A Lifecycle Modeling Language (LML) Approach to Zero Trust Architecture

Andy Tapia

11 October 2024

Agenda

- Introduction to ZTA
- LML & Innoslate Overview
- DoD ZTA Reference Architecture
- LML for Requirements Analysis
- LML for Systems Modeling & Design
- LML for V&V Testing
- LML for Implementation
- Closing Thoughts

Introduction to Zero Trust Architecture

- Cybersecurity Framework
  - “Never Trust, Always Verify”
  - Assumes that threats can be both internal & external
- Core Ideas
  - Least Privilege Access
    - Users should only have what they need
  - Micro-Segmentation
    - Isolate and guard sensitive data
  - Continuous Monitoring
    - Ongoing risk assessments of user/device behaviors

Rise of Zero Trust

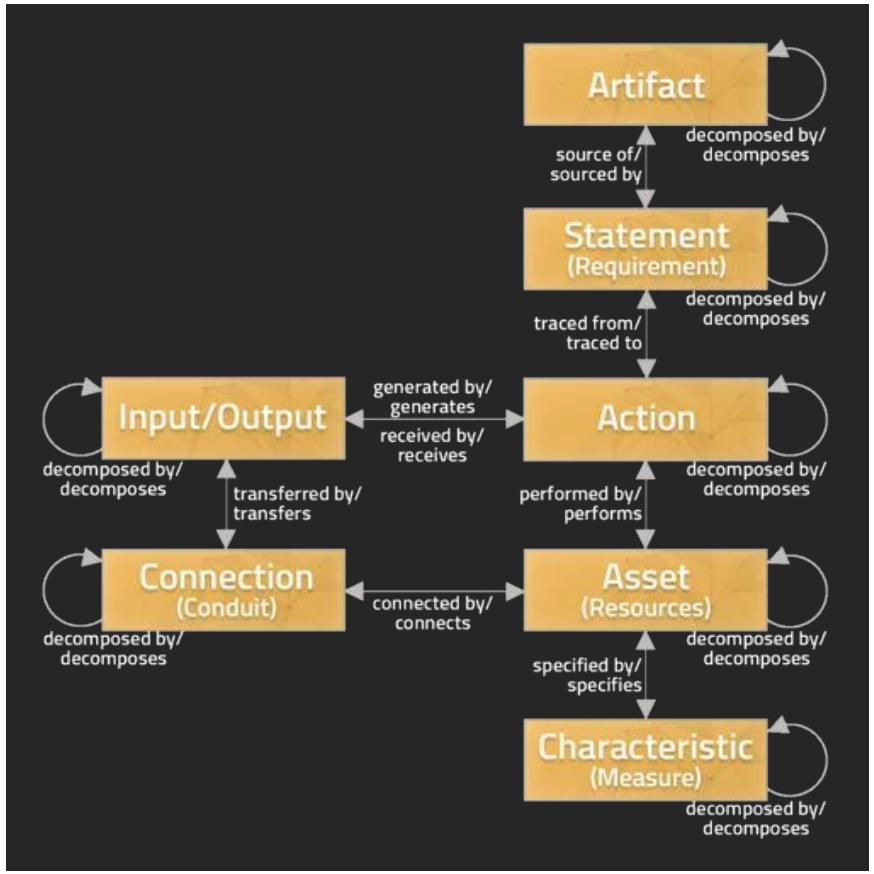
- ZTA is rising in interest due to internal threats & persistent nation-state actors
  - Breaches no longer a case of “if” but “when”
  - Breaches are increasing in number and cost of damages are on the rise
  - Need to minimize damages and amount of data leaked
- Data breach victims surpassed 1 billion in first half of 2024
  - 409% increase from the same time period last year
- ZTA offers a modern cybersecurity defense solution
  - Enhanced security posture against persistent threats
  - Reduced risk of data breaches and insider threats
  - Improved compliance with security regulations and standards
  - Enables new concepts such as resilience

Lifecycle Modeling Language (LML) Overview

- LML is an open-standard modeling language designed by the Lifecycle Modeling Organization
- Based on the entity, relationship, and attribute (ERA) meta-meta model modified by adding attributes to relationships
  - Includes 12 entity classes and 8 subclasses that can be connected bi-directionally
- Simplified, concise, & extendable ontology
  - Designed for the full SE lifecycle process
  - Enables traceability from requirements to implementation
  - Supports project management
  - Accessible to non-SE stakeholders

LML Entities

- Action
- Artifact
- Asset
  - Resource
- Characteristic
  - Measure
- Connection
  - Conduit
  - Logical
- Risk
- Time
- Cost
- Decision
- Input/Output
- Location
  - Orbital
  - Physical
  - Virtual
- Statement
  - Requirement



LML Specification Relationships

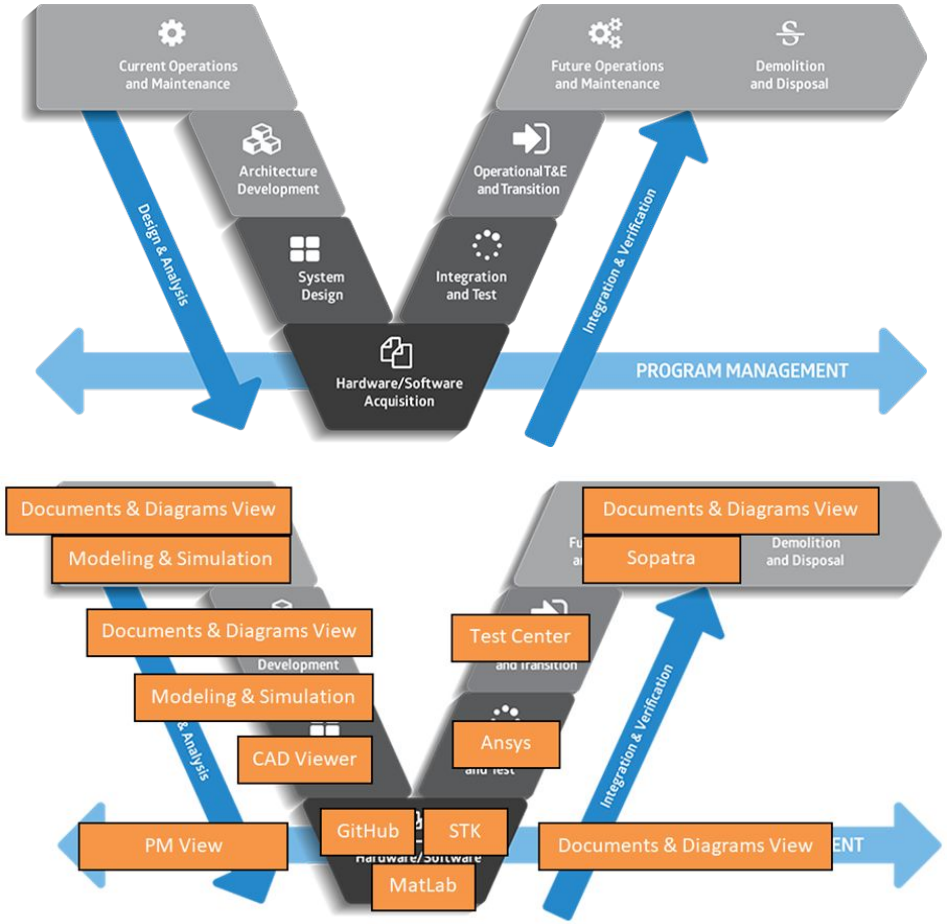
LML Relationships

	Action	Artifact	Asset (Resource)	Characteristic (Measure)	Connection (Conduit, Logical)	Cost	Decision	Input/Output	Location (Orbital, Physical, Virtual)	Risk	Statement (Requirement)	Time
Action	decomposed by* related to*	references	(consumes) performed by (produces) (seizes)	specified by	-	incurs	enables results in	generates receives	located at	causes mitigates resolves	(satisfies) traced from (verifies)	occurs
Artifact	referenced by	decomposed by* related to*	referenced by	referenced by specified by	defines protocol for referenced by	incurs referenced by	enables referenced by results in	referenced by	located at	causes mitigates referenced by resolves	referenced by (satisfies) source of traced from (verifies)	occurs
Asset (Resource)	(consumed by) performs (produced by) (seized by)	references	decomposed by* orbited by* related to*	specified by	connected by	incurs	enables made responds to results in	-	located at	causes mitigates resolves	(satisfies) traced from (verifies)	occurs
Characteristic (Measure)	specifies	references specifies	specifies	decomposed by* related to* specified by*	specifies	incurs specifies	enables results in specifies	specifies	located at specifies	causes mitigates resolves specifies	(satisfies) spacifies traced from (verifies)	occurs specifies
Connection (Conduit, Logical)	-	defined protocol by references	connects to	specified by	decomposed by* joined by* related to*	incurs	enables results in	transfers	located at	causes mitigates resolves	(satisfies) traced from (verifies)	occurs
Cost	incurred by	incurred by references	incurred by	incurred by specified by	incurred by	decomposed by* related to*	enables incurred by results in	incurred by	located at	causes incurred by mitigates resolves	incurred by (satisfies) traced from (verifies)	occurs
Decision	enabled by result of	enabled by references result of	enabled by made by responded by result of	enabled by result of specified by	enabled by result of	enabled by incurs result of	decomposed by* related to*	enabled by result of	located at	causes enabled by mitigated by result of resolves	alternative enabled by traced from result of	date resolved by decision due occurs
Input/Output	generated by received by	references	-	specified by	transferred by	incurs	enables results in	decomposed by* related to*	located at	causes mitigates resolves	(satisfies) traced from (verifies)	occurs
Location (Orbital, Physical, Logical)	locates	locates	locates	locates specified by	locates	locates	locates	locates	decomposed by* related to*	locates mitigates	locates (satisfies) traced from (verifies)	occurs
Risk	caused by mitigated by resolved by	caused by mitigated by references resolved by	caused by mitigated by resolved by	caused by mitigated by resolved by specified by	caused by mitigated by resolved by	caused by incurs mitigated by resolved by	caused by enables mitigated by results in resolved by	caused by mitigated by resolved by	located at mitigated by	caused by* decomposed by* related to* resolved by*	caused by mitigated by resolved by	occurs mitigated by
Statement (Requirement)	(satisfied by) traced to (verified by)	references (satisified by) sourced by traced to (verified by)	(satisified by) traced to (verified by)	(satisified by) specified by traced to (verified by)	(satisified by) traced to (verified by)	incurs (satisified by) traced to (verified by)	alternative of enables traced to results in	(satisified by) traced to (verified by)	located at (satisfied by) traced to (verified by)	causes mitigates resolves	decomposed by* traced to* related to*	occurs (satisified by) (verified by)
Time	occurred by	occurred by	occurred by	occurred by specified by	occurred by	occurred by	date resolves decided by occurred by	occurred by	occurred by	occurred by mitigates	occurred by (satisfies) (verifies)	decomposed by* related to*

LML Specification Relationship Matrix

Overview of Innoslate

- Developed by SPEC Innovations, based on LML
- Cloud-based MBSE tool that supports the SE lifecycle process
- Features
  - Full lifecycle traceability
  - Modeling capabilities
  - Simulation & analysis
  - Verification & validation testing
  - Built-in project management tools



Innoslate Features & V-Model Mapping

DoD Zero Trust Reference Architecture

- DoD Zero Trust Reference Architecture implemented in Innoslate
  - Stored within a secure, cloud-based, and centralized database
  - Text artifacts parsed within Innoslate
  - Diagrams replicated to show the same or similar information
  - Traceability included where possible

Documents View: Parsed ZTA RA Sections

MENU Dashboard Database Diagrams Documents Charts Test Center

SPEC Innovations / Zero Trust Architecture Sandbox

Manage Documents Dashboard

Save Dashboard Layout | Add Widget | Create Document

All Existing Documents

Showing All Documents | Sorted by Number | order:number class:"Artifact" label:"Concep... | Find a Document...

- Concept of Operations Document — Section 1: Zero Trust Purpose and Strategic Goals
- Concept of Operations Document — Section 10: References (Concept of Operations Document)
- Concept of Operations Document — Section 9: Appendix (AV-2) (Concept of Operations Document)
- Concept of Operations Document — Section 8: Zero Trust Transition Architecture Planning (FFP)
- Concept of Operations Document — Section 7: Zero Trust Architecture Patterns
- Concept of Operations Document — Section 6: Zero Trust Security Assessment
- Concept of Operations Document — Section 5: Zero Trust Technical Positions
- Concept of Operations Document — Section 4: Zero Trust Use Cases
- Concept of Operations Document — Section 3: Zero Trust Capabilities
- Concept of Operations Document — Section 2: Zero Trust Pillars and Principles
- Requirements Document — Zero Trust Implementation High-Level Goals
- Concept of Operations Document — NIST Special Publication 800-207: Zero trust (ZT) is the term for an evolving set of cybersecurity paradigms that move

1 - 13 | Tiles per Page 30

Documents View Dashboard

Using LML for Requirements Analysis

- Requirements Document
  - Quality checker ensures heuristics compliance
  - Tags can be added for database queries
  - Change requests & rationales can be made
  - Can be extended to include additional custom attributes
- Requirements taken from DoD ZTA RA
  - CV-1 Vision & Goals

Requirements Document: CV-1 Vision & Goals

MENU | Dashboard | Database | Diagrams | Documents | Charts | Test Center

Filter | Baselines | Comments

All Document Entities 10
Only Requirements 7
Only Statements 3

Labels (AND)

Acronym	0
Analysis	0
Assumption	0
Constraint	0
Definition	0
Demonstration	0
Environmental Requirement	0
Functional Requirement	0
Goal	0
Inspection	0
Interface Requirement	0
Modeling & Simulation	0
Objective	0
Performance Requirement	0
Purpose	0
Reference	0
Reliability Requirement	0
Safety Requirement	0
Scope	0

Back | New State... | Open | More | Reports | Search the Document...

Entity	Rationale	Quality Score	Labels
1 Vision Next generation cybersecurity architecture that precludes default trust of any actor, system, network, or service operating outside or within the security perimeter using a data-centric approach to establish continual verification of each user, device, application, and transaction.	N/A	N/A	*No labels to display.*
2 Goal Secure and defend DoD information, systems, and critical infrastructure against malicious cyber activity, including DoD information on non-DoD-owned networks using Zero Trust.	N/A	N/A	*No labels to display.*
2.1 Objective 1 Detect, deter, deny, defend, and recover from malicious cyber activity across all operational environments.		33%	*No labels to display.*
2.2 Objective 2 Develop a scalable, resilient, auditable, and defendable framework centered on the protection of DoD's most critical, mission-essential data, applications, assets, and services (DAAS).		22%	*No labels to display.*
3 Strategic Requirements	N/A	N/A	*No labels to display.*
3.1 Application of existing and emerging cyber technologies to systematically improve enterprise network defenses predicated on foundational Zero Trust concepts.		44%	*No labels to display.*
3.2 Elimination of the concept of trusted networks, devices, personas, or processes.		44%	*No labels to display.*
3.3 Moving security away from the legacy "castle and moat" approach which focuses on a strong network perimeter.		33%	*No labels to display.*
3.4 Implementation of security in a more consistent and efficient manner.		44%	*No labels to display.*
3.5 Positioning authentication and security mechanisms throughout the architecture to monitor,		22%	*No labels to display.*

Hierarchy

Vision and Goals Document

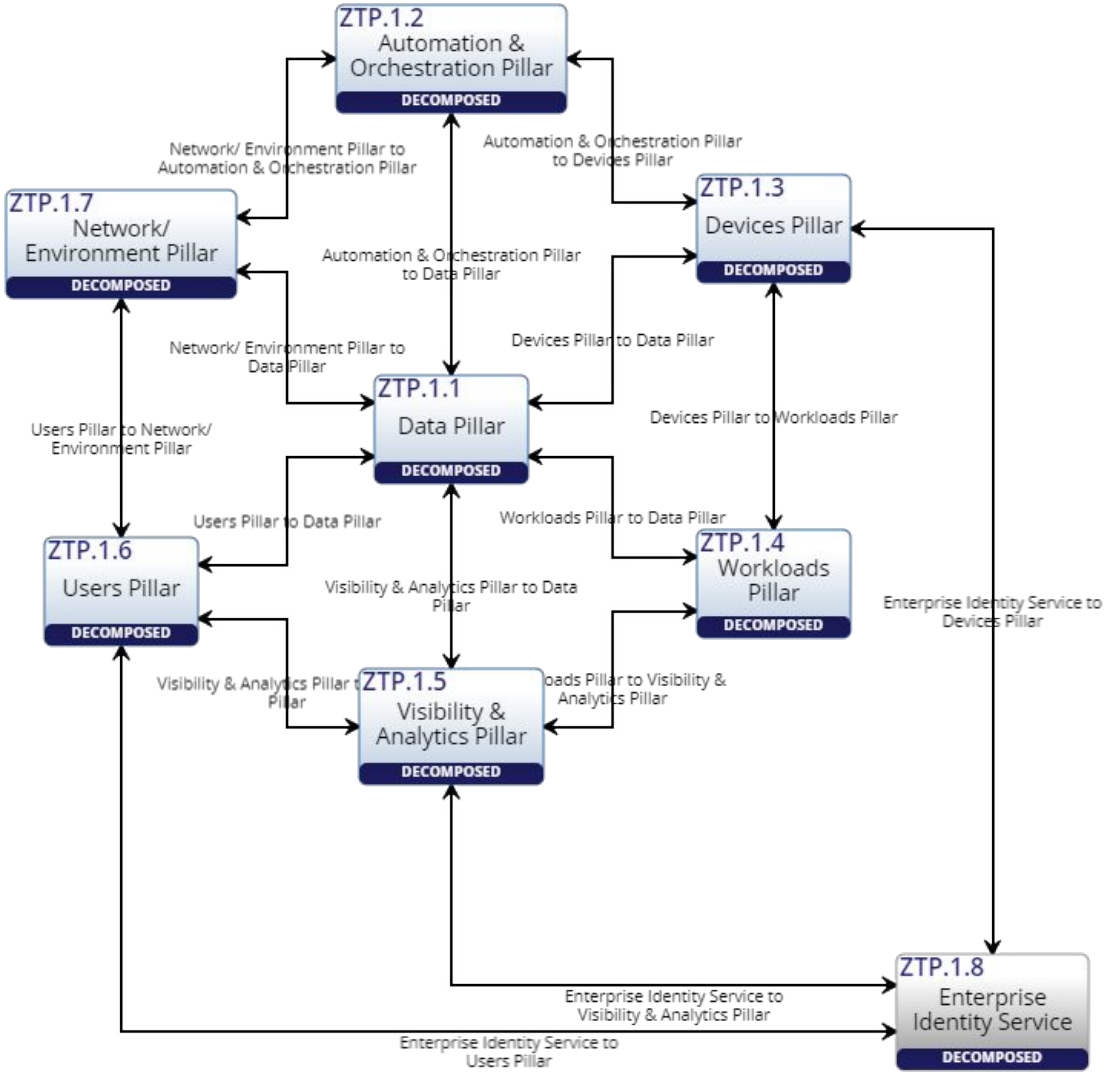
- 1 Vision
- 2 Goal
  - 2.1 Objective 1
  - 2.2 Objective 2
- 3 Strategic Requirements
  - 3.1
  - 3.2
  - 3.3
  - 3.4
  - 3.5

Requirements Document View

Using LML for Systems Modeling & Design

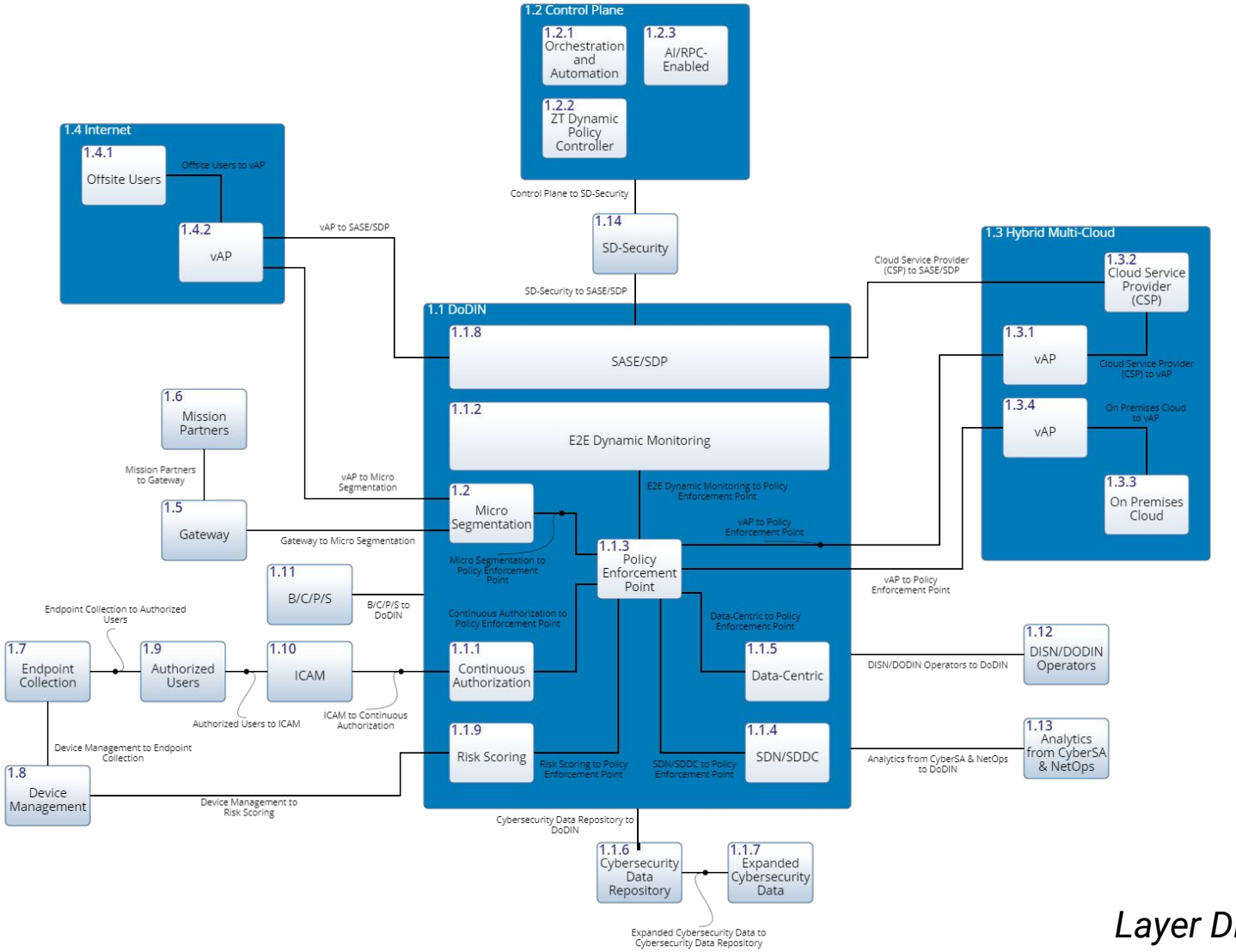
- Asset Diagram & Layer Diagram
  - Develop system context
  - Design physical & functional architectures
- Action Diagram
  - Design system functionality & capabilities
- System architecture & capability products from DoD ZTA RA
  - ZT Pillars
  - As-Is & To-Be OV-1s
  - ZT System Capabilities

Asset Diagrams: ZT Pillars



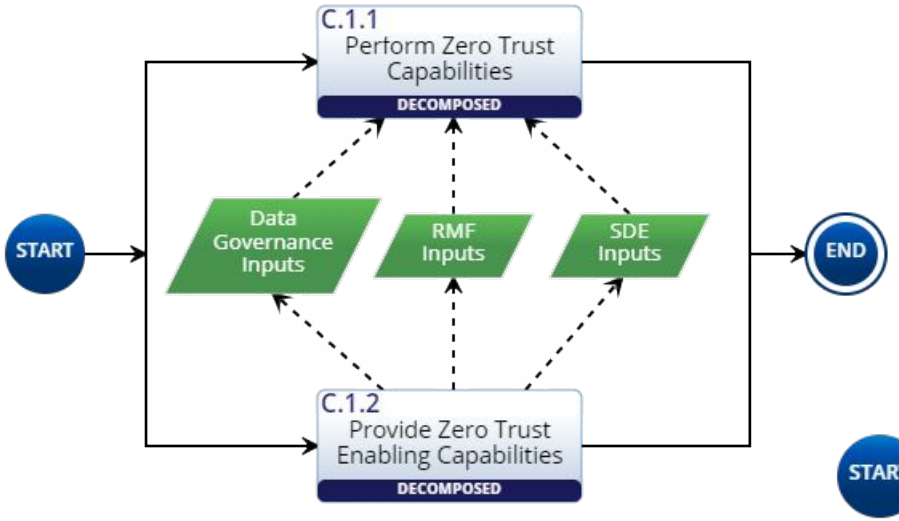
Asset Diagram View

Layer Diagram: “To-Be”Overall Target Environment (OV-1)

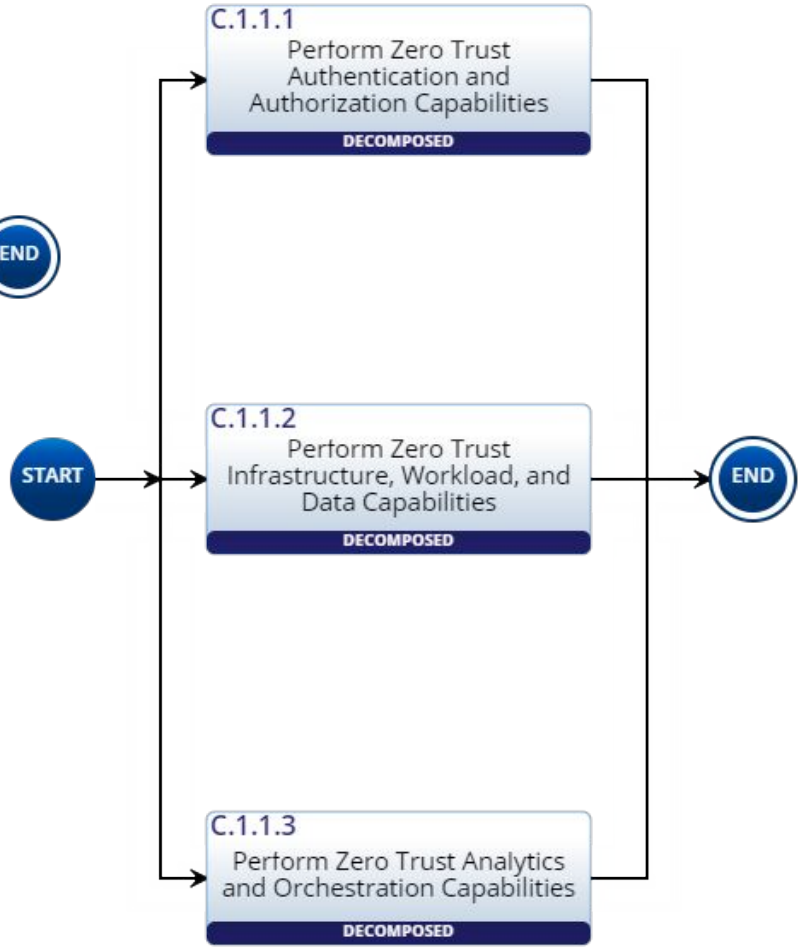


Layer Diagram View

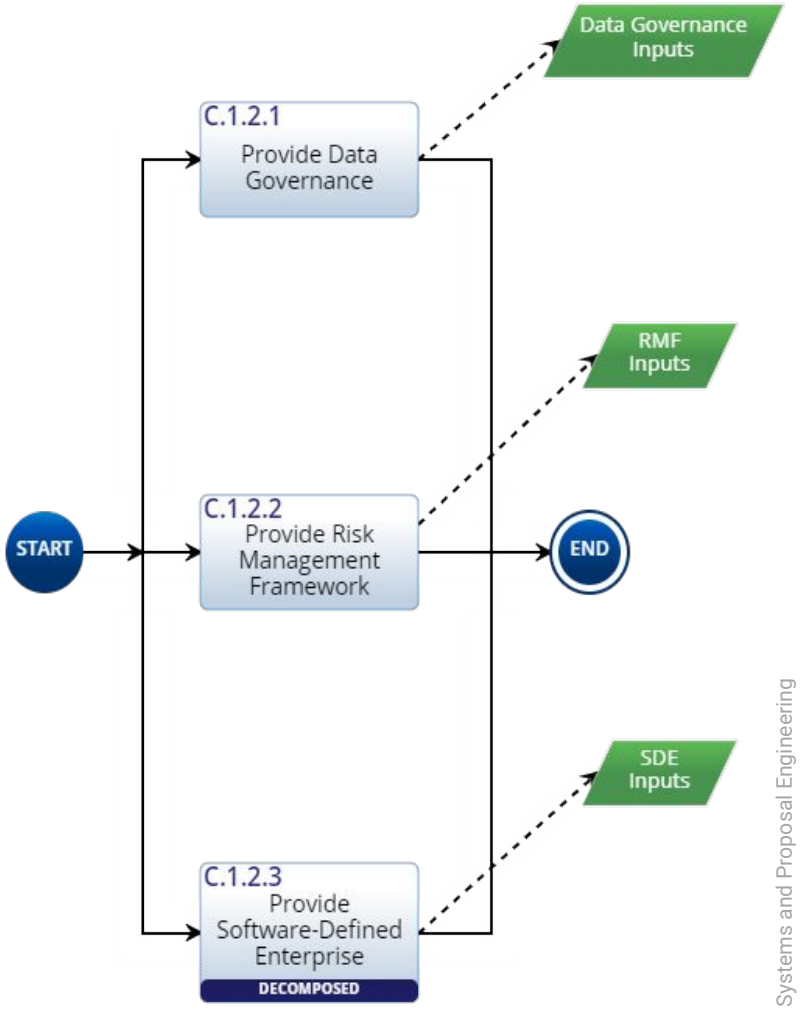
Action Diagram: ZT System Capabilities



Action Diagram View



Action Diagram Decomposed View (C.1.1)



Action Diagram Decomposed View (C.1.2)

Using LML for V&V Testing

- Test Center
  - Can hold V&V requirements with roll-up visuals
  - Also can hold general testing for metrics
  - Completes traceability when linked to requirements
- Risk Diagram & Risk Burn-Down Chart
  - Record risks and how mitigations impact them
- None from DoD ZTA RA

Test Center: V&V Testing

MENU | Dashboard | Database | Diagrams | Documents | Charts | Test Center

Filter | Test Cycles | Comments

All Test Suite Entities 40

Only Test Cases 40

Labels (AND)

Verification Test Case 33

DoDAF Product 0

Modified Model 0

STK/MATLAB Integration Model 0

Back | Open | Reports | Search the Test Suite...

Entity	Expected Result	Actual Result	Status	Status Roll-Up
VRT.1 Suspension Testing dual suspension of rover.			Passed	3
VRT.1.1 Dual-Suspension #1 1. Place rover on level terrain, 0 degree incline 2. Add item of clearance height of 3 cm on either side of rover 3. Confirm rover dual-suspension ability	Rover clears over 3 cm item	Successfully cleared	Passed	Passed
VRT.1.2 Dual- Suspension #2 1. Place rover on level terrain, 0 degrees incline 2. Add item of clearance height of 6 cm on either side of rover 3. Confirm rover dual-suspension ability	Rover clears over 6 cm item	Successfully cleared	Passed	Passed
VRT.1.3 Dual- Suspension #3 1. Place rover on level terrain, 0 degrees incline 2. Add item of clearance height of 8 cm on either side of rover 3. Confirm rover dual-suspension ability	Rover clears over 8 cm item	Successfully cleared	Passed	Passed
VRT.2 Rover Velocity Testing rover average velocity with or without materials load and with and without inclines.			Failed	2 4
VRT.2.1 Average Velocity w… 1. Create 10 meter straight away course 2. Set rover on flat surface 3. Confirm no materials are loaded in storage unit 4. Record the rover's time to drive 10 meter straight course	Average velocity of 0.35 +/- 0.05 m/s	Confirmed consistent average velocity of 0.3432 m/s	Passed	Passed

Hierarchy

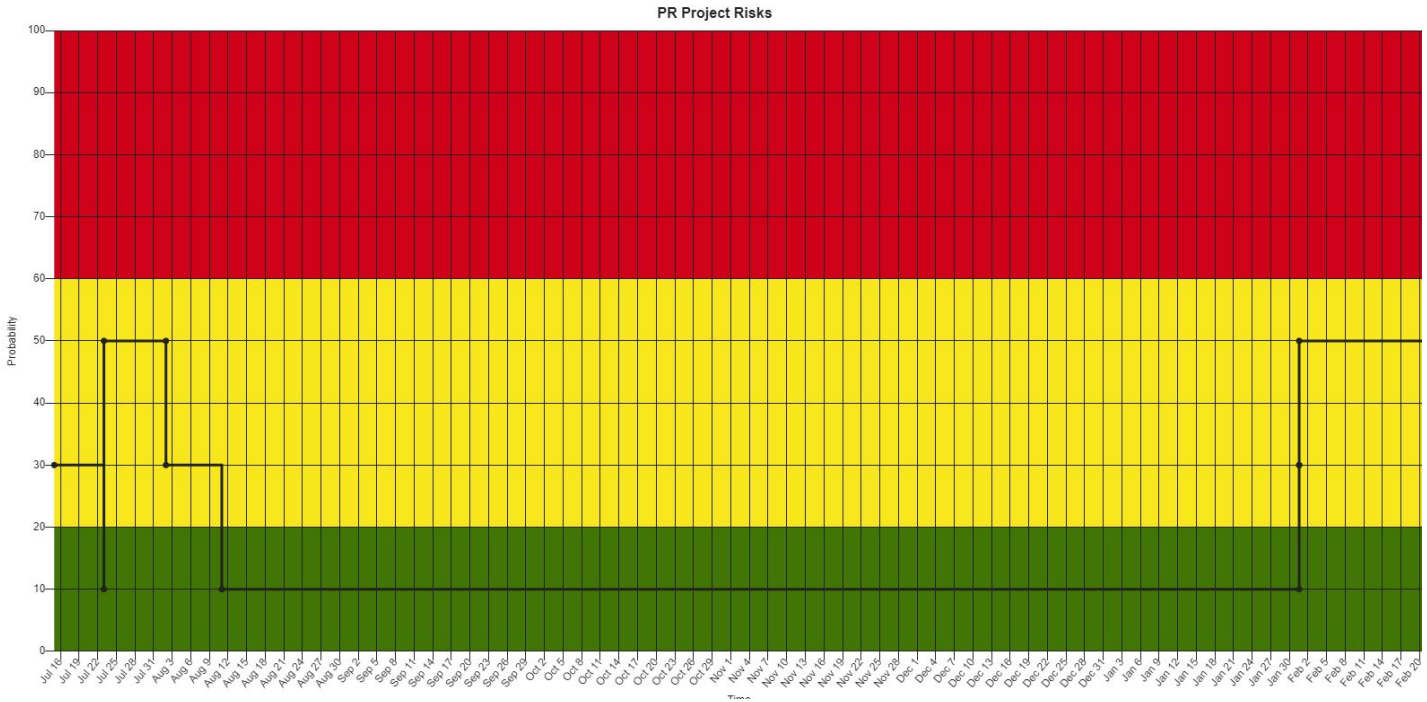
SPECTER Prototype Testing T…

- VRT.1 Suspension
  - VRT.1.1 Dual-Suspension #1
  - VRT.1.2 Dual- Suspension #2
  - VRT.1.3 Dual- Suspension #3
- VRT.2 Rover Velocity
  - VRT.2.1 Average Velocity w/o M…
  - VRT.2.2 Average Velocity w/ Ma…
  - VRT.2.3 Average Velocity on De…
  - VRT.2.4 Average Velocity on De…
  - VRT.2.5 Average Velocity on Inc…
  - VRT.2.6 Average Velocity on Inc…
- VRT.3 Rover Operations
  - VRT.3.1 Time to Go Up an Inclin…
  - VRT.3.2 Time to Go Up an Inclin…
  - VRT.3.3 Time to Go Down a De…
  - VRT.3.4 Time to Go Down a De…
  - VRT.3.5 Average Runtime
  - VRT.3.6 Charge Time
  - VRT.3.7 Detect Obstacles
  - VRT.3.8 Visuals in the Dark
- VRT.4 Storage Capacity
  - VRT.4.1 Maximum Storage Cap…
  - VRT.4.2 Moving Storage Capac…
  - VRT.4.3 Leakage Test
- VRT.5 Rover Excavation

Test Center View: Lunar Rover Prototype Testing

Risk Diagram & Risk-Burn Down: Risk Register & Mitigation

Risk Diagram View: Lunar Rover Project Risks

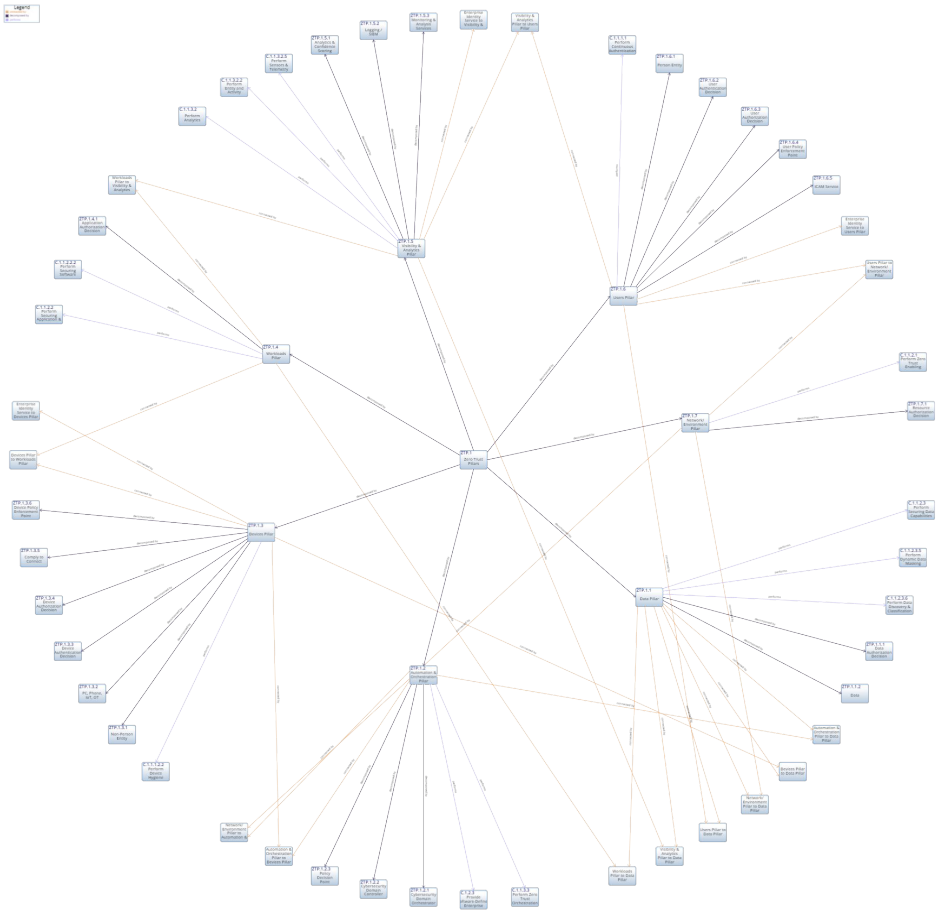


Risk Burn-Down View: Lunar Rover Project Risks

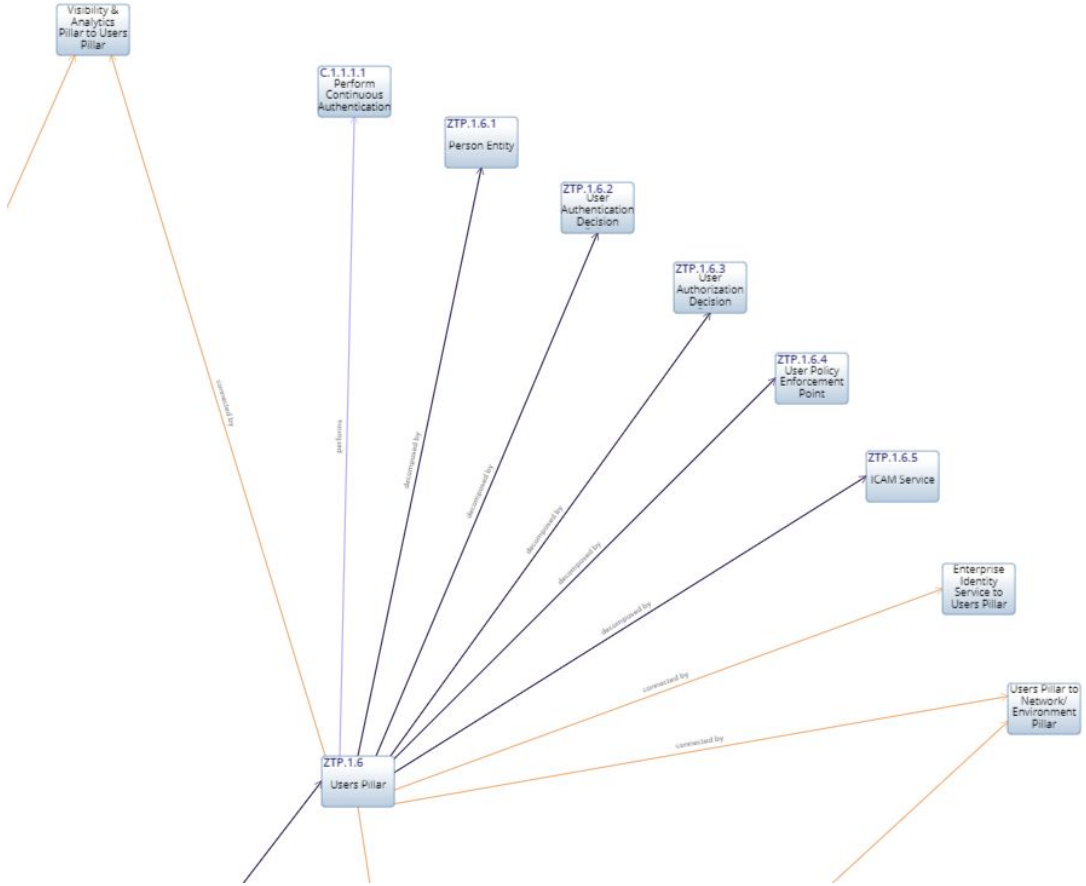
Using LML for Implementation

- Spider Diagram
  - Visualize relationships of all types for a specific entity
  - Trace entities all the way back to requirements or even decisions
  - Supports notion of digital thread
- Zero Trust Pillars from DoD ZTA RA
  - Visualize all relationships
    - Parents/children
    - Interfaces
    - Performing assets

Spider Diagrams: ZT Pillars



Spider Diagram View



Spider Diagram View (Zoomed In)

Closing Thoughts

- Zero Trust Principles & Concepts
  - Becoming more important and perhaps will be necessary for critical industries
- LML
  - A viable alternative that can be used effectively by systems engineers
  - More accessible and understandable by non-SE stakeholders
  - Flexible and adaptable to new changes in technology
  - Encouraging to see it being used in more applications

Thank You

References

- https://www.usatoday.com/story/money/2024/07/18/data-breach-what-to-do/74441060007/
- https://www.lifecyclemodeling.org/
- https://dodcio.defense.gov/Portals/0/Documents/Library/(U)ZT_RA_v2.0(U)_Sep22.pdf