk.

Variables that indicated increased risk of capture or harm to the thief (attack risk) included alarms as well as more subtle indicators, such as the continual presence of a car parked in the driveway. The one security measure that didn't have a statistically significant effect was an indicator of increased security strength, as the presence or absence of a dead bolt on the door. While it's surprising that attempts to increase security strength didn't deter burglars, the study is consistent with the findings of Richard Wright and Scott Decker, who surveyed 108 burglars in the St. Louis, Missouri area.[5] They confirmed that alarms were a strong deterrent (with 56 of 86 burglars reporting that they would never target a house they knew to have an

> **Although regression models have yet to be successfully applied to forecasting risks in computer security, they've proved successful in other security domains.**

alarm[5]), and that burglars are difficult to deter once they've already trespassed onto a property. When confronted with dead bolts, the burglars in their study said they would break the bolt or enter the house through a window, rather

than retreating and finding a new home to break into. In the words of one of the burglars interviewed, "As long as houses are made of wood and glass, I can get 'em."[5] In other words, no house with wood doors or glass windows

Network attackers prefer staging an attack anonymously to staging an attack that is more likely to lead to prosecution.

could be made strong enough to prevent burglars from entering. Deterrents that increase the risk that thieves will be caught are more likely to be effective.

We can use similar statistics to forecast the risk of automobile theft. New defensive and theft strategies have evolved, albeit so slowly that we can forecast theft rates using information such as the vehicle's year, model, and location.

Given regression models' potential for measuring security risk in other domains, it's not surprising that such studies have been proposed to study computer fraud[6] and insider attacks.[7] However, models that can forecast computer security risks have not been forthcoming.

One reason breach forecasting models have remained elusive is that computer systems are far more complex and heterogeneous than homes. While homes are built of a small set of common materials and serve a common function, software can be written using different algorithms coded in different languages to serve various purposes. Because different software packages perform dramatically different functions for their users, their architectures are varied, as are the attack classes to which they're vulnerable. As software rapidly evolves, and as this evolution affects the choices made by the adversaries who would attack it, the threats posed to computer systems do not exhibit the stationarity found in burglary.

The other reason these models have yet to be realized is that accurate or meaningful metrics for gauging security strength is lacking. Security strength is an important deterrent when protecting systems connected to global networks, such as the Internet, because a skilled criminal could launch an attack with little risk of being identified or of facing retribution. Whereas once criminals could flee to foreign countries after committing a crime, a network lets adversaries seek refuge first and then attack from a safe distance or a foreign jurisdiction.

Network attackers prefer staging an attack anonymously to staging an attack that is more likely to lead to prosecution. To do this, they can hide their identities by routing their communications through a sequence of dis-

tant systems—the longer the trail, the harder it is to trace its source. If each step in the route lies in a different jurisdiction, and the trail can only be unraveled one step at a time, tracking the adversary requires gaining each jurisdiction's cooperation. If enough steps lie between the adversary and the target, tracking the attack source is likely to take far more time than the actual attack. Unless all compromised parties keep meticulous logs, the trail will disappear before it can be traced.

Keeping in mind that many attacks on information systems can be committed from a safe distance, the risk of capture or harm is far less a deterrent for information criminals than it is for burglars. Should tools for anonymous communications and financial transactions become available to information criminals, the risk of monetizing their crimes (such as withdrawing a modified bank balance) will also decrease.

When an adversary can attack with impunity, the time, effort, and other resources required to break into a system, which for other crimes can be overshadowed by the risk of being caught, become significant factors in the adversary's choice of target. The stronger the system, the more resources the adversary must expend to achieve his or her goals.

While the means with which attacks are performed and safeguards are constructed could continue to evolve at a rapid pace, it's reasonable to expect that, other things being equal, the breach rate will increase along with the number of potential adversaries. We can also reasonably expect that adversaries will continue to choose targets that are easier to attack (weaker), less likely to result in personal harm or capture, and have greater loot than the alternatives.

Measuring security strength

For any given threat, a system is only as strong as it is difficult for an adversary to succeed in attacking it. Safeguards traditionally used to prevent certain attack classes, such as cryptosystems, have long been judged by how difficult they were to circumvent. Security strength metrics exist to quantify the time, effort, and other resources required to bypass a system's safeguards to perform the attack.

Because security strength metrics gauge security from an adversary's perspective, they complement security risk metrics, which measure security from the defender's perspective. The measure of resources needed to breach a system is fundamentally an economic one.

Several researchers have used formal methods of computer science, such as information theory, to address questions of security strength. For example, Claude Shannon used information theory to prove a one-time pad cipher was secure against certain attack classes regardless of the attacker's available resources.[8,9] Adi Shamir used similar techniques to prove his secret-sharing scheme's security properties.[10] However, applicability of computational approaches to security strength is extremely limited. Even

today's best public-key cryptosystems are only as secure as it is difficult to solve certain algorithmic problems. Given the lack of progress in proving security properties for highly constrained problems, it's unlikely formal proofs could be applied to measure the strength of large and complex networked software systems.

Although software's complexity hinders security strength measurement, it has other properties that aid in it. The ease with which we can create verifiably exact replicas of software ensures that security strength measurement costs can be amortized over all copies. We can even test identical copies for vulnerabilities in parallel. The ease with which end users can compare digital data lets them verify for themselves that their software is that which was demonstrated to have a measured security strength.

The practice of gauging a software system's strength has long revolved around the search for vulnerabilities. Software tools designed to attack systems by exploiting vulnerabilities often appear soon after vulnerabilities are publicized, and thus the strength of systems with known vulnerabilities is almost always too small to be worth measuring. Rather, the measurements of interest are those for which the system under analysis has no known vulnerabilities, or at least none for which all known vulnerabilities require inordinate resources to exploit.

Cost of finding vulnerabilities

To breach a system with no known vulnerabilities, attackers must find a new vulnerability and develop a technique to exploit it. A software system's strength is thus dominated by the cost to find a vulnerability and create an exploit. Because primitive exploits are often required to demonstrate that a vulnerability exists, these costs, considered together, are the cost of finding a vulnerability. (It's possible for the cost of exploiting a vulnerability to exceed the cost of discovering it—an extreme example is a backdoor guarded by public-key cryptography; that is, a vulnerability intentionally placed in the code to grant access to anyone who knows a secret key. Although such a vulnerability might be easy to discover, the secret key could be impossible to find because it doesn't need to be included in the code itself. Writing an exploit without knowing the secret key requires attackers to break the public-key cryptosystem.)

Measuring security strength is difficult because it's a function of an adversary's costs, not the defender's. It's impossible to research the skill of every potential adversary, the value they place on their own time, and the other resource costs they must expend to find vulnerabilities in a software program. What's more, adversaries themselves aren't likely to know how much time and resources they must expend to accomplish their goals.

Assume that finding a vulnerability is an essential step in breaching a system. Adversaries can perform cer-

tain tasks to look for vulnerabilities, from inspecting code to writing and executing tests. To maximize their productivity, they start with the tasks that have the greatest chance of success per unit cost (those that are expected to be the most profitable). The chance of finding a vulnerability increases with total investment (the first derivative is positive), but the chance of success for each additional dollar invested is smaller than for the previous dollar (the second derivative is negative, resulting in diminishing returns).

An economically rational individual will only perform tasks as long as the expected return is greater than the expected cost. If the individual believes a vulnerability is worth r dollars, the cost of a task i is c_i, and the probability that task i will result in the discovery of a vulnerability is p_i, then he or she will perform the task only when $c_i \leq p_i \cdot r$, or equivalently when

$$\frac{c_i}{p_i} \leq r.$$

In fact, while a risk-neutral individual will continue to perform tasks when

$$\frac{c_i}{p_i} < r$$

and be indifferent to performing tasks when

$$\frac{c_i}{p_i} = r,$$

a risk-averse adversary won't search for vulnerabilities when

$$\frac{c_i}{p_i} > r - \in$$

for some positive measure of risk aversion $\in$.

There is no reason to believe that, given this uncertainty, individuals will expend the same amount of resources to find a vulnerability as they would if they knew the cost beforehand. Although expectations regarding each task's profitability can be updated between tasks (c_i and p_i

Adversaries intent on breaching a computer system might search for vulnerabilities themselves or pay someone else to do it.

don't need to be calculated until task $i - 1$ is complete) updating these estimates doesn't guarantee that they'll accurately measure the true probability of finding a vulnerability. It's the perceived expectation of the cost—not the true

cost—that determines how many resources adversaries are willing to spend to find vulnerabilities in a system. Thus, a metric of security strength that incorporates cost perceptions might be more valid than one that includes only true costs. The perceived cost to find a vulnerability, or perceived *cost-to-break*,[11] is a metric of security strength that can be measured by determining the price that someone would have to pay to acquire a vulnerability.

Adversaries intent on breaching a computer system might search for vulnerabilities themselves or pay someone else to do it. If we assume the worst, then the individual with the lowest cost of finding and exploiting a vulnerability and the individual with the most to gain from this knowledge are one and the same. Equivalently, they might be different individuals who contract to exchange the vulnerability information for a price.

Market price

The market price to find and report a vulnerability, or MPV, estimates the cost of finding a vulnerability in a software system as measured by a market in which anyone, including those we consider our adversaries, can participate. The *MPV bid price* is the highest price offered to buy a previously unreported vulnerability, whereas the *MPV ask price* is the lowest price a seller demands in order to report a vulnerability. Transactions occur when the bid price and the ask price meet. At all other times, the MPV falls within the range of the bid price (the lower bound) and the ask price (the upper bound).

The reward a firm offers to find vulnerabilities in its product, which constitutes the bid price for reporting a vulnerability, can be used as a metric of security strength in models that forecast security risk. However, this value is a lower bound and could differ from market perceptions of the true price one would pay to obtain the next vulnerability. Once that vulnerability is known, it can be repaired, and will no longer be of interest. Our interest thus always lies with the market price for the next vulnerability discovered. The quantity we'd want to measure precisely is thus always one transaction in the future, with the transaction's price and time are always one step away.

To more accurately gauge these values, regression models could be built using data generated by vulnerability reporting markets. We could estimate MPV by modeling it as a function of the current bid price, the price at which previous vulnerabilities were reported, the amount of time that has passed since each was reported, and any other factors of import. We could use the same approach to forecast the probability that a new vulnerability will be reported for a given price within a specified time period.

Given the impunity with which remote criminals can probe our software and networks, the security of our systems against remote attacks rests on how difficult we can make it for our adversaries to perform their attacks. This is a question of security strength. Because vulnerabilities and safeguards in networked software systems aren't stationary, historical security risk data that doesn't include measures of security strength is of little value for forecasting future risk. Unfortunately, today's software products are often released without any measure of their strength. Until they are, we won't be able to create accurate security risk models for remote network attacks. □

Acknowledgments

References

1. J.H. Stock and M.W. Watson, *Introduction to Econometrics*, Pearson Education, 2003.
2. Y. Shachmurove, G. Fishman, and S. Hakim, *The Burglar as a Rational Economic Agent*, tech. report CARESS working paper 9707, Ctr. for Analytic Research in Economics and the Social Sciences, Univ. of Pennsylvania, June 1997.
3. S. Hakim, G.F. Rengert, and Y. Shachmurove, *Knowing Your Odds: Home Burglary and the Odds Ratio*, tech. report CARESS Working Paper 0014, Ctr. for Analytic Research in Economics and the Social Sciences, Univ. of Pennsylvania, Sept. 2000.
4. T. Budd, *Burglary of Domestic Dwellings: Findings from the British Crime Survey*, tech. report, UK Home Office Crime Reduction Programme Unit, 1999.
5. R.T. Wright and S.H. Decker, *Burglars on the Job: Streetlife and Residential Break-Ins*, Northeastern Univ. Press, 1994, p. 25.
6. L.C.J. Mercer, "Fraud Detection via Regression Analysis," *Computers & Security*, vol. 9, no. 4, 1990, pp. 331–338.
7. C. Shannon, "A Mathematical Theory of Communication," *Bell System Tech. J.*, vol. 27, July 1948, pp. 398–403.
8. C. Shannon, "A Mathematical Theory of Communication," *Bell System Tech. J.*, vol. 27, Oct. 1948, pp. 623–656.
9. A. Shamir, "How to Share a Secret," *Comm. ACM*, vol. 22, no. 11, 1979, pp. 612–613.
10. S.E. Schechter, "Quantitatively Differentiating System Security," *First Workshop on Economics and Information Security*, 2002; www.sims.berkeley.edu/resources/affiliates/workshops/econsecurity/econws/31.pdf.

***Stuart E. Schechter** is a member of the Information Assurance group at MIT Lincoln Laboratory. He received a PhD in computer science from Harvard University. Although his dissertation explored economic approaches to measuring and improving the security of software and computing systems, Schechter is also active in the development and testing of new technologies to protect networks from malicious code. Contact him at ses@ll.mit.edu.*