Quantitative Information Flow Analysis
CSCI 5271 Guest Lecture
Seonmo (Sean) Kim

Motivation
• An output has some data of an input.
• If the input contains some sensitive data, then output, too.
• The output should contain the intended amount of the input.
• An adversary wants to know the input by observing the output.

Motivation
• Consider two functions:

```c
int numCheck(int input){
    if (input == 1234) {
        return 1;
    }
    return 0;
}

int numCheck2(int input){
    if (input mod 2 == 0) {
        return input;
    }
    return 1;
}
```

• The number of output values?
• 2 vs 2^n+1

Motivation
• There are many applications related to QIF analysis
• AI, games, financial programs, etc.
• Scalability

Quantitative Information Flow (QIF)
• Given a (deterministic or probabilistic) program P which takes a high input H and produces a low output L
• An adversary observes L and P may leak information from H (secret) to L (public)
• Measure the amount of information leaked about H
Early models of QIF

- Used the Shannon mutual information $I(X;Y)$
- Uncertainty
  - $I(L;H) = H(H) - H(L)$
  - Information leaked = initial uncertainty – remaining uncertainty
    - The adversary's initial uncertainty before observing $L$
    - The adversary's remaining uncertainty after observing $L$
- $H(H) - I(L;H) = H(H) - H(L)$

Shannon entropy: initial uncertainty

- $H(X) = - \sum_{x} \Pr[X=x] \cdot \log_2 \Pr[X=x]$
**Alternative Measurement**

- **Vulnerability**
  - $V(X) = \max_{x} \Pr[X=x]$  
- **min-entropy**
  - $H_{\min}(X) = -\log V(X)$  
  - $H_{\min}(\mathcal{X}) = -\log V(\mathcal{X})$  
- **Information leaked** $= H_{\min}(H) - H_{\min}(H|L)$
  - Let $|X|$ be the number of possible values of $X$
  - $V(H) = \frac{1}{|X|} V(H|L) = \frac{1}{|X|}$  
  - $H_{\min}(H) - H_{\min}(H|L) = \log |H| - \log_{2}(|H|/|L|) = \log_{4}|L|$  

**Applications**

- **Image anonymization and KBattleship (PLDI 2008)**
  - Computing a maximum flow of information
- **Error reporting system (ASPLOS 2008)**
- **Heartbleed (VMCAI 2018)**
  - Using the model counting technique to measure the leakage

---

**Image Anonymization**

- **Image Anonymization**
  - **KBattleship**
  - Computing a maximum flow of information
  - **Heartbleed (VMCAI 2018)**
  - Using the model counting technique to measure the leakage
Flowcheck
• Dynamic analysis tool to measure an upper-bound estimate of the amount of information leaked
• Dynamic tainting
• Static control-flow regions
• \( c = d = a + b \)

Error Reporting System
• Scenario

Error Reporting System
• Symbolic Execution
  • Generates path conditions based on symbolic or concrete inputs

Measuring privacy loss
• For each condition \((op(f \cdot g))\), compute a summary for \( f \) and \( g \)
• Use a set of rules to compute the bound given the summaries
• Example
  • \((\text{add (bitwise-and x 1)} 3)\)
  • \((\text{bitwise-and x 1}) > 0 \text{ or } 1\)
  • \((\text{add (bitwise-and x 1)} 3) > 3 \text{ or } 4\)

Heartbleed

Exact Model Counting
• Brute-force counting
  • Go through every seat
  • Simple, but hard to scale
Exact Model Counting

• Brute-force counting
  • Go through every seat
  • Simple, but hard to scale

• DPLL-style counting
  • Detect a region that is empty
  • Faster, but still accounts for every seat

Approximate model counting

• Random sampling
  • Randomly pick a region
  • Count the number and scale up

Approximate model counting

• Random sampling
  • Randomly pick a region
  • Count the number and scale up

Approximate model counting

• Random hashing (AAAI 2006)
  • Everyone flips a coin $k$ times
  • Leave if a tail is ever shown
  • Count the persons $n$
  • Approximately $2^n / n$ persons

Q & A

Thank You:)