Firewalls and Beyond: Engineering IT Security

Dennis Clancey
Jeffrey Glick
Daniel Kang
United States Military Academy
West Point, NY

Advisor: Elizabeth W. Schott

Abstract

Problem

A new university requires defensive measures to protect its network from unauthorized access, alteration of data, and unavailability. Without implementing defensive measures, the university is exposed to an expected loss of $8.9 million per year. Rite-On Consulting Firm has been tasked to conduct a risk analysis of information technology security for the university and to propose a model that minimizes costs while maintaining the highest possible level of security. This analysis addresses emerging technologies as an implied task.

Considerations

Our model stresses flexibility and simplicity. The model is run in Microsoft Excel, common software. Any company can cheaply tailor this powerful model to its individual needs. It can easily be updated to accommodate new tools and policies that reduce an organization's risk.

Results

Our model optimizes the mix of security tools and procedures. For the network-based measures, the new university should use the Network Defense Firewall, Enterprise Inoculation anti-virus program, Network Eye IDS, and a strong password policy. Additionally, the university should disallow wireless connections, have unmonitored personal use, and require user training.

The UMAP Journal 25 (2) (2004) 143–156.

The host-based decisions are divided into three subnetworks.

- The first subnetwork (admissions office, registrar, and health center) should use the Lava firewall, Bug Killer Anti-virus, and Robust Solutions service redundancy.
- Both the second subnetwork (academic departments and dormitories) and the third subnetwork (athletic department and bookstore) should use Intelliscan firewall, Bug Killer AV, Sonic Data data redundancy, and Web King SR.

Conclusions

The model provides the optimal balance of security and risk, based on associated costs. By simply altering the relative importance of security to each network resource, our model can recalculate an optimal solution with three clicks of a button. We are confident that the model for determining the optimal set of security tools and policies will greatly enhance the profitability of the new university for which it is designed. Our procedure and methodology could be used by other universities, businesses, and organizations trying to establish an optimal level of security in an information network.

Introduction

The creation of a new university requires the development of an information technology network with defensive measures protecting the university's assets from unauthorized access, alteration of data, and availability. The new university is expected to lose \$8.9 million per year if no effective defensive measures are implemented. However, each defensive measure is extremely costly, and designing an affordable and effective defense requires careful analysis of the costs and benefits of various combinations of defensive measures.

We develop a model to minimize the costs and maximize the benefits in creating a secure network. The model assumes a law of diminishing return with every additional defensive measures.

Using a Monte Carlo simulation, the development of the model requires several critical assumptions. We ran 500 iterations of the simulation to find the optimal combination of defensive mechanisms.

The model reveals that the optimal suite of defensive measures costs \$1.4 million and is expected to lower expected losses to \$1.7 million, for a net savings of \$5.9 million.

Problem Assumptions

- **All policies are network-wide.** For example, if we decide on a strong password policy, all resources on the network will be in accordance with that

policy. Different policies for different departments are not allowed.

- Likewise, **network-based security measures (tools) are employed across the entire network.** If a particular type of network-based firewall is chosen, it is used to protect the entire network.
- Moreover, **each type of network-based tool can be chosen only once.** That means only one option for firewall can be used (and it can only be used once). Vertically stacking identical security measures at a network level produces no added benefit.
- **Normally distributed observations:** The performance data of each tool will follow a normal distribution if additional observations are taken. This was the basis for our creation of iterations; these iterations of independently performing tools was the basis of our Monte Carlo approach.
- **Sub-networks:** The network is additionally divided into three subnetworks, and we assume that each asset on a particular subnetwork has similar vulnerabilities. This assumption simplifies the use of host-based tools while making it easier for administrators to control uniform defensive measures.
- **Combinations:** A combination of tools that cover the same defensive measure is not allowed. For instance, two different firewalls cannot be employed at the same time. This is a model simplification that recognizes that the benefits of similar tools will do little to improve the systems when used together.

Problem Approach

We develop a model that uses marginal-benefit/marginal-cost analysis and considers both the cost of defensive measures and the opportunity cost associated with assumed risks. We create and implement a four-step method to develop the model: Network Infrastructure, Data Analysis, Risk Analysis, and Cost Analysis.

Network Infrastructure

The network infrastructure depends on the number and function of the computers within each department. This breakdown of computers by department was founded on both given information and estimates:

Departments are grouped into subnetworks based on similar functions and security needs. The network topology (**Figure 1**) creates constraints for the implementation of defensive measures. All hosts in each subnetwork must assume identical defensive measures. The model allows each subnetwork to select an optimal array of defensive measures best suited to its hosts.

Table 1.
Breakdown of computers by department.

Department	Computers
10 Academic Departments	1,230
Dormitory Complex	15,000
Department of Intercollegiate Athletics	30
Bookstore	15
Admissions Office	40
Registrar's Office	35
Health Center	35
TOTAL	16,385

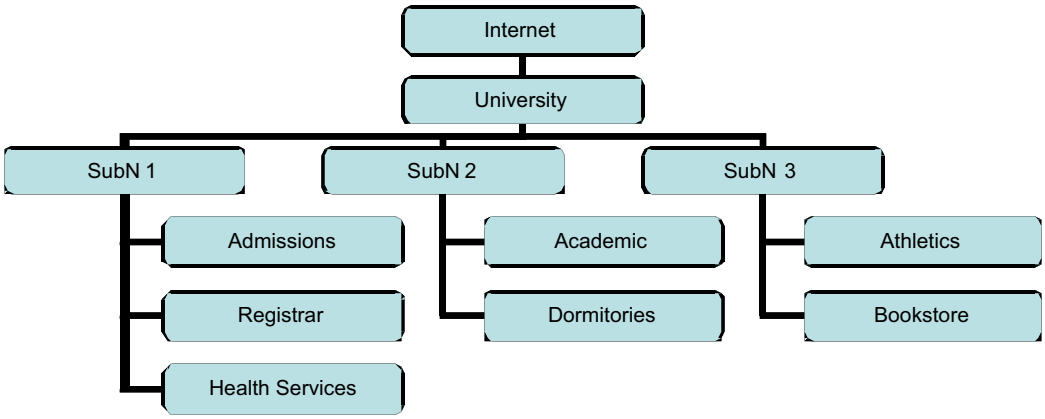

Figure 1. Proposed university network topology.

Data Analysis

Every tool and policy has associated costs and benefits. The direct costs come in the form of procurement costs, maintenance costs, and training costs. The benefits are measured by the degree to which a tool can improve (or detract from) user productivity, confidentiality, integrity, and availability. An improvement results in reduction in opportunity cost. For instance, if a particular tool improves confidentiality by 9%, then the opportunity costs associated with confidentiality will be reduced by 9%.

Quantitative information was provided in the problem statement enclosures for each piece of data: upper bound value, lower bound value, mean value, and variability level (concentration of the data about the mean).

Not knowing the standard deviation, the number of data observations, and the exact distribution, we simulate values, using Crystal Ball (a spreadsheet add-in with random-number generator capabilities [Decisioneering 2004]) and taking into account the possible range, the mean, and the variability. The Central Limit Theorem implies that if the number of observations is sufficiently

large, then both their sum and their mean have approximately normal distributions, even when individual variables themselves are not normally distributed [Devore 2000].

We also consider issues relating to the spread of the data (distance between the minimum and maximum measured values). Extreme levels of variability do not necessarily follow the normal distribution; in cases of high variability, the distributions are likely to be flatter ("fatter in the tails") than the normal distribution. In cases of low variability, the curves will be more sharply peaked than the normal distribution.

The function `CB.Normal`(μ,σ,min,max) in Crystal Ball returns a value from a truncated normal distribution with mean μ and standard deviation σ and minimum and maximum values as specified.

To estimate the standard deviations, we divide the range ($\max - \min$) by a specified factor depending on the level of variability. We wanted nearly all of the spread to be covered by three standard deviations. We settled on the values in **Table 2**.

Table 2.
Estimation of standard deviation.

Variability	Typical	Estimate of st'd dev.
high	0.32	range/6
medium	0.20	range/5
low	0.10	range/4

We were concerned about the accuracy of the simulated data in instances of an asymmetrical distribution (e.g., min = 0.05, mean = 0.17, max = 0.20). Crystal Ball creates a normal distribution with the inputted mean and standard deviation and then truncates it at the upper and lower boundaries. The mean of the resulting distribution can differ from the intended mean, as we confirmed from trial simulations After all considerations, we designed a spreadsheet that would generate actual values for all relevant costs and factors, taking into account levels of variability and ranges of values.

Risk Analysis

Table 1 of the problem statement quantifies the opportunity costs in dollars for various risks and apportions the risk to the categories of confidentiality, integrity, and availability. The university projects a total opportunity cost of $8.93 million if a network is built without defensive measures.

The next step in the risk analysis process involves the calculation of a subjective vulnerability score for each department. Vulnerability "is a weakness in the security system, for example, in procedures, design, or implementation, that might be exploited to cause loss or harm ... a particular system may be

vulnerable to unauthorized data manipulation because the system does not verify a user's identity before allowing data access" [Pfleeger and Pfleeger 2003, 6]. A vulnerability differs from a threat, which is a "set of circumstances that has the potential to cause loss or harm" [Pfleeger and Pfleeger 2003, 6]. We use a threat/vulnerability work table to quantify each risk based on a 1–9 scale, thereby allowing each asset and risk category to be prioritized based on a summed value of the vulnerability scores. The priority system allows the model to focus control measures on risks that have the greatest impact (highest opportunity cost) and highest probability of affecting the asset. **Table 3** shows the assigned vulnerability scores.

Table 3.
Vulnerability work table.

		Impact		
		Low	Med	High
Probability	High	3	6	9
	Med	2	5	8
	Low	1	4	7

The table breaks vulnerability into two factors, probability and impact. Probability refers to the likelihood of the threat occurring, while the impact is the cost associated with a manifestation of that actual threat. A category with low probability and high impact is something that doesn't occur very often, but if it does happen, could be fairly costly. Something with high probability and low impact could happen all the time but the costs would be minimal.

Another worksheet, entitled "Risk Analysis Vulnerability Weighting System," allows the person conducting the risk assessment to give each department a vulnerability score.

Cost Analysis

Cost analysis creates a relationship between the opportunity cost associated with assuming risks and the cost of implementing defensive measures. Our model calls for a cost-benefit optimization. The sum of all these costs (in dollars) that the university is still exposed to in the form of risk (given a particular security combination) is represented by C_R.

The second main category of costs is the total cost C_T of security tools, which includes all aspects of security (training costs, tools, policies implementation, etc.).

The sum of the two main categories of cost is the total expected expenditure on security related matters, $E(TC_S)$:

$$E(TC_S) = C_T + C_R.$$

The total cost C_T is the sum of each tool cost, multiplied by the quantity:

$$C_T = \sum(\text{amt}_T \times \text{cost}_T).$$

For network-based security measures, the amount of the tool is always assumed to be 1. On the contrary, many host-based measures have multiple costs (per computer or per network).

The risk cost C_R has three components: confidentiality, integrity, and availability. The implementation of each tool leads to a corresponding change in opportunity cost associated with each component. The specific opportunity costs that make up C_R (e.g., litigation, service reconstruction, consumer confidence, etc.) are not necessarily important. However, the model is concerned with the degree to which a particular security measure changes opportunity costs in terms of confidentiality, integrity, and availability. Thus, C_R can be broken down as

$$C_R = C_{R_c} + C_{R_i} + C_{R_a}.$$

The subcosts that make up C_R depend on two pieces of information:

- the total original cost of each component in the absence of security measures (T_oC_c, T_oC_i, T_oC_a); and
- the degree to which that original value is decreased, the ξ-factor.

So we have

$$C_{R_c} = T_oC_c - \xi_c.$$

The complexity of this model is increased when you consider all possible combinations of multiple tools. Most notably, you cannot simply add the percentages of improvement when multiple tools are used. If you use two tools, each with a confidentiality improvement of 20%, it would be inaccurate to assume that the combined improvement is 20% + 20% = 40%; in particular, the improvement to risk cannot reach 100%.

We assume that the magnitude of incremental addition would decrease more slowly with lower levels of improvement than with higher levels. The best formula we could find to replicate this phenomenon is the $\tanh function$. The function $y = \tanh 1.05x$ is very nearly equal to x very closely until a 40% degree of improvement ($x = 0.40$), at which point the function starts to level off toward an asymptote of 1. Since $\tanh$ is symmetrical about 0, this formula performs in the same fashion for factors than detract or improve a given factor level.

The final step in this model is creating a formula for optimization. As the opportunity cost of risk decreases, the cost of tools increases. We need to minimize the overall costs incurred by the system,

$$\min(C_T + C_R).$$

We use the `Solver` function in Microsoft Excel to perform the optimization.

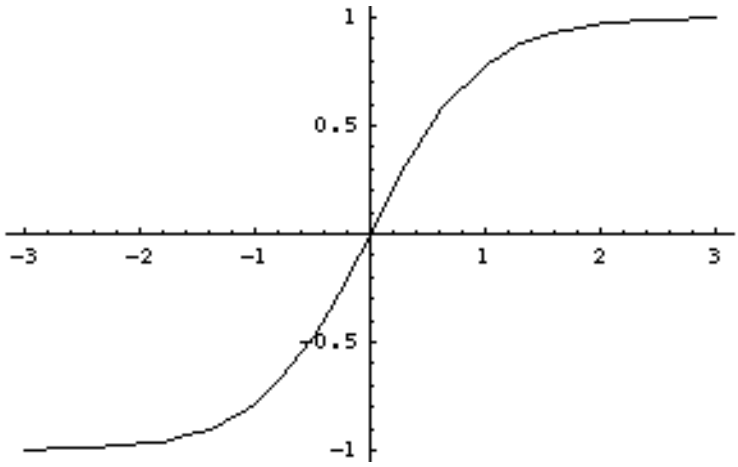

Figure 2. Net improvement vs. sum of improvement effects ($y = \tanh 1.05x$).

We use the truncated normal distributions to generate 500 random numbers (based on that distribution) for each data item, with each number representing a different iteration.

The decision variables are the amount of each tool that the network would use. Excel would search through all the possible combinations of decision variables and choose the set of decision variables that minimizes the cost equation over the 500 iterations.

We constrained the `Solver` function to

- force all decision variables to be integers (to eliminate the possibility of `Solver` recommending the use of a fraction of a resource, such as 54.34% of a firewall);
- force all decision variables to be nonnegative (so we would not recommend -2 firewalls); and
- choose each tool at most once for the network or each subnetwork (to avoid `Solver` recommending relying on 16 network firewalls), via constraining that the sum of all decision variables for a given tool should be less than or equal to 1.

The network policies have additional constraints. For instance, we assumed that we must select either a strong password policy or no password policy, so the sum of their decision variables must equal 1. Similar constraints apply to the use of wireless- and personal-use policies. Network-based decision variables are split into the subnetworks, for which similar constraints are made. `Solver` could choose a different combination of security measures for each subnetwork's host computers.

The degree to which a host-based system used on a particular network improved the overall network was based on the relative weight of importance of that subnetwork. For instance, if Subnetwork 1 accounts for 50% of the risk

to overall confidentiality, and a tool improves it 20%, then the use of that tool improves the overall network by $20\% \times 50\% = 10\%$. It was in this use of weights that the host-based options were chosen along side network-based tools and policies. The sum of factor improvements renders the value of ξ.

`Solver` ran every possible combination and found which combination minimized total cost the most over the 500 iterations.

Results

The optimal suite of defensive measures costs $1.37 million and is expected to lower its expected losses to $1.70 million, for a net savings of $5.86 million.

The tools recommended are:

- **Network Based Tools**
  - Network Defense Firewall
  - Enterprise Inoculation Anti-Virus
  - Network Eye IDS
- **Network Policies**
  - Strong Password Policy
  - Disallow Wireless
  - Unmonitored personal use
  - User training required
- **Host Based Tools**

Subnetwork 1	**Subnetwork 2**	**Subnetwork 3**
(Adm., Reg., Hlth)	(Acad. and Dorm)	(Athl. and Bkstr)
Lava Firewall	Intelliscan Firewall	Intelliscan Firewall
Bug Killer Anti-virus	Bug Killer AV	Bug Killer AV
Robust Solutions SR	Sonic Data DR	Sonic Data DR
	Web King SR	Web King SR

Strengths and Weaknesses

Strengths

The optimization model takes into account the delicate balance between the opportunity costs of the security risks (value of the assets) and the costs of implementing each additional defensive measure.

The model takes into account the proper use of the defensive measures by optimizing each subnetwork according to its function and requirements. The

risk category of integrity would not affect Subnetwork 2 (Academic Departments and Dormitory Complex) as significantly as Subnetwork 1 (Registrar's Office and Admissions Office). Thus, different defensive measures are utilized for each subnetwork and the respective host computers.

By generating reliable observations (based on the normal distribution of supplied data), we simulated the performance of each allowable combination of tools and all possible defensive performances. Every possible outcome of these two factors was considered (over 500 iterations) to produce an optimal solution.

Weaknesses

University Infrastructure: The proposed infrastructure is a simplified topology of the university's network, but perhaps not the best.

Economics

The optimization model takes into account opportunity costs and the cost for the implementation of each defensive measure. However, information technology security cannot always be quantified. Certain human factors, behaviors, and other x-factors cannot necessarily be incorporated into a quantitative model.

Human Factors

When building the model, we did not differentiate between inside and outside attacks. For instance, users in the dormitory complexes are probably more likely to "hack" the system than users in the admissions dept. The optimal security design probably would have been altered if our model accounted for these specific considerations.

User Productivity

The technical data sheets provided give scores that indicate the degree to which each defensive product reduces opportunity costs in terms of integrity, confidentiality, and availability. Our model picks an optimal array of these products by considering costs broken down into these three categories. However, our model fails to consider another metric, User Productivity. For every product, the data sheets give a score that indicates the degree to which user productivity would be hindered by that defensive measure. Certain designs could lead to excessive slowing of the network, user frustration, prohibition of routine transactions, or reduction of potential profits. We certainly considered this factor, and the model even calculated the net reduction in user productivity (7%); but we did not assign a cost to user productivity and incorporate it into the objective function. Fortunately, 7% is not excessively large, so the reduction in user productivity appears to be acceptable.

Improvement by Combinations

The model did not fully explore the degree to which the combination of different tools would effect overall performance of the system. As a partial solution, we disallowed the use of a single defensive measure twice on the same network. We did not explore the overlap which might be present between separate measures, opting instead for modeling this phenomenon in terms of diminishing degrees of improvement (via the `tanh` function).

Conclusion

Our model for the security of the new university's network provides the optimal balance of security and risk, based on associated costs. As new technologies arise, they can be added to our current decision matrices.

Appendix: Honeynet Analysis

Purpose

To determine whether a university or a search-engine company should consider using a honeynet. This memorandum provides a basic introduction to honeynet strategies. In addition, we highlight innovative techniques for deploying these strategies in a myriad of applicable fields.

Introduction

Bears like honey. Honey is made by bees; bees hate bears.

The bears of IT are blackhats (hackers). Their objective is to wallow neck-deep in a vat of warm, sweet honey. In this analogy, honey is a forbidden commodity—restricted information. True hackers claim a benevolent mission; others, called "crackers," have malicious aims to compromise network resources.

Regardless of an intruder's aims, all can pose threats to a target system. Network administrators (white hats) need to monitor for instances of suspicious activity. On busy networks, the task of pinpointing unauthorized use is incredibly difficult. A hacker can appear and vanish across busy resources like a thief disappearing in the bustling crowd of a Chinese street market. To level this playing field, administrators snipe hackers in open fields, who are lured by the sight of "easy" honey. Here is how:

Honeypot: an information system resource with value that lies in the unauthorized or illicit use of that resource [Spitzner 2003]]. The honeypot resources have no production activity, no authorized activity. Since the honeypot is not

a productive system, any interaction with that resource implies malicious or unauthorized use [Honeynet Project 2003]. This assumption of wrongdoing allows administrators to set up complex systems for observing intruder behavior. In doing so, administrators can learn from observations of new hacker techniques. This information fuels the development of updated anti-intrusion systems.

Honeynet: a network of honeypots created for an intruder to interact with.

Honeytoken: While honeypots are traditionally thought of as computers, (and other physical resources), a honeytoken broadens that paradigm. Honeytokens can be credit-card numbers, Excel spreadsheets, or even a bogus login [Spitzner 2003]. An example might be a medical file database containing an entry "John F. Kennedy." Since there is no actual patient with that name, any interaction with that file is assumed to be unauthorized. These tokens can be spread over the network like honey barbecue sauce.

Honey farm: a configuration in which traditional honeypot locations serve as portals, secretly redirecting intruders to one centralized honeynet system. This organization makes the monitoring of a single environment much easier.

Benefits and Risks

Benefits [Project Honeynet 2003]

The advantage of a honeynet is that it allows an administrator to gain extensive data on the abilities and tactics of system intruders. The architecture of a honeynet is much like a fishbowl. It allows administrators to focus completely on a set of unauthorized actions. The traditional method of searching for hackers involved looking through gigabytes of data of a busy network (busied mostly by legitimate use). Searching busy resources is like searching for a needle in a haystack. The honeypot concept serves as a magnet to those needles—no searching necessary. The compilation of information on intruders allows a system administrator to tailor the defense of the network.

Risks [Project Honeynet 2003]

Harm: An attacker may break into a honeynet and then launch an attack that the system cannot forestall. In this case, an attacker will successfully harm the intended victim. Data control is the primary method of reducing this susceptibility to system failure. Each organization must decide which level of control they want. More control allows the intruder to do less, leaving less to be observed. Less control allows the intruder more flexibility but increases the possibility of an administrator losing control.

Detection: If an intruder is able to identify a honeynet, the value of that resource is dramatically reduced (to an observing administrator). An intruder can

introduce false or bogus data into the honeynet, causing confusion for an administrator. In addition, an intruder might be able to identify the data-control and data-capture tools employed by the honeynet. If this occurs, an intruder can exploit the system architecture to gain access to non-honeynet resources.

Disable: There is risk that an intruder will disable the honeynet functionality. The intruder might be able to do this without the honeynet administrator realizing. This risk can be mitigated by having multiple layers of data control and capture.

Violation: If a honeynet is compromised, an intruder may attempt to use that resource for illegal activity. For example, the intruder might choose to upload and distribute illegal material, such as stolen credit cards or child pornography. This might cost the company painful litigation and additional penalties if they are found to be negligent in securing the resources involved.

Discussion

Although there are many risks associated with creating a honeynet, these risks can be mitigated by using a customized and random configuration, layering, some type of dynamism, or other creative means to make detection of the honeynet and countermeasures against it tough to accomplish. Any organization can find and tailor a honeynet to their acceptable risk exposure.

Recommendation

A university, search-engine company, or any other information system should employ some form of honeypot tactics. Combinations of the strategies allow white hats to seize the initiative in the battle against hackers, crackers, and dishonest employees. Additional cost/benefit analysis should be conducted to create an optimal honeynet configuration.

References

Decisioneering, Inc. 2004. Crystal Ball. Add-in software to Microsoft Excel under Microsoft Windows. `http://www.crystalball.com/crystal_ball/index.html` .

Honeynet Project. 2003. Know your enemy: Honeynets—What a Honeynet is, its value, how it works, and risk/issues involved. `http://project.honeynet.org/papers/honeynet/index.html` . Last modified 12 November 2003.

Devore, Jay L. 2000. *Probability and Statistics for Engineering and the Sciences.* Pacific Grove, PA: Brooks/Cole.

Peltier, Thomas R. 2001. *Information Security Risk Analysis.* New York: CRC Press.

Pfleeger, Charles P., and Shari Lawrence Pfleeger. 2003. *Security in Computing*. 3rd ed. Upper Saddle River, NJ: Prentice Hall.

Ragsdale, Cliff T. 2004. *Spreadsheet Modeling and Decision Analysis.* 4th ed. Mason, OH: South-Western.

Spitzner, Lance. 2003. Honeytokens: The other honeypot. `http://www.securityfocus.com/infocus/1713` . Last updated 21 July 2003.