30th Annual INCOSE international symposium
Cape Town, South Africa
July 18 - 23, 2020

Contextually Aware Agile Security in the Future of Systems Engineering

Rick Dove
Paradigm Shift International
dove@parshift.com

Keith D. Willett
Department of Defense, USA
Keith.Willett@incose.org

Abstract. A recurring principle in consideration of the future of systems engineering is *continual dynamic adaptation*. Context drives change whether it be from potential loss (threats, vulnerabilities) or from potential gain (opportunity-driven). Contextual-awareness has great influence over the future of systems engineering and of systems security. Those contextual environments contain fitness functions that will naturally select compatible approaches and filter out the incompatible, with prejudice. This paper provides archetypes that 1) characterize general systems engineering for products, processes, and operations; 2) characterize the integration of security with systems engineering; and, 3) characterize contextually aware agile security. The purpose of this paper is to provide a conceptual understanding of the problem space and derive general security strategies necessary to deal with that problem space, with the intent to provide an initial foundation for subsequent tactical implementations of agile security approaches.

Introduction

The Future of Systems Engineering (FuSE) is an INCOSE led multi-organization collaborative initiative that has identified a number of specific topics to be investigated (INCOSE nd). The authors of this paper accepted the responsibility to lead and initiate the FuSE topic on security. This paper and another (Dove, Willett 2020) attempt to lay some foundational thought for subsequent security-topic project work.

The future of systems engineering as an evolution of the discipline will necessarily produce systems that are dynamically attentive and responsive to the environments in which they operate. A system interfaces with and interacts with its operating environment, and remains viable (capable of working successfully) and relevant (appropriate to current desires) only to the extent to which it is operationally compatible with the *current order* of its environment (Willett 2020). Environmental factors include shaping forces that define that which is necessary for constituent systems to remain compatible with the environment. A change in environmental shaping forces may require a change in the systems within that environment for those systems to remain viable and relevant; or, in other words, for the system to remain compatible with the environment. One might say that the current order tolerates a clear and present system only to the extent that homeostasis can be maintained or effectively readjusted.

Security is now evolving from a cyber (virtual effects) focus to a cyber-physical system appreciation (virtual and real effects). This is a generally inward view of the system of interest (SoI). There is an important outward looking social dimension to systems that has gone relatively unrecognized beyond the human-social aspect. The social dimension will play a major role in the future of systems engineering, with key implications for system security; i.e., the evolution continues to a *cyber-physical-social* appreciation that includes people, process, technology, and environment.

The social dimension includes the value of *active collaboration* and *teaming*. Active collaboration brings a diversity of knowledge and thought – of particular relevance when complexity is increasing in both SoIs and their operating environments. Teaming brings faster and more effective action, which is of particular relevance when the adversary is increasingly innovative in method and speed of attack.

Teaming depends on engaged collaboration and includes human and socio-technical teams. For security, teaming includes socio-technical systems collaborating to assure SoI value-delivery via safeguarding the SoI, its function and functional exchanges, and its environment.

The social dimension of systems engineering includes leadership and stakeholder *collaborative engagement*, SoI *symbiosis* in a greater System of Systems (SoS), and *compatibility* with the current order. Social aspects for system security include insiders and outsiders as potential system adversaries, adversary social probes, peer behavior awareness and collaboration among interacting systems, and stakeholder personal privacy. Extrospective and introspective awareness and response is both a *systems* (product) and *systems engineering* (process) social aspect in the future of systems engineering.

One aspect of the social dimension of particular interest for security is cohesion – the existence, strength, and dynamics of symbiotic bonds among system elements, both virtual and physical.

FuSE recognizes that systems and their environments exhibit emergent behavior (are complex), that systems belong to and are themselves systems of systems, and that the current order and respective constituency are unpredictably dynamic. The dynamics of the current order require security structures and strategies equally dynamic. The adversary is agile, initiating relentless disruptions to the current order and its constituency that requires equally agile security.

FuSE leadership has asked what will good look like when FuSE delivers systems security. Our succinct response: Security Engineers will be active members of the Systems Engineering team. Security will be rapidly reconfigurable, augmentable, and composable. We will monitor system and component behavior for anomalous behavior. We will use modeling to predict variations and prepare contingent courses of action to be proactive and minimize response time. Security will support rather than impede personal and organizational productivity. System components will be self-protective; e.g., *techno-social contracts* for mutual autonomous protection (Dove, Willett 2020).

FuSE leadership has also asked what is stopping us from doing this now. Our succinct response: SE relates to security engineering as an independent specialty practice. Security is viewed as a non-functional cost. Stakeholders consider legislative and standards compliance sufficient. Actionable research is in early stages. Detailed functional requirements rather than capability outcomes drive SE contracting.

This paper provides a relatively succinct foundation for systems engineers to appreciate the needs and intents of security in the future of systems engineering. Objectives include a characterization of the problem space that shapes the future, strategy considerations for addressing the problem space, and some next steps for moving toward solutions. Methods for deriving strategies are detailed throughout and summarized in concluding remarks.

The characterizations and strategies are a short list of *necessary* considerations with no claim of *sufficiency*. System security is not a bounded *logical* entity; rather it is an integrated *structural* and *behavioral* entity distributed throughout a SoI and its environment, where a SoI may be a SoS at any scale.

Characterizing the Environment

The environment of concern encompasses both external (current order) and internal (value-delivery); and recognizes both threats (potential loss) and opportunities (potential gain). CURVE is a framework with which to characterize the shaping forces of environments that require an *agile-systems engineering*[1] approach (Dove, Schindel 2019). CURVE is extensible to accommodate the integration of security as part of continual dynamic adaptation.

The CURVE framework has elements as follows:

- Caprice: Unknowable situations. Unanticipated system-environment change.
- Uncertainty: Randomness with unknowable probabilities. Kinetic and potential forces present in the system.
- Risk: Randomness with knowable probabilities. Relevance of current system-dynamics understandings.
- Variation: Knowable variables and associated variance ranges. Temporal excursions on existing behavior attractors (a reference to complex system behavior trajectories).
- Evolution: Gradual successive developments. Experimentation and natural selection at work.

CURVE characterization of the general SE problem space, as shown in Table 1, column 1, G-CURVE, arose from the INCOSE Agile Systems Life Cycle Model project, and first appeared in (Dove 2019, p. 10). This general SE CURVE provides structure for the security engineering S-CURVE shown in Table 1 column 2, as the general SE CURVE is equally appropriate for the engineered-system (product), systems engineering (process), and the engineered-workflows (operations).

The focus of this paper is Table 1 column 2, the characterization of the system security engineering problem space (S-CURVE), which integrates security into all three aspects of security product, process, and workflow. Following Table 1 a bulleted summary of S-CURVE elements is offered, followed by a discussion of each bullet for the future of system security engineering. The summary is purposefully a short list to encompass the general issues succinctly. Element numbering in each category provides traceability of derived strategies to the CURVE need.

[1] *Agile-systems* are solutions designed for *continual dynamic adaptation*.

The goal of integrating security and systems engineering is to sustain SoI compatibility with their environments. We don't have to guess at what those environmental shaping forces will look like. William Gibson's frequently repeated quotation tells us why: "The future is already here – it's just not very evenly distributed[2]," i.e., look around, for it is already starting to happen.

Table 1 is a discussion of the security environment S-CURVE (Table 1 column 2). Each element in an S-CURVE category is a general concept with many diverse instances under that concept; however, all instances are amenable to general strategy considerations. The sparse examples indicate that the future is already here, with no intent to provide a lengthy set of evidence.

Table 1: Environment CURVE Characterization

General SE CURVE (G-CURVE)	**FuSE System Security CURVE (S-CURVE)**
Caprice	
1. Survivability (i.e., current order compatibility) 2. Occurrence and nature of emergent behavior 3. Game-changing technologies 4. Availability of symbiotic social relationships	1. Innovative attack and response methods 2. Emergent cascades and complexity effects 3. Artificial intelligence and quantum technologies 4. Collaborative symbiosis
Uncertainty	
1. Relevance (i.e., appropriate to current desires) 2. Cohesion in systems and SoSs 3. Integrity and symbiosis of social relationships	1. Cost vs. value evaluations 2. Operational physical relationships 3. Operational social relationships
Risk	
1. Viability (i.e., capable of working successfully) 2. Cohesion among constituent parts 3. Inadequate recall of lessons learned	1. Design and execution 2. Addressing adversity effectively 3. Knowledge assimilation
Variation	
1. Operational environments 2. Social compatibility 3. Human resource loading	1. Attack and response criticality 2. Peer and community behavior 3. Adequacy of incident response capability
Evolution	
1. More operating environment complexity 2. More SoI complexity 3. Shorter SoI static viability 4. New technology options 5. New malevolent threats to viability 6. Greater social involvement	1. External SoS 2. Internal SoS 3. Growing attack community (skills and scope) 4. Increasing technical innovation 5. Increasing attack value 6. Increasing collaborative connectivity

The following is a discussion of the security environment CURVE (right side of Table 1). Each element in a CURVE category is a general concept with many diverse instances lumped under the concept, but all instances are amenable to general strategy considerations spoken to later. The sparse examples indicate that the future is already here, with no intent to provide a lengthy set of evidence.

Caprice

- **Innovative attack and response methods** are unpredictable with non-deterministic impact on system survivability. Security attack discovery and analysis typically results in fixing known

[2] Gibson's verbally delivers his famous quotation in a 30-Nov-1999 NPR recording. He claims he never wrote it. https://ondemand.npr.org/anon.npr-mp3/npr/totn/1999/11/19991130_totn_science_fiction_becoming_science_fact.mp3. See https://quoteinvestigator.com/2012/01/24/future-has-arrived/ for more detail on origin.

exploited vulnerabilities and augmenting detection mechanisms. This pushes adversaries to continuously find and employ new methods for which there are no safeguards. Resourceful adversaries with interest in high value targets invest significantly to discover and catalogue unexploited system vulnerabilities and await future opportunity. Discovery focuses on both broad infrastructure elements common to many systems (e.g., computer operating systems), as well as peculiarities in more unique systems of interest (e.g., a specific nuclear refinement facility configuration).

- **Emergent cascades[3] and complexity effects** are unpredictable. Increasing system complexity and interconnectedness increases the potential for undesirable emergent cascades; principally an architecture and design issue. Cascades may happen within a system of many components (e.g., 911 World Trade Center collapse), as well as within a system of systems (e.g., Northeast US Blackout of 2003). Undesirable security effects may also occur from unexpected complex interactions of systems within a system of systems.

- **Artificial intelligence (AI) and quantum technologies** have unpredictable employment maturation. AI technology is in its infancy, but evolving rapidly and in early employment by both attacker (Stupp 2019) and protector (Capgemini Research Institute 2019). Practical quantum computing is an undelivered promise as yet, but a broader spectrum of quantum technologies are much closer to deployed application (Wallden, Kashefi 2019), with quantum key distribution already commercially available. Large budget nation states are actively engaged in a game changing arms race on both the attack and protect side, with trickle down effects as private information is publicly disclosed.

- **Collaborative symbiosis is unpredictable**. This can be a threat when practiced by attackers, and an opportunity when practiced by protectors. In both cases symbiotic collaboration can be unpredictable with the unexpected appearance of new collaborators as well as the failure of existing collaborative relationships. Both attacker and protector can benefit from a larger community of knowledge and awareness. Attackers make use of various internet forums that share newly discovered target values, vulnerabilities, and attack methods. Protectors often exhibit reluctance in sharing information, though there are some closed community forums. Nodes in some ad hoc networks collaborate with each other about suspiciously behaving nodes. Failed symbiotic collaborative relationships for protectors are often issues of changes in priorities, e.g., reallocation of security funding to something considered more important, or the sudden withdrawal of an important collaborative partner that is otherwise engaged in a higher priority need.

Uncertainty

- **Cost vs. value evaluations** are uncertain. Security isn’t free, nor can it be guaranteed, and objective calculation of value is problematic; yet security is critical to sustain system relevance. Constraints on development and operational budgets often set allocations for security and provide little flexibility in trade-off decisions being in competition with functional and opera-

[3] Cascade is something arranged or occurring in a series or in a succession of stages so that each stage derives from or acts upon the product of the preceding.www.merriam-webster.com/dictionary/cascade, last accessed 10-Oct-2019.

tional priorities. The perception is that security is a non-functional cost in the category of *nice-to-have*.

- **Operational physical relationships** are uncertain. During system operation physical components my be removed for maintenance or replacement. Components may suffer physical damage, or removal by accident or malicious activity. A change in physical relationships protecting components, regardless of cause, may affect security; e.g., protective perimeter structures or anti-tamper mechanisms.

- **Operational social relationships** are uncertain. During system operation relied-upon teaming and collaborative relationships may be impaired. Examples include unavailable critical subject matter experts, obstructed communication channels, otherwise engaged collaborative sources, and whimsical employee visitor-badge monitoring.

Risk

- **Design and execution** is a risk, whether done by different people or the same person. Done by different people, there is the risk of misinterpreting design intent, assuming design intent is adequate. Done by the same person, there is the risk of insufficient knowledge, experience, and threat evolution awareness to create design intent and execution outcome. In all cases design and execution are at risk from insufficient security appreciation when high level system architecture and design is frozen.

- **Addressing adversity effectively** is a risk. Historically, systems engineers treat security as a non-functional requirement with the objective of satisfying compliance standards. These standards are publicly available and known to the attack community. Compliance standards change slowly, the attack community innovates quickly. Security effectiveness is constrained and enabled by systems engineering decisions made early in the systems engineering process.

- **Knowledge assimilation** is a risk. Assimilation is the process of taking in and fully understanding information or ideas. Systems engineering security decision tradeoffs are based on the assimilated knowledge of decision makers. These decisions are made by the systems engineering team, by business line managers, and by the contracting program manager for contracted systems. In all cases there is a decision risk of competing priorities in the face of insufficient assimilated knowledge.

Variation

- **Attack and response criticality** varies. All attacks are not necessarily equally critical, and multiple attacks of different criticalities may occur simultaneously. Evaluating attack criticality establishes response priority. Giving priority to an attack response may be in conflict with achieving other objectives of the moment, e.g., gaining attack knowledge through observation or completing higher priority objectives. Criticality evaluations may change as an attack progresses.

- **Peer and community behavior** varies. The degree of relied-upon human collaborative and teaming engagement can vary in time (e.g., attention limitations, perceived slight). The degree of community compatibility can vary (e.g., degraded communications, new people and systems or components not compatibly integrated as yet, policy changes in systems within an SoS). Emergent behavior occurs (e.g., personality conflicts, contradicting system and component interactions).

- **Adequacy of incident response capability** varies. Overloaded responders. Insufficient knowledge or skills for the immediate response need.

Evolution

- **External SoS**. Cyber connectivity, shrinking costs and size, and increased capability fuels growth in systems connected to systems for greater capability delivery. IoT is a current example.
- **Internal SoS**. Cyber-physical systems and components are becoming more complex and incorporating new technologies like AI.
- **Growing attack community**. Part of this growth is due to nation state high quantity staffing; part is due to outsourcing computer jobs to low labor rate countries and then moving those jobs to other countries, leaving jobless computer literates in search of income; part is due to growing computer literacy worldwide looking for skill application opportunities.
- **Increasing technical innovation**. The rate of technical innovation is an exponential curve fueled by knowledge that builds on knowledge.
- **Increasing attack value**. The rise and success of ransomware and the strategic investments to establish cyberspace as a warfighter domain[4] are two examples.
- **Increasing collaborative connectivity**. DevSecOps is in early stages with a growing focus. Internet tools for remote collaboration are gaining maturity and methods for effective remote teaming approach the tipping point.

The CURVE characterization establishes goals, in the form of a *situational profile* that should have engineering and operational strategies compatible with the current order. CURVE informs strategy derivation. Strategies to address the CURVE environment come in two forms, one for *engineering considerations*, and one for *operational considerations*. The next section will deal with engineering strategies. A companion paper deals with operational strategies (Dove, Willett 2020).

Strategy Considerations

The dynamics of the environment indicate the need for an agile approach to security. A response situation analysis framework is employed as a general framework suitable for our needs (Dove, LaBarge 2014). Minor liberty is taken with this framework to cast it more directly at strategy delineation, and populate it with security versions of general agile systems engineering response needs that emerged from the INCOSE Agile Systems Engineering Life Cycle Model project (Dove, Schindel 2019). The focus is limited to what appears to be necessary general strategy concepts, with no claim of sufficiency, nor any intent at this point to deal with tactical variations

[4] Many consider "cyber war" to be a mischaracterization; i.e., there is only *war* and that through which to engage in war… land, sea, air, space, and cyberspace as the five warfighter domains.

for realizing the strategies. One purpose of this paper is to provide a conceptual strategy foundation for agile security to guide subsequent work on solution tactics.

Agile security is applied continuously during life cycle operations via *dynamic engineering* (e.g., creating new modular security resources) and *dynamic composability*[5] (e.g., creating and invoking actions composed of appropriate modular resources). Security engineering and action composability never ends – new threats, new vulnerabilities, and new opportunities emerge throughout the operational lifecycle requiring immediate attention.

The engineering process strategies in Table 2 are intended to address reference numbered CURVE elements (e.g., C1 is the first Caprice element, U2 is the second Uncertainty element, etc.) in Table 1. The Need (goal) column in Table 2 identifies strategies intended to maintain and restore compatibility with the system security CURVE characteristics identified in the Intent column of Table 2. ConOpsCon is a short way to express both ConOps (Concept of Operations) and OpsCon (Operational Concept).

Table 2: Security Engineering Strategy Needs and Intents

Need	Intent (S-CURVE considerations)
Proactive strategies for creating and eliminating:	
• Awareness of opportunity and threat • Response actions and options • Assimilated memory • Response action decisions	• All S-CURVE elements • C1, C4, R1 • C1, R3, V3, E1 • C1, V1, E4
Proactive strategies for improving:	
• Awareness of impediments during engineering • Memory in culture, actions/options, ConOpsCon • Action and option effectiveness	• C1, C4, U3, R1 • R3 • C1, V3
Proactive strategies for infrastructure migration anticipation:	
• New fundamentally-different types of opportunities • New fundamentally-different types of threats	• E1, E2, E4, E6 • E3, E4, E5, E6
Proactive strategies for modifying capability:	
• Actions and options appropriate for needs • Personnel appropriate for needs • Processes appropriate for needs	• C1, C4, R1 • R1, U3 • C4, U3, R3, V3
Reactive strategies for correcting:	
• Insufficient awareness • Ineffective actions • Wrong decisions	• C1, C2, V2, V3 • C1, R2, V3 • C4, U1, V1
Reactive strategies for accommodating variable:	
• Effective actions and options • Effective evaluations	• All S-CURVE elements • All S-CURVE elements
Reactive strategies for expanding and contracting:	
• Capacity for necessary simultaneous activities	• C1, C4, R1
Reactive strategies for reconfiguring:	
• Reusable actions • Personnel involved in an activity	• All S-CURVE elements • All S-CURVE elements

[5] Composable: to make or form by combining things, parts, or elements (e.g., modules); https://www.dictionary.com/browse/composable?s=t

The framework in Table 2 has four *proactive* strategy categories, and four *reactive* strategy categories. Proactive strategies are generally triggered by an opportunity to generate new value in process or product (seek gain). Reactive strategies are generally triggered by a threat to process or product which demands an action (avoid or withstand loss).

Proactive creation and elimination

Identify the artifacts/data/knowledge that must be created or eliminated during engineering. Elimination is a constant pruning activity that expunges obsolete or ineffective elements to minimize bloat and preclude mistaken usage. For creation the distinguishing feature is the development of something new that is not currently present in any modifiable form. Strategies are those that require creation of artifacts.

- **Awareness of opportunity and threat** (all S-CURVE elements): Opportunities to take advantage of and impediments during engineering (threats) can happen in virtually any CURVE element. Dynamic engineering awareness can minimize rework and maximize effectiveness. Consider (highly) a participating role for security engineering on the systems engineering team, and means and instrumentation to monitor engineering performance.
- **Response actions and options** (C1, C4, R1): Innovative attacks require the creation of innovative responses, and may require new collaborative partners. Engineering design and development are creative activities with effectiveness risk. Consider tasks and responsibilities for developing and evolving a knowledge base of possible collaborative resources and techniques.
- **Assimilated memory** (C1, R3, V3, E1): Analyze and document innovative attack methods; document and share innovative response methods for assimilation. Analyze incident response for cause-effect and design mitigation. The nature of the evolving external SoS can affect design adequacy in real time, and needs broad assimilation. Consider tasks and responsibilities for memory assimilation with shared stories to enculturate and document in ConOps and OpsCon, and in reusable response methods and attack analysis techniques.
- **Response action decisions** (C1, V1, E4): Enact decisions during engineering when time to decision is critical. During engineering consider building and maintaining quick-use decision-making trees and/or AI capability to automate or human-assist engineering decision making.

Proactive improvement

Identify engineering performance that will be expected to improve over time in the continuous engineering activities. Knowing this can avoid cul-de-sacs in engineering processes that impede likely future engineering performance improvement. The distinguishing feature is performance of existing engineering activity, not the addition of new activity. Strategies are generally those involving competencies and performance factors, and are often the focus of continual, open-ended campaigns.

- **Awareness of impediments during engineering** (C1, C4, U3, R1): Engineering is generally a team activity. Security engineers have specialty knowledge that can beneficially inform system engineering architecture and design. Stay abreast of best practice and innovative security threats and opportunities. Bring in collaborative resources for special needs, and inform tradeoff decisions. Consider (highly) a collaborative role for security engineering on the systems engineering team.

- **Memory in culture, actions/options, ConOpsCon** (R3): Engineering resources come and go. Security engineering never ends as vulnerabilities and threats continue to emerge. New people need to assimilate cultural memory, learn the reusable actions and options, and become familiar with the ConOpsCon. Consider tasks and responsibilities for indoctrinating new people appropriately before their need to perform.
- **Action and option effectiveness** (V1, V3): Effectiveness of actions and options are constrained and enabled by decisions made during engineering, by both the systems engineering team and the security action and option developers. The range of incident response options influences action effectiveness; i.e., the correct response to a trigger event. Consider (highly) a participating role for knowledgeable security engineers on the systems engineering team.

Proactive infrastructure migration anticipation

Identify likely events coming down the road that will require a change in the engineering process infrastructure. Infrastructure includes breadth of human resource capability, policy, ConOps, and OpsCon. The distinguishing feature is a need to change the nature of the infrastructure beyond simply adding or modifying resources. Strategies are generally those that enable the transition to possible and potential next generation capabilities.

- **New fundamentally-different types of opportunities** (E1, E2, E4, E6): Evolution presents opportunities that may require infrastructure modification. Evolution in the external SoS may offer new potential collaborative partnerships, possible technology sharing among friendlies, and early knowledge of coming threats. Evolution in the internal SoS complexity may reveal emergent behavior good for security. Evolution in technology shortens the grace period for becoming aware of, evaluating, and employing useful new technologies. Evolution in collaborative connectivity with maturing remote collaboration methods may offer effective means for distributed teaming.

  Consider an agile architecture (Dove, LaBarge 2014, Dove, Schindel 2019) for security development processes and for security systems that enables affordable and timely infrastructure change; ad hoc briefings to policy makers as evolving opportunities emerge; designated internal or subcontracted responsibilities for monitoring evolution in all opportunity categories; instrumented monitoring of system operation during test and during delivered usage for unexpected emergent security-relevant behavior; early experimental evaluations of promising technologies; and distributed remote teaming.

- **New fundamentally-different types of threats** (E3, E4, E5, E6): Evolution presents threats that may require infrastructure modification. Evolution in attack community growth leads to growth in attack methods. Evolution in technology shortens the grace period for becoming aware of, evaluating, and preparing for mitigation of new potential threats. Evolution in attack values changes the attack intents. Evolution in collaborative connectivity leads to a greater shared knowledge base in the attack comminute and more effective attack teaming.

  Consider an agile architecture for security engineering processes and for security systems that enables affordable and timely infrastructure change; dynamically scalable security engineering resources; designated internal or subcontracted responsibilities for monitoring evolution in all threat categories.

Proactive modification of capability

Identify modifications to existing resources that might need made during engineering. The distinguishing feature is a necessary change in available resource capabilities. Strategies are generally those that require something unlike anything already present, or a change to something that does exist.

- **Actions and options appropriate for needs** (C1, C4): Threats evolve beyond what current actions and options can handle, and time is of the essence. Possible and existing collaborative partners come and go unpredictably. Consider an agile architecture that enables the addition of new collaborators, new action options, and the modification of existing action options in minimal time and no side effects; maintain a stable of potential SME collaborators and know the real-time availability of current collaborators.
- **Personnel appropriate for needs** (R1, U3): Engineering capability has to accommodate emerging and spike needs. Necessary knowledge evolves. Consider reviewing staff knowledge, experience, and capabilities in anticipation of evolving needs; maintaining knowledge of means to augment skills quickly; monitoring degree and breadth of assimilated knowledge.
- **Process appropriate for needs** (C4, U3, R3, V3): The evolving environments, both externally and internally, will create needs for modifying the security engineering processes, Concept of Operations, and Operational Concept, with immediate full team awareness. Consider designated responsibilities for timely modification of ConOpsCon documentation, for full team indoctrination of changes in both external and internal collaborative partners, for accomplishing comprehensive knowledge assimilation, and for sustaining adequacy of incident response strategies.

Reactive correction

Identify what will impair/obstruct engineering agility that will benefit from embedded detection mechanisms. The distinguishing feature is a dysfunction or inadequacy during attempted action. Strategies are generally those that require a recovery from malfunction, recovery from unacceptable side effects, and inability to accomplish a necessary activity.

- **Insufficient awareness** (C1, C2, V2, V3): Effectiveness of engineering activity can be impaired by attacker innovation, undesirable complex system emergent behavior, dysfunctional peer and community behavior, and inadequate incident response capability. Consider proactive monitoring and systemic means for detecting these situations.
- **Ineffective actions** (C1, R2, V3): Innovative attack methods don’t have experience-proven protective measures. Addressing agile adversaries effectively requires agility in engineering actions and options. New protective measures don’t have experience-proven incident-response capability. All are trial and error risks. Consider real-time collaborative team evaluations; an agile architecture for security engineering processes and for security systems that enables rapid experimentation and recovery from ineffective approaches.
- **Wrong decisions** (C4, U1, V1): Collaborative partners can fail to be there when needed. Security cost vs. value decisions are infrequent, but the environment is not static. Attack and response criticality evaluations occur with available information at decision time; but information changes during attack. Consider continuous monitoring and evaluation of collaborative partners for cost benefit ratio; reevaluating security cost vs. value decisions fre-

quently; and an open-ended ability for continuous evaluation of criticality as the environment evolves.

Reactive accommodation of variation

Identify engineering process variables that will need accommodation. The distinguishing feature is predictable variance range but uncertain time of variance occurrence. Strategies are generally those that address variances in resource availability, resource performance, and resource interactions.

- **Effective actions and options (all S-CURVE elements):** There should be a roster of ready-to-use response actions with sufficient options and variations to fit all likely S-CURVE needs. Effectiveness of actions and options are generally variable, as they may not fit a new need optimally, and personnel loading and urgency of response may compromise application effectiveness. Consider procedures for attentive updating of actions and options with lessons learned after every event.
- **Effective evaluations** (all S-CURVE elements): There is an after-action tendency to evaluate the effectiveness of responses to any S-CURVE element on a pass-fail basis. Evaluations that go further can vary in the degree of objective retrospective questioning. Consider having uninvolved third party review and adjudication of evaluations, as ineffective evaluations can haunt.

Reactive expansion and contraction

Identify "quantity based" elastic-capacity range needs on resources/output/activity/other. The distinguishing feature is capacity scalability. Strategies are generally those that can be satisfied with planned capacity bounds, as well as those that have indeterminate and unbounded capacity needs.

- **Capacity for necessary simultaneous activities** (C1, C4, R1): During engineering there are often conflicts between desired feature development, known vulnerability elimination, and design of emergency attack-mitigation. Consider daily (or as needed) responsible decision-maker involvement in reevaluating or affirming the immediate activity priorities; and scalable resource availability.

Reactive reconfiguration

Identify types of resource relationship configurations that will need to change during engineering. The distinguishing feature is the configuration and employment of available resources. Strategies are generally those that that may have to reconfigure existing resource relationships during engineering.

- **Reusable actions** (all S-CURVE elements): All S-CURVE elements have response actions associated variably with awareness, analysis, evaluation, and/or execution. Granular reusable actions may need to be configured into a group appropriate for addressing a particular response need. Consider avoiding or minimizing sequence dependency in action design. The time sequence of granular actions in a group should be reconfigurable as appropriate for the need.

- **Personnel involved in an activity** (all S-CURVE elements): All S-CURVE elements have personnel activities associated variably with awareness, analysis, evaluation, and/or execution. Consider satisfying both need and value in being able to reconfigure the personnel involved in activities for including newly needed skills and for bench depth.

Security as a Functional Requirement

System security engineering is a specialty practice and security engineers traditionally provide comments on system design after its completion, resulting at times in security add-ons. Stakeholders often perceive security as a non-functional cost, or as a constraint on system functionality, rather than as an enabler or contributor to functionality. System designs often include security that *adequately* addresses compliance requirements and *sufficiently* protects stakeholders from liability, without truly making the system operationally secure. Integrating security in the system design is more likely to happen when security is part of functional requirements driving the design.

A *functional requirement* for a system is the specification of a required action or activity. The primary functional goal of any system is to *provide value-delivery*; i.e., produce needed results or desired results. Parts of a system's functional requirements are 1) the system shall provide value-delivery under nominal conditions and 2) the system shall provide value-delivery under adverse conditions. The latter acknowledges the system may experience a disturbance that induces stress to the system that may affect its ability to deliver value; i.e., there is a functional need not only for value-delivery but also to *sustain value-delivery*. This implies that derived requirements in reliability, sustainability, survivability, resistance, resilience, agility, safety, and *security* that contribute to sustaining value-delivery are themselves functional requirements.

Concluding Remarks

The purpose of this paper is to provide an initial conceptual understanding of the problem space facing security in the future of systems engineering, and to derive general security engineering strategies necessary to deal with that problem space. The intent is to provide an initial foundation for subsequent work on tactical approaches for implementing the strategies, and to instigate further research in both the problem space and solution space for agile security.

Summarizing Methods and Results

This paper characterized the general nature of the problem space by employing the CURVE framework that has proved useful in both analyzing and designing system engineering process that are or need to be agile (Dove, Schindel 2019). That framework is a heuristic to prompt thinking in five areas (outlined previously), and produces conclusions about the nature of the problem space that are or should be attended to in solution design. If the CURVE framework is used to analyze the nature of an existing process, it reveals the why behind the what. If used, as done in this paper, to provide grounding for a subsequent solution design with needs dictated by a host environment, it can frame the nature of engineering strategies needed for compatibility with the environment.

The principle objective for the CURVE framework is to characterize the security engineering environment with elements that would consequently need strategy attention. After a number of ad-hoc starts it became clear that structured guidance was needed on the nature of relevant security engineering CURVE elements. To obtain that it was decided to look at security engineering as a systems engineering activity that was simply focused on security aspects, and let a general CURVE for systems engineering structure the nature of the security engineering CURVE. A general systems engineering CURVE had previously been developed from analysis in the INCOSE Agile Systems Engineering Life Cycle Model (ASELCM) project, which first appeared in a 2019 INCOSE Webinar that covered various aspects of the ASELCM project (Dove 2019, p. 10).

The general systems engineering CURVE profile then provided useful structure to the security engineering CURVE profile shown in Table 1. Security engineering is system engineering, albeit often practiced separately from SoI engineering and SE standardized practices.

Delineating general security engineering strategies also took guidance from the ASELCM project findings, with minor security-focused rewording of general systems engineering response strategies (Dove, Schindel 2019).

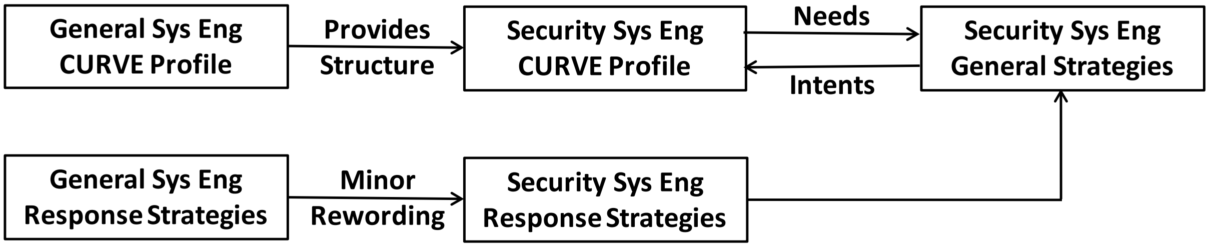

General Sys Eng CURVE Profile
Provides Structure
Security Sys Eng CURVE Profile
Needs
Intents
Security Sys Eng General Strategies
General Sys Eng Response Strategies
Minor Rewording
Security Sys Eng Response Strategies

Figure 1. Deriving security engineering general strategies from general systems engineering response strategies, and relating the intent of security strategies to security CURVE needs.

This paper is intended to provide an initial foundation of thought for system security strategies of particular relevance to *contextually aware agile security*. It does not attempt a comprehensive treatise on all future considerations; but does attempt to be sufficiently general in approach to provide a fairly inclusive strategy foundation.

Key takeaways:

- Agile security is necessary to contend with agile attack.
- Knowledge of the problem space drives strategy for the solution space.
- Vigilant awareness of internal and external process and product environments is essential.
- Systemic monitoring of behavior and performance of both security process and product can identify problems early.
- Systems engineering benefits from integrated security engineering.
- System security is a functional requirement.
- Security engineering is continuous throughout the system life cycle.
- Knowledge relevant to all stakeholders needs effective assimilation.
- Reusability of modular security actions should be facilitated.
- Social interaction and collaboration effectiveness needs strategy attention.
- Adequacy of incident response is constrained and enabled by the adequacy of continuous engineering.

The future of systems engineering and security is being shaped by an evolving environment. Status quo in security engineering is sustained by maintaining solution concepts in a problem space they no longer fit. This paper has proposed an initial foundation for a solution approach that evolves at the rate of the problem space.

Next Steps

Our current focus in the security topic within the FuSE initiative is on foundation development – general considerations that should shape the breadth and depth of necessary future solution strategies. This paper and a companion paper (Dove, Willett 2020) have offered two initial foundations. This paper addresses strategy for security system engineering as a process, the other addresses strategy for engineered system security as an operational product. Neither are considered sufficiently broad to span all that should be considered at the foundational level; but are offered as inspirational models for identifying additional work at the foundation level.

FuSE is a multi-organization collaborative initiative. Next steps for the security topic are active collaboration on foundational work: identifying additional foundational areas to pursue, establishing project participants in those different areas, and beginning those projects. Some of this is likely in process at this paper's publication date; but more is needed. To be a part of this activity contact the lead author of this paper.

References

Capgemini Research Institute 2019. Reinventing Cybersecurity with Artificial Intelligence - The new frontier in digital security. Report 20190711 V06. <www.capgemini.com/wp-content/uploads/2019/07/AI-in-Cybersecurity_Report_20190711_V06.pdf>

Dove, R., R. LaBarge 2014. Fundamentals of Agile Systems Engineering – Part 1. International Council on Systems Engineering, International Symposium, Las Vegas, NV, USA, 30Jun-3Jul. Updated version at <www.parshift.com/s/140630IS14-AgileSystemsEngineering-Part1.pdf>

Dove, R., W. Schindel 2019. Agile Systems Engineering Life Cycle Model for Mixed Discipline Engineering. Proceedings International Symposium. International Council on Systems Engineering. Orlando, FL, USA, July 20-25. <www.parshift.com/s/ASELCM-05Findings.pdf>

Dove, R. 2019. Agile SE Processes 202: Mixed Discipline Continuous Integration. International Council on Systems Engineering. INCOSE Webinar 131, 18-September. Slides at: <www.parshift.com/s/AgileSE-202.pdf>

Dove, R., K.D. Willett. 2020. Techno-Social Contracts for Security Orchestration in the Future of Systems Engineering. International Council on Systems Engineering, International Symposium, Cape Town, South Africa. July 18-23.

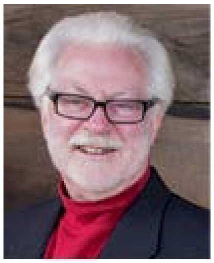
INCOSE. nd. The Future of Systems Engineering. An INCOSE initiative with charter at: <www.incose.org/about-systems-engineering/fuse> accessed 2/28/2020

Stupp, C. 2019. Fraudsters Used AI to Mimic CEO's Voice in Unusual Cybercrime Case. Wall Street Journal. August 30. <www.wsj.com/articles/fraudsters-use-ai-to-mimic-ceos-voice-in-unusual-cybercrime-case-11567157402>

Wallden, P., E. Kashefi. 2019. Cyber Security in the Quantum Era. Communications of the ACM, Vol. 62, No. 4. Pp 120-129. April. <https://cacm.acm.org/magazines/2019/4/235578-cyber-security-in-the-quantum-era/fulltext>

Willett, K.D. 2020. Systems Engineering the Conditions of the Possibility (Towards Systems Engineering v2.0). International Council on Systems Engineering, International Symposium, Cape Town, South Africa. July 18-23.

Biography



Rick Dove is CEO of Paradigm Shift International, specializing in agile systems and security research, engineering, and project management; and an adjunct professor at Stevens Institute of Technology teaching graduate courses in agile and self-organizing systems. He chairs the INCOSE working groups for Agile Systems and Systems Engineering, and for Systems Security Engineering. He is an INCOSE Fellow, and author of *Response Ability, the Language, Structure, and Culture of the Agile Enterprise*.

Dr. Keith D. Willett is a Data Scientist / Enterprise Security Architect for the U.S. Department of Defense with focus on future technologies, science of security, and systems security engineering. He is co-chair of the INCOSE Systems Security Engineering WG and active in working groups for Agile Systems and Systems Engineering, System Resilience, and System Science. He is the author of *Information Assurance Architecture* and many papers on the future of systems engineering and security.