ECE 650
Systems Programming & Engineering

Spring 2018

Dynamic Host Configuration Protocol (DHCP) and Domain Name System (DNS)

Tyler Bletsch

Duke University

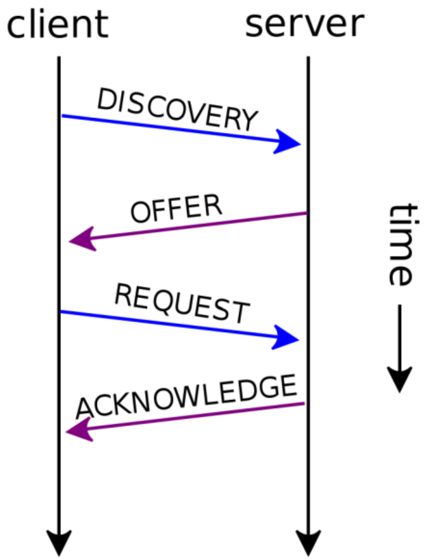
Slides are adapted from Brian Rogers (Duke)

Dynamic Host Configuration Protocol (DHCP)

(It’s just one slide)

Dynamic Host Configuration Protocol (DHCP)

- DHCP: Allow hosts to enter a network and ask "what IP should I use for myself?"
- How it works:
  1. Client sends an IP broadcast "DISCOVERY" request (destination 255.255.255.255 UDP port 67)
  2. DHCP server on network sends an "OFFER" with IP address and other config (gateway router, DNS servers, maybe other stuff)
     - *Note: multiple offers might be provided by multiple DHCP servers (but usually it's just one)*
  3. Client sends a broadcast REQUEST for one of the offers
  4. DHCP server sends ACKNOWLEDGE back
  5. Client now has an IP address and basic config info



- DHCP can also be used to start network-boot (PXE), commonly used for diskless clusters, OS auto-install, etc.

Domain Name System (DNS)

(Many slides)

Purpose of DNS

- Map an easy-to-remember name to an IP address
- IP Address
  - IPv4: four numbers separated by decimals, e.g. 120.74.53.1
    - Each number called an 'octet' with a value of 0-255
    - Some are reserved as special, e.g. 127.0.0.1 = localhost
  - IPv6: eight hex numbers separated by colons
- Implications
  - Without DNS, to send IP packet, must remember IP addresses manually! ...and they could change!
  - With DNS, we can use the name directly:
    - www.google.com or www.cnn.com
- DNS also provides inverse look-up that maps IP address to name

Design Goals of DNS

- Primary goal is a consistent namespace used to refer to resources
  - Consistent: same names should refer to same resources
  - Resources: IP addresses, mail servers
- Enable distributed management
  - Size of the name database will be large
  - Updates (changes, additions, removals) will be frequent
- Design goals determine its structure
  - Hierarchical name space
  - Distributed directory service

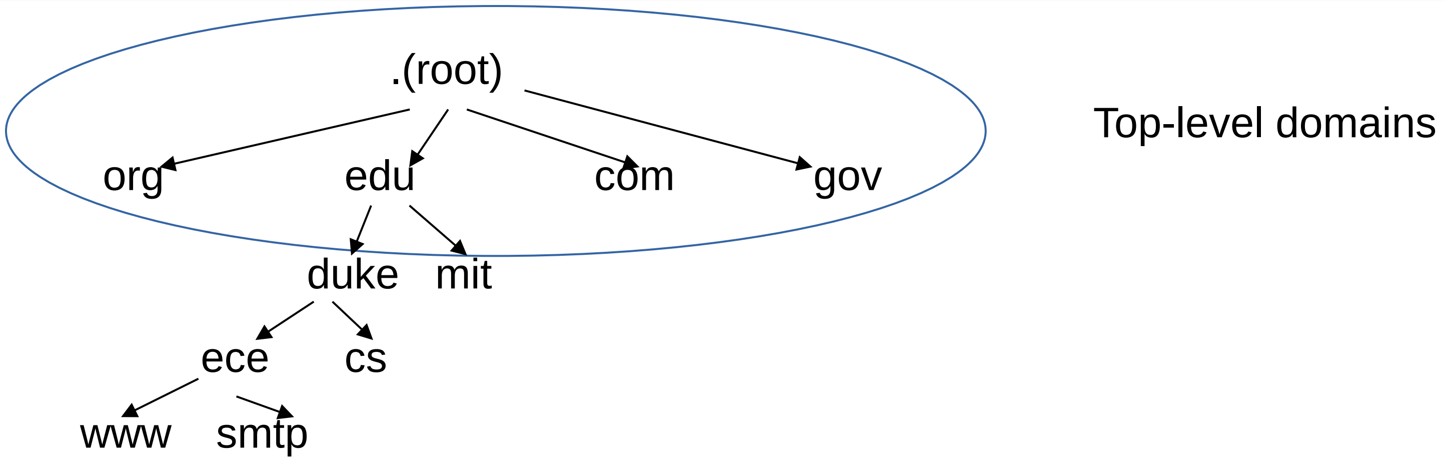
Before there was DNS...

- There was the HOSTS.TXT file (or what about /etc/hosts now)
- Maintained at SRI Network Information Center (NIC)
- Before DNS (1985), the name-to-IP address was done by downloading this single file from a central server with FTP
  - No hierarchical structure to the file
  - Still works on most OSes; can be used to define local names

DNS Architecture Components

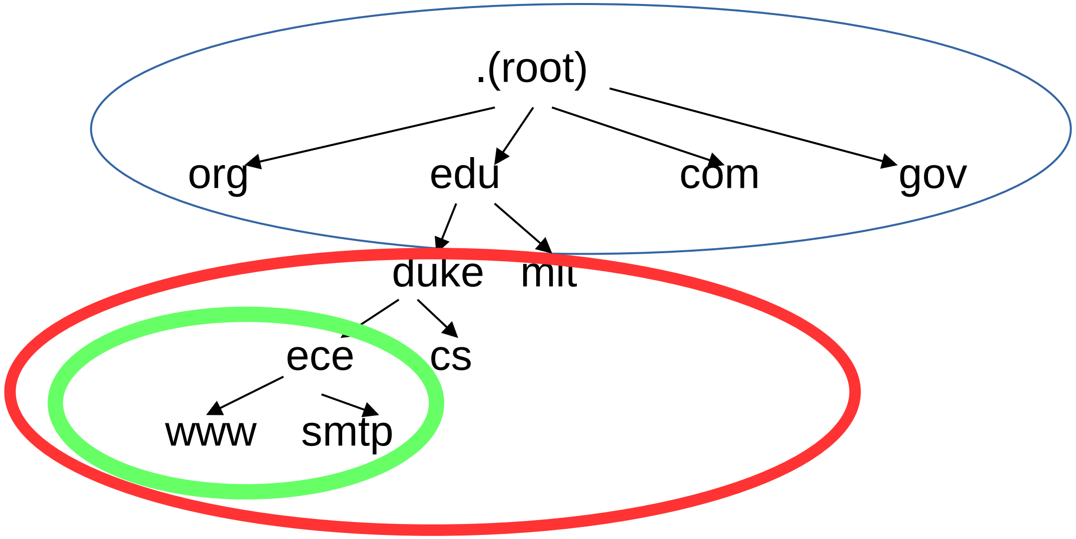
- Domain namespace and resource records
- Name servers
- Name resolution

Domain Namespace



- Domain namespace is a hierarchical and logical tree structure
- Label from a node to root in the DNS tree represents a DNS name
- Each subtree below a node is a DNS domain
  - DNS domain can contain hosts or other domains (subdomains)
- Examples of DNS domains: .edu, duke.edu, ece.duke.edu

Domain Namespace



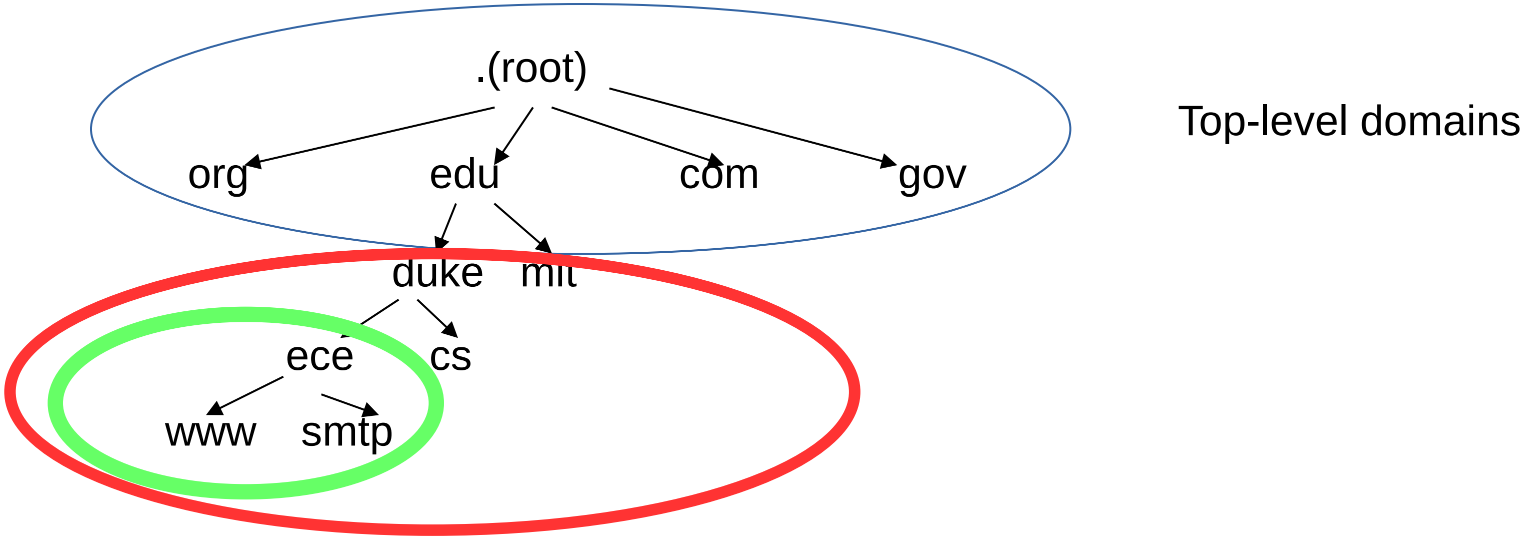
Top-level domains

- Red is managed by Duke
- Green is managed by ECE
- Below top-level domain, administration of name space is delegated to organizations
  - Each organization can further delegate

Domain Names

- Names of hosts can be assigned independent of host locations on a link layer network or IP network
  - This means the host name can remain the same
  - Even if IP address of the host changes (e.g. DHCP)

Fully Qualified Domain Names



- Every node in the DNS domain tree can be identified by a FQDN
  - Fully Qualified Domain Name
- FDQN (from right to left) consists of labels (“ece”, “duke”, “edu”) separated by a period from the root to the node
- Each label can be up to 63 characters; full DNS name <= 255 chars
- FDQN contains characters, digits, dashes; not case-sensitive

Top-Level Domains

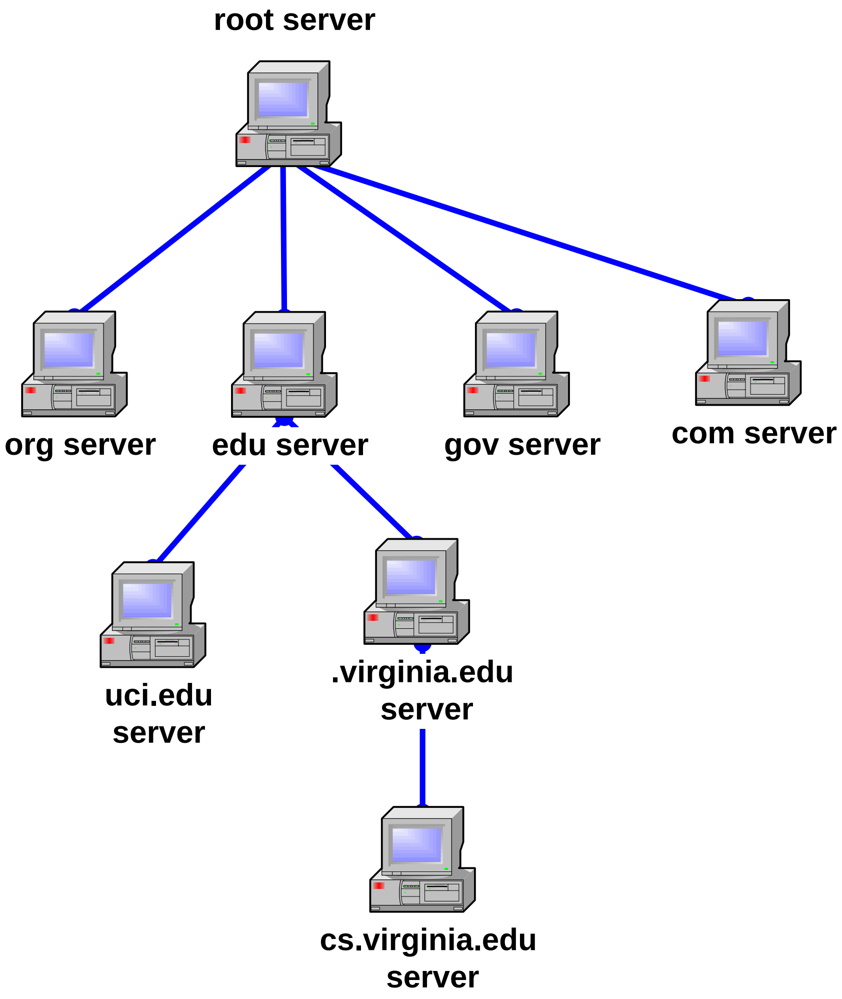
- Three types of top-level domains:
  - Generic Top Level Domains (gTLD)
    - 3 char code indicates the function of the organization
    - Use primarily within the US (e.g. gov, mil, edu, org, com, net)
  - Country Code Top Level Domain (ccTLD)
    - 2 char country or region code (e.g. us, jp, uk)
  - Reverse Domain
    - Special domain used for IP address-to-name mapping
    - in-addr.arpa
- More than 200 top-level domains

DNS Architecture

- Domain name space and resource records (RRs)
  - Domain namespace is a hierarchical tree structure
  - A domain can be delegated to an organization
  - We'll discuss resource records shortly
- Name servers
  - Domain name hierarchy exists only in the abstract
  - A host's name servers are specified in /etc/resolv.conf
- Name resolution

Hierarchy of Name Servers

- Resolution of the hierarchical namespace is done by hierarchy of name servers
- Namespace is partitioned into zones. A zone is a contiguous portion of the DNS namespace
- Each server is responsible (authoritative for a zone)
- DNS server answers queries about host names in its zone



DNS Domains and Zones

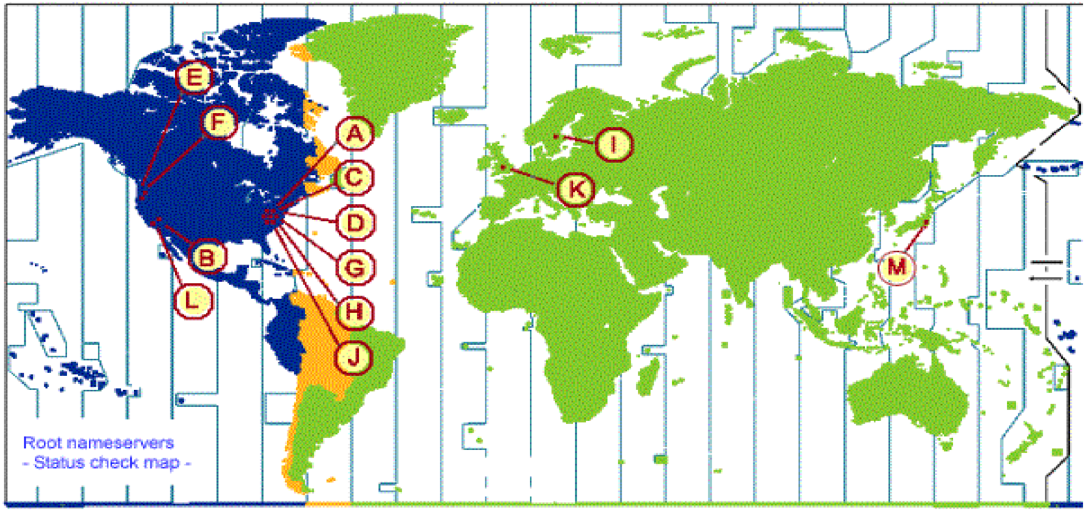
- Each zone is anchored at a specific domain node
- A DNS domain is a subtree of the namespace
- A zone is a portion of the DNS namespace
  - Generally stored in a file
  - Could consist of multiple nodes
- A server can divide part of its zone and delegate it to other servers
- A name server implements the zone information as a collection of resource records (RRs)

Name Servers

- Each zone has a primary and secondary name server
  - For reliability
  - Primary server maintains a zone file with zone info
    - Updates made to the primary server
  - Secondary server copies data stored at the primary server
- Adding a new host:
  - When new host is added (e.g. “newmachine.ece.duke.edu”)
  - Administrator adds the IP info on the host (IP address, name) to a configuration file on the primary server

Root Name Servers

Map of the Root Servers



- Root name servers know how to find authoritative name servers for all top-level zones
- There are 13 (virtual) root name servers
- Root servers are critical for proper functioning of name resolution

Resource Records

- A zone file includes a collection of Resource Records (RRs)
- (Name, Value, Type, Class, TTL)
  - Name and Value
  - Type specifies how the “Value” should be interpreted
    - e.g. “NS” means name is a domain and value is name of authoritative name server for this domain
    - e.g. “A” means a machine name and IP address
  - Class: allows other entities to define record types (“IN” for Internet is most widely used currently)
  - TTL: how long should the RR be cached (more on this later)

Resource Records

```
$TTL 86400

mylab.com. IN SOA PC4.mylab.com. admin@mylab.com. (
                       1 ; serial
                       28800 ; refresh
                       7200 ; retry
                       604800 ; expire
                       86400 ; minimum ttl
                       )
;


mylab.com.      IN      NS      PC4.mylab.com.
;


localhost               A       127.0.0.1
PC4.mylab.com.  A       10.0.1.41
PC3.mylab.com.  A       10.0.1.31
PC2.mylab.com.  A       10.0.1.21
PC1.mylab.com.  A       10.0.1.11
```

- Max age of cached data in seconds

- Start of authority (SOA) record.
  Means: “This name server is authoritative for the zone Mylab.com”
- PC4.mylab.com is the name server
- admin@mylab.com is the email address of the person in charge

- Name server (NS) record:
  one entry for each authoritative name server

- Address (A) records:
  one entry for each host address

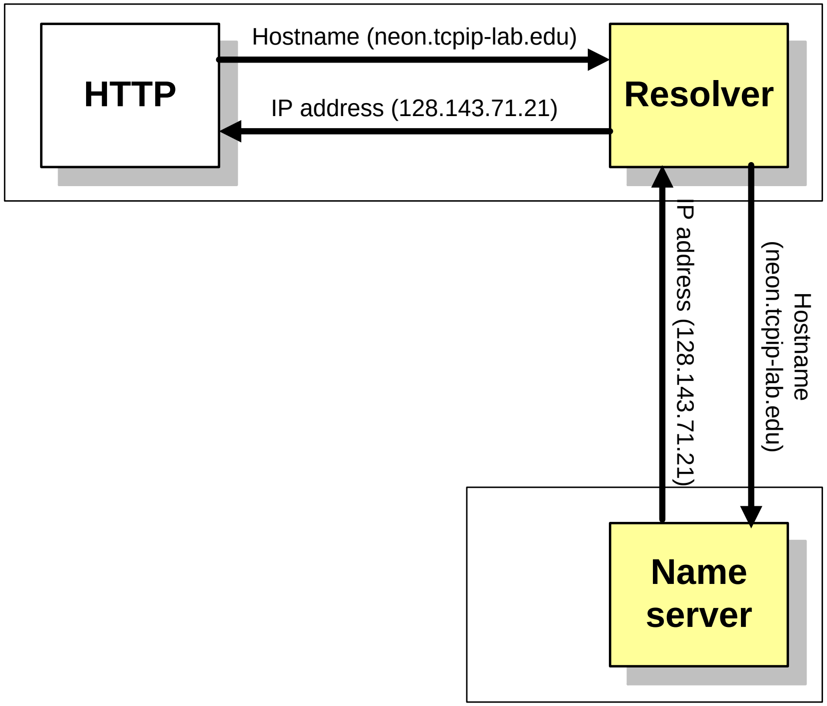
Domain Name Resolution

1. User program issues a request for the IP address of a hostname gethostbyname()

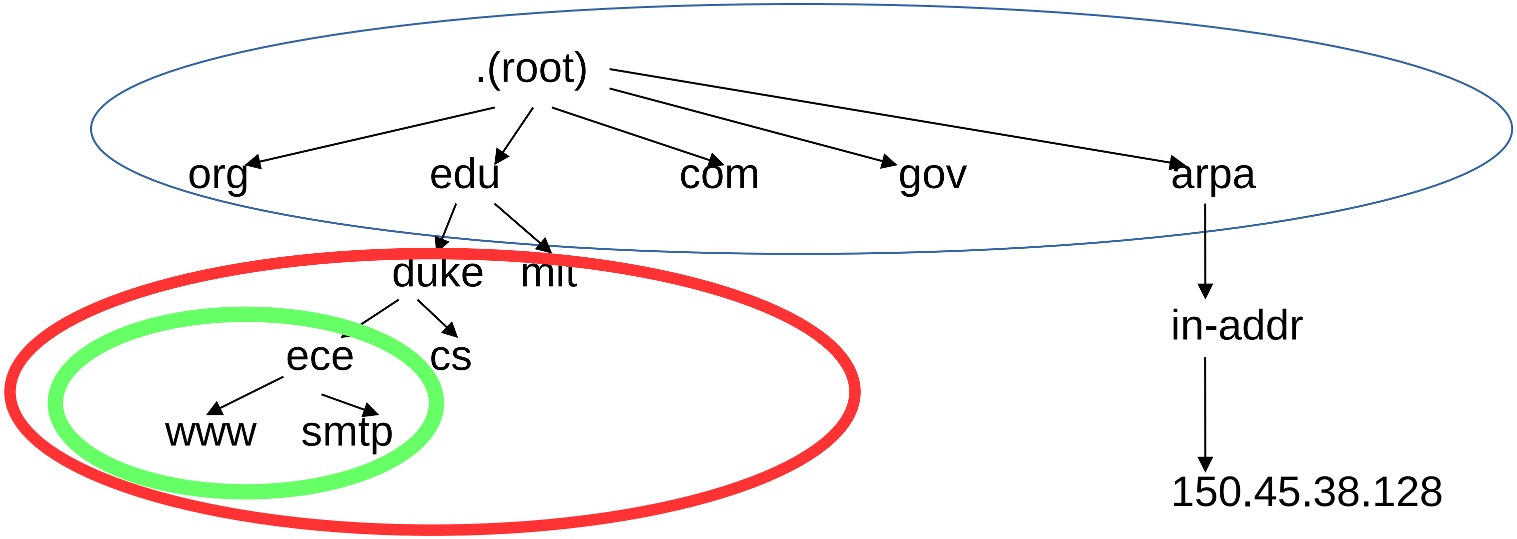
2. Local resolver formulates a DNS query to the name server of the host

3. Name server checks if it is authorized to answer the query.
   - If yes, it responds.
   - Otherwise, it will query other name servers, starting at the root tree

4. When the name server has the answer it sends it to the resolver.


HTTP
Hostname (neon.tcpip-lab.edu)
IP address (128.143.71.21)
Resolver
IP address (128.143.71.21)
Hostname (neon.tcpip-lab.edu)
Name server

Inverse Query



- What is the host name for IP address 150.45.38.128?
  - IP address is converted to domain name: 150.45.38.128.in-addr.arpa
  - Resolver sends query for this address

Caching

- To reduce DNS traffic, name servers cache info
  - e.g. Domain name / IP address mappings
- When entry for a query is cached, the server does not contact other servers
- Note: if an entry is sent from a cache, the reply from the server is marked as “unauthoritative”
- Caching-only servers
  - Only purpose is to cache results
  - Do not contain zone info or a zone database file

Negative Caching

- Two kinds of negative responses
  - Name in question does not exist
  - Name in record exists, but requested data does not
- Negative responses will be cached too

Modern follow-ons

- DNS with DHCP integration
  - When a new host uses DHCP to get on the network, the DHCP server can tell the DNS server about it, then the DNS server can answer requests for that host by name
- **Multicast DNS (mDNS)** and **Link-Local Multicast Name Resolution (LLMNR)**
  - Resolve hostnames when there’s no local DNS server
  - Allows “automagic” host discovery on individual networks
  - Zero configuration – they’re self-organizing protocols