SYM-AM-19-055

# PROCEEDINGS
## OF THE
# SIXTEENTH ANNUAL ACQUISITION RESEARCH SYMPOSIUM

## WEDNESDAY SESSIONS
## VOLUME I

**Acquisition Research:**
**Creating Synergy for Informed Change**

**May 8–9, 2019**

**Published: April 30, 2019**



Prepared for the Naval Postgraduate School, Monterey, CA 93943.

ACQUISITION RESEARCH PROGRAM
GRADUATE SCHOOL OF BUSINESS & PUBLIC POLICY
NAVAL POSTGRADUATE SCHOOL

# Uncovering Cascading Vulnerabilities in Model-Centric Acquisition Programs and Enterprises

**Donna H. Rhodes—**is a principal research scientist at the Massachusetts Institute of Technology, and director of the Systems Engineering Advancement Research Initiative (SEAri). She conducts research on human-model interaction, model curation, model-centric decision making, and innovative approaches for enterprise transformation under the digital paradigm. Previously, she held senior management positions at IBM, Lockheed Martin, and Lucent. Rhodes is a Past President and Fellow of the International Council on Systems Engineering (INCOSE) and an INCOSE Founders Award recipient. She received her PhD in Systems Science from T. J. Watson School of Engineering at Binghamton University. [rhodes@mit.edu]

**Jack Reid—**is a graduate student with the Space Enabled Research Group at the Massachusetts Institute of Technology. Reid is currently a doctoral student at MIT with research interests concerning the design and management of complex sociotechnical systems, particularly with regard to the anticipation of emergent and cascading behavior. While a master's student, he was a research assistant in the Systems Engineering Advancement Research Initiative (SEAri), performing research on vulnerability assessment methods, model-centric enterprises, and complexity and emergence. He received an MS in both Aeronautics & Astronautics and Technology & Policy at MIT. [jackreid@mit.edu]

## Abstract

Digital engineering changes how systems are acquired and developed through the use of model-centric practices and toolsets. Enterprises face new challenges in this transformation, including potential for emergent vulnerabilities within digital engineering environments. While vulnerability analysis of products and systems is standard practice, examining vulnerabilities within the enterprise itself is less common. This research is responsive to the imperatives of the newly released DoD Digital Engineering Strategy that calls for enterprises to mitigate cyber risks and secure digital engineering environments against attacks from internal and external threats, mitigate known vulnerabilities that present high risk to DoD networks and data, and to mitigate risk posed by collaboration and access to vast amount of information in models. This paper presents progress on the ongoing research that focuses on uncovering cascading vulnerabilities as related to digital engineering practice and supporting environments, with special focus on cybersecurity-related vulnerabilities. The approach uses Cause-Effect Mapping (CEM) as a mechanism for better enabling program leaders to anticipate and respond to vulnerabilities within the enterprise. The current investigation is examining enterprise-level vulnerabilities and investigating potential interventions.

## Introduction

Vulnerability assessment of products and systems has been actively investigated in recent years, resulting in a family of useful techniques now commonly accepted as good practice (LeSaint, Reed, & Popick, 2015). The assessment of vulnerabilities within the enterprises performing engineering has received relatively little attention. While many of the existing techniques for systems vulnerability assessment will still be useful, some adaptation and additional techniques are necessary. The urgency of investigating this has increased as a result of digital engineering transformation as it changes how systems are acquired and developed through the use of model-centric engineering practices and new types of environments within the enterprise.

Ongoing research has investigated the use of Cause-Effect Mapping as a mechanism for better enabling program leaders to anticipate and respond to vulnerabilities

as related to model-centric enterprises and their enabling environments (Mekdeci et al., 2012; Rovito & Rhodes, 2016; Reid & Rhodes, 2018b). A Reference Cause-Effect Map (CEM) for model-centric enterprises resulting from the work to date shows promise for considering the cascading vulnerabilities and potential intervention options. In the continuing investigation, the Reference CEM and other analytic techniques are being further developed and evaluated. Intervention approaches are being identified and mapped to cascading vulnerability chains, providing options for mitigation.

## Background

Background is provided in the following subsections to characterize digital engineering and model-centric enterprises. Prior research papers (Reid & Rhodes, 2018a, 2018b) provide additional background information.

### *Digital Engineering (Model-Centric Engineering)*

Digital engineering (sometimes referred to as model-centric engineering) involves using integrated models across disciplines, subsystems, lifecycle stages, and analyst groups. It uses models as "authoritative source of truth," to reduce document handoff and allow for more continuous evaluation. By collaborating through models, there is reduced communication time and rework in response to requirement changes. Most discussions to date focus on engineering practices and methods to overcome implementation difficulties. In any system, however, non-technical factors (human factors, business, and organizational) influence engineering effectiveness and model-centric decisions (Reid & Rhodes, 2017; German & Rhodes, 2017).

Current program leaders have significant experience with processes for acquiring and developing systems, and use this experience to identify and mitigate vulnerabilities. Limited experience exists with digital engineering practice and model-centric supporting environments, however. This situation, coupled with the increased model integration and model longevity, means that emergent uncertainties (policy change, budget cuts, disruptive technologies, threats, changing demographics, etc.) and related programmatic decisions (e.g., staff cuts, reduced training hours) may lead to cascading vulnerabilities within digital engineering enterprises, potentially jeopardizing program success. New practices and enablers are needed to assist program leaders in identifying vulnerabilities within the digital engineering environment, and to determine where interventions can most effectively be taken.

### *Model-Centric Environments*

Model-centric environments have many elements, including computing infrastructure, networks, software tools, models, data sets, data storage, and human actors. These environments may come under attack from internal and/or external threats. Some of these elements exist in traditional engineering, but some are new or changed under digital engineering practice (Reid & Rhodes, 2016). New modes of collaboration through models and data are emerging. The quantity of and types of models, digital artifacts, and data has greatly increased. Collaboration between the many enterprises involved through digital engineering (government agencies, contractors, suppliers, etc.) results in significant increases in data flowing across networks. As new toolsets are introduced into enterprise, there are potential risks related to how proficient the workforce is in using these tools and whether there are sufficient controls in place in the management of the digital artifacts produced, as well as the overall supporting infrastructure. The DoD Digital Engineering Strategy (2018) calls for the mitigation of these risks and vulnerabilities (Figure 1).

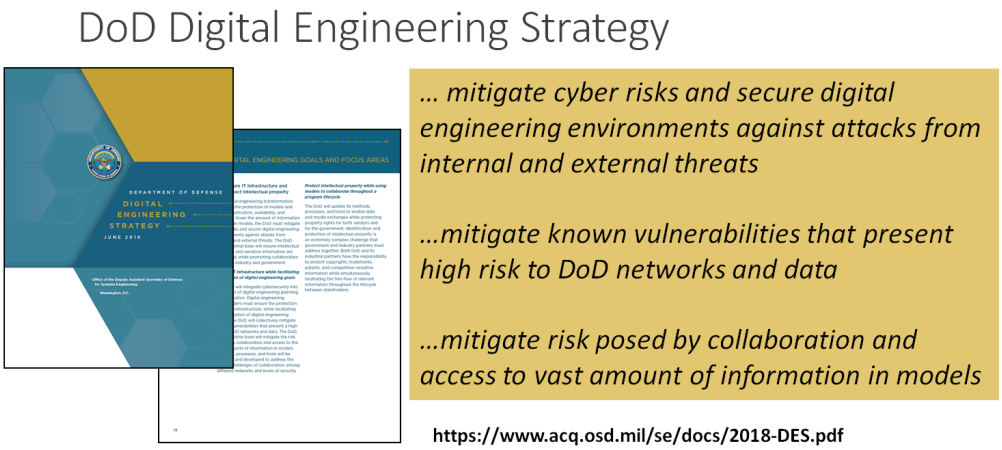


**Figure 1. DoD Digital Engineering Strategy Calls for Mitigation of Risks and Vulnerabilities**

(DoD, 2018)

## Vulnerabilities as Causal Chains

Vulnerabilities are effectively expressed as the causal series of events connecting a hazard to the system and/or failure that results. Cause-Effect Mapping is a vulnerability assessment approach that consists of a mapping of causal chains that connect an exogenous hazard to a system degradation or failure, termed a terminal event (Mekdeci et al., 2012). Terminal events are broadly defined and include any form of value loss. A casual chain can be defined as a series of events, with each event causing or being an integral part of the cause, or the next link in the chain. A hazard (spontaneous event) is a system or environmental state that has the potential to disrupt the system. A vulnerability is defined as causal means by which one or more hazards results in the system disruption/value loss. Accordingly, a vulnerability chain is defined as a conceptualization and representation of vulnerability as a causal chain, emphasizing that vulnerabilities are not discrete events.

### *Vignette*

Figure 2 shows a very simple example of a vulnerability chain, where an external trigger disrupts effectiveness of engineering activities, as triggered by increased cost of the commercial software used by the enterprise. This is illustrative of how a rather simple external change may cascade into interim impacts, and ultimately lead to a failure later in the program.

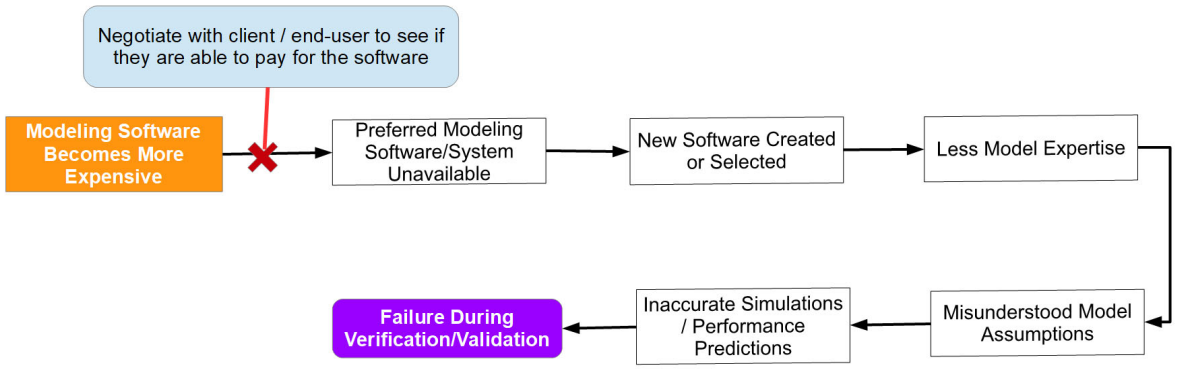


**Figure 2. Example Vulnerability Chain With Intervention Point (in Blue)**

Describing this as a vignette, the vulnerability is as follows:

> A particular piece of simulation software that your company has used on similar projects in the past is licensed from commercial software vendor. The license contract is up for renewal soon and the price goes up significantly. This could result in the preferred modeling software being unavailable for use in this program leading to the selection of an alternate software tool that the team has less (or no) experience with. Due to this lack of experience with the new software, assumptions underlying the model may be misunderstood by analysts and thus inaccurate simulation results are generated. This may not be noticed until either verification or validation when the system or subsystem does not behave according to the predicted performance levels.

One identified intervention point is shown in the blue box in Figure 2. Executing this intervention would require that program leadership recognizes when the external trigger is imminent or occurring and act quickly to avoid loss of modeling capability. Alternately, there may be other points of intervention along the chain. While this analysis is quite simple, more sophisticated applications of graph theory and probabilistic modeling can be conducted using a well-developed Reference CEM. For instance, if probabilities, likelihoods, or time scales of each event transition are known, techniques such as Markov Chain Modeling, Monte Carlo Analysis, and Bayesian Networks can be brought to bear, weighting each arc of the graph instead of treating them equally (Reid, 2018).

## Cause-Effect Mapping

Cause-Effect Mapping (CEM) has been demonstrated as a useful approach to vulnerability analysis for systems, programs and enterprises (Mekdeci et al., 2012; Rovito & Rhodes, 2016; Reid & Rhodes, 2018a, 2018b). An example CEM for a supply chain case vulnerability assessment (Rovito & Rhodes, 2016) is shown in Figure 3. The hazards are external to the perspective of the defined user, and are thus sometimes called external triggers. An intermediary event is any unintended state change of a system's form or operations which could jeopardize value delivery of the program and/or enterprise. Interventions are actions that eliminate or mitigate a vulnerability to break the causal chain.

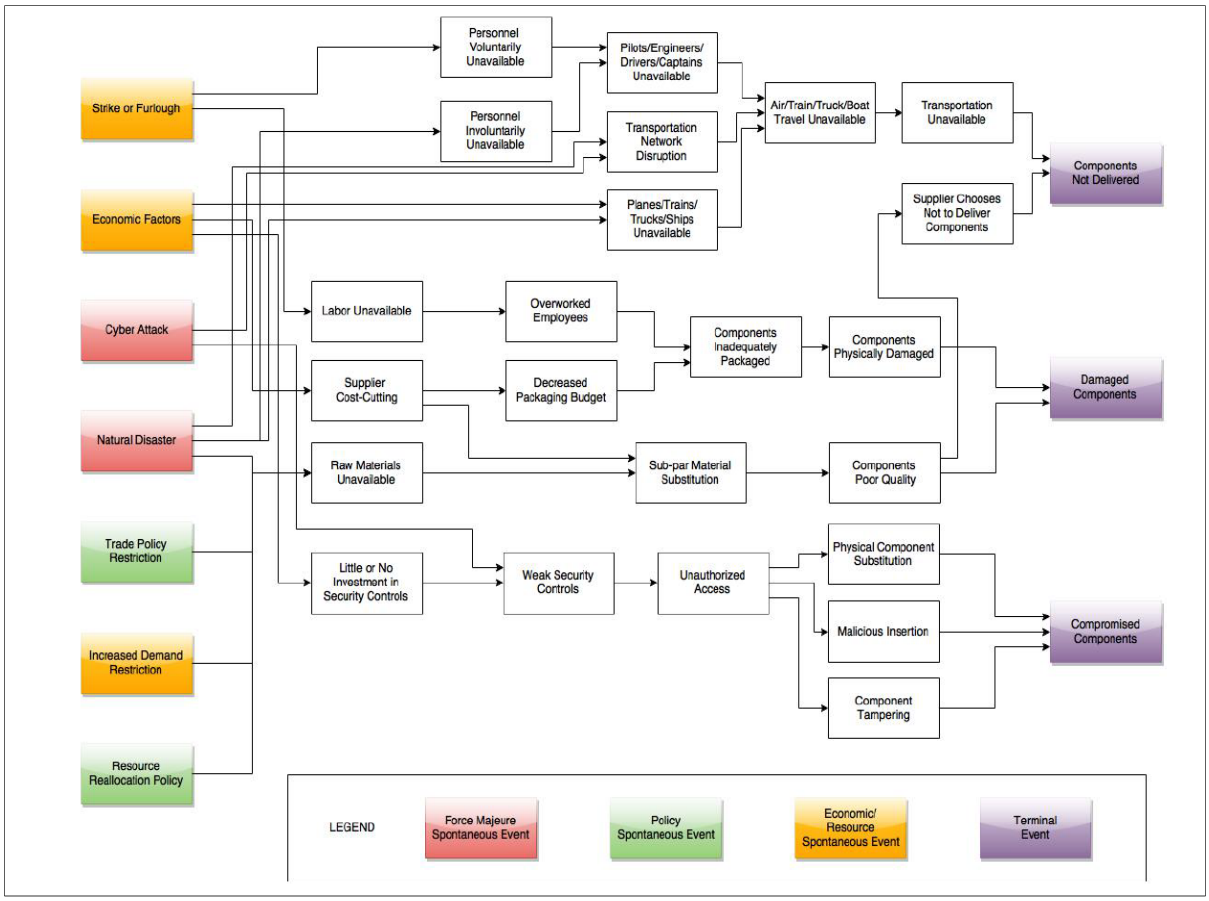


**Figure 3. Example CEM of a Supply Chain**
(Rovito & Rhodes, 2016)

A CEM is created for a specific class of decision-maker (e.g., program manager). The hazards (referred to as "spontaneous events") are exogenous from the point of view of the decision-maker for which the CEM was constructed. In this way, the cause-effect mapping approach avoids "blaming someone else" by making all hazards exogenous. The decision-maker has control over only the intermediary events. While not necessarily at fault for any of the vulnerabilities, the decision maker has the responsibility and authority to choose if, and how, to address these.

As shown in Figure 4, a causal chain may have multiple points for breaking the chain, for instance to correct weak security controls and/or to prevent unauthorized access. The first might be a policy/process intervention and the latter might be a technology intervention. The decision to execute one/both of the interventions will depend upon unique factors, such as the cost to implement, color of money available, specifics of the situation, and so forth.

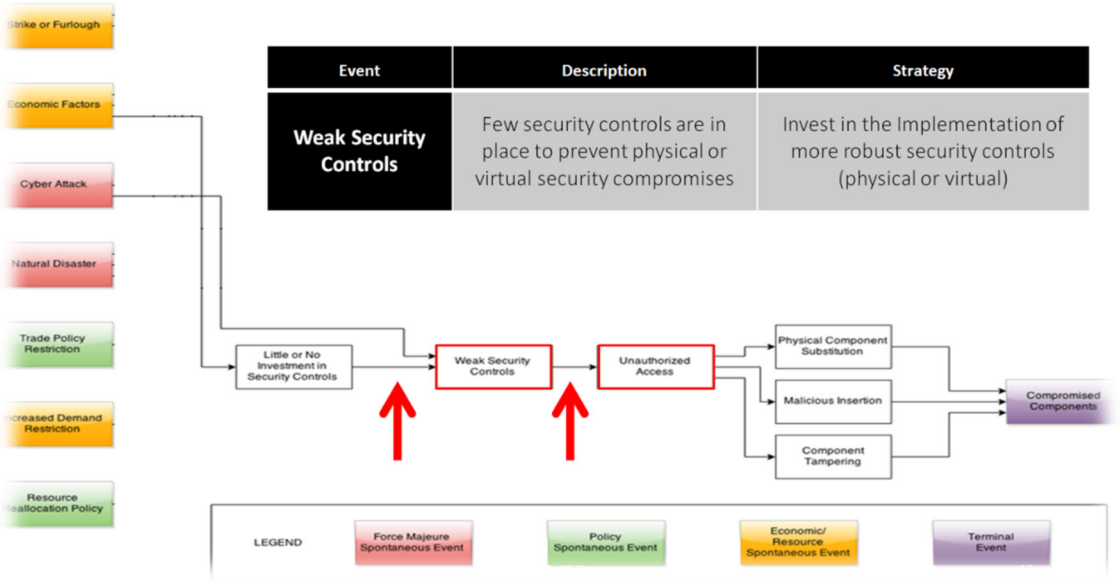


**Figure 4. Example of Two Alternative Placements for an Intervention in Causal Chain**

The basic steps to create a new CEM are not application specific and are detailed in Rovito and Rhodes (2016) and Reid and Rhodes (2018b). The stakeholder generates the CEM (or tailors a Reference CEM) by listing potential hazards posed to the program and then traces the consequences of each of these hazards through the intermediary events to the final terminal events. The process is then done in reverse: taking the terminal events, adding in any that are still missing, and working backwards on how these might come about. The causal connections between each intermediary event are examined to see if there are any additional connections not previously noticed. Finally, lessons learned databases, case studies, and other experts are consulted to generate additional hazards, intermediary events, causal connections, and interventions, as well as to verify existing ones. It is envisioned that any of these steps can take place either formally, using automated tools to enumerate possible vulnerabilities, or informally, relying upon the stakeholder's own experience. CEM is fundamentally a qualitative analysis method, though it can be readily adapted into a more quantitative form, by specifying probabilities of transition to each intermediary (Reid & Rhodes, 2018b).

CEM has previously been applied in a case study of a Maritime Security System of Systems (Mekdeci et al., 2012) and to a supply chain case (Rovito & Rhodes, 2016). More recently, an earlier phase of this research developed a Reference CEM for use by program managers to assess enterprise-level vulnerabilities in the digital engineering/model-centric environment (Reid & Rhodes, 2018b). This work, which was based upon literature reviews, interviews with experts, and other sources, sought to provide program leaders with an entry point into for considering such vulnerabilities. Potential use cases are discussed in Reid and Rhodes (2018b). Key benefits include increased understanding of the causal path and the interrelationships between vulnerabilities.

## Cause-Effect Map for Model-Centric Programs and Enterprises

CEM provides an effective way to describe cascading vulnerabilities within a digital engineering enterprise. Figure 5 shows the Reference CEM generated in this research using literature reviews and interviews with experts, among other sources. Nineteen intervention

points are identified as potential opportunities for breaking causal chains that may be triggered by external events.

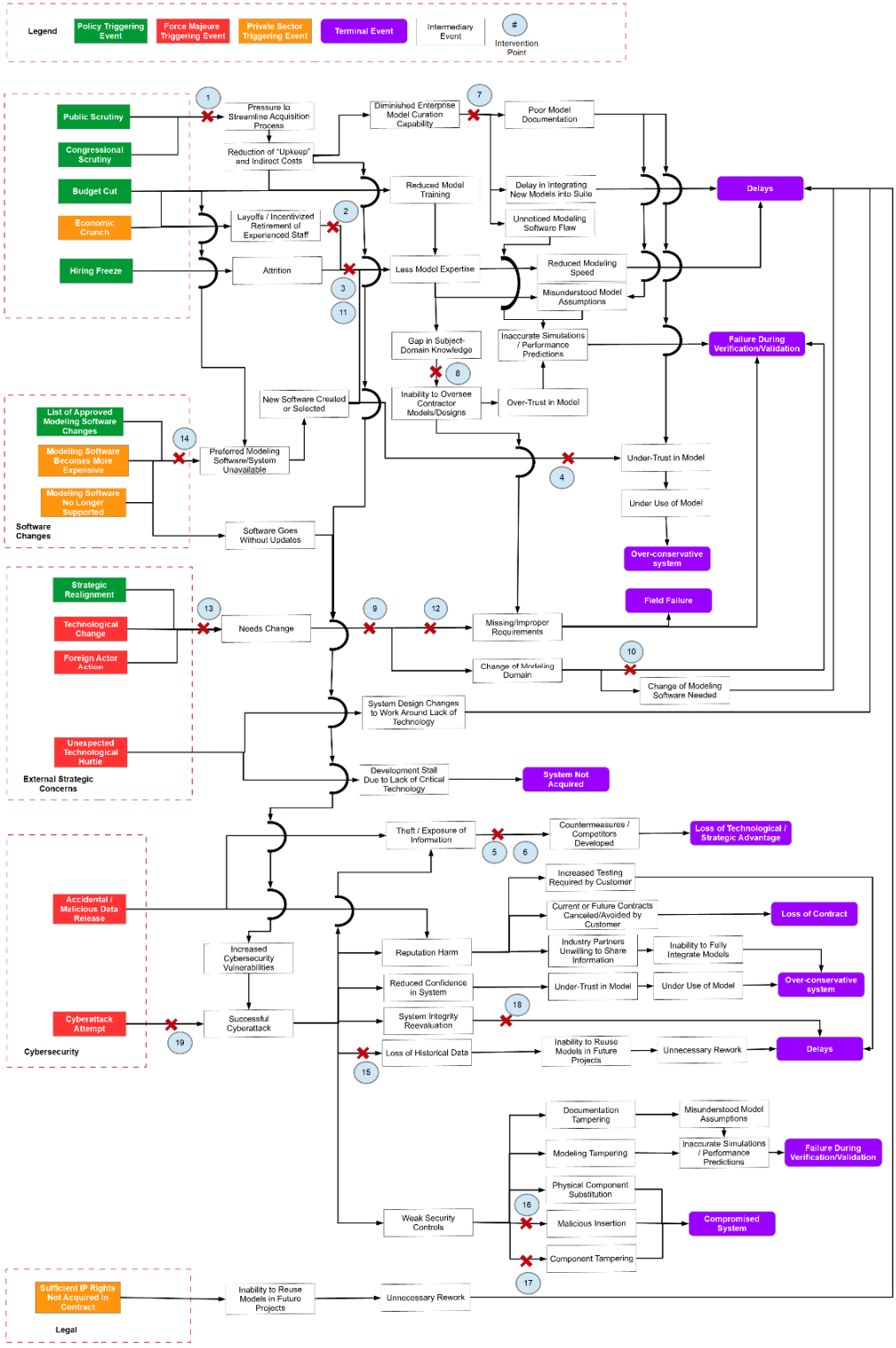


**Figure 5. Reference Cause-Effect Map**
(Reid, 2018)

External triggers that result in similar vulnerability chains are grouped together in the map. By “similar,” we mean that these vulnerability chains either involve many of the same intermediary events or that they involve the same part of the program. For this map, the external triggers were classified into three different domains, defined as follows:

- Force Majeure (red boxes): This is a general term for an event that is the result of actions beyond the possibility of the program enterprise (not just the program manager) to influence. Thus it includes both malicious action and general, unforeseeable events such as Technological Change.
- Policy (green boxes): An event that is the result of intentional decisions made at the organizational or enterprise level. In the case of a government-run program, this includes oversight from Congress and the general public. Non-government organizations may still be impacted indirectly by such oversight, but their proximal triggering event would be different.
- Private Sector (orange boxes): Any event that is the result of the actions of one or more private-sector firms outside the program enterprise.

The purple boxes are the terminal events.

The intervention points on the Reference CEM (Figure 5) are shown in Table 1, where an invention action is defined for each point.

**Table 1. Intervention Points for the Reference CEM Shown in Figure 5**

(Reid, 2018)

| Point # | Intervention Action |
|---|---|
| 1 | Initiate internal assessment and a public relations strategy |
| 2 | Initiate various non-monetary benefits (e.g., 9/80 schedule) to encourage employees to stay |
| 3 | Seek to share resources and employees with other programs |
| 4 | Hire employees with prior experience with the new software |
| 5 | Compartmentalize sensitive information |
| 6 | Obfuscate sensitive data with false or misleading information |
| 7 | Create documentation and curation processes within the program |
| 8 | Institute handover periods to benefit from contractor expertise |
| 9 | Reevaluate the training regime and needed fields of expertise |
| 10 | Increase the amount of testing conducted |
| 11 | Increase use of contractors/consultants to maintain expertise level |
| 12 | Reevaluate the requirements with the client and other stakeholders |
| 13 | Design for modularity to minimize impact on system |
| 14 | Negotiate with client/end-user to see if they are able to pay for the software |
| 15 | Maintain isolated but readily accessible back-ups of data |
| 16 | Conduct reviews/comparisons of models between lifecycle stages |
| 17 | Use multiple independent simulations or component checkers |
| 18 | Maintain isolated, independent backup equipment while primary equipment is evaluated |
| 19 | Conduct regular “red-team”/penetration test exercises |

### *Observations on Intervention Points*

Reid (2018) found that intervention points identified in the Reference CEM (Figure 5) tend to be in the first half of the vulnerability chains, with several immediately after an external trigger. This suggests the need for monitoring for potential or imminent external triggers and being ready to respond as soon as, or even in advance of, their manifestation.

The Reference CEM can be used to guide the attention to various vulnerabilities. For instance, it should be noted that within the “active modeling” set of intermediate events (inside the blue box of Figure 6) there are relatively few intervention points identified, despite the high number of vulnerability chains that pass through that section of the Reference CEM. The primary intervention point identified in that section, number 7, is “Create documentation and curation processes within the program” (see Table 1).

This relative lack of intervention points may represent the unfamiliarity of program leaders with digital engineering processes and how to intervene in them. This suggests that further work would be useful in identifying potential interventions in this section of the map and educating program leaders concerning their availability and use.

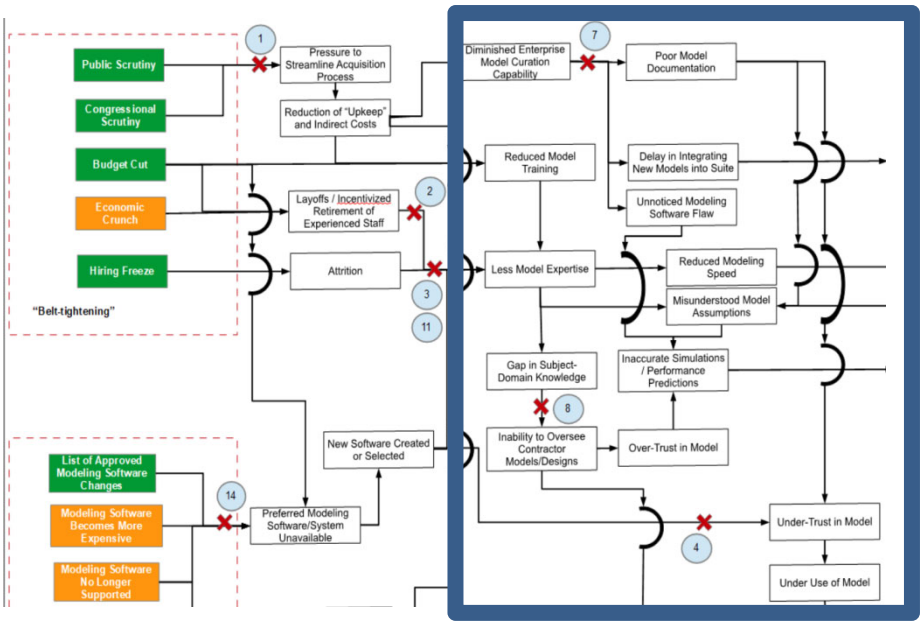


**Figure 6. Excerpt of the Reference CEM Highlighting the “Active Modeling” Portion**
(Reid, 2018)

While this portion of the chain has one intervention identified, certain vulnerability chains have multiple intervention points identified at multiple stages. For instance, several of the vulnerability chains that pass through the Needs Change event have three intervention points each (and the others have at least two), as shown in Figure 5.

According to Reid (2018), this suggests there may not be as much of a concern about these vulnerabilities, due to the multiple options of intervention available and the fact that several are positioned multiple events into the chain, giving significant time for response.

An experienced program leader will find some of the listed intervention points to be common sense. For instance, one of the interventions (number 12) following the Needs Change event (see Table 1) is “Reevaluate requirements with the client and other stakeholders.” This degree of occasional obviousness is not unique to CEM but is true of all vulnerability assessment techniques. The point of these techniques is not just to identify new vulnerabilities and interventions, but to consistently track and assess them so that all options

are available. A case in point is that even experienced pilots still use a checklist (and surgeons really should be; Haynes, Berry, and Gawande, 2015).

It should be noted that the Reference CEM shown in this paper does omit vulnerabilities and interventions that are entirely unchanged. For example, practices like the security clearance system and restricting the use of digital storage media will remain necessary, effective interventions that are not significantly impacted by MCE environments. Some historically successful methods may be conflict with MCE environments, for example, the use of SCIFs has been quite successful in preventing unauthorized access to data. The typical use of a SCIF in design, where a small number of engineers work on a task isolated from the outside world, is not directly compatible with an MCE environment structured around model integration and collaboration across teams and locations. While this problem has been previously considered and ways to mitigate this conflict have been proposed (e.g., Reid & Rhodes, 2016), no silver bullet to resolving these tensions exists and it is likely that the increased use of MCE will result in both the exacerbation of some current vulnerabilities and the creation of new ones.

### *Cybersecurity Vulnerabilities*

Literature review and interview-based research have provided useful insights throughout the research. As the initial research progressed, the importance and urgency of considering the cybersecurity vulnerabilities shaped the second phase of study to focus more specifically on these. Reid (2018) conducted interviews with systems engineers and program managers from a variety of fields, including defense, aerospace, manufacturing, and semiconductors. The interviews explored these program cybersecurity vulnerabilities in general, and in context of model-centric approaches. Four issues commonly were cited:

- Cybersecurity needs to be thoroughly considered much earlier than it commonly is, preferably in the proposal generation stage.
- Program managers and systems engineers are sometimes intimidated by cybersecurity issues and thus seek to pass them onto specialists later in the acquisition process.
- MBSE and MCE toolset developers and proponents have not done a thorough enough job of considering programmatic cybersecurity vulnerabilities, though the tools are thought to be quite effective at designing for cybersecurity in regard to end-systems.
- Traditional programmatic cybersecurity defensive practices tends to quite effective in traditional engineering programs, but the increased use of MCE, particularly for multi-site collaboration, could change this (Reid & Rhodes, 2018a).

### *Non-Technical Influences and Impacts*

One set of vulnerabilities that came up repeatedly in both the interviews and experiment sessions in our research (Reid & Rhodes, 2018a) were those that passed through the reputation harm intermediate event, as shown in Figure 7.

Despite the frequency that the potential for this vulnerability was raised by experts, few interventions were proposed for post-breach. According to Reid (2018), this suggests that leaders of digital engineering enterprises may need better understanding of potential vulnerabilities leading to breaches in context of digital engineering, as well as more knowledge on how to respond to breaches, particularly prominent ones, instead of solely how to prevent them. While there is evidence in the private sector suggesting that the

reputation harm incurred by a prominent breach does not significantly impact the firm (Lange & Burger, 2017), contractors to the government are known to suffer significant financial penalties due to breaches, even when such a breach is unrelated to their government duties (Braun, 2014; Overly, 2017). In a defense acquisition environment, there is thus significant incentive to having program leadership (and the enterprise as a whole) well-prepared to respond to major breaches.

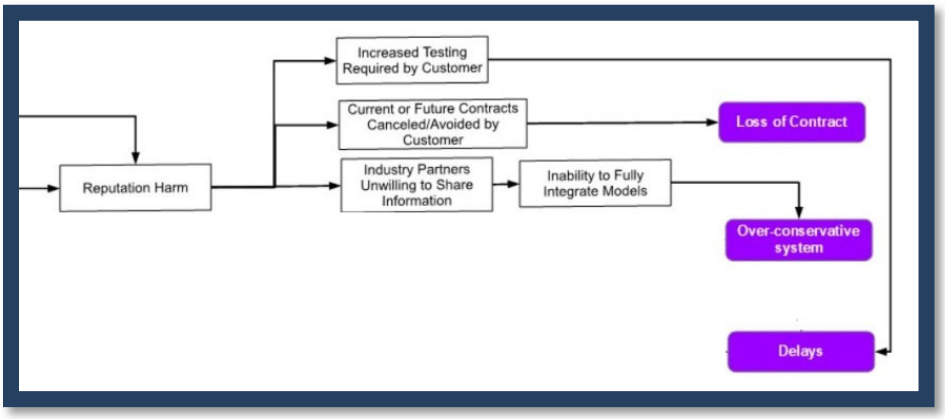


**Figure 7. Reputation Harm Vulnerabilities**
(Reid, 2018)

***Relevant Research From Other Fields***

Huff, Medal, and Griendling (2018) present a methodology for performing vulnerability assessment and decision analysis of critical infrastructure using the approach of model-based systems engineering. The work focuses on physical security of critical infrastructure. Some of their findings may provide useful insights for vulnerability assessment of infrastructure within model-centric enterprises.

The literature on the manufacturing sector offers interesting observations and new research of relevance to vulnerability assessment of model-centric enterprise environments. Burnson (2017), discussing a recent Deloitte study on cyber vulnerabilities in manufacturing supply chains, states “one-third of all manufacturers sampled admitted to not having performed any cyber risk assessments of the industrial connected devices operating on factory floors.” While data is not available, from discussions with experts in the engineering domain, it seems likely that there would be a similar situation in regard to whether cyber risk assessments have been performed for model-centric engineering environments with connected hardware and software.

DeSmit et al. (2016) discuss research on cyber-physical vulnerability assessment in manufacturing systems that uses an approach that employs intersection mapping. According to these authors, “no literature is aimed at assessing cyber-physical vulnerabilities for manufacturing systems.” With similarities of manufacturing facilities with facilities used in model-centric enterprises, their research may offer useful insights to our research. DeSmit et al. (2016) describe their approach as “based on the principle that vulnerabilities in manufacturing systems occur at intersections (and intra-sections, referred to collectively as intersections) of cyber, physical, cyber-physica,l and human entities that embody a manufacturing system.” Similar to the CEM approach, their method maps intersections and assesses the impact at intersection nodes. They evaluate five characteristics: loss of information, inconsistency, relative frequency, lack of maturity, and time until detection. In

their method, vulnerability impact assessment (Low, Medium High) is assessed for the characteristics at each of the nodes. This offers an interesting approach to qualitative assessment measures for vulnerability. Another noteworthy facet of their work that resonates with our research is that human entities are included in defining intersections.

## Discussion

Knowledge gathered in this research indicates that program leaders do not formally grapple with vulnerabilities within the program and overall enterprise to the extent they do with vulnerabilities related to the end-system. Cause-Effect Mapping, with re-conceptualizing vulnerabilities as causal chains, enables program and enterprise leaders to identify connections, categories, and potential interventions in the vulnerability chains. The research indicates identifying external triggers and representing vulnerabilities as chains enables a more detailed assessment of how interim cascading events can result in significant terminal outcomes. Use of the CEM approach assists in understanding these causal chains, and decomposes a vulnerability in a manner that encourages finding multiple options for mitigation. Particular choices for disrupting a harmful causal chain are useful for considering where and when to place interventions based on the specific nature of the situation.

### *Limitations*

While a fully-developed generalized CEM Reference Map could provide overall benefit to digital engineering programs, the fact that enterprise and programs are unique makes it difficult to accomplish this without much more extensive application and study. Secondly, digital engineering practice and supporting infrastructure are still evolving, so limited knowledge exists at present. Nonetheless, programs and enterprises may derive significant benefit by the activity of constructing a reference map for their unique situation. The process of generating the map invokes thoughtful discussion and anticipating potential hazards that may have been introduced as a result of the digital transformation. The approach of considering vulnerabilities as casual chains yields rich discussion, regardless of whether an overall map is developed. This research has demonstrated the approach to constructing a CEM Reference Map and illustrates content included in the map; a fully-developed comprehensive reference map will require a more extensive investigation.

### *Research Directions*

There are several areas of desired future research direction. First, additional study is needed on leading indicators of vulnerability in digital engineering enterprises, along with potential mitigation strategies. Specific approaches to quantification of interventions in breaking vulnerability causal chains is desired, as related to cost, benefit, importance, frequency, etc. Additional research on dynamic simulation using System Dynamics (SD) with CEM is a promising area to explore given the complexities that will be inherent in a fully populated reference CEM (further discussion is found in Reid, 2018). Implementation of an interactive method used to perform vulnerability assessment using a reference map is a future area of inquiry. Additional research is needed to identify relevant investigation in the systems engineering field; for example, Wach and Salado (2018) describe a plan to discover patterns of unknown vulnerabilities associated with SysML. And, further collaborative research with government and industry is desired to identify additional vulnerability chains and enable testing and scaling the method.

### *Summary*

In summary, digital engineering transformation naturally introduces new vulnerabilities within programs and enterprises. Causal chains provide a useful way to understand how external triggers lead to cascading intermediate events that result in

specific outcomes. Understanding a vulnerability chain provides program leaders with increased knowledge and options for inserting interventions to avoid undesired vulnerability outcomes. With more experience and knowledge of vulnerabilities inherent in digital engineering practice and infrastructure, the systems community may find it valuable to establish a generalized Reference CEM that can guide future programs and enterprises to assess and manage vulnerabilities, leading to more successful program outcomes. Related research on model curation views a CEM Reference Map as an enabling tool (Rhodes, 2019) for vulnerability assessment of enterprises.

## References


Braun, S. (2014, September 10). OPM plans to terminate contracts with USIS. Federal News Radio. Retrieved from https://federalnewsradio.com/management/2014/09/opm-plans-to-terminate-contracts-with-usis

Burnson, P. (2017, March). New Deloitte study identifies cyber vulnerabilities in manufacturing supply chains. Supply Chain Management Review. Retrieved from https://www.scmr.com/article/new_deloitte_study_identifies_cyber_vulnerabilities_in_manufacturing_supply

DeSmit, Z., Elhabashy, A., Wells, L., & Camelio, J. (2016). Cyber-physical vulnerability assessment in manufacturing systems. In 44th Proceedings of the North American Manufacturing Research Institution of SME (Procedia Manufacturing), 5, 1060–1074.

DoD. (2018, June). Department of Defense systems engineering strategy. Retrieved from https://www.acq.osd.mil/se/docs/2018-DES.pdf

German, E. S,. & Rhodes, D. H. (2017, May). Model-centric decision-making: Exploring decision-maker trust and perception of models. In Proceedings of the 15th Conference on Systems Engineering Research.

Haynes, A. B., Berry, W. R., & Gawande, A. A. (2015, May). What do we know about the safe surgery checklist now? Annals of Surgery, 261(5), 829–830.

Huff, J., Medal, H., & Griendling, K. (2019, March). A model-based systems engineering approach to critical infrstructure vulnerability assessment and decision analysis. Systems Engineering. 22, 114–133.

Lange, R., & Burger, E. W. (2017). Long-term market implications of data breaches, not. Journal of Information Privacy and Security, 13(4).

LeSaint, J., Reed, M., & Popick, P. (2015, April). System security engineering vulnerability assessments for mission-critical systems and functions. In 2015 Annual IEEE Systems Conference (SysCon) Proceedings (pp. 608–613).

Mekdeci, B., Ross, A. M., Rhodes, D. H., & Hastings, D. E. (2012, March). A taxonomy of perturbations: Determining the ways that systems lose value. In Proceedings of the 6th Annual IEEE Systems Conference. IEEE.

Overly, S. (2017, October). IRS temporarily suspends contract with Equifax. Politico. Retrieved from https://www.politico.com/story/2017/10/12/irs-equifax-contract-suspended-243732

Reid, J. B. (2018). Assessing and mitigating vulnerability chains in model-centric acquisition programs (Master's thesis). Cambridge, MA: Massachussetts Institute of Technology.

Reid, J. B., & Rhodes, D. H. (2016, March). Digital system models: An investigation of the non-technical challenges and research needs. In Proceedings of the Conference on Systems Engineering Research.

Reid, J. B., & Rhodes, D. H. (2018a, May). Applying cause-effect mapping to assess cybersecurity vulnerabilities in model-centric acquisition program environments. In Proceedings of the 15th Annual Acquisition Research Symposium. Monterey, CA: Naval Postgraduate School.

Reid, J. B., & Rhodes, D. H. (2018b, May). Assessing vulnerabilities in model-centric acquisition programs using cause-effect mapping. In Proceedings of the 15th Annual Acquisition Research Symposium. Monterey, CA: Naval Postgraduate School.

Reymondet, L., Rhodes, D. H., & Ross, A. M. (2016, April). Considerations for model curation in model-centric systems engineering. In Proceedings of the 10th Annual IEEE Systems Conference. IEEE.

Rhodes, D. H. (2018, April). Using human-model interaction heuristics to enable model-centric enterprise transformation. In Proceedings of the 12th Annual IEEE Systems Conference. IEEE.

Rhodes, D. H. (2019, April). Model curation: Requisite leadership and practice in digital engineering enterprises. In Proceedings of the 17th Conference on Systems Engineering Research.

Rovito, S. M., & Rhodes, D. H. (2016, April). Enabling better supply chain decisions through a generic model utilizing cause-effect mapping. In Proceedings of the 10th Annual IEEE Sytems Conference. IEEE.

Wach, P., & Salado, A. (2018, March). A research plan to discover patterns of unknown vulnerabilities associated with adopting SysML. In Proceedings of the 16th Conference on Systems Engineering Research.

## Acknowledgement & Disclaimer

This material is based upon work supported by the Acquisition Research Program under Grant No. HQ00341810013. The views expressed in written materials or publications, and/or made by speakers, moderators, and presenters, do not necessarily reflect the official policies of the Department of Defense nor does mention of trade names, commercial practices, or organizations imply endorsement by the U.S. Government.


ACQUISITION RESEARCH PROGRAM
GRADUATE SCHOOL OF BUSINESS & PUBLIC POLICY
NAVAL POSTGRADUATE SCHOOL
555 DYER ROAD, INGERSOLL HALL
MONTEREY, CA 93943

www.acquisitionresearch.net