Security Metrics: An Introduction and Literature Review*

George O.M. Yee
Dept. of Systems and Computer Engineering
Carleton University, Ottawa, Canada K1S 5B6
gmyee@sce.carleton.ca

Abstract

This chapter provides an introduction to and a literature review for security metrics. It begins by describing the need for security metrics, followed by a discussion of the nature of security metrics, including what makes a good security metric, what security metrics have been used in the past, and how security metrics can be scientifically based. This is followed by suggestions for starting a security metrics program within an organization and a discussion of the feasibility of an intelligent security dashboard driven by metrics. The chapter concludes with a literature review that summarizes security metrics publications (including research papers) where one could obtain more information.

1. Introduction

We live in a world where attacks against computer systems are a fact of life. Barely a day goes by without headlines appearing about the latest systems compromised, in terms of web sites being brought down by DDoS (Distributed Denial of Service) attacks, or the loss of privacy due to malware infected computers. In response, systems owners have invested more and more funds into various forms of protection mechanisms (e.g. firewalls, biometrics, data encryption) as well as improve the design of systems and work flows to be more resistant against these attacks. However, the return on these investments or the subsequent increase in the level of security, has been largely unknown, leading to the following dilemmas:

- How much more do I need to spend to be "safe" from attack?
- Will the changes made to my software to improve security be effective?
- Are my company's work flows or processes sufficiently secure?
- How will adding the third party software component impact security?
- How can legislation requiring certain levels of security be enforced if the level of security is unknown?

---

* DOI of published version: https://doi.org/10.1016/B978-0-12-803843-7.00032-6

These questions can be answered if there was some way to measure the level of security of a cyber system. Properly defined, effective, *security metrics* appear to be the solution. An analogy can be made with performance engineering, where there is a need to know if the performance of a computer system is sufficient to satisfy users when under a certain processing load. A performance analysis to obtain *performance metrics* such as throughput and service time is an effective approach to knowing if the performance is sufficient, and if not, identify the location of performance bottlenecks. Similarly, in the security domain, it should be possible to perform a security analysis of a computer system to obtain security metrics that would indicate if the system is secure from various forms of attack, and if not, where and what are the vulnerabilities.

Security metrics do exist and are being used. However, most of them are far from giving the results described above. They can be ineffective and not meaningful. For example, a traditional metric is the number of viruses detected and eliminated, say at a firewall. This metric is not meaningful since a) it says nothing about the number of viruses that were not detected and got through, and b) why are there so many viruses trying to get through in the first place [1]. Rather, a security metric should (adapted from [1]):

- Measure quantities that are meaningful for establishing the security posture of a computer system or of an organization,
- Be reproducible,
- Be objective and unbiased,
- Be able to measure a progression toward a goal over time.

More details on what makes a good security metric (or what makes a bad security metric) are given in Section 3.

The above qualities of a good security metric also describe certain metrics that have a basis in science, such as the throughput metric in performance engineering. Throughput measures the number of jobs completed per second by a computing system. It is a quantitative measure of a computing system based on the laws of physics. It is also meaningful, reproducible, objective and unbiased, and can measure the improving performance of a system over time toward a throughput goal. This leads to the question of what sort of scientific framework could give rise to such science-based metrics. This will be further discussed in Section 3.

An interesting practical application of security metrics is in determining the security posture of a computer system in real time. One envisages a security dashboard that displays security metrics associated with vulnerability points. The dashboard would display security alerts corresponding to strategic subsets and groupings of the metrics that exceed critical thresholds. Security officers monitoring the dashboard would then be able to take remedial action, upon which the security alerts would be replaced by “system back to normal” messages. One can further envisage the dashboard as having intelligence sufficient to recommend courses of remedial action appropriate to particular security

alerts. An assessment of the possibility of achieving this vision based on commercial systems now available will be given in Section 5.

The objectives of this chapter are:

- Introduce the reader with little or no background in security metrics to the topic,
- Discuss the nature of security metrics,
- Explain how one can get started in using security metrics, and
- Show the reader where to find further information by presenting the results of a literature search (including research papers) on security metrics.

The rest of this chapter is organized as follows. Section 2 further elaborates on the need for security metrics. Section 3 discusses the nature of security metrics, including the need to put security metrics on a scientific basis and what that means. Section 4 gives an overview on how one could get started using security metrics. Section 5 provides an assessment of the feasibility of achieving a security dash board that is driven by security metrics. Section 6 presents the results of a literature search on security metrics, and Section 7 gives conclusions.

2. Why Security Metrics?

Over 100 years ago, Lord Kelvin, the distinguished British mathematical physicist and engineer observed that measurement is vital to knowledge and to continued progress in physical science. Lord Kelvin stated that: “To measure is to know,” and “If you can not measure it, you can not improve it.”

This observation is evident in many activities in our modern world. One has only to recall school exams, and at the time of this writing, the 2012 London Olympics. Indeed, the Olympics is fraught with measurement, and an athlete depends on measuring his or her progress in order to improve in his or her chosen sport. Returning to the topic of computer systems performance that was mentioned in the Introduction, measuring the performance of a computer system prior to its deployment in a highly demanding environment is the only way to know in advance if it will perform adequately once deployed. As well, in order to improve the performance, one has to know where the performance bottlenecks lie, which can only be found by measuring it. Thus, it appears that Lord Kelvin's words are applicable to many modern activities, as they were during his time.

The above observations on measurements are also relevant to the information technology (IT) world. Organizations and consumers rely on information technology to deliver goods and services. Information technology heads are challenged to use computer systems effectively and to protect them from security threats and risks. There have been many past efforts to develop security measurements that could help organizations make informed decisions about the design of systems, the selection of controls, and the efficiency of security operations. But the development of standardized metrics for computer system security has been a difficult challenge, and past efforts have only met with partial success (see Section 3).

Security metrics are needed to:

- Provide a quantitative and objective basis for security operations,
- Support decision making, e.g. is investment in more security controls needed?
- Support software quality since software security is part of software quality,
- Support the reliable maintenance of security operations, e.g. how often do users need to change their passwords?
- Support the incremental improvement of software's resistance to attacks.

This is by no means an exhaustive list but it does serve to illustrate the usefulness of good security metrics.

In addition, [2] provides the following main uses of security metrics:

- "Strategic support – Assessments of security properties can be used to aid different kinds of decision making, such as program planning, resource allocation, and product and service selection.
- Quality assurance – Security metrics can be used during the software development lifecycle to eliminate vulnerabilities, particularly during code production, by performing functions such as measuring adherence to secure coding standards, identifying likely vulnerabilities that may exist, and tracking and analyzing security flaws that are eventually discovered.
- Tactical oversight – Monitoring and reporting of the security status or posture of an IT system can be carried out to determine compliance with security requirements (e.g., policy, procedures, and regulations), gauge the effectiveness of security controls and manage risk, provide a basis for trend analysis, and identify specific areas for improvement."

It should be clear from the above that security metrics play very important roles in today's computing systems.

3. The Nature of Security Metrics

3.1 Traditional Security Metrics

3.1.1 The Organization Perspective

Traditional security metrics have not been developed in a rigorous systematic manner [1] and have, in fact, given rise to false impressions of security, leading to unsafe or ineffective implementations of security controls.
Security metrics as quantifiers of the effectiveness of the organization's security practices over time have always been difficult to define and evaluate. How can an organization determine whether it is secure? This can only be truly determined by the organization undergoing a real crisis. Yet, such a crisis is exactly what the security controls were designed to prevent. Thus, a security metric is given the task of measuring the security of the organization, which is discernable only when the organization is in a security crisis and thereby defeats the whole point of having the metric in the first place. An important requirement of a security metric then is that it is capable of measuring the security level of the organization at any time and not only when the organization is in a crisis.

The bottom line is that management needs some way to measure the organization's security level. Organizations need to ask:

- How many *security controls* does it take to be "safe"?
- *When* does the organization know it is "safe"?
- How can the cost of new security controls be *justified*?
- Is the organization getting *good value* for its money?
- How can the organization *compare* its security posture with other similar organizations and with best practices?

Traditionally, these questions are answered using risk assessment. In particular, the answers relate to how much residual risk the organization is willing to accept, depending on business needs and budget limits. However, risk management may be a red herring and not necessarily lead to stronger security.

Consider, for example, a risk assessment that lists a number of threats along with the cost to mitigate each threat or risk. Some items on the list would be very low cost while other items will be very expensive (see Figure 1). Often, management may choose to purchase the most security controls for the least amount of money, possibly ignoring the most expensive controls. Management assumes that buying more inexpensive controls is better value than buying fewer expensive ones. Thus, there is a tendency to buy large numbers of less expensive security tools and avoid the more expensive, less glamorous controls. The latter tends to be organizational in nature, requiring cultural change (e.g. disaster recovery plan) rather than specific self-contained solutions (such as firewalls and intrusion detection systems (IDSs). In carrying out such a purchase policy, management believes it is buying more security for less money.

However, how can it be said that more security is purchased? What is the increase in security achieved by each additional purchase and how can the organization determine this? How do we even know that the purchases have been made in the correct order? Perhaps the organization is being exposed to *more* risk because of the haphazard way in which the security controls were obtained?

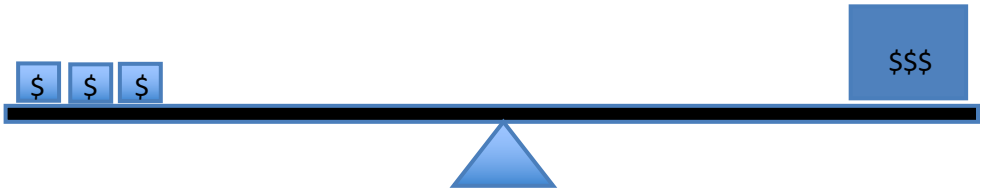

Figure 1 - Is buying more inexpensive security controls better than buying fewer but more expensive ones?

Security metrics programs need to be built from the ground up to allow for new approaches to these traditional security metrics problems. A more systematic and even scientifically based approach (see below) to security metrics can:

- Come up with reproducible and justifiable measurements
- Objectively measure something of value to the organization
- Determine real progress in security posture
- Apply to a broad range of organizations while producing consistent results

- Fix the order in which security controls should be applied
- Help determine the resources needed to apply to a security program

3.1.2 Issues with Definition and Application

A measurement has to be combined with time to be called a metric. Further, a metric by itself is not going to save an organization that is in trouble with its security posture. It is necessary to think through and analyze the true meaning of the metric. The art is to develop security metrics that are simple and provide useful management information corresponding to security-related objectives. The metrics have to inform the organization by demonstrating progress.

Clearly, a security metric needs to count or measure *something*. But count or measure what? How can security be measured? Consider the following security metrics (in italics) [1] that are common but problematic in that they ignore the attendant core issues:

- *Number of computer viruses or malware detected.* The intended use of this metric is to measure the effectiveness of the anti-malware controls. However, it fails to consider why so much malware is getting through in the first place, and what about the malware that got through but were undetected!
- *Number of security incidents and investigations.* This presumably measures the effectiveness of security events monitoring. However, it does not consider the thresholds at which the incident or investigation is triggered. Nor does it consider their causes, e.g. are incidents triggered due to flaws in work processes?
- *Cost of security breaches.* This metric is intended to measure the true business loss due to security failures. However, it ignores the residual risks that the organization chose to live with. Further, it does not delineate between costs incurred for normal operations despite safeguards that were in place, or costs that result from abnormal conditions such as crises or disasters.
- *Resources assigned to security functions.* The intended use of this metric is to measure the true business cost of running a security program. However, it fails to consider the causes of high cost such as the possibility that people may be less productive due to inefficient tools or procedures.
- *Compliance with security policy.* This metric is intended to measure the level of compliance or adhesion to security goals. However, it can be misleading since it fails to consider how compliance is related to effectiveness, the order

> of compliance which may be relevant, and what happens once compliance is achieved – will the security program be complete then?

A security metric for an organization should measure the quality of the organization's security program and be capable of showing progress. The latter is important so that one can know if new investments in improving security are making any difference. None of the above metrics really possess these capabilities.

The following illustration [1] shows why incident totals are unreliable:

> "Imagine a small town with one police officer. He does no other police work other than patrolling the highway with a radar gun, pulling over hundreds of speeders. Now imagine a large town with many police officers. They do not use radar guns and have caught very few speeders but have a large defensive driving program and an active anti-drunk-driving program. Is the small town safer than the large town? The count of speeders is only as good as the sensing mechanism, but that number has no depth to it. What about the small town with nonspeeders who are drunk—are they not potentially more dangerous?"

Consider the anti-malware tool in light of this illustration. The fact that the tool detected a large amount of malware probably makes the security team feel good that the tool works and that so much malware has been caught. However, this says very little about the organization's security level. Why is so much malware present in the first place? How much malware remain undetected? What does it say about the quality of the security program? In fact, just the opposite may be what we want – the tool doesn't detect any malware because malware is unable to penetrate the security controls that are in place!

Time spent on a security-related task (e.g. software patching, security incident investigation) is often used as a security metric. This may be useful from a project management point of view in order to ensure that there is sufficient time to complete a project, but it is next to useless as a measure of security. This is because more time spent does not necessarily translate into better security. For example, the additional time may have been due to inefficient procedures or work processes. Moreover, such procedures may have been responsible for triggering the incidents that called for the investigation (security-related task) in the first place!

The business cost of a security incident is another unreliable security metric. This metric comes with the built-in assumption that something bad has happened but what if that something has already been considered as acceptable to the organization in terms of the

residual risk it is willing to live with? On the other hand, the security incident may in fact have been caused by poor security practice. Or, another possibility is that one of these two possibilities happened but the incident management was so good that the costs were kept to a minimum. How can these three possibilities be separated? This metric may measure the effectiveness of incident response in terms of minimizing the business cost but it may not be a good rating of the quality of the organization's security practices since it cannot distinguish whether or not the costs were due to poor security practices.

Security metrics should have the following **ideal characteristics** (in italics, adapted from [1] and mentioned in the Introduction):

- *Measure quantities that are meaningful for establishing the security posture of a computer system or of an organization.* As discussed above, some traditional security metrics fail to measure the security level, which should be their first objective.
- *Have results that are reproducible.* This means that the value of the security metric should be the same as the original value if re-evaluated by another party, given that the factors upon which the metric is evaluated remain the same. It will be seen below that this is a key requirement of being "scientifically-based".
- *Be objective and unbiased.* This requirement is self explanatory.
- *Be able to measure a progression toward a goal over time.* As mentioned above, it is important to be able to measure over time whether or not investments in improving security have in fact improved security.

Unfortunately, traditional security metrics found in practice lack one or more of the above characteristics. Such metrics were haphazard and opportunistic, in the sense that whatever measures were readily available were taken up and reported. Moving even beyond the ideal characteristics of a security metric above, security metrics should be "scientifically based". The meaning of this will be discussed in the next section.

3.2 Scientifically Based Security Metrics

It would be very useful to have computer security based on science, similar to computer systems performance being based on the science of physics. For then security could be analyzed, just as performance is analyzed, and security metrics could be systematically derived and predicted, just as performance metrics are derived and predicted. Unfortunately, it is not known at the time of this writing that security can be based on science, due to at least two fundamental problems, quoted from [3] as follows:

> Problem 1: "The first is the difference between mathematical abstractions [for security metrics] and real implementations. The gap between theoretical cryptography results and practical cryptanalysis illustrates this: although no one has found a fast factoring algorithm, RSA implementations are regularly broken because of side channels (such as timing and power consumption), poor random-number generation, insecure key storage, message formats and padding, and programming bugs[1]. For system security, the gap between models simple enough to use for metrics and actual implementations is even larger. To make progress, we need metrics that work on more concrete models of actual systems, or ways to build systems that refine models without introducing security vulnerabilities." In other words, security metrics must be simple enough to understand, but by so being, they cannot capture enough of what is going on in a computer system to accurately reflect security levels.
>
> Problem 2: "The second problem is that it seems unlikely that we can reason well about adversary creativity. This argues for metrics that assume that adversaries can efficiently search the entire space of possible actions. Perhaps we can develop complexity metrics that analyze that space and the maximum effectiveness of different search strategies." In other words, to understand the security of a computer system, it is necessary to understand attacker creativity in creating new attacks. However, at the present time, we are not very good at capturing and predicting this behaviour. This then calls for metrics that assume that the attacker is capable of launching every possible attack, which are difficult to design since we would need to know every possible attack.

However, the above assumes basing security on what [3] calls the "weak sense of science" and the "strong sense of science" (explained below). It is possible to base security on the "methodological sense of science". Let us examine these senses of science more closely.

There are three interpretations of science that can be considered for security metrics [3], as follows:

---

[1] J.P. Degabriele, K.G. Paterson, and G.J. Watson, "Provable Security in the Real World," *IEEE Security & Privacy*, vol. 9, no. 3, 2011, pp. 33–41.

- Weak sense: science as the generalization and systematization of knowledge – for example, consider the body of knowledge within physics, where laws and descriptions of behaviour have been systematized and interwoven into an integral whole.

- Strong sense: science used to develop laws with which predictions can be made – for example, in physics, laws of motion have been developed and used to predict the future position of planets.

- Methodological sense: science used for research by forming hypotheses and proving or disproving the hypotheses with experiments. The results of the experiments must be confirmable by independent experimenters. Hence the experiments must be repeatable and yield the same results. This is the embodiment of the scientific method and established sciences have in fact been built up in this fashion.

Basing security metrics on the weak sense of science is currently at best unknown as there has not been sufficient research to show that it is even possible outside of perhaps a highly specialized sub-area of security. Basing the metrics on the strong sense is likewise untenable since it is more likely that laws can be developed only after systematization of the knowledge, i.e. the strong sense is more likely after the weak sense has been established. This leaves the methodological sense, which appears to be a possible basis upon which a framework for computer security metrics can be developed. Such a development however is currently undergoing research and is beyond the scope of this chapter. However, the reader is invited to consult this author's papers on this subject in the near future.

Assuming that scientifically based security metrics are available, the following question now arises: What type of security metrics should be used - those based on science or those having the ideal characteristics (Section 3.1.2)? It is recommended that both types can be used. The choice can be made based on practicality and system requirements. Scientifically based metrics would be more rigorous and therefore require more work to define and evaluate. Perhaps scientifically based metrics can be reserved for critical systems such as those involving public utilities, public safety, hospital systems, and defence applications, which have more stringent requirements for security.

4. Getting Started with Security Metrics

As a security professional in your organization, you would like to start a security metrics program and start using security metrics. But how do you start such a program? This section is partly based on [4] and consists of some essential items that must first be in place, after which suggestions are given (in no particular order) to guide you in starting a security metrics program in your organization.

Essentials

The following items are essential for any successful security metrics program:

- Design your security metrics based on the ideal characteristics (Section 3.1.2).
- Make sure you collect and store all the data needed for the metrics as specified in your design of the metrics. This may sometimes be automated by programming the system to automatically output the data needed to a repository.
- Obtain a picture of how metrics-minded your organization is through discussion with management and co-workers. Ensure that everyone understands and buys in to processes that include metrics, which will be critical when you collect the data needed for the metrics.

Suggestions for Security Metrics Design

- **Base your security metrics on the ideal characteristics.** Strive to base your security metrics on the ideal characteristics described in Section 3.1.2.
- **Use your service level agreement to guide your metrics design.** Your organization's security policies or service level agreements will point to areas for which security metrics may be needed. Use them to refine your measurement targets. By so doing, you will be relating what you measure to what is expected of you, and the value of your results will be more immediately recognized by your organization (especially upper management).
- **Start with basic measurements, understand them, then expand.** Start with a basic metric that is easy to understand and then work to make that metric more useful or replace it with a better one that you've discovered along the way. Be

well organized and prioritize your efforts so that you can build up and maintain a portfolio of metrics that have maximal value.

Suggestions for organization and management

- **Form a team of stakeholders as early as possible.** As soon as possible, contact and put together a cross-functional team of metrics-minded people to build the plan around collecting, analyzing, reporting, interpreting and responding to security metrics. Work with the experts who understand the data, and the management who will need to champion changes throughout the organization.
- **Define your metrics data repository** – a central agreed location for storing trusted data required for metrics evaluation. This will help to create confidence and trust in the data, and possibly save you much time defending the data later on, should questions arise over its reliability.
- **Be consistent in using your metrics.** Don't spend a month on observing and analyzing and then move on if nothing is found. Consistent, steady vigilance is the key to identifying trends or variances - erratic monitoring and analysis will mislead you into a false sense of security and reduce your ability to continuously reflect and refine based on known patterns.
- **Be ready to change based on your findings.** A common behavioral pattern is to take a finding, create a counter-measure around it, and then never look back. Be intellectually and ethically honest when you make new discoveries, particularly if they show a need to change an established rule, position, or policy. Learn to be comfortable with the idea that you may learn something new which will require a policy or process change.
- **Be open to incorporating expertise and data from others.** Since attacks are often not limited to one area, you may need to integrate data from other system components into your analysis. In this case, ask for input from teams who know these other components better. They may shed light on interdependencies or relationships that are critical to better metric design. Leverage the findings established together with these teams to extract any support that may be needed from managers.
- **Test your analytics.** Carry out a "Metrics Penetration Test" (MPT), which is a test to determine if your analytic procedures will zero in on the behaviors you are trying to isolate. For example, have a colleague attempt to crack a login password at an odd hour of the day to see if your "Unusual Login Attempts" metric triggers the flags you expect to see. Incorporate the results from these MPTs in operational reviews to continue evolving and maturing your analytic methodologies.

Additional advice for establishing a security metrics program can be found in books such as Jaquith [5] and Hayden [6]. In addition, the NIST publication [7] provides guidelines for establishing measures for assessing security controls and other security-related activities.

5. Metrics in Action – Towards an Intelligent Security Dashboard

SIEM (Security Information and Event Management) describes security dashboard-like commercial applications or services that are widely available from security vendors. According to Wikipedia[2], SIEM technology "provides real-time analysis of security alerts generated by network hardware and applications. SIEM solutions come as software, appliances or managed services, and are also used to log security data and generate reports for compliance purposes". The same Wikipedia page also gives the following list of SIEM capabilities:

- **"Data Aggregation:** SIEM/LM (log management) solutions aggregate data from many sources, including network, security, servers, databases, applications, providing the ability to consolidate monitored data to help avoid missing crucial events.
- **Correlation:** looks for common attributes, and links events together into meaningful bundles. This technology provides the ability to perform a variety of correlation techniques to integrate different sources, in order to turn data into useful information.
- **Alerting:** the automated analysis of correlated events and production of alerts, to notify recipients of immediate issues.
- **Dashboards:** SIEM/LM tools take event data and turn it into informational charts to assist in seeing patterns, or identifying activity that is not forming a standard pattern.
- **Compliance:** SIEM applications can be employed to automate the gathering of compliance data, producing reports that adapt to existing security, governance and auditing processes.

---

[2] "Security information and event management", accessed Mar. 27, 2012 at: http://en.wikipedia.org/wiki/Security_information_and_event_management

- **Retention:** SIEM/SIM [Security Information Management] solutions employ long-term storage of historical data to facilitate correlation of data over time, and to provide the retention necessary for compliance requirements."

The question at hand is: Is it possible to use SIEM technology as a base upon which to build an intelligent security dashboard that displays security alerts and responds to the alerts by either suggesting corrective action, automatically taking corrective action (depending on the action) or both? It appears from the above list of SIEM capabilities, that the "alerting" and "dashboard" capabilities map directly to the security dashboard's display of security alerts, and that the "data aggregation" and "correlation" capabilities map directly to the security dashboard's suggesting or taking of corrective action. Thus it does seem viable to use SIEM technology as the base to build the security dashboard. Additional research is required to determine what security metrics should be used to trigger the security alerts. As well, research is needed to know how to construct the security dashboard's corrective action engine, which may be built using artificial intelligence techniques. The construction of an intelligent security dashboard based on SIEM technology does indeed appear feasible.

6. Security Metrics in the Literature

This section presents the results of a literature search using the following sources: the Internet, the IEEE Xplore and ACM Digital Library databases, and the home pages of university researchers. The publications found can be categorized as concerning:

- The nature of security metrics
- Measuring the security of a computer system
- Managing IT Security Risks
- Measuring the effectiveness of a security process

“Measuring the security of a computer system” differs from the other categories in that i) it considers the component make-up of the computer system, ii) it is about the use of scientific tools (e.g. modeling) to measure the security, and iii) it only measures the security of the computer system and not other aspects such as operational security or security process. Note that a publication may appear in more than one category.

It should also be noted that a publication may not fit neatly within a particular category. This is natural since security metrics may have a different meaning for different people, and different authors approach the subject from their own varied backgrounds and environments. The above categories do, however, help to group the papers in a broad sense. Of course, the security metrics coverage within these publications is limited by the data sources searched, since not all research is published or published in these sources. Nevertheless, one can say that given the dominance of IEEE and ACM publication repositories over other sources, this coverage is reasonably high.

The following subsections divide up the publications into tables according to each of the categories mentioned above. References to the publications in each table are of the form "Table n [i, j, k, ...]", for publications i, j, k, ... in Table n.

6.1 The Nature of Security Metrics

Table 1 lists the publications in this category, which treats questions such as “how is a security metric defined?” (e.g. Table 1 [3, 4, 7, 8]), “what makes a good security metric?” (e.g. Table 1 [4, 6, 7, 8]), “what is a security metrics taxonomy?” (e.g. Table 1 [3]), “are security metrics scientifically based?” (e.g. Table 1 [2, 5]), “what are good areas for security metrics research?” (e.g. Table 1 [5]), and “who are the U.S. industrial and government players in security metrics, and what security metrics initiatives have they undertaken?" (Table 1 [1]). Publications Table 1 [2, 4, 5, 6] elaborate the ideas of Section 3 by discussing what makes a good security metric and a scientific basis for security metrics.

Table 1– The nature of security metrics

No.	Publication	Summary
1	“Measuring Cyber Security and Information Assurance”, IATAC SOAR, May 8, 2009.	Broad coverage of US, including laws, standards, best practices, government programs, industry initiatives, measurable data, tools and technologies.
2	S. Stolfo, S. Bellovin, D. Evans, “Measuring Security”, IEEE Security & Privacy, May/June 2011.	Discusses a scientific basis for security and security metrics with examples and ideas for research; focuses on security of a computer system.
3	R. Savoia, “Towards a Taxonomy for Information Security Metrics”, QoP’07, 2007.	Proposes a high-level security metrics taxonomy for ICT product companies; gives an example of a security metrics taxonomy.
4	D. Chapin, S. Akridge, “How Can Security Be Measured?”, Information Systems Control Journal, Vol. 2, 2005.	Discusses what is wrong with traditional security metrics, giving characteristics of good metrics; discusses security maturity models with examples.
5	Wayne Jansen, “Directions in Security Metrics Research”, NIST, April 2009.	Overviews security measurement and proposes possible research areas such as formal models of security measurement

		and artificial intelligence techniques.
6	O. Saydjari, “Is Risk a Good Security Metric?”, Panel, Proceedings of QoP’06, 2006.	Succinct descriptions of risk as a security metric, alternative security metrics, and what makes a good metric.
7	Andrew Jaquith, Security Metrics: Replacing Fear, Uncertainty, and Doubt, Addison-Wesley, 2007.	Discusses security metrics for enterprise application; security metrics applied broadly, not only to computing systems but also to all sorts of enterprise processes.
8	Lance Hayden, IT Security Metrics: A Practical Framework for Measuring Security & Protecting Data, McGraw-Hill Osborne Media, June 2010.	Similar to the Jaquith book in its focus on the enterprise; covers security metrics in terms of effectiveness, implementation, operations, compliance, costs, people, organizations; includes 4 case studies.

6.2 Measuring the Security of a Computer System

Table 2 lists the publications in this category, which contains the largest number of publications of all the categories. These papers propose a range of techniques that use metrics to evaluate the security of a computer system and mostly apply to the software. The techniques involve the computer system's components and generally do not treat supporting areas such as software development practice, security operations, or security process.

The papers here are based on various frameworks, and serve to illustrate the earlier discussion of a scientific framework for computer system security metrics.

Table 2– Measuring the security of a computer system

No.	Publication	Summary
1	S. Stolfo, S. Bellovin, D. Evans, “Measuring Security”, IEEE Security & Privacy, May/June 2011.	Discusses scientific basis for security and security metrics with examples and ideas for research; focuses on security of a

		computer system.
2	Wayne Jansen, "Directions in Security Metrics Research", NIST, April 2009.	Overviews security measurement and proposes possible research areas such as formal models of security measurement and artificial intelligence techniques.
3	M. Howard, J. Pincus, J. Wing, "Measuring Relative Attack Surfaces", in *Computer Security in the 21st Century*, Springer, pp. 109-137, 2005.	Proposes "attack surfaces" as a measure of one system's security relative to another; an attack surface is described along 3 dimensions: targets and enablers, channels and protocols, and access rights.
4	M. Howard, "Attack Surface: Mitigate Security Risks by Minimizing the Code You Expose to Untrusted Users", 2004.	Practical advice to developers on how to reduce the attack surface of their code; based on actual Microsoft products such as Windows XP and Windows Server 2003.
5	L. Wang, A. Singhal, S. Jajodia, "Toward Measuring Network Security Using Attack Graphs", Proceedings of QoP'07, 2007.	Proposes a framework for assessing the security of a network based on attack graphs or access paths for attack, e.g. given two networks, if one has more paths of attack than the other, it is the less secure of the two; references Table 2 [8] for attack resistance.
6	S. Noel, L. Wang, A. Singhal, S. Jajodia, "Measuring security risks of networks using attack graphs," International Journal of Next-Generation Computing, Vol. 1, No. 1, pp 113-123, 2010.	An expanded version of Table 2 [5]; provides a method for quantitatively analyzing the security of a network using attack graphs; the attack graphs are first populated with known vulnerabilities and likelihoods of exploitation and then "exercised" to obtain a metric of the overall security and risks of the network.
7	L. Wang, S. Jajodia, A. Singhal, S.	Proposes "k-zero day safety" as a security

	Noel, "k-Zero day safety: Measuring the security risk of networks against unknown attacks," Proc. 15th European Symposium on Research in Computer Security (ESORICS 2010), Springer-Verlag Lecture Notes in Computer Science (LNCS), Vol. 6345, 20-22 September, pages 573-587, 2010.	metric that counts the number of unknown zero day vulnerabilities that would be required to compromise a network asset, regardless of what those vulnerabilities might be. The metric is defined in terms of an abstract model of networks and attacks. Algorithms for computing the metric are included.
8	L. Wang, A. Singhal, S. Jajodia, "Measuring the overall security of network configurations using attack graphs," Proc.21st Annual IFIP WG 11.3 Working Conference on Data and Applications Security (DBSec 2007), Springer Lecture Notes in Computer Science, Vol. 4602, Steve Barker and Gail-Joon Ahn, eds., Redondo Beach, CA, pages 98-112, 2007.	Proposes an attack graph-based attack resistance metric for measuring the relative security of network configurations; incorporates two composition operators for computing the cumulative attack resistance from given individual resistances and accounts for the dependency between individual attack resistances; referenced by Table 2 [5] for attack resistance.
9	L. Wang, T. Islam, T. Long, A. Singhal, S. Jajodia, "An Attack Graph-Based Probabilistic Security Metric", Proc. 22nd Annual IFIP WG 11.3 Working Conference on Data and Applications Security (DBSEC 2008) , Springer-Verlag Lecture Notes in Computer Science (LNCS), Vol. 5094, pages 283-296, 2008.	Proposes an attack graph-based metric for the security of a network that incorporates the likelihood of potential multi-step attacks combining multiple vulnerabilities in order to reach the attack goal; the definition of the metric is claimed to have an intuitive and meaningful interpretation that is useful in real world decision making.
10	A. Singhal, X. Ou, "Techniques for Enterprise Network Security Metrics", Fifth Cyber Security and Information Intelligence Research Workshop (CSIIRW '09), Knoxville, TN, USA, 2009.	Presents an attack graph-based method for evaluating the security of a network based on likelihood of attack (similar to Table 2 [9]); stresses the derivation of the metric based on composition of component vulnerabilities whose security levels are already known. This is a short paper with

		accompanying slides.
11	M. Frigault, L. Wang, A. Singhal, S. Jajodia, "Measuring Network Security Using Dynamic Bayesian Network", Proceedings of QoP'08, 2008.	A Dynamic Bayesian Network (DBN) model is used to capture the dynamic nature of vulnerabilities that change over time. An attack graph is converted to a DBN by applying conditional probabilities to the nodes, calculated from the Common Vulnerabilities Scoring System (CVSS) [8]. The security of the network is calculated from the probabilities of the attacks being successful.
12	M. Frigault, L. Wang, "Measuring Network Security Using Bayesian Network-Based Attack Graphs", Annual IEEE International Computer Software and Applications Conference, 2008.	Proposes measuring network security using Bayesian network-based attack graphs so that relationships such as exploiting one vulnerability makes another vulnerability easier to exploit may be captured; differs from Table 2 [11] in that Table 2 [11] uses dynamic Bayesian networks whereas this paper uses regular Bayesian networks; Table 2 [11] refers to Table 2 [12] but not the other way around.
13	L. Krautsevich, F. Martinelli, A. Yautsiukhin, "Formal approach to security metrics. What does 'more secure' mean for you?", Proceedings of ECSA 2010, 2010.	Initial proposal and analysis of a number of mathematically based definitions of security metrics such as "number of attacks", "minimal cost of attack", "maximal probability of attack", and even "attack surface" from Table 2 [3].
14	C. Wang, W. Wulf, "Towards a Framework for Security Measurement", Proceedings of 20th National Information Systems Security Conference, 1997.	Proposes an initial framework for estimating the security strength of a system by decomposing the system into its security sensitive components and assigning security scores to each component; aggregate the component scores to get an estimate for the security

		strength of the system.
15	P. Halonen, K. Hätönen, “Towards holistic security management through coherent measuring”, Proceedings of ECSA 2010, 2010.	Discusses the problems of applying security metrics to telecommunication systems; compares security metric taxonomies, and discusses the need for security impact metrics; presents a broad view of security metrics.
16	D. Mellado, E. Fernández-Medina, M. Piattini, “A Comparison of Software Design Security Metrics”, Proceedings of ECSA 2010, 2010.	A survey of various security metrics and standards that may be applicable to software design; compares the relevance of the various approaches to security properties such as authenticity and confidentiality.
17	J. Wang, H. Wang, M. Guo, M. Xia, “Security Metrics for Software Systems”, Proceedings of ACMSE ‘09, 2009.	Presents a security metrics formulation in terms of weaknesses and vulnerabilities, rated by CVSS scores for CVE (Common Vulnerabilities and Exposures [9]) names; does not show how one would determine such scores for a brand new piece of software; not clear how the final security metric can be used to improve security.
18	R. Scandariato, B. De Win, W. Joosen, “Towards a Measuring Framework for Security Properties of Software”, Proceedings of QoP ‘06, 2006.	Claims that software has security properties that can be measured, much like it has maintainability properties such as complexity; proposes a number of software security properties along with corresponding metrics.
19	O. Saydjari, “Is Risk a Good Security Metric?”, Panel, Proceedings of QoP’06, 2006.	Succinct descriptions of risk as a security metric, alternative security metrics, and what makes a good metric.
20	Z. Dwaikat, F. Parisi-Presicce, “Risky Trust: Risk-Based Analysis of Software Systems”, Proceedings	Proposes an approach to evaluate the security of a software system in development; security requirements are

	of SESS'05, 2005.	derived and a method is given for evaluating the likelihood of requirements violation based on the individual risks of system components.
21	Y. Liu, I. Traore, A.M. Hoole, "A Service-oriented Framework for Quantitative Security Analysis of Software Architectures", Proceedings of 2008 IEEE Asia-Pacific Services Computing Conference, 2008.	Proposes a User System Interaction Effect (USIE) model for systematically deriving and analyzing security concerns in service oriented architectures. The model is claimed to provide a foundation for software services security metrics and one such metric is defined and illustrated.
22	Y. Liu, I. Traore, "Properties for Security Measures of Software Products", Applied Mathematics & Information Sciences, I(2), pp. 129-156, 2007.	Describes and formalizes properties that characterize security-related internal software attributes; these properties form a framework that can be used to rigorously identify and evaluate new security metrics; this framework is claimed to be sound but not complete; the properties are claimed to be necessary but not sufficient conditions for good security metrics.
23	Y. Liu, I. Traore, "UML-based Security Measures of Software Products", Proceedings of International Workshop on Methodologies for Pervasive and Embedded Software (MOMPES'04), 2004.	Proposes the USIE model mentioned above for Table 2 [21] (probably first publication of the model) and derives it from UML (Unified Modeling Language [10]) sequence diagrams; this model can be used as a basis for architectural level security metrics and as an example, confidentiality metrics are defined based on the model.
24	E. Chew, M. Swanson, K. Stine, N. Bartol, A. Brown, W. Robinson, "Performance Measurement Guide for Information Security", NIST SP	Provides guidelines for developing, selecting, and implementing information system level and security program level measures for assessing the implementation, performance, and impact

	800-55, Revision 1, 2008.	of security controls and other security related activities.
25	"Recommended Security Controls for Federal Information Systems and Organizations", NIST SP 800-53, 2009.	Describes recommended security controls; includes risk assessment as a control; this publication is used by the "Performance Measurement Guide for Information Security" (Table 2 [24]) as a basis for developing security measures.
26	T.E. Hart, M. Chechik, D. Lie, "Security Benchmarking Using Partial Verification", Proceedings of HotSec 08, 2008.	Proposes quantifying insecurity using the partial results of verification attempts - instrumented code (assertions) is property checked until a failure is found. The aggregate of such failures determine the level of insecurity of the software.
27	T. Maibaum, "Challenges in Software Certification", SQRL Report 59, McMaster University, May 2010.	Considers the requirements of software certification, proposing that certification should be product based, not development process based; considers the Common Criteria (CC) [11] as a possible product based model for certification; although this paper is on software certification, it is relevant to security metrics in that it describes the elements of the CC that are pertinent to evaluating the security of a software product.

6.3 Managing IT Security Risks

Table 3 lists the publications in this category, which treats the management of risks for IT vulnerabilities taking into account the probability and impact of occurrence. Managing risks is a process, made up of a) identifying risks, b) assessing risks, and c) reducing the risks to acceptable levels using established procedures. In addition, some of the papers (e.g. Table 3 [4]) provide guidance on selecting security controls for mitigating the identified risks.

Security risk management metrics serve to:

- quantify the risks,
- calculate the risks using formulas, and
- quantify the effectiveness of the risk management process.

Here, the difference between the first and second points is that "quantify the risks" is not limited to producing numbers, e.g. stating that "plan B" is riskier than "plan A".

Note that currently, the quantifications of risks and the risk management process are applicable to all of IT, including, for example, operations and software development. They do not focus on evaluating the security of a computer system with sufficient detail. Therefore, the papers in this category, in a broad sense, extend beyond the evaluation of the security of a computer system. Papers that use risks in conjunction with system components and metrics to evaluate security (not risk management) have been placed in Table 2.

Table 3 – Managing IT security risks

No.	Publication	Summary
1	Andrew Jaquith, Security Metrics: Replacing Fear, Uncertainty, and Doubt, Addison-Wesley, 2007.	Discusses security metrics for enterprise application; security metrics applied broadly, not only to computing systems but also to all sorts of enterprise processes.
2	Lance Hayden, IT Security Metrics: A Practical Framework for Measuring Security & Protecting Data, McGraw-Hill Osborne Media, June 2010.	Similar to the Jaquith book in its focus on the enterprise; covers security metrics in terms of effectiveness, implementation, operations, compliance, costs, people, organizations; includes 4 case studies.
3	J. Talbot, M. Jakeman, Security Risk Management Body of Knowledge, book, Wiley, 2009.	Describes the security risk management process; discusses the pros and cons of various risk measures, including risks of threats and attacks.
4	G. Stoneburner, A. Goguen, A. Feringa, “Risk Management Guide for Information Technology	Provides a foundation for developing a risk management program; contains definitions and guidelines for assessing

	Systems", NIST SP 800-30, 2002.	and mitigating risks within IT systems.

6.4 Measuring the Effectiveness of a Security Process

Table 4 lists the publications in this category, which cover metrics that evaluate the effectiveness of security processes or show where an organization is at in terms of a security maturity model. Note that security processes usually describe security within an enterprise. On the other hand, a security maturity model can apply to an enterprise, a geographical region, and even a country.

Table 4 – Measuring the effectiveness of a security process

No.	Publication	Summary
1	Andrew Jaquith, Security Metrics: Replacing Fear, Uncertainty, and Doubt, Addison-Wesley, 2007.	Discusses security metrics for enterprise application; security metrics applied broadly, not only to computing systems but also to all sorts of enterprise processes.
2	Lance Hayden, IT Security Metrics: A Practical Framework for Measuring Security & Protecting Data, McGraw-Hill Osborne Media, June 2010.	Similar to the Jaquith book in its focus on the enterprise; covers security metrics in terms of effectiveness, implementation, operations, compliance, costs, people, organizations; includes 4 case studies.
3	D. Chapin, S. Akridge, "How Can Security Be Measured?", Information Systems Control Journal, Vol. 2, 2005.	Discusses what is wrong with traditional security metrics, giving characteristics of good metrics; discusses security maturity models with examples.
4	S.S. Alaboodi, "Towards Evaluating Security implementations Using the	Extensions and abstractions of the ISMM security maturity model are proposed with

	Information Security Maturity Model (ISMM)", MASc thesis, University of Waterloo, 2007.	the goals of using the extended model to identify the security level of implementations as well as promote the optimization of IT and security expenditures.
5	Carnegie-Mellon University, "The Systems Security Engineering Capability Maturity Model (SSE-CMM) – Model Description Document", Version 3, June 15, 2003. Accessed Mar. 16, 2012, at: http://www.sse-cmm.org/model/model.asp	Describes essential characteristics of a sound security engineering process; addresses security engineering activities that span the entire security engineering lifecycle, including process metrics; applies to all types and sizes of security engineering organizations, including commercial, government, and academic organizations.
6	R.F. Lentz, "Advanced Persistent Threats & Zero Day Attacks", slide presentation, 2010.	Describes the stages of the Cyber Security Maturity Model, which can be measures of where an organization stands in terms of its security posture.
7	R.F. Lentz, "Cyber Security Maturity Model", slide presentation, 2011.	Describes advanced persistent threats and the stages of the Cyber Security Maturity Model; appears to be an updated version of Table 4 [6].

7. Conclusions

Security metrics provide a quantitative basis for security operations and security-related decision making. They can be used to measure security improvements over time and can therefore show if a series of new investments in security controls is giving better security. Traditional security metrics have been selected haphazardly and have been problematic in that they often targeted aspects of a computer system that were irrelevant to the question at hand. Security metrics should be based on the ideal characteristics given in Section 3.1.2. Security metrics for application in critical areas such as public safety and healthcare should be scientifically based once scientifically based security metrics are available and mature. Key aspects of starting a security metrics program for an organization include designing the security metrics based on the ideal characteristics, and forming a security metrics cross-functional team. An intelligent security dashboard that not only displays alerts but also automatically handles them is an exciting application of security metrics. It appears that current SIEM technology can be a basis for such a dashboard but more research work is needed. Security metrics publications cover the nature of security metrics, measuring the security of a computer system, managing risk, and measuring the effectiveness of a security process.

References

Other references are given in Section 6. Some of the following references also appear in Section 6 and are listed here to allow for ease of citation from the text.

[1] D. Chapin, S. Akridge, “How Can Security Be Measured?”, Information Systems Control Journal, Vol. 2, 2005.

[2] W. Jansen, “Directions in Security Metrics Research”, NIST NISTIR 7564, April 2009.

[3] S. Stolfo, S. Bellovin, D. Evans, “Measuring Security”, IEEE Security & Privacy, May/June 2011.

[4] J. Gottlieb, "10 Tips for Getting Started with Security Metrics", available as of Sept. 5, 2012 from: http://threatpost.com/en_us/blogs/10-tips-getting-started-security-metrics-081712

[5] A. Jaquith, Security Metrics: Replacing Fear, Uncertainty, and Doubt, Addison-Wesley, 2007.

[6] Lance Hayden, IT Security Metrics: A Practical Framework for Measuring Security & Protecting Data, McGraw-Hill Osborne Media, June 2010.

[7] E. Chew, M. Swanson, K. Stine, N. Bartol, A. Brown, W. Robinson, “Performance Measurement Guide for Information Security”, NIST SP 800-55, Revision 1, 2008.

[8] Wikipedia, "CVSS", available as of Dec. 23, 2012 from: http://en.wikipedia.org/wiki/CVSS

[9] Wikipedia, "Common Vulnerabilities and Exposures", available as of Dec. 23, 2012 from: http://en.wikipedia.org/wiki/Common_Vulnerabilities_and_Exposures

[10] Wikipedia, "Unified Modeling Language", available as of Dec. 23, 2012 from: http://en.wikipedia.org/wiki/Unified_Modeling_Language

[11] Wikipedia, "Common Criteria", available as of Dec. 23, 2012 from: http://en.wikipedia.org/wiki/Common_Criteria