IBM T.J. Watson Research Center

Private Use of Untrusted Web Servers via Opportunistic Encryption

Mihai Christodorescu

mihai@us.ibm.com

May 22, 2008

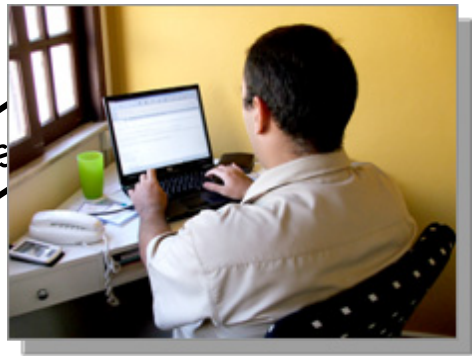
What is the Problem?

Program Committee

Read, comment

Conference Submission
Web Server

Full acces



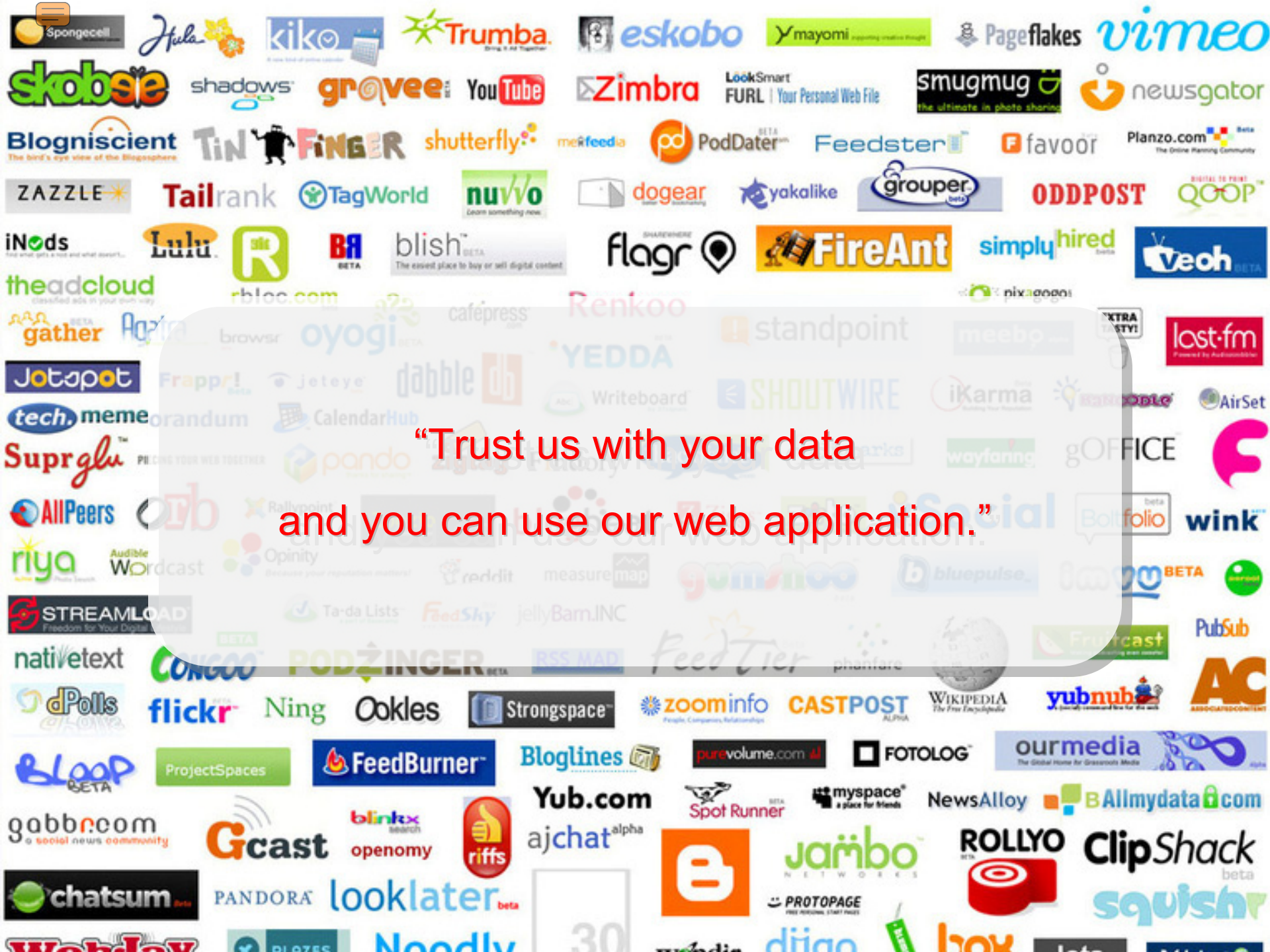
Software Developer and
Server Administrator

Submit

Submitting Authors

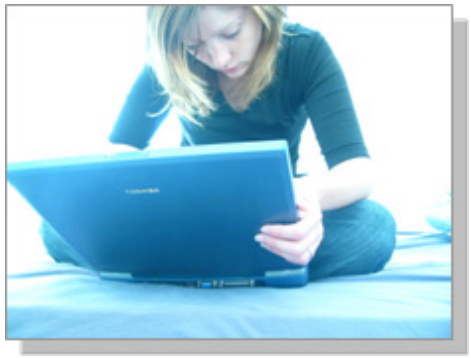
Can the PC trust the server with conference data?

Photos courtesy of Bruno Neves, David Lat, Justyna Furmanczyk, Whrelf Siemens.


"Trust us with your data
and you can use our web application."

Overview of Today's Talk

Problem: Web applications require trust in the server.

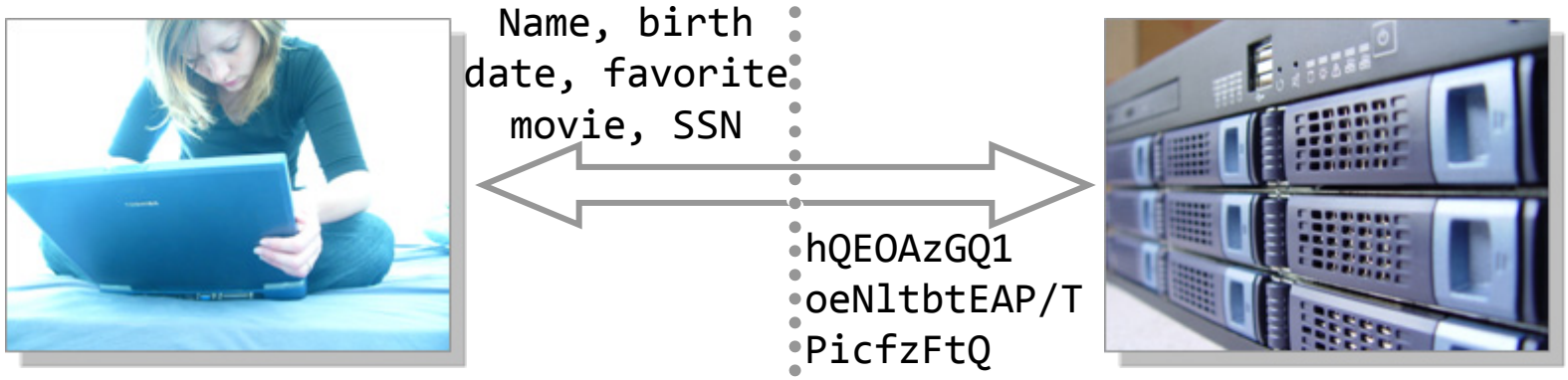


Name, birth date, favorite movie, SSN

Photos courtesy of Elena Buetler, Whrelf Siemens.

Overview of Today's Talk

Problem: **Web applications require trust in the server.**



Solution: **Encrypt all data leaving your computer.**

Photos courtesy of Elena Buetler, Whrelf Siemens.

Not a New Idea…

- **We have elegant solutions for securing data in remote storage:** [Fu*etal*2000], [Kallahalla*etal*2003], …

- **What about data on remote servers?**
  - Richard Schwartz proposed “host-proof hosting” in 2005
  - Marco Barulli proposed “zero-knowledge [sic] web applications” in 2007

Today: a research agenda to make web applications that use encrypted data ubiquitous.

Surely We Already Have Some Solutions!?!

- **P3P**
- **privacy-oriented web-service models**
- **privacy obligations**
- **certification**
- **…**

They all *assume* that the web-application provider is trustworthy.

But… Web Providers Can Change Policies

- **Famous words:**

  "Please note that this Privacy Policy may change from time to time."

  from *Google's Privacy Policy*

  "I am altering the deal. Pray I don't alter it any further."

  from Darth Vader's M.O.

- **PayPal issued 15 policy changes in 4 years.**

But… Companies Can Pass Data to Others

- **Outsourcing partners, etc.:**

  "We provide [your personal] information to our subsidiaries, affiliated companies or other trusted businesses or persons for the purpose of processing personal information on our behalf."

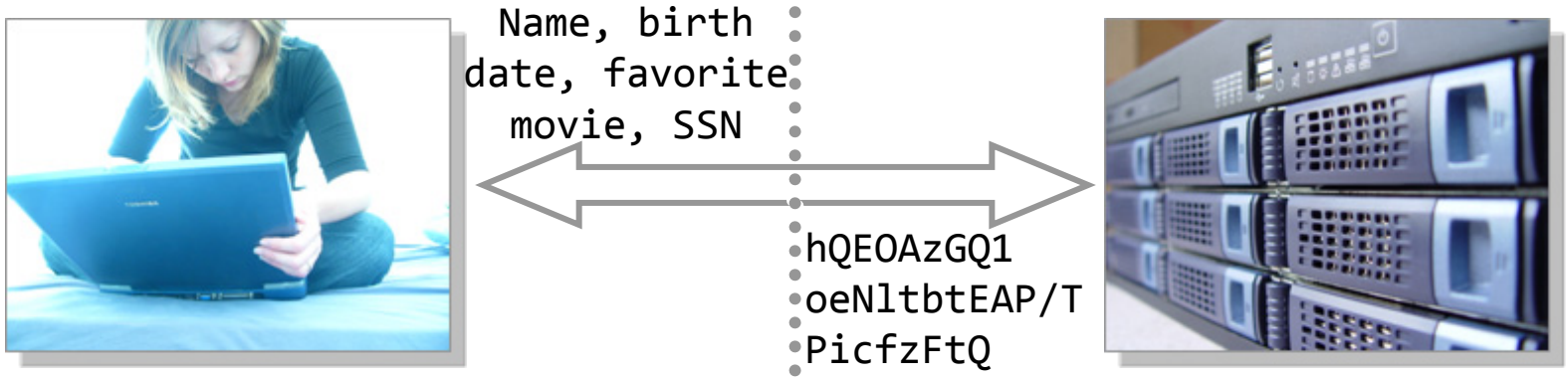
  from *Google's Privacy Policy*

Do you even know where the data might end up?

But… Servers Can Get Hacked

- U.K. drug-store chain Boots lost info of 35,000 people
- Bank of NY Mellon lost info of 1,376 SAIC investors
- Hong Kong hospitals lost data of 3,000 patients
- U.S. military contractor stole info on 17,000 employees
- Bank of Ireland lost data of 30,000 customers
- University of Miami lost data of 47,000 people
- Server stolen from Central Collection Bureau, IN, contained records for 700,000 people
- Intrusion at Okemo Mountain Resort, VT, compromised 46,000 payment card transactions
- Agilent lost laptop with records for 51,000 employees
- Insider theft at Certegy Check Services Inc. compromises data of 8,500,000 consumers

Solution: Encrypt All Data Leaving Your Computer



Requirements

- **User experience should be unchanged.**
- **Web-app provider should use the same programming model.**
- **No misuse of user data on the web server.**

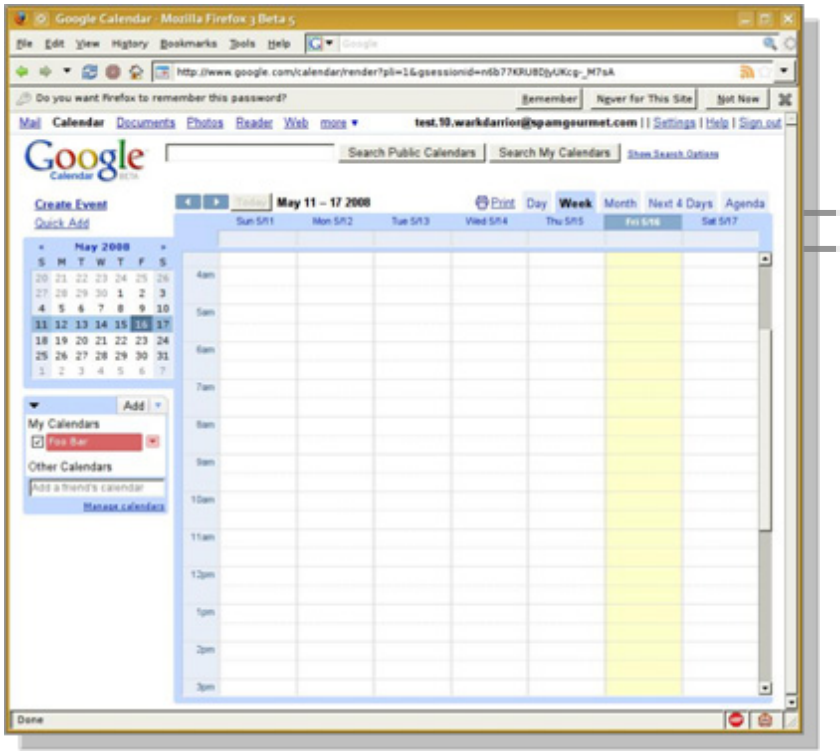
Sticky-policy paradigm:

Submitted data is associated with a usage policy. Association holds even if data is further disclosed.

[KarjothSchunterWaidner2002]

How would this work?

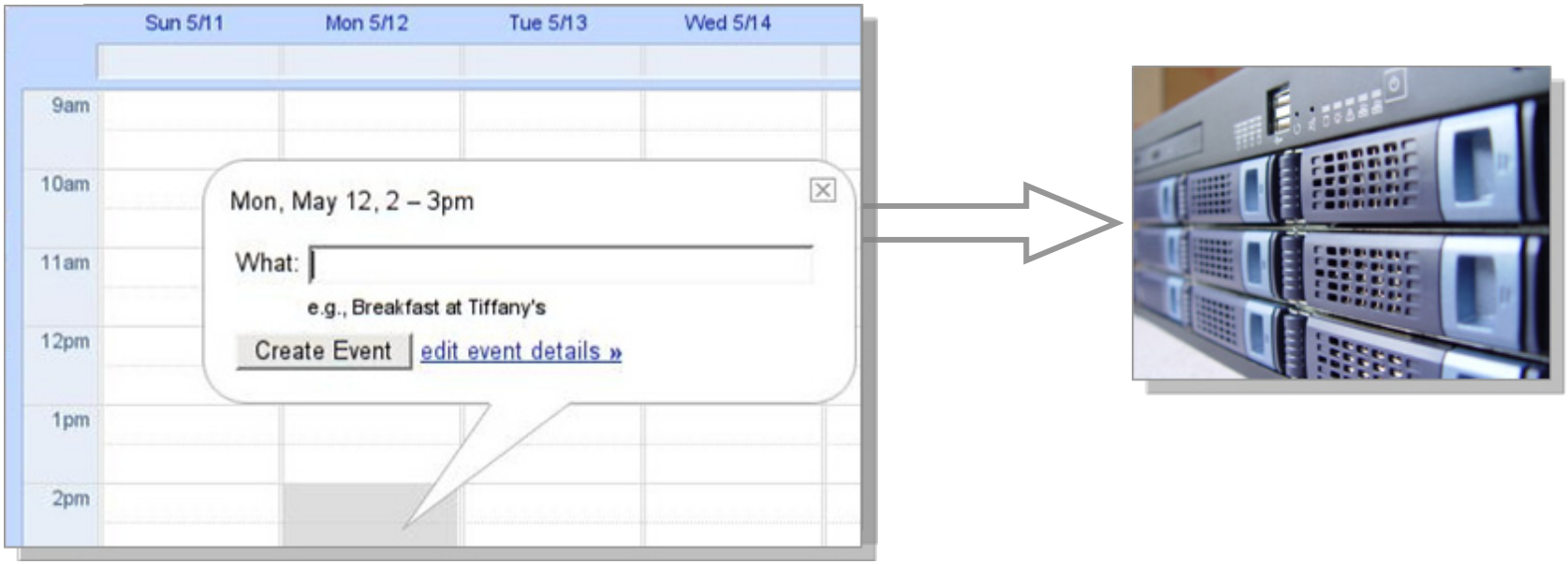
- **Google Calendar example**



- Web-based application
- Code on both the client and the server
- Clean, user-friendly UI
- Sharing, embedding, mashup features

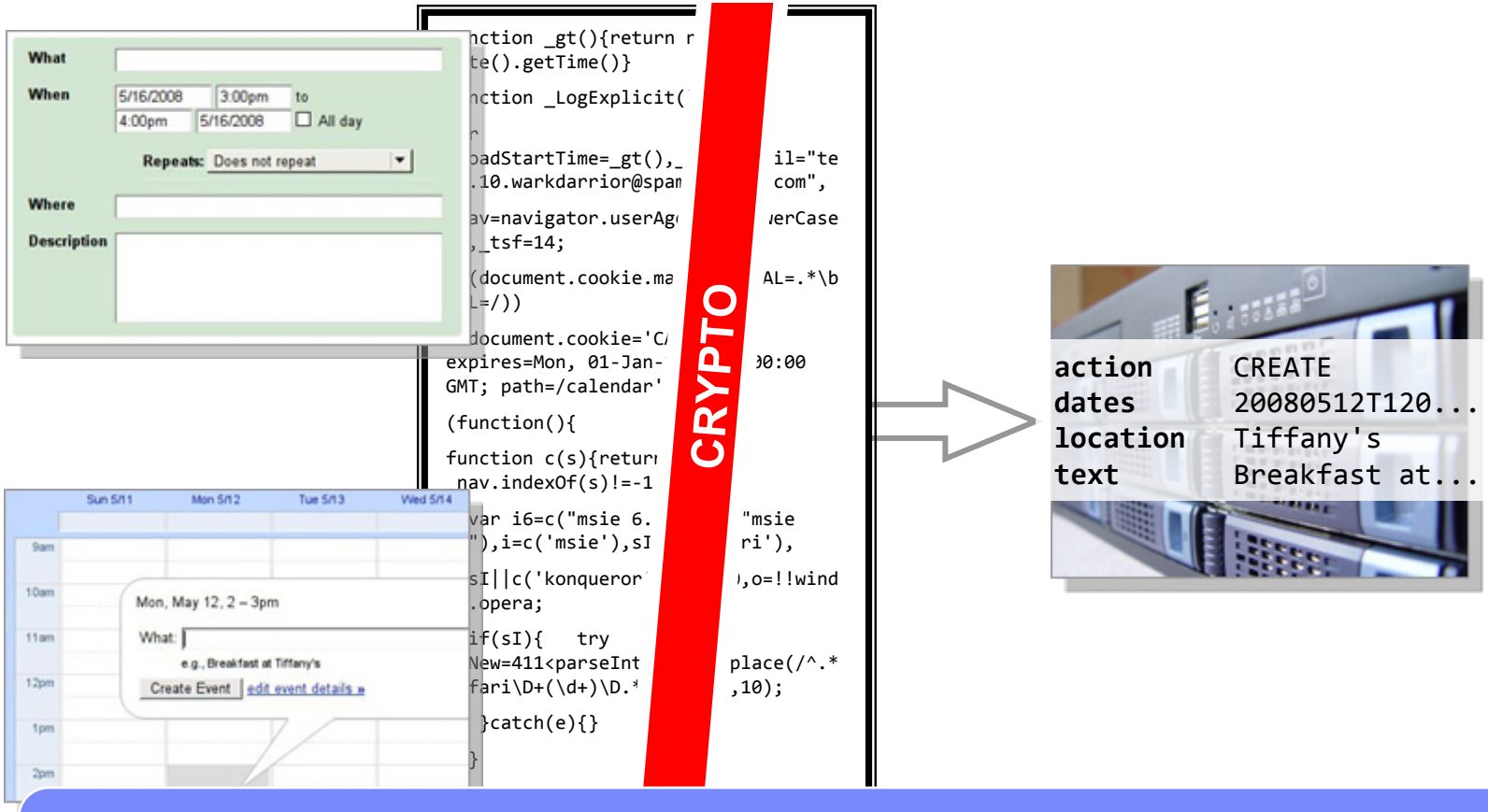
Basic Google Calendar Interaction

- **Create a new event:**



Not all inputs can be encrypted in the UI…

Basic Google Calendar Interaction: UI Challenge



Partition client-side code into a UI component and a network component, mediated by a crypto layer.

Basic Google Calendar Interaction: Key Challenge

- **Encryption key:**
  - Hidden from the web-app provider
  - Transparent to the user
- **User still needs to authenticate to the web app.**

password = web-app credential

password $\longrightarrow$ **key** $\longrightarrow$ **web-app credential**

PRF(pwd) **E_K(domain)**

[RossJacksonMiyakeBonehMitchell2005]

Calendar Mashups

- **Client-side mashups – OK**
- **Server-side mashups…**

Server-Side Calendar Features

- **Notifications : server must compute over event dates**

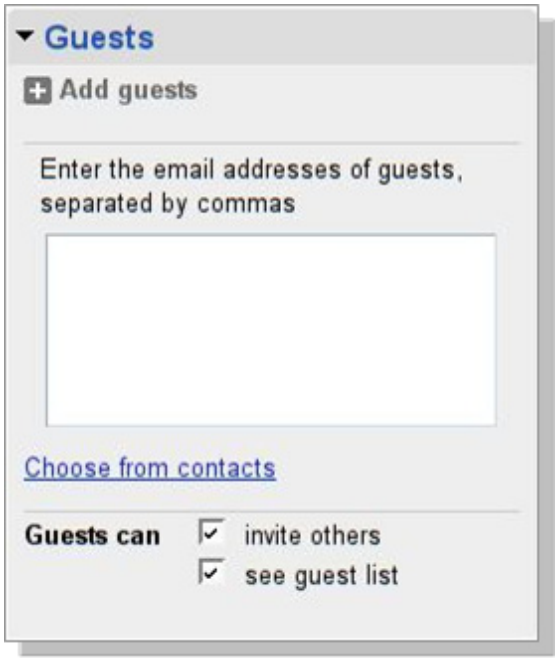
- **Search : server must perform partial matches over event titles and descriptions**

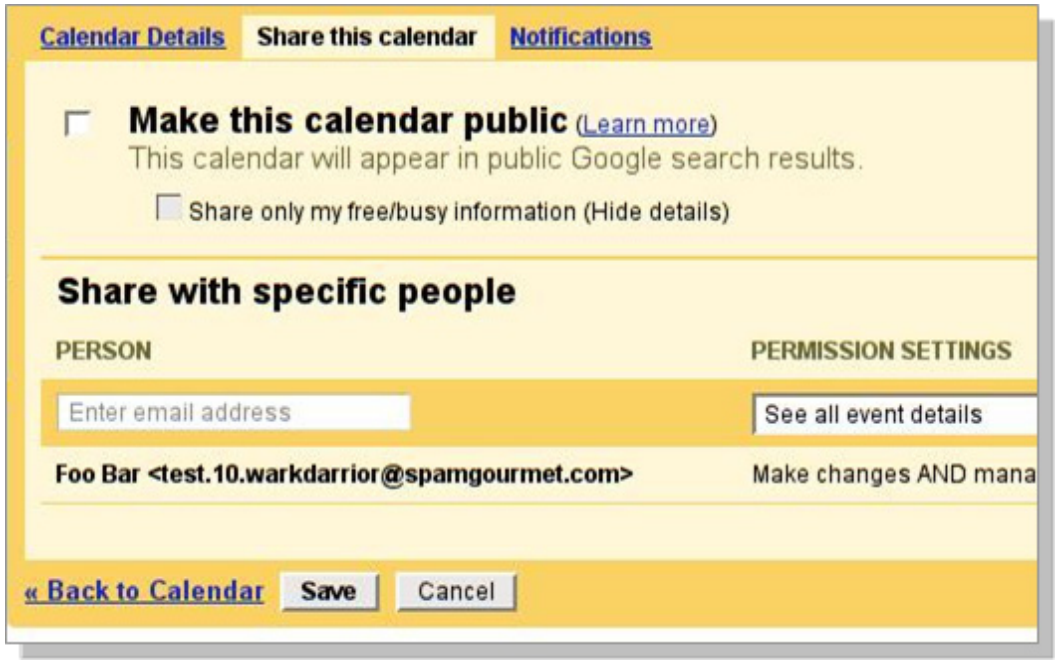
Use different cryptographic schemes for different data types, to allow for server-side computation.

Server-Side Calendar Features: Challenges

- **We need cryptographic schemes that support operations over encrypted data.**
  - Homomorphic encryption: $E_K(x) + E_K(y) = E_K(x+y)$
  - More general: Cryptocomputing [SanderYoung2001]
- **We need to know the type of data entered by the user.**
  - Operations performed on the server
  - Search, date arithmetic, …

Calendar Sharing





- **Need to communicate the encryption key to the other people.**

Calendar Sharing: Challenges

- **Caveat:**

  If you share with *everyone*, you might as well not encrypt.

- **Options for key distribution:**
  - Via email, IM, phone, …
  - Identity-based encryption?
  - Proxy re-encryption?

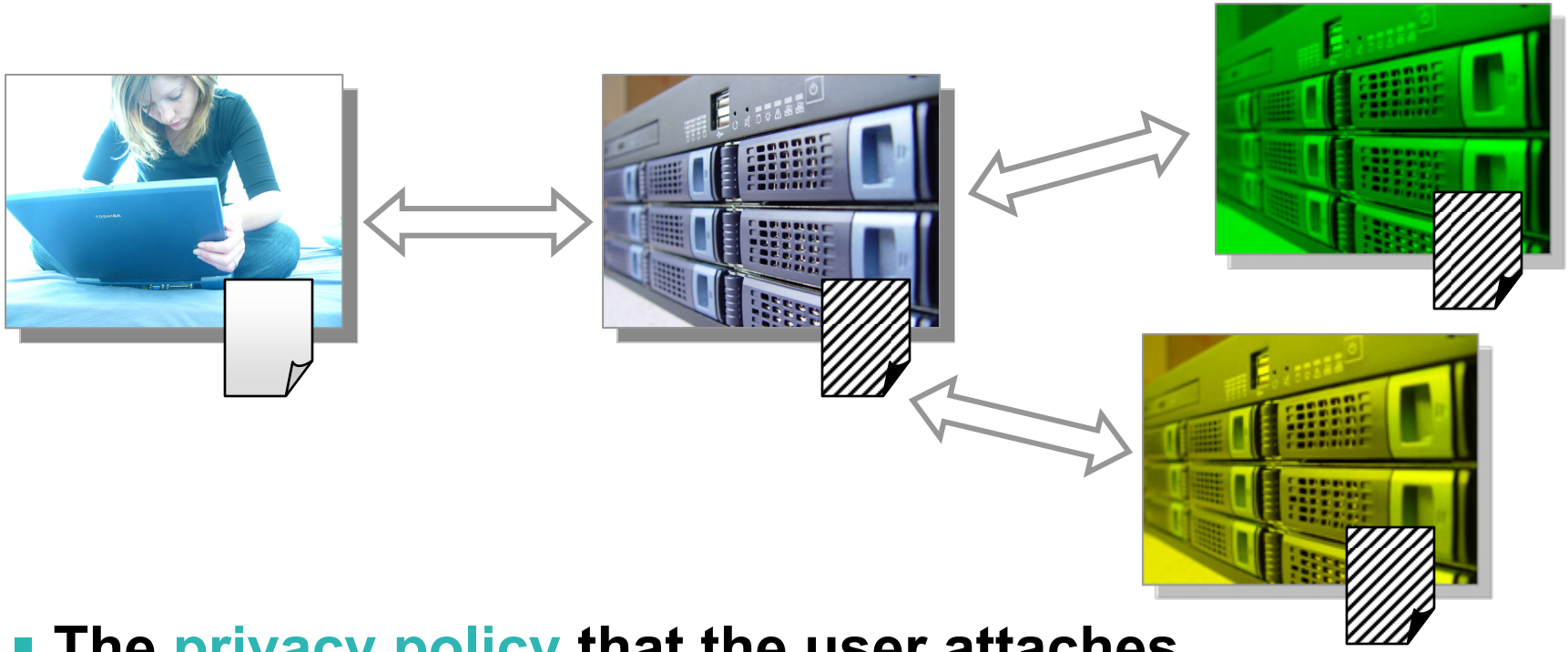
Research Challenges

- **Automatic partitioning of client-side code**
- **Encryption schemes that allow for computation over ciphertexts**
- **Server-side types for user data**
- **Key-distribution protocols to enable data sharing**
- **Reencryption on password updates, data portability**

How Do We Know We Succeeded?

- **When the user does not know it is there (or is just aware…).**
  - Client-side XSS and CSRF attacks still work!
- **When the web server does not know it is there.**
- **When anyone trying to misuse the data on the web server fails.**
  - Client-side attacks (e.g., spyware) still a problem!

Implications



- The privacy policy that the user attaches to data becomes sticky.
- The privacy restrictions placed by the web application become obvious to the user.

Future Work

- **Automatic privilege separation for client-side code**
  - Javascript, Java, Flash, …
- **New crypto schemes**
  - Ciphertext search, comparison, date calculation, …
- **Discovery of server-side data types**
  - Black-box analysis? User-driven cooperative analysis?
- **Key distribution**
- **Data integrity**

Questions?

Private Use of Untrusted Web Servers via Opportunistic Encryption

Mihai Christodorescu
mihai@us.ibm.com