CS 260 02: Information Security Lab

Chengyu Song

Slides based on CS6265 taught by Prof. Taesoo Kim
Information Security Lab

• A special course: supervised, hands-on laboratory
• Designed for seniors and above
  • Prerequisite: OS, system programming, architecture
  • Background: low-level programming (assembly, C)
Course Goal: Capture-the-flag
Topics

• Reverse engineering
• Binary exploitation
• Binary analysis
• Memory forensic
• etc.

Weekly Structure

- **Fri**: Cover single topic (e.g., stack overflow)
  - 30 min: discus last week's challenges (be prepared, two fastest solvers)
  - 30 min: cover this week's topic
- **Mon**: Tutorial
  - 30 - 60 min: bring your laptop!
- **Wed**: Backup (holidays, conferences)
Weekly Structure

- **Thu**: Office hour (last minute questions) 5-6pm
- **Thu**: Submission deadline **11:59 pm**
  - Submit: flag, writeup/exploit of each challenge
Course Grading

- **100% Lab** (if you didn't turn in a single lab, you will get F)
- No midterm and final exams
- **40%: A, 30-40%: B, 30-20%: C and below**
- See Game Rules
Scoring Each Lab (Game Rules)

• Approximately **10 challenges**
• 20 pt (flag) x 1.0 (write-up/exploit) = 20 pt (each challenge)
• So, 200 pt (20 pt x 10 challenges) in each lab
• **Bonus**: two fastest solvers get 10 and 5 bonus pt
• **Late policy**: 50% of the original points (an extra week)
Misconduct Policy

- Cheating vs. collaboration
- Refer Academic Integrity Policy
- **Never ever** use/copy other students' code/write-up
- Please write down names of your collaborators
Administrative

• No TA
• Contact: csong@cs.ucr.edu
• Website: http://www.cs.ucr.edu/~csong/seclab/17/index.html
• Scoreboard: <>
• Join Piazza
  • Be active: bonus points on top participants
# Next Two Weeks

<table>
<thead>
<tr>
<th>Monday</th>
<th>Tuesday</th>
<th>Wednesday</th>
<th>Thursday</th>
<th>Friday</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan 9</td>
<td>Jan 10</td>
<td>Jan 11</td>
<td>Jan 12</td>
<td>Jan 13</td>
</tr>
<tr>
<td><strong>First day of class</strong></td>
<td></td>
<td></td>
<td></td>
<td><strong>LEC 1:</strong> Warm-up: x86, Tools</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td><strong>TUT 1:</strong> GDB/x86</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td><strong>Preparation:</strong> Read asm</td>
</tr>
<tr>
<td>Jan 16</td>
<td>Jan 17</td>
<td>Jan 18</td>
<td>Jan 19</td>
<td>Jan 20</td>
</tr>
<tr>
<td><strong>Martin Luther King Day</strong></td>
<td></td>
<td></td>
<td></td>
<td><strong>LEC 2:</strong> Warm-up: x86, Shellcode, Tools</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td><strong>TUT 2:</strong> PEDA, Shellcode (video/slides)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td><strong>Preparation:</strong> Read x86, shellcode (video/slides)</td>
</tr>
</tbody>
</table>