

# Duplicating Encoding Profiles

Oct 2021

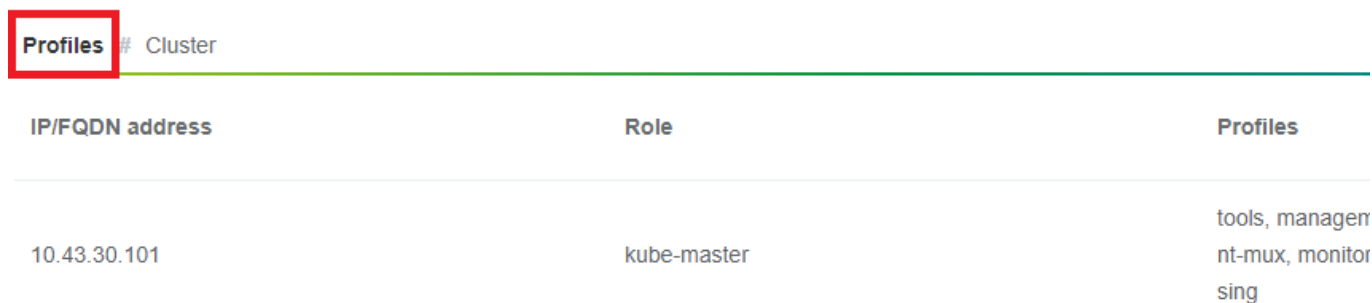
## Introduction

IPHE comes with a set of pre-defined encoding profiles, but no way to manually add a new one from the MDT UI (at the time of writing).

To create a new encoding profile, the best way to do this is to duplicate an existing MDT profile. We just then need to edit a few lines in the profile and re-upload it.

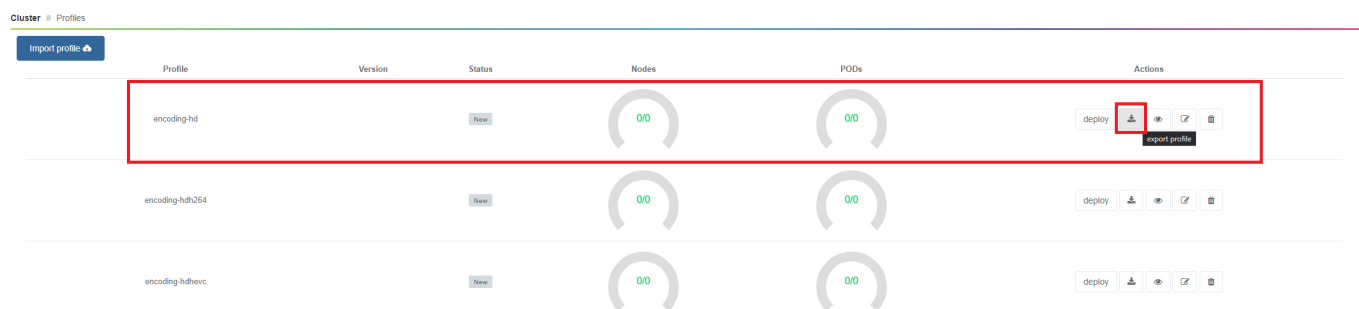
## Download an existing encoding profile

Open the MDT application (<https://ipaddress:30443>) and login using mdt-admin / changeme.



IP/FQDN address	Role	Profiles
10.43.30.101	kube-master	tools, management-mux, monitoring

Now look for an appropriate encoding profile to download (I normally take encoding-hd) and select the 'export profile' button.



Profile	Version	Status	Nodes	PODs	Actions
encoding-hd	New		0/0	0/0	<div> <div>export profile</div> </div>
encoding-hd264	New		0/0	0/0	<div> <div></div> </div>
encoding-hdhevc	New		0/0	0/0	<div> <div></div> </div>

The encoding profile (a .yaml file) will now download.

— Alan Walker 2021/09/21 14:49 — Alan Walker 2021/09/21 14:49

## Editing the encoding profile

Open the .yaml file (preferably in an editor that properly supports .yaml file formats (not notepad)) and save it to a new filename. If this is for a PoD per server then you can just save it as a new encoding filename like 'encoding-hd-custom.yaml' for example. If this is for a PoD per service, then save it including the service name, encoding-hd-bbc1.yaml for example.

Below is the format of the default encoding-hd yaml file:

```
mdt-kinds:
- kind: profile
  name: encoding-hd
  version: ''
  packages:
    - deploy-package: encoding-live-worker-hd
      version: 25.14.76+4.0.19
- kind: deploy-package
  deploy-package:
    name: encoding-live-worker-hd
    version: 25.14.76+4.0.19
  chart:
    name: encoding-live-worker
    version: 25.14.76
  namespace: '{{ products_var.namespace }}'
  deploy_list:
    - mkel-hd1
  options:
    mdt:
      deployment_stage: 5
      upgrade:
        policy: immediate
    helm:
      wait: false
      timeout: 300
  values:
    image:
      serverdaemon:
        repository: '{{ products_var.docker_registry }}/mediakind/server-daemon'
        tag: '{{ products_var.server_daemon.tag }}'
      worker:
        repository: '{{ products_var.docker_registry }}/mediakind/encoding-live-worker'
      redis:
        repository: '{{ products_var.docker_registry }}/redis'
        tag: '{{ products_var.redis.tag }}'
      alarmProxy:
        repository: '{{ products_var.docker_registry }}/mediakind/alarm-proxy'
        tag: '{{ products_var.alarm_proxy.tag }}'
      initcontainer:
        repository: '{{ products_var.docker_registry }}/busybox'
        tag: 1.30.1
    alarmsViaRedisInWorkerPod: '{{ products_var.features.alarmsViaRedisInWorkerPod }}'
    el_worker:
      server_mngt_nic: '{{ products_var.server_mngt_nic }}'
      mpeg4cfg: |
        perChannelLicensing=true
    qsv:
      enable: false
    mongo:
      name: '{{ products_var.mongo.name }}'
      database: '{{ products_var.mongo.replicasetname }}'
      replica: '{{ products_var.mongo.replica }}'
    muhook:
      enable: '{{ products_var.features.muconv }}'
      interface: '{{ products_var.server_mngt_nic }}'
      process_name: '*liveTranscoder'
      log_level: 4
      log_output: stderr
    unified_ui:
      name: api-gateway
    rabbitmq:
      name: '{{ products_var.rabbitmq.name }}'
    etcd_mu:
      name: '{{ products_var.etcd.name }}'
      port: '{{ products_var.etcd.port }}'
    redis:
      name: '{{ products_var.redis.name }}'
  volumes:
    log:
      enable: '{{ products_var.log_volume }}'
```

```
license_server_addr: '{{ products_var.license_server }}'
server_processing_name: LiveEncoder
nodeSelector: '{{ mdt.current_profile.profile_name_label }}'
resources:
  requests:
    cpu: 9
privilegedInitContainer:
  enabled: true
  command:
    - sysctl
    - '-w'
affinity:
  podAntiAffinity:
    requiredDuringSchedulingIgnoredDuringExecution:
      - topologyKey: kubernetes.io/hostname
        labelSelector:
          matchLabels:
            app: encoding-live-worker
```

We need to change a few lines, see below for a side-by-side comparison, where the original text is on the left and the edited text on the right.

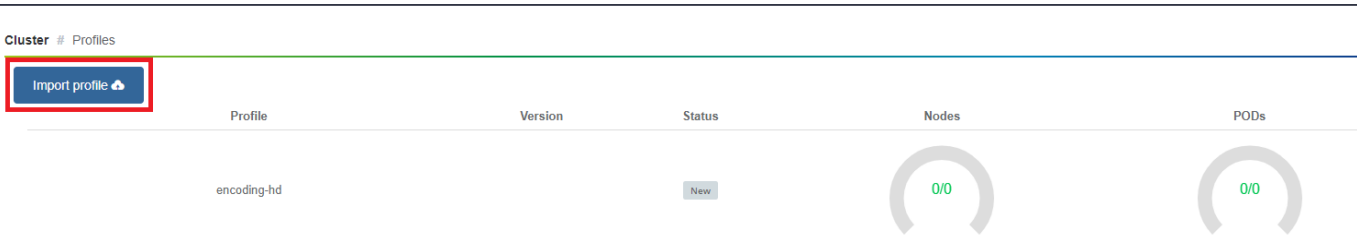
1mdt-kinds:	1mdt-kinds:
2 - kind: profile	2 - kind: profile
3 name: encoding-hd	3 name: encoding-sky-hd-01
4 version: ''	4 version: ''
5 packages:	5 packages:
6 - deploy-package: encoding-live-worker-hd	6 - deploy-package: encoding-live-worker-sky-hd-01
7 version: 25.14.76+4.0.19	7 version: 25.14.76+4.0.19
8 - kind: deploy-package	8 - kind: deploy-package
9 deploy-package:	9 deploy-package:
10 name: encoding-live-worker-hd	10 name: encoding-live-worker-sky-hd-01
11 version: 25.14.76+4.0.19	11 version: 25.14.76+4.0.19
12 chart:	12 chart:
13 name: encoding-live-worker	13 name: encoding-live-worker
14 version: 25.14.76	14 version: 25.14.76
15 namespace: '{{ products_var.namespace }}'	15 namespace: '{{ products_var.namespace }}'
16 deploy_list:	16 deploy_list:
17 - mkel-hd1	17 - sky-01
18 options:	18 options:
19 mdt:	19 mdt:
20 deployment_stage: 5	20 deployment_stage: 5
21 upgrade:	21 upgrade:
22 policy: immediate	22 policy: immediate
23 helm:	23 helm:
24 wait: false	24 wait: false
25 timeout: 300	25 timeout: 300
26 values:	26 values:
27 image:	27 image:
28 serverdaemon:	28 serverdaemon:

The first three lines we edit are just for the profile names, and the associated deploy package. Line 17 is where we change the deploy list names, if you are working with a PoD per node, then you add an entry here for each node that you want this encoding profile for, so if you want 5 encoding profiles of this type, you enter the names of those five encoding profiles (mkel1, mkel2, mkel3, mkel4 and mkel5 for instance).

If you are working with PoD per service, you would enter the service name here, and will only have a single item in your deploy list, bbc1 for instance. You will have to create this file for each service PoD you will require.

Adding the new encoding profile to MDT

Once you have created your new encoding profiles, you can upload them to MDT. To do this use the 'Import Profile' button at the top right of the MDT web UI (ensure you are in profile view).



Now select your profile

Import Profile

enc\_hd\_sky01.yaml

Select file

Cancel

Ok

Once uploaded you will see a confirmation panel.

Success

Profile added successfully

OK

Click okay. Your new encoding profile will be at the bottom of the MDT profile page.

encoding-channel4-hd-01	Deployed	4/4	1/1	reset	upload	eye	edit
encoding-itr-hd-01	Deployed	4/4	1/1	reset	upload	eye	edit
encoding-sky-hd-01	New	0/0	0/0	deploy	upload	eye	edit

To make it active, edit the profile, add some nodes and deploy it.

Edited encoding profile

Below is the edited file that was used in this process.

```
mdt-kinds:
- kind: profile
  name: encoding-sky-hd-01
  version: ''
packages:
- deploy-package: encoding-live-worker-sky-hd-01
  version: 25.14.76+4.0.19
```

```

- kind: deploy-package
  deploy-package:
    name: encoding-live-worker-sky-hd-01
    version: 25.14.76+4.0.19
  chart:
    name: encoding-live-worker
    version: 25.14.76
  namespace: '{{ products_var.namespace }}'
  deploy_list:
    - sky-01
  options:
    mdt:
      deployment_stage: 5
      upgrade:
        policy: immediate
    helm:
      wait: false
      timeout: 300
  values:
    image:
      serverdaemon:
        repository: '{{ products_var.docker_registry }}/mediakind/server-daemon'
        tag: '{{ products_var.server_daemon.tag }}'
      worker:
        repository: '{{ products_var.docker_registry }}/mediakind/encoding-live-worker'
      redis:
        repository: '{{ products_var.docker_registry }}/redis'
        tag: '{{ products_var.redis.tag }}'
      alarmProxy:
        repository: '{{ products_var.docker_registry }}/mediakind/alarm-proxy'
        tag: '{{ products_var.alarm_proxy.tag }}'
      initcontainer:
        repository: '{{ products_var.docker_registry }}/busybox'
        tag: 1.30.1
    alarmsViaRedisInWorkerPod: '{{ products_var.features.alarmsViaRedisInWorkerPod }}'
    el_worker:
      server_mngt_nic: '{{ products_var.server_mngt_nic }}'
      mpeg4cfg: |
        perChannelLicensing=true
    qsv:
      enable: false
    mongo:
      name: '{{ products_var.mongo.name }}'
      database: '{{ products_var.mongo.replicasetname }}'
      replica: '{{ products_var.mongo.replica }}'
    muhook:
      enable: '{{ products_var.features.muconv }}'
      interface: '{{ products_var.server_mngt_nic }}'
      process_name: '*liveTranscoder'
      log_level: 4
      log_output: stderr
    unified_ui:
      name: api-gateway
    rabbitmq:
      name: '{{ products_var.rabbitmq.name }}'
    etcd_mu:
      name: '{{ products_var.etcd.name }}'
      port: '{{ products_var.etcd.port }}'
    redis:
      name: '{{ products_var.redis.name }}'
    volumes:
      log:
        enable: '{{ products_var.log_volume }}'
    license_server_addr: '{{ products_var.license_server }}'
    server_processing_name: LiveEncoder
    nodeSelector: '{{ mdt.current_profile.profile_name_label }}'
    resources:
      requests:
        cpu: 7
        memory: 4Gi

```

```
limits:
  cpu: 15
  memory: 10Gi
privilegedInitContainer:
  enabled: true
  command:
    - sysctl
    - '-w'
affinity:
  podAntiAffinity:
    requiredDuringSchedulingIgnoredDuringExecution:
      - topologyKey: kubernetes.io/hostname
        labelSelector:
          matchLabels:
            app: encoding-live-worker
```

From:  
<http://cameraangle.co.uk/> - WalkerWiki - [wiki.alanwalker.uk](http://wiki.alanwalker.uk)

Permanent link:  
[http://cameraangle.co.uk/doku.php?id=duplicating\\_encoding\\_profiles](http://cameraangle.co.uk/doku.php?id=duplicating_encoding_profiles)

Last update: **2023/03/09 22:35**

