Realistic, Extensible DNS and mDNS Models for INET/OMNeT++

2nd OMNeT++ Community Summit, 2015

Andreas Rain, Daniel Kaiser, Marcel Waldvogel

University of Konstanz, Konstanz, Germany
<first>.<last>@uni-konstanz.de

September 4th, 2015

What is this work about?

DNS

- Design networks using DNS
- Design new extensions to DNS
- Evaluate performance and validate behavior

Privacy Extension

- Find new ways to enhance the privacy of users
- Validate your design

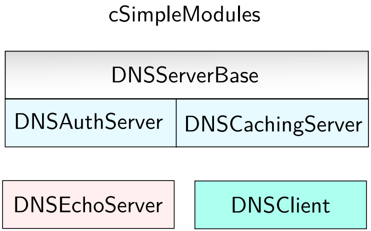
mDNS/DNS-SD

- Use mDNS for discovery
- Evaluate mDNS in combination with a new multicast transport protocol as a use case

Stateless DNS

- Discovery without infrastructure (more or less)
- Test Stateless DNS and check whether it fits your needs

DNS Simulation Model



cSimpleModules

DNSServerBase

DNSAuthServer

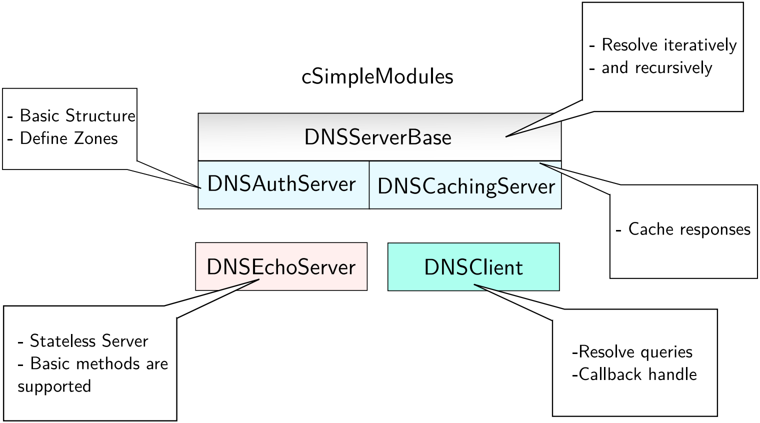
DNSCachingServer

DNSEchoServer

DNSClient

Figure: Overview of the simple modules belonging to the DNS model.

DNS Simulation Model



- Resolve iteratively
- and recursively

- Basic Structure
- Define Zones

- Cache responses

- Stateless Server
- Basic methods are supported

-Resolve queries
-Callback handle

Figure: Overview of the simple modules belonging to the DNS model.

Design DNS zones using the BIND syntax

Example Configuration

```
$TTL 86400 ; 24 hours, $TTL used for all RRs
ORIGIN uni-konstanz.de.
@ IN SOA pan.rz.uni-konstanz.de.   hostmaster.uni-konstanz.de. (
          2003080800 ; sn = serial number
          172800     ; ref = refresh = 2d
          900        ; ret = update retry = 15m
          1209600    ; ex = expiry = 2w
          3600       ; nx = nxdomain ttl = 1h
         )
  IN  NS pan.rz.uni-konstanz.de.    ; in the domain
  IN  NS uranos.rz.uni-konstanz.de. ; slave
  IN  MX imap.uni-konstanz.de.      ; external mail
  IN  A  134.34.240.80              ; ip of origin
; server host definitions
pan.rz       IN A      134.34.3.3    ; this server
uranos.rz    IN A      134.34.3.2    ; the slave server
imap         IN A      134.34.240.42 ; mail server imap
www          IN CNAME  proxy-neu.rz  ; test on
proxy-neu.rz IN A      134.34.240.80 ;
```

Figure: Example zone configuration based on BIND syntax.

Capabilities, Limitations, and Challenges

Capabilities

- Model DNS networks
- Hierarchical structures
- Recursive and iterative resolving
- `A, AAAA, NS, PTR, SRV, CNAME, TXT`
- Name compression

Limitations

- Manual modeling
- Bailiwick rules
- Not all record types
- Dynamic zone updates
- DNSSec

Challenges

- Dynamic generation
- Extensible design
- Mapping of rules
- RFC ↔ Implementation-specific
- Integration

mDNS Simulation Model

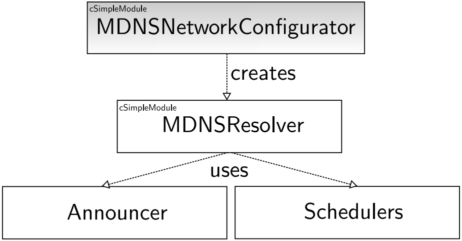
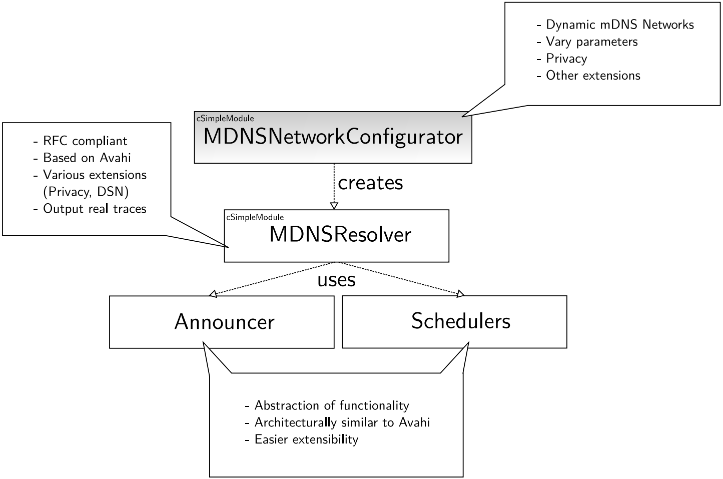


Figure: Structure of the mDNS simulation model and various components.

mDNS Simulation Model



- Dynamic mDNS Networks
- Vary parameters
- Privacy
- Other extensions

- RFC compliant
- Based on Avahi
- Various extensions (Privacy, DSN)
- Output real traces

- Abstraction of functionality
- Architecturally similar to Avahi
- Easier extensibility

Figure: Structure of the mDNS simulation model and various components.

Dynamic mDNS resolver networks

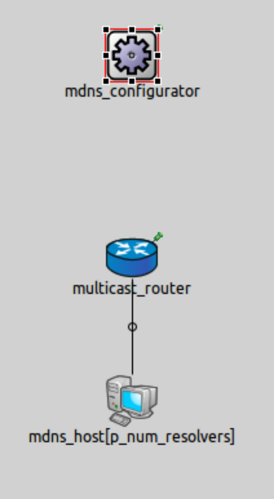


Figure: Dynamic mDNS network in its basic form.

Parameters:

- Number of Resolvers
- Number of Private Resolvers
- Maximum amount of **friends**
- Minimum amount of **friends**
- Maximum amount of services
- Minimum amount of services
- Ratio of public to private services

Capabilities, Limitations, and Challenges

Capabilities

- mDNS and DNS-SD
- Dynamic mDNS network generation
- Our privacy extension for mDNS
- Name compression

Limitations

- Shared resource records not handled differently
- Dynamic services
- Internal messages are not used to query or announce
- Not all resource record types are supported

Challenges

- Scheduling
- Reference implementations
- Dynamic generation
- Extensibility
- Integration

Extensions

Privacy

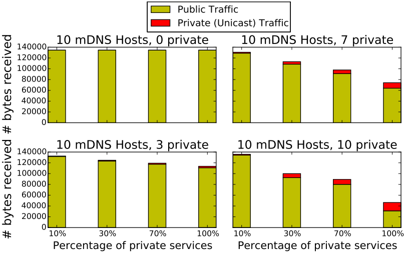


Figure: Evaluation of traffic reduction by the privacy extension.

Stateless DNS

- Combine with other protocols
- Validate behavior
- Add new functionality

Implement your own extension!

Usage

Example DNSCache

1. Extend the DNSCache interface.
2. Implement the methods and thus your caching strategy.
3. Simply change the DNSCache implementation used in the server.

Example DNSServer

1. Extend the **DNSServerBase** class (if needed).
2. Implement **handleQuery**
3. Return **DNSPacket** to send it
4. or nothing when recursion is initiated

Conclusions & Future Work

Possible future work:

- Dynamic generation of DNS networks
- Implementation of DNSSec
- DNS caching analysis
- Evaluation of other extensions
- Better integration with INET

What we are working on:

- Evaluation of the impact of mDNS on WLANs.
- Simulations performed on the **bwUniCluster** . . .
- . . . with up to 800 Simulations in parallel.

References I

- R. Arends, R. Austein, M. Larson, D. Massey, and S. Rose, “DNS security introduction and requirements,” March 2005, RFC 4033.
- “Avahi,” http://avahi.org, Internet Resource, last visited on May 24th, 2015.
- S. Cheshire and M. Krochmal, “DNS-based service discovery,” February 2013, RFC 6763.
- ——, “Multicast DNS,” February 2013, RFC 6762.
- D. Kaiser, M. Fratz, M. Waldvogel, and V. Dietrich, “Stateless DNS,” University of Konstanz, Tech. Rep. KN-2014-DiSy-004, Dec 2014.
- D. Kaiser, A. Rain, M. Waldvogel, and H. Strittmatter, “A multicast-avoiding privacy extension for the Avahi zeroconf daemon,” *Netsys 2015*, March 2015.
- D. Kaiser and M. Waldvogel, “Adding privacy to multicast DNS service discovery,” in *Trust, Security and Privacy in Computing and Communications (TrustCom), 2014 IEEE 13th International Conference on*. IEEE, 2014, pp. 809–816.
- ——, “Efficient privacy preserving multicast dns service discovery,” in *High Performance Computing and Communications, 2014 IEEE 6th Intl Symp on Cyberspace Safety and Security, 2014 IEEE 11th Intl Conf on Embedded Software and Syst (HPCC, CSS, ICESS), 2014 IEEE Intl Conf on*. IEEE, 2014, pp. 1229–1236.
- P. Mockapetris, “Domain names - implementation and specification,” November 1987, RFC 1035.
- S. Son and V. Shmatikov, “The hitchhiker’s guide to DNS cache poisoning,” in *Security and Privacy in Communication Networks*. Springer, 2010, pp. 466–483.