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.

```
# 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

Amennyiben a klaszter kvórummal rendelkezik, az alábbi utasít

```
# kubectl get nodes -l node-role.kubernetes.io/control-plane
NAME     STATUS   ROLES                    AGE   VERSION
kube01   Ready    control-plane,postgres   66d   v1.36.2
kube02   Ready    control-plane,postgres   66d   v1.36.2
kube03   Ready    control-plane,postgres   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)' ; echo
; done
### kube01 állapot ###
  NetworkUnavailable   False   Tue, 07 Jul 2026 09:42:26 +0200   Tue, 07 Jul
2026 09:42:26 +0200   FlannelIsUp                  Flannel is running on
this node
  MemoryPressure       False   Thu, 09 Jul 2026 08:43:44 +0200   Tue, 07 Jul
```

```
2026 09:41:07 +0200   KubeletHasSufficientMemory   kubelet has sufficient
memory available
  DiskPressure         False   Thu, 09 Jul 2026 08:43:44 +0200   Tue, 07 Jul
2026 09:41:07 +0200   KubeletHasNoDiskPressure     kubelet has no disk
pressure
  PIDPressure          False   Thu, 09 Jul 2026 08:43:44 +0200   Tue, 07 Jul
2026 09:41:07 +0200   KubeletHasSufficientPID      kubelet has sufficient
PID available
  Ready                True    Thu, 09 Jul 2026 08:43:44 +0200   Tue, 07 Jul
2026 09:41:07 +0200   KubeletReady                kubelet is posting ready
status

### kube02 állapot ###
  NetworkUnavailable   False   Tue, 07 Jul 2026 09:53:13 +0200   Tue, 07 Jul
2026 09:53:13 +0200   FlannelIsUp                 Flannel is running on
this node
  MemoryPressure       False   Thu, 09 Jul 2026 08:40:29 +0200   Tue, 07 Jul
2026 09:52:22 +0200   KubeletHasSufficientMemory   kubelet has sufficient
memory available
  DiskPressure         False   Thu, 09 Jul 2026 08:40:29 +0200   Tue, 07 Jul
2026 09:52:22 +0200   KubeletHasNoDiskPressure     kubelet has no disk
pressure
  PIDPressure          False   Thu, 09 Jul 2026 08:40:29 +0200   Tue, 07 Jul
2026 09:52:22 +0200   KubeletHasSufficientPID      kubelet has sufficient
PID available
  Ready                True    Thu, 09 Jul 2026 08:40:29 +0200   Tue, 07 Jul
2026 09:52:22 +0200   KubeletReady                kubelet is posting ready
status

### kube03 állapot ###
  NetworkUnavailable   False   Tue, 07 Jul 2026 10:03:38 +0200   Tue, 07 Jul
2026 10:03:38 +0200   FlannelIsUp                 Flannel is running on
this node
  MemoryPressure       False   Thu, 09 Jul 2026 08:44:12 +0200   Tue, 07 Jul
2026 10:02:27 +0200   KubeletHasSufficientMemory   kubelet has sufficient
memory available
  DiskPressure         False   Thu, 09 Jul 2026 08:44:12 +0200   Tue, 07 Jul
2026 10:02:27 +0200   KubeletHasNoDiskPressure     kubelet has no disk
pressure
  PIDPressure          False   Thu, 09 Jul 2026 08:44:12 +0200   Tue, 07 Jul
2026 10:02:27 +0200   KubeletHasSufficientPID      kubelet has sufficient
PID available
  Ready                True    Thu, 09 Jul 2026 08:44:12 +0200   Tue, 07 Jul
2026 10:02:27 +0200   KubeletReady                kubelet is posting ready
st
```

Egy control plane kiesés

Egy control plane

Kettő control plane kiesés

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=1783579621

Last update: **2026/07/09 06:47**