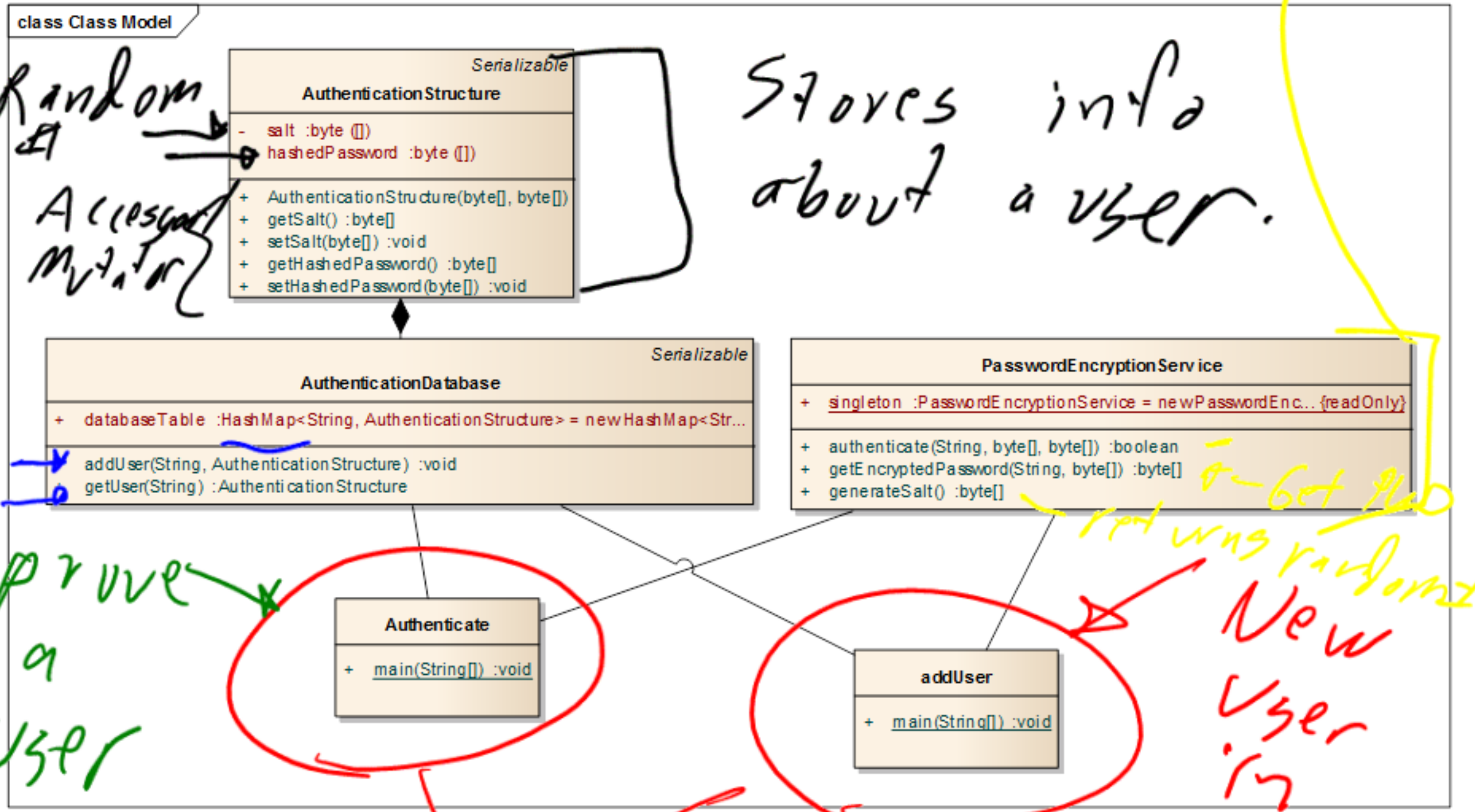
Secure Software Development
Design Principles

Objectives

- Explain the design for an appropriate mechanism for encrypting and storing passwords
- List and explain the Secure Design Principles
- Describe the concept of trust domains and trust boundaries
- Critique architectures based on trust allocations
- Critique a modern software application from a security standpoint

AN Encryption Example



Design Case Study

- A system is setup on campus at MSOE to log failed login attempts
  - Logs entered user name
  - Logs attempted password

⇒ Discloses a lot about passwords

What flaws do we see in design?

tiger ⇒ tgier

User name tiger

password bob

Design Flaws

- Not following coding standards —
- Improper implementation of least privilidge ~ Short fall about
- Software fails insecurely ~
- Authentication methods easily bypassed ⇒ AKA Microsoft BK
- Security through obscurity –
- Improper error handling – Fails unsecu
- Weak input validation –

Cross Scripting

Secure Design Principles

- Principle of Least Privilege
- Separation of Duties
- Defense in Depth
- Fail Secure
- Economy of Mechanisms
- Complete Mediation
- Open design
- Least Common Mechanisms
- Psychological Acceptability
- Leveraging Existing Components

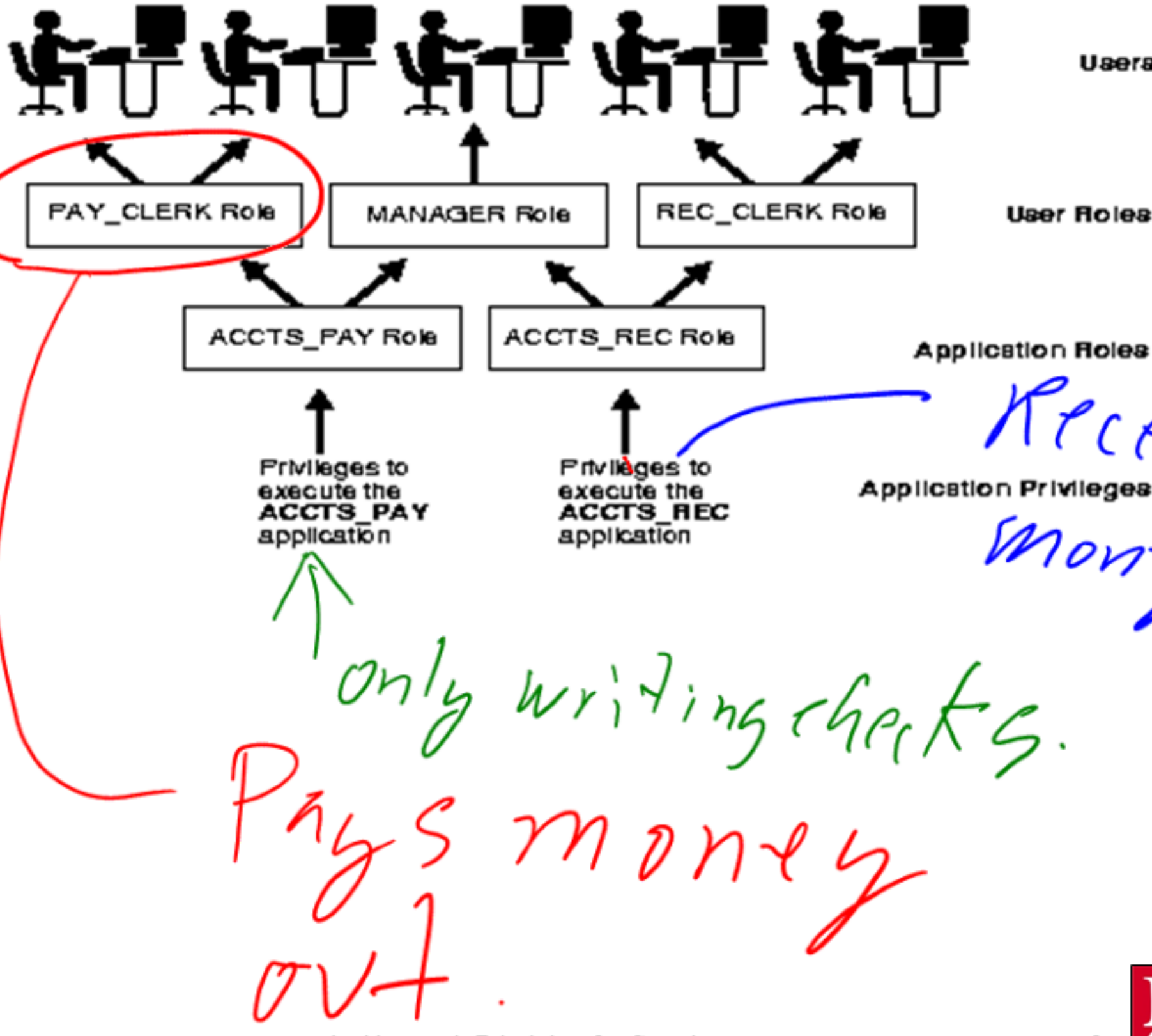
Reuse what others have done.

The Principle of Least Privilidge

The Principle of Least Privilege

- "[The Principle of Least Privilege] requires that each subject in a system be granted the most restrictive set of privileges (or lowest clearance) needed for the performance of authorized tasks. The application of this principle limits the damage that can result from accident, error, or unauthorized use."
  - Department of Defense (DOD-5200.28-STD), also known as the orange book

What does this mean to us in software?



Unix Security Levels and Sudo

Separation of Duties

- Design is compartmentalized
  - Split keys for cryptographic functions
- Development roles
  - Programmer does not review his own code
  - Programmer does not deploy code onto production system

Only have a component do what it needs to do.

Defense in Depth

- Layered defense towards security
  - The breach of a single vulnerability does not result in complete or total system compromise
- Deters curious hacker / nondetermined hacker

- Use of input validation, prepared statements / stored procedures

Security Zones

Access separated based on Zone person is in.

Fail Secure

- Software reliably functions when attacked
- Is rapidly recoverable in the event of a failure
- Fails to a secure state if a failure occurs

True Crypt and fail secure

Economy of Mechanism

- The more complex the design of the software, the more likelihood for a security failure there is
  - Unnecessary functionality or unneeded security mechanisms should be avoided
  - Strive for operational ease of use

Complete Mediation

- Every access to every object must be checked for authority every time the object is accessed.
- **Example 1**
  - When a UNIX process tries to read a file, the operating system determines if the process is allowed to read the file. If so, the process receives a file descriptor encoding the allowed access. Whenever the process wants to read the file, it presents the file descriptor to the kernel. The kernel then allows the access. If the owner of the file disallows the process permission to read the file after the file descriptor is issued, the kernel still allows access. This scheme violates the principle of complete mediation, because the second access is not checked. The cached value is used, resulting in the denial of access being ineffective.

Index of /

https://myweb.msoe.edu/?user=sebern&path=msoe/Winter2011/ce2800/ce2800.shtml

Suggested Sites | Web Slice Gallery | Church Sound Syste... | LXR linux/include/li... | Professional Audio ... | Other bookmarks

Index of /

Index of /

https://myweb.msoe.edu/?user=schilling&path=msoe/Winter2011/ce2800/ce2800.shtml

Suggested Sites | Web Slice Gallery | Church Sound Syste... | LXR linux/include/li... | Professional Audio ... | Other bookmarks

Index of /

Name	Last modified	Size	Description
favicon.ico	08-Jun-2004 17:12	766	
graphics/	01-Nov-2004 10:37	-	
index.html	18-Nov-2010 10:17	1.9K	
local/	08-Apr-2009 13:56	-	
msoe.ico	08-Jun-2004 17:12	766	
robots	06-Jul-2007 16:12	0	
test.cgi	25-Feb-2008 14:22	65	
test.py	25-Feb-2008 14:22	65	

Apache/2.2.8 (Ubuntu) mod_auth_kerb/5.3 DAV/2 SVN/1.4.6 mod_jk/1.2.25 mod_ldap_userdir/1.1.12-20070601 PHP/5.2.4-2ubuntu5.12 with Suhosin-Patch mod_ssl/2.2.8 OpenSSL/0.9.8g mod_perl/2.0.3 Perl/v5.8.8 Server at myweb.msoe.edu Port 443

Complete Mediation Bad Design

Credit Card's Billing Name & Address:

First Name:

Last Name:

Address:

City:

State/Province:

Zip/Postal Code:

Country:

Process Now

(do not click more than once)

Open Design

- All information about crypto systems is public knowledge except the key, and the security of the system against cryptanalysis attacks is dependent on the secrecy of the key
- Not Security through obscurity

Least common mechanisms

- Mechanisms common to more than one user or process should not be shared
  - Design should compartmentalize or isolate the functions by user roles

Psychological Acceptability

- The security principle should be designed to maximize usage, adoption, and automatic application

- Discuss strong passwords as an example

Leverage Existing Components

- Use existing components when possible

Trust Relationship

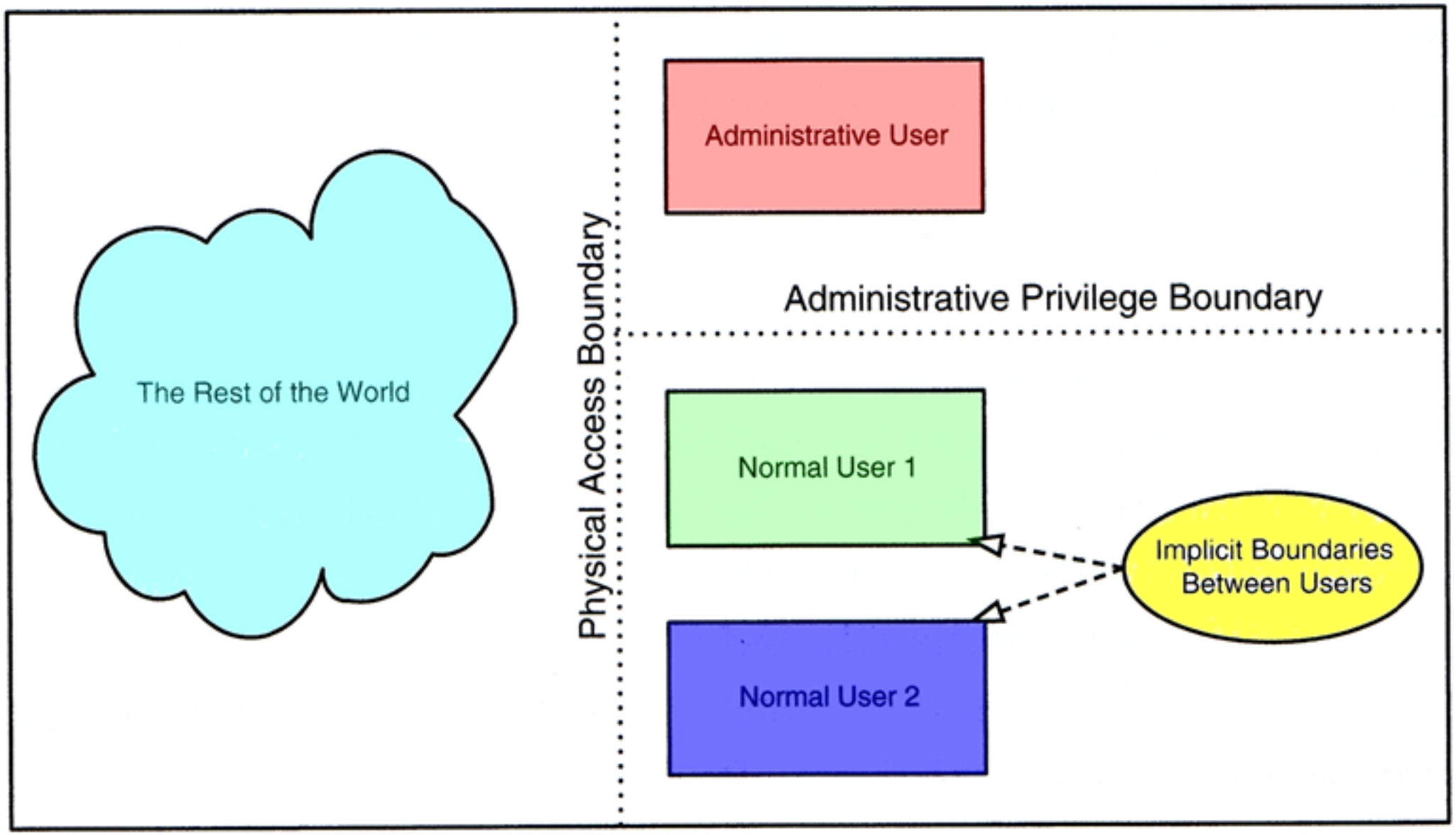
- Every communication between parties must have some degree of trust associated with it
  - Trust relationship
- For simply communications systems, each system has full trust and allows the other complete access to its communication facilities
  - Not very secure

Trust Boundaries

- Distinguishes between regions of shared trust
  - Region of shared trust is a trust domain

Windows 98 Trust Boundary Example

Simple trust relationships



Security Trust – Defense in Depth

- Layering protections so that the compromise of one is mitigated
- Running services and daemons as low privileged accounts
- Isolating different functions to different pieces of hardware
- Demilitarized zones
- Stack and heap guards

Design Gotcha's

- Strong coupling
  - Strong coupling indicates a high level of trust amongst components
  - High exposure of internal interfaces
  - High risk of problems
  - Data validation error prone and difficult
- Strong cohesion
  - Strong cohesion indicates module handles only one specific task

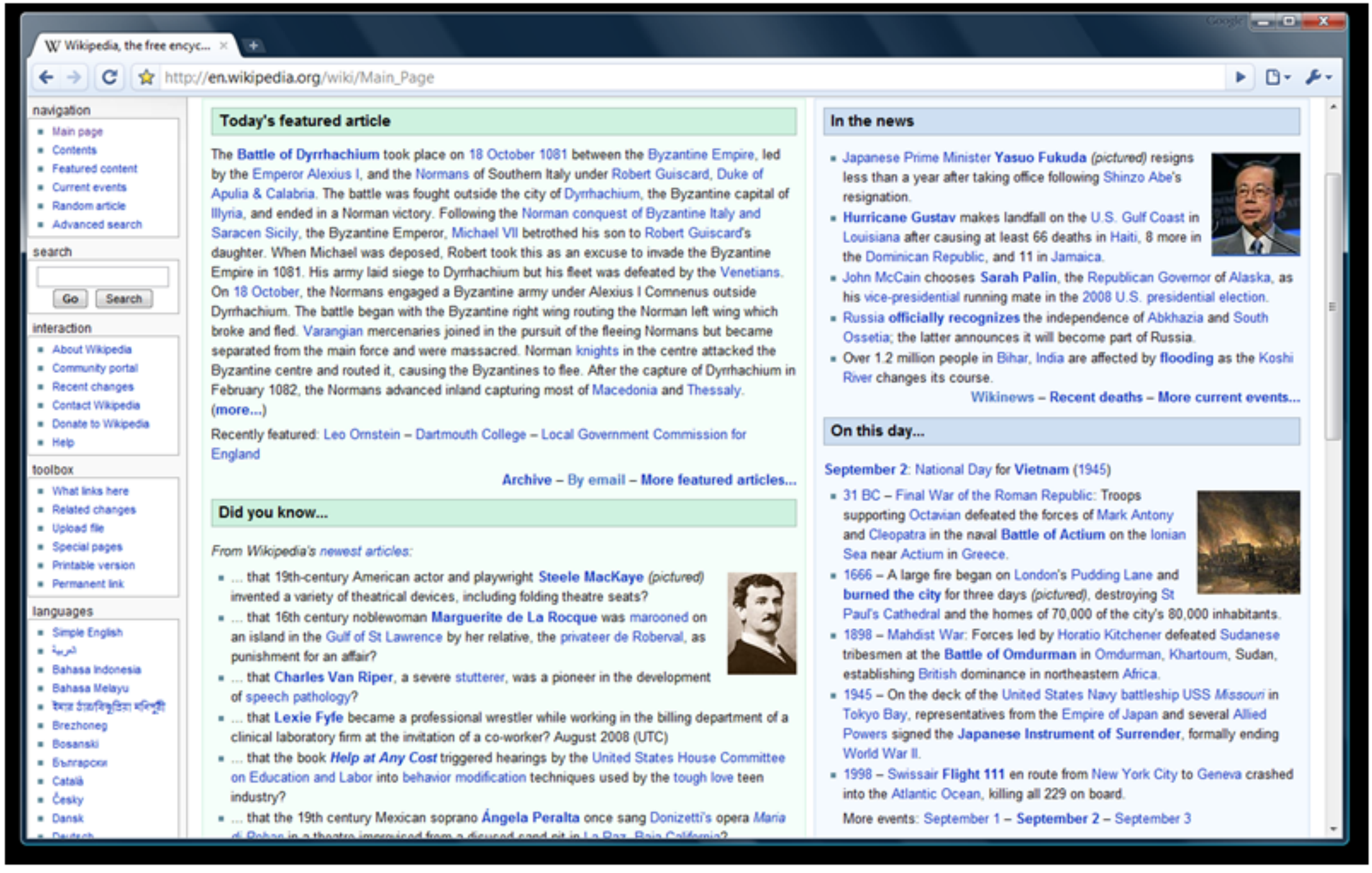
Design Gotcha's

- Modules which cross trust boundaries
  - Design decomposition which fail to decompose modules along trust boundaries

- Shatter class of vulnerabilities

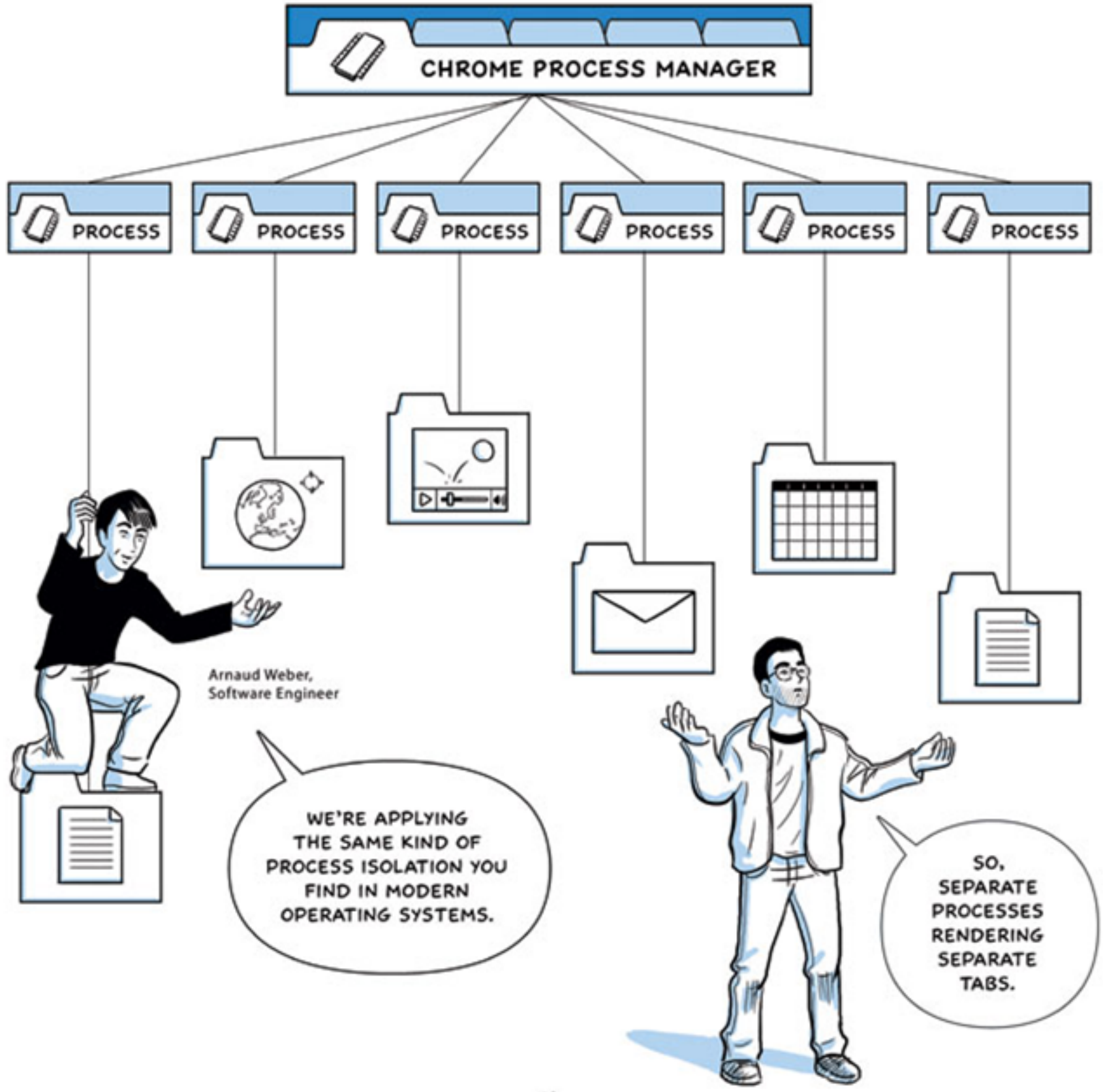
Strong coupling exploit

Google Chrome

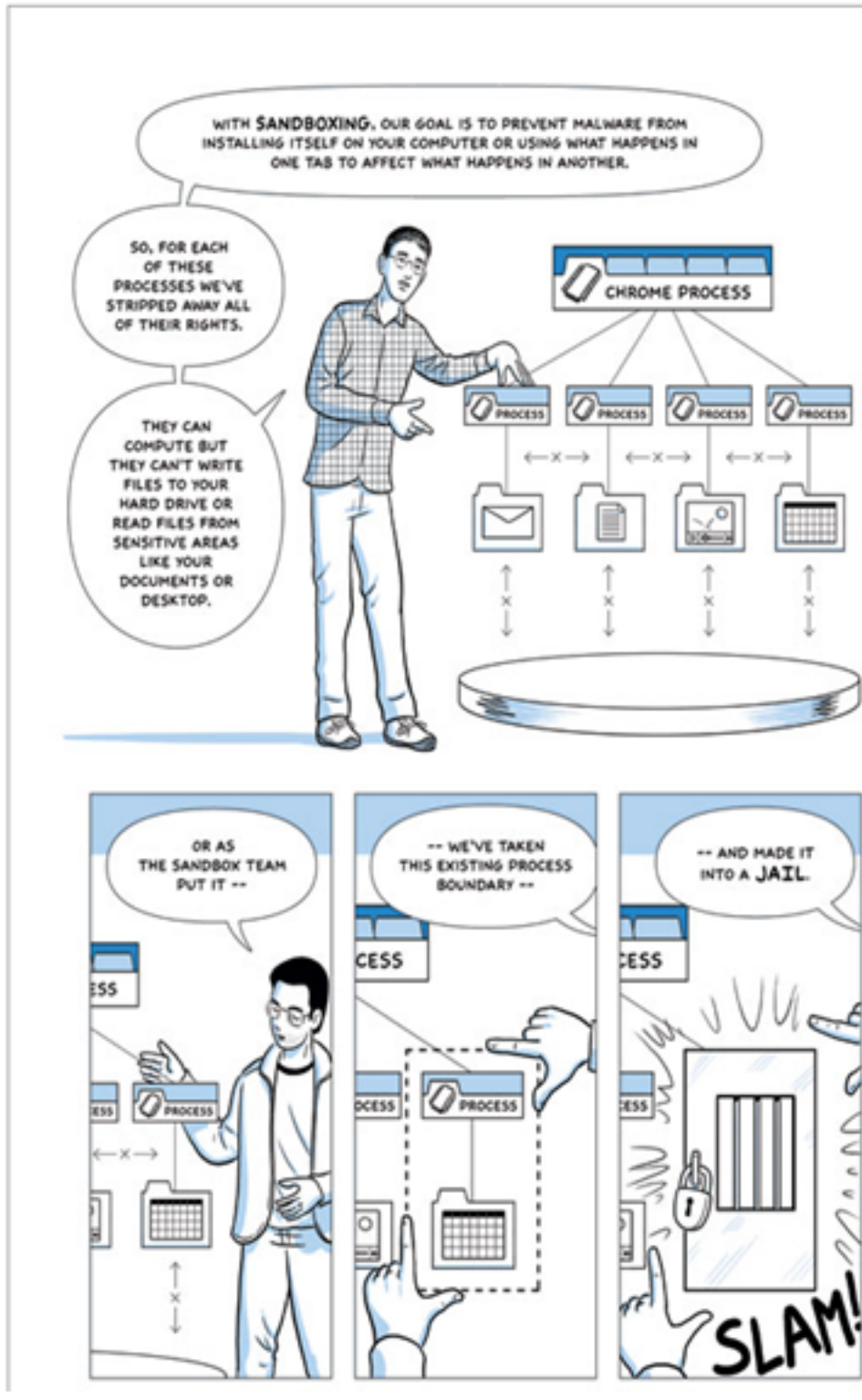


Google Chrome

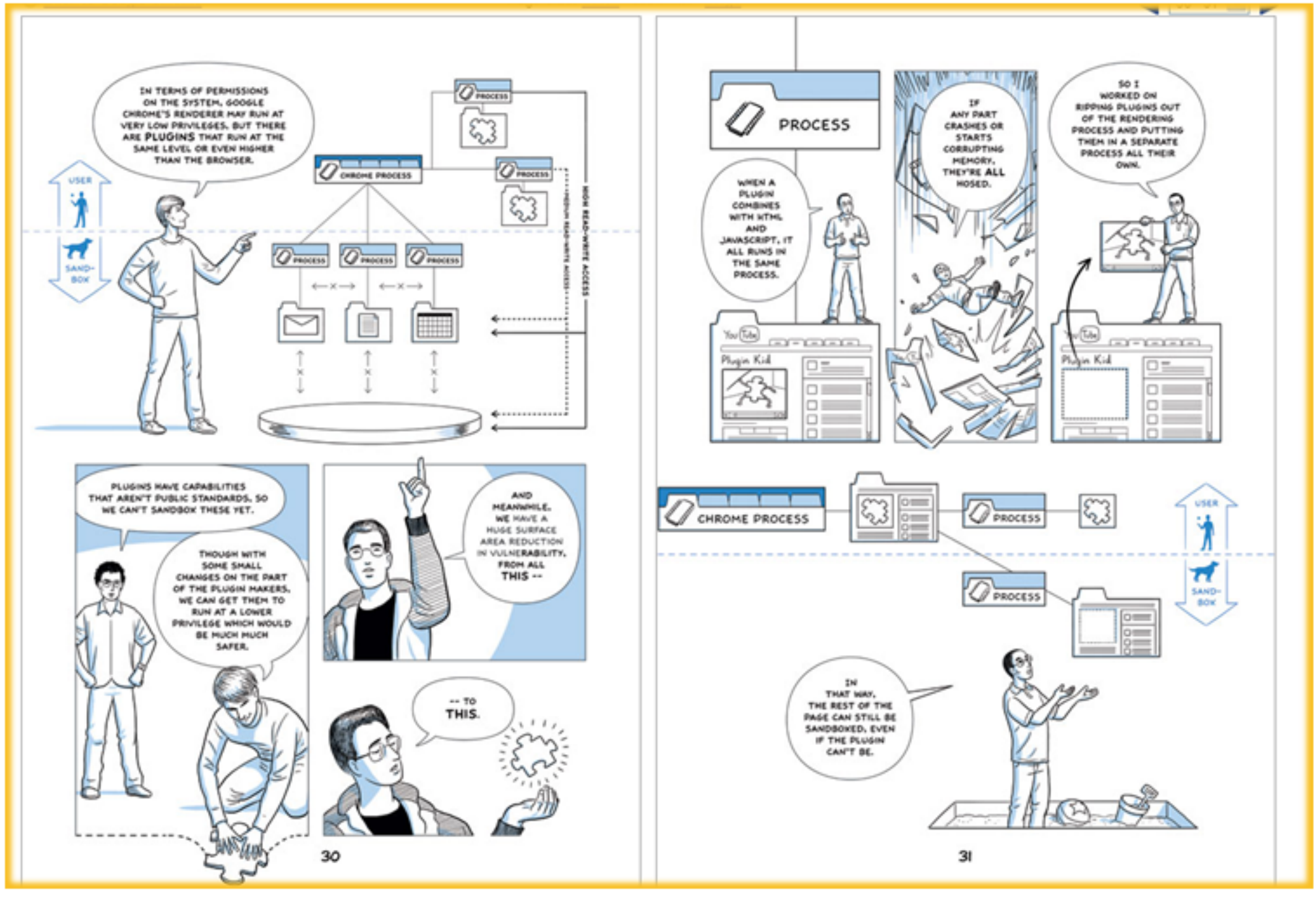
Google Chrome



Google Chrome



Google Chrome




Bank of America | Onli...
2009 Sports Illustrated...
Google Chrome - Wiki...
https://sitekey.bankofamerica.com/sas/signonScreen.do?state=WI
Bank of America
Online Banking
En Español
Sign In
Enter Online ID:
somebody
(6 - 32 characters)
Save this Online ID (How does this work?)
Sign In
Where do I enter my Passcode
Forgot or need help with your ID?
Not using Online Banking?
Enroll now
for Online Banking
Learn more
about Online Banking
Service Agreement
Go to Online Banking for
a state other than Wisconsin
Secure Area
Home . Locations . Contact Us . Help . Sign in . Site Map
Personal Finance . Small Business . Corporate & Institutional
About the Bank . In the Community . Finance Tools & Planning . Privacy
& Security
Bank of America, N.A. Member FDIC. Equal Housing Lender
©2009 Bank of America Corporation. All rights reserved.
USA
start
2 Firefox
Windows Task...
Presentation1
Lecture6Archit...
Jasc Paint Sho...
Bank of Ameri...
9:59 PM


Bank of America | Onli...
2009 Sports Illustrated...
Google Chrome - Wiki...
http://sportsillustrated.cnn.com/2009_swimsuit/
MODELS
NBA DANCERS
TENNIS STARS
ON LOCATION
VIDEO
SWIMSUIT GOODIES
VIDEO LINEUP
ALL VIDEO
ARIEL MEREDITH
CHENEY LARSCHIED
MELISSA HARO
ALISON PRESTON
DANIELA HANTUCHOVA
Brooklyn
DECKER
VIDEO
PHOTOS
ALL
MODELS
Brooklyn Decker was photographed by Raphael Mazzucco in Canouan Island, The Grenadines. Swimsuit by Sugarwater.
SWEEPSTAKES
Sports Illustrated
start
2 Firefox
Windows Task...
Presentation1
LectureArchit...
Jasc Paint Sho...
2009 Sports Il...
10:02 PM

Google Chrome

- Each tab is its own process
  - Not thread
  - "prevent malware from installing itself" or "using what happens in one tab to affect what happens in another",
- Can not write files or read from sensitive areas (e.g. documents, desktop)
- two levels of security, user and sandbox
  - *sandbox* can only respond to communication requests initiated by the *user*.[34]
- Plugins are run in separate processes
  - communicate with the renderer in dedicated per-tab processes.[1]
- *Incognito* mode prevents the browser from storing any history information or cookies
  - Referred to as a porn mode