
Servidor Kea DHCP

Índex

1. Què és Kea?	1
1.1. Diferències respecte a ISC DHCP	1
2. Funcionalitats principals	2
2.1. Suport DHCPv4 i DHCPv6	2
2.2. Arquitectura modular	2
2.3. Alt rendiment i multifil	2
2.4. API REST potent	2
2.5. Backends de dades flexibles	2
2.6. Alta disponibilitat (HA)	3
2.7. Hooks modules (extensions)	3
2.8. Interfície gràfica: Stork	3
3. Instal·lació de Kea a Ubuntu Server 26.04	4
3.1. Prerequisits	4
3.2. Actualitza el sistema	4
3.3. Instal·lació del paquet	5
4. Configuració de kea-dhcp4	7
4.1. Estructura bàsica del fitxer de configuració	9
4.2. Explicació de les opcions principals	12
interfaces-config	12
lease-database	12
valid-lifetime i max-valid-lifetime	13
subnet4	13
option-data	13
5. Reserves d'amfitrions (<i>host reservations</i>)	14
6. Gestió del servei	15
6.1. Verifica l'estat dels serveis	15
6.2. Atura, inicia i reinicia	15
6.3. Recarrega la configuració sense reiniciar	15
6.4. Consulta els registres	15
7. Configuració de kea-dhcp6	16
8. Stork: tauler de control gràfic per a Kea	16
8.1. Arquitectura d'Stork	17
8.2. Instal·lació al servidor Ubuntu	17
Pas 1: Instal·lació de dependències	17
Pas 2: Instal·la PostgreSQL (prerequisit del servidor)	17
Pas 3: Afegeix el dipòsit d'Stork	17
Pas 4: Instal·la el servidor Stork	18
Pas 5: Crea la base de dades	18
Pas 6: Configura el servidor	19
Pas 7: Inicia el servei	19
Pas 8: Verifica l'estat del servei	19
8.3. Accedeix a la interfície web	20

8.4. Funcionalitats principals de la interfície	21
8.5. Instal·lació de l'agent Stork al mateix servidor	22
Pas 1. Instal·la l'agent	22
Pas 2. Registre	22
Pas 3. Autorització	23
Pas 4. Modifica el port de l'agent	24
Pas 5. Inicia l'agent	24
8.6. Instal·lació de l'agent Stork a una màquina que no és el servidor	25
Pas 1. Actualitza el sistema	25
Pas 2. Instal·la les dependències	25
Pas 3. Afegeix el dipòsit d'Stok	25
Pas 4. Instal·la l'agent	25
Pas 5. Registre	25
Pas 6. Autorització	26
Pas 7. Inicia l'agent	26
8.7. Configuració de l'agent	26
8.8. Fitxers d'Stok	26
9. Configuració del DNS dinàmic (DDNS)	27
10. Versions disponibles	27
11. Fitxers i directoris importants	27
12. Recursos addicionals	27

Cicle formatiu: CFGM Sistemes Microinformàtics i Xarxes (SMX) / CFGS Administració de sistemes informàtics en xarxa (ASIX)

Mòdul: 0227 - Serveis de xarxa / 0375 - Serveis de xarxa i Internet

Sistema operatiu: Ubuntu Server 26.04 LTS

1. Què és Kea?

Kea és el servidor DHCP de codi obert desenvolupat per l'*Internet Systems Consortium* (ISC), l'organització responsable també de BIND 9. Va ser creat per substituir l'antic `isc-dhcp`, que va arribar al final de la seva vida útil l'any 2022. Kea està dissenyat per a entorns de xarxa moderns i és, avui dia, el servidor DHCP de codi obert més desplegat del món.

Kea és una implementació moderna i modular del protocol DHCP (Dynamic Host Configuration Protocol), que permet assignar de manera automàtica adreces IP i altres paràmetres de xarxa als dispositius connectats.



Figura 1: Kea logo

NOTA

A Ubuntu Server 23.04 i versions posteriors (inclosa la 26.04), Kea és el servidor DHCP recomanat. El paquet `isc-dhcp-server` està obsolet i no és compatible des de l'Ubuntu 24.04 LTS.

1.1. Diferències respecte a ISC DHCP

Característica	ISC DHCP (antic)	Kea (nou)
Estat	End of Life (2022)	Actiu i mantingut
Arquitectura	Monolítica	Modular (daemons separats)
Multifil (<i>multithreaded</i>)	No	Sí
API REST	No	Sí (JSON)
Backend de base de dades	Fitxer pla	Fitxer, MySQL, PostgreSQL
Interfície gràfica	No	Sí (Stork)
Alta disponibilitat (HA)	Limitada	Avançada

2. Funcionalitats principals

2.1. Suport DHCPv4 i DHCPv6

Kea inclou daemons separats per a IPv4 i IPv6, que permet gestionar ambdós protocols de manera independent o simultània.

2.2. Arquitectura modular

El paquet kea instal·la diversos components:

- **kea-dhcp4-server**: servidor DHCP per a IPv4.
- **kea-dhcp6-server**: servidor DHCP per a IPv6.
- **kea-ctrl-agent**: agent de control amb API REST (JSON) per gestionar els servidors remotament.
- **kea-dhcp-ddns-server**: servei de DNS dinàmic (DDNS) que actualitza el DNS en funció dels esdeveniments d'assignació de lloguer (*lease*).

2.3. Alt rendiment i multifil

Kea és *multithreaded*, cosa que li permet gestionar entorns de gran escala amb durades de lloguer curtes, l'escenari més exigent per a un servidor DHCP.

2.4. API REST potent

Kea fa servir fitxers de configuració JSON que es poden modificar remotament mitjançant ordres `set` i recarregar sense aturar ni reiniciar el servidor. Això permet una gestió dinàmica i sense interrupcions.

2.5. Backends de dades flexibles

Les dades de xarxa (lloguers, reserves d'amfitrions, configuració) es poden emmagatzemar en:

- **Fitxer CSV** (*memfile*): opció predeterminada, lleugera i sense dependències.
- **MySQL**: permet compartir dades entre múltiples servidors Kea.
- **PostgreSQL**: alternativa relacional per a entorns empresarials.

L'ús d'una base de dades compartida permet implementar estratègies d'alta disponibilitat i facilita la gestió centralitzada de reserves d'amfitrions.

2.6. Alta disponibilitat (HA)

Kea admet diverses estratègies d'alta disponibilitat per garantir la continuïtat del servei DHCP en cas de fallada d'un servidor. Les configuracions HA es gestionen mitjançant *hooks modules*.

2.7. Hooks modules (extensions)

Les funcionalitats opcionals s'activen amb mòduls carregables dinàmicament (*hooks modules*). Alguns exemples:

- **host_cmds**: gestió remota de reserves d'amfitrions via Stork.
- **stat_cmds**: estadístiques en temps real.
- **lease_cmds**: gestió de lloguers via API.
- **RBAC i Configuration Backend**: mòduls amb llicència comercial (disponibles per a subscriptors ISC).

Es poden escriure hooks propis en C++.

2.8. Interfície gràfica: Stork

Stork és el tauler de control gràfic per a Kea. Permet:

- Monitorar múltiples servidors Kea des d'una plataforma centralitzada.
- Visualitzar l'estat de salut del sistema (CPU, memòria, versions, *uptime*).
- Controlar la utilització dels pools d'adreces i les xarxes compartides.
- Monitorar parelles d'alta disponibilitat i detectar esdeveniments de *failover*.
- Gestionar reserves d'amfitrions DHCPv4 i DHCPv6 des d'una interfície gràfica.

3. Instal·lació de Kea a Ubuntu Server 26.04

3.1. Prerequisits

- Ubuntu Server 23.04 o superior.
- Accés amb privilegis sudo.

3.2. Actualitza el sistema

```
sudo apt update && sudo apt upgrade -y
```

3.3. Instal·lació del paquet

```
sudo apt install kea
```

Durant la instal·lació, el sistema mostrarà un missatge interactiu (debconf) per configurar l'autenticació del servei kea-ctrl-agent. Cal triar una de les tres opcions:

- **Cap acció** (opció predeterminada): el kea-ctrl-agent **no** s'iniciarà.
- **Configurar amb contrasenya aleatòria**: recomanat per a proves.
- **Configurar amb una contrasenya específica**: recomanat per a producció.

AVÍS

Si no es configura cap contrasenya, el servei kea-ctrl-agent no s'iniciarà, ja que cal autenticació per evitar accessos no autoritzats a l'API administrativa.

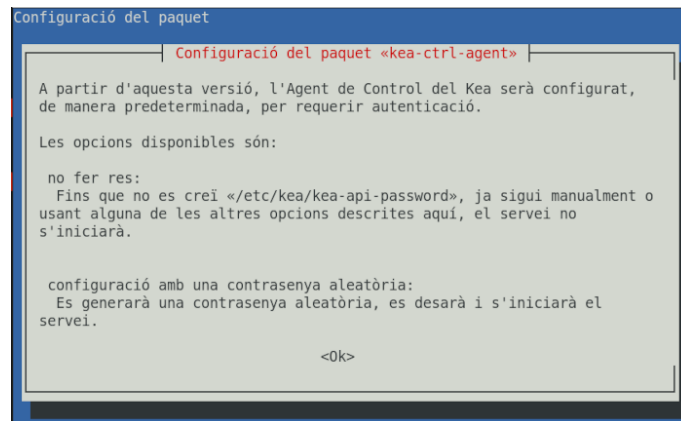


Figura 2: Configuració del paquet kea-ctrl-agent 1/4

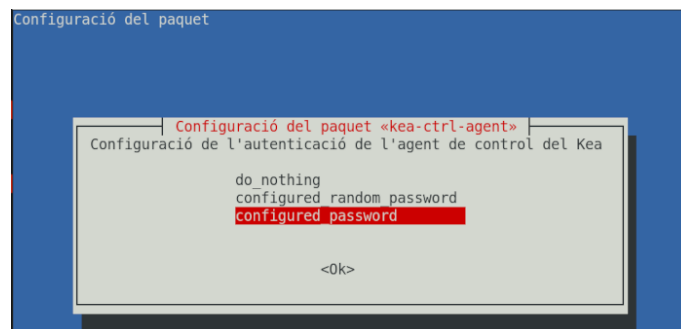


Figura 3: Configuració del paquet kea-ctrl-agent 2/4

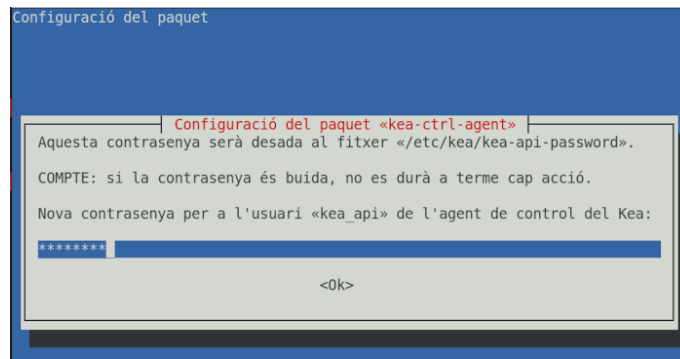


Figura 4: Configuració del paquet kea-ctrl-agent 3/4

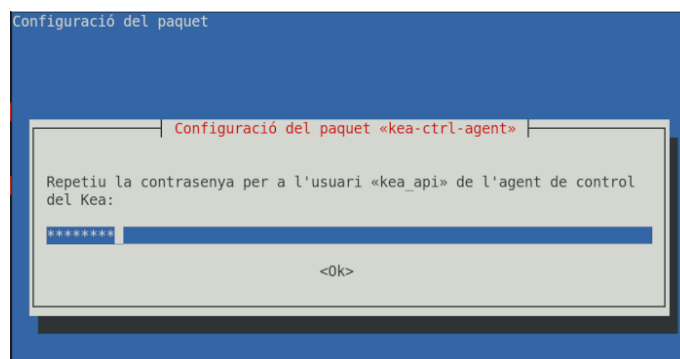


Figura 5: Configuració del paquet kea-ctrl-agent 4/4

La contrasenya s'emmagatzema a /etc/kea/kea-api-password amb propietari root : _kea i permisos 0640.

```
root@server:/etc/kea# ls -l
total 68
-rw-r----- 1 root _kea      8 Jun 25 18:15 kea-api-password
-rw-r--r-- 1 root root    5697 Mar 25 10:23 kea-ctrl-agent.conf
-rw-r--r-- 1 root root    3025 Mar 25 10:23 kea-dhcp-ddns.conf
-rw-r--r-- 1 root root   23850 Mar 25 10:23 kea-dhcp4.conf
-rw-r--r-- 1 root root   21099 Mar 25 10:23 kea-dhcp6.conf
drwxr-xr-x 2 root root    4096 Jun 25 18:15 radius
root@server:/etc/kea#
```

Per canviar la contrasenya posteriorment:

```
sudo dpkg-reconfigure kea-ctrl-agent
```

O bé editant el fitxer directament:

```
sudo nano /etc/kea/kea-api-password
```

Si UFW està habilitat, habilita el servei DHCP. El servidor DHCP utilitza [67/UDP].

```
sudo ufw allow bootps
```

4. Configuració de kea-dhcp4

El fitxer de configuració principal del servidor DHCPv4 es troba a `/etc/kea/kea-dhcp4.conf`

Aquest és el contingut per defecte:

```
{
  "Dhcp4": {
    "interfaces-config": {
      "interfaces": [ ]
    },
    "control-socket": {
      "socket-type": "unix",
      "socket-name": "kea4-ctrl-socket"
    },
    "lease-database": {
      "type": "memfile",
      "lfc-interval": 3600
    },
    "expired-leases-processing": {
      "reclaim-timer-wait-time": 10,
      "flush-reclaimed-timer-wait-time": 25,
      "hold-reclaimed-time": 3600,
      "max-reclaim-leases": 100,
      "max-reclaim-time": 250,
      "unwarned-reclaim-cycles": 5
    },
    "renew-timer": 900,
    "rebind-timer": 1800,
    "valid-lifetime": 3600,
    "option-data": [
      {
        "name": "domain-name-servers",
        "data": "192.0.2.1, 192.0.2.2"
      },
      {
        "code": 15,
        "data": "example.org"
      },
      {
        "name": "domain-search",
        "data": "mydomain.example.com, example.com"
      },
      {
        "name": "boot-file-name",
        "data": "EST5EDT4\\,M3.2.0/02:00\\,M11.1.0/02:00"
      }
    ]
  }
}
```

```

    {
      "name": "default-ip-ttl",
      "data": "0xf0"
    }
  ],
  "client-classes": [
    {
      "name": "voip",
      "test": "substring(option[60].hex,0,6) == 'Aastra'",
      "next-server": "192.0.2.254",
      "server-hostname": "hal9000",
      "boot-file-name": "/dev/null"
    }
  ],
  "subnet4": [
    {
      "id": 1,
      "subnet": "192.0.2.0/24",
      "pools": [ { "pool": "192.0.2.1 - 192.0.2.200" } ],
      "option-data": [
        {
          "name": "routers",
          "data": "192.0.2.1"
        }
      ],
      "reservations": [
        {
          "hw-address": "1a:1b:1c:1d:1e:1f",
          "ip-address": "192.0.2.201"
        },
        {
          "client-id": "01:11:22:33:44:55:66",
          "ip-address": "192.0.2.202",
          "hostname": "special-snowflake"
        },
        {
          "duid": "01:02:03:04:05",
          "ip-address": "192.0.2.203",
          "option-data": [ {
            "name": "domain-name-servers",
            "data": "10.1.1.202, 10.1.1.203"
          } ]
        }
      ],
        {
          "client-id": "01:12:23:34:45:56:67",
          "ip-address": "192.0.2.204",
          "option-data": [
            {
              "name": "vivso-suboptions",
              "data": "4491"
            },
            {
              "name": "tftp-servers",

```

```

        "space": "vendor-4491",
        "data": "10.1.1.202, 10.1.1.203"
    }
  ],
  },
  {
    "client-id": "01:0a:0b:0c:0d:0e:0f",
    "ip-address": "192.0.2.205",
    "next-server": "192.0.2.1",
    "server-hostname": "hal9000",
    "boot-file-name": "/dev/null"
  },
  {
    "flex-id": "'s0mEVaLue'",
    "ip-address": "192.0.2.206"
  }
]
}
],
"loggers": [
{
  "name": "kea-dhcp4",
  "output-options": [
    {
      "output": "stdout",
      "pattern": "%-5p %m\n",
    }
  ],
  "severity": "INFO",
  "debuglevel": 0
}
]
}
}
}

```

4.1. Estructura bàsica del fitxer de configuració

El fitxer utilitza format **JSON**. A continuació es mostra una configuració típica per assignar adreces d'un rang predefinit:

```

{
  "Dhcp4": {
    "interfaces-config": {
      "interfaces": ["eth0"]
    },
    "control-socket": {
      "socket-type": "unix",
      "socket-name": "/run/kea/kea4-ctrl-socket"
    },
    "lease-database": {

```

```

        "type": "memfile",
        "lfc-interval": 3600
    },
    "valid-lifetime": 600,
    "max-valid-lifetime": 7200,
    "subnet4": [{
        "id": 1,
        "subnet": "192.168.1.0/24",
        "pools": [{
            "pool": "192.168.1.150 - 192.168.1.200"
        }],
        "option-data": [
            {
                "name": "routers",
                "data": "192.168.1.254"
            },
            {
                "name": "domain-name-servers",
                "data": "192.168.1.1, 192.168.1.2"
            },
            {
                "name": "domain-name",
                "data": "exemple.local"
            }
        ]
    }
  ]
}

```

Una altra configuració:

```

{
  "Dhcp4": {
    "interfaces-config": {
      # especifica les interfícies de xarxa d'escolta
      "interfaces": [ "enp0s3" ]
    },
    # configuració per a lloguers caducats (els següents són els
    ↪ valors per defecte)
    "expired-leases-processing": {
      "reclaim-timer-wait-time": 10,
      "flush-reclaimed-timer-wait-time": 25,
      "hold-reclaimed-time": 3600,
      "max-reclaim-leases": 100,
      "max-reclaim-time": 250,
      "unwarned-reclaim-cycles": 5
    },
    # Temporitzador T1 que controla quan el client comença els
    ↪ processos de renovació (segons)
    "renew-timer": 900,
    # Temporitzador T2 que controla quan el client inicia els
    ↪ processos de reenllaç (segons)
  }
}

```

```

"rebind-timer": 1800,
# quant de temps són vàlides les adreces (lloguers) proporcionades
↪ pel servidor (segons)
"valid-lifetime": 3600,
"option-data": [
  {
    # especifica el teu servidor DNS
    # per especificar diverses entrades, separa-les amb comes
    "name": "domain-name-servers",
    "data": "10.0.2.10"
  },
  {
    # especifica el teu nom de domini
    "name": "domain-name",
    "data": "thos.local"
  },
  {
    # especifica la teve base de cerca de domini
    # per especificar diverses entrades, separa-les amb comes
    "name": "domain-search",
    "data": "thos.local"
  }
],
"subnet4": [
  {
    "id": 1,
    # especifica la subxarxa on s'utilitza el DHCP
    "subnet": "10.0.2.0/24",
    # especifica el rang d'adreces IP que es llogaran
    "pools": [ { "pool": "10.0.2.100 - 10.0.2.200" } ],
    "option-data": [
      {
        # especifica la porta d'enllaç
        "name": "routers",
        "data": "10.0.2.1"
      }
    ]
  }
],
# configuració de registre
"loggers": [
  {
    "name": "kea-dhcp4",
    "output_options": [
      {
        "output": "/var/log/kea/kea-dhcp4.log"
      }
    ]
  },
  {
    "severity": "INFO",
    "debuglevel": 0
  }
]
}

```

```
}
```

4.2. Explicació de les opcions principals

interfaces-config

Defineix la interfície de xarxa on el servidor escoltarà les peticions DHCP.

```
"interfaces-config": {  
  "interfaces": ["enp0s3"]  
}
```

Per escoltar a totes les interfícies:

```
"interfaces": ["*"]
```

lease-database

Configura on s'emmagatzemen els lloguers d'adreces IP.

```
"lease-database": {  
  "type": "memfile",  
  "lfc-interval": 3600  
}
```

- type: pot ser memfile, mysql o postgresql.
- lfc-interval: interval en segons per netejar (*Lease File Cleanup*) els registres caducats. En aquest exemple, cada hora.

Per usar **MySQL** com a backend:

```
"lease-database": {  
  "type": "mysql",  
  "host": "localhost",  
  "name": "kea",  
  "user": "kea",  
  "password": "contrasenya_segura"  
}
```

valid-lifetime i max-valid-lifetime

Temps de validesa del lloguer en segons.

```
"valid-lifetime": 600,  
"max-valid-lifetime": 7200
```

- **valid-lifetime**: durada predeterminada si el client no en sol·licita cap (600 s = 10 minuts).
- **max-valid-lifetime**: durada màxima permesa (7200 s = 2 hores).

subnet4

Defineix les subxarxes gestionades pel servidor.

```
"subnet4": [{  
  "id": 1,  
  "subnet": "10.0.2.0/24",  
  "pools": [{  
    "pool": "10.0.2.100 - 192.168.1.200"  
  }]  
}]
```

- **id**: identificador únic de la subxarxa.
- **subnet**: rang CIDR de la xarxa.
- **pools**: rang d'adreces IP assignables als clients.

option-data

Opcions DHCP addicionals enviades als clients.

```
"option-data": [  
  {  
    "name": "routers",  
    "data": "10.0.2.1"  
  },  
  {  
    "name": "domain-name-servers",  
    "data": "10.0.2.10, 10.0.2.11"  
  },  
  {  
    "name": "domain-name",  
    "data": "thos.local"  
  }  
]
```

Opcions comunes:

Nom de l'opció	Descripció
routers	Passarel·la predeterminada (<i>gateway</i>)
domain-name-servers	Servidors DNS
domain-name	Domini de cerca DNS
broadcast-address	Adreça de difusió
ntp-servers	Servidors de temps NTP
domain-search	Llista de dominis de cerca

5. Reserves d'amfitrions (*host reservations*)

Kea permet assignar una adreça IP fixa a un dispositiu concret, identificat per la seva adreça MAC.

```
"reservations": [
  {
    "hw-address": "aa:bb:cc:dd:ee:ff",
    "ip-address": "10.0.2.150",
    "hostname": "impressora-oficina"
  }
]
```

Aquesta secció s'afegeix dins del bloc subnet4 corresponent.

És possible veure l'adreça IP llogada al fitxer següent des del servidor DHCP als clients DHCP.

```
sudo cat /var/lib/kea/kea-leases4.csv
```

```
address,hwaddr,client_id,valid_lifetime,expire,subnet_id,fqdn_fwd,fq_
↪ dn_rev,hostname,state,user_context,pool_id
10.0.2.100,00:0c:29:e5:f5:43,01:00:0c:29:e5:f5:43,3600,1777029931,1,
↪ 0,0,rx-0.thos.local.,0,,0
10.0.2.101,52:54:00:d1:7f:bd,ff:56:50:4d:98:00:02:00:00:ab:11:b3:df:
↪ a8:ce:66:2b:f0:1e,3600,1777030048,1,0,0,ubuntu,0,,0
```

6. Gestió del servei

6.1. Verifica l'estat dels serveis

```
sudo systemctl status kea-dhcp4-server
sudo systemctl status kea-dhcp6-server
sudo systemctl status kea-ctrl-agent
sudo systemctl status kea-dhcp-ddns-server
```

6.2. Atura, inicia i reinicia

```
sudo systemctl restart kea-dhcp4-server
sudo systemctl stop kea-dhcp4-server
sudo systemctl start kea-dhcp4-server
```

6.3. Recarrega la configuració sense reiniciar

La manera recomanada és fer servir kea-shell amb l'API REST per recarregar la configuració sense interrompre el servei:

```
sudo kea-shell --host 127.0.0.1 --port 8000 \
  --auth-user kea-api \
  --auth-password-file /etc/kea/kea-api-password \
  --service dhcp4 config-reload
```

Prem Ctrl+D per enviar l'ordre. El servidor hauria de respondre:

```
[ { "arguments": { "hash": "F4BB02E0CC1B1F54004168FCF2DE952CF6E1F76D"
↵ 3B98D13167D3C0DC485F3257" }, "result": 0, "text": "Configuration
↵ successful." } ]
```

6.4. Consulta els registres

```
sudo journalctl -u kea-dhcp4-server
sudo journalctl -u kea-dhcp4-server -f # en temps real
```

Un missatge d'èxit a la recàrrega té aquest aspecte:

```
DHCP4_DYNAMIC_RECONFIGURATION_SUCCESS dynamic server reconfiguration
succeeded with file: /etc/kea/kea-dhcp4.conf
```

7. Configuració de kea-dhcp6

El fitxer de configuració del servidor DHCPv6 es troba a `/etc/kea/kea-dhcp6.conf`

L'estructura és similar a la del servidor DHCPv4, però usa la clau `Dhcp6` i les subxarxes es defineixen amb `subnet6`. Exemple bàsic:

```
{
  "Dhcp6": {
    "interfaces-config": {
      "interfaces": ["eth0"]
    },
    "lease-database": {
      "type": "memfile",
      "lfc-interval": 3600
    },
    "preferred-lifetime": 3000,
    "valid-lifetime": 4000,
    "subnet6": [{
      "id": 1,
      "subnet": "2001:db8:1::/64",
      "pools": [{
        "pool": "2001:db8:1::1 - 2001:db8:1::ffff"
      }]
    }]
  }
}
```

8. Stork: tauler de control gràfic per a Kea

Stork és una aplicació independent d'ISC que proporciona una interfície web centralitzada per monitorar i gestionar servidors Kea (i també BIND 9). No ve inclosa amb el paquet kea i cal instal·lar-la a part.



Figura 6: Stork logo

8.1. Arquitectura d'Stork

Stork es compon de dos elements:

- **stork-server**: el servidor central que serveix la interfície web. Requereix una base de dades **PostgreSQL**. Normalment s'instal·la en una màquina dedicada.
- **stork-agent**: l'agent que s'instal·la a cada màquina on s'executa Kea. Recull les dades i les envia al servidor via gRPC amb TLS.

8.2. Instal·lació al servidor Ubuntu

Pas 1: Instal·lació de dependències

```
sudo apt install curl apt-transport-https ca-certificates gnupg
```

Pas 2: Instal·la PostgreSQL (prerequisit del servidor)

```
sudo apt install postgresql postgresql-contrib
```

Habilita perquè s'iniciï automàticament en cada arrencada del sistema:

```
sudo systemctl enable --now postgresql
```

Pas 3: Afegiu el dipòsit d'Stork

Els paquets d'Stork no es troben als dipòsits oficials d'Ubuntu, sinó als dipòsits de **Cloudsmith** que manté ISC.

```
curl -sLf 'https://dl.cloudsmith.io/public/isc/stork/setup.deb.sh' |  
↪ sudo bash
```

Resposta esperada:

```
Executing the setup script for the 'isc/stork' repository ...  
  
OK: Checking for required executable 'curl' ...  
OK: Checking for required executable 'apt-get' ...  
OK: Detecting your OS distribution and release using system methods  
↪ ...  
^^^: ... Detected/provided for your OS/distribution, version and  
↪ architecture:  
>>>:  
>>>: ... distro=ubuntu version=26.04 codename=resolute  
↪ arch=x86_64  
>>>:  
OK: Checking for apt dependency 'apt-transport-https' ...
```

```
OK: Checking for apt dependency 'ca-certificates' ...
OK: Checking for apt dependency 'gnupg' ...
OK: Checking for apt signed-by key support ...
OK: Importing 'isc/stork' repository GPG keys ...
OK: Checking if upstream install config is OK ...
OK: Installing 'isc/stork' repository via apt ...
OK: Updating apt repository metadata cache ...
OK: The repository has been installed successfully - You're ready
↪ to rock!
```

Pas 4: Instal·la el servidor Stork

```
sudo apt install isc-stork-server
```

Pas 5: Crea la base de dades

L'eina `stork-tool` (inclosa amb el paquet) crea la base de dades automàticament:

```
sudo -u postgres stork-tool db-create --db-name stork --db-user
↪ stork-server
```

La sortida mostrarà les credencials generades. Observa les tres dades essencials. Apunta la contrasenya.

- `database_name="stork"`
- `password="yWRvhBrI7uyFV+diGQJD2w4nBoqXyvKc"`
- `user="stork-server"`

```
INFO[2026-06-26 04:49:25]      connection.go:44    Checking
↪ connection to database
INFO[2026-06-26 04:49:25]      connection.go:109 Connected to
↪ database /var/run/postgresql/.s.PGSQL.5432
INFO[2026-06-26 04:49:25]      connection.go:44    Checking
↪ connection to database
INFO[2026-06-26 04:49:25]      connection.go:109 Connected to
↪ database /var/run/postgresql/.s.PGSQL.5432
INFO[2026-06-26 04:49:25]      main.go:100      Created database
↪ and user for the server with the following credentials
↪ database_name="stork" password="yWRvhBrI7uyFV+diGQJD2w4nBoqXyvKc"
↪ user="stork-server"
```

Pas 6: Configura el servidor

Edita el fitxer de configuració:

```
sudo nano /etc/stork/server.env
```

Descomenta i edita la línia de l'usuari de base de dades:

```
STORK_DATABASE_NAME=stork
STORK_DATABASE_USER_NAME=stork-server
STORK_DATABASE_PASSWORD=yWRvhBrI7uyFV+diGQJD2w4nBoqXyvKc # canvia'l
↪ pel teu
```

Pas 7: Inicia el servei

```
sudo systemctl enable --now isc-stork-server
```

Pas 8: Verifica l'estat del servei

```
sudo systemctl status isc-stork-server
```

Sortida esperada:

```
● isc-stork-server.service - ISC Stork Server
   Loaded: loaded (/usr/lib/systemd/system/isc-stork-server.service;
   ↪ enabled; preset: enabled)
   Active: active (running) since Fri 2026-06-26 04:57:43 UTC; 8s ago
  Invocation: 9bbc954afc7b443299c299a2155c1209
     Docs: man:stork-server(8)
    Main PID: 4842 (stork-server)
      Tasks: 7 (limit: 1718)
     Memory: 25.7M (peak: 25.7M)
        CPU: 169ms
    CGroup: /system.slice/isc-stork-server.service
            └─4842 /usr/bin/stork-server
```

8.3. Accedeix a la interfície web

Un cop el servei és actiu, obre un navegador i accedeix a:

```
http://<ip-del-servidor>:8080
```

Les credencials per defecte són:

- **Usuari:** admin
- **Contrasenya:** admin

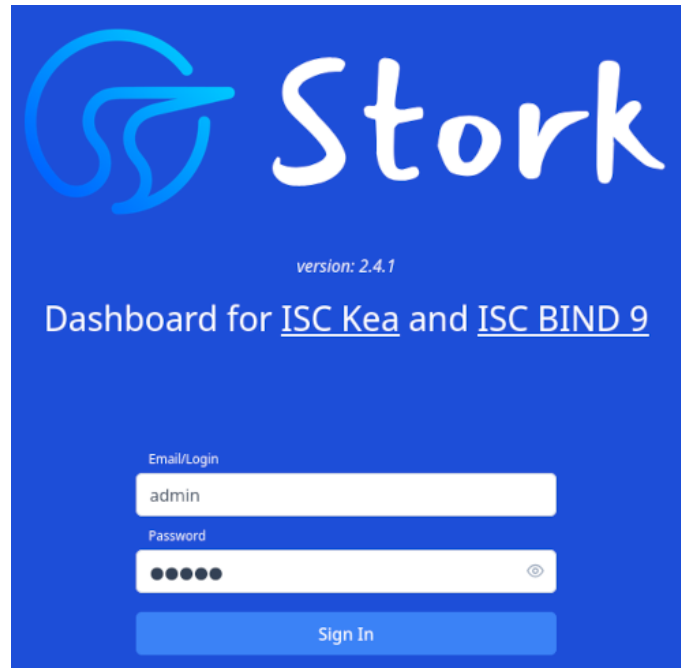


Figura 7: Interfície web

AVÍS

Canvia la contrasenya immediatament després del primer accés. La mida mínima de la contrasenya ha de ser de 12 caràcters.

Change password

You must change your password. Please enter your current password and create a new one to continue.

New password setting

Current password

●●●●●

New password

●●●●●●●●●●●●

Confirm password

●●●●●●●●●●●●

[Save](#)

Figura 8: Canvi de contrasenya

8.4. Funcionalitats principals de la interfície

Des del tauler d'Stork pots:

- Veure l'estat de tots els servidors Kea monitorats (CPU, memòria, *uptime*, versions).
- Controlar la utilització dels pools d'adreces IP i detectar zones de saturació.
- Monitorar parelles d'alta disponibilitat i esdeveniments de *failover*.
- Gestionar reserves d'amfitrions via GUI (requereix el hook `host_cmds`).
- Gestionar subxarxes i pools (requereix el hook `subnet_cmds`).
- Consultar i filtrar fitxers de log i de configuració de Kea des del navegador.
- Integrar-se amb **Prometheus** i **Grafana** per a monitoratge avançat i visualització de tendències.

8.5. Instal·lació de l'agent Stork al mateix servidor

Pas 1. Instal·la l'agent

```
sudo apt install isc-stork-agent
```

Pas 2. Registre

L'agent s'ha de registrar per aconseguir els certificats TLS per establir la connexió gRPC amb el servidor Stork.

El procés de registre:

1. L'agent genera un parell de claus i un CSR.
2. Envia el CSR al servidor Stork.
3. Un administrador ha d'aprovar el registre des de la interfície web de l'Stork Server (o es pot configurar aprovació automàtica).
4. Un cop aprovat, l'agent descarrega els certificats signats.

Cal executar `stork-agent register` apuntant a l'URL del servidor Stork.

Sintaxi:

```
sudo su stork-agent -s /bin/sh -c 'stork-agent register \  
--server-url http://<IP_SERVIDOR>:<PORT_SERVIDOR> \  
--agent-host <IP_SERVIDOR> \  
--agent-port <PORT_AGENT> \  
--non-interactive'
```

Substitueix `<IP_SERVIDOR>:<PORT_SERVIDOR>` i `<PORT_AGENT>` pels teus valors.

Exemple:

```
sudo su stork-agent -s /bin/sh -c 'stork-agent register --server-url \  
↪ http://10.0.2.10:8080 --agent-host 10.0.2.10 --agent-port 8081 \  
↪ --non-interactive'
```

Resposta esperada:

```
INFO[2026-06-28 12:59:10] register.go:93 There are no agent  
↪ certificates - they will be generated.  
INFO[2026-06-28 12:59:10] register.go:450 =====  
↪ =====  
INFO[2026-06-28 12:59:10] register.go:451 AGENT TOKEN:  
↪ D1C2D0BEDCF542C63C8FA00E160B12AC5C3EA469B545020BBEAE9E240135  
INFO[2026-06-28 12:59:10] register.go:452 =====  
↪ =====  
INFO[2026-06-28 12:59:10] register.go:455 Authorize the machine  
↪ in the Stork web UI  
INFO[2026-06-28 12:59:10] register.go:468 Try to register agent  
↪ in Stork Server
```

```
INFO[2026-06-28 12:59:10] register.go:274 Machine registered
INFO[2026-06-28 12:59:10] register.go:346 Stored agent-signed
↪ cert and CA cert
INFO[2026-06-28 12:59:10] main.go:317 Registration completed
↪ successfully
```

Pas 3. Autorització

Accedeix a la interfície web i, des de la interfície web, ves a **Services** → **Machines** → **Unauthorized** i autoritza la màquina.

⚠ **Unregistered machines** Found 1 machine requesting registration

Figura 9: Avís de màquina no registrada

🏠 > Services > Machines

Machines

🗑 Clear Authorized ✕ 🔍 Search machines

<input type="checkbox"/>	Hostname	Location 📍	Agent Token 🔑
<input type="checkbox"/>	10.0.2.10	10.0.2.10:8080	8320D886DF0B87E4AE8AA70A6D2A92EBB6301D9422025B29EEE9CAAC7927E851

Figura 10: Màquina no registrada

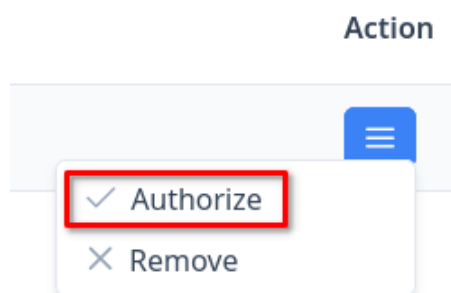


Figura 11: Autoritza

Pas 4. Modifica el port de l'agent

Edita el fitxer de configuració de l'agent Stork `/etc/stork/agent.env`

```
sudo nano /etc/stork/agent.env
```

Descomenta la línia i escriu el port configurat al registre.

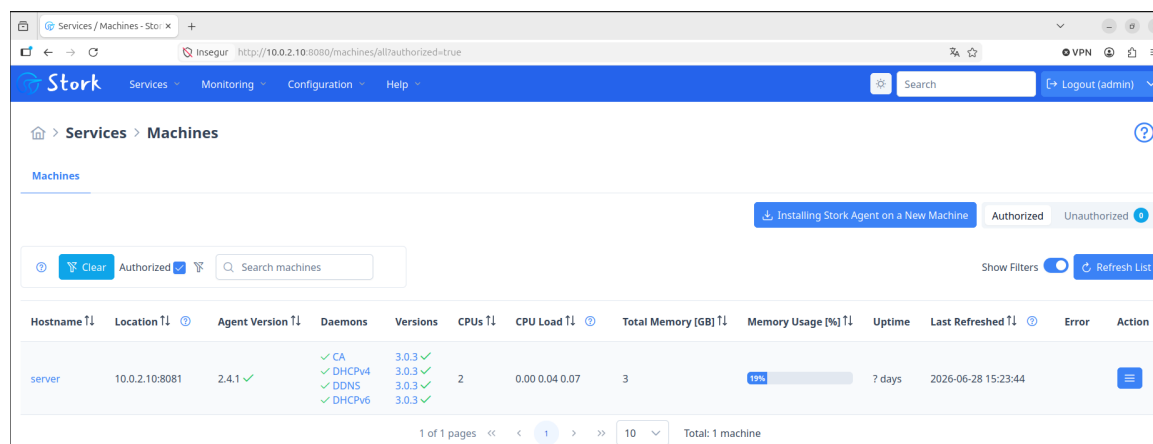
```
STORK_AGENT_PORT=8081
```

Pas 5. Inicia l'agent

Inicia l'agent i habilita'l perquè s'iniciï automàticament en cada arrencada del sistema:

```
sudo systemctl enable --now isc-stork-agent
```

Passats uns segons podràs veure les dades a la interfície d'Stork



The screenshot shows the Stork web interface in a browser. The page title is "Services > Machines". There is a search bar and a "Refresh List" button. Below the search bar is a table with the following columns: Hostname, Location, Agent Version, Daemons, Versions, CPUs, CPU Load, Total Memory (GB), Memory Usage (%), Uptime, Last Refreshed, Error, and Action. The table contains one row for a machine named "server" with location "10.0.2.10:8081" and agent version "2.4.1". The daemons listed are CA, DHCPv4, DDNS, and DHCPv6, all with version "3.0.3". The CPU load is "0.00 0.04 0.07", total memory is "3 GB", and memory usage is "19%". The uptime is "7 days" and the last refreshed time is "2026-06-28 15:23:44".

Hostname	Location	Agent Version	Daemons	Versions	CPUs	CPU Load	Total Memory (GB)	Memory Usage (%)	Uptime	Last Refreshed	Error	Action
server	10.0.2.10:8081	2.4.1 ✓	✓ CA ✓ DHCPv4 ✓ DDNS ✓ DHCPv6	3.0.3 ✓ 3.0.3 ✓ 3.0.3 ✓ 3.0.3 ✓	2	0.00 0.04 0.07	3	19%	7 days	2026-06-28 15:23:44		

Figura 12: Tauler de control d'Stork

8.6. Instal·lació de l'agent Stork a una màquina que no és el servidor

Pas 1. Actualitza el sistema

```
sudo apt update && sudo apt upgrade -y
```

Pas 2. Instal·la les dependències

```
sudo apt install curl apt-transport-https ca-certificates gnupg
```

Pas 3. Afegeix el dipòsit d'Stork

```
curl -sLf 'https://dl.cloudsmith.io/public/isc/stork/setup.deb.sh' |  
↪ sudo bash
```

Pas 4. Instal·la l'agent

```
sudo apt install isc-stork-agent
```

Pas 5. Registre

Cal executar `stork-agent register` apuntant a l'URL del servidor Stork.

Sintaxi:

```
sudo su stork-agent -s /bin/sh -c 'stork-agent register \  
--server-url http://<IP_SERVIDOR>:<PORT_SERVIDOR> \  
--agent-host <IP_MÀQUINA> \  
--agent-port <PORT_MÀQUINA> \  
--non-interactive'
```

Substitueix `<IP_SERVIDOR>:<PORT_SERVIDOR>`, `<IP_MÀQUINA>` i `<PORT_MÀQUINA>` pels teus valors.

Exemple:

```
sudo su stork-agent -s /bin/sh -c 'stork-agent register --server-url  
↪ http://10.0.2.10:8080 --agent-host 10.0.2.11 --agent-port 8080  
↪ --non-interactive'
```

Pas 6. Autorització

Accedeix a la interfície web i, des de la interfície web, ves a **Services** → **Machines** → **Unauthorized** i autoritza la màquina.

Pas 7. Inicia l'agent

Inicia l'agent i habilita'l perquè s'iniciï automàticament en cada arrencada del sistema:

```
sudo systemctl enable --now isc-stork-agent
```

8.7. Configuració de l'agent

La configuració general és:

STORK_AGENT_HOST - l'adreça IP de la interfície de xarxa o el nom DNS que stork-agent ha d'utilitzar per rebre connexions del servidor; el valor per defecte és 0.0.0.0 (és a dir, escolta a totes les interfícies).

STORK_AGENT_PORT - el número de port que l'agent ha d'utilitzar per rebre connexions del servidor; el valor per defecte és 8080.

STORK_AGENT_LISTEN_STORK_ONLY - això només habilita la funcionalitat de Stork, és a dir, desactiva els exportadors de Prometheus; el valor per defecte és fals.

STORK_AGENT_LISTEN_PROMETHEUS_ONLY - això només habilita els exportadors de Prometheus, és a dir, desactiva la funcionalitat d'Stork; el valor per defecte és fals.

STORK_AGENT_SKIP_TLS_CERT_VERIFICATION - això omet la verificació del certificat TLS quan stork-agent es connecta a Kea a través de TLS i Kea utilitza certificats autosignats; el valor per defecte és fals.

8.8. Fitxers d'Stork

Ruta	Descripció
/etc/stork/server.env	Configuració del servidor Stork
/etc/stork/agent.env	Configuració de l'agent Stork

9. Configuració del DNS dinàmic (DDNS)

El servei `kea-dhcp-ddns-server` actualitza automàticament els registres DNS quan s'assigna o allibera un lloguer DHCP. Consulta com fer la configuració a [DDNS](#)

10. Versions disponibles

A juny de 2026, les versions actives de Kea són:

Versió	Estat	Publicació	Fi de suport
3.2.0	Estable actual	Juny 2026	Juliol 2027
3.0.3	LTS (Long-Term Support)	Març 2026	Juny 2028
3.1.9	Desenvolupament	Maig 2026	---

La versió disponible als dipòsits oficials d'Ubuntu 26.04 a juny de 2026 és la **3.0.3**, i pot diferir de l'última versió publicada per ISC. Per a la versió més recent, es poden fer servir els dipòsits oficials de Cloudsmith que ISC manté per a Ubuntu, Debian, RHEL i Fedora.

11. Fitxers i directoris importants

Ruta	Descripció
<code>/etc/kea/kea-dhcp4.conf</code>	Configuració del servidor DHCPv4
<code>/etc/kea/kea-dhcp6.conf</code>	Configuració del servidor DHCPv6
<code>/etc/kea/kea-ctrl-agent.conf</code>	Configuració de l'agent de control REST
<code>/etc/kea/kea-dhcp-ddns.conf</code>	Configuració del servei DDNS
<code>/etc/kea/kea-api-password</code>	Contrasenya de l'API REST
<code>/var/lib/kea/</code>	Fitxers de lloguers (<i>memfile</i>)
<code>/run/kea/</code>	Sòcols de control Unix

12. Recursos addicionals

- [Documentació oficial de Kea \(ISC\)](#)
- [Manual d'administració de Kea \(ARM\)](#)
- [Guia d'Ubuntu Server: instal·lar isc-kea](#)
- [Dipòsit de Kea a GitLab](#)
- [Migració des d'ISC DHCP a Kea](#)
- [Comparativa d'estratègies HA de Kea](#)
- [Llista de correu kea-users](#)
- [Documentació oficial d'Stork](#)
- [Guia ràpida d'instal·lació d'Stork \(ISC KB\)](#)
- [Dipòsit d'Stork a GitLab](#)

Versions d'aquest document

- HTML - [kea.html](#)
- PDF - [kea.pdf](#)
- ODT - [kea.odt](#)
- MD - [kea.md](#)

[Domini Públic \(CC0\)](#)