An Overview of FAIR-CAM

FAIR - Controls Analytics Model

Introduction

The problem statement

How much risk reduction did your organization get from the most recent major investment it made? How much additional risk would the organization face if, due to budget cuts, one or more controls is removed or downgraded in some fashion?

Unfortunately, defensible answers to these and similar questions have evaded the risk management profession. For a profession whose bread and butter is the application and management of controls, this fact should be a huge red flag. Clearly, something important has been missing from our professional toolkit.

This document[1] discusses the *Factor Analysis of Information Risk Controls Analytics Model* (FAIR-CAM), which fills that gap. In doing so, it lays a foundation for what is essentially a whole new chapter within the risk management body of knowledge. Please note that although this document will cover the entire model, it will only go into detail on selected parts of the model. This is simply because deep coverage of the entire model would require an entire book (which is on the to-do list).

What FAIR-CAM is – and isn't

> "*In the 19th century we had a relatively advanced understanding of anatomy, but we had a terrible understanding of physiology. We knew what was happening, but we didn't know why it was happening.*"
>
> *A retired surgeon*

FAIR-CAM is an extension of the FAIR model, which has been an open industry standard for over a decade. It specifically focuses on the control dimension of risk measurement.

But does the world need yet another control framework? The simple answer is probably not, but FAIR-CAM isn't a control framework. You see, common control frameworks describe controls (or control objectives) in a way that is roughly analogous to anatomy in the practice of medicine. They describe which controls can or should be part of a risk management program, and what those controls should look like. What they don't tell you is how those controls function, either individually or as a complex system of interdependent parts, in order to reduce or maintain risk levels. FAIR-CAM, on the other hand, provides what is analogous to medical physiology, which

[1] Readers who are familiar with the original Introduction to FAIR-CAM document will notice a few updates to the model. These changes have resulted from lessons learned in applying the model.

enables a much deeper and clearer understanding of how the control landscape works. In doing so, it also provides the means of measuring the risk reduction value of controls.

FAIR-CAM works alongside existing control frameworks and acts as a bridge between those frameworks and risk. We'll see an example of that further on.

One last point is that even though FAIR-CAM was "born" in the cybersecurity domain, it can be applied to any type of risk scenario.

Challenges

Before getting into the details of FAIR-CAM and how it can be used, it's important to understand some challenges we face.

The first challenge is that control assessments and control technologies provide most of the risk-related data we have, but very often it isn't the data we need. This is in large part due to the fact that control assessment practices and control technologies were developed without an explicit understanding of control physiology. Going forward, this situation should improve as processes and technologies evolve within an understanding of control physiology. For now however, simply recognize that empirical data may be hard to come by for some control functions. This means that for some measurements, especially in the nearer term, calibrated subject matter expert estimates and benchmarks may be the best (or only) available options. This will, however, still represent a significant improvement over current methods in the quality of our risk measurements.

Another challenge is that common control frameworks weren't developed with an explicit understanding of control physiology. As a result, some control descriptions are either vague or overly broad in their descriptions. Sometimes both. Vague descriptions make it difficult to know exactly which physiological function the control fulfills. Overly broad descriptions mean that a control may fulfill multiple physiological functions, which makes it much more difficult to translate control "ratings" into actual units of measurement.

A last challenge is that FAIR-CAM is quite a bit more complex than what most risk professionals are used to. For some people this is a show-stopper. They aren't interested in learning something new and more complex. Unfortunately, the control landscape is an inherently complex system. Therefore, FAIR-CAM isn't "making it complex," it's simply describing how the control landscape works. Some of the physicians of the mid-1800's were undoubtedly put-off by the complexity of human physiology as the medical profession began to develop that science. The point is that if we want to be able to accurately measure and cost-effectively manage our risk landscapes, we have to understand and account for control complexity.

Licensing and Use

The FAIR-CAM™ ontology is intended to serve as an international standard for controls physiology. In order to support this objective this work is licensed under a Creative Commons Attribution-Non Commercial-No Derivatives 4.0 International Public License (the link can be

found at https://creativecommons.org/licenses/by- nc-nd/4.0/legalcode). To further clarify the Creative Commons license related to FAIR-CAM™ content, you are authorized to copy and redistribute the content as a framework for use by you, within your organization and outside of your organization, for non-commercial purposes only, provided that (i) appropriate credit is given to the FAIR Institute, and (ii) a link to the license is provided. Additionally, if you remix, transform, or build upon the FAIR-CAM™ ontology, you may not distribute the modified materials. Users of FAIR-CAM™ are also required to refer to (http://www.fairinstitute.org/FAIR-CAM/) when referring to the model in order to ensure that users are employing the most up-to-date guidance. Commercial use of FAIR-CAM™ is subject to licensing through the FAIR Institute.)

The FAIR-CAM model

Controls vs. control functions

The first thing we have to understand is the difference between controls and control functions. This is actually pretty straight-forward to understand using the human anatomy versus physiology analogy. In the human body, the stomach is part of the anatomy. Its function (or one of the several functions it serves) is digestion. Switching to a cybersecurity context, awareness training is a control. A couple of the functions it serves include:

- Communicating organization policies and objectives
- Improving personnel capabilities (e.g., recognizing phishing emails)

The outcome is a reduction in human error, which results in reduced risk.

Controls definition

An explicit definition for controls is:

- Controls are anything that can be used to directly or indirectly affect the frequency or magnitude of loss from one or more loss event scenarios.

This relatively broad definition opens up the scope of what constitutes a control. As we'll see later on, this is important because our profession has not been explicitly recognizing a few key controls.

Control Functions definition

An explicit definition for control functions is:

- Control functions are the ways in which a control can affect the frequency or magnitude of loss, either directly or indirectly.

To be fair, our profession has nibbled at the edges of control functions for a long time with “prevention”, “detection”, and “response” (as well as the other functions in NIST CSF). The problem is that three, five, or seven functions doesn’t come close to capturing the complex reality of the control landscape, and does not support accurate measurement. In fact, it contributes to confusion and poor measurements.

For example, NIST CSF’s Detection Function/Continuous Monitoring Category includes Subcategories for both the detection of loss events (e.g.,DE.CM-4 Malicious code is detected) and detection of control deficiencies (DE.CM-8 Vulnerability scans are performed), which are entirely different in purpose and effect. Yet it is routine to see organizations (and cybersecurity solutions) aggregate the scores from these (and other very different functions) into overall scores. This generates results that inaccurately describe an organization’s security posture and drives poor decisions.

Two additional reasons why there’s a need for more functional granularity is that:

- Having more functions enables us to account for key dependencies between controls (which will be covered later in the document)
- Each FAIR-CAM function has a distinct unit of measurement for measuring and communicating control efficacy. We lose any opportunity for accurate measurement if we munge together control functions that have different units of measurement.

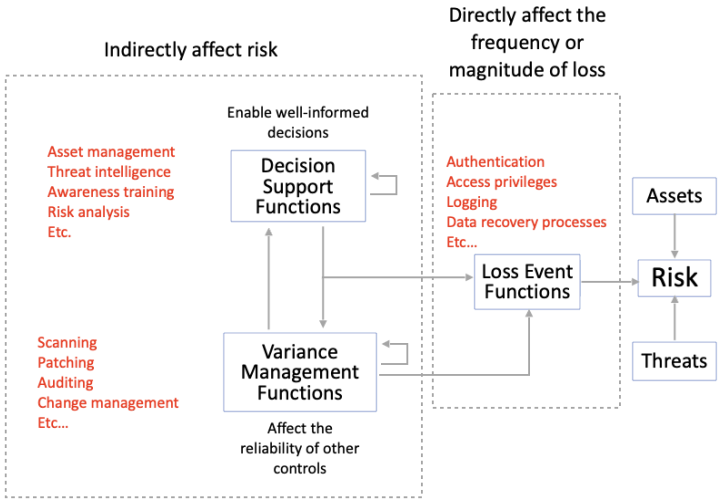
The point here is that our profession has focused on control anatomy – passwords, access privileges, firewalls, logging, incident response, backups, policies, awareness training, etc, with only a superficial, informal, and incomplete notion of control physiology. But we have to understand how controls function, both individually and as part of a complex system, in order to perform accurate risk measurement and intelligently prioritize our application of limited resources.

Functional Domains

In this section we will be covering the three sub-models (functional domains) that make up FAIR-CAM:

1. Loss Event Control Functions
2. Variance Management Control Functions
3. Decision Support Control Functions

The diagram below illustrates these functional domains and their relationships:



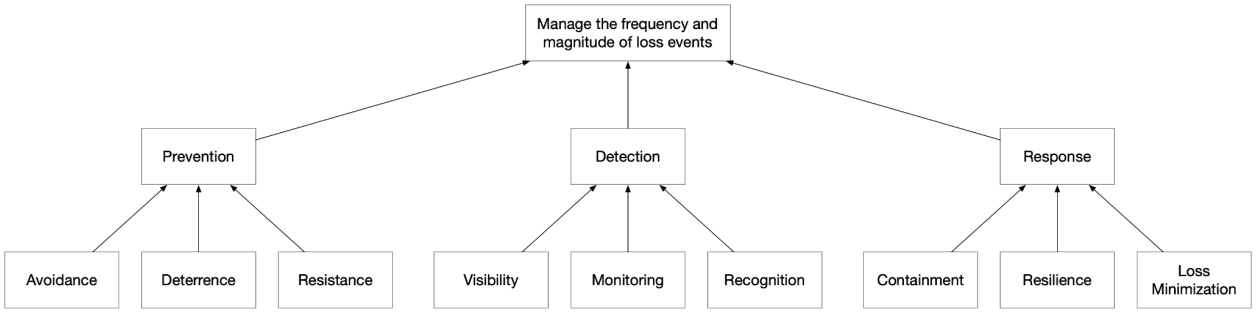
The text in red are examples of controls that fulfill one or more functions within a domain.

It's obvious that Loss Event Functions affect risk directly, and we'll spend a lot of time on that in this document. It's also pretty easy to understand how Variance Management (VM) and Decision Support (DS) controls affect risk indirectly by their effect downstream on Loss Event (LE) controls. We'll see examples of how this works, as well. What might be less obvious is how Variance Management and Decision Support controls can affect each other, and themselves. A few simple examples should help:

- Auditing (which is a VM control) can and should be applied to threat intelligence, awareness training, and risk analysis (which are DS controls). Auditing also can be applied to patching and change management (which are VM controls).
- Asset management (a DS control) informs the use of scanning and change management (which are VM controls).
- Threat intelligence (a DS control) is crucial for awareness training and risk analysis (which are also DS controls).

Loss Event Control Functions

Loss event controls directly affect the frequency or magnitude of loss.



The table below describes each of the functions in this domain, and provides a few examples of controls that fulfill each function. NOTE: The examples do not represent a comprehensive list of controls for each function!

Functional Category	Function	Description	Example controls
Loss Event Prevention	Avoidance	Reduce the frequency of contact between threat agents and the assets they might adversely affect.	• Email filtering at the perimeter • URL filtering
	Deterrence	Reduce the probability of malicious or unauthorized actions after a threat agent has come into contact with an asset. (Generally only applicable to insiders like employees, contractors, etc.)	• Activity logging • Auditing
	Resistance	Reduce the likelihood that a threat agent's act will result in a loss event.	• Authentication • Access privileges • Personnel (e.g., resistiveness to phishing attacks) • Anti-malware • Software
Loss Event Detection	Visibility	Provide evidence of activity that may be anomalous or illicit in nature.	• Logging • Anti-malware • EDR technologies
	Monitoring	Manual or automated review data provided by Visibility controls.	• Anti-malware • SIEM solutions • EDR technologies • Manual log reviews
	Recognition	Enable differentiation of normal activity from abnormal activity.	• Anti-malware (signatures and heuristics) • SIEM solutions • EDR technologies • Trained threat analyst
Loss Event Response	Containment	Enable termination of threat agent access or activities that could continue to be harmful.	• Incident response • Affected systems segregation
	Resilience	Maintain or restore normal operations.	• Backup and recovery systems and processes • System redundancy

	Loss Reduction	Reduce the amount of realized losses.	• Insurance • Fraud recovery • Legal actions • Contractual claw-backs

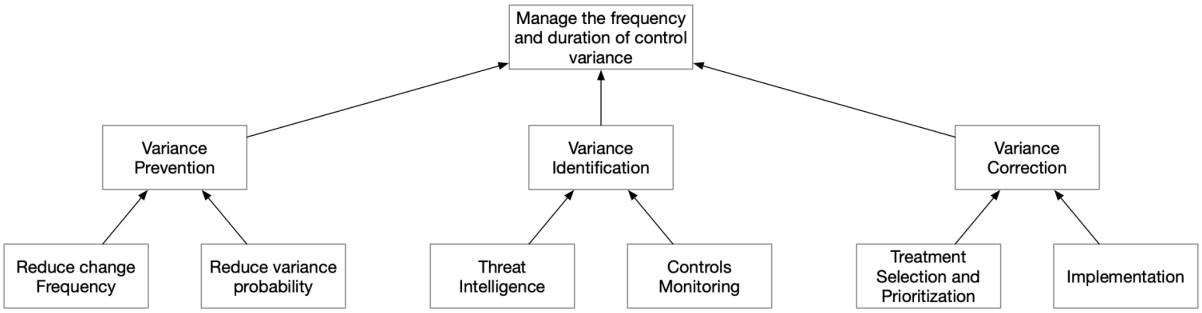
If you read the example controls closely you may have been surprised to see “software” listed as a resistive control. This is actually a very important and largely misunderstood feature of the cybersecurity control landscape, which we’ll touch on again later.

You may have noticed that some of the control examples (often these are control technologies or processes) fulfill more than one function. This is important to recognize and account for, for a couple of reasons:

- Controls that fulfill multiple functions can sometimes provide more risk reduction value
- The efficacy of each function a control fulfills can be independently measured. This not only improves our ability to account for the control’s overall effectiveness, it also provides a more complete means of comparing one control’s value versus another.

Variance Management Control Functions

Variance management controls affect the reliability of other controls.



The table below describes each of the functions in this domain, and provides a few examples of controls that fulfill each function. Here again, the examples do not represent a comprehensive list of controls for each function.

Functional Category	Function	Description	Example controls
Variance Prevention	Reduce Change Frequency	Reduce the frequency of changes that might introduce variant control conditions.	• Restrict local admin privileges • Change control processes • Pre-implementation configuration

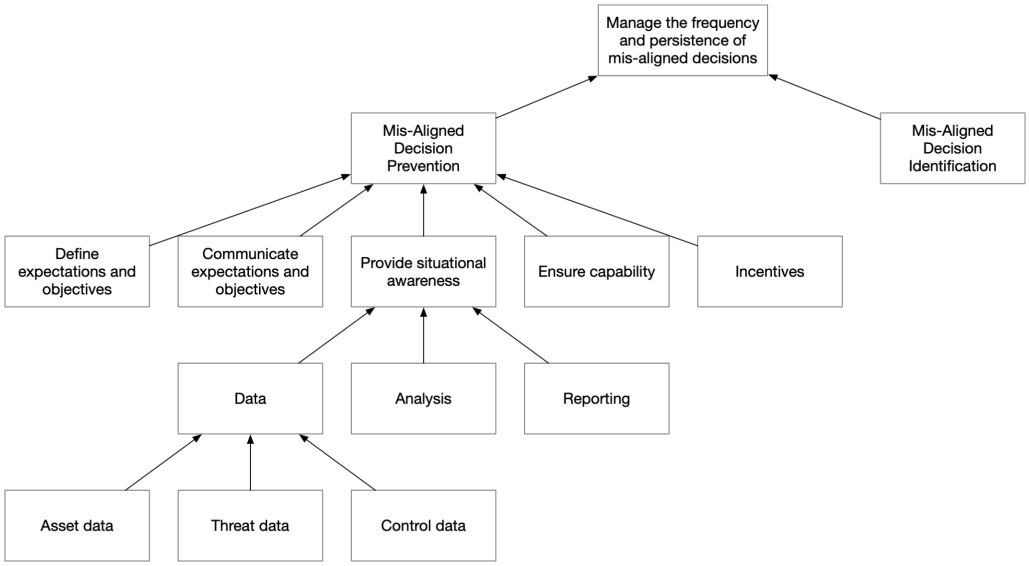
			validation
	Reduce Variance Probability	Reduce the probability of variance being introduced when changes to systems, networks, etc. occur.	• Personnel (enabled by various Decision Support controls – e.g., policies, training, etc.) • Pre-release software testing
Variance Identification	Threat Intelligence	Enable the recognition of changes in threat capabilities that result in loss event controls no longer being as effective as intended.	• Threat intelligence providers • Industry publications • Industry ISAC's
	Controls Monitoring	Enable the recognition that variant control conditions exist.	• Auditing • Scanning • Attack & penetration exercises
Variance Correction	Treatment Selection and Prioritization	Ensure that effective remediation activities are appropriately prioritized.	• Personnel (enabled by various Decision Support controls – e.g., policies, training, etc.)
	Implementation	Correct variant control conditions.	• Patching • System reconfiguration • Anti-malware updates

You'll note that a couple of these functions explicitly reference personnel as controls, which reinforces one of the features of the control landscape that some people find surprising, or at least unfamiliar. Specifically, that humans are controls. You may have noticed Personnel listed as a Resistive control against phishing attacks in the Loss Event Functional Domain, which most people accept pretty readily. In the Variance Management domain we see additional instances of personnel as controls. These are not the only ways in which personnel fulfill control functions, but these are perhaps the most obvious and common ones.

Note that the deeper you get into FAIR-CAM, the more obviously important this fact becomes. This may provide clarity and supportive evidence for those in the profession who've always emphasized the importance of personnel in risk management.

Decision Support Control Functions

Decision support controls affect the ability to make well-informed decisions.



The tables below describe each of the functions in this domain, and provide a few examples of controls that fulfill each function. And, as always, the examples do not represent a comprehensive list of controls for each function!

NOTE: Because of this model's deeper structure, the information has been broken out into three tables.

Functional Category	Function	Description	Example controls
Mis-aligned Decision Prevention	Define Expectations and Objectives	Define an organization's risk management expectations and objectives.	• Policies and standards • Processes
	Communicate Expectations and Objectives	Ensure that responsible persons are aware of and understand the organization's risk management expectations and objectives.	• Education and awareness training • Policy update emails
	Provide Situational Awareness	Provide decision-makers with an understanding of the current state and its relevance to the organization's expectations and objectives.	• See the Situational Awareness table below
	Ensure Capability	Ensure that the decision-maker has the necessary skills, authority, and resources to make decisions that are aligned with the organization's expectations and objectives.	• Skills training • Capacity management • Appropriate tooling

	Incentives	Ensure that personnel are motivated on a personal level to make decisions aligned with the organization's expectations and objectives.	• Specific MBOs • Compensation plans • Termination policies • Bonus structures
Mis-aligned Decision Identification		Identify decisions that are not aligned with organization expectations or objectives.	• Root cause analysis • Incident post-mortems • Auditing

NOTE: The correction of mis-aligned decisions occurs through existing functions, such as: clarifying definitions of expectations and objectives, improving communications, creating incentives, etc.

Functional Category	Function	Description	Example controls
Provide Situational Awareness	Data	Provide data as inputs to analysis controls.	• See the Data table below
	Analysis	Accurately synthesize asset, threat, and control data for decision-makers.	• Risk analysis tools and processes
	Reporting	Provide analysis results to decision-makers in a time and manner that meets their needs.	• Quarterly board reports • Cost-benefit analyses

Functional Category	Function	Description	Example controls
Data	Asset Data	Provide data related to the assets at risk.	• CMDB solutions • Asset inventories • Dataflow maps
	Threat Data	Provide data related to the threat landscape. Note that this is broader in scope than the Threat Intelligence function in Variance Management, which focuses solely on new threat capabilities.	• Threat intelligence • Internal threat assessments

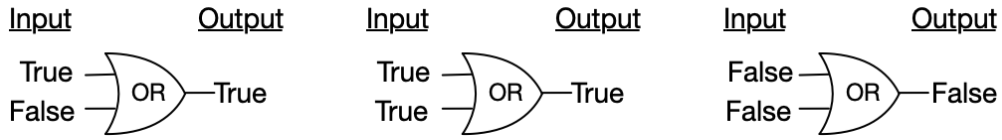
	Control Data	Provide data related to the controls environment..	• Auditing • Scanning • Regulatory exams

Relationships and dependencies

For those of you who aren't familiar with Boolean logic, there is a lot of information available online, including this article (https://computer.howstuffworks.com/boolean.htm). A brief explanation is simply that two or more variables (e.g., control functions) can be related in ways that are important when we're analyzing controls. More specifically, controls can have a Boolean AND relationship, or a Boolean OR relationship.

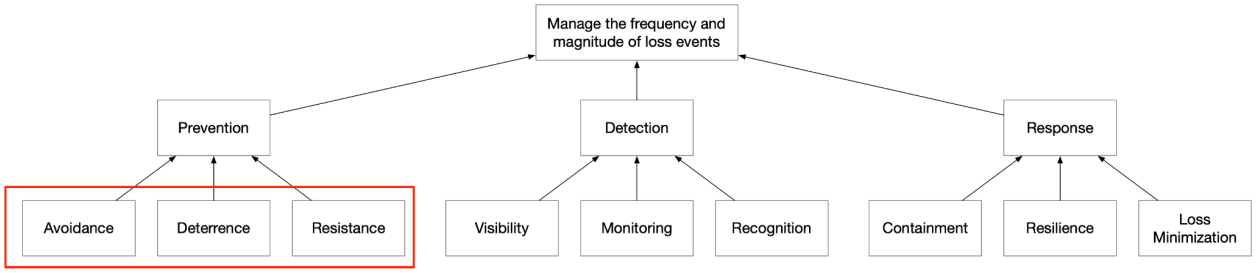
Boolean OR

In a Boolean OR relationship, the outcome is only dependent on one of the inputs. For example, let's say you have two inputs, either of which can be in a True state or a False state. If either of the inputs is in a True state (it doesn't matter which one), the output will be True.



How this plays out in controls is that some functions have a Boolean OR relationship with one another, meaning if a control for either function is operating in its intended state (i.e., it isn't variant), the function's objective is fulfilled.

In the diagram below, the three functions under Loss Event Prevention have a Boolean OR relationship with one another. In other words, if controls that serve any one of the three functions are operating as intended, a loss event is unlikely to occur.

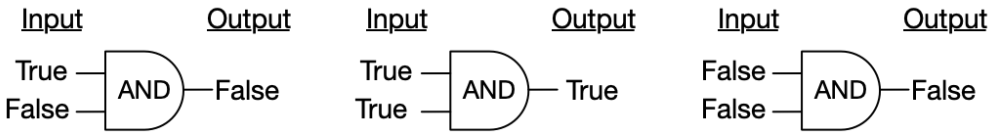


For example, if we have implemented an Avoidance control (e.g., IP address filtering) and a Resistance control (e.g., multi-factor authentication), and either of them is functioning as intended, then a loss event is unlikely to occur for the scenarios in which they're both relevant. If the IP address filtering control is working correctly, the attacker can't reach the target. If the MFA control is working effectively, then even if the attacker reaches the target, the attack is unlikely to succeed. Clearly, this example is a bit over-simplified, but it illustrates the point.

One of the implications is that these functions offer one form of defense-in-depth, which we'll discuss a bit more later in the document.

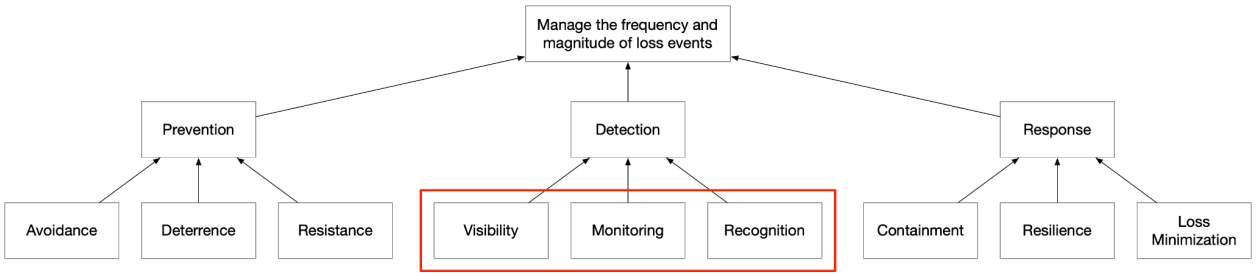
Boolean AND

In a Boolean AND relationship, the outcome is dependent on all of the inputs. For example, let's say you have two inputs, either of which can be in a True state or a False state. Both of the inputs have to be in a True state in order for the output to be True.



How this plays out in controls is that some functions have a Boolean AND relationship with one another, meaning if a control for any of the related functions is not operating in its intended state (i.e., it is variant), the function's objective will not be fulfilled.

In the diagram below, the three functions under Loss Event Detection have a Boolean AND relationship with one another. In other words, if controls that serve any one of the three functions are in a variant condition, detection is unlikely to occur.



For example, if we've implemented logging but nobody looks at the logs, or someone looks at the logs but isn't capable of distinguishing between normal versus abnormal activity, then detection doesn't occur.

These Boolean relationships exist throughout the three FAIR-CAM models, and failure to account for them in controls analysis can significantly affect the efficacy of control-related decisions..

How the model works

NOTE: This section only discusses the efficacy of controls that fulfill FAIR-CAM Loss Event control functions. Even within the LE control functions, this section will concentrate primarily on

Preventative functions. Detection and Response controls will be discussed at a high level later in the document. A comprehensive description of how to apply the entire FAIR-CAM model is beyond the scope of this document.

It's also important to note that the approach described below is different from how susceptibility (a.k.a., vulnerability) was derived in "classic" FAIR analyses. The new approach is inherently more reliable and far more easily supported with empirical evidence.

A control's risk reduction value

One of the most important applications of FAIR-CAM is to measure the risk reduction value of controls. Amongst other things, this enables:

- Cost-benefit analysis of proposed new or improved controls,
- Cost-effective prioritization of control gap remediation
- An understanding of how much risk would increase if specific existing controls are removed or degraded

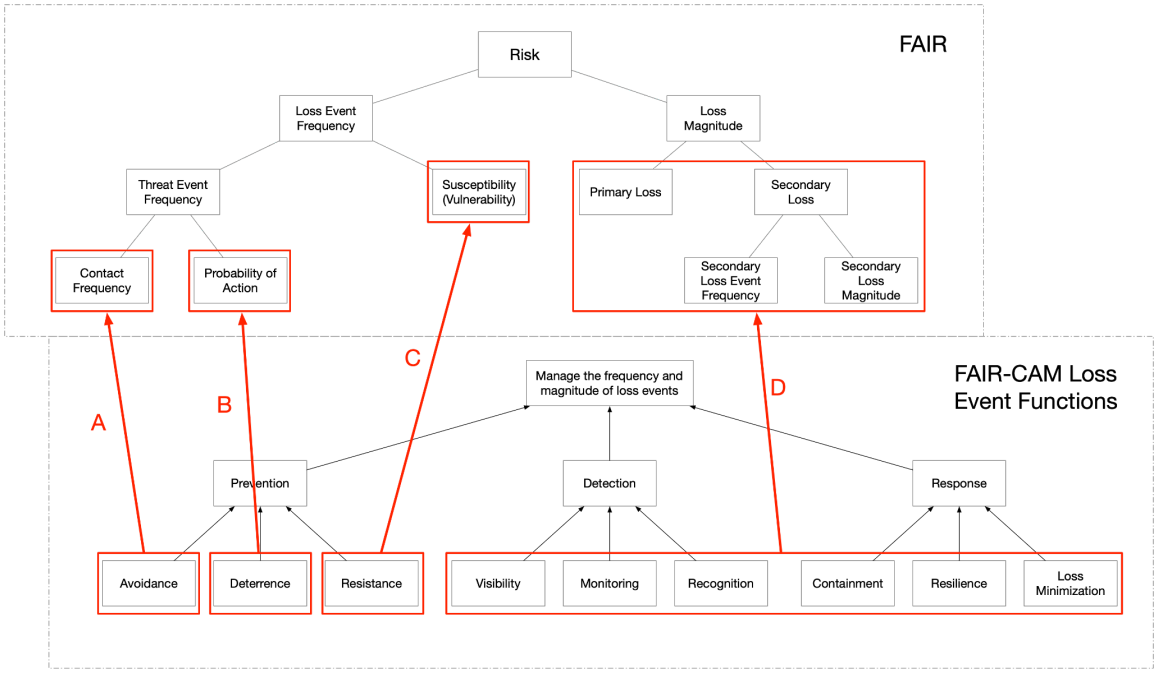
In order to establish a control's risk reduction value, we have to understand a number of things:

1. Which loss event scenarios the control is relevant to
2. How much risk currently exists from those scenarios
3. How the control does (or could) affect risk within those scenarios
4. How effective the control is expected to be when it's operating normally
5. How effective the control is expected to be when it's operating in a degraded mode
6. How frequently the control is forecasted to be in a degraded condition
7. How long the control is forecasted to persist in a degraded condition when it becomes degraded
8. Which other controls the control of interest is dependent on, and the operational efficacy of those controls
9. Whether other controls exist that are also relevant to the same scenarios, and the operational efficacy of those controls

That's a lot to consider, which is one of the reasons why FAIR-CAM is challenging to apply manually. Nonetheless, in the following sections I'll dig into bullets 3 through 7 above. After that, I'll share an example.

How controls affect risk

The diagram below illustrates how the different Loss Event Control Functions map to the FAIR model, and therefore how they affect risk.



A. Avoidance controls reduce Contact Frequency between threats and assets
B. Deterrence controls reduce the probability of illicit action if contact occurs
C. Resistive controls reduce the probability of successful illicit actions
D. Detection and Response controls reduce the magnitude of loss when an event occurs

Note that many controls fulfill more than one FAIR-CAM function. For example, EDR solutions can, depending on how they're configured, fulfill the following functions for the systems they're installed on:

- Loss Event Prevention/Resistance
- Loss Event Detection/Visibility
- Loss Event Detection/Monitoring
- Loss Event Detection/Recognition
- Loss Event Response/Containment

Units of measurement

Each FAIR-CAM function has a specific unit of measurement, which means that the efficacy of controls which fulfill a function would be measured using that unit of measurement. For example, the unit of measurement for the LE Prevention/Avoidance function is percentage – in other words, by what percentage would we expect Contact Frequency between threats and assets to be reduced by a control.

The units of measurement vary amongst FAIR-CAM functions, with some being percentages, and others being frequencies, duration, or monetary values. The fact that units of measurement vary between functions is one of the key reasons why FAIR-CAM has as many functions as it

does. You simply can't munge together the efficacy of controls whose units of measurement are different.

Note that explicit units of measurement is one of the key advantages FAIR-CAM provides over other approaches to control efficacy, which tend to be qualitative and ordinal in nature. Having real (vs. abstract) units of measurement also means that control efficacy can be empirically measured.

What do we mean by "efficacy"?

Before we get into the analytic details, we need to be on the same page regarding what "control efficacy" means. Specifically, all controls fulfill one or more of the functions within the FAIR-CAM model, and a control will fulfill those functions to a greater or lesser degree. In this context, efficacy is a measurement of how well a control performs a particular function.

Intended, Degraded, and Operational efficacy

In measuring the risk reduction value of a control it's important to recognize that control efficacy falls into three categories: Intended, Degraded, and Operational efficacy.

- **Intended Efficacy** is how well the control is expected to perform when it's operating as intended. For example, when configured appropriately, access privileges are 100% effective in preventing unauthorized actions.
- A control's **Degraded Efficacy** is how well the control is expected to perform when it's in a "variant condition" – i.e., not operating as intended. For example, when configured inappropriately, access privileges are 0% effective in preventing unauthorized actions.
- A control's **Operational Efficacy** is derived from a combination of Intended and Degraded efficacy, as well as the how often and for how long the control is in a degraded condition. How to derive Operational Efficacy will be discussed below. Operational Efficacy is what's used in determining a control's risk reduction value.

Note that Intended and Degraded Efficacy values are not always 100% and 0% respectively. More commonly they range between 0% and 100%.

Variance

Loss events only occur when controls are missing, fail, or are insufficient.

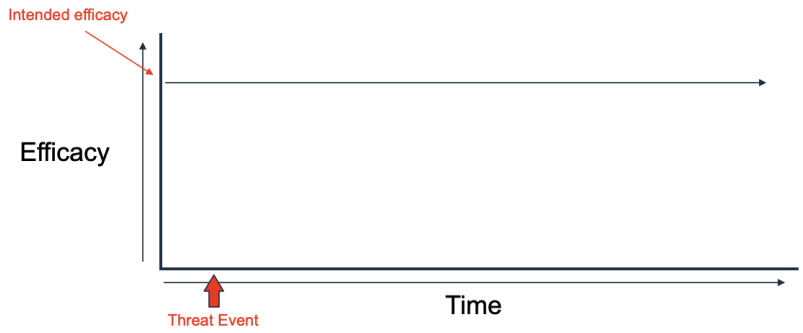
Controls that are not operating at their Intended Efficacy are referred to as being in a "variant" condition. A few examples of variant controls include:

- Software that is susceptible to code exploitation
- Access privileges that are inappropriately configured
- Out-of-date anti-malware

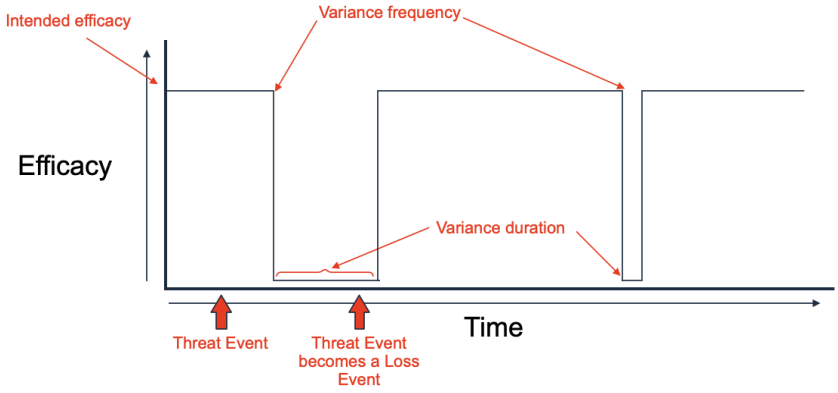
- Passwords that don't meet minimum criteria for length and complexity

How frequently controls become variant and how long they stay variant has a significant effect on their Operating Efficacy.

The diagrams that follow illustrate how this works. In the first diagram, a control (let's assume it's access privileges) is intended to not be susceptible to unauthorized data deletion. If someone tries to delete data who isn't authorized to do so, their attempt fails.



In reality however, almost all controls experience variance occasionally, which is illustrated in the diagram below.



In this case, if a bad actor tries to delete data when privileges are misconfigured (i.e., are variant), the attacker will be successful.

This example was especially simple because access privileges are binary in nature – i.e., when they're in their intended state they're 100% effective. When they're variant, they're 0% effective. Not all controls are binary in nature, which means the Intended Efficacy and Operational Efficacy of those controls should be represented as ranges or distributions.

The Operational Efficacy of binary controls boils down to the percentage of time in a year that the control is operating in its intended state (i.e., 100% effective). This is determined by its frequency and duration of variance. The formula below provides the percentage of time a control is in its intended state given VF and VD values (i.e., its "reliability"):

$$Reliability = (1 - (VF/365))^{VD}$$

You can also think of this as the probability that, on any given day, the control will be in its intended state. For binary controls, the Operating Efficacy is simply derived from:

$$OpEff = IntEff * Reliability$$

For example:

- A binary control with a VF of once per year, and a VD of 36 days has an operational efficacy of roughly 91%.
- If that control has the same VF but its VD is 7 days instead of 36 days it will have an operational efficacy of roughly 98%.

The formula for deriving the operational efficacy for non-binary controls is significantly more complex.

One of the insights from this is the temporal relationship that exists between variance and TEF. In other words, assets that face high-activity threat landscapes (e.g., are Internet-facing) can't tolerate as much variance as those that face low-activity threat landscapes. That may seem obvious, but using FAIR-CAM enables us to explicitly and empirically measure these aspects of our landscape, and make informed choices about which controls to leverage and how they need to be managed.

Where do we get VF and VD data?

In an ideal world, all Variance Frequency (VF) and Variance Duration (VD) data would come from empirical sources like audit findings, vulnerability scans, various cybersecurity platforms, etc. Unfortunately, those data may not be readily available for all controls, especially when initially applying FAIR-CAM. Until organizations can accumulate these data over a period of time it is reasonable to use calibrated subject matter expert estimates or industry benchmark data as a starting point.

What contributes to VF and VD?

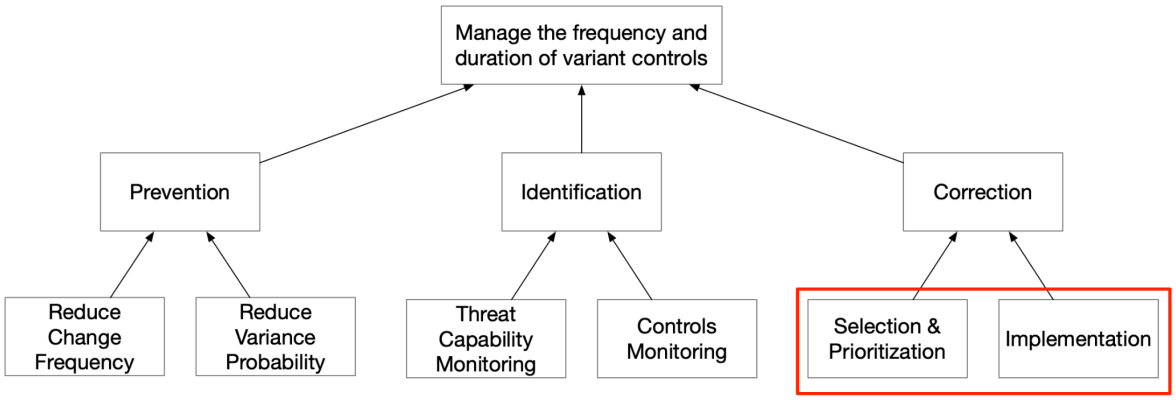
Variance Frequency and Duration are outcomes from the efficacy of Variance Management and Decision Support controls. For example, VF for access privileges within an organization is largely a function of the organization's:

- Policies related to access management
- Communication of those policies to responsible line management
- Line management's awareness of of the importance of access management
- The difficulty of managing access privileges in accordance with policy, and
- Whether management personnel are appropriately incentivized to fulfill their access management responsibilities

Each of these maps to one of the functions within the Decision Support Functions.

How this ties into Variance Management is that in this case the VM control is a human – typically a line manager who's responsible for requesting access privilege changes for personnel who leave or change roles. These personnel are making decisions about whether to fulfill their responsibilities, which Decision Support controls affect. Better DS controls result in better decision making.

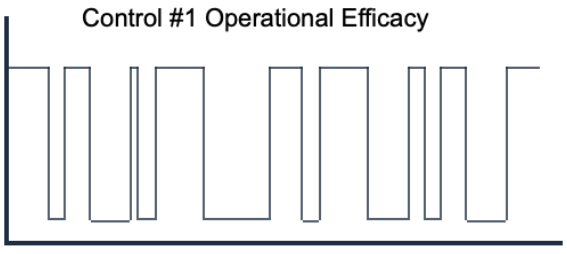
VD for access privileges is a function of how frequently privileges are reviewed (e.g., quarterly, annually, etc.), which falls under Variance Management/Controls monitoring, as well as how quickly any variant conditions are remediated, which falls under the Selection & Prioritization and Implementation sub-functions of Variance Management/Correction.

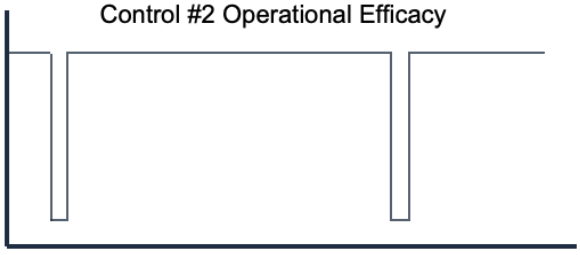


The better an organization manages VF and VD, the lower those values will be, which translates into better overall Operational Efficacy of the controls being managed.

This provides the basis for another key insight. Specifically, organizations can improve their risk postures by adding more controls, improving the reliability of existing controls, or both. In some cases, one approach will be more cost-effective than another. Using FAIR-CAM, organizations will be able to evaluate these options and make informed decisions that align with their needs and constraints.

The diagrams below illustrate (somewhat extremely) the Operational Efficacy of controls that are managed differently from a variance perspective.





Applying FAIR-CAM

Before we step through an example analysis it's important to ensure that we're on the same page regarding measurement quality. Those who have been trained in performing FAIR analyses can skip ahead to the example analysis.

Accuracy vs. Precision

There is a significant amount of confusion regarding this topic, which often creates challenges in risk-related conversations and decision-making. There is, however, a simple way to keep them straight:

- Think of "accuracy" as truthfulness
- Think of "precision" as exactness

For example, a statement that my annual income is $745,322.78 is highly precise, but inaccurate (unfortunately). On the other hand, a statement that my annual income is between $80k and $300k is not precise, but it is accurate. It contains the truth.

The most important thing to keep in mind is that measurements with lower precision are acceptable, but inaccurate measurements are not. There is a caveat though. If precision becomes too low, measurements aren't useful in decision-making. For example, if we simply say that the probability of some loss event is between 0% and 100%, well, that's guaranteed to be accurate, but it's too imprecise to be useful.

Several other things to keep in mind include:

- By faithfully reflecting measurement uncertainty in our inputs, we increase the odds of our measurements being accurate. I.e. – when we have poor data, the use of wider ranges to reflect that fact will increase the odds of the range containing the actual value.

- Decision-makers can (and sometimes should) make different decisions when the information they're working from has more uncertainty. As a result, we have an obligation to ensure they have the best opportunity to make a well-informed decision, and uncertainty is an important piece of information about a measurement.

- Sometimes the sources of measurement uncertainty are correctable, at least to some extent (e.g., by increasing log retention from 90 days to 365 days). If measurement uncertainty and the sources of the uncertainty aren't part of the conversation, then there's a much lower chance that those opportunities for improvement will be recognized and leveraged.

- And contrary to what some people believe, faithfully representing measurement uncertainty will in most cases increase confidence in our measurements – at least for people who understand measurement. Any self-respecting business executive will scoff at impossibly precise risk measurements, and likely will (and should) discount such a measurement's value.

The bottom line here is that measurement accuracy (truthfulness) is our primary objective. High measurement precision is a “nice to have” but is never a priority.

Example analysis

As was noted earlier, it isn’t feasible to perform FAIR-CAM analysis outside of a FAIR-CAM enabled software application, so this example will be relatively simple and high-level in nature. For simplicity I’ll also be using discrete input and output values rather than ranges or distributions. I’ll also assume the controls are binary in nature.

In this example, we’ll examine how much less risk an organization would have if it implements a web application firewall (WAF) in front of a high-value Internet-facing web application. To perform this analysis we’ll follow the steps outlined in the earlier section “A control’s risk reduction value”.

1. Which loss event scenarios is the control relevant to?
    - In this case, we’re concerned about compromise of sensitive customer information by cyber criminals using a code exploitation attack against the web application.
2. How much risk currently exists from those scenarios?
    - For the sake of illustration, we’ll assume that the loss magnitude from this event would be $10M. This is true with or without the WAF because the WAF doesn’t affect loss magnitude.
    - We’ll also assume that the probability of experiencing a loss event in the next 12 months without the WAF is 10%, based on the frequency of attacks and the Operating Efficacy of existing preventative controls.
        i. Note that the primary existing preventative control in this scenario is the web application software itself. When it has been coded securely, it is not susceptible to code exploitation – i.e., it’s a strong resistive control.
        ii. This condition can change when: a) a new version of the application is released that has not been coded securely and is susceptible to code exploitation, or b) the threat community develops a new zero day exploit.
    - Given the probability and impact of this loss event scenario, the current annualized amount of risk is $1M.
3. How the control does (or could) affect risk within those scenarios?
    - A WAF is resistive in nature, as it blocks specific types of attacks. Consequently, it and the software are “stacked resistive controls”, which is one form of functional defense-in-depth.
4. How effective is the control expected to be when it’s operating normally?
    - WAFs are generally considered to be highly effective against run-of-the-mill attacks, but are less effective against highly sophisticated attackers. This means that WAFs are non-binary in nature. For this example, however, I’m going to assume the control’s Intended Efficacy is 100%.
5. How effective is the control expected to be when it’s in a variant condition?

    - WAFs are considered variant when they are either misconfigured or not updated. Consequently, the Degraded Efficacy of a WAF would be less than its Intended Efficacy. Because we're treating the WAF as if it was binary, its Degraded Efficacy is 0%.
6. How frequently is the control forecasted to be in a variant condition?
    - Variance frequency is a function of how well an organization manages a control. For the sake of illustration, we'll say that this organization doesn't manage technical solutions like WAFs very well, with variance occurring twice per year (VF = 2).
7. How long the control is forecasted to persist in a degraded condition when it becomes variant?
    - Variance duration is a function of how quickly a variant condition is identified and resolved. Our hypothetical organization in this example only checks the WAF and similar technologies every 90 days.
    - Because this is a high-value web application, we're going to assume that any identified variance in the WAF would be immediately addressed. In reality, organizations often do not or cannot immediately address variant conditions, which adds time to VD.
    - Given the above, for our example the average length of time the WAF remains variant is 45 days.
8. Which other controls is the control being analyzed dependent on, and what is the operational efficacy of those controls?
    - As stated earlier, a WAF's operational efficacy is a function of how it's managed. For a WAF, intrinsic variance frequency (i.e., variance caused by the organization itself) can be reduced through effective administrator training, as well as change management controls.
    - For our WAF, part of the variance duration is managed via the weekly testing, which sets a maximum of seven days the WAF could be variant before it's identified as needing attention.
    - Note that the more reliable a Variance Management control (like weekly testing) is the more reliable any downstream controls like the WAF will be.
9. Do other controls exist that are also relevant to the same scenarios, and what are the operational efficacy of those controls?
    - As was mentioned earlier, the web application software itself is also relevant as a control for this scenario. In step 2 above, we accounted for the efficacy of this control in arriving at the current loss event probability values.

Given all of the above, adding a WAF is forecasted to reduce the probability of compromise in this scenario from 10% down to approximately 2%. This 8% reduction in probability, times the loss magnitude range, results in a risk reduction value of $800k[2]. This value can be compared against the cost of buying and maintaining the WAF to determine whether it's a good investment.

[2] Note that these numbers are for illustration purposes only, and do not reflect the risk reduction value any particular organization would realize from implementing a WAF.

Clearly, this example is simplified, but hopefully it provides some clarity regarding how this type of analysis works. It also should reinforce the point that FAIR-CAM analyses require FAIR-CAM enabled software, which can account for multiple scenarios and many controls simultaneously. Such software also can continually monitor the condition of controls and use that telemetry to update variance frequency and duration values and provide near-real time risk updates.

Just scratching the surface

FAIR-CAM provides the foundation for gaining many new insights in age-old cybersecurity principles. Unfortunately, covering all of these opportunities in-depth would require a book. In the sections below however, I provide a very high level overview on a few of these topics.

Working with control frameworks

As more organizations have become aware of FAIR-CAM, there has been increasing interest in how it can be combined with common control frameworks like NIST-CSF, the CIS Controls, etc. In this section I'll discuss a few of the opportunities and challenges associated with this.

As was briefly described at the beginning of this document, existing control frameworks describe controls anatomy – i.e., the controls that could or should be part of a cybersecurity program. FAIR-CAM provides a description of how those controls function, both individually and collectively, to affect risk. Consequently, when we map elements within a control framework like NIST-CSF[3] to FAIR-CAM, we gain a much more complete and accurate understanding of how those controls affect risk. Ideally, we then should be able to measure the risk reduction value of those controls. Unfortunately, there are several significant challenges with this:

- Control descriptions within many of these frameworks are somewhat (or sometimes very) ambiguous, which makes it difficult to reliably map them to FAIR-CAM functions. (e.g., NIST CSF PR.AC-3 "Remote access is managed.")
- Many controls are broadly defined, resulting in the need to map them to multiple FAIR-CAM functions. For example, NIST CSF PR.AC-1 "Identities and credentials are issued, managed, verified, revoked, and audited for authorized devices, users, and processes" maps to multiple FAIR-CAM functions. When a control is mapped to multiple functions, the question has to be asked whether the rating that control has been given (e.g., "2") apply equally to all of the functions?
- Control scoring scales are inevitably ordinal in nature (red, yellow, green, 1-through-5 scales, etc.), which makes it necessary to translate them into quantitative

[3] Note that NIST CSF does include "functions", however those functions are far too high level to be useful in risk measurement. Furthermore, the premise underlying how CSF controls (i.e., control outcomes) have been mapped to those functions wasn't focused on risk measurement, but rather program evaluation. This means that control mappings within CSF often don't align with how those same controls would be mapped to FAIR-CAM.

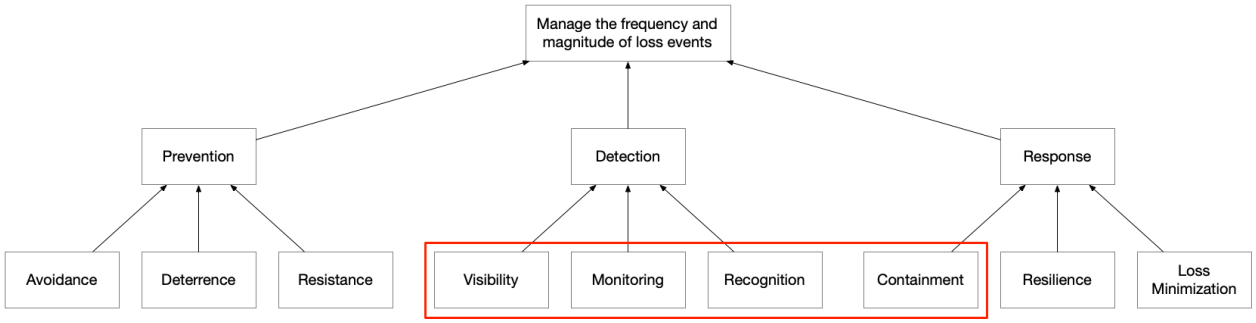
measurements. Translating these ordinal values accurately is inherently non-trivial, and is made even more difficult because those scales often aren't well defined (or in some cases they aren't formally defined at all).

Some frameworks, however, are more clearly defined and simpler to map than others. Note that the mapping and translation of the most commonly used frameworks is being done at the time this paper is being written (late 2023). These mappings will be published as they are completed.

Detection and response controls

In many cybersecurity loss events, detection and response controls can affect loss magnitude by reducing the amount of dwell time a threat actor has to complete their illicit actions.

Within FAIR-CAM, the Visibility, Monitoring and Recognition sub-functions within the Loss Event Detection function, have a Boolean AND relationship to one another. This means that if any of the three are missing, defective, or insufficient, detection doesn't occur. Furthermore, the Containment sub-function in the Response function also has to be operating sufficiently to terminate a threat actor's access. This means that it, too, has a Boolean AND relationship with the Detection sub-functions.



From a measurement perspective:

- Visibility's unit of measurement is % – i.e., the probability that Visibility controls (e.g., logging) will have captured the information needed to detect an event.
- Monitoring's unit of measurement is time – i.e., how much time elapses between monitoring events (e.g., someone looks at the logs).
- The unit of measurement for Recognition controls (e.g., checksums, user activity baselines, etc.) is % – i.e., the probability that illicit activity can be differentiated from legitimate activity.
- The unit of measurement for Containment controls (e.g., incident response) is time – i.e., how long will it take to terminate a threat actor's access and activities.

In order for the efficacy of these controls to be evaluated, an “Event Velocity” variable has to be defined. Event Velocity is the time it takes from initial compromise until maximum loss is assured.

Taken together, it is possible to determine the probability that an event is detected and contained before maximum loss occurs.

Defense-in-depth

Using FAIR-CAM as a lens to examine DiD with, we find that there are two broad categories of DiD:

- Architectural – This category of DiD is what many people think of when the topic comes up – i.e., segmenting and layering the network environment. Each layer of the architecture effectively acts as an Avoidance control for deeper layers. In other words, we avoid contact between a threat actor and the assets at risk when we stop or contain the actor in an outer layer. This reduction in contact frequency with inner layers can be explicitly measured using FAIR-CAM.
- Functional – This category of DiD isn’t entirely unique to FAIR-CAM, but FAIR-CAM allows us to understand, discuss, and apply it more explicitly. An example of this form of DiD was mentioned briefly in the analysis example given earlier. Functional DiD primarily provides resilience to control failure. Note that there are several subcategories of Functional DiD, but a discussion of those is beyond the scope of this paper.

The primary benefit here is that FAIR-CAM enables us to measure the risk reduction value of DiD.

Population susceptibility

It’s common to see organizations determine that their susceptibility to some type of attack (e.g., an authentication-based attack) is 50% if half of their systems have a particular control in place (e.g., MFA). But this isn’t how it works in reality.

When our risk scenario is one in which an attacker is expected to probe multiple systems in the course of an attack, then we have to determine the probability of one or more of the systems that make up the “attack surface” being susceptible when the attack takes place.

Earlier in this document I described how susceptibility is strongly affected by variant control conditions. This is a relatively straightforward analytic problem when we’re dealing with a single system (e.g., a single user end-point). It becomes a bit more challenging however, when we want to measure the susceptibility of a population of systems – i.e., when the attack surface area is larger than one potential point of attack. At least two things make this more challenging:

- Whether all of the systems in the population have the same controls in place, which can be especially challenging when shadow IT is involved
- Whether all of the systems in the population are managed in the same way, which affects the frequency and duration of variance (also a problem for shadow IT)

Covering this topic thoroughly would require a white paper of its own. The bottom line is that population susceptibility is often approached incorrectly, which can severely affect the accuracy of risk measurement. FAIR-CAM provides a very effective way to deal with this measurement challenge.

Control sets

What is the risk reduction value of data backups? We intuitively understand that backups play a role in FAIR-CAM's Loss Event Response/Resilience function, which affects loss magnitude, but how do we measure data backup's effect on risk?

Data backups are an example of a control that, by itself, doesn't reduce risk. In addition to the backed-up data you also need to have recovery procedures in place, recovery systems to put the data on, as well as key pieces of infrastructure. In other words, in order for successful recovery from a data integrity or availability event, all of these pieces of the control puzzle have to be in place and functioning properly. This is another example of the Boolean AND relationships discussed earlier. In this case however, these controls don't fulfill separate functions in the FAIR-CAM model (like the visibility, monitoring and recognition functions do in LE Detection), rather they work together to fulfill a single function. In FAIR-CAM, controls with a Boolean AND dependency to one another that fulfill a single function are referred to as "control sets."

There are other common controls that form control sets, perhaps most notably some of the password-related controls (e.g., password length, password complexity, login failure lockouts, etc.). There isn't (yet) a comprehensive library of these sets and their member controls, but it's useful to understand that control sets exist, particularly when it isn't obvious how a control's effect on risk can be measured.

Measuring the value of Variance Management Controls

Measuring the risk reduction value of VM controls is a significantly different process than what we discussed for measuring the risk reduction value of LE controls, primarily because VM controls affect risk indirectly. Here again, an in-depth description of how this works is beyond the scope of this paper. That said, the process typically involves:

- Identifying the downstream LE controls that a VM control affects. A relatively simple example is vulnerability scanning, which was described earlier as fulfilling the VM Identification/Control Monitoring function. The most obvious downstream LE control that vulnerability scanning affects is software.

- Identifying loss event scenarios that the downstream LE control is relevant for. In our example, software is relevant as a resistive control for any loss event scenario where code exploitation is a potential method of attack.

- Determine the current amount of risk for the relevant scenario(s). This provides a baseline for later comparison.

- Determining how the downstream control is affected by the VM control. Because vulnerability scanning fulfills the VM Identification/Control Monitoring function, it lets us know when a vulnerable condition exists (or doesn't exist).

- Estimating or measuring the effect of a change to the VM control on the LE control's efficacy. For example, if we eliminate vulnerability scanning altogether, how does the efficacy of software as a resistive control change? Or perhaps the question is whether we should increase the frequency of vulnerability scanning, which reduces the amount of time it takes to identify vulnerable conditions, and increases the operational effectiveness of the downstream LE control.

As we saw with LE controls, we also have to be aware of, and account for the fact that VM controls have Boolean relationships with other controls. This can sometimes (but not always) complicate the analysis.

Fortunately, it's often fairly easy to recognize deficiencies in VM controls by their downstream effects on other controls. For example, if we have problems with operating system software that repeatedly goes variant because of users changing settings, we can evaluate whether there are VM Prevention controls we can improve or implement (e.g., limiting user admin privileges). If we make that change to user privileges we can over time empirically measure the reduction in software variance, which translates to a measurable improvement in software's operational efficacy as a resistive control.

Measuring the value of Decision Support Controls

Measuring the risk reduction value of DS controls is often even more challenging than it is for VM controls. This is true for several reasons, including:

- Decision support controls tend to be more global in their effects, which means they often affect many downstream controls, and thus many loss event scenarios.

- The DS control functions are riddled with Boolean dependencies.

- Empirical data can be very hard to find regarding the condition of DS control conditions. This isn't because the data are inherently difficult to find or generate, but rather because our profession hasn't prioritized the measurement of these controls.

The good news is that we can often take a similar approach to what's described above for VM controls – i.e., we can make changes to a DS control and (over time) measure its effect on downstream controls. For example, implementing meaningful incentives might reduce the

frequency of managers failing to update access privileges when personnel change roles or leave the organization.

Measuring program maturity

Historically, the risk management profession has defined and measured program maturity using frameworks like NIST CSF, ISO, etc. With FAIR-CAM, an alternative approach is available that can be empirically validated (as opposed to using abstract ordinal maturity scores), and which can be integrated into day-to-day, as well as strategic, decision-making..

Leveraging this approach first requires acceptance of a specific definition for what constitutes a "mature" risk management program. Specifically:

> *A mature risk management program is one that enables the organization to cost-effectively achieve and maintain an acceptable level of risk.*

There's a lot to unpack in this, which goes well beyond what can be covered here. Suffice it to say that FAIR-CAM enables an approach to defining and measuring program maturity that is quite different from typical methods. Some of the outcomes of this different approach include:

- Being able to explicitly and quantitatively align the organization's risk objectives with its other imperatives (e.g., revenue growth, cost management, etc.)
- Clearly understanding where the organization stands relative to its risk objectives
- Being able to define KRIs and KPIs that are explicitly tied to risk objectives
- Being able to drive day-to-day decision-making that is aligned with the organization's risk objectives

Documenting more information about this approach is on the to-do list, so stay tuned to the FAIR Institute.

Third-party risk

Third-party risk measurement and management (TPRM) has long been one of the most challenging aspects of cybersecurity. Common TPRM practices today are almost universally considered to be unreliable at best, and a significant waste of resources at worst. FAIR-CAM enables a couple of novel approaches to TPRM, both of which are substantial topics in their own right. Briefly, however:

- You can treat third-parties as an extension of your network perimeter, and can use FAIR-CAM to measure the efficacy of the controls in those systems. The challenge here is the lack of complete, accurate and timely information about the condition of those controls over time. This can be improved through questionnaires that seek information explicitly relevant to FAIR-CAM, and through better sources of third-party telemetry.

- You also can use FAIR-CAM to evaluate the maturity of a third-party's risk management practices. This has significant implications in terms of simplicity and reliability, but it

requires adopting a very specific definition of “maturity” that is not common in the industry. As mentioned briefly in the “Maturity” section above, the specifics of how this is applied are outside the scope of this document.

Wrapping up

Hopefully, this document has provided you with a basic understanding of what FAIR-CAM is, why it was developed and the value it can provide. Needless to say, there is a lot that wasn’t covered in this overview, which needs to be covered in subsequent documentation. As the community expands its use of FAIR-CAM, there will undoubtedly be many lessons learned, which will enable us to continually refine the model and apply it more broadly and more effectively.