



# KubeSmith: A Framework for Hardening Security Policies in Cloud-Native Environments

---

**CHIHYEON CHO**

Department of Computer Science & Engineering

Incheon National University

# Subtitle

---

- Background
- Problem Statement
- Design
- Evaluation
- Conclusion and Future Work

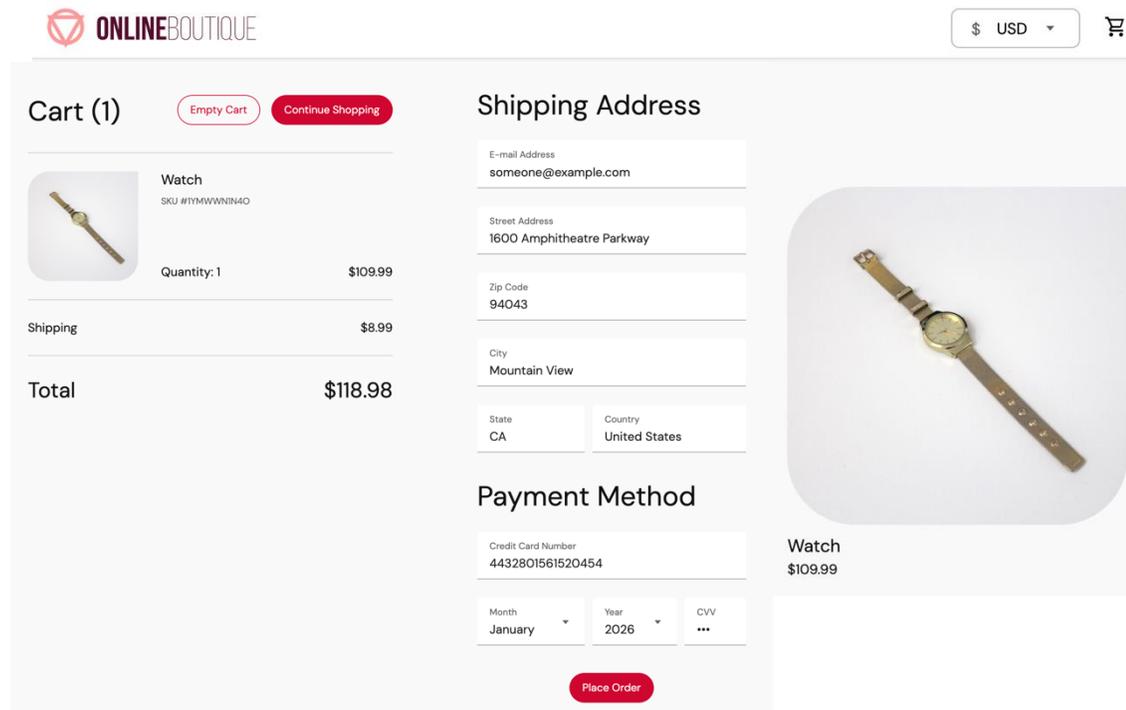
# Background

---

# Microservice

- The most widely used **architecture** to practically implement the characteristics of Cloud Native

## Client



The screenshot shows a checkout page for 'ONLINEBOUTIQUE'. The page is titled 'Client' and features a currency selector set to 'USD' and a shopping cart icon. The main content is divided into three sections: 'Cart (1)', 'Shipping Address', and 'Payment Method'. The 'Cart (1)' section shows a 'Watch' with SKU #YMWNNIN40, a quantity of 1, and a price of \$109.99. Shipping is \$8.99, and the total is \$118.98. The 'Shipping Address' section includes fields for E-mail Address (someone@example.com), Street Address (1600 Amphitheatre Parkway), Zip Code (94043), City (Mountain View), State (CA), and Country (United States). The 'Payment Method' section includes a Credit Card Number (4432801561520454), Month (January), Year (2026), and CVV (\*\*\*). A 'Place Order' button is at the bottom. A large image of the watch is shown on the right side of the page.

ONLINEBOUTIQUE

USD

Cart (1) [Empty Cart](#) [Continue Shopping](#)

 Watch  
SKU #YMWNNIN40  
Quantity: 1 \$109.99

Shipping \$8.99

Total \$118.98

Shipping Address

E-mail Address  
someone@example.com

Street Address  
1600 Amphitheatre Parkway

Zip Code  
94043

City  
Mountain View

State  
CA

Country  
United States

Payment Method

Credit Card Number  
4432801561520454

Month  
January

Year  
2026

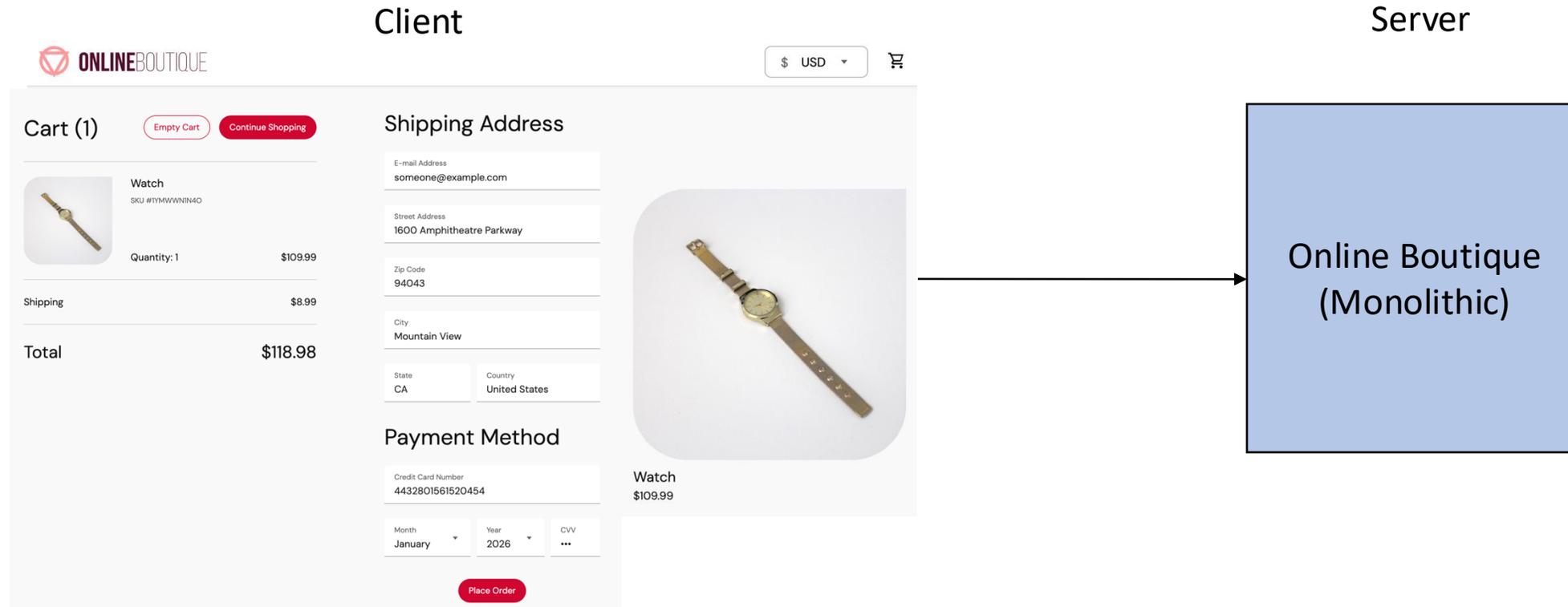
CVV  
\*\*\*

[Place Order](#)

Watch  
\$109.99

# Microservice

- Past - Monolithic



# Microservice

- Cloud Native - Microservice

Client

The screenshot shows a checkout page for 'ONLINEBOUTIQUE'. It features a cart summary on the left, a shipping address form, a product image of a watch, and a payment method form. Red boxes highlight the 'Shipping Address' and 'Payment Method' sections. A shopping cart icon is also highlighted in the top right corner.

Cart (1) Empty Cart Continue Shopping

	Watch SKU #TYMWWNIN40	
Quantity: 1		\$109.99
Shipping		\$8.99
<b>Total</b>		<b>\$118.98</b>

Shipping Address

E-mail Address  
someone@example.com

Street Address  
1600 Amphitheatre Parkway

Zip Code  
94043

City  
Mountain View

State  
CA

Country  
United States

Payment Method

Credit Card Number  
4432801561520454

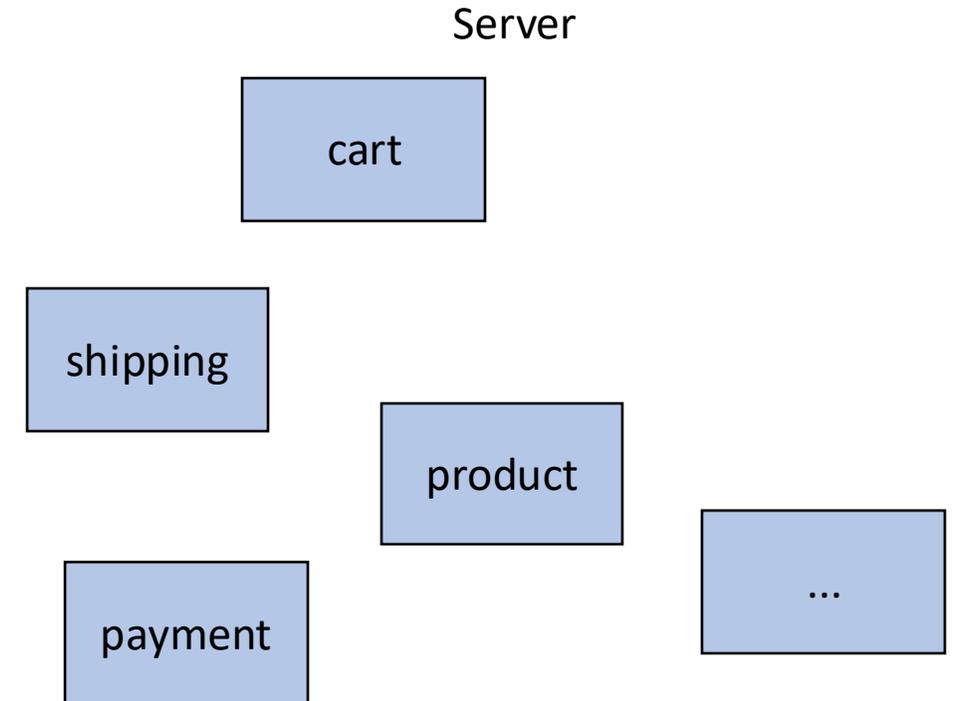
Month  
January

Year  
2026

CVV  
...

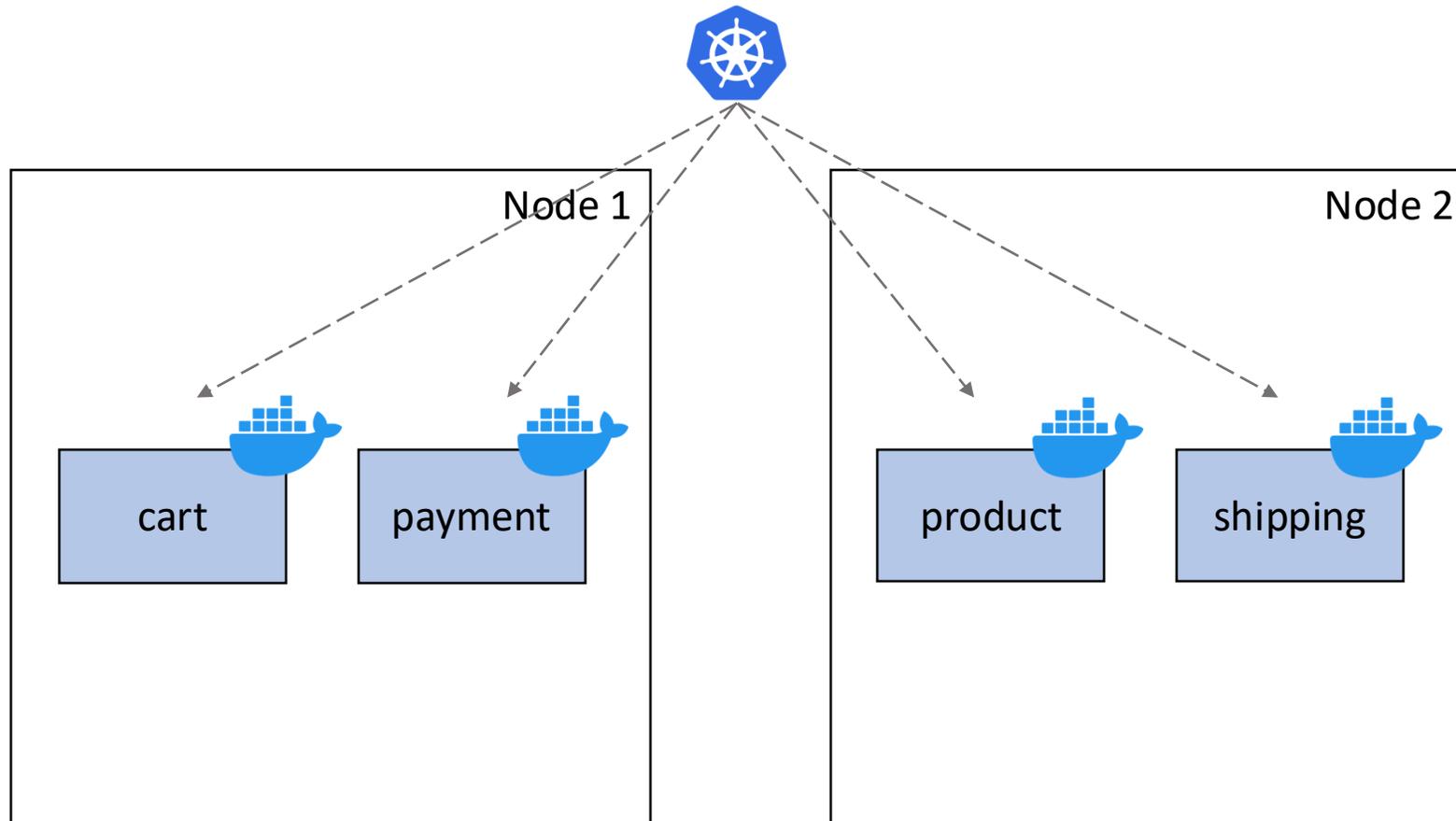
Place Order

Watch  
\$109.99



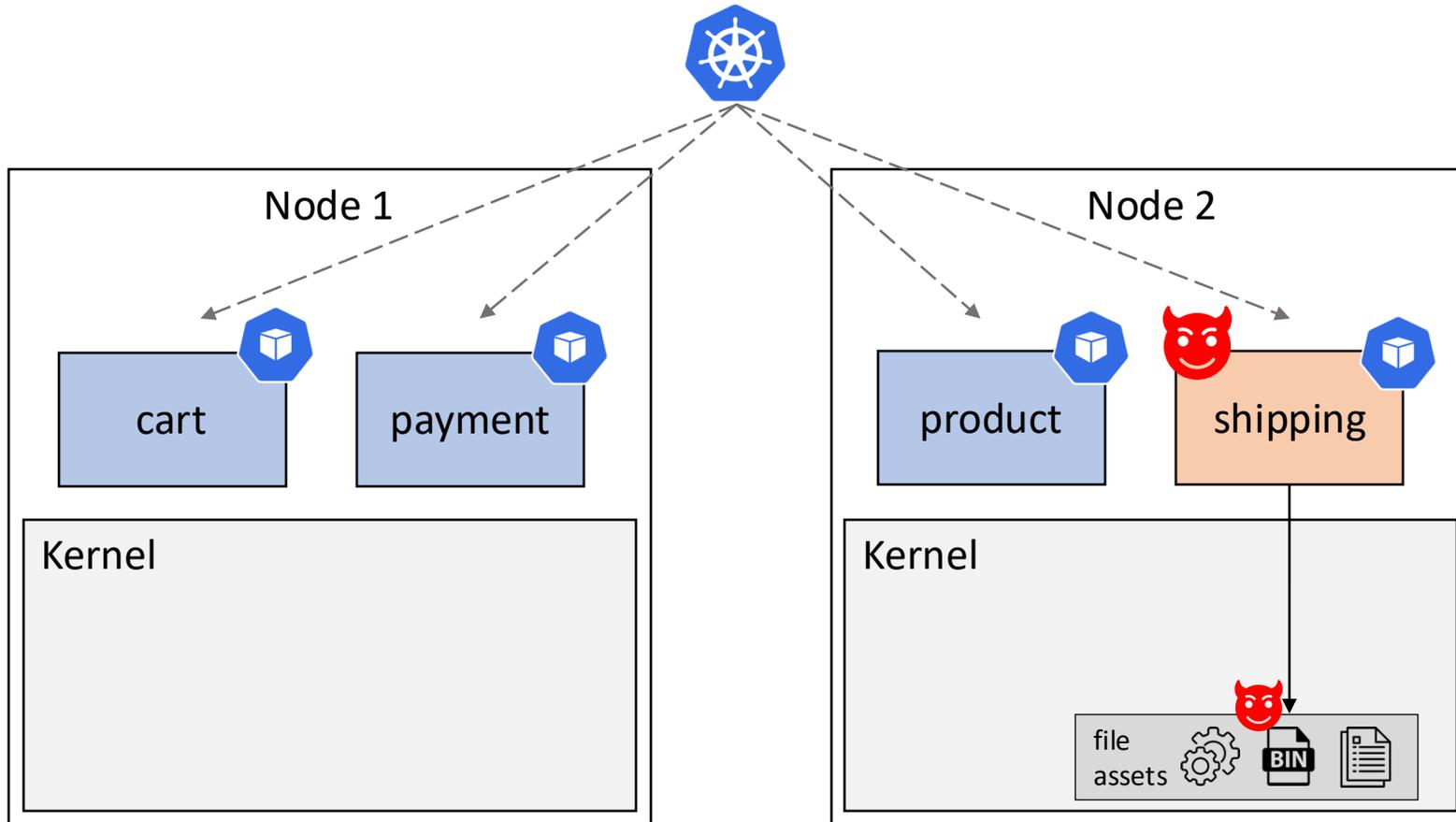
# Container and Kubernetes

- Container Orchestration



# Attack Types & Techniques

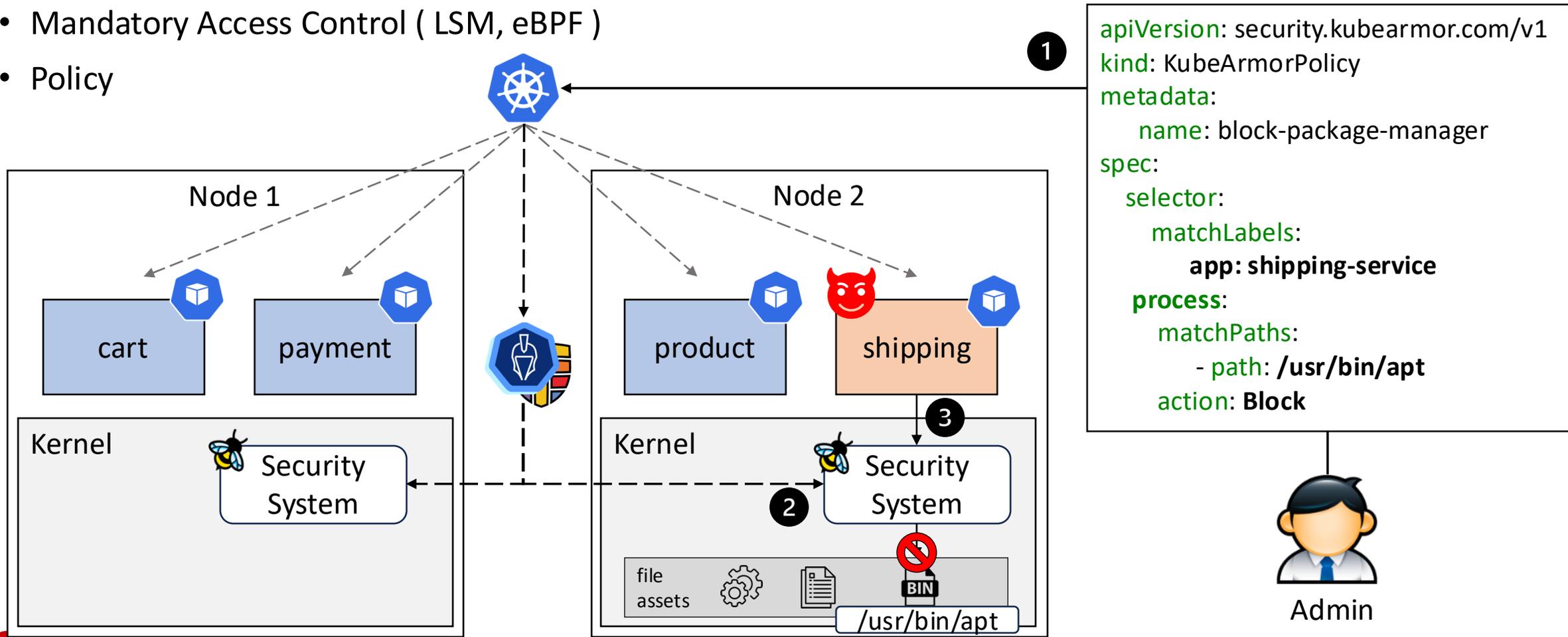
- Compromise the container



Attack Type	Attack Techs
Disclose credential Information	filesystem access
	file name access
Execute Arbitrary Code	Command and Scripting Interpreter

# Cloud Native Runtime Security Enforcement System

- KubeArmor, Tetragon
- Mandatory Access Control ( LSM, eBPF )
- Policy



# Problem Statement

---

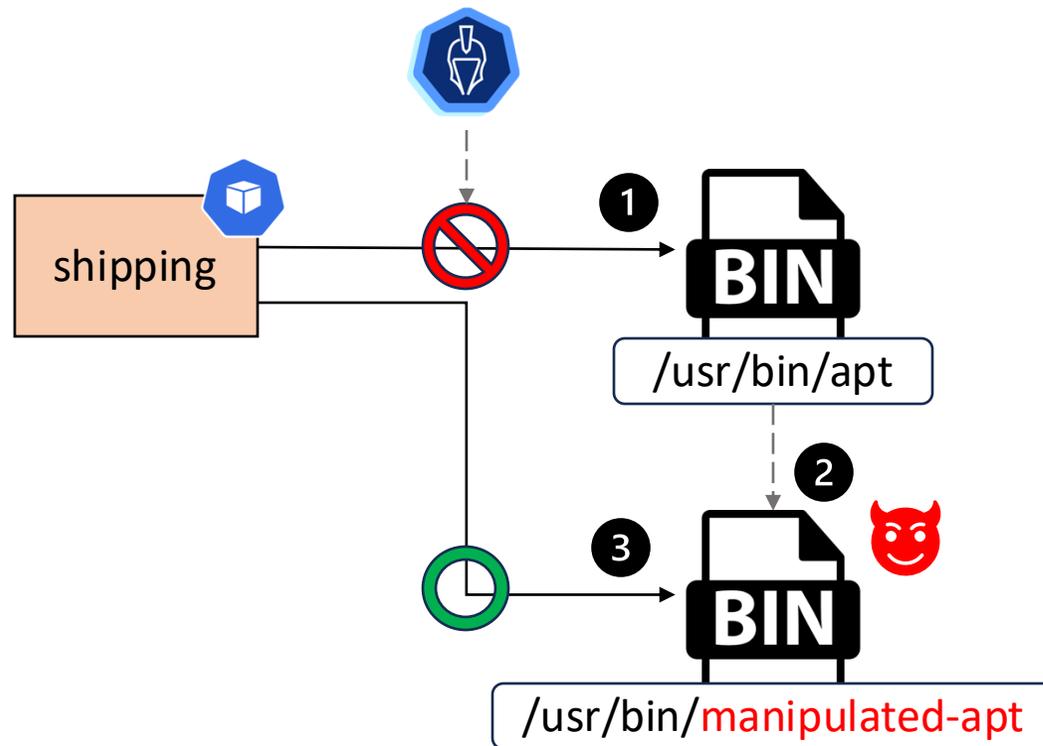
# Problem Statement - Overview

---

- Bypass of the security policy
  - Path Manipulation
  - Misconfiguration

# Bypass of security policy (1) – Path Manipulation

- Path Manipulation



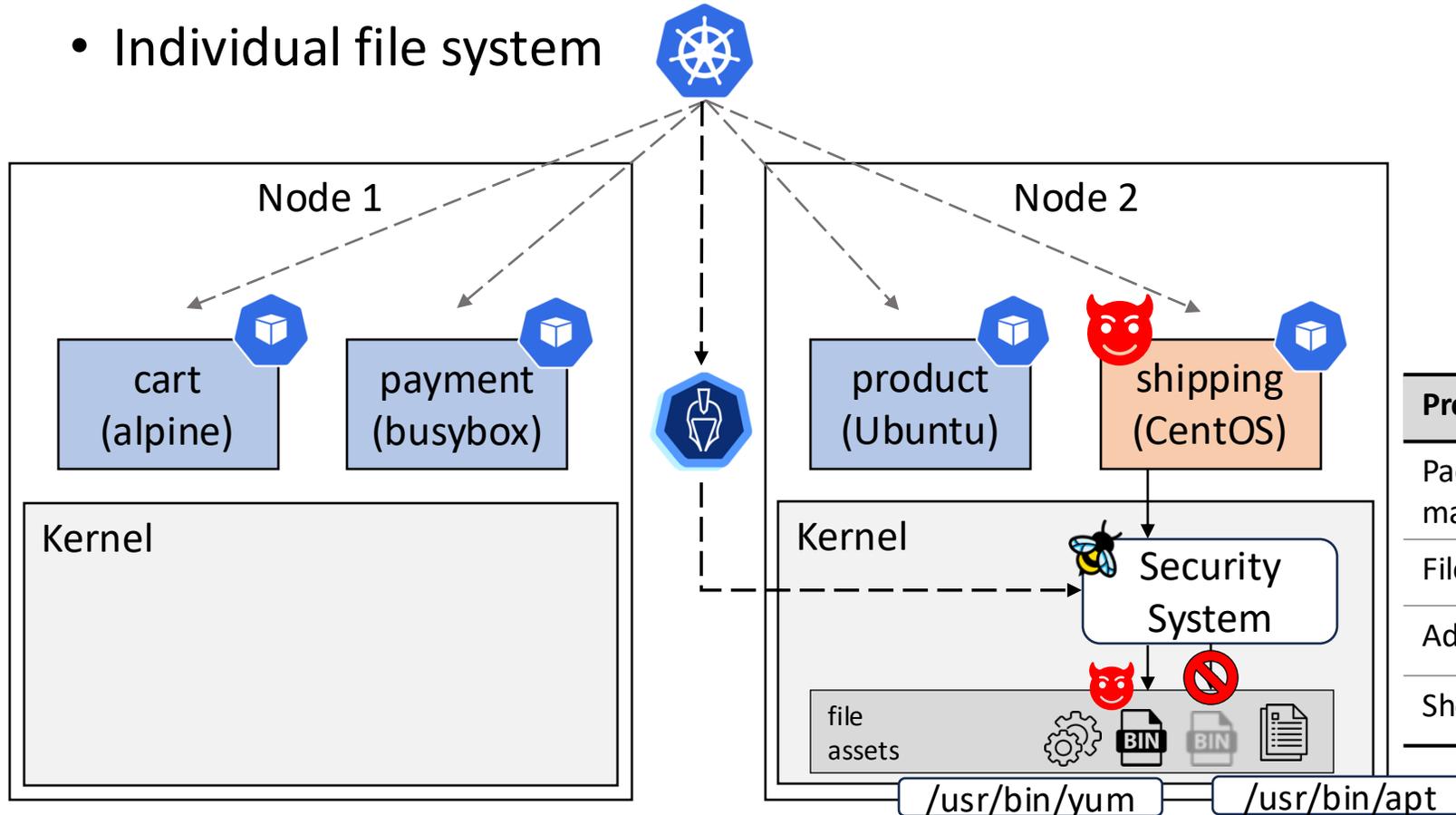
(1) Path Manipulation

Manipulation Type	Command (linux)
Path Modification	mv
File Duplication	cp
Hardlink	ln

(2) Manipulation Type

# Bypass of security policy (2) – Misconfiguration

- Misconfiguration
- Individual file system



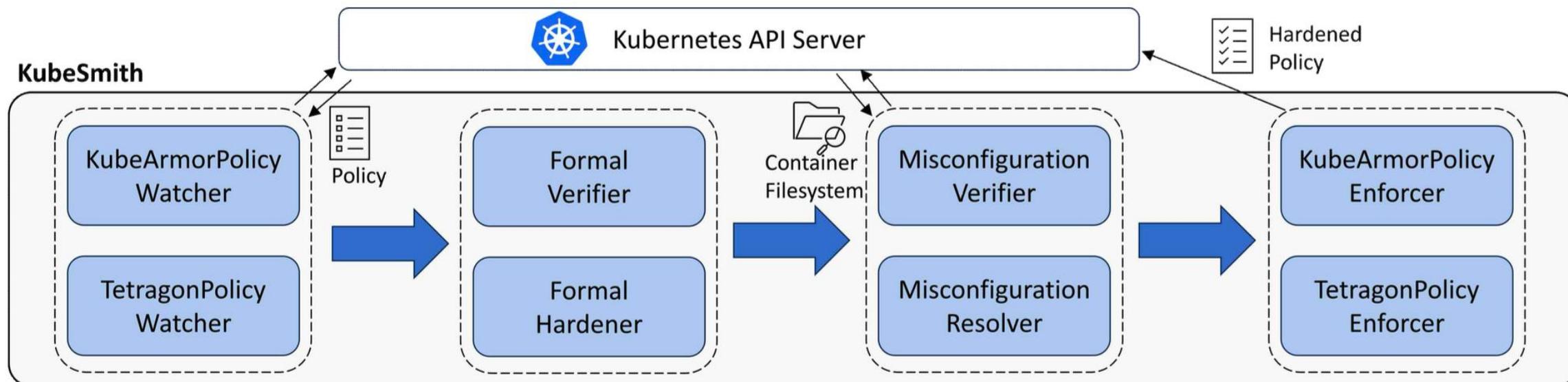
Program Type	Ubuntu	CentOS
Package manager	<code>/usr/bin/apt</code> <code>/usr/bin/apt-get</code>	<code>/usr/bin/yum</code> <code>/usr/bin/rpm</code>
File Editor (vim)	<code>/usr/bin/vim.basic</code>	<code>/usr/bin/vi</code>
Add User	<code>/usr/sbin/useradd</code>	<code>/usr/sbin/adduser</code>
Shell	<code>/usr/bin/dash</code>	<code>/usr/bin/bash</code>

# KubeSmith Design

---

# KubeSmith Overview

- Verification and Hardening **Policy**
- Watch → Verify & Harden → Enforce



# Path Manipulation – Formal Verifier & Hardener

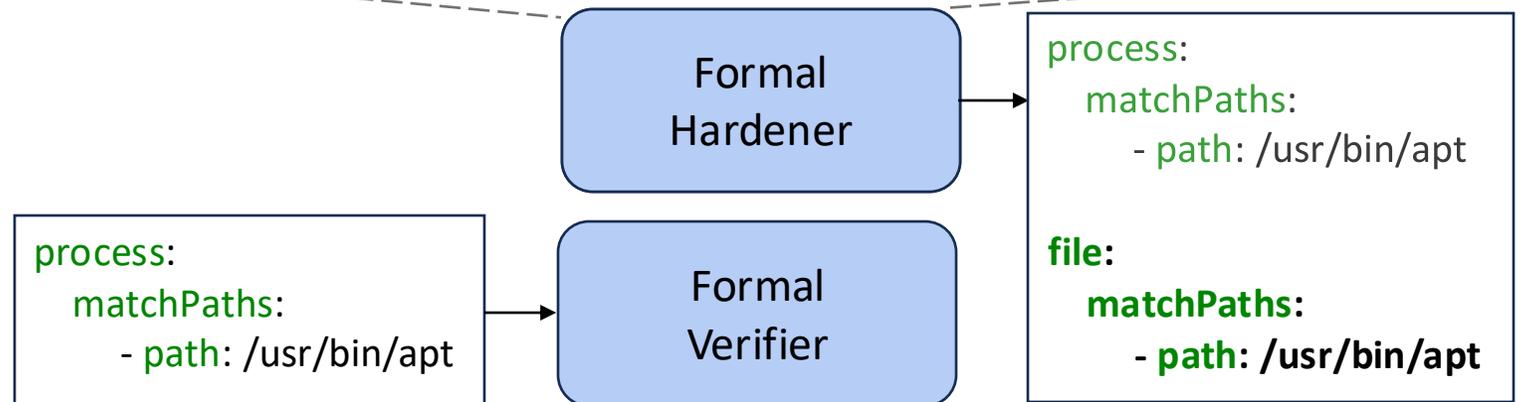
Identification of potential **path manipulation** in policies and hardening through **policy modification**

(1) Manipulation type – Response Mapping

Manipulation type	LSM Hook	KubeArmor rule	Tetragon rule
Path Modification	path_rename	File	-
File Duplication	file_open		hook: "file_open"
Hardlink	path_link		-

(2) Identify security policy

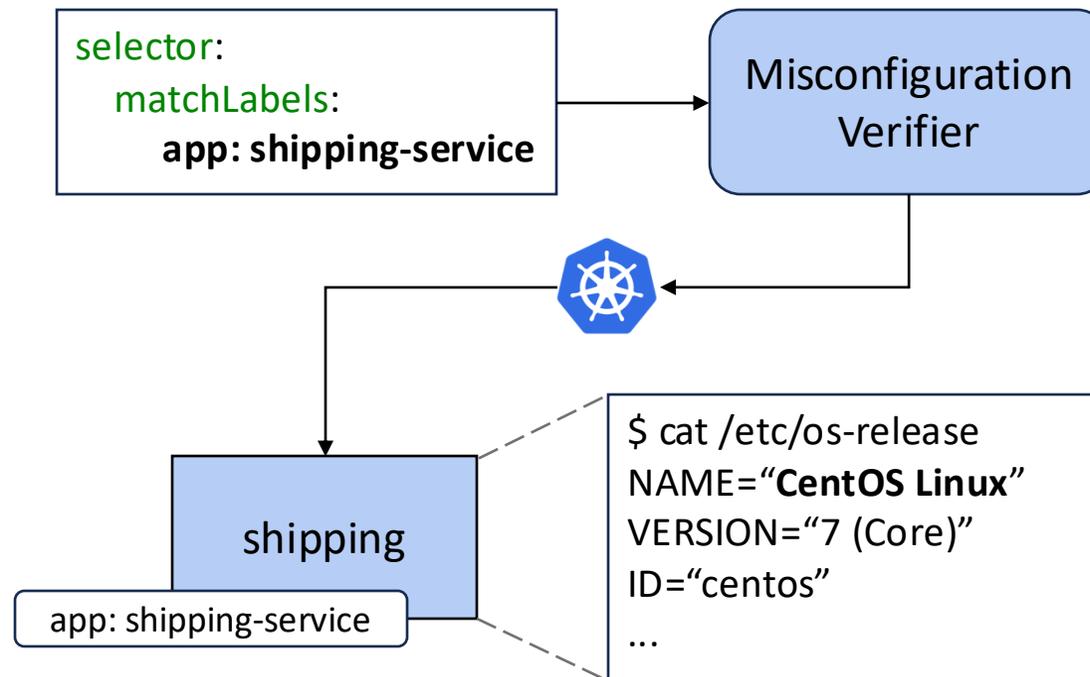
(3) Harden security policy



# Misconfiguration – Misconfiguration Verifier & Resolver

Identify **misconfigurations** between policies and the actual container environment, and **resolve** them

## (1) Extraction container filesystem information



# Misconfiguration – Misconfiguration Verifier & Resolver

Identify **misconfigurations** between policies and the actual container environment, and **resolve** them

## (2) Matching the container – program

```
process:  
  matchPaths:  
    - path: /usr/bin/apt
```

```
NAME="CentOS Linux"  
VERSION="7 (Core)"  
ID="centos"  
...
```



```
yum  
dnf
```

You are a Linux system binary mapper.  
Your task is to identify equivalent command-line binaries across Linux distributions.

binary: **{binary name}**

Your job: Search within the **{image}** and list the binary (or binaries) that serve the same purpose as the input binary.

Constraints:

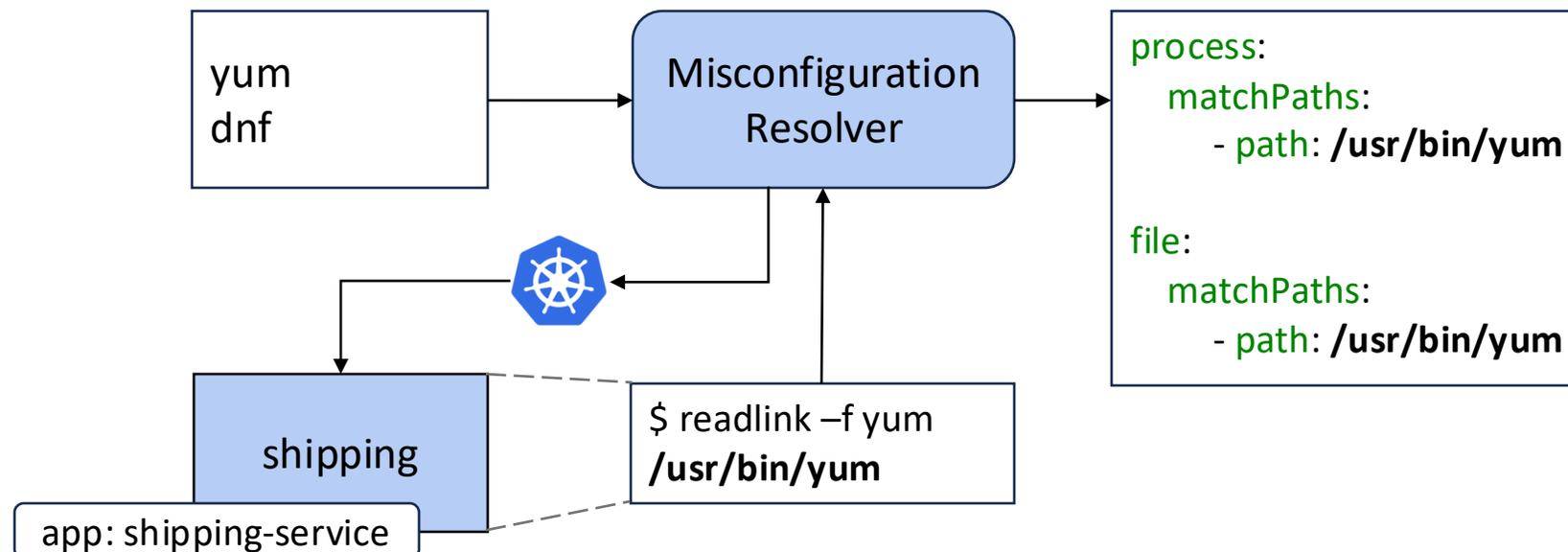
- Output only actual binary names as they appear
- No explanation, no code blocks, no formatting
- ...



# Misconfiguration – Misconfiguration Verifier & Resolver

Identify **misconfigurations** between policies and the actual container environment, and **resolve** them

(3) Path correction based on filesystem lookup



# Evaluation

---

# Evaluation – Formal Verifier & Hardener

- Evaluate whether all three types of manipulable paths can be blocked
  - KubeArmor – All
  - Tetragon – only file duplication

```

root@target-pod:/# apt update
bash: /usr/bin/apt: Permission denied

root@target-pod:/# mv /usr/bin/apt /usr/bin/manipulated-apt
root@target-pod:/# manipulated-apt update
Get:1 http://deb.debian.org/debian bookworm InRelease [...]
Get:2 http://deb.debian.org/debian bookworm-updates InRelease [...]
...
    
```

without – (A)

```

root@target-pod:/# apt update
bash: /usr/bin/apt: Permission denied

root@target-pod:/# mv /usr/bin/apt /usr/bin/manipulated-apt
mv: cannot move '/usr/bin/apt' to '/usr/bin/manipulated-apt':
Permission denied
    
```

with – (B)

(1) Use case

Manipulation type	LSM Hook	KubeArmor rule	Tetragon rule
Path Modification	path_rename	File	-
File Duplication	file_open		hook: "file_open"
Hardlink	path_link		-

(2) Coverage

# Evaluation – Misconfiguration Verifier & Resolver

- 5 example policies from the KubeArmor repository
- 3 containers, each using a different image
- Policy Example
  - /usr/bin/apt, /usr/bin/apt-get

Program Type	Ubuntu	CentOS	Busybox
Package Manager	/usr/bin/apt /usr/bin/apt-get	/usr/bin/yum	-
File Editor	/usr/bin/vim.basic	/usr/bin/vi	/bin/vi
Add User	/usr/sbin/useradd	/usr/sbin/adduser	/bin/adduser
Shell	/usr/bin/dash	/usr/bin/bash	/bin/sh

(1) Program name

Program Type	Ubuntu		CentOS		Busybox	
	w/o	w/	w/o	w/	w/o	w/
Package Manager (apt)	2/2	2/2	<b>0/1</b>	<b>1/1</b>	-	-
File Editor (vim)	<b>0/1</b>	<b>1/1</b>	<b>0/1</b>	<b>1/1</b>	1/1	1/1
Add User (adduser)	1/1	1/1	1/1	1/1	<b>0/1</b>	<b>1/1</b>
Shell (sh)	<b>0/1</b>	<b>1/1</b>	1/1	1/1	1/1	1/1

(2) Coverage

# Conclusion

---

# Conclusion and Future Work

---

## Conclusion

- Bypass of security policy
- KubeSmith

## Future Work

- White list policy generation