DNS Dynamic Updates

Susan Thomson, Yakov Rekhter, Jim Bound, Peter Ford

IETF DNSIND Working Group

July 1994

Definition and Goal

- Ability to update a zone without manual configuration
- However, do not attempt to eliminate manual configuration
- Goal is to make the minimal changes necessary, and still provide a useful service.

Motivation

- To complement dynamic address assignment
- To help enable host address autoconfiguration
  - when booting "out of the box"
  - when host booting at different network location
    - because workstation moved to different office
    - because computer is "portable", e.g. laptop
- To ease address renumbering
  - when changing providers
- Mechanism NOT limited only to address registration

Requirements

- Enable records to be added, deleted, modified
- Also enable records to "time out"
- Node creation and deletion still manual
- This implies node names must be preregistered
- Believe this is not an impractical restriction
  - only system administrator has node creation authority
  - if security in use, key must be preregistered anyway
- There may be a requirement to ease system administrator's task, but out of our scope

Other Requirements

- Updates must be able to be made secure
  - Authentication, integrity, protection vs replay
  - Access control on node, record type basis
- Updates can be made to a zone from anywhere
- Must be compatible with auto address assignment
- Must be compatible with existing servers and resolvers
- Changes to existing DNS must be minimal
- Changes independent of version of IP in use

Architecture

- Principles:
  - Entity that “owns” node updates it
  - Lifetime of record less than that of data in record
- Operation:
  - Assume domain names in zones pre-registered
  - Host acquires address
  - Host acquires or knows domain name
  - Typically, host updates name-to-address mapping
  - Typically, address server updates inverse mapping
  - Updates occur at authoritative name servers only

Mechanisms

- New attribute associated with each record
  - expiration time
- New operation to update records associated with existing node
  - REPLACE operation
- Some incremental zone transfer and notify mechanism assumed
  - Ongoing work in this working group
- Authentication, integrity, protection vs replay
  - Ongoing work in DNSSEC working group

Expiration Time

- Expiration time is associated with each record
- Determines when record becomes invalid (out-of-date)
- Record must not be cached beyond expiry
- So, TTL set to MIN(time to expiry, TTL)
- Implementation uses expiration time in SIG record
- Implies security mechanism implemented, although security not necessarily in use

REPLACE operation

- REPLACE replaces the complete set of records of the specified domain name, class and type by the specified set.
- Domain name and class must exist.
- Update is permanent until records expire.
- Only primary servers can perform REPLACE operation
- Update distributed to other authoritative servers asynchronously using (incremental) zone transfer

Name Server REPLACE Behavior

- Name server must support QUERY and REPLACE
- On receiving a REPLACE:
  - Authoritative servers return error if name does not exist or is an alias.
  - Primary name server performs update, returns authoritative answer
  - Secondary returns an authoritative refusal
  - Non-authoritative servers provide delegation info

Extensions to Name Server QUERY Behavior

- On a QUERY:
  - Only valid (non-expired) records must be returned
  - TTL must not exceed time to expiry

Resolver REPLACE Behavior

- Resolver must update at least one primary
- To determine primary, full-service resolver may
  - query for SOA record, use MNAME field as hint
  - update all authoritative name servers
- If update successful, resolver should cache update
- Resolver should cache knowledge about primary servers
- Stub resolver may be configured with address of primary name server that it needs to update

Extensions to Resolver QUERY Behavior

- Invalid records should not be used or cached
- TTL should not exceed time-to-expiry

Implementation Issues

- Message format remains very similar, but
  - New opcode for REPLACE; new error code
  - REPLACE request fills in question and answer section
- Expiration time field as defined for SIG record
- Format of master file can be retained
  - Updating master file can be done by logging changes, and rewriting master file periodically.
- Access control mechanism
  - May use special type of expired record to indicate that updates allowed to corresponding RR type

Issues

- Availability vs Reliability
  - Use single primary (traditional model)
  - Use multiple primaries
  - Other