Proceedings of
The Eighth Annual Meeting
of
Metricon

Friday, March 1, 2013
San Francisco, CA

Metricon is a forum for lively, practical discussion in the area of security metrics.

DRAFT FOR PARTICIPANT REVIEW

Please use Word Track Changes and Comments features or just sent comments via email to:

jennifer@bayuk.com

Metricon 8

Table of Contents

1. Executive Summary

The goal of Metricon 8 was to bring together practitioners in security metrics, review both the state of the art and the state of the practice in security metrics, and leverage the collective wisdom of participants to take the first steps toward a taxonomy or framework for metrics in areas that are of significant value to enterprise security programs.

The event consisted of direction-setting discourse, a panel consisting of leading metrics data publishers, a panel consisting of enterprise security practitioners, short talks on emerging trends, and facilitated group sessions focused on metrics of common interest. The outcome was a short list of key metrics in these areas:

- Data Breach Costs
- Malware Identification
- Vulnerability Management
- Systems Development Controls
- Information Security Program
- Cyber Security Risk
- Business Impact

This report includes the scope and approach of the Metricon 8 workshop, as well as detailed descriptions of the metrics identified as key indicators of effective information security. An appendix lists workshop participants and roles.

The goal was achieved in that the collective wisdom of participants was leveraged to take the first steps toward a taxonomy or framework for metrics in areas that are of significant value to enterprise security programs. However, these were baby steps that left the security metrics community profoundly aware of how large the gap is between the state of the art in security metrics and the metrics needed by enterprise security practitioners.

2. Scope and Approach

The day began with a discussion of goals and objectives led by the program chair. Participants self-identified areas of interest, loosely based on a list provided in the program agenda. Facilitator-led break-out groups aligned with these areas of interest, and these produced an initial set of metrics. The plan for each facilitator-led group session was threefold:

1. Create a series of scenarios associated with topic areas.
2. Define a set of metrics that will best inform decisions regarding these scenarios.
3. Review published data to see what we can pull from it and conduct a gap analysis.

Elements expected to compose the metrics definitions were listed in the program:[1]

Name:	Descriptive label
Measure:	The unit of quantitative measurement(s).
Scenarios:	Describe the scenarios where the metric would be useful.
Frequency:	Propose time periods for collection of data that is used for measuring changes over time.
Formula:	Describe the calculation to be performed that results in a numeric expression of a metric.
Indicators:	Provide information about the meaning of the metric and its performance trend.

After completing steps 1 and 2, the groups reported their preliminary results to in a general session augmented by an "enterprise" panel with CISO-level enterprise security experience. First the panel commented on the outcome, then the discussion opened to all participants. Groups were expected to use this feedback to refine their metrics lists.

The idea was to incorporate an evaluation of existing industry data sources with an eye toward identifying alignments, gaps, and overlaps as these reports relate to the needs of the enterprise security professional. To this end, this enterprise panel was followed by a series of "lightning talks" on emerging issues, so named because they were limited to 5-10 minutes each. Following the lightning talks, a diverse panel of metrics data publishers were asked to describe what is in their reports and to discuss how they expect enterprise security practitioners to make use of the data in the report to make decisions.

Topics covered in the lightning talks were:

- *Pete: please fill in lightning talk topics*

The members of the data publishing panel represented firms who collect and publish data in three different segments of the security breach lifecycle. Each had a different concern about how their data was being used. The segments and corresponding concerns are briefly summarized as follows:

- Independent publisher of vulnerabilities and threats
    - Budget and clarity constraints prevent us from covering every single vulnerability, so what criteria should we use to determine inclusion?
    - We need to evolve with technology, but how do we know when we make changes that these will not diminish utility to our subscribers?
- Security service provider publisher of incident metrics in progress
    - Do we correctly recognize a compromise?
    - How do we know we have set severity levels appropriately?
- Security forensics firm publisher of post-mortem data on security breaches
    - What data can we collect that will lead us to root cause?
    - How can we use data on compromised customers to help others?

[1] These key features of metrics were adapted from NIST SP 800-55.

The resulting discussion was open to all participants. A major topic of discussion was the burden on practitioners to absorb results from multiple reports that share no common references on methodology or metrics terminology. One insightful comment, that it was admitted has occasionally surfaced on the securitymetrics.org mail list, was on the need for the Center of Disease Control (CDC) of information security breaches. It was observed that breach information has sometimes been aggregated from disparate sources and that doing this the wrong way can be very misleading.

3. Key Metrics

The key metrics produced by each group are below organized by the scenario faced by a security practitioner. The scenario is briefly described in text, and supported by a table that lists several of the elements expected to compose the metrics definitions as defined in the workshop program. Each scenario is followed by a description of what the metrics would *indicate* to a decision-maker. As defined in the program, these indicators provide information about the meaning of the metric and its potential performance trends.

3.1. Data Breach Costs

This group focused on ways to measure the cost of a security breach. They examined indicators of impact, whether initial, downstream or cascading. They also identified characteristics of data breach events that would require additional losses to be calculated. Measureable attributes of these subsets of data breach loss calculations are listed in a table corresponding to each category. Note that the losses described in the first table represent the minimum set of attributes that are common to all breach losses, and so losses for data breaches in subsequent tables should be added to those in the first table. Of course, any given breach may have unique loss characteristics, so it is to be expected that real loss calculations would combine items from multiple tables among others not listed.

Scenario 1. **All Security Breaches**				
Metric Name	**Measures**[2]	**Frequency**	**Formula**	**Unit**
Breach Count	#Internally_Detected_Breaches #Externally_Detected_Breaches	Increment with each occurrence	#Internally_Detected_ Breaches + #Externally_Detected_ Breaches	Count
Forensics	#Systems Cost_Per_System	Per breach	#Systems * Cost_Per_System	Currency
Investigate	Investigation_Labor Legal_Advice Internal_Staff_Time	Per breach	Sum of all measures	Currency

[2] To indicate a single, multi-word, measure description, underlines are used instead of spaces to connect the words.

Remediate	Consultants Reimage_Systems Upgrade_Systems Internal_Staff_Time	Per breach	Sum of all measures	Currency
Opportunity	Estimated_Productivity_ of_ Displaced_Staff Economic_Impact_of_Project_ Milestones_Not_Met	Per breach	Sum of all measures	Currency

As indicators, metrics in this scenario can be used to calculate the *operational cost* of a data security breach, independent of the value of information compromised. Several elements of this scenario are therefore reusable (and therefore referenced) by scenarios created by other workshop groups. Note that the first metric, the number of breaches wherein each breach is assigned an incremental number, serves as a tracking mechanism to ensure that all appropriate data is collected per breach. The distinction between internally and externally detected breaches is relevant because breaches that are externally detected may not easily be mapped to compromised systems. This situation also surfaces in Scenario 10.

A unique scenario in data breaches are breaches that result in the compromise of personally identifiable information (PII). PII breach losses have unique characteristics, and the level of activity in each of the loss calculation areas will be dependent on the jurisdiction of the multiple government entities that regulate such events.

Scenario 2. **PII Data Breaches**

Metric Name	**Measures**	**Frequency**	**Formula**	**Unit**
Insurance	Policy_Cost Number_of_Breaches	Annual	Policy_Cost / Number_of_Breaches	Currency
Notification	Mail_or_Automation Response_Handling Internal_Staff_Time	Per PII record	Sum of all cost measures in scope of breach	Currency
PII Remediation	Credit_Coverage_for_ Victims	Per PII record	Sum of all measures in scope of breach	Currency
Regulatory	Filing_Process_Execution Amending_Executive_ Reports	Quarterly	Sum of all measures in scope of breach	Currency

As indicators, metrics in this scenario can be used in cost-benefit analysis decisions with respect to cyber security insurance and notification technology alternatives. In combination with Scenario 1, it can be used to justify the cost of PII security measures.

Note that it is not assumed that all breached will be revealed to the public or will results in data misuse that leads to fraud. However, breaches that have these consequences will have additional loss attributes. These are captured in the scenario of potential downstream impact.

Scenario 3. **Downstream Impact of Data Breaches**				
Metric Name	**Measures**	**Frequency**	**Formula**	**Unit**
Legal	Case_Preparation_Costs Court_Fees Settlement_Fees Settlement_Compliance_Process Charges_for_Missing_Contractual_ Service_Level_Agreement_Targets	Per breach	Sum of all measures in scope of breach	Currency
Fraud	Asset_Loss Legal_Prosecution	Per breach	Sum of all measures in scope of breach	Currency
Reputation	Lost_Business Public_Relations_Overtime New_Public_Relations_Campaign	Per breach	Sum of all measures in scope of breach	Currency
Regulatory	Filing_Process Execution Amending_Executive_Reports	Per breach	Sum of all measures in scope of breach	Currency

As in, and in combination with, Scenarios 1 and 2, these indicators can be used in cost-benefit analysis decisions with respect to cyber security insurance and/or protective measures. They may also be used to evaluate the cost-benefit of settling rather than trying or defending cyber security court cases.

This group also discussed reasons why an enterprise should collect data breach metrics even if the breach was not PII or Public. These involved:

- Determining risk tolerance
- Driving investment
- Planning for robustness, considerations of scale
- To calculate the cost/benefit of diverting funds reserved for notification into prevention

The enterprise panel commented that they would like to see measurements that would be necessary to measure these dimensions, but the group left that task to future work.

3.2. Malware Identification

A significant part of the discussion on malware focused on how to to best allocate resources between diff technology approaches. It is well known that different anti-virus vendors have different false positive rates, and combinations of technologies are often used to identify malware. Hence, one scenario was devoted to measures with which to compare alternative technologies, and another focused on measures of effectiveness.

Scenario 4. **Signature-Based Blacklist Malware Blocking**				
Metric Name	**Measures**	**Frequency**	**Formula**	**Unit**
Block Benefit	Number_of_Blocks Malware_Hit_Rate Averted_Remediation[3]	Per day	Number_of_Blocks * Hit_Rate * Averted_Remediation	Currency
Opportunity Block	Number_of_Blocks False_Positive_Rate Block_Opportunity_Cost[4]	Per day	Number_of_Blocks * False_Positive_Rate * Block_Opportunity_Cost	Currency

[3] See Scenario 1 for a breakdown of remediation costs.

Block Technology Cost	License_Fees Management_Servers Infrastructure_Integration Technology_Staff_Support User_Inconvenience	Per month, amoritized	Sum of all cost measures	Currency

As indicators, metrics in this scenario can be used in cost-benefit analysis decisions with respect to signature-based blocking technology. The product that had the highest malware hit rate and lowest false positive rate at the lowest Block Technology Cost should be preferred.

Scenario 5. **Data Leakage**

Metric Name	**Measures**	**Frequency**	**Formula**	**Unit**
Egress Monitors	#Devices_Hitting_Known_ Bad_Sites #Gateways_Used_for_ Sensitive_Data_Exfiltration	Per instance	Existence test	True/False
External Reports	Presence_of_Enterprise_ Data_Found_on_Known_ Malware_Operator_Sites	Per instance	Existence test	True/False

Metrics in this scenario are an independent indicator that can be used to determine whether or not an existing combination of anti-malware technology is effective. Of course, where these metrics yield "true" results, the instance of data leakage must be investigated to determine the root cause, which may or may not be malware. Regardless, where data is known to have been compromised, these metrics should be folded into the Security Breach metrics described in Scenarios 1-3.

The group also discussed the inadequacy of using blocks as a unit of measure in Scenario 4 because multiple blocks may be due to a single piece of malware on a single device, or due to the bad behavior of a single user. The concepts that *block rates* should be substituted for blocks was discussed, but the concept was not fully fleshed out.

3.3. Vulnerability Management

The mission of this group differs from that of the malware identification group in that it was focused on the mitigation rather than the identification of vulnerabilities (or "vulns"). The idea is that there are always vulnerabilities, and metrics should be used to make decisions about which ones to fix. They also faced the scenario wherein multiple vulnerabilities should be fixed, but scarce resources require decisions on the priority of one fix over another. The vulnerability management group had three types of decisions in mind:

---

[4] See Scenario 1 for a breakdown of opportunity costs.

- Focus on most important systems
- Use budgets effectively
- Measure good IT operations

The last is important because traditional vulnerability management metrics count the number of vulnerabilities found in systems. Yet if this approach is used and no vulnerabilities are found, systems cannot be declared to be invulnerable because they tests may not include vulnerabilities that are in the systems, and the test themselves often yield false negatives.[5] As these measures cannot be practically applied to claim that security is good, they have been mocked as "badness-ometers," a scale on which every measure is bad, with no measure of good.[6] Nevertheless, the group had a hard time coming up with prioritization metrics without including badness-ometers, and the first two goals are merged into one scenario in the table below.

Scenario 6. **Priority Management**

Metric Name	**Measures**	**Frequency**	**Formula**	**Unit**
System Value	System_Transaction_ Revenue Loss_Avoidance[7]	Daily	Sum of measures per system[8]	Currency per System List
Sensitivity	System_Connects_to_ Sensitive_Data	Daily	Existence test	System List
Vulnerability Level	CVSS Scores[9] Environmental_Factors	Per vulns	Use CVSS Scores, which specify Environmental_Factors to map onto a three-level ordinal scale	One of: (High, Medium, Low)
Badness-ometer	Total_Number_Target_Systems Target_Systems_with_Known_ Vulns Target_Systems_with_Severe_ Vulns	As testing schedule permits	Match Total_Number_ Target_Systems to System Value and Sensitivity lists sorted by currency and data sensitivity, filter by Target_Systems_with_ Severe_Vulns, breaking ties with higher Vulnerability Levels	Ordered list of systems

---

[5] Doupé, A., M. Cova, and G. Vigna, *Why Johnny Can't Pentest: An Analysis of Black-Box Web Vulnerability Scanners in Detection of Intrusions and Malware, and Vulnerability Assessment,* Lecture Notes in Computer Science, C. Kreibich and M. Jahnke, Editors, Springer Berlin Heidelberg, 2010, p. 111-131.
[6] McGraw, G., *Software Security*, Addison-Wesley, 2006.
[7] Measured using assumptions that the vulnerability was exploited and corresponding baseline losses from Scenario 1.
[8] The team acknowledges that the definition of "system" needs work, it may actually amount to application or business technology process.
[9] Mell, P., K. Scarfone, and S. Romanosky, *A Complete Guide to the Common Vulnerability Scoring System Version 2.0*, 2007, Forum of Incident Response and Security Teams (FIRST).

Note that it is important to measure the total number of target systems in the badness-ometer metric because this may be used to ensure that no system will escape the measurement process. As indicators, metrics in this scenario can be used to set priorities for vulnerability remediation. Note that there is no assumption that remediation exist, nor that the remediate activity chosen will be effective. Some aspects of remediation effectiveness metrics are addressed in the Security Program Effectiveness Scenario number 9.

The third goal, that of supporting technology operations decision-making with respect to closing vulnerabilities, emerged as a unique scenario, though not fully fleshed out. Where vulnerabilities are so numerous that allocated resources cannot cover the highest, then the metrics from Scenarios 1-3 in combination with assessments on incident likelihood and remediation effectiveness may be used to evaluate the cost-benefit of additional resource allocation. This obviously covers more ground than vulnerability management, and is essential to facilitate vulnerability management. It reflects the group's conclusion that good security is not likely in the absence of sound technology operations.

3.4. System Development Controls

This group looked at systems and software development lifecycle (generically referred to hereafter as "SDLC") security control decisions. The idea was to come up with a few metrics that show which development activities result in fewer security incidents.

Scenario 7. **Activities to Include in SDLC**

Metric Name	**Measures**	**Frequency**	**Formula**	**Units**
Requirements	#Identified_Defects #Severe_Defects #False_Positives Cost	Per release *and* Per Developer	#(Identified_Defects *mapped to* Identified_Defects i*n rows above)* Cost / (#Identified_Defects - #False_Positives) Cost / #Severe_Defects	#New_Defects #Old_Defects *and* Currency per (Severe) Defect
Code Review	#Identified_Defects #Severe_Defects #False_Positives Cost	Per release *and* Per Developer	#(Identified_Defects *mapped to* Identified_Defects i*n rows above)* Cost / (#Identified_Defects - #False_Positives) Cost / #Severe_Defects	#New_Defects #Old_Defects *and* Currency per (Severe) Defect
Abuse Case Tests	#Identified_Defects #Severe_Defects #False_Positives Cost	Per release *and* Per Developer	#(Identified_Defects *mapped to* Identified_Defects i*n rows above)* Cost / (#Identified_Defects - #False_Positives) Cost / #Severe_Defects	#New_Defects #Old_Defects *and* Currency per (Severe) Defect
Static Analysis	#Identified_Defects #Severe_Defects #False_Positives Cost	Per release *and* Per Developer	#(Identified_Defects *mapped to* Identified_Defects i*n rows above)* Cost / (#Identified_Defects - #False_Positives) Cost / #Severe_Defects	#New_Defects #Old_Defects *and* Currency per (Severe) Defect
Dynamic (& Fuzz) Analysis	#Identified_Defects #Severe_Defects #False_Positives Cost	Per release *and* Per Developer	#(Identified_Defects *mapped to* Identified_Defects i*n rows above)* Cost / (#Identified_Defects - #False_Positives) Cost / #Severe_Defects	#New_Defects #Old_Defects *and* Currency per (Severe) Defect

User Acceptance Tests	#Identified_Defects #Severe_Defects #False_Positives Cost	Per release *and* Per Developer	#(Identified_Defects *mapped to* Identified_Defects i*n rows above)* Cost / (#Identified_Defects - #False_Positives) Cost / #Severe_Defects	#New_Defects #Old_Defects *and* Currency per (Severe) Defect
Penetration Tests	#Identified_Defects #Severe_Defects #False_Positives Cost	Per release *and* Per Developer	#(Identified_Defects *mapped to* Identified_Defects i*n rows above)* Cost / (#Identified_Defects - #False_Positives) Cost / #Severe_Defects	#New_Defects #Old_Defects *and* Currency per (Severe) Defect
Operations Reports	#Identified_Defects #Severe_Defects #False_Positives Cost	Per release *and* Per Developer	#(Identified_Defects *mapped to* Identified_Defects i*n rows above)* Cost / (#Identified_Defects - #False_Positives) Cost / #Severe_Defects	#New_Defects #Old_Defects *and* Currency per (Severe) Defect

As indicators, metrics in this scenario can be used in two ways. The first is to add up all currency units to identify the total cost (in terms of the staff time, technology, and technology support devoted to the activity) of software security efforts per developer or per software release. The second, and more informative, would be to correlate software security improvement with activities typically recommended to be performed in the system development lifecycle. If different software projects use subsets of the scope of available activities, then the projects can be compared to see if some combinations are more effective overall than others. These metrics may also indicate developer and development manager quality, as they are reused in Scenario 8.

Scenario 8. **SDLC Assessment**

Metric Name	**Measures**	**Frequency**	**Formula**	**Unit**
Security Training	Requirements Design Secure_Coding	As new techniques evolve	For each developer, which modules taken and passed	True/False per Developer
Software Management	Scenario 7 Metrics for all software releases under a given software manager	Quarterly	Use available criteria to rate each software manager, focus on finding defects early and lowering total cost	Ordinal Software manager rank
Impact	Operations_Defects[10] Remediation[11]	Per incident, per software manager	Combine with software management rank to charts historical trends	Currency

As indicators, metrics in this scenario can be used to correlate software security quality metrics from Scenario 7 with developer training programs and software management organizations, potentially to support decisions for developer training and organizational improvement. These metrics may also be used to assign developers and software organizations to security-critical system components.

Discussion of this topic included potential strategies to maximize the cost-benefit of using independent penetration tester to minimize the dependency on the (in)experience of

[10] Measured as in the Operations Reports of software security defects metrics of Scenario 7.
[11] Where discovered during a breach, measured using the cost of security breach metrics from Scenarios 1 thru 3. Where no breach has been known to occur, measured using the Remediation metrics of Scenario 1.

individual testers. There were also concerns about the software development environment itself that remain to be addressed. For example, does software development in the cloud increase the potential for accidentally or maliciously introduced vulnerabilities? Moreover, it was acknowledged that, as these indicators are all defect-driven, they are all badness-ometers, and thus cannot be used to declare that software is secure, just that is it not known to be not secure.

3.5. Information Security Program

This group operated on the principal that the effectiveness of an information security program should be measured by outcome. This typically means adequate protection of information and infrastructure and prevention of security breaches. Hence, the group chose to focus on incident handling rather than controls to determine effectiveness. It is assumed that a design for security exists, that controls correspond to the design, and that the program has a method of identifying deviation from those controls. The team considered two scenarios, that of control effectiveness, and that of control improvement through information sharing.

Scenario 9. **Control Effectiveness**

Metric Name	**Measures**	**Frequency**	**Formula**	**Unit**
Compromised Controlled Devices	#Controlled_Devices (CDs) #Compromised_CDs	Daily, count devices only once per day, upon compromise	#Compromised_CDs / #Controlled_Devices	Percent Compromised CDs
Mean Time to Detect	Time_CD_Compromised Time_CD_Compromise_ Detected #CD_Compromises	Upon occurrence, As well as aggregates for trend analysis	Sum over CDs (Time_CD_Compromise_ Detected - Time_CD_Compromised) / #CD_Compromises	Minutes
Time to Triage	Time_CD_Compromise_ Detected Time_CD_Compromise_ Response_Decision #CD_Compromises	Upon occurrence, As well as aggregates for trend analysis	Sum over CDs (Time_CD_Compromise_ Response_Decision - Time_CD_Compromise_ Detected) / #CD_Compromises	Minutes
Time to Stabilize	Time_CD_Compromise_ Response_Decision Time_CD_Impact_ Averted #CD_Compromises	Upon occurrence, As well as aggregates for trend analysis	Sum over CDs (Time_CD_Impact_ Averted / Time_CD_Compromise_ Response_Decision) / #CD_Compromises	Minutes
Time to Report	Time_CD_Compromise_ Detected Time_CD_Compromise_ Reported #CD_Compromises	Upon occurrence, As well as aggregates for trend analysis	Sum over CDs (Time_CD_Compromise_ Reported - Time_CD_Compromise_ Detected) / #CD_Compromises	Minutes

Note that the count of compromised devices, although counted daily, is not a point of time, end of day count. Instead, it is the number of controlled devices that were in a compromised state at any time during the day. This indicates an unintended change in a security attribute, the controlled device is thus out of control. This differs from the #CD_Compromises per day measure, which would include every instance of CD compromise, no matter how many times per day. Note that the time of detection must allow for a response. For example, a log entry that is not monitored would not count as a detection until and unless some human read it, or some automated process triggered a recovery response or an alarm based on it. As it is often the case that multiple compromises will be detected in a short amount of time, multiple compromises may correlate to a single stabilize and/or report activity. These should nevertheless be measured individually so that distinct events may be later analyzed in aggregate from multiple angles.

This overlaps with the third goal of the Vulnerability Management group, that of supporting technology operations decision-making with respect to closing vulnerabilities. The difference is that not every incident to which a security group must respond is based on a known defect. Often the root cause is not known, and may in fact be authorized access not considered to be a vulnerability at the time of system design. The “time to stabilize” may that require innovative measures to adapt to an unforeseen circumstance for which there is no obvious solution.

The next scenario considered by this group was the extent to which a security program can assimilate intelligence (intel) on known attacks on other organizations to protect itself from similar attacks. In this scenario, it is assumed that an organization has a way to make use of such intelligence to determine whether their own systems are similarly vulnerable. However, unlike Scenario 9, externally reported intelligence does not necessarily mean that vulnerable systems are compromised. The challenge for a security program is to identify whether or not the organization is vulnerable, as well as whether or not a breach has occurred. If it is determined the organization is vulnerable, but no breach has occurred, the report should be folded into the vulnerability management metrics of Scenario 6 and the Operations Reports of Scenario 7. If it is determined that the device is vulnerable, and that vulnerability conflicts with the security program’s current definition of a controlled device, it should be folded into the metrics of Scenario 9. If it is determined that a breach had occurred, it should be folded into the Security Breach metrics of Scenarios 1-3. Scenario 10 is thus represents a bridge between external information sharing programs and the internal security program.

Scenario 10. **Information Sharing**				
Metric Name	**Measures**	**Frequency**	**Formula**	**Unit**
Intel-Driven Detection	Time_of_Intel_Report #CDs_in_Scope_of_Intel #Vulns_in_Scope_of_Intel #Breaches_in_Scope_of_Intel Time_to_Identify_Vulns Time_to_Identify_CDs Time_to_Identify_Breaches	Upon intel report	If #Breaches_in_Scope_of_Intel > 0, then (Time_to_Identify_Breachess - Time_of_Intel_Report) Elseif #CDs_in_Scope_of_Intel > 0, then (Time_to_Identify_CDs - Time_of_Intel_Report) Elseif #Vulns_in_Scope_of_Intel > 0, then (Time_to_Identify_Vulns - Time_of_Intel_Report)	minutes
Breach Detection from Sharing Intelligence	#Breaches[12] #Breaches_Identified_via_ Intel[13]	Upon intel report *and* aggregate trends	# Breaches_Identified_via_ Intel / #Breaches	Percent
CD Detection from Sharing	#Controlled_Devices[14] #Intel_Exploitable_CDs	Upon intel report *and* aggregate trends	#Intel_Exploitable_CDs / #Controlled_Devicess	Percent
Defect Detection from Sharing	#Identified_Defects[15] #Intel_Exploitable_Defects	Upon intel report *and* aggregate trends	#Intel_Exploitable_Defect / # Identified_Defects	Percent

As indicators, metrics in this scenario can be used to determine the ability of the security program to assimilate information on new threats. An obvious example is a vendor report of a newly introduced security patch. To some extent, these may also be used to determine the utility of membership in intelligence sharing organizations or vendor-provide cyber threat intelligence services. If the majority of incidents in an organization seem to originate from external reports, it may also be indicative of the need for a more proactive security program.

Members of this group commented that for these program effectiveness measures to be realized, that automated detection techniques for both configuration drift (for Scenario 9) and signature search (as may be required in Scenario 10) may need to advance "an order of magnitude above" where we are today.

[12] This corresponds to the Breach Count in Scenario 1.
[13] This would be a subset of Scenario 1's externally identified breaches, those that were reported via information sharing activities as opposed to other external reports, for example, customer complaints.
[14] Measured as in Scenario 9.
[15] Measured as Operations Reported Defects as in Scenario 7.

3.6. Cyber Security Risk

All the metrics discussed in the workshop may be generically referred to as security risk metrics. The group focused on risk of adopting new technology or evaluating an existing one. They adopted a case study approach to using security metrics to analyze a new technology introduction scenario. The example they chose was mobile device deployment.

Scenario 11. **Mobile Device Deployment Decisions**				
Metric Name	**Measures**	**Frequency**	**Formula**	**Unit**
Environmental	Threat_Actors Threat_Actions Known_Attack_Targets Attack_Frequency Industry_Alerts Type_of_Organization Geography	continuous	Map Geography and Type_of_Organization to Threat_Actors Threat_Actions Known_Attack_Targets Attack_Frequency Industry_Alerts	Probability of being a target
Mobile Device Management (MDM)	Mobile_Device_Config_Drift Rate_of_Drift Severity_of_Deviation Vendor_Vuln_Reports Available_Patches	continuous	Ascertain confidence level in control environment	Confidence level
Help Desk	Help_Desk_Trouble_Ticket_ Fields_Related_to_Device Authorized_Device_User_List Authorized_Device_ Application_List	continuous	No direct link can be assumed, but patterns of help desk calls related to mobile devices may be analyzed and compared with device usage patterns	Probability of device misuse
Asset Monitoring	Transactions_on_Assets_ Affected_by_Device_ Activity Authorized_Device_User_List Authorized_Device_ Application_List Expected_Device_Usage_ Patterns Actual_Device_Usage_ Patterns	continuous	No direct link can be assumed, but patterns of underlying asset movement (e.g. orders, payments, shipments, etc) using mobile devices may be analyzed and compared with device usage patterns	Probability of device misuse

As indicators, metrics in this scenario are expected to provide threat, control, and asset information to inform risk decisions with respect to using the new technology. With such a broad charter, they are more varied that those in previous scenarios. Because of the assumption that the technology is new, there is discomfort on relying on verification of secure configuration, but more emphasis on situational awareness over the entire end-to-end mobile landscape. In addition, thought the group noted that vendor reports are often a useful type of risk indicator, but that their interpretation is at risk due to their reluctance to adopt a common vocabulary with respect to security risk (such as the CVSS[16]). They

[16] Mell, Ibid.

also note that data leakage metrics such as those in Scenario 5 are useful in identifying security risks in the Mobile environment.

3.7. Business Impact

Although the data breach cost group did specify metrics to quantify the business impact of a security breach, that group's focus was only on incidents. This business impact group more broadly considered the business impact of security in dimensions other than incidents. Not all security business impact is negative. For example, a security program may prevent losses due to operational mistakes as well as internal fraud. The group chose to focus on scenarios where security is obviously part of service delivery, and sought high-level metrics that would tie security metrics to customer expectations for business partnerships. In these relationships, customer typically have regulatory requirements to review vendor security, and vendors must therefore expend resources on not only on security programs but on outward-facing customer security assurance measures.

Scenario 12. **Business Impact**

Metric Name	**Measures**	**Frequency**	**Formula**	**Unit**
Reactive Evidence	#Different_Customer_Security_Surveys * Cost_of_Completing_Survey Legal_Contract_Vetting_Costs + #Days_to_Produce_Evidence Expected_Customer_Revenue_Per_Day	Per contract	(#Different_Customer_Security_Surveys * Cost_of_Completing_Survey) + Legal_Contract_Vetting_Costs + (#Days_to_Produce_Evidence * Expected_Customer_Revenue_Per_Day)	Currency
Proactive Evidence	Cost_of_Producing_Independent_Security_Audit_Report Price_of_Agile_Security_Response #Customers_with_Contractual_Security_Requirements	Measure per contract, trend over time	(Cost_of_Producing_Independent_Security_Audit_Report + Price_of_Agile_Security_Response) / #Customers_with_Contractual_Security_Requirements	Currency
Parasite Load	Breach_Frequency Customer_Monthly_Loss_Post_Breach Expected_Customer_Revenue_Per_Month	Measure per month, trend over time	Breach_Frequency * Customer_Monthly_Loss_Post_Breach * Expected_Customer_Revenue_Per_Month	Currency

As indicators, metrics in this scenario should help executives decide how to champion security measures that will minimize customer security risk while maximizing profitability. It should help focus on auditable security measures on issues of importance to service delivery and customer satisfaction. As technology services reach maturity levels capable of sustained service delivery, they would be expected to find lower costs in proactive approaches to producing security evidence

4. Summary and Next Steps

Although workshop participants were provided with no parameters other than reminders of published reports with which most were familiar, they converged on units of measure for their assigned areas. For instance, workshop participants concluded that malware remediation effectiveness is measured best in currency while information security program effectiveness is best measured in time. Although these concepts are fundamental to technology management, they have not traditionally been highlighted in security metrics frameworks.

Workshop participants also concluded that vulnerability management is an exercise in prioritization and secure development is an exercise in correlation. While these two ideas are not particularly ground-breaking, neither do that map neatly onto current industry practice in security metrics. Rather, in most security metrics programs, security measures are assumed to be effective, and deviation from planned activities in vulnerability remediation and secure development are always considered weaknesses.

Etc etc etc – reviewers, please chime in with your own conclusions!

Appendix: Participants

Workshop Participants:

Jim Acquaviva	nCircle
Phil Agacoli	Cox Communications
Anthony Arrott	Trend Micro
Wade Baker	Verizon
Jennifer L. Bayuk	Jennifer L. Bayuk, LLC
Chris Berry	Sensage Services
Nathaniel Boggs	Columbia University
Stephen Boyer	BitSight Technologies
Katherine Brocklehurst	nCircle
Krag Brotby	Brotby & Associates
David Charing	Canadian Imperial Bank of Commerce
Steve Christey	MITRE
Anton Chuvakin	Gartner
Myles Conley	Auspices LLC
Earl Crane	National Security Staff, The White House
Keesha M. Crosby	Tri-Guard Risk Solution, LTD
Fred Doolittle	Chevron Information Technology Company
Steve Dotson	Travelport
Thomas Elegante	Zions Bancorporation
Jussi Eronen	CERT-FI
Matthew H. Fleming	Homeland Security Studies and Analysis Institute
Patrick M. Florer	Risk Centric Security, Inc.
Doug Foster	USG
Summer C. Fowler	Carnegie Mellon University
Gary Golomb	Cylance, Inc.
Grant Hansen	Zions Bancorporation
Paula Hant	salesforce.com
Lance Hayden	Cisco
Josh Huston	Exultium
Jay Jacobs	Verizon
Andrew Jaquith	Silversky
Jack Jones	CXOWARE, Inc.
Ramon Krikken	Gartner
Jason Leuenberger	Starbucks
Pete Lindstrom	Spire Security, LLC
Ivan Macalintal	Trend Micro
Michael Makstman	Kaiser Permanente
Robert Markel	Virgin America
Raffael Marty	pixlcloud
Adam Montville	Tripwire, Inc.

Bill Telletier	LMIG
Alex Proskura	Auspicatus
Andy Rappaport	CORE Security
Michael Roytman	Risk I/O
Bob Rudis	Liberty Mutual
Ben Sapiro	The Dominion
Mahesh Saptarshi	Symantec
Aaron Schaub	State Auto Insurance
David F. Severski	Seattle Children's
Lindsey Smith	Tripwire, Inc.
Wyman Stocks	NetApp
Salvatore J. Stolfo	Columbia University
Morey Straus	VMware
Russell Thomas	George Mason University
Ryan Ward	Avatier Corporation
Evan Wheeler	Omgeo
Suzanne Widup	Verizon
Walt Williams	Lattice Engines
Mathew Woodyard	Zions Bancorporation
Kai Yu	Trend Micro

Facilitators:

Facilitator 1:	Data Breach Costs	Ben Shapiro
Facilitator 2:	Malware Identification	Patrick Florer
Facilitator 3:	Vulnerability Management	Andy Jaquith
Facilitator 4:	System Development Controls	Evan Wheeler
Facilitator 5:	Information Security Program	Matt Fleming
Facilitator 6:	Cyber Security Risk	Bob Rudis
Facilitator 7:	Business Impact	Myles Connelly

Lightning Talks:

Pete – Please list speakers with name and affiliation

Enterprise Panelists:

Jennifer Bayuk	Jennifer L Bayuk, LLC
Fred Doolittle	Chevron
Steve Dotson	Travelport

Data Publisher Panelists:

Wade Baker	Verizon
Steve Christey	MITRE
Andy Jacquith	SilverSky

Metricon 8 Conference Committee:

Pete Lindstrom, Chair	Spire Security
Bob Rudis	Liberty Mutual
Walt Williams	Lattice Engines
Chris Porter	Verizon
Gunnar Peterson	Arctec Group

Metricon Steering Committee:

Jennifer Bayuk	Jennifer L. Bayuk, LLC
Dan Geer	InQTel
Andrew Jaquith	SilverSky