OMOS: A Framework for Secure Communication in Mashup Applications

Saman Zarandioon Danfeng (Daphne) Yao Vinod Ganapathy
Department of Computer Science
Rutgers University
Piscataway, NJ 08854
{samanz,danfeng,vinodg}@cs.rutgers.edu

December 2008

Mashups

Introduction
Mashups
Architecture
Security in client-side services
OMOS
Experiments

- What is a **Mashup** application?

Mashups

Introduction
Mashups
Architecture
Security in client-side services
OMOS
Experiments

- What is a **Mashup** application?
  - Seamlessly combine contents from multiple heterogeneous data sources.
  - Overal goal: more integrated and convenient end-user experience.
  - Becoming very popular - Web 2.0

Mashups

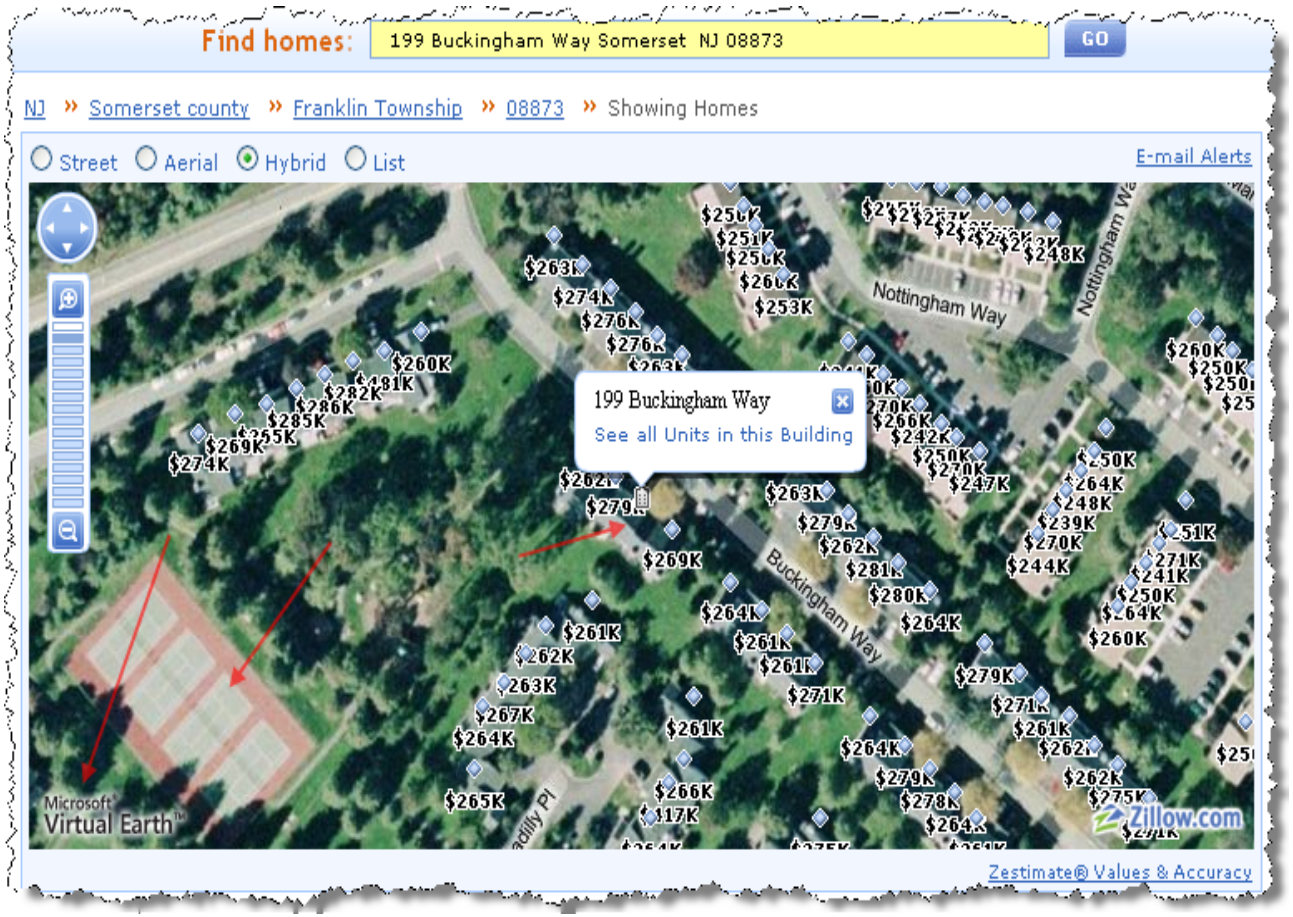
Introduction
Mashups
Architecture
Security in client-side services
OMOS
Experiments

- What is a **Mashup** application?
- My favorite mashup website *Zillow*!

Mashups

Introduction
Mashups
Architecture
Security in client-side services
OMOS
Experiments

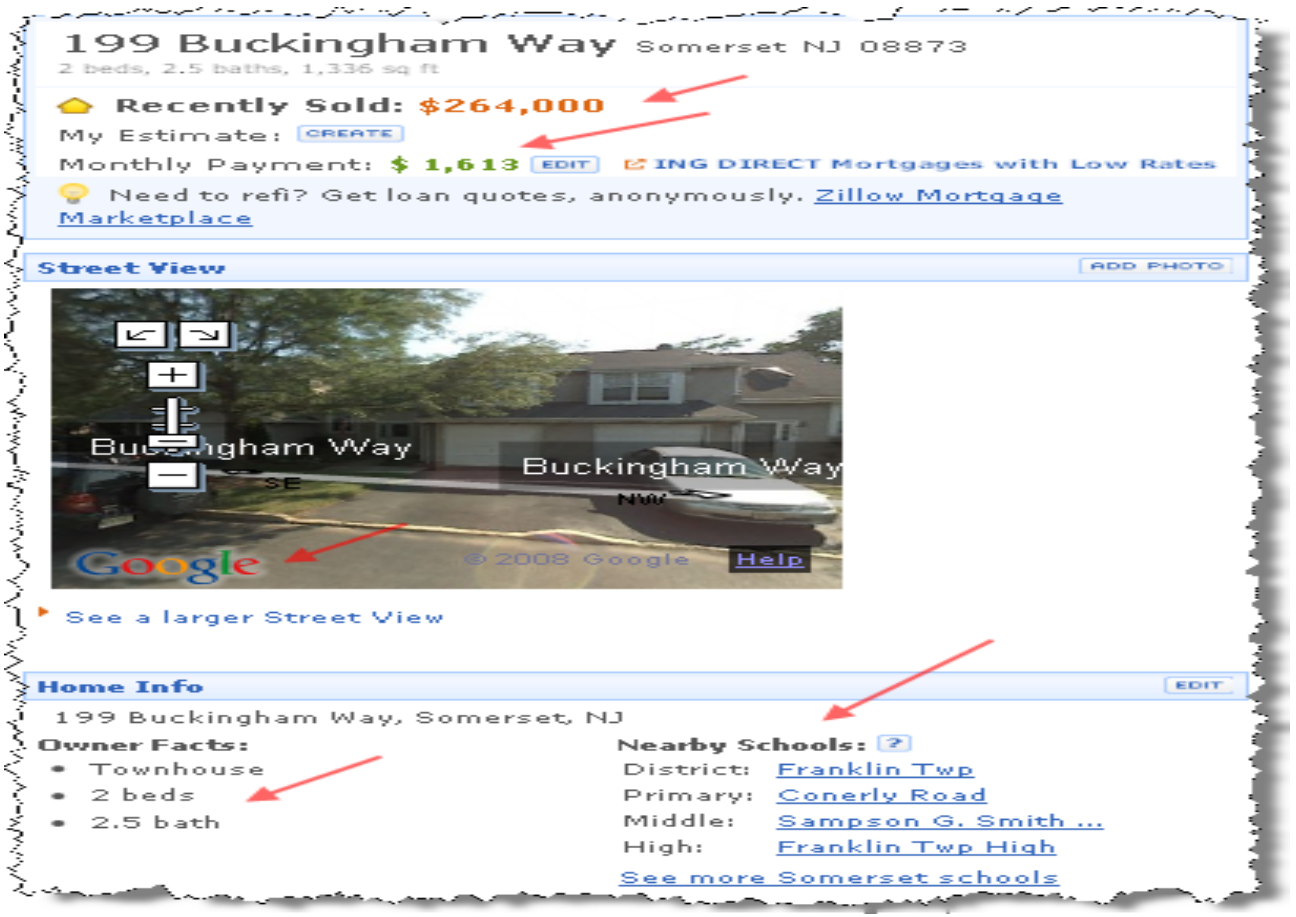
Zillow.com®
Your Edge in Real Estate



Mashups

Introduction
Mashups
Architecture
Security in client-side services
OMOS
Experiments

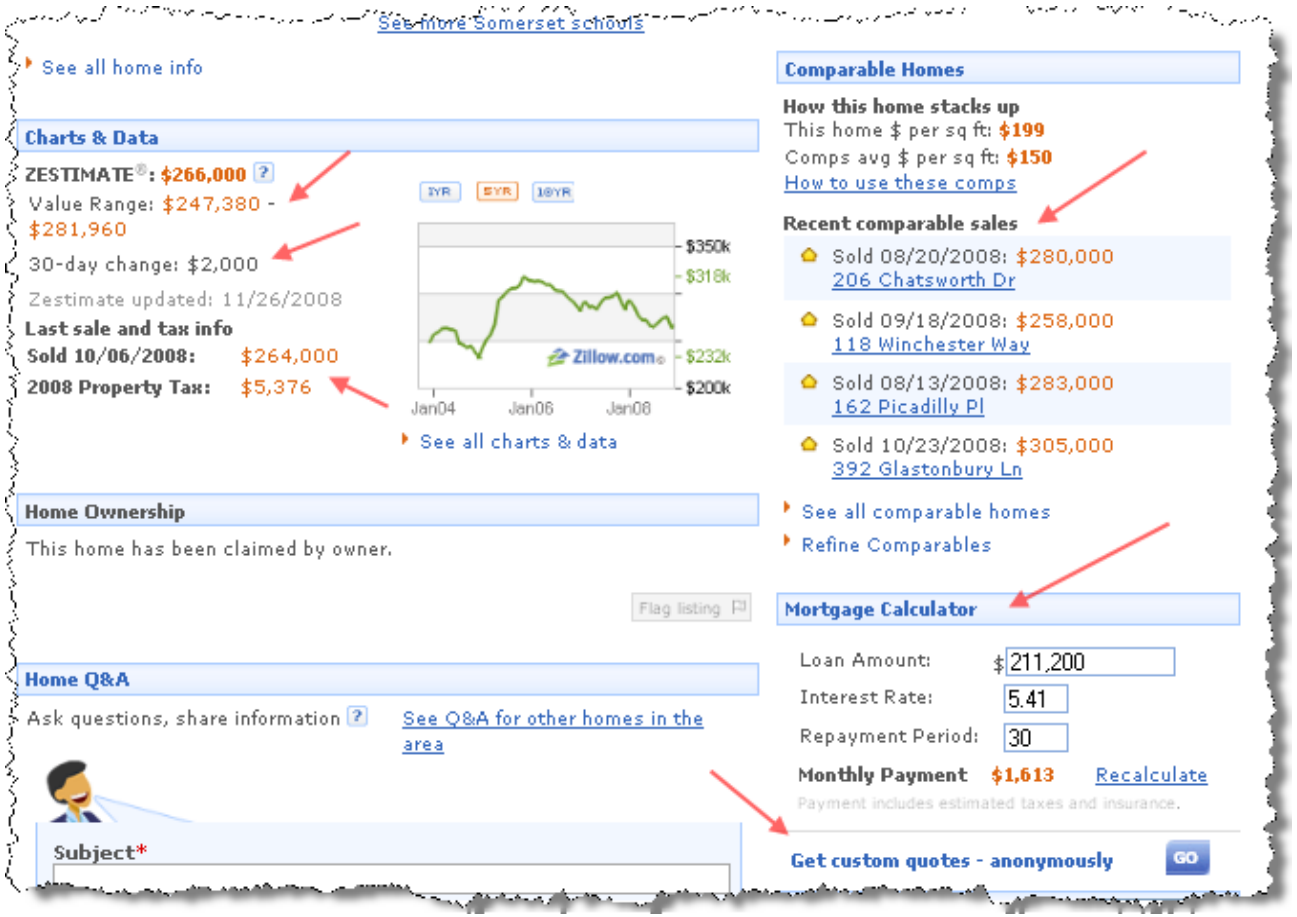
Zillow.com®
Your Edge in Real Estate



Mashups

Introduction
Mashups
Architecture
Security in client-side services
OMOS
Experiments

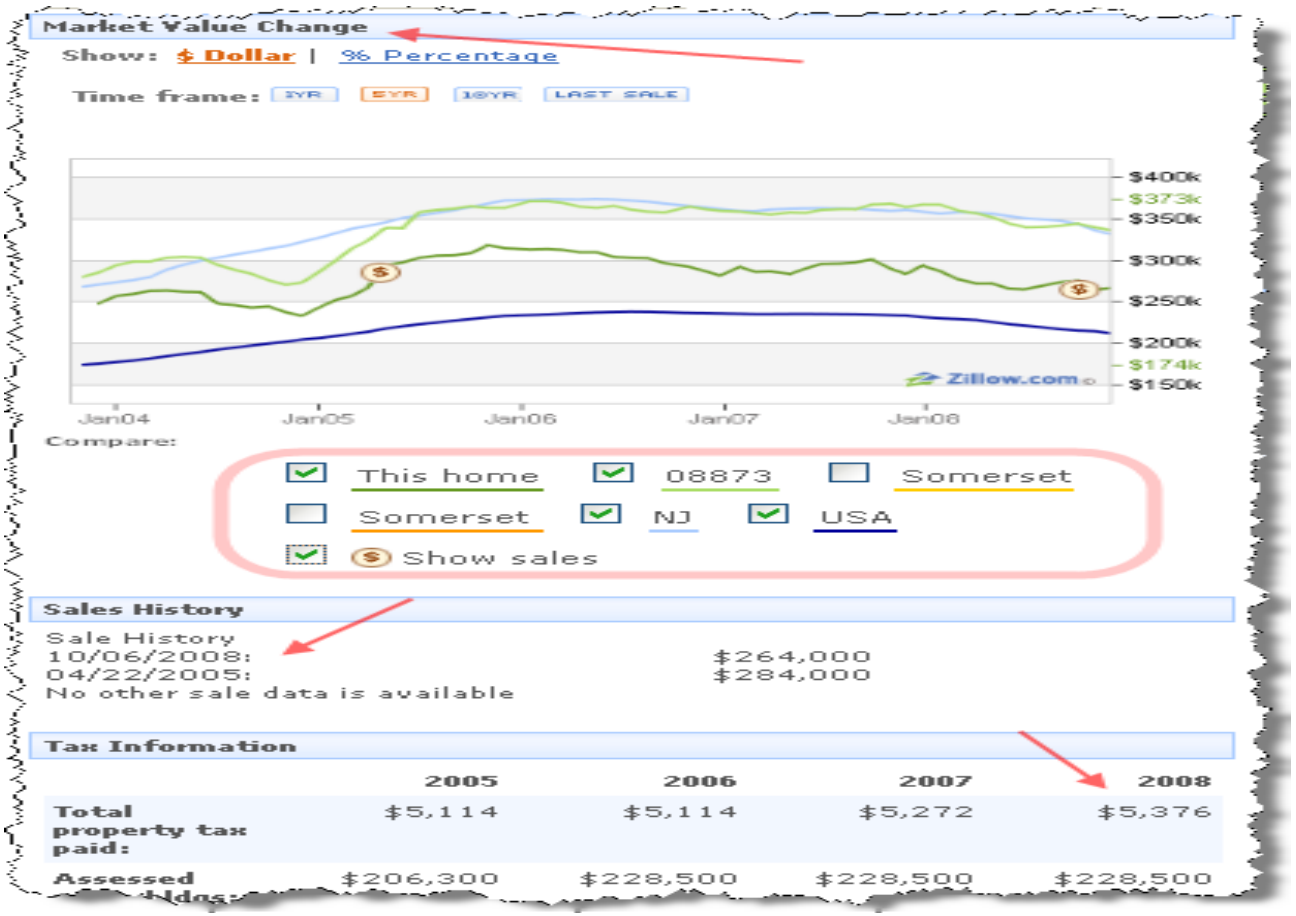
Zillow.com®
Your Edge in Real Estate



Mashups

Introduction
Mashups
Architecture
Security in client-side services
OMOS
Experiments

Zillow.com®
Your Edge in Real Estate



Mashups

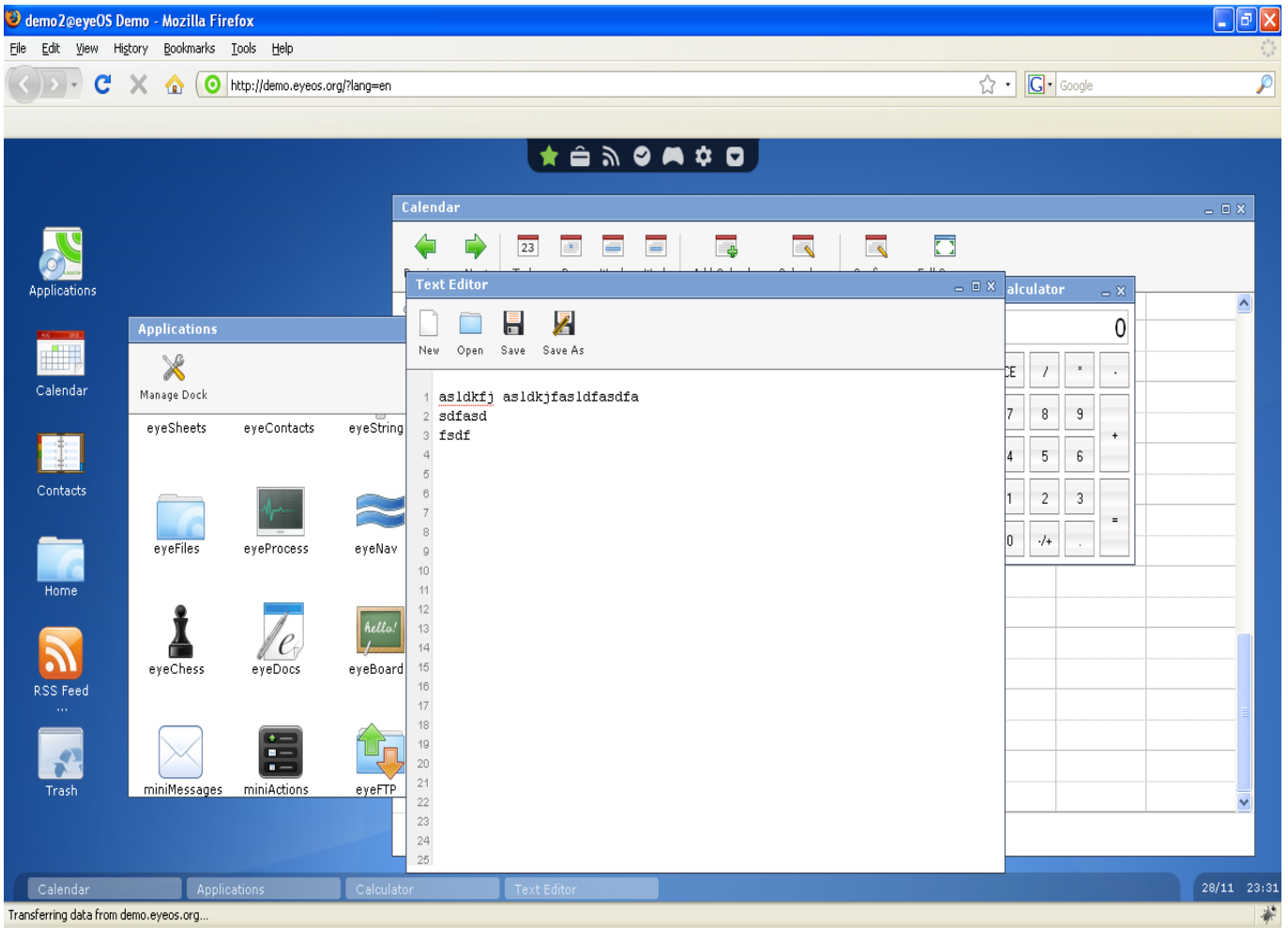
Introduction
Mashups
Architecture
Security in client-side services
OMOS
Experiments

- What is a **Mashup** application?
- My favorite mashup website *Zillow*!
- **Web desktop** (webtop) (e.g. eyeOS, DesktopTwo, G.ho.st, Netvibes, and Online OS).

Mashups

Introduction
Mashups
Architecture
Security in client-side services
OMOS
Experiments

Open Source Platform .org



Architecture

Introduction
Mashups
Architecture
Security in client-side services
OMOS
Experiments

Ways that service providers can expose their services:

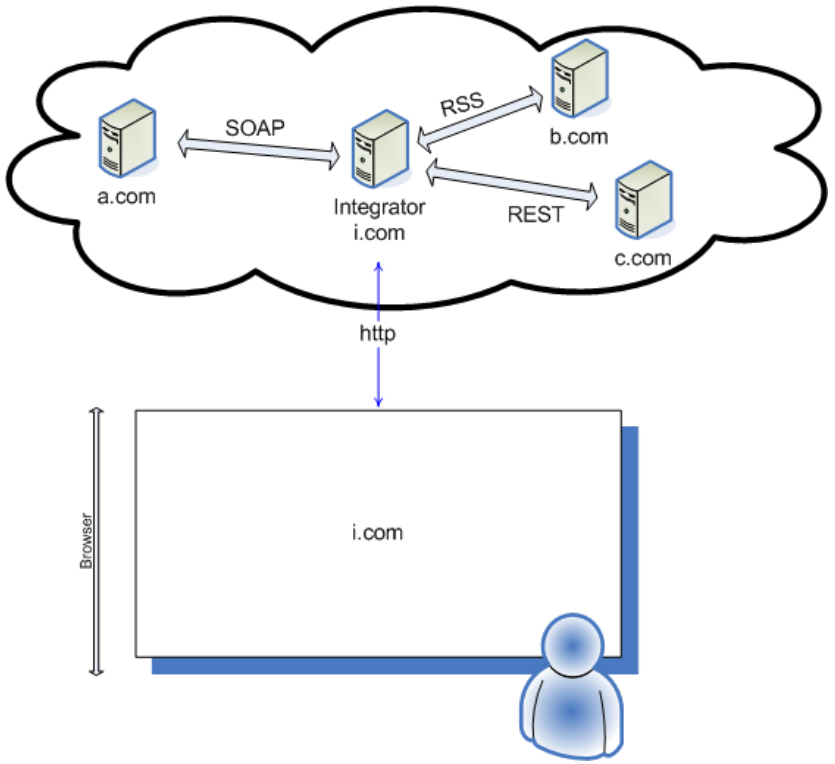
- **Server-side services**

Architecture

Introduction
Mashups
Architecture
Security in client-side services
OMOS
Experiments

Ways that service providers can expose their services:

▷ **Server-side services**



Architecture

Introduction
Mashups
Architecture
Security in client-side services
OMOS
Experiments

Ways that service providers can expose their services:

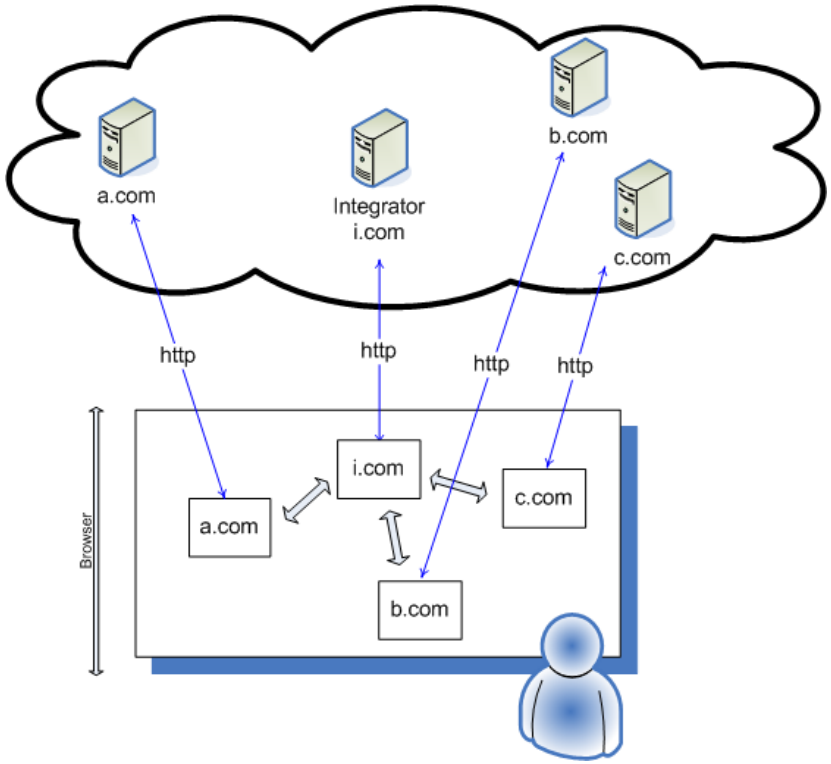
- **Server-side services**
- **Client-side services**

Architecture

Introduction
Mashups
Architecture
Security in client-side services
OMOS
Experiments

Ways that service providers can expose their services:

- ▷ **Server-side services**
- ▷ **Client-side services**



User is involved; AJAX-oriented; More responsive/efficient

Security in client-side services

Introduction
Mashups
Architecture
Security in client-side services
OMOS
Experiments

- Service providers use **ad-hoc non-secure** methods.

Security in client-side services

Introduction
Mashups
Architecture
Security in client-side services
OMOS
Experiments

- Service providers use **ad-hoc non-secure** methods.
- Consumers need to *trust* service providers: Not suitable when dealing with sensitive personal data.

Security in client-side services

Introduction
Mashups
Architecture
Security in client-side services
OMOS
Experiments

- Service providers use **ad-hoc non-secure** methods.
- Consumers need to *trust* service providers: Not suitable when dealing with sensitive personal data.
- HTML, JavaScript and browsers are not designed to support client-side communication.

Security in client-side services

Introduction
Mashups
Architecture
Security in client-side services
OMOS
Experiments

- Service providers use **ad-hoc non-secure** methods.
- Consumers need to *trust* service providers: Not suitable when dealing with sensitive personal data.
- HTML, JavaScript and browsers are not designed to support client-side communication.
- Trade-Off Between Usability and Security: All or Nothing, Complete isolation vs. complete exposure.

Security in client-side services

Introduction
Mashups
Architecture
Security in client-side services
OMOS
Experiments

- Service providers use **ad-hoc non-secure** methods.
- Consumers need to *trust* service providers: Not suitable when dealing with sensitive personal data.
- HTML, JavaScript and browsers are not designed to support client-side communication.
- Trade-Off Between Usability and Security: All or Nothing, Complete isolation vs. complete exposure.

Overview

Introduction
OMOS
Overview
Mashlet
Secure Frame-to-frame Communication
Communication Stack
MDP Layer
MHTTP Layer
Experiments

- **OpenMashupOS (OMOS)** is a mashup framework that is designed to support secure client-side services.

Overview

Introduction
OMOS
Overview
Mashlet
Secure
Frame-to-frame
Communication
Communication
Stack
MDP Layer
MHTTP Layer
Experiments

- **OpenMashupOS (OMOS)** is a mashup framework that is designed to support secure client-side services.
- **Design Goals:**

Overview

Introduction
OMOS
Overview
Mashlet
Secure Frame-to-frame Communication
Communication Stack
MDP Layer
MHTTP Layer
Experiments

- **OpenMashupOS (OMOS)** is a mashup framework that is designed to support secure client-side services.
- **Design Goals:**
  - To be compatible with all major browsers without *any* change or extension to the browsers.

Overview

Introduction
OMOS
Overview
Mashlet
Secure
Frame-to-frame
Communication
Communication
Stack
MDP Layer
MHTTP Layer
Experiments

- **OpenMashupOS (OMOS)** is a mashup framework that is designed to support secure client-side services.
- **Design Goals:**
  - To be compatible with all major browsers without *any* change or extension to the browsers.
  - To provide a *powerful* abstraction that is *flexible* and *easy* to **understand** and **use** by mashup developers.

Overview

Introduction
OMOS
Overview
Mashlet
Secure Frame-to-frame Communication
Communication Stack
MDP Layer
MHTTP Layer
Experiments

- **OpenMashupOS (OMOS)** is a mashup framework that is designed to support secure client-side services.
- **Design Goals:**
  - To be compatible with all major browsers without *any* change or extension to the browsers.
  - To provide a *powerful* abstraction that is *flexible* and *easy* to **understand** and **use** by mashup developers.
  - To guarantee *mutual authentication*, *data confidentiality*, and *message integrity* for communication between service provider and consumer.

Overview

Introduction
OMOS
Overview
Mashlet
Secure Frame-to-frame Communication
Communication Stack
MDP Layer
MHTTP Layer
Experiments

- **OpenMashupOS (OMOS)** is a mashup framework that is designed to support secure client-side services.
- **Design Goals:**
  - To be compatible with all major browsers without *any* change or extension to the browsers.
  - To provide a *powerful* abstraction that is *flexible* and *easy* to **understand** and **use** by mashup developers.
  - To guarantee *mutual authentication*, *data confidentiality*, and *message integrity* for communication between service provider and consumer.

Mashlet

Introduction
OMOS
Overview
Mashlet
Secure
Frame-to-frame Communication
Communication Stack
MDP Layer
MHTTP Layer
Experiments

- **Mashlet** is a client side component that runs in the browser under the privilege of the principal that is defined by the domain name of the server that hosts the mashlet.

Mashlet

Introduction
OMOS
Overview
Mashlet
Secure Frame-to-frame Communication
Communication Stack
MDP Layer
MHTTP Layer
Experiments

- **Mashlet** is a client side component that runs in the browser under the privilege of the principal that is defined by the domain name of the server that hosts the mashlet.
- Mashlets should be able to communicate securely on the client side, meaning that the communication protocol guarantees:

Mashlet

Introduction
OMOS
Overview
Mashlet
Secure Frame-to-frame Communication
Communication Stack
MDP Layer
MHTTP Layer
Experiments

- **Mashlet** is a client side component that runs in the browser under the privilege of the principal that is defined by the domain name of the server that hosts the mashlet.
- Mashlets should be able to communicate securely on the client side, meaning that the communication protocol guarantees:
  - **Mutual Authentication**
  - **Confidentiality**
  - **Message Integrity**

Mashlet

Introduction
OMOS
Overview
Mashlet
Secure Frame-to-frame Communication
Communication Stack
MDP Layer
MHTTP Layer
Experiments

- **Mashlet** is a client side component that runs in the browser under the privilege of the principal that is defined by the domain name of the server that hosts the mashlet.
- Mashlets should be able to communicate securely on the client side, meaning that the communication protocol guarantees:
  - **Mutual Authentication**
  - **Confidentiality**
  - **Message Integrity**

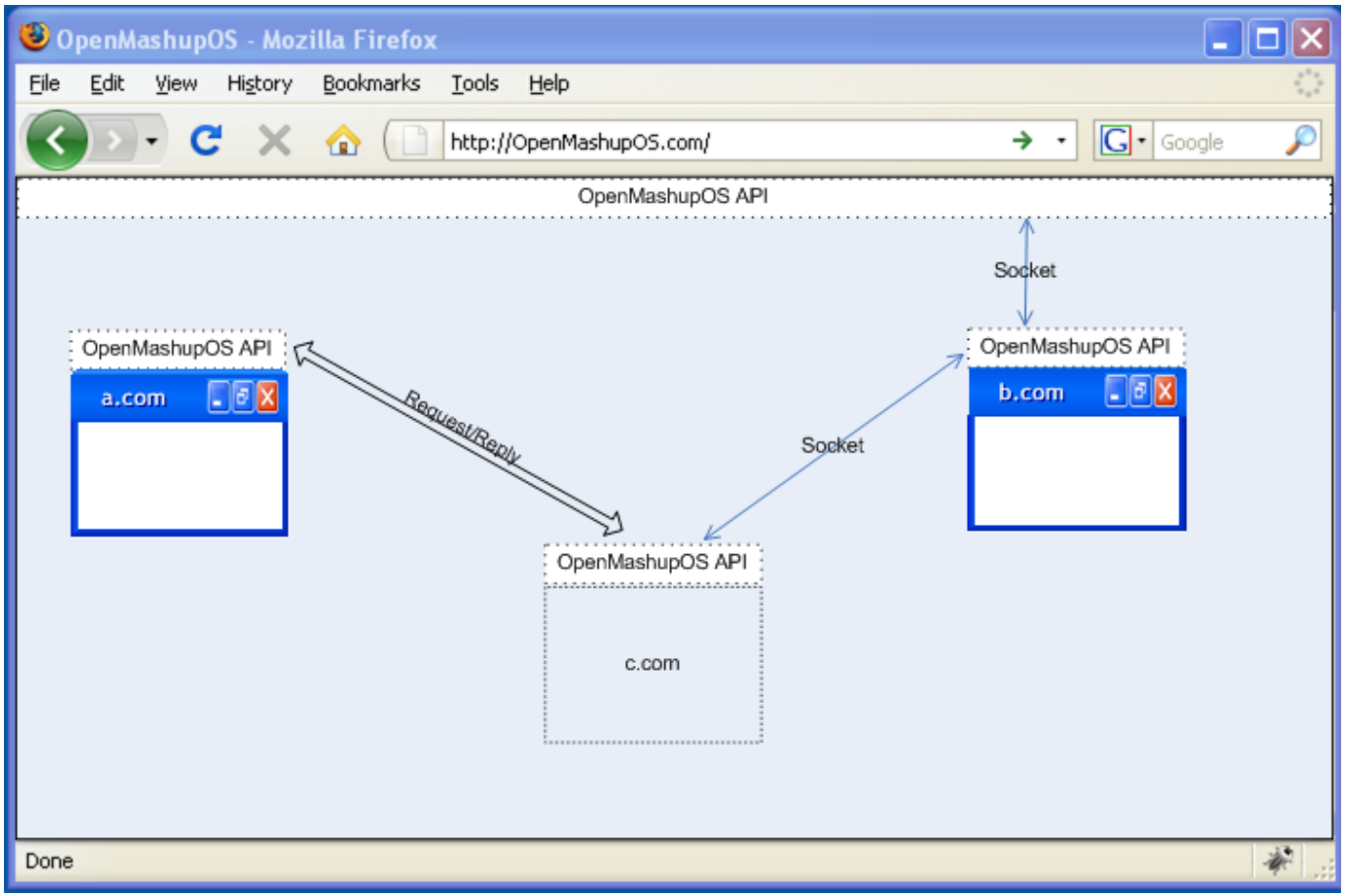
Mashlet

Introduction
OMOS
Overview
Mashlet
Secure Frame-to-frame Communication
Communication Stack
MDP Layer
MHTTP Layer
Experiments

- **Mashlet** is a client side component that runs in the browser under the privilege of the principal that is defined by the domain name of the server that hosts the mashlet.
- Mashlets should be able to communicate securely on the client side, meaning that the communication protocol guarantees:
  - **Mutual Authentication**
  - **Confidentiality**
  - **Message Integrity**

Mashlet

Introduction
OMOS
Overview
Mashlet
Secure
Frame-to-frame Communication
Communication Stack
MDP Layer
MHTTP Layer
Experiments



Using OMOS API, mashlets can communicate with their siblings and parents.

Secure Frame-to-frame Communication

Introduction
OMOS
Overview
Mashlet
Secure Frame-to-frame Communication
Communication Stack
MDP Layer
MHTTP Layer
Experiments

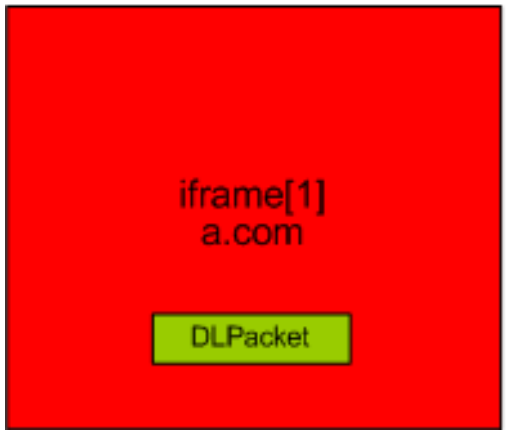
Security of OMOS communication protocol relies on *Same Origin Policy (SOP)*:

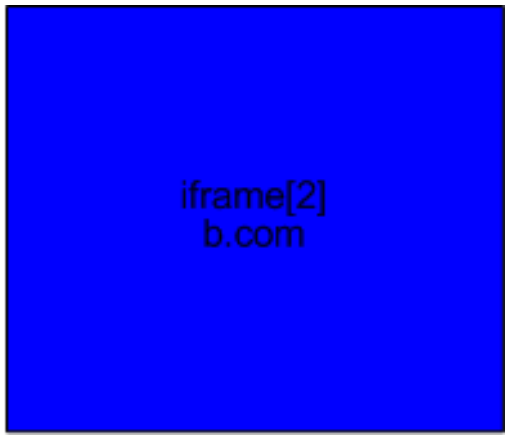
- Protects confidentiality of domains against each other. (DOM elements, events, cookies, . . .)
- URL property of an iframe is write-only.
- Partial change of URL is not allowed.

Secure Frame-to-frame Communication

Introduction
OMOS
Overview
Mashlet
Secure Frame-to-frame Communication
Communication Stack
MDP Layer
MHTTP Layer
Experiments

Destination Frame Address	Source Frame Address	End point ID	Secret Key	Sequence Number	List of data fragments

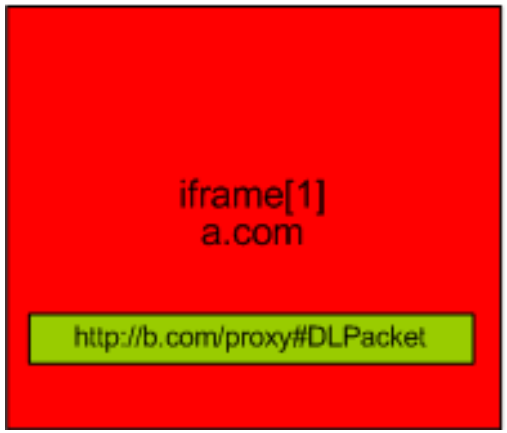


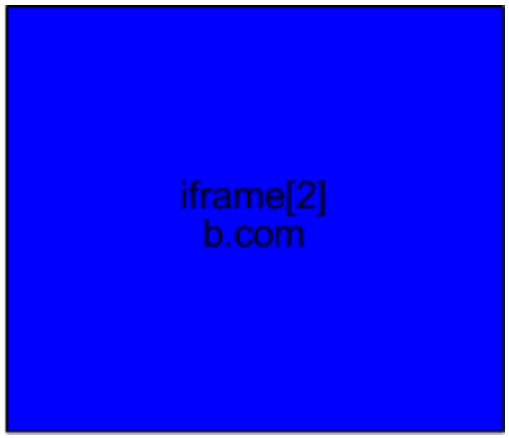


Secure Frame-to-frame Communication

Introduction
OMOS
Overview
Mashlet
Secure Frame-to-frame Communication
Communication Stack
MDP Layer
MHTTP Layer
Experiments

Destination Frame Address	Source Frame Address	End point ID	Secret Key	Sequence Number	List of data fragments

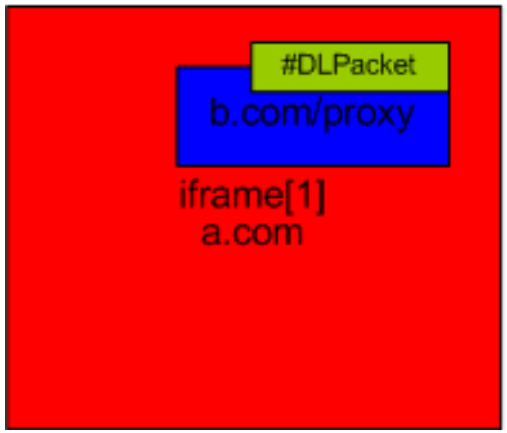


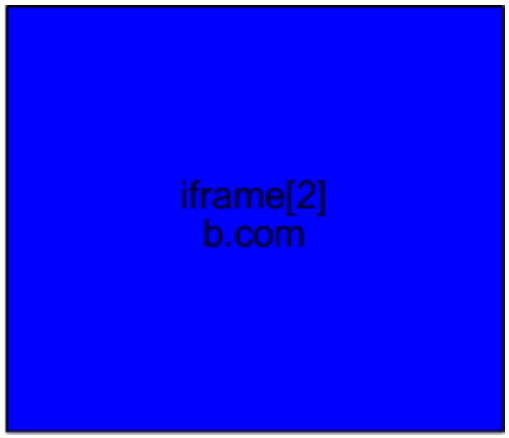


Secure Frame-to-frame Communication

Introduction
OMOS
Overview
Mashlet
Secure Frame-to-frame Communication
Communication Stack
MDP Layer
MHTTP Layer
Experiments

Destination Frame Address	Source Frame Address	End point ID	Secret Key	Sequence Number	List of data fragments

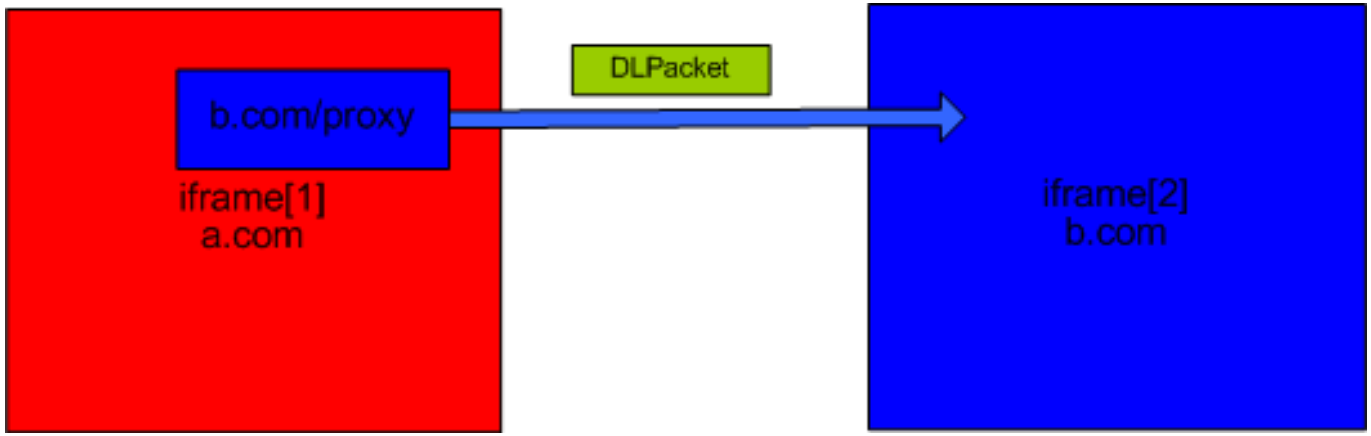




Secure Frame-to-frame Communication

Introduction
OMOS
Overview
Mashlet
Secure Frame-to-frame Communication
Communication Stack
MDP Layer
MHTTP Layer
Experiments

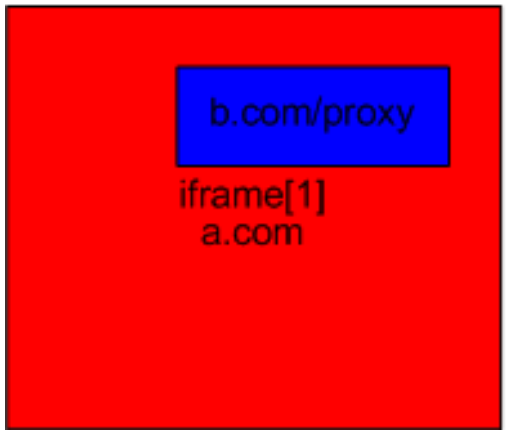
Destination Frame Address	Source Frame Address	End point ID	Secret Key	Sequence Number	List of data fragments

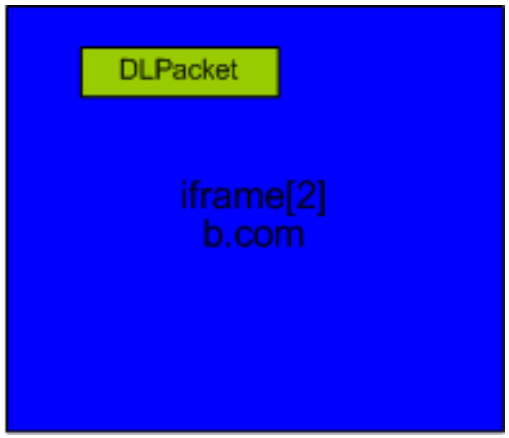


Secure Frame-to-frame Communication

Introduction
OMOS
Overview
Mashlet
Secure Frame-to-frame Communication
Communication Stack
MDP Layer
MHTTP Layer
Experiments

Destination Frame Address	Source Frame Address	End point ID	Secret Key	Sequence Number	List of data fragments

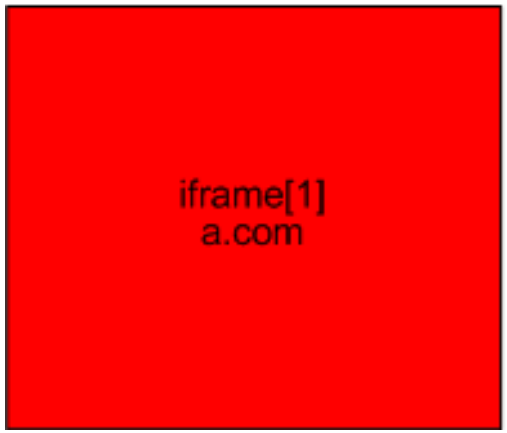


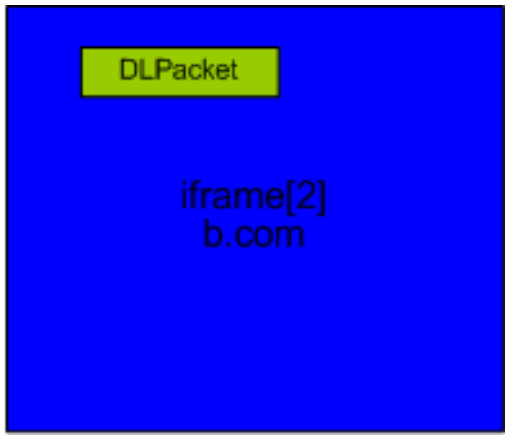


Secure Frame-to-frame Communication

Introduction
OMOS
Overview
Mashlet
Secure Frame-to-frame Communication
Communication Stack
MDP Layer
MHTTP Layer
Experiments

Destination Frame Address	Source Frame Address	End point ID	Secret Key	Sequence Number	List of data fragments

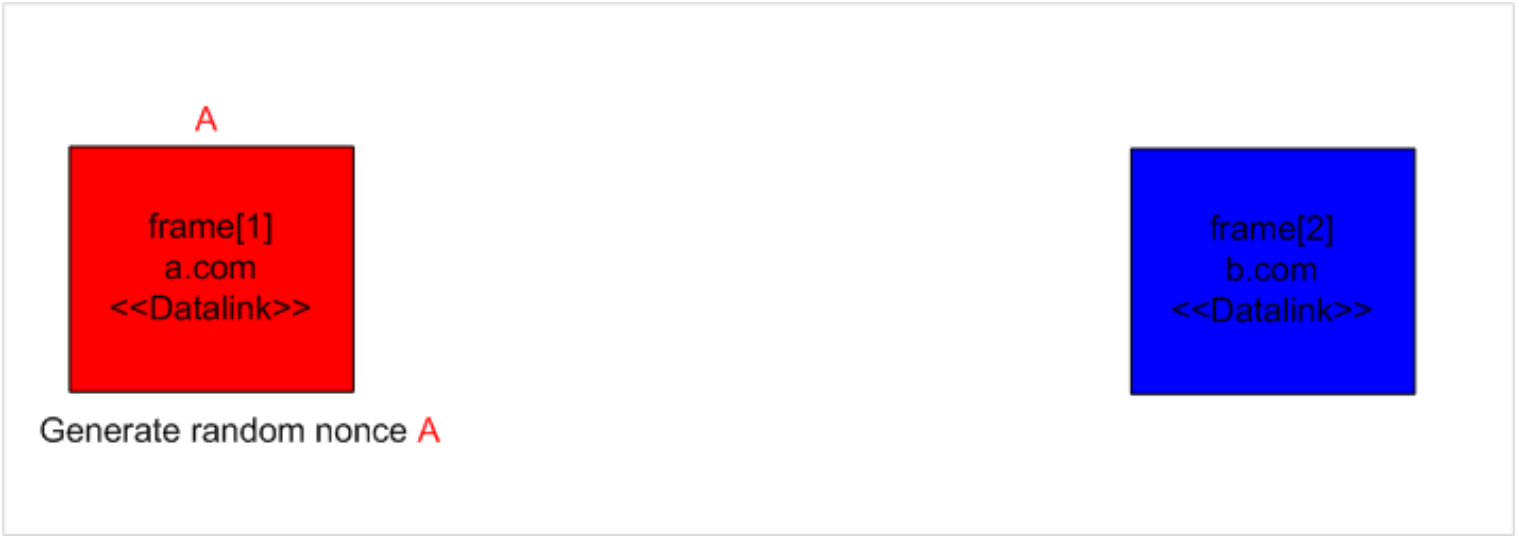




Secure Frame-to-frame Communication

Introduction
OMOS
Overview
Mashlet
Secure Frame-to-frame Communication
Communication Stack
MDP Layer
MHTTP Layer
Experiments

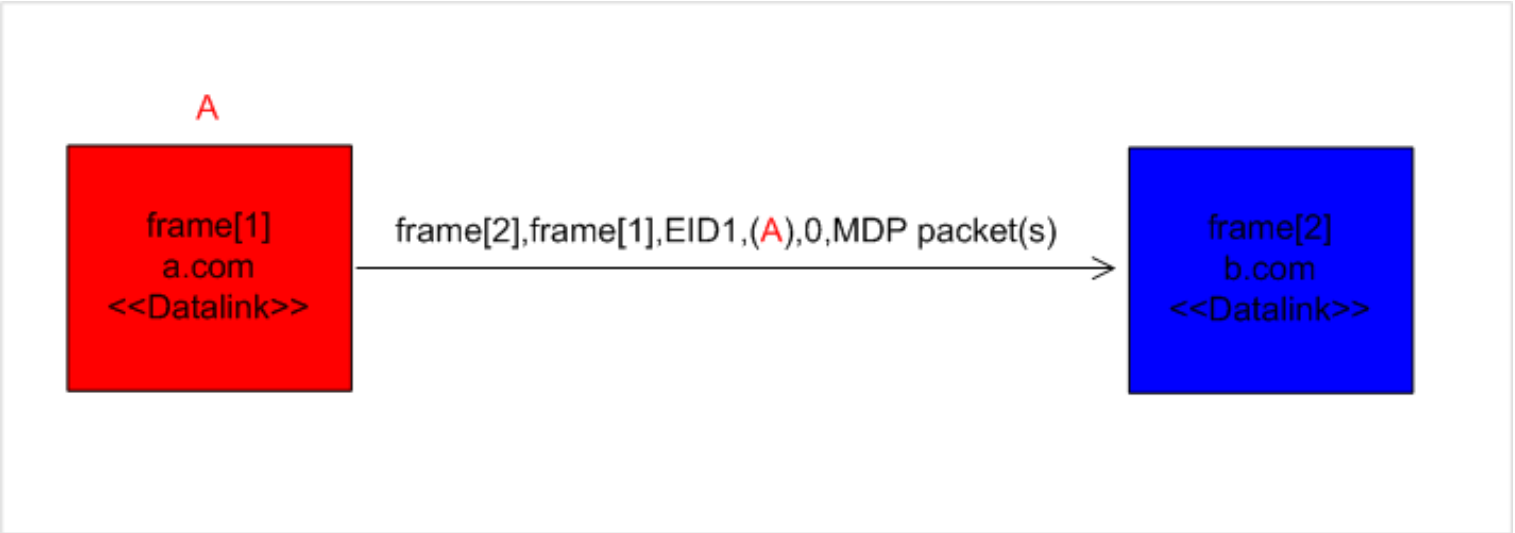
Key exchange protocol:



Secure Frame-to-frame Communication

Introduction
OMOS
Overview
Mashlet
Secure Frame-to-frame Communication
Communication Stack
MDP Layer
MHTTP Layer
Experiments

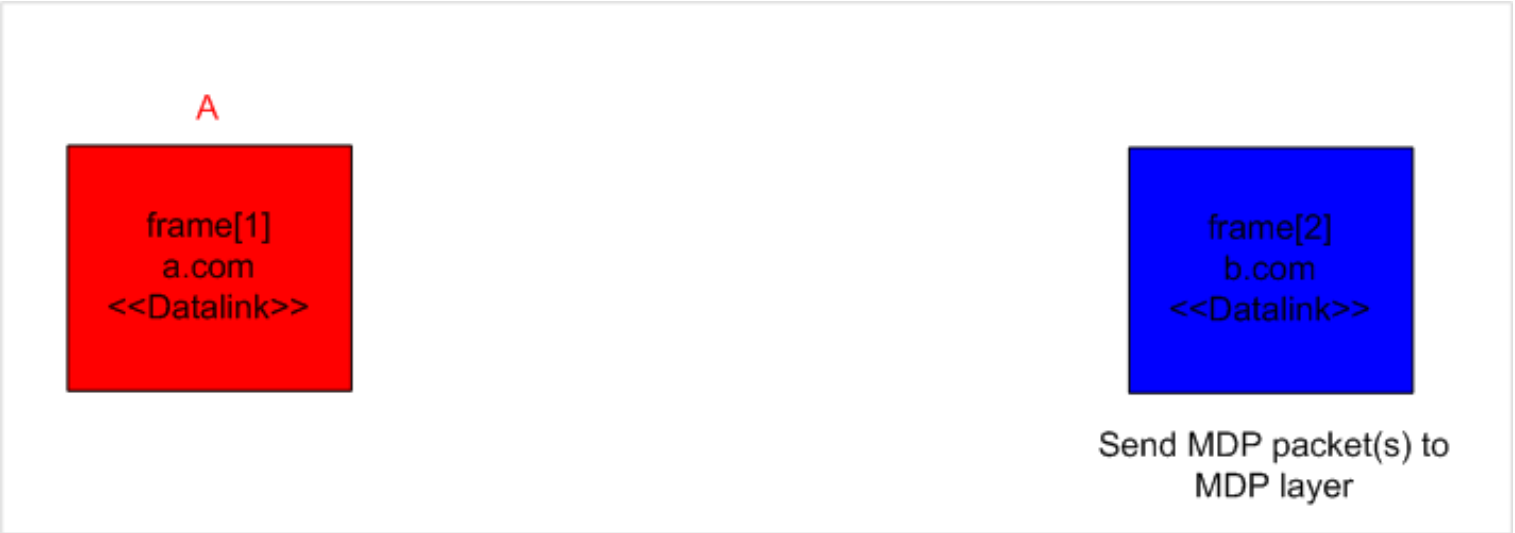
Key exchange protocol:



Secure Frame-to-frame Communication

Introduction
OMOS
Overview
Mashlet
Secure Frame-to-frame Communication
Communication Stack
MDP Layer
MHTTP Layer
Experiments

Key exchange protocol:



Secure Frame-to-frame Communication

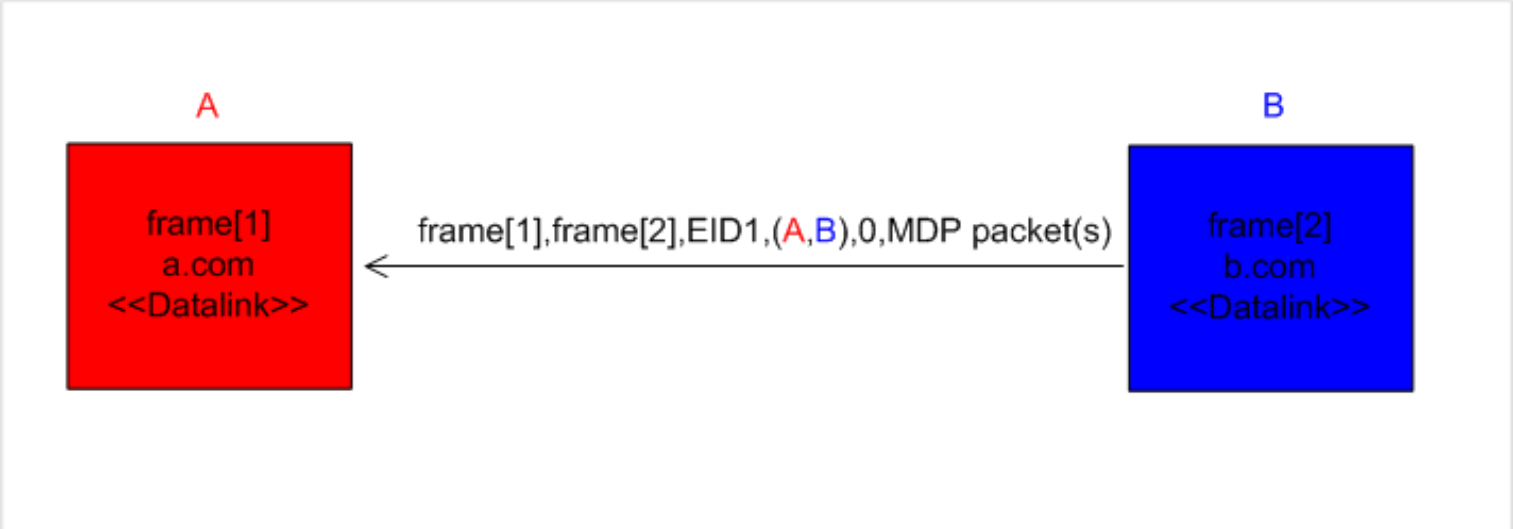
Introduction
OMOS
Overview
Mashlet
Secure Frame-to-frame Communication
Communication Stack
MDP Layer
MHTTP Layer
Experiments

Key exchange protocol:

Secure Frame-to-frame Communication

Introduction
OMOS
Overview
Mashlet
Secure Frame-to-frame Communication
Communication Stack
MDP Layer
MHTTP Layer
Experiments

Key exchange protocol:



Secure Frame-to-frame Communication

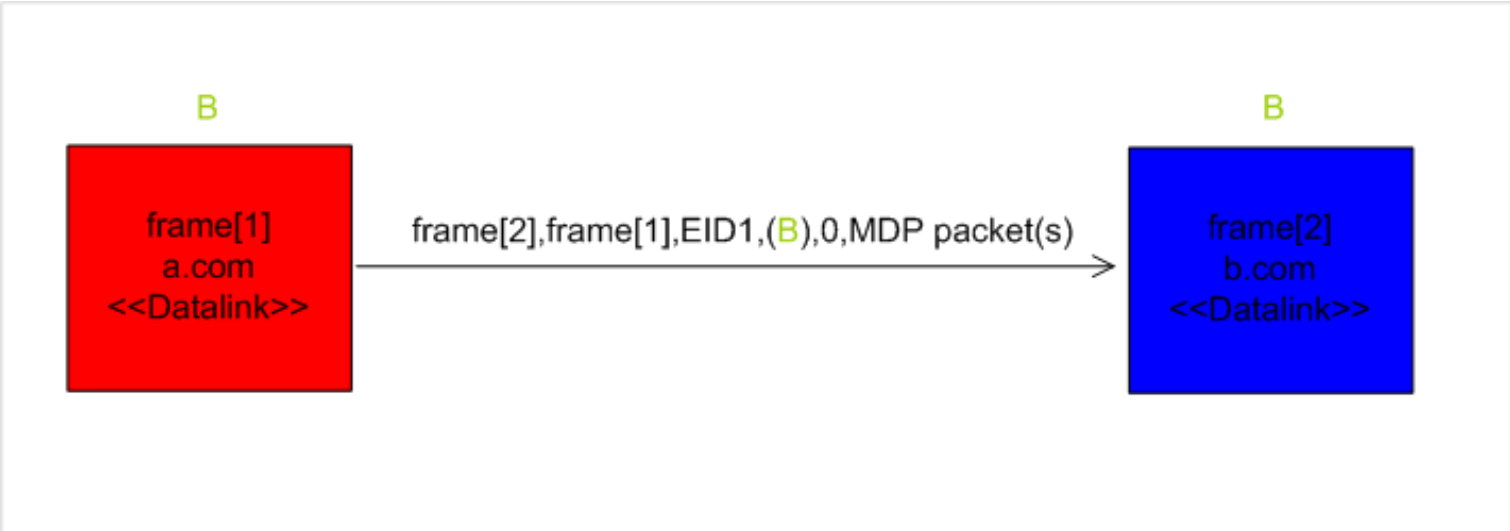
Introduction
OMOS
Overview
Mashlet
Secure Frame-to-frame Communication
Communication Stack
MDP Layer
MHTTP Layer
Experiments

Key exchange protocol:

Secure Frame-to-frame Communication

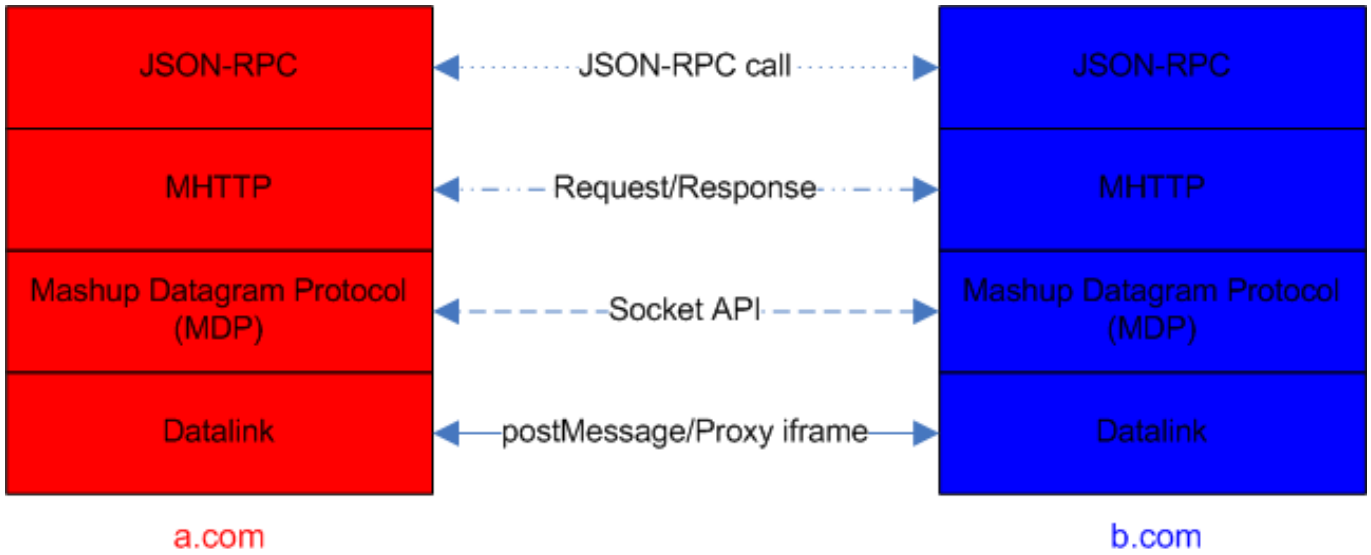
Introduction
OMOS
Overview
Mashlet
Secure Frame-to-frame Communication
Communication Stack
MDP Layer
MHTTP Layer
Experiments

Key exchange protocol:



Communication Stack

Introduction
OMOS
Overview
Mashlet
Secure
Frame-to-frame Communication
Communication Stack
MDP Layer
MHTTP Layer
Experiments

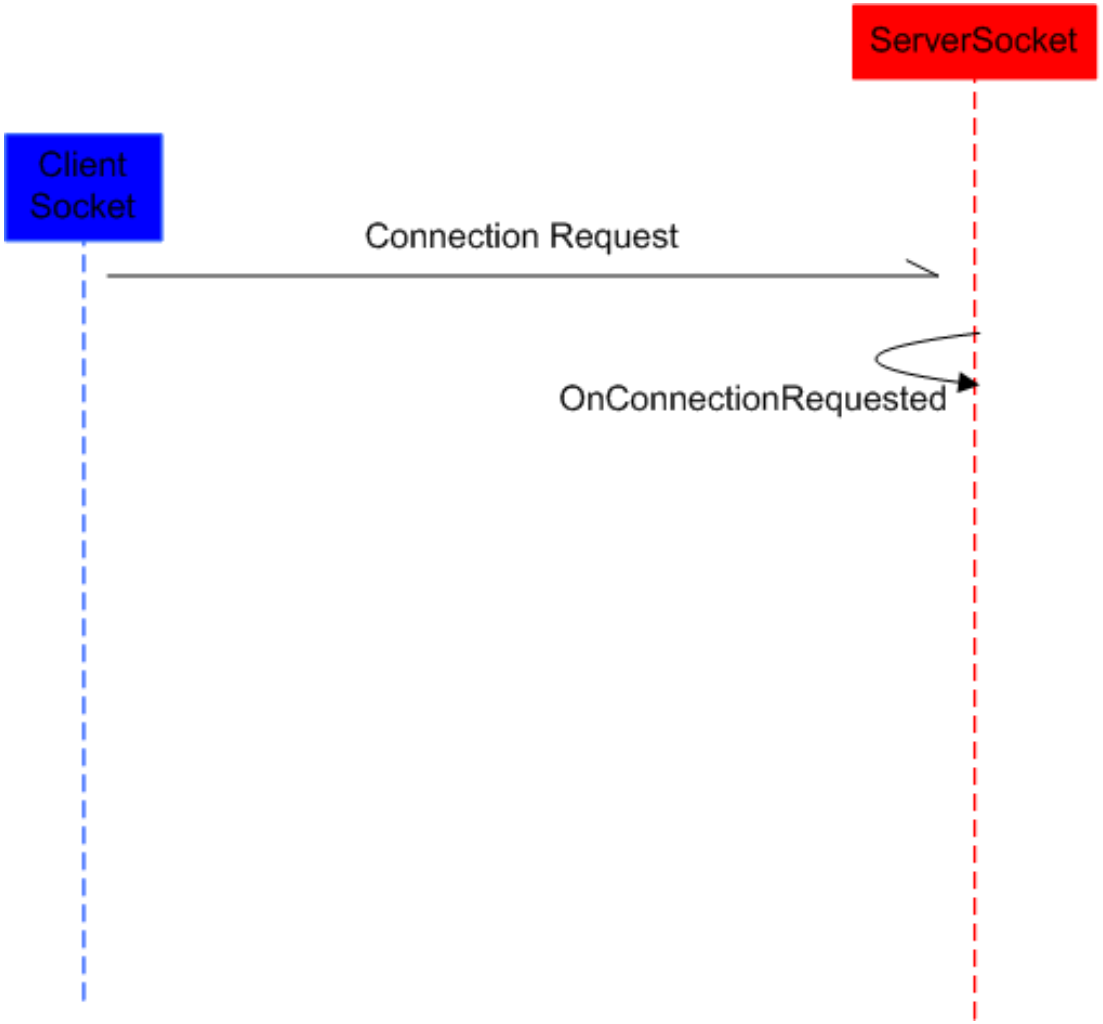


Each layer hides complex implementation details of communication in lower layers.

MDP Layer

Introduction
OMOS
Overview
Mashlet
Secure
Frame-to-frame
Communication
Communication
Stack
MDP Layer
MHTTP Layer
Experiments

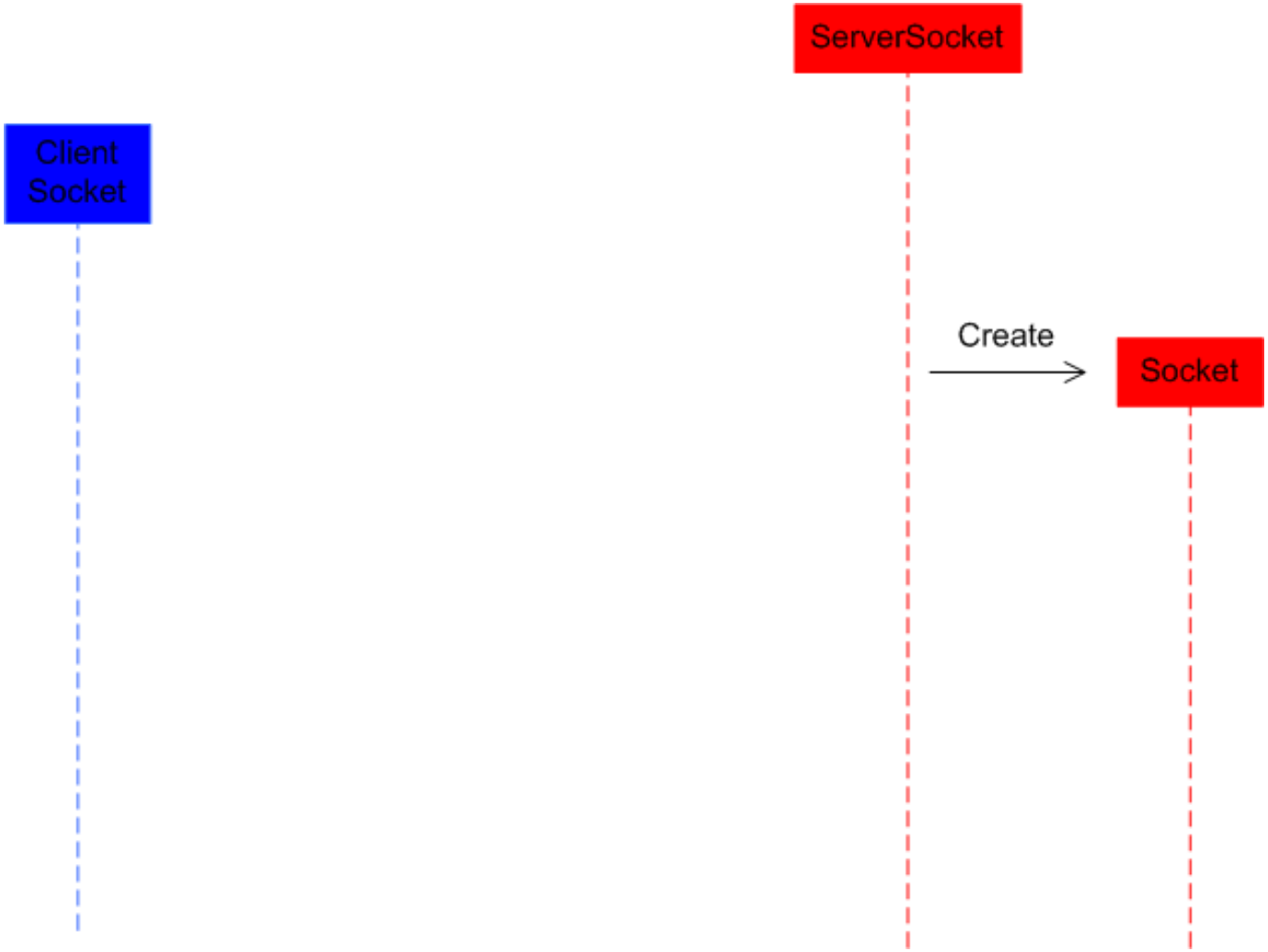
3-way Handshake



MDP Layer

Introduction
OMOS
Overview
Mashlet
Secure
Frame-to-frame
Communication
Communication
Stack
MDP Layer
MHTTP Layer
Experiments

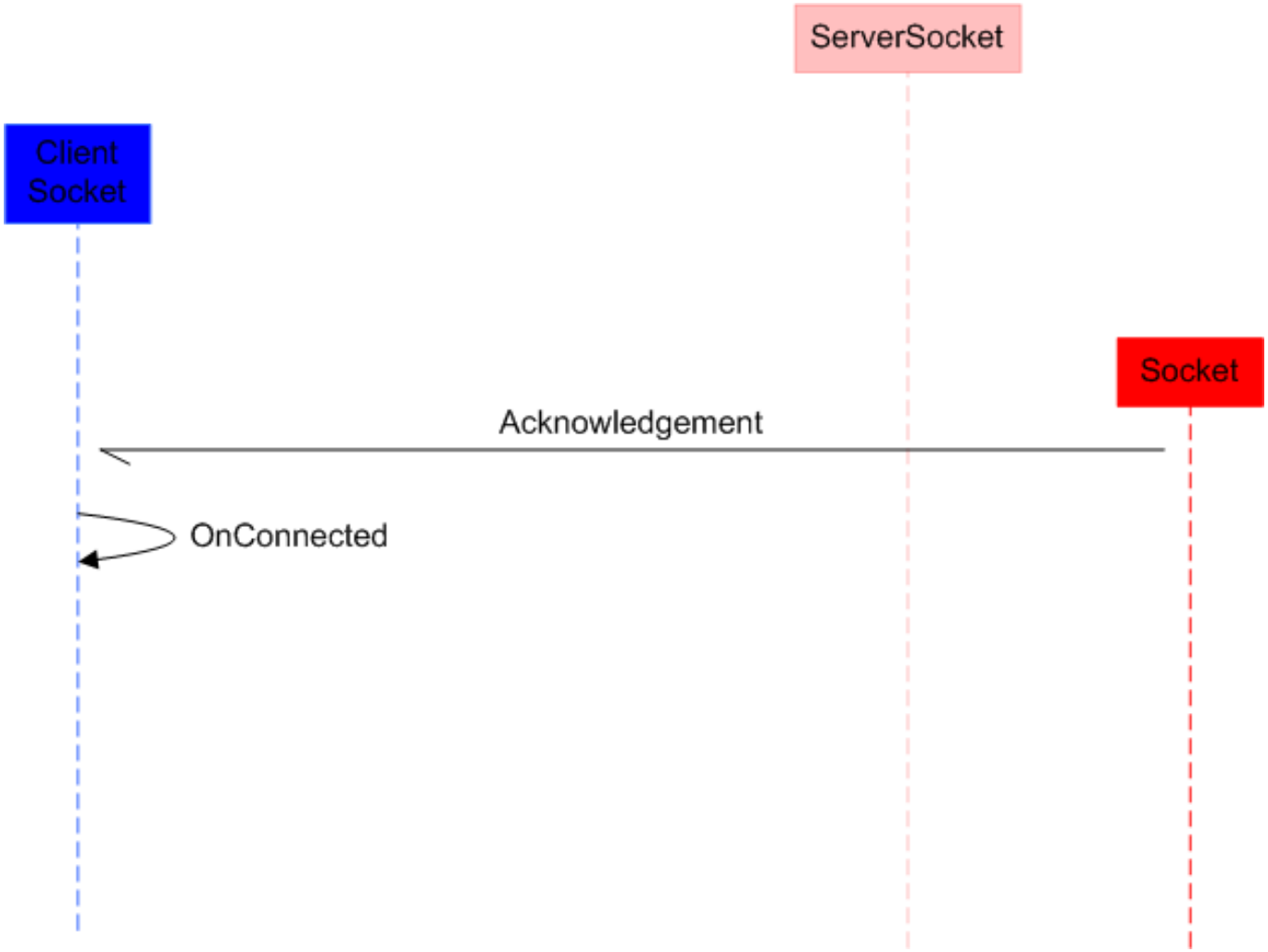
3-way Handshake



MDP Layer

Introduction
OMOS
Overview
Mashlet
Secure
Frame-to-frame
Communication
Communication
Stack
MDP Layer
MHTTP Layer
Experiments

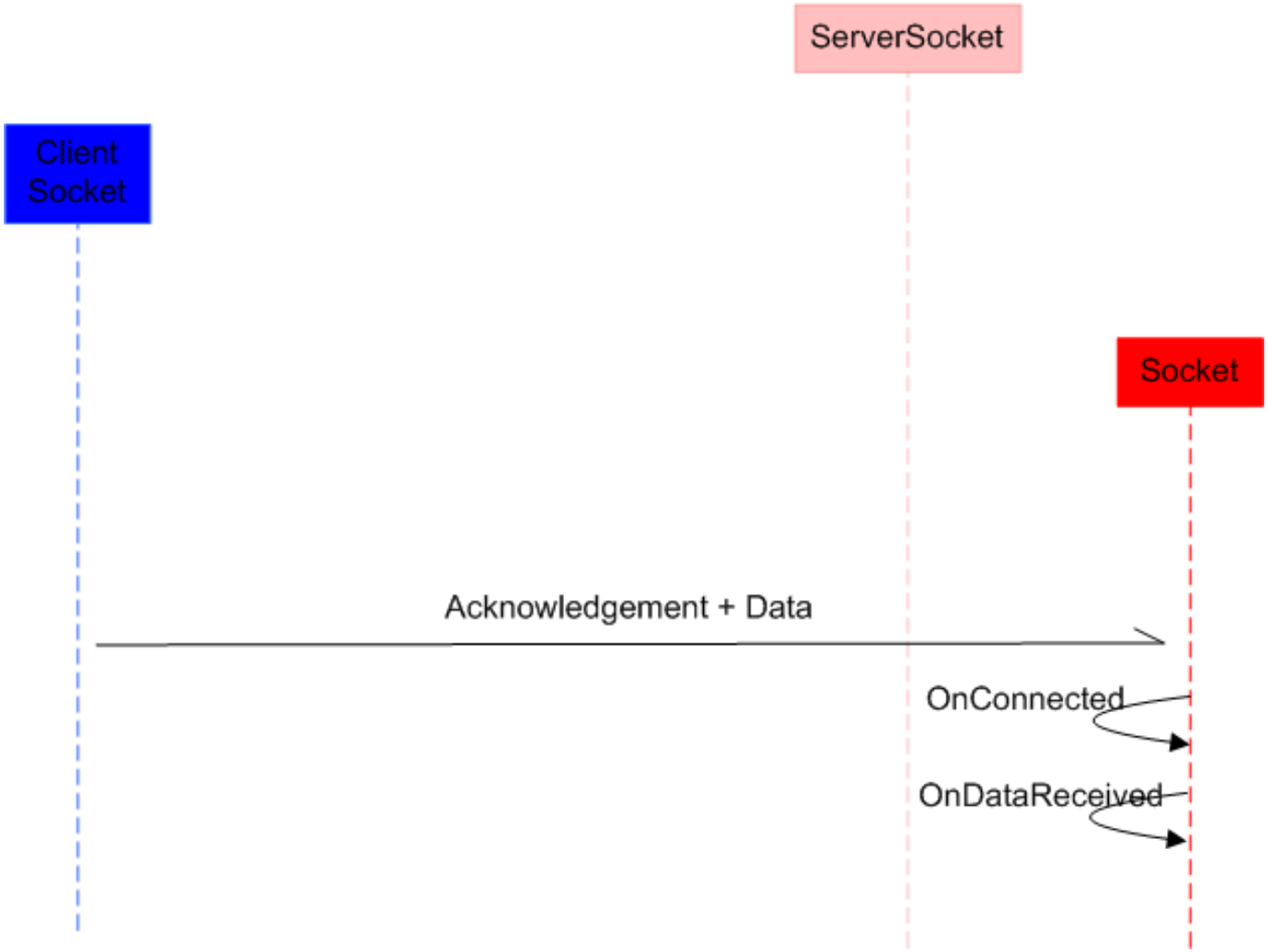
3-way Handshake



MDP Layer

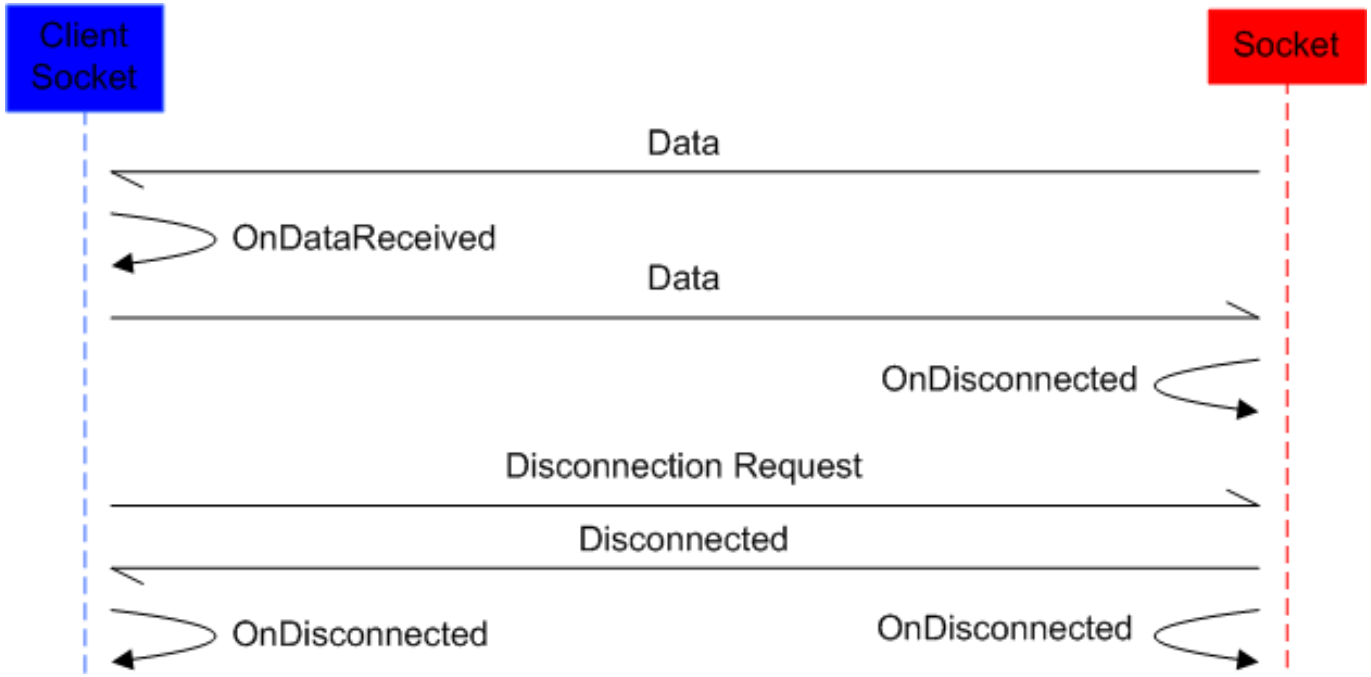
Introduction
OMOS
Overview
Mashlet
Secure
Frame-to-frame
Communication
Communication
Stack
MDP Layer
MHTTP Layer
Experiments

3-way Handshake



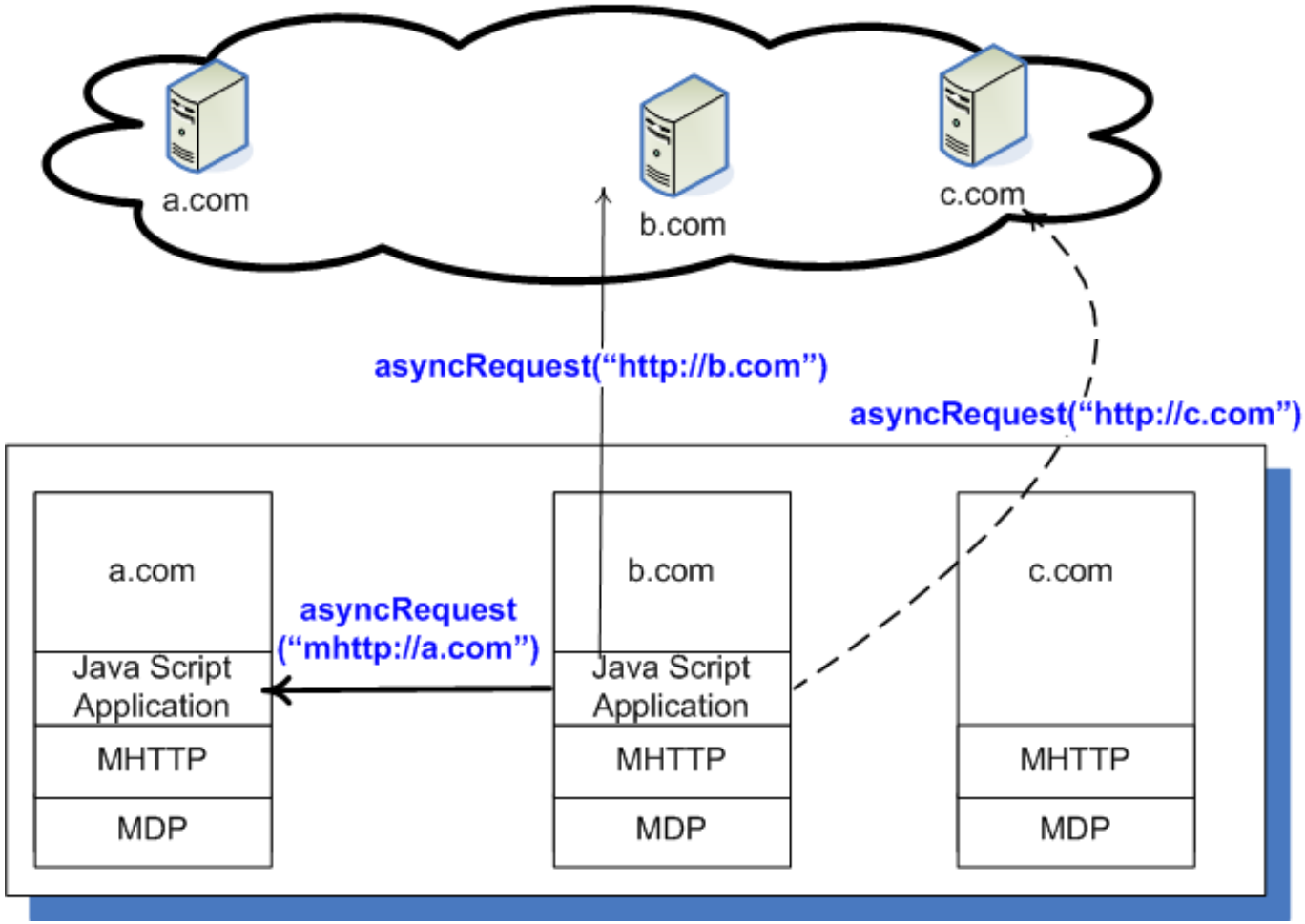
MDP Layer

Introduction
OMOS
Overview
Mashlet
Secure
Frame-to-frame
Communication
Communication
Stack
MDP Layer
MHTTP Layer
Experiments



MHTTP Layer

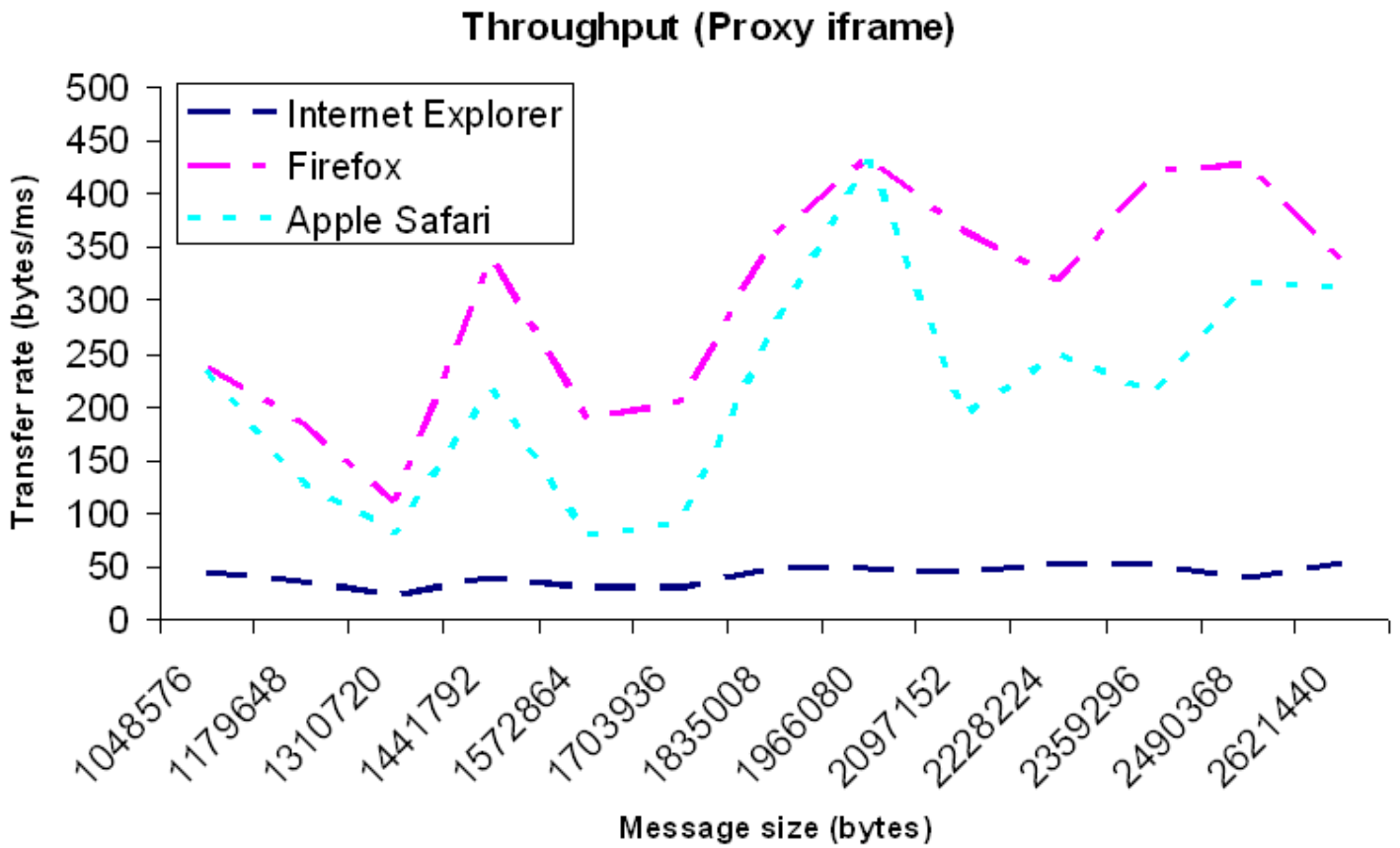
Introduction
OMOS
Overview
Mashlet
Secure
Frame-to-frame
Communication
Communication
Stack
MDP Layer
MHTTP Layer
Experiments

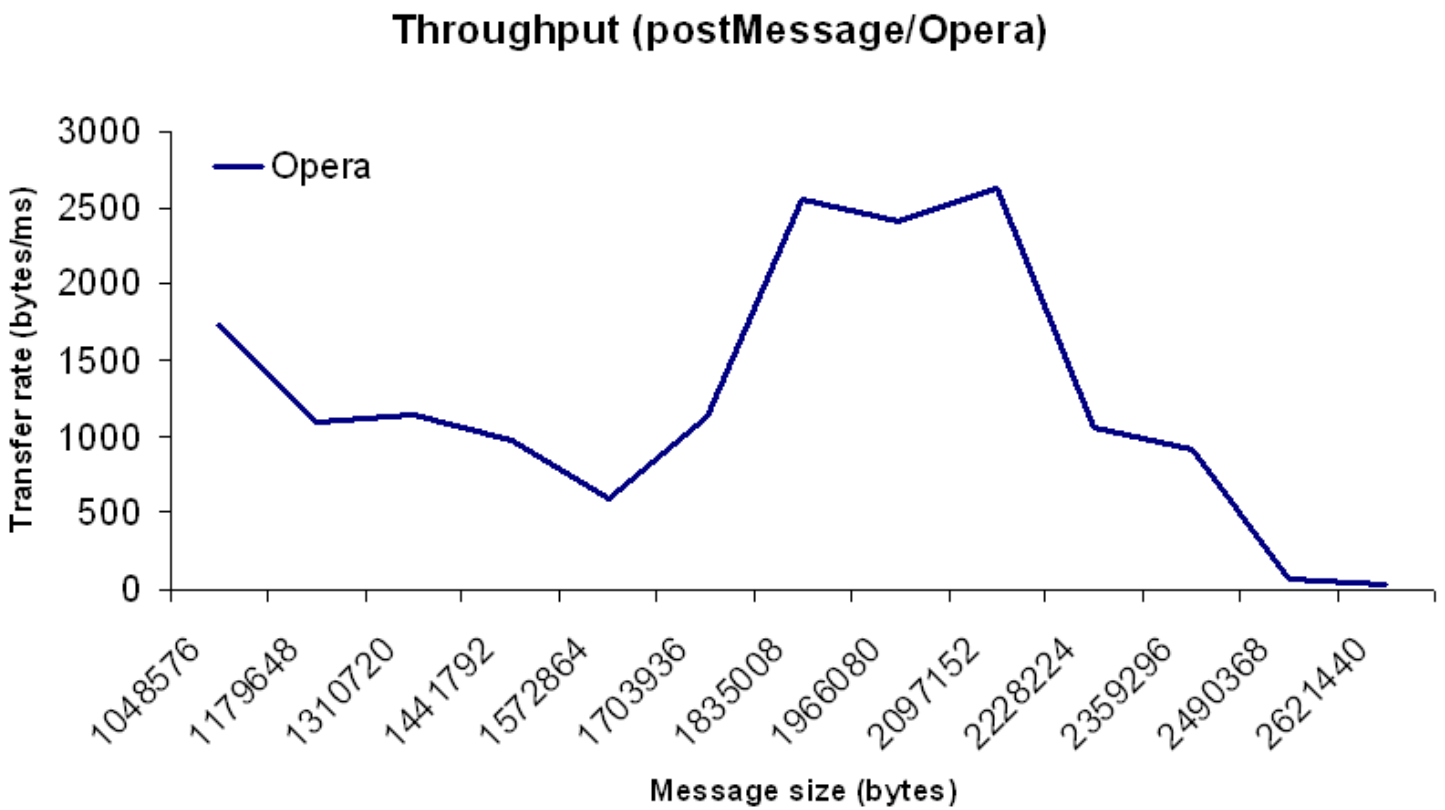


Versatile asyncRequest: mashlet-to-mashlet, same-domain & cross-domain mashlet-to-server communication.

Results

Introduction
OMOS
Experiments
Results



Results

Introduction
OMOS
Experiments
Results