

커널 수준에서 완성하는 컨테이너 보안 격리

UbuCon Korea X MiniDebConf Korea 2026

안이수 (Leesoo Ahn)

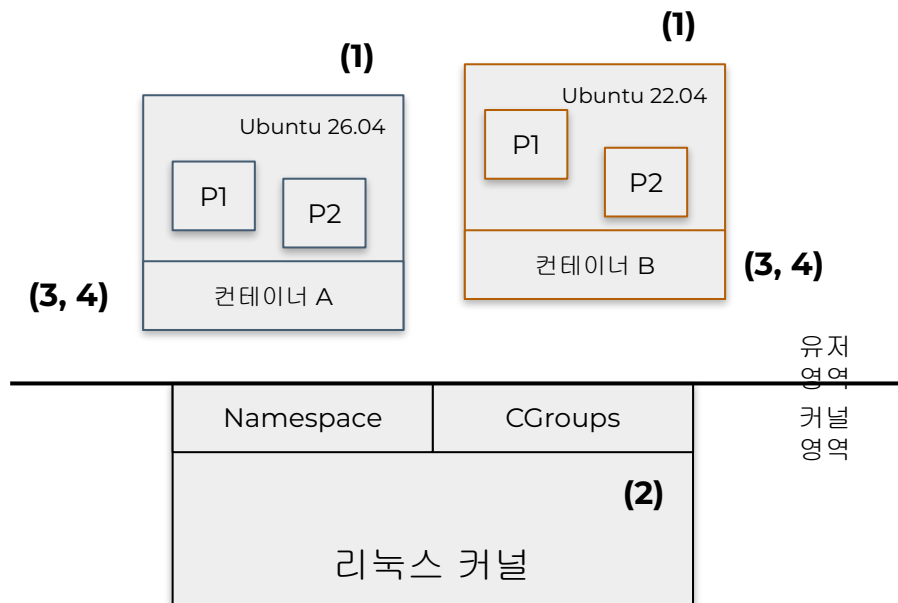


목차



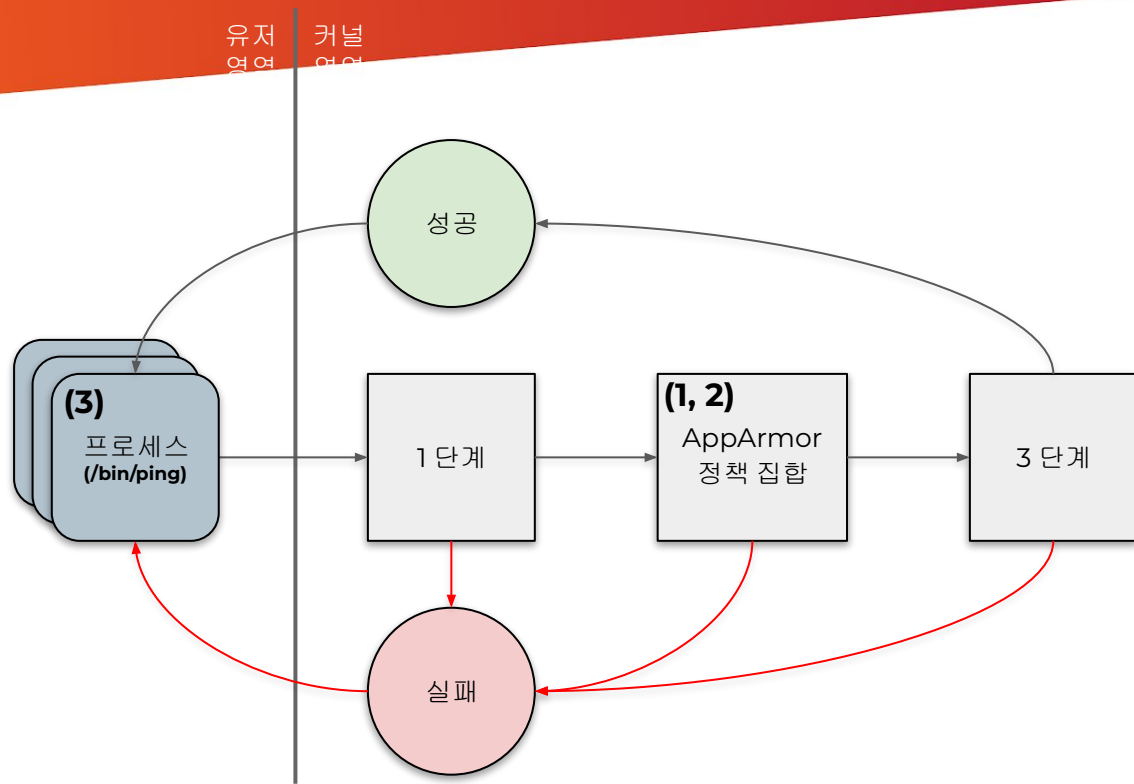
- 리눅스 컨테이너와 AppArmor 개요
- AppArmor Policy 운영 모델 비교
 - Profile, Namespace
- Policy Namespace 시연
- 자동차 S/W 보안을 위한 가상화 기술

리눅스 컨테이너 개요



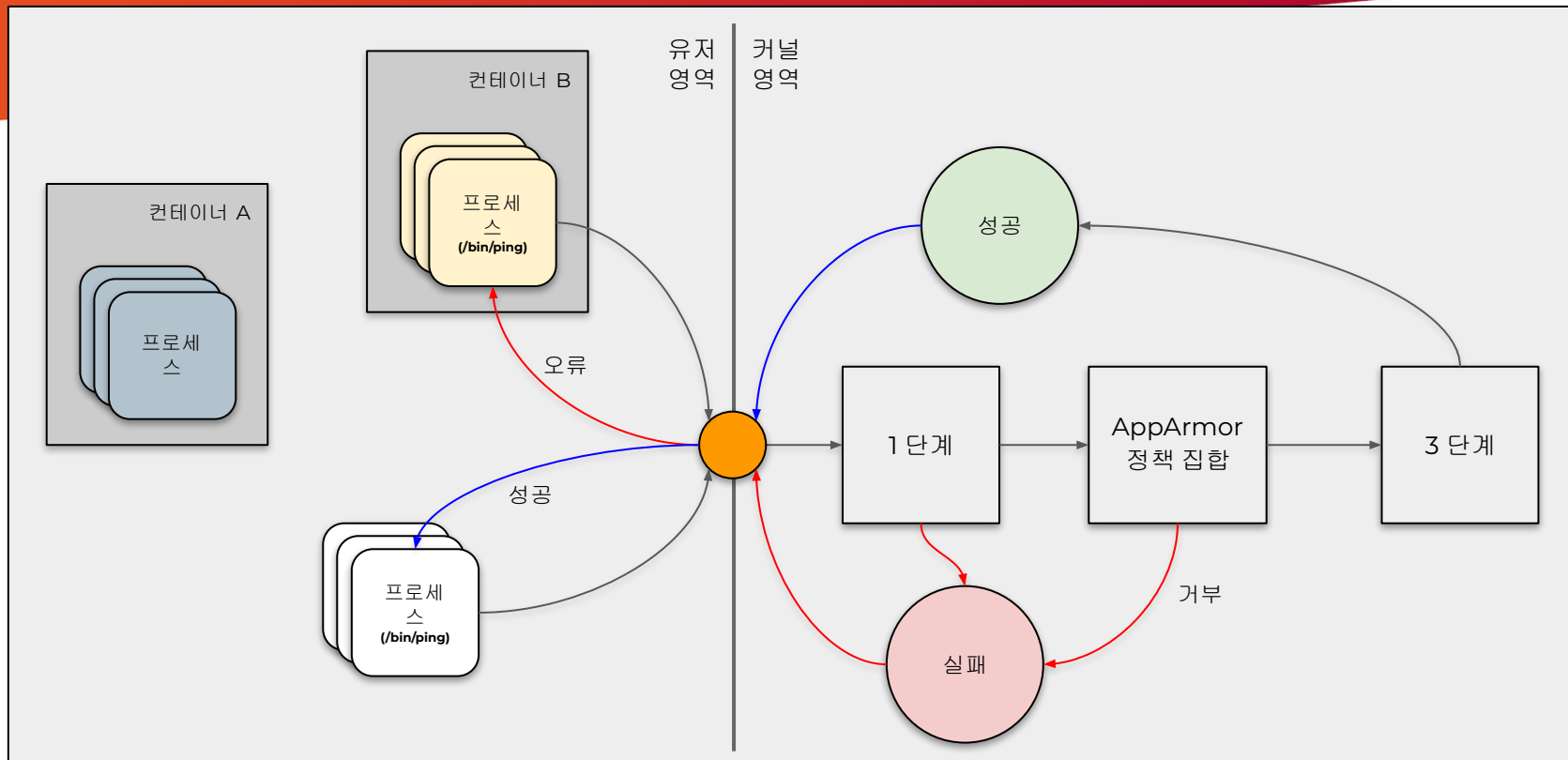
1. 경량 VM
2. 리눅스 커널 공유
3. 프로세스 및 FS 격리
4. 서로 다른 배포판 기능 사용
(예: 시스템 라이브러리)

AppArmor 개요

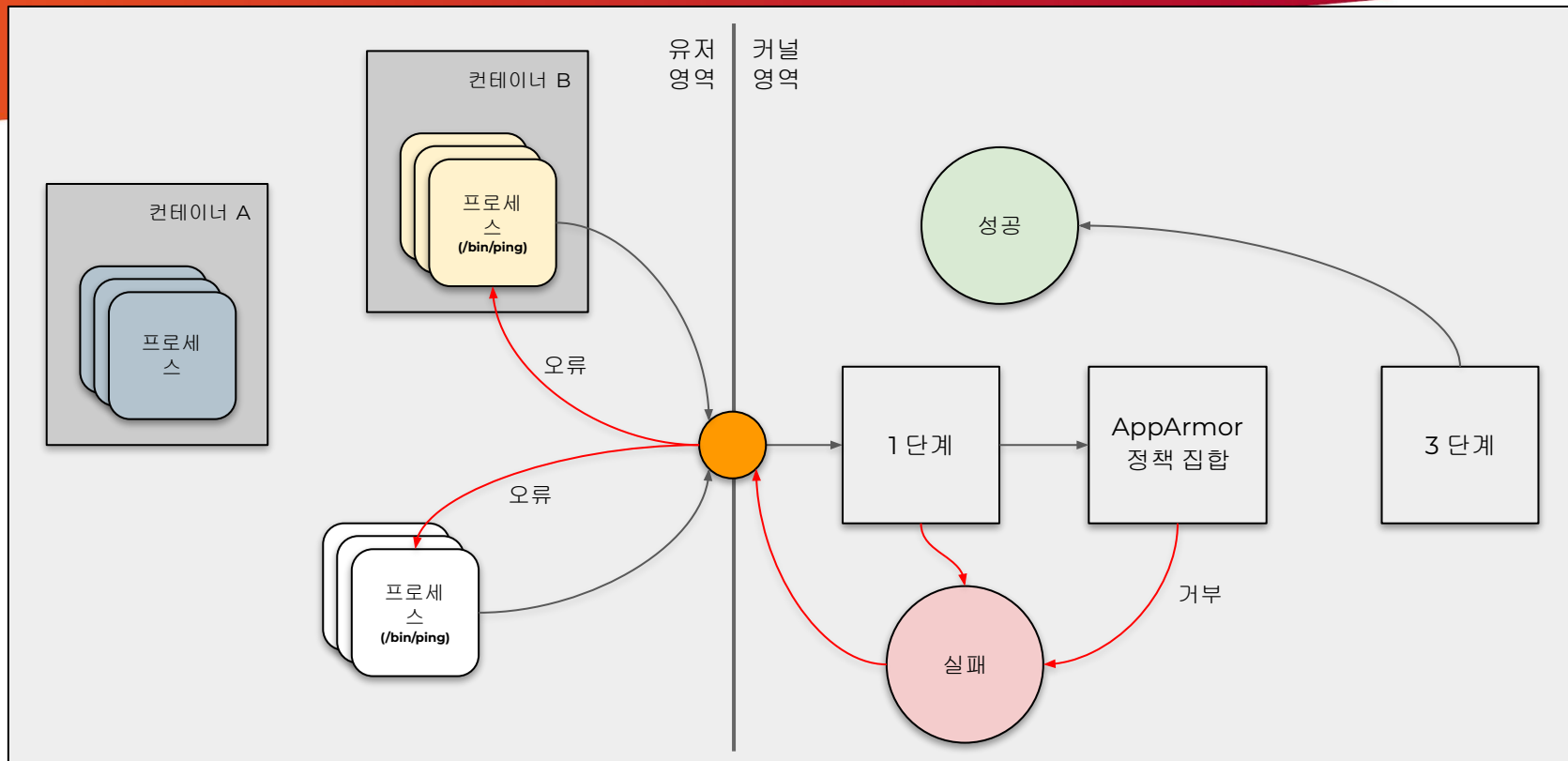


1. 커널 수준 보안 모듈
2. 정책 집합(Policy Set) 기반
3. 경로 기반(Path-based) 제어
(예: /bin/ping)

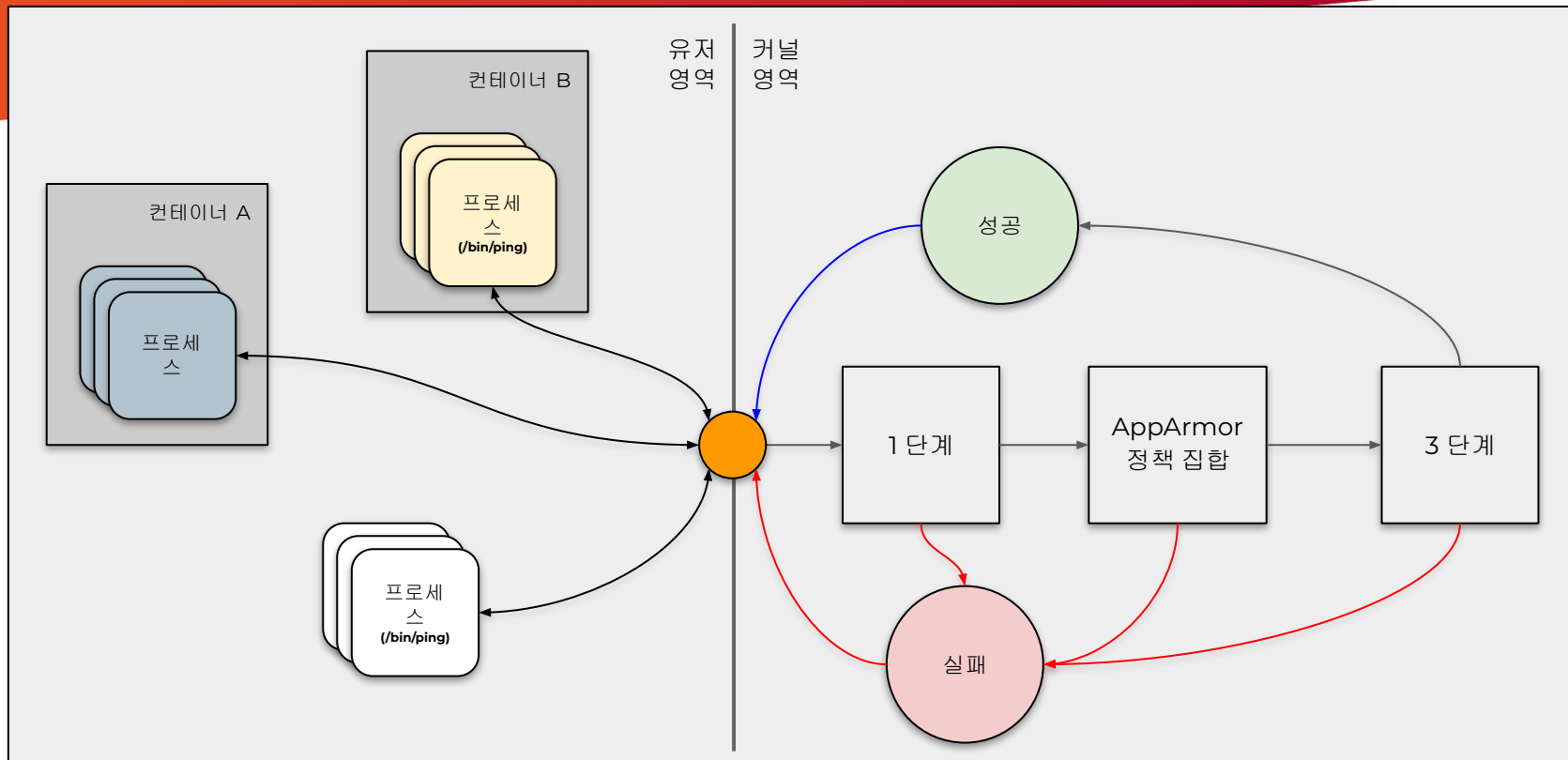
컨테이너 격리 시나리오



컨테이너 격리 시나리오



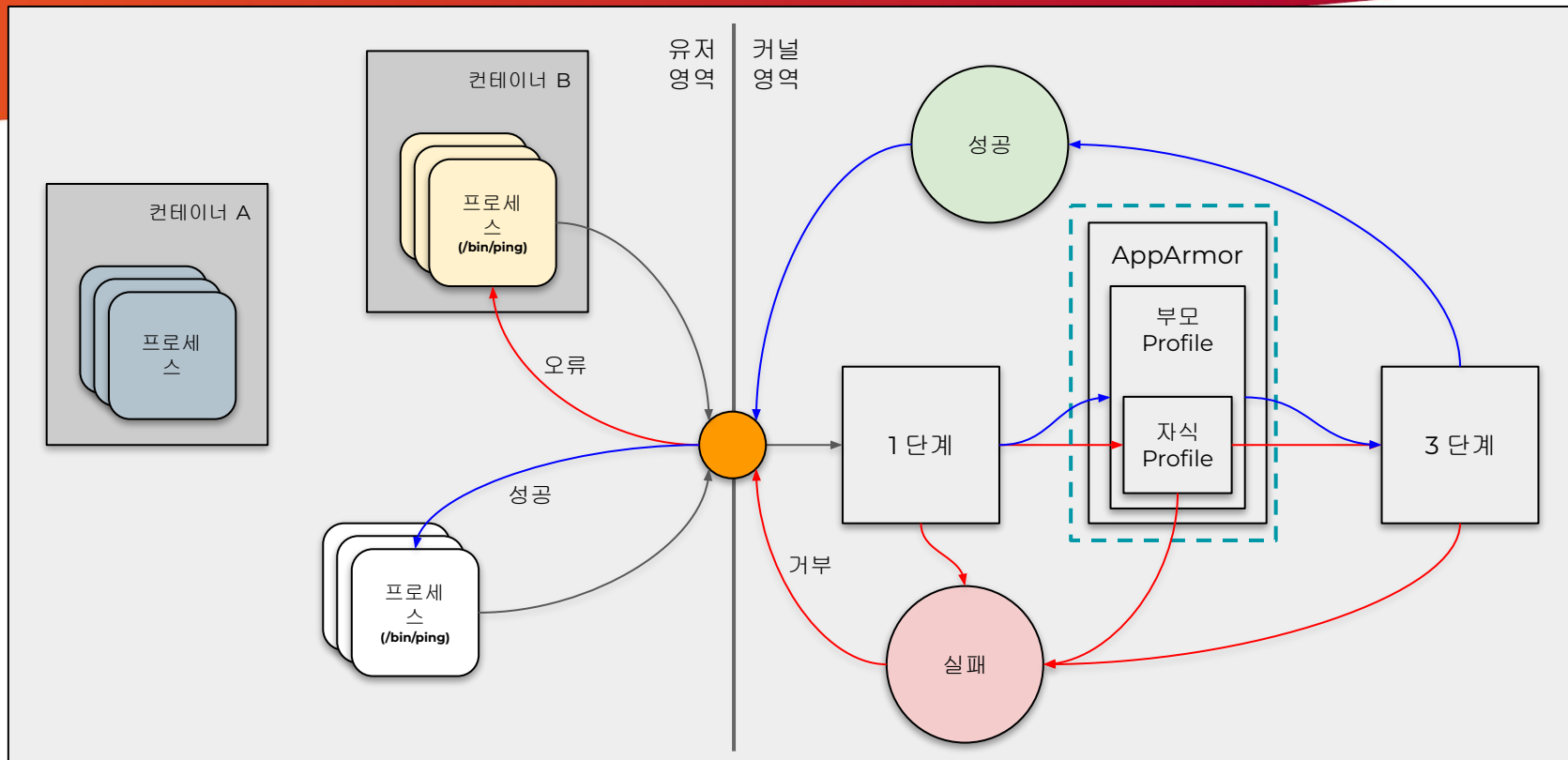
컨테이너 격리 시나리오



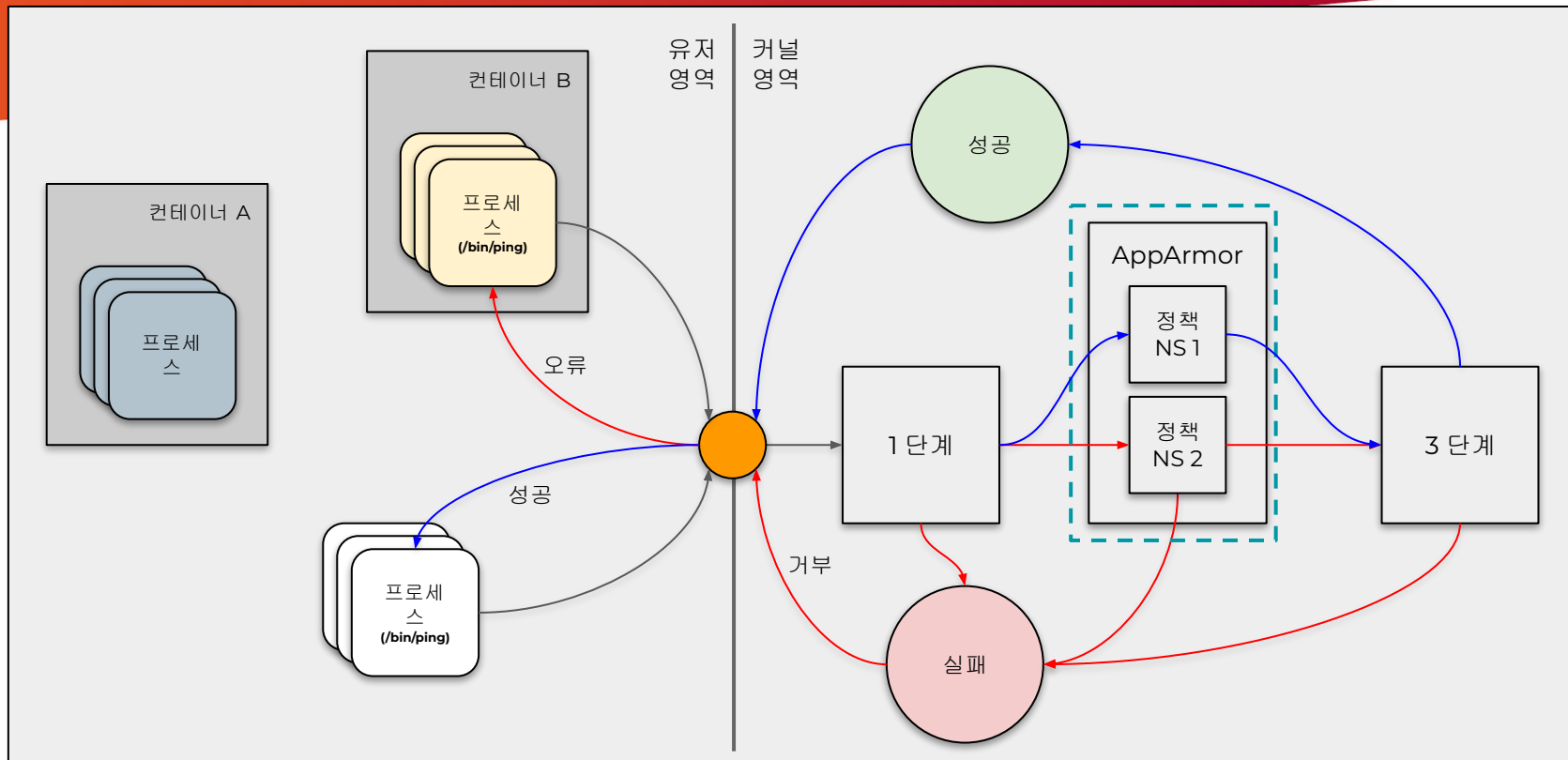
AppArmor Policy 운영 모델



Profile 기반 모델



Policy Namespace 기반 모델



Policy Namespace 시연



Profile vs. Policy Namespace 모델



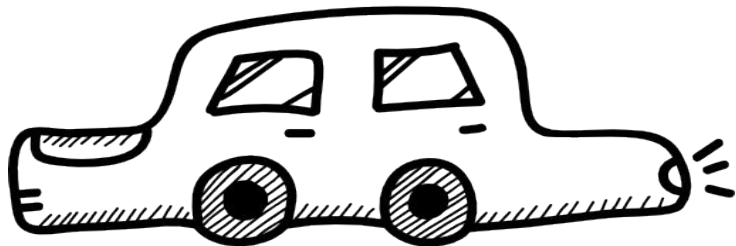
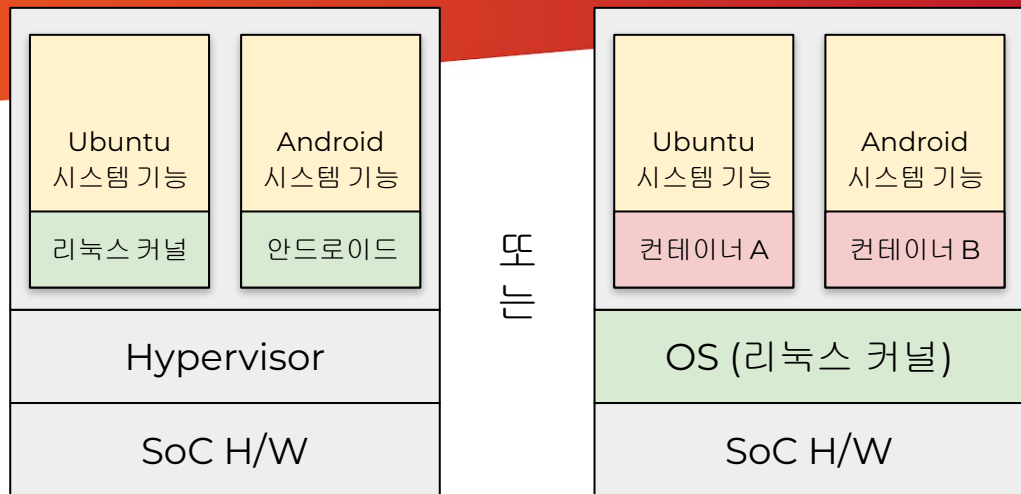
Profile

- Nested Namespace 불가능
- 제한적인 AppArmor 운영
- 경량 Policy 설계
- 학습 곡선이 낮은 편
- 집, 학교에서 운영시 적합

Policy Namespace

- Nested Namespace 가능
- 자유로운 AppArmor 운영
- 복잡한 Policy 설계
- 학습 곡선이 높은 편
- 대규모 Container 운영시 적합

자동차 S/W 보안을 위한 기술 비교



1. 서로 다른 시스템 운영
2. 빠르고 쉬운 통합
3. 자원 제어
(Resource Control)
4. 보안 강화
(Security Hardening)

Q & A



Appendix



```
> cat docker-deny-ping
#include <tunables/global>

profile docker-default flags=(attach_disconnected,mediate_deleted) {
    # deny ping bins
    audit deny /bin/ping rx,
    audit deny /usr/bin/ping rx,

    # allow everything else
    capability,
    network,
    file,
    mount,
    umount,
    pivot_root,
    ptrace,
    signal,
    dbus,
    unix,
}
```

Appendix



- `apparmor_parser -r -W -n "restrict-ns" <profile-file>`
- `docker run -- -it --security-opt apparmor=":restrict-ns://docker-default" ubuntu:24.04 bash`
- `apparmor_parser -R -W -n "restrict-ns" <profile-file>`