Kadapter:
Kata 컨테이너를 위한 보안 집행 프레임워크

Chihyeon Jo, Her Jin, Seungsoo Lee*

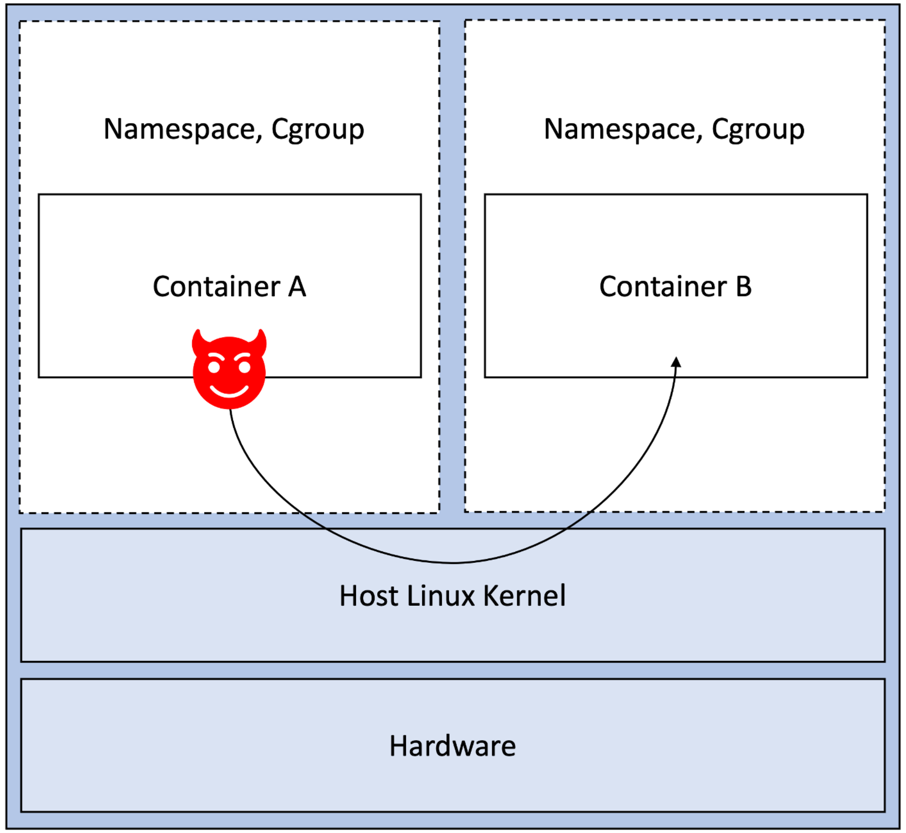
Department of Computer Science & Engineering

Incheon National University

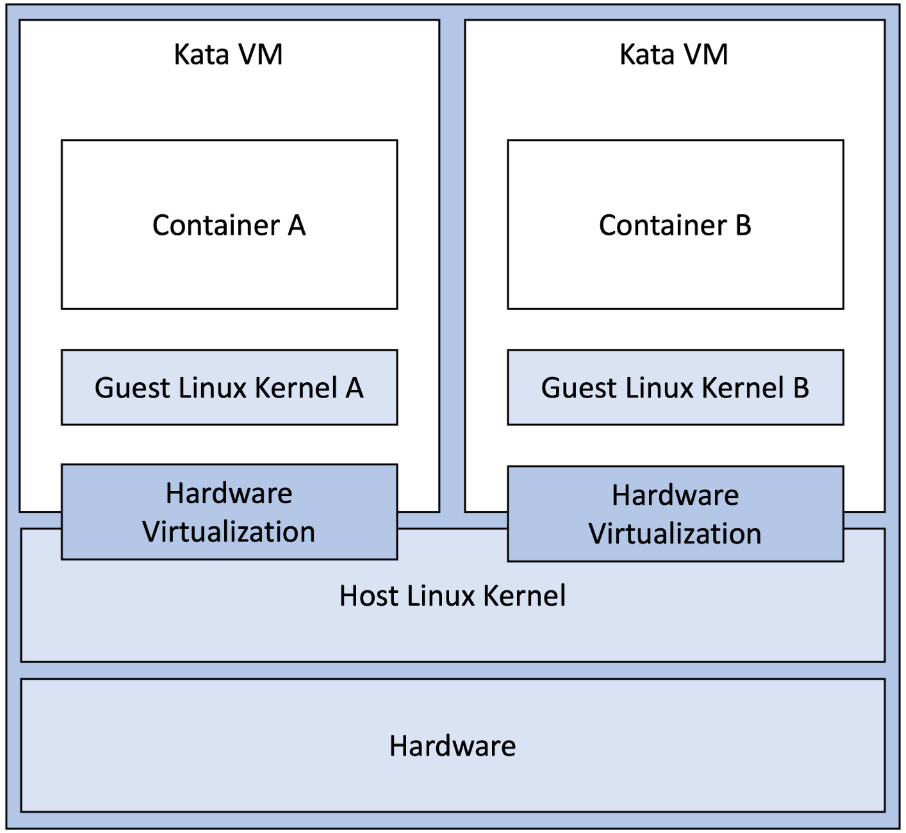
Background: Kata containers

- Dual isolation of containers as lightweight virtual machines

Traditional Container

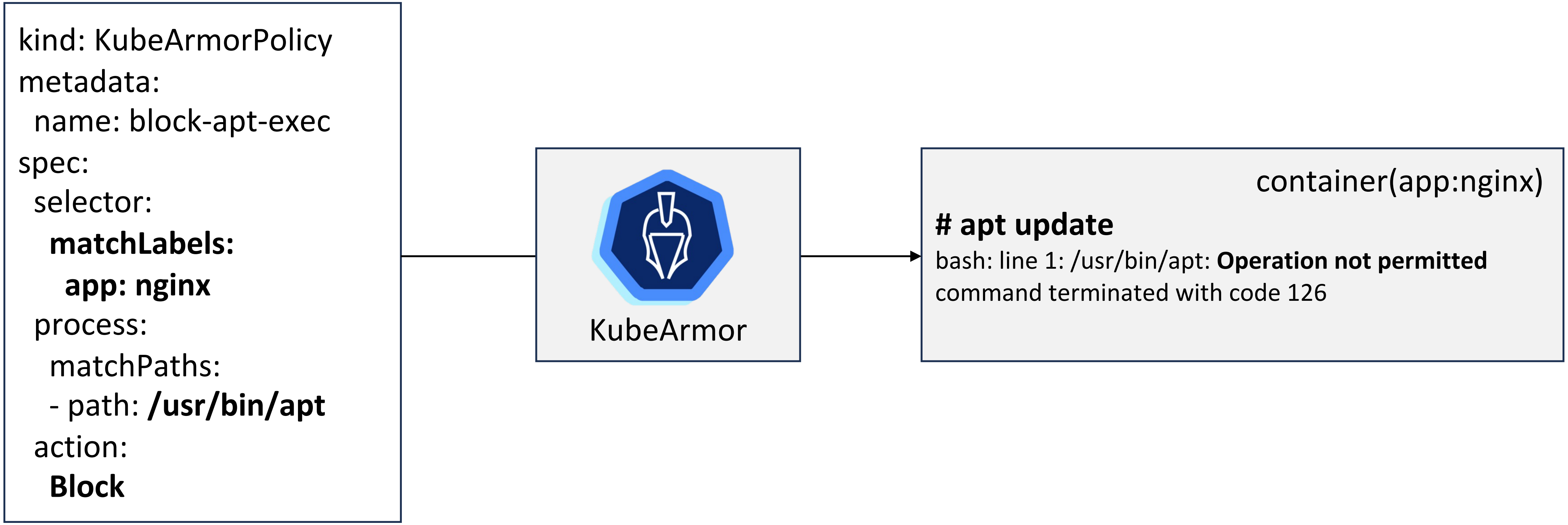


Kata container



Background: KubeArmor

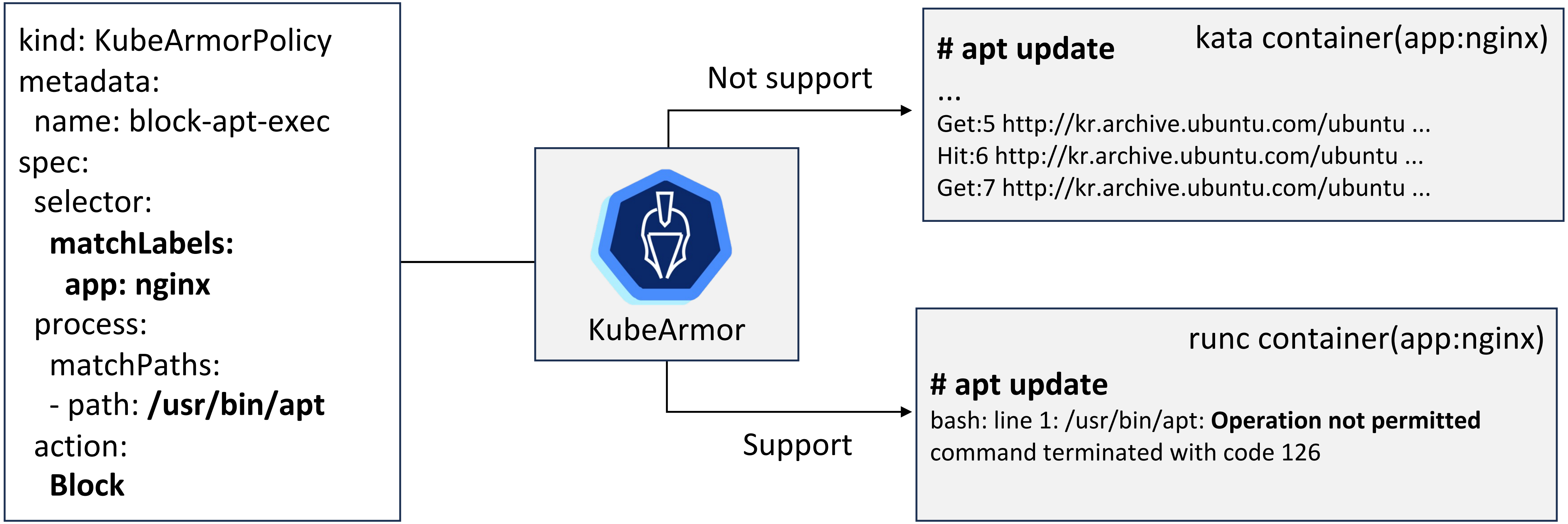
- Restrict the behavior of pods, containers, and nodes at the system level
- Runtime Security is an important one since most of the attacks are manifested in Runtime.



```
kind: KubeArmorPolicy
metadata:
  name: block-apt-exec
spec:
  selector:
    matchLabels:
      app: nginx
  process:
    matchPaths:
    - path: /usr/bin/apt
  action:
    Block
```

```
# apt update
bash: line 1: /usr/bin/apt: Operation not permitted
command terminated with code 126
```

Motivation



```
kind: KubeArmorPolicy
metadata:
  name: block-apt-exec
spec:
  selector:
    matchLabels:
      app: nginx
  process:
    matchPaths:
    - path: /usr/bin/apt
  action:
    Block
```

```
# apt update
...
Get:5 http://kr.archive.ubuntu.com/ubuntu ...
Hit:6 http://kr.archive.ubuntu.com/ubuntu ...
Get:7 http://kr.archive.ubuntu.com/ubuntu ...
```

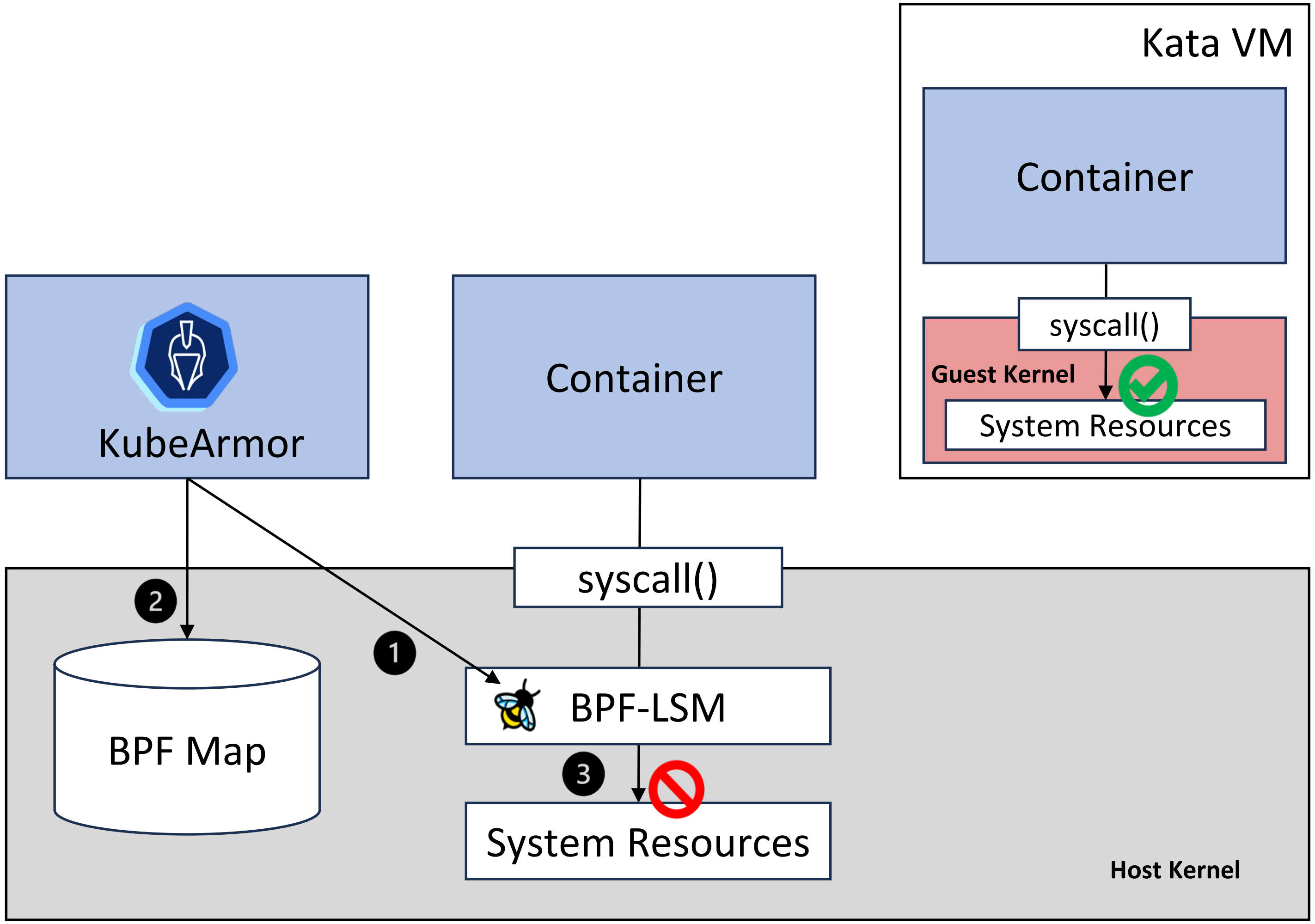
```
# apt update
bash: line 1: /usr/bin/apt: Operation not permitted
command terminated with code 126
```

There is no **security enforcement framework** that supports the **Kata container.**

Analysis of KubeArmor

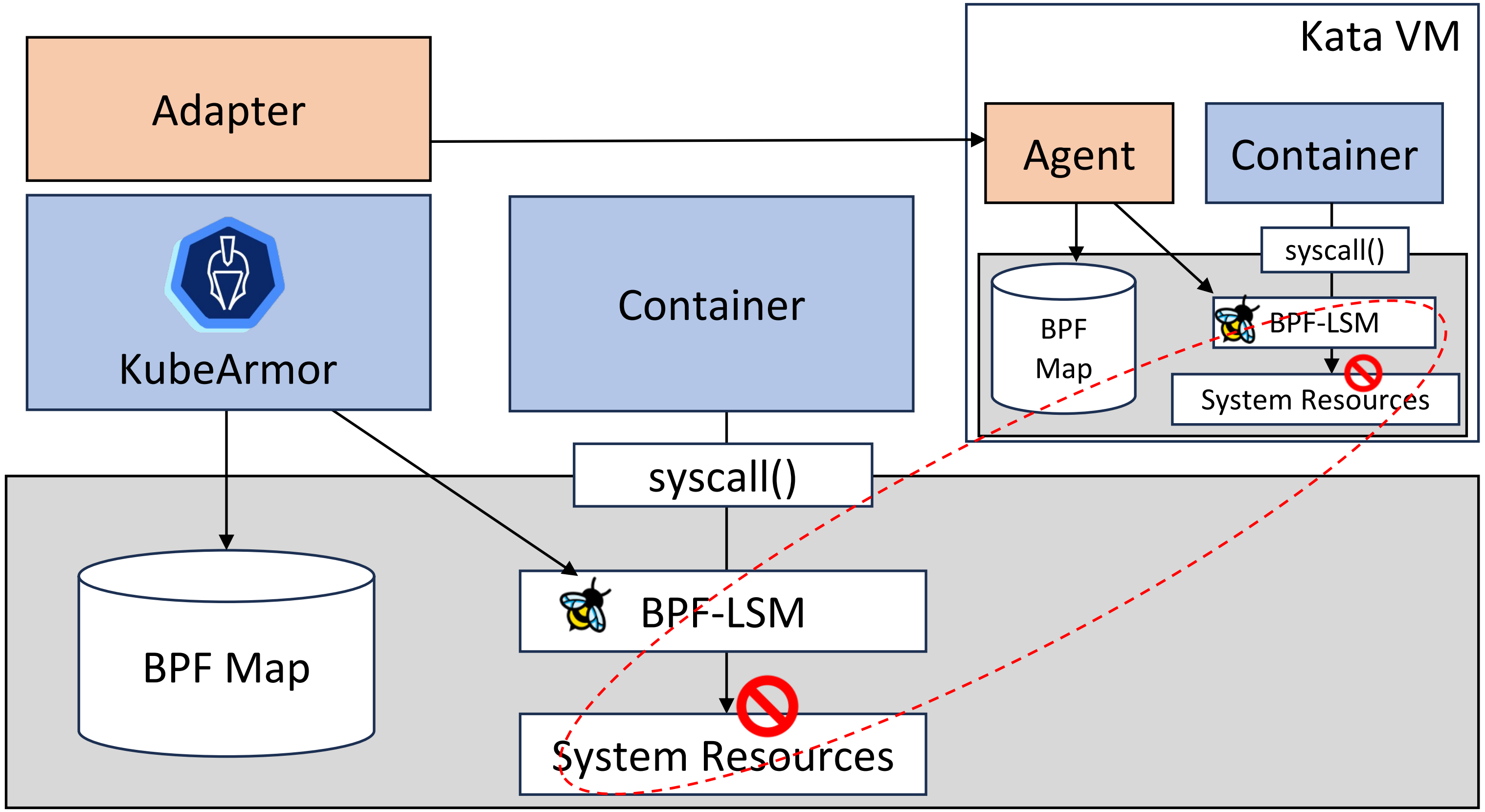
1. KubeArmor attaches BPF program to LSM Hook
2. It detects policy-related resources and store them in the BPF map
3. BPF-LSM program queries the BPF map to block actions that violate policies

However, Kata container is isolated by **individual guest kernel**



Kadapter Design: Approach

1. The same BPF-LSM programs and BPF maps are loaded into the guest kernel of the Kata virtual machine as the host kernel.
2. When a policy related to Kata containers is detected, it is sent inside the Kata virtual machine for enforcement



Kadapter Design: Component

Kadapter Componenet

Kadapter

Detect and send Kata container related policies on Master node

Kadapter-Agent

Manage BPF-LSM Program and enforce policies inside the Kata virtual machine

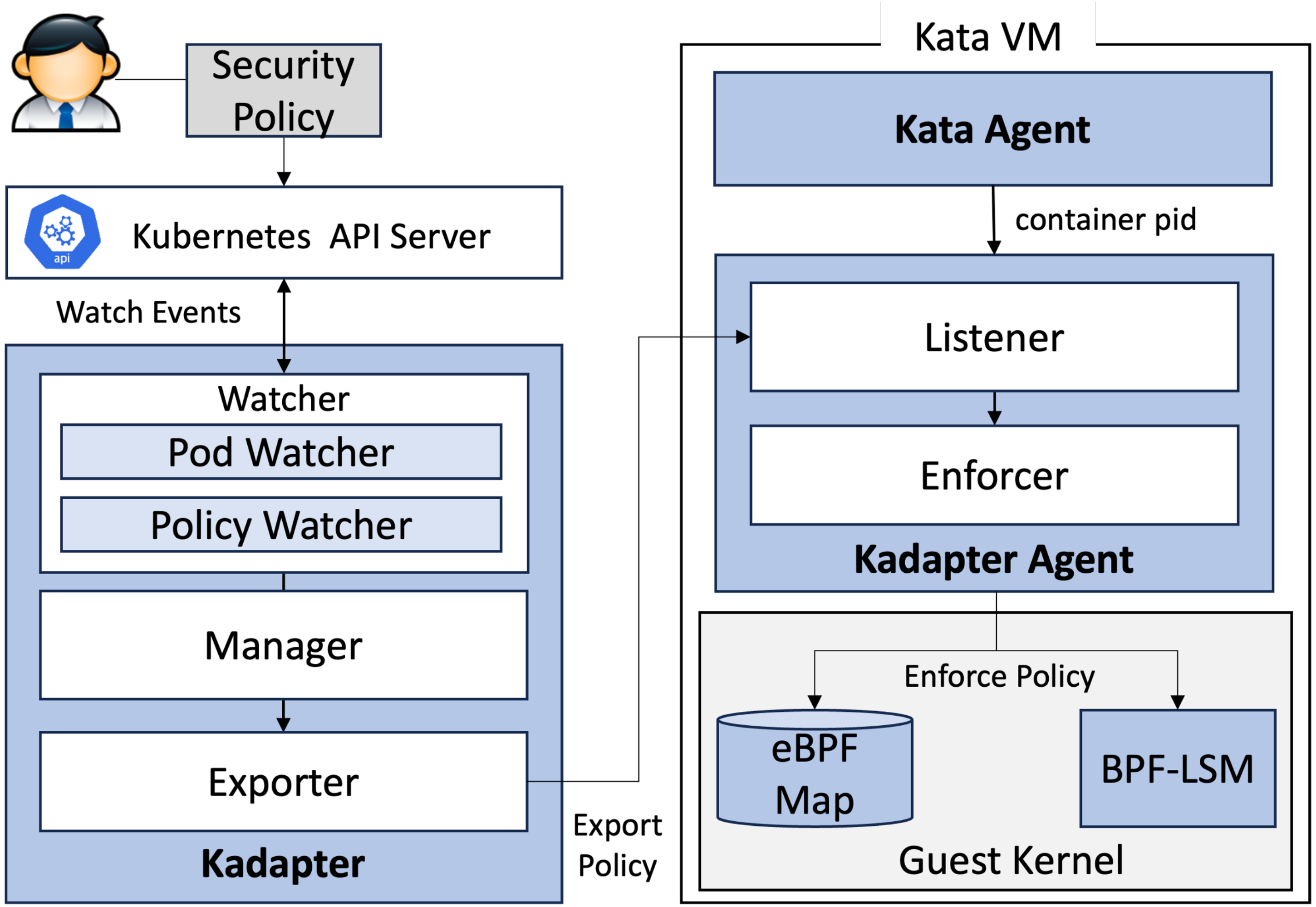
Modified Component

Guest Kernel

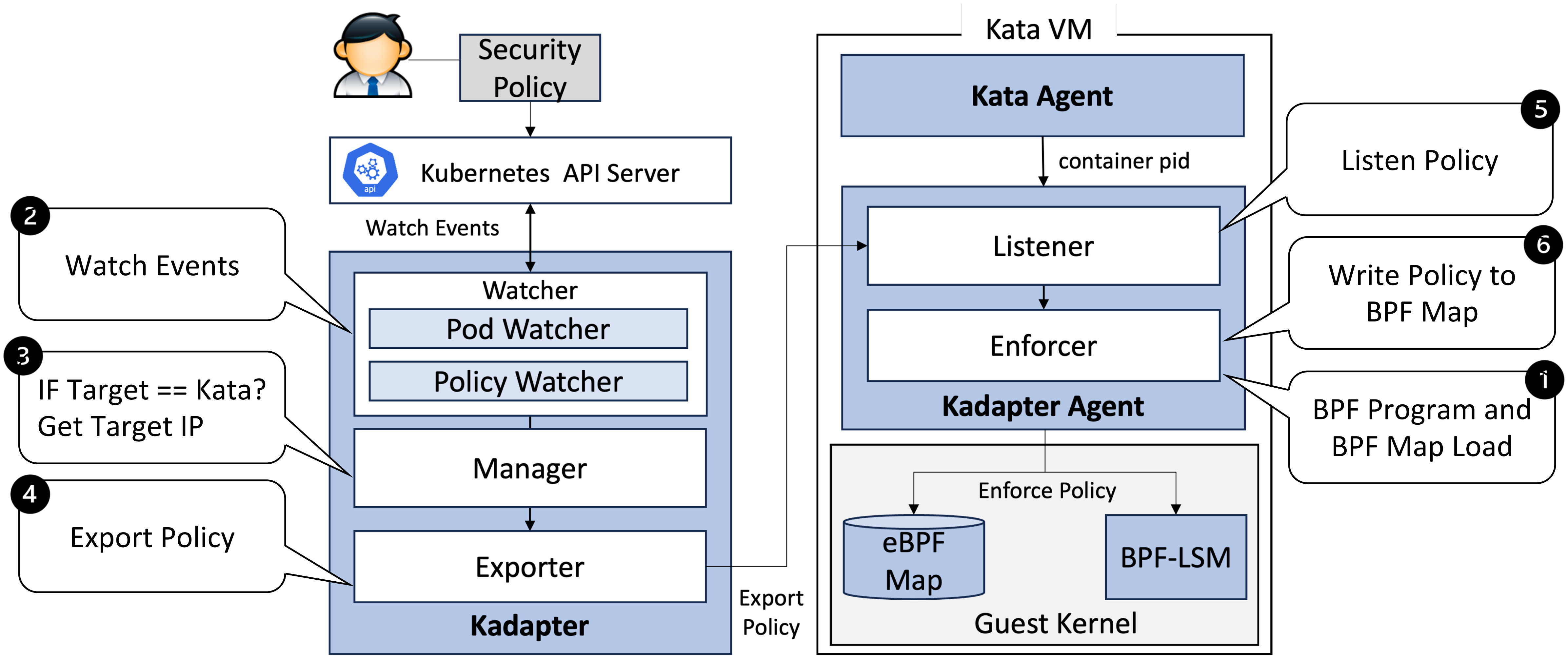
Rebuilded to support BPF Program

Kata Agent

Modified to obtain the PID of the container inside the Kata virtual machine

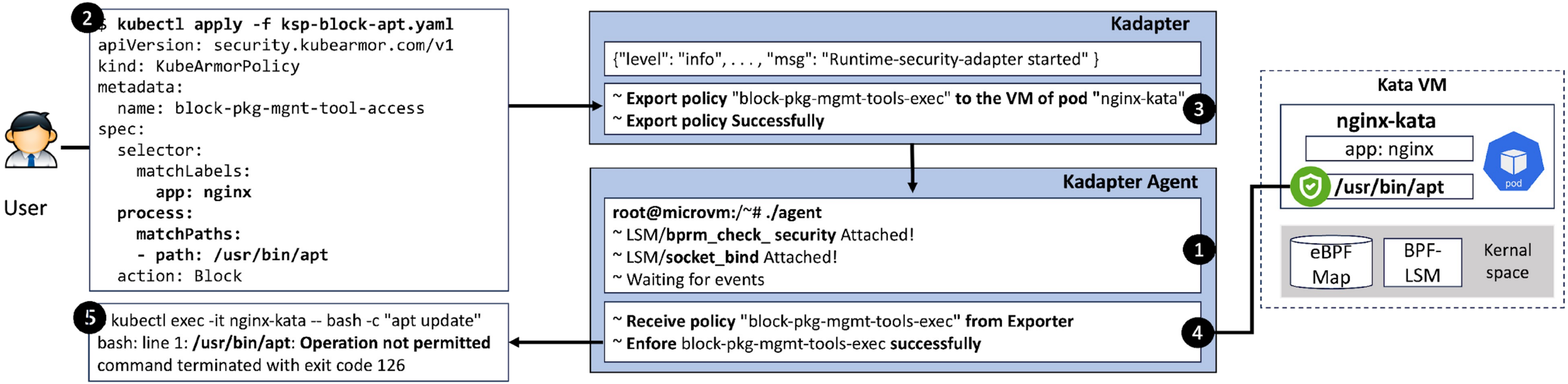


Kadapter Design: Workflow



Evaluation: Use case(1)

Test the feasibility with scenarios that limit process execution for containers with "app=nginx" labels



```
$ kubectl apply -f ksp-block-apt.yaml
apiVersion: security.kubearmor.com/v1
kind: KubeArmorPolicy
metadata:
  name: block-pkg-mgnt-tool-access
spec:
  selector:
    matchLabels:
      app: nginx
  process:
    matchPaths:
    - path: /usr/bin/apt
  action: Block
```

```
{"level": "info", . . . , "msg": "Runtime-security-adapter started" }
```

```
~ Export policy "block-pkg-mgmt-tools-exec" to the VM of pod "nginx-kata"
~ Export policy Successfully
```

```
root@microvm:/~# ./agent
~ LSM/bprm_check_ security Attached!
~ LSM/socket_bind Attached!
~ Waiting for events
```

```
~ Receive policy "block-pkg-mgmt-tools-exec" from Exporter
~ Enfore block-pkg-mgmt-tools-exec successfully
```

```
kubectl exec -it nginx-kata -- bash -c "apt update"
bash: line 1: /usr/bin/apt: Operation not permitted
command terminated with exit code 126
```

Evaluation: Use case(2)

Verified that policies for network and file operations are enforced correctly.

Network

File

Policy

```
kind: KubeArmorPolicy
...
spec:
  selector:
    matchLabels:
      app: nginx
  network:
    matchProtocols:
    - protocol: icmp
  action:
    Block
```

```
kind: KubeArmorPolicy
...
spec:
  selector:
    matchLabels:
      app: nginx
  file:
    matchDirectories:
    - dir: /run/secrets/kubernetes.io/serviceaccount/
action:
    Block
```

Result

```
nginx-kata# ping 8.8.8.8
ping: 1.1.1.1: Address family for hostname not
supported
command terminated with exit code 2
```

```
nginx-kata# cat /run/secrets/kubernetes.io/...
cat: /run/secrets/kubernetes.io/...: Permission denied
```

Conclusion and Future Work

- Design runtime security enforcement system for kata container, and demonstrate that it is feasible through experiments

- Increase policy management convenience by expanding to Kata container without modifying existing KubeArmor policies

- In future work, improve performance by minimizing the overhead that occurs in policy enforcement