Putting DNS in Context

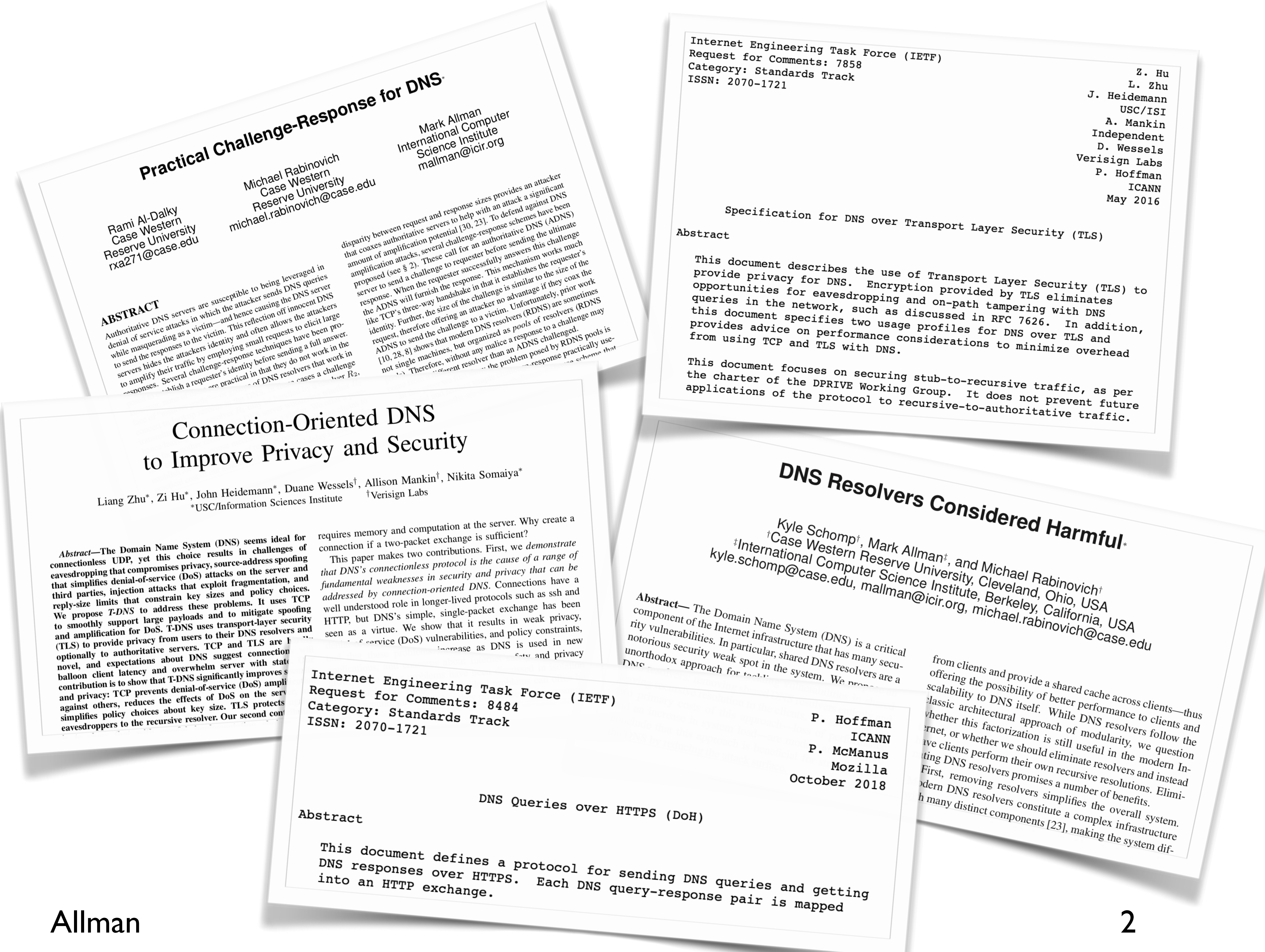
Mark Allman

International Computer Science Institute

IMC 2020

October 2020

"I was drivin' through the misty rain
Searchin' for a mystery train
Boppin' through the wild blue
Tryin' to make a connection with you"



Practical Challenge-Response for DNS

Rami Al-Dalky, Case Western Reserve University, rxa271@case.edu
Michael Rabinovich, Case Western Reserve University, michael.rabinovich@case.edu
Mark Allman, International Computer Science Institute, mallman@icir.org

ABSTRACT

Authoritative DNS servers are susceptible to being leveraged in denial of service attacks in which the attacker sends DNS queries while masquerading as a victim—and hence causing the DNS server to send the responses to the victim. This reflection off innocent DNS servers hides the attackers identity and often allows the attackers to amplify their traffic by employing small requests to elicit large responses. Several challenge-response techniques have been proposed to establish a requester's identity before sending a full answer. …

disparity between request and response sizes provides an attacker that coaxes authoritative servers to help with an attack a significant amount of amplification potential [30, 23]. To defend against DNS amplification attacks, several challenge-response schemes have been proposed (see § 2). These call for an authoritative DNS (ADNS) server to send a challenge to requester before sending the ultimate response. When the requester successfully answers this challenge the ADNS will furnish the response. This mechanism works much like TCP's three-way handshake in that it establishes the requester's identity. Further, the size of the challenge is similar to the size of the request, therefore offering an attacker no advantage if they coax the ADNS to send the challenge to a victim. Unfortunately, prior work [10, 28, 8] shows that modern DNS resolvers (RDNS) are sometimes not single machines, but organized as *pools* of resolvers …

Internet Engineering Task Force (IETF)
Request for Comments: 7858
Category: Standards Track
ISSN: 2070-1721

Z. Hu
L. Zhu
J. Heidemann
USC/ISI
A. Mankin
Independent
D. Wessels
Verisign Labs
P. Hoffman
ICANN
May 2016

Specification for DNS over Transport Layer Security (TLS)

Abstract

This document describes the use of Transport Layer Security (TLS) to provide privacy for DNS. Encryption provided by TLS eliminates opportunities for eavesdropping and on-path tampering with DNS queries in the network, such as discussed in RFC 7626. In addition, this document specifies two usage profiles for DNS over TLS and provides advice on performance considerations to minimize overhead from using TCP and TLS with DNS.

This document focuses on securing stub-to-recursive traffic, as per the charter of the DPRIVE Working Group. It does not prevent future applications of the protocol to recursive-to-authoritative traffic.

Connection-Oriented DNS to Improve Privacy and Security

Liang Zhu*, Zi Hu*, John Heidemann*, Duane Wessels†, Allison Mankin†, Nikita Somaiya*
*USC/Information Sciences Institute †Verisign Labs

***Abstract*—The Domain Name System (DNS) seems ideal for connectionless UDP, yet this choice results in challenges of eavesdropping that compromises privacy, source-address spoofing that simplifies denial-of-service (DoS) attacks on the server and third parties, injection attacks that exploit fragmentation, and reply-size limits that constrain key sizes and policy choices. We propose *T-DNS* to address these problems. It uses TCP to smoothly support large payloads and to mitigate spoofing and amplification for DoS. T-DNS uses transport-layer security (TLS) to provide privacy from users to their DNS resolvers and optionally to authoritative servers. …**

requires memory and computation at the server. Why create a connection if a two-packet exchange is sufficient?

This paper makes two contributions. First, we *demonstrate that DNS's connectionless protocol is the cause of a range of fundamental weaknesses in security and privacy that can be addressed by connection-oriented DNS.* Connections have a well understood role in longer-lived protocols such as ssh and HTTP, but DNS's simple, single-packet exchange has been seen as a virtue. We show that it results in weak privacy, … (DoS) vulnerabilities, and policy constraints, …

DNS Resolvers Considered Harmful

Kyle Schomp†, Mark Allman‡, and Michael Rabinovich†
†Case Western Reserve University, Cleveland, Ohio, USA
‡International Computer Science Institute, Berkeley, California, USA
kyle.schomp@case.edu, mallman@icir.org, michael.rabinovich@case.edu

Abstract— The Domain Name System (DNS) is a critical component of the Internet infrastructure that has many security vulnerabilities. In particular, shared DNS resolvers are a notorious security weak spot in the system. …

from clients and provide a shared cache across clients—thus offering the possibility of better performance to clients and scalability to DNS itself. While DNS resolvers follow the classic architectural approach of modularity, we question whether this factorization is still useful in the modern Internet, or whether we should eliminate resolvers and instead have clients perform their own recursive resolutions. Eliminating DNS resolvers promises a number of benefits. First, removing resolvers simplifies the overall system. Modern DNS resolvers constitute a complex infrastructure with many distinct components [23], making the system dif-

---

Internet Engineering Task Force (IETF)
Request for Comments: 8484
Category: Standards Track
ISSN: 2070-1721

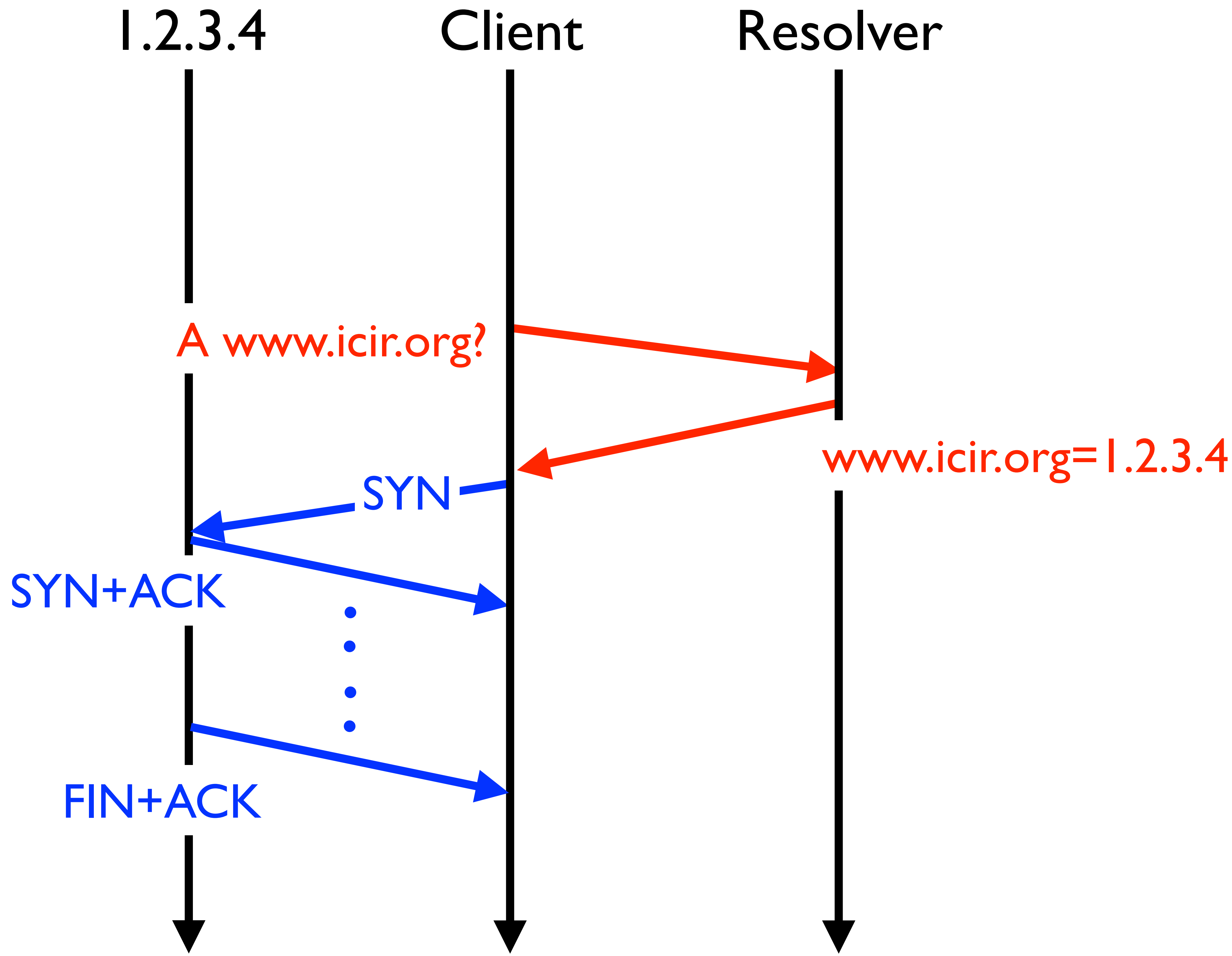
P. Hoffman
ICANN
P. McManus
Mozilla
October 2018

DNS Queries over HTTPS (DoH)

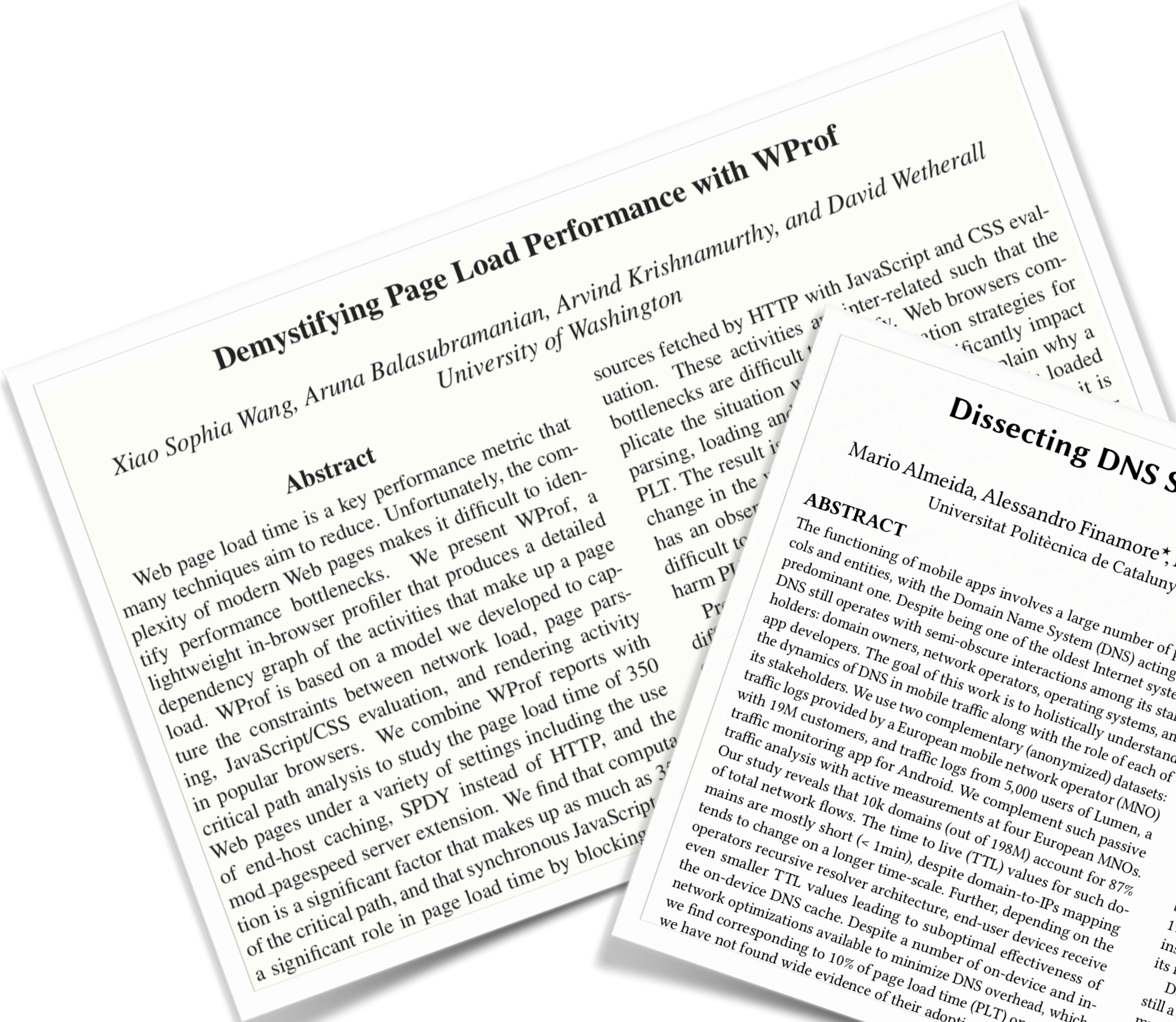
Abstract

This document defines a protocol for sending DNS queries and getting DNS responses over HTTPS. Each DNS query-response pair is mapped into an HTTP exchange.

Textbook DNS Usage



Previous Work



Demystifying Page Load Performance with WProf

Xiao Sophia Wang, Aruna Balasubramanian, Arvind Krishnamurthy, and David Wetherall
University of Washington

Abstract

Web page load time is a key performance metric that many techniques aim to reduce. Unfortunately, the complexity of modern Web pages makes it difficult to identify performance bottlenecks. We present WProf, a lightweight in-browser profiler that produces a detailed dependency graph of the activities that make up a page load. WProf is based on a model we developed to capture the constraints between network load, page parsing, JavaScript/CSS evaluation, and rendering activity in popular browsers. We combine WProf reports with critical path analysis to study the page load time of 350 Web pages under a variety of settings including the use of end-host caching, SPDY instead of HTTP, and the mod_pagespeed server extension. We find that computation is a significant factor that makes up as much as 3... of the critical path, and that synchronous JavaScript ... a significant role in page load time by blocking ...

sources fetched by HTTP with JavaScript and CSS evaluation. These activities are inter-related such that the bottlenecks are difficult ... Web browsers complicate the situation ... strategies for parsing, loading and ... significantly impact PLT. The result is ... explain why a change in the ... loaded has an obser... it is difficult to ... harm PL...

Pr... dif...

Dissecting DNS Stakeholders in Mobile Networks

Mario Almeida, Alessandro Finamore★, Diego Perino★, Narseo Vallina-Rodriguez†, Matteo Varvello‡★
Universitat Politècnica de Catalunya, ★Telefónica Research, †IMDEA Networks/ICSI, ‡AT&T

ABSTRACT

The functioning of mobile apps involves a large number of protocols and entities, with the Domain Name System (DNS) acting as a predominant one. Despite being one of the oldest Internet systems, DNS still operates with semi-obscure interactions among its stakeholders: domain owners, network operators, operating systems, and app developers. The goal of this work is to holistically understand the dynamics of DNS in mobile traffic along with the role of each of its stakeholders. We use two complementary (anonymized) datasets: traffic logs provided by a European mobile network operator (MNO) with 19M customers, and traffic logs from 5,000 users of Lumen, a traffic monitoring app for Android. We complement such passive traffic analysis with active measurements at four European MNOs. Our study reveals that 10k domains (out of 198M) account for 87% of total network flows. The time to live (TTL) values for such domains are mostly short (< 1min), despite domain-to-IPs mapping tends to change on a longer time-scale. Further, depending on the operators recursive resolver architecture, end-user devices receive even smaller TTL values leading to suboptimal effectiveness of the on-device DNS cache. Despite a number of on-device and in-network optimizations available to minimize DNS overhead, which we find corresponding to 10% of page load time (PLT) on average, we have not found wide evidence of their adoption in the wild.

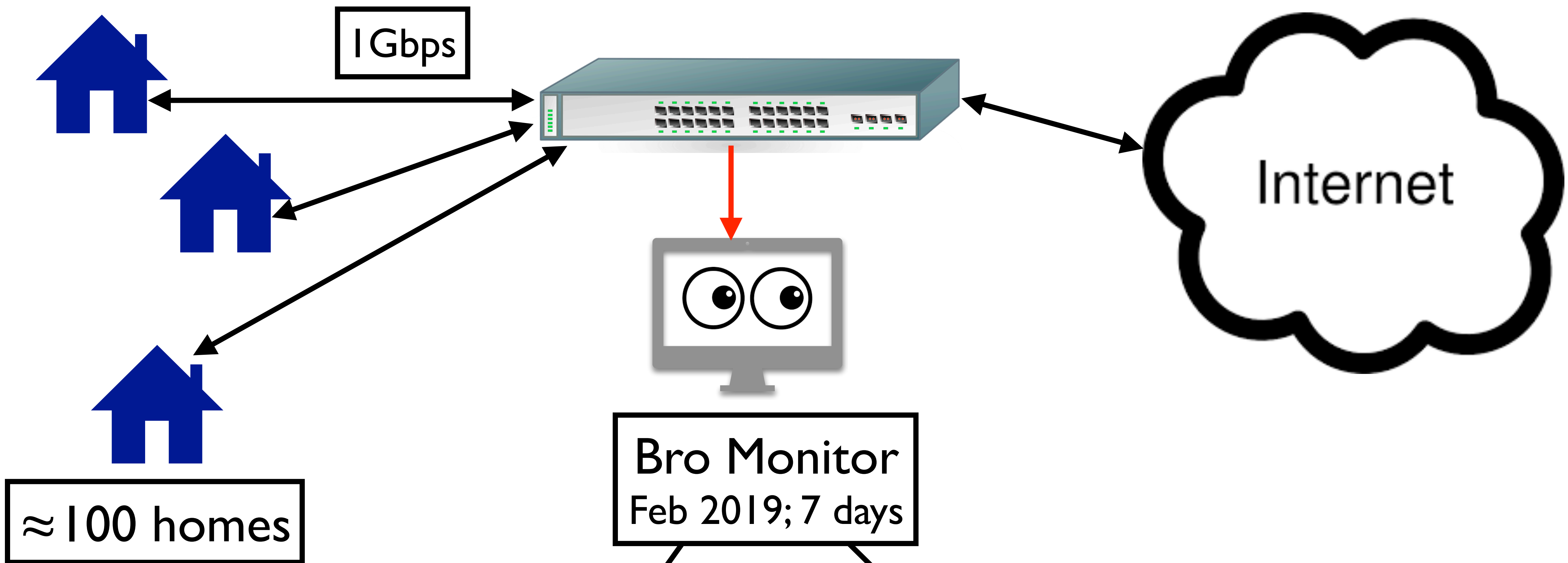
1 INTRODUCTION

According to recent estimates, mobile traffic is expected to have a sevenfold increase by 2021 [9]. At such growing rate, improving network architectures and understanding quality of experience (QoE) are fundamental steps towards shaping and optimizing current and future mobile networks. A recent study [20] revealed that, for some mobile operators, the Domain Name System (DNS) accounts for the highest fraction of flows and up to 50% of the total traffic. However, it is unclear *who* is responsible for such traffic load, and its impact on the end-user QoE.

Many previous studies investigated QoE in mobile networks focusing on inefficiencies of the access technology and network path [5, 7, 17, 19, 22, 25, 31], and in-path proxies [1, 14, 27]. DNS has been considered as part of general traffic performance studies [7, 17, 19], and (small scale) active experiments [23]. Less attention was instead given to DNS in mobile networks and, most importantly, its impact on users QoE.

Despite recent evolutions such as 4G, mobile networks last mile is still a shared access medium suffering from high latency. Hence communications should be optimized (or avoided) when possible. Considering DNS, on-device DNS caches and app-specific optimizations such as pre-fetching [15] are adopted to minimize DNS lookups. However, DNS *stakeholders*—domain owners, mobile network op-

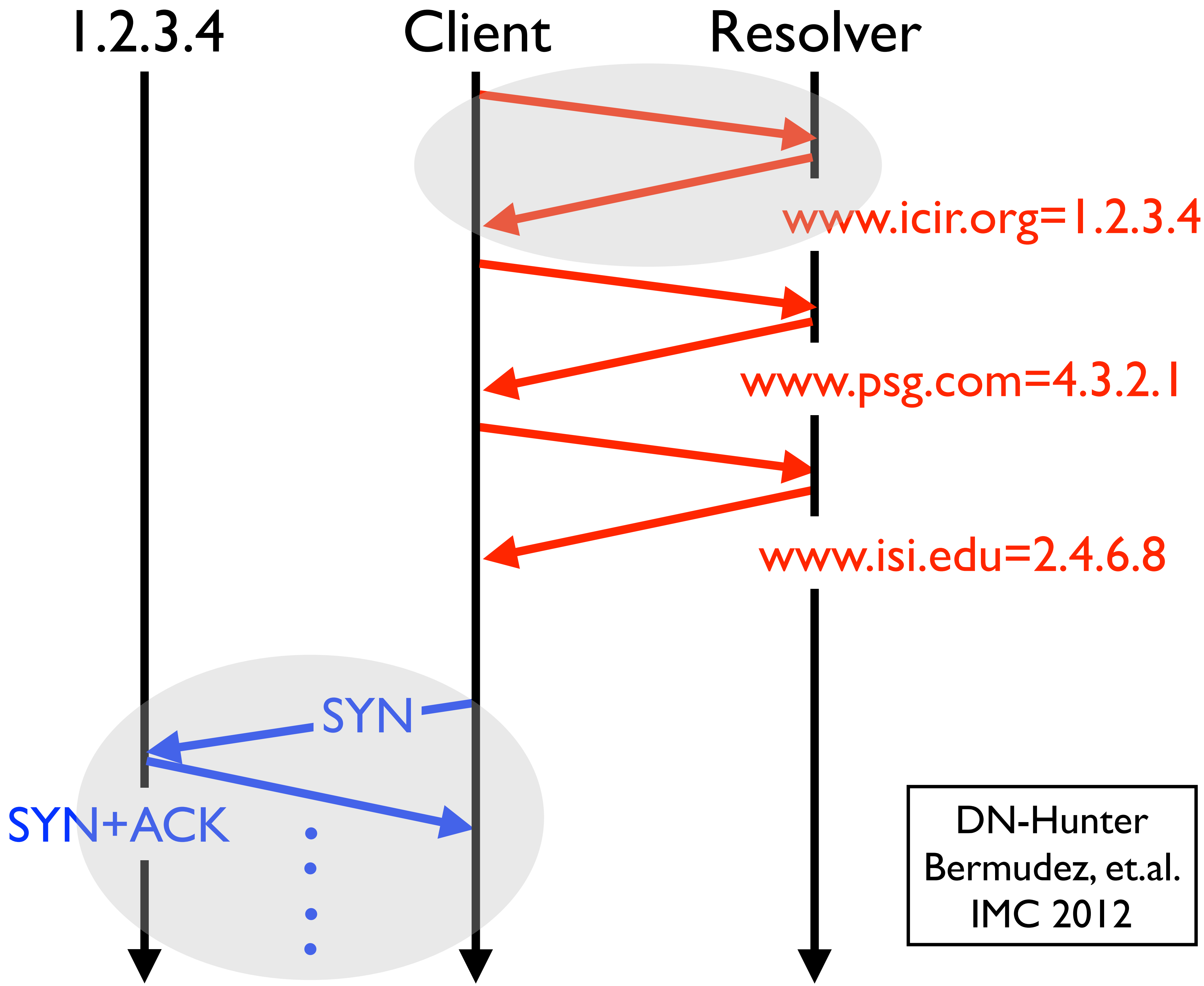
Dataset From
Case Connection Zone



9.2M DNS Transactions	
ISP	72.8%
Google	12.9%
OpenDNS	9.4%
Cloudflare	3.9%

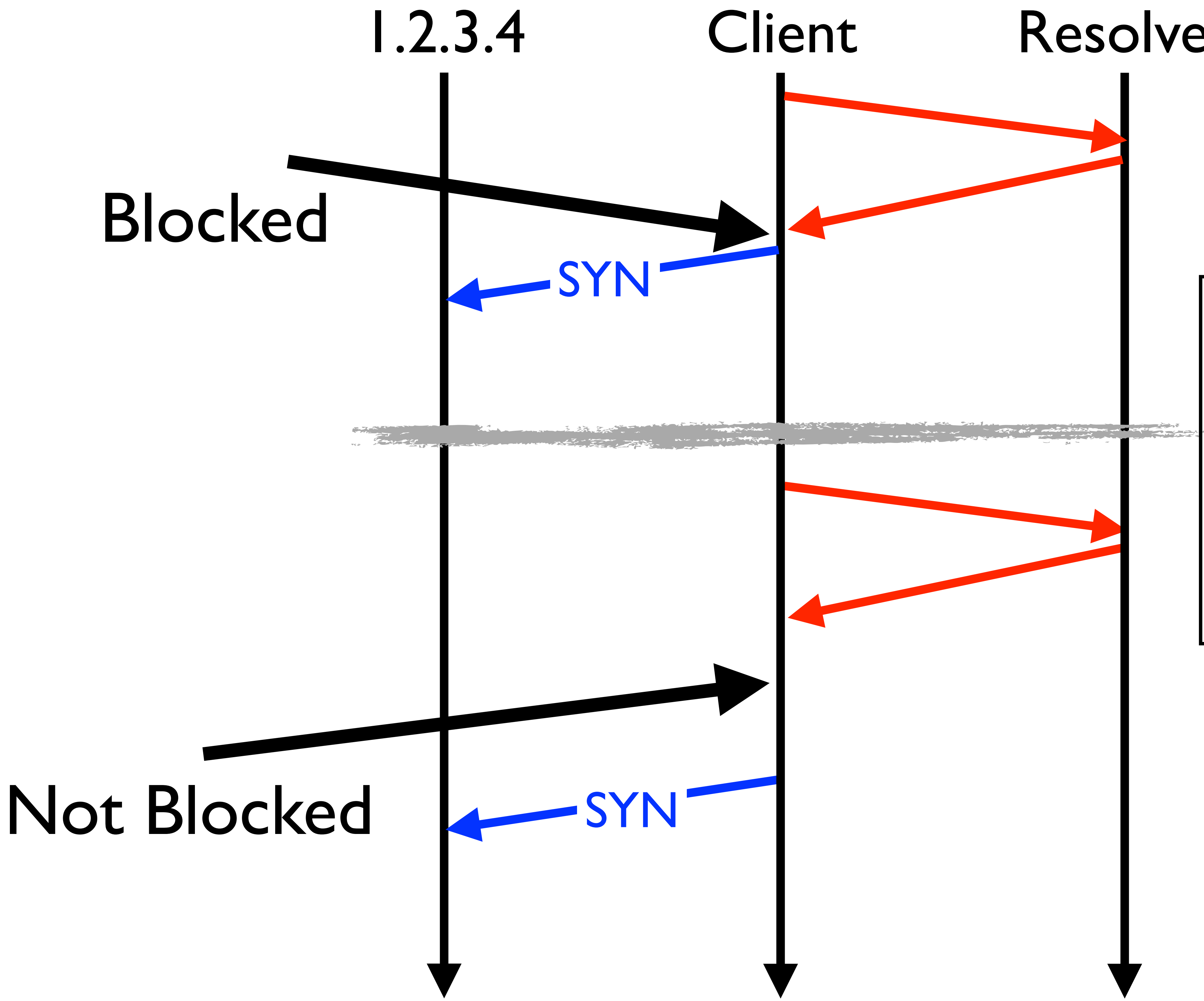
11.2M App. Transactions	
TCP	88%
UDP	12%

Methodology: Pairing



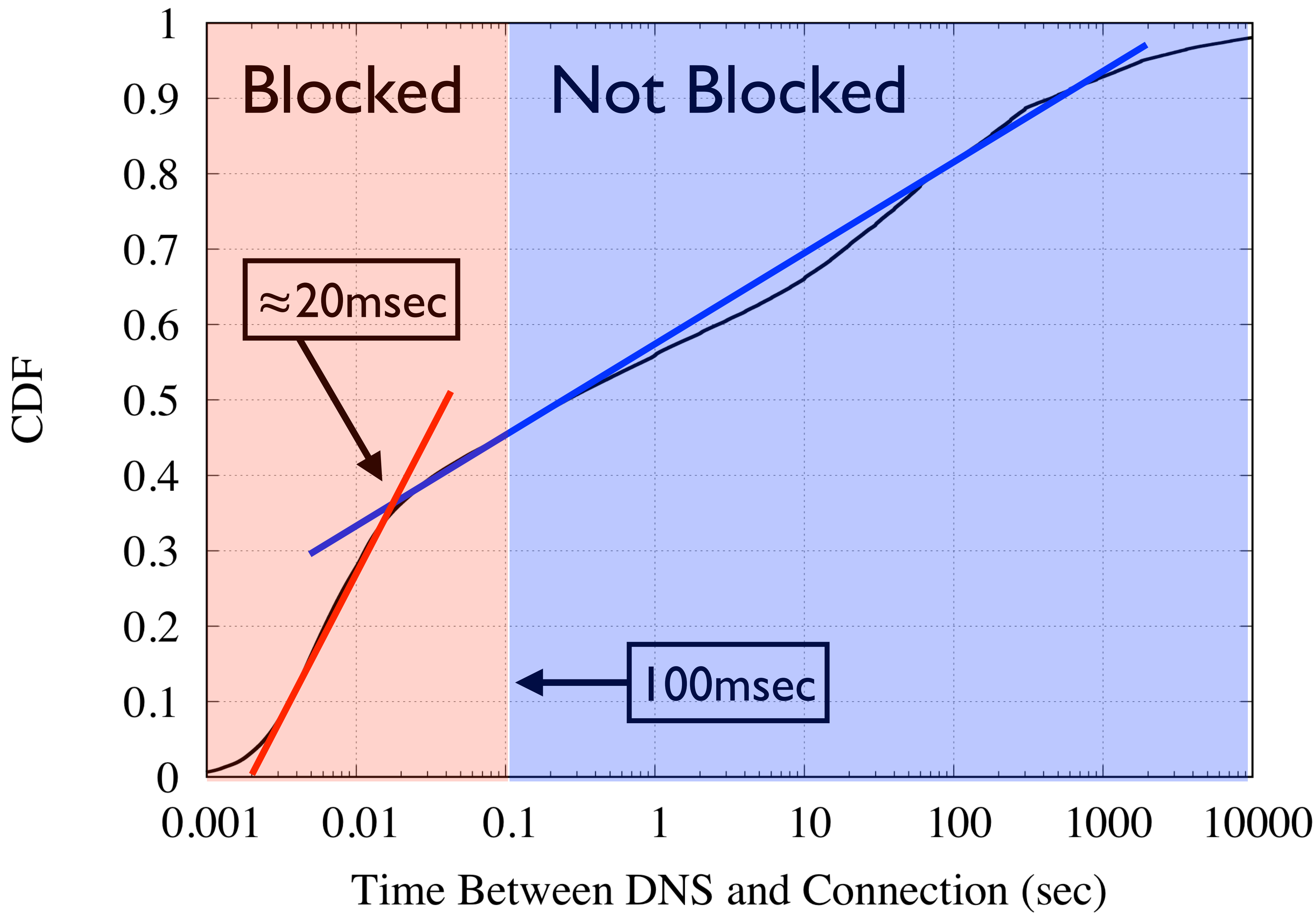
DN-Hunter
Bermudez, et.al.
IMC 2012

Methodology: Blocking



How do we determine whether a connection is "blocked"?

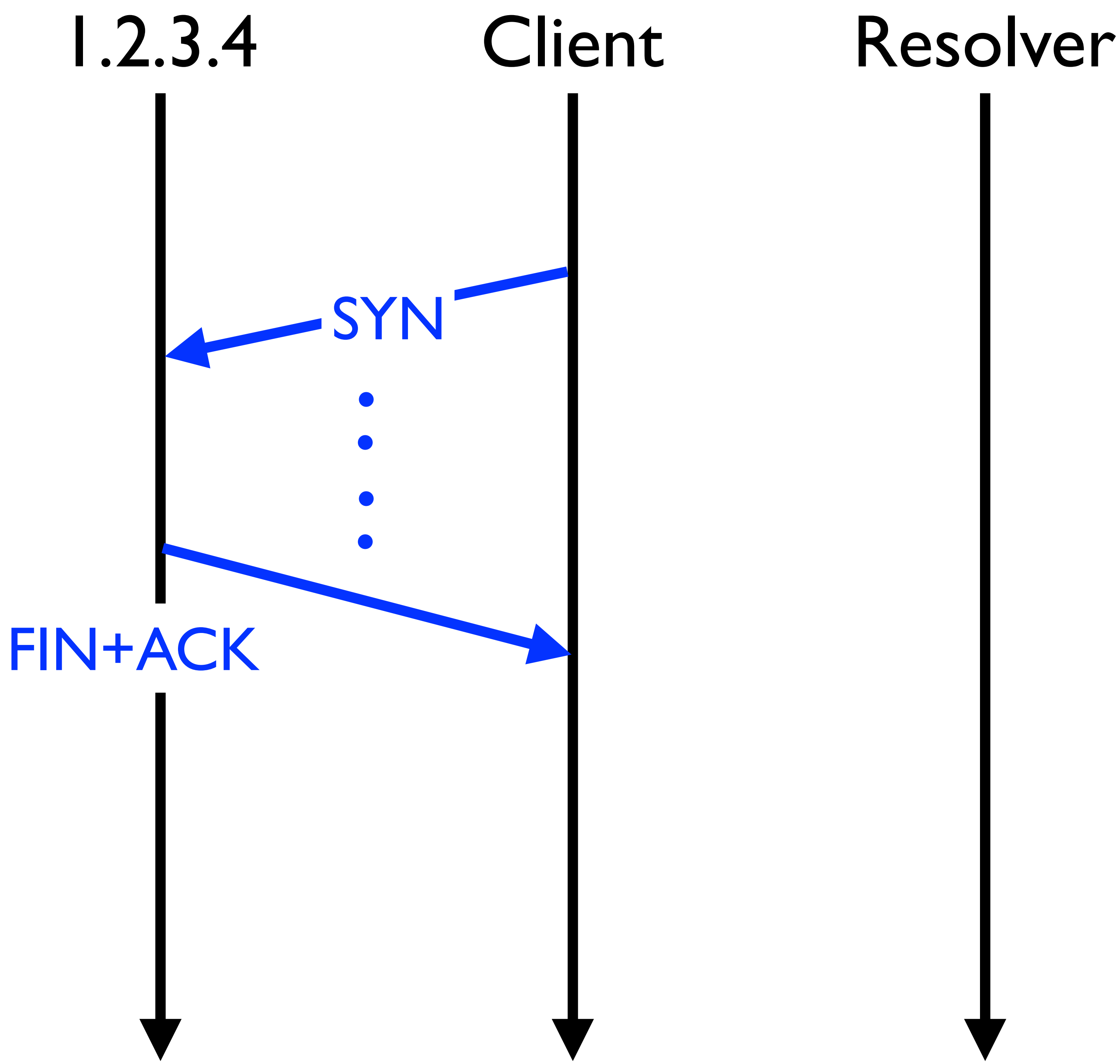
Methodology: Blocking



DNS Record Origin

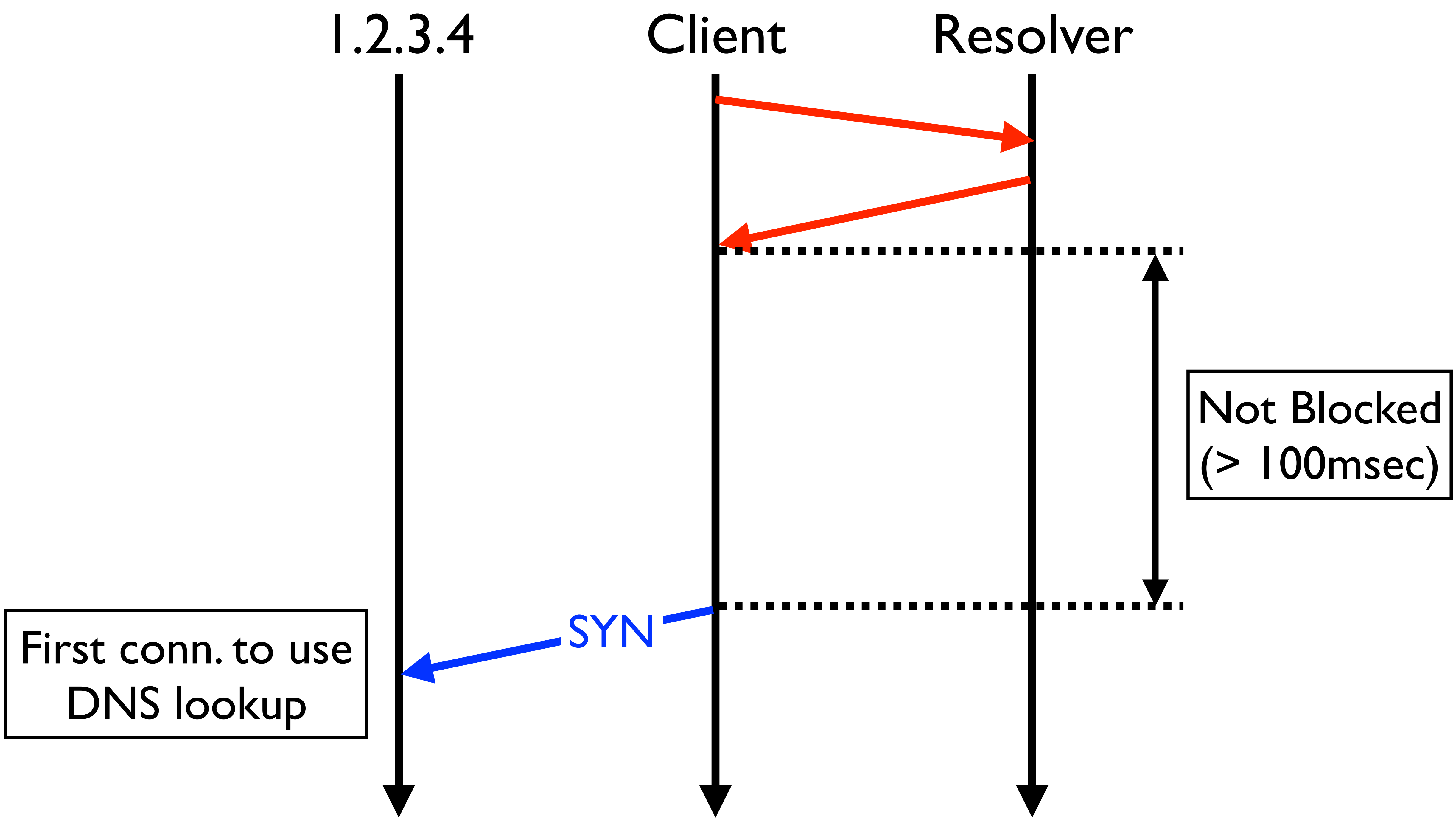
- We'll now use our two primitives—pairing and blocking—to catagorize application transactions based on where the corresponding DNS information comes from

No DNS

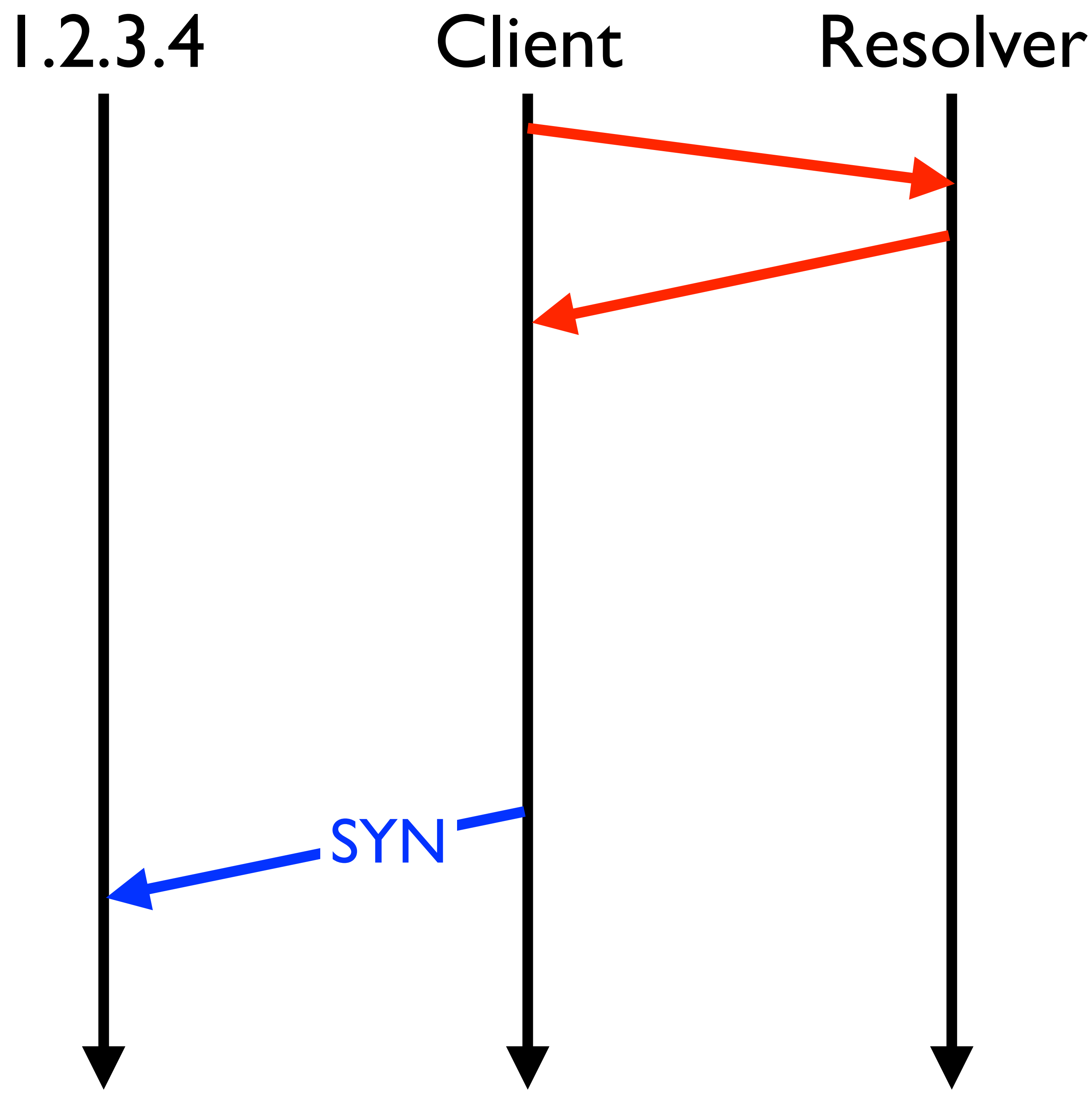


- 7.2% of conns.
- 82% are p2p
- the remaining cases seem to largely represent hard-coded IPs

Prefetched

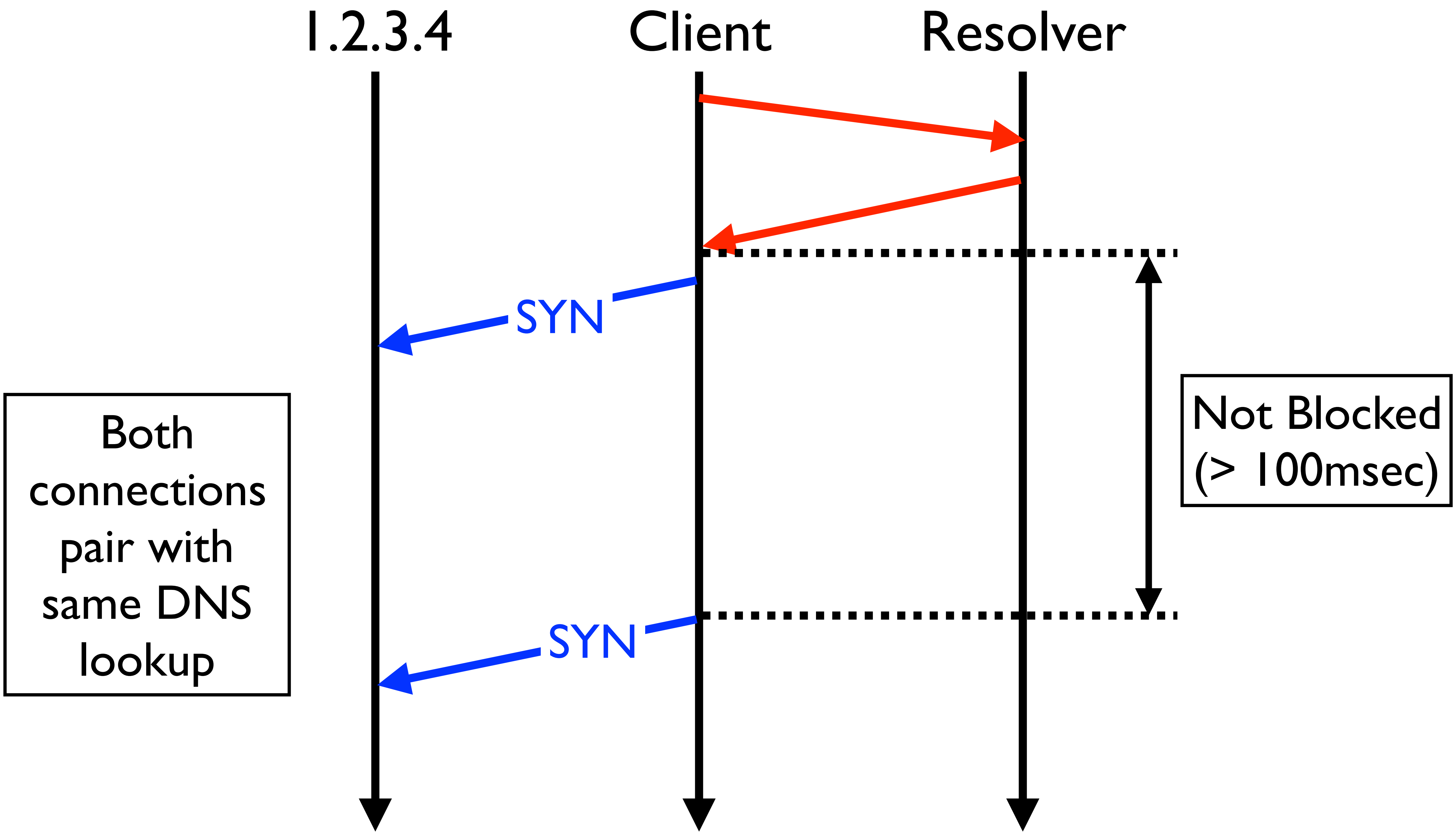

1.2.3.4
Client
Resolver
Not Blocked
(> 100msec)
SYN
First conn. to use
DNS lookup

Prefetched



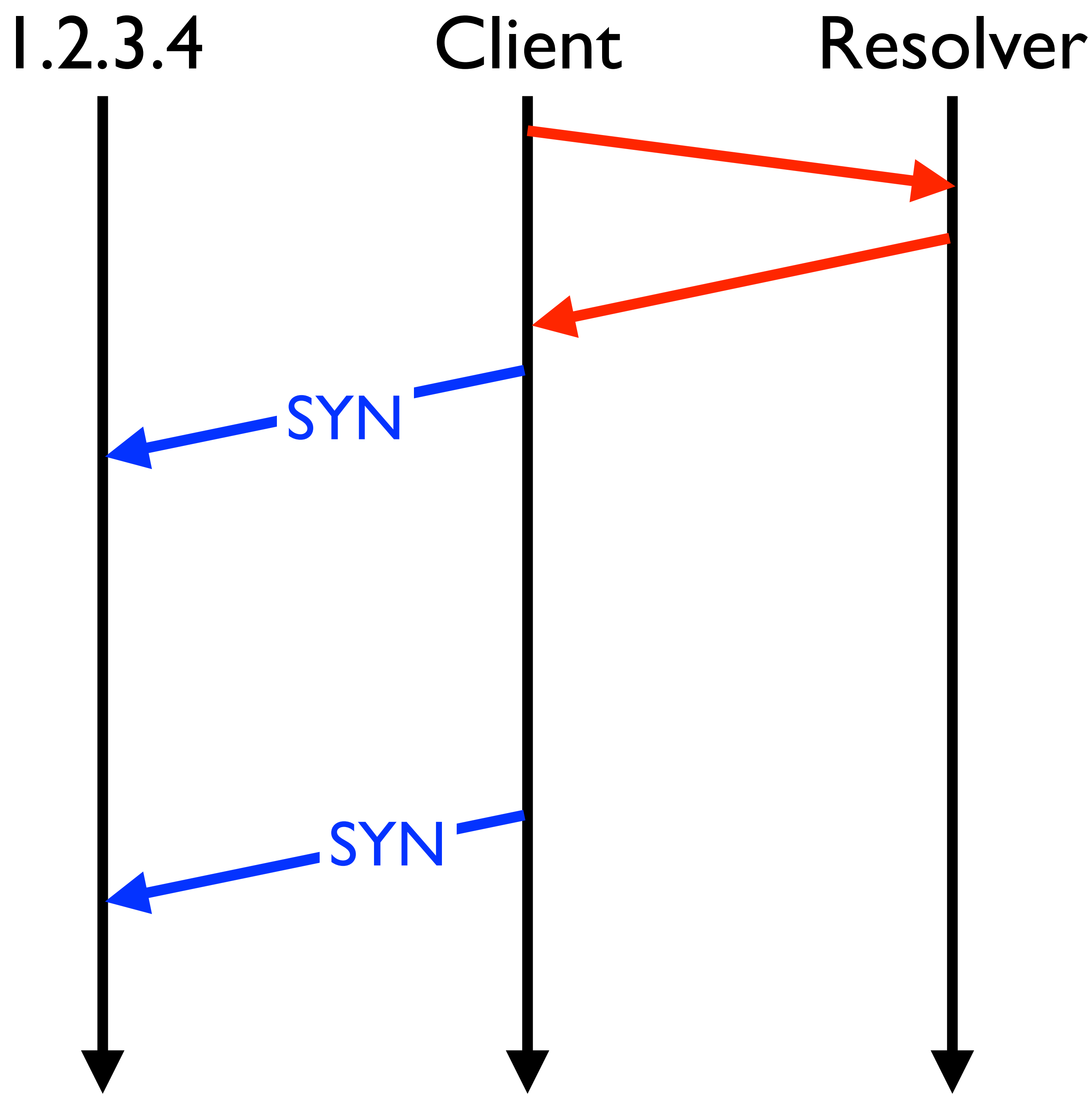
- 7.8% of conns. (875K conns.)
- 3.1M DNS lookups never used
- ≈22% of speculative lookups are used

Local Cache



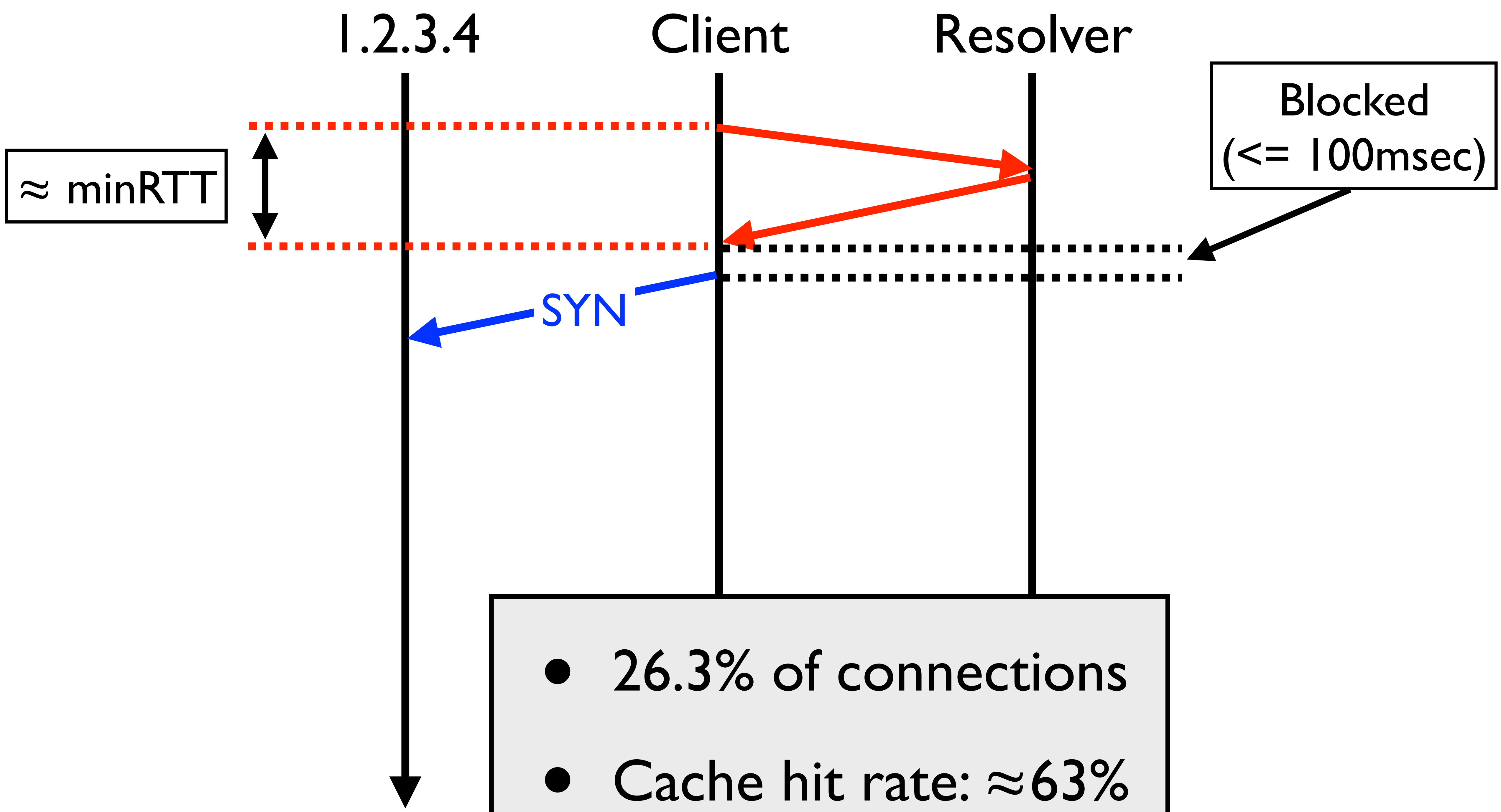
1.2.3.4
Client
Resolver
SYN
SYN
Not Blocked
(> 100msec)
Both connections pair with same DNS lookup

Local Cache



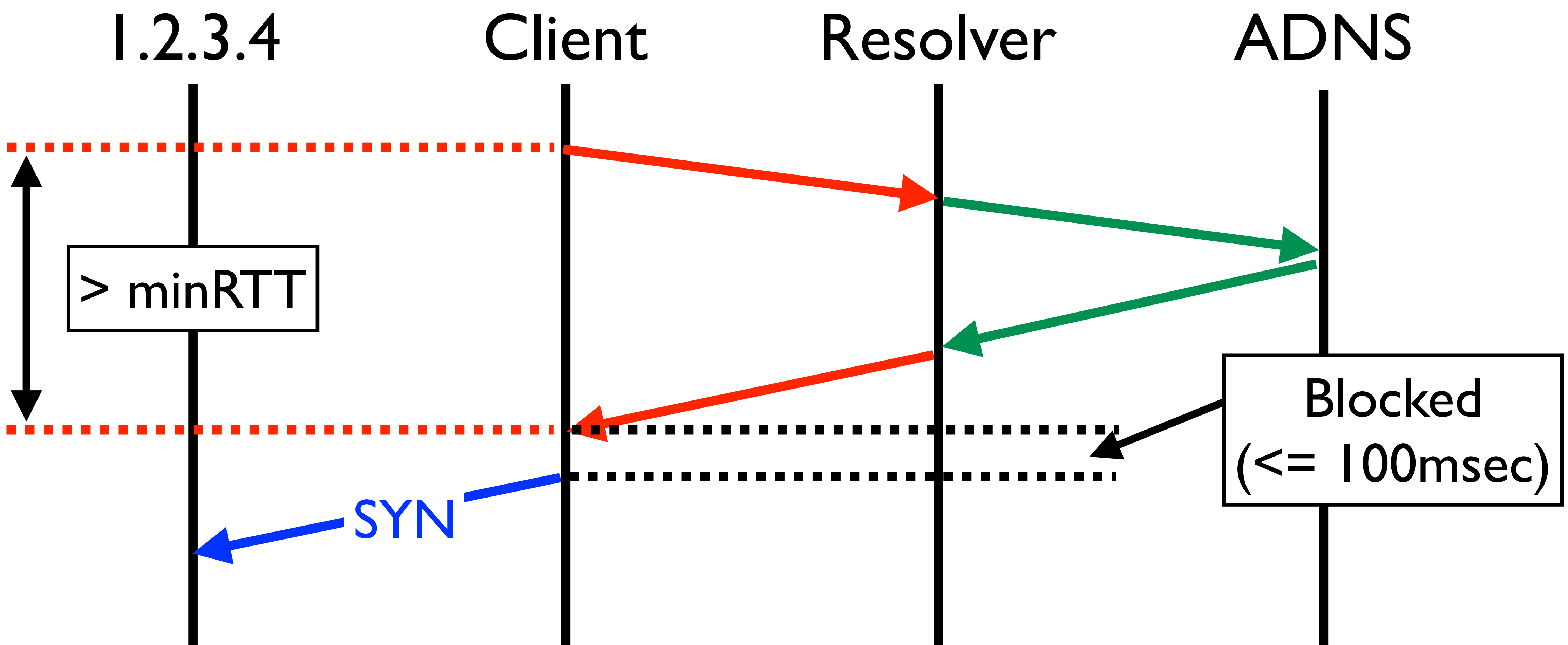
- 42.9% of conns.
- 22.2% of conns. using local cache use *expired* records!

Shared Resolver Cache



- 26.3% of connections
- Cache hit rate: ≈63%

Auth Resolution Required



- 15.7% of connections
- we cannot determine how many lookups the resolver does
- likely still benefits from caching

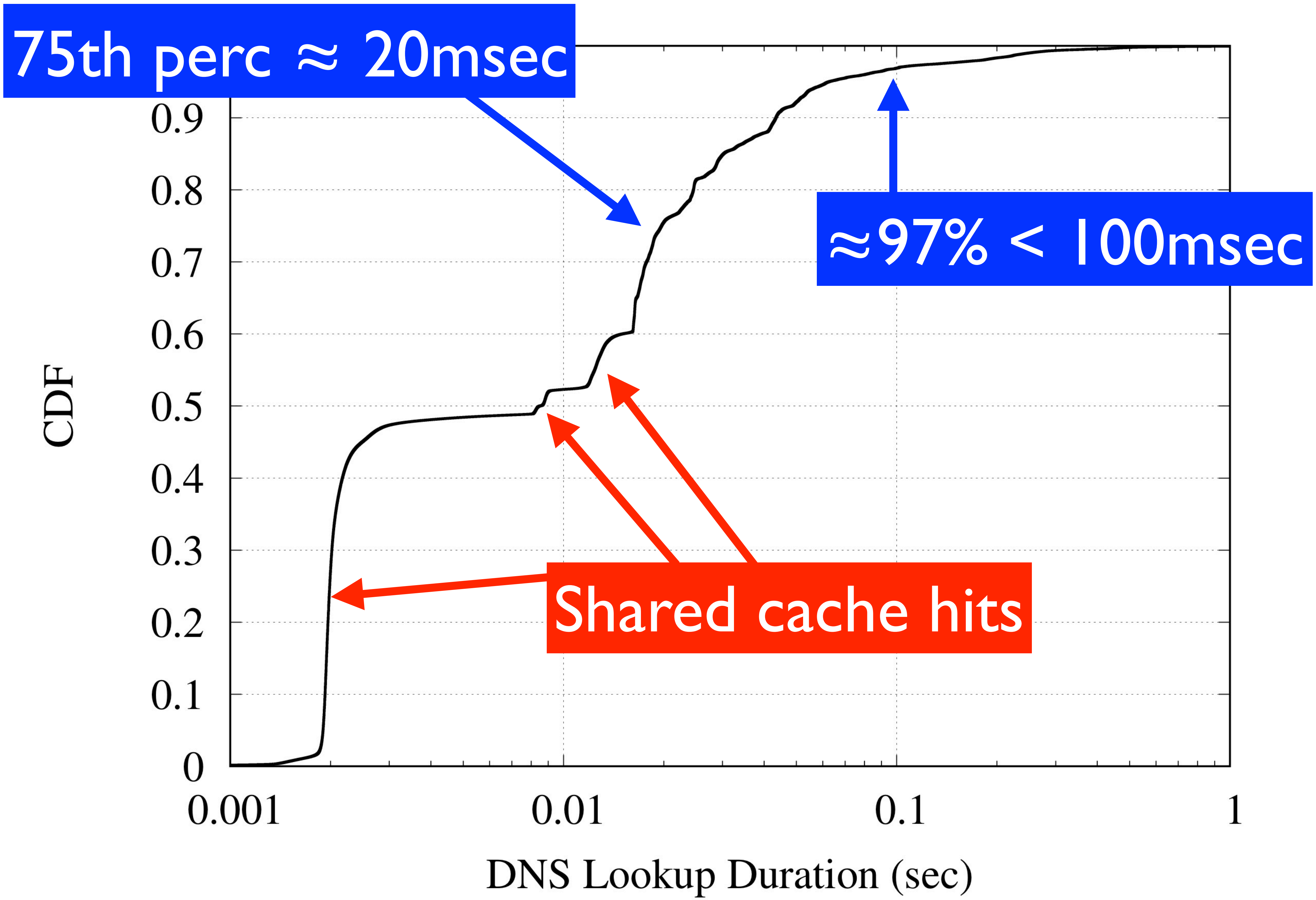
DNS Record Origin

	% Connections
No DNS	7.2
Local Cache	42.9
Prefetched	7.8
Shared Resolver Cache	26.3
Auth Resolution Required	15.7

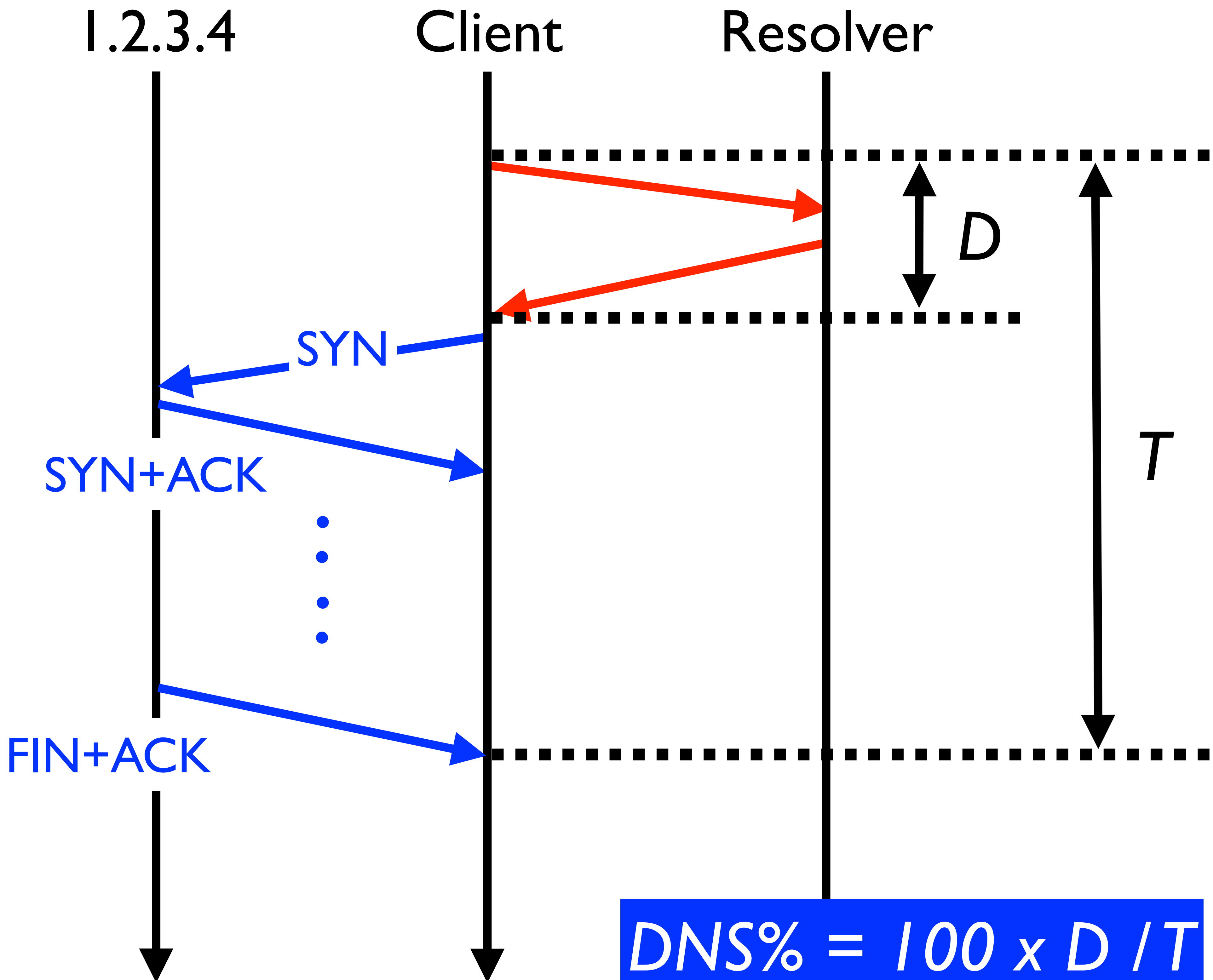
57.9% of conns. have *no direct DNS costs*

42.1% of conns. don't have needed DNS information on hand

Absolute DNS Cost

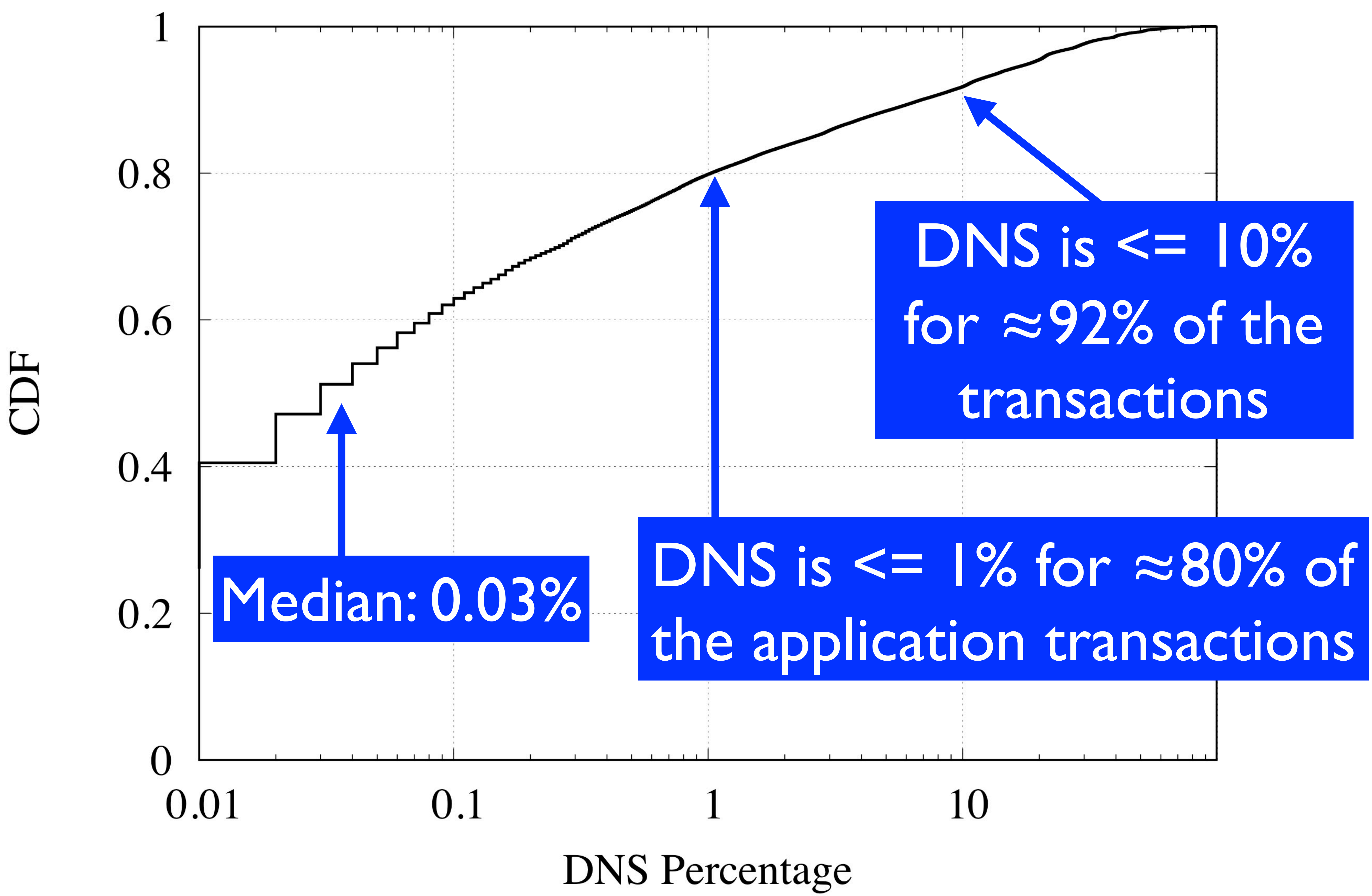


Relative DNS Cost



DNS% = 100 x D / T

Relative DNS Cost



DNS Cost

- We say application connections requiring a DNS lookup experience *"significant DNS cost"* when *both* the following criteria are met:
  - DNS lookup duration > 20msec
  - DNS percentage contribution > 1%

DNS Cost

		DNS Duration	
		<= 20msec	> 20msec
DNS Percent	<= 1%	64.0	15.9
	> 1%	11.5	8.6

64% of transactions show an *insignificant* DNS cost using *both* criteria

27% of transactions show an *insignificant* DNS cost using *one* criteria

<9% of transactions show an *significant* DNS cost using *both* criteria

Also In The Paper ...

- More methodological details
  - including ethical considerations in appendix
- Analysis as a function of resolver
  - ISP vs. Google vs. OpenDNS vs. Cloudflare
- Brief analysis of how to improve DNS further

Questions? Comments?

Mark Allman, *mallman@icir.org*
https://www.icir.org/mallman/
@mallman_icsi

EXTRA SLIDES