ZT Principles Applied to existing applications – DRAFT 0.5

ZT Working Group

03/07/2022

- Proceeding to evaluate OSS coverage of NCCoE Functional ZT Compositions
- All efforts will be confined to the OASIS OCA ZT Working Group, until we are ready to establish an official OASIS OCA Project/Sub-Project.

Current NCCoE “Crawl Phase” Functional ZT Element Composition

- Category 1
  - Discovery, Classification and Compliance of assets -
- Category 2
  - Mobile Platform management –
  - Mobile Endpoint Security –
  - Mobile Endpoint (Emergent) Threat Detection & Response
- Category 3
  - Enterprise Endpoint Management –
  - Enterprise Endpoint Security
  - Emergent Endpoint (Emergent) Threat Detection & Response - EDR/NDR/XDR/MDR
- Category 4
  - Enterprise Infrastructure Management, Integration and Automation – Directories, Provisioning, Orchestration
- Category 5
  - Security Services Edge/WAN Edge Services – Gateway PEP (ZT 0.1 policy)

Objective 1 (2 weeks): OSS Coverage of Category 1

- Category 1 - Discover, Classification and Compliance of Assets
  - Nessus - https://www.tenable.com/downloads/nessus?loginAttempted=true
  - Nmap - https://nmap.org/

  - OpenVAS – https://cdn-cybersecurity.att.com/docs/whitepapers/AV-OpenSourceNetworkSecurity.pdf (Fork of Nessus)
  - Nikto - https://www.kali.org/tools/nikto/
  - OpenSCAP - https://www.open-scap.org/tools/ (FedRAMP, FISMA, DISA …)
  - Open-Audit - https://opmantek.com/network-discovery-inventory-software/
  - Mitre Vulcan - https://vulcan.mitre.org/ (STIGs)
  - OSQuery - https://www.osquery.io/

  - Open FISMA+ - http://openfisma.org/

- Q1: Any additions to this list?
- Q2: Any deletions from this list?

OASIS OCA ZT Working Group Status Update

1. Available OSS tech
   - Surveying open source security tools/services that can cover the ZT technology profiles of current NCCoE proposed implementations (8 current proposals). See last update for cross vendor example of minimally adequate proposal.
   - Could use suggestions of comparable OSS security portfolio options, especially in EDR (0-day threat hunting, analysis, mitigation …)
   - Probably 2-4 weeks
2. Adequate capability & controls:
   - Using NCCoE mapped controls, determine functional adequacy of available OS tech.
   - See next slide from: https://www.nccoe.nist.gov/sites/default/files/library/project-descriptions/zta-project-description-final.pdf
   - <depends on 1 above>
   - About 2 weeks, if I can get the current docs (may be tough for less active efforts with sparse design, integration , planning documentation). Code is not nearly enough.
3. Adequate integrability: OSS portfolio must have enough context in identifiers and predicates (tool specific policy) to allow semantic alignment on subjects, objects and their relationship/dependency.
   - Easier for cloud friendly tech (for k8s, SM, functions … across dynamic workloads)
   - Far less so for tools aimed only at conventional application security scenarios. See NIST NCCoE evaluation architecture from last update (below)
   - Definitionally, ZT leverages micro-segmentation, software defined-ness and /or proxies, to realize correct logical placement of PEP between subject and object , across normal system dynamics. ZT is intentionally anti-VPN/VLAN/static NW architecture/policy and anti-"implicit trust" (see SP 800-207)
   - About 4 weeks for mapping into analogous evaluative design.
   - <depends on 2>
4. Will be pursuing collaboration discussions with NCCoE after ZT Kickoff meeting. Strict dependency on NIST, but 1-3 above do not depend on such collaboration, and will be pursued independently, since it leveraged only publicly available NCCoE documentation.
5. Minimal result from effort:
   - A concrete and NIST conformant proposed OSS ZT implementation, that is one to one comparable with current proposed implementations functionally.
   - A concrete controls-driven evaluation plan that is fully comparable to the comparative/evaluative results NIST will generate (put possibly not socialize).
   - Caveat: we will not have access to the commercial verification tech NIST is using in NCCoE, but may be able to use either the Mitre simulator or Cobalt Strike, or … (will resolve ASAP)
6. Question: Does this sound reasonable?
7. Question: Will we do an actual evaluation/validation? Significant methodological work. –or- do we recommend an OSS proposed implementation for NIST to evaluate , just as they are doing for commercial alternatives? (much better option, in my view)

NCCoE: NIST CSF for ZT - Identify, Protect , Detect, Respond

What's "Good Enough" ZT? Pages 11-14 of https://www.nccoe.nist.gov/sites/default/files/library/project-descriptions/zta-project-description-final.pdf

Cybersecurity Framework v1.1		
Function	**Category**	**Subcategory**
IDENTIFY (ID)	Asset Management (ID.AM)	ID.AM-1: Physical devices and systems within the organization are inventoried.
		ID.AM-2: Software platforms and applications within the organization are inventoried.
		ID.AM-5: Resources (e.g., hardware, devices, data, time, personnel, and software) are prioritized based on their classification, criticality, and business value.
	Risk Assessment (ID.RA)	ID.RA-1: Asset vulnerabilities are identified and documented.
		ID.RA-3: Threats, both internal and external, are identified and documented.
PROTECT (PR)	Identity Management, Authentication, and Access Control (PR.AC)	PR.AC-1 Identities and credentials are issued, managed, verified, revoked, and audited for authorized devices, users, and processes.
		PR.AC-3 Remote access is managed.
		PR.AC-4 Access permissions and authorizations are managed, incorporating the principles of least privilege and separation of duties.
		PR.AC-5: Network integrity is protected (e.g., network segregation, network segmentation).
		PR.AC-6: Identities are proofed and bound to credentials and asserted in interactions.

+SBOM

Cybersecurity Framework v1.1		
Function	**Category**	**Subcategory**
		PR.AC-7 Users, devices, and other assets are authenticated (e.g., single-factor, multifactor) commensurate with the risk of the transaction (e.g., individuals' security and privacy risks and other organizational risks).
	Data Security (PR.DS)	PR.DS-2 Data in transit is protected.
		PR.DS-5: Protections against data leaks are implemented.
		PR.DS-6 Integrity-checking mechanisms are used to verify software, firmware, and information integrity.
		PR.DS-8: Integrity-checking mechanisms are used to verify hardware integrity.
	Information Protection Processes and Procedures (PR.IP)	PR.IP-1: A baseline configuration of IT/industrial control systems is created and maintained, incorporating security principles (e.g., concept of least functionality).
		PR.IP-3: Configuration change control processes are in place.

Cybersecurity Framework v1.1		
Function	**Category**	**Subcategory**
DETECT	Anomalies and Events (DE.AE)	DE.AE-2: Detected events are analyzed to understand attack targets and methods.
		DE.AE-3: Event data are collected and correlated from multiple sources and sensors.
		DE.AE-5: Incident alert thresholds are established.
	Security Continuous Monitoring (DE.CM)	DE.CM-1: The network is monitored to detect potential cybersecurity events.
		DE.CM-2: The physical environment is monitored to detect potential cybersecurity events.
		DE.CM-4: Malicious code is detected.
		DE.CM-5: Unauthorized mobile code is detected.
		DE.CM-6: External service provider activity is monitored to detect potential cybersecurity events.
		DE.CM-7 Monitoring for unauthorized personnel, connections, devices, and software is performed.
		DE.CM-8: Vulnerability scans are performed.
	Detection Processes (DE.DP)	DE.DP-5: Detection processes are continuously improved.
RESPOND	Mitigation (RS.MI)	RS.MI-1: Incidents are contained.
		RS.MI-2: Incidents are mitigated.

Multiple telemetry/sensor types collected and correlated (separation of Network, Endpoint, Service … telemetry/analytics is not the intent)

Known Payload Ident Required

Known Vuln Ident Required

Granular Isolation

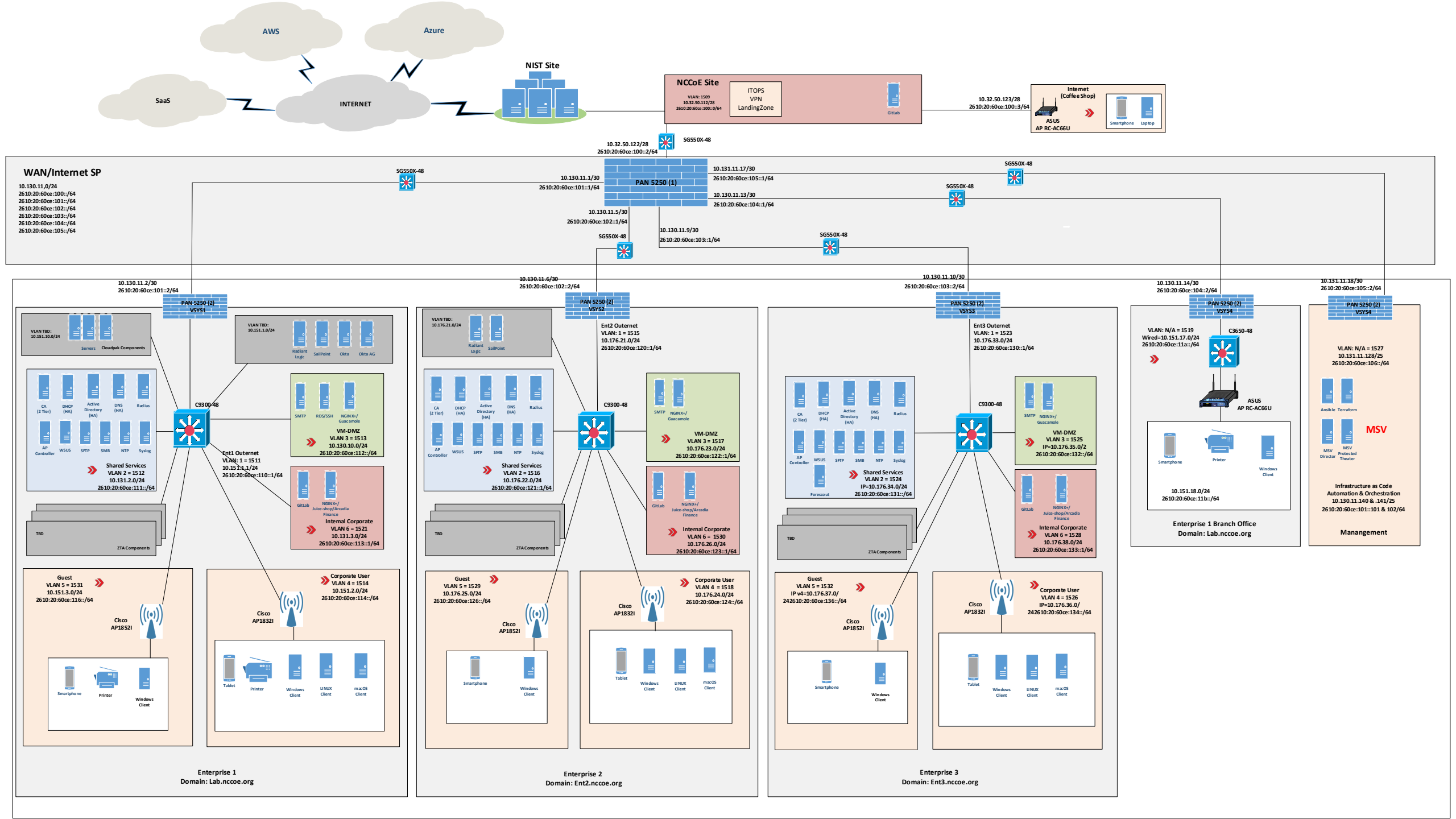
NSA: Container level at least.`

LP + LF

No single point of policy enforcement

Scope and Status of NCCoE Implementing ZT Effort

- Cultivate, configure, and evaluate Zero Trust hybrid solutions, using commercial collaborators and their technologies.
- Starting product constellations for an identity-based ZT crawl phase
- Example proposed implementation (1 of 8): to be evaluated using FireEye Verifier…
  - Cisco (Secure Endpoint - AMP)
  - Forescout (EyeSight)
  - IBM (MaaS360)
  - Ivanti (Unified Endpoint Mgmt. Platform)
  - Lookout (Mobile Endpoint Security)
  - McAfee (MVISION - ENS, Endpoint Detection and Response, Insights, Mobile, and Device Control)
  - Microsoft (Endpoint Manager & Defender for Endpoint, Endpoint Manager MDM + MAM, Intune)
  - Palo Alto (Prisma Access- Global Protect, Cortex XDR)
  - PC Matic (Full Endpoint Suite - PC Matic Pro, Ransomware Lifetime, RDP Lifetime)
  - Symantec (Symantec Endpoint Protection Complete)
- Significant design gaps, getting here as consumed 11 months


AWS
Azure
SaaS
INTERNET
NIST Site
NCCoE Site
VLAN: 1509
10.32.50.112/28
2610:20:60ce:100::0/64
ITOPS
VPN
LandingZone
Gitlab
10.32.50.123/28
2610:20:60ce:100::3/64
Internet
(Coffee Shop)
ASUS
AP RC-AC66U
Smartphone
Laptop
10.32.50.122/28
2610:20:60ce:100::2/64
SG550X-48
WAN/Internet SP
10.130.11.0/24
2610:20:60ce:100::/64
2610:20:60ce:101::/64
2610:20:60ce:102::/64
2610:20:60ce:103::/64
2610:20:60ce:104::/64
2610:20:60ce:105::/64
PAN 5250 (1)
10.130.11.1/30
2610:20:60ce:101::1/64
10.131.11.17/30
2610:20:60ce:105::1/64
10.130.11.13/30
2610:20:60ce:104::1/64
10.130.11.5/30
2610:20:60ce:102::1/64
10.130.11.9/30
2610:20:60ce:103::1/64
10.130.11.2/30
2610:20:60ce:101::2/64
PAN 5250 (2)
VSYS1
10.130.11.6/30
2610:20:60ce:102::2/64
PAN 5250 (2)
VSYS2
10.130.11.10/30
2610:20:60ce:103::2/64
PAN 5250 (2)
VSYS3
10.130.11.14/30
2610:20:60ce:104::2/64
PAN 5250 (2)
VSYS4
10.131.11.18/30
2610:20:60ce:105::2/64
PAN 5250 (2)
VSYS4
C9300-48
C3650-48
VLAN TBD:
10.151.10.0/24
Servers
Cloudpak Components
VLAN TBD:
10.151.1.0/24
Radiant Logic
SailPoint
Okta
Okta AG
CA (2 Tier)
DHCP (HA)
Active Directory (HA)
DNS (HA)
Radius
AP Controller
WSUS
SFTP
SMB
NTP
Syslog
Shared Services
VLAN 2 = 1512
10.131.2.0/24
2610:20:60ce:111::/64
TBD
ZTA Components
SMTP
RDS/SSH
NGINX+/ Guacamole
VM-DMZ
VLAN 3 = 1513
10.130.10.0/24
2610:20:60ce:112::/64
Ent1 Outernet
VLAN: 1 = 1511
10.151.1.1/24
2610:20:60ce:110::1/64
GitLab
NGINX+/ Juice-shop/Arcadia Finance
Internal Corporate
VLAN 6 = 1521
10.131.3.0/24
2610:20:60ce:113::1/64
Guest
VLAN 5 = 1531
10.151.3.0/24
2610:20:60ce:116::/64
Cisco AP1852I
Smartphone
Printer
Windows Client
Corporate User
VLAN 4 = 1514
10.151.2.0/24
2610:20:60ce:114::/64
Cisco AP1832I
Tablet
Printer
Windows Client
LINUX Client
macOS Client
Enterprise 1
Domain: Lab.nccoe.org
VLAN TBD:
10.176.21.0/24
Radiant Logic
SailPoint
Ent2 Outernet
VLAN: 1 = 1515
10.176.21.0/24
2610:20:60ce:120::1/64
Shared Services
VLAN 2 = 1516
10.176.22.0/24
2610:20:60ce:121::1/64
SMTP
NGINX+/ Guacamole
VM-DMZ
VLAN 3 = 1517
10.176.23.0/24
2610:20:60ce:122::1/64
Internal Corporate
VLAN 6 = 1530
10.176.26.0/24
2610:20:60ce:123::1/64
Guest
VLAN 5 = 1529
10.176.25.0/24
2610:20:60ce:126::/64
Corporate User
VLAN 4 = 1518
10.176.24.0/24
2610:20:60ce:124::/64
Enterprise 2
Domain: Ent2.nccoe.org
Ent3 Outernet
VLAN: 1 = 1523
10.176.33.0/24
2610:20:60ce:130::/64
Shared Services
VLAN 2 = 1524
IP=10.176.34.0/24
2610:20:60ce:131::/64
Forescout
VM-DMZ
VLAN 3 = 1525
IP=10.176.35.0/2
2610:20:60ce:132::/64
Internal Corporate
VLAN 6 = 1528
10.176.38.0/24
2610:20:60ce:133::1/64
Guest
VLAN 5 = 1532
IP v4=10.176.37.0/
242610:20:60ce:136::/64
Corporate User
VLAN 4 = 1526
IP=10.176.36.0/
242610:20:60ce:134::/64
Enterprise 3
Domain: Ent3.nccoe.org
VLAN: N/A = 1519
Wired=10.151.17.0/24
2610:20:60ce:11a::/64
10.151.18.0/24
2610:20:60ce:11b::/64
Printer
Windows Client
Enterprise 1 Branch Office
Domain: Lab.nccoe.org
VLAN: N/A = 1527
10.131.11.128/25
2610:20:60ce:106::/64
Ansible
Terraform
MSV Director
MSV Protected Theater
MSV
Infrastructure as Code
Automation & Orchestration
10.130.11.140 & .141/25
2610:20:60ce:101::101 & 102/64
Manangement

OASIS OCA ZT Questions
Proposed Next Steps

- Question 1 – Are there open source functional equivalents of the current proposed implementation in the NCCoE?
- Question 2 – If there are such alternatives, should OASIS OCA ZT WG consider configuring and analyzing the open source alternative implementations?
- Observations based on NIST call – NIST NCCoE has indicated interest in collaborating on such an effort if we are.
  - We could leverage NIST definitions because they define the business need and attestation standard.
  - We could leverage NIST Use cases and testing plan, for true comparability.
- Question 3 – May I draft a proposal sketch to NIST, to formalize access to evaluation plans, decisions, use cases?

Previous update

- <pending updates in February to SP 800-161 Rev 1, DRAFT 3>
  - Attestation formats
  - Formalization of the roles of DevSecOps, curation and Threat Intel feeds in ZT
  - Formalization of the ZT Risk Model (including vendor Risk/reputation)
- <pending CISA updates on SBOM and ZT MM>
- <pending ENISA pursuant to NIS2 vote in October>
- <pending ZT extension to FERC/NERC CIP, by CISA/NIST>
- …

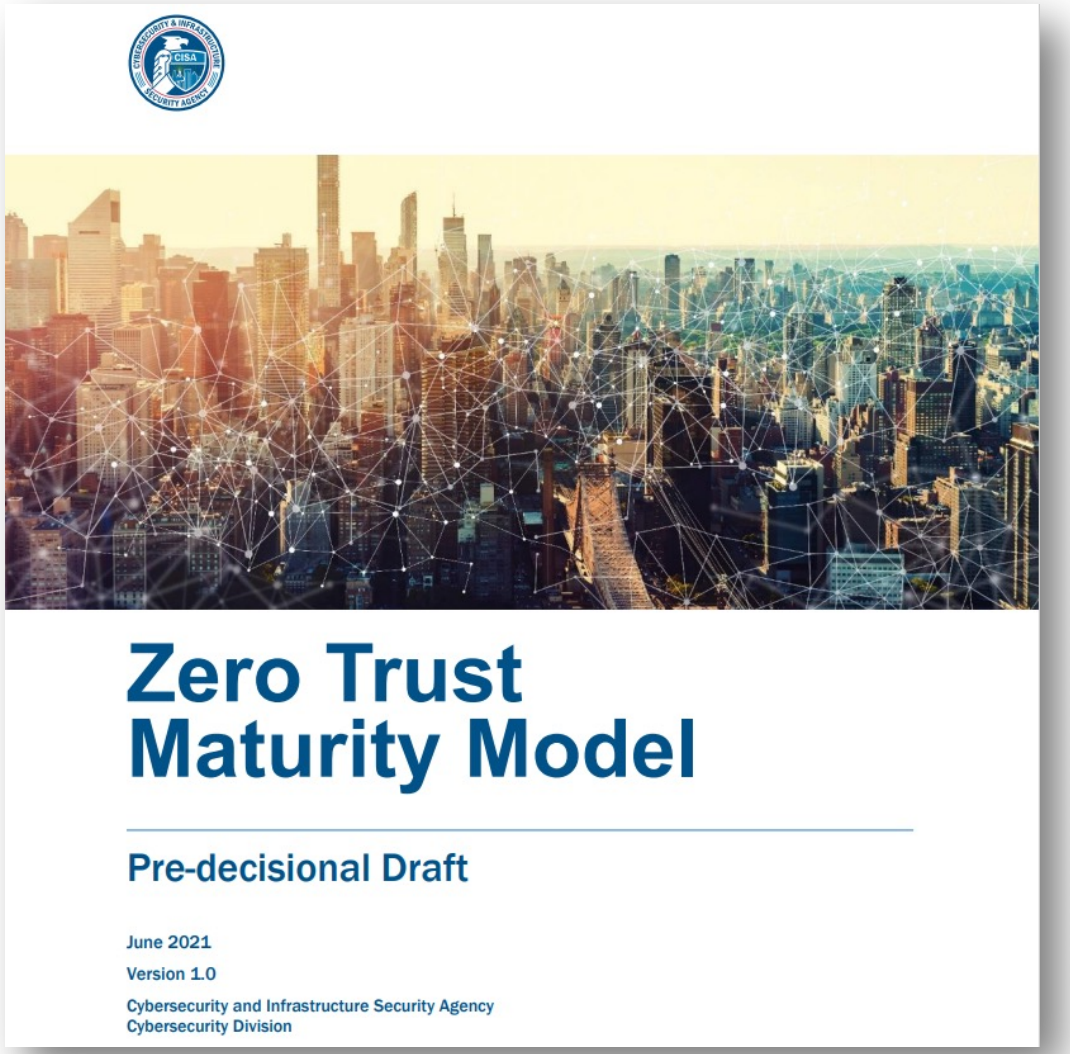
Goals

1. Compare guidance applying ZT principles to existing systems.
    1. NIST
    2. UK NCSC - Emerging
    3. EU NIS2 ENISA – Emerging
    4. Commercial
2. Establish recommendations for integration ZT and non-ZT systems.
3. Evaluate what resulting ZT + non-ZT Policy might looklike.

The need for ZT + non-ZT guidance

CISA Observations from the CISA ZT Maturity Model

CISA ZT MM (DRAFT)



Zero Trust Maturity Model

Pre-decisional Draft

June 2021

Version 1.0

Cybersecurity and Infrastructure Security Agency
Cybersecurity Division

https://www.cisa.gov/sites/default/files/publications/CISA%20Zero%20Trust%20Maturity%20Model_Draft.pdf

	Identity	Device	Network / Environment	Application Workload	Data
Traditional	• Password or multifactor authentication (MFA) • Limited risk assessment	• Limited visibility into compliance • Simple inventory	• Large macro-segmentation • Minimal internal or external traffic encryption	• Access based on local authorization • Minimal integration with workflow • Some cloud accessibility	• Not well inventoried • Static control • Unencrypted
	Visibility and Analytics	*Automation and Orchestration*	*Governance*		
Advanced	• MFA • Some identity federation with cloud and on-premises systems	• Compliance enforcement employed • Data access depends on device posture on first access	• Defined by ingress/egress micro-perimeters • Basic analytics	• Access based on centralized authentication • Basic integration into application workflow	• Least privilege controls • Data stored in cloud or remote environments are encrypted at rest
	Visibility and Analytics	*Automation and Orchestration*	*Governance*		
Optimal	• Continuous validation • Real time machine learning analysis	• Constant device security monitor and validation • Data access depends on real-time risk analytics	• Fully distributed ingress/egress micro-perimeters • Machine learning-based threat protection • All traffic is encrypted	• Access is authorized continuously • Strong integration into application workflow	• Dynamic support • All data is encrypted
	Visibility and Analytics	*Automation and Orchestration*	*Governance*		

Figure 2: High-Level Zero Trust Maturity Model

CISA MM Transitioning to ZT [Page 4]

- Transitioning to Zero Trust:

1. Identify Actors on the Enterprise.
2. Identify Assets Owned by the Enterprise.
3. Identify Key Processes and Evaluate Risks Associated with Executing Process.
   - The Basis for Business Impact (Risk)
4. Formulating Policies for the ZTA Candidate.
5. Identifying Candidate Solutions.
6. Initial Deployment and Monitoring

https://www.cisa.gov/sites/default/files/publications/CISA%20Zero%20Trust%20Maturity%20Model_Draft.pdf

Observations from CISA ZT MM Document

[Page ii]

- The path to zero trust is an incremental process that will take years to implement.
- Legacy infrastructure and systems may not support a zero trust implementation.

[Page 2] Zero trust presents a shift from a location-centric model to a more data-centric approach for fine-grained security controls between users, systems, data and assets that change over time; for these reasons, moving to a ZTA is non-trivial. This provides the visibility needed to support the development, implementation, enforcement, and evolution of security policies. More fundamentally, zero trust may require a change in an organization's philosophy and culture around cybersecurity. The path to zero trust is a journey that will take years to implement.

[Page 3] 6. Challenge The Federal Government faces several challenges in transitioning to ZTA. First, legacy systems rely on "implicit trust"; this concept conflicts with the core principle of adaptive evaluation of trust within a ZTA. Additionally, existing infrastructures are also built on implicit trust and must either be rebuilt or replaced. To rebuild or replace information technology (IT) infrastructure and mission systems requires a significant investment on the part of agencies. Lastly, there is no consensus on or formal adoption of a maturity model for ZTA. While proposals for maturity models have been put forth, current initiatives for kickstarting zero trust adoption are often focused on the network layer and do not present a holistic approach for transition

https://www.cisa.gov/sites/default/files/publications/CISA%20Zero%20Trust%20Maturity%20Model_Draft.pdf

NIST Application of ZT consistent security measures to existing systems in use.

NIST Security Measures for EO-Critical Software Use – ZT aligned, but not ZT complete

NIST: Security Measures for EO-Critical Software Use

Aimed at near term implementation, in contrast to the years required to implement full zero trust and potential the replacement or development of existing systems.

- *(i) Within 60 days of the date of this order, the Secretary of Commerce acting through the Director of NIST, in consultation with the Secretary of Homeland Security acting through the Director of CISA and with the Director of OMB, shall publish guidance outlining security measures for critical software as defined in subsection (g) of this section, including applying practices of least privilege, network segmentation, and proper configuration.*
- *(j) Within 30 days of the issuance of the guidance described in subsection (i) of this section, the Director of OMB acting through the Administrator of the Office of Electronic Government within OMB shall take appropriate steps to require that agencies comply with such guidance.*

- *NIST has identified security measures that are fundamental for meeting these objectives. These "Security Measures for EO-Critical Software Use" are not intended to be comprehensive, nor are they intended to eliminate the need for other security measures that federal agencies implement as part of their existing requirements and cybersecurity programs. Agencies should continue their efforts to secure systems and networks that EO-critical software runs on and to manage cyber supply chain risk (see FAQ #4), as well as implement zero trust practices (see FAQ #5), which depend on the fundamental security measures. The intent of specifying these security measures is to assist agencies by defining a set of common security objectives for prioritizing the security measures that should be in place to protect EO-critical software use.*

https://www.nist.gov/itl/executive-order-improving-nations-cybersecurity/security-measures-eo-critical-software-use

FAQ: EO-Critical SW Use

https://www.nist.gov/itl/executive-order-improving-nations-cybersecurity/security-measures-eo-critical-software-use-1

5.What is the relationship between this guidance and zero trust architecture?

Section 3 of the EO directs each federal agency to plan to implement zero trust architecture. All of the security measures for EO-critical software defined in this guidance are also components of a zero trust architecture, although by no means are they complete. Agencies developing plans for migrating to zero trust architecture can incorporate the security measures for EO-critical software use into those plans. For more information on zero trust architecture, see the following Federal Government resources:

- *DISA and NSA, Department of Defense (DOD) Zero Trust Reference Architecture Version 1.0*
- *NIST, SP 800-207, Zero Trust Architecture*
- *NSA, Embracing a Zero Trust Security Model*

Objective 1:

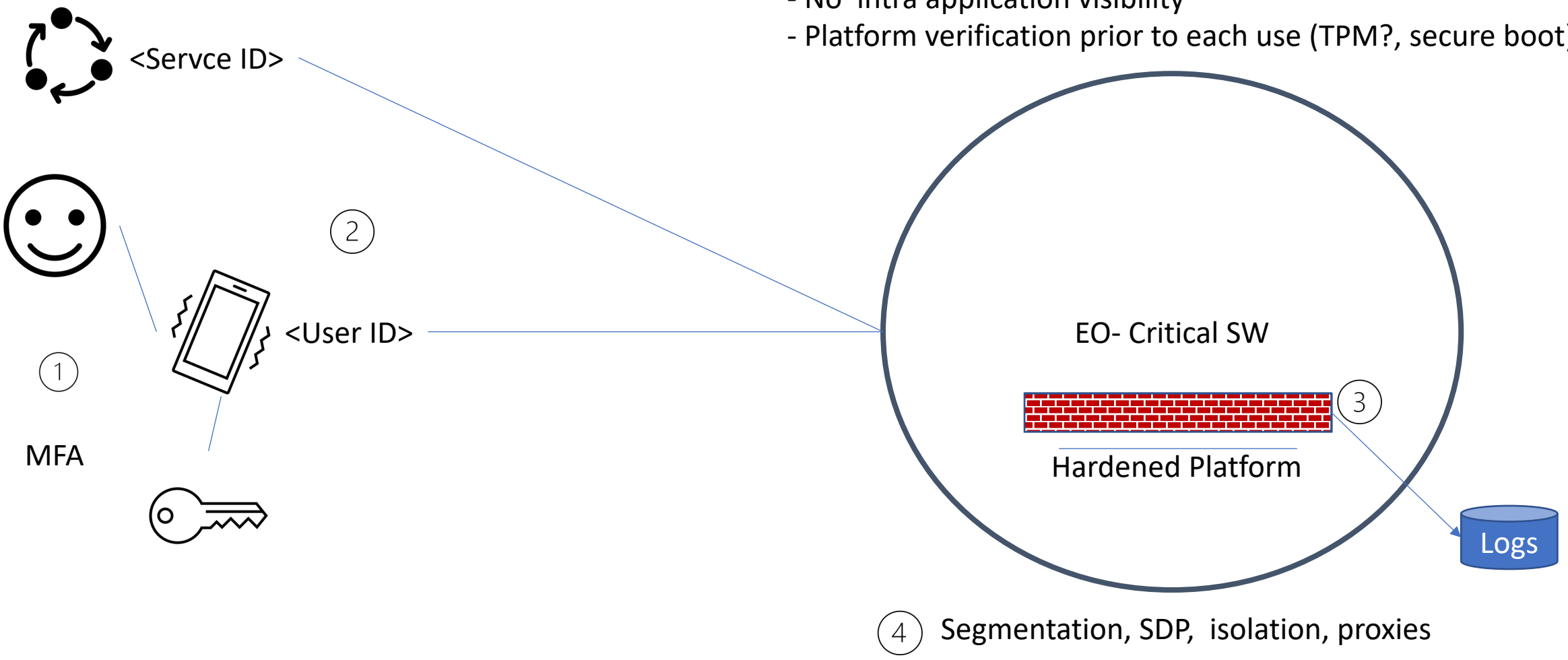
Objective 1: Protect EO-critical software and EO-critical software platforms from unauthorized access and usage.

- SM 1.1: Use multi-factor authentication that is verifier impersonation-resistant for all users and administrators of EO-critical software and EO-critical software platforms. (See FAQ #7.)
- SM 1.2: Uniquely identify and authenticate each service attempting to access EO-critical software or EO-critical software platforms.
- SM 1.3: Follow privileged access management principles for network-based administration of EO-critical software and EO-critical software platforms. Examples of possible implementations include using hardened platforms dedicated to administration and verified before each use, requiring unique identification of each administrator, and proxying and logging all administrative sessions to EO-critical software platforms.
- SM 1.4: Employ boundary protection techniques as appropriate to minimize direct access to EO-critical software, EO-critical software platforms, and associated data. Examples of such techniques include network segmentation, isolation, software-defined perimeters, and proxies.

Objective 1

Differences from ZT:

- Coarser Granularity – bigger than a “container”
- Discovery of accessing Services by Observation
- No intra application visibility
- Platform verification prior to each use (TPM?, secure boot)

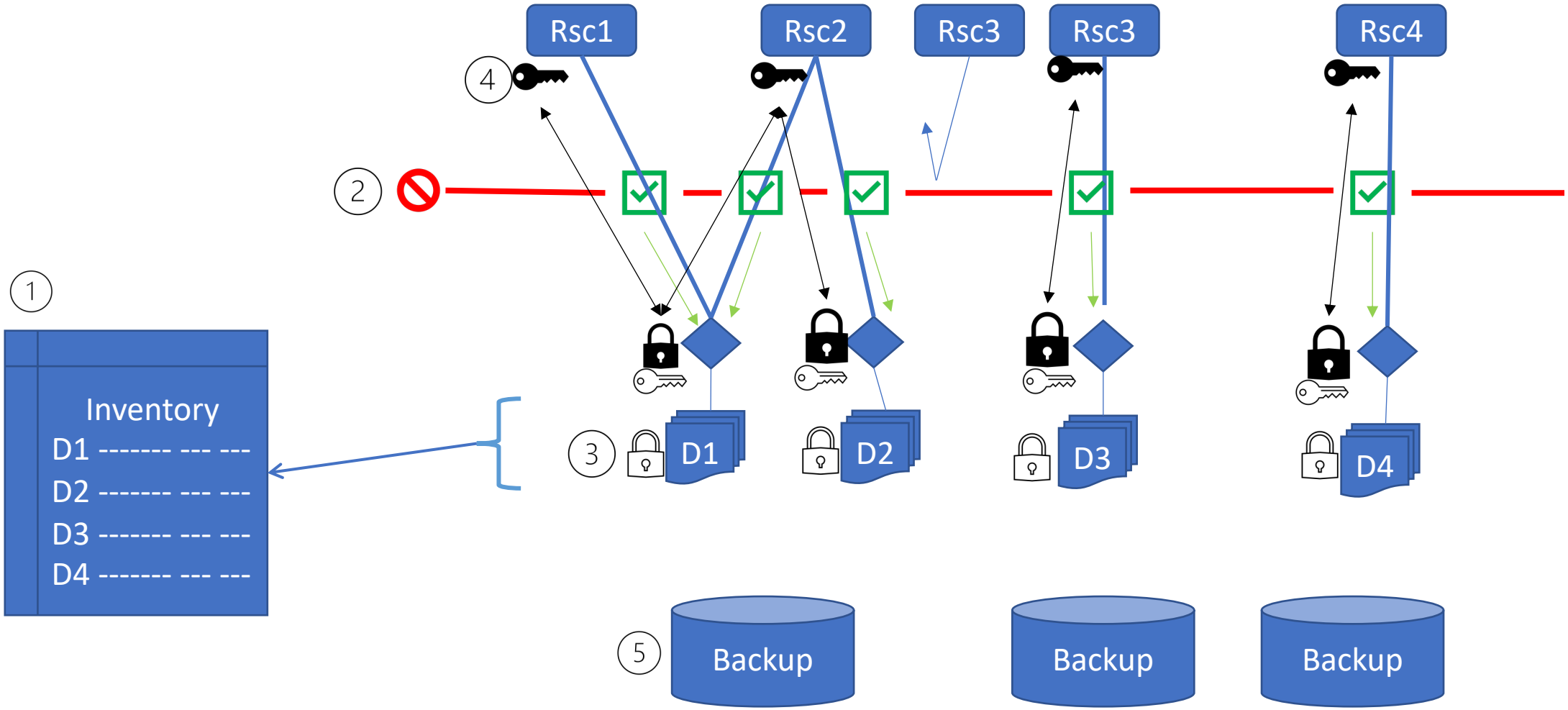


Objective 2:

Objective 2: Protect the confidentiality, integrity, and availability of data used by EO-critical software and EO-critical software platforms. (See FAQ #6.)

- SM 2.1: Establish and maintain a data inventory for EO-critical software and EO-critical software platforms.
- SM 2.2: Use fine-grained access control for data and resources used by EO-critical software and EO-critical software platforms to enforce the principle of least privilege to the extent possible.
- SM 2.3: Protect data at rest by encrypting the sensitive data used by EO-critical software and EO-critical software platforms consistent with NIST's cryptographic standards.
- SM 2.4: Protect data in transit by using mutual authentication whenever feasible and by encrypting sensitive data communications for EO-critical software and EO-critical software platforms consistent with NIST's cryptographic standards.
- SM 2.5: Back up data, exercise backup restoration, and be prepared to recover data used by EO-critical software and EO-critical software platforms at any time from backups.\\

Objective 2 Diagram



Objective 3:

Objective 3: Identify and maintain EO-critical software platforms and the software deployed to those platforms to protect the EO-critical software from exploitation.

- SM 3.1: Establish and maintain a software inventory for all platforms running EO-critical software and all software (both EO-critical and non-EO-critical) deployed to each platform.
- SM 3.2: Use patch management practices to maintain EO-critical software platforms and all software deployed to those platforms. Practices include: rapidly identify, document, and mitigate known vulnerabilities
- SM 3.3: Use configuration management practices to maintain EO-critical software platforms and all software deployed to those platforms. Practices include: (hardened security configurations enforce the principles of least privilege, separation of duties, and least functionality)

Objective 4:

Objective 4: Quickly detect, respond to, and recover from threats and incidents involving EO-critical software and EO-critical software platforms.

- SM 4.1: Configure logging to record the necessary information about security events involving EO-critical software platforms and all software running on those platforms.
- SM 4.2: Continuously monitor the security of EO-critical software platforms and all software running on those platforms.
- SM 4.3: Employ endpoint security protection on EO-critical software platforms to protect the platforms and all software running on them. Capabilities include:
  - protecting the software, data, and platform by identifying, reviewing, and minimizing the attack surface and exposure to known threats.
  - permitting only verified software to execute (e.g., file integrity verification, signed executables, allow listing)
- SM 4.4: Employ network security protection to monitor the network traffic to and from EO-critical software platforms to protect the platforms and their software using networks. Capabilities include:
- SM 4.5: Train all security operations personnel and incident response team members, based on their roles and responsibilities, on how to handle incidents involving EO-critical software or EO-critical software platforms.

Objective 5

Objective 5: Strengthen the understanding and performance of humans' actions that foster the security of EO-critical software and EO-critical software platforms.

- SM 5.1: Train all users of EO-critical software, based on their roles and responsibilities, on how to securely use the software and the EO-critical software platforms.
- SM 5.2: Train all administrators of EO-critical software and EO-critical software platforms, based on their roles and responsibilities, on how to securely administer the software and/or platforms.
- SM 5.3: Conduct frequent awareness activities to reinforce the training for all users and administrators of EO-critical software and platforms, and to measure the training's effectiveness for continuous improvement purposes.

Objective 2

Objective 3

Objective 4

Objective 5

Commercial Example - Microsoft

Focus: Zero Trust Model helps by:

- Applying controls and technologies to discover Shadow IT.
- Ensuring appropriate in-app permissions.
- Limiting access based on real-time analytics.
- Monitoring for abnormal behavior.
- Controlling user actions.
- Validating secure configuration options.

Initial Deployment Objectives:

- I. Gain visibility into the activities and data in your applications by connecting them via APIs.
- II. Discover and control the use of shadow IT.
- III. Protect sensitive information and activities automatically by implementing policies.

Additional deployment objectives:

- IV. Deploy adaptive access and session controls for all apps.
- V. Strengthen protection against cyber threats and rogue apps.
- VI. Assess the security posture of your cloud environments

https://docs.microsoft.com/en-us/security/zero-trust/deploy/applications

Next Steps – In Process

Feel free to adjust, correct refine, …

Next Steps

- Diagram for each Objective
  - Is the diagram for Objective 1 an adequate first iteraction for each objective?
- Identify differences from more conformant ZT
- Commercial examples of applying ZT Principles to existing systems
  - Microsoft
  - AWS
  - Google

  * <other group contributors may want to consider summarizing these or others>

Questions – for consider ation by the broader group

- Does emerging guidance (non-ZT) provide abstractions for aligning postures for integrated ZT and non-ZT ?
  - Isolation
  - Authentication & verification
  - Trust-ability of the platform
  - Completeness of the User, Service universe
- Are there limits to integration that should be considered?
  - Directional limitations?
  - Restrictions on change (to address supply chain absent devops)?
  - Policy on detection/nonconformance in non-ZT domain?