Oblique: Accelerating Page Loads Using Symbolic Execution

nsdi '21

James Mickens
Harvard University
Why I’m Interested in Load Time Optimization

THIS HAPPENED TO ME LAST NIGHT

Internet speed test

Megabits per second

Testing upload...

2.34 Mbps download

Sheraton Palo Alto Hotel

See photos

See outside
Outline

● How A Browser Loads a Web Page
● Impediments to Optimization
● Oblique
● What I Learned While Working On Oblique
How Does A Browser Load a Page?

User phone

GET http://foo.com

Web server

<html>
<head>
  <title>Hi!</title>
  <link rel="stylesheet" href="style.css">
  <script src="foo.js"/>
</head>
<body>
  <div>
    <p>Some content</p>
    <img src="cats.jpg"/>
    <img src="dogs.jpg"/>
  </div>
</body>
</html>
How Does A Browser Load a Page?

- The HTML contains tags that reference external objects.
- To fully load the page, the browser must fetch all of these external objects!

User phone

Web server
How Does A Browser Load a Page?

- The HTML contains tags that reference external objects.
- To fully load the page, the browser must fetch all of these external objects!

HTTP GET to http://foo.com/c.css

User phone

Web server
How Does A Browser Load a Page?

User phone

GET http://foo.com/first.js

Web server

HTML

<css>
<script>
<img>
<script>

Web server

JS

User phone

GET http://foo.com

Web server
How Does A Browser Load a Page?

```javascript
if (cookie.loggedln) {
    fetch('http://foo.com/' + cookie.userId + '.jpg');
} else {
    fetch('http://foo.com/defaultAvatar.jpg');
}
```

User phone

Web server
How Does A Browser Load a Page?

```javascript
if(cookie.loggedIN){
    fetch(`http://foo.com/` + cookie.userID + `.jpg`);
} else {
    fetch(`http://foo.com/defaultAvatar.jpg`);
}
```

GET http://foo.com/defaultAvatar.jpg
How Does A Browser Load a Page?

Universe 1
- Some objects to fetch are **statically known**, given the HTML
- The if/else JS statement created two different **universes** because of **dynamic content resolution**
- Branches may check **personalized** state like a **cookie**!
- Some dynamically-selected content may be **personalized**!

Universe 2
A single page load may be associated with many potential universes for each user!

```javascript
if(cookie.loggedln)
    fetch("http://foo.com/" + cookie.userid + "_.jpg");
else
    fetch("http://foo.com/defaultAvatar.jpg");

if((userAgent == "Chrome Mobile") ||
    ( userAgent == "Safari iOS") ){
    fetch("http://foo.com/Android.js");
    fetch(...);
} else{
    fetch("http://foo.com/default.js");
    fetch(...);
}
```
A single page load may be associated with many potential universes for each user!

```
if((userAgent == "Chrome Mobile") || (userAgent == "Safari iOS")){
    fetch("http://foo.com/Android.js");
    fetch(...);
} else{
    fetch("http://foo.com/default.js");
    fetch(...);
}
```
Outline

- How A Browser Loads a Web Page
- Impediments to Optimization
- Oblique
- What I Learned While Working On Oblique
Two Problematic Trends in Web Traffic

- Mobile traffic is growing (> 50%)
  - Many mobile users (particularly in emerging markets) stuck behind high-latency 3G/4G links
  - Even 5G links often suffer from 4G latencies
  - Latency, not bandwidth, often determines page load times!

- Traffic is shifting to HTTPS (> 90%)
  - The crypto is cheap . . .
  - . . . but how can we accelerate encrypted mobile traffic while preserving confidentiality and integrity?

Remote Dependency Resolution (e.g., Amazon Silk, Parcel)
Two Problematic Trends in Web Traffic

✅ Enables outsourcing of web acceleration

❌ Breaks end-to-end TLS security: cleartext user data (e.g., cookies and User-Agent string) are exposed to third party

Remote Dependency Resolution (e.g., Amazon Silk, Parcel)
Two Problematic Trends in Web Traffic

✔ Doesn't expose cleartext TLS data to third-party origins

✘ Analysis must be run by the first party: outsourcing would break TLS security

First-party offline analysis server

Loads page multiple times, identifies the stable set URLs

Server uses <link> prefetch and H2 push to prewarm client cache

User phone

First-party web server

Vroom (SIGCOMM 2017)
Outline

● How A Browser Loads a Web Page
● Impediments to Optimization
● Oblique
● What I Learned While Working On Oblique
Oblique: The Big Idea™

- An offline third-party server loads a web page symbolically
  - The symbols are sensitive user values like cookies and User-Agent strings
  - Output of analysis is a list of symbolic URLs (e.g., \{{{cookie[“userId”]}}}.html) fetched by each universe

- Have the user's browser:
  - Concretize symbolic client state (thereby picking a universe)
  - Concretize the symbolic URLs
  - Prefetch the symbolic URLs

- User-specific data is never revealed to the third party!
Output of Symbolic Page Load: A Path Constraint Tree

UserAgent

Other browser types

== "Chrome Mobile"

cookie[ "darkMode" ]

== ( "no" || "" )

== "yes"

light-mode.css
gui.js
default.html

dark-mode.css
gui.js

{{cookie[ "userId" ]}}.html
Oblique: End-to-end Workflow

1. Developer uploads page content to Oblique’s third-party analysis server
2. Oblique returns a path constraint tree for the page
3. Developer uploads page content + path constraint tree to first-party web servers
4. Later, user fetches the page’s HTML + path constraint tree
5. Oblique’s JavaScript library concretizes path constraint tree, prefetches objects
teach me this symbolic analysis of which thee speak
Symbolic Analysis

1. Distributor generates initial concrete values for client symbols (e.g., Cookie="cat=yes", User-Agent="MobileChrome")
2. Executor launches a web browser
3. Browser fetches concrete page HTML from first-party servers
4. Browser fetches more concrete objects
   a. CSS and images handled as normal
   b. JS evaluated using a concolic engine
Symbolic Analysis

1. Distributor generates initial concrete values for client symbols (e.g., Cookie="cat=yes", User-Agent="MobileChrome").
2. Executor launches a web browser.
4. Browser fetches more concrete objects:
   a. CSS and images handled as normal.
   b. JS evaluated using a concolic engine.

As JS executes on concrete data, Oblique tracks symbolic path constraints and symbolic URLs!

```javascript
var baseUrl = "foo.com/";
var rndId = Math.random().toString();
if(document.cookie.indexOf("cat")===0){
  fetch(baseUrl + rndId + "/cat.jpg");
} else{
  fetch(baseUrl + "/dog.jpg");
}
```

Concrete URL: foo.com/0.3274/cat.jpg

Symbolic URL: foo.com/\{\{rnd\_0\}\}/cat.jpg
Symbolic path constraint:
```javascript
document.cookie = "cat"\{w*\}
```
Symbolic Analysis

5. Once the page load finishes, the symbolic path constraint, e.g.,
   \( \text{document.cookie} = \text{“cat”}{\backslash w^*} \) is sent to executor

6. Executor asks the SMT solver to invert part of the constraint

7. Solver performs inversion, e.g.,
   \( \text{document.cookie} = (^\text{“cat”} \backslash w^*) \) and concretizes the new constraint, e.g.,
   \( \text{document.cookie} = \text{“x81b5”} \)

8. Solver returns the new test input to the distributor who inserts the input into a priority queue

   This input would explore a new universe!
Symbolic Analysis

The longer we run the symbolic analysis, the more universes we discover!

what if we don't find them all

it's ok
<table>
<thead>
<tr>
<th>Input name</th>
<th>HTTP header</th>
<th>JavaScript variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>User Agent</td>
<td>User-Agent</td>
<td>navigator.userAgent</td>
<td>The local browser type, e.g., &quot;Mozilla/5.0 (Windows; U; Win98; en-US; rv:0.9.2) Gecko/20010725 Netscape6/6.1&quot;</td>
</tr>
<tr>
<td>Platform</td>
<td>Included in User-Agent</td>
<td>navigator.platform</td>
<td>The local OS, e.g., &quot;Win64&quot;</td>
</tr>
<tr>
<td>Screen characteristics</td>
<td>N/A</td>
<td>window.screen.*</td>
<td>Information about the local display, e.g., the dimensions and pixel depth</td>
</tr>
<tr>
<td>Host</td>
<td>Host</td>
<td>location.host</td>
<td>Specifies the virtual host and port number to use</td>
</tr>
<tr>
<td>Referer</td>
<td>Referer</td>
<td>document.referrer</td>
<td>The URL of the page whose link was followed to generate a request for the current page</td>
</tr>
<tr>
<td>Origin</td>
<td>Origin</td>
<td>location.origin</td>
<td>Like Referer, but only includes the origin part of the referring URL</td>
</tr>
<tr>
<td>Last modified</td>
<td>Last-Modified (response)</td>
<td>document.lastModified</td>
<td>Set by the server to indicate the last modification date for the returned re- source</td>
</tr>
<tr>
<td>Cookie</td>
<td>Cookie (request), Set-Cookie (response)</td>
<td>document.cookie</td>
<td>A text string containing &quot;key=value&quot; pairs</td>
</tr>
</tbody>
</table>

Table 1: Symbolic inputs to a client-side page load.
Overview

1. Developer uploads page content to Oblique's third-party analysis server
2. Oblique returns a path constraint tree for the page
3. Developer uploads page content to first-party web servers
4. Later, user loads the page
5. Oblique's JavaScript library concretizes path constraint tree, prefetches objects
Overview

1. Developer uploads page content to Oblique’s third-party analysis server
2. Oblique returns a path constraint tree for the page
3. Developer uploads page content to first-party web servers
4. Later, user loads the page
5. Oblique’s JavaScript library concretizes path constraint tree, prefetches objects

```
var baseUrl = "foo.com/";
var rndId = Math.random().toString();
if(document.cookie.indexOf("cat")===0){
  fetch(baseUrl + rndId + "\cat.jpg");
}else{
  fetch(baseUrl + "\dog.jpg");
}
```

foo.com/\{\{rnd\}\}/cat.jpg

Client generates a random number on-the-fly and then prefetches the concretized URL!

```
document.cookie == "cat"(*)
```

"cat=OfCourse; id=42"

```
foo.com/dog.jpg
```

Yes

No
Client determines which URLs to fetch without having to parse the page's HTML or evaluate the page's JavaScript!

Client generates a random number on-the-fly and then prefetches the concretized URL!
LET’S BE HONEST WITH EACH OTHER
Program Analysis Techniques Somewhat Don’t Work

- Taint tracking
  - Implicit flows
  - XOR'ing something with itself should clear taint
  - More generally, some kinds of code (e.g., hash functions) should destroy taint BUT HOW DO YOU KNOW?

- Formal methods: AIN'T NOBODY GOT TIME TO WRITE BIG SYSTEMS IN A FORMAL LANGUAGE

- Symbolic analysis
  - You almost never have enough time
  - Constraint solvers can't solve all constraints
  - Black-box code can be hard to model
Limitations of Oblique

● Only certain native methods are modelled
  ○ For example, if the JavaScript string variable `s` contains symbolic data derived from User-Agent, then `String.charAt(s)`'s return value will properly capture that symbolic data
  ○ In contrast, Oblique always treats `Intl.DateTimeFormat(s)` as fully concrete, possibly hurting path coverage

● Oblique’s concolic analysis may time out, hurting path coverage

● Oblique’s concolic analysis can’t issue HTTP requests that are nonidempotent
Evaluation Setup

- A Mahimahi derivative recorded content from 200 popular pages
- Digital Ocean VM ran:
  - Oblique web server
  - Vroom server
  - RDR server
- User device was a Galaxy S10e phone running Chromium v78
  - End-to-end RTT b/w phone and Digital Ocean VM was ~47ms
  - We used `netem` to inject added latency in some experiments
Prefetch hit rate (static+dynamc URLs)

Prefetch hit rate (only dynamic URLs)
Outline

● How A Browser Loads a Web Page
● Impediments to Optimization
● Oblique
● What I Learned While Working On Oblique
Excitement about paper

This paper is amazing

EVERYTHING I LOVE TURNS TO ASH

This paper is okay

Time
Why Losing Wimbledon Hurts So Much

By Kevin Craft

July 9, 2011

Tennis, more than any other professional sport, seems to amplify the pain of losing. When a tennis player loses a match, he or she must face the pain all alone. There are no teammates to turn to, no caddy to give an enthusiastic pat on the back as the match slips from one's grasp. A losing tennis player must shake hands, wave to the crowd and sit there alone, contemplating what could have been.

In his book Strokes of Genius, an account of the epic 2008 Wimbledon final between Federer and Nadal, author Jon Wertheim quotes Toni Nadal, Rafa's uncle and coach, saying "Victory does not feel so good as losing feels bad. When you have a son, you are happy. But it's no comparison to the sadness you feel losing a son." While his metaphor may be a bit strong--losing a sporting event and losing a child are not commensurate in terms of grief--that quote underscores the truth about winning and losing in sports.

Sports can seem like a cruel joke perpetrated on the athletes that entertain us week after week. This is because no athlete can escape the paradox at the heart of sports, which is that the agony of defeat always outweighs the thrill of victory. Listen to a great athlete articulate why he or she sacrifices so much and works so hard to compete at a high level, and more often you'll hear them say that a fear of losing is the chief source of motivation. Rarely if ever will an athlete say that a joy of winning is what keeps them going, and this careful choice of words is critical to understanding the psychology of sports. Winning may bring a certain level of satisfaction, but losing inspires a visceral feeling of pain, like a sharp punch to the gut, that can stay with someone long after competition is complete. And this means that even if an athlete wins half of the time over the course of a career, he or she will end up experiencing more painful emotions than joyful ones.
In academia (and life in general), other people’s failures are often invisible to you!
Build a Supportive Community

- You are not the only one who:
  - Has papers get rejected
  - Occasionally says something incorrect during a meeting
  - Isn’t sure which career path is the best one

- A strong support group is important
  - Talk to other students (even in other departments!)
  - Make time to not do work
  - Don’t be afraid to talk to mental health professionals
Conclusion

● Prefetching helps to reduce page load times
● Prior systems generate prefetch lists by:
  ○ Breaking TLS integrity, or
  ○ Preventing third-party outsourcing of the analysis
● Oblique uses symbolic execution to eliminate the design tension
  ○ Oblique's third-party server can model user-specific data as symbols
  ○ Symbols are only resolved by clients!
● Oblique reduces page loads by up to 31%, outperforming Vroom and RDR by up to 17%