

# Zonemaster

A new DNS testing tool

*Patrik Wallström, .SE*

*afnic*

**.SE**

# Background

- DNSCheck does not give truly deterministic results
- ZoneCheck written in Ruby, old legacy code
- AFNIC and .SE needed better tools for DNS measurements
- Decision to create a joint “reference” tool

# The Collaboration

- Based on DNSCheck and ZoneCheck
  - Joint requirements and specifications
- Collaboration between AFNIC and .SE
- A new tool with a great name

# Requirements

- For version 1.0:
  - All test cases from DNSCheck and ZoneCheck
  - All functionality from DNSCheck and ZoneCheck
  - Modular code
  - Faster, and lower impact on network!

R08!?

Req	Requirement description	Level and Test Case
R01	UDP connectivity	CONNECTIVITY
R02	TCP connectivity	CONNECTIVITY
R03	Address in a private network	ADDRESS
R04	Address should not be part of a bogon prefix	ADDRESS
R05	Illegal symbols in domain name	SYNTAX
R06	Dash ('-') at start or beginning of domain name	SYNTAX
R07	Double dash in domain name	SYNTAX
R09	At least two nameservers for the domain	DELEGATION
R10	Identical addresses	DELEGATION
R11	Nameserver addresses on same subnet	CONNECTIVITY
R12	Nameserver addresses are all on the same subnet	CONNECTIVITY
R13	Delegation response fit in a 512 byte UDP packet	DELEGATION
R14	Delegation response with additional fit in a 512 byte UDP packet	DELEGATION
R15	NS record present	BASIC
R16	NS authoritative answer	DELEGATION
R17	NS name has a valid domain/hostname syntax	SYNTAX



# Report on Requirements to Test Case mapping

Req	Requirement description	Level	Test Case ID	Test Description
R01	UDP connectivity	Connectivity-TP	CONNECTIVITY01	UDP connectivity
R02	TCP connectivity	Connectivity-TP	CONNECTIVITY02	TCP connectivity
R03	address in a private network	Address-TP	ADDRESS01	Name server address must be globally routable
R04	address should not be part of a bogon prefix	Address-TP	ADDRESS04	IPv6 address not part of bogon prefix
R05	illegal symbols in domain name	Syntax-TP	SYNTAX01	No illegal characters in the domain name
R06	dash ('-') at start or beginning of domain name	Syntax-TP	SYNTAX02	No hyphen ('-') at the start or end of the domain name
R07	double dash in domain name	Syntax-TP	SYNTAX03	There must be no double hyphen ('--') in position 3 and 4 of the domain name

# CONNECTIVITY01: UDP connectivity

---

## Test case identifier

CONNECTIVITY01: UDP connectivity

## Objective

DNS queries are sent using UDP on port 53, as described in section 4.2.1 of [RFC 1035](#).

The objective for this test is that all the authoritative name servers for the domain are accessible over UDP on port 53

## Inputs

The domain to be tested.

## Ordered description of steps to be taken to execute the test case

1. Obtains the IP address of the name servers from [Method4](#) and [Method5](#)
2. A SOA query is sent over UDP to distinct IP addresses of each name server found in step 1.
3. If all queries in step 2 receive a DNS answer (bogus responses are not checked here) then the test case succeed.

## Outcome(s)

If there is any name server that fails to answer queries over port 53 using UDP, this test case fails

## Special procedural requirements

If either IPv4 or IPv6 transport is disabled, ignore the evaluation of the result of any test using this transport protocol.  
Log a message reporting on the ignored result.

## Intercase dependencies

None

# Implementation

- Written in Perl...
  - Engine
  - CLI
  - Backend
  - GUI (Javascript + Perl)



# Zonemaster Engine

- A Perl library (`Zonemaster::*`)
- That implements
  - All functionality for testing
  - All the test cases
  - Functionality for logging results
  - Its own resolver
    - That uses `Net::LDNS` (built on `Idns`) instead of `Net::DNS`

# Zonemaster CLI

- Takes its input from a user
- Executes a test
- Publishes a log
  - Log as text, raw text - or JSON

# Zonemaster Backend

- JSON-RPC-interface to the Engine
- Used by the GUI to execute tests
- Can also be used by other applications for batch testing

# Zonemaster GUI

- The UI to run tests
- ... and present the result
- Is available at [zonemaster.net](https://zonemaster.net)

## Domain name

iis.se



### Advanced options

## Test # 8828

Executed at 2015-04-27T10:25+0200

Link

<http://zonemaster.net/test/8828>

**Basic** [Advanced](#) [Export](#) [History](#)

- SYSTEM
- BASIC
- ADDRESS
- CONNECTIVITY
- CONSISTENCY
- DNSSEC
- DELEGATION
- NAMESERVER
- SYNTAX
- ZONE

# Releases

- The four components can be released independently
- “Zonemaster Distribution” releases

The screenshot displays the GitHub repository for 'Zonemaster Distribution'. On the left, a table lists the modules and their versions. On the right, a release history sidebar shows the progression from v1.0.0 to v1.0.3. A red arrow points from the 'zonemaster-engine' row in the table to the v1.0.3 release entry in the sidebar.

Module	Version	Commit Hash
<a href="#">zonemaster-engine</a>	1.0.0	a9d6c832ddc...
<a href="#">zonemaster-cli</a>	1.0.0	42f413136caa...
<a href="#">zonemaster-backend</a>	1.0.0	85eea4f8a0da...
<a href="#">zonemaster-gui</a>	1.0.0	d2f5acc91f8cb4e51b703da91d743d05122170aa...

Release History:

- v1.0.3 (20 days ago) - a7adeb9 - zip, tar.gz
- v1.0.2 (on Feb 24) - 853d23c - zip, tar.gz
- v1.0.1 (on Dec 30, 2014) - b62c965 - zip, tar.gz
- v1.0.0 (on Dec 11, 2014) - f615054 - zip, tar.gz

# Installation

```
sudo apt-get install build-essential libfile-slurp-perl libjson-perl liblist-moreutils-  
perl libio-socket-inet6-perl libmodule-find-perl libmoose-perl libfile-sharedir-perl  
libhash-merge-perl libreadonly-perl libmail-rfc822-address-perl libintl-xs-perl  
libssl-dev libdevel-checklib-perl libtest-fatal-perl libtie-simple-perl libio-capture-  
perl libgeography-countries-perl libidn1 1-dev
```

**sudo cpan -i Zonemaster**

```
sudo apt-get install libmoosex-getopt-perl libtext-reflow-perl libmodule-install-perl
```

**sudo cpan -i Zonemaster::CLI**

# How to run a test?

```
$> zonemaster-cli blipp.com
```

```
Seconds Level      Message
```

```
=====
```

```
8.25 NOTICE      Nameserver boa.blipp.com has an IP address  
(2001:1b40:5600:900:4321:1234:9b92:4bc0) with mismatched PTR result (snake.blipp.com.).
```

```
8.25 NOTICE      Nameserver boa.blipp.com has an IP address (95.154.217.14) with  
mismatched PTR result (snake.blipp.com.).
```

```
8.25 NOTICE      Nameserver vic20.blipp.com has an IP address (2001:16d8:ff00:2a9::2)  
with mismatched PTR result (cl-682.sto-01.se.sixxs.net.).
```

```
33.47 NOTICE      Nameserver boa.blipp.com/2001:1b40:5600:900:4321:1234:9b92:4bc0 allow  
zone transfer using AXFR.
```

```
33.57 NOTICE      Nameserver boa.blipp.com/95.154.217.14 allow zone transfer using AXFR.
```

```
34.40 NOTICE      Nameserver vic20.blipp.com/192.195.142.21 allow zone transfer using  
AXFR.
```

```
34.42 NOTICE      Nameserver vic20.blipp.com/2001:16d8:ff00:2a9::2 allow zone transfer  
using AXFR.
```



```

usage: zonemaster-cli [-?h] [long options...]
-h -? --usage --help Prints this usage information.
--version Print version information and exit.
--level The minimum severity level to display. Must be
one of CRITICAL, ERROR, WARNING, NOTICE, INFO or
DEBUG.

--locale The locale to use for messages translation.
--json Flag indicating if output should be in JSON or
not.
--raw Flag indicating if output should be translated
to human language or dumped raw.
--time Print timestamp on entries.
--show_level Print level on entries.
--show_module Print the name of the module on entries.
--ns A name/ip string giving a nameserver for
undelegated tests. Can be given multiple times.
--save Name of a file to save DNS data to after running
tests.
--restore Name of a file to restore DNS data from before
running test.
--ipv4 Flag to permit or deny queries being sent via
IPv4. --ipv4 permits IPv4 traffic, --no-ipv4
forbids it.
--ipv6 Flag to permit or deny queries being sent via
IPv6. --ipv6 permits IPv6 traffic, --no-ipv6
forbids it.
--list_tests Instead of running a test, list all available
tests.
--test Specify test to run. Should be either the name
of a module, or the name of a module and the
name of a method in that module separated by a
"/" character (Example: "Basic/basic1"). The
method specified must be one that takes a zone
object as its single argument. This switch can
be repeated.
--stop_level As soon as a message at this level or higher is
logged, execution will stop. Must be one of
CRITICAL, ERROR, WARNING, NOTICE, INFO or DEBUG.
--config Name of configuration file to load.
--policy Name of policy file to load.
--ds Strings with DS data on the form
"keytag,algorithm,type,digest"
--count Print a count of the number of messages at each
level
--progress Boolean flag for activity indicator. Defaults to
on if STDOUT is a tty, off if it is not.
--encoding Name of the character encoding used for command
line arguments
--nstimes At the end of a run, print a summary of the
times the zone's name servers took to answer.
--dump_config Print the effective configuration used in JSON
format, then exit.
--dump_policy Print the effective policy used in JSON format,
then exit.

```



# CLI Features

- Pre-delegation tests
- Record a dump of queries+answers
- Use answers from a recorded dump
- Stop execution at specific log level
- Use different policy evaluating the results
- Record name server response times
- Human readable output in Swedish and French ;)

# Log levels

- DEBUG 1, 2, 3
- INFO
- NOTICE
- WARNING
- ERROR
- CRITICAL



Configured  
by  
policy

# Policy

```
$> zonemaster-cli --dump_policy
{
  "ADDRESS" : {
    "NAMESERVERS_IP_WITH_REVERSE" : "INFO",
    "NAMESERVER_IP_PRIVATE_NETWORK" : "ERROR",
    "NAMESERVER_IP_PTR_MATCH" : "INFO",
    "NAMESERVER_IP_PTR_MISMATCH" : "NOTICE",
    "NAMESERVER_IP_WITHOUT_REVERSE" : "WARNING",
    "NO_IP_PRIVATE_NETWORK" : "INFO",
    "NO_RESPONSE_PTR_QUERY" : "WARNING"
  },
  . . .
}
```

# Log - Human readable

```
$> zonemaster-cli blipp.com
```

```
Seconds Level      Message
```

```
=====
```

```
8.25 NOTICE      Nameserver boa.blipp.com has an IP address  
(2001:1b40:5600:900:4321:1234:9b92:4bc0) with mismatched PTR result (snake.blipp.com.).
```

```
8.25 NOTICE      Nameserver boa.blipp.com has an IP address (95.154.217.14) with  
mismatched PTR result (snake.blipp.com.).
```

```
8.25 NOTICE      Nameserver vic20.blipp.com has an IP address (2001:16d8:ff00:2a9::2)  
with mismatched PTR result (cl-682.sto-01.se.sixxs.net.).
```

```
33.47 NOTICE     Nameserver boa.blipp.com/2001:1b40:5600:900:4321:1234:9b92:4bc0 allow  
zone transfer using AXFR.
```

```
33.57 NOTICE     Nameserver boa.blipp.com/95.154.217.14 allow zone transfer using AXFR.
```

```
34.40 NOTICE     Nameserver vic20.blipp.com/192.195.142.21 allow zone transfer using  
AXFR.
```

```
34.42 NOTICE     Nameserver vic20.blipp.com/2001:16d8:ff00:2a9::2 allow zone transfer  
using AXFR.
```

# Log - Raw

```
$> zonemaster-cli --raw blipp.com
```

```
8.13 NOTICE ADDRESS:NAMESERVER_IP_PTR_MISMATCH  
address=2001:1b40:5600:900:4321:1234:9b92:4bc0, names=snake.blipp.com., ns=boa.blipp.com
```

```
8.13 NOTICE ADDRESS:NAMESERVER_IP_PTR_MISMATCH address=95.154.217.14,  
names=snake.blipp.com., ns=boa.blipp.com
```

```
8.13 NOTICE ADDRESS:NAMESERVER_IP_PTR_MISMATCH address=2001:16d8:ff00:2a9::2,  
names=cl-682.sto-01.se.sixxs.net., ns=vic20.blipp.com
```

```
33.60 NOTICE NAMESERVER:AXFR_AVAILABLE  
address=2001:1b40:5600:900:4321:1234:9b92:4bc0, ns=boa.blipp.com
```

```
33.74 NOTICE NAMESERVER:AXFR_AVAILABLE address=95.154.217.14, ns=boa.blipp.com
```

```
34.57 NOTICE NAMESERVER:AXFR_AVAILABLE address=192.195.142.21, ns=vic20.blipp.com
```

```
34.59 NOTICE NAMESERVER:AXFR_AVAILABLE address=2001:16d8:ff00:2a9::2,  
ns=vic20.blipp.com
```

# Logg - JSON

```
$> zonemaster-cli --json blipp.com
```

```
[{"args": {"address": "2001:1b40:5600:900:4321:1234:9b92:4bc0", "names": "snake.blipp.com.", "ns": "boa.blipp.com"}, "level": "NOTICE", "module": "ADDRESS", "tag": "NAMESERVER_IP_PTR_MISMATCH", "timestamp": 8.25753903388977}, {"args": {"address": "95.154.217.14", "names": "snake.blipp.com.", "ns": "boa.blipp.com"}, "level": "NOTICE", "module": "ADDRESS", "tag": "NAMESERVER_IP_PTR_MISMATCH", "timestamp": 8.25794792175293}, {"args": {"address": "2001:16d8:ff00:2a9::2", "names": "cl-682.sto-01.se.sixxs.net.", "ns": "vic20.blipp.com"}, "level": "NOTICE", "module": "ADDRESS", "tag": "NAMESERVER_IP_PTR_MISMATCH", "timestamp": 8.26008701324463}, {"args": {"address": "2001:1b40:5600:900:4321:1234:9b92:4bc0", "ns": "boa.blipp.com"}, "level": "NOTICE", "module": "NAMESERVER", "tag": "AXFR_AVAILABLE", "timestamp": 33.5192708969116}, {"args": {"address": "95.154.217.14", "ns": "boa.blipp.com"}, "level": "NOTICE", "module": "NAMESERVER", "tag": "AXFR_AVAILABLE", "timestamp": 33.6047399044037}, {"args": {"address": "192.195.142.21", "ns": "vic20.blipp.com"}, "level": "NOTICE", "module": "NAMESERVER", "tag": "AXFR_AVAILABLE", "timestamp": 34.4463429450989}, {"args": {"address": "2001:16d8:ff00:2a9::2", "ns": "vic20.blipp.com"}, "level": "NOTICE", "module": "NAMESERVER", "tag": "AXFR_AVAILABLE", "timestamp": 34.4668040275574}]
```

# Anatomy of a log entry

```
[  
  {  
    "tag" : "TEST_CASE_TAG",  
    "timestamp" : 6.35820984840393,  
    "module" : "TEST-MODULE",  
    "level" : "MESSAGELEVEL",  
    "args" : {  
      "ns" : "nameserver.tld",  
      "address" : "127.0.0.1"  
    }  
  },  
  {...}  
]
```



# Build your own applications

```
use Zonemaster;  
  
# execute a test  
my @log = Zonemaster->test_zone( 'blipp.com' );  
my $json = Zonemaster->logger->json( 'NOTICE' );  
  
# get name servers for a zone  
my $zone = Zonemaster->zone( 'another-zone.com' );  
my $nameservers = join ( ' ', @{ $zone->ns_names } );
```

# “zonemaster-info”

expo\$~>zonemaster-info ripe.net

Zone: ripe.net  
Parent zone: net

Glue nameserver list:

ns3.nic.fr/192.134.0.49  
ns3.nic.fr/2001:660:3006:1::1:1  
pri.authdns.ripe.net/193.0.9.5  
pri.authdns.ripe.net/2001:67c:e0::5  
sec1.apnic.net/2001:dc0:2001:a:4608::59  
sec1.apnic.net/202.12.29.59  
sec3.apnic.net/2001:dc0:1:0:4777::140  
sec3.apnic.net/202.12.28.140  
sns-pb.isc.org/192.5.4.1  
sns-pb.isc.org/2001:500:2e::1  
tinnie.arin.net/199.212.0.53  
tinnie.arin.net/2001:500:13::c7d4:35

Authoritative nameserver list:

ns3.nic.fr/192.134.0.49  
ns3.nic.fr/2001:660:3006:1::1:1  
pri.authdns.ripe.net/193.0.9.5  
pri.authdns.ripe.net/2001:67c:e0::5  
sec1.apnic.net/2001:dc0:2001:a:4608::59  
sec1.apnic.net/202.12.29.59  
sec3.apnic.net/2001:dc0:1:0:4777::140  
sec3.apnic.net/202.12.28.140  
sns-pb.isc.org/192.5.4.1  
sns-pb.isc.org/2001:500:2e::1  
tinnie.arin.net/199.212.0.53  
tinnie.arin.net/2001:500:13::c7d4:35

Parent has DS record(s) for ripe.net.

Key tag: 65306 Digest algorithm: SHA-256  
Key tag: 65306 Digest algorithm: SHA-1

DNSKEY records:

Tag: 65306 Algorithm: RSA/SHA1  
Tag: 964 Algorithm: RSA/SHA1  
Tag: 14716 Algorithm: RSA/SHA1  
Tag: 8937 Algorithm: RSA/SHA1

SOA record:

MNAME: pri.authdns.ripe.net.  
RNAME: dns.ripe.net.  
Serial: 1431002041  
Refresh: 3600  
Retry: 600  
Expire: 864000  
Minimum: 300

SOA record signature(s):

Keytag: 14716  
Algorithm: RSA/SHA1  
Labels: 2  
Signer: ripe.net.  
Inception: 2015-05-07 13:38:32  
Expiration: 2015-06-06 14:38:32

# “zonemaster-collector”

- My own tool for mass collection
- Multi-threaded use of Zonemaster
- Storage in MongoDB
- Easy to query a lot of test data
- <https://github.com/pawal/zonemaster-collector>
  - Examples in the “howtomongo” document

# Mass collection

```
$ collect.pl -f 10000.txt --mongo --db results --collection foo --threads 35
```

# Query example 1

```
# results in a list of open resolvers
```

```
db.domains.aggregate(  
  { $match: { "result.tag": "IS_A_RECURSOR" } },  
  { $unwind: "$result" },  
  { $match: { "result.tag": "IS_A_RECURSOR" } },  
  { $project: { "name":1, "result.args": 1, "_id": 0 } }  
);  
  
{  
  "result" : {  
    "args" : {  
      "ns" : "ns2.example.com",  
      "address" : "12.34.56.78",  
      "dname" : "xx--domain-cannot-exist.xx--illegal-syntax-tld"  
    }  
  },  
  "name" : "example.com"  
}, ...
```

# Query example 2

```
db.domains.aggregate(  
  { $match: { "result.level": "ERROR" } },  
  { $unwind: "$result" },  
  { $match: { "result.level": "ERROR" } },  
  { $match: { "result.args.ns": { $exists: true } } },  
  { $project: { "name":1, "result": 1, "_id": 0 } },  
  { $group: { _id: "$result.args.ns", nscount: { $sum: 1 } } },  
  { $sort: { nscount: -1 } },  
  { $limit: 25 }  
);
```

```
{  
  "result" : [  
    {  
      "_id" : "ns1.example.com",  
      "nscount" : 233  
    },  
    {  
      "_id" : "ns2.example.com",  
      "nscount" : 232  
    },  
    ...  
  ]  
}
```

# Future improvements

- Fix all the log imperfection
- Implement new test requirements
- “IANA test profile” - different test profiles/policies
- HTML export of results

# What do you need?

- Batch testing from CLI?
- Anycast testing?
- Mobile apps?
- Any specific test case? (They are easy to write and just “plugin” to the existing framework.)



# TRTF

- The test specifications will be further refined by a new CENTR task force, the TRTF
- The goal is to have a best practices document
- Published as an informational RFC

# Test it yourself!

- <https://github.com/dotse/zonemaster>
- <http://lists.iis.se/cgi-bin/mailman/admin/zonemaster-users>
- <http://lists.iis.se/cgi-bin/mailman/admin/zonemaster-devel>