ASAP: A Low-Latency Transport Layer

Wenxuan Zhou, Qingxi Li, Matthew Caesar, Brighten Godfrey
Not so fast, Internet...
Not so fast, Internet...

• Fetching a popular website
Not so fast, Internet...

- Fetching a popular website
  
  time wget www.phdcomics.com
Not so fast, Internet...

- Fetching a popular website

  time wget www.phdcomics.com
  real 895ms
Not so fast, Internet...

- Fetching a popular website

```bash
time wget www.phdcomics.com
real 895ms
100%[===================================================>] 10,560  65.2K/s  in 200ms
```
Not so fast, Internet...

- Fetching a popular website

  time wget www.phdcomics.com
  real 895ms
  100%[------------------>] 10,560  65.2K/s  in 200ms

- Single packet latency

  ping 69.17.116.124  (www.phdcomics.com)
  time=76.3ms
Not so fast, Internet...

• Fetching a popular website
  time wget www.phdcomics.com
  real 895ms
  100%[================================] 10,560 65.2K/s in 200ms

• Single packet latency
  ping 69.17.116.124 (www.phdcomics.com)
  time=76.3ms

• Why is the web so slow?
What's a few hundred milliseconds?
What’s a few hundred milliseconds?

![Graph showing the effect of a few hundred milliseconds on searches per user. The graph includes a trend line and actual data points.](image)

[Jake Brutlag, Google, 2009]
What's a few hundred milliseconds?

4 searches per user

+200ms

+400ms

--- actual

— trend

-0.2%

0%

0.2%

$200 M/yr

week 1  week 2  week 3  week 4  week 5  week 6

[Jake Brutlag, Google, 2009]
Protocols cause delay

Client

Web server
Protocols cause delay
Protocols cause delay
Protocols cause delay

Client

Web server

We wish...
Protocols cause delay

Client → Local DNS → Auth. DNS → Web server
Protocols cause delay

Client

Local DNS

Auth. DNS

Web server
Protocols cause delay

Client
Local DNS
Auth. DNS
Web server
Protocols cause delay

Client
Local DNS
Auth. DNS
Web server
Protocols cause delay

Client

Local DNS

Auth. DNS

Web server
Protocols cause delay

Client
Local DNS
Auth. DNS
Web server

Name Resolution
Protocols cause delay
Protocols cause delay
Protocols cause delay
Protocols cause delay

Client
Local DNS
Auth. DNS
Web server

Name Resolution Handshake

TCP
Sources of Delay

Client
Local DNS

Auth. DNS
Web server
Sources of Delay

Client

Local DNS

Auth. DNS

Web server

Name Resolution
Sources of Delay

Client
Local DNS
Auth. DNS
Web server

Name Resolution
TCP Handshake

Route
Saturday, December 3, 2011
Sources of Delay

- Client
- Local DNS
- Auth. DNS
- Web server

Name Resolution
TCP Handshake
Waiting for First Byte
Sources of Delay

Client
Local DNS
Auth. DNS
Web server

Name Resolution
TCP Handshake
Waiting for First Byte
Data Transmission
Sources of Delay

- Client: 41.5%
- Local DNS: 20.5%
- Auth. DNS: 24.6%
- Web server: 13.4%

Fat pipe primarily helps here.

- Name Resolution: 7%
- TCP Handshake
- Waiting for First Byte
- Data Transmission
Our Design: ASAP
Goals:
Our Design: ASAP

Goals:

- Reduce latency
Goals:

- Reduce latency
- Retain security and deployability
Goals:

- Reduce latency
- Retain security and deployability

Key idea 1: Piggyback connection setup on name resolution
Our Design: ASAP

Goals:

- Reduce latency
- Retain security and deployability

Key idea 1: Piggyback connection setup on name resolution

- Add connection information into name lookup message.
Our Design: ASAP

Goals:

- Reduce latency
- Retain security and deployability

**Key idea 1:** Piggyback connection setup on name resolution
  - Add connection information into name lookup message.

**Key idea 2:** Bypass handshaking
Our Design: ASAP

Goals:

- Reduce latency
- Retain security and deployability

Key idea 1: Piggyback connection setup on name resolution

- Add connection information into name lookup message.

Key idea 2: Bypass handshaking

- Servers respond to the request with data directly.
Our Design: ASAP

Client → Local DNS → Auth. DNS → Web server
Our Design: ASAP

Client

Local DNS

Auth. DNS

Web server
Our Design: ASAP

Client ➔ Local DNS ➔ Auth. DNS ➔ Web server

CI: connection information
Our Design: ASAP

Address of “Cl.x.com”?

Client  
Local DNS  
Auth. DNS  
Web server

Cl: connection information
Our Design: ASAP

Client

Local DNS

Auth. DNS

Web server

Address of “Cl.x.com”?

Source IP, Source Port, Dest IP, Dest Port, Sequence #,...
HTTP GET index.html

CI: connection information

Saturday, December 3, 2011
Our Design: ASAP

Client

Local DNS

Auth. DNS

Web server

Cl: connection information
Our Design: ASAP

Client → Local DNS → Auth. DNS → Web server

CI: connection information
Our Design: ASAP

Client -> Local DNS -> Auth. DNS -> Web server

Address of “Cl.x.com”?

Cl: connection information
Our Design: ASAP

- Client
- Local DNS
- Auth. DNS
- Web server

**Cl**: connection information
Our Design: ASAP

CI: connection information
Our Design: ASAP

Client

Local DNS

Auth. DNS

Web server

Cl: connection information
Our Design: ASAP

Client

Local DNS

Auth. DNS

Web server

CI: connection information
Our Design: ASAP

Cl: connection information
Our Design: ASAP

Cl: connection information
Our Design: ASAP

Client ─── Local DNS ─── Auth. DNS ─── Web server

Data

CI: connection information
Our Design: ASAP

Client

Local DNS

Auth. DNS

Web server

**CI: connection information**
Our Design: ASAP

Cl: connection information
ASAP Improvement

Client
Local DNS
Auth. DNS
Web server
ASAP Improvement

Client

Local DNS

Auth. DNS

Web server
ASAP Improvement

Current Internet

Client

Local DNS

Auth. DNS

Web server
Time we save:
ASAP Improvement

**Time we save:**

- **Client**
  - Local DNS
  - Auth. DNS
  - Web server

**ASAP**

- **Current**
- **Internet**

**Time we save:**

Saturday, December 3, 2011
ASAP Improvement

Time we save:
Problem: Breaks DNS caching
Problem: Breaks DNS caching

Solution: Client also sends a standard DNS query
Challenges and Solutions

Problem: Breaks DNS caching

Solution: Client also sends a standard DNS query

Problem: All queries traverse a single Auth. DNS
Challenges and Solutions

Problem: Breaks DNS caching

Solution: Client also sends a standard DNS query

Problem: All queries traverse a single Auth. DNS

Solution: Another layer of Auth. DNS at servers
Challenges and Solutions

Problem: Breaks DNS caching

Solution: Client also sends a standard DNS query

Problem: All queries traverse a single Auth. DNS

Solution: Another layer of Auth. DNS at servers

Problem: Eliminating 3-way handshake makes DoS attacks easier
Challenges and Solutions

**Problem:** Breaks DNS caching

- **Solution:** Client also sends a standard DNS query

**Problem:** All queries traverse a single Auth. DNS

- **Solution:** Another layer of Auth. DNS at servers

**Problem:** Eliminating 3-way handshake makes DoS attacks easier

- **Solution:** Hand reusable certificates to clients
Challenges and Solutions

Problem: Breaks DNS caching

• Solution: Client also sends a standard DNS query

Problem: All queries traverse a single Auth. DNS

• Solution: Another layer of Auth. DNS at servers

Problem: Eliminating 3-way handshake makes DoS attacks easier

• Solution: Hand reusable certificates to clients
Security Problem: DoS Attack

- Reflection/Amplification DoS attack

Diagram:
- Attacker
- Victim
- Web server

Saturday, December 3, 2011
Security Problem: DoS Attack

**Reflection/Amplification DoS attack**

I’m the Victim, I want this data.

Attacker  →  Web server
Victim    →  Web server
Victim    →  Web server
Victim    →  Web server
Victim    →  Web server
Security Problem: DoS Attack

Reflection/Amplification DoS attack

I’m the Victim, I want this data.

Attacker

Victim

Data

Web server

Web server

Web server

Web server
**Goal:** Servers verify client’s address without adding an RTT
Security Mechanism

**Goal:** Servers verify client’s address without adding an RTT

**Solution:** Cryptographic proofs
**Goal:** Servers verify client’s address without adding an RTT

**Solution:** Cryptographic proofs
**Goal:** Servers verify client’s address without adding an RTT

**Solution:** Cryptographic proofs

**Choices of PV:**
- Web server
- CDN
- Trusted 3rd party
- …
Goal: Servers verify client’s address without adding an RTT

Solution: Cryptographic proofs

Choices of PV:
- Web server
- CDN
- Trusted 3rd party
- …
Goal: Servers verify client’s address without adding an RTT

Solution: Cryptographic proofs

Choices of PV:
- Web server
- CDN
- Trusted 3rd party
- …
Goal: Servers verify client’s address without adding an RTT

Solution: Cryptographic proofs

Choices of PV:
- Web server
- CDN
- Trusted 3rd party
- …
Goal: Servers verify client’s address without adding an RTT

Solution: Cryptographic proofs

Choices of PV:
- Web server
- CDN
- Trusted 3rd party
- ...
**Goal:** Servers verify client’s address without adding an RTT

**Solution:** Cryptographic proofs

**Choices of PV:**
- Web server
- CDN
- Trusted 3rd party
- …
Problem: Eavesdropping
Problem: Eavesdropping
Problem: Eavesdropping
Problem: Eavesdropping
Problem: Eavesdropping
Problem: Eavesdropping

Solution: Multiple PVs
Problem: Eavesdropping

Solution: Multiple PVs
Problem: Eavesdropping

Solution: Multiple PVs
Problem: Eavesdropping

Solution: Multiple PVs
Problem: Eavesdropping

Solution: Multiple PVs
Problem: Eavesdropping

Solution: Multiple PVs
Results: Eavesdropping Defense

**Attackability**

- 1 PV ASAP: 100%
- 2 PVs ASAP: 20%
- 3 PVs ASAP: 0%
- TCP: 19%

Saturday, December 3, 2011
Results: Eavesdropping Defense

**Attackability**

- **One PV** is vulnerable to the attacker
- **Two PVs** reduce the attacker’s effectiveness dramatically

<table>
<thead>
<tr>
<th>1 PV ASAP</th>
<th>2 PVs ASAP</th>
<th>3 PVs ASAP</th>
<th>TCP</th>
</tr>
</thead>
<tbody>
<tr>
<td>100%</td>
<td>0%</td>
<td>0%</td>
<td></td>
</tr>
</tbody>
</table>

Saturday, December 3, 2011
Deployability

ASAP requires changes to some parts of the Internet

But only to devices under the client’s and server’s control

- Auth. DNS, End host clients, Server

This simplifies deployability
Experimental Results

ASAP has both benefits and costs
ASAP has both **benefits** and **costs**

- Improved latency, at the cost of some additional processing overhead
ASAP has both **benefits** and **costs**

- Improved latency, at the cost of some additional processing overhead

Metrics
ASAP has both **benefits** and **costs**

- Improved latency, at the cost of some additional processing overhead

**Metrics**

- Reduced latency
Experimental Results

ASAP has both **benefits** and **costs**

- Improved latency, at the cost of some additional processing overhead

**Metrics**

- Reduced latency
- Processing overhead
ASAP has both benefits and costs

- Improved latency, at the cost of some additional processing overhead

Metrics

- Reduced latency
- Processing overhead

Implementation
ASAP has both **benefits** and **costs**

- Improved latency, at the cost of some additional processing overhead

**Metrics**

- Reduced latency
- Processing overhead

**Implementation**

- Linux Kernel
ASAP has both **benefits** and **costs**

- Improved latency, at the cost of some additional processing overhead

**Metrics**

- Reduced latency
- Processing overhead

**Implementation**

- Linux Kernel
- User-space on planet-lab
Results: Latency of a Download

Round Trip Time

Fraction of trials

Time (seconds)

Saturday, December 3, 2011
Results: Latency of a Download

Round Trip Time

ASAP
Results: Latency of a Download

Round Trip Time

- ASAP
- TCP

fraction of trials

time (seconds)
ASAP provides a significant reduction in latency over TCP.
Results: Computational Overhead

![Graph showing computational overhead with time (ms) on the x-axis and fraction of trials on the y-axis. The graph includes curves for Server, PV, and Client.]
- Client, PV, and server overheads are low
- A single PV server could handle tens of millions of clients
### Related Work: TCP Fast Open

<table>
<thead>
<tr>
<th>Similarities</th>
<th>ASAP</th>
<th>TCP Fast Open</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bypass three way handshake</td>
<td>Use a security proof verifying address ownership</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Differences</th>
<th>ASAP</th>
<th>TCP Fast Open</th>
</tr>
</thead>
<tbody>
<tr>
<td>Piggyback connection setup on name resolution</td>
<td>Name resolution not involved</td>
<td></td>
</tr>
<tr>
<td>Require changes to Auth. DNS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Certificates for security proof (can be used across domains)</td>
<td>Cookies for security proof (easier to compute)</td>
<td></td>
</tr>
</tbody>
</table>
ASAP merges functions of DNS and TCP, and eliminates 3WH to reduce latency of interactive requests
ASAP merges functions of DNS and TCP, and eliminates 3WH to reduce latency of interactive requests

- Reduces latency by up to 2 RTTs
ASAP merges functions of DNS and TCP, and eliminates 3WH to reduce latency of interactive requests

- Reduces latency by up to 2 RTTs
- Retains protection against attacks
Conclusions

ASAP merges functions of DNS and TCP, and eliminates 3WH to reduce latency of interactive requests

- Reduces latency by up to 2 RTTs
- Retains protection against attacks

Implementation:
Conclusions

ASAP merges functions of DNS and TCP, and eliminates 3WH to reduce latency of interactive requests

- Reduces latency by up to 2 RTTs
- Retains protection against attacks

Implementation:

- User-space and Kernel-space
Conclusions

ASAP merges functions of DNS and TCP, and eliminates 3WH to reduce latency of interactive requests

- Reduces latency by up to 2 RTTs
- Retains protection against attacks

Implementation:

- User-space and Kernel-space

Available at: [http://www.cs.illinois.edu/~wzhou10/asap.tar.gz](http://www.cs.illinois.edu/~wzhou10/asap.tar.gz)
Backup: Security Mechanism

Client

Web Server

Provenance Verifier (PV)
Backup: Security Mechanism

1. \( \{K_{\text{pub}}, d_c\} \)
Backup: Security Mechanism

1. \( \{K^c_{pub}, d_c\} \)

2. \( PC=\{K^c_{pub}, a_c, t, d\} \ K^{pv}_{priv} \)

Client → Provenance Verifier (PV) → Web Server

Saturday, December 3, 2011
Backup: Security Mechanism

1. \( \{K_{\text{pub}}^c, d_c\} \)

2. \( \text{PC}=\{K_{\text{pub}}^c, a_c, t, d\} K_{\text{priv}}^{pv} \)

3. meta, PC, RC, data
   
   where
   
   \( \text{RC}=\{\text{hash(meta, data)}, t_{\text{req}}\} K_{\text{priv}}^c \)

Client → Provenance Verifier (PV)

Web Server