Secure Software Development
More Design

Architecture Review

Objectives

Explain the relationship between client server and peer to peer architectures and the security risks associated with each system

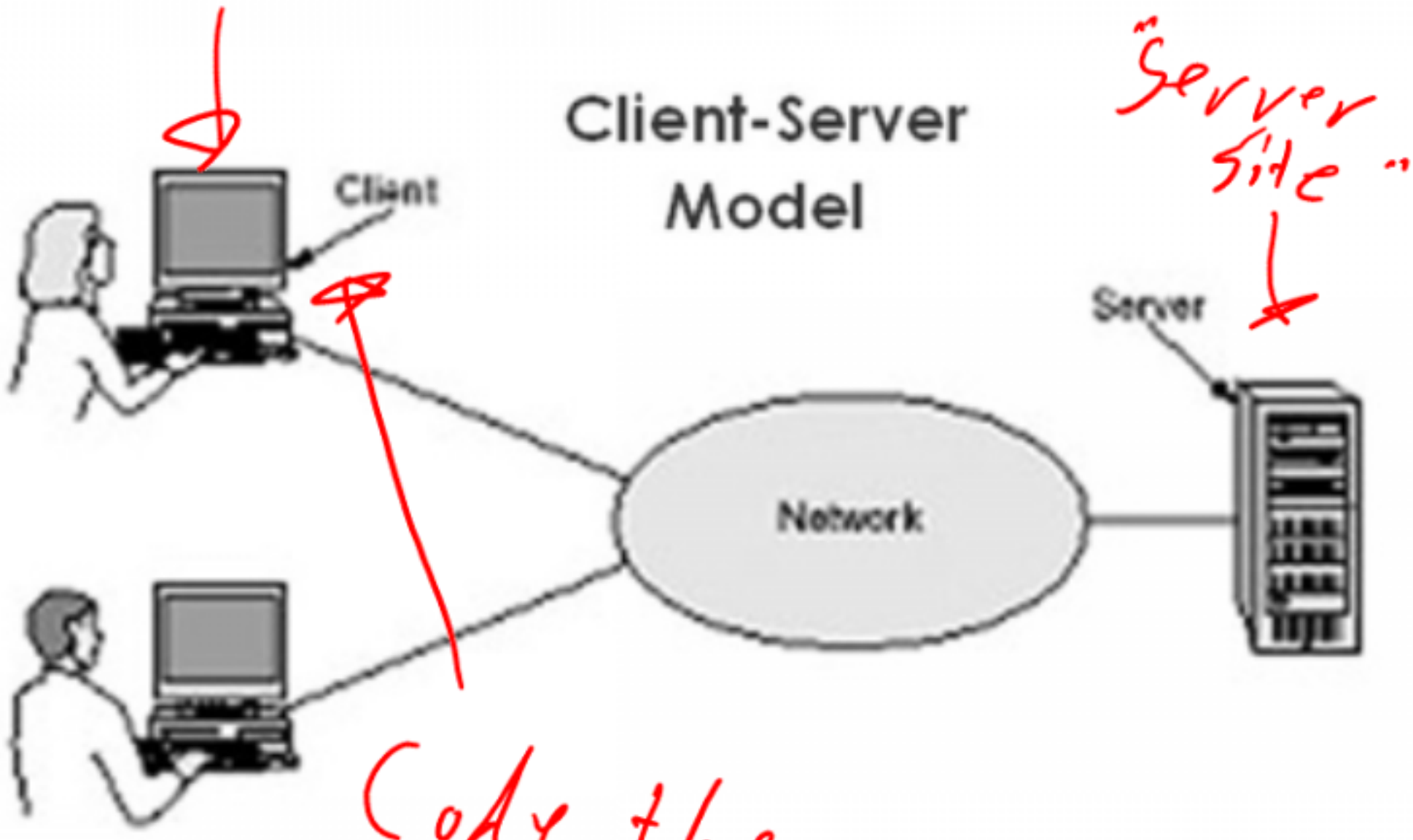
Explain the concept of a thin client

Explain the concept of a fat client

Define an N tier software architecture

Explain the problems with a 1 tier architecture

Client Server Model



"web browser"

"Server side"

Code the runs the UI & other stuff

Thick versus Thin Clients

- Thick client
  - Client is very complex and has lots of logic in it.
  - Client is responsible for many issues related to the software product
- Thin client
  - Client has very little complexty.
  - Most complexity is on the server.

Thick versus Thin Clients

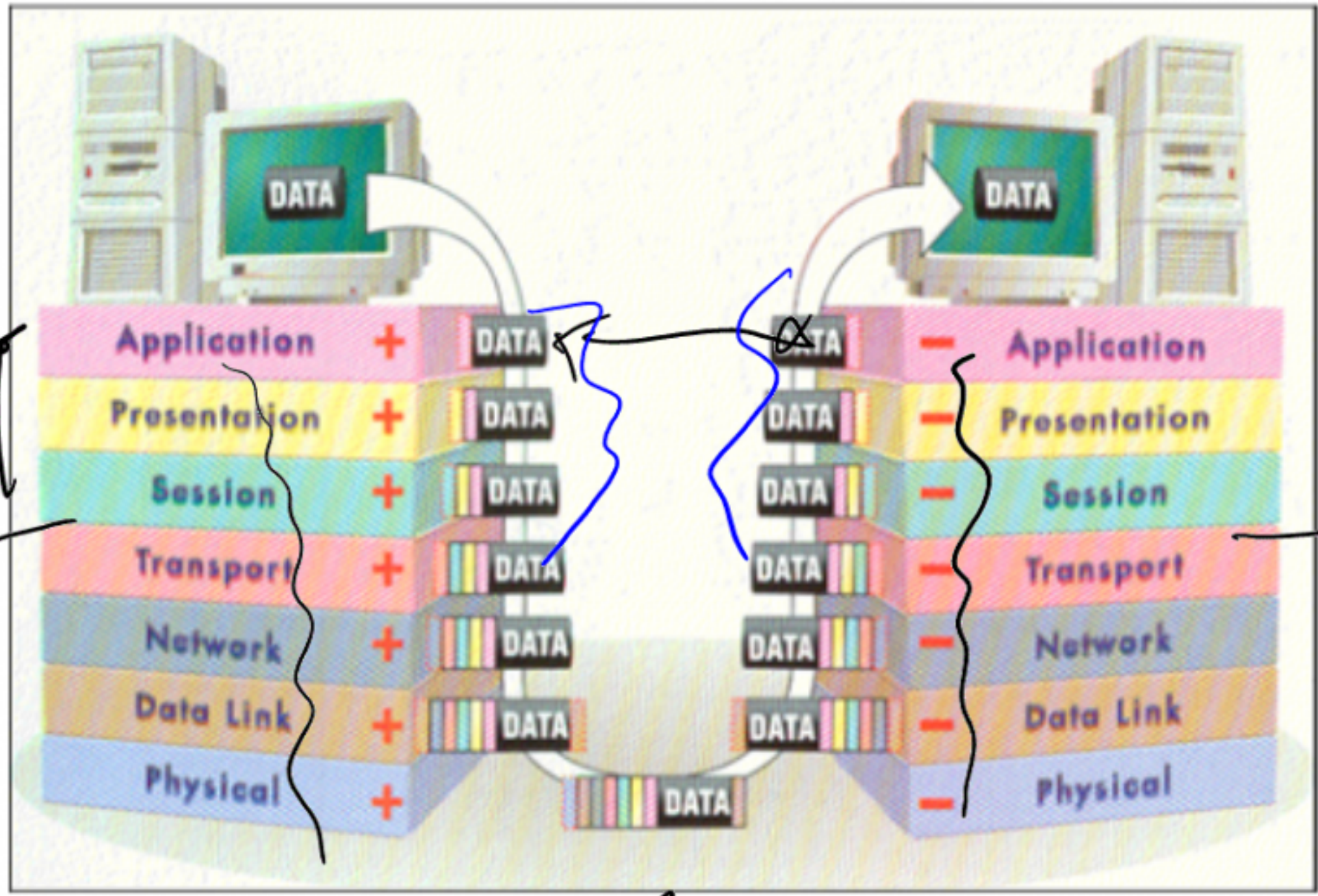
- Thick client
  - Client is very complex and has lots of logic in it.
  - Client is responsible for many issues related to the software product
- Thin client
  - Client has very little complexty.
  - Most complexity is on the server.

Peer to Peer

There is no "server". All machines are @ the same level.

⇒ How do I validate who the peer is?

OSI Layers


DATA
DATA
Application
Presentation
Session
Transport
Network
Data Link
Physical
DATA
DATA
DATA
DATA
DATA
DATA
DATA
DATA
DATA
DATA
Application
Presentation
Session
Transport
Network
Data Link
Physical

3 tier architecture

Presentation tier

The top-most level of the application is the user interface. The main function of the interface is to translate tasks and results to something the user can understand.

Logic tier

This layer coordinates the application, processes commands, makes logical decisions and evaluations, and performs calculations. It also moves and processes data between the two surrounding layers.

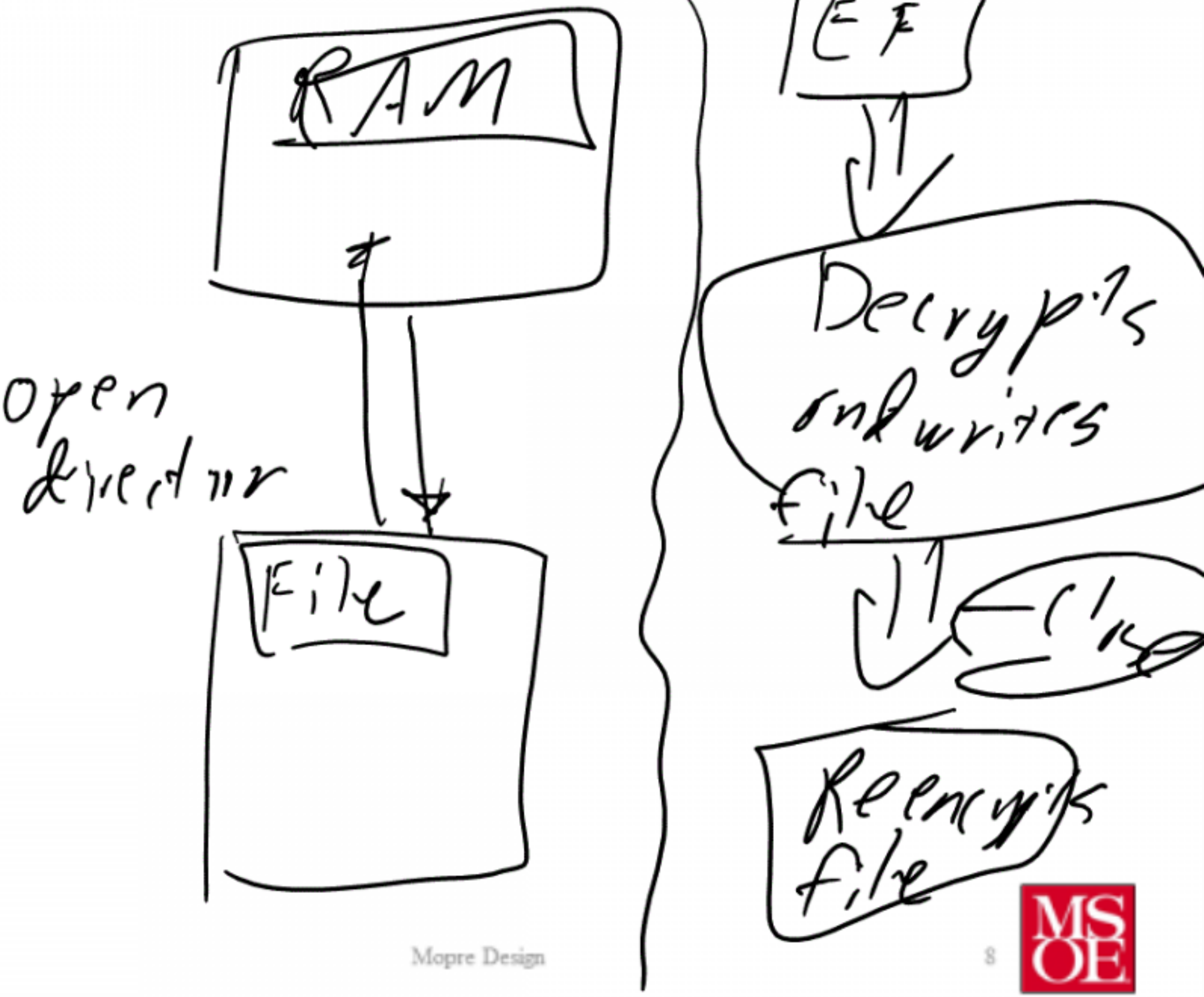
Data tier

Here information is stored and retrieved from a database or file system. The information is then passed back to the logic tier for processing, and then eventually back to the user.

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 Media 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

LO Don't reinvent the wheel

Rate the Threats

- Determine the risk of each threat that is posed
- Risk = Probability * Damage Potential

DREAD

- **D**amage potential: How great is the damage if the vulnerability is exploited?
- **R**eproducibility: How easy is it to reproduce the attack?
- **E**xploitability: How easy is it to launch an attack?
- **A**ffected users: As a rough percentage, how many users are affected?
- **D**iscoverability: How easy is it to find the vulnerability?

Dread Rubric

	Rating	High (3)	Medium (2)	Low (1)
D	Damage potential	The attacker can subvert the security system; get full trust authorization; run as administrator; upload content.	Leaking sensitive information	Leaking trivial information
R	Reproducibility	The attack can be reproduced every time and does not require a timing window.	The attack can be reproduced, but only with a timing window and a particular race situation.	The attack is very difficult to reproduce, even with knowledge of the security hole.
E	Exploitability	A novice programmer could make the attack in a short time.	A skilled programmer could make the attack, then repeat the steps.	The attack requires an extremely skilled person and in-depth knowledge every time to exploit.
A	Affected users	All users, default configuration, key customers	Some users, non-default configuration	Very small percentage of users, obscure feature; affects anonymous users
D	Discoverability	Published information explains the attack. The vulnerability is found in the most commonly used feature and is very noticeable.	The vulnerability is in a seldom-used part of the product, and only a few users should come across it. It would take some thinking to see malicious use.	The bug is obscure, and it is unlikely that users will work out damage potential.

- Attacker obtains authentication credentials by monitoring the network.
- SQL commands injected into application.

Example

Threat	D	R	E	A	D	Total	Rating
Attacker obtains authentication credentials by monitoring the network.	3	3	2	2	2	12	High
SQL commands injected into application.	3	3	3	3	2	14	High

- Mitigate the risks…

Afterwards

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 ← Bad
  - 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

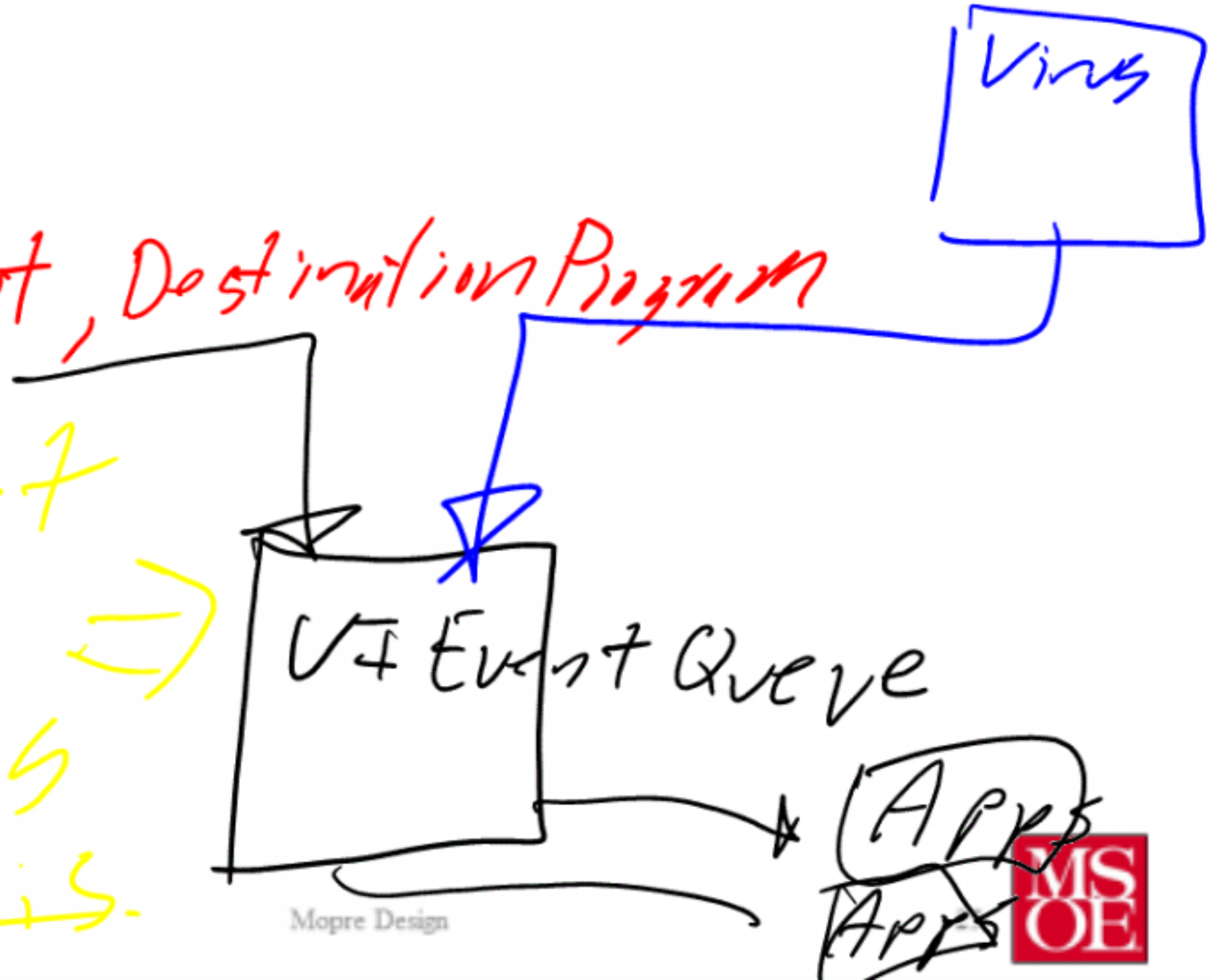
- Modules which cross trust boundaries
  - Design decomposition which fail to decompose modules along trust boundaries

BAD

- Shatter class of vulnerabilities

=> Windows 95 (I think)

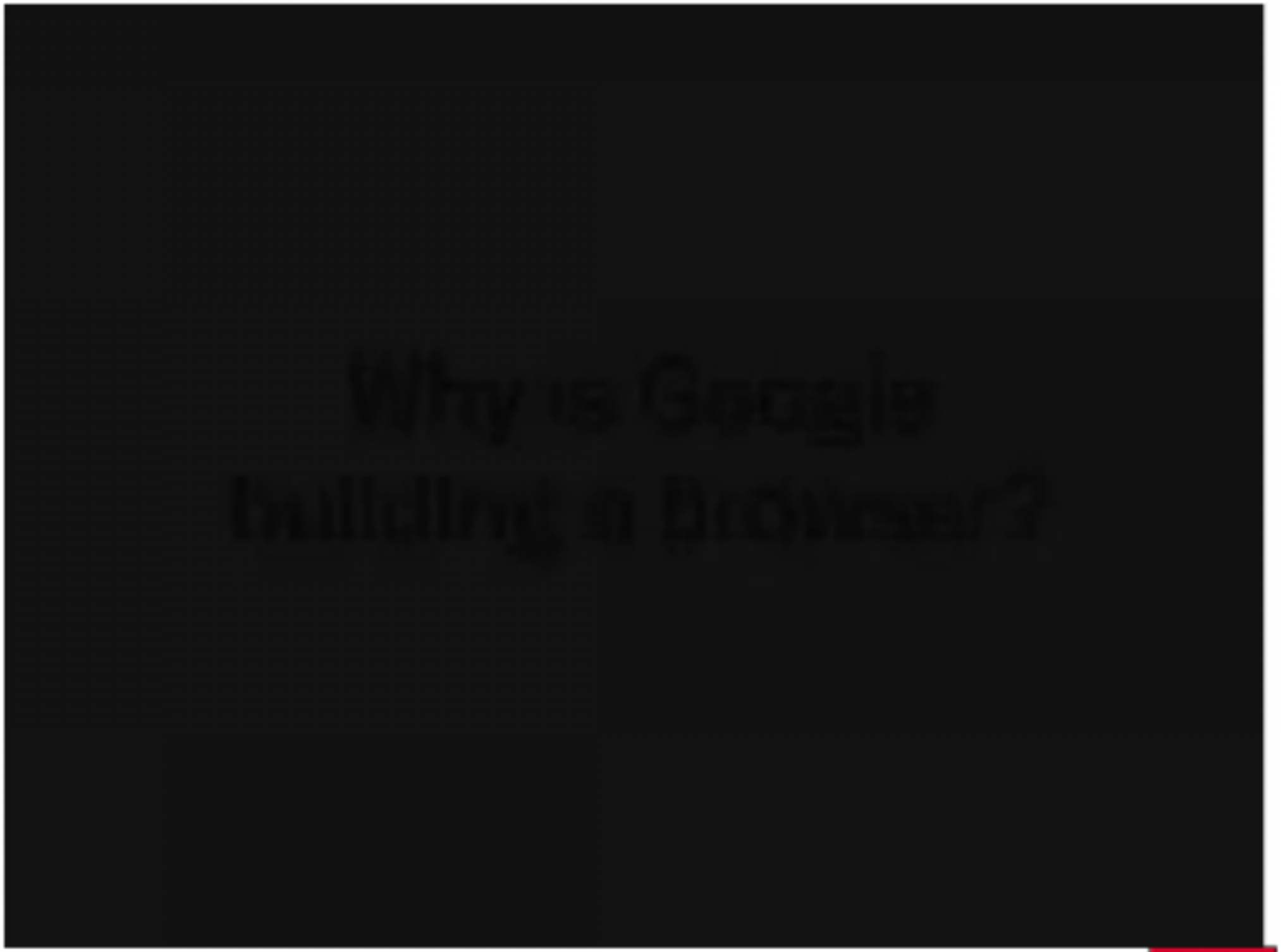
Strong coupling exploit



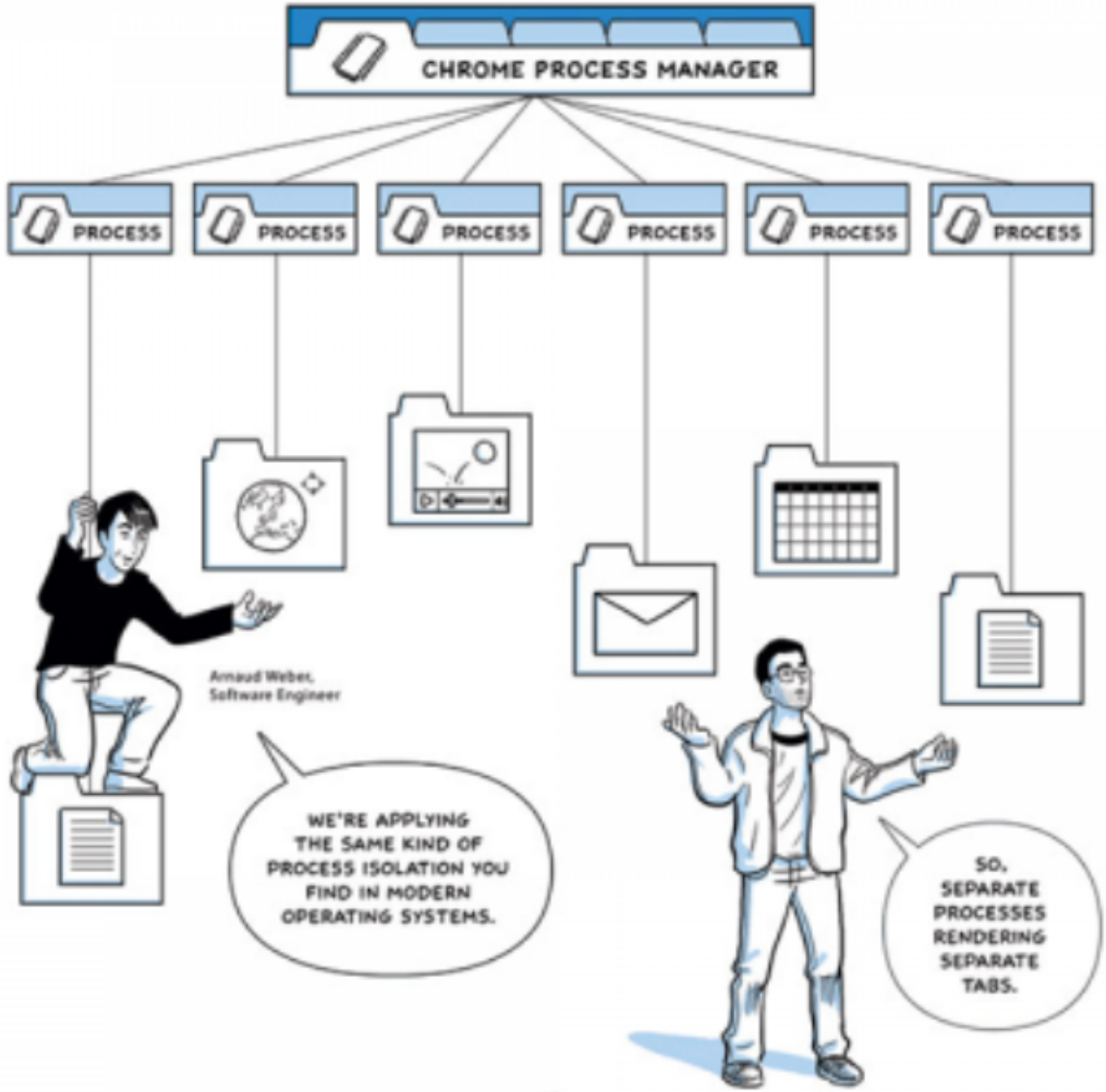
Must do things like this.

Google Chrome

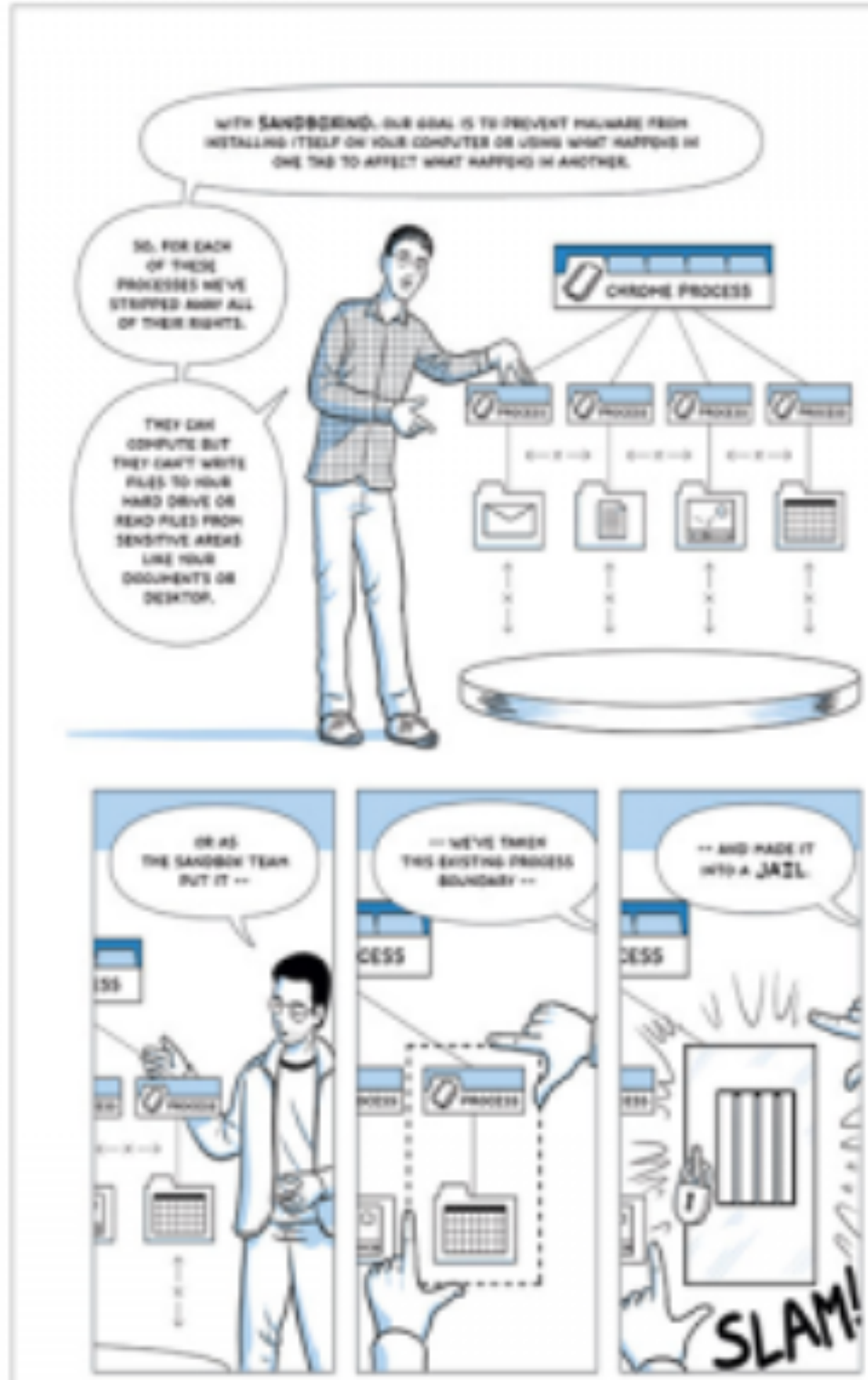
Google Chrome



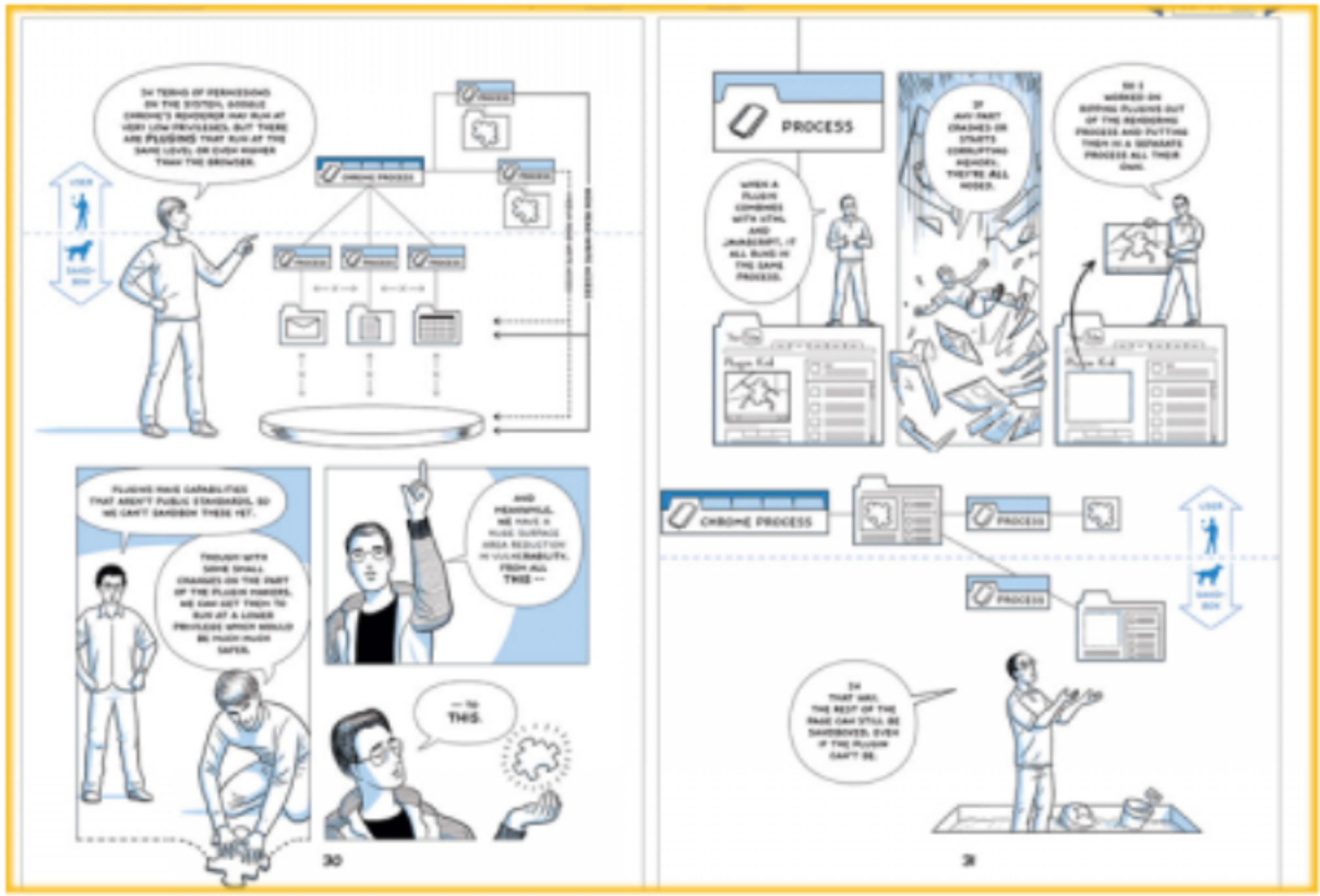
Google Chrome



Google Chrome



Google Chrome




Bank of America
Online Banking
Sign In
Enter Online ID:
somebody
Save this Online ID
Sign In
Where do I enter my Passcode
Not using Online Banking?
Learn more
about Online Banking
Service Agreement
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.
start


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
SWEEPSTAKES
start

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