Reverse Engineering

- **“Design recovery** is a subset of reverse engineering in which domain knowledge, external information, and deduction or fuzzy reasoning are added to the observations of the subject system."
- The objective of design recovery is to identify meaningful higher-level abstractions beyond those obtained directly by examining the system itself
- [ElliotChikofsky and JamesCross, Reverse Engineering and Design Recovery: A Taxonomy, IEEE Software 7(1):13-17, 1990.]

Reverse Engineering

- Reverse engineering often precedes re-engineering but is sometimes worthwhile in its own right
  - The design and specification of a system may be reverse engineered so that they can be an input to the requirements specification process for the system's replacement
  - The design and specification may be reverse engineered to support program maintenance and reengineering

Reverse Engineering as a process

- Like any other activity, reverse engineering is also a process
- There is a guide that we can follow to help us generate information that
- Can be helpful to both the analyst and stakeholders

Reverse Engineering as a process

- Seeking approval
  - Ethics requires anyone carrying out reverse engineering of software to have approval from the owner of the software
  - Some companies are more lenient about their software getting reversed without approval, but it is customary today that any vulnerabilities found should be reported directly to the owner and not publicized
  - It is up to the owner to decide when to report the vulnerability to the community
  - This prevents attackers from using a vulnerability before a software patch gets released

Reverse Engineering as a process

- Seeking approval
  - It is a different story when malware or hacking is involved
  - Of course, reversing malware doesn't need approval from the malware author
  - Rather, one of the goals of malware analysis is to catch the author
  - If not sure, always consult a lawyer or a company's legal department

Reverse Engineering as a process

- Static Analysis
  - Without any execution, viewing the file's binary and parsing each and every byte provides much of the information needed to continue further
  - Simply knowing the type of file sets the mindset of the analyst in a way that helps them to prepare specific sets of tools and references that may be used
  - Searching text strings can also give clues about the author of the program, where it came from, and, most likely, what it does

Reverse Engineering as a process

- Dynamic Analysis
  - This type of analysis is where the object being analyzed gets executed
  - It requires an enclosed environment so that behaviors that may compromise production systems do not happen
  - Setting up enclosed environments are usually done using virtual machines, since they can then easily be controlled
    Tools that monitor and log common environment actions are implemented during dynamic analysis