DEPARTMENT: CYBERSECURITY

Bad Security Metrics

Part 1: Problems

David Flater
NIST

Editors:
Rick Kuhn, NIST;
d.kuhn@nist.gov

Tim Weil, Alcohol
Monitoring Systems;
tweil.ieee@gmail.com

Security metrics are numerous and in high demand. Unfortunately, measuring security accurately is difficult and many security metrics are problematic.[1]

The problems with security metrics can be complicated and subtle. However, using measurement theory, it's possible to determine quickly that many metrics are unfit for the purposes for which they are used without venturing into subtle or subjective analysis.

This two-part series doesn't call out questionable metrics that are in use. Instead, it focuses on defining the problem conceptually and revealing a path forward for improving both security metrics and how people use them.

SCALE THEORY

Scale theory is a small part of the broad discipline of measurement theory. It was popularized in 1946 by Stanley S. Stevens.[2] Scale theory isn't without problems, but is so well known that it's often the most practical way to frame a discussion. The scales needed for this discussion are introduced briefly in Table 1.

Table 1. Description of scale types.

Name	Description
Ratio (and absolute)	The scale of most physical measurements of length, mass, time, and so on, where units can be converted through simple multiplication. A ratio scale with a supposedly unique, "natural" unit is sometimes called the absolute scale.
Interval	The scale of Celsius and Fahrenheit temperatures, which have meaningful units but arbitrary zero points.
Ordinal	The scale of relative ranks and grades, and generally of ordered values with no meaningful units, where the subtraction of one value from another does not yield a meaningful quantity.
Nominal (and dichotomic)	The scale of naming or classifying things. Nominal values have neither magnitude nor order. The two-valued nominal scale of a generic two-

Published by the IEEE Computer Society
1520-9202/18/$33.00 ©2018 IEEE

	way partitioning (usually yes/no or true/false) has been called the dichotomic or dichotomous scale.

The following sections identify six characteristics of bad security metrics and explain how they are problematic.

MAKE-BELIEVE MEASURAND

The *International Vocabulary of Metrology*[3] defines "measurand" as the "quantity intended to be measured," and it defines "quantity" as a "property of a phenomenon, body, or substance where the property has a magnitude that can be expressed as a number and a reference." For example, "5 m" means 5 times the standard unit of measure, which is the meter.

The first problem we encounter with security metrics is simply that they are often put forth as measurements of security as if security were a quantity to be measured. A statement such as "X is 5 times as secure as Y" is meaningless on its face unless the general term "security" is given a specific interpretation. For example, if X is claimed to be 5 times as secure as Y because an attack against it is expected to take 5 times as long, then security has been interpreted to mean a specific quantity of time for which a standard unit of measurement already exists.[4] Without such an interpretation, there can be no unit of security, and security can't be a quantity in the metrological sense.

Furthermore, security isn't a property of an IT artifact in anything remotely like the way that mass is a property of a physical artifact. It isn't inherent or intrinsic to the artifact. As described in *A Rational Foundation for Software Metrology*:

> *To be meaningful, security for any system, software or physical, must be defined according to some specification of the protection that it will provide in a particular threat environment. Assumptions must be made regarding the capabilities and ingenuity of the adversary and the pace of future technological progress, all of which are unknown.*[5]

Unlike a normal physical measurement that captures information about the present state of a system, an assessment of security is primarily a forecast about what can or will happen in the future.[6] It can be an observation of present facts only in the case that security is absent.

MISUSE OF ORDINAL SCALE

The non-comparability of differences on an ordinal scale makes them unfit for the purpose of making security-relevant decisions. Consider the example of choosing an email filter to keep out phishing, and consider the two different metrics shown in Table 2 that could be used to evaluate competing products. Assume we have narrowed the choices to filter A and filter B, with the attributes shown in Table 3.

If we only had metric 2, we would have an indication that filter A performed better than filter B, but we would have no way to evaluate the magnitude of the performance advantage and no rational way to decide whether it was worth the money. Only a wealthy fool would buy the top-performing tool without looking at the price–performance tradeoff.

This example was simplified by the assumption that a better metric existed, but the argument holds generally: ordinal metrics give no indication of the magnitudes of differences, and therefore can't be used to evaluate the tradeoffs of security against cost, usability, and so on that are critically important to real-world security planning.

Table 2. Example metrics for email filters.

Metric	Description	Scale
1	Proportion of phishing attempts blocked, for a representative sample	Ratio/absolute
2	Relative ranking of the products' performance (1 means best, 2 means second-best, and so on)	Ordinal

Table 3. Product choice scenario.

Filter	Metric 1	Metric 2	Price
A	83%	1	$10,000
B	82%	2	Free

FALSE PRECISION

In Table 3, results were stated without any indication of uncertainty. One might assume that this means they are sufficiently accurate for any reasonable use to which one would put them. This often isn't the case.

Without an indication of the uncertainty of results, we have no reason to believe that the difference in performance was actual. Generally accepted measurement practice calls us to test the products not with one sample but with many different ones, and then report the mean performance with a confidence interval. The results would then look something like Table 4. This rendering is consistent with the first, but the confidence intervals reveal that the difference between products isn't statistically significant. The results without uncertainty, therefore, could be worse than useless, potentially misleading someone into paying $10,000 extra for performance that's actually worse.

Table 4. Product choice scenario with confidence intervals.

Filter	Metric 1	Metric 2	Price
A	(83 ± 3)%	1 to 2	$10,000
B	(82 ± 5)%	1 to 2	Free

MISLEADING SCALE

A misleading scale is a way of expressing a result that leads the reader to make inferences that are unsound or untrue. For example, the numerical range of an ordinal measure is completely arbitrary and meaningless, but often they're scaled to produce a maximum value of 10 or 100. This leads the audience to confuse them with probabilities and proportions and then use them as if they were on a ratio scale.

Another common misleading practice is using a count of rule violations or nonconformities as a "badness" metric. The problem here is the induced assumption that having n rule violations is n times as bad as having a single rule violation, or alternately, that violating n different rules is n times as bad as violating a single rule. The count is legitimately on a ratio or absolute scale, but only while it's interpreted literally as a count of rule violations and nothing more. As soon as it's interpreted as a metric of badness, it becomes misleading. For example, if two different rules address two different attack vectors, either of which would allow for complete system compromise with high likelihood and at negligible cost to the attacker, it's hard to argue that violating both rules simultaneously is even approximately twice as bad as violating only one. It's somewhat analogous to multiplying an infinite value (infinite badness) by a constant.

COMBINING DISPARATE MEASURES

Attempts to synthesize a metric for nontrivial qualities often resort to weighted sums or more complex functions of disparate measures that in principle aren't even comparable with one another as quantities. The combination functions have no theoretical basis or measurement principle to justify their derivation. The mathematical notation might lead a reader to assume that the functions are derived using sound principles, but it might instead be a veneer that hides flaws. The functions might merely have been designed to produce the desired results for the cases tested.

Combining disparate measures can make it difficult or impossible to produce a valid metric. Properties that might be lost include:

- Correlation: in repeated use on different objects, the measured quantity value should exhibit a consistent correlational relationship to the measurand.
- Tracking: a change in the measurand should always result in the measured quantity value moving in the same direction.
- Consistency: measured quantity values should always preserve a relative ordering by the measurand.
- Predictability: for metrics that are used in forecasting, prediction error should always be within the specified range or tolerance limit.
- Discriminative: high and low values of the true quantity should be distinguishable from the measured quantity values.

The above list, reproduced from *A Rational Foundation for Software Metrology,*[5] is a reinterpretation of criteria that appeared in ISO/IEC TR 9126-2:2003, 3:2003, and 4:2004, Software engineering—Product quality, §A.2.2.

NAIVE USE OF HUMAN INPUT

When humans are the measuring instruments, subjectivity and human factors introduce distortion. For example, a respondent who is asked to evaluate something on a five-point scale might perceive the five levels as covering different-sized intervals or might "grade on a curve" based on a belief about what the distribution of results is supposed to be. To mitigate such distortion, social scientists have been migrating to more robust models, such as the graded response model within item response theory,[7] whose validity depends on less onerous assumptions. Unfortunately, those advances have not transferred to analogous activities in IT risk assessment, such as estimation of likelihoods and impacts. Unless human factors are taken into consideration, the uncertainty associated with various human-produced scores will be large.

CONCLUSION

This article, the first of a two-part series, provided background on scale theory and explored different types of problems that afflict security metrics. Part two will continue with an explanation of how to avoid these issues, and finish with conclusions.

ACKNOWLEDGMENTS

Thanks to Paul Black and Barbara Guttman of NIST for their helpful review comments and suggestions.

REFERENCES

1. S.L. Pfleeger and R.K. Cunningham, "Why Measuring Security Is Hard," *IEEE Security & Privacy*, vol. 8, no. 4, 2010, pp. 46–54.
2. S.S. Stevens, "On the Theory of Scales of Measurement," *Science*, vol. 103, no. 2684, 1946, pp. 677–680.
3. *International Vocabulary of Metrology—Basic and General Concepts and Associated Terms (VIM)*, Joint Committee for Guides in Metrology, 2012; www.bipm.org/en/publications/guides/vim.html.
4. *The International System of Units (SI)*, BIPM, 2006; www.bipm.org/en/publications/si-brochure.
5. D. Flater et al., *A Rational Foundation for Software Metrology*, report NIST IR 8101, NIST, 2016; doi.org/10.6028/NIST.IR.8101.
6. D. Flater, "Mostly Sunny with a Chance of Cyber," *Proc. NIST Workshop Software Measures and Metrics to Reduce Security Vulnerabilities*, 2016; www.nist.gov/node/1114701.
7. *Item Response Theory: Parameter Estimation Techniques*, CRC Press, 2004.

ABOUT THE AUTHOR

David Flater is a computer scientist in the Software and Systems Division of NIST's Information Technology Laboratory. Contact him at david.flater@nist.gov.

Disclaimer: Products may be identified in this document, but such identification does not imply recommendation by the US National Institute of Standards and Technology or the US Government, nor that the products identified are necessarily the best available for the purpose.

DEPARTMENT: CYBERSECURITY

Bad Security Metrics

Part 2: Solutions

David Flater
NIST

Editors:
Rick Kuhn, NIST;
d.kuhn@nist.gov

Tim Weil, Alcohol
Monitoring Systems;
tweil.ieee@gmail.com

In last issue’s Cybersecurity department (“Bad Security Metrics Part 1: Problems”[1]), I introduced scale theory and identified the following characteristics of bad security metrics, explaining how they are problematic.

Make-believe measurand. Security is treated as if it were an intrinsic property of something, like the mass of a physical object.

Misuse of ordinal scale. A number with no measurement unit is used as the basis for security-relevant decisions.

False precision. The uncertainty of a measurement is ignored.

Misleading scale. Results are presented in ways that lead readers to make inferences that are unsound or untrue.

Combining disparate measures. Quantities that in principle aren’t comparable with one another are mathematically combined without a justification of how this yields a valid measurement.

Naive use of human input. Distortions introduced by subjectivity and human factors are not addressed.

In part 2, I present six principles to help avoid the problems discussed in part 1 and move forward using sound measurement.

DESCRIBE THE MEASURAND ACCURATELY

The key is to represent and communicate the observable facts—not add to them, take away from them, or mutate them in the process. This is a semantic condition (not a syntactic one), and it places a burden on the listener as well as the speaker. In the absence of standard units of measure, the description of the quantity is extremely important for the correct interpretation of the result. The speaker must take great pains to state exactly what was measured (for example, “proportion of tests passed”) without suggesting further implications or interpretations (for example, “percent compliant”).

SEPARATE METRICS FOR DISTINCT DIMENSIONS

Combining disparate measures leads to problems with the validity of the result. Avoiding this requires pushing back against constant societal demands for single numbers and simple answers

Published by the IEEE Computer Society
1520-9202/18/$33.00 ©2018 IEEE

to complex questions. When there are multiple kinds of security involved, don't let them be reduced to a single "badness" number; use separate metrics for distinct dimensions.

A Rational Foundation for Software Metrology states: "Security is conventionally considered to have at least three primary aspects—confidentiality, integrity, and availability—that have tradeoffs among them." (For example, controls that intentionally lock out attackers tend to unintentionally lock out authentic users as well, harming availability.) "Reducing security to a single score may provide desired simplicity for a go/no-go decision, but a truly informative description of security requires multiple quantities—the inputs to the decision function, not its output."[2]

Shari Pfleeger and Robert Cunningham made a similar statement: "'Security' is shorthand for describing a collection of attributes that capture security's many dimensions. Rather than agonize over finding a security metric, we can use different metrics for different attributes."[3]

USE DICHOTOMIC METRICS APPROPRIATELY

There are many situations in which a test with a yes or no result is extremely valuable. For example, "Does this system have known vulnerabilities that could be exploited to foil our plan?"

A "no" answer is basically a null result (it neither confirms nor refutes the hypothesis that the system is "adequately secure"). However, if the answer is "yes," then deploying the system would presumably be postponed until the vulnerability was fixed. The worth of the metric is in the avoidance of fielding a system known to be insecure.

Stating the result in yes or no terms instead of as a count of detected vulnerabilities makes the result less open to abuse, such as assuming that two vulnerabilities are twice as bad as one. To revisit an example given in part 1 of this article, the correct scale for the metric addressing two different attack vectors (either of which would allow for complete system compromise with high likelihood and at negligible cost to the attacker) is dichotomic: either you guard against known dangerous attacks or you fail.

USE NOMINAL METRICS APPROPRIATELY

There are also many opportunities to make productive, valid use of nominal metrics. For example, the set of roles or privileges required by an app is a nominal indication of your security exposure to the app. Most roles and privileges are not comparable to one another, yet each resource to which the app is granted access extends your exposure in some way.

Strict subsumption is an exceptional case in which you can objectively state that one app requires more exposure than another. This happens when the set of roles or privileges required by one app is a proper superset of those required by another app, or when a given role or privilege obviates the need for others (as *root* often does). In most cases, however, you end up trying to compare things like "allow app to determine device's location" with "allow app to obtain audio from device's microphone." You cannot objectively quantify the exposures that these two different privileges add; their impacts on different users, including the one who never travels and the one who never says anything, are different. But as long as each required role or privilege is interpreted appropriately, the set provides meaningful information.

The most pragmatically useful security metrics are likely to be simple ones that communicate easily observed but inconvenient truths.

USE RATIO AND ABSOLUTE METRICS WHEN FEASIBLE

There are security-relevant metrics that genuinely satisfy the conditions of the ratio scale and whose only problem is being taken out of context. *Performance Measurement Guide for Information Security* defines 19 metrics, 18 of which are proportions ("percentages").[4] For example, Security Budget Measure 1 is the proportion of the agency's information system budget devoted to information security, a ratio of two monetary amounts. This is a well-defined quantity. Problems arise only when it is used as a surrogate for information security. The non-proportion metric is Audit Record Review Measure 1, which is the average frequency at which audit records are reviewed and analyzed for inappropriate activity.

An estimate of the number of "guesses" required for an attack on a secure hash function to succeed is another example of a well-defined metric on a ratio scale. However, it is vitally important that the listener understand the context: an adversary is not prevented from accomplishing the objective much faster through a lucky guess or by finding a better search algorithm.

ADDRESS UNCERTAINTY AND HUMAN FACTORS

When a single number is being calculated with no apparent uncertainty, pause and consider whether this number is truly the desired measurement result or whether it is merely an indicator or estimator of something of broader scope. In the false precision example in part 1, the performance of a particular version of a product with respect to a particular sample of input was being used as an estimate of how future updated versions would perform with respect to a larger population of inputs. Even if we assume (unwisely) that past performance is a reliable predictor of future performance, it is still necessary to characterize the variability of the input quantities and the consequential uncertainty of the calculated result. Software security specialists might not have been trained in the estimation of measurement uncertainty. Fortunately, canonical references are free to download,[5,6] and NIST routinely offers a three-day training course in the fundamentals of uncertainty analysis (www.nist.gov/programs-projects/statistical-metrology-short-courses).

When the source of a quantity is not an objective measurement but a human who was asked to assign numbers, be aware that reliably collecting data from humans is a challenge on par with making software secure. The difficulty and cost of doing either of these well fosters a pattern of neglect and unpleasant surprise among the unwary. Just like questions about security and uncertainty, questions about human factors need more than a shrug for an answer. For all of these questions, there are resources and specialists who can help, and the first milestone on the path forward is simply ensuring that available knowledge gets used where applicable.

CONCLUSION

Many IT security metrics are ordinal and therefore cannot rationally be used for evaluating real-world tradeoffs of security against cost, usability, and so on. Those that are not ordinal still might not be indicative of real-world security. Real-world studies and well-designed experiments are required to determine the extent to which they correlate with desired outcomes. Such studies and experiments are fraught with complications, but they can still provide useful information.

It is recognized in IT security practice that the bulk of the problem results not from sophisticated attacks that are academically interesting, but from organizations and individuals failing to take simple steps to improve security. The most pragmatically useful security metrics, therefore, are likely to be simple ones that communicate easily observed but inconvenient truths.

ACKNOWLEDGMENTS

Thanks to Paul Black and Barbara Guttman of NIST for their helpful review comments and suggestions.

REFERENCES

1. D. Flater, “Bad Security Metrics Part 1: Problems,” *IT Professional*, vol. 20, no. 1, IEEE, 2018, pp. 64–68.
2. D Flater et al., *A Rational Foundation for Software Metrology*, government report NIST IR 8101, NIST, 2016; doi.org/10.6028/NIST.IR.8101.
3. S.L. Pfleeger and R.K. Cunningham, “Why Measuring Security Is Hard,” *IEEE Security & Privacy*, vol. 8, no. 4, 2010, pp. 46–54.
4. E. Chew et al., *Performance Measurement Guide for Information Security*, government report NIST SP 800-55 Rev. 1, NIST, 2008; doi.org/10.6028/NIST.SP.800-55r1.
5. *Evaluation of Measurement Data—Guide to the Expression of Uncertainty in Measurement*, standard JCGM 100:2008, JCGM, 2008; www.bipm.org/utils/common/documents/jcgm/JCGM_100_2008_E.pdf.
6. *Evaluation of Measurement Data—Supplement 1 to the “Guide to the Expression of Uncertainty in Measurement”—Propagation of Distributions using a Monte Carlo Method*, standard JCGM 101:2008, JCGM, 2008; www.bipm.org/utils/common/documents/jcgm/JCGM_101_2008_E.pdf.

ABOUT THE AUTHOR

David Flater is a computer scientist in the Software and Systems Division of NIST’s Information Technology Laboratory. Contact him at david.flater@nist.gov.

Disclaimer: Products may be identified in this document, but such identification does not imply recommendation by the US National Institute of Standards and Technology or the US Government, nor that the products identified are necessarily the best available for the purpose.