After Mid-Term Exam Hours

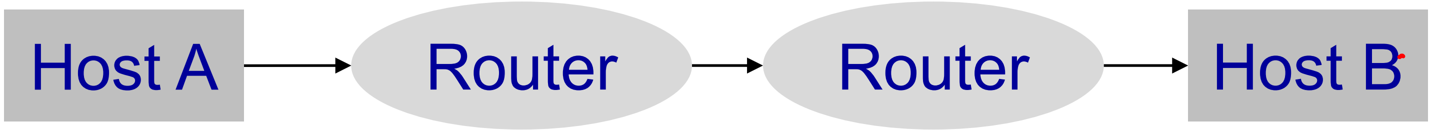
- Mid-Term Exam
  - @SF-3202
  - 2 hours
- Students asked “what about the remaining 1 hour, will any class be taught during the 1-hour after exam?”
  - The scheduled exam place is not in our regular lecture room
  - Some students may submit exam paper early
  - We can do a 1-hour video lecture instead to save your time from traveling between lecture room and exam room

ECE568 Lecture 06:
Network Security 01 – Spoofing Attack

Wei Huang
Department of Electrical and Computer Engineering
University of Toronto

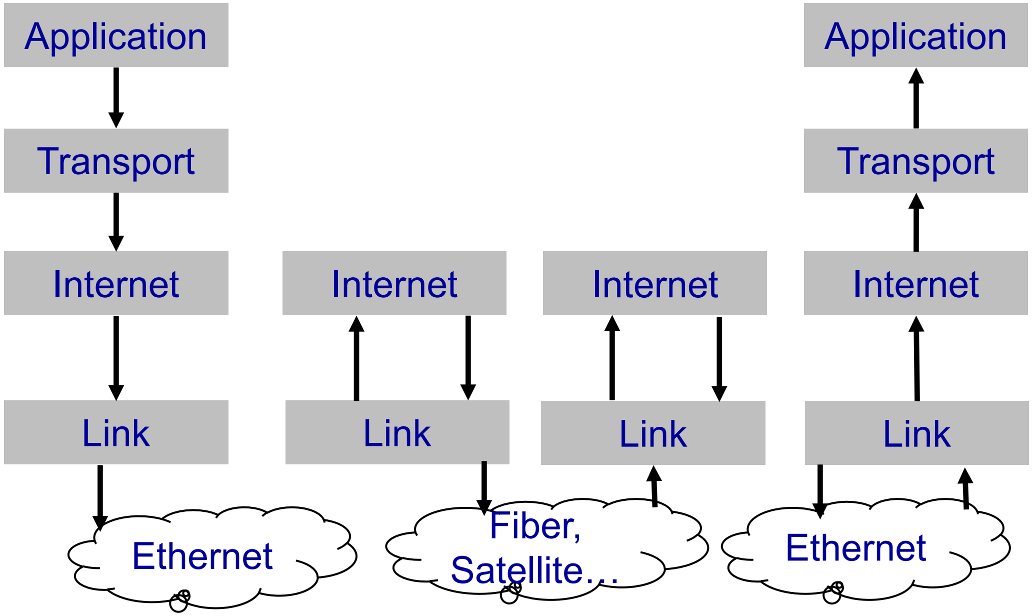
OSI Network Stack model

Host Connections



Host A → Router → Router → Host B

Network Stack Connections



Application

Transport

Internet

Link

Ethernet

Internet

Link

Fiber,
Satellite…

Internet

Link

Ethernet

Application

Transport

Internet

Link

L7: DNS, BGP, SMTP, NTP

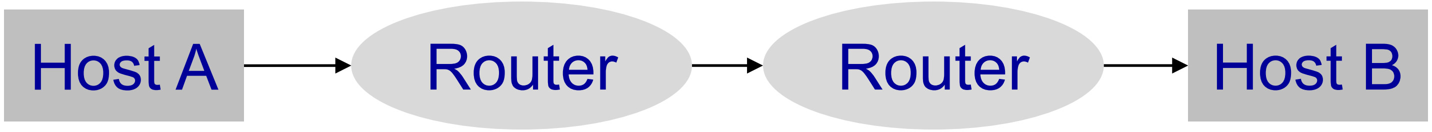
L4: TCP, UDP

L3: IP, ICMP, IPSec

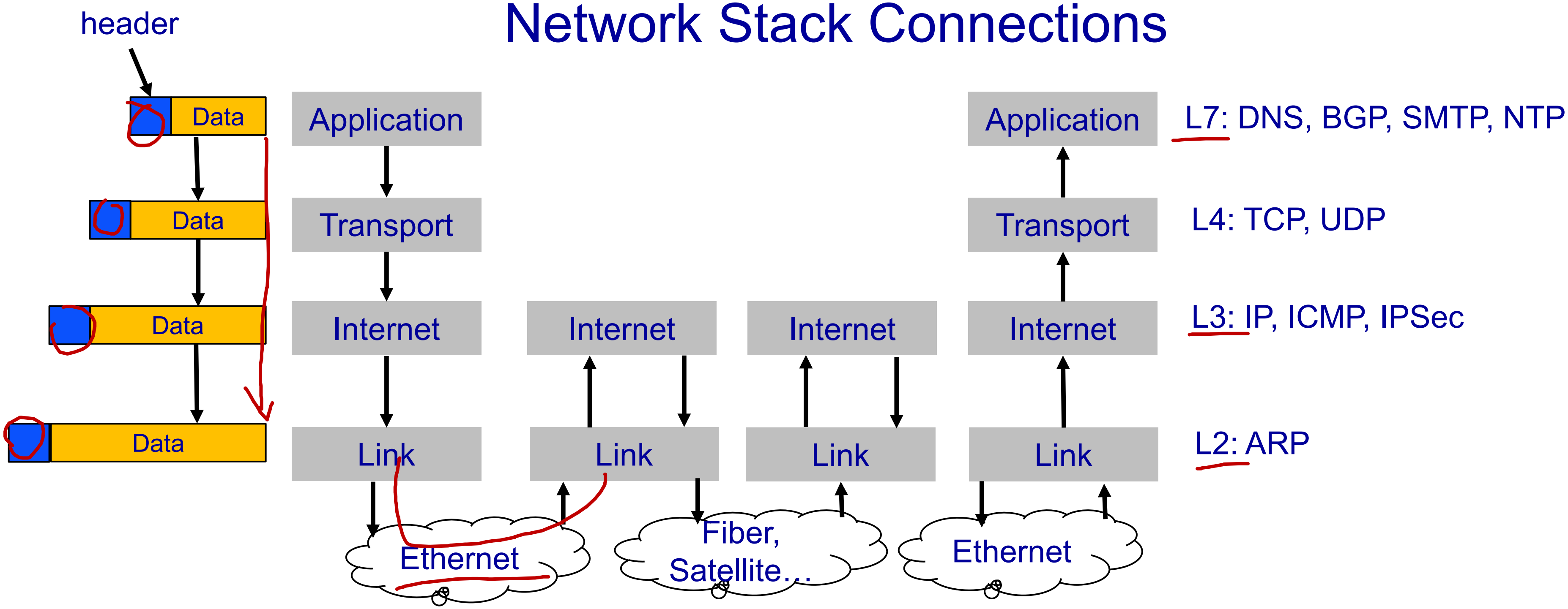
L2: ARP

OSI Network Stack model

Host Connections



Network Stack Connections



L7: DNS, BGP, SMTP, NTP

L4: TCP, UDP

L3: IP, ICMP, IPSec

L2: ARP

Network attacks

Many of our Internet protocols were designed assuming that all parties with access to the Internet were trusted

  - There was almost no security, and any checks that were instituted were primarily for finding misconfigured systems, rather than dealing with malicious systems

- Broadly two categories of attacks:
  - Spoofing: fake the identity of a victim
  - Denial of service: Prevent communication between victims

Spoofing

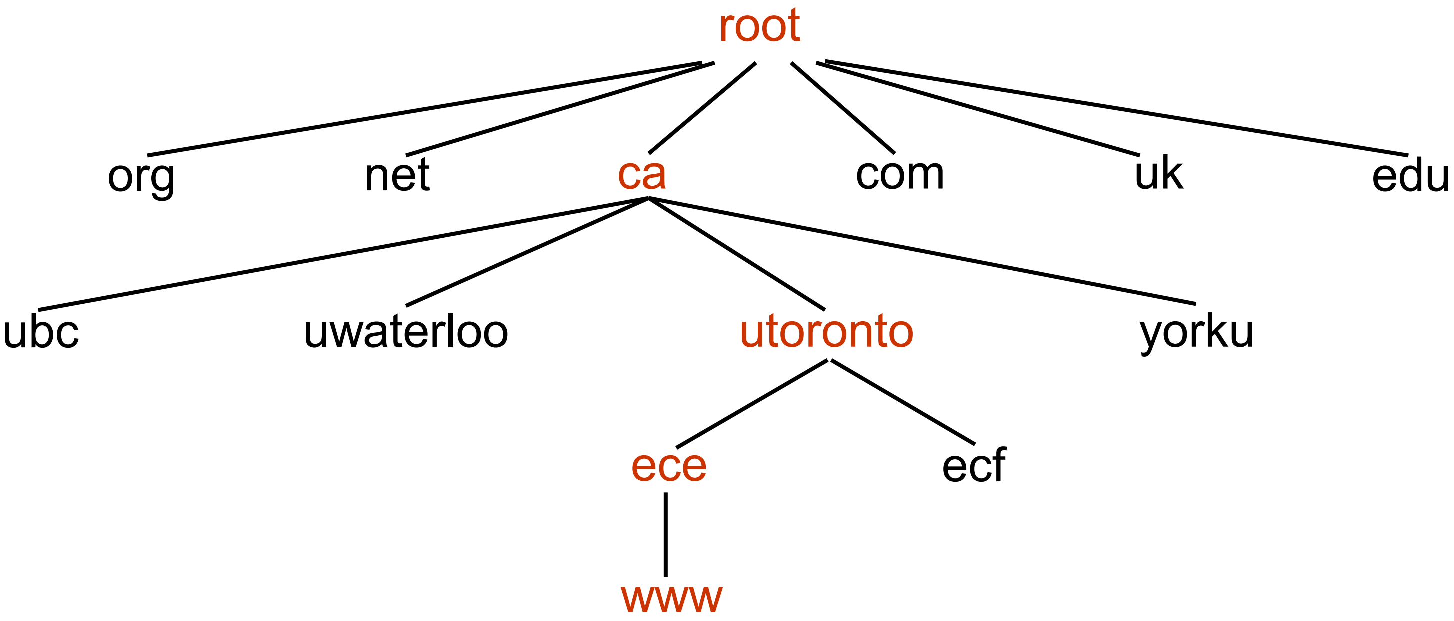
- Networks are really not designed to authenticate the source
  - Why TLS/SSL is so important
  - In general, sender can write any address they want into the source address
    - e.g. NAT can rewrite source
- Spoofing can happen at many levels:
  - Layer 7/Application: BGP, DNS
  - Layer 3: TCP
  - Layer 2: ARP

Domain Name System (DNS)

The Domain Name System (DNS) is a hierarchical naming system for resources on the Internet

- It maps symbolic names to numeric IP addresses

www.ece.utoronto.ca ↔ 128.100.131.138

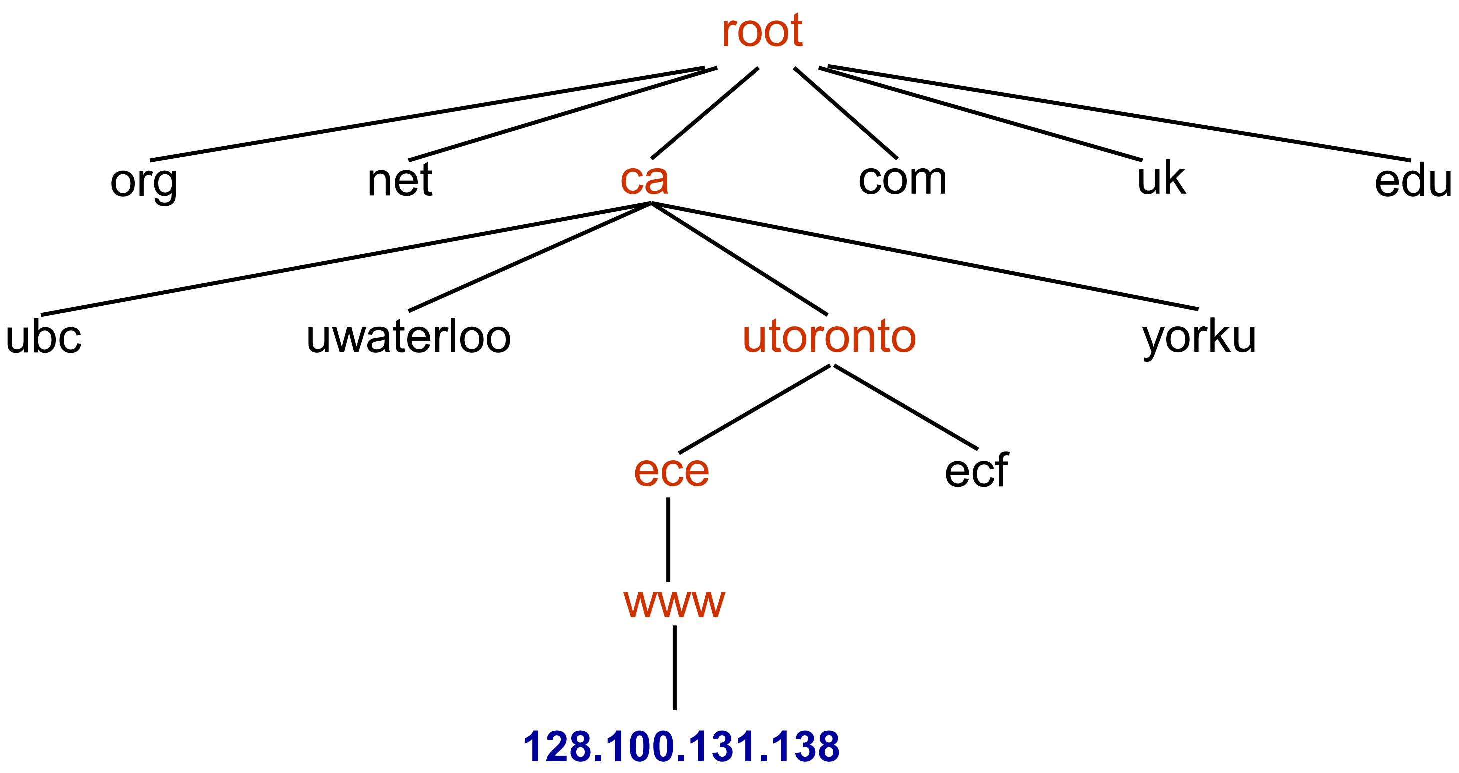


DNS Name Servers

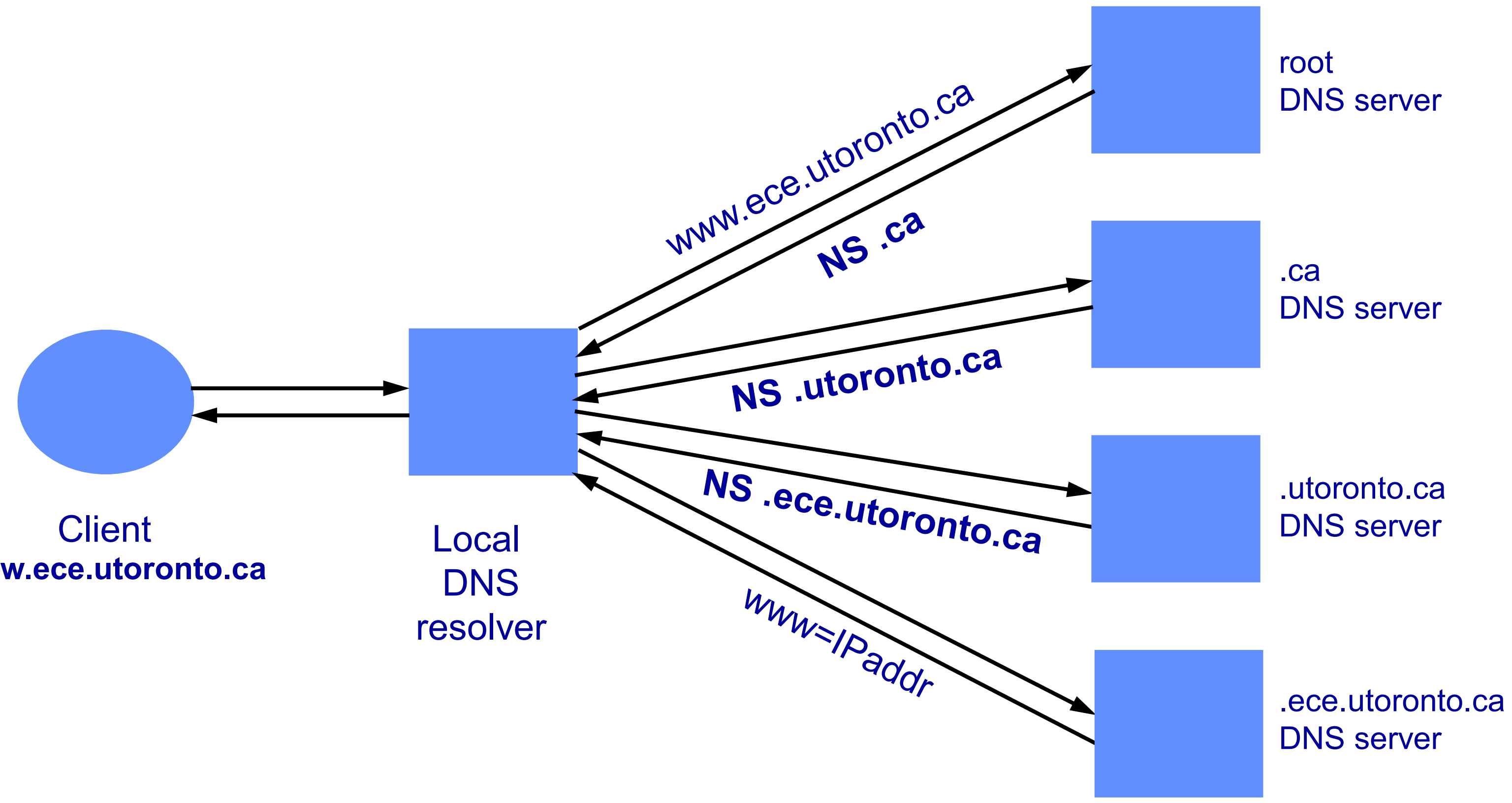
DNS maps names to IP addresses using a set of **authoritative name servers**

- Each domain has an authoritative name server that is responsible for the DNS mappings for its domain, and in turn can assign other authoritative name servers for their sub-domains. There can also be caching name servers that replicate mappings for load balance
- Example:
  - NS for utoronto.ca is ns1.utoronto.ca
  - NS for ece.utoronto.ca is ugsparc0.eecg.utoronto.ca
- This hierarchy makes DNS distributed, and helps avoid the need for a single central register to be continually consulted and updated

DNS Name Server



DNS Lookup Example



DNS Lookup

A client performs a **DNS lookup** (query) to the local DNS software called a **resolver**

- The resolver starts by querying the name server at the top level of the DNS hierarchy
- Each name server replies with information about the authoritative server (name of the server and possibly IP address) one level down the hierarchy
- The resolver repeats the previous step until the IP address is returned
- Each query has a unique **query id** that helps associate the response with the request

DNS Caching

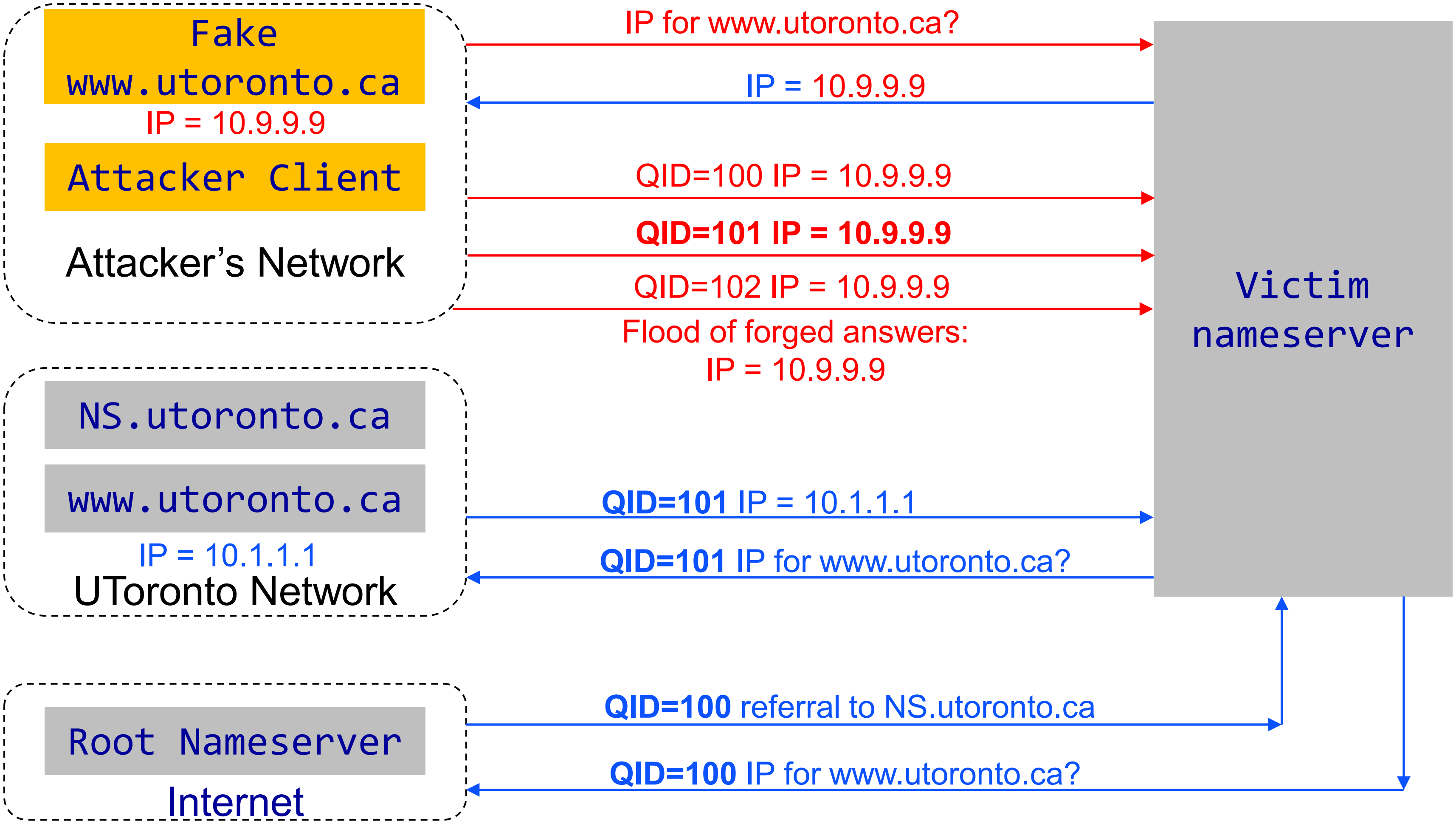
- DNS responses are cached at name servers
  - Allows quick response for repeated queries
- DNS negative queries are cached also
  - Saves time for nonexistent sites, e.g. misspelling
- Cached data periodically times out
  - Lifetime (TTL) of data is controlled by owner of data
  - TTL ranges from seconds to days
    - Higher TTL is more efficient
      - Less DNS requests are made
  - Shorter TTL allows better load balancing
    - The same name is mapped to different IP addresses to spread load among web servers in a server farm

DNS Cache Poisoning

Users/hosts trust DNS mappings at name servers, although these mappings are not authenticated

- If an attacker is able to update a DNS server’s cache with bogus mappings, then hosts would be served these **poisoned** mappings
- How is it possible to poison a DNS cache?
  - Exploit vulnerability in DNS software
    - *e.g.*, BIND v4.9 had buffer overflow
  - Spoof DNS response
    - For a single host
    - For an entire subdomain
    - Let’s see how the spoofing works

Basic (Naïve) DNS Poisoning Attack



Fake www.utoronto.ca

IP = 10.9.9.9

Attacker Client

Attacker's Network

NS.utoronto.ca

www.utoronto.ca

IP = 10.1.1.1

UToronto Network

Root Nameserver

Internet

Victim nameserver

IP for www.utoronto.ca?

IP = 10.9.9.9

QID=100 IP = 10.9.9.9

QID=101 IP = 10.9.9.9

QID=102 IP = 10.9.9.9

Flood of forged answers:
IP = 10.9.9.9

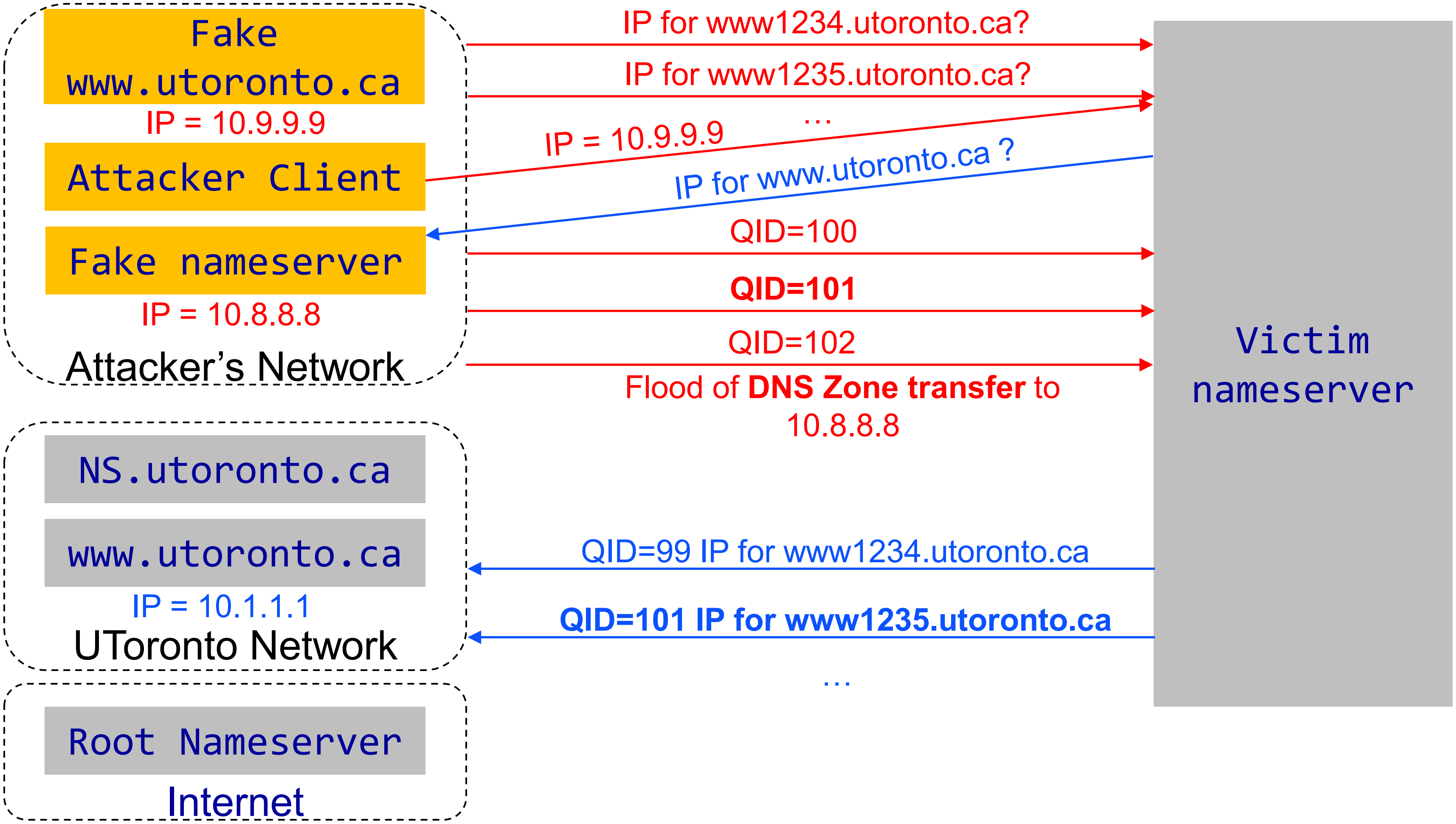
QID=101 IP = 10.1.1.1

QID=101 IP for www.utoronto.ca?

QID=100 referral to NS.utoronto.ca

QID=100 IP for www.utoronto.ca?

Effective Kaminsky attack


Fake
www.utoronto.ca
IP = 10.9.9.9
Attacker Client
Fake nameserver
IP = 10.8.8.8
Attacker's Network
NS.utoronto.ca
www.utoronto.ca
IP = 10.1.1.1
UToronto Network
Root Nameserver
Internet
IP for www1234.utoronto.ca?
IP for www1235.utoronto.ca?
…
IP = 10.9.9.9
IP for www.utoronto.ca ?
QID=100
QID=101
QID=102
Flood of DNS Zone transfer to
10.8.8.8
Victim
nameserver
QID=99 IP for www1234.utoronto.ca
QID=101 IP for www1235.utoronto.ca
…