Goals for this lecture

☞ **Brief overview of HTTPS:**

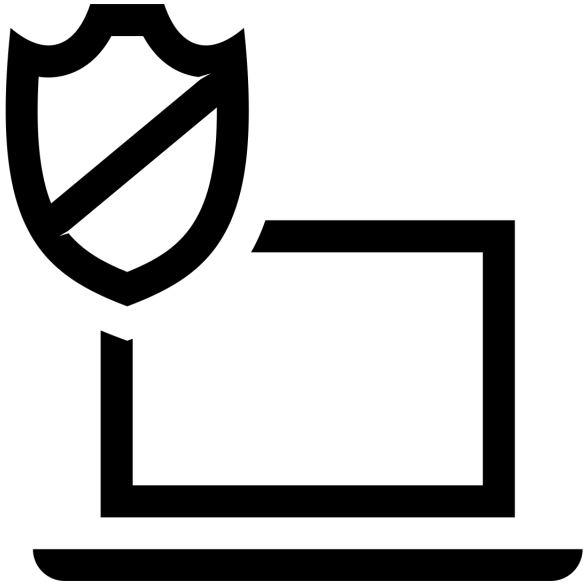
- How the SSL/TLS protocol works (very briefly)
  How to use HTTPS

☞ **Integrating HTTPS into the browser:**

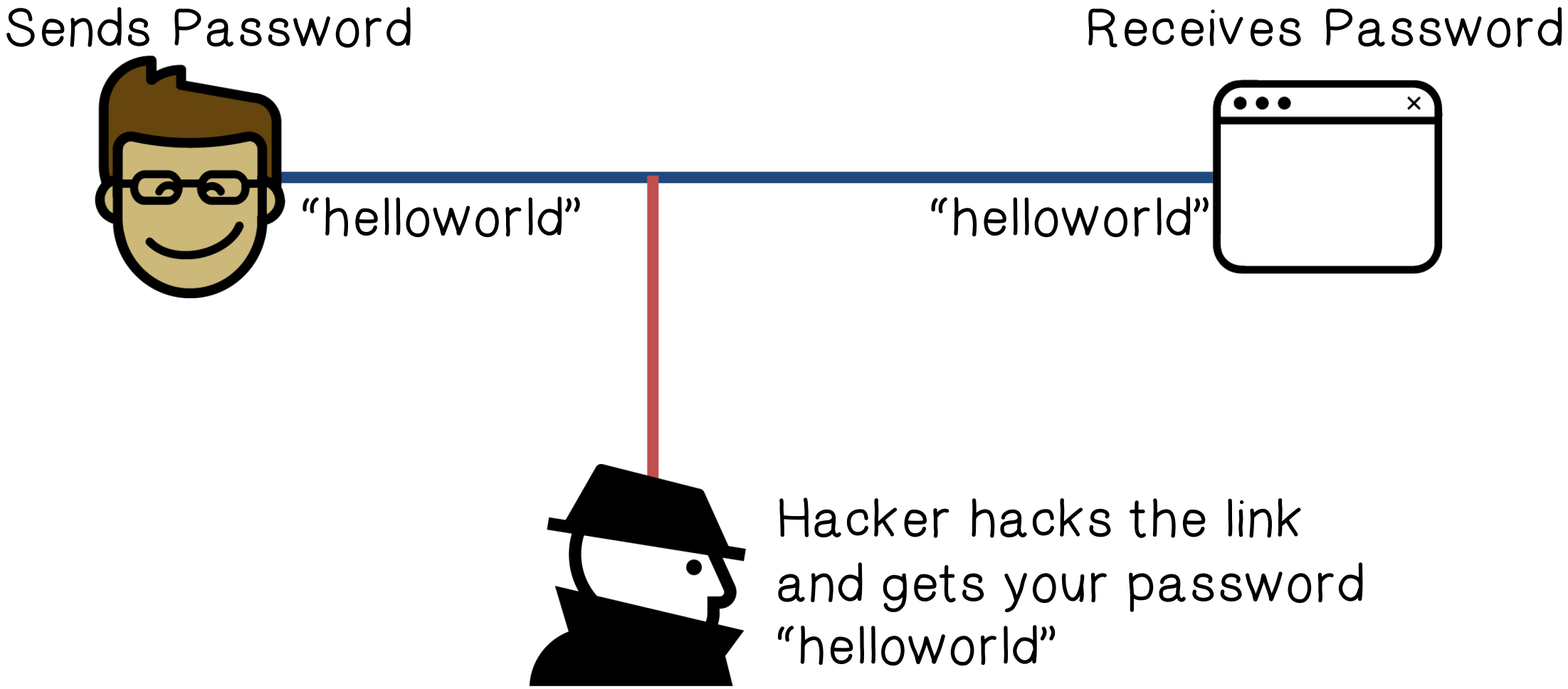
- Lots of user interface problems to watch for

HTTPS

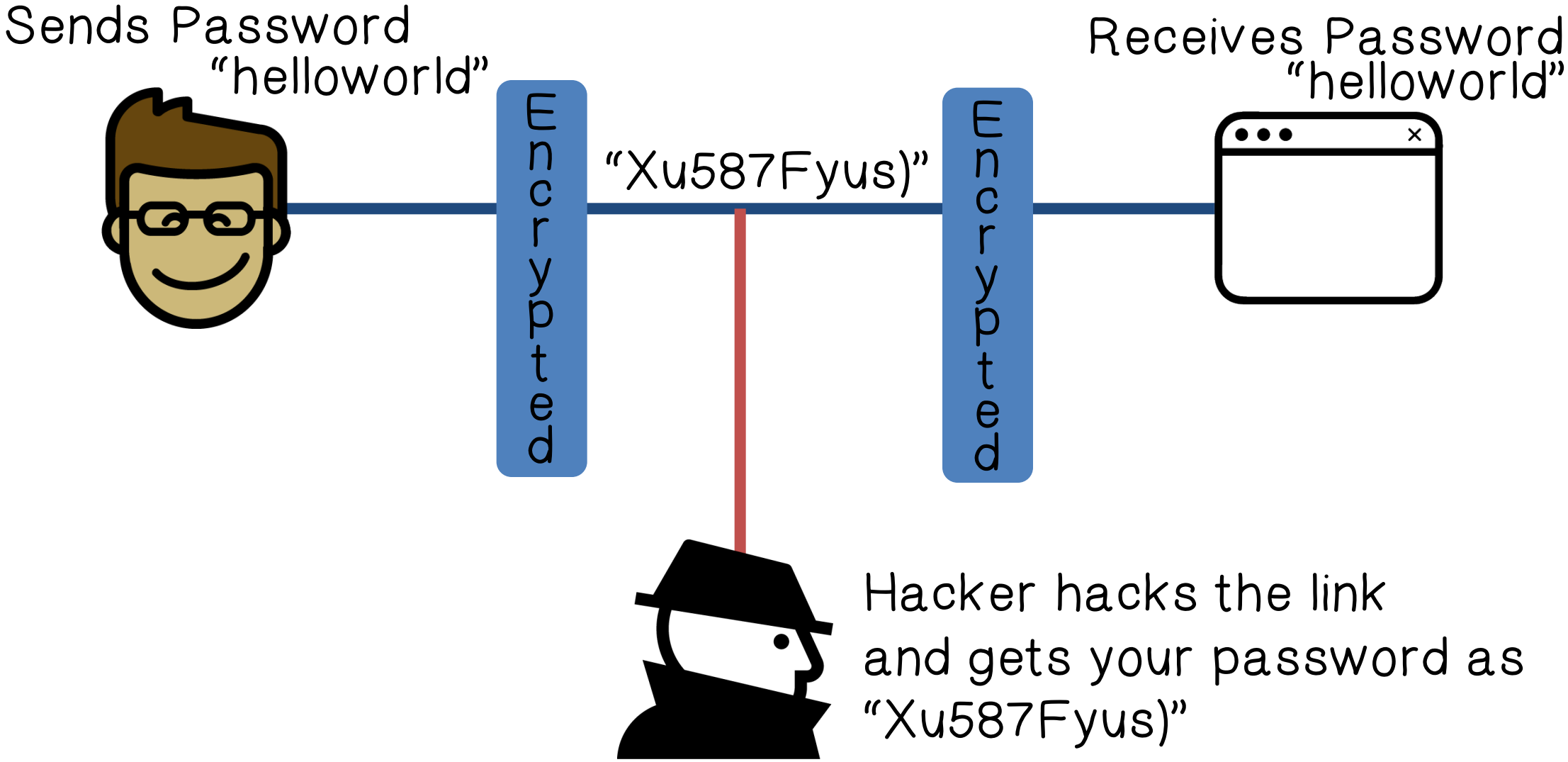
HTTPS: Hyper Text Protocol Secure= all communication between your browser and a website is encrypted.



HTTP


Sends Password
Receives Password
"helloworld"
"helloworld"
Hacker hacks the link
and gets your password
"helloworld"

HTTPS

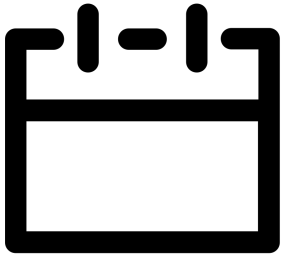

Sends Password
"helloworld"
Encrypted
"Xu587Fyus)"
Encrypted
Receives Password
"helloworld"
Hacker hacks the link
and gets your password as
"Xu587Fyus)"

HTTPS

- Creates a secure channel over an insecure network
- Is reasonable protection against man-in-the-middle attacks
- Can still provide security even when only one side of the communication is secure

Why is HTTPS not used for all web traffic?

- Crypto slows down web servers (but not by much if done right.)
- Some ad-networks do not support HTTPS (2015 stats: 20%)
  - Reduced revenue for publishers



August 2014: Google boosts ranking of sites supporting HTTPS

HTTPS Quiz

HTTPS can encrypt the underlying HTTP protocol which means quite a bit of data can be encrypted.

Select all of the items that can be encrypted by HTTPS:

- [x] Request URL
- [x] Query parameters
- [x] Headers
- [x] Cookies
- [ ] Host Addresses
- [ ] Port numbers
- [ ] The amount of transferred data
- [ ] Length of the session

Threat Model: Network Attacker

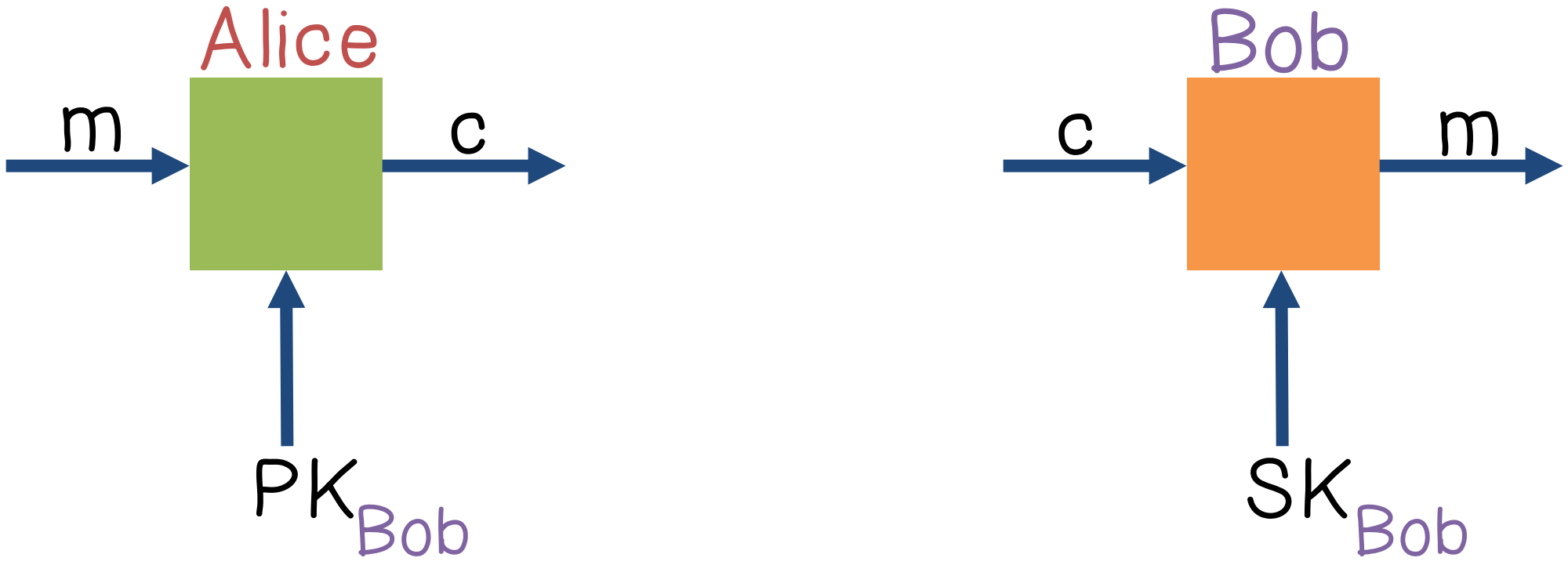
☞ Network Attacker:

- Controls network infrastructure: Routers, DNS
- Eavesdrops, injects, blocks, and modifies packets

Examples:

- Wireless network at Internet Café
- Internet access at hotels (untrusted ISP)

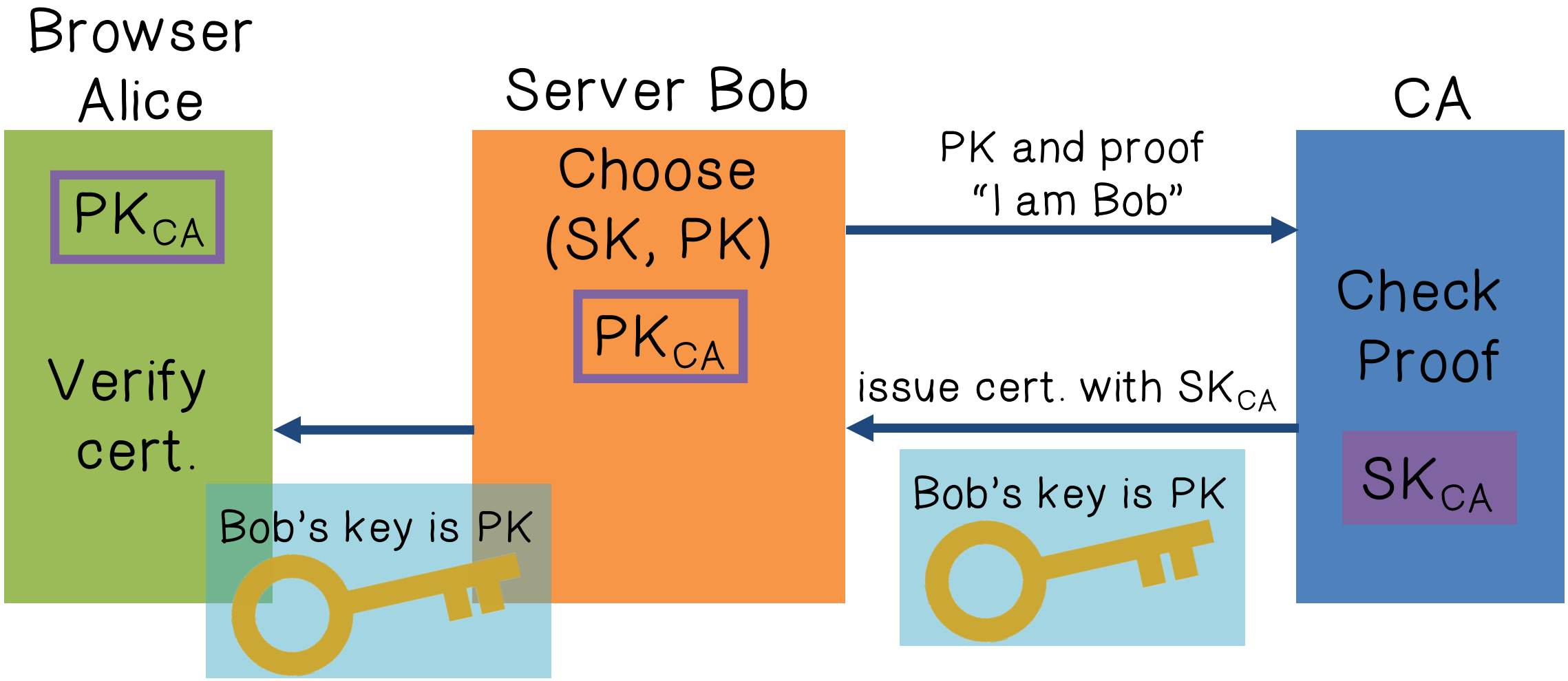
SSL/TLS overview



Bob: generates (SK_{Bob}, PK_{Bob})

Alice: using PK_{Bob} encrypts messages and only Bob can decrypt

☞ How does Alice (browser) obtain PK_{Bob}?

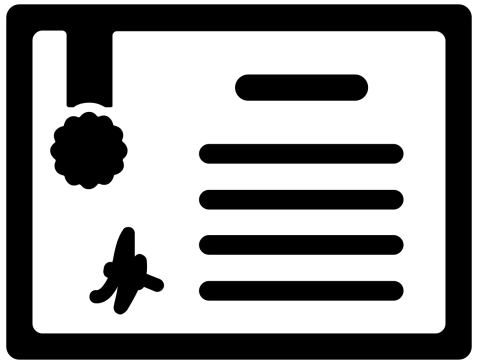


Important Fields:

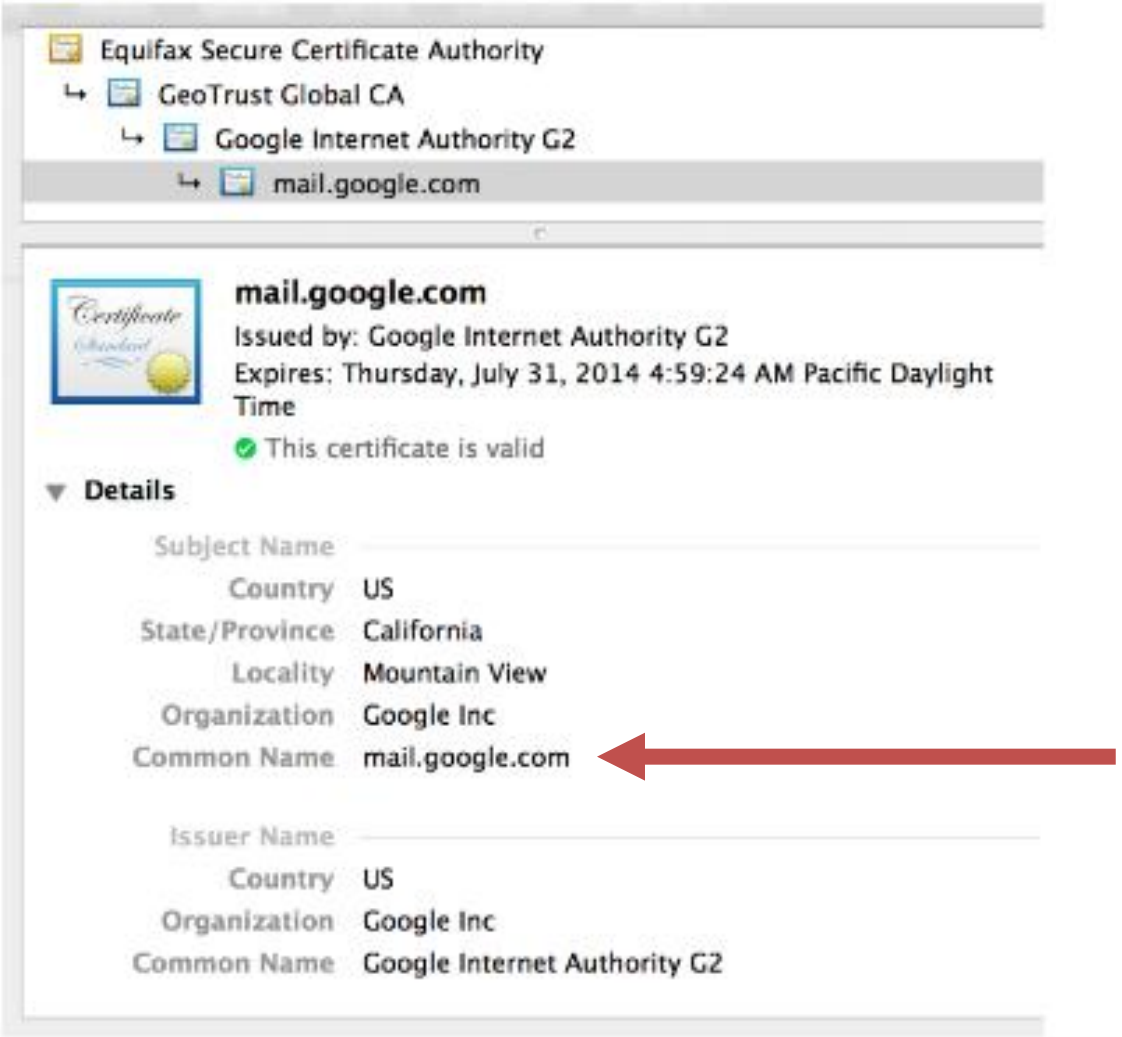
Serial Number	5814744488373890497
Version	3
Signature Algorithm	SHA-1 with RSA Encryption (1.2.840.113569.1.1.5)
Parameters	none
Not Valid Before	Wednesday, July 31, 2013 4:59:24 AM Pacific Daylight Time
Not Valid After	Thursday, July 31, 2014 4:59:24 AM Pacific Daylight Time

Public Key Info:

Algorithm	Elliptic Curve Public Key (1.2.840.10045.2.1)
Parameters	Elliptic Curve secp256r1 (1.2.840.10045.3.1.7)
Public Key	65 bytes: 04 71 6C DD E0 0A C9 76...
Key Size	256 bits
Key Usage	Encrypt, Verify, Derive
Signature	256 bytes: 8A 38 FE D6 F5E7 F6 59 ...



Certificate Examples



Certificates on the Web

Subject's CommonName can be:

- An explicit name, e.g. cc.gatech.edu
- A wildcard cert, e.g. *.gatech.edu

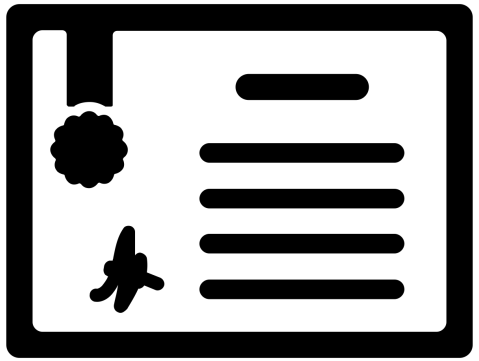
Matching Rule:

- "*" must occur in leftmost component, does not match "." For example: "*.a.com" matches "x.a.com" but not "y.x.a.com"

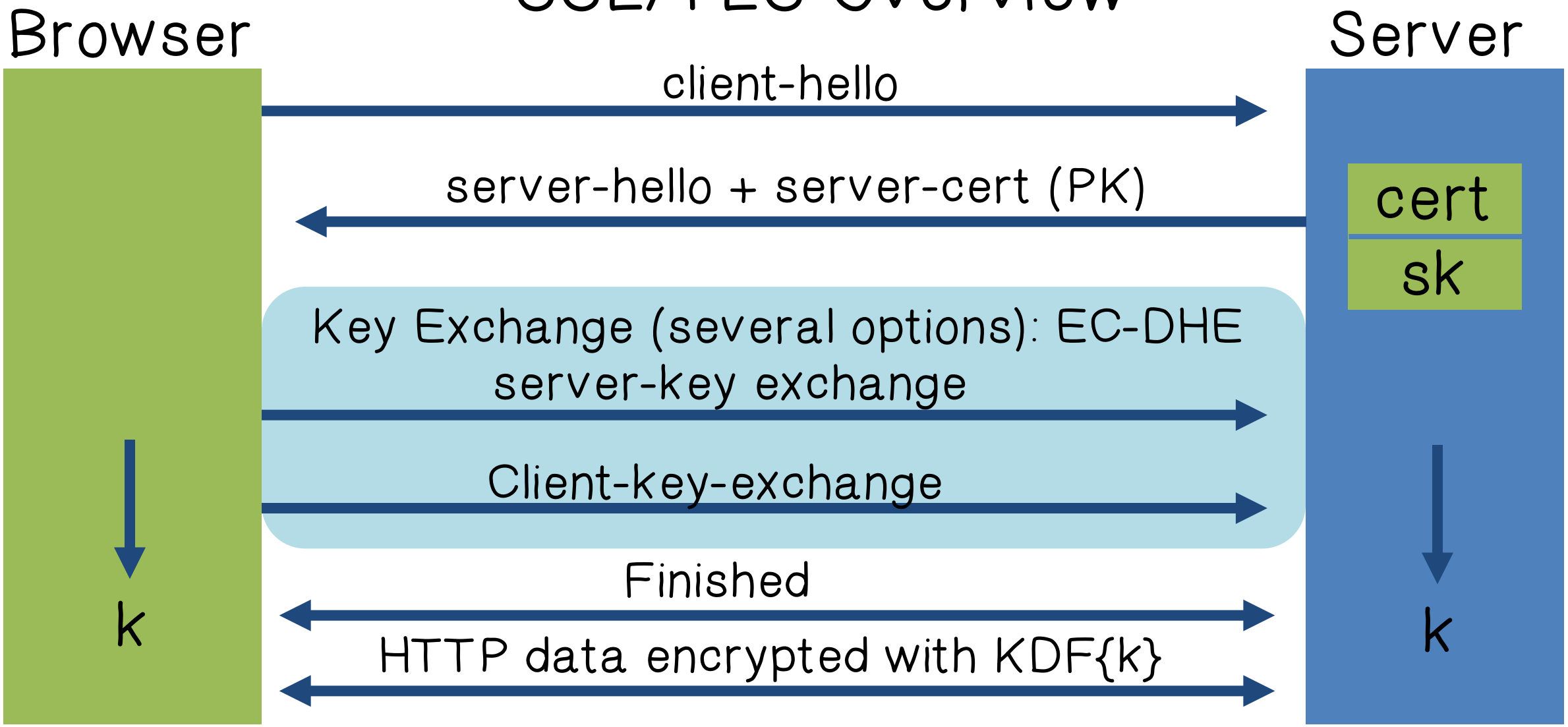
Certificates on the Web

Browsers accept certificates from a large number of CAs:

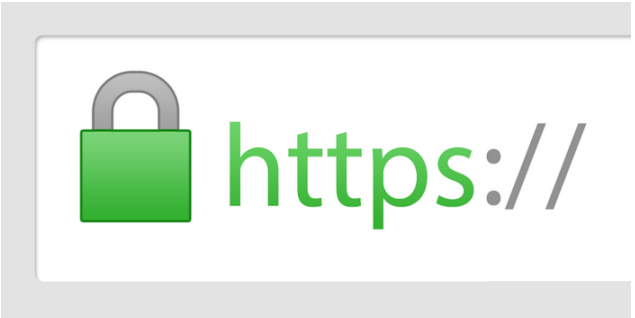
- Top level CAs ≈ 60
- Intermediate CAs ≈ 1200



SSL/TLS Overview



HTTPS in the Browser: The Lock Icon



Intended Goal:

- Provide the user with identity of page origin
- Indicate to user that page contents were not viewed or modified by a network attacker

☞ **In Reality: Many problems**

When is the lock icon displayed?

☞ All elements on the page fetched using HTTPS.

☞ For all elements:

- HTTPS cert issued by a CA trusted by browser
- HTTPS cert is valid (e.g., not expired)

Domain in URL matches:

CommonName or SubjectAlternativeName in cert.

When is the lock icon displayed?

fetched using HTTPS.

Extension	Subject Alternative Name (2.5.29.17)
Critical	No
DNS Name	*.google.com
DNS Name	*.android.com
DNS Name	*.appengine.google.com
DNS Name	*.cloud.google.com
DNS Name	*.google-analytics.com
DNS Name	*.google.ca
DNS Name	*.google.cl
DNS Name	*.google.co.in
DNS Name	*.google.co.jp
DNS Name	*.google.co.uk
DNS Name	*.google.com.ar
DNS Name	*.google.com.au

a CA trusted by browser
e.g., not expired)
nes:
r SubjectAlternativeName

HTTPS Disadvantages Quiz

Which of the following are real disadvantages to using HTTPS

- [ ] Browser caching won't work properly
- [x] You need to buy an SSL certificate
- [x] Mixed modes issue- loading insecure content on a secure site
- [ ] HTTPS uses a lot of server resources
- [x] Proxy caching problems- public caching cannot occur
- [ ] HTTPS introduces latencies

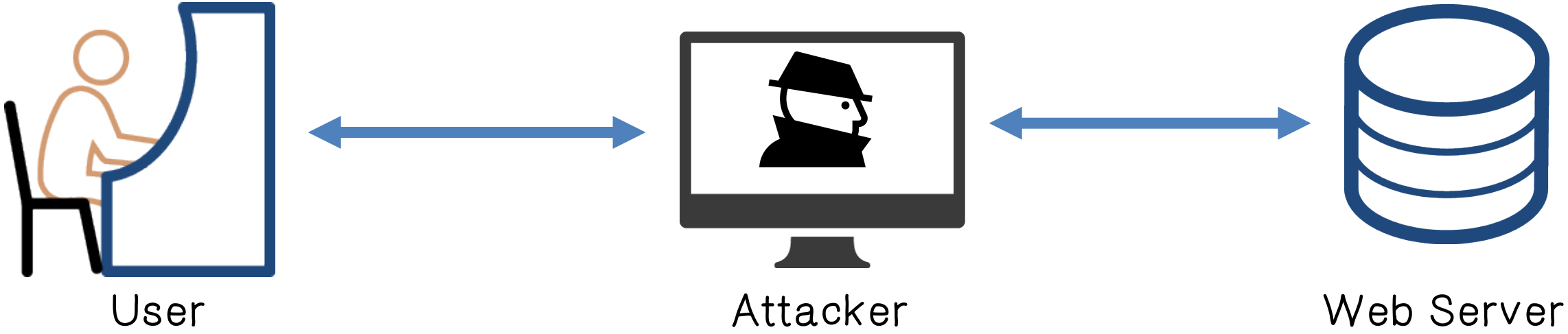
Problems with HTTPS and the lock icon

- Upgrade from HTTP to HTTPS
- Forged certs
- Mixed content: HTTP and HTTPS on the same page

Problems with HTTPS and the lock icon

☞ Upgrade from HTTP to HTTPS

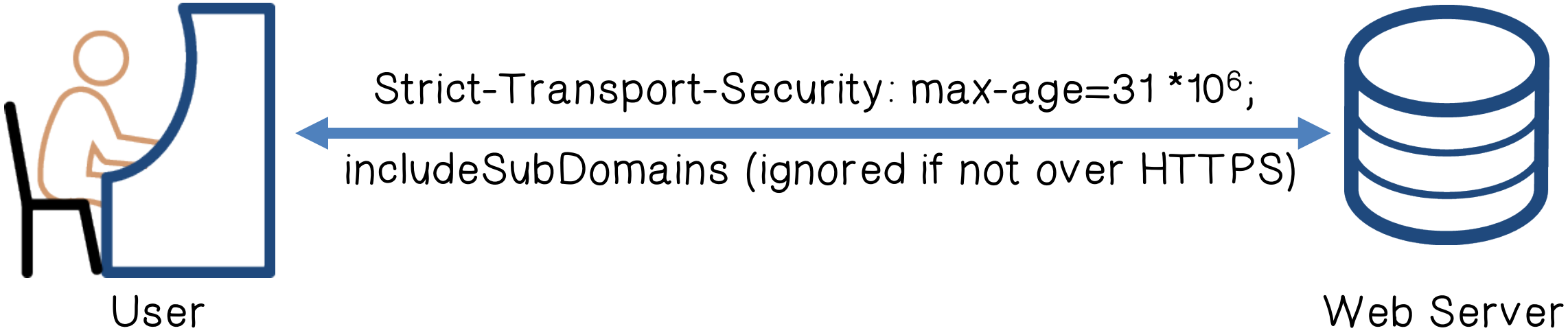
SSL_strip attack: prevent the upgrade [Moxie '08]



Problems with HTTPS and the lock icon

Upgrade from HTTP to HTTPS

Strict Transport Security (HSTS)



Problems with HTTPS and the lock icon

Upgrade from HTTP to HTTPS

Strict Transport Security (HSTS)

- Header tells browser to always connect over HTTPS
- Subsequent visits must be over HTTPS (self signed certs result in an error)
  - Browser refuses to connect over HTTP or if self-signed cert, requires that entire site be served over HTTPS
- HSTS flag deleted when user "clears private data": security vs. privacy

Certificates: wrong issuance

2011

Comodo and DigiNotar CAs hacked, issue certs for Gmail Yahoo!, Mail...

2013

TurkTrust issued cert for gmail.com (discovered by pinning)

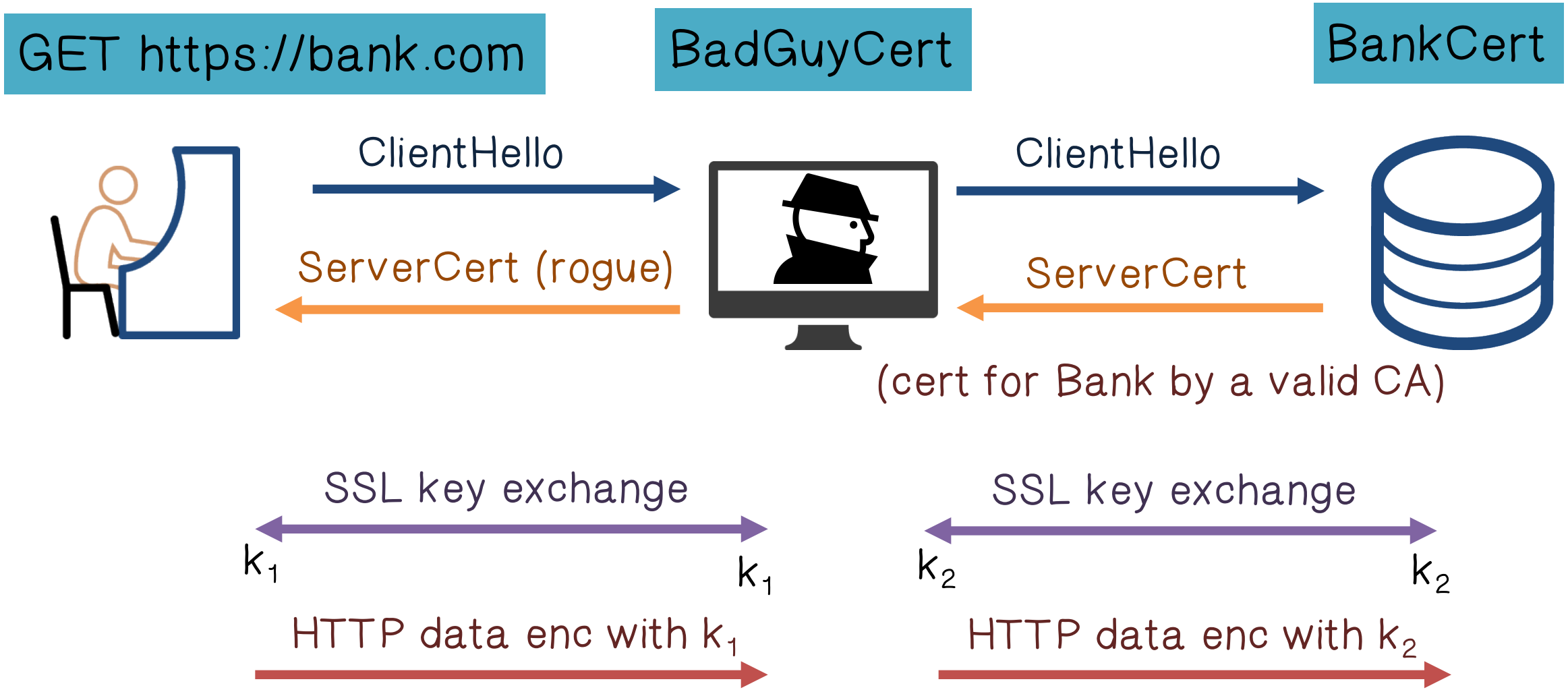
2014

Indian NIC (intermediate CA trusted by the root CA India CCA) issue certs for Google and Yahoo! Domains

2015

MCS (intermediate CA cert issued by CNNIC) issued certs for Google Domains

Man in the middle attack using rogue cert



Solutions to Certificate Issues

1 Dynamic HTTP public-key pinning (RFC 7469)

- Let a site declare CAs that can sign its cert (similar to HSTS)
- On Subsequent HTTPS, browser rejects certs issued by other CAs
- TOFU: Trust on First Use

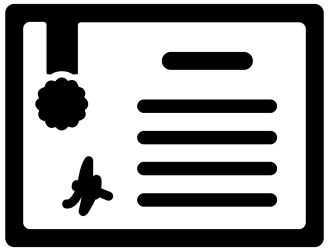
Solutions to Certificate Issues

HPKP example (HTTP header from server)

Public-Key-Pins: max-age=2592000;

- pin-sha256="E9CZ9INDbd+2eRQozYqqbQ2yXLVKB9+xcprMF+44U1g=";
- pin-sha256="LPJNul+wow4m6DsqxbninhsWHlwfp0JecwQzYpOLmCQ=";
- report-uri="https://example.net/pkp-report"

☞ Examine browser's pinning DB: chrome://net-internals/#hsts



Solutions to Certificate Issues

2 Certificate Transparency: [LL'12]

- Idea: CAs must advertise a log of all certs they issued
- Browser will only use a cert if it is published on log server
  - Efficient implementation using Merkle hash trees
- Companies can scan logs to look for invalid issuance