**Audit Workpaper: Security Architect Review and Approval Process**

**Control Objective (Why is the control performed and how does it manage an issue?)**

The control ensures compliance with NAB’s security requirements for container deployment pipelines and clusters by mitigating risks associated with insecure container configurations. This promotes a secure software development lifecycle and reduces the likelihood of unauthorized access, privilege escalation, or operational instability.

**Control Frequency (How often is the control performed?)**

* Periodic reviews conducted at defined intervals.
* Additional reviews are performed during significant pipeline or cluster updates.

**Control Ownership (Who performs the control and who is responsible for the control?)**

* **Performer:** Security Architect.
* **Responsible Party:** Security Architect, with support from DevOps and application teams.

**Control Type**

* **Preventative.**

**Control Classification**

* **Manual** (involves human oversight and decision-making).

**Where is it performed (Location/System)?**

* **Location:** Across all tenant environments.
* **System:** Deployment pipelines, clusters, and associated configuration management tools (e.g., CI/CD platforms, Kubernetes clusters).

**Background Detail (if required)**

* NAB requires a formalized security architecture approval process to validate configurations for pipelines and clusters. This process ensures alignment with security baselines, industry standards, and regulatory requirements for containerized environments.

**How is the control performed?**

* **Inputs and Sources:**
  + Configuration files, pipeline settings, and cluster parameters.
  + Security standards and NAB compliance requirements.
* **What is done to process the inputs:**
  + Review configuration files and cluster parameters to verify compliance with required security tools and baselines.
  + Validate alignment with NAB’s security architecture requirements.
* **Relevant Thresholds/Limits:**
  + Ensure all configurations meet predefined security baselines.
  + Verify that critical security tools (e.g., vulnerability scanners, monitoring agents) are implemented.
* **How outliers are tracked, managed, escalated:**
  + Any non-compliant configurations are flagged and escalated to DevOps or tenant teams for remediation.
  + Escalated issues are logged and tracked until resolution.
* **How the control review is evidenced:**
  + Review outcomes are documented in approval logs or reports.
  + Evidence includes approved configuration files, meeting minutes, or email confirmations from the Security Architect.

This structured approach ensures secure and compliant use of containerized environments across NAB

**Audit Workpaper: Security Guidelines and Approval for Container Orchestration Layers**

**Control:**

The Security Architect periodically reviews and approves security configurations for pipelines and clusters, ensuring secure use of containers in line with NAB requirements.

**Control Test: Validating Security Guidelines for Orchestration Layers**

**Test Step:**

Validate the availability of security guidelines, threat assessments, and periodic reviews by the Security Architect for all orchestration layers in use.

**Control Attribute:**

* **Control Objective: To validate that security guidelines, threat assessments, and annual reviews are in place for all container orchestration layers used in NAB, ensuring compliance with NAB’s Cloud Security Assurance Methodology (CSAM) and protecting against vulnerabilities.**

**Test Evaluation Criteria:**

* **Defined by:** NAB’s Cloud Security Assurance Methodology (CSAM) policy and standard operating procedures.
* **Reasonability:** Security guidelines should exist for all orchestration layers, based on threat assessments, and must be reviewed annually to align with NAB requirements.

**Key Test Steps:**

1. **Sub-sample selection and details:**
   * Reviewed container orchestration data from Crowdstrike Report (12th Oct FY2024), AWS EKS/ECS inventory (12th Nov), and Azure AKS inventory (27th Oct).
   * Sampling method: Selected all available orchestration layers used within NAB based on provided evidence.
2. **What was looked at:**
   * [Evidence 1] Crowdstrike report on container inventory for visibility issues.
   * [Evidence 2] EKS and ECS inventory detailing orchestration layers for AWS containers.
   * [Evidence 3] AKS inventory listing orchestration for Azure-hosted containers.
   * Security guidelines for each orchestration layer.
3. **How the test was executed:**
   * Analyzed evidence to classify containers by orchestration layers (EKS, ECS, AKS, EC2-hosted, and VM-hosted).
   * Validated the existence of threat assessments, security guidelines, and annual reviews by the Security Architect for each orchestration layer.

**Outcome:**

1. **Result or Condition:**
   * **AWS:**
     + EKS and ECS have security guidelines and annual reviews.
     + EC2-hosted containers lack both threat assessments and security guidelines.
   * **Azure:**
     + AKS has security guidelines and annual reviews.
     + VM-hosted containers lack threat assessments and security guidelines.
2. **Comparison Against Expectations:**
   * **Expected:** Security guidelines, threat assessments, and periodic reviews must exist for all orchestration layers.
   * **Actual:** EC2-hosted and VM-hosted containers do not meet the expectations, violating the CSAM framework.
3. **Rationale for Pass/Fail:**
   * The control failed as there were no threat assessments or security guidelines for EC2-hosted and VM-hosted containers, leaving them vulnerable and non-compliant with NAB security policies.

**Control Test Result:**

* **Fail:** The control did not ensure periodic review and security guidelines for all orchestration layers.

**Exceptions:**

* **Exception 1:** No threat assessment or security guidelines were available for EC2-hosted containers on AWS.
* **Exception 2:** No threat assessment or security guidelines were available for VM-hosted containers on Azure.

**Next Steps:**

The Security Architect must establish security guidelines and conduct periodic reviews for containers without orchestration layers to align with NAB’s CSAM policy.

4o

Top of Form

**Control Test Result: Design Effectiveness Failure**

The control failed to ensure the design effectiveness of security governance for containerized environments and orchestration layers. Specifically, **ECS and EC2/VM-hosted containers** lack both approved baseline configurations and security guidelines, leaving them highly vulnerable to misconfigurations, unauthorized access, and potential exploitation. Additionally, **EKS and AKS**, while partially compliant, have incomplete and unapproved security baselines and are missing periodic reviews for unmanaged containers, creating significant gaps in adherence to NAB security standards and CIS benchmarks. The absence of comprehensive security documentation, threat assessments, and periodic reviews undermines the organization’s ability to enforce consistent security practices, increasing the risk of non-compliance, privilege escalation attacks, and exposure of critical assets to malicious actors. Immediate corrective action is required to address these deficiencies and strengthen the security posture across all containerized environments.

**Control Test Result:**

**Fail:** The control failed to ensure the presence of approved baseline configurations and tools documentation for all containerized environments. This poses a significant risk as:

* **ECS and EC2/VM-hosted containers** lack approved security baselines, leaving them vulnerable to misconfigurations, unauthorized access, and potential exploitation.
* **EKS and AKS** configurations, while partially compliant, remain incomplete and unapproved, creating a gap in ensuring consistent adherence to CIS benchmarks and NAB security standards.

**Risk Highlight:**  
Without approved security baselines and validated tools documentation:

* Containers may run with insecure configurations, exposing critical assets to threats such as privilege escalation, unauthorized access, and resource mismanagement.
* The lack of alignment with industry standards increases the likelihood of non-compliance with NAB policies and regulatory frameworks.

Immediate action is required to address these deficiencies and mitigate associated risks.

4o

**You said:**

similarly for this Control Test Result: Fail: The control did not ensure periodic review and security guidelines for all orchestration layers.

**ChatGPT said:**

ChatGPT

**Control Test Result:**

**Fail:** The control failed to ensure periodic review and security guidelines for all orchestration layers. This creates significant risks as:

* **ECS and EC2/VM-hosted containers** lack both threat assessments and documented security guidelines, leaving these environments unprotected and vulnerable to exploitation.
* **EKS and AKS** guidelines, while compliant, do not extend to unmanaged containers, creating inconsistencies and gaps in overall security governance.

**Risk Highlight:**  
Without periodic reviews and comprehensive security guidelines:

* Orchestration layers and unmanaged containers are exposed to potential misconfigurations, unauthorized access, and privilege escalation attacks.
* The absence of documented standards increases non-compliance with NAB policies and regulatory frameworks, weakening the overall security posture.

Immediate action is needed to develop, review, and implement security guidelines for all orchestration layers and unmanaged container environments.

**Audit Workpaper: Validation of Approved Security Tools and Configurations for Containers**

**Control:**

The Security Architect periodically reviews and approves all security configurations for container deployment pipelines and clusters, including required security tools (**what**), to ensure compliance with NAB’s security requirements and mitigate risks associated with insecure configurations while promoting a secure software development lifecycle (**why**). This formalized review process is conducted during significant pipeline or cluster updates and at regular intervals (**when**) across all tenant environments (**where**) by the Security Architect in collaboration with DevOps and application teams (**who**). Configurations are validated through a structured approval process that ensures alignment with security standards (**how**) and comprehensive oversight of containerized environments (**how much**).

**Control Test: Validation of Security Tools and Baseline Configurations**

**Test Step:**

To verify that all software tools and security configurations on containers have been reviewed and approved by the Security Architect.

**Control Attribute:**

* Documentation of the approved software tools for containers.
* Documentation of the approved baseline security configurations for containers.
* Periodic review of the documentation by the Security Architect.

**Test Evaluation Criteria:**

* **Defined by:** CIS benchmarks and NAB’s Cloud Security Assurance Methodology (CSAM).
* **Reasonability:** The approved documentation for baseline security configurations and allowed software tools should exist for all containerized environments. The periodic review process must be validated for its alignment with CIS benchmarks and NAB policies.

**Key Test Steps:**

1. **Sub-sample selection and details:**
   * Reviewed orchestration services used in NAB: EKS, AKS, ECS, and containers running directly on EC2 or Virtual Machines (VMs).
   * Validated security-related documentation provided by Group Security and NEF teams.
2. **What was looked at:**
   * **Documentation of minimum approved baseline security configurations:**
     + Verified the existence of a Kubernetes security baseline configuration document (e.g., Kubernetes.xlsx).
   * **Documentation of approved software tools for containers:**
     + Checked whether approved tools list was documented to ensure only secure tools are installed.
   * Evidence reviewed: Kubernetes.xlsx, container inventory reports from EKS, ECS, AKS, and Crowdstrike report.
3. **How the test was executed:**
   * Validated Kubernetes.xlsx for alignment with CIS benchmarks for EKS and AKS.
   * Checked documentation for ECS and EC2/VM-hosted containers to confirm if baseline configurations and tools are approved.
   * Verified if periodic reviews of these documents were conducted by the Security Architect.

**Outcome:**

1. **Result or Condition:**
   * **EKS and AKS:**
     + Security baseline configurations are being developed and documented (e.g., Kubernetes.xlsx) by Group Security. However, the documentation is incomplete and pending approval by the Security Architect.
   * **ECS and EC2/VM-hosted Containers:**
     + No approved security baseline configurations or documentation of allowed software tools are available for ECS or containers hosted directly on EC2 or Virtual Machines.
2. **Comparison Against Expectations:**
   * **Expected:** Approved documentation for security baselines and tools should exist for all containerized environments, aligned with CIS benchmarks and NAB’s security policies. Periodic reviews should be evidenced.
   * **Actual:**
     + Partial compliance for EKS and AKS (in-progress documentation).
     + Non-compliance for ECS and EC2/VM-hosted containers (no documentation or review).
3. **Rationale for Pass/Fail:**
   * The control failed as there is no approved documentation for baseline configurations and software tools for ECS and EC2/VM-hosted containers. EKS and AKS documentation is incomplete and not yet approved.

**Control Test Result:**

* **Fail:** The control did not ensure approved baseline configurations and tools documentation for all containerized environments.

**Exceptions:**

1. **Exception 1:** ECS and EC2/VM-hosted containers lack approved security baselines and tools documentation.
2. **Exception 2:** EKS and AKS security baseline documentation is incomplete and pending approval by the Security Architect.
3. **Control Assessment: Design Effectiveness Evaluation**
4. **1. Is this the right control to mitigate the risk?**
5. **Result:** **Fail**  
   **Rationale:** While the control aims to mitigate risks associated with insecure container configurations, it is not effectively designed to address gaps in security baselines and periodic reviews for all containerized environments and orchestration layers. ECS and EC2/VM-hosted containers lack comprehensive security guidelines, and periodic reviews for EKS and AKS are incomplete.  
   **Linked Exception:** Exception 1 and 2 regarding the absence of approved security baselines and lack of periodic review.
6. **2. Does the control achieve its objective?**
7. **Result:** **Fail**  
   **Rationale:** The control does not ensure that all containers adhere to approved security baselines or include validated software tools. This leaves critical gaps in mitigating risks, especially for ECS and unmanaged container environments.  
   **Linked Exception:** Exception 1 and 2 regarding incomplete and missing security baselines.
8. **3. Is the control performed by the right people with requisite skills, knowledge, and experience?**
9. **Result:** **Pass**  
   **Rationale:** The control is performed by the Security Architect and relevant teams, who possess the necessary expertise and knowledge to implement and review security configurations.
10. **4. Does the control have adequate segregation of duties?**
11. **Result:** **Pass**  
    **Rationale:** The control maintains adequate segregation, as the Security Architect performs reviews independently of the deployment and operations teams, ensuring objectivity in validation.
12. **5. Is the control performed at the right time or in the right stage of the process?**
13. **Result:** **Fail**  
    **Rationale:** The control is not consistently performed during significant pipeline updates or cluster deployments for all orchestration layers, especially for unmanaged containers.  
    **Linked Exception:** Exception 1 regarding unmanaged environments and incomplete reviews.
14. **6. Is the control performed at the right frequency?**
15. **Result:** **Fail**  
    **Rationale:** The control lacks the periodicity required for effective governance. Annual reviews were incomplete for EKS, and no reviews were conducted for ECS or EC2/VM-hosted containers.  
    **Linked Exception:** Exception 2 regarding incomplete and inconsistent review frequencies.
16. **7. Is the control sustainable?**
17. **Result:** **Fail**  
    **Rationale:** The absence of a centralized repository for managing container security baselines and tools makes the control unsustainable. Additionally, the lack of clear guidelines for ECS and unmanaged containers hinders scalability.  
    **Linked Exception:** Exception 1 regarding missing documentation and sustainability challenges.
18. **8. How does it manage/escalate an issue?**
19. **Result:** **Fail**  
    **Rationale:** The control lacks mechanisms for detecting and escalating issues, as no comprehensive guidelines exist for ECS and unmanaged containers. Furthermore, there is no evidence of tracking or escalating gaps in periodic reviews.  
    **Linked Exception:** Exception 2 regarding escalation and documentation gaps.
20. **9. Is the control evidenced?**
21. **Result:** **Fail**  
    **Rationale:** Documentation for security baselines is incomplete for EKS and AKS and entirely missing for ECS and EC2/VM-hosted containers. The lack of sufficient evidence undermines the control’s effectiveness.  
    **Linked Exception:** Exception 1 and 2 regarding missing documentation.
22. **10. Does the control have adequate management focus?**
23. **Result:** **Fail**  
    **Rationale:** Insufficient attention has been given to addressing gaps in unmanaged containers and completing the baseline security configurations. Management oversight is limited in enforcing adherence to NAB’s security standards across all environments.  
    **Linked Exception:** Exception 2 regarding gaps in management focus and execution.

Bottom of Form