Guidance for Containers and Container Orchestration Tools

Joel Weisz, Standards Manager Emerging Standards
PCI Security Standards Council

Randy Bartels, VP, Security Services
KirpatrickPrice
Special Interest in Containers Orchestration Tools
Agenda

Overview of History and SIG Process

2021 SIG Paper

2023 SIG Process
History

• Community driven

• Over a dozen guidance documents produced since 2010

• Documentation is used by the entire payment card industry
History

- Community driven
- Over a dozen guidance documents produced since 2010
- Documentation is used by the entire payment card industry
History

- Community driven
- Over a dozen guidance documents produced since 2010
- Documentation is used by the entire payment card industry
SIG Process

SIG Proposal from Community

Selection of Proposals for Vote
SIG Process

1. SIG Proposal from Community
2. Selection of Proposals for Vote
3. Objectives Created from Proposal
4. Content Developed through Iterative Process
Role Of SIG Participants

**SIG Members**
- Provide subject matter expertise and user experiences
devolve content

**PCI SSC Chair**
- Facilitates discussions,
preparation of deliverables,
and approvals

**Payment Brands’ Technical Working Group & Management Committee**
- Review and final approval
Role Of SIG Participants

SIG Members
Provide subject matter expertise and user experiences to develop content

PCI SSC Chair
Facilitates discussions, preparation of deliverables, and approvals

Payment Brands’ Technical Working Group & Management Committee
Review and final approval
Role Of SIG Participants

SIG Members
Provide subject matter expertise and user experiences develop content

PCI SSC Chair
Facilitates discussions, preparation of deliverables, and approvals

Payment Brands’ Technical Working Group & Management Committee
Review and final approval
Role Of SIG Participants

SIG Members
Provide subject matter expertise and user experiences; develop content

PCI SSC Chair
Facilitates discussions, preparation of deliverables, and approvals

Payment Brands’ Technical Working Group & Management Committee
Review and final approval
It’s an Opportunity to Help the Community

• A collaborative effort of stakeholders, providing best of industry background, knowledge and experience
• Addresses issues important to the community
• Experience-based input
• An opportunity to provide direct input
It’s an Opportunity to Help the Community

• A collaborative effort of stakeholders, providing best of industry background, knowledge and experience
• Addresses issues important to the community
• Experience-based input
• An opportunity to provide direct input
It’s an Opportunity to Help the Community

- A collaborative effort of stakeholders, providing best of industry background, knowledge and experience
- Addresses issues important to the community
- Experience-based input
- An opportunity to provide direct input
It’s an Opportunity to Help the Community

- A collaborative effort of stakeholders, providing best of industry background, knowledge and experience
- Addresses issues important to the community
- Experience-based input
- An opportunity to provide direct input
It’s an Opportunity to Help the Community

- A collaborative effort of stakeholders, providing best of industry background, knowledge and experience
- Addresses issues important to the community
- Experience-based input
- An opportunity to provide direct input
Guidance for Containers and Container Orchestration Tools

Process and Results
Containers

A little background

• Dependency
• Elasticity
Containers

A little background

• Dependency
• Elasticity
Containers

A little background

• Dependency
• Elasticity
Orchestration is used for containers at scale

Complexity creates risk
Container Orchestration Tools

A little more background

- Orchestration is used for containers at scale
- Complexity creates risk
Container Orchestration Tools

A little more background

• Orchestration is used for containers at scale

• Complexity creates risk
Why do we need this document?

Guidance Purpose

• Stakeholder questions:
  • What are containers?
  • What are container orchestration tools?
  • What are the best practices for secure implementation of these technologies?
Why do we need this document?

Stakeholder questions:

- What are containers?
- What are container orchestration tools?
- What are the best practices for secure implementation of these technologies?
The Result

What did we do?

• Foundational
  • Combination of background information and best practices

• Industry specific guidance
  • Consideration of payment environments

• Applicable
  • Based on industry specific use cases
The Result

What did we do?

• Foundational
  • Combination of background information and best practices

• Industry specific guidance
  • Consideration of payment environments

• Applicable
  • Based on industry specific use cases
What did we do?

• Foundational
  • Combination of background information and best practices

• Industry specific guidance
  • Consideration of payment environments

• Applicable
  • Based on industry specific use cases
Foundational Background Information

Figure 1: Architectural Differences between Traditional, Virtualized, and Containerized Deployment
## Threats, Best Practices, and Use Cases

<table>
<thead>
<tr>
<th>Threat</th>
<th>Best Practice</th>
<th>Applicable to Use Case</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.2 Vulnerabilities present on container orchestration tool hosts (commonly Linux VMs) will allow for compromise of container orchestration tools and other components.</td>
<td>a. Host operating system of all the nodes that are part of a cluster controlled by a container orchestration tool should be patched and kept up to date. With the ability to reschedule workloads dynamically, each node can be patched one at a time, without a maintenance window.</td>
<td>X</td>
</tr>
<tr>
<td>10.3 As container orchestration tools commonly run as containers in the clusters, any container with vulnerabilities may allow compromise of container orchestration tools.</td>
<td>a. All container images used for applications running in the cluster should be regularly scanned for vulnerabilities, patches should be regularly applied, and the patched images redeployed to the cluster.</td>
<td>X X</td>
</tr>
<tr>
<td><strong>11. Resource Management</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11.1 A compromised container could disrupt the operation of applications due to excessive use of shared resources.</td>
<td>a. All workloads running via a container orchestration system should have defined resource limits to reduce the risk of &quot;noisy neighbors&quot; causing availability issues with workloads in the same cluster.</td>
<td>X</td>
</tr>
<tr>
<td><strong>12. Container Image Building</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12.1 Container base images downloaded from untrusted sources, or which contain unnecessary packages, increase the risk of supply chain attacks.</td>
<td>a. Application container images should be built from trusted, up-to-date minimal base images.</td>
<td>X</td>
</tr>
</tbody>
</table>
# Threats, Best Practices, and Use Cases

<table>
<thead>
<tr>
<th>Threat</th>
<th>Best Practice</th>
<th>Applicable to Use Case</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.2   Vulnerabilities present on container orchestration tool hosts (commonly Linux VMs) will allow for compromise of container orchestration tools and other components.</td>
<td>a. Host operating system of all the nodes that are part of a cluster controlled by a container orchestration tool should be patched and kept up to date. With the ability to reschedule workloads dynamically, each node can be patched one at a time, without a maintenance window.</td>
<td>Baseline Case</td>
</tr>
<tr>
<td>10.3   As container orchestration tools commonly run as containers in the clusters, any container with vulnerabilities may allow compromise of container orchestration tools.</td>
<td>a. All container images used for applications running in the cluster should be regularly scanned for vulnerabilities, patches should be regularly applied, and the patched images redeployed to the cluster.</td>
<td>X</td>
</tr>
<tr>
<td>11.1   A compromised container could disrupt the operation of applications due to excessive use of shared resources.</td>
<td>a. All workloads running via a container orchestration system should have defined resource limits to reduce the risk of “noisy neighbors” causing availability issues with workloads in the same cluster.</td>
<td></td>
</tr>
<tr>
<td>12.1   Container base images downloaded from untrusted sources, or which contain unnecessary packages, increase the risk of supply chain attacks.</td>
<td>a. Application container images should be built from trusted, up-to-date minimal base images.</td>
<td></td>
</tr>
</tbody>
</table>
3.2.2 Development and Management of Containerized Applications

3.2.2.1 Description

Creating and managing a container-based workflow for application development and deployment involves several steps, including the initial creation of the container images to be used by the application, the flow of the images as artifacts through the company's CI/CD pipeline, secure storage of the images in a container registry, and their ongoing management and updating.

Phases of the deployment process include:

- Initial development targets application deployment using a container-based on a common base image. The container image is used by Continuos Integration processes in the SDLC.
- The container image is placed into a container registry during testing and deployment.
- The container image is deployed into a production environment to be managed by a container orchestration system.

3.2.2.2 Graphic Representation of the Use Case

![Diagram of Container Build Process]

Figure 6: Container Build Process
# Example Threat Scenarios

## 3.2.2.3 Example Threat Scenario

When building container images, a common requirement is to use secrets—for example, credentials or API keys—to access private data stores to retrieve information. If those secrets are embedded in the resulting container images, attackers can extract the secrets and gain unauthorized access to resources including source code repositories, CI/CD systems, or even container orchestration APIs.

**Example implementation of selected best practices:**

<table>
<thead>
<tr>
<th>Best Practice</th>
<th>Result of Best Practice to Address the Security Threat</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.1.a All secrets needed for the operation of applications hosted on the orchestration platform should be held in encrypted dedicated secrets management systems.</td>
<td>Where secrets are required for running containers, a dedicated secrets management system is employed to ensure that secrets are securely encrypted and made available to only the containers which require them. These systems can determine which containers require access to a specific secret and then inject those secrets into the running container as a mounted file.</td>
</tr>
<tr>
<td>12.4.a Secrets should not be included in application images. Where secrets are required during the building of an image (for example to provide credentials for accessing source code), this process should leverage container builder techniques to ensure that the secret will not be present in the final image.</td>
<td>If an attacker can access source code repositories, CI/CD systems, or the container API, proper management of secrets—for example, not being included in application images, including binary files—prevents these secrets from being used to access additional resources. Ensuring that secrets are not embedded in images can be achieved by using techniques such as multi-stage builds. Here separation between source code compilation and the final container image is achieved by having multiple build processes, and only copying compiled application programs and necessary configuration files to the final stage.</td>
</tr>
</tbody>
</table>
Summary of Guidance

• Useful to our diverse stakeholders
• Background on containers and container orchestration tools
• Best practices to address threats
• Real world use cases and scenarios
2023 SIG Process

- SIG Proposal from Community
- Review and Consolidation
- Selection of Proposals for Vote
- SIG 2023

- September 1, 2022 to October 28, 2022
- January 30, 2023 to February 2023
Member Driven Effort!