Stork

(Managing and Monitoring Kea DHCP)

Carsten Strotmann and the ISC Team
Welcome to our Webinar on Stork and Kea DHCP
In this Webinar

- What is Stork?
- New Features of Stork
- A Demo
- Stork Installation and Configuration
- Monitoring with Prometheus and Grafana
- Managing DHCP Reservations
- Hands-On Stork
What is Stork?

- Stork is an open source Monitoring and Management System for Kea DHCP and BIND 9
  - Web-UI
  - REST-API
  - Prometheus Exporter for Kea DHCP and BIND 9 (https://prometheus.io)
  - Integration into Grafana visualization (https://grafana.com/grafana/)
Stork Architecture

DHCP Server
- Kea DHCPv4
- Kea DHCPv6
- Kea DDNS
- Kea Control Agent
- Stork Agent

DNS Server
- BIND 9
- RNDC
- Statistics
- Stork Agent
Grafana Monitoring Only Setup

DHCP Server
- Kea DHCPv4
- Kea DHCPv6
- Kea DDNS

Kea Control Agent

Stork Agent

DNS Server
- BIND 9
- RNDC

Statistics

Prometheus

Grafana
Stork and Grafana Monitoring (and Management)
Update on 2020 Stork Video
What's new

- This webinar does not cover all Stork functions in full detail
- See the video from 2020 for additional information on Stork (https://www.youtube.com/watch?v=5aF9NBIKhqQ)
## New: Reservation management

### Host Reservations

<table>
<thead>
<tr>
<th>DHCP Identifiers</th>
<th>IP Addresses</th>
<th>IPv6 Prefixes</th>
<th>Hostname</th>
<th>Global/Subnet</th>
<th>App Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>hw-address=(00:23:8b:f2:b8:13)</td>
<td>172.22.1.88</td>
<td></td>
<td>lemote</td>
<td>172.22.1.0/24</td>
<td>kea@172.22.1.8</td>
</tr>
<tr>
<td>hw-address=(00:26:b0:d6:a4:e0)</td>
<td>172.22.1.27</td>
<td></td>
<td>macbookpro</td>
<td>172.22.1.0/24</td>
<td>kea@172.22.1.8</td>
</tr>
<tr>
<td>hw-address=(00:30:48:d5:22:b1)</td>
<td>172.22.1.11</td>
<td></td>
<td>csos2-2</td>
<td>172.22.1.0/24</td>
<td>kea@172.22.1.8</td>
</tr>
<tr>
<td>hw-address=(00:30:65:6f:7f:5a)</td>
<td>172.22.1.20</td>
<td></td>
<td>imac</td>
<td>172.22.1.0/24</td>
<td>kea@172.22.1.8</td>
</tr>
<tr>
<td>hw-address=(00:40:ca:dd:f2:80)</td>
<td>172.22.1.32</td>
<td></td>
<td>amilopro</td>
<td>172.22.1.0/24</td>
<td>kea@172.22.1.8</td>
</tr>
<tr>
<td>hw-address=(00:50:da:42:91:c7)</td>
<td>172.22.1.75</td>
<td></td>
<td>cyrix686</td>
<td>172.22.1.0/24</td>
<td>kea@172.22.1.8</td>
</tr>
<tr>
<td>hw-address=(00:d0:b7:ad:3d:3c)</td>
<td>172.22.1.24</td>
<td></td>
<td>csmobile</td>
<td>172.22.1.0/24</td>
<td>kea@172.22.1.8</td>
</tr>
<tr>
<td>hw-address=(00:e0:7d:a3:2:e7)</td>
<td>172.22.1.4</td>
<td></td>
<td>nfssrv</td>
<td>172.22.1.0/24</td>
<td>kea@172.22.1.8</td>
</tr>
<tr>
<td>hw-address=(00:00:09:1e:72:da)</td>
<td>172.22.1.16</td>
<td></td>
<td>hp715-80</td>
<td>172.22.1.0/24</td>
<td>kea@172.22.1.8</td>
</tr>
<tr>
<td>hw-address=(10:9:ad:4f:0b:da)</td>
<td>172.22.1.21</td>
<td></td>
<td>macmini2</td>
<td>172.22.1.0/24</td>
<td>kea@172.22.1.8</td>
</tr>
</tbody>
</table>
New: Global Search / Leases Search

<table>
<thead>
<tr>
<th>Subnets</th>
<th>Hosts</th>
<th>Machines</th>
<th>Apps</th>
</tr>
</thead>
</table>

3 of 1 pages  <<  <  1  >  >>  10  

All content © 2023 Internet Systems Consortium, Inc.
New: View Logs

<table>
<thead>
<tr>
<th>Logger</th>
<th>Severity</th>
<th>Output Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>kea-dhcp4</td>
<td>info</td>
<td>stdout</td>
</tr>
<tr>
<td>kea-dhcp4</td>
<td>info</td>
<td>/var/log/kea/kea-dhcp4.log</td>
</tr>
</tbody>
</table>

Log /var/log/kea/kea-dhcp4.log from the app kea@172.22.1.8 on the machine [3] 172.22.1.8

2023-02-20 11:23:59.720 INFO (kea-dhcp4.commands/1520.1402171739223064) COMMAND RECEIVED Received command 'config-get'

2023-02-20 11:23:59.720 INFO (kea-dhcp4.commands/1520.1402171739223064) COMMAND RECEIVED Received command 'statistic-get-all'

2023-02-20 11:23:59.720 INFO (kea-dhcp4.commands/1520.1402171739223064) COMMAND RECEIVED Received command 'configuration-get'

2023-02-20 11:23:59.720 INFO (kea-dhcp4.statistics-books/1520.1402171739223064) STAT.Configuration.GET stat-lease-get command successful, parameters: (all subsets) rows found

2023-02-20 11:23:59.720 INFO (kea-dhcp4.commands/1520.1402171739223064) COMMAND RECEIVED Received command 'statistic-get'

2023-02-20 11:23:59.720 INFO (kea-dhcp4.commands/1520.1402171739223064) COMMAND RECEIVED Received command 'version-get'

2023-02-20 11:23:59.720 INFO (kea-dhcp4.commands/1520.1402171739223064) COMMAND RECEIVED Received command 'status-get'

2023-02-20 11:23:59.720 INFO (kea-dhcp4.commands/1520.1402171739223064) COMMAND RECEIVED Received command 'config-get'

2023-02-20 11:23:59.720 INFO (kea-dhcp4.commands/1520.1402171739223064) COMMAND RECEIVED Received command 'version-get'

2023-02-20 11:23:59.720 INFO (kea-dhcp4.commands/1520.1402171739223064) COMMAND RECEIVED Received command 'status-get'

2023-02-20 11:23:59.720 INFO (kea-dhcp4.commands/1520.1402171739223064) COMMAND RECEIVED Received command 'config-get'

2023-02-20 11:23:59.720 INFO (kea-dhcp4.commands/1520.1402171739223064) COMMAND RECEIVED Received command 'version-get'

2023-02-20 11:23:59.720 INFO (kea-dhcp4.commands/1520.1402171739223064) COMMAND RECEIVED Received command 'status-get'

2023-02-20 11:23:59.720 INFO (kea-dhcp4.commands/1520.1402171739223064) COMMAND RECEIVED Received command 'config-get'
New: RPS (Response Per Second) statistics

<table>
<thead>
<tr>
<th>Host</th>
<th>App Version</th>
<th>App Name</th>
<th>Daemon</th>
<th>Status</th>
<th>RPS (15min)</th>
<th>RPS (24h)</th>
<th>HA State</th>
<th>Detected Failure w/HA</th>
<th>Uptime</th>
</tr>
</thead>
<tbody>
<tr>
<td>home01</td>
<td>Kea 2.2.0</td>
<td>kea@172.22.1.8</td>
<td>dhcp4</td>
<td></td>
<td>34</td>
<td>2056</td>
<td>🌐 not configured</td>
<td></td>
<td>1 m 13 d 22 h 54 min 35 s</td>
</tr>
</tbody>
</table>
New: Kea configuration checkers
<table>
<thead>
<tr>
<th>State</th>
<th>Name</th>
<th>Description</th>
<th>Selectors</th>
<th>Triggers</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enabled</td>
<td>canonical_prefix</td>
<td>The checker verifying if subnet prefixes are in the canonical form.</td>
<td>✉️kea-dhcp-daemon</td>
<td>✿ manual ✿ config change</td>
</tr>
<tr>
<td>Enabled</td>
<td>disposable_shared_network</td>
<td>The checker verifying if a shared network can be removed because it is empty or contains only one subnet.</td>
<td>✉️kea-dhcp-daemon</td>
<td>✿ manual ✿ config change</td>
</tr>
<tr>
<td>Enabled</td>
<td>disposable_subnet</td>
<td>The checker verifying if a subnet can be removed because it includes no pools and no reservations. The check is skipped when the host_cmds hook library is loaded because host reservations may be present in the database.</td>
<td>✉️kea-dhcp-daemon</td>
<td>✿ manual ✿ config change ✿ host reservations change</td>
</tr>
<tr>
<td>Enabled</td>
<td>host_cmds_presence</td>
<td>The checker verifying if the host_cmds hooks library is loaded when host backend is in use.</td>
<td>✉️kea-dhcp-daemon</td>
<td>✿ manual ✿ config change</td>
</tr>
<tr>
<td>Enabled</td>
<td>out_of_pool_reservation</td>
<td>The checker suggesting the use of out-of-pool host reservation mode when there are subnets with all host reservations outside of the dynamic pools.</td>
<td>✉️kea-dhcp-daemon</td>
<td>✿ manual ✿ config change ✿ host reservations change</td>
</tr>
<tr>
<td>Enabled</td>
<td>overlapping_subnet</td>
<td>The checker verifying if subnet prefixes do not overlap.</td>
<td>✉️kea-dhcp-daemon</td>
<td>✿ manual ✿ config change</td>
</tr>
<tr>
<td>Enabled</td>
<td>stat_cmds_presence</td>
<td>The checker verifying if the stat_cmds hooks library is loaded.</td>
<td>✉️kea-dhcp-daemon</td>
<td>✿ manual ✿ config change</td>
</tr>
</tbody>
</table>
New: Dump machine configuration to file

- **Kea App** (kea@172.22.1.8)
  - Version 2.2.0
  - DHCPv4 ✓ DHCPv6 ☐ DDNS ✓ CA

- **BIND 9 App**
  - Version BIND 9.18.11 (Stable Release) <cid:>
  - ✓ named
New: View Kea JSON Configuration

```
Dhcp4:
  allocator: "iterative"
  authoritative: false
  boot-file-name: ""
  calculate-tee-times: false
  control-socket: {...}
  ddns-generated-prefix: "myhost"
  ddns-override-client-update: false
  ddns-override-no-update: false
  ddns-qualifying-suffix: ""
  ddns-replace-client-name: "never"
  ddns-send-updates: true
  ddns-update-on-renew: false
  ddns-use-conflict-resolution: true
  decline-probation-period: 86400

dhcp-ddns:
  enable-updates: false
  max-queue-size: 1024
  ncr-format: "JSON"
  ncr-protocol: "UDP"
  sender-ip: "0.0.0.0"
  sender-port: 0
  server-ip: "127.0.0.1"
  server-port: 53001
```
New: Prometheus exporter and Grafana Dashboard Updates

- Extended Kea DHCP exporter for Prometheus
- The Stork Agent now contains a BIND 9 exporter for Prometheus
  - The exporter is based on bind_exporter from Digital Ocean
    https://github.com/prometheus-community/bind_exporter
Stork Installation
Stork Installation

- The hands-on workshop webpage at https://webinar.defaultroutes.de/webinar/14-kea-stork-workshop.html contains a step by step guide of how to install Kea-DHCP, BIND 9, Stork, PostgreSQL, Prometheus and Grafana on Red Hat EL 9 compatible systems
(Manual) Installation from packages

- ISC offers ready made packages for major Linux distributions (Red Hat compatible, Debian compatible, Alpine Linux)
  - https://cloudsmith.io/~isc/repos/stork/setup/
- Choice between open-source and support subscription repositories
(Automated) Installation using installation scripts

- The Stork manual (https://stork.readthedocs.io/en/latest/install.html) contains instructions on shell scripts that can be downloaded from the Cloudsmith repositories that will automate the installation process.
  - The scripts support Debian, Red Hat compatible systems and Alpine Linux
  - In security sensitive environments, these scripts should be first downloaded, inspected and executed with care
Installation from Container Images

- The Stork Source code repository contains source code and scripts to generate an extensive demo environment with multiple Kea-DHCP instances, BIND 9 and Stork
  - These can be used as a starting point for own Container images (using Docker, Podman or similar container engines)
Installation of Database

- Stork requires a PostgreSQL database installation
  - works fine sharing one PostgreSQL instance between Stork and Kea-DHCP
  - Requires the pgcrypto extension
Stork Server and Agent Configuration
Stork Server and Agent Configuration

• Stork-Server and Stork-Agent are configured through environment variables or through command line switches
  ■ There are no configuration files

• These environment variables are injected into the process environment from a service management system (such as systemd)

• Further configuration is done through the Stork Web-UI and stored in the PostgreSQL database
### database settings
STORK_DATABASE_HOST=127.0.0.1
STORK_DATABASE_PORT=5432
STORK_DATABASE_NAME=stork_db
STORK_DATABASE_USER_NAME=stork
STORK_DATABASE_PASSWORD=secure-password

### REST API settings
STORK_REST_HOST=127.0.0.1
STORK_REST_PORT=9877
STORK_REST_STATIC_FILES_DIR=/usr/share/stork/www

### enable Prometheus /metrics HTTP endpoint for exporting metrics from the server to Prometheus. It is recommended to secure this endpoint (e.g. using HTTP proxy).
STORK_SERVER_ENABLE_METRICS=true

### Logging parameters
STORK_LOG_LEVEL=WARNING
CLICOLOR=false
Configuration of Stork Agent (env files)

STORK_AGENT_HOST=127.0.0.1
STORK_AGENT_PORT=9878

STORK_AGENT_LISTEN_STORK_ONLY=false
STORK_AGENT_LISTEN_PROMETHEUS_ONLY=false

STORK_AGENT_PROMETHEUS_KEA_EXPORTER_ADDRESS=127.0.0.1
STORK_AGENT_PROMETHEUS_KEA_EXPORTER_PORT=9879
STORK_AGENT_PROMETHEUS_KEA_EXPORTER_INTERVAL=60
STORK_AGENT_PROMETHEUS_KEA_EXPORTER_PER_SUBNET_STATS=true

STORK_AGENT_PROMETHEUS_BIND9_EXPORTER_ADDRESS=127.0.0.1
STORK_AGENT_PROMETHEUS_BIND9_EXPORTER_PORT=9119
STORK_AGENT_PROMETHEUS_BIND9_EXPORTER_INTERVAL=60

STORK_AGENT_SERVER_URL=http://127.0.0.1:9877
STORK_AGENT_SKIP_TLS_CERT_VERIFICATION=true

### Logging parameters

### Set logging level. Supported values are: DEBUG, INFO, WARN, ERROR
STORK_LOG_LEVEL=WARN
CLICOLOR=false
Securing Stork
Securing communication with TLS

- The communication between Stork Agents and the Stork Server can be secured with TLS encryption based on x.509 certificates
  - Public CA certificates, as well as private CA or "self-signed" certificates are possible
- The communication between Stork Server and the PostgreSQL database can also be secured with TLS
Stork-Tool

- `stork-tool` is a new command line utility that can be used to import or export TLS certificates from Stork.
- The tool can also be used to maintain the Stork database (init, upgrade, downgrade and checking the database schema version).
Importing the Grafana Dashboards
Importing the Grafana Dashboards

- Stork comes with ready made Grafana Dashboard configurations for Kea-DHCP4, Kea-DHCP6 and BIND 9 metrics data
- The dashboard definitions can be found as JSON source in /usr/share/stork/examples/grafana
- The files can be imported into Grafana using copy-n-paste

```
# ls -l /usr/share/stork/examples/grafana
total 108
-rw-rw-rw-. 1 root root 43169 Jan 31 09:02 bind9-resolver.json
-rw-rw-rw-. 1 root root 28474 Jan 31 09:02 kea-dhcp4.json
-rw-rw-rw-. 1 root root 36025 Jan 31 09:02 kea-dhcp6.json
```
Importing the Grafana Dashboards
Importing the Grafana Dashboards
Manage Reservations with Stork
Reservations in the SQL database

- Kea DHCP (both the IPv4 and IPv6 DHCP server) can either have leases in the configuration or in a SQL database system (PostgreSQL or MySQL/MariaDB)
  - For conflicting reservation information, the configuration file has priority over the database content
Database configuration

```json
[...
  "hosts-database": {
    "type": "postgresql",
    "host": "localhost",
    "name": "kea_host_db",
    "user": "kea",
    "password": "secure-password"
  },
[...]
```
Host Commands Hook

- Stork will display the reservations available both in the configuration file and in the database.
- To be able to manage the reservations in the database, the (non-free) `hosts-cmds` hook needs to be installed and loaded into the Kea-DHCP Server.
Host Commands Hook

```
[...]
    "hooks-libraries": [
        {
            "library": "/usr/lib64/kea/hooks/libdhcp_stat_cmds.so",
            "parameters": {} 
        },
        {
            "library": "/usr/lib64/kea/hooks/libdhcp_host_cmds.so",
            "parameters": {} 
        }
    ],
[...]
```
# Entering a new Reservation

## DHCP Host Reservations

<table>
<thead>
<tr>
<th>DHCP Identifiers</th>
<th>IP Addresses</th>
<th>IPv6 Prefixes</th>
<th>Hostname</th>
<th>Global/Subnet</th>
<th>App Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>hw-address=(80:01:03:04:05:06)</td>
<td>192.0.2.20</td>
<td></td>
<td>some-host</td>
<td>192.0.2.0/24</td>
<td>kea@127.0.0.1</td>
</tr>
<tr>
<td>hw-address=(80:80:80:80:90:ff)</td>
<td>2001:db8::1</td>
<td>2001:db8:100::1</td>
<td>ipv6-host</td>
<td>global</td>
<td>kea@127.0.0.1</td>
</tr>
</tbody>
</table>
Entering a new Reservation
Entering a new Reservation
Entering a new Reservation

• A new reservation might not be shown immediately in the Stork Web-UI
  ▪ It will be written into the database immediately and will be active
  ▪ It will be shown after the next sync of the Stork-Agent with the Kea-DHCP Server (can take a few seconds)
Host-Reservation Demo
Upcoming ISC Webinar

- 30 Mar - Configuring vendor options in Kea
- 20 Apr - Netbox and Kea DHCP
- 16 May - Migrating to Kea from ISC DHCP
- 07 Jun - Using the new dynamic templates in Kea
Questions / Answers
Hands-On:

- Installing Kea/Bind/Stork/Prometheus/Grafana
  https://webinar.defaultroutes.de/webinar/14-kea-stork-workshop.html