

[Logiciel](#)

# Exemple de fichier unbound.conf

</usr/share/doc/unbound/examples/unbound.conf>

```
#
# Example configuration file.
#
# See unbound.conf(5) man page, version 1.9.4.
#
# this is a comment.

#Use this to include other text into the file.
#include: "otherfile.conf"

# The server clause sets the main parameters.
server:
    # whitespace is not necessary, but looks cleaner.

    # verbosity number, 0 is least verbose. 1 is default.
    verbosity: 1

    # print statistics to the log (for every thread) every N
seconds.
    # Set to "" or 0 to disable. Default is disabled.
    # statistics-interval: 0

    # enable shm for stats, default no.  if you enable also enable
    # statistics-interval, every time it also writes stats to the
    # shared memory segment keyed with shm-key.
    # shm-enable: no

    # shm for stats uses this key, and key+1 for the shared mem
segment.
    # shm-key: 11777

    # enable cumulative statistics, without clearing them after
printing.
    # statistics-cumulative: no

    # enable extended statistics (query types, answer codes,
status)
    # printed from unbound-control. default off, because of speed.
    # extended-statistics: no

    # number of threads to create. 1 disables threading.
    # num-threads: 1
```

```
# specify the interfaces to answer queries from by ip-address.
# The default is to listen to localhost (127.0.0.1 and ::1).
# specify 0.0.0.0 and ::0 to bind to all available interfaces.
# specify every interface[@port] on a new 'interface:'
labelled line.
# The listen interfaces are not changed on reload, only on
restart.
# interface: 192.0.2.153
# interface: 192.0.2.154
# interface: 192.0.2.154@5003
# interface: 2001:DB8::5

# enable this feature to copy the source address of queries to
reply.
# Socket options are not supported on all platforms.
experimental.
# interface-automatic: no

# port to answer queries from
# port: 53

# specify the interfaces to send outgoing queries to
authoritative
# server from by ip-address. If none, the default (all)
interface
# is used. Specify every interface on a 'outgoing-interface:'
line.
# outgoing-interface: 192.0.2.153
# outgoing-interface: 2001:DB8::5
# outgoing-interface: 2001:DB8::6

# Specify a netblock to use remainder 64 bits as random bits
for
# upstream queries. Uses freebind option (Linux).
# outgoing-interface: 2001:DB8::/64
# Also (Linux:) ip -6 addr add 2001:db8::/64 dev lo
# And: ip -6 route add local 2001:db8::/64 dev lo
# And set prefer-ip6: yes to use the ip6 randomness from a
netblock.
# Set this to yes to prefer ipv6 upstream servers over ipv4.
# prefer-ip6: no

# number of ports to allocate per thread, determines the size
of the
# port range that can be open simultaneously. About double
the
# num-queries-per-thread, or, use as many as the OS will allow
you.
# outgoing-range: 4096
```

```
# permit unbound to use this port number or port range for
# making outgoing queries, using an outgoing interface.
# outgoing-port-permit: 32768

# deny unbound the use this of port number or port range for
# making outgoing queries, using an outgoing interface.
# Use this to make sure unbound does not grab a UDP port that
some
# other server on this computer needs. The default is to avoid
# IANA-assigned port numbers.
# If multiple outgoing-port-permit and outgoing-port-avoid
options
# are present, they are processed in order.
# outgoing-port-avoid: "3200-3208"

# number of outgoing simultaneous tcp buffers to hold per
thread.
# outgoing-num-tcp: 10

# number of incoming simultaneous tcp buffers to hold per
thread.
# incoming-num-tcp: 10

# buffer size for UDP port 53 incoming (SO_RCVBUF socket
option).
# 0 is system default. Use 4m to catch query spikes for busy
servers.
# so-rcvbuf: 0

# buffer size for UDP port 53 outgoing (SO_SNDBUF socket
option).
# 0 is system default. Use 4m to handle spikes on very busy
servers.
# so-sndbuf: 0

# use SO_REUSEPORT to distribute queries over threads.
# at extreme load it could be better to turn it off to
distribute even.
# so-reuseport: yes

# use IP_TRANSPARENT so the interface: addresses can be non-
local
# and you can config non-existing IPs that are going to work
later on
# (uses IP_BINDANY on FreeBSD).
# ip-transparent: no

# use IP_FREEBIND so the interface: addresses can be non-local
# and you can bind to nonexisting IPs and interfaces that are
down.
# Linux only. On Linux you also have ip-transparent that is
```

```
similar.  
# ip-freebind: no  
  
# EDNS reassembly buffer to advertise to UDP peers (the actual  
buffer  
# is set with msg-buffer-size). 1472 can solve fragmentation  
(timeouts)  
# edns-buffer-size: 4096  
  
# Maximum UDP response size (not applied to TCP response).  
# Suggested values are 512 to 4096. Default is 4096. 65536  
disables it.  
# max-udp-size: 4096  
  
# max memory to use for stream(tcp and tls) waiting result  
buffers.  
# stream-wait-size: 4m  
  
# buffer size for handling DNS data. No messages larger than  
this  
# size can be sent or received, by UDP or TCP. In bytes.  
# msg-buffer-size: 65552  
  
# the amount of memory to use for the message cache.  
# plain value in bytes or you can append k, m or G. default is  
"4Mb".  
# msg-cache-size: 4m  
  
# the number of slabs to use for the message cache.  
# the number of slabs must be a power of 2.  
# more slabs reduce lock contention, but fragment memory  
usage.  
# msg-cache-slabs: 4  
  
# the number of queries that a thread gets to service.  
# num-queries-per-thread: 1024  
  
# if very busy, 50% queries run to completion, 50% get timeout  
in msec  
# jostle-timeout: 200  
  
# msec to wait before close of port on timeout UDP. 0  
disables.  
# delay-close: 0  
  
# msec for waiting for an unknown server to reply. Increase  
if you  
# are behind a slow satellite link, to eg. 1128.  
# unknown-server-time-limit: 376
```

```
# the amount of memory to use for the RRset cache.
# plain value in bytes or you can append k, m or G. default is
"4Mb".
# rrset-cache-size: 4m

# the number of slabs to use for the RRset cache.
# the number of slabs must be a power of 2.
# more slabs reduce lock contention, but fragment memory
usage.
# rrset-cache-slabs: 4

# the time to live (TTL) value lower bound, in seconds.
Default 0.
# If more than an hour could easily give trouble due to stale
data.
# cache-min-ttl: 0

# the time to live (TTL) value cap for RRsets and messages in
the
# cache. Items are not cached for longer. In seconds.
# cache-max-ttl: 86400

# the time to live (TTL) value cap for negative responses in
the cache
# cache-max-negative-ttl: 3600

# the time to live (TTL) value for cached roundtrip times,
lameness and
# EDNS version information for hosts. In seconds.
# infra-host-ttl: 900

# minimum wait time for responses, increase if uplink is long.
In msec.
# infra-cache-min-rtt: 50

# the number of slabs to use for the Infrastructure cache.
# the number of slabs must be a power of 2.
# more slabs reduce lock contention, but fragment memory
usage.
# infra-cache-slabs: 4

# the maximum number of hosts that are cached (roundtrip,
EDNS, lame).
# infra-cache-numhosts: 10000

# define a number of tags here, use with local-zone, access-
control.
# repeat the define-tag statement to add additional tags.
# define-tag: "tag1 tag2 tag3"

# Enable IPv4, "yes" or "no".
```

```
# do-ip4: yes

# Enable IPv6, "yes" or "no".
# do-ip6: yes

# Enable UDP, "yes" or "no".
# do-udp: yes

# Enable TCP, "yes" or "no".
# do-tcp: yes

# upstream connections use TCP only (and no UDP), "yes" or
"no"
# useful for tunneling scenarios, default no.
# tcp-upstream: no

# upstream connections also use UDP (even if do-udp is no).
# useful if if you want UDP upstream, but don't provide UDP
downstream.
# udp-upstream-without-downstream: no

# Maximum segment size (MSS) of TCP socket on which the server
# responds to queries. Default is 0, system default MSS.
# tcp-mss: 0

# Maximum segment size (MSS) of TCP socket for outgoing
queries.
# Default is 0, system default MSS.
# outgoing-tcp-mss: 0

# Idle TCP timeout, connection closed in milliseconds
# tcp-idle-timeout: 30000

# Enable EDNS TCP keepalive option.
# edns-tcp-keepalive: no

# Timeout for EDNS TCP keepalive, in msec.
# edns-tcp-keepalive-timeout: 120000

# Use systemd socket activation for UDP, TCP, and control
sockets.
# use-systemd: no

# Detach from the terminal, run in background, "yes" or "no".
# Set the value to "no" when unbound runs as systemd service.
# do-daemonize: yes

# control which clients are allowed to make (recursive)
queries
# to this server. Specify classless netblocks with /size and
```

```
action.  
    # By default everything is refused, except for localhost.  
    # Choose deny (drop message), refuse (polite error reply),  
    # allow (recursive ok), allow_setrd (recursive ok, rd bit is  
forced on),  
    # allow_snoop (recursive and nonrecursive ok)  
    # deny_non_local (drop queries unless can be answered from  
local-data)  
    # refuse_non_local (like deny_non_local but polite error  
reply).  
    # access-control: 0.0.0.0/0 refuse  
    # access-control: 127.0.0.0/8 allow  
    # access-control: ::0/0 refuse  
    # access-control: ::1 allow  
    # access-control: ::ffff:127.0.0.1 allow  
  
    # tag access-control with list of tags (in "" with spaces  
between)  
    # Clients using this access control element use localzones  
that  
    # are tagged with one of these tags.  
    # access-control-tag: 192.0.2.0/24 "tag2 tag3"  
  
    # set action for particular tag for given access control  
element  
    # if you have multiple tag values, the tag used to lookup the  
action  
    # is the first tag match between access-control-tag and local-  
zone-tag  
    # where "first" comes from the order of the define-tag values.  
    # access-control-tag-action: 192.0.2.0/24 tag3 refuse  
  
    # set redirect data for particular tag for access control  
element  
    # access-control-tag-data: 192.0.2.0/24 tag2 "A 127.0.0.1"  
  
    # Set view for access control element  
    # access-control-view: 192.0.2.0/24 viewname  
  
    # if given, a chroot(2) is done to the given directory.  
    # i.e. you can chroot to the working directory, for example,  
    # for extra security, but make sure all files are in that  
directory.  
    #  
    # If chroot is enabled, you should pass the configfile (from  
the  
    # commandline) as a full path from the original root. After  
the  
    # chroot has been performed the now defunct portion of the  
config  
    # file path is removed to be able to reread the config after a
```

```
reload.  
#  
# All other file paths (working dir, logfile, roothints, and  
# key files) can be specified in several ways:  
#   o as an absolute path relative to the new root.  
#   o as a relative path to the working directory.  
#   o as an absolute path relative to the original root.  
# In the last case the path is adjusted to remove the unused  
portion.  
#  
# The pid file can be absolute and outside of the chroot, it  
is  
# written just prior to performing the chroot and dropping  
permissions.  
#  
# Additionally, unbound may need to access /dev/random (for  
entropy).  
# How to do this is specific to your OS.  
#  
# If you give "" no chroot is performed. The path must not end  
in a /.  
# chroot: "/etc/unbound"  
  
# if given, user privileges are dropped (after binding port),  
# and the given username is assumed. Default is user  
"unbound".  
# If you give "" no privileges are dropped.  
# username: "unbound"  
  
# the working directory. The relative files in this config are  
# relative to this directory. If you give "" the working  
directory  
# is not changed.  
# If you give a server: directory: dir before include: file  
statements  
# then those includes can be relative to the working  
directory.  
# directory: "/etc/unbound"  
  
# the log file, "" means log to stderr.  
# Use of this option sets use-syslog to "no".  
# logfile: ""  
  
# Log to syslog(3) if yes. The log facility LOG_DAEMON is used  
to  
# log to. If yes, it overrides the logfile.  
# use-syslog: yes  
  
# Log identity to report. if empty, defaults to the name of  
argv[0]
```



```
# (usually "unbound").
# log-identity: ""

# print UTC timestamp in ascii to logfile, default is epoch in
seconds.
# log-time-ascii: no

# print one line with time, IP, name, type, class for every
query.
# log-queries: no

# print one line per reply, with time, IP, name, type, class,
rcode,
# timetoresolve, fromcache and responsesize.
# log-replies: no

# log with tag 'query' and 'reply' instead of 'info' for
# filtering log-queries and log-replies from the log.
# log-tag-queryreply: no

# log the local-zone actions, like local-zone type inform is
enabled
# also for the other local zone types.
# log-local-actions: no

# print log lines that say why queries return SERVFAIL to
clients.
# log-servfail: no

# the pid file. Can be an absolute path outside of chroot/work
dir.
# pidfile: "/etc/unbound/unbound.pid"

# file to read root hints from.
# get one from https://www.internic.net/domain/named.cache
# root-hints: ""

# enable to not answer id.server and hostname.bind queries.
# hide-identity: no

# enable to not answer version.server and version.bind
queries.
# hide-version: no

# enable to not answer trustanchor.unbound queries.
# hide-trustanchor: no

# the identity to report. Leave "" or default to return
hostname.
# identity: ""
```

```
# the version to report. Leave "" or default to return package
version.
# version: ""

# the target fetch policy.
# series of integers describing the policy per dependency
depth.
# The number of values in the list determines the maximum
dependency
# depth the recursor will pursue before giving up. Each
integer means:
#   -1 : fetch all targets opportunistically,
#   0: fetch on demand,
#   positive value: fetch that many targets
opportunistically.
# Enclose the list of numbers between quotes ("").
# target-fetch-policy: "3 2 1 0 0"

# Harden against very small EDNS buffer sizes.
# harden-short-bufsize: no

# Harden against unseemly large queries.
# harden-large-queries: no

# Harden against out of zone rrsets, to avoid spoofing
attempts.
# harden-glue: yes

# Harden against receiving dnssec-stripped data. If you turn
it
# off, failing to validate dnskey data for a trustanchor will
# trigger insecure mode for that zone (like without a
trustanchor).
# Default on, which insists on dnssec data for trust-anchored
zones.
# harden-dnssec-stripped: yes

# Harden against queries that fall under dnssec-signed
nxdomain names.
# harden-below-nxdomain: yes

# Harden the referral path by performing additional queries
for
# infrastructure data. Validates the replies (if possible).
# Default off, because the lookups burden the server.
Experimental
# implementation of draft-wijnngaards-dnsexp-resolver-side-
mitigation.
# harden-referral-path: no
```

```
# Harden against algorithm downgrade when multiple algorithms
are
# advertised in the DS record. If no, allows the weakest
algorithm
# to validate the zone.
# harden-algo-downgrade: no

# Sent minimum amount of information to upstream servers to
enhance
# privacy. Only sent minimum required labels of the QNAME and
set QTYPE
# to A when possible.
# qname-minimisation: yes

# QNAME minimisation in strict mode. Do not fall-back to
sending full
# QNAME to potentially broken nameservers. A lot of domains
will not be
# resolvable when this option is enabled.
# This option only has effect when qname-minimisation is
enabled.
# qname-minimisation-strict: no

# Aggressive NSEC uses the DNSSEC NSEC chain to synthesize
NXDOMAIN
# and other denials, using information from previous NXDOMAINs
answers.
# aggressive-nsec: no

# Use 0x20-encoded random bits in the query to foil spoof
attempts.
# This feature is an experimental implementation of draft
dns-0x20.
# use-caps-for-id: no

# Domains (and domains in them) without support for dns-0x20
and
# the fallback fails because they keep sending different
answers.
# caps-whitelist: "licdn.com"
# caps-whitelist: "senderbase.org"

# Enforce privacy of these addresses. Strips them away from
answers.
# It may cause DNSSEC validation to additionally mark it as
bogus.
# Protects against 'DNS Rebinding' (uses browser as network
proxy).
# Only 'private-domain' and 'local-data' names are allowed to
have
# these private addresses. No default.
```

```
# private-address: 10.0.0.0/8
# private-address: 172.16.0.0/12
# private-address: 192.168.0.0/16
# private-address: 169.254.0.0/16
# private-address: fd00::/8
# private-address: fe80::/10
# private-address: ::ffff:0:0/96

# Allow the domain (and its subdomains) to contain private
addresses.
# local-data statements are allowed to contain private
addresses too.
# private-domain: "example.com"

# If nonzero, unwanted replies are not only reported in
statistics,
# but also a running total is kept per thread. If it reaches
the
# threshold, a warning is printed and a defensive action is
taken,
# the cache is cleared to flush potential poison out of it.
# A suggested value is 10000000, the default is 0 (turned
off).
# unwanted-reply-threshold: 0

# Do not query the following addresses. No DNS queries are
sent there.
# List one address per entry. List classless netblocks with
/size,
# do-not-query-address: 127.0.0.1/8
# do-not-query-address: ::1

# if yes, the above default do-not-query-address entries are
present.
# if no, localhost can be queried (for testing and debugging).
# do-not-query-localhost: yes

# if yes, perform prefetching of almost expired message cache
entries.
# prefetch: no

# if yes, perform key lookups adjacent to normal lookups.
# prefetch-key: no

# deny queries of type ANY with an empty response.
# deny-any: no

# if yes, Unbound rotates RRSset order in response.
# rrsset-roundrobin: no
```

```
# if yes, Unbound doesn't insert authority/additional sections
# into response messages when those sections are not required.
# minimal-responses: yes

# true to disable DNSSEC lameness check in iterator.
# disable-dnssec-lame-check: no

# module configuration of the server. A string with
identifiers
# separated by spaces. Syntax: "[dns64] [validator] iterator"
# most modules have to be listed at the beginning of the line,
# except cachedb(just before iterator), and python (at the
beginning,
# or, just before the iterator).
# module-config: "validator iterator"

# File with trusted keys, kept uptodate using RFC5011 probes,
# initial file like trust-anchor-file, then it stores
metadata.
# Use several entries, one per domain name, to track multiple
zones.
#
# If you want to perform DNSSEC validation, run unbound-anchor
before
# you start unbound (i.e. in the system boot scripts). And
enable:
# Please note usage of unbound-anchor root anchor is at your
own risk
# and under the terms of our LICENSE (see that file in the
source).
# auto-trust-anchor-file: "/etc/unbound/root.key"

# trust anchor signaling sends a RFC8145 key tag query after
priming.
# trust-anchor-signaling: yes

# Root key trust anchor sentinel (draft-ietf-dnsop-kskroll-
sentinel)
# root-key-sentinel: yes

# File with DLV trusted keys. Same format as trust-anchor-
file.
# There can be only one DLV configured, it is trusted from
root down.
# DLV is going to be decommissioned. Please do not use it any
more.
# dlv-anchor-file: "dlv.isc.org.key"

# File with trusted keys for validation. Specify more than one
file
# with several entries, one file per entry.
```

```
# Zone file format, with DS and DNSKEY entries.
# Note this gets out of date, use auto-trust-anchor-file
please.
# trust-anchor-file: ""

# Trusted key for validation. DS or DNSKEY. specify the RR on
a
# single line, surrounded by "". TTL is ignored. class is IN
default.
# Note this gets out of date, use auto-trust-anchor-file
please.
# (These examples are from August 2007 and may not be valid
anymore).
# trust-anchor: "nlnetlabs.nl. DNSKEY 257 3 5
AQPzzTWMz8qSWIQLfRnPckx2BiVmKVN6LPup03mbz7FhLSnm26n6iG9N
Lby97Ji453aWZY3M5/xJBS0S2vWtco2t8C0+xe01bc/d6ZTy32DHchpW
6rDH1vp86Ll+ha0tmwyy9QP7y2bVw5zSbFCrefk8qCUBgfHm9bHzMG1U BYtEIQ=="
# trust-anchor: "jelte.nlnetlabs.nl. DS 42860 5 1
14D739EB566D2B1A5E216A0BA4D17FA9B038BE4A"

# File with trusted keys for validation. Specify more than one
file
# with several entries, one file per entry. Like trust-anchor-
file
# but has a different file format. Format is BIND-9 style
format,
# the trusted-keys { name flag proto algo "key"; }; clauses
are read.
# you need external update procedures to track changes in
keys.
# trusted-keys-file: ""

# Ignore chain of trust. Domain is treated as insecure.
# domain-insecure: "example.com"

# Override the date for validation with a specific fixed date.
# Do not set this unless you are debugging signature inception
# and expiration. "" or "0" turns the feature off. -1 ignores
date.
# val-override-date: ""

# The time to live for bogus data, rrsets and messages. This
avoids
# some of the revalidation, until the time interval expires.
in secs.
# val-bogus-ttl: 60

# The signature inception and expiration dates are allowed to
be off
# by 10% of the signature lifetime (expir-incep) from our
```

```
local clock.
    # This leeway is capped with a minimum and a maximum.  In
seconds.
    # val-sig-skew-min: 3600
    # val-sig-skew-max: 86400

    # Should additional section of secure message also be kept
clean of
    # unsecure data. Useful to shield the users of this validator
from
    # potential bogus data in the additional section. All unsigned
data
    # in the additional section is removed from secure messages.
    # val-clean-additional: yes

    # Turn permissive mode on to permit bogus messages. Thus,
messages
    # for which security checks failed will be returned to
clients,
    # instead of SERVFAIL. It still performs the security checks,
which
    # result in interesting log files and possibly the AD bit in
    # replies if the message is found secure. The default is off.
    # val-permissive-mode: no

    # Ignore the CD flag in incoming queries and refuse them bogus
data.
    # Enable it if the only clients of unbound are legacy servers
(w2008)
    # that set CD but cannot validate themselves.
    # ignore-cd-flag: no

    # Serve expired responses from cache, with TTL 0 in the
response,
    # and then attempt to fetch the data afresh.
    # serve-expired: no
    #
    # Limit serving of expired responses to configured seconds
after
    # expiration. 0 disables the limit.
    # serve-expired-ttl: 0
    #
    # Set the TTL of expired records to the serve-expired-ttl
value after a
    # failed attempt to retrieve the record from upstream. This
makes sure
    # that the expired records will be served as long as there are
queries
    # for it.
    # serve-expired-ttl-reset: no
```

```
# Have the validator log failed validations for your
diagnosis.
# 0: off. 1: A line per failed user query. 2: With reason and
bad IP.
# val-log-level: 0

# It is possible to configure NSEC3 maximum iteration counts
per
# keysize. Keep this table very short, as linear search is
done.
# A message with an NSEC3 with larger count is marked
insecure.
# List in ascending order the keysize and count values.
# val-nsec3-keysize-iterations: "1024 150 2048 500 4096 2500"

# instruct the auto-trust-anchor-file probing to add anchors
after ttl.
# add-holddown: 2592000 # 30 days

# instruct the auto-trust-anchor-file probing to del anchors
after ttl.
# del-holddown: 2592000 # 30 days

# auto-trust-anchor-file probing removes missing anchors after
ttl.
# If the value 0 is given, missing anchors are not removed.
# keep-missing: 31622400 # 366 days

# debug option that allows very small holddown times for key
rollover,
# otherwise the RFC mandates probe intervals must be at least
1 hour.
# permit-small-holddown: no

# the amount of memory to use for the key cache.
# plain value in bytes or you can append k, m or G. default is
"4Mb".
# key-cache-size: 4m

# the number of slabs to use for the key cache.
# the number of slabs must be a power of 2.
# more slabs reduce lock contention, but fragment memory
usage.
# key-cache-slabs: 4

# the amount of memory to use for the negative cache (used for
DLV).
# plain value in bytes or you can append k, m or G. default is
"1Mb".
# neg-cache-size: 1m
```



[illegible]

```
# Add example.com into ipset
# local-zone: "example.com" ipset

# If unbound is running service for the local host then it is
useful
# to perform lan-wide lookups to the upstream, and unblock the
# long list of local-zones above. If this unbound is a dns
server
# for a network of computers, disabled is better and stops
information
# leakage of local lan information.
# unblock-lan-zones: no

# The insecure-lan-zones option disables validation for
# these zones, as if they were all listed as domain-insecure.
# insecure-lan-zones: no

# a number of locally served zones can be configured.
#   local-zone: <zone> <type>
#   local-data: "<resource record string>"
# o deny serves local data (if any), else, drops queries.
# o refuse serves local data (if any), else, replies with
error.
# o static serves local data, else, nxdomain or nodata answer.
# o transparent gives local data, but resolves normally for
other names
# o redirect serves the zone data for any subdomain in the
zone.
# o nodefault can be used to normally resolve AS112 zones.
# o typettransparent resolves normally for other types and
other names
# o inform acts like transparent, but logs client IP address
# o inform_deny drops queries and logs client IP address
# o inform_redirect redirects queries and logs client IP
address
# o always_transparent, always_refuse, always_nxdomain,
resolve in
#   that way but ignore local data for that name
# o noview breaks out of that view towards global local-zones.
#
# defaults are localhost address, reverse for 127.0.0.1 and
::1
# and nxdomain for AS112 zones. If you configure one of these
zones
# the default content is omitted, or you can omit it with
'nodefault'.
#
# If you configure local-data without specifying local-zone,
by
# default a transparent local-zone is created for the data.
```

```
#
# You can add locally served data with
# local-zone: "local." static
# local-data: "mycomputer.local. IN A 192.0.2.51"
# local-data: 'mytext.local TXT "content of text record"'
#
# You can override certain queries with
# local-data: "adserver.example.com A 127.0.0.1"
#
# You can redirect a domain to a fixed address with
# (this makes example.com, www.example.com, etc, all go to
192.0.2.3)
# local-zone: "example.com" redirect
# local-data: "example.com A 192.0.2.3"
#
# Shorthand to make PTR records, "IPv4 name" or "IPv6 name".
# You can also add PTR records using local-data directly, but
then
# you need to do the reverse notation yourself.
# local-data-ptr: "192.0.2.3 www.example.com"

# tag a localzone with a list of tag names (in "" with spaces
between)
# local-zone-tag: "example.com" "tag2 tag3"

# add a netblock specific override to a localzone, with zone
type
# local-zone-override: "example.com" 192.0.2.0/24 refuse

# service clients over TLS (on the TCP sockets), with plain
DNS inside
# the TLS stream. Give the certificate to use and private
key.
# default is "" (disabled). requires restart to take effect.
# tls-service-key: "path/to/privatekeyfile.key"
# tls-service-pem: "path/to/publiccertfile.pem"
# tls-port: 853

# cipher setting for TLSv1.2
# tls-ciphers: "DHE-RSA-AES256-GCM-SHA384:DHE-RSA-AES128-GCM-
SHA256:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES128-GCM-
SHA256:DHE-RSA-AES256-SHA256:DHE-RSA-AES128-SHA256:ECDHE-RSA-
AES256-SHA384:ECDHE-RSA-AES128-SHA256"
# cipher setting for TLSv1.3
# tls-ciphersuites:
"TLS_AES_128_GCM_SHA256:TLS_AES_128_CCM_8_SHA256:TLS_AES_128_CCM_S
HA256:TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256"

# Add the secret file for TLS Session Ticket.
# Secret file must be 80 bytes of random data.
# First key use to encrypt and decrypt TLS session tickets.
```

```
# Other keys use to decrypt only.
# requires restart to take effect.
# tls-session-ticket-keys: "path/to/secret_file1"
# tls-session-ticket-keys: "path/to/secret_file2"

# request upstream over TLS (with plain DNS inside the TLS
stream).
# Default is no. Can be turned on and off with unbound-
control.
# tls-upstream: no

# Certificates used to authenticate connections made upstream.
# tls-cert-bundle: ""

# Add system certs to the cert bundle, from the Windows Cert
Store
# tls-win-cert: no

# Also serve tls on these port numbers (eg. 443, ...), by
listing
# tls-additional-port: portno for each of the port numbers.

# DNS64 prefix. Must be specified when DNS64 is use.
# Enable dns64 in module-config. Used to synthesize IPv6 from
IPv4.
# dns64-prefix: 64:ff9b::0/96

# DNS64 ignore AAAA records for these domains and use A
instead.
# dns64-ignore-aaaa: "example.com"

# ratelimit for uncached, new queries, this limits recursion
effort.
# ratelimiting is experimental, and may help against
randomqueryflood.
# if 0(default) it is disabled, otherwise state qps allowed
per zone.
# ratelimit: 0

# ratelimits are tracked in a cache, size in bytes of cache
(or k,m).
# ratelimit-size: 4m
# ratelimit cache slabs, reduces lock contention if equal to
cpucount.
# ratelimit-slabs: 4

# 0 blocks when ratelimited, otherwise let 1/xth traffic
through
# ratelimit-factor: 10
```

```
# override the ratelimit for a specific domain name.
# give this setting multiple times to have multiple overrides.
# ratelimit-for-domain: example.com 1000
# override the ratelimits for all domains below a domain name
# can give this multiple times, the name closest to the zone
is used.
# ratelimit-below-domain: com 1000

# global query ratelimit for all ip addresses.
# feature is experimental.
# if 0(default) it is disabled, otherwise states qps allowed
per ip address
# ip-ratelimit: 0

# ip ratelimits are tracked in a cache, size in bytes of cache
(or k,m).
# ip-ratelimit-size: 4m
# ip ratelimit cache slabs, reduces lock contention if equal
to cpucount.
# ip-ratelimit-slabs: 4

# 0 blocks when ip is ratelimited, otherwise let 1/xth traffic
through
# ip-ratelimit-factor: 10

# Limit the number of connections simultaneous from a netblock
# tcp-connection-limit: 192.0.2.0/24 12

# select from the fastest servers this many times out of 1000.
0 means
# the fast server select is disabled. prefetches are not sped
up.
# fast-server-permil: 0
# the number of servers that will be used in the fast server
selection.
# fast-server-num: 3

# Specific options for ipsecmod. unbound needs to be
configured with
# --enable-ipsecmod for these to take effect.
#
# Enable or disable ipsecmod (it still needs to be defined in
# module-config above). Can be used when ipsecmod needs to be
# enabled/disabled via remote-control(below).
# ipsecmod-enabled: yes
#
# Path to executable external hook. It must be defined when
ipsecmod is
# listed in module-config (above).
# ipsecmod-hook: "./my_executable"
#
```

```
# When enabled unbound will reply with SERVFAIL if the return
value of
# the ipsecmod-hook is not 0.
# ipsecmod-strict: no
#
# Maximum time to live (TTL) for cached A/AAAA records with
IPSECKEY.
# ipsecmod-max-ttl: 3600
#
# Reply with A/AAAA even if the relevant IPSECKEY is bogus.
Mainly used for
# testing.
# ipsecmod-ignore-bogus: no
#
# Domains for which ipsecmod will be triggered. If not defined
(default)
# all domains are treated as being whitelisted.
# ipsecmod-whitelist: "example.com"
# ipsecmod-whitelist: "nlnetlabs.nl"

# Python config section. To enable:
# o use --with-pythonmodule to configure before compiling.
# o list python in the module-config string (above) to enable.
# It can be at the start, it gets validated results, or just
before
# the iterator and process before DNSSEC validation.
# o and give a python-script to run.
python:
# Script file to load
# python-script: "/etc/unbound/ubmodule-tst.py"

# Remote control config section.
remote-control:
# Enable remote control with unbound-control(8) here.
# set up the keys and certificates with unbound-control-setup.
# control-enable: no

# what interfaces are listened to for remote control.
# give 0.0.0.0 and ::0 to listen to all interfaces.
# set to an absolute path to use a unix local name pipe,
certificates
# are not used for that, so key and cert files need not be
present.
# control-interface: 127.0.0.1
# control-interface: ::1

# port number for remote control operations.
# control-port: 8953
```

```
# for localhost, you can disable use of TLS by setting this to
"no"
# For local sockets this option is ignored, and TLS is not
used.
# control-use-cert: "yes"

# unbound server key file.
# server-key-file: "/etc/unbound/unbound_server.key"

# unbound server certificate file.
# server-cert-file: "/etc/unbound/unbound_server.pem"

# unbound-control key file.
# control-key-file: "/etc/unbound/unbound_control.key"

# unbound-control certificate file.
# control-cert-file: "/etc/unbound/unbound_control.pem"

# Stub zones.
# Create entries like below, to make all queries for 'example.com'
and
# 'example.org' go to the given list of nameservers. list zero or
more
# nameservers by hostname or by ipaddress. If you set stub-prime
to yes,
# the list is treated as priming hints (default is no).
# With stub-first yes, it attempts without the stub if it fails.
# Consider adding domain-insecure: name and local-zone: name
nodefault
# to the server: section if the stub is a locally served zone.
# stub-zone:
#   name: "example.com"
#   stub-addr: 192.0.2.68
#   stub-prime: no
#   stub-first: no
#   stub-tls-upstream: no
#   stub-no-cache: no
# stub-zone:
#   name: "example.org"
#   stub-host: ns.example.com.

# Forward zones
# Create entries like below, to make all queries for 'example.com'
and
# 'example.org' go to the given list of servers. These servers
have to handle
# recursion to other nameservers. List zero or more nameservers by
hostname
# or by ipaddress. Use an entry with name "." to forward all
queries.
# If you enable forward-first, it attempts without the forward if
```

```
it fails.
# forward-zone:
#   name: "example.com"
#   forward-addr: 192.0.2.68
#   forward-addr: 192.0.2.73@5355 # forward to port 5355.
#   forward-first: no
#   forward-tls-upstream: no
#   forward-no-cache: no
# forward-zone:
#   name: "example.org"
#   forward-host: fwd.example.com

# Authority zones
# The data for these zones is kept locally, from a file or
# downloaded.
# The data can be served to downstream clients, or used instead of
# the
# upstream (which saves a lookup to the upstream). The first
# example
# has a copy of the root for local usage. The second serves
# example.org
# authoritatively. zonefile: reads from file (and writes to it if
# you also
# download it), master: fetches with AXFR and IXFR, or url to
# zonefile.
# With allow-notify: you can give additional (apart from masters)
# sources of
# notifies.
# auth-zone:
#   name: "."
#   master: 199.9.14.201 # b.root-servers.net
#   master: 192.33.4.12 # c.root-servers.net
#   master: 199.7.91.13 # d.root-servers.net
#   master: 192.5.5.241 # f.root-servers.net
#   master: 192.112.36.4 # g.root-servers.net
#   master: 193.0.14.129 # k.root-servers.net
#   master: 192.0.47.132 # xfr.cjr.dns.icann.org
#   master: 192.0.32.132 # xfr.lax.dns.icann.org
#   master: 2001:500:200::b # b.root-servers.net
#   master: 2001:500:2::c # c.root-servers.net
#   master: 2001:500:2d::d # d.root-servers.net
#   master: 2001:500:2f::f # f.root-servers.net
#   master: 2001:500:12::d0d # g.root-servers.net
#   master: 2001:7fd::1 # k.root-servers.net
#   master: 2620:0:2830:202::132 # xfr.cjr.dns.icann.org
#   master: 2620:0:2d0:202::132 # xfr.lax.dns.icann.org
#   fallback-enabled: yes
#   for-downstream: no
#   for-upstream: yes
# auth-zone:
```



```
# name: "example.org"
# for-downstream: yes
# for-upstream: yes
# zonefile: "example.org.zone"

# Views
# Create named views. Name must be unique. Map views to requests
using
# the access-control-view option. Views can contain zero or more
local-zone
# and local-data options. Options from matching views will
override global
# options. Global options will be used if no matching view is
found.
# With view-first yes, it will try to answer using the global
local-zone and
# local-data elements if there is no view specific match.
# view:
# name: "viewname"
# local-zone: "example.com" redirect
# local-data: "example.com A 192.0.2.3"
# local-data-ptr: "192.0.2.3 www.example.com"
# view-first: no
# view:
# name: "anotherview"
# local-zone: "example.com" refuse

# DNSCrypt
# Caveats:
# 1. the keys/certs cannot be produced by unbound. You can use
dnscrypt-wrapper
# for this:
https://github.com/cofyc/dnscrypt-wrapper/blob/master/README.md#usage
# 2. dnscrypt channel attaches to an interface. you MUST set
interfaces to
# listen on `dnscrypt-port` with the following snippet:
# server:
# interface: 0.0.0.0@443
# interface: ::0@443
#
# Finally, `dnscrypt` config has its own section.
# dnscrypt:
# dnscrypt-enable: yes
# dnscrypt-port: 443
# dnscrypt-provider: 2.dnscrypt-cert.example.com.
# dnscrypt-secret-key: /path/unbound-conf/keys1/1.key
# dnscrypt-secret-key: /path/unbound-conf/keys2/1.key
# dnscrypt-provider-cert: /path/unbound-conf/keys1/1.cert
# dnscrypt-provider-cert: /path/unbound-conf/keys2/1.cert
```

```
# CacheDB
# Enable external backend DB as auxiliary cache.  Specify the
# backend name
# (default is "testframe", which has no use other than for
# debugging and
# testing) and backend-specific options.  The 'cachedb' module
# must be
# included in module-config, just before the iterator module.
# cachedb:
#     backend: "testframe"
#     # secret seed string to calculate hashed keys
#     secret-seed: "default"
#
#     # For "redis" backend:
#     # redis server's IP address or host name
#     redis-server-host: 127.0.0.1
#     # redis server's TCP port
#     redis-server-port: 6379
#     # timeout (in ms) for communication with the redis server
#     redis-timeout: 100

# IPSet
# Add specify domain into set via ipset.
# Note: To enable ipset needs run unbound as root user.
# ipset:
#     # set name for ip v4 addresses
#     name-v4: "list-v4"
#     # set name for ip v6 addresses
#     name-v6: "list-v6"
#
```

## Voir aussi

- (fr) <http://Article>

---

Basé sur « [Article](#) » par Auteur.

From:

<http://www.nfrappe.fr/doc-0/> - **Documentation du Dr Nicolas Frappé**

Permanent link:

<http://www.nfrappe.fr/doc-0/doku.php?id=logiciel:internet:unbound:config:dist>



Last update: **2022/08/13 22:14**