AppArmor Profile

Purpose: To confine programs to a limited set of resources

→ Linux capabilites, file permissions, network access, etc

Two modes

1. *Enforcing* mode - blocks access to disallowed resources
2. *Complain* mode - only reports violations

AppArmor allows for more secure deployment by restricting what containers can do, and by provides system logs of container activity.

However! It is not an final solution against exploits and other security problems, other measures should be taken for protection and defense.

Objectives

- See an example of how to load a profile on a node
- Learn how to enforce the profile on a Pod
- Learn how to check that the profile is loaded
- See what happens when a profile is violated
- See what happens when a profile cannot be loaded

Caution! Before you begin:

Ensure Kubernetes is running version 1.4 or higher (1.4 is when AA was finally supported)

AA kernel module is enabled, check:

```
cat /sys/module/apparmor/parameters/enabled
Y
```

Container runtime supports AA?(Docker does)

Profile is loaded, you must specify an AA profile that each container should run with. Check this file to view loaded profiles:

```
/sys/kernel/security/apparmor/profiles
```

Note: AA profiles are specified per container, add an annotation to the Pod's metadata:

```
container.apparmor.security.beta.kubernetes.io/<container_name>: <profile_ref>
```

Options for <profile-ref>

- `runtime/default` to apply the runtime's default profile
- `localhost/<profile_name>` to apply the profile loaded on the host with the name `<profile_name>`
- `unconfined` to indicate that no profiles will be loaded

Kubernetes AA enforcement checks if all prerequisites are met, and then forward the profile selection to the container runtime for enforcement; if not, the Pod will be rejected, and will not run.

Verify profile was applied

```
kubectl get events | grep Created
```

or check its proc attr

```
kubectl exec <pod_name> cat /proc/1/attr/current
```

AppArmor profile example

AppArmor Administration

Disable AppArmor

Tools to help with creating AA profiles

aa-genprof and aa-logprof generate profile rules by monitoring an application's activity and logs, and admitting the actions it takes.

bane is an AA prfile generator for Docker that uses a simplified profile language

API Reference

Full AppArmor Documentation

Creating an AppArmor profile, step by step