MTR170001
MITRE TECHNICAL REPORT

MITRE

Cyber Resiliency Design Principles

Selective Use Throughout the Lifecycle and in Conjunction with Related Disciplines

Deborah Bodeau
Richard Graubart
January 2017

Dept. No.: J83C
Project No.: 03177M01-CA

This technical data was produced for the U. S. Government under Contract No. FA8702-17-C-0001, and is subject to the Rights in Technical Data-Noncommercial Items Clause DFARS 252.227-7013 (JUN 2013)

Bedford, MA

Abstract

Cyber resiliency is increasingly an explicit concern for systems, missions, and programs. Therefore, systems engineers and architects seek ways to apply cyber resiliency concepts and to integrate resilience-enhancing technologies into architectures and designs. This paper presents a representative set of cyber resiliency design principles and describes factors to use in selecting a set appropriate to a given system, program, or system-of-systems. These cyber resiliency design principles can be used, in varying ways and to different degrees, throughout the system lifecycle, and in conjunction with design principles from related disciplines, including security, resilience engineering, survivability, and evolvability.

Acknowledgments

The authors gratefully acknowledge the work of those who have defined and applied cyber resiliency design principles in a variety of situations, in particular Kate Arndt, Ken Cox, Harriet Goldman, Bill Heinbockel, Ellen Laderman, Rosalie McQuaid, Linda Morrison, Jeff Picciotto, and Mindy Rudell. In addition, the authors are grateful for the review and improvements provided by Shawn Fagan, Harriet Goldman, Bill Heinbockel, Ellen Laderman, and Linda Morrison.

Table of Contents

List of Figures

List of Tables

1 Introduction

Cyber resiliency is *the ability to anticipate, withstand, recover from, and adapt to adverse conditions, stresses, attacks, or compromises on cyber resources.* Cyber resiliency (or system resiliency[1], when the definition explicitly includes cyber attacks among the forms of adversity to which a system must be resilient) is an emergent[2] property of a system or a system-of-systems. This document presents a representative set of design principles for cyber resiliency, which can be applied in a variety of settings. In particular, these design principles can be used to inform activities and processes which are part of systems security engineering (SSE), as defined by NIST SP 800-160 [1].

This document is intended for systems engineers and architects with a working knowledge of cyber resiliency concepts and technologies, who are seeking to apply those concepts and technologies by (1) identifying the corresponding cyber resiliency design principles that apply to a given system and environment; (2) aligning (and possibly combining) the applicable cyber resiliency design principles with design principles from other specialty systems engineering disciplines; and (3) analyzing how well a given design, implementation, or as-deployed system applies a given cyber resiliency design principle. This document builds on a body of existing work on cyber resiliency, including the definition and evolution of the Cyber Resiliency Engineering Framework (CREF) [2] [3], processes for cyber resiliency assessment [4] [5], alignment of cyber resiliency with the multi-tiered approach to risk management defined by the Joint Task Force Transformation Initiative [6] [7], the series of Secure & Resilient Cyber Architectures Invitationals [8] [9] [10], and application of cyber resiliency to systems and programs at a variety of stages. For more information on cyber resiliency documentation, concepts, and guidance, see [11].

Cyber resiliency is increasingly recognized as a necessary attribute of systems and missions, as awareness has increased of sophisticated and evolving cyber threats [12]. The concern for cyber resiliency is recent relative to the lifespans of many systems and acquisition programs. Thus, the need to improve cyber resiliency of existing systems, systems-of-systems (SoS), and enterprise architectures (EAs) is a significant and growing concern. The cyber resiliency design principles described in this paper can be applied in different ways at multiple stages in the system development lifecycle (SDLC) or the acquisition lifecycle (ALC), including the operations and maintenance (O&M) stage, and can be used in a wide variety of system development models, including agile and spiral development.

The Need for Cyber Resiliency Design Principles

As a systems engineering area, cyber resiliency is related to other specialty disciplines, including security, resilience, survivability, and evolvability. However, cyber resiliency assumes an advanced cyber threat – persistent, stealthy, strategic, evolving, capable of discovering (and sometimes even creating) new vulnerabilities and developing tactics, techniques, and procedures (TTPs) to exploit those vulnerabilities in unforeseen ways. In addition, cyber resiliency is motivated by mission assurance, the overarching goal of which is to ensure that mission objectives can be achieved, "fighting through" attacks by intelligent, sophisticated, and strongly motivated adversaries. Because other disciplines do not make the same threat assumptions, their design principles cannot suffice to achieve cyber resiliency.

[1] See the May 2016 Second Public Draft of NIST SP 800-160 [20]. Note that NIST now plans to publish several of the Appendices to that draft – including the resiliency appendix – as separate publications.

[2] An emergent property is a property of a complex system which arises from interactions among the entities that make up that system. An emergent property can be accidental, but it can also be the result of engineering decisions. Examples of emergent properties that are intended by engineering processes include security [126], safety [127], and resilience ([1], p. 8). NIST SP 800-160 observes that "Emergent properties are typically qualitative in nature, are subjective in their nature and assessment, and require consensus agreement based on evidentiary analysis and reasoning." ([1], p. 9)

This introductory section provides background on design principles, an overview of this document, and notes on terminology.

1.1 Design Principles

In this document, the phrase *"design principles"* refers to distillations of experience designing, implementing, integrating, and upgrading systems that systems engineers and architects can use to guide design decisions and analysis. A design principle typically takes the form of a terse statement or a phrase identifying a key concept, accompanied by one or more statements that describe how that concept applies to system design (where "system" is construed broadly to include operational processes and procedures, and may also include development and maintenance environments).

Design principles are typically defined by specialty engineering disciplines. Figure 1 illustrates design principles from the specialty disciplines of Security ([1], Appendix F), Resilience Engineering [13], Survivability [14], and Evolvability [15].[3] The figure illustrates the fact that different specialty disciplines often share some design principles. For example, *Redundancy* is identified for Resilience Engineering, Survivability, and Evolvability; *Modularity and Layering* is a Security design principle, while *Layered Defense* is a Resilience Engineering design principle. However, the meanings of these apparently common design principles cannot be assumed to be identical; a design principle for a specialty discipline carries with it the assumptions, system and risk models, and priorities specific to that discipline. Thus, the relationship between apparently identical or similar design principles from different disciplines can be characterized in terms of *alignment*: Engineers from the specialty disciplines can combine such design principles into a system- or program-specific design principle, providing amplifying discussion to clarify what the design principle means in the context of the system or program, its mission requirements and operational environment, and the risks it can serve to mitigate. Alternately or in addition, systems engineers can develop questions to be answered by analysis of a system design, or via analysis and testing of an as-built or as-deployed system, and define metrics or other evidence to support the analysis.

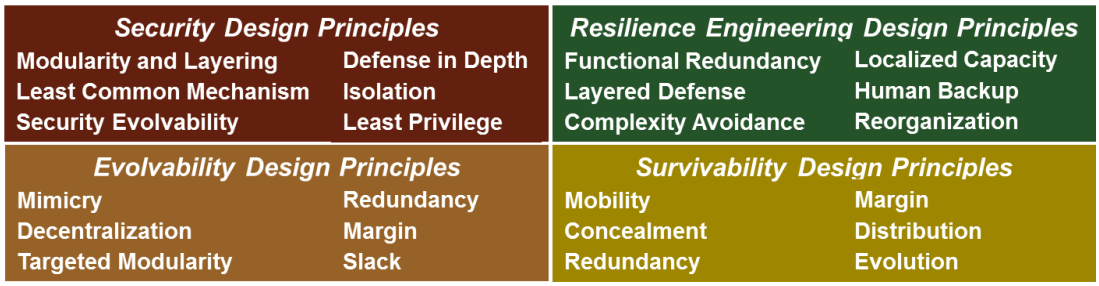

Figure 1. Representative Examples of Design Principles from Different Specialty Disciplines

The presence of "design" in the phrase "design principle" might suggest that the usefulness of design principles is limited to the early stages in the SDLC. However, some design principles are relevant to the design (or redesign) of processes, either for making more effective use of systems as those systems are being implemented, or during O&M. Early in the lifecycle, statements of design principles can be incorporated into a Security Plan and/or contractual requirements [8].[4] Design documentation then can include explanations of how the design applies or is consistent with the principles. A design principle can guide the selection, de-selection, or tailoring of requirements; the allocation of requirements to specific location(s) in an architecture; the choice of specific technical solutions or of how such solutions are implemented or integrated; and decisions about how to define operational processes and procedures consistent with an overall concept of operations (CONOPS). Later in the lifecycle, a design principle can

[3] See Appendix C for more details on design principles for these specialty disciplines, as well as safety engineering.
[4] Contractual requirements related to design principles typically appear in a Statement of Work (SOW), rather than in a Functional Requirements Document (FRD).

guide the selection, de-selection, or tailoring of recommended changes to the system (including changes in how it is used).

In the six-step process defined by the Risk Management Framework (RMF, [16] [17] [18]), requirements reflect the functional decomposition and allocation of security controls to the system architecture. A design principle is not a functional requirement, but it can be used to guide the selection, tailoring, or de-selection of security controls.[5] A design principle can also be used to guide the decomposition and allocation of security controls, as well as guiding implementation decisions.

1.2 Overview of This Document

Some design principles can be derived directly from the Cyber Resiliency Engineering Framework (CREF), which is described in Appendix A. Since cyber resiliency is a relatively new area, no single set of design principles has achieved consensus (as has been achieved, for example, with respect to security). However, MITRE's experience in articulating design principles for specific programs or systems, at different points in the lifecycle and for different types of systems, has demonstrated that a meaningful set of design principles needs to include statements that package one or more objectives and techniques together. In addition, MITRE has brought together a community of practice at the series of Secure & Resilient Cyber Architectures Invitationals [8] [9], where further experiences have been shared. Based on experience applying cyber resiliency, a representative set of cyber resiliency design principles has been developed. This set is presented in Section 2.

For any given system, system-of-systems, or program, a set of cyber resiliency design principles can be selected (and tailored, to be expressed in terms more meaningful in the context of the architecture and CONOPS for missions and for system operations) using those presented in this paper as a starting point. ***Meaningful design principles provide the basis for engineering analysis and (where possible) metrics, to speak directly to the concerns of stakeholders.*** Section 2 provides examples of specific restatements and possible metrics, and Appendix B provides alternative statements of cyber resiliency design principles. Note that the metrics identified in Section 2 only address how well (e.g., how completely, how consistently) each principle is applied; metrics and other form of evidence for *how effective* an application of a design principle is, given a threat model, will be the topic of a future report.

To be *useful*, the set of design principles should not be too large; experience suggests a set on the order of a dozen.[6] Thus, the set presented in Section 2 is a starting point, with the expectation that some will be deemed inapplicable. ***No cyber resiliency design principle is universally applicable.*** Whether a principle is relevant to a given situation depends on a variety of factors. When a principle is relevant, the statements describing how it applies will be tailored based on those factors. Section 3 describes factors to consider. Among those factors is the relationship among design principles. Even in a relatively mature discipline such as security, established design principles cannot all be satisfied simultaneously. Cyber resiliency design principles must be used in conjunction with those from related disciplines – security, resilience in general, survivability, or evolvability. Relationships discussed in Section 3 are explored in more detail in Appendix C.

Three appendices are also provided. Appendix A provides background on cyber resiliency. Appendix B provides background on sources of potential cyber resiliency design principles, and presents some additions or alternatives to the design principles presented in Section 2. Appendix C presents mappings from design principles for related disciplines to the cyber resiliency design principles. Note that the

[5] For a mapping of the security controls in NIST SP 800-53R4 to the cyber resiliency techniques defined in the Cyber Resiliency Engineering Framework, see Appendix H of the Second Public Draft of NIST SP 800-160 [20]. An earlier mapping can be found in [6].

[6] The set will typically include a mixture of strategic and structural design principles. See Section 2 for an explanation of these terms. Note that, as discussed in [3], the use of some cyber resiliency techniques can interfere or conflict with the use of others. A similar observation can be made about design principles.

details in Appendices B and C are intended for systems engineers seeking to align design principles from different specialty disciplines, rather than for the general reader.

Two significant topics are outside the scope of this document: metrics or other evidence for evaluating the relative effectiveness of applications of design principles, and methods for performing cost-benefit analyses. MITRE research on metrics and other evidence is underway, and will be the subject of a subsequent report. The economics of cyber resiliency is the focus of a research workshop, proceedings of which are forthcoming [19].

1.3 Notes on Terminology

Systems engineering relies on well-defined terms, but specialty disciplines differ in nuanced connotations of the same term. In addition, the cyber resiliency engineering discipline continues to evolve. Therefore, the following notes are intended to aid in understanding terms as used in this document.

"Resilience" or "resiliency"? "Resilience" and "resiliency" are alternative spellings, with "resilience" being more common. The term "cyber resiliency" was chosen for MITRE's Cyber Resiliency Engineering Framework (CREF, [2] [4] [3]), to avoid creating the impression that cyber resiliency engineering was simply resilience engineering with "cyber" as a modifier. Cyber resiliency engineering draws upon resilience engineering, as demonstrated by the CREF goals, but also from cybersecurity and survivability; it explicitly addresses advanced cyber threats; and it is intended to serve as a bridge between the disciplines of mission assurance, cybersecurity, and resilience engineering. The definition of cyber resiliency in this document is consistent with that of system resiliency in the 2nd Public Draft of NIST SP 800-160 [20] and with the definition of operational resilience in DoDI 8500.01 [21].

Since the publication of the CREF, the term "cyber resilience" has gained use, but is being used to refer to organizational resilience against cyber threats, with a strong emphasis on effective implementation of good cybersecurity practices (e.g., using the NIST Cybersecurity Framework [22]) and COOP. For example, the DHS Cyber Resilience Review (CRR, [23]), which is based on the SEI CERT Resilience Management Model (RMM, [24]), focuses on good practices against conventional adversaries. Discussions of "cyber resilience" focus on improved risk governance (e.g., making cyber risk part of enterprise risk); improved cyber hygiene to include incident response procedures and ongoing monitoring; and threat information sharing (see, for example, [25] [26] [27]). These aspects of Governance and Operations are all vitally important to an organization's cyber preparedness strategy [28]. However, discussions of "cyber resilience" generally omit the Architecture and Engineering aspect, which is the focus of the CREF and of the design principles discussed in this paper.

"Conventional" or "advanced"? The phrases "conventional cybersecurity" and "conventional cyber threats" are increasingly used to establish a basis for discussing cyber resiliency. While some uses of "conventional" are pejorative, that is not the intended use in this document. Rather, the intent is to make a connection with the concept of "the conventional threat," as translated into the cyber domain. For much of the 20th century, the phrases "conventional threats," "conventional weapons," and "conventional warfare" were widely used, both to refer to situations covered by international treaty conventions and to set a context for discussion of "unconventional" (or "non-conventional") situations. In the cyber domain, "conventional threats" are those addressed by established standards of good practice, and particularly by the baselines in NIST SP 800-53R4 [29].

Terminology for non-conventional threats in the cyber domain continues to evolve, but typically involves "advanced" to express a degree of sophistication in TTPs, particularly sophistication in the malware the adversary can develop. The term "advanced persistent threat" (APT) is commonly used, particularly to indicate an adversary (or class of adversaries) able to overcome perimeter defenses, access control and privilege management mechanisms, and intrusion detection to maintain a long-term presence on targeted systems. While some sources restrict the term APT to adversaries seeking to exfiltrate data, the term is

increasingly used to include advanced adversaries seeking disruption and undermining of mission effectiveness. For clarity and for consistency with the Defense Science Board (DSB) report [12], this document uses the phrase "advanced cyber threat".

"Structural" or "strategic"? As discussed by Ricci et al. [15], design principles can be characterized as (i) *strategic* to be applied throughout the systems engineering process, guiding the direction of engineering analyses and possibly also programmatic decisions (e.g., decisions about supply chain risk management (SCRM)), or (ii) *structural* – directly affecting the architecture and design.

A *strategic* design principle expresses aspects of an organization's risk management strategy. A strategic design principle should relate to how the organization frames risk – what threat assumptions it makes, which cyber resiliency goals are highest priority, and what constraints apply to the selection and implementation of architectural and design decisions, security controls, and products or technologies. A strategic design principle implies the need for specific *analytic techniques or methodologies*. Strategic design principles can drive the selection of structural design principles. Strategic design principles are typically not restated.

For a *structural* design principle, more specific restatements can be made, focusing on different approaches to applying the principle[7] or expressed in terms of the mission or enterprise architecture in which the principle is to be applied. Analytic *questions* can be posed regarding the layers or locations in the architecture to which a structural design principle is applied; *metrics or other evidence* can be defined to support analysis of how well the principle is applied.

What are the distinctions between components, system elements, assets, and resources? While a design principle can guide design decisions, it can also guide analysis of how – and how effectively – a given design or as-deployed system applies the principle. Such analysis can involve more specific restatements of the principle as relevant to the type of system, and metrics – typically in the form of counts or percentages – of the objects to which the principle applies. Thus, precision in terminology matters, whether in the initial statement or in restatements or metrics definitions; otherwise, the analysis will be misdirected.

This paper uses terminology from NIST SP 800-160 [1], the NIST glossary [30], and CNSSI No. 4009 [31] where possible. Specifically, the terms *component*, *system element*, and *asset* are used. A *system element* is described conceptually, in terms of what it does (e.g., a database management system, an identity validation subsystem) or how it is implemented, as a person, process, physical object, or technological object – i.e., as a hardware, software, or communications component. System elements are implemented or instantiated as hardware components, software components, data stores, communications channels, input channels, people, and processes.

While NIST SP 800-160 redirects its definition to that of system element, the term *component* generally refers to a technical or physical object rather than including people and procedures; the connotation of the term is of being discrete and replaceable. The terms system element and component both allow for recursion: a system element can be made up of other system elements; a component can have sub-components.

An *asset* is something of value, where the value may be intrinsic (e.g., a financial asset, a physical asset) or may be derived from its uses (e.g., a mission asset, with the associated attributed of criticality) or from other characteristics (e.g., an information asset, with an associated confidentiality impact level). Thus, an asset can take the form of a component or system element. An *information asset* is information of value,

[7] For example, in Section 2.2.5 below on the *Plan and manage diversity* design principle, one restatement focuses on geographic diversity and another on diversity-in-depth.

typically in the form of a data store (e.g., file, database). In addition, a service (e.g., a domain name service (DNS) or identity and access management (IdAM) service) or capability can be an asset.

In addition, this paper (like NIST SP 800-160, the NIST glossary, and CNSSI No. 4009) uses the term *resource* (or *system resource*). While these publications do not provide a definition[8], the term in the context of a cyber resiliency objective, technique, or design principle means a cyber resource. Cyber resources are defined as "separately manageable resources in cyberspace, including information in electronic form, as well as information systems, systems-of-systems, network infrastructures, shared services, and devices." [9] Thus, a cyber resource can be a system element, a service or capability offered by a system element, or information viewed in terms of how it can be used (e.g., processing, communications, storage, information in usable form) to achieve mission objectives. Where the key attribute of an asset is its value, the key attribute of a resource is its utility (which may be expressed in terms of capacity, quality, or readiness). Thus, a resource is an asset viewed through the lens of its intended or potential uses [32].

What does "trusted" mean? The term "trusted" has been used in the context of security for decades. In a larger context, trustworthiness involves meeting critical requirements [1], whether security-related or other, and can be identified with dependability [33]. As discussed in Appendix A.4, multiple dimensions of trustworthiness can be identified. These include safety, security, privacy, resilience, and reliability [34]. Within each dimension, multiple aspects can be identified. For example, in the context of security, key properties are confidentiality, integrity, and availability; historically, a trusted subject has been one that is capable of violating a confidentiality policy – but (based on assurance evidence) is assumed not to violate that policy [35].

In the context of cyber resiliency, "trust" refers to confidence that critical requirements will be met, critical properties will be preserved, or critical attributes will be assured. That is, "trust" is defined in terms of criticality, as in NIST SP 800-160, and thus is highly contextual. Because trustworthiness has multiple dimensions, and each dimension can have different aspects, critical requirements need to be weighted, and trade-offs among them made. Therefore, "trust" or "trustworthiness" is not a binary property.

[8] CNSSI No. 4009 defines "information resources" as "information and related resources, such as personnel, equipment, funds, and information technology" but this definition is too high-level to be of use in this paper. NIST SP 800-160 uses the term "resources" in the same way, but also uses it in such contexts as "human resources," "infrastructure resources," "collaboration resources," "system resources, services, and capabilities," and "methods, processes, and tools" to perform maintenance.

2 Representative Cyber Resiliency Design Principles

This section presents a representative set[9] of cyber resiliency design principles, which can be applied in different ways at multiple stages in the lifecycle, including the operations and maintenance stage, and which can be used in a wide variety of system development models, including agile and spiral development. The principles identified in this section are intended to serve as a starting point for systems engineers and architects – for any given situation, only a subset[10] will be selected, and those will be tailored or re-expressed in terms more meaningful to the program, system, or system-of-systems to which they apply. Section 3 discusses selection factors.

Table 1 presents the representative set of cyber resiliency design principles. These principles were generalized from those developed as a result of performing cyber resiliency analyses, defining architectures, or identifying requirements for specific programs, systems, or systems-of-systems.[11] Some of the activities used as sources for this representative set were for existing systems and programs, while others were for new starts. Some of the design principles discussed below bundle together ideas from the CREF, while others are nearly identical to CREF-derived principles as defined in Appendix A. The cyber resiliency design principles are strongly informed by, and can be aligned with, design principles from other specialty disciplines. This is indicated by the color coding.

Table 1. Representative Cyber Resiliency Design Principles

Strategic Cyber Resiliency Design Principles			
Focus on common critical assets.		**Support agility and architect for adaptability.**	
Reduce attack surfaces.	**Assume compromised resources.**	**Expect adversaries to evolve.**	
Structural Cyber Resiliency Design Principles			
Limit the need for trust.	**Control visibility and use.**	**Contain and exclude behaviors.**	**Layer and partition defenses.**
Plan and manage diversity.	**Maintain redundancy.**	**Make resources location versatile.**	
Leverage health and status data.	**Maintain situational awareness.**	**Manage resources (risk) adaptively.**	
Maximize transience; minimize persistence.	**Determine ongoing trustworthiness.**	**Change or disrupt the attack surface.**	**Make unpredictability and deception user transparent.**
Key to Aligned Disciplines:			
Security	*Resilience Engineering & Survivability*	*Evolvability*	*Unique to Consideration of Advanced Cyber Threats*
Warning: For any given mission, system, or program, only a subset of these principles will be relevant selection must be based on a variety of considerations, including lifecycle stage, type of system, and relevant design principles from other disciplines. In addition, more specific restatements may prove more useful in guiding analysis and assessment.			

[9] While the set of principles defined in this section is drawn from multiple sources, it is intended to be representative rather than exhaustive. See Appendix B for more detail on sources, as well as alternative and additional statements.

[10] As noted in Section 1.1, some principles cannot simultaneously be satisfied easily, if at all. For example, strategies to reduce the attack surface typically conflict with those to provide agility. The inability to satisfy all design principles in a set is not unique to cyber resiliency; Benzel et al. discuss this for security [107].

[11] Program Managers increasingly recognize the need to improve the cyber resiliency of existing or partially-developed systems and SoS. While such systems may lack requirements specific to cyber resiliency, their design and implementation typically will include mechanisms for security, survivability, continuity of operations planning (COOP), and reliability, maintainability, and availability (RMA), which can be repurposed or modified in low- or no-cost ways to support cyber resiliency. Therefore, a cyber resiliency analysis (CRA, [5]) can be used effectively at any point in the lifecycle to identify potential ways to improve cyber resiliency. One product of a CRA can be a set of cyber resiliency design principles that are consistent with the program's lifecycle stage and other constraining factors.

As discussed by Ricci et al. [15], design principles can be characterized as (i) *strategic* to be applied throughout the systems engineering process, guiding the direction of engineering analyses, or (ii) *structural* – directly affecting the architecture and design. Both strategic and structural cyber resiliency design principles can be reflected in cybersecurity-related artifacts (see Appendix G of [18]). Strategic cyber resiliency design principles are discussed in Section 2.1, while structural principles are discussed in Section 2.2.

Cyber resiliency design principles support different cyber resiliency objectives and call for the use of different cyber resiliency techniques. As indicated in Section 2.3, the representative set of cyber resiliency design principles covers the set of cyber resiliency objectives and techniques. The strategic design principles emphasize the Transform and Re-Architect[12] objectives, while these two objectives are lightly represented in the structural design principles.

2.1 Strategic Design Principles for Cyber Resiliency

The strategic design principles inform engineering analyses and risk analyses throughout the SDLC, and highlight different structural design principles, cyber resiliency techniques, and approaches to applying those techniques. Of the five strategic cyber resiliency design principles, the first two (*Focus on common critical assets* and *Support agility and architect for adaptability*) draw strongly from Resilience Engineering and Survivability, but emphasize consideration of malicious cyber activities. The next two (*Reduce attack surfaces* and *Assume compromised resources*) draw from security, but become more important in light of advanced cyber adversaries. The remaining principle (*Expect adversaries to evolve*) is unique to cyber resiliency, due to its focus on advanced adversaries.[13]

Strategic design principles are driven by an organization's risk management strategy – in particular, by its risk framing. Risk framing includes such considerations as assumptions about the threat the organization needs to be prepared for, the constraints on risk management decision making (including which risk response alternatives are irrelevant), and organizational priorities and trade-offs. From the standpoint of cyber resiliency, one way to express priorities is in terms of which cyber resiliency goals[14] are most important. Each strategic design principle supports achievement of one or more cyber resiliency objectives. Each strategic design principle can be used to identify analytic resources – methodologies, tools, frameworks, taxonomies, and datasets – which can be used in analysis of how (and how well) a given system, system-of-systems, or mission (as-deployed or as-designed) applies the principle. These relationships are indicated in Table 2.

An organization's risk management strategy is constrained by such factors as financial resources; legal, regulatory, and contractual requirements, as reflected in organizational policies and procedures; legacy investments; and organizational culture [36]. An organization can define an order of precedence for responding to identified risks, analogous to the safety order of precedence ([32]; see Appendix C.5), such as "harden, sensor, isolate, obfuscate." Together with the strategic design principles selected and tailored to a given program, mission, or system, such an order of precedence can guide the selection and application of structural design principles at different locations in an architecture.

[12] In the discussion in this section, cyber resiliency objectives and techniques are capitalized, while names of design principles are *italicized*.

[13] Both *Support agility and architect for adaptability* and *Expect adversaries to evolve* assume a changing environment. However, the former assumes "normal" changes, i.e., changes in the technical and operational environment, and thus can be closely aligned with design principles for Resilience Engineering and Evolvability. The latter assumes a changing threat environment, and thus can be aligned with security design principles for digital services (see Appendix C.1.3-4).

[14] Cyber resiliency goals, objectives, and techniques are defined in the Cyber Resiliency Engineering Framework (CREF). See Appendices A and B for more information.

Table 2. Strategic Cyber Resiliency Design Principles in Context

Strategic Cyber Resiliency Design Principle	Risk Framing Elements of Risk Management Strategy	Cyber Resiliency Objectives	Relevant Analytic Resources
Focus on common critical assets.	**Threat assumptions**: Conventional adversary. Advanced adversary seeking path of least resistance. **Risk response constraints**: Limited programmatic resources. **Risk response priorities**: Anticipate, Withstand, Recover	Understand Prevent / Avoid Continue Reconstitute Re-Architect	Criticality Analysis [37], Crown Jewels Analysis (CJA, [38]), Cyber Mission Impact Analysis (CMIA, [39]), Dagger [40], Cyber Failure Mode, Effect, and Criticality Analysis (Cyber FMECA, [41]), Functional Dependency Network Analysis (FDNA, [42]), Mission Information Risk Analysis (MIRA) using the Mission Assurance Analytics Platform (MAAP) [43], Mission Thread Analysis (MTA, [44]), Contingency Tabletop Exercise (TTX) [45]
Support agility and architect for adaptability.	**Threat assumptions**: None **Risk response constraints**: Missions to be supported, and mission needs, can change rapidly. **Risk response priorities**: Recover, Evolve.	Prepare Reconstitute Transform Re-Architect	Cyber Security Game (CSG, [46]) (using network topology model and CMIA process model), War gaming (e.g., Cyber SIMEX, cyber range)
Reduce attack surfaces.	**Threat assumptions**: Conventional adversary. Advanced adversary seeking path of least resistance. **Risk response constraints**: Limited operational resources to monitor and actively defend systems. **Risk response priorities**: Anticipate	Understand Prevent / Avoid Constrain Transform Re-Architect	Attack Surface Analysis (can be integrated with or informed by Security Code Review), SCRM Analysis, OPSEC Analysis
Assume compromised resources.	**Threat assumptions**: Advanced adversary. **Risk response constraints**: Ability to assure trustworthiness of system elements is limited. **Risk response priorities**: Anticipate, Withstand	Understand Prepare Continue Constrain Reconstitute Transform Re-Architect	Modeling & Simulation (M&S) e.g., CSG, AMICA; CMIA, FDNA, MTA, Cyber FMECA; War Gaming (e.g., Cyber SIMEX, cyber range); Red Teaming (Cyber TTX, Voice of the Offense, adversarial T&E); Voice of the Adversary (VoA)
Expect adversaries to evolve.	**Threat assumptions**: Advanced adversary. Adversary can change TTPs and goals unpredictably. **Risk response priorities**: Anticipate, Evolve	Understand Prepare Transform Re-Architect	Cyber Attack Lifecycle (CAL) or Cyber Kill Chain (CKC) models, ATT&CK [47], CAPEC [48], War Gaming and Red Teaming, Threat Information Sharing, VoA

Strategic design principles could be used in a programmatic Security Plan[15] as part of documenting the philosophy of protection and systems security engineering strategy. As noted above, the expectation that all these strategies can be applied simultaneously is unrealistic. For example, agility can change the determination of which assets are critical, and approaches to supporting agility can increase the attack surface. Thus, the Security Plan needs to identify the set of strategic design principles selected for and

[15] As described in [18], "The Security Plan provides an overview of the security requirements for the system, system boundary description, the system identification, common controls identification, security control selections, subsystems security documentation (as required), and external services security documentation (as required)." The selected and tailored cyber resiliency and security design principles could be included in, or serve as a structuring mechanism for, this overview.

tailored to the given system, SoS, or program, and to capture guidance on how to make trade-offs among the structural design principles they highlight.

In addition, strategic design principles – or more concrete statements drawn from the discussion of those principles – can be represented in a Statement of Work.[16]

2.1.1 Focus on Common Critical Assets

Limited organizational and programmatic resources need to be applied where they can provide the greatest benefit. This results in a strategy of focusing first on assets which are critical and common, then on those which are either critical or common.

A focus on *critical* assets – resources valued due to their importance to mission accomplishment – has long been central to contingency planning, continuity of operations planning [49], and operational resilience [50], as well as to safety analysis [41]. Criticality analysis is central to the development and implementation of a Program Protection Plan (PPP, [51]). Critical assets can be identified using a variety of mission-oriented analysis techniques, ranging from Business Impact Analysis (BIA, [49]) and Mission Impact Analysis (MIA, [39]) to Crown Jewels Analysis (CJA, [38]) and Functional Dependency Network Analysis (FDNA, [42]), as well as Mission Threat Analysis (MTA, [44]) and Mission Information Risk Analysis (MIRA, [43]). Failure Modes, Effects, and Criticality Analysis (FMECA) takes a safety-oriented approach.

Assets that are *common* to multiple missions or business functions are potential high-value targets for cyber attackers, either because those assets are critical or because their compromise increases the attackers' options for lateral motion or persistence.[17]

Once an asset has been identified as critical or common, further analysis involves

1. Identifying how the asset is used in different operational contexts (e.g., normal operations, abnormal operations, crisis or emergency operations, failover). An asset that is common to multiple missions may be critical to one mission in one context but not in a second, but critical to a second mission only in the second context.
2. Determining which properties or attributes make the asset critical (e.g., availability, correctness, non-observability) or high-value (e.g., providing access to a set of critical system elements, providing information which could be used in further malicious cyber activities), and what would constitute an acceptable (e.g., safe, secure) failure mode. Again, properties which are critical to one mission may be non-essential to another, and a failure mode which is acceptable from the standpoint of security may be unacceptable from the standpoint of safety.
3. Determining which strategies to use to ensure critical properties, taking into consideration the different usage contexts and potential malicious cyber activities. Examples of strategies for ensuring correctness and non-observability properties include disabling non-critical functionality, restoration to default / known-good settings, and selectively isolating or disabling data flows to or from components. Articulating trade-offs among critical properties and acceptable failure modes is central to effective risk management.

Based on the strategy or strategies that best fit a given type of asset, the most relevant structural design principles can be determined.

This design principle makes common infrastructures (e.g., networks), shared services (e.g., identity and access management services), and shared data repositories high priority for the application of selected

[16] Requirements in a Statement of Work related to design principles could take the form: "The contractor shall describe how the design, implementation, integration, and/or maintenance procedures …"
[17] See [47] for a taxonomy of post-exploit malicious cyber activities.

cyber resiliency techniques. It recognizes that risk mitigation resources are limited, and enables systems engineers to focus resources where they will have the greatest potential risk mitigation. This design principle also highlights the importance of analysis methods such as Cyber FMECA [41].

2.1.2 Support Agility and Architect for Adaptability

Not only does the threat landscape change as adversaries evolve, so do technologies and the ways in which individuals and organizations use them. This design principle is motivated by the Evolve cyber resiliency goal. It recognizes the need for both agility and adaptability as part of the risk management strategy, in response to the risk framing assumption that unforeseen changes will occur in the threat, technical, and operational environment through a system's lifespan.

The term "agility" is used in many ways. In Resilience Engineering, it means "effective response to opportunity and problem, within a mission" ([52], quoted in [53]). In that context, resilience supports agility, and counters brittleness.[18] In the context of cyber resiliency, *agility* is the property of a system or an infrastructure which can be reconfigured, in which resources can be reallocated, and in which components can be reused or repurposed, so that cyber defenders can define, select, and tailor cyber courses of action for a broad range of disruptions or malicious cyber activities (MCA). This strategy is consistent with the vision that the "infrastructure allows systems and missions to be reshaped nimbly to meet tactical goals or environment changes" [54]. Agility enables the system and operational processes to incorporate new technologies and/or adapt to changing adversary capabilities.

Adaptability is the property of an architecture, design, and implementation which can accommodate changes to the threat model, mission threads and systems, and technologies without major programmatic impacts. A variety of strategies for agility and adaptability have been defined. These include modularity and controlled interfaces, to support plug-and-play; externalization of rules and configuration data; and removal or disabling of unused components to reduce complexity. Application of this design principle early in the lifecycle can reduce sustainment costs and modernization efforts.

This design principle means that analyses of alternative architectures and designs need to look for sources of brittleness (e.g., reliance on a single operating system or communications channel; allowing single points of failure; reliance on proprietary interface standards; use of large and hard-to-analyze multi-function modules). Thus, analyses need to consider adaptive response, diversity, and redundancy, and the coordinated defense capabilities that enable cyber defenders to make effective use of these techniques. In addition, analyses need to consider where and how to use "cyber maneuver" or moving target defenses [55], as well as deception [56]. Finally, analyses need to consider where and how an architecture, design, or as-deployed system is bound to assumptions about the threat, operational, and technical environments.

2.1.3 Reduce Attack Surfaces

A large attack surface is difficult to defend, requiring ongoing effort to monitor, analyze, and respond to anomalies. Reducing attack surfaces reduces ongoing costs and makes the adversary concentrate efforts on a small set of locations, resources, or environments that can be more effectively monitored and defended.

At a minimum, the term "attack surface" refers to "accessible areas where weaknesses or deficiencies in information systems (including the hardware, software, and firmware components) provide opportunities for adversaries to exploit vulnerabilities." [29] "The attack surface is the system's exposure to reachable and exploitable vulnerabilities; in other words, any hardware, software, connection, data exchange, service, removable media, etc. that might expose the system to potential threat access." [57] While some uses of the term focus on externally exposed vulnerabilities, the assumption that an adversary will

[18] "A brittle system is one that is unable to adapt to unanticipated disturbances or disruptions and, consequently, breaks down." [95]

penetrate an organization's systems means that internal exposures – vulnerabilities which can be reached by lateral movement within a system or infrastructure – are also part of the attack surface.

Conceptually, the term can also cover aspects of the operational, development, and maintenance environments that an adversary can reach and that could contain vulnerabilities. In particular, a system's supply chain(s) can present additional attack surfaces. More broadly, a mission or an organization can be said to have an attack surface, which might include people and processes. To accommodate these broader interpretations of the term, the design principle refers to "attack surfaces" (plural).

This design principle is often used in conjunction with the *Focus on critical assets* principle.[19] Analysis of internal attack surfaces can reveal unplanned and unexpected paths to critical assets. It makes identification or discovery of attack surfaces a priority in design analyses[20], as well as analyses of development, configuration, and maintenance environments (e.g., by considering how using free and open source software (FOSS) or commercial off-the-shelf (COTS) products which cannot be tailored in those environments expands attack surfaces). It may be infeasible in some architectures (e.g., Internet of Things, bring-your-own-device) or procurement environments (e.g., limited supply chain), for which the *Assume compromised resources* principle is highly relevant.

As indicated in Table 3, several alternative strategies for reducing an attack surface can be identified. These strategies are expressed by different controls in NIST SP 800-53R4 [29], and apply different cyber resiliency techniques. (In Table 3, the **bolding** in the discussion of the control indicates how the control supports the strategy.) These strategies can be reflected by different structural principles. For example, design decisions related to the *Maximize transience* and *Change or disrupt the attack surface* structural principles can reduce the duration of exposure, while application of the *Limit the need for trust* principle can reduce exposure.

While the security controls in Table 3 focus on attack surfaces of and within a system, the strategies apply more broadly, to the attack surfaces of a mission or an organization. For example, Operations Security (OPSEC) can reduce the exposure of the mission or organization to adversary reconnaissance; other supply chain protections can reduce the exposure of key components to tampering.

Table 3. Strategies for Reducing an Attack Surface

Strategy	Representative Security Control Supporting the Strategy	Related Techniques
Reduce the extent ("area") of the attack surface	"Attack surface reduction includes, for example, applying the principle of least privilege, employing layered defenses, **applying the principle of least functionality (i.e., restricting ports, protocols, functions, and services), deprecating unsafe functions, and eliminating application programming interfaces (APIs) that are vulnerable to cyber attacks**." SA-15 (6) DEVELOPMENT PROCESS, STANDARDS, AND TOOLS | ATTACK SURFACE REDUCTION	Adaptive Response Realignment
Reduce the exposure ("aperture" or structural accessibility) of the attack surface	"Attack surface reduction includes, for example, **applying the principle of least privilege, employing layered defenses,** applying the principle of least functionality (i.e., restricting ports, protocols, functions, and services), deprecating unsafe functions, and eliminating application programming interfaces (APIs) that are vulnerable to cyber attacks." SA-15 (6) DEVELOPMENT PROCESS, STANDARDS, AND TOOLS | ATTACK SURFACE REDUCTION	Privilege Restriction Coordinated Defense

[19] The combination of focusing on critical assets and reducing the attack surface reflects abnegation: "A strategy of abnegation is founded on the presumption that critical systems should be supported by cyber capabilities that are no more extensive than required to perform their core mission." [133]

[20] For example, SA-11 (7) *DEVELOPER SECURITY TESTING | ATTACK SURFACE REVIEWS* calls for analysis of design and implementation changes.

Strategy	Representative Security Control Supporting the Strategy	Related Techniques
	"**Component isolation** reduces the attack surface of organizational information systems." SC-7 (20) BOUNDARY PROTECTION | DYNAMIC ISOLATION / SEGREGATION	Adaptive Response Segmentation / Isolation
Reduce the duration (temporal accessibility) of attack surface exposure	"This control mitigates risk from advanced persistent threats by significantly reducing the targeting capability of adversaries (i.e., window of opportunity and available attack surface) to initiate and complete cyber attacks." "This control mitigates risk from advanced persistent threats (APTs) by significantly reducing the targeting capability of adversaries (i.e., **window of opportunity** and available attack surface) to initiate and complete cyber attacks." SI-14 NON-PERSISTENCE	Non-Persistence

2.1.4 Assume Compromised Resources

Many system architectures treat many if not all resources as non-malicious. This assumption is particularly prevalent in cyber-physical systems (CPS) and Internet of Things (IoT) architectures [58]. However, systems and their components, ranging from chips to software modules to running services, can be compromised for extended periods without detection [12]. In fact, some compromises may never be detected.[21] Thus, the assumption that some system resources have been compromised is prudent. Note that while the assumption that some resources cannot be trusted is well-established from the standpoint of security (i.e., the compromised resources cannot be trusted to follow established security policies), the concept of trustworthiness is broader. By compromising a resource, an adversary can affect its reliability, ability to enforce privacy policies, or the safety of the larger system or environment of which the resource is a part [34] [59], or can use the resource in an attack on other systems.

This design principle implies the need for analysis of how the system architecture reduces the potential consequences of a successful compromise – in particular, the duration and degree of adversary-caused disruption, as well as the speed and extent of malware propagation. As indicated in Table 2, an increasing number of modeling and simulation techniques support analysis of the potential systemic consequences stemming from the compromise of a given resource or set of resources. Such analysis includes identifying different types or forms of systemic consequences (e.g., unreliable or unpredictable behavior of services, unreliable or unpredictable availability of capabilities, data of indeterminate quality), and linking these systemic consequences to mission consequences (e.g., mission failure, safety failure) or organizational consequences (e.g., loss of trust or reputation).

2.1.5 Expect Adversaries to Evolve

Advanced cyber adversaries invest time, effort, and intelligence-gathering to improve existing and develop new TTPs. Adversaries evolve in response to opportunities offered by new technologies or uses of technology, as well as to the knowledge they gain about defender TTPs. In (increasingly short) time, the tools developed by advanced adversaries become available to less sophisticated adversaries. Therefore, systems and missions need to be resilient in the face of unexpected attacks. This design principle therefore supports a risk management strategy which includes but goes beyond the common practice of searching for and seeking ways to remediate vulnerabilities (or classes of vulnerabilities); a system which has been hardened in the sense of remediating known vulnerabilities will remain exposed to evolving adversaries.

[21] A classic example of a compromise which would never have been known had not its creator chosen to disclose it is the bugging of the C compiler to create a backdoor in all Unix systems [136].

This design principle implies the need for analyses in which the adversary perspective is explicitly represented by intelligent actors who can play the role of an adaptive or evolving adversary. For implemented systems, such analyses are typically part of Red Teaming or War Gaming. Analyses can use threat intelligence or repositories of attack patterns (e.g., ATT&CK, CAPEC) to provide concrete examples, but care must be exercised not to be constrained by those examples. Earlier in the SDLC, Voice of the Adversary (VoA) is a design analysis technique in which one or more team members play the role of an adversary to critique alternatives by taking into consideration possible goals, behaviors, and cyber effects assuming varying degrees of system access or penetration. Such design analysis can use models or taxonomies of adversary behaviors (e.g., CAL or CKC models, CAPEC or ATT&CK classes), as well as languages or taxonomies of cyber effects (e.g., [60]).

This design principle also highlights the value of the Deception and Diversity cyber resiliency techniques. Deception can cause adversaries to reveal their TTPs prematurely from the perspective of their cyber campaign plans, enabling defenders to develop countermeasures or defensive TTPs. Diversity can force an adversary to develop a wider range of TTPs to achieve the same objectives.

2.2 Structural Design Principles for Cyber Resiliency

As noted above, strategic design principles express the organization's risk management strategy. Structural design principles support strategic design principles, as shown in Table 4. As Table 1 illustrated, the first four structural design principles are closely related to protection strategies and security design principles, and can be applied in mutually supportive ways. The next three are closely related to design principles for Resilience Engineering and Survivability. The next three are driven by the concern for an operational environment (which includes cyber threats) which changes on an ongoing basis, and are closely related to design principles for Evolvability. The final four are strongly driven by the need to manage the effects of malicious cyber activities, even when those activities are not observed.

Table 4. Structural Design Principles Support Different Strategic Design Principles

Structural Design Principle	Strategic Design Principle				
	Focus on common critical assets	*Support agility and architect for adaptability*	*Reduce attack surfaces*	*Assume compromised resources*	*Expect adversaries to evolve*
Limit the need for trust			X	X	
Control visibility and use	X		X	X	
Contain and exclude behaviors	X			X	X
Layer and partition defenses	X			X	
Plan and manage diversity	X	X		X	
Maintain redundancy	X	X			
Make resources location-versatile	X	X			X
Leverage health and status data	X			X	X
Maintain situational awareness	X	X			X
Manage resources (risk-) adaptively	X	X			X
Maximize transience; minimize persistence			X	X	X
Determine ongoing trustworthiness	X			X	X
Change or disrupt the attack surface			X	X	X
Make unpredictability and deception user-transparent					X

The structural design principles are intended to serve as a starting point for tailoring. For example, separate cyber resiliency design principles are stated for diversity and redundancy. For a given system, a more specific single design principle might be stated, which expresses how diversity and redundancy are

to be used together to provide agility, and how to make trade-offs between providing agility and reducing the attack surface.[22] Structural design principles (tailored for a given system or program) can be reflected in contractual requirements in two complementary ways. First, the Statement of Work can call for system documentation to describe how the design and implementation applies or is consistent with the design principles, or more concrete statements drawn from the discussion of those principles. Second, the programmatic Security Plan can map requirements to the selected design principles.[23]

For any given structural design principle, systems engineers can ask the following analytic questions:

- *Where* does the architecture or design apply the principle? Locations at which a principle can be applied can be identified in terms of
  - A systems engineering view, representing the system [61] or mission threads [44] via diagrams or models[24].
  - A layered view, as used in cybersecurity matrices. Layers can include hardware, software, networks, automation, suppliers, and human users [62], or hardware / firmware, networking / communications, system / network component, operating system, cloud / virtualization / middleware infrastructure, application / service, information store, information stream / feed, system / system-of-systems [4]. In addition, layers can include operating procedures, configuration settings, and physical environment (e.g., facility, mobile platform or vehicle). These environmental layers are of particular importance when applying a design principle to an as-built or as-deployed system.
- *How* does the architecture or design apply the principle? What is enforced upon system designers, what is built into the system, and what is expected of system users, administrators, and/or cyber defenders? For example,
  - Are specific system elements added? Are existing system elements repurposed?
  - Are specific interfaces or data flows precluded, and if so, how (e.g., physically, via configuration settings)?
  - Are specific functions precluded, and if so, how?
- *How well* is the design principle applied?
- *How effective* is this application of the design principle against malicious cyber activities, given a threat model[25]? [Note that this question is outside the scope of this document, and will be addressed in a subsequent report.]

For each of the structural design principles, a quick statement of the key concepts is followed by a somewhat longer discussion. One or two representative examples of more specific restatements are

[22] Such a system-specific design principle would be closely related to the concept of degeneracy as defined in [132]: "Degeneracy is the capacity for different elements to perform the same functions. … Unlike pure redundancy, degeneracy creates functional diversity and a high level of agility. … Parallel degenerate systems, in a cyber-context, can present additional risks for attack. By using multiple disparate systems at the same time, the attack surface available for an attacker is increased. However, used in sequence, the use of multiple, disparate systems allows for robust processes with faster incident response and greater flexibility."

[23] While specific statements tailored from the discussions of the selected design principles can be expressed as functional requirements, such requirements are unlikely to be testable.

[24] For IoT systems, elements in a model can be characterized as sensors, aggregators, communications channels, external utilities, and decision triggers [135]. For CPS, a high-level model includes sensors, actuators, physical systems, and controllers; these can be decomposed further and mapped to the physical, control, and cyber layers [137].

[25] A threat model represents an adversary's characteristics (e.g., capabilities, intent, targeting), TTPs, and activities or behaviors (frequently represented using threat scenarios, or in more fragmentary form as in CAPEC, and built out of threat events, e.g., as identified in ATT&CK).

presented, along with examples of *where* and *how* questions to be answered via analysis. In the context of those more specific restatements, examples of evidence or metrics are identified which might be used to evaluate *how well* the design principle is applied. Examples of potential evidence or metrics can be evaluated via a variety of methods, including analysis [A], modeling [M], use of asset inventory or configuration assessment tools [CT], testing [T], red teaming [RT], and analysis of operational data [O]. Note that ***these are not cyber resiliency metrics***; rather, they support analysis of whether and how well (e.g., how completely, how consistently) design principles are applied. However, they may serve as indicators of, or inputs to metrics for, cyber resiliency attributes. A future report will discuss metrics and other form of evidence for *how effective* an application of a design principle is – how much it improves cyber resiliency and/or how much it affects adversary activities, given a threat model.

Because the structural principles in this section draw from multiple sources and situations, some of the structural design principles presented below overlap, while others are not simultaneously satisfiable. This section is intended to serve as a starting point for the selection and tailoring of a set of system- or program-specific cyber resiliency design principles. See Sections 3.1-3.3 for discussion of specific engineering considerations related to such factors as stage in the SDLC, stakeholder concerns, and relationships with design principles from other specialty disciplines. See Section 3.4 for discussion of the questions of "How does the design principle apply a risk management strategy?" and "How does application of the design principle affect the adversary?"

2.2.1 Limit the Need for Trust

Key concept: Limiting the number of system elements that need to be trusted reduces the level of effort needed for assurance, as well as for ongoing protection and monitoring.

Discussion: As noted in Section 1.3.3, trustworthiness can be defined as "The attribute of [an entity] that provides confidence to others of the qualifications, capabilities, and reliability of that entity to perform specific tasks and fulfill assigned responsibilities." [31]) Assertions of trustworthiness (e.g., "this software can be relied upon to enforce the following security policies … with a high level of confidence") are meaningless without some form of validation or demonstration (e.g., analysis, testing). In the absence of some form of assurance (which can be costly and can be invalidated by changes in the environment), assertions of trustworthiness constitute assumptions. Reducing the size of the set of trusted entities (whether individuals, software components, or hardware components) by minimizing assumptions about what is or can be trusted reduces the attack surface, and lowers assurance costs.

Application of this design principle is easiest early in the SDLC, where the motivation of the Prevent / Avoid objective is clearest. When a system already exists, changes – to the operational concept (consistent with the Transform objective), or to the system's architecture (applying the Re-Architect objective, and the Realignment technique) can increase costs. One approach to applying this design principle (using the Coordinated Defense and Privilege Restriction techniques) is through limitations on inheritance, so that privileges or access rights associated with one class of component are not automatically propagated to classes or instances created from the original one. (While limitations on inheritance can increase the burden on developers or administrators initially, they can also reduce the complexity associated with multiple inheritance.)

This design principle supports the strategic design principles of *Reduce attack surfaces* and *Assume compromised resources*. However, its application increases the difficulty of applying the *Support agility and architect for adaptability* strategic design principle. For example, the use of Privileged Access Workstations (PAWs) (which applies this design principle, and uses Segmentation / Isolation in conjunction with the Purposing approach to Realignment) provides strong protections but means the workstations can be used only for a single purpose [63].

This design principle can be used in conjunction with *Determine ongoing trustworthiness*; if a system element is assumed or required to have a given level of trustworthiness, some attestation mechanism is needed to verify that it has – and continues to retain – that level. Minimizing the number of elements with trustworthiness requirements reduces the level of effort involved in determining ongoing trustworthiness. It can also be used in conjunction with *Plan and manage diversity*; the managed use of multiple sources of system elements, services, or information can enable behavior or data quality to be validated by comparison.

Table 5. Examples of Restatements of *Limit the Need for Trust*

Example of Restatement	Examples of Questions for Analysis	Examples of Metrics
Provide processes and procedures to determine required aspects of trustworthiness, to identify system elements for which those aspects are necessary, and to reduce the size of the set of trusted system elements	Which aspects of trustworthiness (e.g., security, safety, privacy, reliability, resilience) are required to ensure that the system meets its mission and supporting requirements? How are the trustworthiness requirements for a given system element determined (e.g., analysis, modeling, ad hoc)? Are system elements assumed to have a given minimum level of trustworthiness with respect to one or more aspects? If so, why?	Number of system elements (or types of elements); percentage of these for which trustworthiness requirements are identified [A] Percentage of system elements for which no trustworthiness requirements have been identified which are treated as trustworthy [A]
Provide the capability for attestation of or assurance in the trustworthiness attribute(s) of a system element	How can the level of trustworthiness of a given system element be determined? Must this be determined in a static way (e.g., via developmental assurance evidence), or can it be determined dynamically?	Number of system elements (or types of elements) for which trustworthiness requirements have been identified; percentage of such elements for which dynamic attestation is feasible [A]

2.2.2 Control Visibility and Use

Key concept: Controlling what can be discovered, observed, and used increases the effort needed by an adversary seeking to expand their foothold in or increase their impacts on cyber resources.

Discussion: Controlling visibility counters adversary attempts at reconnaissance, from outside or within the system. Thus, the adversary must work harder to identify potential targets, whether for exfiltration, modification, or disruption. Visibility of data can be controlled by such mechanisms as encryption, data hiding, or data obfuscation. Visibility of how some resources are used can also be controlled directly, for example by adding chaff to network traffic. Visibility into the supply chain, development process, or system design can be limited via operations security (OPSEC), deception [64], and split or distributed design and manufacturing. Process obfuscation is an area of active research [65]. And an increasing number and variety of deception technologies (e.g., deception nets) can be applied at the system level.

Controlling use counters adversary activities in the Control, Execute, and Maintain phases of the cyber attack lifecycle. To limit visibility or to control use, access to system resources can be controlled from the perspectives of multiple security disciplines, including physical, logical (see the discussion of privileges below), and hybrid (e.g., physical locations in a geographically distributed system or a complex embedded system). Restrictions on access and use can be based on information sensitivity, as in standard security practices. Restrictions can also be based on criticality – importance to achieving mission objectives. While some resources can be determined to be mission-critical or mission-essential *a priori*, the criticality of other resources can change dynamically; a resource which is vital to one phase of mission processing can become unimportant after that phase is completed.

Many systems or components provide the capability to define and manage privileges associated with software, services, processes, hardware, communications channels, and individual users. Assignment of privileges ideally should reflect judgments of operational need (e.g., need-to-know, need-to-use) as well as trustworthiness. Restriction of privileges is well-established as a security design principle (Least Privilege). Privilege restrictions force adversaries to focus efforts on a restricted set of targets, which can be assured (in the case of software), validated (in the case of data), or monitored (in the case of individuals, communications channels, processes, and services).

Non-Persistence and Segmentation can also limit visibility; thus, this principle can be applied in conjunction with the *Contain and exclude behaviors* and *Maximize transience* principles.

Table 6. Examples of Restatements of *Control Visibility and Use*

Example of Restatement	Examples of Questions for Analysis	Examples of Metrics
Restrict external visibility of system behaviors.	What can be observed about system behaviors from the vantage point of a system or individual interacting directly (i.e., without going through an intervening component such as a firewall, network gateway, or proxy server) with it (e.g., latency in responses to queries at the application or network layer, responses to failed login attempts)? What controls, if any, are placed on external systems which might attempt to interface with the system? (For example, are queries from sources not on a whitelist dropped without acknowledgement?) What can be observed about system behaviors from the vantage point of a system which indirectly interfaces with it? How, if anything, do intervening components obscure or obfuscate system behaviors (e.g., by introducing latency, by replicating queries or responses)?	Number of directly interfaced systems [A, CT] Number of indirectly interfaced systems (if knowable) [A, CT] Minimum or average time needed for an external entity to determine whether the system responds to a given type of query [M, RT] Minimum or average time needed for an external entity to estimate system load based on latency in response to queries [M, RT]
Restrict internal visibility and use of system resources, based on type of resource and on privileges of entity (user, process) seeking information about the resources.	How much can a system element operating at one layer (e.g., application, OS, or network) observe about an element operating at another layer? For example, can an application instance determine the network paths between it and the data stores or other applications it uses? How much can a system element operating at a given layer observe about its peer elements (i.e., those operating at the same layer)? For example, what can a service running in a virtual machine (VM) on a hardware platform observe about other services running in different VMs on the same platform? How do interacting system elements defend themselves against man-in-the-middle (MITM) attacks?	Minimum or average time needed for an application instance to identify the locations of the resources on which it depends [M, RT] Number or percentage or user/roles with visibility into mission critical data [A]

Example of Restatement	Examples of Questions for Analysis	Examples of Metrics
Provide the capability to dynamically assign, and then control use based on, a criticality attribute to those system resources which are mission-critical or mission-essential in at least one mission thread.	What assumptions about criticality are made about system resources? Are some types of resources assumed to be non-critical? Are some types of resources assumed to be high-criticality? How are mission threads identified – statically, as part of system design; dynamically, by observing patterns of use; or somewhere in between? How are the system resources used by mission threads identified? How is the criticality of a resource to a mission thread determined? For what types of system resources (if any) can criticality be represented, as an attribute which can be set by an administrator or determined automatically and visualized by an administrator? How is the criticality of a system resource used to determine controls on its usage? How are changes to the criticality of a system resource made?	Number or percentage of system resources which are mission-critical or mission-essential in at least one mission thread [A] Percentage of such resources for which a criticality attribute can be assigned [A] Percentage of such resources for which a criticality attribute can be dynamically assigned [A] Minimum, average, and maximum time between assignment / re-assignment of a criticality attribute to a resource and its use [M, RT]

2.2.3 Contain and Exclude Behaviors

Key concept: Limiting what can be done and where actions can be taken reduces the possibility or extent of the spread of compromises or disruptions across components or services.

Discussion: The behavior of a system element – what resources it uses, which system elements it interacts with, or when it takes a given action – can vary based on many legitimate circumstances. However, analysis of the mission or business process can identify some behaviors which are always unacceptable, and others which are acceptable only under specific circumstances. Excluding behaviors prevents such behaviors from having undesirable consequences. Behaviors can be excluded *a priori* with varying degrees of assurance, from removing functionality to restricting functionality or use, with trade-offs between assurance and flexibility. (For example, user activity outside specific time windows can be precluded.) In addition, behaviors can be interrupted based on ongoing monitoring, when that monitoring provides a basis for suspicion.

Containing behaviors involves restricting the set of resources or system elements which can be affected by the behavior of a given system element. Such restriction can, but does not have to, involve a temporal aspect. Containment can be achieved *a priori*, via pre-defined privileges and segmentation. Alternately or additionally, adaptive response and dynamic isolation can be applied. (For example, a sandbox or deception environment can be dynamically created in response to suspicious behavior, and subsequent activities can be diverted there.)

Table 7. Examples of Restatements of *Contain and Exclude Behaviors*

Example of Restatement	Examples of Questions for Analysis	Examples of Metrics
Provide mechanisms to disable any non-mission critical system element exhibiting suspicious behavior.	Which system elements cannot be disabled? Which system elements can be disabled, and how?	Number of system elements (or types of elements); percentage of these which cannot be disabled; percentage which can be automatically disabled; percentage which can be disabled by administrator action [A, CT]

Example of Restatement	Examples of Questions for Analysis	Examples of Metrics
Provide mechanisms for static and dynamic isolation of components.	Which components cannot be isolated? That is, which components are common assets, providing shared services or information which needs to be shared to ensure consistent decisions? Which can be isolated a priori, e.g., by running applications and services in their own virtual containers isolated from other applications, services, and much of the operating system? Which components can be isolated dynamically, e.g., by reconfiguration of VPNs or by physically removing network links?	Number of common (non-isolatable) components [A] Number of potentially isolatable components; percentage of these which are isolated a priori; percentage of these which can be isolated dynamically [A] Minimum or average time needed to isolate a component [M, RT]
Provide mechanisms for restriction or containment of activities (i.e., related sequences of actions or events), a priori or dynamically.	What types of activities can be identified, tracked, and thus potentially be restricted – e.g., activities by an individual user, activities from a specific device or network node, activities by a scheduled service? On what basis are different types of activities restricted? On what basis can activities be tracked and identified as potentially suspicious (e.g., time of day, physical location, network location)? What mechanisms are provided to constrain the effects of activities (e.g., sandboxing, time-based restrictions, process termination, VM termination)? On what basis are these mechanisms invoked – automatically, or in response to suspicious activities?	Time between identification of suspicious activities and the restriction or containment of subsequent activities associated with that user [M, CT]

2.2.4 Layer Defenses and Partition Resources

Key concept: The combination of defense-in-depth and partitioning increases the effort required by an adversary to overcome multiple defenses.

Discussion: *Defense-in-depth* – "integrating people, technology, and operations capabilities to establish variable barriers across multiple layers and missions" [31] – is a well-established security strategy. It "describes security architectures constructed through the application of multiple mechanisms to create a series of barriers to prevent, delay, or deter an attack by an adversary." [1] Multiple mechanisms to achieve the same objective or provide equivalent functionality can be used at a single layer (e.g., different COTS firewalls to separate zones in a DMZ) or at different layers (e.g., detection of suspicious behavior at the application, OS, and network layers). To avoid inconsistencies which could result in errors or vulnerabilities, the multiple mechanisms must be managed consistently.

Layering of defenses restricts the adversary's movement vertically in a layered architecture; a defense at one layer prevents a compromise at an adjacent layer from propagating. *Partitioning* – separating sets of resources into effectively separate systems, with controlled interfaces (e.g., cross domain solutions or CDSs) between them – restricts the adversary's movement horizontally or laterally. Partitioning can limit the adversary's visibility; see *Control visibility and use*. It can also serve to *Contain and exclude behaviors*. Partitioning can be based on administration and policy, as in security domains [1], or can be based on the missions the system elements in the partition support. Partitions can be implemented physically or logically [66], at the network layer and within a platform (e.g., via hard or soft partitioning [67]). Note that partitioning may entail limiting resource sharing, making fewer resources common; if resources are replicated, the *Maintain redundancy* principle needs to be applied.

Table 8. Examples of Restatements of *Layer Defenses and Partition Resources*

Example of Restatement	Examples of Questions for Analysis	Examples of Metrics
Provide multiple mechanisms to achieve security policy or cyber resiliency objectives, at different architectural layers.	For which security policy objectives (e.g., protect, detect, react; confidentiality, integrity, availability, non-repudiation) and which cyber resiliency objectives are multiple mechanisms provided? For each objective or functional requirement for which multiple mechanisms are provided: What is the balance between multiple mechanisms at the same layer and multiple mechanisms at different layers? How is consistency across those mechanisms ensured? Are any automated checks made of policy-related settings (e.g., roles, privileges) or objective-related configuration settings (e.g., resource allocation priorities)? How quickly can an administrative change be propagated across the set of mechanisms that provide the same or similar functionality?	For each objective or functional requirement: Number of different mechanisms at each layer [A]; number of different layers [A] Minimum, maximum, average time to validate consistency of a change in settings [M, T] Minimum, maximum, average time for a change in settings to be complete across all system elements to which it applies [M, T]
Provide mechanisms for static and dynamic partitioning of system elements.	How are partitions defined (e.g., in terms of security policies to enforce, missions to be supported, organizational elements in control)? How are partitions implemented (e.g., physically, logically)? How are interfaces between partitions controlled? How are interactions between partitions monitored? How are new partitions created? Who has the authority to create a new partition?	Minimum or average time needed to establish a new partition [M, T] Minimum time needed to migrate elements between partitions [M, T]
Limit shared resources; provide mechanisms to dedicate resources to distinct instances of services, applications, or missions.	Which resources are shared across multiple services, applications, or missions? Why are these resources shared rather than dedicated? For each mission-critical system element: What resources does the system element depend on? Which resources are dedicated to it, and which are shared resources? For the shared resources, is the mission-critical system element given priority?	For each mission-critical system element: Percentage of resources it uses which are dedicated to it; percentage which are shared [A, CT]

2.2.5 Plan and Manage Diversity

Key concept: Diversity is a well-established resilience technique, removing single points of attack or failure. However, architectures and designs must take cost and manageability into consideration to avoid introducing new risks.

Discussion: Diversity (usually in conjunction with Redundancy [68]) is a well-established technique for improving system resilience [69] [70]. For cyber resiliency, Diversity avoids the risk of a monoculture, in which compromise of one component can propagate to all other such components. Diversity offers the benefit of providing alternative ways to provide required functionality, so that if a component is compromised, one or more alternative components which provide the same functionality can be used.

Multiple approaches to diversity can be identified; these include architectural diversity, design diversity, synthetic (or automated) diversity[26], information diversity, diversity of command, control, and

[26] Synthetic diversity in conjunction with randomization, a form of Unpredictability, is a form of Moving Target Defense (MTD).

communications (C3) paths (including out-of-band communications), and supply chain diversity [3] [1], as well as geographic diversity[27] and diversity in operating procedures. In addition, some incidental architectural diversity often results from procurement over time and differing user preferences [3]. (Incidental diversity is often more apparent than real; that is, different products can present significantly different interfaces to administrators or users, while incorporating identical components.)

However, diversity can be problematic in several ways: First, it can increase the attack surface. Rather than trying to compromise a single component and propagate across all such components, an adversary can attack any component in the set of alternatives, looking for a path of least resistance to establish a foothold. Second, it can increase demands on developers, system administrators, maintenance staff, and users, by forcing them to deal with multiple interfaces to equivalent components. This translates into increased lifecycle costs. (These costs have historically been acceptable in some safety-critical systems.) This can also increase the risks that inconsistencies will be introduced, particularly if the configuration alternatives for the equivalent components are organized differently.

Third, diversity can be more apparent than real (e.g., multiple different implementations of the same mission functionality all running on the same underlying OS, applications which reuse software components). Thus, analysis of the architectural approach to using diversity is critical. For embedded systems, some approaches to diversity raise a variety of research challenges [71]. And finally, the effectiveness of diversity against adversaries is not an absolute: analysis of diversity strategies is needed to determine the best alternative in the context of adversary TTPs [72] [73] [74].

Therefore, this design principle calls for the use of diversity in system architecture and design to take manageability into consideration. It also calls for consideration of diversity in operational processes and practices, including non-cyber alternatives such as out-of-band measures for critical capabilities.[28] To reduce cost and other impacts, this design principle is most effective when used in conjunction with the *Focus on common critical assets* strategic principle and the *Maintain redundancy* and *Layer and partition defenses* structural principles. Measurements related to this design principle can focus on the degree of diversity, manageability, or both.

Table 9. Examples of Restatements of *Plan and Manage Diversity*

Example of Restatement	Examples of Questions for Analysis	Examples of Metrics
Provide equivalent non-identical capabilities at geographically diverse sites. [Note: This restatement is also a restatement of *Maintain redundancy*.]	Which capabilities are provided in different but equivalent ways at geographically diverse sites? What is the basis for determining which capabilities are diversified in this way (e.g., ease of implementation, mission criticality, criticality as a function supporting multiple missions)? How is the equivalence of capabilities determined (e.g., analysis, testing)? What effects, if any, on operational processes and procedures result from equivalent-but-not-identical capabilities? How are these equivalent capabilities managed (e.g., access or privilege management, resource allocation, configuration settings)? How can the equivalency of the management actions be determined (e.g., analysis, testing)? How are management actions for one capability converted into management actions for an equivalent capability (e.g., administrator analysis, documented instructions for administrators, automated scripts)?	Number of capabilities diversified both architecturally and geographically (Diversity) [A] Percentages of mission-critical and of mission-essential functions so diversified (Diversity) [A] Time required to convert management actions for one capability into equivalent management actions for each equivalent capability (Manageability) [M, RT, O]

[27] Geographic diversity can be used to support the *Make resources location-versatile* structural design principle.
[28] See [133], p. 21.

Example of Restatement	Examples of Questions for Analysis	Examples of Metrics
Provide diversity-in-depth for selected mission threads; that is, for a given mission thread, ensure that the different ways to execute the thread do not rely on a single common type of component at any architectural layer. [Note: This is also a restatement of *Layer and partition defenses*.]	For which mission threads is diversity-in-depth provided? On what basis are these threads selected? For a given mission thread, at how many layers is diversity provided (including not only technical layers, but also procedures, configuration settings, and physical environment)? Are non-cyber resources (e.g., paper, face-to-face communications, physical locks) used to provide diversity? How is the equivalency of the different ways to execute a mission thread established (e.g., analysis, testing)? How is the equivalency of the different ways to execute a mission thread maintained over time, given how resources are (or will be) managed locally or with respect to their primary functions?	Number of architectural layers at which diversity is provided (Diversity) [A] Number of different ways (including non-cyber) to execute the thread (Diversity) [A] Number of differently-managed components involved in thread execution (Manageability) [A]
Provide diversity for critical system elements.	To which critical system elements is diversity applied? How is it applied – e.g., architecturally (using different standards), via parallel design efforts, or synthetically?	Percentage of critical system elements to which diversity is applied [A]

2.2.6 Maintain Redundancy

Key concept: Redundancy is key to many resilience strategies, but can degrade over time as configurations are updated or connectivity changes.

Discussion: Redundancy is a well-established design principle in Resilience Engineering and Survivability [69]. Approaches to Redundancy include surplus capacity and replication (e.g., cold spares, hot or inline spares), and can be implemented in conjunction with backup and failover procedures. It can enhance the availability of critical capabilities, but requires that redundant resources be protected.

Because malware can propagate across homogeneous resources, Redundancy for cyber resiliency needs to be applied in conjunction with Diversity, and needs to be considered at multiple levels or layers in a layered architecture [68]. Thus, Redundancy, all the more so when used in conjunction with Diversity, can increase complexity and present scalability challenges.

The extent of Redundancy must be established and maintained through analysis, looking for single points of failure and shared resources. Trends to convergence can undermine Redundancy; for example, an organization using VOIP (Voice over Internet Protocol) for its phone system cannot assert alternate communications paths for phone, email, and instant messaging.

Because maintaining surplus capacity or spare components increases lifecycle costs, this design principle is most effective when used in conjunction with the *Focus on common critical assets* strategic principle. As the discussion above indicates, it is also most effective in conjunction with the *Plan and manage diversity* and *Layer and partition defenses* structural principles.

Table 10. Examples of Restatements of *Maintain Redundancy*

Example of Restatement	Examples of Questions for Analysis	Examples of Metrics
Duplicate resources in multiple locations, keeping the resources synchronized.	Which resources (e.g., information stores, services, communications paths) are duplicated? What is the basis for determining which resources are duplicated (e.g., ease of implementation, mission criticality, criticality as a function supporting multiple missions)? How are authorized changes to one resource (e.g., database update, change to the privilege or configuration settings for a service, software update) propagated to its duplicates? What mechanisms can detect inconsistencies between resources which should be identical?	Percentage of critical assets which are duplicated [A] Average and/or maximum time between an authorized change to one resource and replication of that change to all duplicate resources [M, RT, O]
Maintain surplus capacity for information storage, processing, and/or communications.	For which types of resources (e.g., information storage, processing, communications) is surplus capacity provided? On what basis are the specific resources for which surplus capacity will be provided selected? (For example, are these resources needed to handle mission or operational contingencies?) How much surplus capacity is provided? [Note that capacity measurements, and hence measurements of surplus capacity, can be expressed in terms of Key Performance Parameters (KPPs).] How is surplus capacity maintained? What evidence or triggers are used to determine that the surplus has been depleted?	For selected resources: Percentage of capacity which is surplus [A] Average and/or maximum time to determine that the amount of surplus capacity has dropped below an acceptable level [M, RT, O]

2.2.7 Make Resources Location-Versatile

Key concept: A resource bound to a single location (e.g., a service running only on a single hardware component, a database located in a single datacenter) can become a single point of failure, and thus a high-value target.

Discussion: Location-versatile resources are those which do not require a fixed location, and which can be relocated or reconstituted to maximize performance, avoid disruptions, and better avoid becoming a high-value target for an adversary. Different approaches can be used to provide location-versatile resources. These include virtualization, replication, distribution (of functionality or stored data), physical mobility, and functional relocation. Replication is a well-established approach for high-availability systems, using multiple parallel processes [75], as well as high-availability data (sometimes referred to as data resilience) using database sharding (although this can present security challenges [76]).

Replication and distribution can be across geographic locations, hardware platforms, or (in the case of services) VMs. While replication can take the form of redundancy, it can also involve providing ways to reconfigure resources to provide equivalent functionality. Data virtualization – data management which enables applications to retrieve and use data without specific knowledge of the data's location or format – supports distribution, and reduces the likelihood that local (persistent and unmaintained) data stores will proliferate. Composable services enable alternative reconstitution of mission capabilities; diverse information sources can be used for alternative reconstitution of mission data.

Application of this principle involves the use of Dynamic Positioning, often in conjunction with Redundancy and/or Diversity. This principle supports the *Support agility and architect for adaptability* strategic principle, and can be used in conjunction with the *Maximize transience; minimize persistence*

and *Change or disrupt the attack surface* structural principles. Some approaches to reconstitution of mission capabilities can conflict with the *Control visibility and use* structural principle.

Table 11. Examples of Restatements of *Make Resources Location-Versatile*

Example of Restatement	Examples of Questions for Analysis	Examples of Metrics
Distribute resources to eliminate single points of failure.	What types of resources can be distributed? How are resources that are distributed selected (e.g., based on criticality)? How are resources distributed (e.g., across geographic locations, across hardware platforms, across VMs)? What mechanisms are used to enable distributed resources to be accessed or used as a system element? What mechanisms are provided to ensure that, if one portion of a distributed resource is damaged or unavailable, the capabilities it provided can be reconstituted?	Number of resource types which can be distributed; percentage of mission-critical resources which are distributed [A] Minimum and average time to reconstitute damaged resource [T, RT, O]
Replicate resources to eliminate single points of failure.	What resources are replicated? How are resources selected for replication? How are resources replicated (e.g., physically, virtually)? What mechanisms are used for replication (e.g., database mirroring)? How is consistency (an attribute of integrity) among instances of a replicated resource ensured or validated?	Number of resource types which can be replicated; percentage of mission-critical resources which are replicated [A] Minimum and average time to ensure consistency among instances of a replicated resource [T, O]
Provide mechanisms to enable resources to be relocated or alternately reconstituted.	What types of resources can be relocated? How are relocatable resources selected? How can resources be relocated (e.g., between geographic locations, hardware platforms, VMs)? What mechanisms are used to relocate resources and ensure that they can be located and used? What types of resources can be reconstituted out of other resources? What alternative reconstitution mechanisms are provided?	Number of resource types which can be relocated; percentage of mission-critical resources which can be relocated [A] Minimum and average time to ensure that a relocated or reconstituted resource can be located and used [T, O]

2.2.8 Leverage Health and Status Data

Key concept: Health and status (H&S) data can be useful in supporting situational awareness, indicating potentially suspicious behaviors, and predicting the need for adaptation to changing operational demands.

Discussion: In some architectures, many system components are security-unaware, incapable of enforcing a security policy (e.g., an access control policy) and hence of monitoring policy compliance (e.g., auditing or alerting on unauthorized access attempts). However, virtually every system component provides H&S data, to indicate its availability (or unavailability) for use. These include components of CPS, particularly components in space systems and in the emerging IoT. In addition, system components present H&S data to orchestration components (e.g., application or service on a virtual platform in a cloud to a cloud orchestrator) or service-providing components (e.g., application to OS, device to network) so that those components can allocate and scale resources more effectively. Correlation of monitoring data, including H&S data, from multiple layers or types of components in the architecture can identify potential problems early, so they can be averted or contained.

As architectural convergence between information technology (IT) and operational technology (OT) or the IoT increases [34], application of this structural principle will support the *Expect adversaries to evolve* strategic principle. Given the increasing number and variety of "smart" components in the IoT, application of this principle may be driven by the *Focus on critical components* principle.

In addition, components can erroneously or maliciously report H&S data, by design (as in the case of VW diesel engines) or due to compromise (e.g., as resulted from Stuxnet). Thus, application of this principle may be more effective in conjunction with the *Determine ongoing trustworthiness* principle.

Table 12. Examples of Restatements of *Leverage Health and Status Data*

Example of Restatement	Examples of Questions for Analysis	Examples of Metrics
Analyze monitoring and self-reporting of health and status for indicators of abnormal behavior.	Which types of system elements or components are capable of self-reporting health and status (e.g., applications or services, devices)? To which other system elements or components do they report (e.g., cloud orchestrator, network manager)? For which types or instances is H&S reporting enabled? On what basis is this determined? How do system elements or components to which H&S data is reported use such data (e.g., performance monitoring, resource allocation)? Is any trend analysis or comparison with a baseline performed? How do system elements or components to which H&S data is reported respond to beyond-acceptable changes in that data (e.g., terminate a service or process, a virtual machine, or a network connection; alert Security Information and Event Management (SIEM) services)?	Number of types of components or elements capable of reporting H&S [A] Percentage of such types which are critical [A] Percentages of such types (critical vs. non-critical) for which H&S reporting is enabled [A] Minimum or average time needed for to determine whether H&S reporting merits a response by a cyber defender [M, RT]
Fuse monitoring and self-reporting of health and status data with other monitoring and analysis data to develop indicators of abnormal behavior.	At what layers do system elements generate H&S data, and to system elements at which layers is that data reported? At what layers do system elements perform analytic monitoring of security-related and other behaviors (e.g., performance)? How is security monitoring data combined with other monitoring data, and with H&S data? What mechanisms (e.g., fusion, pattern analysis, trend analysis) are provided to enable cyber defenders or analysts to develop indicators of abnormal behavior?	Percentage of system elements whose behaviors are monitored, either via H&S data, security-related monitoring, or other forms of monitoring [A] Minimum or average time needed to develop new indicators [RT, O]

2.2.9 Maintain Situational Awareness

Key concept: Situational awareness – including awareness of possible performance trends and the emergence of anomalies –informs decisions about cyber courses of action to ensure mission completion.

Discussion: In the context of cybersecurity and cyber resiliency, situational awareness encompasses awareness of *system elements*, *threats*, and *mission dependencies* on system elements [77].[29] Awareness of system elements can rely on security posture assessment, security monitoring, and performance monitoring, and can be achieved in conjunction with the *Leverage health and status data* principle. Awareness of threats involves ingesting and using threat intelligence, recognizing that adversaries evolve.

[29] As a capability of a Security Operations Center (SOC), situational awareness provides "regular, repeatable repackaging and redistribution of the SOC's knowledge of constituency assets, networks, threats, incidents, and vulnerabilities to constituents. This capability goes beyond cyber intel distribution, enhancing constituents' understanding of the cybersecurity posture of the constituency and portions thereof, driving effective decision making at all levels." [124]

Awareness of system elements and of threats – via gathered and correlated data, and processing capabilities – can be centralized or distributed, and can be enterprise-internal or cross-enterprise (e.g., via a managed security service provider or MSSP).

Awareness of mission dependencies can be determined *a priori*, as part of system design (e.g., using CJA, MIA, or BIA). Alternately or additionally, mission dependencies can be identified in the course of mission operations, by tracking and analyzing resource use. This more dynamic approach supports agility and adaptability, and supports capabilities to *Control visibility and use* and *Contain and exclude behaviors*. While cyber situational awareness remains an active area of research [78] [79], analytic capabilities are increasingly being offered [80], and cyber situational awareness is maturing through tailored applications in specific environments.

Table 13. Examples of Restatements of *Maintain Situational Awareness*

Example of Restatement	Examples of Questions for Analysis	Examples of Metrics
Provide capabilities to correlate cybersecurity posture data with other data about system elements.	Of which types of system elements are security monitoring and analysis capabilities aware? Of which types are they unaware? For which types of system elements can non-security-related monitoring (e.g., performance, H&S) be performed? How are different forms of monitoring fused to provide an overall picture of the status of system elements with respect to expected behaviors? (For example, are physical access events correlated with system use?) What are the gaps in situational awareness? Which types of system elements are not represented? Which types of unexpected behaviors are not detectable?	Number of types of system elements; percentage of types subject to security monitoring; percentage of types subject to other forms of monitoring; percentage of types for which no monitoring is performed [A] Minimum and average time between triggering event and detection of its impacts on system elements [M, RT]
Provide capabilities to correlate or apply threat intelligence with cybersecurity posture data.	What capabilities to ingest threat intelligence are provided? What capabilities are provided to apply threat intelligence (e.g., look for damage to or increase monitoring of targeted resources)?	Minimum and average time between arrival of threat intelligence and its effective use [RT]
Provide capabilities to represent the mission impacts of changes in the posture of system elements.	How are mission dependencies on system elements identified (e.g., a priori, in real time, both)? Is the system capable of identifying changes in mission dependencies over time (e.g., in the course of mission operations)? How is mission posture represented to mission owners and operators? How are the implications of changes in the posture of system elements reflected in the representation of the mission posture? Are those changes reflected only for mission-critical system elements, or for all system elements?	Minimum and average time between a change in mission dependencies on system elements and when that change is represented to mission operators [M, RT] Minimum and average time between a change in the posture of a mission-critical system element and the reporting of the resulting change in mission posture [M, RT]

2.2.10 Manage Resources (Risk-) Adaptively

Key concept: Risk-adaptive management supports agility, providing supplemental risk mitigation throughout critical operations, despite disruptions or outages of components.

Discussion: Risk-adaptive management has been developed in multiple contexts. Cybersecurity mechanisms include risk-adaptive access control (RAdAC) for systems, highly adaptive cybersecurity services (HACS) providing such functionality as penetration testing, incident response, cyber hunting, and risk and vulnerability assessment [81] for programs, and integrated adaptive cyber defense (IACD) for the enterprise and beyond [82].

Strategies for risk-adaptive management include *changing the frequency of planned changes* (e.g., resetting encryption keys, switching between OSs or platforms, or changing configuration of internal routers), *increasing security restrictions* (e.g., requiring reauthentication periodically within a single session, two-factor authentication for requests from remote locations, or two-person control on specific actions; increasing privilege requirements based on changing criticality), *reallocating resources* (e.g., reallocating processing, communications, or storage resources to enable graceful degradation; repurposing resources), and *discarding or isolating suspected system elements* (e.g., terminating a service or locking out a user account; quarantining processing; diverting communications to a deception environment). Thus, strategies for implementing this design principle can be applied in conjunction with strategies for implementing *Control visibility and use* (dynamically changing privileges) *Contain and exclude behaviors* (disabling resources, dynamic isolation), *Layer defenses and partition resources* (dynamic partitioning), *Plan and manage diversity* (switching from one resource to an equivalent), and *Make resources location-versatile* (reconstituting resources).

To be *risk*-adaptive, the selection and application of a strategy needs to be based on situational awareness: the management decisions are based on indications of changes in adversary characteristics, characteristics of system elements, or patterns of operational use which change the risk posture of the system or the mission it supports. Alternately, strategies can be applied unpredictably, to address unknown risks.

Table 14. Examples of Restatements of *Manage Resources (Risk-) Adaptively*

Example of Restatement	Examples of Questions for Analysis	Examples of Metrics
Provide capabilities or mechanisms to change the frequency of planned changes based on changes in the risk posture.	For what types of resources, and at what layers in the architecture, are planned changes made? What mechanisms are used to make those changes (e.g., configuration management, cloud service orchestration management, security management)? For which types of resources, and where in the architecture, are change control mechanisms used risk-adaptively (i.e., based on cyber situational awareness)? How are the mechanisms used risk-adaptively (e.g., manually or human-in-the-loop (HITL), semi-automatically or human-on-the-loop (HOTL), automatically)? How are the changes reflected in cyber situational awareness?	Number of types of resources to which change management controls apply; percent of these to which change management controls are applied risk-adaptively [A] Minimum and average time between a triggering event and the change in frequency of a planned change [M, T]
Provide capabilities or mechanisms to change security restrictions based on changes in the risk posture.	For what types of resources are security restrictions (e.g., access or usage control) applied? What mechanisms are used to set the policies or requirements for those restrictions (e.g., identity and access management (IdAM) controls)? For which types of resources, and where in the architecture, are security restriction mechanisms used risk-adaptively (i.e., based on cyber situational awareness)? How are the mechanisms used risk-adaptively (e.g., HITL, HOTL, automatically)? How are the changes reflected in cyber situational awareness or security administration visualization tools?	Number of types of resources to which security restrictions apply; percent of these to which restrictions are changed risk-adaptively [A] Minimum and average time between a triggering event and the change in security restrictions [M, T]

2.2.11 Maximize Transience; Minimize Persistence

Key concept: Use of transient system elements minimizes the duration of exposure to adversary activities, while periodically refreshing to a known good state can expunge malware or corrupted data.

Discussion: Non-persistence is a strategy to *Reduce attack surfaces* in the temporal dimension [83]. Virtualization technologies, which simulate the hardware and/or software on which other software runs [84], enable processes, services, and applications to be transient. At the network layer, technologies for network virtualization, network functions virtualization, software defined networking, and just-in-time connectivity can support non-persistence [85]. Data virtualization provides a strategy for reducing persistent local data stores. As noted above, this principle is synergistic with *Make resources location-versatile*. Since transient resources can be virtually isolated, this principle can also be used in conjunction with *Contain and exclude behaviors*.

Logical transient system elements (processes, files, connections) need to be expunged – removed in such a way that no data is left behind on the shared resources.[30] If a running process or service has been compromised by malicious software which changes its behavior or corrupts the data it offers to other system elements, expunging it – either by bringing it down or by moving it and deleting the prior instance – also expunges the compromise. This can be done in response to suspicious behavior, or can be deliberately unpredictable.

In addition, cyber-physical system elements can be made attritable and expendable, as in the case of unmanned air systems (UAS) [86]. These physically transient system elements also need mechanisms for ensuring that no data is left behind.

Instantiation of a transient resource depends on being able to *Determine ongoing trustworthiness* of the resources from which it is constructed. Support for such validation can include, for example, gold copies of software and configuration data; policy data for network function virtualization; and data quality validation as part of data virtualization.

Table 15. Examples of Restatements of *Maximize Transience; Minimize Persistence*

Example of Restatement	Examples of Questions for Analysis	Examples of Metrics
Provide resources which can be instantiated on demand and expunged when no longer needed.	What types of resources can be instantiated on demand (e.g., data, services, network connections)? For each type of resource which can be instantiated on demand and is not, what is the rationale for making it persistent? What mechanisms are provided to instantiate resources on demand? How are resources expunged (e.g., processors, storage, connections released; memory erased) when no longer needed? How much assurance can be placed in the expunging process?	Minimum, average, and maximum time to instantiate a given resource [M, T, O] Minimum, average, and maximum time to expunge an instance of a given resource [M, T, O] Minimum, average, and maximum time for an adversary to replicate an expunged resource, or to determine that it cannot be replicated [RT]

[30] See NIST SP 800-53R4, controls SC-4 (Information in Shared Resources) and MP-6 (Media Sanitization).

Example of Restatement	Examples of Questions for Analysis	Examples of Metrics
Provide resources which periodically refresh to a known good states.	For what types of resources can a known good state be defined? For what types of resources can a refresh be performed? How are the resources for which the refresh capability is used selected? What mechanisms are used to refresh resources to a known good state? What mechanisms are used to determine whether a resource is in a known good state?	Minimum, average, and maximum time to refresh a given resource to a known good state [M, T, O] Minimum, average, and maximum time to determine whether a given resource is in a known good state [M, T, O]

2.2.12 Determine Ongoing Trustworthiness

Key concept: Periodic or ongoing validation of the integrity or correctness of data or software can increase the effort needed by an adversary seeking to modify or fabricate data or functionality. Similarly, periodic or ongoing analysis of the behavior of individual users, system components, and services can increase suspicion, triggering responses such as closer monitoring, more restrictive privileges, or quarantine.

Discussion: In the Control phase of the cyber attack lifecycle, an adversary can modify system components (e.g., modify software, replace legitimate software with malware), system data (e.g., modify configuration files, fabricate entries in an authorization database, fabricate or delete audit data), or mission data (e.g., deleting, changing, or inserting entries in a mission database; replacing user-created files with fabricated versions). These modifications enable the adversary to take actions in the Execute and Maintain phases. Periodic or ongoing validation can detect the effects of adversary activities before those effects become too significant or irremediable.

A variety of Substantiated Integrity mechanisms can be used to identify suspicious changes. Changes can be to properties or to behavior. Some behaviors – for example, the frequency with which a service makes requests, the latency between a request to it and its response, the size of requests or responses it makes – can be validated by other services. Other behaviors – for example, processor, memory, disk use, or network use – can be validated by other system components (e.g., the OS's task manager). Note that making the behavior capable of being validated can impede the use of unpredictability.

This principle is strongly synergistic with *Manage resources (risk-) adaptively*: Some changes can trigger the use of Privilege Restriction or Analytic Monitoring mechanisms; others can trigger quarantine via Segmentation. However, such mechanisms can add processing, transmission, and storage overhead. Therefore, this structural principle is most effective in support of the *Focus on common critical assets* strategic principle.

Ideally, any system element which cannot be determined to be trustworthy should be assumed to be compromised; in practice, that assumption is hard to apply. This principle is consistent with the weaker assumption that some resources will be compromised, and calls for mechanisms to detect and respond to evidence of compromise.

Mechanisms to determine trustworthiness need to be applied in a coordinated way, across architectural layers, among different types of system elements, and (if applicable) with Insider Threat controls.

Table 16. Examples of Restatements of *Determine Ongoing Trustworthiness*

Example of Restatement	Examples of Questions for Analysis	Examples of Metrics
Make the behavior of services capable of being validated.	For which services can behaviors be validated? How were these services chosen – based on criticality, based on ease of implementing the validation? Which behaviors can be validated? How are behaviors validated – by what other system elements, with a specific frequency or sampling rate, or for each interaction, and on what basis? How are thresholds (both minimum and maximum) established? If based on past behavior, how long a time window is used? (In effect, how easily and how quickly can an adversary change the behavior profile by causing the service's behavior to drift over time?) What notifications are made or what actions are taken if the behavior of a service exceeds or falls below its expected thresholds?	Number of distinct services that the system may run, percentage which are critical, and percent of critical and non-critical services for which behaviors can be validated [A] Minimum, average, and maximum time between when service behavior diverged from expected and when the divergence was detected [M, T, RT]
Provide or employ services to validate the integrity of device configurations, software modules, and critical data.	For which resource types (e.g., database records, data files, device configuration files, executable modules) can the integrity be validated? For which resources will integrity be validated? How were these resources chosen – based on criticality, based on ease of implementing the validation? How is the integrity of a resource validated? How easily can the validation mechanism be circumvented (e.g., by substituting a new signature)? When is the integrity of a resource validated – on a periodic basis, on a random basis, on the basis of a triggering event? How do integrity validation mechanisms interact with mechanisms for protected backup and restoration?	Number of resource types for which integrity can be validated, and percent of resources of each type for which integrity is validated [A] Minimum, average, and maximum time between when a resource is corrupted and when that corruption is detected [M, T, RT, O]

2.2.13 Change or Disrupt the Attack Surface

Key concept: Disruption of the attack surface can cause the adversary to waste resources, make incorrect assumptions about the system or the defender, or prematurely launch attacks or disclose information.

Discussion: Disruption of the attack surface can also lead an adversary to reveal their presence. A growing set of moving target defenses are intended to change or disrupt a system's attack surface. Moving Target Defense (MTD) is an active area of research and development. See, for example, [87] [88] as well as the proceedings of the three ACM MTD workshops.

MTD can be categorized in terms of the *layer* or level at which the defenses are applied, e.g., data, software, runtime environment, platform, and network [89]. However, MTD can be applied at other layers; for example, when this design principle is used in conjunction with the *Make resources location-versatile* principle, MTD can also be applied at the physical or geographic levels. MTD is particularly well suited to cloud architectures [90], where implementation is at the middleware level.

MTD can also be categorized in terms of *strategy*: move, morph, or switch. Resources can be moved; for example, execution of a service can be moved from one platform or virtual machine to another. This approach, which leverages Dynamic Positioning, can be used in conjunction with the *Make resources location-versatile* principle. The terms "cyber maneuver" and MTD are often reserved for morphing: making changes to the properties of the runtime environment, software, data, platform, or network [91] or by using configuration changes in conjunction with Diversity and Unpredictability or randomization [87]

[88] [92], rather than including relocation or distribution. Data or software can be morphed, using synthetic diversity; the behavior of system elements can be morphed via configuration or resource allocation changes. Morphing can be part of a Deception strategy. Finally, switching can leverage diversity and distributed resources; mission applications which rely on a supporting service can switch from one implementation of the service to another. Switching can also be used in conjunction with Deception, as when adversary interactions with the system are switched to a deception environment.

This structural design principle supports the *Expect adversaries to evolve* strategic principle. It can support the *Reduce attack surfaces* strategic principle; alternately, it can support the *Assume compromised resource*s principle. When Unpredictability is part of the way this principle is applied, it must be used in conjunction with the *Make unpredictability and deception user-transparent* structural principle.

Table 17. Examples of Restatements of *Change or Disrupt the Attack Surface*

Example of Restatement	Examples of Questions for Analysis	Examples of Metrics
Provide capabilities or mechanisms to move resources from one system element to another.	What types of resources at a given layer can be moved (e.g., data stores, services)? On what basis are such resources selected (e.g., criticality)? How are resources moved – what mechanisms are provided, and what underlying mechanisms do they depend on (e.g., VMs)? What causes resources to be moved (e.g., suspicious behavior, performance load on a given system element, randomization)?	Number of resource types which can be relocated, and percent of resources of each type for which relocation mechanisms are provided [A] Minimum, average, and maximum time between when the triggering event and completion of resource relocation [M, T, RT, O]
Provide mechanisms to switch from one resource to an equivalent set of different resources.	For what types of resources are equivalent different versions provided? On what basis are resources chosen (e.g., criticality)? What causes the use of a resource by a given system element to be switched to an equivalent resource? What constitutes a triggering event?	Number of resource types for which equivalent alternatives are possible; percentage of resources of each type for which alternatives are offered [A] Minimum, average, and maximum time between the triggering event and completion of the switch [M, T, RT, O]
Provide capabilities or mechanisms to change the behavior or attributes of resources.	At what layers (e.g., network, platform, OS, application or service, data store) can behavior or attributes (e.g., performance, latency, privilege requirements, other usage constraints) be changed? How are the behavior or attributes of a resource changed (e.g., configuration changes, resource allocation changes)? What causes resource behavior or attributes to be changed? What constitutes a triggering event? How are the behavior or attributes of a resource changed (e.g., automated mechanism, operator or administrator intervention)?	For each layer at which resource characteristics (behavior, attributes) can be changed: Number of resource types whose characteristics can be changed, and percent of resources of each type for which reconfiguration mechanisms are provided [A] Minimum, average, and maximum time between when the triggering event and completion of resource reconfiguration [M, T, RT, O]

2.2.14 Make Unpredictability and Deception User-Transparent

Key concept: Deception and unpredictability can be highly effective techniques against cyber adversaries, leading them to reveal their presence or TTPs, or to waste effort. However, when improperly applied, these techniques can also confuse users.

Discussion: Deception and unpredictability are intended to increase adversaries' uncertainty – about the system's structure and behavior, about what effects an adversary might be able to achieve, and about what actions cyber defenders might take in response to suspected MCA. See [64] for a detailed discussion of deception and its role in active cyber defense. Deception includes obfuscation, which increases the effort needed by the adversary, and can hide mission activities long enough for the mission to complete without adversary disruption. Active deception can divert adversary activities, causing the adversary to waste resources and reveal TTPs, intent, and targeting.

Unpredictability can apply to characteristics, structure, or behavior. Unpredictable characteristics (e.g., configurations, selection of an equivalent element from a diverse set) force the adversary to develop a broader range of TTPs. Unpredictable structure (e.g., dynamically changing partitions or isolating components) undermine the adversary's reconnaissance efforts. Unpredictable behavior (e.g., response latency) increases uncertainty about effects and about whether system behavior indicates defender awareness of MCA. Unpredictability and deception can be applied separately, as well as synergistically [93]. These two techniques can be highly effective against advanced adversaries.

However, deception and unpredictability, if implemented poorly, can also increase the uncertainty of end users and administrators about how the system will behave. User and administrator confusion reduces overall resilience, reliability, and security. This uncertainty can in turn make detection of unauthorized or suspicious behavior more difficult. This design principle calls for a sound implementation, which makes system behaviors directed at the adversary transparent to end users and to administrative users.

Table 18. Examples of Restatements of *Make Unpredictability and Deception User-Transparent*

Example of Restatement	Examples of Questions for Analysis	Examples of Metrics
Obfuscate or hide data and evidence of system behavior.	How much effort is required to obtain hidden data? How difficult is it to locate, collect, and/or decrypt? Which system behaviors are concealed? How does the system defend against traffic analysis and performance analysis? Do the obfuscation or concealment mechanisms impede system administrators and cyber defenders from obtaining the information they need to investigate suspicious behavior?	Number of resource types which can be obfuscated or hidden; percentage of mission-critical resources which are concealed or obfuscated [A] Minimum and average time to obtain or infer information about concealed resources by a legitimate administrator or cyber defender [T], by an adversary [RT]
Provide mechanisms to support active deception strategies, as defined and applied by cyber defenders.	What deception strategies does the system architecture support? What level of effort is required to maintain the deception? How effective are the deception strategies the system provides? How is effectiveness determined? Do or can the deception mechanisms entrap or fool legitimate system users?	Minimum, average, and maximum dwell time of an adversary in a deception environment [M, RT] Average number of times during a mission operation (or over a fixed time period) a legitimate user is entrapped or fooled [T, O]

Example of Restatement	Examples of Questions for Analysis	Examples of Metrics
Incorporate features whose behaviors (e.g., random delays in responses) or characteristics (e.g., address randomization) are unpredictable.	What behaviors or characteristics are unpredictable by design? What behaviors or characteristics can be made unpredictable by the actions of system administrators or cyber defenders? Do or can the unpredictability mechanisms lead legitimate system users into making errors (e.g., re-entering a command or data that should only be entered once)? Do or can the unpredictability mechanisms lead system administrators or cyber defenders into making errors (e.g., assuming that suspicious behavior is the effect of the unpredictability mechanisms, treating the behavior of the unpredictability mechanisms as suspicious)?	Average number of times during a mission operation (or over a fixed time period) a legitimate user is confused due to unpredictability [T, O] Average number of times during a mission operation (or over a fixed time period) a system administrator or cyber defender makes an error attributable to unpredictability [T, O]

2.3 Cyber Resiliency Design Principles, Objectives, and Techniques

The activities from which the design principles presented in Sections 2.1 and 2.2 used the CREF as a common structuring mechanism, but found in many cases that CREF-based design principles as defined in Appendix B.1 were too generic. However, as Table 19 indicates, the representative design principles can be mapped to the CREF: they support different cyber resiliency objectives, and call for the use of different cyber resiliency techniques.

All the cyber resiliency objectives are supported by multiple design principles. However, two objectives are supported to a lesser extent: Prepare and Re-Architect. Prepare involves creating and maintaining a set of realistic courses of action, which are based on the architecture, design, and implementation, rather than driving them. Re-Architect is supported by most of the strategic design principles, and (in conjunction with the organization's risk management strategy) drives the selection of structural design principles. Similarly, all the cyber resiliency techniques are used by multiple design principles.

Table 19. Mapping Cyber Resiliency Design Principles to Objectives and Techniques

Cyber Resiliency Design Principle	Objective								Technique													
	Understand	Prepare	Prevent / Avoid	Continue	Constrain	Reconstitute	Transform	Re-Architect	Adaptive Response	Analytic Monitoring	Coordinated Defense	Deception	Diversity	Dynamic Positioning	Dynamic Representation	Non-Persistence	Privilege Restriction	Realignment	Redundancy	Segmentation / Isolation	Substantiated Integrity	Unpredictability
Focus on common critical assets	S		S	S		S		S		U	U		U		U	U		R		U	U	
Support agility and architect for adaptability		S				S	S	S	U				U	U		U		R	U			
Reduce attack surfaces	S		S		S		S	S										U		U		
Assume compromised resources		S		S	S		S	S	U						U	U	U	U		U		
Expect adversaries to evolve	S	S					S	S		U		U	U		U			U				U
Limit the need for trust			S		S		S				U						R	U			U	
Control visibility and use			S	S	S							U				U	R			U		
Contain and exclude behaviors				S	S			S	U	U			U			U	U			R	U	
Layer and partition defenses			S	S				S		U	R		U	U					U	R		
Plan and manage diversity		S		S	S	S					R		R						U			X
Maintain redundancy				S	S	S					U							U	R				
Make resources location-versatile				S	S	S			U				U	R		U			U			U
Leverage health and status data	S						S			R					R						U	
Maintain situational awareness	S						S			R					R							
Manage resources (risk-) adaptively		S		S		S			R	U	U	U		U			U	U	U	U		U
Maximize transience; minimize persistence			S		S				U							R					U	U
Determine ongoing trustworthiness	S		S	S	S	S					U										R	
Change or disrupt the attack surface			S						R			U	U	U		U						U
Make unpredictability and deception user-transparent	S		S								R	R										R

Key:

S – Supports achieving the objective

R – **Requires** use of cyber resiliency technique (or one or more approaches)

U – **Can use** cyber resiliency technique (or one or more approaches)

X – **Can be combined with** use of cyber resiliency technique (or one or more approaches) to achieve specific effects on adversary

3 Applying Cyber Resiliency Design Principles

When applying cyber resiliency design principles, systems engineers and architects need to consider the environment in which those principles will be used. Representative environmental factors are discussed in Section 3.1; these factors can determine which design principles are appropriate, and can also support the tailoring of design principle statements to be more understandable and meaningful in the target environment. In selecting, tailoring, or de-selecting cyber resiliency design principles, systems engineers and architects also need to take stakeholder priorities into consideration; this topic is briefly discussed in Section 3.2.

A given system, program, or system-of-systems needs to meet multiple design objectives, not simply those related to cyber resiliency. Therefore, cyber resiliency design principles must be selected, tailored, and applied in conjunction with design principles for related disciplines, including cybersecurity, survivability, and evolvability. The relationships between design principles for those disciplines and cyber resiliency design principles are discussed in Section 3.3, with supporting details in Appendix C. Finally, depending on how an organization has defined or articulated its risk management strategy, the selection and tailoring of cyber resiliency design principles can be driven by that strategy. Some examples are provided in Section 3.4. Figure 2 provides a roadmap to this section.

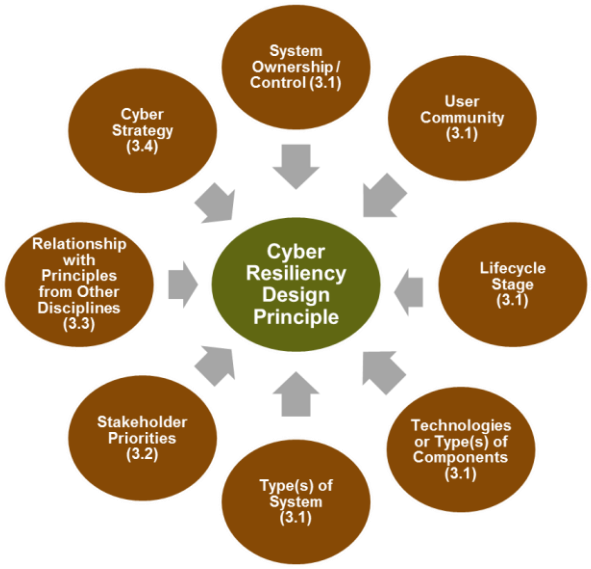

Figure 2. Factors to Consider in Selecting and Applying Cyber Resiliency Design Principles

3.1 Environmental Factors

Systems engineers and architects need to consider a variety of factors when deciding which cyber resiliency design principles are appropriate for a system, program, or system-of-systems. Factors to consider relate to the environment in which the selected design principles will be applied. Environmental factors, with ranges of values defined for purposes of characterizing design principle applicability, include:

- *Life-cycle stage*[31]. Five stages are identified: (i) requirements development (pre-Milestone [MS] A), (ii) preliminary design (pre-MS B), (iii) detailed design and implementation (pre-MS C), (iv) production and deployment (pre-IOC), and (v) operations and support (post-IOC).

[31] Life-cycle stages are identified using the DoD Acquisition Lifecycle [128].

- Pre-MS A, design principles are articulated based on strategic objectives, particularly on strategies for managing mission and organizational risks due to dependence on the system.
- Pre-MS B, design principles are applied to the architecture and refined: Design reviews are supported by explanations of how the design applies the design principles. Design principles from multiple disciplines or problem domains are aligned; where conflicts exist, heuristics for making trade-offs are identified. Requirements are defined and allocated based on design principles.
- Pre-MS C, design principles are applied to the detailed design and implementation, and the results of the application are documented: Design reviews and analyses are supported by explanations of how – and how effectively – the design applies the design principles. Trade-offs among applications of conflicting design principles from multiple disciplines or problem domains are explained; the resulting risks or concerns identified, resolved, or tracked.
- Pre-IOC, the application (and corresponding documentation) of design principles continues, with an increasing emphasis on implementation of the principles via procedures and other environmental controls. Program reviews and analyses (including analyses of test results) are supported by explanations of how – and how effectively – the implementation and supporting procedures apply the design principles. Trade-offs and resulting risks or concerns are identified, resolved, or tracked.
- Post-IOC, design principles continue to be applied. The emphasis is on implementation of the principles via procedures and other environmental controls. However, design principles are also applied to maintenance processes and system upgrades. Trade-offs and resulting risks or concerns continue to be identified, resolved, or tracked.

- *Type of system or system-of-systems*. Four broad (and non-disjoint) classes are identified: (i) enterprise IT; (ii) shared service; (iii) common infrastructure; and (iv) embedded systems, including platform IT (PIT), operational technology (OT), industrial control systems (ICS), and cyber-physical systems (CPS).
- *Type(s) of system components*. Three broad classes are identified: (i) commodity components, including general-purpose IT platforms (e.g., desktops, laptops, tablets) and applications (e.g., database management systems), general-purpose networking (e.g., routers), commercial off-the-shelf (COTS) specialized subsystems (e.g., intrusion detection systems), and COTS SCADA or embedded components; (ii) custom-developed components; and (iii) virtualized platforms and cloud computing infrastructures, which can be either commoditized or customized.
- *System ownership / control*. Three broad classes are identified: (i) single owner / operator (O/O), (ii) federated group of O/Os under a common governance structure (e.g., in a cloud environment), and (iii) loosely federated group of O/Os.

The strategic cyber resiliency design principles are largely insensitive to these factors; selection of strategic design principles is driven by stakeholder priorities and beliefs, as well as the organization's risk management strategy. Table 20 provides examples of how these factors apply to the structural cyber resiliency design principles identified in Table 1. Table 20 is not exhaustive; it is intended to indicate how relevant the different principles may be in different environments.

An additional factor is *user community*. Three broad classes can be identified: (i) organization-internal (e.g., employees), (ii) organization-identified (e.g., issued organizational credentials), and (iii) self-identified (e.g., members of the general public). The following cyber resiliency design principles are of particular concern if the user community is self-identified: *Limit the need for trust, Control visibility and use, Contain and exclude behaviors, Layer and partition defenses, Maintain situational awareness, Maximize transience, Change or disrupt the attack surface,* and *Make deception and unpredictability user-transparent*.

Table 20. Environmental Factors Influencing the Use of Cyber Resiliency Structural Design Principles

Structural Design Principle	Lifecycle Stages	Type(s) of Systems	Type(s) of Components	System Ownership / Control
Limit the need for trust.	All	All. Potentially challenging for embedded.	Commodity and custom-developed. Can add cost to custom-developed.	Single O/O. Challenging to define privileges / basis for trust consistently in federated, possibly impossible in loosely federated.
Control visibility and use.	Pre-MS C. Implies commitment to sustain.	All. Potentially challenging for embedded.	Commodity and cloud / virtualized. Potentially feasible with custom-developed but may present technical challenges.	Single O/O; may be compatible with federated O/Os; highly challenging in loose federations.
Contain and exclude behaviors.	Pre-MS C. Implies commitment to sustain.	All, though runs counter to convergence trend. Potentially challenging for embedded.	Feasible for commodity, highly compatible with custom-developed, highly compatible with cloud / virtualized environments.	Highly relevant to reduce risk in loose federations but may run counter to convergence trends; relevant in federations and in large or complex single O/O.
Layer and partition defenses.	Pre-MS C.	All. Potentially challenging for embedded.	Commodity and custom-developed. Can add cost to custom-developed.	Single O/O. Challenging to operate consistently in federated, possibly impossible in loosely federated
Plan and manage diversity.	All. Note that in later stages, may be applied more to people and processes than to technology.	All. Technical and governance challenges for EIT, shared services, common infrastructures. Adds cost, complexity. Potentially challenging for embedded.	Challenging for commodity (including cloud / virtualization elements). Feasible but can add cost and complexity to custom-developed.	Single O/O. Unplanned diversity is often a side effect of federation or loose federation, but presents technical and governance challenges to manage.
Maintain redundancy.	Pre-MS C. Implies commitment to sustain.	All. Adds cost, complexity.	Easy for commodity, feasible but can add cost and complexity to custom-developed, can be compatible with virtualized / cloud environments.	Single O/O. Unplanned redundancy is often a side effect of federation or loose federation, but presents technical and governance challenges to maintain.
Make resources location-versatile.	Pre-MS C. Implies commitment to sustain.	All. Adds cost, complexity. Potentially challenging for embedded.	Often feasible for commodity, potentially feasible with custom-developed but may present technical challenges, highly compatible with virtualized / cloud environments.	Usually a side effect in loose federations and in federations, relevant in large or complex single O/O.

Structural Design Principle	Lifecycle Stages	Type(s) of Systems	Type(s) of Components	System Ownership / Control
Leverage health and status data.	All.	All, but highly relevant to embedded.	All.	Single O/O. Challenging in federated, possibly impossible in loosely federated.
Maintain situational awareness.	Pre-MS C. Implies commitment to sustain.	All. Potentially challenging for embedded.	Often feasible for commodity, potentially feasible with custom-developed but may present technical challenges, technical challenges for virtualized / cloud environments.	Highly compatible with single O/O; may be compatible with federated O/Os; highly challenging in loose federations.
Manage resources (risk-) adaptively.	Pre-MS C. Implies commitment to sustain.	All. Potentially challenging for embedded.	Often feasible for commodity, potentially feasible with custom-developed if capabilities are built in, highly compatible with virtualized / cloud environments.	Usually a side effect in loose federations, relevant in federations and in large or complex single O/O.
Maximize transience.	Pre-MS C. Implies commitment to sustain.	All. Potentially very challenging for embedded.	Potentially feasible for commodity, potentially feasible with custom-developed, highly compatible with virtualized / cloud environments.	Highly relevant to reduce risk in loose federations but may run counter to convergence trends, relevant in federations and in large or complex single O/O.
Determine ongoing trustworthiness.	Pre-MS C.	All. Potentially challenging for embedded.	Challenging for commodity, potentially feasible with custom-developed but may present technical challenges, independent of whether environment is virtualized / cloud.	Highly compatible with single O/O; may be compatible with federated O/Os; highly challenging in loose federations.
Change or disrupt the attack surface.	Pre-MS C. Implies commitment to sustain.	All. Potentially very challenging for embedded.	Highly challenging for commodity, potentially feasible with custom-developed but may present technical challenges, facilitated by virtualized / cloud environment.	Highly compatible with single O/O; may be compatible with federated O/Os; highly challenging in loose federations (may occur incidentally).
Make deception and unpredictability user-transparent.	Pre-MS C. Implies strong commitment to sustain.	All. May be challenging for shared services and common infrastructures. Potentially very challenging for embedded.	Challenging for commodity, potentially feasible with custom-developed but may present technical challenges, can be highly compatible with virtualized / cloud environments.	Highly compatible with single O/O; may be compatible with federated O/Os but presents practical and governance challenges; highly challenging or infeasible in loose federations.

It should be noted that the environmental factors used to select design principles are inherently different from those used to select, de-select, tailor, or augment security controls. See [94] for a discussion of such control-related factors.

3.2 Stakeholder Priorities

Different cyber resiliency design principles support different cyber resiliency objectives. As noted in [4], different stakeholders prioritize cyber resiliency goals and objectives differently. For example, a mission commander or business process manager, who is responsible for execution of a specific mission or business process in a given timeframe, can emphasize the Withstand goal, and the Continue and Constrain objectives, at the expense of all others. The mission or business process owner, who is responsible not only for the success of the current mission or business process but also for future viability, also considers the Recover goal and the Reconstitute objective as well as the Anticipate goal and the Prepare and Prevent / Avoid objectives, and may prioritize these more highly than Withstand, Continue, and Constrain in some circumstances. Cyber defenders are more likely to emphasize Anticipate (with Understand as well as Prepare and Prevent / Avoid) and to a lesser extent Evolve (with Understand and Transform as the higher priority objectives), while systems engineers and architects are more likely to emphasize Evolve, with Re-Architect and Transform as higher priority. Figure 3 illustrates how different stakeholder objectives highlight different high-level design principles expressed in terms of cyber resiliency objectives (see Appendix B.1) and thereby different specialty disciplines.

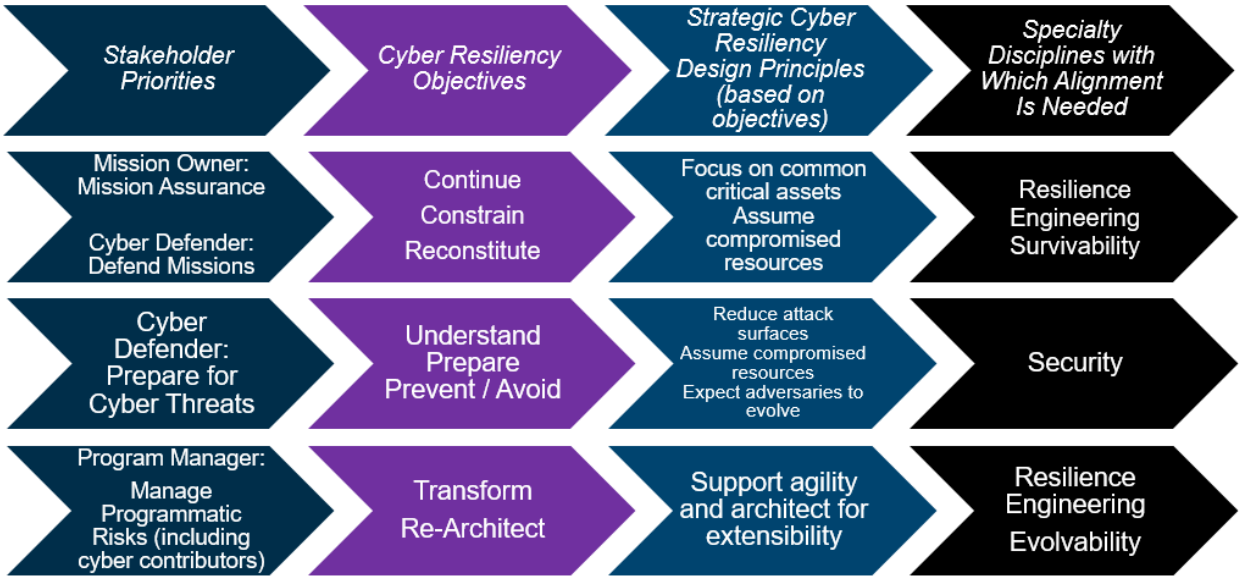

Figure 3. Stakeholder Priorities Highlight Cyber Resiliency Objectives and Corresponding High-Level Design Principles

In addition to the environmental factors identified in Section 3.1, other political, operational, economic, and technical (POET) factors can make some cyber resiliency techniques more or less relevant to a given program, system, or system-of-systems. See Appendix E of [4] for representative POET factors.

A cyber resiliency analysis elicits stakeholder priorities, and produces a common picture of the relative priorities of the cyber resiliency goals and objectives, as well as of the relative relevance of the cyber resiliency techniques. These preferences can be used in determining which cyber resiliency design principles to select or de-select: The structural cyber resiliency design principles are mapped to cyber resiliency objectives and techniques directly in Table 19.

3.3 Design Principles from Related Specialty Disciplines

As noted in Section 1.1, design principles can be drawn from a variety of specialty disciplines. The task for architects and systems engineers is to select, tailor, and make trade-offs among the design principles from those disciplines in order to guide analysis and design decisions.

Cyber resiliency builds upon and is informed by a variety of related specialty disciplines, including cybersecurity, resilience engineering, survivability, and evolvability for systems-of-systems. However, some design principles from those disciplines conflict with some cyber resiliency design principles, while others provide strong synergy with specific cyber resiliency design principles, can support specific cyber resiliency objectives, or can make effective use of cyber resiliency techniques. This section provides a brief discussion of design principles from those disciplines and how they relate to cyber resiliency. See Appendix C for details.

3.3.1 Security

Cyber resiliency assumes a foundation of good cybersecurity practices. Security design principles have been variously articulated; see Appendix C.1. Many of the cyber resiliency design principles are consistent with (and could be used with or as alternatives to) security design principles. In particular, and as indicated in Table 1, two strategic principles (*Reduce attack surfaces* and *Assume compromised resources*) and four structural principles (*Limit the need for trust*, *Control visibility and use*, *Contain and exclude behaviors*, and *Layer and partition defenses*) are strongly aligned with security design principles.

However, security design principles do not support achievement of the full range of cyber resiliency objectives, nor do they apply the full range of cyber resiliency techniques. Security design principles focus on the *Prevent / Avoid* and to a lesser extent the *Understand* cyber resiliency objectives, either primarily or secondarily. (Some also relate to the *Transform* and *Re-Architect* objectives.) This is due to the difference in threat models: security design principles primarily assume a conventional adversary (e.g., an insider, an intruder who can be detected and repulsed), while cyber resiliency design principles assume an advanced cyber threat.

3.3.2 Resilience Engineering and Survivability

Cyber resiliency assumes a foundation of good practice in overall system resilience, including contingency planning [3]. The focus of Resilience Engineering is on full or partial recovery following a threat that disrupts system functionality. Key attributes are capacity, flexibility, tolerance, and cohesion. Similarly, the discipline of Survivability typically assumes a finite-duration disturbance, rather than the potentially unbounded duration of cyber adversary activities within a compromised system.

The cyber resiliency design principles in Table 1 collectively support Resilience Engineering and Survivability design principles, specifically in the context of cyber threats. In particular, and as indicated in Table 1, two strategic principles (*Focus on critical assets* and *Support agility and architect for adaptability*) and three structural principles (*Plan and manage diversity*, *Maintain redundancy*, and *Make resources location-versatile*) are strongly aligned with Resilience Engineering and Survivability design principles. However, due to differences in the underlying threat models, the Resilience Engineering design principles (and other design principles related to resilience and survivability) do not cover the full set of cyber resiliency concerns. In particular, design principles for Resilience Engineering and Survivability do not take an advanced and evolving adversary into consideration.

3.3.3 Evolvability, Anti-Fragility, and Changeability

Consistent with the goals defined for resilience engineering [95], one of the cyber resiliency goals is "evolve" (or "adapt"). The need for systems to be able to evolve has long been recognized in the context of security and survivability [96]. Design principles for evolvable systems-of-systems (SoS), developed

for the military domain, are more broadly applicable [15]. Many evolvability design align well with cyber resiliency design principles; some are only indirectly related; and one is problematic in the context of cybersecurity and cyber resiliency (see C.4 for details). As with design principles from other specialty disciplines, the evolvability design principles do not cover the full range of cyber resiliency objectives and techniques.

A closely related concept is that of anti-fragility, in which a system changes its behavior based on the circumstances of its use [97]. In the anti-fragility literature, the property focuses on the ability to evolve and improve; robustness is identified with the ability to withstand and resilience with the ability to recover. Since cyber resiliency includes all four goals (anticipate, withstand, recover, and evolve), anti-fragility in the context of cyber threats cannot be easily differentiated from cyber resiliency. Research into the concept is ongoing, with a few alternatives for design principles offered. These include openness, feedback loops, independent supervising authorities, and highly reliable and resilient components [98]; and modularity, weak links (to limit failure propagation), redundancy, diversity, and fast failure [99].

Another closely related concept is that of changeability. In an ongoing analysis of system qualities and trade-offs [100], resilience is achieved through dependability and changeability. Changeability in turn is achieved through adaptability and maintainability; the latter includes repair-ability, valid-ability, and modifiability. Some of the statements of opportunity for modifiability strongly resemble resilience design principles [100].

3.4 Design Principles as Expressions of a Risk Management Strategy

Design principles can be viewed as expressions of an organization's (or a mission's or a program's) risk management strategy. An organization's risk management strategy expresses its risk frame, defines how the organization will perform risk assessment, determines which risk responses it will take for different levels and types of risk (subject to the constraints, risk tolerance, and priorities and trade-offs captured in its risk frame), and establishes how risks will be monitored (including monitoring of indicators or other relevant information) [36]. As illustrated in Figure 4, for purposes of selecting design principles, a few aspects of an organization's risk management strategy are key; these are indicated in yellow italics.

Figure 4. Aspects of Risk Management Strategy Relevant to Selection of Design Principles

As illustrated in Figure 5, the risk assumptions in the organization's risk frame highlight strategic cyber resiliency design principles specific to consideration of advanced cyber threats, while risk mitigation approaches aligned with Resilience Engineering and Survivability highlight strategic cyber resiliency design principles consistent with constraints.

An organization's risk management strategy can also lead it to state explicitly the relative priorities of the cyber resiliency goals and objectives (see [3] and Appendix B.1), and to translate these into statements of strategy.

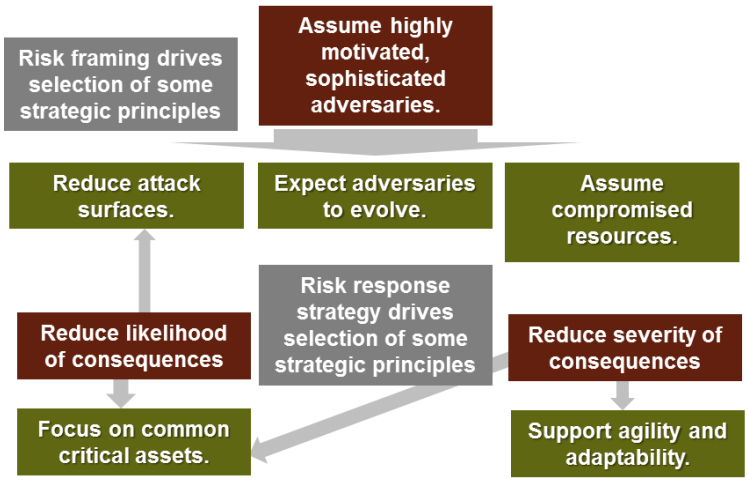

Figure 5. The Risk Management Strategy Highlights Different Strategic Design Principles

An organization's risk management strategy can lead it to develop specific strategies for improving threat intelligence and actively defending against advanced cyber adversaries. These include use of forensic analysis, active deception, and threat information sharing. These strategies lead to requirements for defensive capabilities which may result in requirements for a specific system or program. (For example, the organization can define requirements for a deception environment, which can lead to interface requirements for mission system so that the deception environment can be kept realistic.) These strategies do not *per se* imply strategic design principles. However, these strategies do lead to structural design principles to support the required defensive capabilities:

- Data capture to support forensic analysis: Leverage health and status data
- Incorporation of requirements related to deception or unpredictability: Make unpredictability and deception user-transparent

In addition, analytic resources (vocabularies, frameworks, metrics) to determine effects on adversary activities can be used in conjunction with analytic resources that support strategic or structural design principles. See Appendix H of [20].

4 Conclusion

Today's mission systems – including systems-of-systems – are often the products of large-scale efforts, involving many developers, integrators, configuration managers and system administrators, and maintenance staff over an extended period. In addition, many systems have a long lifespan. Design principles provide a common touchstone, enabling those involved in constructing, operating, and maintaining a system to understand the precepts that guided – and must continue to guide – the system's architecture, design, and implementation. As concern increases for cyber resiliency (or system resiliency, when malicious cyber activities are explicitly considered as a form of adversity), so does the need to include cyber resiliency design principles in a program's or a system's set of design principles, and to reflect cyber resiliency in the corresponding Program Protection Plan or Security Plan.

This paper has presented a representative set of cyber resiliency design principles to serve as a starting point. Cyber resiliency must be achieved in concert with other emergent system properties, including security, survivability, and evolvability. This paper has articulated relationships between design principles developed by those disciplines and cyber resiliency. No design principle is universally applicable. Therefore, this paper has shown how analysis of the context in which a system, system-of-systems, or mission architecture in which cyber resiliency is needed can produce a relevant and useful set of cyber resiliency design principles, to guide design and implementation decisions.

5 References

[1] NIST, "NIST SP 800-160, Systems Security Engineering: Considerations for a Multidisciplinary Approach in the Engineering of Trustworthy Secure Systems," 15 November 2016. [Online]. Available: http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-160.pdf.

[2] D. Bodeau and R. Graubart, "Cyber Resiliency Engineering Framework (MTR110237, PR 11-4436)," September 2011. [Online]. Available: http://www.mitre.org/sites/default/files/pdf/11_4436.pdf.

[3] D. Bodeau, R. Graubart, W. Heinbockel and E. Laderman, "Cyber Resiliency Engineering Aid - The Updated Cyber Resiliency Engineering Framework and Guidance on Applying Cyber Resiliency Techniques, MTR140499R1, PR 15-1334," May 2015. [Online]. Available: http://www.mitre.org/sites/default/files/publications/pr-15-1334-cyber-resiliency-engineering-aid-framework-update.pdf or http://www.defenseinnovationmarketplace.mil/resources/20150527_Cyber_Resiliency_Engineering_Aid-Cyber_Resiliency_Techniques.pdf.

[4] D. Bodeau and R. Graubart, "Cyber Resiliency Assessment: Enabling Architectural Improvement (MTR 120407, PR 12-3795)," May 2013. [Online]. Available: http://www.mitre.org/sites/default/files/pdf/12_3795.pdf.

[5] D. Bodeau and R. Graubart, "Structured Cyber Resiliency Analysis Methodology (SCRAM) (PR Case No. 16-0777)," May 2016. [Online]. Available: https://www.mitre.org/publications/technical-papers/structured-cyber-resiliency-analysis-methodology.

[6] D. Bodeau, Graubart and Richard, "Cyber Resiliency and NIST Special Publication 800-53 Rev.4 Controls (MTR 130531, PR 13-4037)," September 2013. [Online]. Available: http://www.mitre.org/sites/default/files/publications/13-4047.pdf.

[7] R. Graubart and D. Bodeau, "The Risk Management Framework and Cyber Resiliency (PR Case No. 16-0776)," May 2016. [Online]. Available: https://www.mitre.org/publications/technical-papers/the-risk-management-framework-and-cyber-resiliency.

[8] The MITRE Corporation (ed.), "2nd Secure and Resilient Cyber Architectures Workshop: Final Report," 2012. [Online]. Available: https://registerdev1.mitre.org/sr/2012_resiliency_workshop_report.pdf.

[9] The MITRE Corporation (ed.), "2015 Secure and Resilient Cyber Architectures Invitational (PR case no. 16-1199)," May 2016. [Online]. Available: http://www2.mitre.org/public/sr/2015-Secure-and-Resilient-Cyber-Architectures-Report-16-1199.pdf.

[10] The MITRE Corporation (ed.), "6th Annual Secure and Resilient Cyber Architectures Invitational," April 2017. [Online]. Available: https://www.mitre.org/sites/default/files/publications/pr-17-0914-proceedings-of-sixth-annual-secure-and-resilient-cyber-architectures-invitational.pdf.

[11] The MITRE Corporation, "Cyber Resiliency Resource List," May 2016. [Online]. Available: http://www2.mitre.org/public/sr/Cyber-Resiliency-Resources-16-1467.pdf.

[12] DoD Defense Science Board, "Task Force Report: Resilient Military Systems and the Advanced Cyber Threat," January 2013. [Online]. Available: http://www.acq.osd.mil/dsb/reports/ResilientMilitarySystems.CyberThreat.pdf.

[13] SEBoK, "Resilience Engineering," Systems Engineering Body of Knowledge, 2 July 2015. [Online]. Available: http://sebokwiki.org/wiki/Resilience_Engineering.

[14] M. G. Richards, A. M. Ross, D. E. Hastings and D. H. Rhodes, "Empirical Validation of Design Principles for Survivable System Architecture," in *Proceedings of the 2nd Annual IEEE Systems Conference*, Montreal, Quebec, Canada, 2008.

[15] N. Ricci, D. H. Rhodes and A. M. Ross, "Evolvability-Related Options in Military Systems of Systems," in *Conference on Systems Engineering Research (CSER 2014)*, Redondo Beach, CA, 2014.

[16] NIST, "Guide for Applying the Risk Management Framework to Federal Information Systems, NIST SP 800-37 Rev. 1," February 2010. [Online]. Available: http://csrc.nist.gov/publications/nistpubs/800-37-rev1/sp800-37-rev1-final.pdf.

[17] DoD CIO, "DoDI 8510.01, Risk Management Framework (RMF) for DoD Information Technology (IT)," 12 March 2014. [Online]. Available: http://www.dtic.mil/whs/directives/corres/pdf/851001_2014.pdf.

[18] DoD, "DoD Program Manager's Guidebook for Integrating the Cybersecurity Risk Management Framework (RMF) into the System Acquisition Lifecycle, Version 1.0," 26 May 2015. [Online]. Available: https://acc.dau.mil/adl/en-US/722603/file/80119/Cybersecurity%20Guidebook%20v1_0%20with%20publication%20notice.pdf.

[19] IEEE, "The 1st IEEE Workshop on Cyber Resiliency Economics (CRE16)," August 2016. [Online]. Available: http://paris.utdallas.edu/cre16/.

[20] NIST, "2nd Public Draft, NIST SP 800-160, Systems Security Engineering: Considerations for a Multidisciplinary Approach in the Engineering of Trustworthy Secure Systems," 4 May 2016. [Online]. Available: http://csrc.nist.gov/publications/drafts/800-160/sp800_160_second-draft.pdf.

[21] DoD CIO, "DoDI 8500.01, Cybersecurity," 14 March 2014. [Online]. Available: http://www.dtic.mil/whs/directives/corres/pdf/850001_2014.pdf.

[22] NIST, "Framework for Improving Critical Infrastructure Security, Version 1.0," 12 February 2014. [Online]. Available: http://www.nist.gov/cyberframework/upload/cybersecurity-framework-021214.pdf.

[23] DHS, "Assessments: Cyber Resilience Review (CRR)," US-CERT, [Online]. Available: https://www.us-cert.gov/ccubedvp/assessments.

[24] CERT Program, "CERT® Resilience Management Model, Version 1.0: Improving Operational Resilience Processes," May 2010. [Online]. Available: http://www.cert.org/archive/pdf/10tr012.pdf.

[25] CRO Forum, "Cyber resilience: The cyber risk challenge and the role of insurance," December 2014. [Online]. Available: http://www.thecroforum.org/wp-content/uploads/2014/12/Cyber-Risk-Paper-version-24.pdf.

[26] Bank for International Settlements and International Organization of Securities Commissions, "Guidance on cyber resilience for financial market infrastructures," June 2016. [Online]. Available: https://www.bis.org/cpmi/publ/d146.pdf.

[27] Global Forum to Advance Cyber Resilience, "Global Forum to Advance Cyber Resilience," 2016. [Online]. Available: http://gfacr.org/.

[28] The MITRE Corporation, "Cyber Prep 2.0: Motivating Organizational Cyber Strategies in Terms of Preparedness (PR 15-0837)," The MITRE Corporation, Bedford, MA, 2015.

[29] NIST, "Security and Privacy Controls for Federal Information Systems and Organizations (NIST SP 800-53 R4)," April 2013. [Online]. Available: http://dx.doi.org/10.6028/NIST.SP.800-53r4.

[30] NIST, "Glossary of Key Information Security Terms, NISTIR 7298, Revision 2," May 2013. [Online]. Available: http://nvlpubs.nist.gov/nistpubs/ir/2013/NIST.IR.7298r2.pdf.

[31] CNSS, "Committe on National Security Systems (CNSS) Glossary (CNSS Instruction No. 4009)," 26 April 2015. [Online]. Available: https://www.cnss.gov/CNSS/openDoc.cfm?hldYMe6UHW4ISXb8GFGURw==.

[32] INCOSE, INCOSE Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities, 4th Edition, INCOSE-TP-2003-002-04, San Diego, CA: INCOSE, 2015.

[33] IFIP, "IFIP Working Group 10.4 on Dependable Computing and Fault Tolerance," 2014. [Online]. Available: http://www.dependability.org/wg10.4/.

[34] CPS PWG, "Framework for Cyber-Physical Systems, Release 1.0," May 2016. [Online]. Available: https://s3.amazonaws.com/nist-sgcps/cpspwg/files/pwgglobal/CPS_PWG_Framework_for_Cyber_Physical_Systems_Release_1_0Final.pdf.

[35] National Computer Security Center, "A Guide to Understanding Security Modeling in Trusted Systems, NCSC-TG-10 Version 1," National Computer Security Center, Fort Meade, MD, 1992.

[36] NIST, "NIST SP 800-39, Managing Information Security Risk: Organization, Mission, and Information System View," March 2011. [Online]. Available: http://csrc.nist.gov/publications/nistpubs/800-39/SP800-39-final.pdf.

[37] DoD CIO/USD(AT&L), "Protection of Mission Critical Functions to Achieve Trusted Systems and Networks (TSN), DoDI 5200.44," 5 November 2012. [Online]. Available: http://www.dtic.mil/whs/directives/corres/pdf/520044p.pdf.

[38] The MITRE Corporation, "Systems Engineering Guide: Crown Jewels Analysis," 2011. [Online]. Available: http://www.mitre.org/publications/systems-engineering-guide/enterprise-engineering/systems-engineering-for-mission-assurance/crown-jewels-analysis.

[39] S. Musman and A. Temin, "A Cyber Mission Impact Assessment Tool (PR 14-3545)," in *2015 IEEE International Symposium on Technologies for Homeland Security (HST)*, Waltham, MA, 2015.

[40] J. A. Pendergrass, S. C. Lee and C. D. McDonell, "Theory and Practice of Mechanized Software Analysis," *Johns Hopkins APL Technical Digest*, vol. 32, no. 2, pp. 499-508, 2013.

[41] NAVAIR, "Cyber Failure Mode, Effects, and Criticality Analysis (FMECA) Methodology, SWP4000-001," NAVAIR, 2014.

[42] P. R. Garvey and C. A. Pinto, Advanced Risk Analysis in Engineering Enterprise Systems, New York, NY: CRC Press, 2012.

[43] T. Llanso, P. A. Hamilton and M. Silberglitt, "MAAP: Mission Assurance Analytics Platform," in *2012 IEEE Conference on Technologies for Homeland Security (HST)*, Boston, MA, 2012.

[44] C. Woody and C. Alberts, "Evaluating Security Risks Using Mission Threads," *CrossTalk*, pp. 15-19, September / October 2014.

[45] NIST, "NIST SP 800-84, Guide to Test, Training, and Exercise Programs for IT Plans and Capabilities," September 2006. [Online]. Available: http://csrc.nist.gov/publications/nistpubs/800-84/SP800-84.pdf.

[46] S. Musman, "Playing the Cyber Security Game: A Rational A pproach to Cyber Security and Resilience Decision Making (MTR 150371, PR 15-3140)," The MITRE Corporation, McLean, VA, 2016.

[47] The MITRE Corporation, "Adversarial Tactics, Techniques, and Common Knowledge (ATT&CK™)," The MITRE Corporation, 2015. [Online]. Available: https://attack.mitre.org/wiki/Main_Page.

[48] The MITRE Corporation, "CAPEC: Common Attack Pattern Enumeration and Classification," 2013. [Online]. Available: http://capec.mitre.org/.

[49] NIST, "NIST SP 800-34 Rev. 1, Contingency Planning Guide for Federal Information Systems," 11 November 2010. [Online]. Available: http://csrc.nist.gov/publications/nistpubs/800-34-rev1/sp800-34-rev1_errata-Nov11-2010.pdf. [Accessed 19 May 2011].

[50] R. A. Caralli, J. H. Allen, D. W. White, L. R. Young, N. Mehravari and P. D. Curtis, "CERT® Resilience Management Model, Version 1.2," February 2016. [Online]. Available: http://www.cert.org/downloads/resilience/assets/cert-rmm-v1-2.pdf.

[51] DoD DASD(SE), "Program Protection Plan Outline & Guidance, Version 1.0," July 2011. [Online]. Available: http://www.acq.osd.mil/se/docs/PPP-Outline-and-Guidance-v1-July2011.pdf.

[52] S. Jackson, "A Multidisciplinary Framework for Resilience to Disasters and Disruptions," *Journal of Integrated Design and Process Science*, vol. 11, no. 2, pp. 91-108, 2007.

[53] S. Sheard, "A Framework for System Resilience Discussions," in *INCOSE International Symposium, 18*, Utrecht, the Netherlands, Wiley, 2008, p. 1243–1257.

[54] S. King, "National and Defense S&T Strategies & Initiatives," 25-26 July 2012. [Online]. Available: http://www.cyber.st.dhs.gov/wp-content/uploads/2012/08/Dr_Steven_King-_ASD_RE.pdf.

[55] P. McDaniel, T. Jaeger, T. F. La Porta, N. Papernot, R. J. Walls, A. Kott, L. Marvel, A. Swami, P. Mohapatra, S. V. Krishnamurthy and I. Nametiu, "Security and Science of Agility," in *Proceedings of the First ACM Workshop on Moving Target Defense (MTD'14)*, Scottsdale, AZ, 2014.

[56] P. Beraud, A. Cruz, S. Hassell and S. Meadows, "Using Cyber Maneuver to Improve Network Resiliency," in *MILCOM*, Baltimore, MD, 2011.

[57] DoD, "Department of Defense Cybersecurity Test and Evaluation Guidebook, Version 1.0," 1 July 2015. [Online]. Available: http://www.dote.osd.mil/docs/TempGuide3/Cybersecurity_TE_Guidebook_July1_2015_v1_0.pdf.

[58] C. Folk, D. C. Hurley, W. K. Kaplow and J. F. Payne, "The Security Implications of the Internet of Things," AFCEA International Cyber Committee, February 2015. [Online]. Available: http://www.afcea.org/site/sites/default/files/files/AFC_WhitePaper_Revised_Out.pdf.

[59] NIST, "Exploring the Dimensions of Trustworthiness: Challenges and Opportunities Workshop," 30-31 August 2016. [Online]. Available: https://www.nist.gov/news-events/events/2016/08/exploring-dimensions-trustworthiness-challenges-and-opportunities.

[60] A. Temin and S. Musman, "A Language for Capturing Cyber Impact Effects, MTR 100344, PR 10-3793," The MITRE Corporation, Bedford, MA, 2010.

[61] SEBoK, "System Architecture," Systems Engineering Body of Knowledge, 2015. [Online]. Available: http://sebokwiki.org/wiki/System_Architecture.

[62] S. Borg and J. Bumgarner, "The US-CCU Cyber-Security Matrix: A New Type of Check List for Defending Against Cyber Attacks (DRAFT Version 2)," 22 September 2016. [Online]. Available: http://www.usccu.us/documents/US-CCU%20Cyber-Security%20Matrix%20(Draft%20Version%202).pdf.

[63] Microsoft, "Privileged Access Workstations," Microsoft TechNet, 25 May 2016. [Online]. Available: https://technet.microsoft.com/en-us/library/mt634654.aspx.

[64] K. E. Heckman, F. J. Stech, R. K. Thomas, B. Schmoder and A. W. Tsow, Cyber Denial, Deception and Counter Deception: A Framework for Supporting Active Cyber Defense (Advances in Information Security 63), Switzerland: Springer, 2015.

[65] CEF, "Center for Encrypted Functionalities," 2016. [Online]. Available: http://web.cs.ucla.edu/cef/index.html.

[66] DISA, "Terms and Conditions: Applicable to all Service Level Agreements," October 2016. [Online]. Available: http://www.disa.mil/~/media/files/disa/services/computing/termsandconditions.pdf.

[67] Oracle, "Oracle Partitioning Policy: Hardware/Server Partitioning," 5 April 2016. [Online]. Available: http://www.oracle.com/us/corporate/pricing/partitioning-070609.pdf.

[68] J. P. Sterbenz, D. Hutchison, E. K. Çetinkaya, A. Jabbar, J. P. Rohrer, M. Schöller and P. Smith, "Redundancy, diversity, and connectivity to achieve multilevel network resilience, survivability, and disruption tolerance," *Journal of Telecommunications Systems*, vol. 56, no. 1, pp. 17-31, 2014.

[69] J. P. Sterbenz, D. Hutchison, E. K. Çetinkaya, A. Jabbar, J. P. Rohrer, M. Schöller and P. Smith, "Resilience and survivability in communication networks: Strategies,principles, and survey of disciplines," 17 March 2010. [Online]. Available: http://www.ittc.ku.edu/resilinets/papers/Sterbenz-Hutchison-Cetinkaya-Jabbar-Rohrer-Scholler-Smith-2010.pdf.

[70] A. Höller, T. Rauter, J. Iber and C. Kreiner, "Towards Dynamic Software Diversity for Resilient Redundant Embedded Systems (Lecture Notes in Computer Science 9274)," in *Proceedings of Software Engineering for Resilient Systems: 7th International Workshop, SERENE 2015*, Switzerland, Springer, 2015, pp. 16-30.

[71] A. Höller, T. Rauter, J. Iber, G. Macher and C. Kreiner, "Software-Based Fault Recovery via Adaptive Diversity for COTS Multi-Core Processors," in *The 6th International Workshop on Adaptive Self-tuning Computing Systems (ADAPT 2016)*, Prague, Czech Republic, 2016.

[72] S. Brahma, K. Kwiat, P. K. Varshney and C. Kamhoua, "CSRS: Cyber Survive and Recover Simulator," in *Proceedings of the 2016 IEEE 17th International Symposium on High Assurance Systems Engineering*, Orlando, FL, 2016.

[73] K. M. Carter, H. Okhravi and J. Riordan, "Quantitative Analysis of Active Cyber Defenses Based on Temporal Platform Diversity," January 31 2014. [Online]. Available: https://arxiv.org/abs/1401.8255.

[74] G. Cybenko and J. Hughes, "No Free Lunch in Cyber Security," in *Proceedings of the First ACM Workshop on Moving Target Defense (MTD'14)*, Scottsdale, AZ, 2014.

[75] M. Bougeret, H. Casanova, Y. Robert, F. Vivien and D. Zaidouni, "Using replication for resilience on exascale systems, Research Report 7830," December 2011. [Online]. Available: http://graal.ens-lyon.fr/~yrobert/onlinepapers/RR-INRIA-7830.pdf.

[76] A. Zahid, R. Masood and M. A. Shibli, "Security of sharded NoSQL databases: A comparative analysis," in *2014 Conference on Information Assurance and Cyber Security (CIACS)*, 2014.

[77] The MITRE Corporation, "Situation Awareness," Cybersecurity: Strengthening Cyber Defense, 2013. [Online]. Available: http://www.mitre.org/work/cybersecurity/focus/awareness.html.

[78] S. Jajodia, P. Liu, V. Swarup and C. Wang, Cyber Situational Awareness: Issues and Research, Springer, 2010.

[79] A. Kott, C. Wang and R. F. Erbacher, Cyber Defense and Situational Awareness, Springer, 2014.

[80] DISA, "DISA's Big Data Platform and Analytics Capabilities," 16 May 2016. [Online]. Available: http://disa.mil/newsandevents/2016/Big-Data-Platform.

[81] GSA, "Highly Adaptive Cybersecurity Services (HACS)," 8 November 2016. [Online]. Available: http://www.gsa.gov/portal/content/151154.

[82] IACD, "Integrated Adaptive Cyber Defense," Johns Hopkins University Applied Physics Laboratory, 2016. [Online]. Available: https://secwww.jhuapl.edu/iacdcommunityday/About.

[83] H. Goldman, "Building Secure, Resilient Architectures for Cyber Mission Assurance," 2010. [Online]. Available: http://www.mitre.org/sites/default/files/pdf/10_3301.pdf.

[84] NIST, "NIST SP 800-125, Guide to Security for Full Virtualization Technologies," January 2011. [Online]. Available: http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-125.pdf.

[85] Cloud Security Alliance, "Security Position Paper: Network Function Virtualization," 26 February 2016. [Online]. Available: https://downloads.cloudsecurityalliance.org/assets/research/virtualization/Security_Position_Paper-Network_Function_Virtualization.pdf.

[86] R. D. McMurry, "Statement of Maj Gen Robert D. McMurry, USAF on Fiscal Year 2017 Air Force Research Laboratory to the House Armed Services Committee on Emerging Threats and Capabilities," 28 September 2016. [Online]. Available: http://www.defenseinnovationmarketplace.mil/resources/HHRG-114-AS26-Wstate-McMurryUSAFR-20160928.pdf.

[87] S. Jajodia, A. K. Ghosh, V. Swarup, C. Wang and X. S. Wang, Moving Target Defense: Creating Asymmetric Uncertainty for Cyber Threats (Advances in Information Security, Vol. 54), Springer, 2011.

[88] S. Jajodia, A. K. Ghosh, V. S. Subrahmanian, V. Swarup, C. Wang, X. S. Wang and (editors), Moving Target Defense II: Application of Game Theory and Adversarial Modeling (Advances in Information Security), New York: Springer, 2012.

[89] H. Maleki, S. Valizadeh, W. Koch, A. Bestavros and M. van Dijk, "Markov Modeling of Moving Target Defense Games," in *Proceedings of the Third ACM Workshop on Moving Target Defense (MTD 2016)*, Vienna, Austria, 2016.

[90] S. Shetty, X. Yuchi and M. Song, Moving Target Defense for Distributed Systems, Switzerland: Springer, 2016.

[91] H. Okhravi, M. A. Rabe, T. J. Mayberry, W. G. Leonard, T. R. Hobson, D. Bigelow and W. W. Streilein, "Survey of Cyber Moving Targets, ESC-EN-HA-TR-2012-109, Technical Report 1166," 25 September 2013. [Online]. Available: http://www.www.dtic.mil/cgi-bin/GetTRDoc?AD=ADA591804.

[92] R. Zhuang, S. A. DeLoach and X. Ou, "Towards a Theory of Moving Target Defense," in *Proceeding of Moving Target Defense Workshop (MTD'14)*, Scottsdale, AZ, 2014.

[93] R. Sun, M. Bishop, N. C. Ebner, D. Oliveira and D. E. Porter, "The Case for Unpredictability and Deception as OS Features," *;login:*, vol. 40, no. 4, pp. 12-17, 2015.

[94] R. D. Graubart and D. J. Bodeau, "Beyond the Baselines: Identifying Assumptions for Security Controls (DRAFT)," The MITRE Corporation, Bedford, MA, 2016.

[95] A. M. Madni and S. Jackson, "Towards a Conceptual Framework for Resilience Engineering," *IEEE Systems Journal, Vol. 3, No. 2,* June 2009.

[96] H. Lipson, "Evolutionary Systems Design: Recognizing Changes in Security and Survivability Risks (CMU/SEI-2006-TN-027)," September 2006. [Online]. Available: http://www.sei.cmu.edu/reports/06tn027.pdf.

[97] N. N. Taleb, Antifragile: Things That Gain from Disorder, Random House, 2012.

[98] E. Verhulst, "Applying systems and safety engineering principles for antifragility," in *1st International Workshop "From Dependable to Resilient, from Resilient to Antifragile Ambients and Systems" (ANTIFRAGILE 2014)*, 2014.

[99] K. J. Hole, Anti-fragile ICT Systems, Springer, 2016.

[100] B. Boehm, "System Qualities Ontology, Tradespace and Affordability (SQOTA) Project – Phase 4, Technical Report SERC-2016-TR-101," 10 February 2016. [Online]. Available: http://www.sercuarc.org/wp-content/uploads/2014/05/SERC-2016-TR-101-Phase-4_RT-137.pdf.

[101] NIST, "Guide for Conducting Risk Assessments, NIST SP 800-30 Rev.1," September 2012. [Online]. Available: http://csrc.nist.gov/publications/nistpubs/800-30-rev1/sp800_30_r1.pdf.

[102] S. Musman and S. Agbolosu-Amison, "A Measurable Definition of Resiliency Using "Mission Risk" as a Metric," March 2014. [Online]. Available: http://www.mitre.org/sites/default/files/publications/resiliency-mission-risk-14-0500.pdf.

[103] The MITRE Corporation, "Industry Perspective on Cyber Resiliency (home page)," Industry Perspective on Cyber Resiliency, 2015. [Online]. Available: http://www2.mitre.org/public/industry-perspective/index.html.

[104] J. P. Anderson, "Computer Security Technology Planning Study, ESD-TR-73-51, Vol. II," October 1972. [Online]. Available: http://seclab.cs.ucdavis.edu/projects/history/papers/ande72.pdf.

[105] J. H. Saltzer and M. D. Schroeder, "The Protection of Information in Computer Systems," *Proceedings of the IEEE* , vol. 63, no. 9, pp. 1278-1308, 1975.

[106] M. Gegick and S. Barnum, "Design Principles," Build Security In, 13 May 2013. [Online]. Available: https://buildsecurityin.us-cert.gov/articles/knowledge/principles/design-principles.

[107] T. V. Benzel, C. E. Irvine, T. E. Levin, G. Bhaskara, T. D. Nguyen and P. C. Clark, "Design Principles for Security (ISI-TR-605, NPS-CS-05-010)," 20 September 2005. [Online]. Available: http://cisr.nps.edu/downloads/techpubs/nps_cs_05_010.pdf.

[108] CESG, "Security Design Principles for Digital Services," Communications-Electronics Security Group, 14 March 2016. [Online]. Available: https://www.cesg.gov.uk/guidance/security-design-principles-digital-services-0.

[109] The MITRE Corporation, "Secure Administration: Securing the Keys to the Kingdom," Industry Perspective on Cyber Resiliency, 2015. [Online]. Available: http://www2.mitre.org/public/industry-perspective/slicksheets/fasecure_administration.html.

[110] J. Hughes and G. Cybenko, "Quantitative Metrics and Risk Assessment: The Three Tenets Model of Cybersecurity," Technology Innovation Management Review, August 2013. [Online]. Available: https://timreview.ca/article/712.

[111] J. Hughes and G. Cybenko, "Three tenets for secure cyber-physical system design and assessment," in *Cyber Sensing 2014, Proceedings of SPIE - The International Society for Optical Engineering, 2014*, 2014.

[112] OWASP, "Security by Design Principles," Open Web Application Security Project , August 2016. [Online]. Available: https://www.owasp.org/index.php/Security_by_Design_Principles.

[113] G. Coker, J. Guttman, P. Losocco, A. Herzog, J. Millen, B. O'Hanlon, J. Ramsdell, A. Segall, J. Sheehy and B. Sniffen, "Principles of remote attestation," *International Journal of Information Security - Special Issue: 10th International Conference on Information and Communications Security (ICICS)*, vol. 10, no. 2, pp. 63-81, 2011.

[114] N. Ferguson, B. Schneier and T. Kohno, Cryptography Engineering: Design Principles and Practical Applications, Wiley, 2010.

[115] Resilient Design Institute, "The Resilient Design Principles," [Online]. Available: http://www.resilientdesign.org/the-resilient-design-principles/.

[116] Cyber Operations, Analysis, and Research (COAR), "Cyber Resilience in Active Defense Techniques," Risk and Infrastructure Science Center, Argonne National Laboratory, 22 December 2015. [Online]. Available: http://coar.risc.anl.gov/655-2/.

[117] M. G. Richards, D. E. Hastings, D. H. Rhodes, A. M. Ross and A. L. Weigel, "Design for Survivability: Concept Generation and Evaluation in Dynamic Tradespace Exploration," in *Second International Symposium on Engineering Systems*, Cambridge, MA, 2009.

[118] D. Bodeau, J. Brtis, R. Graubart and J. Salwen, "Resiliency Techniques for System of Systems: Extending and Applying the Cyber Resiliency Engineering Framework to the Space Domain (MTR 130515, PR 13-3513)," September 2013. [Online]. Available: http://www.mitre.org/sites/default/files/publications/13-3513-ResiliencyTechniques_0.pdf.

[119] DoD, "Manual for the Operation of The Joint Capabilities Integration and Development System (JCIDS)," 12 February 2015. [Online]. Available: https://dap.dau.mil/policy/Documents/2015/JCIDS_Manual_-_Release_version_20150212.pdf.

[120] Joint Chiefs of Staff, "Cyber Survivability Endorsement Implementation Guide DRAFT Version 1.0 (U//FOUO)," 2016.

[121] DOE, "DOE Standard: Integration of Safety into the Design Process, DOE-STD-1189-2008," March 2008. [Online]. Available: https://energy.gov/sites/prod/files/2013/06/f1/DOE-STD-1189-2008.pdf.

[122] DoD, "Department of Defense Standard Practice: System Safety, MIL-STD-882E," 11 May 2012. [Online]. Available: http://www.system-safety.org/Documents/MIL-STD-882E.pdf.

[123] ISO/IEC JTC 1/SC 7, ISO/IEC/IEEE 15288:2015 - Systems and software engineering -- System life cycle processes, Geneva, Switzerland: International Standards Organization, 2015.

[124] C. Zimmerman, "Ten Strategies of a World-Class Cybersecurity Operations Center," October 2014. [Online]. Available: http://www.mitre.org/sites/default/files/publications/pr-13-1028-mitre-10-strategies-cyber-ops-center.pdf.

[125] Advameg, Inc., "Computer Security," Encyclopedia of Business, 2nd edition, 2016. [Online]. Available: http://www.referenceforbusiness.com/encyclopedia/Clo-Con/Computer-Security.html.

[126] N. Husted and S. Myers, "Emergent Properties & Security: The Complexity of Security as a Science," in *New Security Paradigms Workshop (NSPW'14)*, Victoria, British Columbia, 2014.

[127] J. Black and P. Koopman, "System Safety as an Emergent Property in Composite Systems," in *International Conference on Dependable Systems and Networks (DSN09)*, 2009.

[128] DAU, "Defense Acquisition Portal," Defense Acquisition University (DAU), [Online]. Available: https://dap.dau.mil/aphome/das/Pages/Default.aspx.

[129] K. G. Partridge and L. R. Young, "CERT Resilience Management Model (CERT-RMM) V1.1: NIST Special Publication Crosswalk Version 2," April 2011. [Online]. Available: http://resources.sei.cmu.edu/asset_files/TechnicalNote/2011_004_001_15371.pdf.

[130] National Science and Technology Council, "Federal Cybersecurity Research and Development Strategic Plan," February 2016. [Online]. Available: https://www.whitehouse.gov/sites/whitehouse.gov/files/documents/2016_Federal_Cybersecurity_Research_and_Development_Stratgeic_Plan.pdf.

[131] E. M. Hutchins, M. J. Cloppert and R. M. Amin, "Intelligence-Driven Computer Network Defense Informed by Analysis of Adversary Campaigns and Intrusion Kill Chains," Proceedings of the 6th International Conference on Information-Warfare & Security (ICIW 2011), March 2011. [Online]. Available: http://www.lockheedmartin.com/content/dam/lockheed/data/corporate/documents/LM-White-Paper-Intel-Driven-Defense.pdf.

[132] D. Ormrod and B. Turnbull, "The Military Cyber-Maturity Model: Preparing Modern Cyber-Enabled Military Forces for Future Conflicts," in *11th International Conference on Cyber Warfare and Security: ICCWS2016, 17-18 March 2016*, Boston, MA, 2016.

[133] R. J. Danzig, "Surviving on a Diet of Poisoned Fruit: Reducing the National Security Risks of America's Cyber Dependencies," July 2014. [Online]. Available: http://www.cnas.org/sites/default/files/publications-pdf/CNAS_PoisonedFruit_Danzig_0.pdf.

[134] NIST, "Federal Information Processing Standards Publication (FIPS PUB) 199: Standards for Security Categorization of Federal Information and Information Systems," February 2004. [Online]. Available: http://csrc.nist.gov/publications/fips/fips199/FIPS-PUB-199-final.pdf.

[135] NIST, "NIST SP 800-183, Networks of 'Things'," July 2016. [Online]. Available: http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-183.pdf.

[136] K. Thompson, "Reflections on Trusting Trust," *Communication of the ACM. Vol. 27, No. 8,* pp. 761-763, August 1984.

[137] A. Hahn, R. Thomas, I. Lozano and A. Cardenas, "A multi-layered and kill-chain based security analysis framework for cyber-physical systems," *International Journal of Critical Infrastructure Protection,* vol. 11, pp. 39-50, 2015.

[138] R. E. Smith, "A Contemporary Look at Saltzer and Schroeder's 1975 Design Principles," *IEEE Security & Privacy,* vol. 10, no. 6, pp. 20-25, 2012.

[139] The MITRE Corporation, "Active Defense Strategy for Cyber," 8 November 2012. [Online]. Available: https://www.mitre.org/sites/default/files/publications/active_defense_strategy.pdf.

Appendix A Background on Cyber Resiliency

Cyber resiliency is concerned with addressing *all* threats to cyber resources, whether such threats are cyber or non-cyber (e.g., kinetic) in nature. But cyber resiliency *focuses* on addressing the advanced cyber threat [12], also known as the advanced persistent threat (APT) [36].[32] The resources associated with the APT, its stealthy nature, its persistent focus on the target of interest, and its ability to evolve in the face of defender actions make it a highly dangerous threat. Moreover, APT actors can construct tactics, techniques, and procedures (TTPs) to take advantage of or make their behavior appear to result from other forms of adversity, including human error, structural failure, or natural disaster. Thus, engineering decisions focusing on potential effects of APT activities can be expected to provide the ability to anticipate, withstand, recover from, and adapt to a broad suite of adverse conditions and stresses on cyber resources. This ability maximizes mission continuity despite the presence of an adversary in a system, including an adversary which may be masquerading as other representative adverse events such as software and operator errors, failures of supporting infrastructures (e.g., power), and natural events with cyber effects (e.g., solar weather that affects satellite communications).

A.1 Cyber Resiliency Engineering Framework

The Cyber Resiliency Engineering Framework (CREF), illustrated in Figure 6, organizes the cyber resiliency domain into a set of goals, objectives, and techniques [2] [3]. *Goals* (defined in the Glossary) are high-level statements of intended outcomes, which help scope the cyber resiliency domain. In keeping with the fact that cyber resiliency is concerned with all threats, the goals are derived from those defined by the discipline of Resilience Engineering [95].

Objectives, presented in Appendix B, are more specific statements of intended outcomes that serve as a bridge between techniques and goals. Objectives are expressed so as to facilitate assessment, making it straightforward to develop questions of "how well," "how quickly," or "with what degree of confidence or trust" can each objective be achieved. Objectives enable different stakeholders to assert their different resiliency priorities based on mission or business functions.

Cyber resiliency *techniques* (also presented in Appendix B) characterize approaches to achieving one or more cyber resiliency objectives that can be applied to the architecture or design of mission/business functions and the cyber resources that support them. Each technique refers to a set of related approaches and technologies.

[32] Terminology referring to attackers continues to evolve. The Defense Science Board refers to the cyber threat, and defines six tiers of attackers [12]. The publications of the Joint Task Force Transformation Initiative refer to the advanced persistent threat [29]. The Federal Cybersecurity Research and Development Strategic Plan refers to malicious cyber activities (MCA) [130].

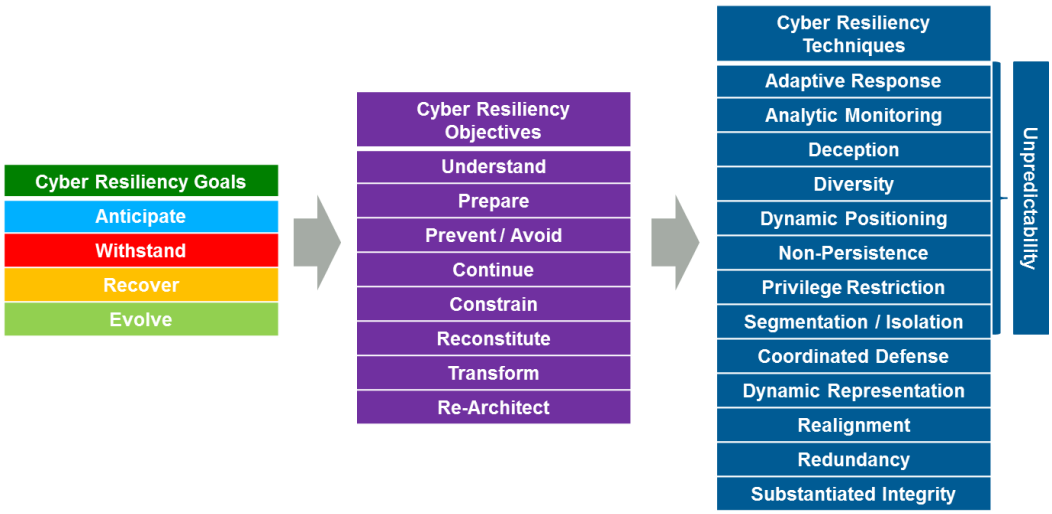

Figure 6. Cyber Resiliency Engineering Framework

A.2 Cyber Resiliency Design Principles and Other Constructs

Figure 7 illustrates the relationships between cyber resiliency design principles and some other key constructs considered in systems engineering. A given cyber resiliency design principle supports one or more cyber resiliency objectives, and uses one or more cyber resiliency techniques. It justifies the selection, tailoring, or de-selection of security controls related to cyber resiliency, and the allocation of those controls in the architecture. In many cases, it aligns with, amplifies, or can serve as an alternative to one or more design principles from related disciplines, including security, resilience engineering, survivability, and evolvability.

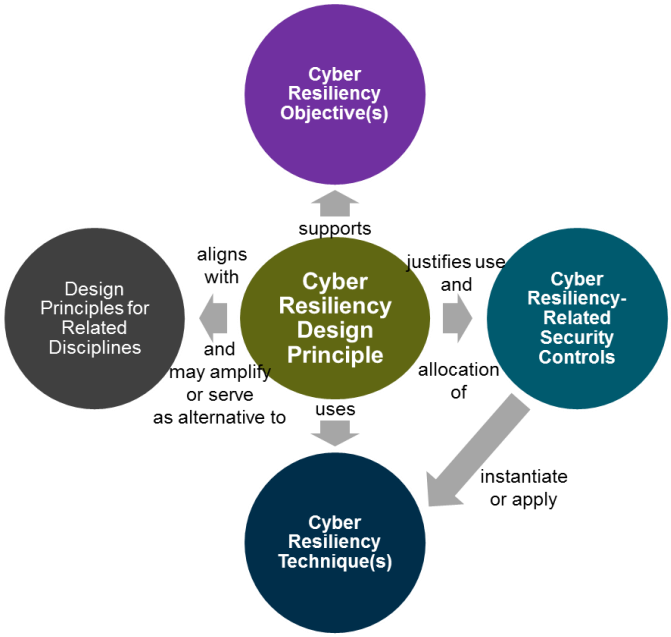

Figure 7. Cyber Resiliency Design Principles in Relation to Other Key Constructs

A.3 Threat Model for Cyber Resiliency

Unlike in such related disciplines as survivability, safety, and resilience engineering, the threat model in cyber resiliency centers on the advanced cyber threat and on the effects of malicious cyber activities. Advanced cyber threat actors execute cyber campaigns that can involve multiple systems and organizations, and can extend for periods of months or even years. A variety of structures or models of cyber campaigns[33] have been defined; the one illustrated in Figure 8 is consistent with Appendix E of [101]. For a taxonomy of post-exploit MCA, see Adversarial Tactics, Techniques, and Common Knowledge (ATT&CK™) [47].

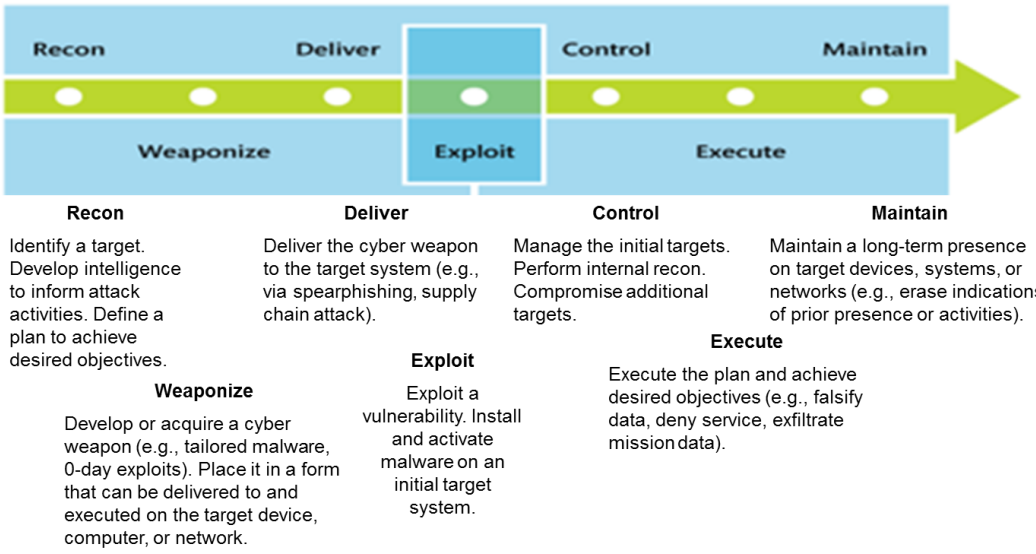

Figure 8. Cyber Attack Lifecycle

Survivability and resilience engineering use a disruption threat model; a simple version is illustrated in Figure 8, while a more nuanced version is illustrated in Figure 10. In either case, a discernable disruptive event is assumed.

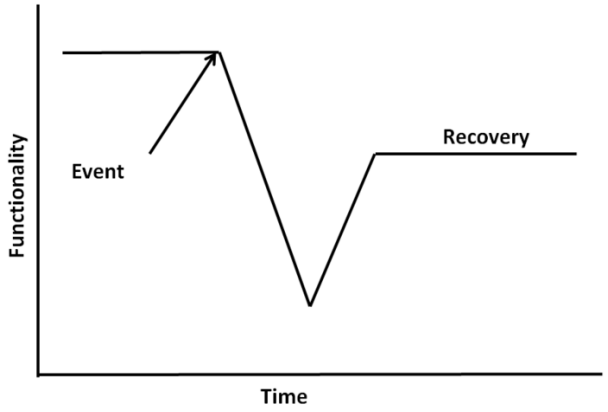

Figure 9. Disruption Model for Survivability or Resilience Engineering[34]

[33] Such a model is often referred to as a cyber attack lifecycle [3] or a cyber kill chain [131] [139].
[34] This graphic is taken from the Systems Engineering Body of Knowledge (SEBoK), http://sebokwiki.org/wiki/File:Disruption_Diagram.PNG.

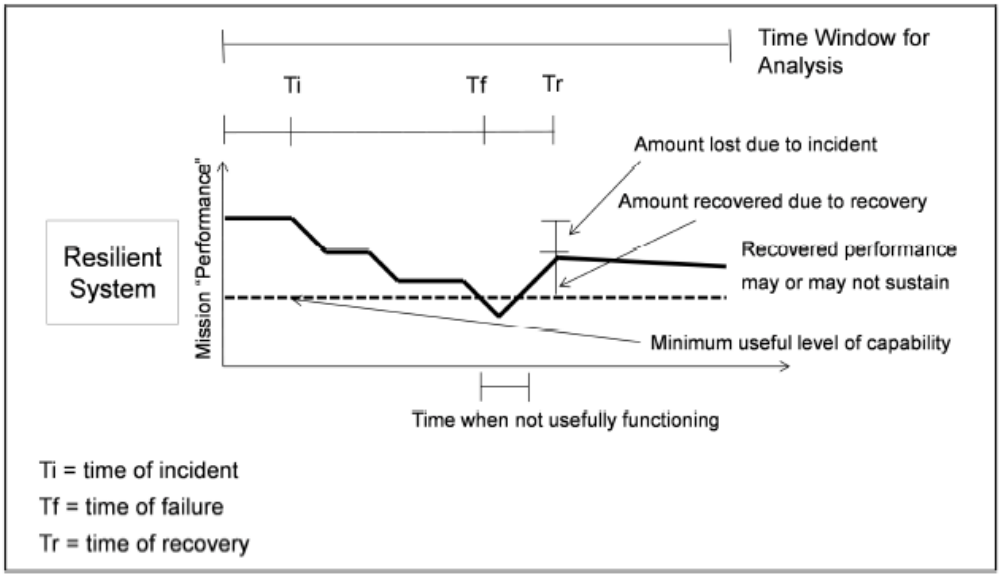

Figure 10. Performance Curve Illustrating Aspects of Resilience (Figure 1 of [102])

These models are relevant to some classes of cyber attacks (e.g., distributed denial of service or DDoS attacks), in which adversary activity has easily discernable effects. However, when malicious cyber activities follow a cyber attack lifecycle, additional models need to be considered. Figure 11 illustrates a cyber campaign in which destructive malware degrades or denies mission capabilities.

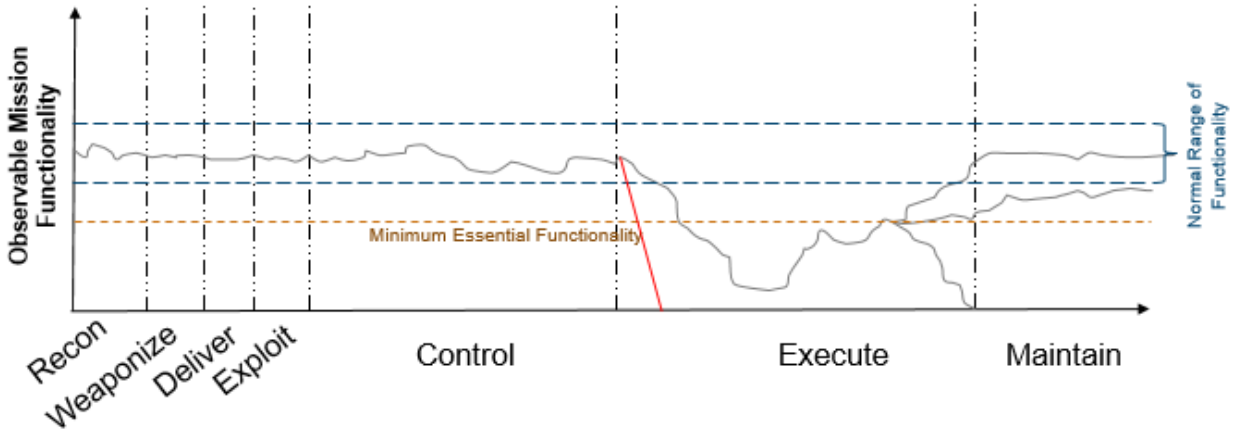

Figure 11. Cyber Resiliency Against Destructive Malware

In the figure, the Recon and Weaponize stages are compressed, since they generally cannot be observed. During the Control stage, the adversary extends their knowledge of the system and the missions it supports, hides or removes evidence of activities, and acquires control of more system components; because stealth is intrinsic to the adversary's plan, the adversary avoids creating disruption (in fact, the adversary can even take actions to improve system performance), or creates transient and minor disruptions to trick performance and intrusion detection tools into redefining "normal." In the Execute stage, the adversary directs malware to take actions (e.g., deny service, corrupt data in ways that make it useless, cause physical harm) – which can include actively impeding recovery. As the red line indicates, destructive malware can reduce the functionality of a component or system to nil. Alternately, cyber defenders can use cyber resiliency techniques to restore functionality to a minimum essential level, or can

choose to move mission support operations to another system entirely. (In the last case, cyber defenders may seek to bring the system down in a controlled way, to preserve forensic evidence.)

Figure 12 illustrates a cyber attack lifecycle in which the adversary's goal is either data exfiltration or fabrication, or undetected usurpation of capabilities (e.g., using a collection of IoT devices to launch a denial-of-service attack, without leading the owners / operators of the devices to suspect that their devices have been compromised). In this situation, the adversary seeks to avoid any disruption. However, as the red line indicates, if and when the consequences of adversary activities are detected, defender responses can constitute an observable disruption.

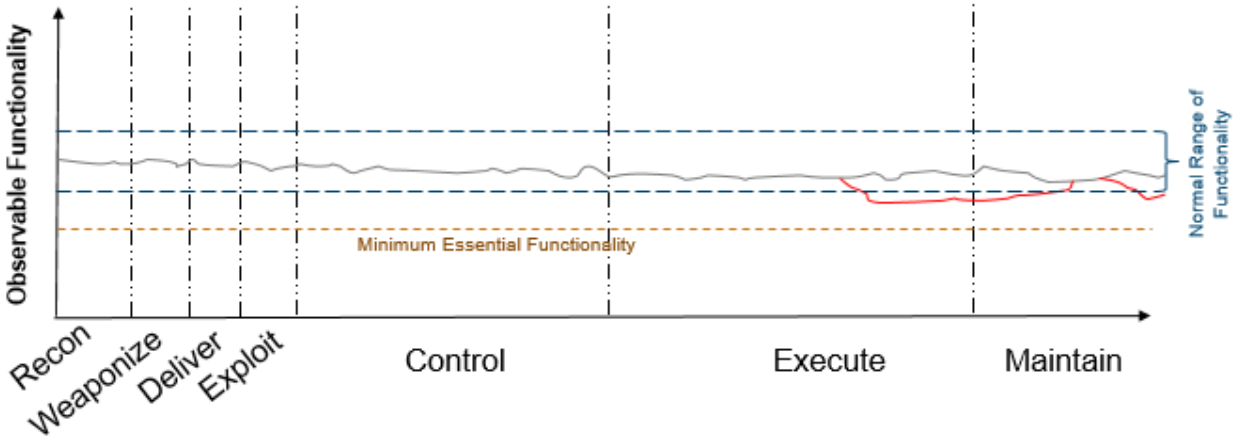

Figure 12. Cyber Resiliency Against Data Exfiltration or Fabrication

A.4 Cyber Resiliency and Trustworthiness

Trustworthiness can be defined as "The attribute of [an entity] that provides confidence to others of the qualifications, capabilities, and reliability of that entity to perform specific tasks and fulfill assigned responsibilities." [31]) In the context of systems engineering, trustworthiness "means simply worthy of being trusted to fulfill whatever critical requirements may be needed for a particular component, subsystem, system, network, application, mission, enterprise, or other entity. Trustworthiness requirements can include, for example, attributes of safety, security, reliability, dependability, performance, resilience, and survivability under a wide range of potential adversity in the form of disruptions, hazards, and threats." [1]

The Cyber-Physical Systems Public Working Group (CPS PWG), an open public forum established by NIST, treats safety, security, privacy, resilience, and reliability as dimensions of trustworthiness [34].[35] That approach enables trade-offs among those dimensions in a given architecture to be articulated. Depending on the context, resilience and reliability can be treated as a single concern [59]. Particularly (but not uniquely) for National Security Systems (NSS), mission assurance against advanced adversaries can be considered as an additional concern. From this perspective, the notional relationships between cybersecurity and cyber resiliency and conventional security (with its focus on confidentiality, integrity, and availability of information), conventional resilience and reliability (with a focus on non-adversarial threats), safety, privacy, and mission assurance can be represented as in the figure below.

[35] While these terms are not defined in the CPS Framework, the working distinction between resilience and reliability is rooted in the underlying threat model: Reliability involves ensuring adequate performance in the face of known disruptions, while resilience involves withstanding and recovering from unknown disruptions.

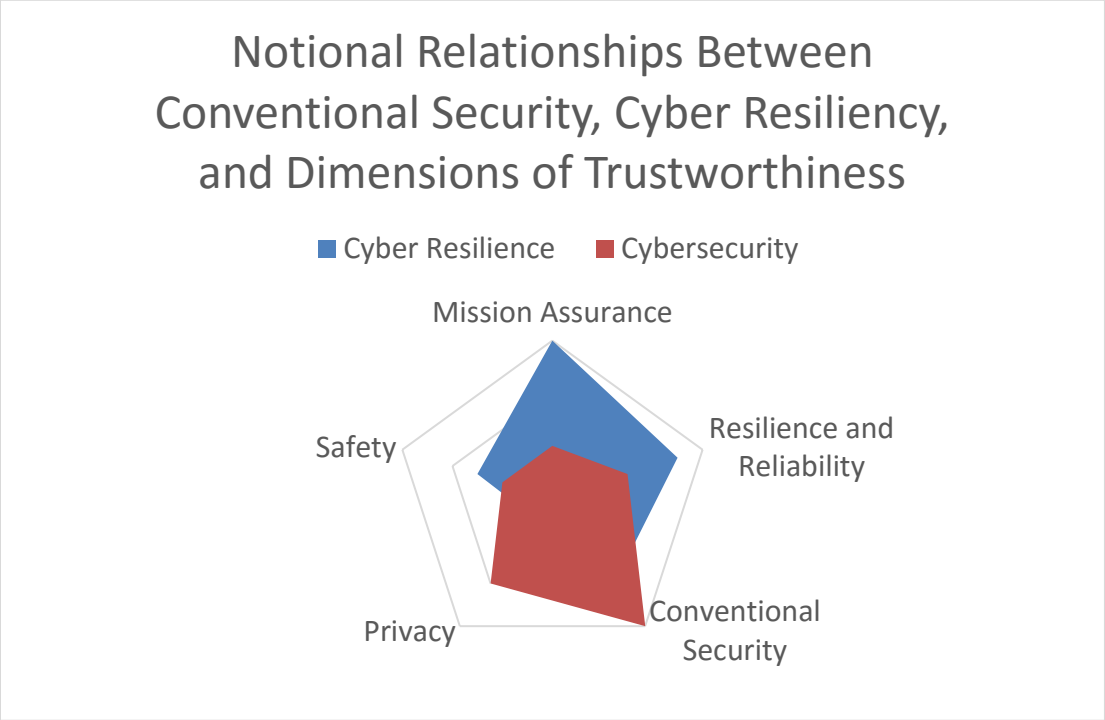

Figure 13. Notional Relationships Among Dimensions of Trustworthiness

As Figure 13 illustrates, cyber resiliency overlaps with and builds upon other aspects of trustworthiness, particularly resilience and conventional security. Cybersecurity (as defined in CNSSI No. 4009 [31]) encompasses conventional security in its consideration of prevention and protection, but overlaps with conventional reliability and resilience in its inclusion of restoration.

Appendix B Sources of Cyber Resiliency Design Principles

The cyber resiliency design principles presented in Section 3 are derived from a variety of sources:

- General principles can be defined directly from the CREF, as described in B.1.
- More specific principles for applying cyber resiliency techniques can be defined in the context of a framework for activities an organization can take, pre- or post- "bang" (where "bang" refers to the detection of a cyber incursion). A set of cyber resiliency design principles with this more operational perspective have been identified by a collaboration of Government and industry organizations [103]. These are identified in B.2.
- Design principles articulated by participants in the series of Secure and Resilient Cyber Architectures Invitational events [8]. These are identified in B.3.
- Strategies and requirements articulated in the context of a specific program or system. Examples are presented in B.4.

The statements presented in this appendix can be used as inputs to restatements of the cyber resiliency design principles in Section 3, or as alternative additional principles.

B.1 General Cyber Resiliency Design Principles Defined Using the CREF

As described in Section 1, the Cyber Resiliency Engineering Framework (CREF) provides a way to structure discussions the cyber resiliency domain. High-level design principles for cyber resiliency can be defined using the objectives defined in the CREF, as shown in Table 21. Design principles derived from the cyber resiliency objectives can be viewed as *strategic*, using the distinction between strategic and structural design principles defined by Ricci et al. [15].

Table 21. Supplementary or Alternative Design Principles from Cyber Resiliency Objectives

Cyber Resiliency Objective	Description	Design Principles
Understand	Maintain useful representations of mission and business dependencies and the status of resources with respect to possible adversity.	**Be transparent**: Maximize transparency and visibility to mission owners and cyber defenders.
Prepare	Maintain a set of realistic courses of action that address predicted or anticipated adversity.	**Be prepared**: Design for operational flexibility.
Prevent / Avoid	Preclude the successful execution of an attack or the realization of adverse conditions.	**Head off adversity**: Apply best practices for cybersecurity and disaster recovery.
Continue	Maximize the duration and viability of essential mission or business functions during adversity.	**Assure mission continuity**: Design systems and mission processes to handle disruption and degradation.
Constrain	Limit damage from adversity.	**Design for damage limitation**: Build in checks and limits, and accommodate procedural work-arounds.
Reconstitute	Restore as much mission or business functionality as possible subsequent to adversity.	**Recover gracefully**: Design systems and mission processes to restore mission capabilities based on criticality.
Transform	Modify mission or business functions and supporting processes to handle adversity more effectively.	**Consider resilience in process design**: Minimize unnecessary dependencies in the design of systems and of mission processes.
Re-architect	Modify architectures to handle adversity more effectively.	**Enable architectural evolution**: Design for environmental evolution and ongoing adoption of cyber resiliency techniques.

An objective-derived strategic design principle could be used in addition to, or as an alternative to, strategic design principles as presented in Table 1. However, as illustrated in Table 19, each strategic design principle presented in Table 1 supports multiple cyber resiliency objectives.

As Table 22 indicates, the descriptions of the cyber resiliency techniques can be viewed as structural design principles. See [3] for descriptions of the rationales for the techniques, as well as the potential effects of different approaches to applying the techniques on adversary activities. Each technique supports multiple cyber resiliency objectives; these are identified in the third column of Table 22. (Note: this is an alternative presentation of the material in Table 4 of [3].)

Table 22. Descriptions of Cyber Resiliency Techniques Can Be Viewed as Design Principles

Cyber Resiliency Technique	Description	Cyber Resiliency Objectives Supported
Adaptive Response	Implement nimble cyber courses of action (CCoAs) to manage risks.	Constrain Continue Reconstitute
Analytic Monitoring	Gather, fuse, and analyze data on an ongoing basis and in a coordinated way to identify potential vulnerabilities, adverse conditions, stresses, attacks, or damage.	Understand Prepare Constrain Reconstitute
Coordinated Defense	Manage multiple distinct mechanisms in a non-disruptive or complementary way.	Prepare Prevent / Avoid Constrain Continue Reconstitute
Deception	Mislead, confuse, or hide critical assets from the adversary.	Understand Prevent / Avoid Continue
Diversity	Use heterogeneity to minimize common mode failures, particularly attacks exploiting common vulnerabilities.	Prevent / Avoid Continue Re-Architect
Dynamic Positioning	Distribute and dynamically relocate functionality or assets.	Understand Prevent / Avoid Continue Re-Architect
Dynamic Representation	Construct and maintain current representations of mission or business posture in light of cyber events and courses of action.	Understand Prepare Transform
Non-Persistence	Generate and retain resources as needed or for a limited time.	Prevent / Avoid Constrain Continue Re-Architect
Privilege Restriction	Restrict privileges required to use system resources, and privileges assigned to users and system entities, based on the type and degree of criticality.	Prevent / Avoid Constrain
Realignment	Align system resources with core aspects of organizational missions or business functions.	Constrain Transform
Redundancy	Provide multiple protected instances of critical resources.	Continue Reconstitute
Segmentation / Isolation	Define and separate (logically or physically) components on the basis of criticality and trustworthiness.	Prevent / Avoid Constrain
Substantiated Integrity	Ascertain whether critical services, information stores, information streams, and components have been corrupted.	Understand Constrain Continue Reconstitute

Cyber Resiliency Technique	Description	Cyber Resiliency Objectives Supported
Unpredictability	Make changes randomly or unpredictably.	Understand Prevent / Avoid Continue

B.2 Cyber Resiliency Design Principles from an Operational Perspective

Treating the definitions of the cyber resiliency techniques as design principles is likely to be too high-level to be useful in a real-world setting. A collaboration with Government and commercial organizations [103] identified best practices for six activities "pre-bang" (i.e., prior to a cyber incursion), for six activities "post-bang," and for disrupting the attack surface. These activities are illustrated in Figure 14.

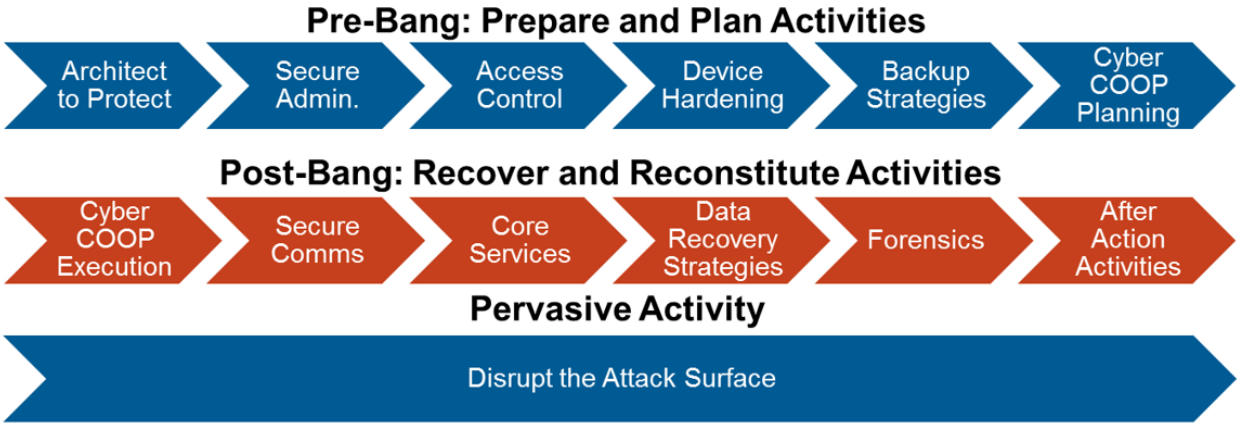

Figure 14. Operational Context for Cyber Resiliency Design Principles

The identified practices include design principles which restate and amplify the general statements of CREF-based techniques, in the context of enterprise information technology (IT). Many of these design principles focus on organizational processes, and thus are highly relevant to as-built systems. The activities and corresponding design principles are shown in Table 23.

Table 23. Cyber Resiliency Design Principles from an Industry Perspective

Organizational Activity	Industry Stated Cyber Resiliency Design Principles
Disrupt the Attack Surface: Make life hard for the adversary.	**Adaptive Response:** Optimize the organization's ability to respond in a timely and appropriate manner to adversary activities, thus maximizing the ability to maintain mission/business operations, limit consequences, and avoid destabilization. **Deception:** Mislead or confuse the adversary, or hide critical assets from the adversary, making them uncertain how to proceed, delaying the effect of their attack, increasing the risk to them of being discovered, causing them to misdirect or waste their attack, and expose their tradecraft (e.g., Attacker TTPs) prematurely. **Dynamic Positioning:** Impede an adversary's ability to locate, eliminate or corrupt mission/business assets, and cause the adversary to spend more time and effort to find the organization's critical assets, thereby increasing the chance of the adversary revealing their actions and tradecraft (e.g., Attacker TTPs) prematurely. **Non-Persistence:** Reduce exposure to corruption or modification; provide a means of curtailing an adversary's advance and potentially expunging an adversary's foothold from the system. **Realignment:** Reduce the attack surface of the defending organization by minimizing the chance that non-mission/business functions could be used as an attack vector. **Unpredictability:** Increase the adversary's uncertainty regarding the cyber defenses that they may encounter, thus making it more difficult for them to ascertain the appropriate course of action.

Organizational Activity	Industry Stated Cyber Resiliency Design Principles
Architect to Protect: Create a foundation for resiliency.	**Segmentation**: Don't give the adversary freedom to move laterally; architect the network and shared services so that portions can be isolated. **Coordinated Defense**: Apply technical Defense-in-Depth effectively using multiple protections at multiple architectural layers; provide processes and tools to manage protections consistently; define business processes that enable cyber defenders to collaborate. **Diversity**: Capitalize on and plan and manage diversity in enterprise systems and processes. Managed diversity reduces the effectiveness of attacks targeted at a particular system type by introducing multiple layers of friction in the cyber attack lifecycle.
Secure Administration: Secure the keys to the kingdom.	**Coordinated Defense**: Develop Administrator Standard Operating Procedures (SOPs) in coordination with business operations and Cyber Courses of Action across multiple administrative domains. **Privilege Restriction**: Apply good practice standards for least privilege, separation of duties, and role-based access control across administrator accounts; limit administrator account access to non-essential capabilities (e.g., e-mail, Internet). **Segmentation**: Designate systems exclusively for administration tasks; physically and logically separate administration and management control channels from the primary enterprise network. As appropriate, further separate activities and functions of privileged users into privileged and non-privileged functions.
Access Control: Constrain what the adversary can do.	**Privilege Restriction**: Minimize the number of services and privileges associated with authorized subjects (e.g., users, platforms, services, and applications) based on roles and groups. By restricting the actions subjects can take when they access resources, you can limit the harm an attacker can achieve. **Coordinated Defense**: Use a layered, defense-in-depth strategy that requires potential adversaries to navigate through and overcome various access control checkpoints to reach their ultimate objective. **Segmentation**: Establish separate domains for critical data and assets. That way, an access compromise in one domain does not affect another. **Analytic Monitoring**: Continuously monitor on-going activities to ensure that required access control mechanisms and settings are in place, are operating correctly, and haven't been comprised.
Device Hardening: Make it harder for components to be compromised.	**Privilege Restriction**: Minimize the ability of a compromised service within a component to compromise other services on that component and spread to other components. Limit the services between components. **Coordinated Defense**: Define a strategy that identifies and isolates unpatched components. Apply security mechanisms consistently across the enterprise.
Backup Strategies: Reconstituting information is key to recovery.	**Segmentation**: Ensure the backup data is isolated from other enterprise services to protect the backups from being impacted by adversary attacks. **Redundancy**: Deploy redundant systems in the Security Operations Center to provide failover capability; maintain protected copies of critical resources; design for spare capacity and secure failover. **Substantiated Integrity**: Ensure that adversaries have not corrupted backups of critical systems (e.g., directory servers, key management systems, payroll systems, etc.); validate data provenance and integrity; validate software, service integrity, and system behavior to ensure they have not been corrupted.
Cyber Continuity of Operations (COOP) Planning: Operationalize resiliency.	**Dynamic Representation**: Construct and maintain current representations of the posture of mission or business processes in light of cyber events and cyber courses of action. Provide cyber situational awareness as a part of overall situational awareness. **Coordinated Defense**: Plan, manage and coordinate multiple procedures and tactics along with the roles/responsibilities and playbooks needed for post-bang recovery. Realignment: Align cyber resources with core aspects of mission and business functions.
Cyber COOP Execution: Face destructive malware.	**Adaptive Response**: Use playbook cyber courses of action (CCoA) based on attack characteristics to recover to minimum core functionality, monitor effectiveness, and update CCoA as needed. Analytic Monitoring: Gather, fuse, and analyze data on an ongoing basis and in a coordinated way to identify potential vulnerabilities, adversary activities, and damage. **Substantiated Integrity**: Provide mechanisms to ascertain the integrity of reconstituted data, services, and information streams.

Organizational Activity	Industry Stated Cyber Resiliency Design Principles
Secure Communications: Protect response and recovery from compromise.	**Segmentation**: Isolate the cybersecurity operations/response center from inbound access of the enterprise network as hostile activity may be occurring in that portion of the network. Where response-related communications flow over the same circuits as ordinary enterprise network traffic, use encryption (e.g., a virtual private network or VPN) to keep it logically separate. **Substantiated Integrity**: Employ data validation mechanisms, to validate the integrity of the data and the authenticity of the sender. **Redundancy**: Ensure that incident response staff have multiple communication paths. This will ensure communications even if the primary secure communications path is compromised or otherwise unavailable.
Core Services: Re-establish a trusted foundation.	**Adaptive Response**: Dynamically reconstitute critical assets or capabilities. Identify and restore functional capabilities based on criticality. **Coordinated Defense**: Coordinate recovery activities to avoid gaps in security coverage. **Redundancy**: Maintain multiple protected instances of hardware, software, and information, enabling a portion of service capacity to be quickly established. **Substantiated Integrity**: Validate data provenance. Validate data, software, and service integrity to ensure they are not corrupt.
Data Recovery Strategies: Assure trustworthiness for continued performance.	**Adaptive Response**: Implement nimble cyber courses of action to manage risks. **Redundancy**: Provide multiple, diverse, protected instances of critical information and supporting services that will be used during the recovery process. **Substantiated Integrity**: Establish mechanisms to determine if critical infrastructure, services, information repositories, information streams and supporting components have been corrupted.
Forensics: Restore trust in the enterprise.	**Analytic Monitoring**: Don't confuse absence of evidence with evidence of absence; refine organization-wide data collection to capture and retain forensics in case of future incidents; improve situational awareness and data access capabilities for efficient evidence retrieval. **Coordinated Defense**: Define business processes for proper incident handling, reporting and investigation; establish partnerships to share forensic techniques and best practices. **Segmentation**: Ensure malware analysis takes place in an isolated environment. **Substantiated Integrity**: Reduce or eliminate compromises to the integrity of incident forensic evidence, such as event logs (both as they are being collected and while in storage).
After Action Activities: Continually improve enterprise resilience.	**Adaptive Response**: Improve existing response capabilities and actuator thresholds; identify and implement new approaches to enhance protection against future incidents. **Analytic Monitoring**: Gather, fuse and analyze data to identify adverse conditions, reveal the extent of adversary activity and identify damage. **Coordinated Defense**: Improve Cyber COOP procedures and define business processes to enable cyber defenders to share incident reports and collaborate on new approaches. **Realignment**: Identify and disable any non-essential system capabilities that might have caused or increased the severity of the incident.

B.3 Principles Identified by Community Brainstorming

A track on cyber resiliency design principles at the 2012 Secure and Resilient Cyber Architectures Invitational identified a number of generally accepted principles, as well as others which were discussed and considered noteworthy by a subset of the participants. In addition, participants proposed operational principles, some of which imply design principles. Of the list developed by participants, some principles are subsumed by one or more of the design principles discussed in Section 2; others relate more strongly to general principles for resilience and reliability. The results of community brainstorming are shown in Tables 24 and 25.

Table 24. Community-Developed Design Principles Related to Cyber Resiliency

Generally Accepted Design Principle	Corresponding Cyber Resiliency Design Principle(s) from Table 1
Design to reduce exposure to attack	Reduce attack surfaces.
Design to reduce persistence of access by the adversary	Maximize transience; minimize persistence.

Generally Accepted Design Principle	Corresponding Cyber Resiliency Design Principle(s) from Table 1
Design to reduce adversary's ability to act	Control visibility and use. Contain and exclude behaviors.
Design to limit the consequences of attack	Contain and exclude behaviors. Manage resources (risk-) adaptively.
Design to minimize common cause failure	Focus on common critical assets. Maintain redundancy.
Design to tolerate compromise	Assume compromised resources. Limit the need for trust.
Design to degrade gracefully	Manage resources (risk-) adaptively.
Design to crash early and recover quickly	Manage resources (risk-) adaptively.
Additional Design Principle	**Corresponding Cyber Resiliency Design Principle(s) from Table 1**
Design for integrity and availability	Determine ongoing trustworthiness.
Design to be threat independent	Expect adversaries to evolve.
Design to be vulnerability independent	Expect adversaries to evolve.
Design such that users will not seek to circumvent security and resilience features	Make unpredictability and deception user-transparent.
Design with distributed and localized decision support	Make resources location-versatile.
Integrate horizontally for resilience of the whole system / mission	Control visibility and use. Layer and partition defenses.
Design components with computational plasticity (alternative functional paths to achieve the computing results)	Support agility and architect for adaptability.
Design for simplicity and modularity to change easily / frequently	Support agility and architect for adaptability.

Table 25. Operational Principles Related to Cyber Resiliency

Operational Principle	Corresponding Cyber Resiliency Design Principle(s) from Table 1
Actively look for bad guys in the system	Assume compromised resources. Expect adversaries to evolve.
Leverage cyber intelligence to inform operations	Assume compromised resources. Expect adversaries to evolve.
Operate to reduce adversary's ability to act	Control visibility and use. Contain and exclude behaviors. Layer and partition defenses.
Operate to contain vulnerabilities	Control visibility and use. Contain and exclude behaviors. Layer and partition defenses.
Operate to reconstitute and recover quickly to an acceptable level of trust	Determine ongoing trustworthiness.
Prioritize operational TTPs based on mission assurance needs	Maintain situational awareness.
Balance / coordinate local defense with global defense	Control visibility and use. Contain and exclude behaviors. Layer and partition defenses.
Operate to control / limit the damage / consequences of attack	Manage resources (risk-) adaptively.
Operate with agility and alternative operational contingencies	Manage resources (risk-) adaptively.
Operate to confuse, deceive, and impede the adversary (but not the mission operators)	Make unpredictability and deception user-transparent.

Operational Principle	Corresponding Cyber Resiliency Design Principle(s) from Table 1
Monitor integrity and availability, and respond accordingly	Maintain situational awareness.
Train operators to understand cyber impact to mission operations to operate in ways that ensure mission execution success	Manage resources (risk-) adaptively.

B.4 Representative Program-Specific Statements

Cyber resiliency design principles will be selected and restated in terms that are meaningful to the system or program to which they are applied. These program-specific statements can be incorporated into the PPP, or into system requirements. Table 26 presents one set of program-specific strategic principles for applying cyber resiliency concepts and techniques; Table 27 presents a second. Table 28 presents a set of program-specific cyber resiliency requirements, illustrating corresponding controls in NIST SP 800-53R4.

Table 26. Examples of Program-Specific Cyber Resiliency Strategies

Program Specific Strategic Principle	Corresponding Cyber Resiliency Design Principle(s) from Table 1
Minimize adversary exposure. Protect hardware, software, all documentation.	Reduce attack surfaces.
Confuse the adversary. Be unpredictable: use diversity and deception to confuse the adversary, change the attack surface via non-persistence, use atypical solutions, etc.	Plan and manage diversity. Maximize transience; minimize persistence. Change or disrupt the attack surface. Make unpredictability and deception user-transparent.
Maximize segmentation. Apply both technical segmentation (wiring, network encryption, DAR encryption, least privilege, etc.) and functional segmentation (user systems from mission systems from control systems, etc.).	Contain and exclude behaviors. Layer and partition defenses.
Use special sauce. Include special capabilities, sensors, and defenses.	Plan and manage diversity.
Architect for extensibility. Efficiently accommodate changes to threat model, mission systems, and defensive capabilities.	Support agility and architect for adaptability.

Table 27. Alternate Examples of Cyber Resiliency Strategies

Program Specific Strategic Principle	Corresponding Cyber Resiliency Design Principle(s) from Table 1
Focus on mission-critical components and data.	Focus on common critical assets. Limit the need for trust.
Reduce exposure time.	Reduce attack surfaces. Maximize transience; minimize persistence. Change or disrupt the attack surface. Make unpredictability and deception user-transparent.
Change and evolve frequently.	Support agility and architect for adaptability. Maximize transience; minimize persistence. Change or disrupt the attack surface.
Use real-time continuous threat collection to inform operations.	Expect adversaries to evolve.
Prioritize detection and recovery.	(Note that detection and recovery capabilities are typically provided by security and disaster recovery requirements. In the context of the program for which this principle was defined, the strategy emphasized use of detection, response, recovery, and reconstitution mechanisms over, for example, protection mechanisms.) Manage resources (risk-) adaptively.

Table 28. Examples of Cyber Resiliency Requirements

Program Specific Cyber Resiliency Requirement	Corresponding Controls	Corresponding Cyber Resiliency Design Principle(s) from Table 1
Design to minimize or eliminate critical cyber assets.	SA-8, SA-14, CP-2(8)	Reduce attack surfaces. Limit the need for trust.
Use standardized protocols and interfaces, to minimize or eliminate reliance on specific products.	[none]	Support agility and architect for adaptability.
Distribute the traffic or message load across all available systems and services.	CP-2, SI-13(4)	Maintain redundancy. Make resources location-versatile.
Provide services by alternating between isolated, heterogeneous instances.	SC-29, SA-20, CP-7, SC-36, PM-8, CP-2(3)(4)(5), CP-10, IR-4(3), SI-13	Plan and manage diversity. Maximize transience; minimize persistence.
Independently monitor the health and status of each service instance.	SA-13, SI-4(17), SI-6	Leverage health and status data.
Be able to automatically refresh any service instance to a known good state.	SI-14(1), SI-4(7)(17), CP-2(3)(4)(5), CP-9	Maximize transience; minimize persistence. Determine ongoing trustworthiness.
Be able to automatically scale the number of service instances to changes in demand.	CP-2(2)(5), SC-6, SI-4(17), SI-14(1), IR-4(2), SC-5(2)	Manage resources (risk-) adaptively.
Be able to dynamically move services between platforms.	SC-30(3), CP-2(3)(4)(5)(6), IR-4(3)(9)	Make resources location-versatile. Manage resources (risk-) adaptively. Maximize transience; minimize persistence.
Be able to revert systems or service instances to a previous version.	CM-2(3), CP-2(3)(4)(5), CP-10, IR-4(2)(3)	Manage resources (risk-) adaptively. Determine ongoing trustworthiness.
Provide, and redirect suspicious user sessions/connections to, a deceptive environment.	SC-30, SC-30(4), SC-29(1), SI-4, IR-4(4)	Make unpredictability and deception user-transparent.
Store persistent data in a distributed, fault-tolerant data store implemented across a geographically diverse set of nodes.	SC-36, CP-2(6), CP-9, SI-7	Maintain redundancy. Make resources location-versatile.
Be able to reconstitute information to a known-good state.	CP-10	Manage resources (risk-) adaptively. Determine ongoing trustworthiness.
Be able to dynamically reconfigure the components based on the mission and threat context.	SC-6, AU-6(10), IR-4(2)	Manage resources (risk-) adaptively.
Be able to dynamically assign or associate a criticality security attribute to cyber resources, based on the resource's criticality to the current operational mission(s).	AC-4, AC-16, CP-2(8), SI-4(16), SC-6	Limit the need for trust. Control visibility and use.
Provide an alert management interface that correlates and prioritizes security and information system events based on the potential mission impact(s) of the affected services or cyber assets/systems.	SI-4(2)(12)(16)	Maintain situational awareness.
Be able to automatically disable any non-mission critical system, service, or network component involved in a potential incident or exhibiting suspicious behavior.	IR-4(5), SC-7(20), SC-6	Contain and exclude behaviors. Manage resources (risk-) adaptively. Determine ongoing trustworthiness.
Be able to create distinct enclaves for each new, concurrent mission/operation, providing isolation and independent management for the services and network/system resources supporting the mission/operation.	SC-7(20)(21)(22), SC-32, AC-4(21), AC-6(4)	Contain and exclude behaviors. Manage resources (risk-) adaptively.

Appendix C Details of Design Principles from Related Domains

This appendix presents details of design principles from the disciplines of security, resilience engineering, system survivability, and evolvability. These details are intended for systems engineers seeking to align design principles from different specialty disciplines, rather than for the general reader.

C.1 Security

Cyber resiliency assumes a foundation of good cybersecurity practices. Many of the cyber resiliency design principles are consistent with (and could be used with or as alternatives to) the security design principles. However, security design principles do not support achievement of the full range of cyber resiliency objectives, nor do they apply the full range of cyber resiliency techniques.

Security design principles have been presented in many forms since the 1972 Anderson Report [104]. In this section, security design principles from three major sources are presented:

- The Saltzer and Schroeder Security Design Principles [105] as rearticulated for the "Building Security In" initiative [106].
- The Security Design Principles from Appendix F of NIST SP 800-160 [1]. These subsume the principles offered in the 2005 study by the Naval Postgraduate School and the Information Sciences Institute of the University of Southern California [107].
- Security Design Principles for Digital Services, offered by the UK National Cyber Security Centre [108].

Additional sources are also noted.

C.1.1 Saltzer and Schroeder / Building Security In

A set of twelve design principles for cybersecurity, based on the work of Saltzer and Schroeder [105], has been widely used and generally accepted as good practice [106].[36] The "Building Security In" principles are presented in Table 29.[37] The first column presents the Security Design Principles. The second column discusses each principle in the context of cyber resiliency. The third column identifies cyber resiliency design principle(s) aligned with (i.e., capable of being used synergistically with) the security design principle. (When multiple cyber resiliency design principles can be aligned with the security design principle, the most easily aligned one is **bolded**.) The fourth column maps the principle to the cyber resiliency objective(s) it supports and to the cyber resiliency techniques which could be used in applying the principle. in the context of cyber resiliency. (When multiple objectives or techniques are identified, but one is primary, that one is **bolded**.)

Almost all the security design principles in [106] support the *Prevent / Avoid* cyber resiliency objective, either primarily or secondarily. This focus is due to the fact that conventional cybersecurity focuses on such threat actors such as individual criminals, hackers, or privilege-abusing insiders, and assumes that MCA can be detected. Several of the security design principles also relate to the Transform and Re-Architect objectives, based on the recognition that assumptions that underpin organizational processes and system architectures are rendered invalid by MCA.

[36] Other sources of security design principles also build on the work of Saltzer and Schroeder. One notable study, which identifies 29 principles, is [107].

[37] See [138] for a review of variants of the Saltzer and Schroeder principles.

Table 29. "Building Security In" Security Design Principles and Cyber Resiliency

Security Design Principle	Discussion in the Context of Cyber Resiliency	*Aligned Cyber Resiliency Design Principle(s)*	Related Cyber Resiliency *Objectives* & Techniques
Secure the Weakest Link	Cyber resiliency is based on the assumption that any system element – whether a human (e.g., user, administrator) or a system component (e.g., COTS or FOSS of unknown provenance) can be compromised. Looking for weak links and increasing security at those points is necessary but not sufficient. See also the discussion of Secure Administration [109].	*Focus on critical common assets.* ***Assume compromised resources.*** *Reduce attack surfaces.*	*Prevent / Avoid* **Segmentation / Isolation** Privilege Restriction
Defense in Depth	Defense-in-depth – supported by good management practices – is an integral part of the Coordinated Defense technique.	*Reduce attack surfaces: Recognize that the development / maintenance environment is part of the attack surface.* ***Layer and partition defenses.*** *Plan and manage diversity.*	***Prevent / Avoid*** *Continue* Redundancy with Diversity **Coordinated Defense** Segmentation / Isolation
Fail Securely	While the intention for security is to fail into a secure state, preserving confidentiality and integrity even with the loss of availability, some approaches (secure defaults, restoration of a secure state, checking return values for failure) are supported by Privilege Restriction and Substantiated Integrity. From the standpoint of cyber resiliency, failing securely is a strategy to minimize mission impacts of degradation, denial, and corruption.	***Focus on critical common assets.*** *Assume compromised resources.*	*Continue* ***Constrain*** *Reconstitute* **Adaptive Response** Privilege Restriction Substantiated Integrity
Least Privilege	Least Privilege is integral to the Privilege Restriction technique. Note, however, that Least Privilege has a strong connotation of access restriction based on relatively static properties, while Privilege Restriction considers mission criticality, which can change dynamically.	*Limit the need for trust.* *Control visibility and use.* ***Contain and exclude behaviors.***	***Prevent / Avoid*** *Constrain* **Privilege Restriction** Realignment (Purposing)
Separation of Privilege	Defining privileges and determining which need to be separated involves a combination of the Coordinated Defense and Privilege Restriction techniques.	*Limit the need for trust.* *Control visibility and use.* ***Contain and exclude behaviors.***	***Prevent / Avoid*** *Constrain* Coordinated Defense **Privilege Restriction** (Privilege-Based Usage Restriction)
Economy of Mechanism	Avoiding unnecessary functionality and the complexity that arises from trying to meet too many needs simultaneously also serves to reduce the attack surface.	*Reduce attack surfaces.*	*Prevent / Avoid* ***Re-Architect*** Realignment (Purposing, Offloading)

Security Design Principle	Discussion in the Context of Cyber Resiliency	*Aligned Cyber Resiliency Design Principle(s)*	Related Cyber Resiliency *Objectives* & Techniques
Least Common Mechanism	Avoid single points of failure and unnecessary shared services. Design to avoid common mode failures. [8]	*Support agility and architect for adaptability.* ***Reduce attack surfaces.*** *Make resources location-versatile.*	*Prevent / Avoid* ***Re-Architect*** Realignment (Purposing) Dynamic Positioning
Be Reluctant to Trust	Watch for evidence of compromise. Validate data and behavior where possible. Design to fail early and recover quickly [8].	*Assume compromised resources.* ***Limit the need for trust.*** *Determine ongoing trustworthiness.*	*Understand* *Prevent / Avoid* ***Transform*** Segmentation / Isolation Least Privilege Analytic Monitoring Substantiated Integrity
Never Assume Your Secrets Are Safe	Actively seek to thwart the adversary, and to gain intelligence about adversary TTPs.	***Assume compromised resources.*** *Limit the need for trust.*	*Understand* *Prevent / Avoid* ***Transform*** Segmentation / Isolation Least Privilege Analytic Monitoring Substantiated Integrity Deception
Complete Mediation	Consider persistent data, connections, services, and relationships to be high-value targets.	*Maximize transience; minimize persistence.*	*Prevent / Avoid* ***Constrain*** Non-Persistence
Ensure Psychological Acceptability	Make the interfaces to diverse mechanisms consistent. Make the mechanisms convenient for or transparent to mission users and system administrators, so that they will not be motivated to circumvent them. [8] Clearly communicate how cyber resilience improves reliability. Keep Unpredictability mechanisms hidden from end users.	*Reduce attack surfaces (keeping in mind ways in which users are part of the attack surface).* ***Make unpredictability and deception user-transparent.***	*Understand* *Prevent / Avoid* ***Transform*** **Coordinated Defense** Deception Realignment Unpredictability
Promote Privacy[38]	Sensitive persistent data can be a high-value target. Note that this concern applies to authentication and authorization data as well as to user and system data.	*Maximize transience; minimize persistence.* ***Contain and exclude behaviors.***	*Prevent / Avoid* *Constrain* ***Transform*** Non-Persistence **Privilege Restriction**

[38] While the discussion of this principle focuses on protecting the confidentiality of personally identifiable information (PII), privacy must not be reduced to that objective. See Appendix J of NIST SP 800-53R4 [29]. Privacy design principles are, however, outside the scope of this document.

C.1.2 NIST SP 800-160

Appendix F of NIST SP 800-160 [1] provides a more comprehensive set than those discussed above, defining design principles in three broad categories: Security Architecture and Design, Security Capability and Intrinsic Behaviors, and Life Cycle Security. The principles for Security Architecture and Design and Security Capability and Intrinsic Behaviors relate primarily to the *Prevent / Avoid* cyber resiliency objective. Several of the security design principles also relate to the *Transform* and *Re-Architect* objectives, based on the recognition that assumptions that underpin organizational processes and system architectures are rendered invalid by MCA. Tables 30-31 characterize the relationships between the Security Design Principles from Appendix F of NIST SP 800-160 and cyber resiliency techniques, using the following key:

- **U, C**: Application of the security design principle **uses** or **can use the** cyber resiliency technique (or one or more approaches to the technique);
- **S**: Application of the security design principle **supports** use of the cyber resiliency technique (i.e., one or more approaches use or can use security design principle); and
- **R**: Application of the security design principle **requires** use of the cyber resiliency technique (or one or more approaches).

Table 30 maps the Design Principles for Security Architecture and Design to the cyber resiliency techniques. Design principles for Security Architecture and Design assume that the purpose of each system component can be well defined, and thus use or can use the Purposing approach to Realignment. They also assume that components will be administered consistently with overarching security policies, and thus require or can use the Coordination & Consistency Analysis approach to Coordinated Defense. Finally, they can make use of Substantiated Integrity to strengthen their application. None of the architectural design principles involve applying the techniques of Analytic Monitoring, Deception, Diversity, Dynamic Positioning, Dynamic Representation, Redundancy, or Unpredictability. (Some of these techniques are more closely related to functionality than to architectural decisions.) However, as the table shows, different security design principles map to different approaches to the Realignment technique.

Table 30. Principles for Security Architecture and Design and Cyber Resiliency

	Cyber Resiliency Techniques										
Design Principles for Security Architecture and Design	Adaptive Response	Non-Persistence	Privilege Restriction	Segmentation / Isolation	Coordinated Defense	Realignment: Purposing	Realignment: Offloading	Realignment: Restriction	Realignment: Replacement	Substantiated Integrity	**Aligned Cyber Resiliency Design Principles**
Clear Abstraction											Control visibility and use.
Least Common Mechanism						C	C				Reduce attack surfaces.
Modularity and Layering			S	U	R	U					Layer and partition defenses.
Partially Ordered Dependencies					R	U				C	
Efficiently Mediated Access						U		U			
Minimized Sharing		C	S			U	C	U			Reduce attack surfaces.
Reduced Complexity						U					Reduce attack surfaces.
Secure Evolvability	S					C	C	C	C		Support agility and adaptability.
Trusted Components										C	Contain and exclude behaviors. Limit the need for trust.

Design Principles for Security Architecture and Design	Cyber Resiliency Techniques: Adaptive Response	Non-Persistence	Privilege Restriction	Segmentation / Isolation	Coordinated Defense	Realignment: Purposing	Realignment: Offloading	Realignment: Restriction	Realignment: Replacement	Substantiated Integrity	Aligned Cyber Resiliency Design Principles
Hierarchical Trust					C					C	
Inverse Modification Threshold								C	C	C	
Hierarchical Protection								C		C	
Minimized Security Elements			S			U		C			Limit the need for trust.
Least Privilege			U, S		R						Limit the need for trust.
Predicate Permission			U, S		R					C	Limit the need for trust.
Self-Reliant Trustworthiness				C	S			U		C	Limit the need for trust. Determine ongoing trustworthiness.
Secure Distributed Composition					S					C	Layer and partition defenses.
Trusted Communication Channels			C	C						R	Contain and exclude behaviors. Limit the need for trust.

Table 31 maps the Design Principles for Security Capability and Intrinsic Behaviors to the cyber resiliency techniques. These design principles support and can use Substantiated Integrity, and can use Redundancy (particularly with Diversity). They partially involve the use of Analytic Monitoring, Diversity, Dynamic Positioning, and some forms of Redundancy, but do not make use of Deception, Dynamic Representation, or Unpredictability.

Table 31. Design Principles for Security Capability and Intrinsic Behaviors and Cyber Resiliency

Design Principles for Security Capability and Intrinsic Behaviors	Cyber Resiliency Techniques: Adaptive Response	Analytic Monitoring	Diversity	Dynamic Positioning	Coordinated Defense	Realignment	Redundancy: Protected Backup & Restore	Redundancy: Surplus Capacity	Redundancy: Replication	Substantiated Integrity	Aligned Cyber Resiliency Design Principles
Continuous Protection	C				R						Determine ongoing trustworthiness.
Secure Metadata Management										C, S	
Self-Analysis		C								U	Determine ongoing trustworthiness. Maintain situational awareness.
Accountability and Traceability		S									Leverage health and status data.
Secure Defaults					C, S					C, S	
Secure Failure and Recovery			C	C			R, S			C	
Economic Security						S					
Performance Security	C			C		U		C	C		
Human Factored Security					S						
Acceptable Security						S					Make unpredictability and deception user-transparent.

Design principles for Life Cycle Security can be applied to all required or desired system properties, including cyber resiliency.

In addition to security design principles, Appendix F of NIST SP 800-160 also defines Approaches to Trustworthy System Development. These approaches apply cyber resiliency techniques in selective, targeted ways: The Reference Monitor Concept applies Segmentation / Isolation and Substantiated Integrity; Defense in Depth applies Coordinated Defense; and Isolation applies Segmentation / Isolation.

The security design principles or approaches to trustworthy system development do not make use of Deception and Unpredictability.

C.1.3 Security Design Principles for Digital Services

Table 32 maps the security design principles for digital services [108] to cyber resiliency objectives, technique, and design principles. Because these design principles specifically recognize the need for resilience in the face of Internet-based attacks, they provide better coverage of cyber resiliency.

Table 32. Security Design Principles for Digital Services and Cyber Resiliency

Goal	Security Design Principle	Cyber Resiliency *Objectives* and Techniques	Aligned Cyber Resiliency Design Principles
Understand your service	Understand your service and the data you will need to operate it	*Understand*	[Supports] Focus on common critical assets.
	Understand the role your suppliers play in securing your service	*Understand*	[Supports] Reduce attack surfaces.
	Have a clear, end-to-end understanding of your service and how it is accessed	*Understand*	[Supports] Focus on common critical assets.
	Ensure the governance arrangements for your system are clear	*Understand* *Prepare*	[Supports] Focus on common critical assets.
	Make it easy for everyone involved in designing and operating the service to know what their role is, and what constitutes acceptable behaviour	*Understand*	[Supports] Focus on common critical assets.
Make services hard to compromise	Validate or transform all external input before processing it	*Prevent / Avoid* Substantiated Integrity	Contain and exclude behaviors.
	Render untrusted content in a disposable environment	*Prevent / Avoid* Non-Persistence Segmentation / Isolation	Contain and exclude behaviors. Maximize transience.
	Only import trustworthy software and verify its legitimacy	*Prevent / Avoid* Substantiated Integrity	Assume compromised resources.
	Design for easy maintenance	*Re-Architect* Coordinated Defense	Support agility and extensibility.
	Use tried and tested frameworks rather than reinventing the wheel	*Re-Architect* Coordinated Defense	Plan and manage diversity. [Conflicts with some approaches to Diversity]
	Reduce your attack surface	*Prevent / Avoid*	Reduce attack surfaces.
	Users with access to data should be identified and authenticated	*Prevent / Avoid* Privilege Restriction	Control visibility and use.
	Make it easy for administrators to manage access control	*Prevent / Avoid* Coordinated Defense	
	Don't design or implement your own cryptographic protections	*Prevent / Avoid* Deception	Plan and manage diversity. [Conflicts with some approaches to Diversity] Make unpredictability and deception user-transparent.
	Protect your management/operations environments from spear-phishing and watering-hole attacks	*Prevent / Avoid* Segmentation / Isolation	Assume compromised resources.
	Make it easy for users to do the right thing	*Prevent / Avoid* Coordinated Defense Privilege Restriction	Plan and manage diversity. Make unpredictability and deception user-transparent.

Goal	Security Design Principle	Cyber Resiliency *Objectives* and Techniques	Aligned Cyber Resiliency Design Principles
Reduce the impact of compromise	Build your service using a segmented approach	*Constrain* Segmentation / Isolation	Contain and exclude behaviors. Layer and partition defenses.
	Anonymise data when it's exported to reporting tools	*Constrain*	Control visibility and use.
	Don't deploy applications or design functionality which enable the running of arbitrary queries against your data set	*Constrain* Privilege Restriction Substantiated Integrity	Contain and exclude behaviors.
	Do not implement functionality that would be damaging if used by unauthorised individuals	*Constrain* *Re-Architect* Privilege Restriction	Expect adversaries to evolve. Contain and exclude behaviors.
	Avoid creating caches or temporary stores of data within the service	*Constrain*	Control visibility and use. Use location-transparent resources.
	Encrypt partially completed forms under a key controlled by the user	*Constrain* Deception: Obfuscation	Control visibility and use.
	Regularly rebuild components that have considerable access to data over a long period of time	*Constrain* Non-Persistence	Maximize transience. Change or disrupt the attack surface.
	Only handle data which is essential to your service	*Constrain* *Re-Architect*	Reduce attack surfaces.
	Retain data for the minimum time necessary	*Constrain* Non-Persistence	Maximize transience.
	Avoid displaying unnecessary or bulk data to users	*Constrain*	Control visibility and use.
	Data model design should allow for tokenisation	*Continue* *Constrain*	Control visibility and use.
	Throttle access to data in line with the role and requirements of the user	*Constrain* Privilege Restriction	Control visibility and use.
	Make it easy to recover following a compromise	*Reconstitute*	Manage resources (risk-) adaptively.
	Design the service to support separation of duties	*Constrain* Privilege Restriction	Control visibility and use.
	Beware of creating a 'management bypass'	*Constrain* Privilege Restriction	Control visibility and use.
Make compromise easy to detect	Ensure that all relevant security events and logs are collected for analysis	*Understand* Analytic Monitoring	Maintain situational awareness.
	Design simple communication flows between your components	*Prevent / Avoid*	
	Detect and prevent malware command and control	*Prevent / Avoid* Non-Persistence Substantiated Integrity	
	Separate your event analysis systems from the core components of the service	*Constrain* Segmentation / Isolation	Layer and partition defenses.
	Make it difficult for attackers to attempt to detect your security rules through external testing	*Prevent / Avoid* Privilege Restriction	Reduce attack surfaces. Control visibility and use.
	Use transaction monitoring to provide additional security for high-risk transactions in digital services	*Prevent / Avoid* Analytic Monitoring	Maintain situational awareness.
	Make it difficult for attackers to probe security-monitoring rules by not stopping transactions immediately on suspicious activity	*Prevent / Avoid* Adaptive Response	Manage resources (risk-) adaptively.

C.1.4 Other Sources

Hughes and Cybenko, building on prior work by the Air Force Research Laboratory (AFRL), identify three tenets for secure design, applicable to CPS as well as to enterprise systems [110] [111], to serve as the basis for cybersecurity metrics. These are:

1. "Focus on What's Critical – systems should include only essential functions."
2. "Move Key Assets Out-of-Band – make mission essential elements and security controls difficult for attackers to reach logically and physically."
3. "Detect, React, Adapt – confound the attacker by implementing sensing system elements with dynamic response technologies."

The first tenet aligns with *Focus on common critical assets* and *Reduce attack surfaces*. The second aligns with *Reduce attack surfaces*, *Control visibility and use*, *Contain and exclude behaviors*, and *Plan and manage diversity*. The third aligns with *Make resources location-versatile*, *Manage resources (risk-) adaptively*, and *Change or disrupt the attack surface*.

The Open Web Application Security Project (OWASP) has identified ten Security-by-Design principles [112]. These are *Minimize attack surface area*, *Establish secure defaults*, *Principle of Least privilege*, *Principle of Defense in depth*, *Fail securely*, *Don't trust services*, *Separation of duties*, *Avoid security by obscurity*, *Keep security simple*, and *Fix security issues correctly*. Some of these align with cyber resiliency design principles; others are restatements of security principles assuming a conventional threat.

Design principles have also been offered for specific security mechanisms, including attestation [113] and encryption [114]. These can be useful in defining system- or program-specific security and resiliency design principles.

C.1.5 Cyber Resiliency Gaps in Security Design Principles

As Table 33 shows, none of the sets of security design principles covers the full range of concerns addressed by the cyber resiliency design principles. In this table, an X indicates the potential for alignment between at least one of the security design principles in the set and the cyber resiliency design principle. However, alignment is not guaranteed, and depends on how the design principles are restated and refined in the context of the system or program to which they are applied.

Table 33. Security Design Principles and Cyber Resiliency Design Principles

Cyber Resiliency Design Principle	Security				
	Building Security In	Security Architecture & Design	Security Capabilities & Intrinsic Behaviors	Security Design Principles for Digital Services	OWASP
Focus on common critical assets	X			X	
Support agility and architect for adaptability		X		X	
Reduce attack surfaces	X	X		X	X
Assume compromised assets	X			X	X
Expect adversaries to evolve				X	
Limit the need for trust	X	X			
Control visibility and use		X		X	X

Cyber Resiliency Design Principle	Security				
	Building Security In	Security Architecture & Design	Security Capabilities & Intrinsic Behaviors	Security Design Principles for Digital Services	OWASP
Contain and exclude behaviors	X			X	
Layer and partition defenses	X	X		X	X
Plan and manage diversity	X			X	
Maintain redundancy					
Make resources location-versatile					
Leverage health and status data			X		
Maintain situational awareness			X	X	
Manage resources (risk-) adaptively				X	
Maximize transience; minimize persistence	X			X	
Determine ongoing trustworthiness	X	X	X		
Change or disrupt the attack surface				X	
Make unpredictability and deception user-transparent			X		

C.2 Resilience Engineering

The discipline of Resilience Engineering has produced a set of design principles [13], related to four key attributes of a resilient system (where "system" explicitly means a human-made system). The key attributes are

- Capacity: the attribute of a system that allows it to withstand a threat. Capacity is achieved via Absorption, Physical Redundancy, Functional Redundancy, and Layered Defense.
- Flexibility: the attribute of a system that allows it to restructure itself in the face of a threat. Flexibility is achieved via Reorganization, Human Backup, Complexity Avoidance, and Drift Correction.
- Tolerance: the attribute of a system that allows it to degrade gracefully following an encounter with a threat. Tolerance is achieved via Localized Capacity, Loose Coupling, Neutral State, and Reparability.
- Cohesion: the attribute of a system that allows it to operate before, during, and after an encounter with a threat. Cohesion is achieved via Inter-Node Interaction.

The cyber resiliency design principles in Table 1 collectively support Resilience Engineering design principles, specifically in the context of cyber threats. However, the Resilience Engineering design principles do not cover the full set of cyber resiliency concerns.

C.2.1 Resilience Design Principles from the Systems Engineering Body of Knowledge

Table 34 presents the resilience engineering design principles from the Systems Engineering Body of Knowledge (SEBoK, [13]) identifies cyber resiliency design principles that could be used in conjunction with these principles, and identifies related cyber resiliency objectives and techniques. It must be emphasized that the model of disruption used to define these design principles does not represent a

stealthy, persistent adversary. Thus, Deception relates only to one resilience engineering design principle, and Unpredictability to none.

Table 34. Resilience Engineering Design Principles

Resilience Engineering Design Principle	*Aligned Cyber Resiliency Design Principles*	Related Cyber Resiliency *Objectives* & Techniques
Absorption: Include adequate margin to withstand a design-level threat.	Note: The Absorption design principle requires the application of other specialties, such as reliability and safety. *Maintain redundancy.*	*Continue* Redundancy (Surplus Capacity)
Physical Redundancy: Make critical components physically redundant.	*Focus on critical common assets.* *Maintain redundancy.*	***Prevent / Avoid*** *Continue* Redundancy
Functional Redundancy: Duplicate critical functions using different means.	*Plan and manage diversity.* *Maintain redundancy.*	*Prevent / Avoid* ***Continue*** *Constrain* *Reconstitute* Diversity Redundancy
Layered Defense: Avoid single points of failure.	*Focus on critical common assets.*	***Prevent / Avoid*** *Continue* **Coordinated Defense** Redundancy
Reorganization: Enable the system to change its own architecture before, during, or after the encounter with the threat.	*Provide agility.* *Expect the adversary to evolve.* *Manage resources (risk-) adaptively.*	*Prevent / Avoid* ***Continue*** *Transform* *Re-Architect* **Adaptive Response** Coordinated Defense Dynamic Positioning Non-Persistence Realignment Segmentation / Isolation
Human Backup: Enable humans to back up automated systems, especially when unprecedented threats are involved.	*Assume compromised resources.* *Provide agility.*	*Reconstitute* ***Transform*** **Adaptive Response** Coordinated Defense Redundancy
Complexity Avoidance: Avoid complex elements, such as software or humans, except where they are essential.	*Reduce the attack surface.*	*Understand* *Transform* ***Re-Architect*** Realignment
Drift Correction: Correct detected threats or conditions before the encounter with the threat.	*Reduce the attack surface.* *Expect the adversary to evolve.* *Maintain situational awareness.* *Change or disrupt the attack surface.* *Leverage health and status data.*	*Understand* ***Prevent / Avoid*** **Adaptive Response** Analytic Monitoring Deception Dynamic Positioning Non-Persistence Realignment Segmentation / Isolation

Resilience Engineering Design Principle	*Aligned Cyber Resiliency Design Principles*	Related Cyber Resiliency *Objectives* & Techniques
Localized Capacity: Concentrate system functionality in individual nodes which stay independent of other nodes.	*Assume compromised resources.* *Limit the need for trust.* *Maximize transience; minimize persistence.* *Control visibility and use.* *Contain and exclude behaviors.*	*Prevent / Avoid* ***Constrain*** Non-Persistence Realignment **Segmentation / Isolation**
Loose Coupling: Check cascading failures by inserting pauses between nodes.	*Focus on critical common assets.* *Assume compromised resources.* *Limit the need for trust.* *Maximize transience; minimize persistence.* *Control visibility and use.* *Contain and exclude behaviors.*	*Prevent / Avoid* ***Constrain*** Adaptive Response Analytic Monitoring Dynamic Positioning Non-Persistence **Segmentation / Isolation**
Neutral State: Bring the system to a neutral state before taking actions.	*Determine ongoing trustworthiness.*	*Reconstitute* Adaptive Response **Substantiated Integrity**
Reparability: Enable the system to be repaired to provide full or partial functionality.	*Make resources location-versatile.* *Manage resources (risk-) adaptively.* *Determine ongoing trustworthiness.*	*Reconstitute* Adaptive Response **Substantiated Integrity**
Inter-Node Interaction: Enable the nodes of a system to communicate, cooperate, and collaborate.	*Provide agility.* *Distribute (rather than localize) resources.* *Maintain situational awareness.* *Leverage health and status data.*	*Understand* Analytic Monitoring Dynamic Representation

C.2.2 Resilience Design Principles for a Broader Context

Resilience can also be understood in a broader context. The ten Resilient Design Principles developed by the Resilient Design Institute for buildings and communities [115] can considered in the context of cyber resiliency. Since these principles were not created for cyber systems, some of them when considered in the context of mission dependence on cyber resources apply to analytic processes in the Structured Cyber Resiliency Analysis Methodology (SCRAM, [5]) rather than to system architecture.

In Table 35, the first column presents the Resilient Design Principles. The second column discusses each principle in the context of cyber resiliency. The third column maps the principle to the cyber resiliency objective(s) it supports, the cyber resiliency techniques (if any[39]) which could be used in applying the principle, and the analytic processes in SCRAM it could inform.

Table 35. Resilient Design Principles and Cyber Resiliency

[39] Because the Resilient Design Principles are not oriented to cyber systems, some principles do not correspond to any techniques.

Resilient Design Principle	Discussion in the Context of Cyber Resiliency	Related Cyber Resiliency *Objectives*, Techniques, or Processes
1. Resilience transcends scales.	**Consider cyber resilience at the component, system-of-systems, and sector scale as well as at the system scale.** To improve cyber resilience at multiple scales, ensure that resilience measures – technologies and organizational or mission processes – can be made consistent at different scales. Minimize the assignment of privileges, to reduce potential consequences of compromise.	*Transform* *Re-Architect* Coordinated Defense Privilege Restriction Cyber Resiliency Analysis (Understand Mission & Threat Context; Analyze Architecture & Mission Threads)
2. Resilient systems provide for basic human needs.	**Recognize the primary importance of mission assurance.** Analyze mission dependencies on cyber systems and constituent resources.	Cyber Resiliency Analysis (Understand Mission & Threat Context; Analyze Architecture & Mission Threads)
3. Diverse and redundant systems are inherently more resilient.	**Apply Redundancy in conjunction with Diversity.** Recognize that Diversity and Redundancy introduce complexity, which must be managed.	*Re-Architect* Adaptive Response Coordinated Defense Redundancy with Diversity
4. Simple, passive, and flexible systems are more resilient.	**Avoid unnecessary complexity.**	*Re-Architect* Privilege Restriction Realignment (Purposing)
5. Durability strengthens resilience.	**Incorporate existing mechanisms rather than fight them.** Recognize that legacy technologies will not disappear, and use them in a manner consistent with cyber resilience.	Cyber Resiliency Analysis (Establish Initial Cyber Resiliency Baseline)
6. Locally available, renewable, or reclaimed resources are more resilient.	**Minimize the attack surface.** Avoid unnecessary external interfaces and the complexity that arises from trying to meet too many needs simultaneously.	*Re-Architect* Realignment (Purposing, Offloading)
7. Resilience anticipates interruptions and a dynamic future.	**Assume compromise, and minimize negative impacts.** Build to achieve cyber resilience objectives, rather than to thwart specific threats or attack activities.	*Re-Architect* Realignment (Purposing) Cyber Resiliency Analysis (Define & Analyze Specific Alternatives)
8. Find and promote resilience in nature.	**Incorporate existing mechanisms rather than fight them.** (See Principle 5.) Analyze how systems are used, and alternative ways they are used under stress, to identify alternatives for operational or procedural improvements.	*Transform* Cyber Resiliency Analysis (Establish Initial Cyber Resiliency Baseline; Define & Analyze Specific Alternatives)
9. Social equity and community contribute to resilience.	**Engage all stakeholders in determining cyber resiliency needs.** Ensure that relative priorities and constraints on potential solutions (e.g., POET factors) are identified. Reflect and respect stakeholder equities when identifying and analyzing alternatives.	*Transform* *Re-Architect* Realignment (Purposing) Cyber Resiliency Analysis (Understand Mission & Threat Context)
10. Resilience is not absolute.	**Engage all stakeholders in determining cyber resiliency needs.** (See Principle 9.)	Cyber Resiliency Analysis (Recommend Courses of Action) *Transform* *Re-Architect*

C.2.3 Other Sources of Resilience Design Principles

Resilience-related design principles can also be found in infrastructure resilience efforts. The Cyber Operations, Analysis, and Research (COAR) team at Argonne National Laboratory has identified six overarching principles or factors for cyber resilience in active defense techniques: adaptability, redundancy, fault tolerance, mitigation, recoverability, and survivability [116]. These principles correspond to cyber resiliency techniques and objectives.

Table 36. Factors for Cyber Resilience and Design Principles

Factors for Cyber Resilience in Active Defense Techniques	*Aligned Cyber Resiliency Design Principles*	Related Cyber Resiliency *Objectives* & Techniques
Adaptability: Be able to change configuration or runtime parameters in response to an external event.	*Manage resources (risk-) adaptively.* *Change or disrupt the attack surface.*	***Continue*** Adaptive Response
Redundancy: Build multiple resources that serve the same function and can replace each other in the event of the loss of primary system resources.	*Plan and manage diversity.* *Maintain redundancy.* *Change or disrupt the attack surface.*	*Prevent / Avoid* ***Continue*** *Reconstitute* Redundancy
Fault Tolerance: Achieve dependability by adapting to failure.	*Plan and manage diversity.* *Maintain redundancy.* *Manage resources (risk-) adaptively.*	***Continue*** *Constrain* Diversity Redundancy
Mitigation: Be able to respond to a failure or support a human in responding to that failure.	*Contain and exclude behaviors.* *Manage resources (risk-) adaptively.*	***Continue*** Adaptive Response **Coordinated Defense** Redundancy Segmentation / Isolation
Survivability: Be able to maintain or provide graceful degradation of operational goals when under attack.	*Support agility and architect for adaptability.* *Expect the adversary to evolve.* *Manage resources (risk-) adaptively.*	*Prevent / Avoid* ***Continue*** **Adaptive Response** Coordinated Defense Dynamic Positioning Non-Persistence Segmentation / Isolation
Recoverability: Design with strategies to provide a means to restore operations quickly and effectively following a service disruption.	*Assume compromised resources.* *Provide agility.*	***Reconstitute*** **Adaptive Response** Coordinated Defense Redundancy Segmentation / Isolation Substantiated Integrity

At another level of detail, one possible element of a risk management strategy is a strategy for resilient response, i.e., for determining the course of action to be taken by a system, a cyber defense team, or an organization in response to indications of adversary activity, which maximizes mission resilience. Actions can be characterized in terms of expected or potential effects on adversary activities, as well as the high-level strategy (or strategies) for resilient response enabled. Table 37 presents a few examples of potential strategies for resilient response. See Table H-6 of [1] for more detail on the potential effects of cyber resiliency techniques on adversary activities.

Table 37. Examples of Strategies for Resilient Response

Strategy	Description	Examples	Related Techniques
Self-heal and Adapt	Maintain as much functionality or capacity as possible, while taking actions that counteract the effects of adversary activities	Refresh virtual machine, thus expunging malware Refresh connections, thus eliminating man-in-the-middle presence	Adaptive Response Non-Persistence
Quarantine	Remove resources from mission support functions, for future analysis and remediation	Dynamically reconfigure internal subnets, to isolate suspected devices and processes Dynamically blacklist ports, protocols, or services Quarantine files, making them inaccessible to all but forensic analysis functions	Adaptive Response Segmentation / Isolation
Discard	Remove resources from mission support permanently	Blacklist resource (e.g., via device identifier) Destroy resource	Segmentation / Isolation

C.3 Survivability

This section updates the mapping of design principles for survivable systems architecture to cyber resiliency objectives and techniques. It also briefly discusses the System Survivability Key Performance Parameter, which is relevant to DoD systems.

C.3.1 Survivable Systems Architecture

Survivability has been defined as

> "the ability of a system to minimize the impact of a finite-duration disturbance on value delivery (i.e., stakeholder benefit at cost), achieved through (1) the reduction of the likelihood or magnitude of a disturbance, (2) the satisfaction of a minimally acceptable level of value delivery during and after a disturbance, and/or (3) a timely recovery." [117]

This definition assumes a finite-duration disturbance, rather than the potentially unbounded duration of cyber adversary activities within a compromised system. Table 38 identifies relationships between principles for survivable systems architecture [14], the cyber resiliency design principles in Section 2, and the cyber resiliency objectives and techniques, updating Appendix B of [118]. When multiple objectives or techniques are related to a survivability principle, the most significant one is **bolded**.

Table 38. Design Principles for Survivable Systems and Cyber Resiliency

Principle for Survivable Systems	*Aligned Cyber Resiliency Design Principles*	Related Cyber Resiliency *Objectives* &Techniques
Reduce Susceptibility		
Prevention: suppress a future or potential disturbance	*Reduce attack surfaces.* *Limit the need for trust.*	*Prepare* ***Prevent / Avoid*** Coordinated Defense Privilege Restriction
Mobility: relocate to avoid detection by an external change agent	*Support agility and architect for adaptability.* *Make resources location-versatile.* *Change or disrupt the attack surface.*	*Prevent / Avoid* **Dynamic Positioning** Unpredictability

Principle for Survivable Systems	*Aligned Cyber Resiliency Design Principles*	Related Cyber Resiliency *Objectives* &Techniques
Concealment: reduce the visibility of a system from an external change agent	*Control visibility and use.* *Make unpredictability and deception user-transparent.*	*Prevent / Avoid* **Deception** (Obfuscation) Segmentation / Isolation
Deterrence: dissuade a rational external change agent from committing a disturbance	*Change or disrupt the attack surface.* *Make unpredictability and deception user-transparent.* Note that implementation of these principles increases the adversary's work factor; the extent to which this serves as a deterrent depends on the adversary.	*Prevent / Avoid* **Adaptive Response** Deception Unpredictability
Preemption: suppress an immediate disturbance	*Contain and exclude behaviors.* *Change or disrupt the attack surface.*	***Constrain*** *Reconstitute* **Adaptive Response** Coordinated Defense Segmentation / Isolation
Avoidance: maneuver away from disturbance	*Make resources location-versatile.* *Manage resources (risk-) adaptively.* *Maximize transience; minimize persistence.* *Change or disrupt the attack surface.*	*Constrain* Adaptive Response Coordinated Response **Dynamic Positioning** Non-Persistence
Vulnerability Reduction		
Hardness: resist deformation	*Layer and partition defenses.* *Limit the need for trust.* *Contain and exclude behaviors.*	*Prevent / Avoid* Coordinated Defense **Privilege Restriction** Segmentation / Isolation
Redundancy: duplicate system functions to increase reliability	*Maintain redundancy.*	*Continue* Adaptive Response **Redundancy** Dynamic Positioning
Margin: allow extra capacity to maintain value delivery despite losses	*Maintain redundancy.*	*Continue* Adaptive Response **Redundancy** Dynamic Positioning
Heterogeneity: vary system elements to mitigate homogeneous disturbances	*Plan and manage diversity.*	*Prevent / Avoid* ***Continue*** Diversity
Distribution: separate critical system elements to mitigate local disturbances	*Make resources location-versatile.* *Contain and exclude behaviors.*	***Continue*** *Constrain* **Dynamic Positioning** Segmentation / Isolation
Failure Mode Reduction: eliminate system hazards through intrinsic design	*Assume compromised resources.* *Reduce the attack surface.*	*Re-Architect* Non-Persistence Privilege Restriction Realignment
Fail-Safe: prevent or delay degradation via physics of incipient failure	*Focus on critical common assets.* *Layer and partition defenses.* *Determine ongoing trustworthiness.*	***Constrain*** *Reconstitute* Coordinated Defense Realignment **Substantiated Integrity**
Evolution: alter system components to reduce disturbance effectiveness	*Provide agility.* *Expect the adversary to evolve.* *Focus on critical common assets.*	*Re-Architect* Realignment (Replacement)

C.3.2 System Survivability Key Performance Parameter

The JCIDS Manual [119] establishes the System Survivability (SS) Key Performance Parameter (KPP) "to ensure the system maintains its critical capabilities under applicable threat environments." In this context, "threat" is by definition adversarial:

> "Threat – The sum of the potential strengths, capabilities, and strategic objectives of any adversary which can limit or negate mission accomplishment or reduce force, system, or equipment effectiveness. It does not include (a) natural or environmental factors affecting the ability or the system to function or support mission accomplishment, (b) mechanical or component failure affecting mission accomplishment unless caused by adversary action, or (c) program issues related to budgeting, restructuring, or cancellation of a program. (Proposed for JP 1-02. SOURCE: CJCSI 5123.01/3170.01)" ([119], Glossary)

System survivability attributes support resilience in the following sense:

> "Resilience is the ability of the collection of systems to support the functions necessary for mission success in spite of hostile action or under adverse conditions. An architecture is "more resilient" if it can provide these functions with higher probability, shorter periods of reduced capability, and across a wider range of scenarios, conditions, and threats." ([119], Appendix C to Enclosure D, Content Guide for the System Survivability KPP)

The JCIDS Manual identifies a variety of potential attributes, some of which relate to design principles for survivable systems architectures. These include situational awareness, maneuverability, durability, added protection / hardening, and redundancy. Appropriate cyber attributes to increase resiliency for a given system, program, or system-of-systems are presented in the draft Cyber Survivability Endorsement Guide [120]. A mapping between the cyber resiliency design principles and these attributes can be made available to DoD organizations. The cyber resiliency design principles can serve as a starting point for identifying key system attributes (KSAs).

C.4 Evolvability

Ricci et al. have identified twelve evolvability design principles, based on analysis of military SoS [15]. Of these, the first four are characterized as *strategic* design principles – they "are not directly related to the architecture of the SoS, but rather suggest strategies for facilitating the achievement of evolvability properties." The remaining principles are characterized as *structural*, directly influencing the structure of a SoS. Table 39 re-interprets the twelve evolvability design principles in the context of cyber resiliency. The first column presents the evolvability design principles. The second column discusses each principle in the context of cyber resiliency, focusing on the cyber resiliency techniques which could be used in applying the principle. All the evolvability design principles relate to the Transform and Re-Architect cyber resiliency objectives.

Table 39. Evolvability Design Principles and Cyber Resiliency

Evolvability Design Principle	Discussion in the Context of Cyber Resiliency
Leverage ancestry: Employ successful design choices of assets, capabilities and/or operations from all prior generations of the system	Does not directly correspond to any of the cyber resiliency design principles. Does implicitly indicate the importance of protecting – and using Substantiated Integrity in – the development environment, to avoid re-use of compromised components and to be able to track component provenance.
Disruptive Architectural Overhaul: Re-architecting significant portions of the existing system or program at the same time in order to reduce the negative impact that making many smaller changes would have	Applying this principle aligns well with *Reduce attack surfaces*. Multiple small changes over time typically increase the size of critical components, introduce unused features, and create unnecessary complexity.

Evolvability Design Principle	Discussion in the Context of Cyber Resiliency
Mimicry: Imitate or duplicate successful design choices of assets, capabilities and/or operations from other systems / domains for a similar purpose	This principle has been applied in various research agendas and research approaches related to cyber resiliency. In particular, it has informed work in Moving Target Defense (MTD, applying the Adaptive Response, Dynamic Positioning, and Non-Persistence techniques), as well as Deception.
Resource Exaptation: Repurposing assets or design choices from prior generations or other systems / domain in order to provide capabilities for which they were not originally selected	Applying this principle is problematic in the context of cyber security and cyber resiliency. It aligns with *Support agility and architect for adaptability*. One of the approaches to the Realignment technique is Purposing: ensuring that cyber resources are used consistent with mission purposes. Resource Exaptation can increase the attack surface, exposing weaknesses which in prior or different environments were not exposed.
Decentralization: Distribute assets, capabilities and/or operations to appropriate multiple locations, rather than have them located in a single location	Applying this principle aligns well with *Make resources location-versatile*.
Targeted Modularity: Isolate parts of the system to reduce interdependencies in order to limit undesirable effects caused by either uncertainties or intentional changes	Applying this principle aligns well with *Contain and exclude behaviors.*
Integrability: Designing interfaces for compatibility and commonality to enable effective and efficient integration of upgraded / new system components and constituents	Applying this principle can support *Determine ongoing trustworthiness*.
Reconfigurability: Create intentional similarities in form and/or function of various system assets, capabilities, and/or operations to facilitate reuse or reallocation	Applying this principle can support *Manage resources (risk-) adaptively*, and can use *Maximize transience; minimize persistence*.
Redundancy: Intentional duplication of selected assets, capabilities and/or operations to enable their future redistribution without compromising existing requirements	Consistent with *Maintain redundancy*.
Scalability: Make design choices that allow scaling of resources and/or assets up or down in order to accommodate uncertainties and emergent needs	Applying this principle can support *Manage resources (risk-) adaptively*.
Margin: Architect for intentional excess capacity in specific capabilities and/or operations to meet emergent needs without compromising existing requirements (i.e., meet or exceed future requirements)	Consistent with *Maintain redundancy*.
Slack: Intentionally under-allocate or over-allocate specific available assets and/or resources in order to reserve excess capacity for accommodating uncertainties (i.e., prevent violation of constraints)	Consistent with *Maintain redundancy*.

C.5 Safety

System Safety Engineering is grounded in processes rather than principles [32] [121] [122]. However, the system safety order of precedence [32] can be viewed as an overall strategy, with each item in the order being viewed as a strategic design principle. These are mapped to the cyber resiliency design principles in Table 40.

Table 40. System Safety Principles and Cyber Resiliency

Safety Design Principle	Cyber Resiliency *Objectives* and Techniques	Cyber Resiliency Design Principle(s)
1. Eliminate hazards through design selection	*Re-Architect* Realignment	Reduce attack surfaces.
2. Reduce risk through design alteration	*Re-Architect* Realignment	Limit the need for trust. Control visibility and use. Contain and exclude behaviors.
3. Incorporate safety devices	*Constrain* Segmentation / Isolation	Assume compromised resources. Layer and partition defenses.
4. Provide warning devices	*Understand* Analytic Monitoring Dynamic Representation Substantiated Integrity	Maintain situational awareness. Leverage health and status data. Monitor ongoing trustworthiness.
5. Develop procedures and training	*Prepare* Adaptive Response Coordinated Defense	Layer and partition resources. Plan and manage diversity.

Appendix D Glossary and Abbreviations

D.1 Glossary

Term	Definition
Advanced Persistent Threat	An adversary with sophisticated levels of expertise and significant resources, allowing it through the use of multiple different attack vectors (e.g., cyber, physical, and deception) to generate opportunities to achieve its objectives, which are typically to establish and extend footholds within the information technology infrastructure of organizations for purposes of continually exfiltrating information and/or to undermine or impede critical aspects of a mission, program, or organization, or place itself in a position to do so in the future; moreover, the advanced persistent threat pursues its objectives repeatedly over an extended period of time, adapting to a defender's efforts to resist it, and with determination to maintain the level of interaction needed to execute its objectives. [36] [31]
Adversary	Individual, group, organization, or government that conducts or has the intent to conduct detrimental activities. [31] [101]
Align (design principles)	Combine related design principles from different specialty disciplines into a system- or program-specific design principle, providing amplifying discussion to clarify what the design principle means in the context of the system or program, its mission requirements and operational environment, and the risks it can serve to mitigate.
Anticipate	(Cyber Resiliency Goal) Maintain a state of informed preparedness for adversity [3] [1]
Asset	(1) An item of value to achievement of organizational mission/business objectives. *Note 1:* Assets have interrelated characteristics that include value, criticality, and the degree to which they are relied upon to achieve organizational mission/business objectives. From these characteristics, appropriate protections are to be engineered into solutions employed by the organization. *Note 2:* An asset may be tangible (e.g., physical item such as hardware, software, firmware, computing platform, network device, or other technology components) or intangible (e.g., information, data, trademark, copyright, patent, intellectual property, image, or reputation). [1] (2) An item, capability, or service of value to achievement of organizational mission or business objectives.
Attack Surface	The set of resources and vulnerabilities that are exposed to potential attack "Attack surfaces of information systems are exposed areas that make those systems more vulnerable to cyber attacks. This includes any accessible areas where weaknesses or deficiencies in information systems (including the hardware, software, and firmware components) provide opportunities for adversaries to exploit vulnerabilities." ([29], Supplemental Guidance for SA-11(6))
Attribute	An attribute is any distinctive feature, characteristic, or property of an object that can be identified or isolated quantitatively or qualitatively by either human or automated means. Source: ISO/IEC 27000 [31]
Component	(1) See *system element*. [1] (2) A part of a system that can be replaced or managed separately from other parts of the system. Examples of components include hardware devices, embedded devices (e.g., sensors, controllers, medical devices such as pacemakers, vehicle automation such as collision avoidance), desktop or laptop computers, servers, routers, firewalls, virtual machine monitors (VMMs) or hypervisors, operating systems (OSs), applications, and databases. When "system" is construed as a socio-technical system, examples also include people and separately managed processes. [3]
Constrain	(Cyber Resiliency Objective) Limit damage from adversity [3] [1]

Term	Definition
Continue	(Cyber Resiliency Objective) Maximize the duration and viability of essential mission or business functions during adversity [3] [1]
Conventional Cyber Threat (or Cyber Adversary)	An adversary addressed by established standards of good practice, and in particular by the baselines in NIST SP 800-53R4 [29]. Conventional cyber adversaries include hackers using malware and TTPs easily recognized by malware and intrusion detection systems, as well as insiders abusing their privileges.
Criticality	An attribute assigned to an asset that reflects its relative importance or necessity in achieving or contributing to the achievement of stated goals. [1]
Criticality Analysis	An end-to-end functional decomposition performed by systems engineers to identify mission critical functions and components. Includes identification of system missions, decomposition into the functions to perform those missions, and traceability to the hardware, software, and firmware components that implement those functions. Criticality is assessed in terms of the impact of function or component failure on the ability of the component to complete the system missions(s). [37]
Cyber	A modifier that indicates a presence in, or involvement with, cyberspace, due to actual or potential accessibility via network communications. [3]
Cyber Resiliency	The ability to anticipate, withstand, recover from, and adapt to adverse conditions, stresses, attacks, or compromises on cyber resources
Cyber Resources	(Cyber Resiliency) Separately manageable resources in cyberspace, including information in electronic form, as well as information systems, systems-of-systems, network infrastructures, shared services, and devices. [9] Thus, a cyber resource can be a system element, a service or capability offered by a system element, or information, viewed in terms of how it can be used (e.g., processing, communications, storage, information in usable form).
Cybersecurity	Prevention of damage to, protection of, and restoration of computers, electronic communications systems, electronic communications services, wire communication, and electronic communication, including information contained therein, to ensure its availability, integrity, authentication, confidentiality, and nonrepudiation. [31]
Design Principle	A succinct statement distilling experience designing, implementing, integrating, and upgrading systems that systems engineers and architects can use to guide design decisions and analysis.
Disruption	An event or set of circumstances that disrupts normal operations
Evolve	(Cyber Resiliency Goal) Adapt mission or business functions and/or supporting capabilities to predicted changes in the technical, operational, or threat environments [3] [1]
Partition	(verb) Separate sets of system elements into effectively separate systems, with controlled interfaces between them (noun) A set of system elements, components, or sub-components functioning as a separately managed system
Prepare	(Cyber Resiliency Objective) Maintain a set of realistic courses of action that address predicted or anticipated adversity [3] [1]
Prevent / Avoid	(Cyber Resiliency Objective) Preclude the successful execution of an attack or the realization of adverse conditions [3] [1]
Privacy	(with respect to personally identifiable information) Protection of individual autonomy by restricting the collection, modification, use, dissemination, and retention of personally identifiable information (PII)
Re-architect	(Cyber Resiliency Objective) Modify architectures to handle adversity more effectively [3] [1]
Reconstitute	(Cyber Resiliency Objective) Restore as much mission or business functionality as possible subsequent to adversity [3] [1]
Recover	(Cyber Resiliency Goal) Restore mission or business functions during and after adversity [3] [1]

Term	Definition
Resource	(1) An asset that is utilized or consumed during the execution of a process [32] (2) (Cyber Resiliency) See *cyber resources*.
Security	(1) Freedom from those conditions that can cause loss of assets with unacceptable consequences. [1] (2) A condition that results from the establishment and maintenance of protective measures that enable an enterprise to perform its mission or critical functions despite risks posed by threats to its use of information systems. Protective measures may involve a combination of deterrence, avoidance, prevention, detection, recovery, and correction that should form part of the enterprise's risk management approach. [31] (3) (CPS, concern as part of trustworthiness aspect) Concerns related to the ability of the CPS to ensure that all of its processes, mechanisms, both physical and cyber, and services are afforded internal or external protection from unintended and unauthorized access, change, damage, destruction, or use. Confidentiality: Preserving authorized restrictions on access and disclosure. Integrity: Guarding against improper modification or destruction of system, and includes ensuring non-repudiation and authenticity and use of a system. Availability: Ensuring timely and reliable access to and use of a system. ([34], p. 31)
Strategic Design Principle	A design principle intended to be applied throughout the systems engineering process, guiding the direction of engineering analyses. (derived from [15])
Structural Design Principle	A design principle which directly affects the architecture and design. (derived from [15]) A structural design principle can be applied to selected locations in the architecture or design.
Survivability	The ability of a system to minimize the impact of a finite-duration disturbance on value delivery (i.e., stakeholder benefit at cost), achieved through (1) the reduction of the likelihood or magnitude of a disturbance, (2) the satisfaction of a minimally acceptable level of value delivery during and after a disturbance, and/or (3) a timely recovery. [117]
System	Combination of interacting elements organized to achieve one or more stated purposes [1] [123]
System Element	Member of a set of elements that constitute a system. *Note 1:* A system element can be a discrete component, product, service, subsystem, system, infrastructure, or enterprise. *Note 2:* Each element of the system is implemented to fulfill specified requirements. *Note 3:* The recursive nature of the term allows the term *system* to apply equally when referring to a discrete component or to a large, complex, geographically distributed system-of-systems. *Note 4:* System elements are implemented by: hardware, software, and firmware that perform operations on data/information; physical structures, devices, and components in the environment of operation; and the people, processes, and procedures for operating, sustaining, and supporting the system elements. [1] [123]
System Resiliency	The ability to anticipate, withstand, recover from, and adapt to adverse conditions, stresses, attacks, or compromises on system resources [1]
Tactics, Techniques, and Procedures	The use of capabilities and resources in relation to each other (tactics); non-prescriptive ways or methods used to perform missions, functions, or tasks (techniques); and standard, detailed steps that prescribe how to perform specific tasks (procedures) [124], adapted
Traditional	(As applied to computer security) Concerned with limiting the physical access to corporate systems and the misappropriation or vandalism of data by internal users [125]
Transform	(Cyber Resiliency Objective) Modify mission / business functions and supporting processes to handle adversity more effectively [3] [1]

Term	Definition
Trustworthiness	(1) Worthy of being trusted to fulfill whatever critical requirements may be needed for a particular component, subsystem, system, network, application, mission, enterprise, or other entity. *Note*: From a security perspective, a trustworthy system is a system that meets specific security requirements in addition to meeting other critical requirements. [1] (2) The attribute of a person or enterprise that provides confidence to others of the qualifications, capabilities, and reliability of that entity to perform specific tasks and fulfill assigned responsibilities. [31]
Understand	(Cyber Resiliency Objective) Maintain useful representations of mission and business dependencies and the status of resources with respect to possible adversity [3] [1]
Voice of the Adversary	A design analysis technique in which one or more team members play the role of an adversary to critique alternatives by taking into consideration possible goals, behaviors, and cyber effects assuming varying degrees of system access or penetration
Withstand	(Cyber Resiliency Goal) Continue essential mission or business functions despite adversity [3] [1]

D.2 List of Abbreviations

AFRL	Air Force Research Laboratory
ALC	Acquisition Lifecycle
APT	Advanced Persistent Threat
ATT&CK	Adversarial Tactics, Techniques & Common Knowledge
BIA	Business Impact Analysis
CAL	Cyber Attack Lifecycle
CAPEC	Common Attack Pattern Enumeration and Classification
CDS	Cross Domain Solution
CKC	Cyber Kill Chain
CERT	Computer Emergency Response Team (SEI)
	Computer Emergency Readiness Team (US-CERT at DHS)
CESG	Communications-Electronics Security Group
CJA	Crown Jewels Analysis
CMIA	Cyber Mission Impact Analysis
CONOPS	Concept of Operations
COOP	Continuity of Operations
COTS	Commercial off-the-shelf
CPS	Cyber-Physical System
CPS PWG	CPS Public Working Group
CRA	Cyber Resiliency Analysis
CREF	Cyber Resiliency Engineering Framework
CRR	Cyber Resiliency Review
CSG	Cyber Security Game
DAR	Data-at-Rest
DDoS	Distributed Denial of Service
DHS	Department of Homeland Security
DLP	Data Loss Prevention
DNS	Domain Name Service (or Server)
DoD	Department of Defense

EA	Enterprise Architecture
FDNA	Functional Dependency Network Analysis
FOSS	Free and Open Source Software
FRD	Functional Requirements Document
GSA	General Services Administration
H&S	Health and Status
HACS	Highly Adaptive Cybersecurity Services
HITL	Human-In-The-Loop
HOTL	Human-On-The-Loop
IACD	Integrated Adaptive Cyber Defense
ICT	Information and Communications Technology
IdAM	Identity and Access Management
IDS	Intrusion Detection System
INCOSE	International Council on Systems Engineering
IOC	Initial Operational Capability
IoT	Internet of Things
IT	Information Technology
JCIDS	Joint Capabilities Integration and Development System
KPP	Key Performance Parameter
KSA	Key System Attribute
M&S	Modeling and Simulation
MCA	Malicious Cyber Activities
MIA	Mission Impact Analysis
MITM	Man-in-the-Middle
MS	Milestone
MTA	Mission Thread Analysis
MTD	Moving Target Defense
NIST	National Institute of Standards and Technology
O&M	Operations & Maintenance
O/O	Owner / Operator
OPSEC	Operations Security
OS	Operating System
OWASP	Open Web Application Security Project
PII	Personally Identifiable Information
PIT	Platform Information Technology
POET	Political, Operational, Economic, and Technical
PoP	Philosophy of Protection
RMA	Reliability, Maintainability, and Availability
RMM	(CERT) Resilience Management Model
SCADA	Supervisory Control and Data Acquisition
SCRAM	Structured Cyber Resiliency Analysis Methodology
SCRM	Supply Chain Risk Management
SDP	System Design Process
SDLC	System Development Lifecycle
SDN	Software-Defined Networking
SE	Systems Engineering
SEBoK	(INCOSE) Systems Engineering Body of Knowledge
SEI	Software Engineering Institute at Carnegie-Mellon University
SIMEX	Simulation Experiment

SOC	Security Operations Center
SoS	System-of-Systems
SOW	Statement of Work
SP	Special Publication
SS	Systems Survivability
SSE	Systems Security Engineering
TCSEC	Trusted Computer System Evaluation Criteria
TTPs	Tactics, Techniques, and Procedures
TTX	Tabletop Exercise
VM	Virtual Machine
VMM	Virtual Machine Monitor
VoA	Voice of the Adversary
VPN	Virtual Private Network