AppArmor Profile Loader

This is a small proof-of-concept daemon to demonstrate how AppArmor profiles can be loaded onto nodes of a Kubernetes cluster. It is not considered production ready, nor will it be supported as a long-term solution.

Running the AppArmor Profile Loader

The example-daemon.yaml provides an example manifest for running the loader as a cluster DaemonSet. In this example, the loader runs in a DaemonSet pod on each node in the cluster, and periodically (every 30 seconds) polls for new profiles in the `apparmor-profiles` configmap (example manifest). It is recommended to run the Daemon and ConfigMap in a separate, restricted namespace:

```
$ kubectl create -f example-namespace.yaml
$ kubectl create -f example-configmap.yaml # Includes the k8s-nginx profile
$ kubectl create -f example-daemon.yaml
```

Check that the profile was loaded:

```
$ POD=$(kubectl --namespace apparmor get pod -o jsonpath="{.items[0].metadata.name}")
$ kubectl --namespace apparmor logs $POD
I0829 22:48:24.917263       1 loader.go:139] Polling /profiles every 30s
I0829 22:48:24.954295       1 loader.go:196] Loading profiles from /profiles/k8s-
nginx:
Addition succeeded for "k8s-nginx".
I0829 22:48:24.954328       1 loader.go:100] Successfully loaded profiles: [k8s-nginx]
```

Trying running a pod with the loaded profile (requires Kubernetes >= v1.4):

```
$ kubectl create -f example-pod.yaml
# Verify that it's running with the new profile:
$ kubectl exec nginx-apparmor cat /proc/1/attr/current
k8s-nginx (enforce)
$ kubectl exec nginx-apparmor touch /tmp/foo
touch: cannot touch '/tmp/foo': Permission denied
error: error executing remote command: command terminated with non-zero exit code:
Error executing in Docker Container: 1
```

Standalone

The loader go binary can also be run as a standalone binary on the host. It must be run with root privileges:

```
sudo loader -logtostderr /path/to/profile/dir
```

Alternatively, it can be run with the supplied loader docker image:

```
PROFILES_PATH=/path/to/profile/dir
sudo docker run \
    --privileged \
    --detach=true \
    --volume=/sys:/sys:ro \
    --volume=/etc/apparmor.d:/etc/apparmor.d:ro \
```

```
    --volume=$PROFILES_PATH:/profiles:ro \
    --name=aa-loader \
    google/apparmor-loader:latest
```

Build the loader

The loader binary is a simple go program, and can be built with `make all-push WHAT=apparmor-loader` (from test/images).

Limitations

The loader will not unload profiles that are removed, and will not update profiles that are changed. This is by design, since there are nuanced issues with changing profiles that are in use.