A MITEL
PRODUCT
GUIDE

Mitel OpenScape UC Application V11

Logging and Troubleshooting Guide

Service Documentation
03/2025

A31003-S50B0-S100-01-7620

Mitel

Notices

The information contained in this document is believed to be accurate in all respects but is not warranted by Mitel Europe Limited. The information is subject to change without notice and should not be construed in any way as a commitment by Mitel or any of its affiliates or subsidiaries. Mitel and its affiliates and subsidiaries assume no responsibility for any errors or omissions in this document. Revisions of this document or new editions of it may be issued to incorporate such changes. No part of this document can be reproduced or transmitted in any form or by any means - electronic or mechanical - for any purpose without written permission from Mitel Networks Corporation.

Trademarks

The trademarks, service marks, logos, and graphics (collectively “Trademarks”) appearing on Mitel’s Internet sites or in its publications are registered and unregistered trademarks of Mitel Networks Corporation (MNC) or its subsidiaries (collectively “Mitel), Unify Software and Solutions GmbH & Co. KG or its affiliates (collectively “Unify”) or others. Use of the Trademarks is prohibited without the express consent from Mitel and/or Unify. Please contact our legal department at iplegal@mitel.com for additional information. For a list of the worldwide Mitel and Unify registered trademarks, please refer to the website: http://www.mitel.com/trademarks.

Contents

1 History of Changes

Issue	Changes	Date
1	First issue of the guide.	01/2025

2 OpenScape UC commonly used loggers and examples

2.1 Most commonly used symphonia loggers and examples

This is a list of the most common symphonia loggers, including examples:

- **log4j_backup_restore.xml** - Logger to be used to troubleshoot backup/ restore issues of Data or Filesystem. Example: Backup of UC data, fails to complete.
- **log4j_bcom.xml** - Logger to be used to troubleshoot call process issues. Example: OWC user A calls OWC user B, but cannot answer the call via call control.
- **log4j_bcom-integrated.xml** - same as above, but to be used in integrated simplex deployments.
- **log4j_bcom-ms.xml** - Logger to be used to troubleshoot call issues and media server issues (If UC media server is on the same server as the BE).
- **log4j_bcom-ms-integrated.xml** - same as above, but to be used in integrated simplex deployments.
- **log4j_cmp_administration.xml** - Logger to be used to troubleshoot CMP related issues. Example: When trying to login to CMP, you get an error that the system is not ready for login or user doesn't have permission to log in.
- **log4j_os4k_administration.xml** - Logger to be used to troubleshoot Delta synchronization and Full synchronization with the OS4k PBX.
- **log4j_conferencing.xml** - Logger to be used to troubleshoot conference issues. User A creates a conference via OWC, adds B and C, but when the conference starts, only user B is alerted.
- **log4j_contactservice.xml** - Logger to troubleshoot contact service related issues (global address book, LDAP). Example: When I search for a contact in the address book, I get a timeout.
- **log4j_domain.xml** - Logger to be used to troubleshoot devices assigned to UC users. Example: UC user A has +1234 number assigned via CMP, but in OWC that device is not visible.
- **log4j_dsa.xml** - Logger that troubleshoots assistant related issues. Example: When I try to create a user via CMP, OSV assistant, I get an exception and the user isn't created.
- **log4j_groupwareservice_contacts_full.xml** - Logger to troubleshoot groupware (Exchange) and contact service related issues. Example: I cannot see in the WebClient my private groupware contacts nor search for them.
- **log4j_licensemanagement.xml** - Logger to troubleshoot licensing related issues. Example: I assigned a new UC licensing file for UC users, but the licenses are not activated.
- **log4j_mediaserver_full_log.xml** - Logger to troubleshoot media server issues. Example: After a 1 hour conference, some participants loose speech path.
- **log4j_openbranch_administration.xml** - Logger to troubleshoot OSB assistant related issues. Example: In CMP->OSB, I cannot list my OSBs.
- **log4j_openscape_sbc_administration.xml** - Logger to troubleshoot SBC assistant related issues. Example: In CMP->SBC, I cannot list my SBCs.
- **log4j_presence.xml** - Logger to troubleshoot presence related issues (IM, user presence, media presence). Example: My presence status turns without any action to unavailable.

- **log4j_pwf_bcom.xml** - Logger to troubleshoot preferred devices and call process. Example: I have set a device list with 3 devices and I receive a call, the first 2 devices ring, but the third doesn't.
- **log4j_remotelogging.xml** - Logger to be used, to troubleshoot, symphonia logs transfer via sftp to Trace Manager. Example: symphonia logs are not delivered to TM.
- **log4j_rules_pwf.xml** - Logger to be used, to troubleshoot cases when setting rules which change prefer devices via UC client. Example: When applying a rule, in which when my presence status is available the call should go to my mobile, the office phone rings.
- **log4j_usermanagement.xml** - Logger to be used, to troubleshoot UM related issues. Example: When trying to provision users via UM, the IM address field is empty
- **log4j_mediaserver_oscc-rtc.xml** - Logger to be used for WebRTC related issues and can be activated on Media Server nodes. Example: WebRTC call related issues, screen sharing, registration etc.
- **log4j_provisioning.xml** - Logger used for scenarios where a new user is created via SPML soap request / OSILA.
- **log4j_spnego_security.xml** - Logger to be used to troubleshoot SSO. After enabling the log4j file log4j_spnego_security.xml (and restarting symphonia), the kerberos communication is logged in the file /var/siemens/common/log/osgi.log. Additionally, more log messages are written to the symphonia.log.

2.2 Most commonly used WebClient logs

2.2.1 Backend Logs

- **AddressService.log** - It is logging the actions/activities concerning the search and add of contacts.
- **CallerIdService.log** - It is logging the presentation of number/name in journal and call control.
- **DomainMgmtService.log** - It is logging the UC users information, such as preferred device, timezone, external identifiers, etc, devices.
- **JMailService.log** - It is logging information about the missed call notification functionality via email to the UC user.
- **JrnService.log** - It is logging information about the journal functionality.
- **PCMuxService.log** - It is logging information about user presence, media/phone presence, and instant message.
- **PSMService.log** - It is logging information about call control, call processing via UC client.
- **XprProxy.log** - It is logging information concerning voicemail/Xpressions related functionality.

2.2.2 Frontend Logs

- **openscapeuc.log** - It is logging information related to the call processing (in WebClient interface, OSMO, DeskApp).
- **HttpdPortal.log** - It is logging information about the http post and get requests of http traffic between the client browser and the FE server. You

can find related info such as user login, compatibility settings of browser used.

- **HttpdPortalAccess.log** - It is logging information for owc servlets and user agents of clients.
- **MsgBrokerService.log** - It is logging eventing information of any action performed in the client and passes it the respective BE webclient delegates. Example: when someone changes the presence status,
- **MsgBrokerService** will catch this event in the frontend and then pass the necessary information to the PCMuxService in the BE.
- **WebtierControllerLog4j.log** - It is logging information about EA Cockpit functionality.

IMPORTANT:

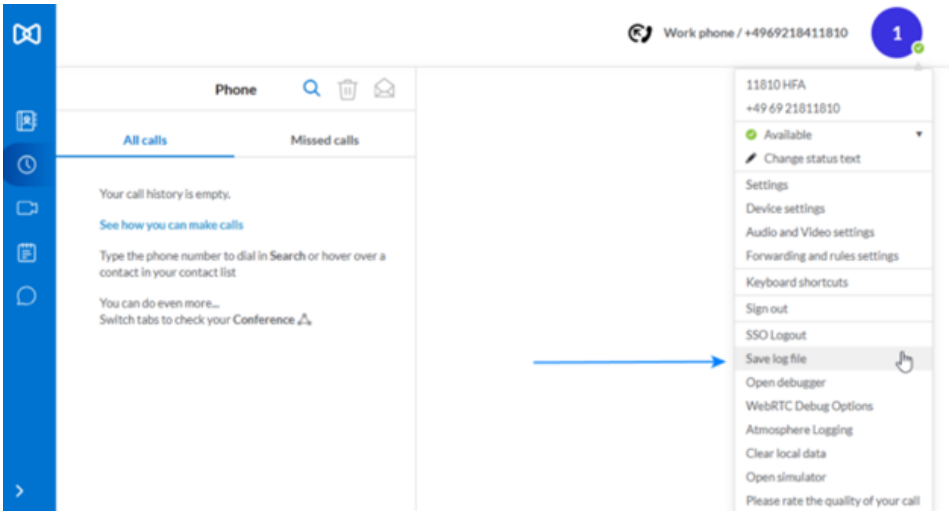
Please note that in case of Integrated simplex and Small Deployment, all above BE and FE logs exist only on the BE server.

Please note that when troubleshooting a Web Client related issue, make sure to have Web Client logs in Level 5 (Debug mode).

Please note that when troubleshooting Smart Client issues (new WebClient Interface), you also need the openscapeuc.log.

2.2.3 WebClientUI logs

The WebClient UI logs record all actions performed in the graphical user interface (GUI). To save these logs, hold down the CTRL key and click on the user's profile picture in the WebClient. Then, select **Save log file** from the drop-down menu, as displayed in the image below:



3 OpenScape UC logs collection

3.1 Logs to verify during UC installation

In case of errors that occur during the installation of OpenScape UC, you must check the following logs:

```
tail -f /var/lib/rancher/rke2/agent/logs/kubelet.log
systemctl status rke2-server.service
```

NOTICE: The rke2 service is responsible for running the Rancher Kubernetes engine.

If the rke2 service is not in the running status, execute the following command or simply start the service again:

```
systemctl restart rke2-server.service
```

If the rke2 service is still not running, execute the command below and provide the command output to the support team:

```
journalctl -u rke2-server.service -f
```

3.2 How to retrieve all OpenScape UC Kubernetes pods

Run the following command to view the list of all OpenScape UC Kubernetes pods:

```
kubectl get pods
```

This command retrieves all UC deployment pods which are in "Running" state and have the "default" namespace, which is a category/classifier.

```
root@ucvll-pooll-j6m5q-78fr9:/home/rancher# kubectl get pods
NAME                          READY   STATUS    RESTARTS      AGE
ucamq-8c775db9c-jfjqv         1/1     Running   0             10d
ucbe-0-785ccb956b-r8cpg       1/1     Running   2 (9d ago)    10d
uckeycloak-65ddf4549d-955jl   1/1     Running   0             10d
uclwfe-78cb4cb5d6-dgd28       1/1     Running   0             10d
ucmsl-7bdb6b9fd7-pv9r5        1/1     Running   0             10d
ucmysql-654fcd4786-g5z25      1/1     Running   0             10d
ucof-fc67b66b9-2vx26          1/1     Running   0             10d
ucoffs-74cd479f6c-fws7l       1/1     Running   0             10d
ucofmysql-5d8fdcc478-4565k    1/1     Running   0             10d
ucowcbe-7b954f5cf8-njxwp      1/1     Running   0             10d
root@ucvll-pooll-j6m5q-78fr9:/home/rancher#
```

Run the following command to continuously track the state of the pods:

```
watch kubectl get pods
```

```
Ctrl + C, to exit
```

It is important to understand the function of each pod in the UC deployment, as well as the specific component or environment it represents.

```
UC Backend pods:
ucbe-x-xxx pod --> UC Backend (the Backend in V10)
ucmysql-xxxx pod --> UC Backend MySQL (MySQL of BE)

UC ActiveMQ pod:
ucamq-xxxx pod --> UC ActiveMQ

UC MediaServer:
ucmsX-xxxx pod --> UC MediaServer (MediaServer in V10, in
 V11 can support up to 4 MediaServer pods)

UC Keycloak:
uckeycloak-xxxx pod --> UC keycloak

UC Webclient:
uclwfe-xxxx pod --> UC Webclient FE (the Frontend in V10)
ucowcbe-xxxx pod --> UC Webclient BE

UC Openfire:
ucof-xxxx pod --> UC Openfire (external openfire in V10)
ucoffs-xxxx pod --> UC Openfire file server (complementary
 to ucof pod, where files are stored, for example chat
 attachments)
ucofmysql-xxxx pod --> UC Openfire MySQL (MySQL of
 Openfire)
```

3.3 OpenScape UC productInventory.sh script

The tool can be applied as follows from the Kubernetes cluster:

```
kubectl exec -it <ucbe_pod_name> -- bash -c
"/opt/siemens/common/bin/productInventory.sh -t"
```

3.4 OpenScape UC systemVersionInfo.sh script

There is a lightweight tool to get all the essential information named `systemVersionInfo` which can be applied during runtime of the system without affecting performance significantly.

The tool `systemVersionInfo` tool provides a collection of version information and the essential configuration information.

The tool can be applied as follows from the Kubernetes cluster:

```
kubectl exec -it <ucbe_pod_name> -- bash -c "/opt/
siemens/servicetools/install/bin/systemVersionInfo.sh
 systemVersionInfo"
```

e.g.

```
Creating Master tar file for the UCBE pod: /tmp/sysVInfo-
ucbe-2025-01-20-T111151-EET.tar
Total bytes written: 1392640 (1.4MiB, 210MiB/s)
Done. Thank you for using systemVersionInfo.sh
```

3.5 OpenScape UC collect.sh script

The `collect.sh` script automatically collects log and dump information of OpenScape UC application. It can be found under `/opt/siemens/common/bin/collect.sh`.

The tool can be applied as follows from the Kubernetes cluster:

```
kubectl exec -it <ucbe_pod_name> -- bash -c "/opt/siemens/
common/bin/collect.sh"
```

e.g.

```
Creating Dump tar file: /tmp/dumpfiles-ucbe-2025-01-20-
T111251-EET.tar
Total bytes written: 3481600 (3.4MiB, 237MiB/s)
-----------------------------------------------------------------
Creating Master tar file: /tmp/logfiles-ucbe-2025-01-20-
T111251-EET.tar
Total bytes written: 7526400 (7.2MiB, 217MiB/s)
SKIPPED: system restart: not required/requested.
Done.
```

You can run the `collect.sh -h` command for more information about the tool.

Selecting logfiles containing traces:

```
START ..... Sun May 15 19:23:07 EEST 2016
END ....... Mon May 16 17:50:45 EEST 2016
..... .....
--------------------------------------------------------------------------
Creating Dump tar file: /tmp/dumpfiles-ucsmall-2016-05-16-T175049-EEST.tar
Total bytes written: 13148160 (13MiB, 501MiB/s)
--------------------------------------------------------------------------
Creating Master tar file: /tmp/logfiles-ucsmall-2016-05-16-T175049-EEST.tar
Total bytes written: 164290560 (157MiB, 97MiB/s)
```

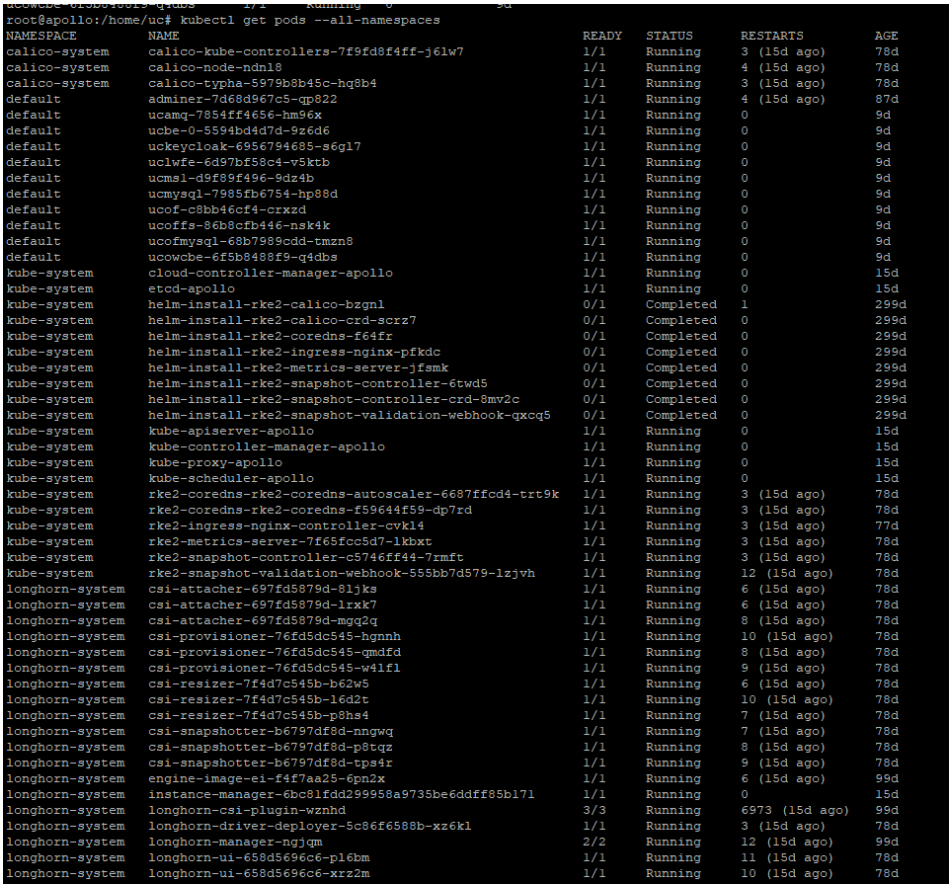
Collected files will be under /tmp/

You can also use the `collect.sh -C -H` command to collect core dumps and heap dumps.

3.6 How to retrieve the system pods

For troubleshooting reasons, you can run the following command to retrieve the "system"/kubernetes cluster pod:

```
kubectl get pods --all-namespaces
```



```
root@apollo:/home/uc# kubectl get pods --all-namespaces
NAMESPACE         NAME                                                READY   STATUS      RESTARTS         AGE
calico-system     calico-kube-controllers-7f9fd8f4ff-j6lw7            1/1     Running     3 (15d ago)      78d
calico-system     calico-node-ndnl8                                   1/1     Running     4 (15d ago)      78d
calico-system     calico-typha-5979b8b45c-hq8b4                       1/1     Running     3 (15d ago)      78d
default           adminer-7d68d967c5-qp822                            1/1     Running     4 (15d ago)      87d
default           ucamq-7854ff4656-hm96x                              1/1     Running     0                9d
default           ucbe-0-5594bd4d7d-9z6d6                             1/1     Running     0                9d
default           uckeycloak-6956794685-s6g17                         1/1     Running     0                9d
default           uclwfe-6d97bf58c4-v5ktb                             1/1     Running     0                9d
default           ucmsl-d9f89f496-9dz4b                               1/1     Running     0                9d
default           ucmysql-7985fb6754-hp88d                            1/1     Running     0                9d
default           ucof-c8bb46cf4-crxzd                                1/1     Running     0                9d
default           ucoffs-86b8cfb446-nsk4k                             1/1     Running     0                9d
default           ucofmysql-68b7989cdd-tmzn8                          1/1     Running     0                9d
default           ucowcbe-6f5b8488f9-q4dbs                            1/1     Running     0                9d
kube-system       cloud-controller-manager-apollo                     1/1     Running     0                15d
kube-system       etcd-apollo                                         1/1     Running     0                15d
kube-system       helm-install-rke2-calico-bzgnl                      0/1     Completed   1                299d
kube-system       helm-install-rke2-calico-crd-scrz7                  0/1     Completed   0                299d
kube-system       helm-install-rke2-coredns-f64fr                     0/1     Completed   0                299d
kube-system       helm-install-rke2-ingress-nginx-pfkdc               0/1     Completed   0                299d
kube-system       helm-install-rke2-metrics-server-jfsmk              0/1     Completed   0                299d
kube-system       helm-install-rke2-snapshot-controller-6twd5         0/1     Completed   0                299d
kube-system       helm-install-rke2-snapshot-controller-crd-8mv2c     0/1     Completed   0                299d
kube-system       helm-install-rke2-snapshot-validation-webhook-qxcq5 0/1     Completed   0                299d
kube-system       kube-apiserver-apollo                               1/1     Running     0                15d
kube-system       kube-controller-manager-apollo                      1/1     Running     0                15d
kube-system       kube-proxy-apollo                                   1/1     Running     0                15d
kube-system       kube-scheduler-apollo                               1/1     Running     0                15d
kube-system       rke2-coredns-rke2-coredns-autoscaler-6687ffcd4-trt9k 1/1     Running     3 (15d ago)      78d
kube-system       rke2-coredns-rke2-coredns-f59644f59-dp7rd            1/1     Running     3 (15d ago)      78d
kube-system       rke2-ingress-nginx-controller-cvkl4                 1/1     Running     3 (15d ago)      77d
kube-system       rke2-metrics-server-7f65fcc5d7-lkbxt                1/1     Running     3 (15d ago)      78d
kube-system       rke2-snapshot-controller-c5746ff44-7rmft            1/1     Running     3 (15d ago)      78d
kube-system       rke2-snapshot-validation-webhook-555bb7d579-lzjvh   1/1     Running     12 (15d ago)     78d
longhorn-system   csi-attacher-697fd5879d-8ljks                       1/1     Running     6 (15d ago)      78d
longhorn-system   csi-attacher-697fd5879d-lrxk7                       1/1     Running     6 (15d ago)      78d
longhorn-system   csi-attacher-697fd5879d-mgq2q                       1/1     Running     8 (15d ago)      78d
longhorn-system   csi-provisioner-76fd5dc545-hgnnh                    1/1     Running     10 (15d ago)     78d
longhorn-system   csi-provisioner-76fd5dc545-qmdfd                    1/1     Running     8 (15d ago)      78d
longhorn-system   csi-provisioner-76fd5dc545-w4lf1                    1/1     Running     9 (15d ago)      78d
longhorn-system   csi-resizer-7f4d7c545b-b62w5                        1/1     Running     6 (15d ago)      78d
longhorn-system   csi-resizer-7f4d7c545b-l6d2t                        1/1     Running     10 (15d ago)     78d
longhorn-system   csi-resizer-7f4d7c545b-p8hs4                        1/1     Running     7 (15d ago)      78d
longhorn-system   csi-snapshotter-b6797df8d-nngwq                     1/1     Running     7 (15d ago)      78d
longhorn-system   csi-snapshotter-b6797df8d-p8tqz                     1/1     Running     8 (15d ago)      78d
longhorn-system   csi-snapshotter-b6797df8d-tps4r                     1/1     Running     9 (15d ago)      78d
longhorn-system   engine-image-ei-f4f7aa25-6pn2x                      1/1     Running     6 (15d ago)      99d
longhorn-system   instance-manager-6bc81fdd299958a9735be6ddff85b171   1/1     Running     0                15d
longhorn-system   longhorn-csi-plugin-wznhd                           3/3     Running     6973 (15d ago)   99d
longhorn-system   longhorn-driver-deployer-5c86f6588b-xz6kl           1/1     Running     3 (15d ago)      78d
longhorn-system   longhorn-manager-ngjqm                              2/2     Running     12 (15d ago)     99d
longhorn-system   longhorn-ui-658d5696c6-pl6bm                        1/1     Running     11 (15d ago)     78d
longhorn-system   longhorn-ui-658d5696c6-xrz2m                        1/1     Running     10 (15d ago)     78d
```

You can find the namespace column on the right side of the command output.

> **IMPORTANT:** Keep in mind that namespace applies to nearly all Kubernetes commands and many more in the Kubernetes API.

If you need to retrieve pods for a specific namespace, run the following command:

```
kubectl get pods -n <namespace_name>
```

The namespace (other than the default namespace) tipically include:

- Kubernetes infrastructure/rancher → **kube-system** namespace
- Data persistency/Volumes → **longhorn-system** namespace
- Networking → **calico-system** + **metallb-system** namespaces

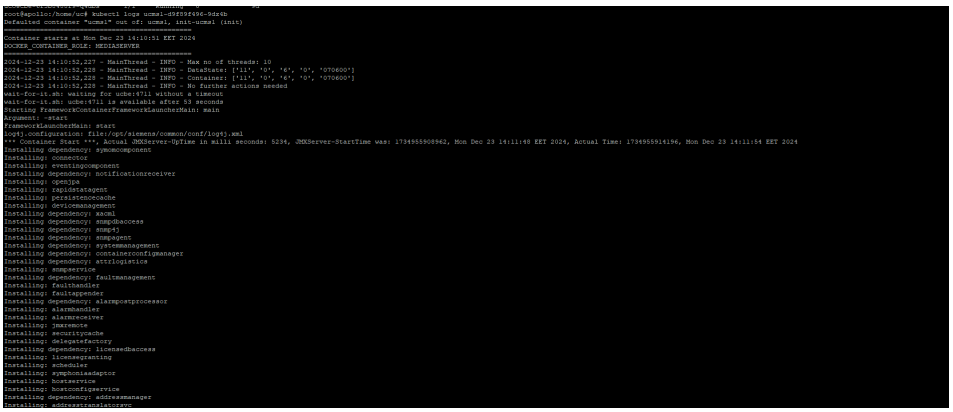
3.7 Top-Level logging commands

Run the following command to retrieve logs for a specific pod:

```
kubectl -n <namespace> logs <pod_name>

If the namespace = "default", you can skip the -n argument
 (it is implied as default)
```

For example, for the MS1 pod, this is the command output for `/var/siemens/common/log/osgi.log`:



3.8 Debug information for pods

Run the following command to retrieve debug information of a specific pod:

```
kubectl describe pod -n <namespace> <pod_name>
```

The example below displays debug information for the kubectl pod.

```
root@apollo:/home/uc# kubectl describe pod -n calico-system calico-kube-
controllers-7f9fd8f4ff-j6lw7
Name:                 calico-kube-controllers-7f9fd8f4ff-j6lw7
Namespace:            calico-system
Priority:             2000000000
Priority Class Name:  system-cluster-critical
Service Account:      calico-kube-controllers
Node:                 apollo/10.11.248.85
Start Time:           Wed, 16 Oct 2024 13:55:10 +0300
Labels:               app.kubernetes.io/name=calico-kube-controllers
                      k8s-app=calico-kube-controllers
                      pod-template-hash=7f9fd8f4ff
Annotations:          cni.projectcalico.org/containerID:
 d2de8b6a8c77925b46f109723973432b8ec238012fd9f5738e6c99b1c02333c4
                      cni.projectcalico.org/podIP: 10.11.248.72/32
                      cni.projectcalico.org/podIPs: 10.11.248.72/32
                      hash.operator.tigera.io/system:
 fdde45054a8ae4f629960ce37570929502e59449
                      kubectl.kubernetes.io/restartedAt: 2024-10-16T13:55:06+03:00
                      tigera-operator.hash.operator.tigera.io/tigera-ca-private:
 628ad7ccb311e4649426c81e565ef001a0a5dc58
Status:               Running
IP:                   10.11.248.72
IPs:
  IP:           10.11.248.72
Controlled By:  ReplicaSet/calico-kube-controllers-7f9fd8f4ff
Containers:
  calico-kube-controllers:
    Container ID:
 containerd://7f8aa6679f45461cd9db8bf8a09c35cf3268e69300d6a5f3950db73d3a9d9586
```

```
    Image:           docker.io/rancher/mirrored-calico-kube-controllers:v3.27.0
    Image ID:        docker.io/rancher/mirrored-calico-kube-
controllers@sha256:d3d056e6be2e3add8641efe693945da157730f4463e4b6c698a8746c6f877224
    Port:            <none>
    Host Port:       <none>
    SeccompProfile:  RuntimeDefault
    State:           Running
      Started:       Wed, 18 Dec 2024 08:50:24 +0200
    Last State:      Terminated
      Reason:        Unknown
      Exit Code:     255
      Started:       Mon, 18 Nov 2024 17:07:55 +0200
      Finished:      Wed, 18 Dec 2024 08:48:57 +0200
    Ready:           True
    Restart Count:   3
    Liveness:        exec [/usr/bin/check-status -l] delay=10s timeout=10s
 period=60s #success=1 #failure=6
    Readiness:       exec [/usr/bin/check-status -r] delay=0s timeout=10s
 period=30s #success=1 #failure=3
    Environment:
      KUBE_CONTROLLERS_CONFIG_NAME:  default
      DATASTORE_TYPE:                kubernetes
      ENABLED_CONTROLLERS:           node
      FIPS_MODE_ENABLED:             false
      KUBERNETES_SERVICE_HOST:       10.43.0.1
      KUBERNETES_SERVICE_PORT:       443
      CA_CRT_PATH:                   /etc/pki/tls/certs/tigera-ca-bundle.crt
    Mounts:
      /etc/pki/tls/cert.pem from tigera-ca-bundle (ro,path="ca-bundle.crt")
      /etc/pki/tls/certs from tigera-ca-bundle (ro)
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-zlr5g
 (ro)
Conditions:
  Type              Status
  Initialized       True
  Ready             True
  ContainersReady   True
  PodScheduled      True
Volumes:
  tigera-ca-bundle:
    Type:      ConfigMap (a volume populated by a ConfigMap)
    Name:      tigera-ca-bundle
    Optional:  false
  kube-api-access-zlr5g:
    Type:                     Projected (a volume that contains injected data from
 multiple sources)
    TokenExpirationSeconds:  3607
    ConfigMapName:           kube-root-ca.crt
    ConfigMapOptional:       <nil>
    DownwardAPI:             true
QoS Class:                   BestEffort
Node-Selectors:              kubernetes.io/os=linux
Tolerations:                 CriticalAddonsOnly op=Exists
                             node-role.kubernetes.io/control-plane:NoSchedule
 op=Exists
                             node-role.kubernetes.io/control-plane:NoSchedule
```

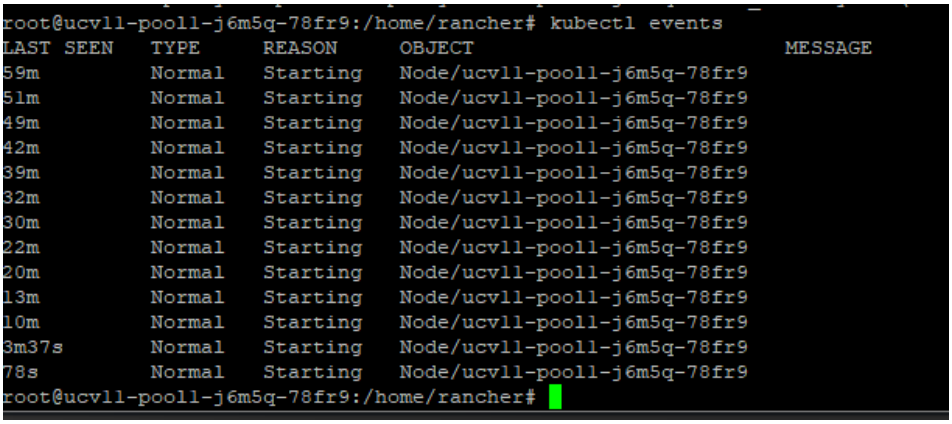
```
                        node-role.kubernetes.io/etcd:NoExecute op=Exists
                        node-role.kubernetes.io/master:NoSchedule
                        node.kubernetes.io/not-ready:NoExecute op=Exists for
 300s
                        node.kubernetes.io/unreachable:NoExecute op=Exists
 for 300s
Events:                 <none>
```

3.9 How to retrieve Events pods

Events pods include useful information regarding potential incidents or errors. Most Event pods are "Normal" pods, such as the startup logs.

Run the following command to retrieve all events in a namespace:

```
kubectl events -n <namespace>
```



You can also retrieve events for a specific pod:

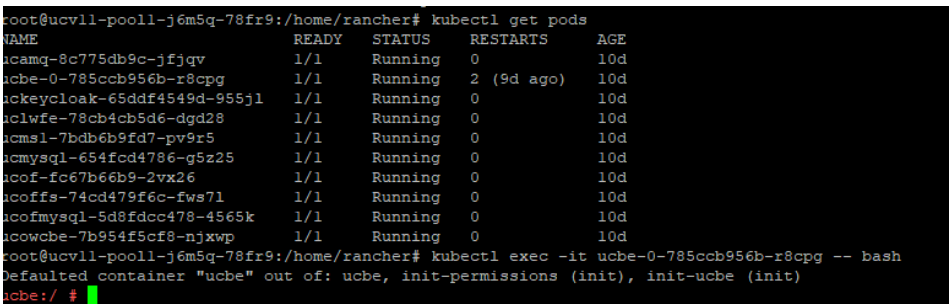
```
kubectl events --for pod/<pod_name>
```

3.10 How to retrieve information from the pod's shell

Run the following command from inside the pod's shell (kubectl exec):

```
kubectl exec -it <pod_name> -- bash
```

```
Where <pod_name> will be one of the pods in the "default"
 namespace (the UC pods)
```

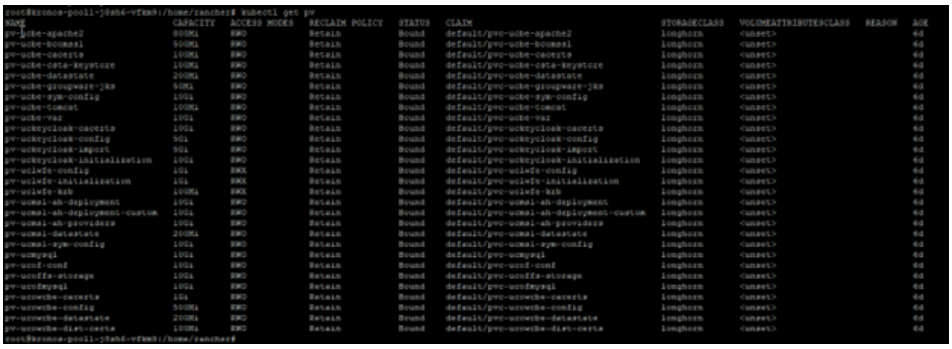


By default, you are placed in the "/" directory. You can check your current location at any time with the "pwd" command.

3.11 Kubernetes Persistent Volumes

The command `kubectl get pv` lists all Persistent Volumes in a Kubernetes cluster.

`kubectl get pvc -n default`



`kubectl describe pvc pvc-ucbe-var -n default`

```
root@kronos-pool1-j8sh6-vfkm9:/home/rancher#
root@kronos-pool1-j8sh6-vfkm9:/home/rancher# kubectl describe pvc pvc-ucbe-var -n default
Name:          pvc-ucbe-var
Namespace:     default
StorageClass:  longhorn
Status:        Bound
Volume:        pv-ucbe-var
Labels:        app=pvc-ucbe-var
               app.kubernetes.io/managed-by=Helm
Annotations:   meta.helm.sh/release-name: uc
               meta.helm.sh/release-namespace: default
               pv.kubernetes.io/bind-completed: yes
Finalizers:    [kubernetes.io/pvc-protection]
Capacity:      10Gi
Access Modes:  RWO
VolumeMode:    Filesystem
Used By:       ucbe-0-54c9788d6d-cz4jd
Events:        <none>
root@kronos-pool1-j8sh6-vfkm9:/home/rancher#
root@kronos-pool1-j8sh6-vfkm9:/home/rancher#
```

3.12 How to retrieve and copy files from a specific pod

Run the following command to retrieve a file from a pod:

```
kubectl cp <pod_name>:<filename_full_path_inside_pod>
 <filename_full_path_rancher_host>

For example, if you want to retrieve the symphonia.log from
 the ucbe pod, you would execute the following:
--> kubectl cp ucbe-0-785ccb956b-r8cpg:var/siemens/common/
log/symphonia.log /home/rancher/symphonia.log

Notice that the <filename_full_path_inside_pod>, does not
 contain the 1st leading "/" (this is the correct syntax,
 otherwise there would be an error)
```

Run the following command to copy a file from a pod:

```
kubectl cp <filename_full_path_rancher_host>
 <pod_name>:<filename_full_path_inside_pod>

For example, if you want to copy a local file (test.log)
 from the ucbe pod, you would execute the following:
--> kubectl cp /home/rancher/test.log ucbe-0-785ccb956b-
r8cpg:var/siemens/common/log/test.log

Again, notice that the <filename_full_path_inside_pod>,
 does not contain the 1st leading "/"
```

3.13 Best practices for restarting a pod

You can run the `kubectl rollout restart` command on a ucowcbe deployment to ensure Kubernetes handles the restart process properly and maintains the desired state.

```
root@kronos-pool1-j8sh6-vfkm9:/home/rancher# kubectl get
 deployment
NAME          READY   UP-TO-DATE   AVAILABLE   AGE
ucamq         1/1     1            1           12d
ucbe-0        1/1     1            1           12d
uckeycloak    1/1     1            1           12d
uclwfe        1/1     1            1           12d
ucms1         1/1     1            1           12d
ucmysql       1/1     1            1           12d
ucof          1/1     1            1           12d
ucoffs        1/1     1            1           12d
ucofmysql     1/1     1            1           12d
ucowcbe       1/1     1            1           12d
root@kronos-pool1-j8sh6-vfkm9:/home/rancher#
```

4 BCom (call scenarios) troubleshooting

- **How to find Bcom Events in symphonia log:**

```
[com.siemens.symphonia.bcom.bcomsvc.impl.RMRSessionManagerImpl] <LogId
BCom="1" /> fireEventViaRMR: fired
```

- **How to find CSTA Requests/Events in symphonia log:**

  ```
  com.siemens.symphonia.bcom.cstaconnector.adaptor.TcpEndPoint
  ```

- **How to search for CSTA Events in symphonia log:**

  Search in log for "/> ***"

  Example for an outgoing call:

  ```
  OSV-CSTA-IP:1040" /> ***SENT: 1335
   <MakeCall><callingDevice>
  OSV-CSTA-IP:1040" /> ***RECEIVED: 1335 <?xml
   version="1.0" encoding="UTF-
  8"?><MakeCallResponse
  OSV-CSTA-IP:1040" /> ***RECEIVED: 9999 <?xml
   version="1.0" encoding="UTF-
  8"?><ServiceInitiatedEvent
  OSV-CSTA-IP:1040" /> ***RECEIVED: 9999 <?xml
   version="1.0" encoding="UTF-
  8"?><OriginatedEvent
  OSV-CSTA-IP:1040" /> ***RECEIVED: 9999 <?xml
   version="1.0" encoding="UTF-
  8"?><NetworkReachedEvent
  OSV-CSTA-IP:1040" /> ***RECEIVED: 9999 <?xml
   version="1.0" encoding="UTF-
  8"?><DeliveredEvent
  OSV-CSTA-IP:1040" /> ***RECEIVED: 9999 <?xml
   version="1.0" encoding="UTF-
  8"?><EstablishedEvent
  OSV-CSTA-IP:1040" /> ***RECEIVED: 9999 <?xml
   version="1.0" encoding="UTF-
  8"?><ConnectionClearedEvent
  ```

- **How to find CSTA link status events in symphonia logs:**

```
[com.siemens.symphonia.bcom.bcomsvc.delegate.EventsFromProviderDelegate]
```

- **How to find active call counts in symphonia logs: (useful for media presence issues):**

  "active calls"

  Example:

  INFO [com.siemens.symphonia.presence.bcom.impl.BComHandlingWorker] **handleNewConnection**:: User test@system **has 1 active calls**

  INFO [com.siemens.symphonia.presence.bcom.impl.BComHandlingWorker] **handleTerminatedConnection**:: User test@system **has 0 active calls**

- **How to check that a user's devices (ONS and additional resources) are monitored by BCom:**

  On BE node go to `/opt/siemens/servicetools/bcom`

  Execute:

  ```
  showBComStatus BE_IP_ADDRESS administrator@system
   "ADMIN_PASSWORD" | grep
  "userid@system"
  ```

  The output should be like this:

  ```
  [INFO] DiscoverService(BComApi): url =
  https://BE_IP_ADDRESS:4709/com/siemens/symphonia/
  clientproxy/RemoteDiscovery
  [INFO] DiscoverService(BComApi): got https://
  FE_IP_ADDRESS/FQDN:4709
  [INFO] ws.url =
  https://FE_IP_ADDRESS/FQDN:4709/com/siemens/symphonia/
  bcom/bcomapisvc/generated/interf/BCo
  mApi
  [INFO] wsEvent.url =
  https://FE_IP_ADDRESS/FQDN:4709/com/siemens/symphonia/
  bcom/bcomapisvc/generated/interf/BCo
  mApi Device +302105566777 [ Pbx +302105566777 ]
   monitored userid@system
  Device 3021055667771;phone-
  context=private.private.302105566@system [ Pbx
   3021055667771 ]
  monitored userid@system Device 3021055667772;phone-
  context=private.private.302105566@system
  [ Pbx 3021055667772 ] monitored userid@system
  ```

  [INFO] httpsSetupDone

  [INFO] DiscoverService(BComApi): url =
- **WebClient tips:**

  How to find Call control windows of a specific user in MsgBrokerService log of WebClient FE:

  Send to: username@system Message: {{"id":"CallInfo"

  How to find Calls initiated by WebClient in MsgBrokerService log of WebClient FE:

  INFO [CC-PlugIn-1] msg: {"id":"MakeCall",

CSTA Request	CSTA Event	Bcom Event
MakeCall Request		
	ServiceInitiatedEvent	NewCommunicationEvent
	OriginatedEvent	DialtoneEvent
	OfferedEvent	InitiatedEvent
AcceptCall Request		
	DeliveredEvent	AlertingEvent/ RingbackEvent

CSTA Request	CSTA Event	Bcom Event
AnswerCall Request		
	EstablishedEvent	ConnectedEvent
ClearConnection Request		
	ConnectionClearEvent	TerminatedEvent

5 MySQL commands for quick troubleshooting

For the recovery of user and password, the mySQL password must be launched from the cluster towards the UCBE pod. Next, the DB interrogation is performed from the BE pod.

```
kubectl exec -it <ucbe_pod_name> -- bash -c "/opt/
siemens/servicetools/install/bin/installlib.sh
 persistence.getDBMSdata -Dmagic=db"
```

The command output is the following:

```
persistence.getDBMSdata:
     [echo] PW: root::dba
```

For the DB interrogation, you can use the following:

- The UCMYSQL pod:

  ```
  kubectl exec -it <ucmysql_pod_name> – bash
  ```

- The corresponding credentials:

  ```
  mysql -uroot -pdba
  ```

  The command output is the following:

  ```
  mysql> SHOW DATABASES;
  +----------------------+
  | Database |
  +----------------------+
  | information_schema |
  | BATCHJOB |
  | OPENBRANCHAPP |
  | OSVSYNCH |
  | dsaadminappcontroller |
  | dsargconnectionconfig |
  | mysql |
  | openscapesbc |
  | performance_schema |
  | symAlarmMgmt |
  | symBackup |
  | symCmpOsee |
  | symCmpSettings |
  | symConferencing |
  | symContactList |
  | symDomain |
  | symFaultMgmt |
  | symGroupware |
  | symGroupwareEWS |
  | symInventory |
  | symLicenseMgmt |
  | symPresence |
  | symPresenceTarget |
  | symRegistration |
  | symRules |
  | symSSLMgmt |
  ```

```
| symScheduler |
| symSnmpTrap |
| symStorage |
| symTellMeWhen |
| symWorkflow |
| symXpressions |
| symcadb |
| sys |
| um4kemsdk |
| umLdapSettings |
| umLocality |
| ums |
+----------------------+
mysql> use symConferencing;
mysql> show TABLES;
+--------------------------+
| Tables_in_symConferencing |
+--------------------------+
| AccessCodes |
| BridgeNumbers |
| BridgeNumbersTenantConfig |
| Compatibility |
| ConferenceProperties |
| Conferences |
| Documents |
| Invitees |
| Occurrences |
| QRTZ_BLOB_TRIGGERS |
| QRTZ_CALENDARS |
| QRTZ_CRON_TRIGGERS |
| QRTZ_FIRED_TRIGGERS |
| QRTZ_JOB_DETAILS |
| QRTZ_LOCKS |
| QRTZ_PAUSED_TRIGGER_GRPS |
| QRTZ_SCHEDULER_STATE |
A31003-S50A0-S100-03-7620, 02/2024
14 OpenScape UC V10, Logging and Troubleshooting Guide
| QRTZ_SIMPLE_TRIGGERS |
| QRTZ_SIMPROP_TRIGGERS |
| QRTZ_TRIGGERS |
| RecordingProperties |
| Recordings |
| symConferencingDBVersion |
| symMessages |
+--------------------------+
mysql> select * from Conferences where
 ConferenceUri='conf.13516';
```

5.1 Examples of queries in case we need to collect general Conferencing related information or for specific users

Querying all conferences in the system, sorted by a field and output to a file

If you want to export a specific database you can execute the following command

mysql -u root -p -D symConferencing -e "select ConferenceUri, CreatorOwner, Count,

ScheduleTime, CreationDate from Conferences order by ScheduleTime;" >>

/tmp/all_conf_output_by_.txt

Number of configured conferences per user

mysql -u root -p -D symConferencing -e "select CreatorOwner, count(*) from Conferences group

by CreatorOwner;" > /tmp/confperuser.txt eg.

CreatorOwner count(*) user1@system 6 user2@system 6 user3@system 97 user4@system 27

Query conferences for a specific user

mysql -u root -p-D symConferencing -e "select ConferenceUri, CreatorOwner, SUBJECT from

Conferences WHERE Creatorowner="@system';"

5.2 If you want to export a specific database you can execute the following command

```
mysqldump -u root -p symPresence > presence.sql
```

Enter password:

5.3 How to access Openfire DB

You can use the **UCOFMYSQL** pod to interrogate the Openfire database:

```
kubectl exec -it <ucofmysql_pod_name> – bash
```

5.4 To export the complete MySQL DB

```
mysqldump -u root -p --extended-insert=FALSE --single-
transaction --all-databases > alldb.sql
```

Enter password:

5.5 To export a single table from Openfire DB e.g. ofPrivacyList

mysqldump -u root -p --extended-insert=FALSE --single-transaction openfire_db

ofPrivacyList>export_ofPrivacyList.sql If not changed, root password can be found under

/root/.mysql.pass or .mysql_secret .

5.6 To export the complete Openfire DB

mysqldump -u root -p --extended-insert=FALSE --single-transaction openfire_db > of_db.sql

5.7 To retrieve information on when UC users logged in last time

SELECT UserURI, FROM_UNIXTIME(LastLogonTime/1000) AS 'LogonDateReadable' FROM

symDomain.symUserAuthentication Auth WHERE

LastLogonTime>0;

You can use "order by UserURI" in order to sort by the userid.

SELECT UserURI, FROM_UNIXTIME(LastLogonTime/1000) AS 'LogonDateReadable' FROM

symDomain.symUserAuthentication Auth WHERE LastLogonTime>0 order by UserURI;

5.8 How to retrieve all the users who did not use their UC account over a long period of time (inactive users)

SELECT UserURI FROM symDomain.symUserAuthentication WHERE lastlogontime + 31536000000

< ((UNIX_TIMESTAMP()*1000)); where 31536000000 stands for 365 days in milliseconds The above

query will also fetch users that have never logged in.

To find out these users separately execute:

SELECT UserURI FROM symDomain.symUserAuthentication WHERE lastlogontime = 0;

You can always enhance the commands with "order by UserURI" at the end in order to sort by the userid.

5.9 How to find out which UC users have been locked

SELECT symDomain.symUsers.UserId FROM symDomain.symUsers WHERE

symDomain.symUsers.UUIDKey IN (SELECT UserKey FROM symDomain. symUserAttributeValues

WHERE symDomain.symUserAttributeValues.AttributeTypeKey = 'sym.attribute.user.10' AND

symDomain. symUserAttributeValues.AttributeValue = 1);

You can use the following query to unlock them:

UPDATE symDomain.symUserAttributeValues SET

symDomain.symUserAttributeValues.AttributeValue = 0 WHERE

symDomain.symUserAttributeValues. AttributeTypeKey = 'sym.attribute.user.10' AND

symDomain.symUserAttributeValues.AttributeValue = 1;

5.10 MySQL commands to retrieve detailed database output only in case it has been identified that there is a DB problem

GLOBAL STATUS

SHOW GLOBAL STATUS;

mysql -u root -p --table -e "show global status;" > global_status.txt

GLOBAL VARIABLES

SHOW GLOBAL VARIABLES;

mysql -u root -p --table -e "show global variables;" > global_variables.txt

INNODB STATUS

SHOW ENGINE INNODB STATUS;

mysql -u root -p --table -e "show engine innodb status;" > engine_innodb.txt

FULL PROCESSLIST

SHOW FULL PROCESSLIST;

mysql -u root -p --table -e "show full processlist;" > full_processlist.txt

6 E/A Cockpit related info

6.1 How to reload the E/A Cockpit application

Open a web browser and use the following URL:

http://:7789/eacockpitosc/main?reload=true

6.2 Log entry for checking that E/A Cockpit has started loading the groups

2018-01-29 11:07:54,755 DEBUG [EACockpit_GroupsResolver] loadGroups; 68;

createGroupsFromSOAP BEGIN ###

6.3 Log entry for checking interruption in the loading procedure

(maybe during the loading procedure the administrator has changed one of the groups and the action

caused the loading procedure to be interrupted) 2018-01-29 11:08:02,077 INFO

[EACockpit_GroupsResolver] #### Reload interrupted ###

6.4 Log entry for checking that a changed group was reloaded

2018-01-29 11:08:08,260 DEBUG [EACockpit_GroupsResolver] loadGroups; 611;

groupToReloadName = 499131847877_8390_8445_8391 In general, the administrator must wait for

the application to finish the group load procedure prior to doing any changes to groups.

Please check in the BE pod the below config file.

/opt/siemens/HiPathCA/config/common/global.cfg which should include

Log size depending on 8k scenario

```
<?x if(<?x $IS_HP8K ?> == "1") ?>
<?x setvar (LOG_INIT_LEVEL4J = "5" ) ?>
<?x setvar (LOG_INIT_LEVEL = "5" ) ?>
<?x setvar (LOG_INIT_MAXDAYS = "3" ) ?>
<?x setvar (LOG_INIT_MAXFILESPERDAY = "15" ) ?>
<?x else ?>
<?x setvar (LOG_INIT_LEVEL4J = "5" ) ?>
<?x setvar (LOG_INIT_LEVEL = "5" ) ?>
<?x setvar (LOG_INIT_MAXDAYS = "5" ) ?>
<?x setvar (LOG_INIT_MAXFILESPERDAY = "30" ) ?>
<?x endif ?>
```

as well as

/opt/siemens/HiPathCA/config/services/default/ServiceController/default/resources/log4j.xml including

the following sequence

<logger name="com.siemens.symphonia.eacockpit">

<level value="DEBUG"/> </logger>

Dedicated logs for the EA-Cockpit functionality can be found under:

/var/siemens/common/log/WebClient/.../WebTierControllerLog4j..log or

/var/siemens/common/log/WebClient/.../WebEngineControllerLog4j..log

Please also provide the configuration files:

/opt/siemens/HiPathCA/WebSpace/Portal/webapps/eacockpit-osc/WEB-INF/groups/*

7 How to solve consumed licenses inconsistencies

In case, for example, “Feature name” “OpenScape UC Appl User” appears as 0 out of XYZ or the number of used licenses value appears greater than the maximum amount available, proceed with activating the License Management Service healing mechanism.

Go to CMP > Maintenance > Inventory > Nodes > select backend node > Show services status > search for *icense* > open “License Management Service” > set value of ActivateHealingMechanism to true and Save the change.

Check that under CMP > Maintenance > Licenses > Information the consumed Licenses are now valid based on uploaded license file.

Restart of the License Management Service NOT required.

After validation of the licenses, change ActivatedHealingMechanism value back to false

8 How to generate a Media Server nativeRTPunit backtrace file

This is mandatory when escalating issues to MS team that are related to nativeRTPunit crashes.

The backtrace file needs to be generated for each nativeRTPunit crash file (*.core):

```
kubectl exec -it <ucms_pod_name> --bash
```

Go to /opt/siemens/mediaserver/application_host/bin and you will see the core file generated e.g. ft2ucms-nativeRTPunit-x-3907.core

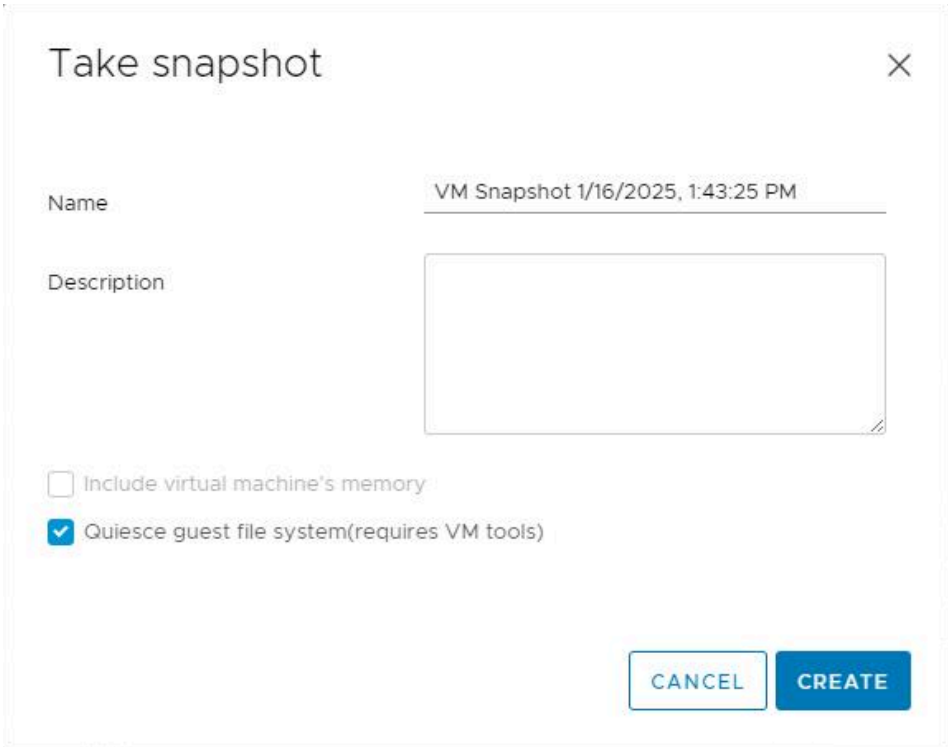
Then execute for this core file:

gdb -ex "set pagination off" -ex "thread appl all bt" -ex "q" ./nativeRTPunit-x86_64 ./ft2ucmsnativeRTPunit-x-3907.core > backtrace-3907.txt

9 OpenScape UC Backup and Restore

In a Kubernetes environment, using CMP for backup and restore is not recommended as it is not functional. Instead, it is advisable to take a snapshot of the Kubernetes virtual machine (VM) for backup, restore, and disaster recovery purposes. In the event of a failure, you can restore the virtual machine by reverting to this snapshot. When taking the snapshot, make sure to select the "Quiesce guest file system" option to ensure that the file system is in a consistent state. Additionally, confirm that the virtual machine is powered off before reverting to the snapshot.

The image below shows an example of taking a snapshot of the Kubernetes VM.



If a restart of the UC is required after reverting to a snapshot, follow the steps below:

Step by Step

1) At the Kubernetes node VM, navigate to `/home/rancher/`.

2) Run the following command and wait until all pods are down:

```
watch kubectl get pod -o wide
```

3) Delete the UC Helm release using the following command:

```
helm delete uc
```

4) Run the following command to estart (or start) the UC pods by installing the UC Helm chart:

```
helm install uc uchelm-<version>.tgz -f values.yaml --
set uc.freshInstall=true
```

Add the corresponding uchelm version, according to your environment.

10 OpenScape UC - outages related to cluster top commands

The following commands should be executed inside the affected pod:

```
root@kronos-pool1-j8sh6-vfkm9:~# kubectl top pod --all-namespaces
NAMESPACE             NAME                                                      CPU(cores)
 MEMORY(bytes)
calico-system         calico-kube-controllers-57df7f8bd8-qvtrc                  2m           37Mi
calico-system         calico-node-99nm6                                         30m          172Mi
calico-system         calico-typha-7d9cf9f74-w24jq                              2m           36Mi
cattle-fleet-system   fleet-agent-0                                             1m           54Mi
cattle-system         cattle-cluster-agent-8699cc4d64-4pdwg                     15m          724Mi
cattle-system         rancher-webhook-5db6f87b7f-pj44t                          4m           48Mi
cattle-system         system-upgrade-controller-684d58bdbc-h7c9s                1m           25Mi
default               ucamq-df8468677-v5gbg                                     2m           70Mi
default               ucbe-0-54c9788d6d-cz4jd                                   69m          5849Mi
default               uckeycloak-7bdcbfd4fd-l2p4b                               4m           494Mi
default               uclwfe-6cfcf55f8d-z7sp2                                   1m           1366Mi
default               ucms1-5f584b6f67-tnwc8                                    202m         1963Mi
default               ucmysql-7bb6cd44c5-f524v                                  7m           796Mi
default               ucof-7fb8f87b5f-sd4zk                                     3m           1808Mi
default               ucoffs-699bc67dd9-5kdpp                                   1m           53Mi
default               ucofmysql-58bcd7c8f8-svvgx                                6m           562Mi
default               ucowcbe-79ffcbcfb6-b5brl                                  3m           781Mi
kube-system           cloud-controller-manager-kronos-pool1-j8sh6-vfkm9         3m           25Mi
kube-system           etcd-kronos-pool1-j8sh6-vfkm9                             41m          129Mi
kube-system           kube-apiserver-kronos-pool1-j8sh6-vfkm9                   140m         741Mi
kube-system           kube-controller-manager-kronos-pool1-j8sh6-vfkm9          24m          86Mi
kube-system           kube-proxy-kronos-pool1-j8sh6-vfkm9                       1m           36Mi
kube-system           kube-scheduler-kronos-pool1-j8sh6-vfkm9                   5m           44Mi
kube-system           rke2-coredns-rke2-coredns-66589bfc87-9rdw5                3m           33Mi
kube-system           rke2-coredns-rke2-coredns-autoscaler-5c6bdd86bc-2g97n     1m           21Mi
kube-system           rke2-ingress-nginx-controller-lzxx9                       2m           132Mi
kube-system           rke2-metrics-server-75866c5bb5-92tgj                      3m           41Mi
kube-system           rke2-snapshot-controller-577cdd5b98-8jpsh                 1m           24Mi
kube-system           rke2-snapshot-validation-webhook-855f5f7d76-p25fp         2m           18Mi
longhorn-system       csi-attacher-77c5cd7b99-6wnrr                             1m           9Mi
longhorn-system       csi-attacher-77c5cd7b99-vgt2k                             1m           9Mi
longhorn-system       csi-attacher-77c5cd7b99-zs4qj                             1m           13Mi
longhorn-system       csi-provisioner-5ccbf6fd5f-bw7mb                          1m           10Mi
longhorn-system       csi-provisioner-5ccbf6fd5f-jdrlj                          1m           16Mi
longhorn-system       csi-provisioner-5ccbf6fd5f-zrm9b                          1m           10Mi
longhorn-system       csi-resizer-57bccf8bcc-lzd79                              1m           14Mi
longhorn-system       csi-resizer-57bccf8bcc-t89tf                              1m           11Mi
longhorn-system       csi-resizer-57bccf8bcc-xmw4z                              1m           16Mi
longhorn-system       csi-snapshotter-88859f4f6-62l44                           1m           17Mi
longhorn-system       csi-snapshotter-88859f4f6-gnfn5                           1m           10Mi
longhorn-system       csi-snapshotter-88859f4f6-pj8gf                           1m           12Mi
longhorn-system       engine-image-ei-4623b511-28kdv                            30m          26Mi
longhorn-system       instance-manager-a227316ae56160cd96eb28226ec27046          18m          54Mi
longhorn-system       longhorn-csi-plugin-k4pd2                                 2m           39Mi
longhorn-system       longhorn-driver-deployer-5b5bdd484f-7fk7k                 1m           12Mi
longhorn-system       longhorn-manager-94f6r                                    50m          132Mi
longhorn-system       longhorn-ui-867df7fc6b-2m7jj                              0m           2Mi
```

```
longhorn-system       longhorn-ui-867df7fc6b-j69rv                              0m           2Mi
metallb-system        metallb-controller-6c9c496dd7-sv742                       2m           38Mi
metallb-system        metallb-speaker-wh6hj                                     6m           40Mi
tigera-operator       tigera-operator-864879b58d-l6tzz                          3m           62Mi

All Pods


root@kronos-pool1-j8sh6-vfkm9:~# kubectl top nodes
NAME                        CPU(cores)   CPU%   MEMORY(bytes)   MEMORY%
kronos-pool1-j8sh6-vfkm9   894m         11%    20294Mi         63%
root@kronos-pool1-j8sh6-vfkm9:~#


root@kronos-pool1-j8sh6-vfkm9:~# kubectl top pod ucamq-df8468677-v5gbg -n default
NAME                     CPU(cores)   MEMORY(bytes)
ucamq-df8468677-v5gbg   2m           70Mi
root@kronos-pool1-j8sh6-vfkm9:~#
root@kronos-pool1-j8sh6-vfkm9:~# kubectl top pod ucbe-0-54c9788d6d-cz4jd -n default
NAME                       CPU(cores)   MEMORY(bytes)
ucbe-0-54c9788d6d-cz4jd   24m          5850Mi
root@kronos-pool1-j8sh6-vfkm9:~#


root@kronos-pool1-j8sh6-vfkm9:~# kubectl top pod --all-namespaces --sort-by=memory
NAMESPACE             NAME                                                      CPU(cores)
 MEMORY(bytes)
default               ucbe-0-54c9788d6d-cz4jd                                   21m          5849Mi
default               ucms1-5f584b6f67-tnwc8                                    213m         1963Mi
default               ucof-7fb8f87b5f-sd4zk                                     2m           1808Mi
default               uclwfe-6cfcf55f8d-z7sp2                                   1m           1366Mi
default               ucmysql-7bb6cd44c5-f524v                                  7m           796Mi
default               ucowcbe-79ffcbcfb6-b5brl                                  3m           781Mi
kube-system           kube-apiserver-kronos-pool1-j8sh6-vfkm9                   161m         748Mi
cattle-system         cattle-cluster-agent-8699cc4d64-4pdwg                     15m          666Mi
default               ucofmysql-58bcd7c8f8-svvgx                                10m          562Mi
default               uckeycloak-7bdcbfd4fd-l2p4b                               4m           494Mi
calico-system         calico-node-99nm6                                         24m          172Mi
longhorn-system       longhorn-manager-94f6r                                    31m          136Mi
kube-system           rke2-ingress-nginx-controller-lzxx9                       3m           132Mi
kube-system           etcd-kronos-pool1-j8sh6-vfkm9                             45m          131Mi
kube-system           kube-controller-manager-kronos-pool1-j8sh6-vfkm9          21m          86Mi
default               ucamq-df8468677-v5gbg                                     2m           70Mi
tigera-operator       tigera-operator-864879b58d-l6tzz                          8m           60Mi
longhorn-system       instance-manager-a227316ae56160cd96eb28226ec27046         22m          54Mi
cattle-fleet-system   fleet-agent-0                                             2m           53Mi
default               ucoffs-699bc67dd9-5kdpp                                   1m           53Mi
cattle-system         rancher-webhook-5db6f87b7f-pj44t                          2m           48Mi
kube-system           kube-scheduler-kronos-pool1-j8sh6-vfkm9                   5m           44Mi
kube-system           rke2-metrics-server-75866c5bb5-92tgj                      3m           42Mi
metallb-system        metallb-speaker-wh6hj                                     6m           40Mi
longhorn-system       longhorn-csi-plugin-k4pd2                                 2m           39Mi
metallb-system        metallb-controller-6c9c496dd7-sv742                       2m           38Mi
calico-system         calico-kube-controllers-57df7f8bd8-qvtrc                  6m           37Mi
calico-system         calico-typha-7d9cf9f74-w24jq                              2m           36Mi
kube-system           kube-proxy-kronos-pool1-j8sh6-vfkm9                       1m           36Mi
```

```
kube-system           rke2-coredns-rke2-coredns-66589bfc87-9rdw5              4m           33Mi
longhorn-system       engine-image-ei-4623b511-28kdv                         29m          26Mi
kube-system           cloud-controller-manager-kronos-pool1-j8sh6-vfkm9       3m           25Mi
cattle-system         system-upgrade-controller-684d58bdbc-h7c9s              1m           24Mi
kube-system           rke2-snapshot-controller-577cdd5b98-8jpsh               1m           24Mi
kube-system           rke2-coredns-rke2-coredns-autoscaler-5c6bdd86bc-2g97n   1m           21Mi
kube-system           rke2-snapshot-validation-webhook-855f5f7d76-p25fp       1m           18Mi
longhorn-system       csi-snapshotter-88859f4f6-62l44                         1m           17Mi
longhorn-system       csi-provisioner-5ccbf6fd5f-jdrlj                        1m           16Mi
longhorn-system       csi-resizer-57bccf8bcc-xmw4z                            1m           16Mi
longhorn-system       csi-resizer-57bccf8bcc-lzd79                            1m           14Mi
longhorn-system       csi-attacher-77c5cd7b99-zs4qj                           1m           13Mi
longhorn-system       csi-snapshotter-88859f4f6-pj8gf                         1m           12Mi
longhorn-system       longhorn-driver-deployer-5b5bdd484f-7fk7k               1m           12Mi
longhorn-system       csi-resizer-57bccf8bcc-t89tf                            1m           11Mi
longhorn-system       csi-provisioner-5ccbf6fd5f-zrm9b                        1m           10Mi
longhorn-system       csi-snapshotter-88859f4f6-gnfn5                         1m           10Mi
longhorn-system       csi-provisioner-5ccbf6fd5f-bw7mb                        1m           10Mi
longhorn-system       csi-attacher-77c5cd7b99-vgt2k                           1m           9Mi
longhorn-system       csi-attacher-77c5cd7b99-6wnrr                           1m           9Mi
longhorn-system       longhorn-ui-867df7fc6b-2m7jj                            0m           2Mi
longhorn-system       longhorn-ui-867df7fc6b-j69rv                            0m           2Mi


root@kronos-pool1-j8sh6-vfkm9:~# kubectl top pod --all-namespaces --sort-by=cpu
NAMESPACE             NAME                                                    CPU(cores)
 MEMORY(bytes)
default               ucms1-5f584b6f67-tnwc8                                  168m         1963Mi
kube-system           kube-apiserver-kronos-pool1-j8sh6-vfkm9                 134m         749Mi
kube-system           etcd-kronos-pool1-j8sh6-vfkm9                           39m          132Mi
longhorn-system       engine-image-ei-4623b511-28kdv                          29m          26Mi
default               ucbe-0-54c9788d6d-cz4jd                                 26m          5849Mi
longhorn-system       longhorn-manager-94f6r                                  22m          136Mi
kube-system           kube-controller-manager-kronos-pool1-j8sh6-vfkm9        21m          86Mi
calico-system         calico-node-99nm6                                       19m          171Mi
longhorn-system       instance-manager-a227316ae56160cd96eb28226ec27046       19m          54Mi
cattle-system         cattle-cluster-agent-8699cc4d64-4pdwg                   15m          666Mi
default               ucmysql-7bb6cd44c5-f524v                                7m           796Mi
metallb-system        metallb-speaker-wh6hj                                   7m           40Mi
default               ucofmysql-58bcd7c8f8-svvgx                              6m           562Mi
kube-system           kube-scheduler-kronos-pool1-j8sh6-vfkm9                 6m           44Mi
kube-system           rke2-metrics-server-75866c5bb5-92tgj                    5m           42Mi
kube-system           cloud-controller-manager-kronos-pool1-j8sh6-vfkm9       4m           25Mi
default               ucof-7fb8f87b5f-sd4zk                                   3m           1808Mi
kube-system           rke2-coredns-rke2-coredns-66589bfc87-9rdw5              3m           33Mi
default               ucowcbe-79ffcbcfb6-b5brl                                3m           781Mi
default               uckeycloak-7bdcbfd4fd-l2p4b                             3m           494Mi
tigera-operator       tigera-operator-864879b58d-l6tzz                        3m           57Mi
kube-system           rke2-ingress-nginx-controller-lzxx9                     2m           132Mi
longhorn-system       csi-provisioner-5ccbf6fd5f-jdrlj                        2m           16Mi
calico-system         calico-typha-7d9cf9f74-w24jq                            2m           36Mi
cattle-fleet-system   fleet-agent-0                                           2m           48Mi
metallb-system        metallb-controller-6c9c496dd7-sv742                     2m           38Mi
cattle-system         rancher-webhook-5db6f87b7f-pj44t                        2m           48Mi
longhorn-system       longhorn-csi-plugin-k4pd2                               2m           39Mi
kube-system           rke2-snapshot-validation-webhook-855f5f7d76-p25fp       2m           18Mi
```

```
default               ucamq-df8468677-v5gbg                                  2m           70Mi
longhorn-system       csi-attacher-77c5cd7b99-6wnrr                          1m           9Mi
kube-system           kube-proxy-kronos-pool1-j8sh6-vfkm9                    1m           36Mi
longhorn-system       csi-provisioner-5ccbf6fd5f-bw7mb                       1m           10Mi
longhorn-system       csi-attacher-77c5cd7b99-vgt2k                          1m           9Mi
longhorn-system       csi-provisioner-5ccbf6fd5f-zrm9b                       1m           10Mi
longhorn-system       csi-resizer-57bccf8bcc-lzd79                           1m           14Mi
longhorn-system       csi-resizer-57bccf8bcc-t89tf                           1m           11Mi
longhorn-system       csi-resizer-57bccf8bcc-xmw4z                           1m           16Mi
longhorn-system       csi-snapshotter-88859f4f6-62144                        1m           17Mi
longhorn-system       csi-snapshotter-88859f4f6-gnfn5                        1m           10Mi
longhorn-system       csi-snapshotter-88859f4f6-pj8gf                        1m           12Mi
longhorn-system       csi-attacher-77c5cd7b99-zs4qj                          1m           13Mi
cattle-system         system-upgrade-controller-684d58bdbc-h7c9s             1m           24Mi
kube-system           rke2-snapshot-controller-577cdd5b98-8jpsh              1m           24Mi
longhorn-system       longhorn-driver-deployer-5b5bdd484f-7fk7k              1m           12Mi
default               uclwfe-6cfcf55f8d-z7sp2                                1m           1366Mi
default               ucoffs-699bc67dd9-5kdpp                                1m           53Mi
kube-system           rke2-coredns-rke2-coredns-autoscaler-5c6bdd86bc-2g97n  1m           21Mi
calico-system         calico-kube-controllers-57df7f8bd8-qvtrc               1m           37Mi
longhorn-system       longhorn-ui-867df7fc6b-j69rv                           0m           2Mi
longhorn-system       longhorn-ui-867df7fc6b-2m7jj                           0m           2Mi
root@kronos-pool1-j8sh6-vfkm9:~#
```

```
root@kronos-pool1-j8sh6-vfkm9:/home/rancher# kubectl exec -it ucbe-0-54c9788d6d-cz4jd  -- bash -c  "ps
 aux --sort -rss"
Defaulted container "ucbe" out of: ucbe, init-permissions (init), init-ucbe (init)
USER         PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
sym            1  1.6  9.4 13526680 3117072 ?    SLsl Jan16 100:43 /opt/siemens/share/ibm-java-
x86_64-80/jre/bin/java -cp :/opt/siemens/common/dist/symphonia-frameworklauncher-current.jar:/
sym          279  0.4  7.6 6958720 2502636 ?     SLl  Jan16  27:35 jsvc.exec -nodetach -home /opt/
siemens/share/ibm-java-x86_64-80/jre -cp :/opt/siemens/share/tomcat/bin/bootstrap.jar:/opt/
root         112  0.0  0.1 2776156 53732 ?       SLl  Jan16   0:13 /opt/siemens/share/ibm-java-
x86_64-80/jre/bin/java -cp :/opt/siemens/common/dist/symphonia-udpforwarder-current.jar:/opt/s
sym           88  0.4  0.0 4358172 18548 ?       Sl   Jan16  29:24 /usr/local/mysqlrouter/bin/
mysqlrouter -c /etc/mysqlrouter/mysqlrouter.conf
root          21  0.0  0.0  20148 12044 ?        Ss   Jan16   0:17 /usr/sbin/httpd-prefork -DSYSCONFIG
 -DSSL -C PidFile /run/httpd.pid -C Include /etc/apache2/sysconfig.d//loadmodule.conf -
wwwrun        77  0.0  0.0  21672  6952 ?        S    Jan16   0:00 /usr/sbin/httpd-prefork -DSYSCONFIG
 -DSSL -C PidFile /run/httpd.pid -C Include /etc/apache2/sysconfig.d//loadmodule.conf -
wwwrun        78  0.0  0.0  21672  6952 ?        S    Jan16   0:00 /usr/sbin/httpd-prefork -DSYSCONFIG
 -DSSL -C PidFile /run/httpd.pid -C Include /etc/apache2/sysconfig.d//loadmodule.conf -
wwwrun        79  0.0  0.0  21672  6952 ?        S    Jan16   0:00 /usr/sbin/httpd-prefork -DSYSCONFIG
 -DSSL -C PidFile /run/httpd.pid -C Include /etc/apache2/sysconfig.d//loadmodule.conf -
wwwrun        80  0.0  0.0  21672  6952 ?        S    Jan16   0:00 /usr/sbin/httpd-prefork -DSYSCONFIG
 -DSSL -C PidFile /run/httpd.pid -C Include /etc/apache2/sysconfig.d//loadmodule.conf -
wwwrun        81  0.0  0.0  21672  6952 ?        S    Jan16   0:00 /usr/sbin/httpd-prefork -DSYSCONFIG
 -DSSL -C PidFile /run/httpd.pid -C Include /etc/apache2/sysconfig.d//loadmodule.conf -
root       15269  0.0  0.0   5208  4600 pts/0    Ss+  Jan17   0:00 bash
root       61830  100  0.0   7336  3192 pts/1    Rs+  20:24   0:00 ps aux --sort -rss
sym          238  0.0  0.0   2740  1264 ?        S    Jan16   0:19 tail -F /var/siemens/common/log/
osgi.log
sym          113  0.0  0.0   2736  1160 ?        S    Jan16   0:00 jsvc.exec -nodetach -home /opt/
siemens/share/ibm-java-x86_64-80/jre -cp :/opt/siemens/share/tomcat/bin/bootstrap.jar:/opt/
```

```
root@kronos-pool1-j8sh6-vfkm9:/home/rancher#
```

10.1 Applying collect.sh script to collect logging data

The tool can be applied as follows from the Kubernetes cluster:

```
kubectl exec -it <ucbe_pod_name> -- bash -c "/opt/siemens/
common/bin/collect.sh"
```

e.g.

```
Creating Dump tar file: /tmp/dumpfiles-ucbe-2025-01-20-
T111251-EET.tar
Total bytes written: 3481600 (3.4MiB, 237MiB/s)
------------------------------------------------------------------
Creating Master tar file: /tmp/logfiles-ucbe-2025-01-20-
T111251-EET.tar
Total bytes written: 7526400 (7.2MiB, 217MiB/s)
SKIPPED: system restart: not required/requested.
Done.
```

You can run the `collect.sh -h` command for more information about the tool.

Selecting logfiles containing traces:

```
START ..... Sun May 15 19:23:07 EEST 2016
END ....... Mon May 16 17:50:45 EEST 2016
..... .....
--------------------------------------------------------------------------
Creating Dump tar file: /tmp/dumpfiles-ucsmall-2016-05-16-T175049-EEST.tar
Total bytes written: 13148160 (13MiB, 501MiB/s)
--------------------------------------------------------------------------
Creating Master tar file: /tmp/logfiles-ucsmall-2016-05-16-T175049-EEST.tar
Total bytes written: 164290560 (157MiB, 97MiB/s)
```

Collected files will be under /tmp/

You can also use the `collect.sh -C -H` command to collect core dumps and heap dumps.

11 OpenScape UC - WebRTC and Web clients

11.1 OpenScape UC WebClient logging

For nearly all WebClient related issues runtime logs are needed. For all deployments except Integrated Simplex they are located in /var/siemens/common /log/webclient. For the Integrated Simplex they are located under /log/webclient/.

The logs are needed in debug-level usually to get the most information out of them and to resolve the issue as soon as possible.

For a more detailed WebClient trace including DEBUG and TRACE messages the variables.

LOG_INIT_LEVEL4J and LOG_INIT_LEVEL need to be set to a value of 5 in the following section of `/longhorn/opt/unify/docker/volumes/ucowcbe-config/common/global.cfg`.

For this, please edit the `global.cfg` file and make sure the log-config-section looks as follows:

```
# Log size depending on 8k scenario
<?x if(<?x $IS_HP8K ?> == "1") ?>
<?x setvar (LOG_INIT_LEVEL4J = "5" ) ?>
<?x setvar (LOG_INIT_LEVEL = "5" ) ?>
<?x setvar (LOG_INIT_MAXDAYS = "3" ) ?>
<?x setvar (LOG_INIT_MAXFILESPERDAY = "15" ) ?>
<?x else ?>
<?x setvar (LOG_INIT_LEVEL4J = "5" ) ?>
<?x setvar (LOG_INIT_LEVEL = "5" ) ?>
<?x setvar (LOG_INIT_MAXDAYS = "5" ) ?>
<?x setvar (LOG_INIT_MAXFILESPERDAY = "30" ) ?> <?x endif ?
>
```

After changing the `global.cfg` file according to the above example, you need to restart WebClient on all nodes of the deployment. You must restart only the UCWCBE pod with the following command:

```
kubectl rollout restart deployment ucowcbe
```

All FE pods should be restarted using the `kubectl rollout restart deployment uclwfe` command.

For the new design of the WebClient interface (openscapeuc.log) please change the value for the root-logger in `/longhorn/opt/unify/docker/volumes/uclwfe-config/common/ log4j-webtier.xml` to ALL:

```
<root>
<priority value="ALL"
 class="com.cycos.media.logging.log4j.ExtendedLevel"/>
 <appender-ref
ref="R"/>
</root>
```

These logs can be found under `/var/siemens/common/log/webclient/openscapeuc` for all deployments apart from Integrated Simplex. For large deployments, look for them on the Frontend nodes.

For the Integrated Simplex deployment, they can be found under `/log/webclient/openscapeuc/`.

Activate webtier log level ALL for openscapeuc logs on the fly (without UC restart)

The administrator is able to reload the log4j settings after having manually edited the **log4jwebtier.xml** file (located under **/opt/siemens/HiPathCA/config /common/log4j-webtier.xml** on each FE node or in the offboard node in case of a Small deployment) and set the log level to ALL.

For the changes to take effect you must restart only the UCFE pods with the `kubectl rollout restart deployment uclwfe` command. After you reproduce the scenario, you must change the log level back to UNEXPECTED. And again for the changes to take effect you must restart only the UCFE pods with the `kubectl rollout restart deployment uclwfe` command.

UI debugger logs

Before starting executing a scenario, start the UI debugger on the WebClient (hold Ctrl on the keyboard and click on your profile and select > Open debugg er) then begin with the scenario and keep the window opened until the scenario execution is finished.

11.2 ODC-PE (Personal Edition) client - trace information

#1: ODC-PE Traces guide

The ODC-PE has its own Trace guide.

Here the last one V2.16 (it's always available within last ODC-PE released. Example: INF-23-000015).

https://nuxeo.unify.com/nuxeo/site/proxy/nxdoc/view/raw/400a11a4-d665-44f1-9773-6d0501f24368

#2: ODC-PE logs

The logs must be activated via Diagnosis tool.

C:\Program Files (x86)\Unify\OpenScape Desktop Client\tools \OpenScapeClientDiagnosis.exe

PS: If for some reason the Diagnosis is not working you can activate manually (via regedit) following the instructions within Trace Guide.

#3: ODC-PE and other Clients are NOT compatible.

ODC-PE cannot be installed(work) if already have other Clients installed like Fusion for Office or Fusion for Notes, etc.

To work, you must UNINSTALL completely the previous Clients, clean/delete regedit keys about it (if uninstall did not clean it automatically) and only after that install the ODC-PE.

#4: ODC-PE SIP and HFA are not compatible.

ODC-PE once installed as SIP or HFA Stackless, can not create a profile for the other provider.

Examples:

Installed ODC-PE SIP => you cannot create a profile for HFA

Installed ODC-PE HFA => you cannot create a profile for SIP.

#5: DUMP 32 bit

ODC-PE (when having problems like CRASH or FREEZING) must have collected (along with ODC-PE logs) the DUMP of it too. But, since ODC-PE it's a 32 bit app, it must be collected a DUMP 32 bit (NOT 64).

Several ways to do that, the easiest is via Windows Task Manager variant 32 bit:

Even if you are using a 64 bits Windows, use the 32-bit TASK MANAGER, because the processes of OpenScape are in 32 bit.

So, let's assume you are running our Desktop Client and a problem happens.

Here are the steps to collect the DUMP 32bit faster:

1) Open the Windows Task Manager for 32-bit:

Since today's most OS are 64-bit, we need to use other Task Manager than the standard one open via CTRL+SHIFT+ESC (that would be for 64 bit). - Open an Explorer Window - Move to the folder C:\Windows\SysWOW64 - Run(double-click) over the exe file " TaskMgr.exe

2) Select the Desired process to collect the DUMP:

With the Task Manager (32 bit) open, goes to desired process to have the DUMP collected. - (via Mouse Rightclick) select the option "Create Dump File" and wait for creation of DUMP file. For complete details/screenshots, please check the OpenScape Desktop client Trace Guide.

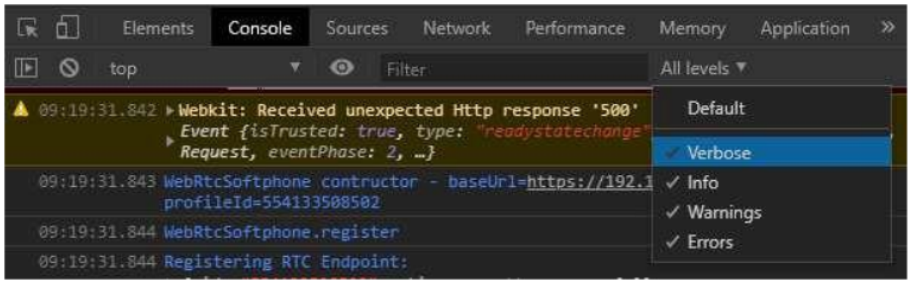
11.3 WebRTC - necessary logs for analysis

- It's always useful to have the full **collect.sh**
- Symphonia logs activated in the active MS pod for WC WebRTC troubleshoot should be the **log4j_mediaserver_oscc-rtc.xml"**
- Important configuration files for WebRTC in the active MS pod are the following:

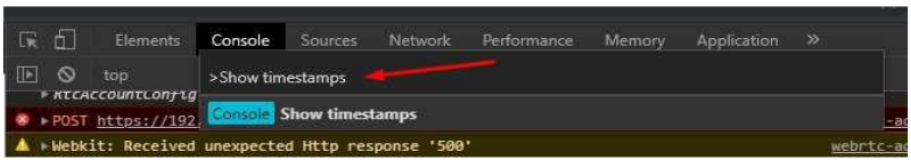
```
/opt/siemens/mediaserver/application_host/providers/turn/turn.component.xml
/opt/siemens/mediaserver/application_host/providers/tomcat/tomcat.component.xml
```

In order to check the basic functionality of WebRTC, proceed with the following steps:

1) Open Chrome or Firefox and navigate to the Web Client page.
2) Open the "Developer tools" (F12), go to the console tab and enable all log levels.
3) The register event should be verified in debug tools.Enable all log levels in browser debugging tool.



4) Add timestamp to debug tools using the option “Run Command” (Ctrl+Shift +P) and command “show timestamps”



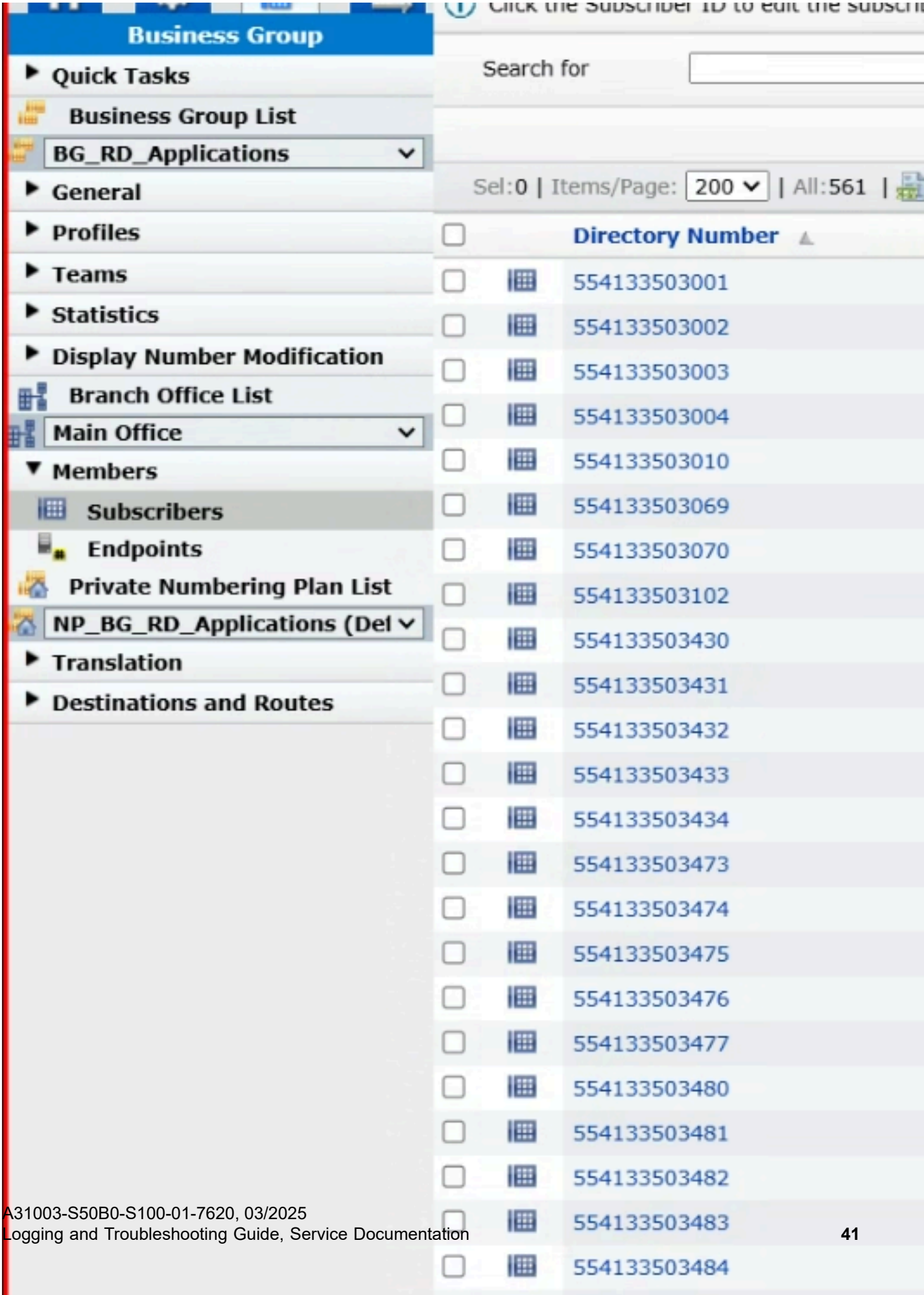
11.4 WebRTC - how to view registered subscribers

On OSV as root:

/opt/solid/bin/solsql -x **pwdfile:/opt/solid/RtpDb/dba.secrets** -e "select ALIAS, DEVICE_ADDR, DEVICE_PORT, REG_STATE, CONTACT_URI from

EPT_CONTACT_DEVICE_T where DEVICE_ADDR like 'MS_IP_ADDRESS'" 'tcp 16760' dba

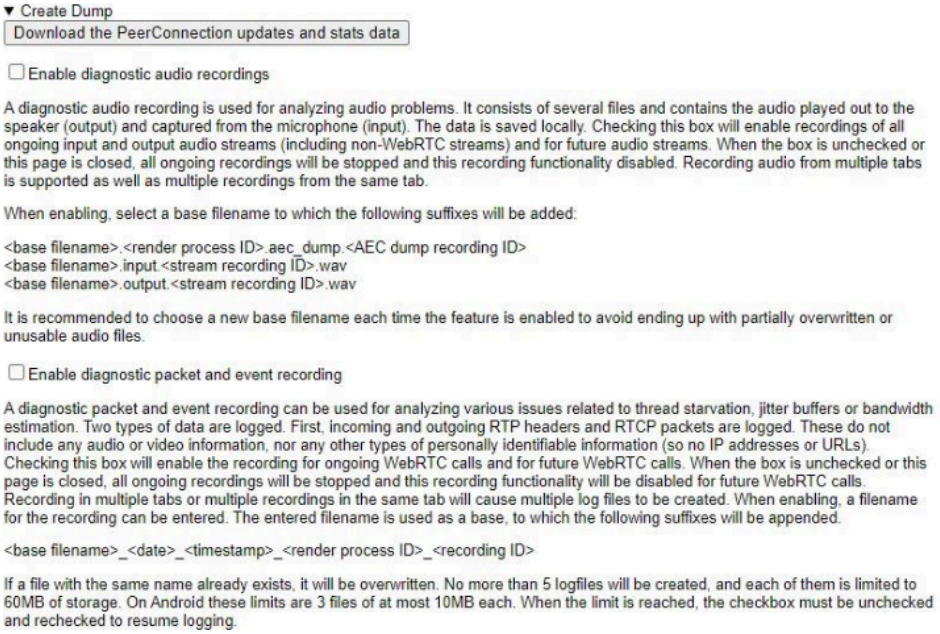
Alternatively, you can view the registered subscribers via CMP:


Business Group
Quick Tasks
Business Group List
BG_RD_Applications
General
Profiles
Teams
Statistics
Display Number Modification
Branch Office List
Main Office
Members
Subscribers
Endpoints
Private Numbering Plan List
NP_BG_RD_Applications (Del
Translation
Destinations and Routes
Search for
Sel:0 | Items/Page: 200 | All:561
Directory Number
554133503001
554133503002
554133503003
554133503004
554133503010
554133503069
554133503070
554133503102
554133503430
554133503431
554133503432
554133503433
554133503434
554133503473
554133503474
554133503475
554133503476
554133503477
554133503480
554133503481
554133503482
554133503483
554133503484

11.5 WebRTC - get all stats for an active WebRTC call

Run this on Chrome browser during an active WebRTC call: `chrome://webrtc-internals/`

Create the dump.



11.6 Manual steps for tracing the UC DesktopApp

Follow the steps below to collect traces from the UC DesktopApp:

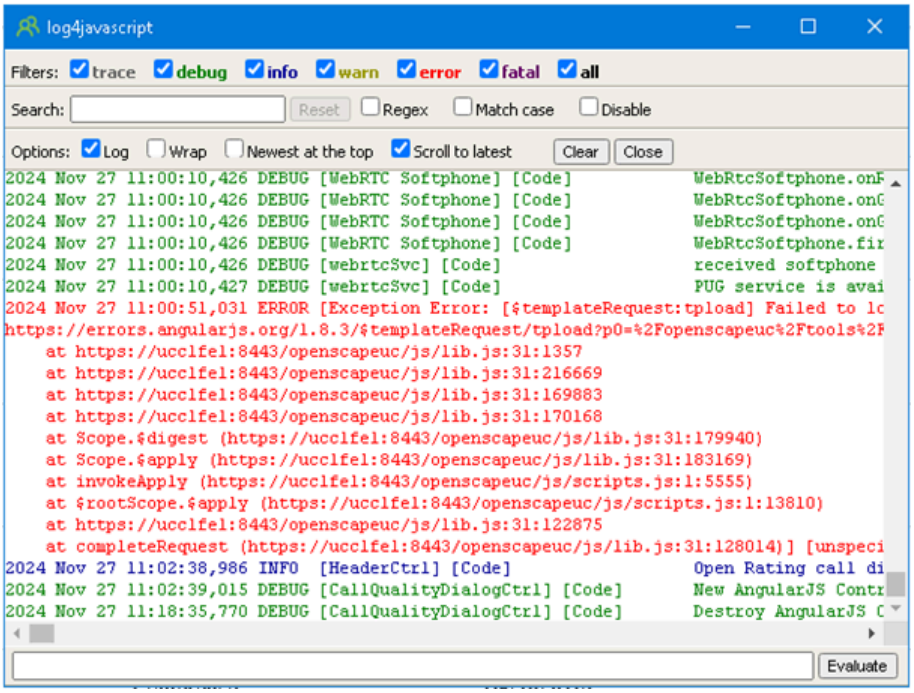
1) Open the UC Desktop App.
2) Open the debugging menu by simultaneously pressing the `Ctrl` key and clicking on your username in the top right corner.

   A drop-down menu opens and the following debug options are displayed:

   a) **Save log file** - exports the log file directly and opens the target folder on the client PC.
   b) **Open Debugger** - opens the debugger window.
   c) **WebRTC Debug Options** - opens an additional dialog window with several configurable options.
   d) **Atmosphere Logging** - provides a Loglevel which will be required by support team (GVS/DEV) in case of issues.
   e) **Clear local data** - clears all data retrieved from the UC server and reloads the DesktopApp.
   f) **Open simulator** - tool owned by DEV team.
   g) **Please rate the quality of your call** - allows you to rate the quality of the last call.
3) To use the debugger, click **Open Debugger**.

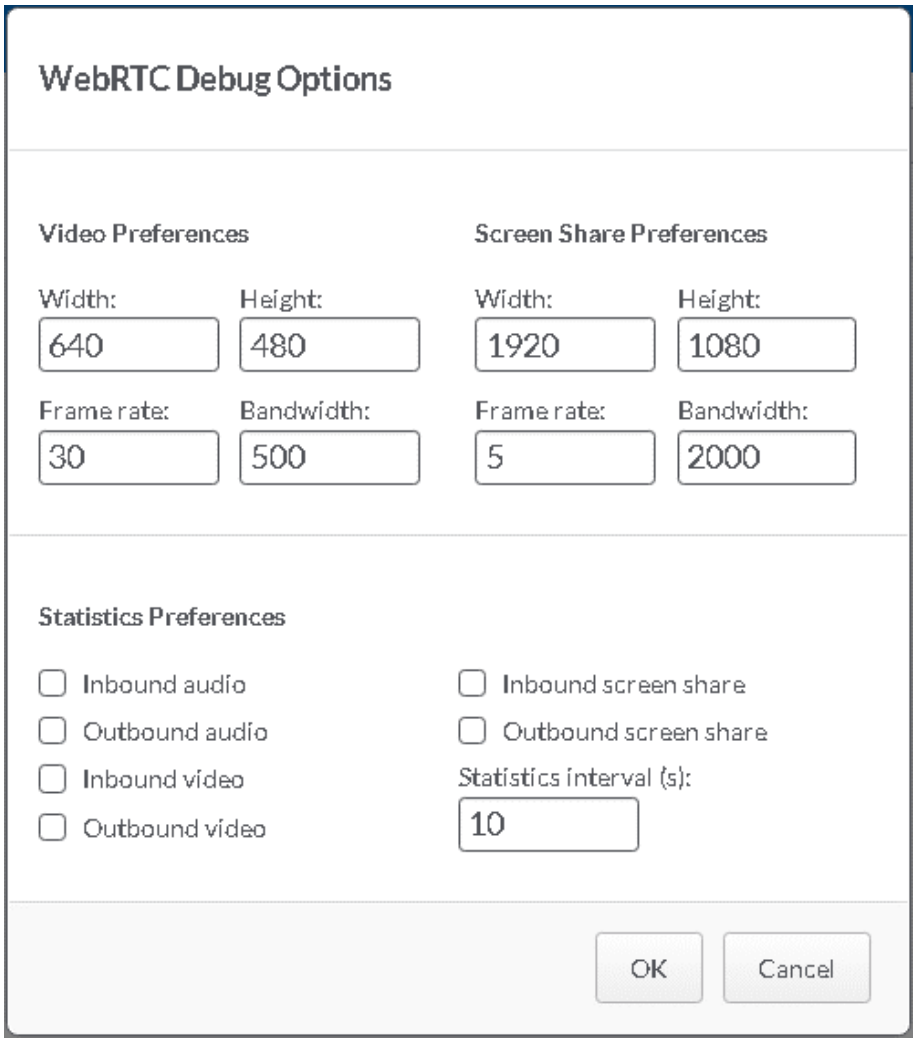
   The Debugger window opens and the contents of the current logfile are displayed, depending on which filters are active. You can also use the

Debugger window to export the logfile. For this, select all contents of the logfile and copy them to a text editor.



4) To use the debug options for WebRTC, click **WebRTC Debug Options**.

The **WebRTC Debug Options** window opens and you can configure the settings for video, screen share and statistics. A configuration example for the debug options preferences is displayed in the image below:



12 OpenScape Mobile related information

12.1 WebEntry Server OSMO Pro required configuration

Several customers use WebEntry Server as application firewall.

The following configuration should exist in WebEntry Servers in order for OSMO Pro to work:

+ SSLProxyEngine On

+ SSLProxyVerify none

+ SSLProxyCheckPeerCN off

+ SSLProxyCheckPeerName off + SSLProxyCheckPeerExpire off enable "**Set-Cookie**" for "**OpenScapeUC**" and "**JSESSIONID**" all traffic after **"OSMO url"/osc-servlets/*** should be allowed.

12.2 REST API

The base URL that used for REST API requests is `http(s)://<domain>/owc-servlets/`

All HTTP status response coded that returned at the requested URLs should send to the client as is without any change in the HTTP status codes. Because this can cause application not able to work properly.

12.3 Authentication

For authentication the required cookies are the ones with names:

- OpenScapeUC
- JSESSIONID

12.4 Eventing

For proper application operation it is also required to be able to establish a WebSocket connection at:

```
ws(s)://<domain>/owc-servlets/ws?
<subscribed-topics>
```

This is a connection that requires support of `101 Switching Protocols`

12.5 Secure Eventing

For secure eventing SSL connections it is required to have proper wright permissions in the application install location otherwise the following error appears:

```
E OSMO : [CertificateUtils] saveCertificate - Error writing
 certificate
```

This issue was noticed in an MDM environment of managed iOS devices.

12.6 How to collect OpenScape Mobile (OSMO) diagnostic data

In case of issues related to the OpenScape Mobile application, the following client logs must be collected:

- **OSMO, iOS:**

  On the Advanced settings screen, set the log level to **max**.

  Then, follow the steps below to send the log files to the support team directly from the OpenScape Mobile application:

  1) Open the OpenScape Mobile application and navigate to **Settings** > **Advanced** > **Log File Management**.
  2) Press the menu button, then tap **Send**.
- **OSMO, android:**

  Before capturing the scenario, navigate to the Settings area of the OpenScape Mobile app, set the log level to **max** and clear the logs.

  Log out the user you want to perform the scenario with from the OSMO application and kill the app from system task manager.

  Start the OSMO application again, log in with your user and perform the scenario.

  Once the scenario is completed, navigate to the Settings area of the OpenScape Mobile app and tap **Send log with e-mail**.

To send the logs, you can either use the default `To` email address that your system administrator has configured on your system, or replace it with another one of your choice.