KEA DHCP

(DDI User Group 12/2022)

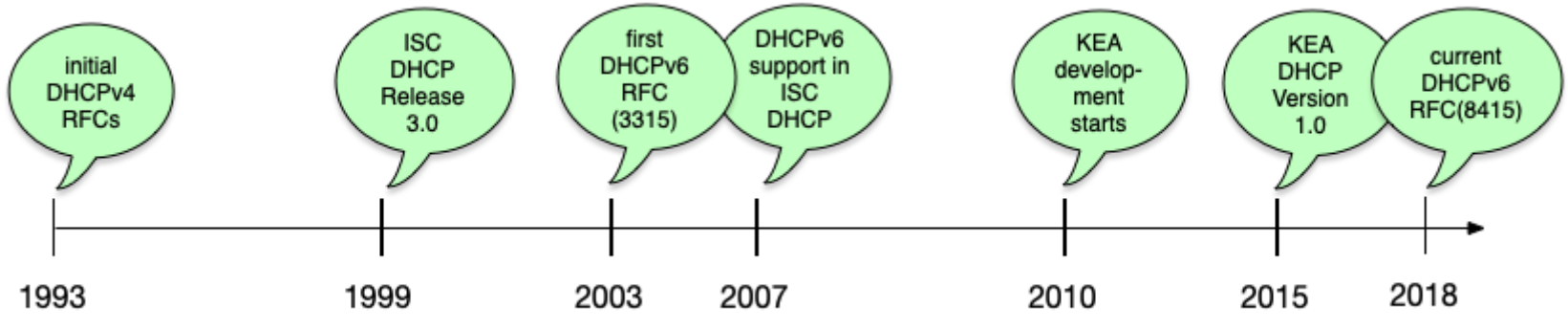
CARSTEN STROTMANN

Created: 2022-12-01 Thu 08:12

ABOUT DHCP

- DHCP is short for Dynamic Host Configuration Protocol
  - The Internet standard protocol to assign IP addresses and network related configuration to TCP/IP connected machines
  - For IPv4: RFC 2131
  - For IPv6: RFC 8415

SHORT HISTORY OF DHCP (INCLUDING ISC-DHCP AND KEA)



ISC DHCP END-OF-LIFE

- ISC has announced (October 5th 2022) that the ISC-DHCP software is not further developed: https://www.isc.org/blogs/isc-dhcp-eol/
- Kea DHCP (also from ISC) is the sucessor

WHAT IS KEA DHCP

KEA DHCP (1/2)

- A modern DHCPv4 and DHCPv6 Server
- Open Source (MPL License)
- JSON/REST API
- Modular design
- High performance (> 1000 leases/seconds with SQL database backend)
- Failover via SQL DB or High-Availability-Hook
- Host reservation support

KEA DHCP (2/2)

- Support for DHCPv6 prefix delegation
- Dynamic reconfiguration
- Dynamic DNS updates
- SQL database backend (MySQL / PostgreSQL …)
- Statistics module
- PXE Boot support

PLATFORMS AND REQUIREMENTS

KEA PLATFORMS (1/2)

- Kea is officially supported on
  - CentOS Linux
  - Fedora Linux
  - Ubuntu Linux
  - Debian Linux
  - FreeBSD Unix

KEA PLATFORMS (2/2)

- Kea is also know to work on
  - MacOS X
  - Alpine Linux
  - OpenBSD Unix
- Kea currently does not work on Windows operating systems

INSTALLATION

KEA DHCP INSTALLATION VIA OPERATING SYSTEM PACKAGES

- Kea DHCP is available in the package repositories of all major Linux and Unix systems
  - If you have support from the operating system vendor (Red Hat, Canonical, Suse), installing from the OS repositories is the best choice
- Kea DHCP can also be installed from source, if you need a special build configuration or the latest features not available in the binary packages

KEA DHCP INSTALLATION VIA ISC PACKAGES

- ISC offers binary packages of Kea DHCP for our users and customers, hosted on Cloudsmith.
  - If you need the latest Kea version, these packages are an alternative to building Kea from source
- The packages provide fast access to the latest bug fixes
  - ISC provides the binary packages along with sources at the time of release

KEA BINARY PACKAGES FROM ISC

- The open source packages contain the base Kea software and the following hooks libraries:
  - Flexible Option
  - Lease Commands
  - High Availability
  - Statistics Commands
  - BOOTP
  - Run Script
  - User Check

PACKAGES FOR SUPPORT CUSTOMERS

- Users of Kea that purchase professional Kea DHCP support from ISC are entitled to special software features that are not available in the open source version:
  - Class Commands
  - Configuration Backend Commands
  - Flexible Identifier
  - Forensic Logging
  - Host Cache
  - Host Commands
  - RADIUS support
  - Subnet Commands
  - GSS-TSIG (Microsoft AD dynamic DNS Updates)

KEA HOOKS

THE KEA HOOKS

- The base Kea software implements the basic DHCPv4 and DHCPv6 functions
- These basic functions can be extended via *hooks*.
  - The *hooks* are libraries that contain extra functions that will be called when a DHCP request is processed
  - Hooks allow the core Kea system to stay lean
  - Installations only load the functions used and needed
  - This reduces the complexity and the attack surface of an installation

TYPES OF HOOKS AVAILABLE

- Hooks that are part of the Kea open source code (source and binary packages)
- Premium hooks that can be purchased online from the ISC website
- Hooks that are available for ISC support subscription customers
- Third party hooks (source code)

PREMIUM/SUBSCRIPTION HOOKS

- The premium/subscription hooks are available in source and binary (package) form
  - Customers can download the hooks for a period of 12 month
  - As the API between Kea and the hooks might change between Kea versions, care must be taken to install hooks that match the Kea version number

KEA CONFIGURATION

JSON

- Configuration files for the **DHCPv4**, **DHCPv6**, **DDNS**, Control Agent, and **NETCONF** modules are defined in an extended JSON format.
- Basic JSON is defined in RFC 7159 and ECMA 404.

```
{
"Dhcp4": {
    "interfaces-config": {
        "interfaces": [ "eth0" ]
    },
    "control-socket": {
        "socket-type": "unix",
        "socket-name": "/tmp/kea-dhcp4.socket"
    },
[...]
```

EXTENDED JSON

- Kea components use an extended JSON with additional features:
  - Shell comments: any text after the hash (#) character is ignored.
  - C comments: any text after double slashes (`//`) is ignored.
  - Multiline comments: any text between `/*` and `*/` is ignored. This commenting can span multiple lines.
  - File inclusion: JSON files can include other JSON files by using a statement of the form `<?include "file.json"?>`.

JSON EDITOR

- When working with KEA, it helps to have an editor that understands the JSON format, can check the syntax and can highlight and reformat JSON data
  - Emacs
  - VIM
  - Visual Studio Code
  - TextMate / BBEdit
  - …

KEA CONFIGURATION FILES

- The main Kea configuration file are
  - `kea-ctrl-agent.conf` - Kea control agent
  - `kea-dhcp-ddns.conf` - Kea dynamic DNS updater
  - `kea-dhcp4.conf` - Kea DHCPv4 server
  - `kea-dhcp6.conf` - Kea DHCPv6 server

A BASIC KEA DHCPV4 CONFIGURATION

NETWORK INTERFACE AND CONTROL SOCKET

- The Kea DHCP server needs to know on which network interfaces the DHCP service should listen on
- The control socket defines the communication interface between the DHCP server process and the administration tools

```
{
"Dhcp4": {
    "interfaces-config": {
        "interfaces": [ "eth0" ]
    },
    "control-socket": {
        "socket-type": "unix",
        "socket-name": "/tmp/kea-dhcp4.socket"
    },
[...]
```

LEASE DATABASE DEFINITION

- Kea DHCP needs to know where to store the lease information. The configuration snippet below defines a *in-memory* database

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

GLOBAL CONFIGURATION AND OPTIONS

- Some DHCP configurations are *global* and apply to all the subnets and pools managed by the DHCP server

```
[...]
    "renew-timer": 900,
    "rebind-timer": 1800,
    "valid-lifetime": 3600,
[...]
```

IPV4-SUBNET AND POOL DEFINITION

- The example of a subnet below with DHCP pool definition includes subnet specific options (default router option: `routers`)

```
[...]
    "subnet4": [
        {
            "subnet": "192.0.2.0/24",
            "pools": [ { "pool": "192.0.2.100 - 192.0.2.200" } ],
            "option-data": [
                {
                    "name": "routers",
                    "data": "192.0.2.1"
                }
            ]
        }
    ]
},
[...]
```

LOGGING

- Kea DHCP has comes with a flexible and powerful logging framework
- The configuration snippet below configures a log-file for the DHCPv4 service

```
[...]
  "loggers": [
    {
        "name": "kea-dhcp4",
        "output_options": [
            {
                "output": "/var/log/kea-dhcp4.log"
            }
        ],
        "severity": "INFO",
        "debuglevel": 0
    }
  ]
}
```

CHECKING THE CONFIGURATION FOR SYNTAX ERRORS

KEA CONFIGURATION SYNTAX CHECK

- After changes to a configuration file, and before reloading the new configuration into the Kea server, the configuration file should be checked for errors
    - Syntax checks can be done with the `-t` (test) parameter

KEA CONFIGURATION CHECK EXAMPLES (1/2)

- KEA-DHCP4 Syntax Check with error (line 33, char 9)

```
# kea-dhcp4 -t /etc/kea/kea-dhcp4.conf
Syntax check failed with: /etc/kea/kea-dhcp4.conf:33.9: syntax erro
```

KEA CONFIGURATION CHECK EXAMPLES (2/2)

- Successful Syntax check (output wrapped for readability)

```
# kea-dhcp4 -t /etc/kea/kea-dhcp4.conf
INFO  [kea-dhcp4.dhcpsrv/51] DHCPSRV_CFGMGR_ADD_IFACE listening on
INFO  [kea-dhcp4.dhcpsrv/51] DHCPSRV_CFGMGR_SOCKET_TYPE_DEFAULT "dh
                             using default socket type raw
INFO  [kea-dhcp4.dhcpsrv/51] DHCPSRV_CFGMGR_NEW_SUBNET4 a new subne
                             192.0.2.0/24 with params: t1=900, t2=1
```

KEA CONTROL AGENT

KEA CONTROL AGENT

- The Kea control agent is a process that provides a HTTP(s) REST interface
- The control agent can be used to dynamically reconfigure the Kea services (without manually changing the configuration files)
- The kea control agent communicates with the running Kea services via unix control sockets

CONFIGURATION OF THE KEA CONTROL AGENT

- By default, the Kea control agent listens on the (first) IPv4 loopback address `127.0.0.1` Port `8000`
  - This can be changed in the configuration file `kea-ctrl-agent.conf`

READING CONFIGURATION DATA

- The REST interface has been designed to be used from a Kea configuration application (such as Kea Stork or Kea Shell)
- However, API calls can be send to the Kea control agent from the command line via the `curl` tool
- Here we send the `config-get` command to the DHCPv4 server

```
[kea-server]# curl -X POST -H "Content-Type: application/json" \
  -d '{ "command": "config-get", "service": [ "dhcp4" ] }' \
  http://127.0.0.1:8000/
```

PRETTY PRINTING THE JSON OUTPUT

- The output is unformatted JSON. The tool `jq` can be used to pretty-print the output

```
[kea-server]# curl -X POST -H "Content-Type: application/json" \
  -d '{ "command": "config-get", "service": [ "dhcp4" ] }' \
  http://127.0.0.1:8000/ | jq
```

JSON QUERIES WITH JQ

- `jq` can be used to filter specific parts of the configuration. The `jq` filter "`.[0].arguments`" can be used to produce a valid KEA configuration file.
    - The example below prints the logging config of the DHCPv4 server:

```
[kea-server]# curl -X POST -H "Content-Type: application/json" \
  -d '{ "command": "config-get", "service": [ "dhcp4" ] }' \
  http://127.0.0.1:8000/ | jq ".[0].arguments.Dhcp4.loggers"
```

Result:

```
[
  {
    "debuglevel": 0,
    "name": "kea-dhcp4",
    "output_options": [
      {
        "output": "/opt/kea/var/log/kea-dhcp4.log"
      }
    ],
    "severity": "INFO"
  }
]
```

LIST API COMMANDS

- The `list-commands` command returns the API commands available for a specific KEA module

```
[kea-server]# curl -X POST -H "Content-Type: application/json" \
  -d '{ "command": "list-commands", "service": [ "dhcp4" ] }' \
  http://127.0.0.1:8000/ | jq
```

DYNAMIC CHANGES TO THE KEA CONFIGURATION FILE (1/5)

- With the REST API, it is possible to
  - Remotely fetch the current running config of a Kea server
  - Change the config
  - And write the config back to the server

DYNAMIC CHANGES TO THE KEA CONFIGURATION FILE (2/5)

- Dump the current configuration into a file

```
curl -s -X POST -H "Content-Type: application/json" \
     -d '{ "command": "config-get",  "service": [ "dhcp4" ] }' \
     http://127.0.0.1:8000/ | jq ".[0]" > kea-dhcp4.tmp
```

DYNAMIC CHANGES TO THE KEA CONFIGURATION FILE (3/5)

- Edit the file
  - Add the `command` and `service` information
  - Make changes to the configuration
  - Remove the `result` from the JSON file

```
{
    "command": "config-set",
    "service": [ "dhcp4" ],
    "arguments": {
        "Logging": {
            "loggers": [
                {
                    "severity": "INFO",
                    "output_options": [
[...]
```

DYNAMIC CHANGES TO THE KEA CONFIGURATION FILE (4/5)

- Send the new configuration to the server

```
[kea-server]# curl -s -X POST -H "Content-Type: application/json" \
   -d @kea-dhcp4.tmp http://127.0.0.1:8000/ | jq
[
  {
    "result": 0,
    "text": "Configuration successful."
  }
]
```

DYNAMIC CHANGES TO THE KEA CONFIGURATION FILE (5/5)

- All dynamic changes are stored in memory
  - To make the changes persistent, write the in-memory configuration back to a file with the `config-write` command (be careful, any comments in the file will be gone and the formatting will be different)

```
[kea-server]# curl -s -X POST -H "Content-Type: application/json" \
                  -d '{ "command": "config-write", "arguments": { "filename": "/etc
                  http://127.0.0.1:8000/ | jq
[
  {
    "arguments": {
      "filename": "/etc/kea/kea-dhcp4-new.json",
      "size": 3248
    },
    "result": 0,
    "text": "Configuration written to /etc/kea/kea-dhcp4-new.json successful"
  }
]
```

FROM ISC-DHCP TO KEA DHCP

ABOUT MIGRATIONS

- Migrations from old to new software products are required from time to time
- Not all software can be upgraded to new functionality
  - Sometimes a clean start is needed to update the underlying implementation
  - Such new implementations often cannot provide 100% compatibility
  - Examples: BIND 8 -> BIND 9, Python2 -> Python3, ISC DHCP -> Kea DHCP

MIGRATION STRATEGIES

- Possible migration strategies
  - Upgrade of the existing infrastructure
  - Re-design of the existing infrastructure

RE-DESIGN OF AN EXISTING INFRASTRUCTURE

- A re-design can be used to remove shortcomings of the old design
  - A migration is a good time to review the current design
  - A new design can make use of modern features that were not available at the time of the original design
  - Maybe it is time to introduce IPv6?
  - Implement new features
    - Automatic configuration management
    - Monitoring
    - Better documentation

KEA MIGRATION ASSISTANT (KEAMA)

- The Kea Migration Assistant (KeaMA) is a tool provided by ISC to help migrating an existing ISC DHCP configuration to Kea DHCP
  - The output of KeaMA can be used as a starting point for creating a Kea DHCP configuration
  - Don't expect miracles from KeaMA: in most cases, it will not be able to convert the complete ISC DHCP configuration

WHAT KEAMA CAN DO

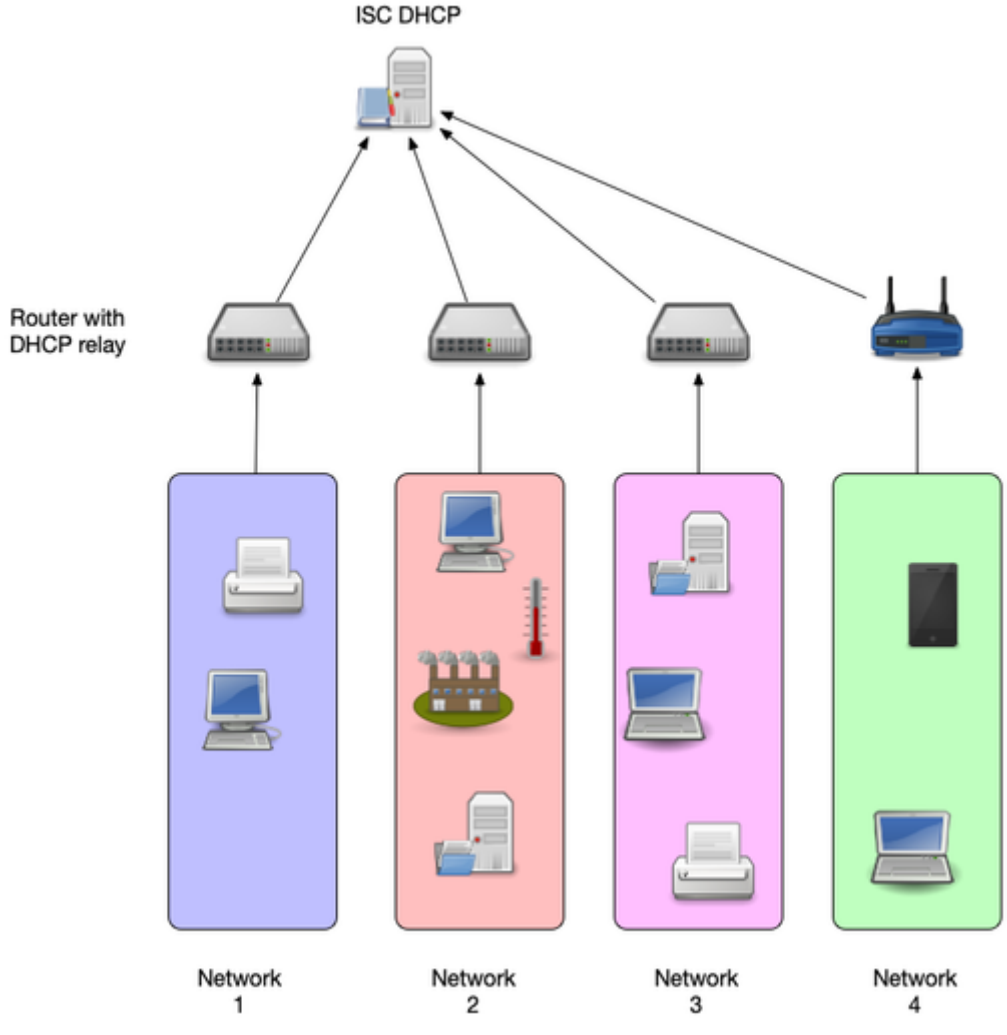
- KeaMA is good at converting simple configuration structures
  - Reservations
  - Subnet and Shared-Network definitions
  - DHCP Option definitions

WHAT KEAMA CANNOT DO

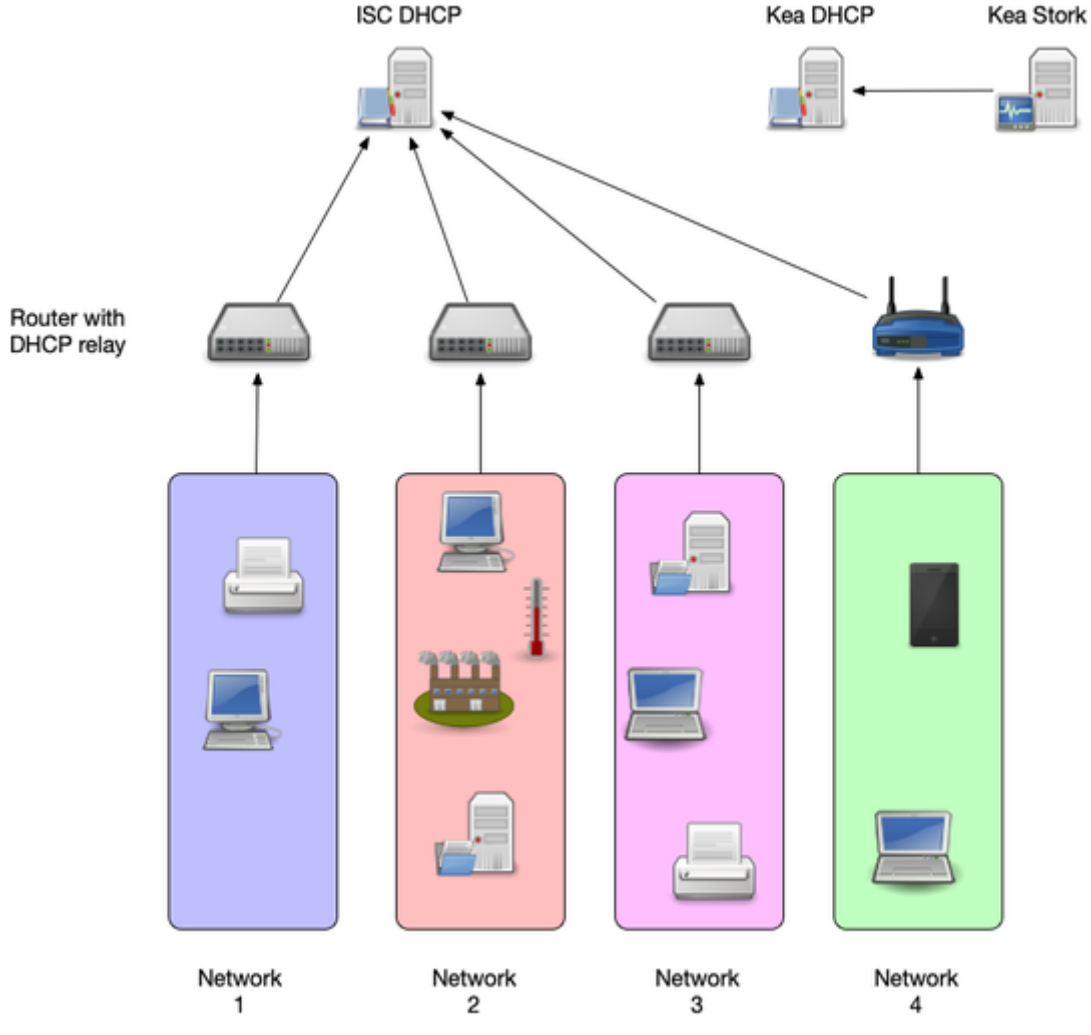
- There are some ISC DHCP configuration statements that cannot be translated into a Kea DHCP configuration
  - configurations where ISC DHCP and Kea DHCP differ
    - Failover vs. High-Availability
    - Support for hardware types other than Ethernet
    - Shared-Network pools
    - ISC DHCP spawning classes
    - …

ISC DHCP TO KEA DHCP MIGRATION PLAN

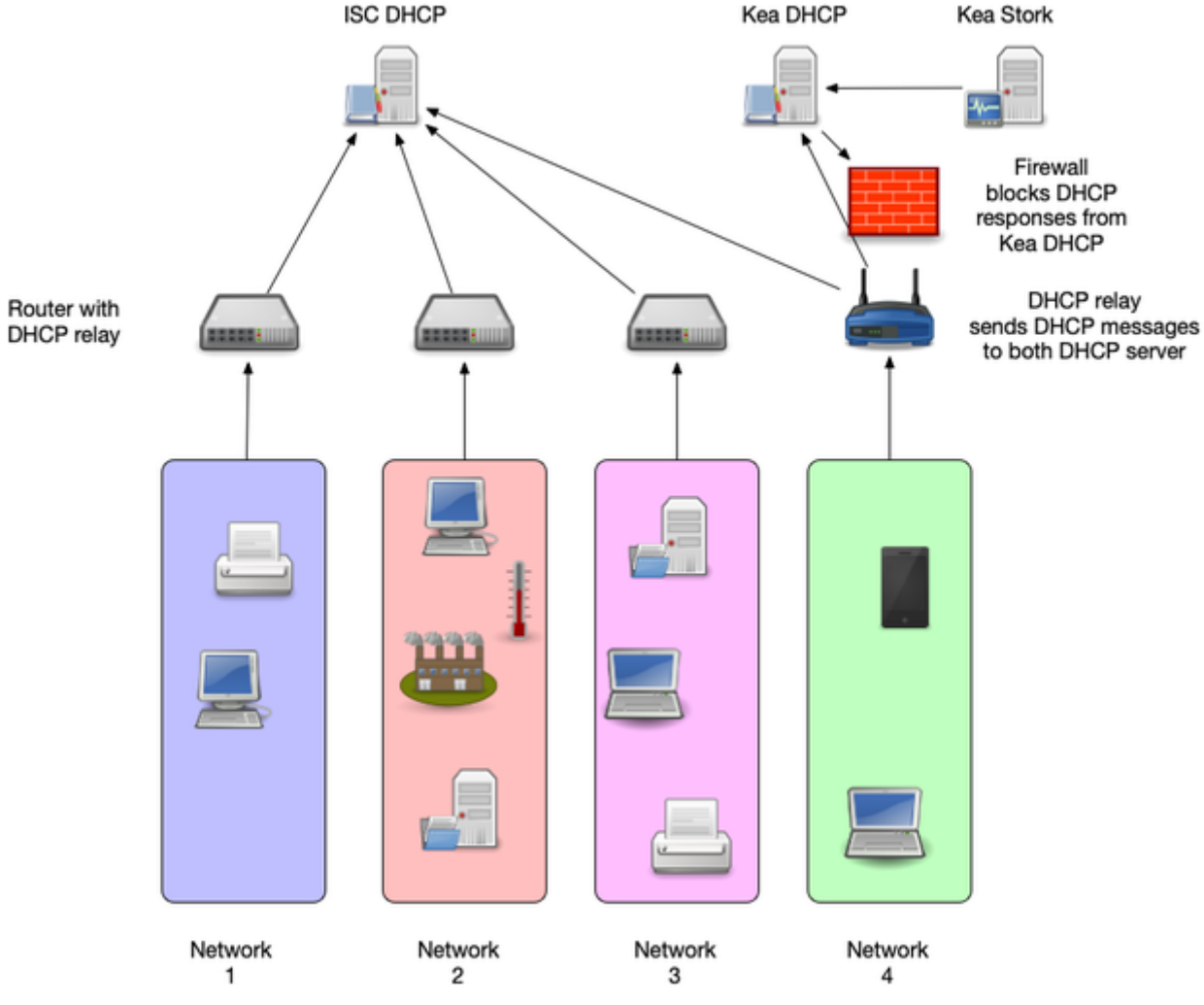
PROPOSED MIGRATION STEPS



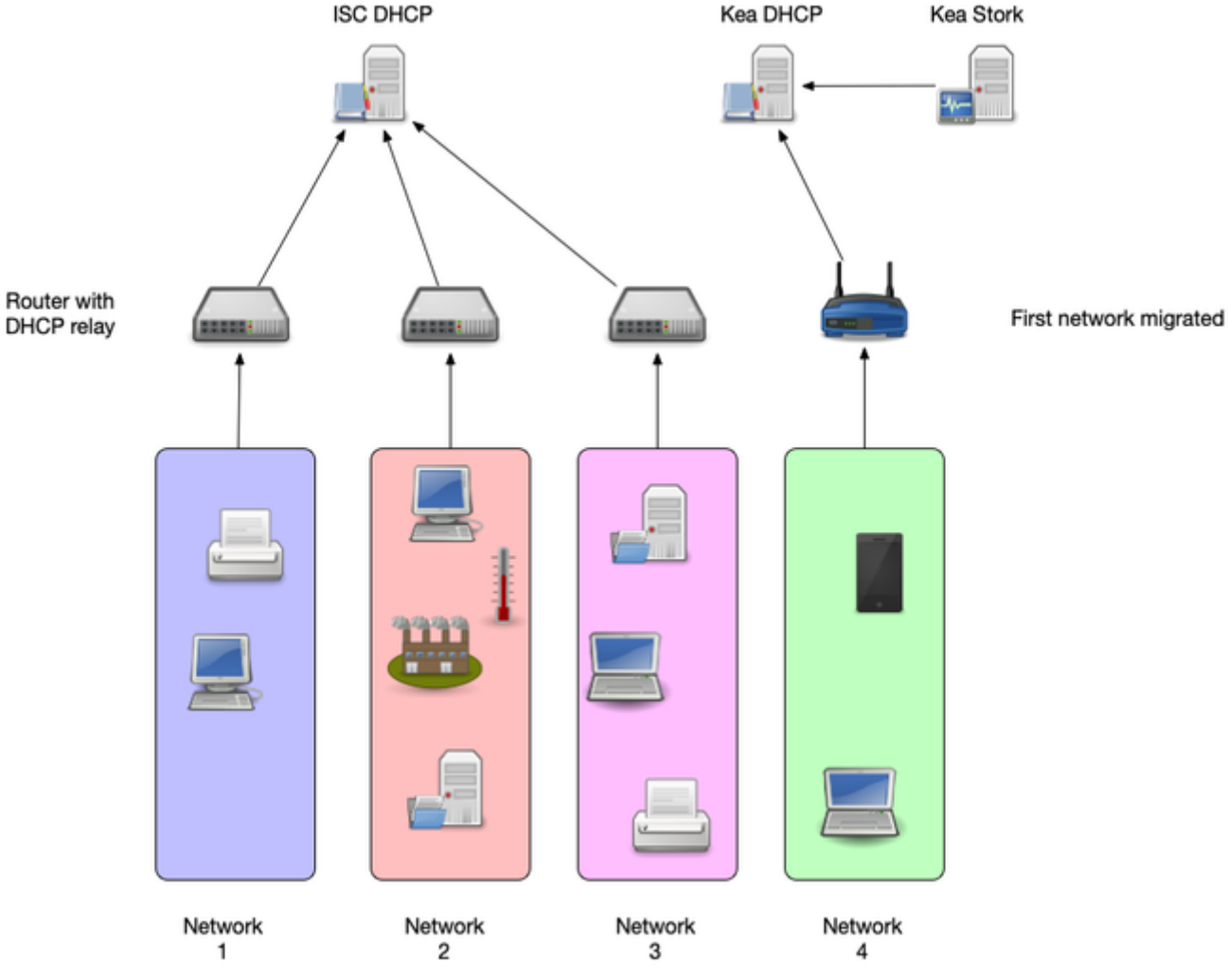
PROPOSED MIGRATION STEPS



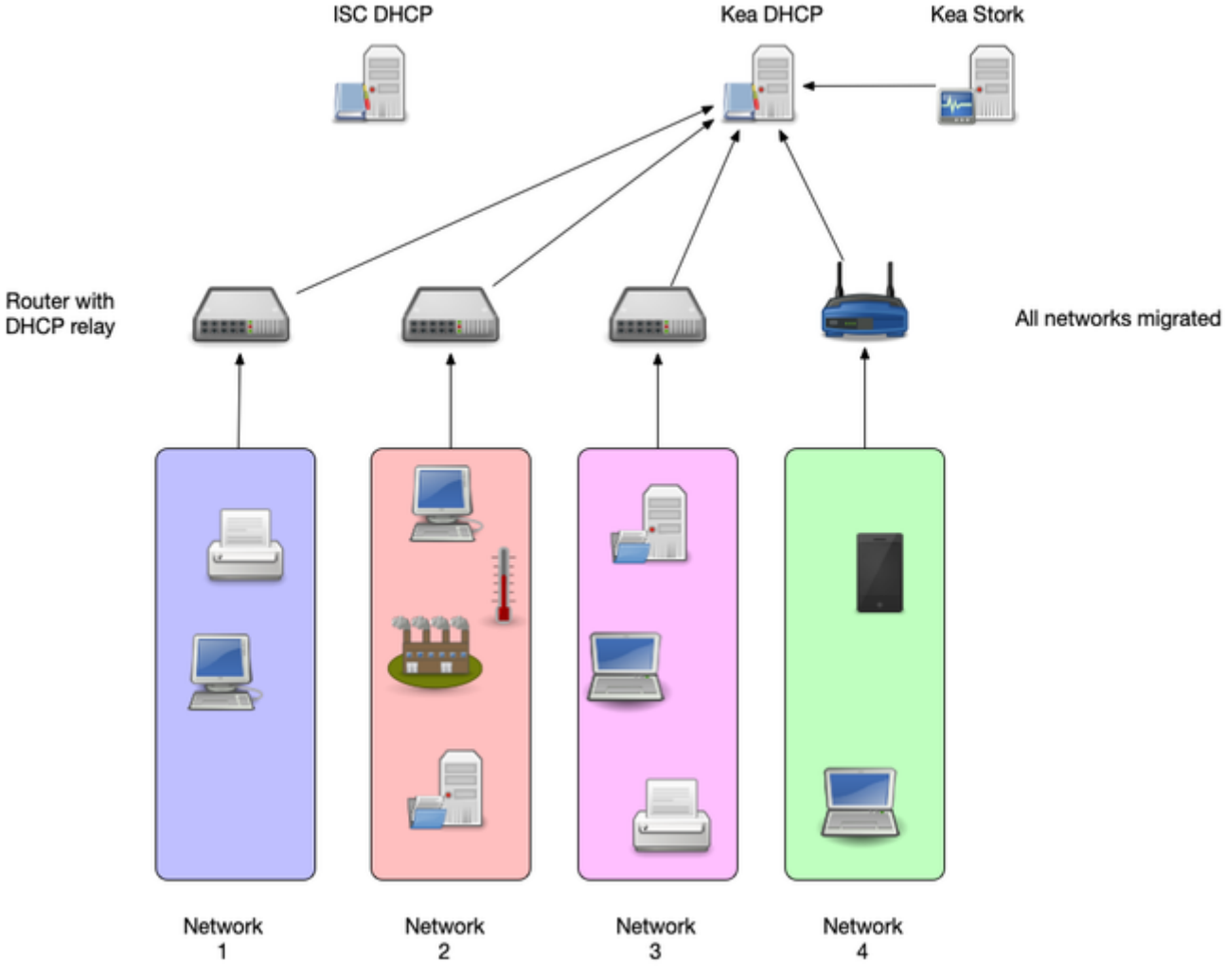
PROPOSED MIGRATION STEPS



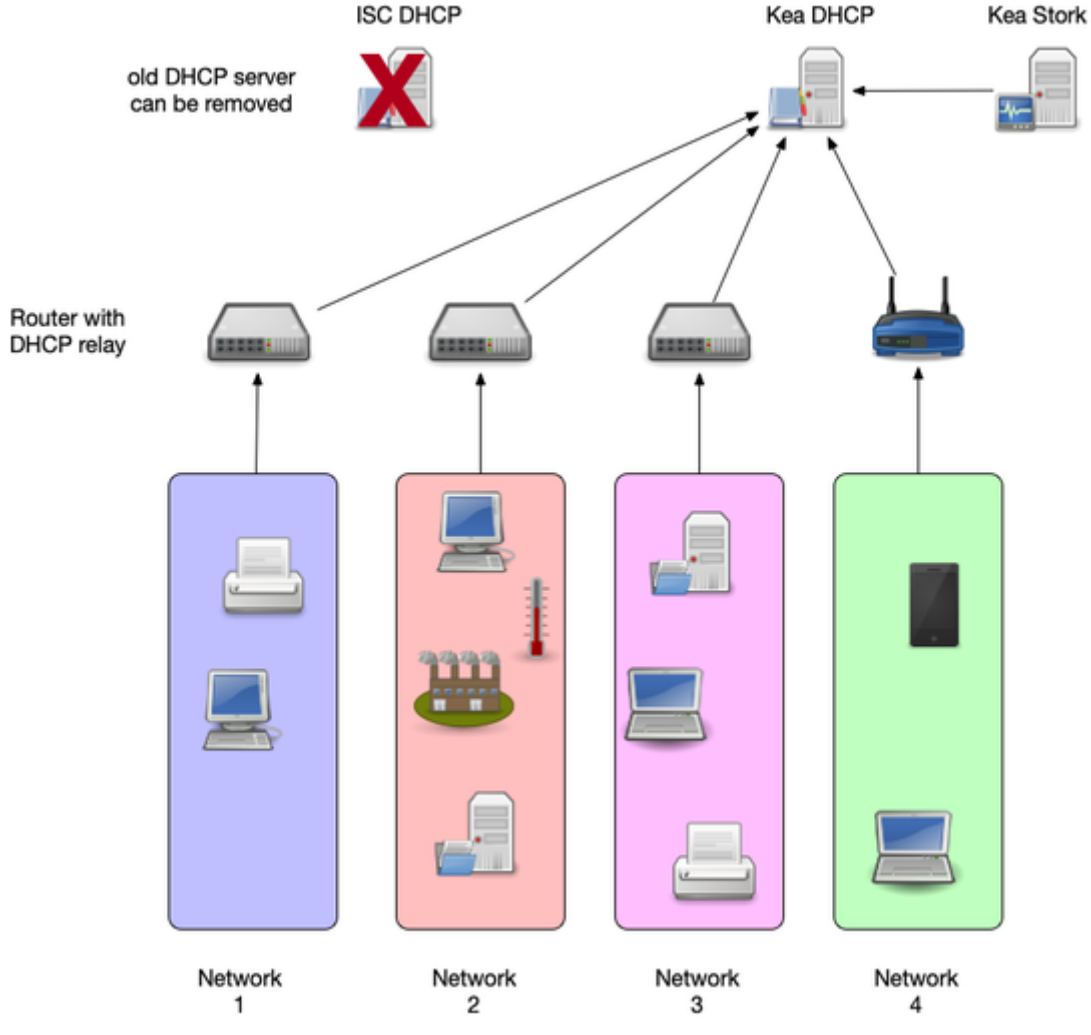
PROPOSED MIGRATION STEPS



PROPOSED MIGRATION STEPS



PROPOSED MIGRATION STEPS



QUESTIONS?

REFERENCES: RFCS, BOOKS, RECOMMENDED WEBPAGES

INTERNET STANDARDS

- DHCPv4
  - RFC 2131 - DHCPv4 Dynamic Host Configuration Protocol
  - RFC 2132 - DHCP Options and BOOTP Vendor Extensions
  - RFC 3396 - Encoding Long Options in the Dynamic Host Configuration Protocol (DHCPv4)
  - RFC 4361 - Node-specific Client Identifiers for Dynamic Host Configuration Protocol Version Four (DHCPv4)
  - RFC 6842 - Client Identifier Option in DHCP

    - RFC 6842 - Client Identifier Option in DHCP Server Replies
- DHCPv6
    - RFC 8415 - Dynamic Host Configuration Protocol for IPv6 (DHCPv6)

BOOKS

- The DHCP Handbook - Understanding, Deploying, and Managing Automated Configuration Services (Ralph Droms, Ted Lemon) 1999
- IP Address Management - Principles and Practice (Timothy Rooney) 2011
- The TCP/IP Guide - A Comprehensive, Illustrated Internet Protocols Reference (Charles M. Kozierok) 2005
- Windows Server 2019 Inside Out (Orin Thomas)

WEBSITES

- ISC Kea Documentation - https://kea.readthedocs.io
- ISC Knowledgebase - https://kb.isc.org/
- The TCP Guide - http://www.tcpipguide.com/
- Microsoft - Dynamic Host Configuration Protocol (DH
  server/networking/technologies/dhcp/dhcp-top
- Alan Clegg – NANOG 76 - DHCP Migration to Kea
  - https://pc.nanog.org/static/published/meeting
- Kea High Availability vs ISC DHCP Failover
  - https://kb.isc.org/docs/aa-01617
- Kea HA Design Document
  - https://gitlab.isc.org/isc-projects/kea/-/wikis/d
- Available 3rd party hooks for Kea DHCP

- Available 3rd party hooks for Kea DHCP
  - https://gitlab.isc.org/isc-projects/kea/-/wikis/H
- Using Host Reservations in Kea
  - https://kb.isc.org/docs/what-are-host-reservat

EXTRA

DIFFERENCES BETWEEN KEA AND ISC DHCP

SHARED NETWORKS

- ISC DHCP permits pools at shared-network level
- In Kea DHCP, a pool must belong to a subnet
- In Kea DHCP, selecting a lease from a shared-network has a performance penalty compared to selecting a lease from a plain subnet

CLIENT CLASSIFICATION

- Kea DHCP does not have a concept similar to `permit` or `deny` in ISC DHCP
  - In ISC DHCP, this is used to permit or deny certain client classes in subnets
  - This can be done in Kea DHCP as well, but the logic is different
- Kea DHCP does not support *spawning* classes (which are used for dynamic lease limit configurations)
  - There is (currently) no per client / per class lease limit in Kea DHCP

DHCP OPTIONS

- ISC DHCP can resolve DNS names to IP addresses for options that require an IP address. Kea DHCP does not resolve DNS names
  - The Kea Migration Assistant can resolve the DNS names into IP addresses while converting an ISC DHCP configuration
- Option inheritance scoping is different between ISC DHCP and Kea DHCP

HIGH-AVAILABILITY

- ISC DHCP supports the IETF DHCPv4 Failover Protocol draft https://datatracker.ietf.org/doc/html/draft-ietf-dhc-failover
- Kea DHCP supports its own DHCP high availability implementation for DHCPv6 and DHCPv4
- Both create a highly available DHCP service, but the implementation and configuration details are different
- Document comparing failover in ISC DHCP to HA in Kea DHCP https://kb.isc.org/docs/aa-01617

KEA HIGH AVAILABILITY VS ISC DHCP FAILOVER (1)

- Number of servers in an HA cluster
  - ISC DHCP: 2
  - Kea DHCP: 2 active + unlimited backup servers
- Failover relationships
  - ISC DHCP: one per subnet
  - Kea DHCP: one per instance
- Load balancing
  - ISC DHCP: Flexible split (RFC3074)
  - Kea DHCP: fixed 50/50 split (RFC3074)

KEA HIGH AVAILABILITY VS ISC DHCP FAILOVER (2)

- Lazy lease updates (MCLT)
  - ISC DHCP: yes (server responds to the client immediately)
  - Kea DHCP: no (server waits for lease update completion before responding to client)
- Send lease updates to external entity
  - ISC DHCP: no
  - Kea DHCP: yes (via backup server or custom hook library)
- Rebalancing pools
  - ISC DHCP: yes
  - Kea DHCP: no

KEA HIGH AVAILABILITY VS ISC DHCP FAILOVER (3)

- Database replication for sharing lease info
  - ISC DHCP: no
  - Kea DHCP: yes (optional)
- API
  - ISC DHCP: omapi
  - Kea DHCP: RESTful API

HOST RESERVATIONS

- In ISC DHCP all host declarations are global
- Kea DHCP supports global and per-subnet/shared-network reservations
- ISC DHCP can have reservations that are not viable on the subnet where the clients are attached
    - Kea does not start if configured with an address or prefix that is not viable on its subnet; it displays an error

SCRIPTS AND TOOLS

- Most 3rd party tools and scripts that work with ISC DHCP do not work with Kea DHCP
  - parse lease file
  - SNMP monitoring agents
  - Configuration orchestration
- Many use cases are covered by the Kea API