September 2005

Features | In Print

A Few Good Metrics

Printer Friendly

Email this story

Information security metrics don't have to rely on heavy-duty math to be effective, but they also don't have to be dumbed down to red, yellow, green. Here are five smart measurements—and effective ways to present them.

By Scott Berinato
CSO

Metrics have a bad rep. Mention metrics to a CISO and immediately his thoughts may well turn to sigmas, standard deviations and, probably, probability. To many, metrics equals statistics.

There's no denying that proven economic principles can—and should—be applied to information security investments. At the same time, a bumper crop of valuable metrics exist that don't require classes on Nobel Prize-winning theories or a working knowledge of the Greek alphabet. You've actually already sowed the seeds of these less dense but equally valuable metrics. They're sitting in your log files, on your network, in the brains of your business unit managers, just waiting to be harvested. You won't need computational prowess to exploit this crop's value, just some legwork and—this is key—the most effective presentation tools.

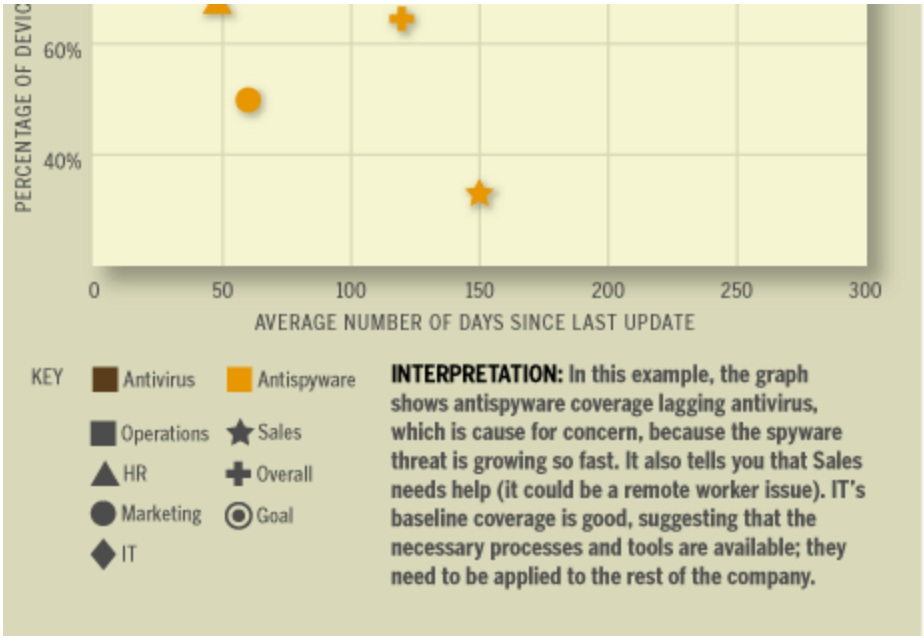
Here we discuss five such metrics, along with some ways to present them visually, as imagined by Andrew Jaquith. Jaquith is a cofounder of the consultancy @stake (which was bought in 2004 by Symantec) and a protégé of infosecurity guru Dan Geer. At @stake he invented a popular analytic methodology that is used to evaluate a client's risk in its application portfolio. He's since left Symantec and joined The Yankee Group. More recently he started Securitymetrics.org, a website open to all security professionals for sharing, contributing and advancing the use of metrics in information security. He's also writing a book, Security Metrics, due out later this year.

Jaquith has sharp, sometimes contrarian opinions on what makes a good metric and what makes for good presentation of metrics. For example, he thinks annual loss expectancy (ALE), a tool used to measure potential losses against probability of losses occurring over time, is useless, because in infosecurity, the L and the E in ALE are wild guesses. Quoting Geer, he says, "The numbers are too poor even to lie with."

He also thinks CISOs are too apt to dumb down visual representations of metrics for their executive counterparts, mistaking simplicity for clarity. He holds a particular grudge against the overuse of the "red, yellow, green" representation of metrics to signify high, medium and low numbers. "A CEO's favorite visualization of metrics is a stock chart, a 1-inch square that contains a month's worth of opening and closing prices, a trend line and several other indicators. Maybe 50 or more data points right there. Don't tell me they can't handle complex data. They can, as long as it's presented well."

By no means does Jaquith (or CSO for that matter) think these five metrics are the final word on infosecurity. Quite the contrary, they're a starting point, relatively easy to ascertain and hopefully smart enough to get CISOs thinking about finding other metrics like these, out in the vast fields of data, waiting to be reaped.

Metric 1: Baseline Defenses Coverage (Antivirus, Antispyware, Firewall, and so on)



This is a measurement of how well you are protecting your enterprise against the most basic information security threats. Your coverage of devices by these security tools should be in the range of 94 percent to 98 percent. Less than 90 percent coverage may be cause for concern. You can repeat the network scan at regular intervals to see if coverage is slipping or holding steady. If in one quarter you've got 96 percent antivirus coverage, and it's 91 percent two quarters later, you may need more formalized protocols for introducing devices to the network or a better way to introduce defenses to devices. In some cases, a drop may stir you to think about working with IT to centralize and unify the process by which devices and security software are introduced to the network. An added benefit: By looking at security coverage, you're also auditing your network and most likely discovering devices the network doesn't know about. "At any given time, your network management software doesn't know about 30 percent of the IP addresses on your network," says Jaquith, because either they were brought online ad hoc or they're transient.

How to get it: Run network scans and canvass departments to find as many devices and their network IP addresses as you can. Then check those devices' IP addresses against the IP addresses in the log files of your antivirus, antispyware, IDS, firewall and other security products to find out how many IP addresses aren't covered by your basic defenses.

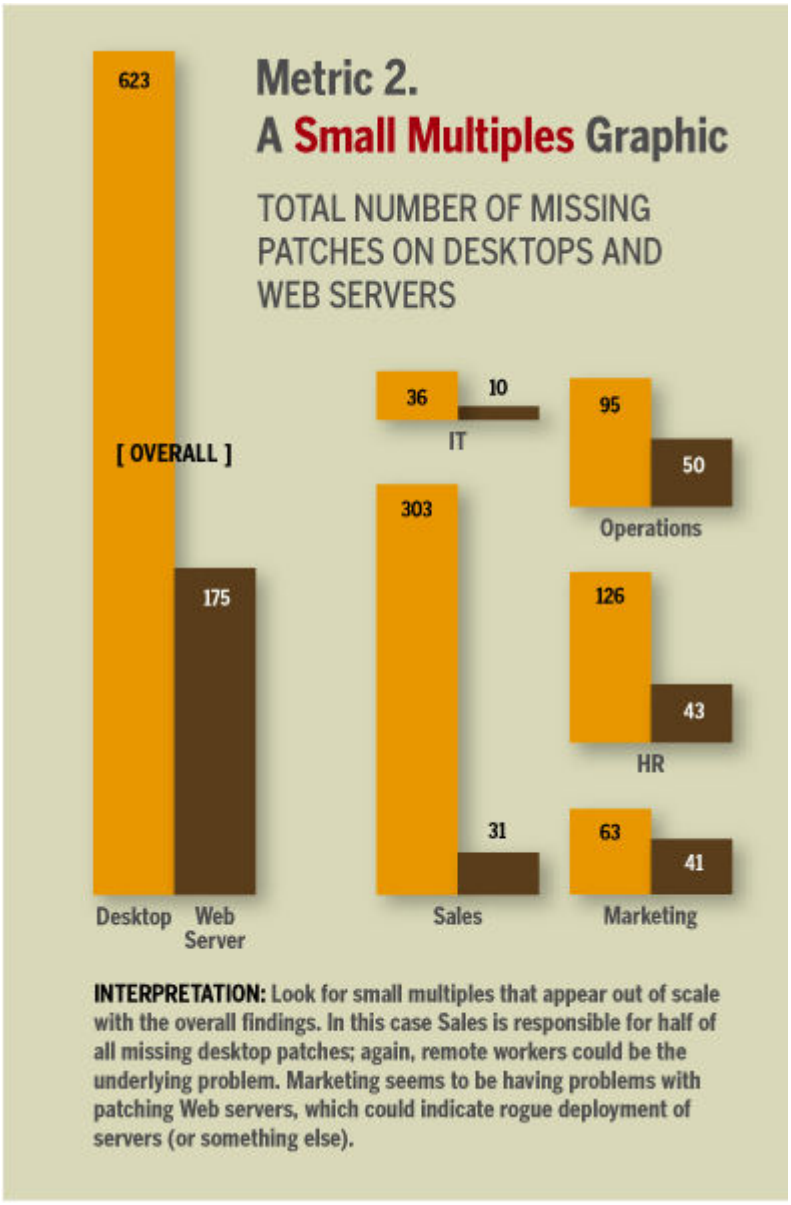
Expressed as: Usually a percentage. (For example, 88 percent coverage of devices by antivirus software, 71 percent coverage of devices by antispyware and so forth.)

Not good for: Shouldn't be used for answering the question "How secure am I?" Maximum coverage, while an important baseline, is too narrow in scope to give any sort of overall idea of your security profile. Also, probably not yet ready to include cell phones, BlackBerrys and other personal devices, because those devices are often transient and not always the property of the company, even if they connect to the company.

Try these advanced versions: You can parse coverage percentages according to several secondary variables. For example, percentage coverage by class of device (for instance, 98 percent antivirus coverage of desktops, 87 percent of servers) or by business unit or geography (for instance, 92 percent antispyware coverage of desktops in operations, 83 percent of desktops in marketing) will help uncover tendencies of certain types of infrastructure, people or offices to miss security coverage. In addition, it's a good idea to add a time variable: Average age of antivirus definitions (or antispyware or firewall rules and so on). That is, 98 percent antivirus coverage of manufacturing servers is useless if the average age of the virus definitions on manufacturing's servers is 335 days. A star company, Jaquith says, will have 95 percent of their desktops covered by antivirus software with virus definitions less than three days old.

One possible visualization: Baseline defenses can be effectively presented with a "you are here" (YAH) graphic. A YAH needs a benchmark—in this case it's the company's overall coverage. After that, a business unit, geography or other variable can be plotted against the benchmark. This creates an easy-to-see graph of who or what is close to "normal" and will suggest where most attention needs to go. YAHs are an essential benchmarking tool. The word "you" should appear many times on one graphic. Remember, executives aren't scared of complexity as long as it's clear. Here's an example: plotting the percentages of five business units' antivirus and antispyware coverage and the time of their last update against a companywide benchmark.

Metric 2: Patch Latency



Patch latency is the time between a patch's release and your successful deployment of that patch. This is an indicator of a company's patching discipline and ability to react to exploits, "especially in widely distributed companies with many business units," according to Jaquith. As with basic coverage metrics, patch latency stats may show machines with lots of missing patches or machines with outdated patches, which might point to the need for centralized patch management or process improvements. At any rate, through accurate patch latency mapping, you can discover the proverbial low-hanging fruit by identifying the machines that might be the most vulnerable to attack.

A Good Metric Must:

1. **Be consistently measured.** The criteria must be objective and repeatable.
2. **Be cheap to gather.** Using automated tools (such as scanning software or

1. password crackers) helps.
2. **Contain units of measure.** Time, dollars or some numerical scale should be included—not just, say, "green," "yellow" or "red" risks.
3. **Be expressed as a number.** Give the results as a percentage, ratio or some other kind of actual measurement. Don't give subjective opinions such as "low risk" or "high priority."

Source: Andrew Jaquith

How to get it: Run a patch management scan on all devices to discover which patches are missing from each machine. Cross-reference those missing patches with a patch clearinghouse service and obtain data on 1. the criticality of each missing patch and 2. when the patches were introduced, to determine how long each missing patch has been available.

Expressed as: Averages. (For example, servers averaged four missing patches per machine. Missing patches on desktops were on average 25 days old.)

Not good for: Companies in the middle of regression testing of patch packages, such as the ones Microsoft releases one Tuesday every month. You should wait to measure patch latency until after regression testing is done and take into account the time testing requires when plotting the information. The metrics might also get skewed by mission-critical systems that have low exposure to the outside world and run so well that you don't patch them for fear of disrupting ops. "There are lots of systems not really open to attack where you say, 'It runs, don't touch it,'" says Jaquith. "You'll have to make a value judgment [on patch latency] in those cases."

Try these advanced metrics: As with baseline coverage, you can analyze patch latency by business unit, geography or class of device. Another interesting way to look at patch latency statistics is to match your average latency to the average latency of exploits. Say your production servers average 36 days on missing patches' latency, but similar exploits were launched an average of 22 days after a patch was made available. Well, then you have a problem. One other potentially useful way to approach patch latency is to map a patch to its percent coverage over time. Take any important patch and determine its coverage across your network after one day, three days, five days, 10 days and so on.

One possible visualization: For data where you can sum up the results, such as total number of missing patches, a "small multiples" graphic works well. With small multiples you present the overall findings (the whole) as a bar to the left. To the right, you place bars that are pieces making up the whole bar on the left. This presentation will downplay the overall findings in favor of the individual pieces. One key in small multiples graphing is to keep the scale consistent between the whole and the parts. This example plots total number of missing patches for the top and bottom quartiles of devices (the best and worst performers). Then it breaks down by business unit who's contributing to the missing patches.

Metric 3: Password Strength

This metric offers simple risk reduction by sifting out bad passwords and making them harder to break, and finding potential weak spots where key systems use default passwords. Password cracking can also be a powerful demonstration tool with executives who themselves have weak passwords. By demonstrating to them in person how quickly you can break their password, you will improve your lines of communication with them and their understanding of your role.

How to get it: Using commonly available password cracking programs, attempt to break into systems with weak passwords. Go about this methodically, first attacking desktops, then servers or admin systems. Or go by business unit. You should classify your devices and spend more time attempting to break the passwords to the more important systems. "If it's a game of capture the flag," Jaquith says, "the flag is with the domain controller, so you want stronger access control there, obviously."

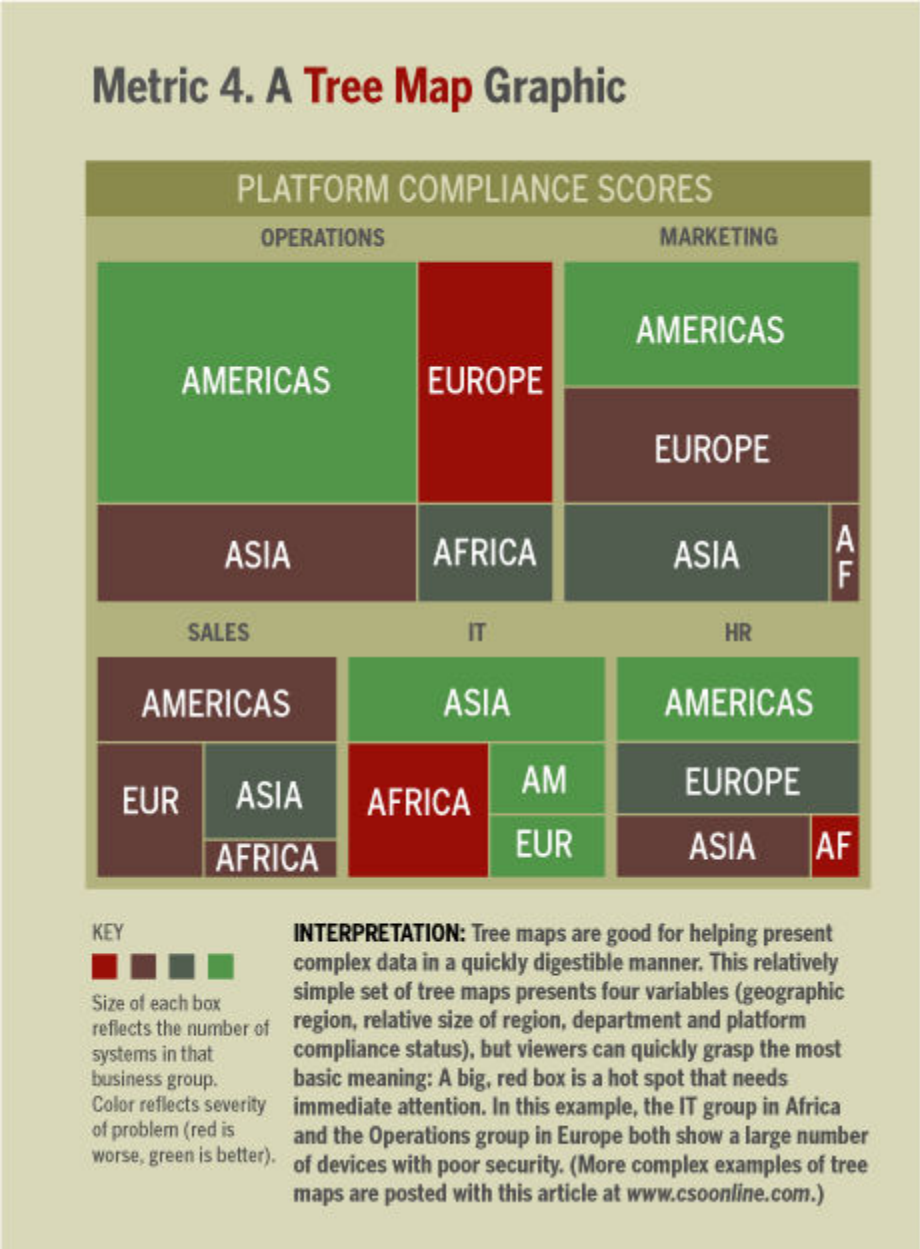
Expressed as: Length of time or average length of time required to break passwords. (For example, admin systems averaged 12 hours to crack.) Can be combined with a percentage for a workgroup view (for example, 20 percent of accounts in business unit cracked in less than 10 minutes). Is your password subject to a lunchtime attack? That is, can it be cracked in the 45 minutes you are away from your desk to nosh?

Not good for: User admonishment, judgment. The point of this exercise is not to punish offending users, but to improve your security. Skip the public floggings and just quietly make sure employees stop using their mother's maiden name for access.

Try this: Use password cracking as an awareness-program audit tool. Set up two groups (maybe business units). Give one group password training. The other group is a control; it doesn't get training. After several months and password resets, try to crack the passwords in both groups to see if the training led to better passwords.

One possible visualization: Both YAH and small multiples graphics could work with this metric. (See the graphics for Metric 1 and Metric 2.)

Metric 4: Platform Compliance Scores



Widely available tools, such as the Center for Internet Security (CIS) scoring toolset, can run tests against systems to find out if your hardware meets best-practice standards such as those set by CIS. The software tools take minutes to run, and test such things as whether ports are left unnecessarily open, machines are indiscriminately shared, default permissions are left on, and other basic but often overlooked security lapses. The scoring system is usually simple, and given how quickly the assessments run, CISOs can in short order get a good picture of how "hardened" their hardware is by business unit, by location or by any other variable they please.

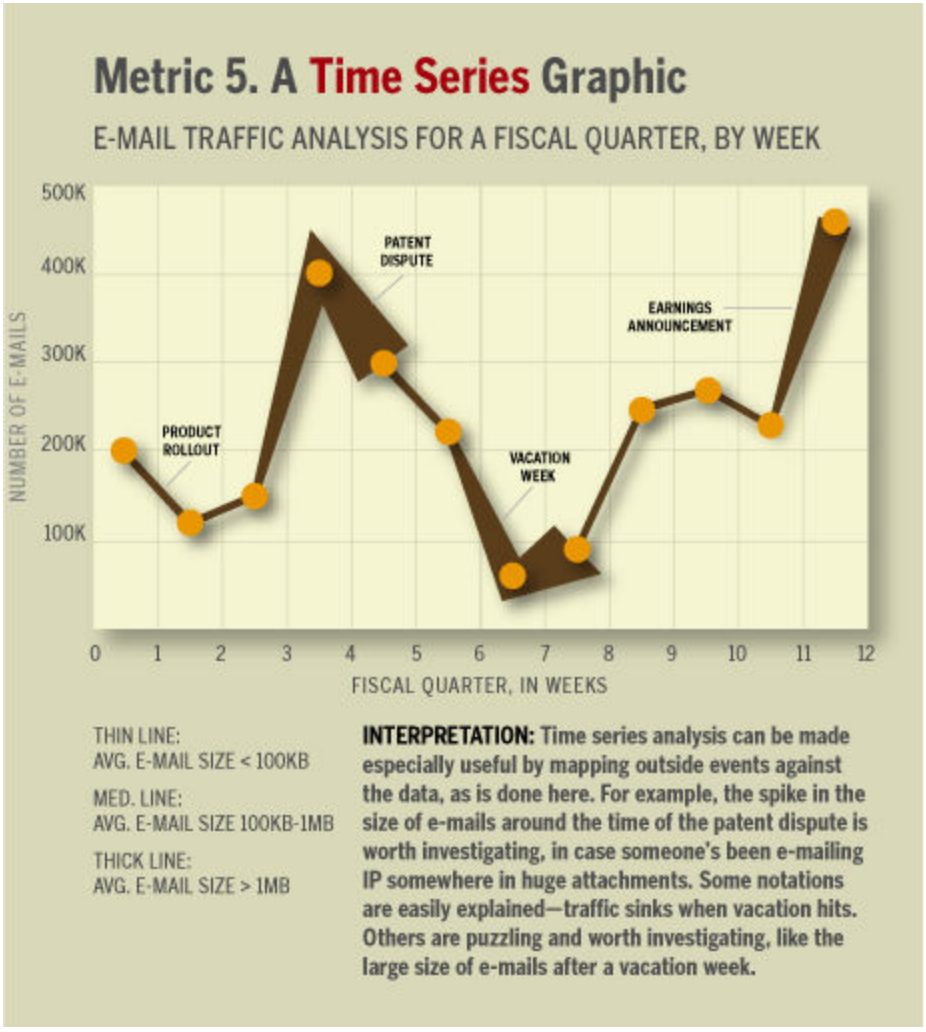
Expressed as: Usually a score from 0 to 10, with 10 being the best. Best-in-class, hardened workstations score a 9 or a 10, according to Jaquith. He says this metric is far more rigorous than standard questionnaires that ask if you're using antivirus software or not. "I ran the benchmark against the default build of a machine with Windows XP Service Pack 2, a personal firewall and antivirus protection, and it scored a zero!" Jaquith notes.

Not good for: Auditing, comprehensive risk assessment or penetration testing. While a benchmark like this may be used to support those advanced security functions, it shouldn't replace them. But if you conduct a penetration test after you've benchmarked yourself, chances are the pen test will go more smoothly.

Try this: Use benchmarking in hardware procurement or integration services negotiations, demanding configurations that meet some minimum score. Also demand baseline scores from partners or others who connect to your network.

One possible visualization: An overall score here is simple to do: It's a number between 1 and 10. To supplement that, consider a tree map. Tree maps use color and space in a field to show "hot spots" and "cool spots" in your data. They are not meant for precision; rather they're a streamlined way to present complex data. They're "moody." They give you a feel for where your problems are most intense. In the case of platform-compliance scores, for instance, you could map the different elements of your benchmark test and assign each element a color based on how risky it is and a size based on how often it was left exposed. Be warned, tree maps are not easy to do. But when done right, they can have instant visual impact.

Metric 5: Legitimate E-Mail Traffic Analysis



Legitimate e-mail traffic analysis is a family of metrics including incoming and outgoing traffic volume, incoming and outgoing traffic size, and traffic flow between your company and others. There are any number of ways to parse this data; mapping the communication flow between your company and your competitors may alert you to an employee divulging intellectual property, for example. The fascination to this point has been with comparing the amount of good and junk e-mail that companies are receiving (typically it's about 20 percent good and 80 percent junk). Such metrics can be disturbing, but Jaquith argues they're also relatively useless. By monitoring legitimate e-mail flow over time, you can learn where to set alarm points. At least one financial services company has benchmarked its e-mail flow to the point that it knows to flag traffic when e-mail size exceeds several megabytes and when a certain number go out in a certain span of time.

How to get it: First shed all the spam and other junk e-mail from the population of e-mails that you intend to analyze. Then parse the legitimate e-mails every which way you can.

Not good for: Employee monitoring. Content surveillance is a different beast. In certain cases you may flag questionable content or monitor for it, if there's a previous reason to do this, but traffic analysis metrics aren't concerned with content except as it's related to the size of e-mails. A spike in large e-mails leaving the company and flowing to competitors may signal IP theft.

Added benefit: An investigations group can watch e-mail flow during an open investigation, say, when IP theft is suspected.

Try this: Monitor legitimate e-mail flow over time. CISOs can actually begin to predict the size and shape of spikes in traffic flow by correlating them with events such as an earnings conference call. You can also mine data after unexpected events to see how they affect traffic and then alter security plans to best address those changes in e-mail flow.

One possible visualization: Traffic analysis is suited well to a time series graphic. Time series simply means that the X axis delineates some unit of time over which something happens. In this case, you could map the number of e-mails sent and their average size (by varying the thickness of your bar) over, say, three months. As with any time line, explain spikes, dips or other aberrations with events that correlate to them.

Metric 6: Application Risk Index

How to get it: Build a risk indexing tool to measure risks in your top business applications. The tool should ask questions about the risks in the application, with certain answers corresponding to a certain risk value. Those risks are added together to create an overall risk score.

Expressed as: A score, or temperature, or other scale for which the higher the number, the higher the exposure to risk. Could also be a series of scores for different areas of risk (for example, business impact score of 10 out of 16, compliance score of 3 out of 16, and other risks score of 7 out of 16).

Industry benchmark: None exist. Even though the scores will be based on observable facts about your applications (such as, is it customer facing? Does it include identity management? Is it subject to regulatory review?). This is the most subjective metric on the list, because you or someone else puts the initial values on the risks in the survey instrument. For example, it might be a fact that your application is customer-facing, but does that merit two risk points or four?

Good for: Prioritizing your plans for reducing risk in key applications—homegrown or commercial. By scoring all of your top applications with a consistent set of criteria, you'll be able to see where the most risk lies and make decisions on what risks to mitigate.

Not good for: Actuarial or legal action. The point of this exercise is for internal use only as a way to gauge your risks, but the results are probably not scientific enough to help set insurance rates or defend yourself in court.

Added benefit: A simple index like this is a good way to introduce risk analysis into information security (if it's not already used) because it follows the principles of risk management without getting too deeply into statistics.

Try this: With your industry consortia, set up an industrywide group to use the same scorecard and create industrywide application risk benchmarks to share (confidentially, of course). One industry can reduce risk for everyone in the sector by comparing risk profiles on similar tools. (Everyone in retail, for example, uses retail point-of-sale systems and faces similar application risks.)

One possible visualization: Two-by-two grids could be used here to map your applications and help suggest a course of action. Two-by-twos break risk and impact into four quadrants: low risk/low impact, low risk/high impact, high risk/low impact, high risk/high impact. A good way to use these familiar boxes is to label each box with a course of action and then plot your data in the boxes. What you're doing is facilitating decision-making by constraining the number of possible courses of action to four. If you need to get things done, use two-by-two grids to push executives into decision making. CIO
