Aggregating, Normalizing and Evaluating Cyber Exposures and Controls to Measure Security Posture and Residual Risk

Primary Author: Duncan Molony

Contributors: Norma Lucero, Patrick Hymes

Abstract & Introduction

Information and cyber security continue to grow in importance, influence and complexity. As organizations deploy and utilize a plethora of different tools, services, and applications to scan, monitor, evaluate, and assess various information technology components on their premises or cloud environments to identify security concerns and validate controls, understanding the outputs from these and the insights about overall security based on their output can be very difficult. One important utilization of the output of these different products lies in the technical detail they provide, which aids in the remediation or mitigation of identified issues. Another usage of the output of these tools is by way of providing the organization's leadership with an aggregate view in how secure their environment is and how much risk is carried by their technical products and solutions. However, the multitude and variety of security data presents a daunting task when evaluating the large amounts of data to derive a clear assessment of consolidated security posture and a quantitative, repeatable way of measuring inherent and residual risk.

When all the data from the various products, processes, services, and controls is collected, normalized, and evaluated at the most granular level using a consistent set of criteria, or factors, across all the information security domains, a flood of seemingly disparate data starts to become informative and meaningful regarding the health of the overall security of a system. Furthermore, when these results are aggregated using a structured and comprehensive model to combine the different issues with an appropriate weighting across the different security domains, the result can provide a single customizable measure or posture 'score' reflecting security posture as well as quantitative measure of risk relevant to the scope of the target system being assessed (device, application, network, or even entire enterprise).

With the often-overwhelming number of remediations required it can be a difficult task to properly prioritize which systems and issues to remediate first. By utilizing a common set of factors and a common methodology to calculate security posture, organizations can better prioritize limited resources to address more critical systems and issues first to strategically improve the measured security posture of the system.

Key actions aimed at improving the resulting score would be available within the detailed data behind the score, thereby increasing the value of the score itself by providing access to actionable guidance of remediations. Continued iterations of refreshing the data, measures, and implementing the actionable guidance will result in improvement to the overall security posture.

By evaluating the effectiveness, importance, and coverage of compensating controls, this same methodology can be adjusted to calculate a score for inherent and residual risk. This can help organizations understand if their technology landscape is operating within the organization's risk tolerance and may help establish a risk posture if one does not already exist.

Defining the common factors

To apply a single methodology using such a wide variety of issues and data sources it is necessary to define a set of common factors which can be used to normalize the data and to assess the impact to the security posture. Each data set needs to be evaluated against the same set of factors, although the exact form of evaluation (calculation) is dependent on the data and the intrinsic characteristics of the data source. (Data from network vulnerability scanning, malware detections, and Identity & Access Management issues would be evaluated using the same set of factors; however, the calculations for those factors may be specific to the data source.)

Which factors to choose is a critical aspect of the validity and acceptance of the model and should be agreed upon by all stakeholders. The specifics of each environment need to be considered when deciding on the factors and how these factors will be evaluated and weighted within the model. Additionally, standing reviews of the methodology and factors used must be conducted at frequent intervals with all stakeholders to analyze the need for any changes.

Five factors with wide applicability and acceptance in most organizations are:

1) Severity – the severity of the issue being evaluated; (severity rating of a CVE, for example)

2) Criticality – the criticality of the system, applications, databases, etc. impacted by the issue being evaluated.

3) Age – how long an issue has been known to exist within the system, application, database, etc.

4) Exposure – this is essentially the attack surface of the issue and environment being evaluated.

5) Exploitability – is there a known active exploit, likely exploit, potential exploit, or no known exploit for the issue.

Another potential factor that should be considered for highly regulated environments with well-defined policies is Compliance. This is an evaluation of the issue of compliance to internal policy requirements and/or external regulatory or governance requirements. While compliance may not represent a direct impact to security posture, it does represent risk that could be regulatory, reputational, and potentially financial in nature. Also, Compliance may be an appropriate factor for specific data sets and can be added to the calculation if needed.

Calculating Factor Scores

Once the factors have been agreed upon, the evaluation of each factor needs to be determined. Given the significant variations in security data, these calculations will need to be flexible and adjustable to the data being analyzed and scored. Consultation with subject matter experts from each security data source and domain is necessary to ensure data is being assessed and scored properly. The factors chosen should apply to most data sets being assessed, but at times a specific factor may not be relevant for a particular data set. To maintain consistency of the model and the factors used for scoring, the model should support a factor which is irrelevant to a particular data set can being set to a neutral value so that it has no impact on the final direction of posture and risk scoring.

Now we will discuss calculations for each factor.

Severity

Industry standards can and should be used as a basis and ***starting point*** for calculating Severity of a finding so that resulting scores and ratings are relatable and relevant to external reviewers. One such standard is the Common Vulnerability Scoring System, or CVSS. This standard provides a set of factors of its own which are used to calculate a score. Not all security tools or domains use CVSS and many security tools and processes leverage text ratings or other numeric rating scales; however, it is necessary for the methodology to have numerical inputs of same scale as this provides for a more granular distinction between findings than text ratings. This granularity also allows for fine-tuning the prioritization of remediation activities. For these instances we will need to have an agreed upon numerical equivalent for the text ratings to ensure a consistent and common scale to the numerical ratings.

A further maturity of this approach would be to enhance the industry or vendor scoring by using system or enterprise specific intelligence to modify those base ratings so they are more impactful and meaningful to the organization. This also allows an organization to evaluate the thresholds for the scoring and adjust if deemed appropriate to the environment's requirements. Organizations may adjust the thresholds between ratings, expand the scoring scale, or even introduce additional ratings such as "Emergency" or "Critical". The results of the organization-specific calculation should always match or increase the score from the industry standard and not decrease the score. The impact of any implemented compensating controls should not influence the severity calculation as these controls should be assessed and accounted for in terms of measuring residual risk and not overall severity of an issue.

Below table includes an example set of ratings and score ranges we will use for our sample model.

Rating	Score Range	Default Score*
Low	0-3.99	2
Moderate	4-6.99	5
High	7-9.99	8
Critical	>=10	10

*Default score is used when only text ratings are available

. To properly weight and prioritize more severe issues a severity calibration is applied to a severity rating to derive the Severity Factor. Using a modified skip Fibonacci sequence as the Severity Calibration provides meaningful deviation to create the necessary separation of final scoring to properly reflect relative impact and aide in prioritization.

Rating	Score Range	Severity Calibration
Low	0-3.99	3
Moderate	4-6.99	8
High	7-9.99	13
Critical	>=10	21

The calculation of the Severity Factor is the product of the Severity Score (SevScore) and the Severity Calibration (SevCal).

Severity Factor = SevScore X SevCal

For example, CVE 2022-22954 has a CVSS v3 score of 9.8. The Severity Calibration for this vulnerability is 13 based on our table above.

Severity Factor = 9.8 X 13 = 127.4

Compare this to CVE 2022-34903 which has a CVSS v3 score of 6.5:

Severity Factor = 6.5 X 8 = 52

Or looking at a vendor test rating of Moderate, we would use the Default Score of 5:

Severity Factor = 5 X 8 = 40

The use of the Severity Calibration creates a much more meaningful distinction between the two vulnerabilities.

Criticality

Most organizations and environments have some assets, data, environments, or zones which are more vital to their operation or more vital to protect for privacy, regulatory or competitive reasons. The organization should have a defined method to classify the criticality level of all their assets whether it be a calculated score or a simple rating system like Low, Medium, High, etc. If the latter, there will need to be an agreed upon numerical value (standard score) assigned to each rating to be used in the calculation.

For this factor, the range of the score is the calibration. If Criticality should have a larger impact the range can be extended, for example setting highest score to 10 instead of 5 by applying the Criticality Calibration to the Calculated Score, if available, or the Standard Score if only text ratings are available.

Table below is a sample of starting values (both calculated and standard) and the calibration factor of 2. This calibration extends the factor score range from 0-5 to 0-10 to enhance the impact of this factor for more critical systems.

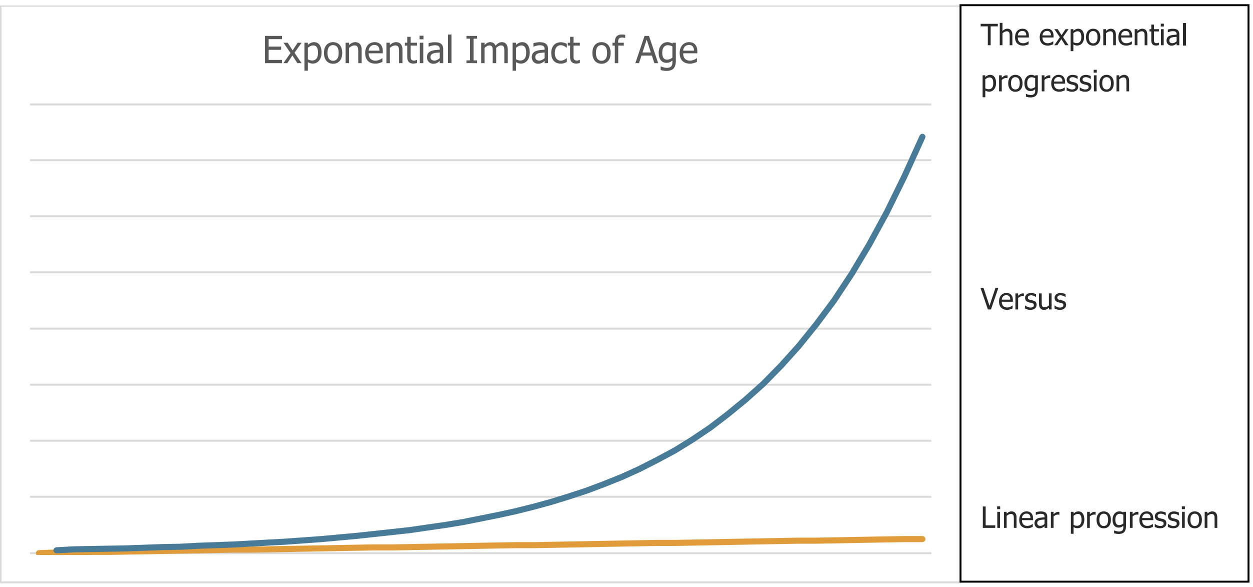
Rating	Calculated Score	Standard Score	Criticality Calibration	Criticality Factor
Low	0-2	1	2	2
Medium	2-4	3	2	6
High	5	5	2	10

The Criticality calibration factors can also be adjusted to further emphasize more critical assets.

Age

The longer an issue exists the more likely it will cause an impact in some way. For this reason, it is important to evaluate how long an issue has been known (to the industry or in the environment) when considering impact to security posture and risk. The age of an issue can be determined in a couple of different ways: 1) when the issue was first recognized and published by a vendor or the industry (CVE's and vendor bug reports are good sources of this data); or 2) when the instance of an issue was first detected within an organization's environment or on a specific device, application, or system. This second method is more environment/organization-specific and relevant. Issues from some domains such as Access Management will likely fall in to the second method. For example, a user with excessive privileged access. In this scenario the age can begin when the issue is identified, or if change records are available, when the excessive privileged access was granted.

To properly represent the increasing risk of age, an appropriate calculation that is not a linear progression of impact should be considered to accurately capture this increased risk. Exponential calculations against a numerical constant can result in the curve representing the desired impact to the security posture scoring. (see chart below).



Applying the exponential approach drastically increases the impact of age the longer an issue goes un-remediated.

For Age factor we are using the exponential function of Euler's number (e) to result in an exponential curve representing increasing impact of the age of a vulnerability. Euler's number is commonly used to calculate decay or growth over time and used in business related calculations such as compound interest. This makes it suitable for the purposes of calculating the impact of increasing age of an issue.

Age Factor = e^x (where x is the age in days of the issue)

At the factor level, one or more calibration points can be introduced into the calculations to allow for adjustments to scoring and thereby prioritization. Let's explore the Age Factor and how it can be adjusted based on expected remediation behaviors within an organization.

Company ABC has defined and published policies for vulnerability management, including clearly defined remediation time frames for high-rated vulnerabilities. The current expectation is most high vulnerability instances will be remediated within 45 days and there is a 15 day grace period before organizational consequences are applied. So in this environment it is expected all high vulnerabilities be remediated within 60 days to avoid organizationally directed consequences. In our factor calculation this can be incorporated as a calibration point (AgeCal).

The AgeCal calibration point will be leveraged in the exponent to help tie the resulting age factor to the policy and expectations of the organization.

$$\text{Age Factor} = e^{(x/\text{AgeCal})}$$

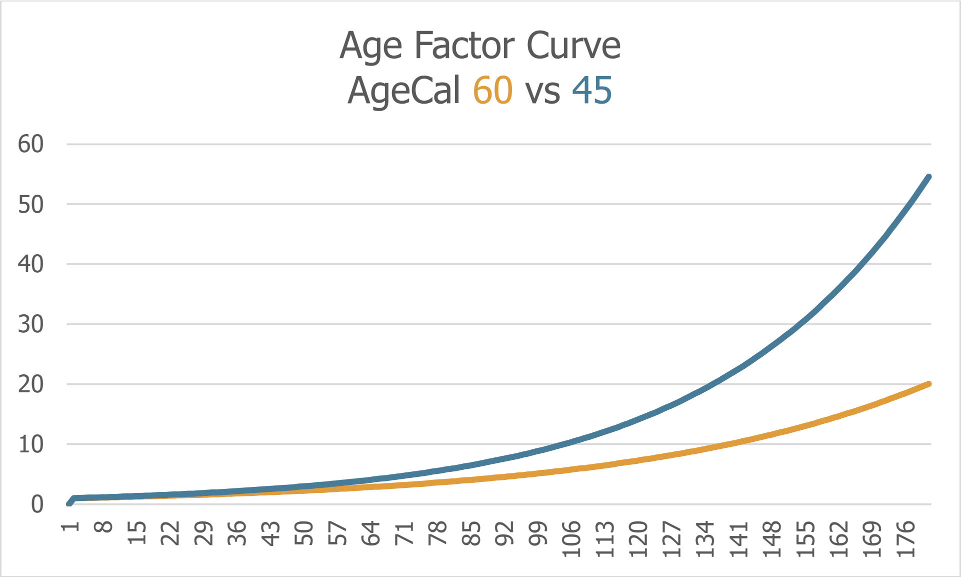
As a practical example of this factor we will consider a vulnerability with an age (based on first detected within the environment) of 58 days. AgeCal is 60 based on Company ABC's policies.

$$\text{Age factor} = e^{(58/60)} = 2.629$$

If the policies and/or priorities of Company ABC changed and the company now wanted to have all vulnerabilities remediated within 30 days with a 15 day grace period, we can change the AgeCal to 45 to drive up the impact of the Age Factor to the overall security posture score.

$$\text{Age factor} = e^{(58/45)} = 3.629$$

Evaluating the results of different AgeCal settings across a range of ages from 1 to 180 provides a good visual of the impact of calibrating the AgeCal number. In the example above the increase in Age Factor score from 2.629 to 3.629 may seem small but will have a notable impact once combined with all the other factors we have defined.



For other issues such as the excessive privileged access we discussed earlier, the Age Factor would likely be set significantly lower as these types excessive access issues should be addressed more swiftly to limit risk. For instance, if we have an issue of excessive privileged access the organization may require it be remediated within 2 days. This would set our Age Cal to 2, so any excessive privileged access issue open beyond 2 days would see a significant increase in the Age Factor. As an example let's examine an excessive privileged access issue that has been in place for 7 days. We would use 7/2 as our exponent for e:

$$\text{Age factor} = e^{(7/2)} = 33.115$$

Comparing this to the vulnerability age factors above it is clear how much more quickly age can impact the Age Factor and ultimately the Security Posture score.

Exposure

Exposure is the evaluation and measure of attack surface. The attack surface is both the system being attacked and the issue being attacked, therefore exposure should be thought of in two ways

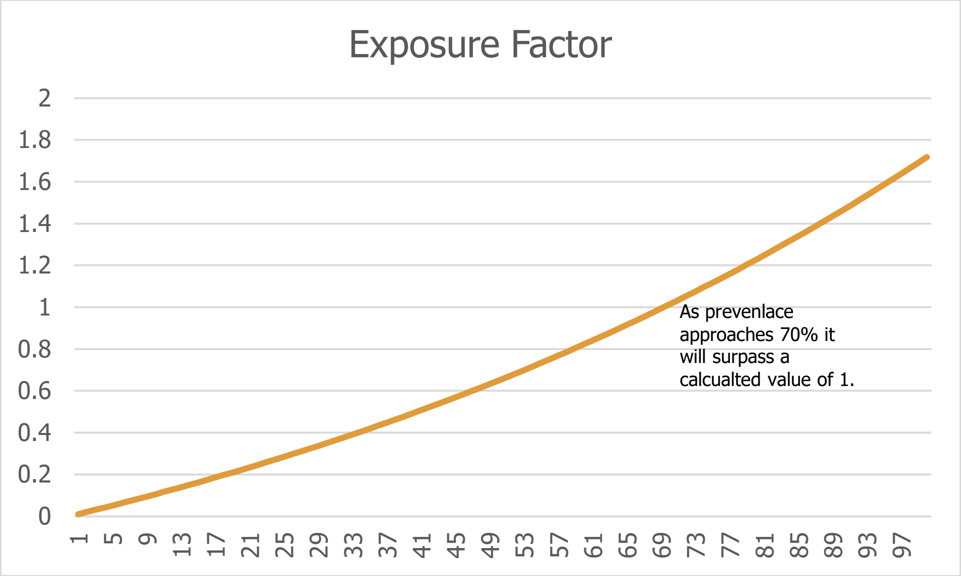
A) Issue Exposure: Exposure of the issue within the enterprise (how many instances of the issue exist in the enterprise);

B) Technology Exposure: Size of the system, application, database with the issue relative to the enterprise (10 out of 1,000 servers, 3 of 5 databases, etc.);

By evaluating both aspects of exposure we can focus the impact of this factor within each unique environment on issues which represent larger percentage of attack surface within the organization and on systems, applications, or databases which account for large percentage of the technology attack surface (total infrastructure).

As with the Age Factor, the larger the attack surface the larger the impact to the overall security posture and risk to the environment. One option is to use straight percentages for these factors (10/1000 = .01, 3/5=.6). This linear progression of resulting scores does not appropriately reflect the increasing impact of large attack surfaces. For this reason, the Exposure Factors will utilize the natural logarithmic function of Euler's number with the exponent being the percentage of exposure to define this increasing impact of greater exposure. Since the result of Euler's number raised to even very small fractions results in a number greater than 1, we will subtract 1 from the result so the remainder is a number between 0 and ~1.718.

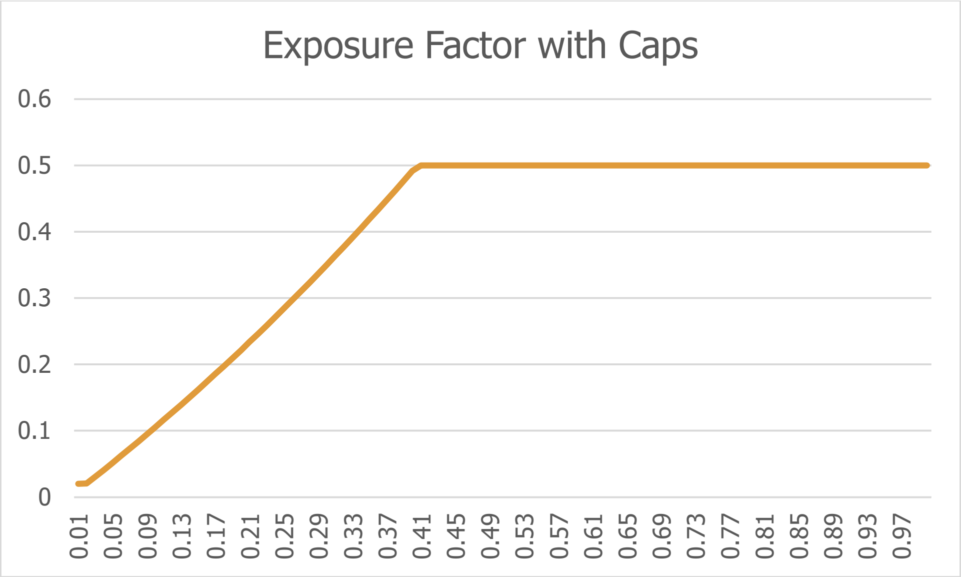
$\text{EXPFactor}= e^{(\text{IssueExpXTechExp})}-1$



As prevenlace approaches 70% it will surpass a calcualted value of 1.

As evidenced by the chart above, the Exposure Factor will exceed 1 as the exposure percentage approaches 70%. Boundaries are needed on the results of the calculation to prevent very low and very high exposure percentages from skewing the results of the Security Posture and risk scores. Field testing and modeling exercises should be performed to define the lower threshold and upper threshold appropriate to each environment. For our example model, we will set the low threshold to 0.02 and the upper to 0.5. This range should make the calculation applicable to environments of most sizes and complexities and can be used as a starting point. These calibration points can be adjusted as needed to suit the environment being measured.

Below is the Exposure factor chart adjusted for the minimum and maximum values.



Exploitability

Many types of issues can be considered exploitable including issues in code (vulnerabilities), configurations, inappropriate access or lack of controls. While the exploitability of an issue is sometimes included in calculating severity, this factor is very relevant and important to security posture and risk calculations. For this reason, we will also consider exploitability as a stand-alone factor. This should not be a complicated factor and can be defined like the table below.

Exploitability	Numerical Factor
No Exploit/Not Applicable	1
Likely exploitable but no known published exploit	1.2
Published exploit concept but no evidence of active attacks	1.5
Active Exploit in the wild	2

The first entry in the table can also be used when exploitability is not a possibility as the value of "1" will have a neutral impact on the calculation of security posture and risk.

If available, Analytic models utilizing Machine Learning and Artificial Intelligence to predict the exploitability of vulnerabilities can be used to provide a better fidelity and to improve the model. The training of such ML/AI needs to be carefully designed and the output validated prior to use and regularly revalidated.

Calculating Security Posture Score – Bringing it all together

Once the factors and calibration points have been decided, calculations can be performed against the relevant data sets. Given the wide range of potential inputs, some boundaries need to be applied to the output to constrain statistically anomalous results. Also, to make the results more easily understandable by the widest audience we will keep the values of the Security Posture between 1 and 100 – the higher the better the Security Posture. The base calculation used is:

100 - (the minimum between (SevFctr * CritFctr*AgeFctr * EXPFctr * Exploitability) and 99)

The Security Posture scores of multiple issues can be aggregated by host, application, location, environment, etc. Security Posture scores from the same Domain, like Access Management can be averaged at the host level to provide a Domain Security Posture score for the host. Likewise, multiple Domain Security Posture scores for the same host can be combined to account for more critical domains. Since some Security Domains are more relevant or important to different organizations, the Domain score can be weighted based on priority of the domain. To maintain the value range of 1-100, the Domain scores will be combined using a weighted average.

Further aggregation can be done for an application and all the hosts supporting that application to provide a Security Posture score for the application itself.

The Security Posture scores for each issue can be used to prioritize remediation activities to focus on those issues with the lowest Security Posture scores (largest negative impact to the Security Posture of the environment).

Mature implementations of this overall model with robust reporting can also provide actionable intelligence to guide the application owner or technology team on the best ways to improve the Security Posture score.

Risk Lens of this Scoring

The model can also be utilized as a methodology to score Inherent Risk. By inverting the scoring where the lower numbers represent lower risk and higher numbers higher risk, the Security Posture Score and be inverted to reflect an Inherent Risk Score.

Security Posture Score:
=100-MIN((SevFctr * CritFctr*AgeFctr * ExpFctr * Exploitability),99)

Inherent Risk Score:
=MIN((SevFctr * CritFctr*AgeFctr * ExpFctr * Exploitability),99)

Having a calculated Inherent Risk score is the first step in evaluating existing security controls and producing a Residual Risk score which can be used in many ways to help the organization manage risk.

Let’s calculate the Security Posture and Inherent Risk scores for a sample issue.

Example: A user has inappropriate Privileged access (an issue rated as Critical) to a 5 of 100 servers. This issue is not very prevalent in the environment. This issue is likely exploitable, but no known exploit activity has been detected. This access has been in place 3 days (expected remediation is 2 days), and these servers have a criticality rating of Medium.

Severity	Sev Factor	Criticality	Crit Factor	Age	Age Factor	IssueExp	ExpFactor	TechExp	Exploitability Factor	Sec Posture	Inherent Risk
10	130	3	6	90	4.4816	0.01	0.0200	0.05	1.2	16.10278	83.89722

This issue has a very high severity, and its age is makes it past the threshold which is driving down the Security Posture score and, as expected, driving up the Inherent Risk score.

Control Evaluation

Security controls provide varying degrees of effectiveness and can be more or less relevant (important) to the mitigation of risk. Another driving factor is the coverage of the control for the scope being assessed. This provides three factors to evaluate Security Controls and a methodology for a numeric value which can be used to assess the impact of the Security Control to the Risk.

Effectiveness Rating	Score
Not Effective/Not Applicable	1
Partially Effective	2
Effective	3

Importance Rating	Score
Not Important/Not Applicable	1
Somewhat Important	2
Important	3

Coverage of a control can vary depending on the control. Some technical controls are implemented at the host level or application level. Process controls are generally implemented at the organization level but can be specific to certain functions (for example: developers). An assessment of the control coverage must be done on a regular basis to provide a coverage percentage for the given control on the scope being assessed. This percentage is the third component of the Control Factor scoring:

Control Factor = Effectiveness X Importance X Coverage Percentage

The calculation will result in a range of values from 0-9.

Applying Control Factor to Inherent Risk to Calculate Residual Risk

Now that we have defined a scoring model for Inherent Risk and for Control Factor, we need to account for the impact the applicable controls will have on the Inherent Risk to calculate Residual Risk. We need to determine what Security Controls are applicable to the issue being assessed. For example, an application vulnerability may rely on multiple security controls for mitigation, including Web Application Firewall, Proxy, Separation of Duties, and several others. Each of the Security Controls will be evaluated using the Control Factor methodology defined above to result in a number between 0 and 9.

Compensating controls cannot eliminate risk; however very effective controls can significantly reduce the risk for as long as the control is in place. Since controls can fail, we also don’t want to overstate the impact of these controls on risk. In many environments there will be multiple controls which can help to reduce the risk. Many of these controls have some overlapping impact to mitigation, so subsequent controls will have a lesser impact on risk reduction. To calculate this spiraling risk reduction, we need a method to evaluate the relative impact of the controls in a successive manner and based on defined order of impact.

To determine a base method to measure a consistent but reducing impact we require an approach which provides correlation between the inherent risk, effect of the control and the residual risk. We will use the Golden Ratio often found in nature, represented by the Greek letter Phi (Φ). One way to represent the Golden Ratio:

For two values, a and b, where a is larger:

a/b = (a+b)/a

The resulting ratio is an irrational number that begins 1.618 (first 10 digits of Phi are 1.618033989)

To align the Control Factor range we have defined (0-9) to the Golden Ratio model while constraining the statistical anomalies, we will apply the following to the Control Factor

ControlFactorΦ = the lesser of 2-(ControlFactor/9) and 1.618

Result will be a number between 1 and 1.618

Control Factor	Control Factor Φ
9	1
8	1.111111111
7	1.222222222
6	1.333333333
5	1.444444444
4	1.555555556
3	1.618
2	1.618
1	1.618

Now let’s address the sequencing of multiple controls. To ensure the model allows for a significant number of controls without requiring an adjustment for each instance of evaluation, we will use a base of 10 possible controls. As more controls are applied the subsequent controls will have less impact to risk reduction accounting for overlapping impact of controls and keeping in line with the idea controls cannot eliminate risk. With these principals in mind the Sequence Ratio is calculated as the sequence number of the control divided by 10 with a maximum value of 1. All subsequent controls past 10 will have a Sequence Ratio of 1.

The order in which controls land in the sequence can be based on an evaluation of the in-scope controls and on relative importance to the issue (primary controls versus secondary), or we can simply order the controls by the Control Factor Φ from low to high. We will use the latter approach for the demonstration below.

We will now apply the Control Factor Φ and Sequence Ratio of 10 controls to the Inherent Risk. The ten controls have Control Factor Φ values of:

1	1	1.22222	1.33333	1.44444	1.55556	1.618	1.618	1.618	1.618

The calculation will evaluate the previous risk score, starting with the Inherent Risk, as a ratio of PHI (Φ) but the sum of the Sequence Ratio and Control Factor Φ will be leveraged to apply appropriate weighting for the control score and the sequence of the control.

$$\text{Residual Risk} = \frac{\text{Starting Risk (inherent risk or Residual Risk n-1)}}{\text{PHI / The lower of (Sequence Ratio + Control Factor } \Phi\text{) and 1.6*}}$$

* 1.6 is used as the maximum value here as it just less that the value of PHI which will ensure there is always some impact from the control

Using the sample issue from above with an Inherent Risk score of 83.89722, the below table is applying the formula for Residual Risk across the 10 controls we defined and evaluated.

Sequence Ratio	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
Control Factor Φ	1	1.222222	1.222222	1.333333	1.444444	1.555556	1.618	1.618	1.618	1.618
Inherent Risk	Residual Risk 1	Residual Risk 2	Residual Risk 3	Residual Risk 4	Residual Risk 5	Residual Risk 6	Residual Risk 7	Residual Risk 8	Residual Risk 9	Residual Risk 10
83.89722	57.03646688	50.13401	47.16533	46.63964	46.11981	45.60578	45.09747	44.59484	44.0978	43.6063

Stepping through the first few iterations:

Residual Risk 1 = 83.89722/(PHI/(0.1+1)) = 57.036467
Residual Risk 2 = 57.036467/(PHI/(0.2+1.222222)) = 50.13401
Residual Risk 3 = 50.13401/(PHI/(0.3+1.222222) = 47.16533

The model is adaptive in the event of a failed control or can be utilized to model the impact to residual risk of failed controls. We will examine the scenario of the failure of the primary control from the example above. Should the primary control fail, the subsequent controls would be computed by moving the Sequence Ratio of each control up one step.

Sequence Ratio	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
Control Factor Φ	1	1.222222	1.222222	1.333333	1.444444	1.555556	1.618	1.618	1.618	1.618
Inherent Risk	Residual Risk 1	Residual Risk 2	Residual Risk 3	Residual Risk 4	Residual Risk 5	Residual Risk 6	Residual Risk 7	Residual Risk 8	Residual Risk 9	Residual Risk 10
83.89722	Failed control	68.55899	64.49927	63.78039	63.06951	62.36657	61.67145	60.98408	60.30438	59.63225

The failure of the primary control has a meaningful impact on the risk reduction of not only the initial secondary control, but also the overall risk reduction of all controls. The final residual risk score in this scenario is 16 points higher (riskier) than the initial calculation with no failed controls.

Assigning the risk rating of the residual risk score is best done by running the model against large data sets over a reasonable amount of time to establish a baseline of results. These results should be assessed against the underlying security and risk data and the specifics of the system being measured. Security and Risk professionals can then assess the ranges for desired risk ratings. An example of a 5-cluster rating scale based on analysis of 6+ months of weekly iterations of millions of records:

Risk Rating	Min Score >	Max Score <
Very Low	0	10
Low	10	25
Moderate	25	40
High	40	70
Very High	70	100

It is important to tune this scale appropriately so that it aligns with the underlying data to avoid minimizing or overstating risk. In the examples cited above, both scenarios (all 10 controls and the failed primary control) would result in the residual risk being High. This is a reasonable result given the very high inherent risk score and the Control Factors of the 10 controls. If the second Control had a Control Factor of 1, like the primary control, the final residual risk score would be 36.38274 and be considered Moderate Risk.

Conclusion

Defining a consistent, structured yet flexible model to evaluate the overwhelming amount of security related data to produce repeatable and tunable scoring and actionable intelligence will help organizations manage their security posture and risk more effectively by focusing the efforts of often limited resources to prioritize those issue with the largest impact.

Even with limited deployments of the model to only high-profile Security Domains, the model can be effective identifying actions to lower risk and improve overall security posture for any environment. The flexibility and customization built within the model in the form of calibration points expands the potential value of this model. Also, as mentioned above, the actual calculations used for the defined Factors can be modified to suit the specific characteristics of the source data being evaluated; thus, providing even more flexibility to this model.