Kubernetes mentés, helyreállítás

Dokumentum kelte:	2026.07.09
Kubernetes verzió:	1.34

Control plane

Control plane helyreállítása előtt győződjünk meg arról, hogy a mentések rendelkezésre állnak. A gépek konfigurációját tartalmazó mentés csak azon a gépen állítható helyre, ahol készült!

Konfiguráció mentése

A mentés az alábbi állományokat tartalmazza:

- /etc/cni
- /etc/containerd
- /etc/kubernetes/manifests/*.yaml
- /etc/kubernetes/pki
- /etc/kubernetes/*.conf
- /root/kubeadm-config.yaml
- /var/lib/kubelet/pki
- /var/lib/kubelet/config.yaml
- /var/lib/kubelet/instance-config.yaml
- /var/lib/kubelet/kubeadm-flags.env

A `/var/lib/kubelet/kubeadm-flags.env` fájlban `kubernetes v1.35` verziótól nincs inicializálási paraméter, ezért kihagyható a mentésből.

A mentést `root` felhasználóként kell futtatni. az alábbi példa napi mentést készít a helyi gép /var/backup alkönvtárába.

```
# mkdir -p /var/backup && tar -czvf /var/backup/k8s-config-$(hostname)-
$(date +'%Y%m%d').tar.gz /etc/cni /etc/containerd
/etc/kubernetes/manifests/*.yaml /etc/kubernetes/pki /etc/kubernetes/*.conf
/root/kubeadm-config.yaml /var/lib/kubelet/pki /var/lib/kubelet/config.yaml
/var/lib/kubelet/instance-config.yaml /var/lib/kubeadm-flags.env
```

ETCD mentés

Az ETCD adatbázisáról egyetlen control plane gépen kell snapshot-ot készíteni. A snapshot `root` felhasználóként az `etcdctl` utasítással készíthető el. A snapshot készítése után visszaállítási tesztet kell végezni. Az ETCD 3.6-os verziójától kezdve a helyreállítási műveleteket az `etcdutl`utasítással tudjuk végrehajtani.

```
# etcdctl version
etcdctl version: 3.6.12
```

```
API version: 3.6

# etcdutl version
etcdutl version: 3.6.12
API version: 3.6
```

```
# mkdir -p /var/backup/ && etcdctl snapshot save /var/backup/etcd-snapshot-
$(hostname)-$(date +'%Y%m%d').db --endpoints=https://127.0.0.1:2379 --
cacert=/etc/kubernetes/pki/etcd/ca.crt --
cert=/etc/kubernetes/pki/etcd/server.crt --
key=/etc/kubernetes/pki/etcd/server.key
...
Snapshot saved at /var/backup/etcd-snapshot-HOSTNAME-YYYYMMDD.db
Server version 3.6.0
```

Snapshot állapot ellenőrzése.

```
# etcdutl snapshot status /var/backup/etcd-snapshot-HOSTNAME-YYYYMMDD.db -w
table
+----------+----------+------------+------------+---------+
|   HASH   | REVISION | TOTAL KEYS | TOTAL SIZE | VERSION |
+----------+----------+------------+------------+---------+
| aac67e22 | 18385870 |        516 |      24 MB |   3.6.0 |
+----------+----------+------------+------------+---------+
```

Egy átmeneti könyvtárban helyreállítási tesztet futtatva győződhetünk meg a snapshot állapotáról.

```
# rm -rf /tmp/etcd-restore-test-$(date +'%Y%m%d') && etcdutl snapshot
restore /var/backup/etcd-snapshot-$(hostname)-$(date +'%Y%m%d').db --data-
dir=/tmp/etcd-restore-test-$(date +'%Y%m%d')

# [ $? -eq 0 ] && echo "OK - Sikeres helyreállítás" || echo "FAILED -
Sikertelen helyreállítás"
OK - Sikeres helyreállítás

# rm -rf /tmp/etcd-restore-test-$(date +'%Y%m%d')
```

Állapot ellenőrzése

Az alábbi utasításokkal ellenőrizhető a klaszter állapota.

```
# kubectl get nodes -l node-role.kubernetes.io/control-plane
NAME     STATUS   ROLES           AGE   VERSION
kube01   Ready    control-plane   66d   v1.36.2
kube02   Ready    control-plane   66d   v1.36.2
kube03   Ready    control-plane   66d   v1.36.2
```

Mindhárom node estében érdemes ellenőrizni a működési feltételeket.

```
# for node in kube01 kube02 kube03 ; do echo "### ${node} állapot ###" ;
kubectl describe node ${node} | egrep
'(Ready|MemoryPressure|DiskPressure|PIDPressure|NetworkUnavailable)' | awk
'{print $1": "$2}' ; echo ; done
### kube01 állapot ###
NetworkUnavailable: False
MemoryPressure: False
DiskPressure: False
PIDPressure: False
Ready: True

### kube02 állapot ###
NetworkUnavailable: False
MemoryPressure: False
DiskPressure: False
PIDPressure: False
Ready: True

### kube03 állapot ###
NetworkUnavailable: False
MemoryPressure: False
DiskPressure: False
PIDPressure: False
Ready: True
```

Ellenőrizzük a POD-ok állapotát.

```
# kubectl -n kube-system get pods
NAME                              READY   STATUS    RESTARTS       AGE
coredns-589f44dc88-5ppfq          1/1     Running   7 (46h ago)    66d
coredns-589f44dc88-nmvss          1/1     Running   7 (46h ago)    66d
etcd-kube01                       1/1     Running   14 (47h ago)   66d
etcd-kube02                       1/1     Running   2 (46h ago)    21d
etcd-kube03                       1/1     Running   2 (46h ago)    21d
kube-apiserver-kube01             1/1     Running   31 (47h ago)   21d
kube-apiserver-kube02             1/1     Running   2 (46h ago)    21d
kube-apiserver-kube03             1/1     Running   5 (46h ago)    21d
kube-controller-manager-kube01    1/1     Running   5 (47h ago)    21d
kube-controller-manager-kube02    1/1     Running   7 (46h ago)    21d
kube-controller-manager-kube03    1/1     Running   4 (46h ago)    21d
kube-proxy-5v48r                  1/1     Running   2 (46h ago)    21d
kube-proxy-hj68x                  1/1     Running   2 (47h ago)    21d
kube-proxy-xrk5x                  1/1     Running   2 (46h ago)    21d
kube-scheduler-kube01             1/1     Running   7 (47h ago)    21d
kube-scheduler-kube02             1/1     Running   4 (46h ago)    21d
kube-scheduler-kube03             1/1     Running   5 (46h ago)    21d
metrics-server-55bf4495db-tsgtq   1/1     Running   6 (46h ago)    26d
```

Áttekintő ellenőrzés.

```
# kubectl get --raw='/readyz?verbose'
[+]ping ok
[+]log ok
[+]etcd ok
[+]etcd-readiness ok
[+]informer-sync ok
[+]poststarthook/start-apiserver-admission-initializer ok
[+]poststarthook/generic-apiserver-start-informers ok
[+]poststarthook/priority-and-fairness-config-consumer ok
[+]poststarthook/priority-and-fairness-filter ok
[+]poststarthook/storage-object-count-tracker-hook ok
[+]poststarthook/start-apiextensions-informers ok
[+]poststarthook/start-apiextensions-controllers ok
[+]poststarthook/crd-informer-synced ok
[+]poststarthook/start-system-namespaces-controller ok
[+]poststarthook/peer-endpoint-reconciler-controller ok
[+]poststarthook/start-cluster-authentication-info-controller ok
[+]poststarthook/start-kube-apiserver-identity-lease-controller ok
[+]poststarthook/start-kube-apiserver-identity-lease-garbage-collector ok
[+]poststarthook/storage-readiness ok
[+]poststarthook/start-legacy-token-tracking-controller ok
[+]poststarthook/start-service-ip-repair-controllers ok
[+]poststarthook/rbac/bootstrap-roles ok
[+]poststarthook/scheduling/bootstrap-system-priority-classes ok
[+]poststarthook/priority-and-fairness-config-producer ok
[+]poststarthook/bootstrap-controller ok
[+]poststarthook/start-kubernetes-service-cidr-controller ok
[+]poststarthook/aggregator-reload-proxy-client-cert ok
[+]poststarthook/start-kube-aggregator-informers ok
[+]poststarthook/apiservice-status-local-available-controller ok
[+]poststarthook/apiservice-status-remote-available-controller ok
[+]poststarthook/apiservice-registration-controller ok
[+]poststarthook/apiservice-discovery-controller ok
[+]poststarthook/kube-apiserver-autoregistration ok
[+]autoregister-completion ok
[+]poststarthook/apiservice-openapi-controller ok
[+]poststarthook/apiservice-openapiv3-controller ok
[+]shutdown ok
readyz check passed
```

ETCD tagok listája.

```
# etcdctl --
endpoints=https://kube01:2379,https://kube02:2379,https://kube03:2379 --
cacert=/etc/kubernetes/pki/etcd/ca.crt --
cert=/etc/kubernetes/pki/etcd/server.crt --
key=/etc/kubernetes/pki/etcd/server.key member list -w table
+------------------+---------+--------+-----------------------------+------
-----------------------+------------+
|        ID        | STATUS  |  NAME  |          PEER ADDRS         |
```

```
CLIENT ADDRS         | IS LEARNER |
+------------------+---------+--------+-----------------------------+------
-----------------------+------------+
| 1546a2d2ac61bd8d | started | kube02 | https://192.168.110.162:2380 |
https://192.168.110.162:2379 |      false |
| 18e2fb4d02a30b2e | started | kube03 | https://192.168.110.163:2380 |
https://192.168.110.163:2379 |      false |
| 20e2ce3086b2e277 | started | kube01 | https://192.168.110.161:2380 |
https://192.168.110.161:2379 |      false |
+------------------+---------+--------+-----------------------------+------
-----------------------+------------+
```

ETCD endpoint állapot.

```
# etcdctl --
endpoints=https://kube01:2379,https://kube02:2379,https://kube03:2379 --
cacert=/etc/kubernetes/pki/etcd/ca.crt --
cert=/etc/kubernetes/pki/etcd/server.crt --
key=/etc/kubernetes/pki/etcd/server.key endpoint status -w table
+---------------------+------------------+---------+-----------------+------
---+--------+-----------------------+--------+-----------+------------+-----
------+------------+--------------------+--------+-------------------------
+-------------------+
|      ENDPOINT       |        ID        | VERSION | STORAGE VERSION | DB
SIZE | IN USE | PERCENTAGE NOT IN USE | QUOTA  | IS LEADER | IS LEARNER |
RAFT TERM | RAFT INDEX | RAFT APPLIED INDEX | ERRORS | DOWNGRADE TARGET
VERSION | DOWNGRADE ENABLED |
+---------------------+------------------+---------+-----------------+------
---+--------+-----------------------+--------+-----------+------------+-----
------+------------+--------------------+--------+-------------------------
+-------------------+
| https://kube01:2379 | 20e2ce3086b2e277 |   3.6.8 |           3.6.0 |   24
MB | 7.2 MB |                   71% | 2.1 GB |     false |      false |
4 |    6990447 |           6990447 |        |                           |
false |
| https://kube02:2379 | 1546a2d2ac61bd8d |   3.6.8 |           3.6.0 |   24
MB | 7.2 MB |                   71% | 2.1 GB |      true |      false |
4 |    6990447 |           6990447 |        |                           |
false |
| https://kube03:2379 | 18e2fb4d02a30b2e |   3.6.8 |           3.6.0 |   24
MB | 7.2 MB |                   71% | 2.1 GB |     false |      false |
4 |    6990447 |           6990447 |        |                           |
false |
+---------------------+------------------+---------+-----------------+------
---+--------+-----------------------+--------+-----------+------------+-----
------+------------+--------------------+--------+-------------------------
+-------------------+
```

Tanúsítványok érvényességének vizsgálata.

```
# kubeadm certs check-expiration
```

```
[check-expiration] Reading configuration from the "kubeadm-config" ConfigMap
in namespace "kube-system"...
[check-expiration] Use 'kubeadm init phase upload-config kubeadm --config
your-config-file' to re-upload it.

CERTIFICATE                EXPIRES                  RESIDUAL TIME
CERTIFICATE AUTHORITY   EXTERNALLY MANAGED
admin.conf                 Feb 15, 2126 06:54 UTC   99y             ca
no
apiserver                  Feb 15, 2126 06:54 UTC   99y             ca
no
apiserver-etcd-client      Feb 15, 2126 06:54 UTC   99y             etcd-ca
no
apiserver-kubelet-client   Feb 15, 2126 06:54 UTC   99y             ca
no
controller-manager.conf    Feb 15, 2126 06:54 UTC   99y             ca
no
etcd-healthcheck-client    Feb 15, 2126 06:54 UTC   99y             etcd-ca
no
etcd-peer                  Feb 15, 2126 06:54 UTC   99y             etcd-ca
no
etcd-server                Feb 15, 2126 06:54 UTC   99y             etcd-ca
no
front-proxy-client         Feb 15, 2126 06:54 UTC   99y             front-
proxy-ca          no
scheduler.conf             Feb 15, 2126 06:54 UTC   99y             ca
no
super-admin.conf           Feb 15, 2126 06:54 UTC   99y             ca
no

CERTIFICATE AUTHORITY   EXPIRES                  RESIDUAL TIME   EXTERNALLY
MANAGED
ca                      Sep 01, 2225 09:27 UTC   199y            no
etcd-ca                 Sep 01, 2225 09:27 UTC   199y            no
front-proxy-ca          Sep 01, 2225 09:27 UTC   199y            no
```

Amennyiben a fenti ellenőrzések sikeresek, akkor a klaszter control-plane gépei egészségesek. Nincs további teendő.

Egy control plane kiesés

Egy control-plane gép kiesését a klaszter probléma nélkül elviseli. Amennyiben a kiesés az ETCD leader-t érinti, akkor a két megmaradt tagból az egyik átveszi a leader szerepet.

A leader szerepkörű ETCD tagot az alábbi utasítással kaphatjuk meg.

```
# etcdctl --
endpoints=https://kube01:2379,https://kube02:2379,https://kube03:2379 --
cacert=/etc/kubernetes/pki/etcd/ca.crt --
```

```
cert=/etc/kubernetes/pki/etcd/server.crt --
key=/etc/kubernetes/pki/etcd/server.key endpoint status -w json | jq -r '.[]
| select(.Status.leader == .Status.header.member_id) | .Endpoint'

https://kube02:2379
```

A leader kiesésekor az ETCD ellenőrző utasítás nem tud kapcsolódni a korábbi leader-hez. A két működő tag új leader-t választ.

```
# etcdctl --
endpoints=https://kube01:2379,https://kube02:2379,https://kube03:2379 --
cacert=/etc/kubernetes/pki/etcd/ca.crt --
cert=/etc/kubernetes/pki/etcd/server.crt --
key=/etc/kubernetes/pki/etcd/server.key endpoint status -w json | jq -r '.[]
| select(.Status.leader == .Status.header.member_id) | .Endpoint'
{"level":"warn","ts":"2026-07-09T09:24:07.518541+0200","logger":"etcd-
client","caller":"v3@v3.6.12/retry_interceptor.go:68","msg":"retrying of
unary invoker failed","target":"etcd-
endpoints://0xc0002103c0/kube02:2379","peer":"Peer{Addr: <nil>, LocalAddr:
<nil>, AuthInfo:
<nil>}","method":"/etcdserverpb.Maintenance/Status","attempt":0,"error":"rpc
error: code = DeadlineExceeded desc = latest balancer error: connection
error: desc = \"transport: Error while dialing: dial tcp
192.168.110.162:2379: connect: connection refused\""}
Failed to get the status of endpoint https://kube02:2379 (context deadline
exceeded)

https://kube01:2379
```

A gépek listázásakor egy idő után a hiányzó gép NotReady állapotba kerül.

```
# kubectl get nodes -l node-role.kubernetes.io/control-plane
NAME     STATUS     ROLES           AGE   VERSION
kube01   Ready      control-plane   66d   v1.36.2
kube02   NotReady   control-plane   66d   v1.36.2
kube03   Ready      control-plane   66d   v1.36.2
```

ETCD endpoint állapot ellenőrzésekor hiányzik a kiesett tag.

```
# etcdctl --
endpoints=https://kube01:2379,https://kube02:2379,https://kube03:2379 --
cacert=/etc/kubernetes/pki/etcd/ca.crt --
cert=/etc/kubernetes/pki/etcd/server.crt --
key=/etc/kubernetes/pki/etcd/server.key endpoint status -w table
{"level":"warn","ts":"2026-07-09T09:34:48.926554+0200","logger":"etcd-
client","caller":"v3@v3.6.12/retry_interceptor.go:68","msg":"retrying of
unary invoker failed","target":"etcd-
endpoints://0xc00027f860/kube02:2379","peer":"Peer{Addr: <nil>, LocalAddr:
<nil>, AuthInfo:
<nil>}","method":"/etcdserverpb.Maintenance/Status","attempt":0,"error":"rpc
error: code = DeadlineExceeded desc = context deadline exceeded while
```

```
waiting for connections to become ready"}
Failed to get the status of endpoint https://kube02:2379 (context deadline
exceeded)
+---------------------+------------------+---------+-----------------+------
---+--------+----------------------+--------+-----------+------------+-----
------+------------+-------------------+--------+-------------------------
+-------------------+
|      ENDPOINT       |        ID        | VERSION | STORAGE VERSION | DB
SIZE | IN USE | PERCENTAGE NOT IN USE | QUOTA  | IS LEADER | IS LEARNER |
RAFT TERM | RAFT INDEX | RAFT APPLIED INDEX | ERRORS | DOWNGRADE TARGET
VERSION | DOWNGRADE ENABLED |
+---------------------+------------------+---------+-----------------+------
---+--------+----------------------+--------+-----------+------------+-----
------+------------+-------------------+--------+-------------------------
+-------------------+
| https://kube01:2379 | 20e2ce3086b2e277 |   3.6.8 |           3.6.0 |   24
MB | 8.0 MB |                   68% | 2.1 GB |      true |      false |
5 |    6996663 |            6996663 |        |                           |
false |
| https://kube03:2379 | 18e2fb4d02a30b2e |   3.6.8 |           3.6.0 |   24
MB | 8.0 MB |                   67% | 2.1 GB |     false |      false |
5 |    6996681 |            6996678 |        |                           |
false |
+---------------------+------------------+---------+-----------------+------
---+--------+----------------------+--------+-----------+------------+-----
------+------------+-------------------+--------+-------------------------
+-------------------+
```

Rövid idejű kiesés

Rövid ideig tartó kiesés esetén a hiba megszűnése után érdemes ellenőrizni az állapotokat.

Gépek listája.

```
# kubectl get nodes -l node-role.kubernetes.io/control-plane
NAME     STATUS   ROLES           AGE   VERSION
kube01   Ready    control-plane   66d   v1.36.2
kube02   Ready    control-plane   66d   v1.36.2
kube03   Ready    control-plane   66d   v1.36.2
```

ETCD endpoint állapot.

```
# etcdctl --
endpoints=https://kube01:2379,https://kube02:2379,https://kube03:2379 --
cacert=/etc/kubernetes/pki/etcd/ca.crt --
cert=/etc/kubernetes/pki/etcd/server.crt --
key=/etc/kubernetes/pki/etcd/server.key endpoint status -w table
+---------------------+------------------+---------+-----------------+------
---+--------+----------------------+--------+-----------+------------+-----
```

```
------+------------+--------------------+--------+-------------------------
+------------------+
|      ENDPOINT       |        ID        | VERSION | STORAGE VERSION | DB
SIZE | IN USE | PERCENTAGE NOT IN USE | QUOTA  | IS LEADER | IS LEARNER |
RAFT TERM | RAFT INDEX | RAFT APPLIED INDEX | ERRORS | DOWNGRADE TARGET
VERSION | DOWNGRADE ENABLED |
+---------------------+------------------+---------+-----------------+------
---+--------+----------------------+--------+-----------+------------+-----
------+------------+--------------------+--------+-------------------------
+------------------+
| https://kube01:2379 | 20e2ce3086b2e277 |   3.6.8 |           3.6.0 |   24
MB | 8.4 MB |                   66% | 2.1 GB |      true |      false |
5 |    6996862 |            6996862 |        |                          |
false |
| https://kube02:2379 | 1546a2d2ac61bd8d |   3.6.8 |           3.6.0 |   24
MB | 8.0 MB |                   68% | 2.1 GB |     false |      false |
5 |    6996863 |            6996861 |        |                          |
false |
| https://kube03:2379 | 18e2fb4d02a30b2e |   3.6.8 |           3.6.0 |   24
MB | 8.3 MB |                   66% | 2.1 GB |     false |      false |
5 |    6996864 |            6996863 |        |                          |
false |
+---------------------+------------------+---------+-----------------+------
---+--------+----------------------+--------+-----------+------------+-----
------+------------+--------------------+--------+-------------------------
+------------------+
```

Újabb ellenőrzéssel meggyőződhetünk a RAFT indexek állapotáról.

etcdctl –endpoints=https://kube01:2379,https://kube02:2379,https://kube03:2379 –cacert=/etc/kubernetes/pki/etcd/ca.crt –cert=/etc/kubernetes/pki/etcd/server.crt –key=/etc/kubernetes/pki/etcd/server.key endpoint status -w table

ENDPOINT	ID	VERSION	STORAGE VERSION	DB SIZE	IN USE	PERCENTAGE NOT IN USE	QUOTA	IS LEADER	IS LEARNER	RAFT TERM	RAFT INDEX	RAFT APPLIED INDEX	ERRORS	DOWNGRADE TARGET VERSION	DOWNGRADE ENABLED
https://kube01:2379	20e2ce3086b2e277	3.6.8	3.6.0	24 MB	9.8 MB	60%	2.1 GB	true	false	5	6998333	6998333			false
https://kube02:2379	1546a2d2ac61bd8d	3.6.8	3.6.0	24 MB	9.6 MB	61%	2.1 GB	false	false	5	6998333	6998333			false
https://kube03:2379	18e2fb4d02a30b2e	3.6.8	3.6.0	24 MB	9.8 MB	60%	2.1 GB	false	false	5	6998333	6998333			false

Tartós kiesés

Egy control-plane node tartós kiesése esetén az ETCD tömöríti a RAFT naplókat, így azok alkalmazására már nincs lehetőség. Ilyenkor az alábbi lehetőségek vannak.

Automatikus helyreállás

Ebben az esetben automatikus helyreállás történik egy friss leader snapshot elküldésével. Ellenőrzéseken kívül nincs további teendő.

Helyreállítás ugyanazon a gépen

A legtisztább eljárás, hogy a két megmaradó gép egyikén töröljük az ETCD tagok és a gépek listájából a gépet. Fontos, hogy a törlegdő gépen ne fusson a `containerd` és a `kubelet` szolgáltatás valamint a kuberneteshez kapcsolódó statikus pod-ok sem. Ezt legegyszerűbben a gép leállításval tudjuk megoldani.

Ellenőrizzük az ETCD tagok listáját.

```
# etcdctl --endpoints=https://127.0.0.1:2379 --
cacert=/etc/kubernetes/pki/etcd/ca.crt --
cert=/etc/kubernetes/pki/etcd/peer.crt --
key=/etc/kubernetes/pki/etcd/peer.key member list -w table
+------------------+---------+--------+-----------------------------+------
------------------------+------------+
|        ID        | STATUS  |  NAME  |          PEER ADDRS         |
CLIENT ADDRS         | IS LEARNER |
+------------------+---------+--------+-----------------------------+------
------------------------+------------+
| 1546a2d2ac61bd8d | started | kube02 | https://192.168.110.162:2380 |
https://192.168.110.162:2379 |      false |
| 18e2fb4d02a30b2e | started | kube03 | https://192.168.110.163:2380 |
https://192.168.110.163:2379 |      false |
| 20e2ce3086b2e277 | started | kube01 | https://192.168.110.161:2380 |
https://192.168.110.161:2379 |      false |
+------------------+---------+--------+-----------------------------+------
------------------------+------------+
```

Távolítsuk el a hibás tagot az azonosító megadásával (ID mező).

```
# etcdctl --endpoints=https://127.0.0.1:2379 --
cacert=/etc/kubernetes/pki/etcd/ca.crt --
cert=/etc/kubernetes/pki/etcd/peer.crt --
key=/etc/kubernetes/pki/etcd/peer.key member remove 18e2fb4d02a30b2e
Member 18e2fb4d02a30b2e removed from cluster adeb3c2ca3243377
```

Újra ellenőrizzük az ETCD tagok listáját.

```
# etcdctl --endpoints=https://127.0.0.1:2379 --
cacert=/etc/kubernetes/pki/etcd/ca.crt --
cert=/etc/kubernetes/pki/etcd/peer.crt --
key=/etc/kubernetes/pki/etcd/peer.key member list -w table
+------------------+---------+--------+-----------------------------+------
------------------------+------------+
|        ID        | STATUS  |  NAME  |          PEER ADDRS         |
CLIENT ADDRS         | IS LEARNER |
+------------------+---------+--------+-----------------------------+------
------------------------+------------+
| 1546a2d2ac61bd8d | started | kube02 | https://192.168.110.162:2380 |
https://192.168.110.162:2379 |      false |
| 20e2ce3086b2e277 | started | kube01 | https://192.168.110.161:2380 |
https://192.168.110.161:2379 |      false |
+------------------+---------+--------+-----------------------------+------
------------------------+------------+
```

Ellenőrizzük a kubernetes gépek listáját.

```
# kubectl get nodes -l node-role.kubernetes.io/control-plane
NAME     STATUS     ROLES           AGE   VERSION
kube01   Ready      control-plane   66d   v1.36.2
kube02   Ready      control-plane   66d   v1.36.2
kube03   NotReady   control-plane   66d   v1.36.2
```

Töröljük a hibás gépet.

```
# kubectl delete node kube03
node "kube03" deleted
```

Ellenőrizzük a gépek listáját.

```
# kubectl get nodes
NAME     STATUS   ROLES           AGE   VERSION
kube01   Ready    control-plane   66d   v1.36.2
kube02   Ready    control-plane   66d   v1.36.2
```

Generáljunk új tokent a gép csatlakoztatásához.

```
# kubeadm token create --print-join-command
kubeadm join 192.168.110.160:6443 --token gef9jk.oafnicbu6hgwhhy7 --
discovery-token-ca-cert-hash
sha256:838e04036d629fbf3e152fc6606bf9a143f8e1ca0b0bc00d747fe1caba5df8f1

# kubeadm init phase upload-certs --upload-certs
[upload-certs] Storing the certificates in Secret "kubeadm-certs" in the
"kube-system" Namespace
[upload-certs] Using certificate key:
7aa4b634e4f0986669dbd19b972ec3c14a80e3ab1909a54d26ca58b0b7ec15ac
```

A snapshot-ból helyreállított gépet indítsuk el oly módon, hogy a `containerd` és a `kubelet` szolgáltatások biztosan **NE** induljanak el. Ezt a legegyszerűbben a gép boot folyamatának módosításával érjük el. A GRUB menüjében a kernel paraméterezését egészítsük ki az alábbiak szerint.

`linux ….. **systemd.mask=containerd.service systemd.mask=kubelet.service**`

Az elindult rendszerben tudjuk ellenőrizni.

```
# systemctl status containerd kubelet
○ containerd.service
     Loaded: masked (Reason: Unit containerd.service is masked.)
     Active: inactive (dead)

○ kubelet.service
     Loaded: masked (Reason: Unit kubelet.service is masked.)
    Drop-In: /usr/lib/systemd/system/kubelet.service.d
             └─10-kubeadm.conf
     Active: inactive (dead)
```

Amennyiben snapshot-ból állt helyre a gép, futtassuk a `kubeadm reset` utasítást. Ez törölni fogja a korábbi klaszter állományait.

```
# kubeadm reset
[reset] Reading configuration from the "kubeadm-config" ConfigMap in
namespace "kube-system"...
[reset] Use 'kubeadm init phase upload-config kubeadm --config your-config-
file' to re-upload it.
...
The reset process does not perform cleanup of CNI plugin configuration,
network filtering rules and kubeconfig files.

For information on how to perform this cleanup manually, please see:
    https://k8s.io/docs/reference/setup-tools/kubeadm/kubeadm-reset/
```

Töröljük a felesleges egyéb állományokat.

```
# rm -rf /etc/cni/net.d/* /root/.kube /var/lib/cni /var/lib/etcd
/var/lib/kubelet
```

Indítsuk újra a gépet.

```
# reboot
```

A gép újraindulása után csatlakoztassuk a gépet control-plane szerepkörben a meglevő klaszterhez.

```
# kubeadm join 192.168.110.160:6443 --token gef9jk.oafnicbu6hgwhhy7 --
discovery-token-ca-cert-hash
sha256:838e04036d629fbf3e152fc6606bf9a143f8e1ca0b0bc00d747fe1caba5df8f1 --
control-plane --certificate-key
```

```
7aa4b634e4f0986669dbd19b972ec3c14a80e3ab1909a54d26ca58b0b7ec15ac
[preflight] Running pre-flight checks
[preflight] Reading configuration from the "kubeadm-config" ConfigMap in
namespace "kube-system"...
[preflight] Use 'kubeadm init phase upload-config kubeadm --config your-
config-file' to re-upload it.
...

This node has joined the cluster and a new control plane instance was
created:

* Certificate signing request was sent to apiserver and approval was
received.
* The Kubelet was informed of the new secure connection details.
* Control plane label and taint were applied to the new node.
* The Kubernetes control plane instances scaled up.
* A new etcd member was added to the local/stacked etcd cluster.

To start administering your cluster from this node, you need to run the
following as a regular user:

    mkdir -p $HOME/.kube
    sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
    sudo chown $(id -u):$(id -g) $HOME/.kube/config

Run 'kubectl get nodes' to see this node join the cluster.
```

Állítsuk be a kubernetes konfigurációt a root felhasználónak.

```
# rm -rf /root/.kube/
# mkdir -p $HOME/.kube
# sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
# sudo chown $(id -u):$(id -g) $HOME/.kube/config
```

Ellenőrizzük a klaszter állapotát.

```
# kubectl get nodes
NAME     STATUS   ROLES           AGE     VERSION
kube01   Ready    control-plane   67d     v1.36.2
kube02   Ready    control-plane   67d     v1.36.2
kube03   Ready    control-plane   5m21s   v1.36.2
```

Ellenőrizzük az ETCD tagok listáját.

```
# etcdctl --endpoints=https://127.0.0.1:2379 --
cacert=/etc/kubernetes/pki/etcd/ca.crt --
cert=/etc/kubernetes/pki/etcd/peer.crt --
key=/etc/kubernetes/pki/etcd/peer.key member list -w table
+------------------+---------+--------+-----------------------------+------
-----------------------+------------+
|        ID        | STATUS  |  NAME  |          PEER ADDRS         |
```

```
CLIENT ADDRS         | IS LEARNER |
+------------------+---------+--------+-----------------------------+------
-----------------------+------------+
| 1546a2d2ac61bd8d | started | kube02 | https://192.168.110.162:2380 |
https://192.168.110.162:2379 |      false |
| 20e2ce3086b2e277 | started | kube01 | https://192.168.110.161:2380 |
https://192.168.110.161:2379 |      false |
| 23fb75a79460e3c9 | started | kube03 | https://192.168.110.163:2380 |
https://192.168.110.163:2379 |      false |
+------------------+---------+--------+-----------------------------+------
-----------------------+------------+
```

Ellenőrizzük az ETCD endpointok állapotát.

```
# etcdctl --
endpoints=https://kube01:2379,https://kube02:2379,https://kube03:2379 --
cacert=/etc/kubernetes/pki/etcd/ca.crt --
cert=/etc/kubernetes/pki/etcd/server.crt --
key=/etc/kubernetes/pki/etcd/server.key endpoint status -w table
+----------------------+------------------+---------+-----------------+------
---+--------+-----------------------+--------+-----------+------------+-----
------+------------+--------------------+--------+-------------------------
+-------------------+
|      ENDPOINT       |        ID        | VERSION | STORAGE VERSION | DB
SIZE | IN USE | PERCENTAGE NOT IN USE | QUOTA  | IS LEADER | IS LEARNER |
RAFT TERM | RAFT INDEX | RAFT APPLIED INDEX | ERRORS | DOWNGRADE TARGET
VERSION | DOWNGRADE ENABLED |
+----------------------+------------------+---------+-----------------+------
---+--------+-----------------------+--------+-----------+------------+-----
------+------------+--------------------+--------+-------------------------
+-------------------+
| https://kube01:2379 | 20e2ce3086b2e277 |   3.6.8 |           3.6.0 |   24
MB | 7.5 MB |                   70% | 2.1 GB |      true |      false |
5 |    7031342 |           7031342 |        |                           |
false |
| https://kube02:2379 | 1546a2d2ac61bd8d |   3.6.8 |           3.6.0 |   24
MB | 7.5 MB |                   70% | 2.1 GB |     false |      false |
5 |    7031343 |           7031343 |        |                           |
false |
| https://kube03:2379 | 23fb75a79460e3c9 |   3.6.8 |           3.6.0 |   24
MB | 7.5 MB |                   70% | 2.1 GB |     false |      false |
5 |    7031343 |           7031343 |        |                           |
false |
+----------------------+------------------+---------+-----------------+------
---+--------+-----------------------+--------+-----------+------------+-----
------+------------+--------------------+--------+-------------------------
+-------------------+
```

A helyreállítás sikeres.

Két control plane kiesés

Két control-plane kiesése esetén a kunernetes klaszter elveszíti a kvórumot. Ebben a helyzetben a worker gépeken futó pod-ok a legutolsó állapotukban maradnak. Nem működik a HPA. Nem lehet módosítani a konfiguráción. Ilyen esetben a control-plane node-ok mielőbbi helyreállítása szükséges.

Három control plane kiesés

Worker

ETCD környezet helyreállítása mentésből

A környezet helyreállítása előtt győződjünk meg arról, hogy minden control-plane gépen telepítve van az etcdctl és az etcdutl program.

```
# etcdctl version
etcdctl version: 3.6.12
API version: 3.6

# etcdutl version
etcdutl version: 3.6.12
API version: 3.6
```

Megjegyzés: 3.5-ös `etcdctl` verzió az utolsó olyan változat, amellyel a helyreállítás elvégezhető. 3.6-os verziótól a helyreállítási műveleteket az `etcdutl` utasítással kell elvégezni.

Mentés állapotának ellenőrzése

Állapot gyors ellenőrzése

```
# etcdutl snapshot status /root/etcd-snapshot-20260606.db -w table
+----------+----------+------------+------------+---------+
|   HASH   | REVISION | TOTAL KEYS | TOTAL SIZE | VERSION |
+----------+----------+------------+------------+---------+
| e1ac0dbb | 17572986 |        749 |      24 MB |   3.6.0 |
+----------+----------+------------+------------+---------+
```

Helyreállítási teszt

Győződjünk meg a mentés helyreállíthatóságáról. A mentést valamelyik control-plane gépen egy átmeneti könyvtárban állítsuk helyre.

```
# rm -rf /tmp/etcd-restore-test
```

```
etcdutl snapshot restore /root/etcd-snapshot-20260606.db --data-
dir=/tmp/etcd-restore-test
2026-07-06T08:54:44+02:00   info    snapshot/v3_snapshot.go:305    restoring
snapshot    {"path": "/root/etcd-snapshot-20260606.db", "wal-dir":
"/tmp/etcd-restore-test/member/wal", "data-dir": "/tmp/etcd-restore-test",
"snap-dir": "/tmp/etcd-restore-test/member/snap", "initial-memory-map-size":
10737418240}
2026-07-06T08:54:44+02:00   info    bbolt    backend/backend.go:203
Opening db file (/tmp/etcd-restore-test/member/snap/db) with mode -rw-------
and with options: {Timeout: 0s, NoGrowSync: false, NoFreelistSync: true,
PreLoadFreelist: false, FreelistType: , ReadOnly: false, MmapFlags: 8000,
InitialMmapSize: 10737418240, PageSize: 0, NoSync: false, OpenFile: 0x0,
Mlock: false, Logger: 0xc000304008}
2026-07-06T08:54:44+02:00   info    bbolt    bbolt@v1.4.3/db.go:322
Opening bbolt db (/tmp/etcd-restore-test/member/snap/db) successfully
2026-07-06T08:54:44+02:00   info    schema/membership.go:138    Trimming
membership information from the backend...
2026-07-06T08:54:44+02:00   info    bbolt    backend/backend.go:203
Opening db file (/tmp/etcd-restore-test/member/snap/db) with mode -rw-------
and with options: {Timeout: 0s, NoGrowSync: false, NoFreelistSync: true,
PreLoadFreelist: false, FreelistType: , ReadOnly: false, MmapFlags: 8000,
InitialMmapSize: 10737418240, PageSize: 0, NoSync: false, OpenFile: 0x0,
Mlock: false, Logger: 0xc0000621e0}
2026-07-06T08:54:44+02:00   info    bbolt    bbolt@v1.4.3/db.go:322
Opening bbolt db (/tmp/etcd-restore-test/member/snap/db) successfully
2026-07-06T08:54:44+02:00   info    membership/cluster.go:424    added
member    {"cluster-id": "cdf818194e3a8c32", "local-member-id": "0", "added-
peer-id": "8e9e05c52164694d", "added-peer-peer-urls":
["http://localhost:2380"], "added-peer-is-learner": false}
2026-07-06T08:54:45+02:00   info    bbolt    backend/backend.go:203
Opening db file (/tmp/etcd-restore-test/member/snap/db) with mode -rw-------
and with options: {Timeout: 0s, NoGrowSync: false, NoFreelistSync: true,
PreLoadFreelist: false, FreelistType: , ReadOnly: false, MmapFlags: 8000,
InitialMmapSize: 10737418240, PageSize: 0, NoSync: false, OpenFile: 0x0,
Mlock: false, Logger: 0xc000062228}
2026-07-06T08:54:45+02:00   info    bbolt    bbolt@v1.4.3/db.go:322
Opening bbolt db (/tmp/etcd-restore-test/member/snap/db) successfully
2026-07-06T08:54:45+02:00   info    snapshot/v3_snapshot.go:333    restored
snapshot    {"path": "/root/etcd-snapshot-20260606.db", "wal-dir":
"/tmp/etcd-restore-test/member/wal", "data-dir": "/tmp/etcd-restore-test",
"snap-dir": "/tmp/etcd-restore-test/member/snap", "initial-memory-map-size":
10737418240}
```

Amennyiben sikeresen lefut a helyreállítás, akkor a teszt sikeres.

Jelenlegi állapot mentése

Kubernetes objektumok állapotának mentése egy control-plane gépen. Erre a helyreállítás után az objektumok állapotának ellenőrzésénél lesz szükség.

```
# mkdir -p /root/k8s-backup
# kubectl get nodes -o wide > /root/k8s-backup/nodes.before.txt
# kubectl get pods -A -o wide > /root/k8s-backup/pods.before.txt
# kubectl get ns > /root/k8s-backup/namespaces.before.txt
# kubectl get cm -A > /root/k8s-backup/configmaps.before.txt
```

Szolgáltatások tiltása és leállítása az összes control-plane gépen.

```
# systemctl disable kubelet containerd
# systemctl stop kubelet containerd
# systemctl reboot
<code>

Korábbi környezet mentése.

<code>
# mkdir -p /root/k8s-backup/pre-restore
# mkdir -p /root/k8s-backup/pre-restore/etc/kubernetes
# mkdir -p /root/k8s-backup/pre-restore/var/lib
# cp -a /etc/kubernetes/manifests /root/k8s-backup/pre-
restore/etc/kubernetes/
# cp -a /etc/kubernetes/pki /root/k8s-backup/pre-restore/etc/kubernetes/
# cp -a /var/lib/etcd /root/k8s-backup/pre-restore/var/lib/
```

From:
https://wiki.r-l.hu/ - **Reverse-Logic wiki**

Permanent link:
https://wiki.r-l.hu/doku.php?id=kubernetes:mentes-helyreallitas&rev=1783591995

Last update: **2026/07/09 10:13**