NDNS: DNS in NDN
Overview

- NDN needs DNS-like system
  - Map names to names (e.g., to scale routing [1])
    - Some names can be reached directly (in FIB) and some need to be mapped to reachable
  - Always-on storage to save and lookup keys
    - Data production & consumption in NDN is asynchronous
- Data-centric design at the networking level require special considerations
  - Namespace design to allow use of in-network caches and realize per-packet data security

NDNS Overview

- **Domain** = NDN name
- **Resource Record**
  - Information associated with domain
- **Zone**
  - Dataset containing records associated with NDNS domain
- **Authoritative Server**
  - Data storage for an NDNS zone
- **Caching Resolver**
  - Helper to discover domain record
- **Stub Resolver**
  - Client module for record discovery

This report is organized as follows: Section 2 presents the NDNS design on naming, query, update and trust model. Then Section 3 gives the implementation including database schema and packet format. The following Section 4 describes the management of NDNS. And Section 5 presents NDNS deployment in NDN testbed and its use cases. And Section 6 summarizes the report at last.

### 2. Design

This section presents the NDNS design. At the beginning of this section, we gives an overview of NDNS. Then NDNS naming convention, query, update and trust model is explained, respectively.

#### 2.1 Overview

Figure 1 is the overview of NDNS. There are different kinds of servers in the system: name server, caching resolver and stub resolver. The servers are connected via NDN network. For current, we assume that zone name announced by name servers are routable globally.

```
/edu
/net...
/net/ndnsim...
/net/ndnsim/docs
/net/ndnsim/www...
```

Figure 1: NDNS Overview

Every zone runs its own name servers which stores RRs. NDNs Queries arrives at name server through NDN network, where caching is build-in. In this scenario, caching resolver which is designed to cache RRs, plays a less important role than it does in traditional DNS.

#### 2.2 Naming Convention

The most important messages in NDNS are query and response. Query represents a question to a NDNS server, while response is answer to the question. Since the semantic of query is overlap with NDN Interest,
NDNS Design Space

- Replicate DNS protocol exactly
  - DNS is data-centric, but at app-level
  - Tradeoffs

- Designing natively data-centric NDNS
  - Data-centricity at the network-level
  - Tradeoffs
Replicating DNS (DNS)

- (!) Interest is a question not to the server, but to the network
- (?) In principle, can do point-to-point, but what could be the “name” of one of an NDNS zone?
- (!) Make NDNS servers be transparent “zone owners”

```
/NDNS/...
/net/NDNS/...
/net/ndnsim/NDNS/
/google/NDNS-R/
```
Replicating DNS (NDNS)

- (?) Interest is a question not to the server, but to the network
- (?) In principle, can do point-to-point, but what could be the “name” of one of an NDNS zone?
- (!) Make NDNS servers be transparent “zone owners” / service providers

```
/NDNS/...
/net/NDNS/...
/net/ndnsim/NDNS/
/google/NDNS-R/
```
Transforming DNS Query to NDN Interest (Option 1)

Root zone
/NDNS

Stub Resolver

/net/NDNS

/net/NDNS

? www.ndnsim.net TXT

/net/NDNS

? www.ndnsim.net TXT

/net/NDNS

? www.ndnsim.net TXT

? www.ndnsim.net TXT

/net/NDNS

/google/NDNS-R/net/ndnsim/www/TXT

/NDNS/net/ndnsim/www/TXT

/net/NDNS/net/ndnsim/www/TXT
Transforming DNS Query to NDN Interest (Option 2)
Selected for recursive NDNS query

- NDNS server can fully utilize its internal cache to return the requested record or next-level referral
- NDN sends query to the closest/available NDNS zone replica
- NDNS servers create response on the fly, e.g., encapsulating original record(s)
- Strict reliance on caching resolvers to scale system

Selected for iterative NDNS query

- NDNS sends query to the closest/available NDNS zone replica
- NDNS servers serve static responses
- Full use of NDN caches / scales independent whether caching resolvers exist or not
- NDNS servers cannot fully utilize caches, as the real question is not known
NDNS Lookup Overview

Figure 2: NDNS Query

Figure 2 shows the flow of a NDNS query. A stub resolver sends a recursive query to caching resolver, and caching resolver accepts the recursive query and translates it to a sequence of iterative queries. Name server answers iterative query according to resource record in its database. Once caching resolver gets the final iterative query response, it constructs the recursive query response which embeds the final iterative response inside.

DNS replies on caching resolver to cache data, otherwise, DNS name servers could not survive under the heavy traffic from the whole Internet. However, NDN network could aggregate identical queries and cache NDNS response, caching resolver in NDNS is not of the same significance as in traditional DNS. But still caching resolver helps to offload computing and save energy, especially for devices like sensor and personal digital assistant.

Two types of records are defined by NDNS per se, i.e., NS (Name Server), ID-CERT (Identity Certificate). ID-CERT RR is used to store certificates as mentioned above. NS RR is used to define the zone boundary. NS RR has three kinds of possible value, i.e., NDNS-RESP, NDNS-NACK, NDNS-AUTH. NDNS-RESP is the positive response to a query, while NDNS-NACK is the negative response. NDNS-AUTH is a kind of response to query name server indicating that zone with the name in current NS query does not exist, but there is a zone with longer name than that contained in the NS query exists.

A zone should stores a NS RR which points out the boundary of its sub-namespace delegation. For example, root zone should contain a NDNS-RESP NS RR pointing out that /ndn is a zone if it is, or answer a NDNS-NACK NS RR to query /NDNS/ndn/NS if /ndn is not further delegated. However, if zone /ndn does not exist but /ndn/ndnsim does, root zone should store NDNS-AUTH NS RR for label /ndn, and NDNS-RESP NS RR for label /ndn/ndnsim.

For current implementation, NS RRs contain nothing in its content field, but it is able to contain the routable identifier of the name servers if necessary.

2.4 Update

When resource record changes, the corresponding entry in the NDNS must be updated. Update is started by the authorized identity, who sends an Interest, embedding update message, towards NDNS authoritative server, who verifies the message and handles the update. Figure 3 shows the process that client follows to update the resource record.
Questions