Security Metrics: A Solution in Search of a Problem

The multifaceted aspects of security programs become clearer with the creation and collection of appropriate metrics

By **Joel Rosenblatt**

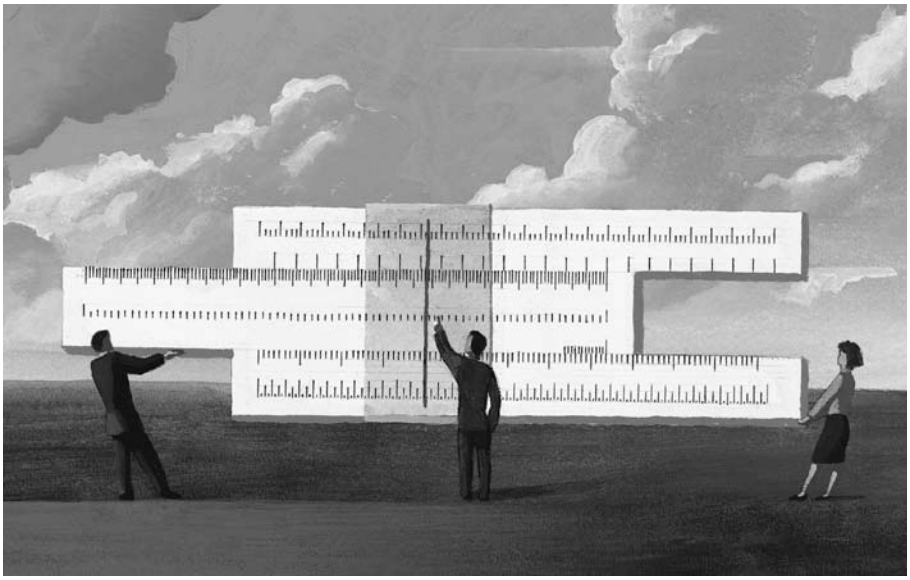
Computer security is one of the most complicated and challenging fields in technology today. As soon as you think you have it figured out, the "bad guys" change all of the rules and you have to start over. A security metrics program provides a major benefit: looking at the metrics on a regular basis offers early clues to changes in attack patterns or environmental factors that may require changes in security strategy.

There are some key rules to observe when collecting metrics:

- Metrics should be collected and generated on a regular basis (ideally, automatically).
- Metrics should be consistent and objective.

The term "security metrics" loosely translates to the standard measurement of computer security. The process of obtaining security metrics both fascinated and confused me. How can you measure something that doesn't happen? As an optimist—for a security guy—I figured that eventually all the problems would be fixed and there would be nothing to measure. Silly me. Once you get sucked into the wonderful world of computer security, you quickly realize several things:

- There is no such thing as perfect security.
- Almost everything done with a computer carries a security or risk component.



- Once you get past the simple stuff (viruses, malware, spam, configuration problems, and updates), then you get to deal with the really hard problems of people and politics.
- Without some form of metrics, it is all but impossible to determine if your security solutions are effective and where improvements are needed.
- In many cases, metrics might make the difference in getting funding approved.

Building a Security Program to Include Metrics

Assume you have been asked to build a security program mostly from scratch, as I was. I really didn't know much about metrics, but—being a geek—I figured that along the way I would count things and make pretty charts and graphs because it was the cool thing to do.

To begin, I came up with the following list of security issues:

- Policy and compliance
- Network and machine monitoring
- Outreach and education
- Legal compliance: DMCA, PCI, FERPA, etc. (see the sidebar)
- ID: authorization and authentication
- Asset protection
- Privacy

Each item in my list turned into multiple projects, and over the past seven years each project has developed into a security program. Your list might differ from mine, of course, and over time lists will change because computer security is dynamic and must respond to external forces (the bad guys). You can keep

moving in the right direction if you remember to think of the big picture and ask yourself "What problem am I trying to solve?"

Starting with my simple list, let's build out each item and look at metrics that can enhance the security program.

Policy and Compliance

I strongly believe that the starting point for any security program is policy. Just having the policies written is not enough, however—you need to have those policies endorsed by the highest level of administration possible. The policy-creation and adoption process dovetails with the governance process, which is another key to a successful program. Governance, or more simply the reporting structure of the computer security department, plays a significant role in the authority of the group. Unless the highest ranking security person obtains the support of key executives, your policies—no matter how well written—will end up as paper tigers.

Policy metrics can start simply. When my group started, we had one computer policy (more like a suggestion); now we have 22 policies that have passed through the vetting process. One metric established by our senior executive vice president, who we call the CEO, was to develop well-written, understandable policies accepted by the university policy committee. I am not personally fond of counting or making a list as a metric, but in this case, it worked. The lesson: provide your executives with the metrics they request.

Compliance, on the other hand, is a much more complicated issue. Collecting compliance metrics can be very tricky. In some cases, adherence to the policy can be enforced using technology; for example, Columbia University has a policy called "Network Bandwidth Quotas" that states:

> To maintain network performance, CUIT [Columbia University Information Technology] has implemented an automated network bandwidth quota system, described below. Individual computers may be limited in either the inbound or outbound direction. Limits are imposed only on off-campus traffic to or from the Internet. Traffic on the university's internal network is not restricted in any way. Internet2 traffic is also unrestricted.

The section of the policy covering the university's automated system of port-agnostic bandwidth control states:

> Each host computer on the Columbia network is assigned two quotas. One quota affects outbound usage, i.e., data sent to the Internet. The second affects inbound usage, i.e., data downloaded from the Internet. A host exceeding either limit in a given hour will have its bandwidth in that direction restricted to a lower rate for the remainder of the hour and the hour following if excessive bandwidth use continues.

Technology enforces the policy, automating compliance and making the metrics very easy to collect—the number of machines violating the policy. We know at all times how many machines are in the penalty box (lowered network speed), and we provide a mechanism for users to check their own systems if they feel that their network performance is bad.

The university's "Desktop and Laptop Security Policy" is much more complicated, with 15 separate points. Because of Columbia's decentralized environment, the central IT organization does not manage most of these systems. To collect information on compliance requires looking at the bigger picture. The problem we wanted to solve was compromised computer systems. We do have a way of finding compromised machines (described in the section Network and Machine Monitoring), but we could also say that those who don't follow the "Desktop and Laptop Security Policy" face having their machines compromised, hence, the number of compromised systems will correlate to compliance with this policy. Of course, this correlation will not be 100 percent, but it gives us a good picture of the state of compliance. Sometimes, security metrics are derived by correlating different factors.

Relevant Legislation

DMCA (Digital Millennium Copyright Act), http://www.copyright.gov/legislation/dmca.pdf

FERPA (Family Educational Rights and Privacy Act), http://www.ed.gov/offices/OM/fpco/ferpa/index.html

PCI (Personal Credit Information/Industry), https://www.pcisecuritystandards.org/

HIPPA (Health Insurance Portability and Accountability Act), http://www.hhs.gov/ocr/hipaa/

GLB (Gramm-Leach-Bliley) Act, http://www.ftc.gov/privacy/privacyinitiatives/glbact.html

These policies and others can be found in our Policy Library: http://www.columbia.edu/cu/administration/policylibrary/category/computing_technology.html.

Network and Machine Monitoring

This area of computer security often becomes the alpha and the omega of the computer security group. While I agree that it is important, it's just one of the cogs in the computer security wheel. This component often takes on a life of its own because the sheer number of data points is overwhelming. Columbia, for example, has approximately 65,000 nodes on its network and around 35,000 active MAC addresses. These machines generate a massive amount of network traffic.

Many companies will come in and sell you a solution for your computer security problems, but many of the solutions generate only a slightly smaller amount of data than the raw network. The next group of companies sells a product that takes the output of the first layer and tells you what your problems *really* are. A third group of companies sells yet another layer of products to finally produce actionable items. I am not criticizing these products; many of them do exactly

what they claim. The problem I wanted to solve, however, was how to find compromised computers and get them off the network without having one of the IT staff sitting at a console playing Whack-a-Mole.

Columbia's IT group has developed software we call PAIRS (Point of contact And Incident Response System) that, in an automated way, does exactly what I wanted—it identifies compromised systems and takes them off the network automatically. Any security program needs to find solutions to its specific problems that also produce metrics offering insight into how well the solutions offered on viruses, malware, random clicking, and all the other evils of the Internet are working. One metric without real meaning but much quoted is how many attacks were launched against the network. Who cares, really? The only important attack is one that succeeds and compromises a system. Why measure and place importance on something with no direct effect on security?

Outreach and Education

In outreach and education, the rubber meets the road for computer security. After all the "easy" stuff is done, you have to deal with users. A primary objection to computer security requirements is that many of the rules and policies seem designed to prevent computer users from doing their work. Asking users to run anti-virus and anti-malware solutions makes sense and doesn't appear too onerous, but when you start telling users to change their 20-character passwords every 30 days and not to reuse a password *ever* or even something close to a previous password—you can see why some of them might get a little edgy.

One of the most important steps in outreach and education is to obtain buy-in, which generates goodwill. An excellent example of the importance of education is shown by the response to the latest highly targeted ploy to gather IDs and passwords, called Spear Phishing. Columbia University, along with many other schools, received convincing e-mails asking for individual's credentials, as shown in Figure 1.

Figure 1

Spear Phishing Message

VERIFY YOUR COLUMBIA EMAIL ACCOUNT NOW

Dear columbia Email Account Owner,

This message is from columbia messaging center to all columbia email account owners. We are currently upgrading our data base and e-mail account center. We are deleting all columbia email account to create more space for new accounts.

To prevent your account from closing you will have to update it below so that we will know that it's a presently used account.

We have been sending this notice to all our columbia email account owners and this is the last notice/verification exercise.

CONFIRM YOUR EMAIL IDENTITY BELOW

Email Username : ....
EMAIL Password : .....
Date of Birth : .....
Country or Territory : .....

Warning!!! Account owner that refuses to update his or her account within Seven days of receiving this warning will lose his or her account permanently.

Thank you for using columbia.edu!
Warning Code:VX2G99AAJ
Thanks,
Columbia.edu Team
COLUMBIA.EDU BETA

This phishing e-mail message includes several obvious mistakes, but it was persuasive enough that several people responded by providing their personal information. The e-mail group had built some tools to pull information from the university's e-mail logs, capturing the e-mail address or UNI (University Network ID, which is also the e-mail address) of anyone who responds to a phishing message and invalidating the password they use to log into university systems to prevent misuse of their account by spammers. We have also found these accounts being sold and then used to access our library systems.

With these tools, we can also build e-mail lists to send warnings to anyone who receives one of the false messages. After an educational e-mail is sent to the general population, the interesting metric measures whether the number of people who respond to the phish goes down as a percentage of the number of people who received it.

Education metrics are tricky to collect, especially in an environment like a university where the population is always changing. The best you can do is keep teaching the same lessons and hope the new students pay attention. An easy metric to collect—and many times the only one—is the number of people who watched the security presentation. I am not convinced this metric has any real significance—watching a security presentation does not mean people absorbed the information. You should collect this metric anyway because it will satisfy some compliance requirements.

Legal Compliance

Legal compliance is a huge can of worms that often gets dumped on the security group. Many of the various regulations and laws are not really IT problems, but because they have a data component, they become a computer security issue.

One of the biggest time sinks has proved to be DMCA compliance. The Digital Millennium Copyright Act requires a provider of Internet service, or ISP (which a lot of universities resemble), to be prepared to locate the owner of an IP address that is violating copyright law and either ask them to stop (a take-down notice) or provide identifying information to the complainant in the case of

a subpoena. Many universities (including Columbia) operate their network using a DHCP service, which assigns IP addresses on demand. The implication of this is that without an infrastructure in place that can turn an IP address and a timestamp into a person, each one of these notices can consume 15 minutes or more of a staff person's time digging through various logs. This law actually requires some tracking because there is a requirement to escalate the penalty applied each time the same person receives a take-down notice.

I have found that once you start tracking something, metrics develop automatically. Columbia's IT group started tracking DCMA notices in 2003 and can display everything from trends to details of a single case. This data collection helped us recognize when the number of notices started going up sharply, prompting us to create a fully automated system to process notices. While the automated system did not affect the number of notices received, it allows us to process them with no additional staff time.

In general, metrics showing legal compliance will be of interest to auditors and senior executives. Generally simple to collect and produce, such metrics will prove useful when questions arise about why something happened.

ID: Authorization and Authentication

A university's ID system represents the major gatekeeper in the organization. Access to almost every major asset is controlled by an ID and password. This mechanism has become less reliable, with keyloggers and phishing scams threatening the integrity of passwords. In response, many organizations are moving toward adding a second factor for authentication, with the assumption that the combination of something you know and something you have is more secure than either alone. We are implementing a token-based system, for example. This works by adding a second factor to the logins of very sensitive IDs. When a system administrator wants to log into a root account, in addition to the ID and password (something you know), the system will prompt for a number that appears on the token (something you have) and that changes every minute.

The authorization piece of this puzzle is usually harder than authentication, although it is fairly straightforward—either you know the magic words or you don't. The tricky part is that now that I know who you are, what are you allowed to do? Metrics in this area always come up in audits: "How many people do you have with root access to the servers and why do you have so many?" Another typical question is, "When was the last time everyone with access to the financial system recertified?" Questions like these constantly arise because they are on every audit checklist ever made. Keeping metrics like these up-to-date will prevent major headaches.

One of the interesting metrics we keep is the number of different ISPs a person uses to log in. By studying these numbers and looking at the geographical distribution of the ISPs, we discovered that logins by the same user from more than seven ISPs in 48 hours usually indicates a compromised password. That allows us to contact the person and get them to change their password.

Asset Protection

When you get right down to it, the big-picture purpose of security is asset protection. The definition of an asset might be flexible, but if an asset is lost, damaged, or stolen, you have a problem. In computer security, assets are mostly data. The information can range from PII (personally identifying information) and IP (intellectual property) to PCI (personal credit information/industry). (If the asset's name has a "P" and an "I" in it, it is really bad if it gets lost or stolen.) Metrics in this area are the same as those in the "Legal Compliance" section—or they are the type reported in the newspaper. A failure in any area of security results in a jump in the asset-protection metric.

Privacy

I included privacy on my security list for two reasons. First, I work at a university, where the individual's privacy is an important concept. Second, I did not want to create an environment that sacrifices privacy for security. I needed that reminder to keep me from doing the easy things in the name of security (look at everything, block everything, report everything) and instead build a security system that only looks for bad behavior without looking at content. The metrics I gather in this area are thank-you notes and the surprised looks I get from people when I explain that we provide security without poking into everyone's business. I realize that the metrics mentioned here completely break my second rule on good metrics requiring that they be measured consistently and be objective, but then, many people will say that privacy has no place on a security list.

Metrics Are the Answer

Metrics are the key to understanding what is going on and whether or not things are working properly. Computer security often takes place in firefighting mode: somebody discovers a problem, IT staff implement a "temporary" fix, and then, on to the next fire. Without a good metrics program, there is no way to discover if the fix made things better or worse the next time a fire breaks out.

A good way to approach the issue is to build metrics into the process, sort of like building security into the whole IT process. This way, you won't have to go back and reengineer a system to find out what's going on.

Another significant benefit to collecting good metrics is that they make asking for money easier. When you have numbers to back up a request, it becomes harder to deny funding, especially if you can correlate those numbers to a security project and then estimate how an increase will improve those numbers.

Metrics are the answer. Now go and find your questions. *e*

Joel Rosenblatt (joel@columbia.edu) is the Manager of Computer and Network Security for Columbia University in New York and Chair of the Security Metrics Project Team of the EDUCAUSE/Internet2 Computer and Network Security Task Force.