Web-QoS2: Web-browsing Quickly and of Course Safely, Too

Zhenyu Zhou and Theophilus Benson
Department of Computer Science, Duke University

Motivation

Text

- HTTPS has sky-rocket

Adopted everywhere because the increasing concern of network security and privacy.

- But blindly

All Objects are retrieved via HTTPS. HTTPS handshake can account for over *42%* of data exchanged.

- With harmful consequences

HTTPS *prevents network functions*, e.g. caches, from inspecting packets and optimizing end-user performance.

Conclusion: The user experience can be hurt seriously by adopting HTTPS everywhere. It may introduce long latency, poor performance or even loss of functionality.

Challenges

- Short loading time and low overhead.
- Security is not compromised.

Goal: Achieve quick and secure page load.

Solution

Key observations:

- Not everything needs to be encrypted.
- The data that indeed need to be encrypted may *NOT* need to be cached.
- HTTPS connections are not well utilized and may be harmful.

Idea: Use HTTP for as many objects as possible.

Classify the Web Content:

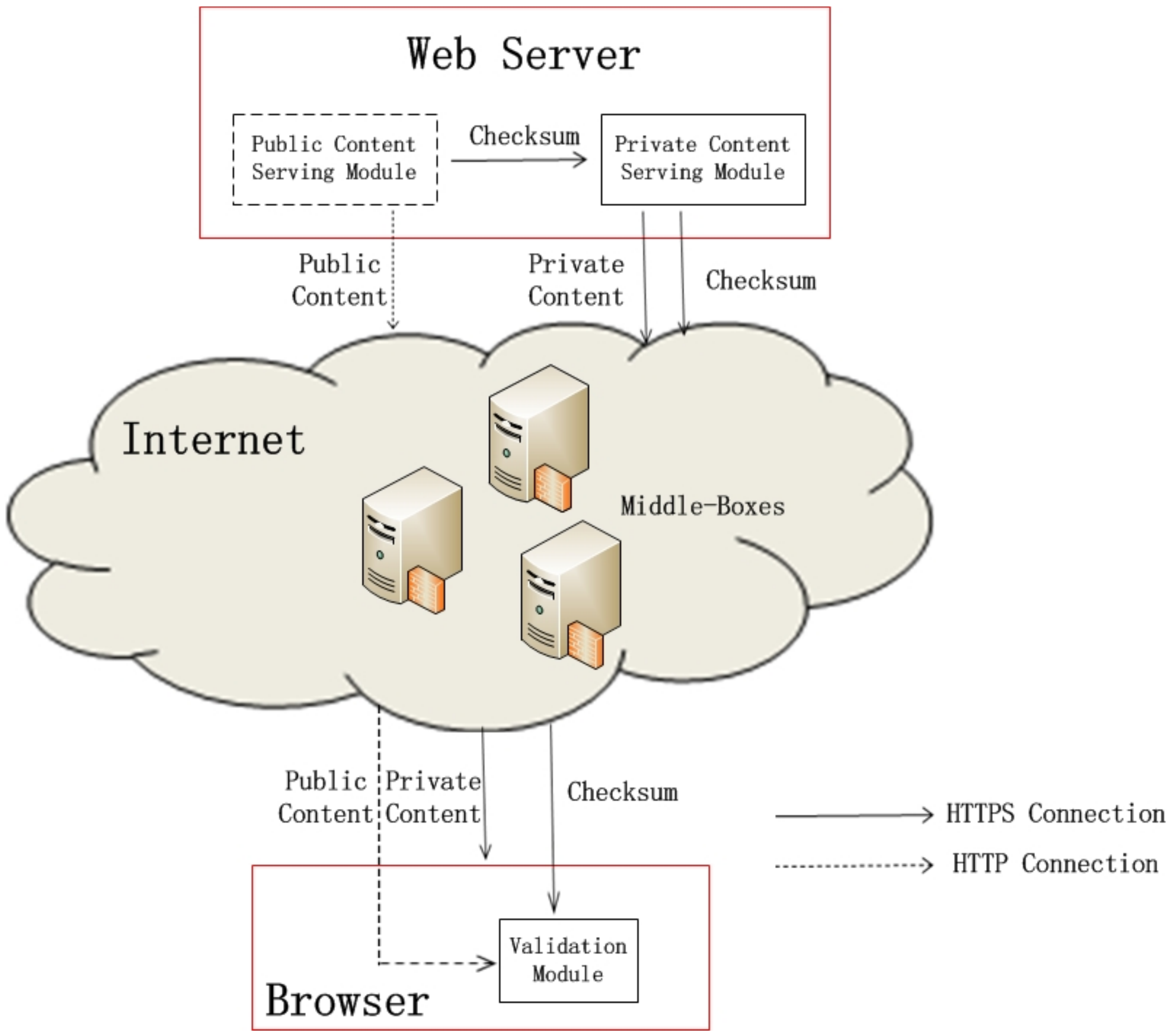
- Public content, can be sent over HTTP.
- Private content, must be sent over HTTPS.

Employ checksums to prevent tampering of data:

- Checksum prevents Man in the Middle Attacks compromising the unsecure data.
- Send checksums over HTTPS channel.

Key insight: Checksum are much smaller than data, sending checksum over HTTPS incurs minimal costs.

QoS2 Architecture



Client Side

A QoS2 enhanced browser is similar to a traditional browser except in the following way:

- Uses the checksum to verify the integrity of unencrypted data

Server Side

A QoS2 web-server is similar to a traditional web server except in the following ways:

- Tags content as either private or public

Tags determine which content is sent over HTTP or HTTPS.

- Calculates and maintains a checksum for each content that is tagged as public

Checksums enable verification of an object's integrity.

- Maintains two connections with every client

A secure connection (over HTTPS) and an unsecure one (over HTTP). The server *uses the secure connection to transfer the checksums*. This ensures that the checksums are not tampered with.

Evaluation

Performance

Experiment Setup:

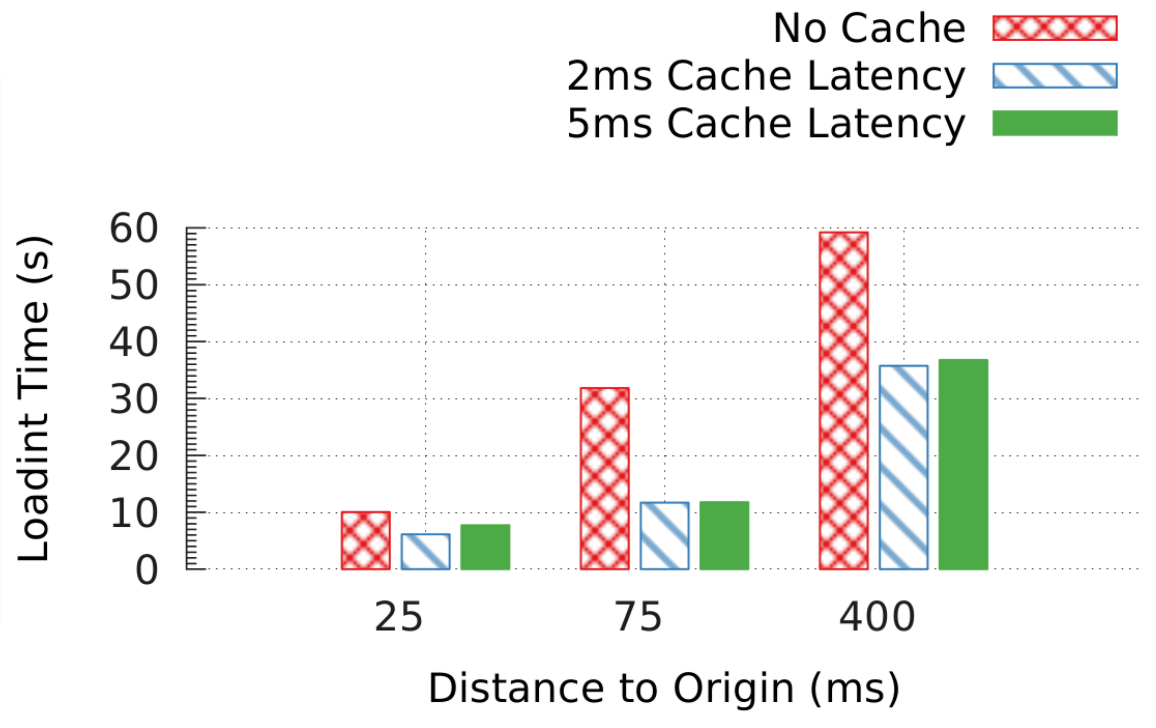
We compare the load time for varying latencies to the origin server and potential proxies.

Latencies follow distribution from Pings to Alexa Top 100 servers.

We make the following observations:

** *A 30% performance improvement* in low latency networks and *potentially as much as 70%* in high latency networks.

** Improvements are a function of both the dependencies between objects and the size of the public objects.



Analysis of page load times under QoS2