Sound Security Metrics

George O. M. Yee, Ph.D., P.Eng., CISSP, SMIEEE

Computer Research Lab, Aptusinnova Inc.

Dept. of Systems and Computer Engineering, Carleton University

Ottawa, Canada

gmyee@sce.carleton.ca | george@aptusinnova.com

Presenter's Biography

George Yee is a research scientist with Aptusinnova Inc., and an Adjunct Research Professor at Carleton University, Ottawa, Canada. He earned his Ph.D. (Electrical Engineering) in 1991 at Carleton University and is a registered professional engineer with Professional Engineers Ontario in Canada. Dr. Yee has over 100 refereed research publications in the form of conference papers, journal articles, and books. He volunteers as an organizer and technical reviewer for scientific conferences. His research interests include the application of mathematics to security and privacy, especially security and privacy for software engineering.

Content

- Introduction to the Problem and to Security Metrics
- Sound Security Metrics – Method for Design and Test
- Applications of the Method
- Conclusions and Future Research

This presentation is based on:

G. O. M. Yee, "Designing sound security metrics," *International Journal of Systems and Software Security and Protection*, vol. 10, no. 1, pp. 1–21, 2019.

Introduction to the Problem

- Today's world is precarious.
- Organizations have spent large sums on security. Returns?
  - Spent enough?
  - Software changes to improve security effective?
  - Work flows and processes secure?
  - Security impact of adding third party SW component?
- Need to assess security level – need properly defined, effective security metrics

Introduction to the Problem

- Many existing security metrics are poorly defined and ineffective.
  - Example: N = number of viruses detected and eliminated at a firewall

    Problem: What about the viruses that were NOT detected and got through?

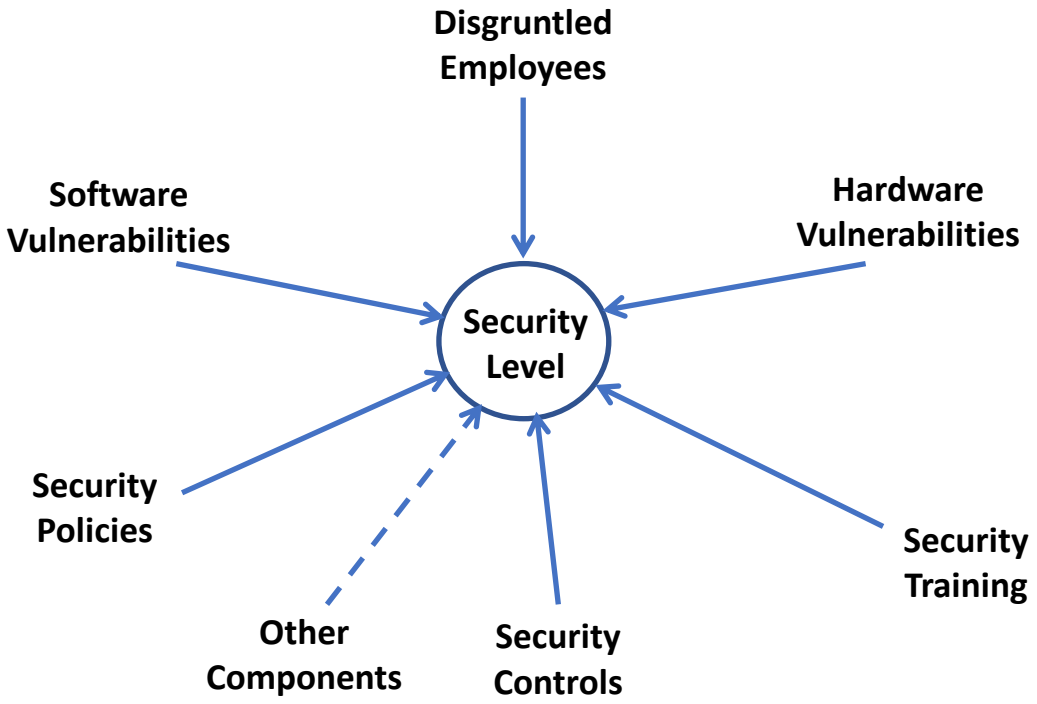
    Suppose 50 viruses detected and eliminated but 100 got through.
- Can we define security metrics that are meaningful and effective?
- What conditions must security metrics satisfy to be considered sound?

Introduction to Security Metrics

- Computer systems must be secure and respect privacy
- Researchers and practitioners have made every effort to achieve this
- But the development of effective security metrics to help them achieve this has been a difficult challenge
- Security Level: the degree to which “something” is considered secure

Introduction to Security Metrics

- Component of the security level: anything that plays a part in determining the security level



- May be very difficult, if not impossible, to identify all the components

Introduction to Security Metrics

- Definition of a security metric
  - A numerical value or set of numerical values or a formula that evaluates to the numerical value or values
  - Measures some component or components of the security level of "something" at a particular point in time
  - "Something" could be a computer system, an organization, a software product, an access control system, and so on
- A security metric may only partially measure the security level since it may be impossible to identify all components of the security level, e.g., SW system vulnerabilities

Introduction to Security Metrics

- A security metric measures the security level at a particular point in time, since the security level can change over time, e.g., new security controls added
- Example security metrics
  - Frequency of changing a banking password
  - Number of times a computer OS is patched for security vulnerabilities in a year
  - Amount spent in an year to deploy security controls

Introduction to Security Metrics

- Objectives of security metrics
  - Provide quantitative and objective basis for security operations, e.g., how often do users need to change their passwords?
  - Strategic support – aid decision making for program planning, resource allocation, selection of products and services, e.g., more security controls needed?
  - Quality assurance for software development – e.g., measuring adherence to secure coding standards, tracking and analyzing security flaws
  - Tactical oversight – e.g., determine compliance with security requirements, gauge the effectiveness of security controls, basis for trend analysis

Introduction to Security Metrics

- Challenges with Definition (measure which components)
  - Example 1: **number of computer viruses or malware detected**
    - Intended to measure the effectiveness of anti-malware controls
    - Leaves out the malware that was undetected and got through
  - Example 2: **number of security incidents reported**
    - Intended as an indicator of the security level of the organization
    - Fails to incorporate incident thresholds (needed to understand the severity of an incident) and causes (incident may be triggered by flaws in work processes)

Introduction to Security Metrics

- Challenges with Application
  - Example 1: **time spent on a security related task** (e.g. SW patching)
    - Intended as an indicator of security – the more time the better the security
    - Spending more time does not necessarily result in better security
  - Example 2: **business cost of a security incident**
    - Look at financial losses to gauge the quality of the organization's security practices
    - A high loss may have been due to something other than poor security practice, e.g., accidental loss

Sound Security Metrics

- Characteristics of Sound Security Metrics Proposed by Researchers
  - Measure meaningful things
  - Reproducible (results can be reproduced by a third party)
  - Objective and unbiased
  - Measure over time some type of progression toward a goal
  - Accurate, precise, valid, correct
  - Consistently measured
  - Cheap to gather
  - Contextually specific

Sound Security Metrics

- Some traditional security metrics fail to have one or more of these characteristics
  - Selected haphazardly or opportunistically
- Some of the traditional metrics discussed above are misleading because they fail to incorporate logically needed elements, e.g., firewall example
- A security metric is “progressive” if over a sufficiently long period, the metric progresses to a value corresponding to an acceptable or maximal security level.
- A security metric is “sufficient” if it behaves appropriately and is not missing any aspect required for it to be effective.

Sound Security Metrics

Definition of a sound security metric

- A security metric is sound if and only if the metric is meaningful, objective, unbiased, not costly or difficult to obtain, sufficient, progressive, and reproducible.

Sound Security Metrics

Method for Designing Sound Security Metrics (MDSSM):

1. **Definition:** Define the quantity (security level component) to be measured.
   - Meaningful, objective, unbiased, can be obtained without undue hardship or costs – if YES, go to step 2. Otherwise repeat.
     - Example quantity: *number of SW security patches issued in a month*

Sound Security Metrics

MDSSM:

2. **Divisibility**: Check if the quantity can be expressed in terms of other constituent components
   - If YES, formulate a mathematical expression for the quantity in terms of the constituent components and go to step 3; if NO, go to step 3.
     - Example: the quantity *number of SW security patches issued in a month* is not further divisible
     - Example: the quantity *outstanding vulnerabilities after threat analysis each month* may be divided and equated to the number of non-secured vulnerabilities from last month plus the number of new vulnerabilities found during threat analysis

Sound Security Metrics

MDSSM:

3. **Sufficiency**: Check that the quantity is a sufficient measure by asking:
   1. If the quantity goes up, do you believe that the security level consistently goes up (or down)? **Yes for sufficiency**
   2. Does the quantity have a direct impact on the security level? **Yes for sufficiency**
   3. Are there any aspects missing from the definition of the quantity that are needed for it be effective as a measure of the component or components of the security level? **No for sufficiency**

   If sufficient (YES, YES, NO), go to step 4. Otherwise, repeat from step 1.

Sound Security Metrics

MDSSM:

- Example for Sufficiency: the quantity *time spent on a security-related task* is not sufficient – spending more time does not mean that the security level will be consistently higher (or lower); thus the answer to the first question is “no”, and the quantity is therefore not sufficient.

Sound Security Metrics

MDSSM:

4. **Progression:** Check that when evaluated over a sufficiently large time period, from past to future, the quantity progresses to a value that corresponds to an acceptable or maximal security level.
   - If YES, go to step 5. If NO, repeat from step 1.
     - Example: Suppose the *cumulative number of SW security patches applied* is updated at the first of a month by adding the patches applied in the last month; over time, this quantity will never decrease, and will reach some maximal value with corresponding maximal security level of the software, where the security level increases with each additional patch applied.

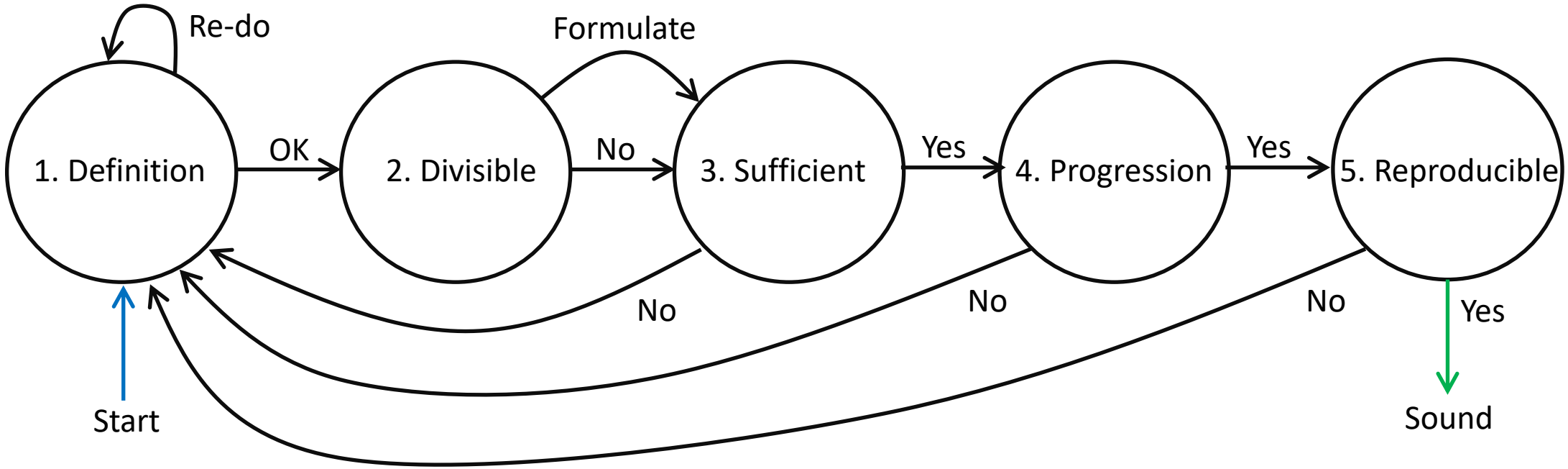
Sound Security Metrics

MDSSM:

5. **Reproducibility:** Check that the quantity is reproducible or verifiable by third-party verifiers, i.e., using identical input and procedures, the verifiers obtain the same value(s) (quantity deterministic) or expected value(s) (quantity non-deterministic)
   - If YES, STOP. Quantity is sound. If NO, repeat from step 1.
   - Example: for the quantity *cumulative number of SW security patches applied,* a third party verifier would perform the same calculation using the same input as the organization using this metric and obtain the same values (deterministic)

Sound Security Metrics

MDSSM Flow Diagram



Sound Security Metrics

MDSSM Application Notes:

- In Step 1, an example of a biased quantity is the *number of viruses detected and eliminated at a firewall* – biases up the security level
- In Step 2, breaking up the quantity into its constituents allows ease of evaluation. Example: cost of a security incident = investigation cost + remediation cost
- In Step 3, answering question 3 may not be obvious. May help to have the "big picture" in mind or ask the opinion of others

Sound Security Metrics

MDSSM Application Notes:

- In Step 4, progression is needed to answer questions such as when do we know that it is “safe” and how can the cost of security controls be justified.
- In Step 5, reproducibility is necessary for verification by others; otherwise, how do we know that a mistake didn’t happen? It is a fundamental requirement of any “fact” or “evidence” of significant importance, that it be verifiable by others.

Sound Security Metrics

MDSSM Strengths:

- Provides rigorous step-by-step checklist for designing a new security metric that is sound
- Applicable for testing existing security metrics to see if they are sound
- Provides metrics that can answer what management needs to know, e.g., an assessment of the organization's security level, knowing when the organization is "safe", how to justify the cost of security controls

Sound Security Metrics

MDSSM Weaknesses:

- Steps 1 and 3 may be challenging to carry out – requires some security and computer systems expertise
  - May be difficult to come up with a security metric that has the desirable characteristics listed in Step 1.
  - Question 3 in Step 3 requires one to think through whether or not the metric will be effective in how it will be applied
- May help to apply MDSSM using a team approach.

Applications of MDSSM

Designing a new security metric – an illustration

- Bob, who is knowledgeable about security and computer systems, is hired as a security consultant for Company A's computer system
- Company A has spent tens of thousands of dollars on securing computer system vulnerabilities
- Company A's Management wants to know if even more vulnerabilities need to be secured in order to be "safe"

Applications of MDSSM

Designing a new security metric – an illustration

- Bob will be working as part of a team that includes the computer system's operations manager.
- The team's objective is to design a new security metric that is sound and can answer Management's question.
- The team apples MDSSM as follows.

Applications of MDSSM

Designing a new security metric – an illustration

STEP 1: Definition

- System security is directly related to the number of secured vulnerabilities: the higher this number, the higher the security; the lower this number, the lower the security
- Therefore, the quantity chosen to measure system security is *the percentage of secured vulnerabilities over all known vulnerabilities (both secured and unsecured)* or PSV for Percentage of Secured Vulnerabilities

Applications of MDSSM

Designing a new security metric – an illustration

STEP 1: Definition

- PSV
  - is meaningful for assessing if more vulnerabilities need to be secured
  - is objective since secured vulnerabilities relate directly to system security
  - is unbiased since its value cannot be understated or overstated
  - can be obtained without undue hardship or cost
- The team considers PSV as having passed all checks and proceeds to Step 2.

Applications of MDSSM

Designing a new security metric – an illustration

STEP 2: Divisibility

The team notices that PSV can be expressed as

$$PSV = (100 \times p) / (p + q) \quad \text{if } p + q > 0$$
$$= 100 \quad \text{if } p + q = 0$$

where p = secured vulnerabilities, q = unsecured vulnerabilities, and (p + q) = all vulnerabilities (both secured and unsecured). This expression behaves as expected when q = 0, p = 0, or p + q = 0. Since PSV is not divisible, the team proceeds to Step 3.

Applications of MDSSM

Designing a new security metric – an illustration

STEP 3: Sufficiency

The team answers the 3 questions given above in order to determine sufficiency. Replacing "quantity" with "PSV":

1. If the PSV goes up, do you believe that the security level consistently goes up (or down)? Yes.
2. Does PSV have a direct impact on the security level? Yes.
3. Are there any aspects missing from the definition of PSV that are needed for it be effective as a measure of the component or components of the security level? As far as the team can tell, No.

The team concludes that PSV is sufficient and proceeds to Step 4.

Applications of MDSSM

Designing a new security metric – an illustration

STEP 4: Progression

- Vulnerabilities are determined and PSV re-calculated at regular intervals, e.g., monthly
- A PSV of at least 95 is considered “safe”
- Management will want to secure vulnerabilities at each opportunity until, eventually, PSV ≥ 95
- The team concludes that PSV satisfies Progression and proceeds to Step 5.

Applications of MDSSM

Designing a new security metric – an illustration

STEP 5: Reproducibility

- The team observes that anyone will calculate the same value for PSV given the same values for p and q.
- The team concludes that PSV is reproducible.

The team has successfully designed a new security metric that is sound using MDSSM. Further, PSV can answer the question of whether or not more vulnerabilities need to be secured in order to be "safe" (Step 4).

Applications of MDSSM

Testing existing security metrics

- MDSSM Step 2 Divisibility is not applicable when testing existing security metrics for soundness
- A security metric *m* is not sound if and only if *m* fails one or more of MDSSM Steps 1, 3, 4, or 5.
- To show that an existing security metric is not sound, it suffices to find one MDSSM step (1, 3, 4, or 5) in which the metric fails.
- To show that an existing security metric is sound, we must show that the metric passes all of MDSSM Steps 1, 3, 4, and 5.

Applications of MDSSM

Testing existing security metrics

Metric: *number of computer viruses or malware detected*

MDSSM Step 1: Definition.

- Does not account for the viruses or malware that were not detected
- Biased toward overstating the effectiveness of the malware detection
- Fails Step 1 and therefore not sound

Applications of MDSSM

Testing existing security metrics

Metric: *number of security incidents reported*

MDSSM Step 3: Sufficiency.

- Answer the 3 questions:
  - Question 1: If this metric goes up, do you believe that the security level will consistently go down? YES.
  - Question 2: Does the metric have a direct impact on the security level? YES.
  - Question 3: Are there aspects missing from this metric that are needed for it to be effective? YES. E.g., a security incident can occur that has nothing to do with security (accident). (need NO here)
- Fails Step 3 and therefore not sound

Applications of MDSSM

Testing existing security metrics

Metric: *time spent on a security related task*

MDSSM Step 3: Sufficiency.

- Answer the 3 questions:
  - Question 1: If this metric goes up, do you believe that the security level consistently goes up (or down)? NO. Spending more time does not mean that the security level will be consistently higher (or lower). The person could just be slow. (Need YES here)
- Fails Step 3 and therefore not sound

Applications of MDSSM

Testing existing security metrics

Metric: *business cost of a security incident*

MDSSM Step 1: Definition.

- Idea: the higher the cost, the lower the quality of security practice
- A high cost does not necessarily mean a low quality of security practice. The high cost may have nothing to do with security practice, e.g. due to the high value of an asset lost.
- Not meaningful and biased
- Fails Step 1 and therefore not sound

Applications of MDSSM

Other researchers have applied MDSSM:

J. Samuel, K. Aalab, and J. Jaskolka, "Evaluating the soundness of security metrics from vulnerability scoring frameworks," in *19th IEEE International Conference on Trust, Security and Privacy in Computing and Communications*, (Guangzhou, China), pp. 442–449, 2020.

- Vulnerability scoring frameworks (e.g., CVSS) aim to estimate the severity of known vulnerabilities in SW dependent systems and provide security metrics for use in security decision-making processes.
- The above work applies MDSSM to determine the soundness of some of these metrics.

Conclusions and Future Research

- Security metrics provide a quantitative basis for security operations, providing actionable information for security decision makers. They can also measure security improvements over time, in order to justify new security controls.
- Some past security metrics have been problematic, missing relevant aspects of security or the system.
- This work has presented MDSSM, a rigorous step-by-step method for designing sound security metrics. MDSSM can also be used to test existing security metrics for soundness to avoid problematic metrics.
- Future research includes improving MDSSM and exploring new types of security metrics (e.g., derived from Big Data).

Thank you for your attention.