SH Dr. Soufiane Hourri

drsoufiane.com

DUT · SEMESTER 3 · COMPUTER SCIENCE

Network Administration

Session 04 · Dynamic DNS: DHCP registers names in DNS

Dr. Soufiane Hourri, PhD, HDR

Associate Professor & AI Consultant

École Supérieure de Technologie de Safi · Cadi Ayyad University

2026-2027

SESSION 04 · OVERVIEW

Today's session

01 The problem and the idea

02 Who may change the zone?

03 Configuration in five steps

04 When leases end

TODAY'S FOCUS

DHCP knows which machine has which address. DNS knows names. Today the DHCP server tells the DNS server, every time a lease changes.

Learning objectives

1. Explain why client names need dynamic updates, and trace an update from DHCP to DNS
2. Explain how a TSIG key protects dynamic updates
3. Configure BIND and Kea so that srv1 registers clients in lab.internal
4. Check the result, and explain what happens when a lease ends or a name is taken

PREREQUISITES

- Session 02: Kea, leases, the host name option
- Session 03: zones, A and PTR records, serial, dig
- Practicals 02 and 03: srv1 and ns1 working

The lab after Session 03

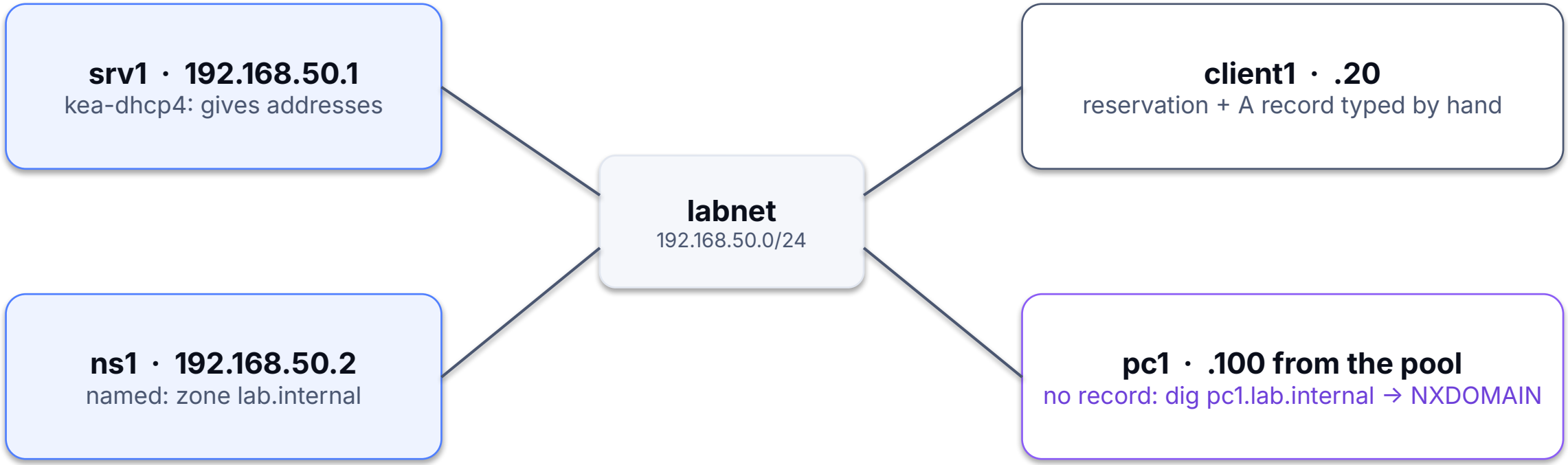


Figure 1. labnet before DDNS. Blue = servers with fixed addresses, white = DHCP clients. pc1 has an address but no name in DNS.

01

CHAPTER

The problem and the idea

Why a zone typed by hand falls behind, and how DHCP can keep it up to date

Why a hand-written zone falls behind

- Pool addresses change
  - pc1 has .100 today; after a long absence it may come back with .137
- Every change means editing two files on ns1
  - A record, PTR record, serial, check, reload: for each client, each time
- Mistakes stay in the zone
  - A forgotten record sends users to the wrong machine, or to nobody

KEY TAKEAWAY

Hand-written records are fine for servers with fixed addresses. For DHCP clients, the zone must follow the leases automatically.

What dynamic DNS is

DEFINITION

Dynamic DNS (DDNS) means that records in a zone are added or deleted over the network, while the DNS server runs, without editing the zone file. Here, the DHCP server sends these changes each time it gives or takes back an address.

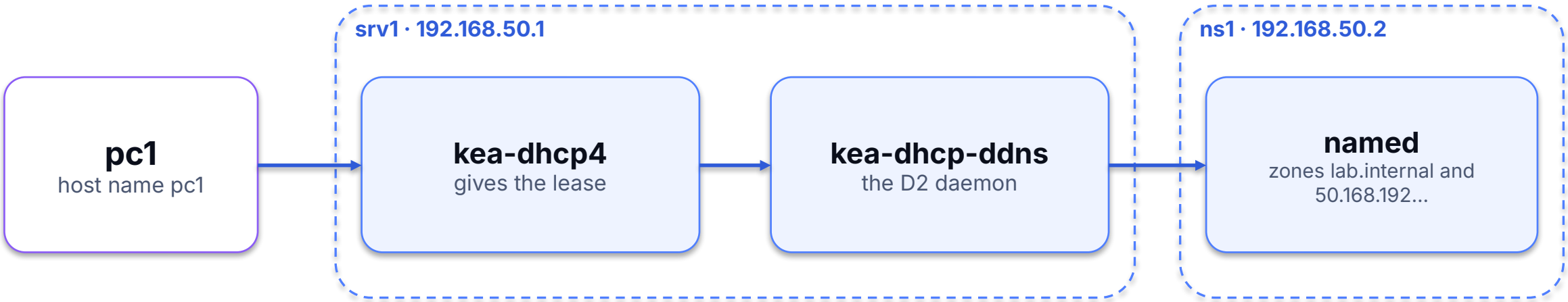
EXAMPLE

pc1 receives 192.168.50.100. A moment later ns1 answers pc1.lab.internal → 192.168.50.100, and the reverse lookup of .100 → pc1.lab.internal.

THE STANDARD BEHIND IT

A DNS UPDATE message (RFC 2136, 1997) asks a primary server to add or delete records. It travels on port 53, like a query.

The path of an update in our lab



1 DORA

pc1 puts its host name in the REQUEST (option 12 or 81)

2 Name change request

kea-dhcp4 asks D2: add pc1 → .100. Local only: 127.0.0.1, port 53001

3 DNS UPDATE

D2 signs the update with the TSIG key and sends it to ns1, port 53

4 Records written

named checks the key, adds A, PTR and DHCID, raises the serial

Figure 2. From a DHCP lease to DNS records. The two Kea programs run on srv1; named runs on ns1.

Who does what

Program	Runs on	Configuration file	Its job in DDNS
kea-dhcp4	srv1	`/etc/kea/kea-dhcp4.conf`	Gives leases; asks D2 to add or remove names
kea-dhcp-ddns (D2)	srv1	`/etc/kea/kea-dhcp-ddns.conf`	Sends signed DNS UPDATE messages to ns1
named	ns1	`/etc/named.conf + zone files`	Checks the key; writes the records
NetworkManager	pc1	`(nothing to change)`	Sends the host name in its DHCP request

Table 1. The four programs involved. Only the first three need configuration today.

Knowledge check

QUESTION

Which program sends the DNS UPDATE message to ns1?

A

pc1, the client

B

kea-dhcp4

C

kea-dhcp-ddns

D

named on srv1

02

CHAPTER

Who may change the zone?

Updates arrive over the network, so ns1 must know who sent them

Test: an update without a key

SRV1 · NSUPDATE, NO KEY

```
$ nsupdate
> server 192.168.50.2
> zone lab.internal
> update add evil.lab.internal 300 A 10.6.6.6
> send
update failed: REFUSED

# nsupdate is the command-line tool
# for DNS UPDATE (package bind-utils)
```

LOG ON NS1

```
update-security: error:
client 192.168.50.1#37471:
update 'lab.internal/IN'
denied
```

TSIG: updates signed with a shared key

DEFINITION

TSIG (Transaction SIGnature, RFC 8945) protects a DNS message with a secret key known only to the sender and the server. The sender adds a signature computed from the message, the key and the current time. ns1 recomputes it and refuses any message whose signature does not match.

IN OUR LAB

One key named ddns-key, algorithm HMAC-SHA256. It is created on ns1 and copied into the D2 configuration on srv1. The key itself never travels in the messages.

WHY THE CLOCK MATTERS

The signature includes the time. By default ns1 accepts at most 5 minutes of difference: srv1 and ns1 need synchronized clocks.

Who should send the updates?

Option 1: each client updates its own name

- Every client needs credentials to write in the zone
- A client could overwrite another machine's name
- Used in Windows domains, where Active Directory provides the credentials (not covered here)

Option 2: the DHCP server updates for everyone

- One key, stored only on srv1 and ns1
- The server knows each MAC address and lease, so it can detect conflicts
- Our choice: kea-dhcp4 updates both the A and the PTR record, even if the client offers to do it

03

CHAPTER

Configuration in five steps

Two steps on ns1, two on srv1, then a test from pc1

Five steps

01

Create the key

ns1: tsig-keygen writes the shared secret

02

Open the zones to signed updates

ns1: named.conf and the dynamic folder

03

Configure D2

srv1: kea-dhcp-ddns.conf with the same key

04

Turn DDNS on in Kea

srv1: three lines in kea-dhcp4.conf

05

Test

pc1 renews; dig and the logs confirm

CH. 03 · STEP 1

Create the key on ns1

BASH · NS1

```
$ sudo tsig-keygen -a hmac-sha256 ddns-key \
      | sudo tee /etc/named/ddns.key

$ sudo chown root:named /etc/named/ddns.key
$ sudo chmod 640 /etc/named/ddns.key

# tsig-keygen comes with the bind package
# ddns-key is the name we choose for the key
```

OUTPUT (YOURS DIFFERS)

```
key "ddns-key" {
  algorithm hmac-sha256;
  secret
"Mc0jHjDGTLl771Ix3N
  +ya6RmPKDJmhkCKcsCK
  EdHInA=";
};
```

named.conf: accept updates signed with the key

NAMED.CONF · NS1

```
include "/etc/named/ddns.key";

zone "lab.internal" IN {
    type primary;
    file "dynamic/lab.internal.zone";
    allow-update { key "ddns-key"; };
};

# the same two changes in the reverse zone:
#   file "dynamic/192.168.50.rev";
#   allow-update { key "ddns-key"; };
```

LINE 1

Load the key created in step 1

LINES 5 AND 10

The zone files move to /var/named/dynamic (next slide)

LINES 6 AND 11

Accept an update only if it is signed with ddns-key

Move the zones where named may write

BASH · NS1

```
$ cd /var/named
$ sudo mv lab.internal.zone 192.168.50.rev dynamic/
$ sudo chown named:named dynamic/*
$ sudo restorecon -Rv dynamic

$ sudo named-checkconf
$ sudo named-checkzone lab.internal \
      dynamic/lab.internal.zone
$ sudo systemctl restart named
$ sudo journalctl -u named -n 10
```

WHY MOVE THE FILES?

named now writes into the zone and keeps a journal file (.jnl). SELinux lets it write only in /var/named/dynamic

LINE 4

mv keeps the old SELinux label; restorecon gives the files the label of the folder

BEFORE MOVING

Delete client1's lines: DDNS will register it too

kea-dhcp-ddns.conf on srv1

KEA-DHCP-DDNS.CONF

```
{ "DhcpDdns": {
  "ip-address": "127.0.0.1", "port": 53001,
  "tsig-keys": [ { "name": "ddns-key",
     "algorithm": "HMAC-SHA256",
     "secret": "Mc0jHjDG...KEdHInA=" } ],
  "forward-ddns": { "ddns-domains": [ {
     "name": "lab.internal.",
     "key-name": "ddns-key",
     "dns-servers": [
        { "ip-address": "192.168.50.2" } ] } ] },
  "reverse-ddns": { "ddns-domains": [ {
     "name": "50.168.192.in-addr.arpa.",
     "key-name": "ddns-key",
     "dns-servers": [
        { "ip-address": "192.168.50.2" } ] } ] } } }
```

LINE 2

D2 listens only on loopback: only programs on srv1 can reach it

LINES 3-5

The same name, algorithm and secret as on ns1

LINES 6-15

Which zones D2 updates, and the server for each. Note the final dots

kea-dhcp4.conf: three lines, then start D2

JSON + BASH · SRV1

```
  "dhcp-ddns": { "enable-updates": true },
  "ddns-qualifying-suffix": "lab.internal.",
  "ddns-override-client-update": true,

$ cd /etc/kea
$ sudo chown --reference=kea-dhcp4.conf kea-dhcp-ddns.conf
$ sudo chmod --reference=kea-dhcp4.conf kea-dhcp-ddns.conf
$ sudo kea-dhcp-ddns -t kea-dhcp-ddns.conf
$ sudo kea-dhcp4 -t kea-dhcp4.conf
$ sudo systemctl enable --now kea-dhcp-ddns
$ sudo systemctl restart kea-dhcp4
```

LINES 1-3

Inside "Dhcp4", next to valid-lifetime: send requests to D2, add lab.internal to short names, update both records

LINES 6-7

Same owner and mode as the file Kea already reads

LINES 8-11

Check both files, start D2 first, then restart kea-dhcp4

Test: pc1 renews, ns1 answers

PC1, THEN SRV1

```
# on pc1
$ hostnamectl hostname
pc1.lab.internal
$ sudo nmcli connection up labnet

# on srv1 (or any labnet machine)
$ dig @192.168.50.2 pc1.lab.internal +noall +answer
$ dig @192.168.50.2 -x 192.168.50.100 +noall +answer
$ dig @192.168.50.2 lab.internal SOA +short
```

REAL OUTPUT

```
pc1.lab.internal. 600 IN A
  192.168.50.100

100.50.168.192.in-addr.arpa.
  600 IN PTR
pc1.lab.internal.

ns1.lab.internal. admin...
  2026110902 3600 900 ...
```

CH. 03 · STEP 5

What the logs say

NS1 · NAMED LOG

```
updating zone 'lab.internal/IN':
  adding an RR at 'pc1.lab.internal' A 192.168.50.100
updating zone 'lab.internal/IN':
  adding an RR at 'pc1.lab.internal' DHCID AAEB9wL1...
updating zone '50.168.192.in-addr.arpa/IN':
  deleting rrset at '100.50.168.192.in-addr.arpa' PTR
updating zone '50.168.192.in-addr.arpa/IN':
  adding an RR at '100.50.168.192.in-addr.arpa'
  PTR pc1.lab.internal.
```

SRV1 · D2 LOG

```
DHCP_DDNS_ADD_SUCCEEDED
successfully added the DNS
mapping addition for this
request: Type: 0 (CHG_ADD)

$ ls /var/named/dynamic
192.168.50.rev
192.168.50.rev.jnl
lab.internal.zone
lab.internal.zone.jnl
```

Editing a dynamic zone with nano

× INCORRECT APPROACH

Open dynamic/lab.internal.zone in nano while named runs, add a line, reload

```
rndc: 'reload' failed: dynamic zone
```

✓ CORRECTION

```
sudo rndc freeze lab.internal
```

Edit, increase the serial, named-checkzone

```
sudo rndc thaw lab.internal
```

Or add the record with nsupdate -k /etc/named/ddns.key

WHY IT FAILS

Recent changes may exist only in the .jnl file. named refuses to reload the file, and at the next restart the file and its journal no longer match. freeze writes the journal into the file and pauses updates; thaw reloads and resumes.

04

CHAPTER

When leases end

Removing names, and refusing a name that already belongs to someone

What happens in DNS at each lease event

Lease event	Kea's action	Result in the zone
New lease with a host name	Asks D2 to add the name	A, PTR and DHCID created
Renewal, same name, same address	Nothing	No change
Released or expired	Asks D2 to remove the name	A, PTR and DHCID deleted
Another host asks for the same name	Asks D2 to add it	Refused: the first host keeps the name
Name typed by hand (no DHCID)	Asks D2 to add it	Refused: Kea never overwrites it

Table 2. Observed in the test lab: a renewal sent no update; an expired lease removed all three records.

Two VMs called pc1: the DHCID decides

SECOND VM NAMED PC1

```
# ns1 log (shortened)
update unsuccessful: pc1.lab.internal:
  'name not in use' prerequisite
  not satisfied (YXDOMAIN)
update unsuccessful: pc1.lab.internal/DHCID:
  'RRset exists (value dependent)'
  prerequisite not satisfied (NXRRSET)

$ dig @192.168.50.2 +short pc1.lab.internal
192.168.50.100
```

WHAT D2 TRIED

```
Try 1: add pc1 if
the name is free
→ refused (YXDOMAIN)

Try 2: replace pc1 if
its DHCID is mine
→ refused (NXRRSET)

DHCID: a hash of the
client's identity
```

Knowledge check

QUESTION

web.lab.internal is a hand-written A record (.3). A DHCP client also named web gets .150. What does dig web.lab.internal return?

A

192.168.50.150

B

Both addresses

C

192.168.50.3

D

NXDOMAIN

Troubleshooting: the name does not appear

What you see	Where	Most likely cause
"DDNS: disabled" when kea-dhcp4 starts	srv1 log	enable-updates missing or false
Nothing in the named log at all	srv1	kea-dhcp-ddns not running, or pc1 sent no host name
update ... denied	ns1 log	No allow-update with the key, or the zone name differs
tsig verify failure (BADSIG)	ns1 log	The secret was mistyped on srv1
FORWARD_REPLACE_REJECTED	D2 log	The name already exists: hand-written or another host
permission denied on a .jnl file	ns1 log	Zone not in dynamic/, or restorecon forgotten

Table 3. Read the logs on both servers: journalctl -u kea-dhcp4, -u kea-dhcp-ddns on srv1 and -u named on ns1.

Recap

KEY TAKEAWAYS

- DDNS: the DHCP server registers each client's name in DNS
- kea-dhcp4 → kea-dhcp-ddns (127.0.0.1:53001) → named (port 53)
- Updates are signed with a shared TSIG key; others are refused
- Dynamic zones live in /var/named/dynamic; edit them with rndc freeze and thaw
- Names are removed when leases end; the DHCID stops name theft

NEXT

- Practical: apply the five steps on ns1 and srv1, then register pc1 and client1
- Session 05: Active Directory, with a Windows Server domain controller

References

(1) Vixie, P. et al. RFC 2136: Dynamic Updates in the Domain Name System (DNS UPDATE). IETF, 1997

(2) Dupont, F. et al. RFC 8945: Secret Key Transaction Authentication for DNS (TSIG). IETF, 2020

(3) Stapp, M., Volz, B., Rekhter, Y. RFC 4702: The DHCP Client FQDN Option. IETF, 2006

(4) Stapp, M., Lemon, T., Gustafsson, A. RFC 4701: The DHCID Resource Record. IETF, 2006

(5) Stapp, M., Volz, B. RFC 4703: Resolution of FQDN Conflicts among DHCP Clients. IETF, 2006

(6) ISC. Kea Administrator Reference Manual 3.0.3, The DHCP-DDNS Server. kea.readthedocs.io

(7) ISC Knowledge Base. Things to be aware of when upgrading to Kea 3.0.0. kb.isc.org

(8) ISC Knowledge Base. Why doesn't my DNS get updated by Kea? (updated August 2023). kb.isc.org

(9) ISC. BIND 9.18 Administrator Reference Manual: dynamic update, rndc. bind9.readthedocs.io

(10) Red Hat. SELinux User's and Administrator's Guide, Berkeley Internet Name Domain. docs.redhat.com

Course page:
drsoufiane.com

SH

Thank you · Questions?

Next: register pc1 and client1 automatically in the practical session

Dr. Soufiane Hourri, PhD, HDR

drsoufiane.com