사이드카 인젝션 공격: 쿠버네티스 환경에서 Mutating Webhook 악용

Sidecar Injection Attack: Exploiting the Mutating Webhook in Kubernetes

Incheon National University, **Chi Hyun Cho, Seungsoo Lee**

초 록

- 쿠버네티스의 많은 프로젝트는 자동화 및 리소스의 일관성 유지를 위해 Mutating Webhook을 사용.
- Mutating Webhook의 리소스 사양을 변경하는 특징이 공격자에 의해 악용될 경우, 클러스터 전반에 악의적인 영향을 미칠 수 있음.
- Mutating Webhook을 악용한 공격 시나리오인 사이드카 인젝션 공격을 소개하여, 안전한 Mutating Webhook의 중요성 강조.
- 악의적인 Mutating Webhook으로부터 클러스터를 보호하기 위한 방어 기법 소개.

Mutating Webhook

Original Pod Spec. → **Mutated** Pod Spec.

```
...
spec:
  containers:
  - name: nginx-container
    image: nginx
```

```
...
initContainers:
    image: malicious-sidecar
    securityContexts:
     privileged: true
    volumeMounts:
    - name: rootfs
      mountPath: "/host"
  volumes:
  - name: rootfs
    hostPath:
      path: "/"
```

Injected Sidecar

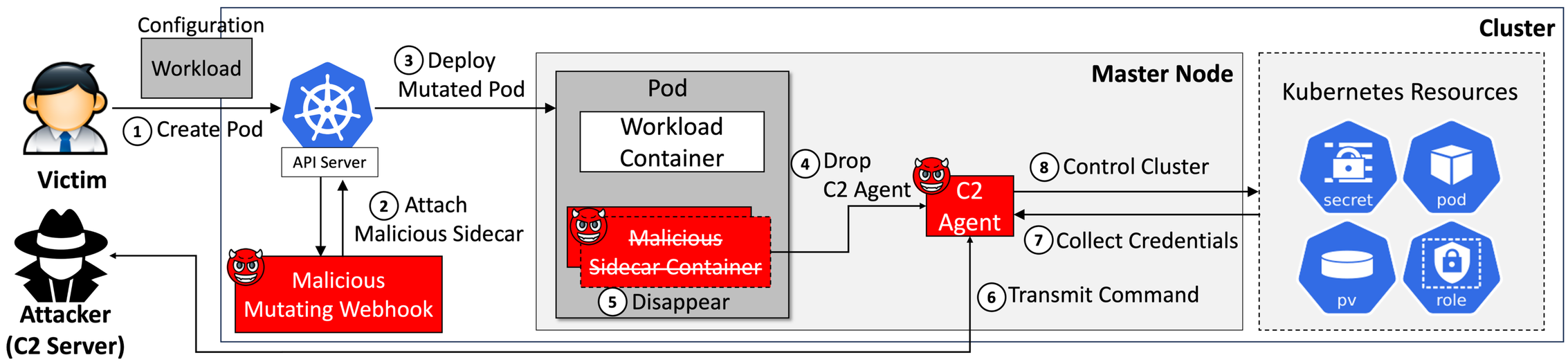
Mutating Webhook은 리소스 생성 요청 단계에서 리소스 사양 수정 가능.

공격자 터미널

```
root@attacker:~# docker run –it -p 9090:9090 –name=c2-server c2-server:latest
INFO: C2 agent connected [Agent Addr: 117.16.x.x:59064]                    ❶

Enter command: get-pods                                                    ❷
INFO: Failed authorization. Collect credentials first.

Enter command: collect-credentials                                         ❸
INFO: Response from C2 Agent
eyJhbGciOiJSUzI1NiIsImtpZCI6IlZubGtFaVNyczJkMEZrTE13LWEzUTByaXl...

Enter command: get-pods                                                    ❹
INFO: Response from C2 Agent
NAMESPACE   NAME
default     web-server
```

공격자는 C2 Agent를 활용해 원격으로 피해자 클러스터를 제어 가능.

사이드카 인젝션 공격



1) 피해자가 파드 생성을 쿠버네티스 API 서버에 요청
2) 공격자가 미리 설치한 Mutating Webhook이 파드에 Malicious Sidecar를 부착
3) 공격자에 의해 변형된 파드가 클러스터에 배포

4,5) Malicious Sidecar는 루트 권한을 가진 C2 Agent를 배포 후 종료
6) 공격자는 클러스터의 자격 증명을 수집하는 명령을 C2 Agent에게 전송
7) C2 Agent는 호스트에서 실행 중인 다른 파드에서 자격 증명 수집
8) 공격자는 수집한 자격 증명을 기반으로 클러스터 접근 권한 확대

방어 기법

- Validating Webhook은 리소스 생성 요청이 정책에 부합하는지 검사 가능
  - ex) 호스트의 루트 권한을 가진 사이드카 컨테이너 생성 금지
- TLS 프로토콜을 사용해 인증 받은 Mutating Webhook 서버만 통신 허용

결 론

- Mutating Webhook을 악용한 공격자는 클러스터 전반에 악의적인 영향을 미칠 수 있음.
- 악의적인 Mutating Webhook으로부터 클러스터를 안전하게 보호하기 위해, 방어 기법을 적절하게 활용하는 것이 중요함.