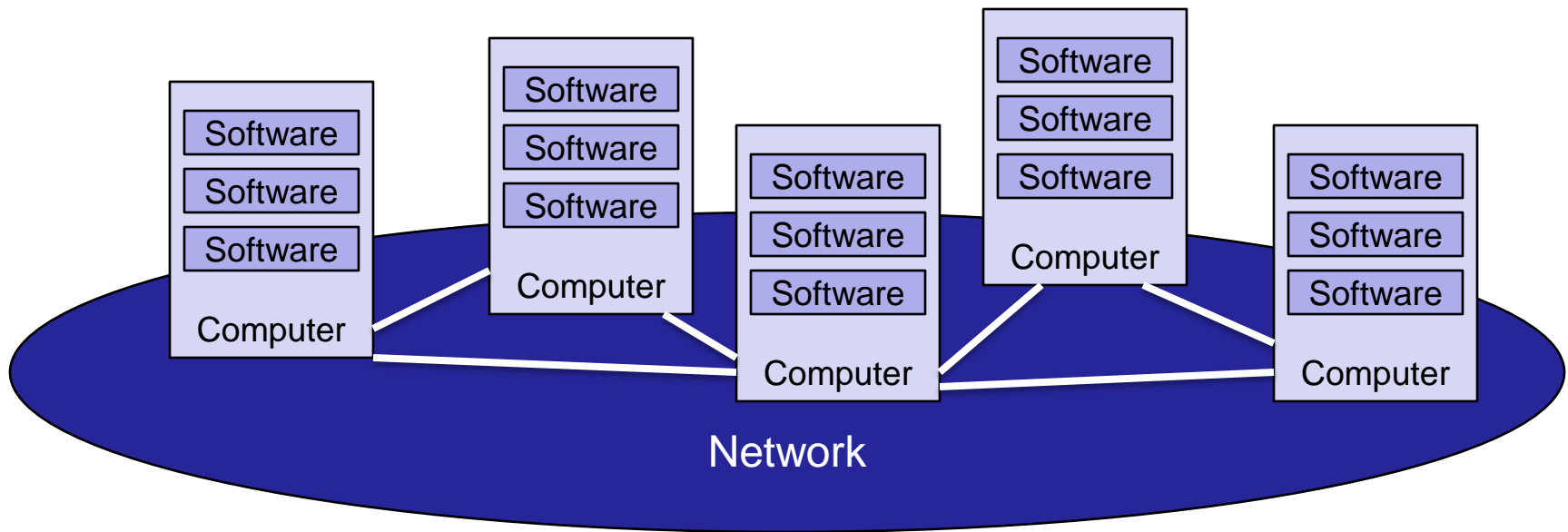


Computer and Network Security Principles



Environment

- IT infrastructures are made up of many components, abstractly:

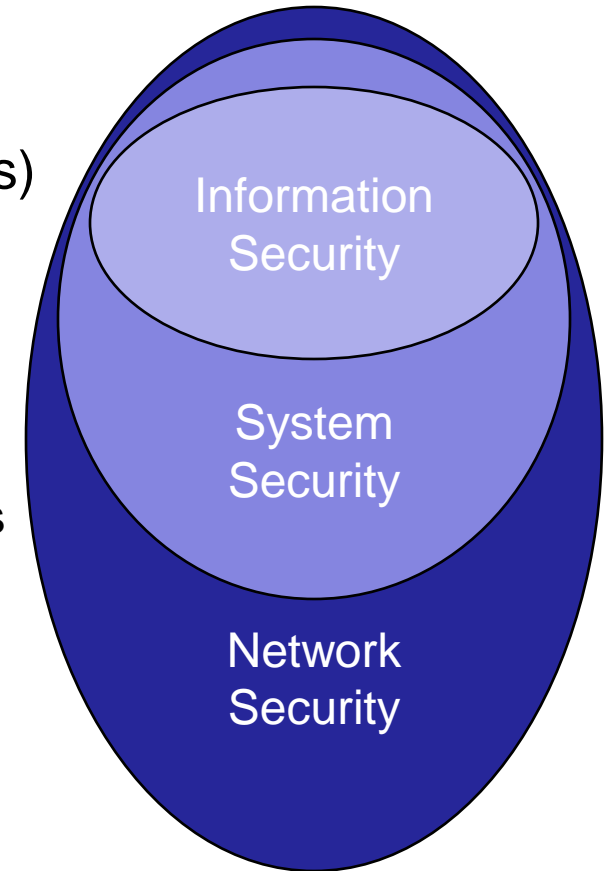


- Each component is uniquely vulnerable to attack



Course Layout

- Course is broken down into various components to reflect components
- Security Principles (1 lecture)
- Cryptography and Information Security (3 lectures)
 - Cryptography: Basic building blocks of security
 - Using cryptography to protect information
- System Security (3 lectures)
 - Attack and defense of individual computer systems
 - Focus on software security
 - Applications, Operating System
- Network Security (3 lectures)
 - Attack and defense of entire networks
 - Focus on network protocol security and protecting network infrastructure devices



Course Goals

- What this course *is not*:
 - Theory-based course on cryptography or security
 - Training course for securing IT devices (i.e. Windows workstations, Cisco routers, etc)
- What this course *is*:
 - Practical, hands-on course on security
 - Cover enough theory to understand issues
 - Focus on real-world security issues, rather than academic ones



Basic Syllabus

- Three mini-courses, back-to-back (plus this intro lecture)
- Each mini-course will involve
 - Homework
 - Project
 - Exam
- Grade Breakdown
 - Exams: 20% each (60% total)
 - Homework/Projects: 40%



Schedule

- Lecturers
 - Charles Clancy (Adjunct Professor, ECE)
 - Nick Petroni (Lecturer, ENTS)
 - Aaron Walters (Lecturer, ENTS)
- Schedule
 - Intro, Sep 4 (Clancy)
 - Cryptography, Sep 11, 18, 25, Oct 2 (Clancy)
 - System Security, Oct 9, 16, 23, 30 (Petroni)
 - Network Security, Nov 6, 13, 20, Dec 4 (Walters)
- No exam during finals week
- Lecture: Thursdays, 5:30pm – 8:15pm
- Office Hours: 8:15pm – 9:00pm (or by appointment)



Security Goals

- Confidentiality
 - Concealment of information
- Integrity
 - Trustworthiness of information
- Availability
 - Access to information



Threats

- Vocabulary
 - **Threat**: potential security vulnerability
 - **Attack**: action exploiting security vulnerabilities
 - **Adversary**: one who implements attacks
- Types of threats
 - **Disclosure**: unauthorized access to information
 - Threat to CONFIDENTIALITY
 - **Deception**: acceptance of false data
 - Threat to INTEGRITY
 - **Disruption**: interruption of correct operation
 - Threat to AVAILABILITY
 - **Usurpation**: unauthorized system control
 - Threat to ALL

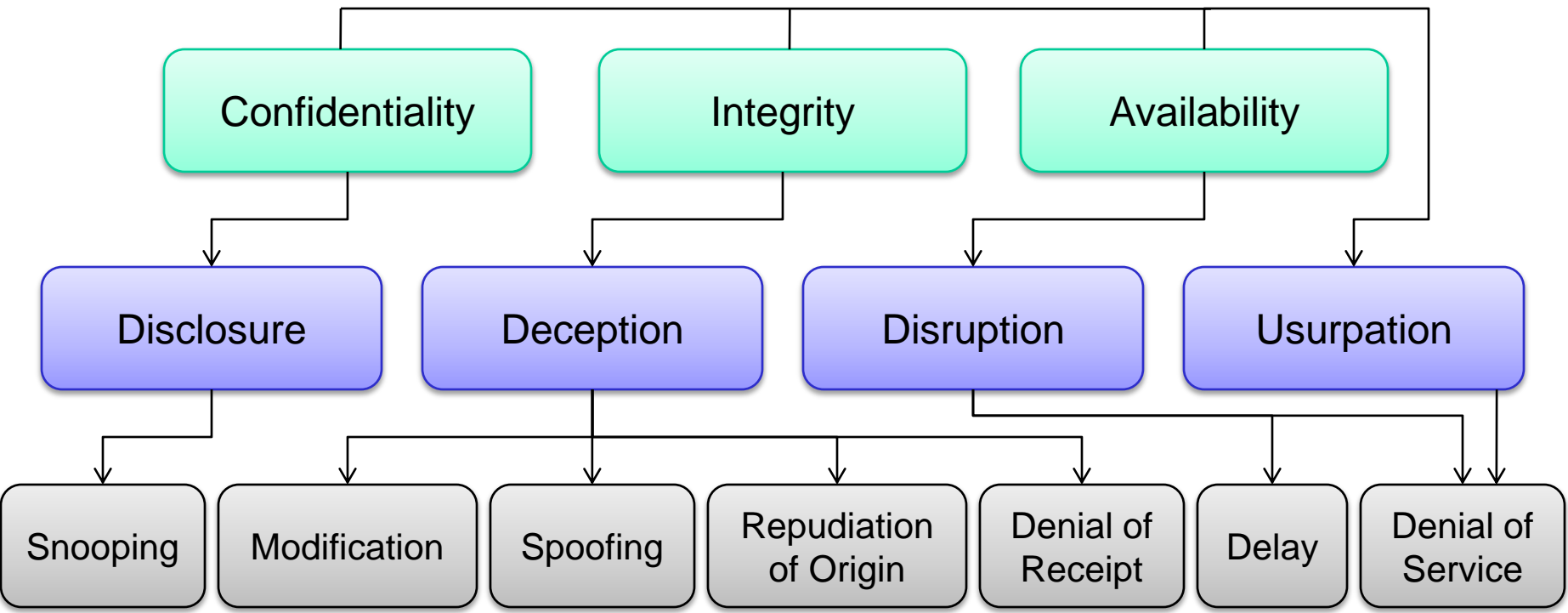


Threats

- Types of attacks
 - Snooping, aka wiretapping
 - Modification, unauthorized change of information
 - Masquerading, aka spoofing, impersonation
 - Repudiation of Origin, false denial that an entity sent/created something
 - Denial of Receipt, false denial that an entity received something
 - Delay
 - Denial of service



Entire Taxonomy



Threat Models

- Threat models define a hypothetical adversary's capabilities
- Based on capabilities, we can determine the types of attacks
- Given attacks we can determine threats
- Given threats we can determine the necessary security goals



Adversary Capabilities

- Can vary depending on the application, environment, system
- Examples...
- Crypto / Information Security
 - Access to keying material
 - Computational capabilities (i.e. laptop vs supercomputer)
- System Security
 - Access to operating system, applications
 - User vs administrator access
 - Physical access to system
- Network Security
 - On-path vs off-path
 - Active vs passive



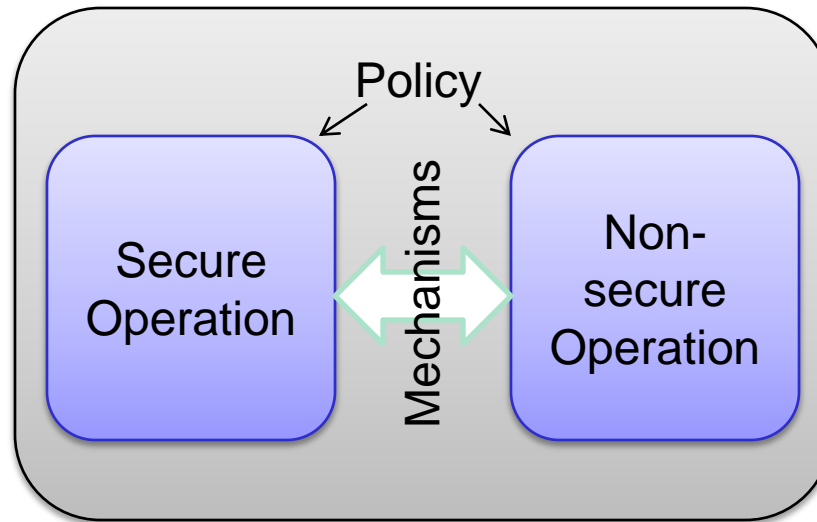
Policy vs Mechanism

- How are security goals achieved?
- *Security Policy*: statement as to what is or is not allowed
- *Security Mechanism*: method, tool, or procedure to enforce policy
- Example 1:
 - Bank policy requires that all home banking transactions have confidentiality and integrity
 - Bank website encrypts communications to web browsers and requires that users log in before accessing accounts
- Example 2:
 - University policy requires that users who forget their password be authenticated before having their password reset
 - University requires users to bring a photo ID to the helpdesk to reset passwords



Assumptions

- Two key assumptions:
 - Policy correctly and unambiguously partitions the set of system states into *secure* and *nonsecure*
 - Security mechanisms prevent the system from entering a *nonsecure* state



Dealing with Attacks

- Prevention
 - Mechanisms prevent attack from being effective
- Detection
 - Mechanisms to detect an attack has occurred
- Recovery
 - Mechanisms to allow a system to be restored to its pre-attack state after an attack is detected
 - Mechanisms to allow a system to function correctly in spite of an attack (more difficult)



Trust

- Implementing policy typically requires some level of trust
- Example:
 - Opening a door requires a key
 - Trusted parties have keys
 - Trusted parties can open the door
 - What about lock pickers? We assume they are trustworthy, and do not open doors without permission.



- Example:
 - Reading your email requires a password
 - Only you know your password, so only you can read your email
 - What about system administrators? We assume they are trustworthy and do not use their system access to read our email.



Assurance

- Trust is imprecise, and rarely “yes” or “no”
- Assurance is a trust metric: specifies how much you trust a system, and what do you trust it to do?
- Examples
 - Who on the Internet do you trust with your credit card information?
 - Would you trust a computer system more if it had up-to-date anti-virus software installed?



Assurance Operational Issues

- Enterprise networks: information assurance is based on a cost-benefit analysis
- Weigh the value of the information with the expense of protecting it
- Financial cost of a security policy violation
 - Theft of funds, intellectual property, state secrets
 - Loss of revenue or reputation
 - Cost of recovering from the attack
- Financial cost of implementing mechanisms
 - Additional staff to audit computer networks
 - Cost of firewalls, secure servers, physical security, etc
 - Cost of increased system complexity



Overall

Determine Adversary Capabilities
Determine Possible Attacks

Enumerate Threats

Establish Security Goals for System Components
(i.e. confidentiality, integrity, availability)

Develop Overall System Security Policy

Implement Security Mechanisms consistent with
the required level of Assurance

