THE TOWER-OF-BABEL PROBLEM, AND SECURITY ASSESSMENT SHARING

ANDREAS BLASS, YURI GUREVICH, AND EFIM HUDIS

Abstract. The tower-of-Babel problem is rather general: How to enable a collaboration among experts speaking different languages? A computer security version of the tower-of-Babel problem is rather important. A recent Microsoft solution for that security problem, called Security Assessment Sharing, is based on this idea: A tiny common language goes a long way. We construct simple mathematical models showing that the idea is sound.

> *And the Lord said: "If, as one people with one language, this is how they began to act, then nothing that they may propose to do is out of their reach. Let's then go down and confound their speech, so that they shall not understand one another."*
>
> — *Genesis 11*

1. Introduction

The Biblical story of the Tower of Babel is well known. What does it tell us? A religious teacher may use it to teach humility. For an atheist historian it may be just a folktale from old and more primitive times. A philosopher may draw our attention to the fact that even a common language does not guarantee mutual understanding. But the story raises a rather general logico-engineering problem. Consider a group of experts with no common language for any two of them. Can they collaborate to achieve a practical goal? If yes then how? In real world, one may seek the help of translators. It may be impossible to find a super-expert that speaks all the languages. It may be too expensive to employ many translators.

The tower-of-Babel problem is quite ubiquitous. Think of medical experts. An otolaryngologist may not understand a podiatrist. Imagine that on your way to a skiing area you meet a fellow skier who tells you that he is a medical doctor. And then you have a skiing accident. You hit a tree, and you are in pain. Suddenly you see your new acquaintance. "Doctor, help!" you shout. The doctor comes. "Sorry," he says, "I can't help. I am a psychiatrist[1]."

Another version of the tower-of-Babel problem is related to distributed databases. The information about one object may be spread over a large number of local databases. Checking for consistency, infection, etc. may be hard and may require expensive transactions. View local databases as experts observing different features of objects in question.

This brings us to the computer security version of the tower-of-Babel problem that we have been working upon. Consider the computer system of an enterprise.

The work of Andreas Blass was partially supported by NSF grant DMS-0653696.

[1] We did not invent the story, but we do not have a reference.

It has many automated experts that are supposed to make the computer system secure. Some of these experts, e.g. firewalls, are primarily security experts but others, e.g. routers, assume security duties on top of their primary duties.

There are *network edge security experts* related to network security, for example, firewalls, routers, gateways, spam experts, virus experts, network intrusion detection experts, and data leak protection experts also known as network extrusion detection experts. There are *host experts* that monitor individual computers, for example, anti-malware and anti-spyware experts, host firewalls, host intrusion and extrusion protection systems, so-called health management experts, registry monitors (specific to Windows operating system). There are *identity management experts*. For example, you may have an expert monitoring password changes. On later Windows systems you have Active Directory and an expert monitoring the elevation of user privileges. Various database and application experts have security duties as well.

These experts speak different languages and may be produced by different vendors. The problem is how to integrate the information available to them.

A natural approach to working from raw data up to usable security assessments is to design a “super-expert” that would collect and analyse data (low level security events) from various sources (sensors, logs) and would deliver the results in a useful form. There are quite a number of such products on the market, and they are known under different names, especially under the names of Security Information Management (SIM), Security Event Management (SEM), and Security Information and Event management (SIEM). Arcsight Enterprise Security [1] won the 2010 ”Best SIM/SIEM solution” award of SC Magazine [4]. In these products, the super-expert usually works directly with the raw data. As a result, the experts described above (firewalls, anti-virus experts, registry monitors, etc.) have very little analysis to do; it is the analysis of the super-expert that is all important. Since different vendors use different event data formats and different delivery methods, there are attempts to standardize those, in particular by MITRE Corporation [3].

The SAS team was skeptical about such super-expert approaches. You need to collect low level security information, lug it all to one central place, and try to analyze it there. This is a large overhead in lugging and keeping such a huge amount of data. Even more importantly, it is hard to program the analysis of that pile of data.

Efim Hudis, the third author of this article, suggested a different approach to the tower-of-Babel problem, an approach combining two ideas. The first idea is to make real use of the original experts, having them evaluate the security implications of what they observe. To motivate his second idea, notice that different languages are not necessarily disjoint. There may be terms understood by all. For example, all physicians understand the terms “male” and “female.” And maybe the experts can learn a few common terms. The second idea is that a tiny language suffices for many practical purposes. The approach may not solve some variants of the tower-of-Babel problem. But it works for the computer security version of the problem.

That tiny-language approach to the computer security problem was implemented in Microsoft as a Security Assessment Sharing system (SAS). The effort was part of the work on Forefront line of security products [2]. Andreas Blass and Yuri Gurevich consulted for the SAS group.

While SAS had been implemented, the group sought to confirm that the tiny communication language of SAS is helpful. One way to get such a confirmation is to construct a mathematical model of SAS and check whether communication helps in that model. This is especially useful when the available experimental data is insufficient. The model should capture the crucial ideas but be simple enough to allow calculations. More ambitiously, the model could be a simple playground helping to fine-tune the SAS architecture. The two logician consultants worked on the modeling problem and tried various approaches, in particular machine learning. At the end, it was not logic or artificial intelligence but old-fashioned probability theory that worked.

In the next section, we quickly describe SAS. The rest of the article is devoted to simple mathematical models of SAS.

Acknowledgment. The authors are deeply grateful to all the SAS group members. Special thanks go to Yossi Malka, Zakie Mashiah, and Shai Rubin.

2. Security Assessment Sharing

Imagine that you (and your group) are responsible for the computer security of an enterprise. You have under your control various automated computer experts — network edge firewalls, anti-virus software, network intrusion prevention systems, email security experts, and so on. Some of them are responsible for the network edge security, some of them monitor various events inside the enterprise. How do you reconcile their reports? For example, what does it mean that "virus was detected and cleaned" on a computer? Is it normal or should you worry? How to compare "cleaned virus" from an anti-malware expert, "abnormal traffic" from a firewall and "failed login" from a Windows Security log? What is more urgent, three "failed login" events on one computer or one "cleaned virus" event on another computer?

Further, as threats become more sophisticated, even individual experts move from black-and-white reports ("virus found") to shades-of-grey reports ("potential malicious activity detected"). It addition to telling you that the recently downloaded and executed code may be malicious, a host intrusion system may also give you the probability that the code is malicious. The intrusion system does not know whether the code is malicious or not; it just detected some suspicious aspects of the execution. But how are you supposed to deal with such shade-of-grey responses?

It would by highly desirable to have an automated system that combines and correlates the reports of the automated security experts in order to produce an overall assessment of system security, in a form useful to system administrators. The Security Assessment Sharing system was built to do just that.

The main idea behind SAS is to have a tiny language that all the experts understand. An expert that detects a security problem uses this language to broadcast to the other experts a security assessment with some very limited information about what it found. The other experts can take that assessment into account when making their own decisions as to whether they see a problem in their, presumably different, areas of concern. The hope is that even very restricted communication of this sort may improve the experts' ability to detect problems, and to avoid issuing "false positive" warnings when no problem actually exists. The central issue addressed in the present paper is the extent to which extremely limited languages may prove useful. This issue is a reflection, in an unusual context, of a familiar concern of mathematical logic, namely the trade-off between two desirable properties of languages: simplicity and expressive power.

SAS is designed to be used by various enterprises and institutions. In the current version of SAS, a security assessment issued by an expert can be represented (for the purposes of this article) as a record with the following fields.

- **Object** The expert specifies the object whose security is assessed. Currently there are two kinds of objects: users (identified by the identity number of their account with the enterprise or institution), and computers (identified by unique network-addressable name).
- **Problem class** The expert specifies the problem with the object. That is done on a very high level. Currently this field has just two possible values: "compromised" and "vulnerable."
- **Severity** The expert assesses the severity of the problem. Currently there are three levels of severity: "high," "medium," and "low".
- **Confidence** The expert indicates how confident it is that it sees a problem. Currently there are three confidence levels: "high," "medium," and "low."
- **Time** This field gives the issue time and the validity period (e.g. 12 hours) of the assessment.
- **Issuer** The expert issuing the assessment is specified.
- **Immediate influencers** The set of valid assessments that the issuer took into account producing the present assessment.

Thus the language is very limited. In particular, the confidence field carries exactly two bits of information. Indeed, in addition to "high," "medium" and "low," the expert has a fourth option: to issue no alert at all.

In general, one could expect better overall results if more information is communicated, but there is a cost, namely that very different sorts of experts, produced by different vendors, need to understand each other's communications. In particular, as far as confidence is concerned, there should be a common understanding among the experts of what high, medium and low confidence levels mean. Here is one natural criterion. Among the occasions when an expert issues a warning with high (resp. medium, resp. low) confidence, there should actually be a problem at least 90% (resp. 50%, resp. 10%) of the time. We shall later see some difficulties with this criterion, but a priori it seems a reasonable specification of what the levels of confidence should mean.

Remark 1. The field "Immediate influencers" needs an explanation. Imagine that expert 1 issues an assessment that provokes expert 2 to issue an assessment that provokes expert 1 to issue a stronger assessment and so on. It makes sense to avoid such self-reinforcing loops. To this end, an expert E ignores all assessments a where E itself appears as an immediate influencer; a is not an immediate influencer of any assessment of E. This simple trick does not remove all self-reinforcing loops but it does remove the most obvious and most damaging loops.

The real-world SAS provides numerous services. We mention here only some of them.

- SAS defines the notion of risk so that every incident is assigned a numerical risk value. SAS orders the current incidents in the risk order.

- SAS has a hierarchy of data abstraction levels, from raw data (session logs, activity logs, scans, and the like) at the bottom to the list of incidents in the risk order at the top. Investigators (e.g. SAS administrators or auditors) may be satisfied with just the list of incidents. If not, they my look at the assessments issued. If this is not enough and they want to see more detailed information, they may dig further until they arrive at raw data.
- Note that the best place to respond to a security incident is often different from the best place to detect it. For example, a firewall may detect a computer worm in the outgoing traffic, but the best place to take care of the worm is elsewhere. SAS helps investigators to find right places to respond to security incidents.

In the rest of this paper, we address the fundamental question underlying the SAS system: Can very limited communication between experts substantially improve their ability to discern real problems from innocent, random fluctuations?

3. Results

We approach the question of the value of limited communication by analyzing some very simple models of the experts' activity and of the system under observation. This analysis leads to several conclusions.

First, even very limited communication between experts can really help.

Second, perhaps surprisingly, the 90%-50%-10% specification of the confidence levels will not work in realistic situations. It needs to be replaced with a description that takes into account the costs of false positives and of false negatives, and we analyze a model that incorporates this cost-based approach.

Third, an expert's decision whether to issue an alert should not, in general, be based on a simple threshold. Even for quite simple probabilistic relationships between the state of the system and the expert's observations, there might, for example, be two thresholds, such that an alert should be issued just when the observation lies between the two.

4. First Model: Threshold Probability

In this section, we describe the first of our simplified models of SAS. We make the following assumptions:

- The system is in one of two states, "normal" and "bad," which we abbreviate as N and B. The a priori probability of being in the bad state is p.
- The system has two observable properties, X and Y, each of which is a real-valued random variable.
- When the system is in the normal state, X and Y are independent and both have the probability density function $\varphi(x)$, whose mean is 0. (In many of our calculations, φ will be taken to be a normal distribution, but the general theory will apply to arbitrary φ.)
- When the system is in the bad state, X and Y are independent and both have the probability density function $\varphi(x-1)$, whose mean is 1.
- There are just two experts. One observes X and the other observes Y. By abuse of language, the experts are named X and Y, respectively.
- Each expert knows p and φ.

- Each expert's goal is to tell whether the probability (given what it observes or otherwise learns) of the bad state is $> \frac{1}{2}$ or not. (So each expert outputs just one bit.)

We first consider what each expert will do in isolation. Afterward, we shall consider what Y will do if, in addition to observing the random variable Y, it has already heard X's one-bit output. Comparing the two, we intend to describe the effect of this one bit of communication.

Remark 2. The assumption that there are only two states is essentially a convention; we lump together all non-normal states as a single "bad" state. We think of the a priori probability p of B as being quite small; 10^{-4} is a plausible value. But the theory presented in this section doesn't depend on whether p is small.

Remark 3. The assumption that there are just two experts is an oversimplification. That their observed random variables have mean 0 in the normal state and mean 1 in the bad state is just an affine scaling of these random variables. That the probability density functions are the same except for a translation is another oversimplification, as is the assumption that, in both the normal and the bad state, X and Y are independent. This last assumption, of independence in each state, is often called a "naïve Bayes" assumption.

To avoid possible confusion, we emphasize that our assumptions about independence are about the conditional distributions of X and Y, conditional on either the normal or the bad state. Conditional on either of the two states, X and Y are independent. It does not follow that they are independent in the absence of conditioning. The easiest way to see this is to imagine that the distribution φ is concentrated in a very small interval around the mean. Then when X is near 0 it is extremely likely that Y is also near 0, and similarly with 1 in place of 0. So there is a very strong correlation between the unconditioned X and Y.

Remark 4. There is another simplification in our assumption that each expert outputs just one bit, rather than the two bits involved in distinguishing the four options of high, medium, and low confidence alerts plus not issuing an alert. Our use of $\frac{1}{2}$ in the last assumption amounts to the 50% threshold mentioned above for medium confidence alerts. Thus, our one-bit alert corresponds to "high or medium" in the more detailed picture.

Remark 5. We shall compute what expert X should do when it observes a particular value x for the random variable X. Our description above says that it should produce an alert if and only if the conditional probability $\mathbb{P}(B|X = x) > \frac{1}{2}$. Strictly speaking, this doesn't make sense, since it refers to a conditional probability where the condition $X = x$ is an event of probability zero (assuming φ is a continuous distribution). We shall use the convention that the expert really doesn't see the exact value $X = x$ but rather an infinitesimal interval of length dx around x, whose probability is $\varphi(x)\,dx$ in the normal state, $\varphi(x-1)\,dx$ in the problem state, and therefore

$$((1-p)\varphi(x) + p\varphi(x-1))\,dx$$

overall. In most of the formulas below, the dx factors would cancel at the end, so we can safely just omit them.

For more rigor, one could begin with finite intervals of length Δx instead of infinitesimal ones and later pass to a limit as $\Delta x \to 0^+$. For even greater rigor, one could invoke the Radon-Nikodym theorem to define (almost everywhere) probabilities conditioned on the value of a continuous random variable. All this rigor, however, would affect only the length, not the results, of what follows.

We compute, using Bayes's theorem, the condition for expert X to issue an alert when it observes the value x for its random variable X. For brevity, let us write simply x to denote the event $X = x$; similarly, let us write N (resp. B) to denote the event that the system is in the normal state N (resp. the bad state B). We use $\mathbb{P}$ for probability and (later) $\mathbb{E}$ for expectation. Then an alert should be issued just in case

$$\frac{1}{2} < \mathbb{P}(B|x) = \frac{\mathbb{P}(B) \cdot \mathbb{P}(x|B)}{\mathbb{P}(B) \cdot \mathbb{P}(x|B) + \mathbb{P}(N) \cdot \mathbb{P}(x|N)} = \frac{p\varphi(x-1)}{p\varphi(x-1) + (1-p)\varphi(x)}.$$

Clearing fractions and simplifying, we find that the criterion for issuing an alert is

$$\frac{\varphi(x-1)}{\varphi(x)} > \frac{1-p}{p}.$$

A simple but plausible distribution φ is the Gaussian normal distribution. If we think of the deviation of X from its mean 0 or 1 (in the normal or bad state, respectively) as arising from an accumulation of small, random fluctuations, then the Gaussian distribution is a standard model for such fluctuations. (This way of viewing the fluctuations might also support our simplifying assumption that the probability distributions for X in the system's two states differ only by a translation.) We therefore pause to calculate the criterion for issuing an alert in the special case where φ is Gaussian.

Using the usual notation σ^2 for the variance, we have

$$\varphi(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-x^2/2\sigma^2}.$$

In this case, $\dfrac{\varphi(x-1)}{\varphi(x)}$ simplifies to $\exp((2x-1)/2\sigma^2)$, and therefore the criterion for issuing an alert simplifies to

$$x > \frac{1}{2} + \sigma^2 \ln\left(\frac{1-p}{p}\right).$$

Returning to the general case, let us now calculate the criterion for expert Y to issue an alert if, in addition to observing $Y = y$, it has heard an alert from expert X. We assume that X's decision to produce this alert was based merely on its observation of a value x for its random variable X, exactly as in the preceding discussion. That is, in addition to $Y = y$, expert Y now knows also that

$$\frac{\varphi(X-1)}{\varphi(X)} > \frac{1-p}{p}.$$

We abbreviate this last event as C (for "communication"). The criterion for Y to issue an alert in this situation is computed exactly as above (with Y and y in place of X and x), except that all probabilities are now conditioned on C (in addition to any conditioning in the previous calculation).

Thus, instead of $p = \mathbb{P}(B)$, we will now have $p' = \mathbb{P}(B|C)$, and in place of $\varphi(x) = \mathbb{P}(x|N)$, we will now have $\varphi'(y) = \mathbb{P}(y|N \wedge C)$. We calculate these new quantities as follows.

For p', we use Bayes's theorem again:

$$p' = \mathbb{P}(B|C) = \frac{\mathbb{P}(B) \cdot \mathbb{P}(C|B)}{\mathbb{P}(B) \cdot \mathbb{P}(C|B) + \mathbb{P}(N) \cdot \mathbb{P}(C|N)} = \frac{p \cdot \mathbb{P}(C|B)}{p \cdot \mathbb{P}(C|B) + (1-p) \cdot \mathbb{P}(C|N)}.$$

It follows that

$$1 - p' = \frac{(1-p) \cdot \mathbb{P}(C|N)}{p \cdot \mathbb{P}(C|B) + (1-p) \cdot \mathbb{P}(C|N)},$$

and therefore

$$\frac{1-p'}{p'} = \frac{(1-p) \cdot \mathbb{P}(C|N)}{p \cdot \mathbb{P}(C|B)}.$$

Next, we calculate $\varphi'(y)$.

$$\begin{aligned}\varphi'(y) = \mathbb{P}(Y = y|N \wedge C) &= \frac{\mathbb{P}((Y = y) \wedge C|N)}{\mathbb{P}(C|N)} \\ &= \frac{\mathbb{P}(Y = y|N) \cdot \mathbb{P}(C|N)}{\mathbb{P}(C|N)} = \mathbb{P}(Y = y|N) = \varphi(y).\end{aligned}$$

The third equality here uses the assumption that X and Y are independent given N (and that the event C depends only on X).

A similar calculation shows that, when everything is conditioned on C, $\varphi(x-1)$ should also become simply $\varphi(y-1)$ in the earlier formulas. Thus, the criterion for Y to issue an alert has become

$$\frac{\varphi(y-1)}{\varphi(y)} > \frac{1-p}{p} \cdot \frac{\mathbb{P}(C|N)}{\mathbb{P}(C|B)}.$$

In the special case of Gaussian $\varphi(x)$, the criterion for Y to issue an alert after hearing X do so is

$$y > \frac{1}{2} + \sigma^2 \ln\left(\frac{1-p}{p} \cdot \frac{\mathbb{P}(C|N)}{\mathbb{P}(C|B)}\right).$$

Since X is more likely to issue an alert when the system is in the bad state than when it is in the normal state, i.e., since $\mathbb{P}(C|N) < \mathbb{P}(C|B)$, our formulas imply that Y's threshold for issuing an alert is lower as a result of X's alert. This is, of course, what one would expect.

All the preceding calculations have been based on the assumption that an expert should issue an alert if, given what it knows, the probability of the bad state B exceeds $1/2$. Very similar calculations apply if the probability threshold, the conditional probability of B that should cause an alert, is set to some other value τ. We do not repeat the calculation but merely record, for use in the next section, the result in the case of a single expert. An alert should be issued if

$$\frac{\varphi(x-1)}{\varphi(x)} > \frac{1-p}{p} \frac{\tau}{1-\tau}.$$

In the case of a normal distribution φ with standard deviation σ, this criterion means

$$x > \frac{1}{2} + \sigma^2 \ln\left(\frac{1-p}{p} \frac{\tau}{1-\tau}\right).$$

5. Computation for First Model

We record in Table 1 some numerical results for various levels of confidence. That is, we vary the parameter τ introduced above, namely the probability such that, when an expert thinks the probability of B exceeds τ, it issues an alert. For simplicity, we have assumed in all cases the same a priori probability of B, namely 10^{-4} and the same probability distribution φ, namely a Gaussian normal distribution with standard deviation 1/2. The value of p was chosen to be rather realistic. The value of σ was chosen so that the conditional distributions of X in the normal and bad states, namely $\varphi(x)$ and $\varphi(x-1)$ differ significantly (so that an expert observing X has a reasonable chance to tell what the state is) but do not differ so completely as to make the expert's job trivial. We now describe and discuss what is in the table.

τ	.01	.10	.30	.50	.70	.90
threshold	1.654	2.253	2.591	2.803	3.014	3.352
$\mathbb{P}(\text{alert}\|N)$	4.71 E−4	3.30 E–6	1.10 E–7	1.04 E–8	8.26 E–10	1.02 E–11
$\mathbb{P}(N\|\text{alert})$	.980	.844	.600	.400	.228	.074
$\mathbb{P}(\text{alert}\|B)$	9.55 E–2	6.10 E–3	7.33 E–4	1.56 E–4	2.80 E–5	1.28 E–6
$\mathbb{P}(\text{no alert}\|B)$	.904	.994	.999	.99984	.99997	.999999

Table 1. Computations with $p = 10^{-4}$ and $\sigma = 1/2$

Each column describes what happens for a particular value of τ, given in the first row. As we go from left to right, τ increases, so the expert is becoming less and less sensitive. (We omitted a few columns from our original calculations, for $\tau = .05$, .20 and .80, to fit the table on the page.)

The second row gives the threshold value t such that the expert will issue an alert if and only if the observed value of X exceeds t. Here we already see a perhaps unanticipated phenomenon. Even though the conditional distribution of X in the bad state B is centered at 1, the threshold t is greater than 1, even for the most sensitive expert ($\tau = .01$). In other words, the expert will not issue an alert unless it observes a value of X quite some distance out in the tails of the normal distributions. This is not a good situation, since it means that the bad state will usually produce no alert; we shall see later just how bad things are.

Once one sees these large thresholds in the table, it is not difficult to explain why they occur. Suppose, for example, that the expert observes a value x of X that is slightly bigger than 1. This value is considerably farther out in the tail of the distribution $\varphi(x)$ for the normal state than in the tail of the distribution $\varphi(x-1)$ for the bad state. Thus, the observed x is far more likely to occur in the bad state than in the normal state. Nevertheless, the a priori probability of the bad state is so low that even this strong observational evidence in favor of B does not raise its probability (as computed by Bayes's theorem) above τ.

The third row of the table gives the conditional probability of an alert being issued when the state is normal. The numbers here are extremely small, because it is very unlikely, in the normal state, for X to get so large as to exceed the thresholds in the preceding row. These small numbers are reassuring; when the system is in the normal state, the expert is unlikely to raise a false alarm.

The fourth row describes the situation from the reverse point of view: Given that an alert was issued, what is the probability that it was a false alarm, i.e., that the state is N? Here we see some unpleasantly large numbers, at least in the left half of the table where the expert is quite sensitive. But these are to be expected. In the first column of numbers, for example, the expert is trying to issue an alert when it thinks the probability of B is at least .01, so we should expect a large number of false alarms. In fact, one might naïvely expect about 99% of the alarms to be false. The actual situation is slightly better, only 98%, because the situations where the expert issues an alarm includes a few where the probability of B is significantly more than τ.

The fifth row gives the conditional probability that an alert is issued given that the state is bad, and the final row gives the complementary probability that the expert fails to warn us when the state is bad. This last row shows a very serious problem with the present approach. Even in the column where $\tau = .01$, so the expert is trying to react to even a slight probability of B, it misses more than 90% of the bad situations. Roughly speaking, even a very mild limitation on the false positives causes the expert to miss most of the real positives.

6. Second Model: Cost

The problem detected in the preceding section can be traced to an excessive fear of false positives. We insisted that, when an alert is issued, there should be at least a certain probability (50% in our original model, 1% in the first scenario of Table 1) that there really is a problem. We did not permit a lot of false alarms. That decision reduced the sensitivity of the experts to the point where they ignored too many cases where there really was a problem. By strictly limiting the rate of false positives, we produced too many false negatives. Of course, this phenomenon is not unexpected — there is always a trade-off between allowing a lot of false positives (by acting with high sensitivity) and allowing a lot of false negatives (by acting with low sensitivity). What was unexpected is the magnitude of the problem. Even in the case of low-confidence alerts, of which up to 90% are allowed to be false positives (and indeed even with 98% false positives), we still had too many false negatives, too many cases where a real problem did not lead even to a (very) low-confidence alert.

In this situation, we cannot expect to simultaneously reduce false positives and false negatives as much as we might like. We are stuck with a trade-off. To make an optimal choice in such a situation, we must ask what are the costs for false positives and for false negatives (or at least what is the ratio of the two costs). We therefore introduce the following model, similar to the one used above, but taking into account the ratio of the cost of a false negative to the cost of a false positive.

For the sake of simplicity in the ensuing calculations, we develop this model only in the case of normal distributions. We shall, after a while, introduce a bit more generality by allowing the variance of the normal distribution to be different in the normal and bad states. But first, let us consider the simplest cost-based model, where the two variances are equal, so the two distributions have the form $\varphi(x)$ and $\varphi(x-1)$, as in our first model.

We begin, as before, by considering the action of an expert acting alone. Our assumptions are therefore as follows.

- There is one expert; it observes a random variable X.
- The system is in one of two states, bad (B) with a priori probability p and normal (N) with a priori probability $1-p$.

- In state N, the distribution of X is normal with mean 0 and standard deviation σ.
- In state B, the distribution of X is normal with mean 1 and standard deviation σ.
- The cost of a false positive is 1, and the cost of a false negative is c.

Remark 6. The value 1 for the cost of a false positive is just a normalization. Whatever the actual cost of a false positive may be, we use that as our unit of cost, and we measure the cost of a false negative relative to it. Thus, independently of any choice of units, c represents the ratio between the cost of a false negative and the cost of a false positive.

Our model regards all false positives as having the same cost (and similarly for false negatives). A more realistic model might take into account that a single false positive costs very little but a large number r of false positives might cost considerably more than r times the cost of a single one, especially if r gets so large that the system administrator starts to simply ignore alerts, thereby making the security system useless.

Remark 7. In the description of p and $1-p$, the phrase "a priori" should be understood as meaning only "before X is observed." In particular, if other experts had been present, p could depend on information obtained by our X expert from listening to others before observing X.

We want to find a threshold value t such that, by issuing an alert if and only if $X > t$, the expert minimizes the expected cost of errors.

For any value of t, the resulting probability of a false positive is

$$(1-p)\cdot\left(1-\Phi\left(\frac{t}{\sigma}\right)\right).$$

Here Φ is the cumulative probability distribution function for the standard normal distribution

$$\Phi(s)=\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{s}e^{-u^2/2}du.$$

So in the formula for the probability of a false positive, the factor $(1-p)$ is the probability of state N, and the second factor is the conditional probability, given N, that $X > t$, i.e., that an alert is issued.

Similarly, the probability of a false negative is

$$p\cdot\Phi\left(\frac{t-1}{\sigma}\right).$$

Therefore, the expected cost is

$$\mathbb{E}(C)=(1-p)\cdot\left(1-\Phi\left(\frac{t}{\sigma}\right)\right)+cp\cdot\Phi\left(\frac{t-1}{\sigma}\right).$$

We want to find the value of t that minimizes this expected cost, so we differentiate $\mathbb{E}(C)$ with respect to t, holding c, p, and σ constant, and we set the derivative equal to 0.

$$0=-(1-p)\Phi'\left(\frac{t}{\sigma}\right)\frac{1}{\sigma}+cp\Phi'\left(\frac{t-1}{\sigma}\right)\frac{1}{\sigma}.$$

Note that the derivative Φ' of the cumulative probability distribution function is the probability density function

$$\Phi'(s) = \frac{1}{\sqrt{2\pi}} e^{-s^2/2}.$$

Substituting this into the previous equation, cancelling some common factors, and bringing all the exponential factors to the left and everything else to the right, we get

$$\exp\left(-\frac{t^2}{2\sigma^2} + \frac{(t-1)^2}{2\sigma^2}\right) = \frac{cp}{1-p}.$$

Taking logarithms of both sides and simplifying, we get

$$t = \frac{1}{2} - \sigma^2 \ln\left(\frac{cp}{1-p}\right). \tag{1}$$

It is worth noting that the relative cost c and the a priori probability p of the bad state occur only in the combination $cp/(1-p)$. This is the ratio between the expected cost of never issuing an alert and the expected cost of always issuing an alert.

Both for comparison with the previous model and for future calculations of what happens when a second expert is present, we need the conditional probability of the bad state B given that an alert is issued. In other words, when an alert is issued, what is the probability that it is correct? To compute this, we again use Bayes's theorem.

Let A be the event that an alert is issued. Then

$$\mathbb{P}(B|A) = \frac{\mathbb{P}(A|B)\mathbb{P}(B)}{\mathbb{P}(A|B)\mathbb{P}(B) + \mathbb{P}(A|N)\mathbb{P}(N)}. \tag{2}$$

Here $\mathbb{P}(B)$ and $\mathbb{P}(N)$ are the a priori probabilities p and $1-p$, respectively. The conditional probability $\mathbb{P}(A|B)$ is the probability that, when the system is in state B, the value of X is larger than the threshold t computed above, so

$$\mathbb{P}(A|B) = 1 - \Phi\left(\frac{t-1}{\sigma}\right)$$

where t is to be obtained from equation (1) above. Similarly,

$$\mathbb{P}(A|N) = 1 - \Phi\left(\frac{t}{\sigma}\right).$$

Substituting into equation (2), we get

$$\mathbb{P}(B|A) = \frac{p \cdot \left(1 - \Phi\left(\frac{t-1}{\sigma}\right)\right)}{p \cdot \left(1 - \Phi\left(\frac{t-1}{\sigma}\right)\right) + (1-p) \cdot \left(1 - \Phi\left(\frac{t}{\sigma}\right)\right)}. \tag{3}$$

This equation together with the formula (1) for t gives the probability that, when an alert is issued, there really is a problem (as a function of p, σ, c).

We now consider the following scenario. There are two agents, one observing a random variable X as above and one observing Y. We assume that Y, like X, is normally distributed, with mean 0 in the normal state and mean 1 in the bad state, and we assume that these two distributions have the same standard deviation σ', which may, however, differ from the standard deviation σ of X (in either state). We also allow the cost ratio c' to be different for Y than it was for X. (Concerning the a priori probability p' of B that expert Y should use, see below.) The X expert sets its threshold t as in (1) above, and, after observing X, either issues an alert or doesn't. The other expert knows, before observing Y, whether the first expert has issued an alert, and it is allowed to use this information in setting its own threshold and, after observing Y, deciding whether to issue an alert.

As in our first model, we make the naïve Bayes assumption that X and Y are conditionally independent given B and are also conditionally independent given N. (Unconditionally, they are of course correlated.) Under this assumption, the analysis becomes easy; in fact, we have already done all the work.

Equation (1) determines the X expert's threshold t. Substituting this value for t in equation (3), we get the probability that there is a problem when the X expert issues an alert. If the X expert in fact issues an alert and, according to our scenario, the Y expert knows about it before observing Y, then this probability, from (3), is what the Y expert should use as its a priori probability p' of B. Then the Y expert sets its threshold t' according to equation (1) with p', c', σ' in place of p, c, σ.

Similarly, if the X expert does not issue an alert, then the Y expert would know this and set its a priori probability p' equal to the probability of B given that the X expert issues no alert. Just as above, that probability is given by Bayes's theorem:

$$\mathbb{P}(B|\neg A) = \frac{\mathbb{P}(\neg A|B)\mathbb{P}(B)}{\mathbb{P}(\neg A|B)\mathbb{P}(B) + \mathbb{P}(\neg A|N)\mathbb{P}(N)} = \frac{p \cdot \Phi\left(\frac{t-1}{\sigma}\right)}{p \cdot \Phi\left(\frac{t-1}{\sigma}\right) + (1-p) \cdot \Phi\left(\frac{t}{\sigma}\right)}.$$

As before, the t here is to be taken from equation (1), and this conditional probability serves as the p' (in conjunction with c', σ') in the calculation of the threshold t' to be used by the Y expert when the X expert declines to issue an alert.

7. Computation for Second Model

Table 2 shows computations for our second, cost-based model. As before, we assume that the a priori probability of B is 10^{-4} and that the distribution φ is Gaussian with standard deviation 1/2. The three columns of the table differ in the assumed ratio c between the cost of a false negative and the cost of a false positive. We now explain the entries in the table.

The four lines under the heading "ALONE" describe what happens when a single expert is active. The threshold row gives the value t such that an alert is issued whenever the observed value of the expert's random ariable exceeds t. The next two rows give the conditional probabilities that the expert incurs a cost, either for issuing no alert when the state is B (a cost of c) or for issuing an alert when the state is N (a cost of 1). The last row in this section of the table gives the overall expected cost incurred by this expert.

The rows under the heading "HELPED" refer to the situation where another expert has already observed X and decided (alone) whether to issue an alert. Our expert knows, before observing its random variable Y, what X decided, and this information is taken into account in determining the appropriate threshold for Y, the threshold called t' in the preceding section.

The first two rows under "HELPED" give the new higher threshold to be used when X issued no alert and the new lower threshold to be used when X did issue an alert.

The row marked "$\mathbb{P}(\text{no} \mid \text{no}, B)$" gives the conditional probability that Y will issue no alert, given that X issues no alert and the actual state is B. Similarly, the next row, "$\mathbb{P}(\text{no} \mid \text{yes}, B)$" gives the conditional probability that Y will issue no

cost ratio	100	10	2
ALONE			
threshold	.788	.932	1.032
$\mathbb{P}(\text{no alert} \mid B)$	.199	.392	.551
$\mathbb{P}(\text{alert} \mid N)$	8.12 E–4	9.69 E–5	1.82 E–5
$\mathbb{E}(\text{cost})$	2.79 E–3	4.89 E–4	1.28 E–4
HELPED			
high threshold	.889	.990	1.070
low threshold	.357	.385	.400
$\mathbb{P}(\text{no} \mid \text{no}, B)$	.328	.484	.610
$\mathbb{P}(\text{no} \mid \text{yes}, B)$	.005	.007	.008
$\mathbb{P}(\text{no alert} \mid B)$	.069	.194	.340
$\mathbb{P}(\text{yes} \mid \text{no}, N)$	1.88 E–4	3.74 E–5	1.82 E–5
$\mathbb{P}(\text{yes} \mid \text{yes}, N)$	7.67 E–2	6.16 E–2	5.47 E–2
$\mathbb{P}(\text{alert} \mid N)$	2.51 E–4	4.33 E–5	1.04 E–5
$\mathbb{E}(\text{cost})$	9.41 E–4	2.38 E–4	7.84 E–5
expected cost of no alerts	1 E–2	1 E–3	2 E–4

Table 2. Computations with $p = 10^{-4}$ and $\sigma = 1/2$, varying costs

alert, given that X does issue an alert and the state is B. These two are combined in the next row to give the probability that Y errs by issuing no alert when the state is B. The next three rows do the analogous computations leading up to the probability that Y errs by issuing an alert when the state is N. Finally, all this information is assembled to compute the expected cost incurred by Y in this situation.

We note that the expected costs in the "helped" situations are significantly lower than in the corresponding "alone" situations. The one bit of information that Y received from X was genuinely useful.

The last row of the table lists, for comparison, the expected costs that would be incurred if the experts never issued alerts. This row is included mainly to prevent false optimism caused by the rather small numbers in both of the previous "$\mathbb{E}(\text{cost})$" rows. The expected costs in these situations are always low, simply because the probability of B is low.

8. Third Model: Cost with Different Variances

In this final section, we generalize the model from Section 6 by allowing the variance of X to be different in the normal and bad states. We write ν^2 and β^2 for these two variances, still assuming Gaussian distributions centered on 0 in the normal state and on 1 in the bad state. We shall carry out the computations only for the case of a single expert. The case of a second expert, who knows whether the first has issued an alert, could be handled by the same methods as in Section 6, but the formulas would become considerably more complicated.

Our objective is to determine, for each value x of the random variable X, what the expert should do if it observes X to have this value. The expert has two choices, namely to issue an alert or not. Either way, it incurs a cost if the decision is wrong, and it should make its decision so as to minimize the expected cost.

Previously, we described the decision process in terms of a threshold t; the expert should issue an alert if and only if the observed value x of X exceeds t. But this assumption is implausible when the standard deviations β and ν differ. For an extreme example, suppose β is very small (say 0.01) and ν is large, say 10. Then if the observed value of X is 3, it is much more likely that the system is in the normal state (and the deviation of X from its mean 0, only 0.3 standard deviations, arose randomly) than that the system is in the bad state (and that the deviation of X from its mean 1, 200 standard deviations, arose randomly). Thus, although (depending on details of p and c), it may be reasonable to issue an alert when $X = 1$, it is not reasonable to do so when $X = 3$. In other words, the threshold idea is too simple-minded.

Suppose the expert observes a value x of X. We calculate the expectation of the cost when it issues an alert and when it doesn't issue an alert. If the expert issues an alert, then it incurs a cost of 1 if the state is in fact normal (so the alert was a false positive). The probability of this is, by Bayes's theorem,

$$\mathbb{P}(N|X=x) = \frac{(1-p)\frac{1}{\sqrt{2\pi}\nu}\exp(-x^2/2\nu^2)}{(1-p)\frac{1}{\sqrt{2\pi}\nu}\exp(-x^2/2\nu^2) + p\frac{1}{\sqrt{2\pi}\beta}\exp(-(x-1)^2/2\beta^2)}. \tag{4}$$

If, on the other hand, the expert declines to issue an alert, then it incurs a cost of c if the state is in fact bad (so the absence of an alert was a false negative). The probability of this is

$$\mathbb{P}(B|X=x) = \frac{p\frac{1}{\sqrt{2\pi}\beta}\exp(-(x-1)^2/2\beta^2)}{(1-p)\frac{1}{\sqrt{2\pi}\nu}\exp(-x^2/2\nu^2) + p\frac{1}{\sqrt{2\pi}\beta}\exp(-(x-1)^2/2\beta^2)}. \tag{5}$$

Note that the denominators in (4) and (5) are the same (and they are the sum of the two numerators, as the two conditional probabilities are complementary).

So the expert should issue an alert if and only if the probability in (4) is less than (or equal to) c times that in (5). (In the "equal to" case, the cost is the same whatever the expert does; in the future, we shall omit mention of this case.)

After canceling the common denominator of (4) and (5) and canceling the $\sqrt{2\pi}$ factors, we find that the expert should issue an alert when

$$(1-p)\frac{1}{\nu}\exp(-x^2/2\nu^2) < cp\frac{1}{\beta}\exp(-(x-1)^2/2\beta^2).$$

Bringing all the exponentials to the left and everything else to the right, we get

$$\exp\left(-\frac{x^2}{2\nu^2} + \frac{(x-1)^2}{2\beta^2}\right) < \frac{cp\nu}{(1-p)\beta}.$$

(Note that again we have c and p only in the combination $cp/(1-p)$.) Taking logarithms and transposing, we get the quadratic (except when $\nu = \beta$) inequality

$$x^2\left(\frac{1}{2\beta^2} - \frac{1}{2\nu^2}\right) - x\frac{1}{\beta^2} + \frac{1}{2\beta^2} - \ln\left(\frac{cp\nu}{(1-p)\beta}\right) < 0. \tag{6}$$

We consider several cases, depending on the properties of the coefficients in this inequality.

Case 1: $\nu = \beta$

Then inequality (6) is linear and simplifies to

$$x > \frac{1}{2} - \beta^2 \ln\left(\frac{cp}{1-p}\right).$$

So the expert should issue an alert if and only if the observed value of X exceeds the threshold indicated here. This confirms what we computed in Section 6. When the two variances are equal, the correct choices for the expert are given by a threshold, as was assumed earlier.

Before turning to the remaining cases, where the inequality (6) is quadratic, it will be useful to clear fractions and to introduce a notation for the discriminant (or, in some authors' terminology, half of the discriminant). We multiply (6) by $2\beta^2\nu^2$ to get

$$(\nu^2-\beta^2)x^2-2\nu^2x+\nu^2-2\nu^2\beta^2\ln\left(\frac{cp\nu}{(1-p)\beta}\right)<0 \tag{7}$$

as the criterion for when to issue an alert. To abbreviate formulas, we introduce the notation l for the logarithm that occurs here and D for the discriminant.

$$\begin{aligned} l :&= \ln\left(\frac{cp\nu}{(1-p)\beta}\right), \\ D :&= \nu^4-(\nu^2-\beta^2)\nu^2(1-2\beta^2 l) \\ &= \nu^2[\nu^2-\nu^2+\beta^2+(\nu^2-\beta^2)2\beta^2 l] = \nu^2\beta^2[1+2(\nu^2-\beta^2)l]. \end{aligned}$$

With these notations, we turn to the remaining cases.

Case 2: $\beta<\nu$ and $D<0$.

Because the discriminant is negative, the quadratic expression on the left side of (7) never vanishes, so its sign is the same for all x. For large x it's positive, as $\beta<\nu$, and so it's positive for all x. Thus, the inequality (7) never holds. In Case 2, therefore, the expert should never issue an alert.

Intuitively, this means that, no matter how much an observed value x looks like what one would expect in the bad state, even if x is the very center 1 of the distribution of X given B, it is nevertheless more reasonable to assume that this x arose as a random fluctuation in the normal state.

In the boundary case, where $\beta<\nu$ and $D=0$, the result is the same except that there is one value of x for which the left side of (7) equals 0. For this x, the expert can decide arbitrarily whether to issue an alert or not; the expected cost is the same either way. For simplicity, we include this boundary case in Case 2, with the instructions to never issue an alert.

Case 3: $\beta<\nu$ and $D>0$.

Now the quadratic function on the left side of (7) has two real zeros, namely

$$t_\pm=\frac{\nu^2\pm\sqrt{D}}{\nu^2-\beta^2}.$$

The inequality (7) holds if and only if x is between these two (since the quadratic function is still positive for large positive or negative values of x). So the expert should issue an alert if and only if $t_-<x<t_+$.

Had we allowed $\beta\neq\nu$ in Section 6, where we assumed that the appropriate action for our expert is to issue an alert if and only if X exceeds some threshold, the calculation would have produced the threshold t_-. We omitted that calculation because, as we explained above and as we now see in detail, the assumption of a single threshold is unjustified. Nevertheless, t_- is somewhat relevant as a threshold; it indicates the value of X *immediately* above which an alert should be issued. But not all X values above t_- should produce alerts; there is a second threshold, t_+, above which alerts should not be issued.

Case 4: $\beta > \nu$ and $D < 0$

Now the left side of (7) is negative for large x and therefore, since it never vanishes, for all x. That is, inequality (7) always holds, and therefore an alert should always be issued.

Case 5: $\beta > \nu$ and $D > 0$

There are two roots $t_\pm$ exactly as in Case 3, but the quadratic expression on the left of (7) is negative for large positive and negative x and is positive when x is between the two roots. Therefore, an alert should be issued if and only if either $x < t_-$ or $x > t_+$.

In view of the importance of the sign of D in the case distinctions above, it may be useful to reformulate the inequality $D < 0$ as follows. First, directly from the definition of D, we obtain, since ν^2 and β^2 are positive, that $D < 0$ if and only if $1 + 2(\nu^2 - \beta^2)l < 0$. If $\nu > \beta$ then this becomes, in view of the definition of l,

$$\frac{1}{\nu^2 - \beta^2} < -2 \ln \left(\frac{cp\nu}{(1-p)\beta} \right) = \ln \left(\left(\frac{(1-p)\beta}{cp\nu} \right)^2 \right),$$

or, equivalently,

$$\exp \left(\frac{1}{\nu^2 - \beta^2} \right) < \left(\frac{(1-p)\beta}{cp\nu} \right)^2 .$$

If, on the other hand, $\nu < \beta$, then when we divide by $\nu^2 - \beta^2$, the inequalities get reversed, and so $D < 0$ is equivalent to

$$\exp \left(\frac{1}{\nu^2 - \beta^2} \right) > \left(\frac{(1-p)\beta}{cp\nu} \right)^2 .$$

References

[1] Arcsight, http://www.arcsight.com/, viewed Mar 14, 2010

[2] Microsoft Forefront homepage, http://www.microsoft.com/forefront/en/us/default.aspx, viewed May 9, 2010.

[3] MITRE, http://www.mitre.com/, viewed on May 14, 2010.

[4] SC Magazine, "Best SIM/SIEM solution," March 2, 2010, http://www.scmagazineus.com/best-simsiem-solution/article/164132/, viewed May 14, 2010.

Math. Dept, University of Michigan, Ann Arbor, MI 48109, USA
E-mail address: ablass@umich.edu

Microsoft, One Microsoft Way, Redmond, WA 98052, USA
E-mail address: gurevich@microsoft.com

Microsoft, One Microsoft Way, Redmond, WA 98052, USA
E-mail address: efimh@microsoft.com