Namespace: kube-system, default

Inspecting an existing audit policy file
Enhancing an existing audit policy file
Configuring audit logging
Identifying a logged event

https://kubernetes.io/docs/tasks/debug/debug-cluster/audit/

Enhancing the Existing Audit Policy File
Edit the existing audit policy file at /etc/kubernetes/audit/rules/auditpolicy.yaml. Add a rule that logs events for ConfigMaps and Secrets at the Metadata level. Add another rule that logs events for Services at the Request level.

```
Kubernetes started
root@controlplane:~$ vi /etc/kubernetes/audit/rules/audit-policy.yaml
root@controlplane:~$ cat /etc/kubernetes/audit/rules/audit-policy.yaml
apiVersion: audit.k8s.io/v1
kind: Policy
omitStages:
  - "RequestReceived"
rules:
  - level: RequestResponse
    resources:
    - group: ""
      resources: ["pods"]
  - level: Metadata
    resources:
    - group: ""
      resources: ["secrets","configmaps"]
  - level: Request
    resources:
    - group:
      resources: ["services"]
root@controlplane:~$ vi /etc/kubernetes/manifests/kube-apiserver.yaml
```

Configuring Audit Logging
Configure the API server to consume the audit policy file. Logs should be written to the file /var/log/kubernetes/audit/logs/apiserver.log. Define a maximum number of 5 days to retain audit log files.

Configure the API server to consume the audit policy file by editing the file /etc/kubernetes/manifests/kube-apiserver.yaml.

...
spec:
  containers:

```
- command:
  - kube-apiserver
  - --audit-policy-file=/etc/kubernetes/audit/rules/audit-policy.yaml
  - --audit-log-path=/var/log/kubernetes/audit/logs/apiserver.log
  - --audit-log-maxage=5
  ...
  volumeMounts:
  - mountPath: /etc/kubernetes/audit/rules/audit-policy.yaml
    name: audit
    readOnly: true
  - mountPath: /var/log/kubernetes/audit/logs/
    name: audit-log
    readOnly: false
...
 volumes:
 - name: audit
   hostPath:
     path: /etc/kubernetes/audit/rules/audit-policy.yaml
     type: File
 - name: audit-log
   hostPath:
     path: /var/log/kubernetes/audit/logs/
     type: DirectoryOrCreate
```

The Pod running the API server should automatically restart. This process may take a couple of minutes. Once fully restarted, you should be able to query for it

```
root@controlplane:~$ kubectl get pods -n kube-system
NAME                                   READY   STATUS    RESTARTS        AGE
coredns-787d4945fb-72fmj               1/1     Running   0               17m
coredns-787d4945fb-7drld               1/1     Running   0               17m
etcd-controlplane                      1/1     Running   0               17m
kube-apiserver-controlplane            1/1     Running   0               4m19s
kube-controller-manager-controlplane   1/1     Running   2 (4m52s ago)   17m
kube-proxy-2cmqq                       1/1     Running   0               16m
kube-proxy-vs2rj                       1/1     Running   0               17m
kube-scheduler-controlplane            1/1     Running   1 (9m59s ago)   17m
root@controlplane:~$
```

Check the API server log files under /var/log/pods if the Pod no longer comes up after a reasonable amount of time.

```
root@controlplane:~$ cd /var/log/pods
root@controlplane:/var/log/pods$ ls
kube-flannel_kube-flannel-ds-2j265_24e7a588-dbc4-4638-b5aa-016eea9dc1b9
kube-system_coredns-787d4945fb-72fmj_1f9f3516-8edf-4f85-86c6-f19cea77cecd
kube-system_coredns-787d4945fb-7drld_7464fb2e-8eba-4812-8595-efd26a080bee
kube-system_etcd-controlplane_8eeb6ee1813b7d5128d8a2e2b9d163da
kube-system_kube-apiserver-controlplane_58e6dcfea6059434c21609f13a63f847
kube-system_kube-controller-manager-controlplane_2c56b37387b28f0fbab22a88992005cb
kube-system_kube-proxy-vs2rj_c12ad298-588f-494f-aa46-7228dd359720
kube-system_kube-scheduler-controlplane_cf1e2959d6bc21ac7d83f247034cc01b
```

Creating a Logged Event
Create a ConfigMap named db-user with the key-value pair username=tom in the default namespace. Ensure that the log file has been created and contains at least one entry that matches the events configured.

One of the logged resources is a ConfigMap on the Metadata level. The following command creates an exemplary ConfigMap object.

$ kubectl create configmap db-user --from-literal=username=tom

The audit log file will now contain an entry for the event:

$ cat /var/log/kubernetes/audit/logs/apiserver.log

```
root@controlplane:/var/log/pods$ kubectl get cm
NAME               DATA   AGE
db-user            1      32s
kube-root-ca.crt   1      24m
root@controlplane:/var/log/pods$ taill -10 /var/log/kubernetes/audit/logs/apiserver.log
Command 'taill' not found, did you mean:
  command 'tkill' from deb lam-runtime (7.1.4-7)
  command 'tail' from deb coreutils (8.32-4.1ubuntu1)
Try: apt install <deb name>
root@controlplane:/var/log/pods$ tail -10 /var/log/kubernetes/audit/logs/apiserver.log
{"kind":"Event","apiVersion":"audit.k8s.io/v1","level":"Metadata","auditID":"b900a976-86
```