Vulnerability and Information Flow Analysis of COTS

Somesh Jha, Bart Miller, Tom Reps
{jha,bart,reps}@cs.wisc.edu
Computer Sciences Department
University of Wisconsin
1210 W. Dayton Street
Madison, WI 53706-1685

Cost of Software Development Motivates Use of COTS

- High cost of software development
  - increased complexity
  - increasing degree of concurrency
  - increasing quality-assurance demands
  - other factors . . .
- Increased deployment of COTS
- CIP/SW TOPIC #6
  - Protecting COTS from the inside

Advantages and Disadvantages of COTS

- Advantages
  - reduced cost
  - promotes modular design
  - partitions the testing effort
- Disadvantages
  - higher risk of vulnerabilities
  - general quality-assurance issues

Unsafe Malicious Code

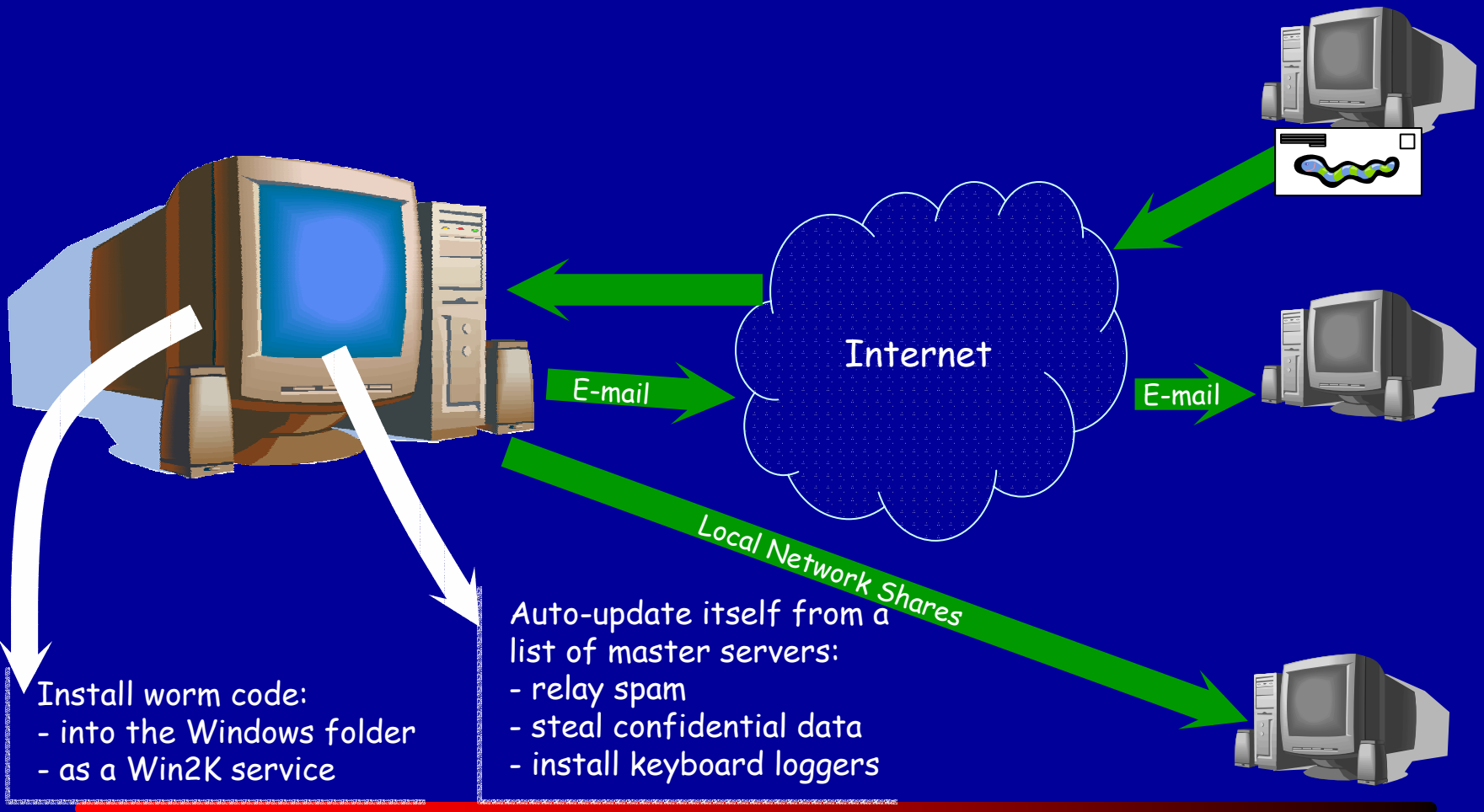
- Viruses
  - Gain access through infected files
- Worms
  - Spread over the network
- Trojans
  - Hide harmful behavior under the guise of useful programs

- Most often: combined code
  - worm + virus + trojan

- Distinguishing characteristics: something observable happens

Malicious Code Example:

Internet worm Sobig.E

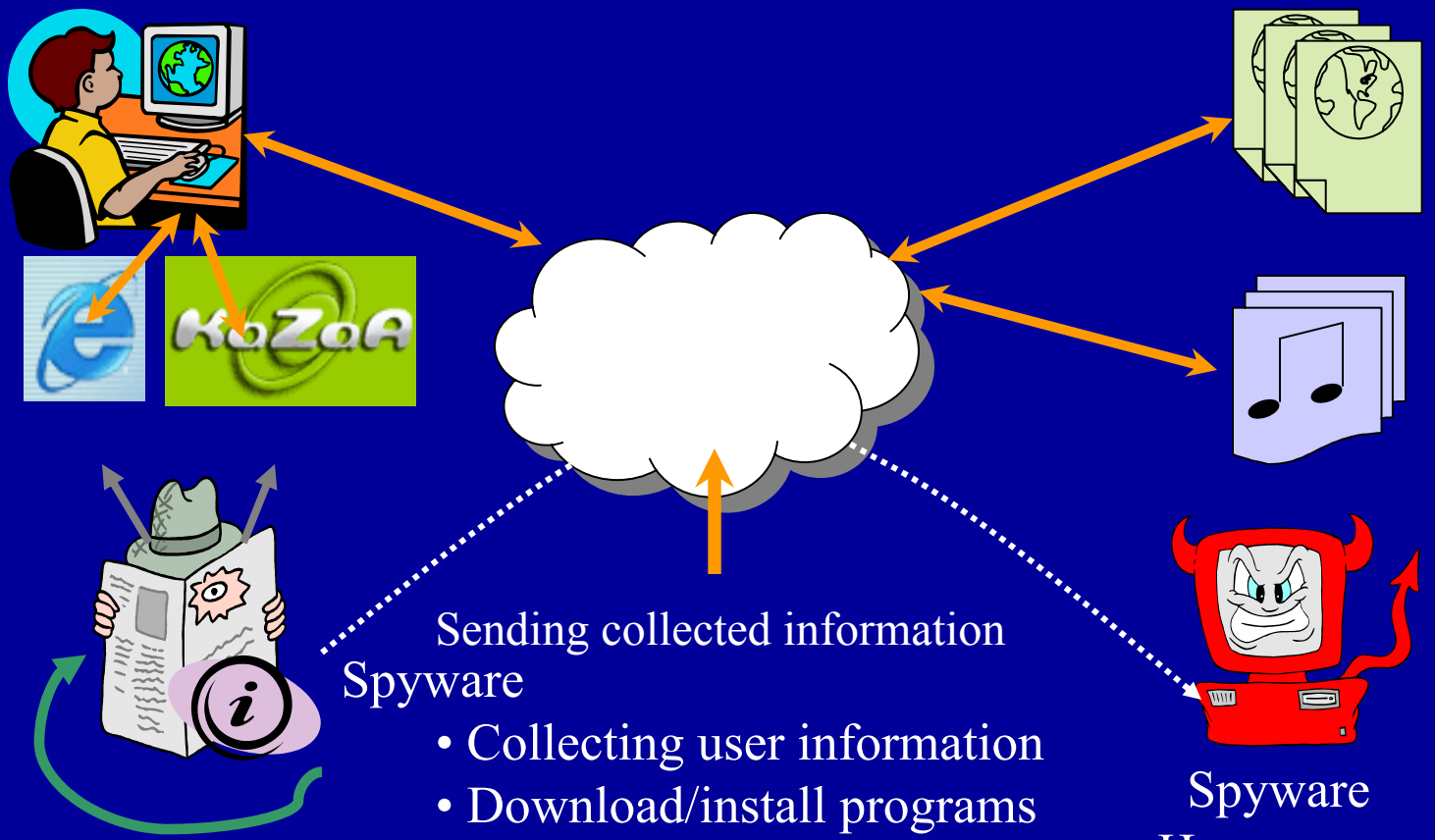


What Is Spyware?

- Spyware is software that
  - Is non-destructive (unlike a virus)
  - Operates in background—not easily observable
  - Is Often installed silently by other software
  - Usually integrated with desired functionality

- Privacy-violating malicious code
  - Provides useful functionality
  - But, "leaks" sensitive information

KaZaa in Operation



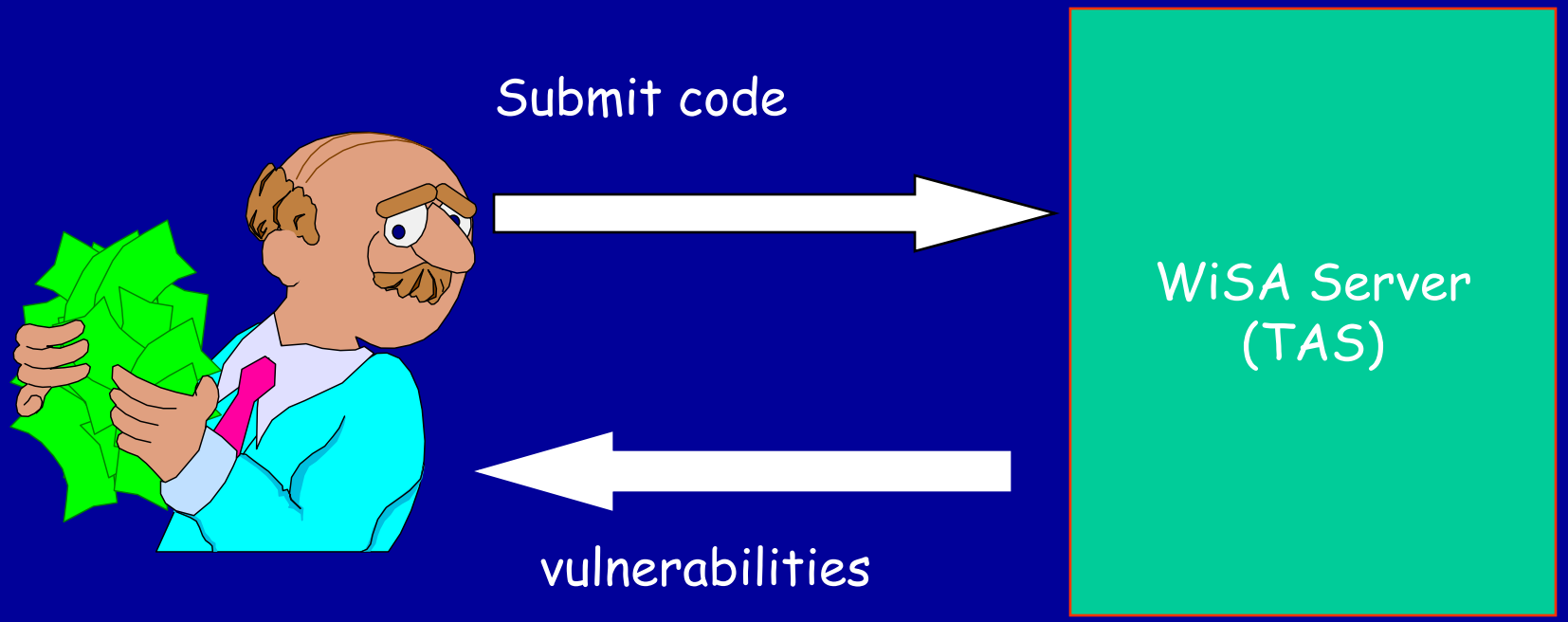
Spyware Summary

- Install a useful program
  - Play DVDs
- But …
  - Also install "spy" software, which monitors user behavior
    - Example: Monitor web traffic
- Aureate Media, Real Networks
- Consult
  - http://grc.com/optout.htm
- Perhaps useful to advisors/managers☺

WiSA: Don't Deploy COTS Without It

- We have proposed the Wisconsin Safety Analyzer
  - vulnerability and
    - Handles unsafe malicious code
  - information flow analysis of COTS
    - Handles privacy-violating malicious code (Spyware)
- Develop technology for static analysis of binaries
- Investigate applications

Trusted verification services



Benefits to DoD

- Reduces risk of deploying COTS
- Capable of discovering vulnerabilities in COTS
  - safety related
  - information-flow related
- Assign assurance levels to COTS components

WiSA Requirements

- Requirement 1
  - cannot mandate that all COTS packages will be written in the same language
  - source code for COTS frequently not available

  ∴ analysis of binaries/multi-lingual techniques
- Requirement 2
  - safety depends on context
  - desire to specify
    - discretionary access control
    - mandatory access control

  ∴ need an expressive specification language

WiSA Requirements

- Requirement 3
  - there are tradeoffs between scalability & precision
    - generally: efficiency vs. precision
    - but sometimes: more precise = more efficient

  ∴ tunable precision

- Requirement 4
  - wish to analyze compositions of COTS packages

  ∴ rely-guarantee reasoning and reason about compositions of vulnerabilities and constructing attack graphs

Initial Focus

- Our initial focus is on analyzing x86 binaries
- Reasons
  - high impact
    - several viruses written for the x86 platform
  - rich language
    - several hard analysis issues will be dealt with
    - can reuse architecture and experience in other settings
- partially addresses requirement 1

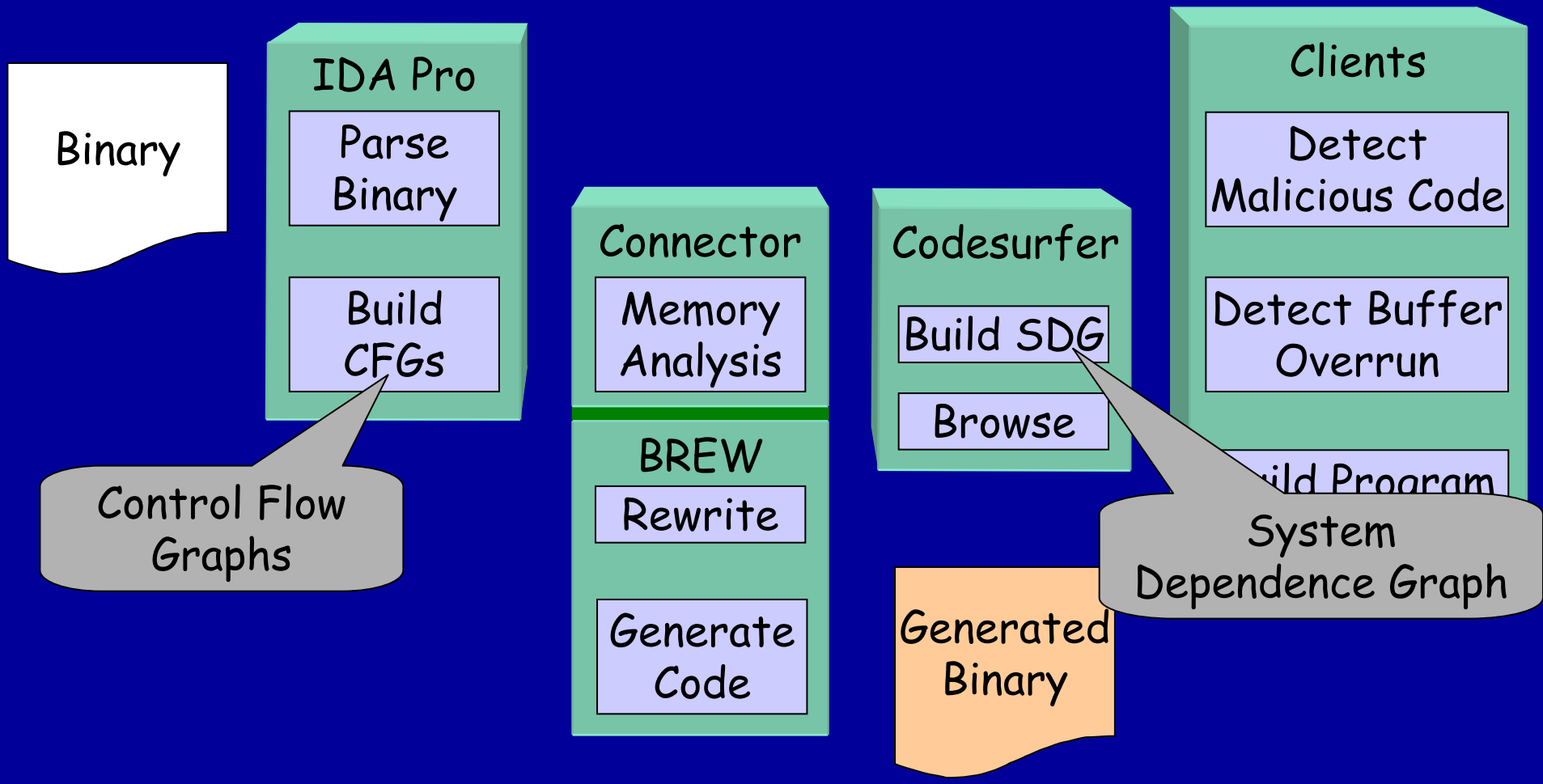
IDA Pro

- Decompilation tool
- Supports several executable file formats like COFF, ELF ....
- Gather as much information as possible
  - e.g. Names of functions, parameters to functions
- Is extensible through a built-in C like language

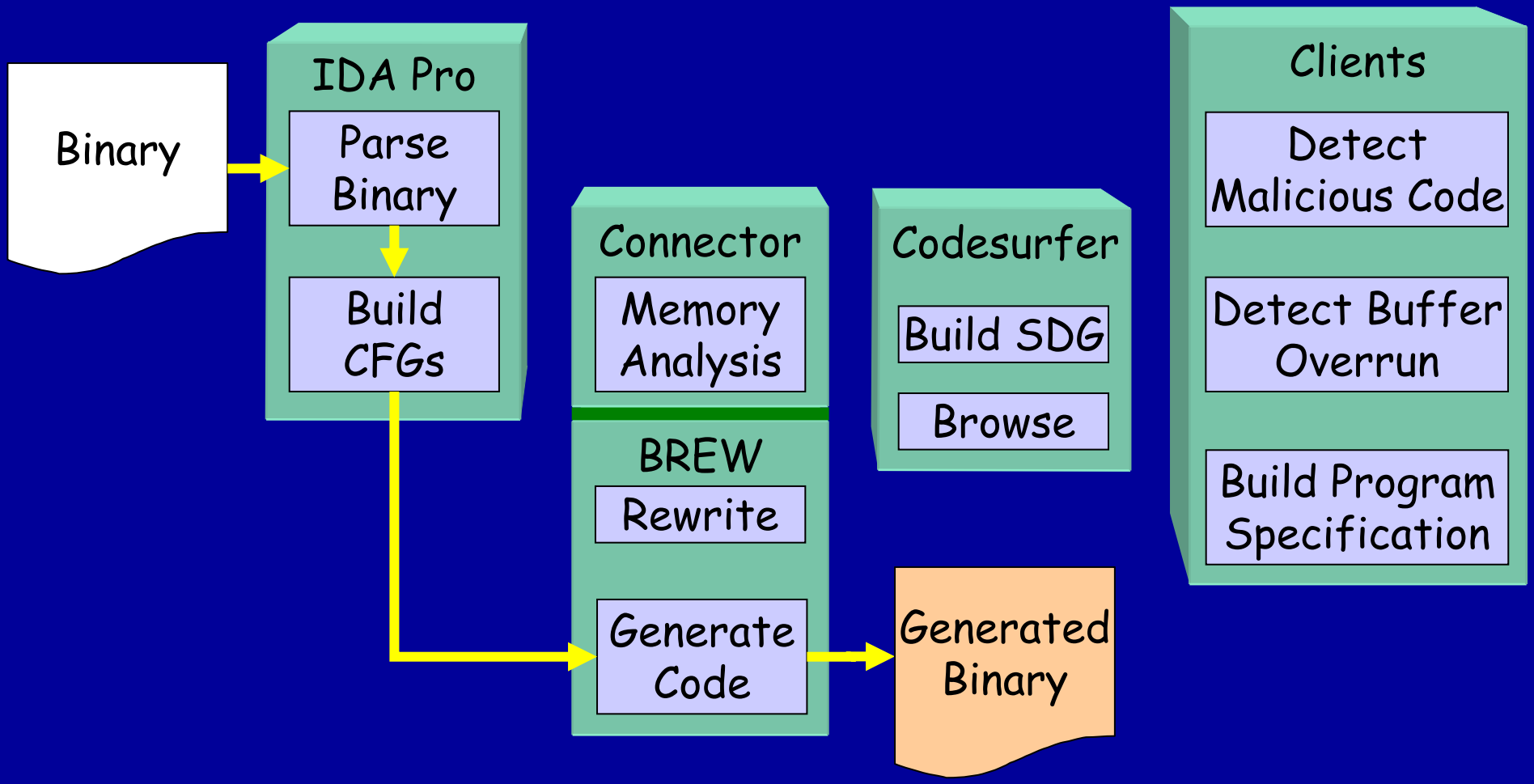
Codesurfer

- A program understanding tool
- Analyzes the data and control dependencies
  - Stores in System Dependence Graph (SDG)
  - Helpful in static analysis
- Provides an API to access the information stored in SDG
- The API can be extended

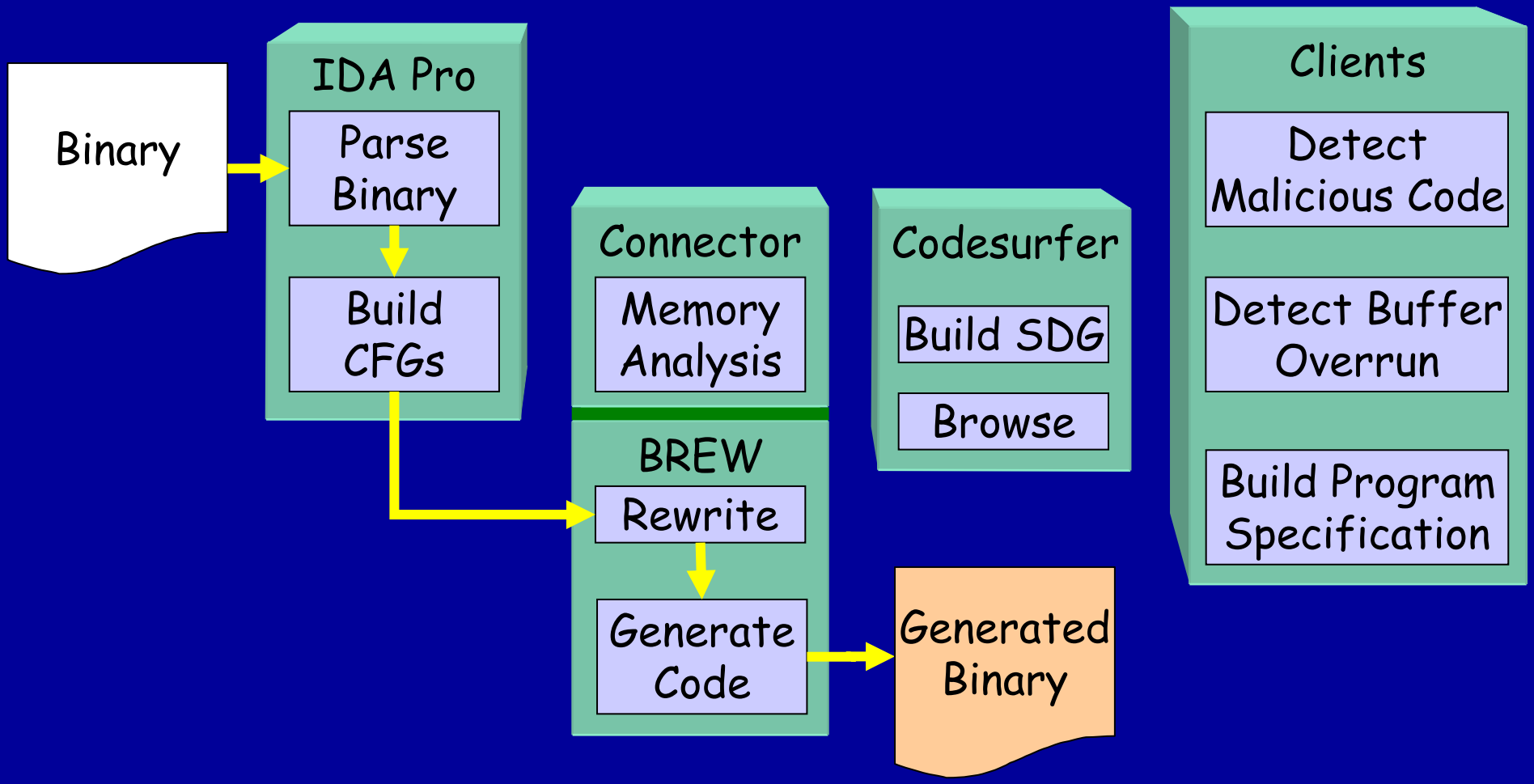
Architecture



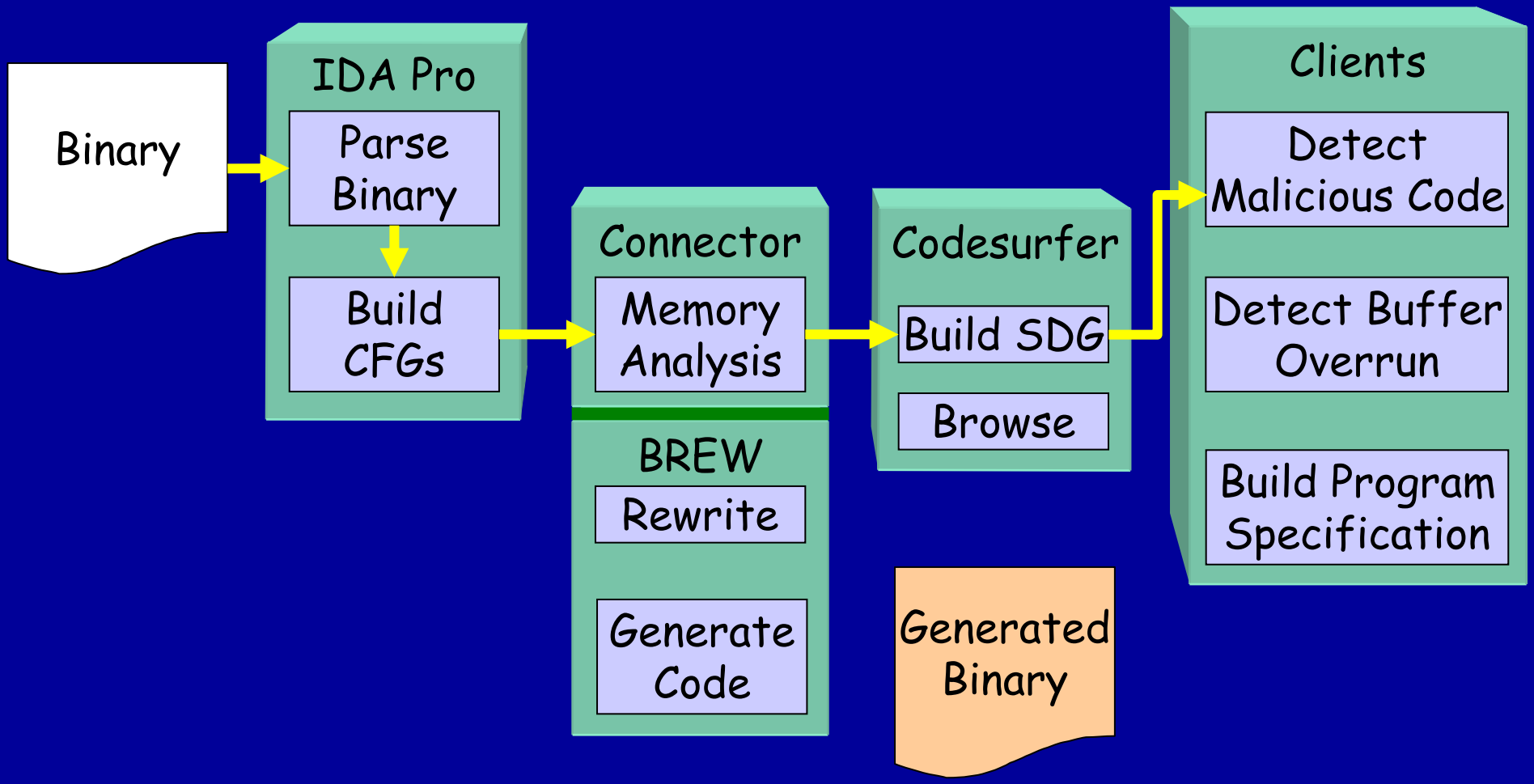
Code Generation



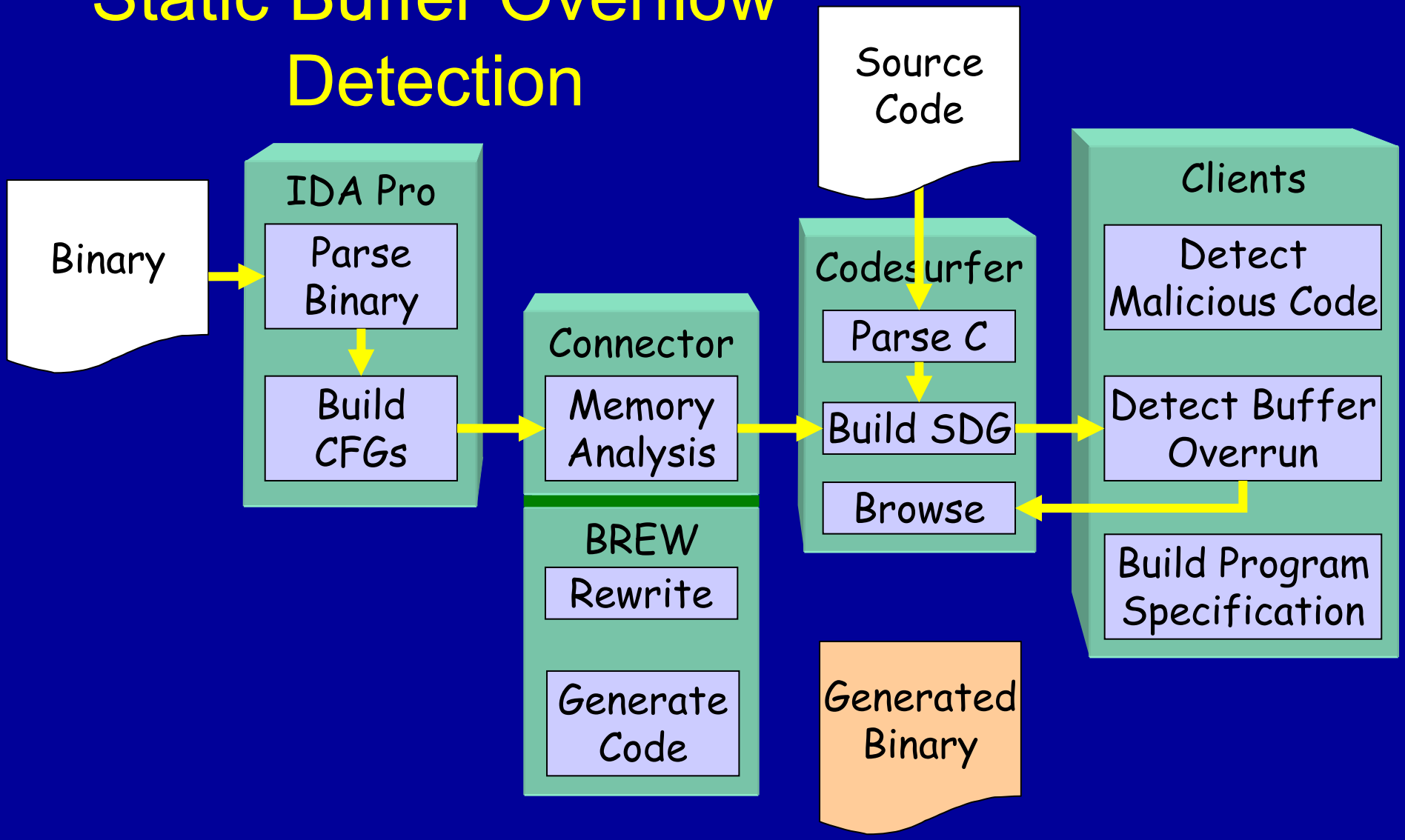
Dynamic Buffer Overflow Detection



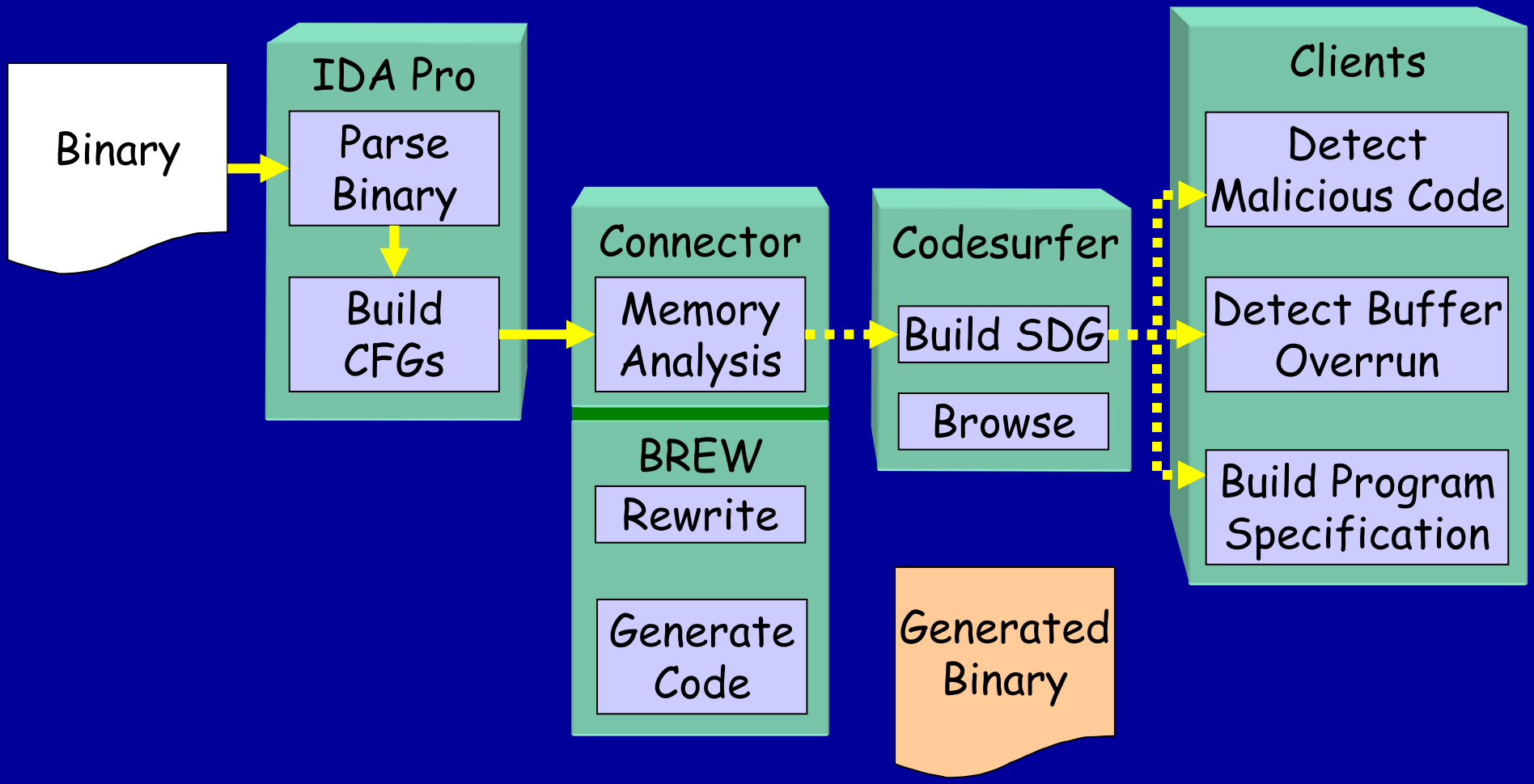
Malicious Code Detection



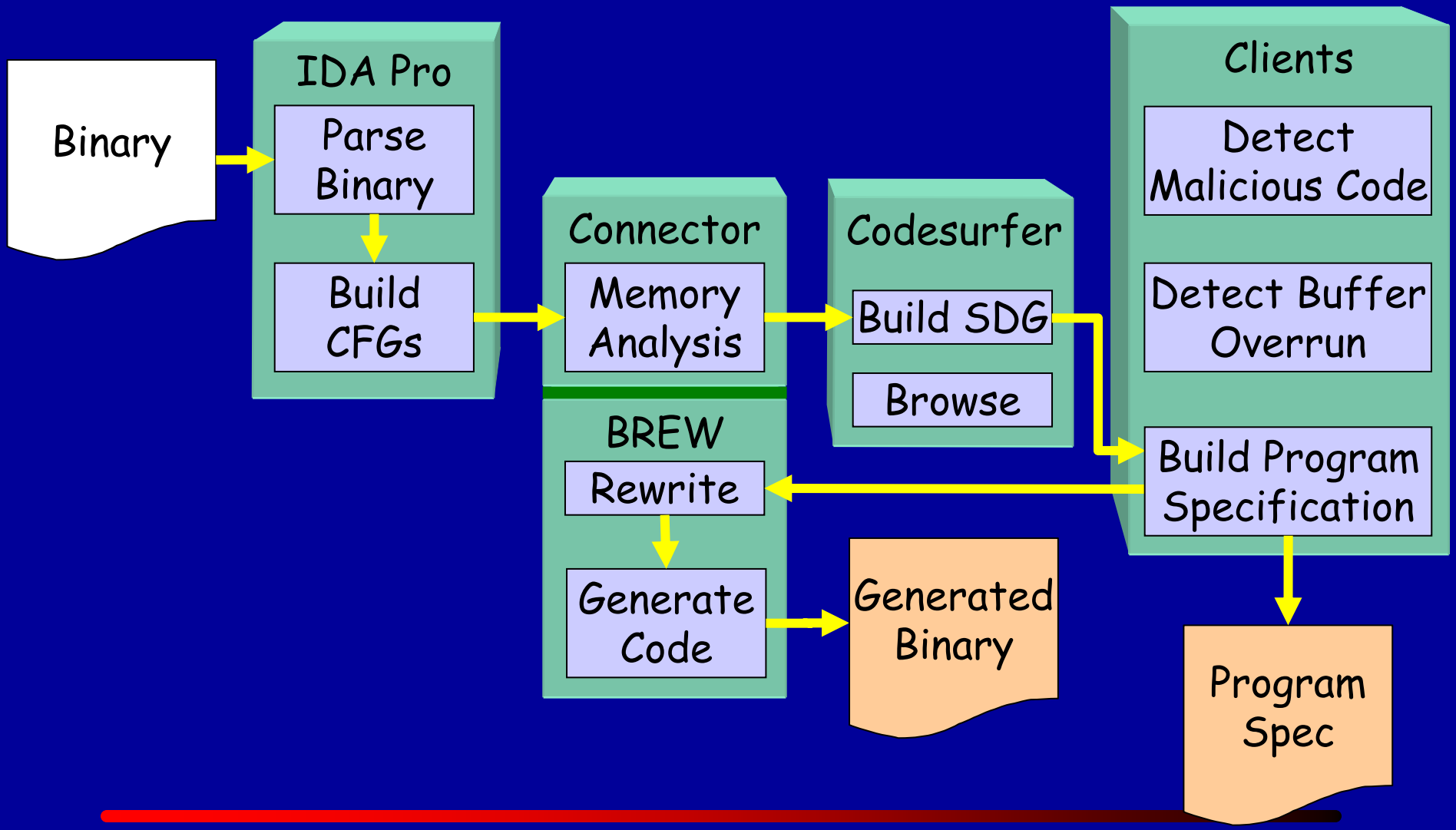
Static Buffer Overflow Detection



Value Set Analysis



Specification-Based Monitoring



Students Supported

- Gogul Balakrishnan
  - Advisor: Tom Reps
  - Going to take his qualifiers in the fall
- Mihai Christodorescu
  - Advisor: Somesh Jha
  - Passed his qualifiers (PL)
  - Prelim soon
- Vinod Ganapathy
  - Advisor: Somesh Jha
  - Passed his qualifiers (PL)

Students Supported (Contd.)

- Jon Giffin
  - Advisors: Somesh Jha and Bart Miller
  - Passed his qualifier (OS)
  - Prelim soon
- Hong Lin
  - Advisor: Bart Miller
  - Going to take her qualifier in the fall
- Hao Wang
  - Advisor: Somesh Jha
  - Passed his qualifier (OS)

Papers

- S. Jha and T. Reps, Analysis of SPKI/SDSI certificates using model checking, *Computer Security Foundations Workshop (CSFW),* June 2002.
- J. Giffin, S. Jha, and B. Miller, Detecting manipulated remote call streams, *Usenix Security Symposium,* August 2002.
- S. Schwoon, S. Jha, T. Reps, and S. Stubblebine, On generalized authorization problems, *Computer Security Foundations Workshop (CSFW),* 2003.

Papers

- M. Christodorescu and S. Jha, Static analysis of executables to detect malicious patterns, *Usenix Security Symposium*, August 2003.
- Four papers under submission
- Mihai's work is being patented
- Many more coming …

Technology Transfer

- Main vehicles for technology transfer
  - Grammatech
    - Tim Tietelbaum and David Melski will talk about this
  - NRL
    - Connie Hietmeyer is planning to visit UW-Madison
    - I am planning to visit her sometime in the fall
  - CERT and other such organizations
    - Disseminate bugs and vulnerabilities found
    - Hopefully …
  - Many other research projects have expressed interest in the infrastructure

Contact Information

- Prof. S. Jha
  - email: jha@cs.wisc.edu
- Prof. B. Miller
  - email: bart@cs.wisc.edu
- Prof. T. Reps
  - email: reps@cs.wisc.edu

- Computer Sciences Dept.
  1210 West Dayton Street
  Madison, WI 53706

Project home page
http://www.cs.wisc.edu/wisa