Warm-Up Exercise …

1. Write down a list of technologies you are using in your application (e.g., database systems, external libraries, platforms you rely on, etc.)

2. For each, determine what security issues/vulnerabilities exist or have been exploited in the past
Liabilities and Protections

**Liability:** what is the software maker “responsible” for in terms of “harms” done to users/consumers?

**Protection:** what consumer protections are there in terms of “harms” done to them?
Legal Issues and Responsibilities …

We’ll focus on liability

- Protection usually refers to legal ability to recoup losses
- Although protection & liabilities aren’t always opposites
- E.g., banks and FDIC

Legal Liabilities

Involved parties in software damages

- Software “producers” … company, vendor, developers
- Software “consumers” … end users, clients, intermediaries
- Third parties, e.g., cyber hackers, thieves, governments

Can be complex: e.g., using Amazon EC2 in your app
Liability depends on type of damage

Broad areas of damages …

- Software that breaks laws (e.g., theft, national security, …)
- Software that results in personal harm (e.g., GPS case)
- Quality issues that lead to direct/indirect financial/privacy loss

We’ll focus on financial/privacy loss which is largely:

**Buyer Beware!**

The state of software liability

With Buyer Beware ...

- Software producers have limited liability
- Software consumers shoulder majority of damages

There are a number of reasons for this …
Why buyer beware?

1. Lack of legislation (“legislative void”)
   - Few laws specific to software
   - Some do exist: CAN-SPAM act of 2003, State data-breach notification laws, CFAA (computer fraud & abuse act)

Why buyer beware?

2. Disputes often fall under “contract law”
   - as opposed to criminal or tort law
   - Tort law requires more than “purely economic damages”
     - typically physical damage (personal or property)
Why buyer beware?

- Purchased software protected under the UCC
  - 1952 Uniform Consumer Code
  - Not a law, but a suggestion to states
  - Has been adopted by most states

Courts, however, have generally allowed software providers to "contract away" liabilities for product deficiencies (via the UCC)

Why buyer beware?

- Most software today is "licensed" and not "purchased"
  - i.e., users don't actually own the software (e.g., GM)
  - "End User License Agreement" (EULA) … downloaded
  - "Terms of Service" (TOS) … e.g., web sites

Courts also generally allow software licenses to disclaim all liabilities for product deficiencies

www.autoblog.com/2015/05/20/general-motors-says-owns-your-car-software/
Why buyer beware?

3. Free software (not directly purchased) is not considered under contract law (“considerations”)
   - No buyer and seller to form a proper contract

Some examples ...

Why buyer beware?

LinkedIn had a class action suit brought against it after hackers caused a data breach, releasing 6.5 million password hashes, 60% of which were later cracked.

- Plaintiffs argued LinkedIn failed to utilize industry standard techniques to protect data, in violation of its own User Agreement and Privacy Policy.
- A federal court in California threw out the case in part on the grounds that the policy was the same for users of the free and premium versions of the service.
- The court found that the complaint “fails to sufficiently allege that Plaintiffs actually provided consideration for the security services which they claim were not provided.”
Why buyer beware?

Facebook was sued for sharing data with third parties, which violated their own published policies.

- The judge ruled:

  “[A] plaintiff who is a consumer of certain services (i.e., who paid fees for those services) may state a claim under certain California consumer protection statutes when a company, in violation of its own policies, discloses personal information about its consumers to the public … Here, by contrast, Plaintiffs do not allege that they paid fees for Defendant’s services.”

Why buyer beware?

4. Other reasons …

- some courts have treated security expectations as inherently unreasonable (since software cannot be made bug free)
- some courts reject claims since 3rd party hackers create the issue, and the vendors don’t benefit from these attacks (e.g., Sony employee data breach case)
End User License Agreements (EULAs)

Shrinkwrap, Clickwrap, Browserwrap agreements

- All represent a contract between a buyer and seller
- To be a valid contract ...
  - Must be conspicuous
  - Must be an offer (seller) and acceptance (buyer)
  - Must be considerations
  - “Meeting of the minds” (know legally binding, shared understanding of terms)
  - Plus others (e.g., legal activities)

Examples of EULAs not passing as contracts

Zappos.com suffered a security breach and was sued
  - The company motioned for arbitration because the TOS on its website required all disputes to be “submitted to confidential arbitration in Las Vegas, Nevada.”
  - Part of a browserwrap agreement that users were not required to affirmatively “accept” (a link was located in the footer of the homepage you had to scroll to see)
  - The District of Nevada Court concluded that plaintiffs never viewed, let alone manifested assent to, the TOS, and held that the arbitration provision was unenforceable

Similar case with Barnes & Noble (suit against cancelled orders)
The two extremes:

1. Buyer beware
2. Hold software producers responsible

Where we seem to be today:

- Somewhat murky middle ground
- But (slowly) heading towards 2?

Some history: Late 1990’s UCITA

- Uniform Computer Information Transactions Act
- To deal with issues with the Uniform Consumer Code
- Proposed making software producers automatically liable for defects/errors

Supporters: large software companies (Microsoft, Adobe, …)

Opposers: IEEE, ACM, Free Software Foundation, FTC, …

Why? …
Why was UCITA opposed?

- Allowed shrinkwrap licenses to override UCITA provisions
- But free software could not override UCITA provisions
- Weakened consumer protections
  - Allowed licensing over purchases
  - Which weakens consumer protections

Only passed in two states, but with amendments
Some states passed anti-UCITA laws

Arguments against Buyer Beware:

1. Software is critical/essential part of people’s lives
2. Most users can’t make security decisions (sophistication)
3. Anti-virus software can’t keep up …
   - Already a multi-billion dollar industry
   - ~15% of antivirus software is fake / malware
Arguments against Buyer Beware:

4. Economics of software is build first, patch later
   - Places burden on users (who must apply patches)
   - E.g., 52% of IE users don’t use latest version

5. Without incentives for quality, situation becomes worse …
   - Which increases the potential damages
   - High-profile cases increases the cost on software makers
   - Which goes against the buyer beware model

Arguments for software producers taking responsibility:

1. Software doesn’t follow standards applied to other industries
   - E.g., as a renter, if you don’t provide industry standard locks, you can be held liable for damages

2. Many of the software problems are easy to fix
   - The marketplace just doesn’t reward it
Liability “extremes”

Arguments for software producers taking responsibility:

3. Not requiring responsibility translates to government subsidy
   - Companies can produce more products faster/cheaper

4. Other countries moving in this directly already ...

Opinions can change quickly ...

The automotive industry as an example

- In the 60’s, courts reluctant to apply tort laws for defects
- Over next 30 years, did an “about face”
- Started in 1966 with the National Traffic and Motor Vehicles Safety Act
- Which gave a federal agency power to create and enforce industry safety standards
Costs are beginning to rise ...

Thousands of data breaches every year! (+ undetected)

In 2016, per company

- Avg total cost of a data breach in US is $7 million
- Avg cost of detecting a breach is $1.6 million
- Avg cost of notification is $590,000 (laws for this)
- Avg cost of lost business is $3.9 million
- Avg cost of handling breach $1.7 million (incl. legal costs)
- 24% chance of a data breach within 24 mo period

* 2016, Ponemon Institute LLC and IBM