CS 6V81-05
Advanced Digital Forensics and Data Reverse Engineering –
Course Overview

Zhiqiang Lin

Department of Computer Science
The University of Texas at Dallas

August 26th, 2011
Outline

1. Overview
   - Course Goals
   - Course Style

2. Course Content
   - Foundations
   - Techniques
   - Tools
   - Applications

3. Course Project

4. Course Policy

5. Summary
What is Digital Forensics

**Digital Forensics** is a branch of forensic science (in relation to computer crime) focusing on the recovery and investigation of material (essentially data) found in digital devices.
More broadly, what is Data Reverse Engineering

Data reverse engineering deals with the problem of
1. What information is stored in a computer system
2. How this information can be extracted and used.
Outline

1 Overview
   - Course Goals
   - Course Style

2 Course Content
   - Foundations
   - Techniques
   - Tools
   - Applications

3 Course Project

4 Course Policy

5 Summary
Outline

1. Overview
   - Course Goals
   - Course Style

2. Course Content
   - Foundations
   - Techniques
   - Tools
   - Applications

3. Course Project

4. Course Policy

5. Summary
The **general goal** is to introduce the current techniques of data reverse engineering used in both research and practice.
Course Goals

The **general goal** is to introduce the current techniques of data reverse engineering used in both research and practice.

Students will

1. **Learn about latest research** from the literature.
Course Goals

The general goal is to introduce the current techniques of data reverse engineering used in both research and practice.

Students will

1. Learn about latest research from the literature.
2. Perform research a semester long research project.
Course Goals

The **general goal** is to introduce the current techniques of data reverse engineering used in both research and practice.

Students will

1. **Learn about latest research** from the literature.
2. **Perform research** a semester long research project.
3. **Practice**
   - Use and understand the tools
   - Participate a CTF
Outline

1 Overview
   - Course Goals
   - Course Style

2 Course Content
   - Foundations
   - Techniques
   - Tools
   - Applications

3 Course Project

4 Course Policy

5 Summary
This course is taught in **seminar-style**. Each student will be expected to
This course is taught in seminar-style. Each student will be expected to

1. **Read** the most recent papers.
This course is taught in seminar-style. Each student will be expected to

1. **Read** the most recent papers.

2. **Present** and **lead** the discussion on one topic.
This course is taught in **seminar-style**. Each student will be expected to

1. **Read** the most recent papers.
2. **Present** and **lead** the discussion on one topic.
3. **Write** the summary on the topic of choosing.
Outline

1. Overview
   - Course Goals
   - Course Style

2. Course Content
   - Foundations
   - Techniques
   - Tools
   - Applications

3. Course Project

4. Course Policy

5. Summary
Outline

1. Overview
   - Course Goals
   - Course Style

2. Course Content
   - Foundations
   - Techniques
   - Tools
   - Applications

3. Course Project

4. Course Policy

5. Summary
Operating System

- **Manages** the electronic device (e.g., computer, smart-phone)
Operating System

- **Manages** the electronic device (e.g., computer, smart-phone)
- **Organizes** data (especially memory)
Operating System

- **Manages** the electronic device (e.g., computer, smart-phone)
- **Organizes** data (especially memory)
- **Provides** common services (API) for execution of various application software
Operating System

- **Manages** the electronic device (e.g., computer, smart-phone)
- **Organizes** data (especially memory)
- **Provides** common services (API) for execution of various application software
File System

File system
File System

File system

- **Organizes** data expected to be retained after a program terminates
File System

File system

- **Organizes** data expected to be retained after a program terminates
- **Provides** procedures (API) to store, retrieve and update data
File system

- **Organizes** data expected to be retained after a program terminates
- **Provides** procedures (API) to store, retrieve and update data
- **Manages** the available space on the device(s) which contain the file system.
File System

File system
- **Organizes** data expected to be retained after a program terminates
- **Provides** procedures (API) to store, retrieve and update data
- **Manages** the available space on the device(s) which contain the file system.
Compiler

Transforms **source code** written in a programming language into **object code**.
Compiler

Transforms **source code** written in a programming language into **object code**.
Transforms **source code** written in a programming language into **object code**.
Compiler transforms source code written in a programming language into object code.
Outline

1. Overview
   - Course Goals
   - Course Style

2. Course Content
   - Foundations
   - Techniques
   - Tools
   - Applications

3. Course Project

4. Course Policy

5. Summary
Data Structure

- Data in computer is usually structured
Data Structure Reverse Engineering

Data Structure

- Data in computer is usually structured
- Program is used to process data
Data Structure

- Data in computer is usually structured
- Program is used to process data
- Data structure can be reverse engineered
Data Structure

- Data in computer is usually structured
- Program is used to process data
- Data structure can be reverse engineered
Data Structure Reverse Engineering

Uncovering the **syntax** and **semantics** of the data structure.

Syntax

- Layout
Data Structure Reverse Engineering

Uncovering the **syntax** and **semantics** of the data structure.

**Syntax**
- Layout
- Size
Uncovering the **syntax** and **semantics** of the data structure.

**Syntax**
- Layout
- Size
- Offset
Data Structure Reverse Engineering

Uncovering the **syntax** and **semantics** of the data structure.

**Syntax**
- Layout
- Size
- Offset

**Semantics**
- Meaning
Uncovering the **syntax** and **semantics** of the data structure.

**Syntax**
- Layout
- Size
- Offset

**Semantics**
- Meaning
- Context
Data Structure Reverse Engineering

State of the art techniques:
State of the art techniques:

- **REWARDS** [NDSS’10]: Dynamic analysis
State of the art techniques:

- **REWARDS** [NDSS’10]: Dynamic analysis
- **TIE** [NDSS’11]: Static + Dynamic
State of the art techniques:

- **REWARDS** [NDSS’10]: Dynamic analysis
- **TIE** [NDSS’11]: Static + Dynamic
- **HOWARD** [NDSS’11]: Dynamic analysis
State of the art techniques:

- **REWARDS** [NDSS’10]: Dynamic analysis
- **TIE** [NDSS’11]: Static + Dynamic
- **HOWARD** [NDSS’11]: Dynamic analysis
Two types of data analysis

Data exists in a computer system in two medias: **Memory** and **Disk**
Two types of data analysis

Data exists in a computer system in two medias: **Memory** and **Disk**

**Memory Analysis**

Analyzing data which is in volatile memory.
Two types of data analysis

Data exists in a computer system in two medias: **Memory** and **Disk**

**Memory Analysis**
Analyzing data which is in volatile memory.

**File Analysis**
Analyzing data which is in persistent disk.
Two types of data analysis

Data exists in a computer system in two medias: **Memory** and **Disk**

**Memory Analysis**
Analyzing data which is in volatile memory.

**File Analysis**
Analyzing data which is in persistent disk.
Outline

1. Overview
   - Course Goals
   - Course Style

2. Course Content
   - Foundations
   - Techniques
   - Tools
   - Applications

3. Course Project

4. Course Policy

5. Summary
Memory Tools

Graph based approach

- Using road map to navigate
- Crash/gdb utility (Linux) [USENIX 2005]
- LiveKD/WinDBG (Windows)
- Volatility
- KOP [CCS'09]

Value-invariant based approach

- Using constant value (or range) to look for data
- Value-invariant [CCS'09]
- ColdBoot [USENIX Security'07]...
Graph based approach

Using road map to **navigate** data

- Crash/gdb utility (Linux) [USENIX 2005]
Memory Tools

Graph based approach

Using road map to **navigate** data

- Crash/gdb utility (Linux) [USENIX 2005]
- LiveKD/WinDBG (Windows)
Memory Tools

Graph based approach

Using road map to **navigate** data

- Crash/gdb utility (Linux) [USENIX 2005]
- LiveKD/WinDBG (Windows)
- Volatility
Memory Tools

Graph based approach

Using road map to **navigate** data

- Crash/gdb utility (Linux) [USENIX 2005]
- LiveKD/WinDBG (Windows)
- Volatility
- KOP [CCS’09]
Memory Tools

Graph based approach

Using road map to **navigate** data
- Crash/gdb utility (Linux) [USENIX 2005]
- LiveKD/WinDBG (Windows)
- Volatility
- KOP [CCS’09]

Value-invariant based approach
Memory Tools

Graph based approach
Using road map to **navigate** data
- Crash/gdb utility (Linux) [USENIX 2005]
- LiveKD/WinDBG (Windows)
- Volatility
- KOP [CCS’09]

Value-invariant based approach
Using constant value (or range) to look for data
- Value-invariant [CCS’09]
Memory Tools

Graph based approach

Using road map to **navigate** data
- Crash/gdb utility (Linux) [USENIX 2005]
- LiveKD/WinDBG (Windows)
- Volatility
- KOP [CCS’09]

Value-invariant based approach

Using constant value (or range) to look for data
- Value-invariant [CCS’09]
- ColdBoot [USENIX Security’07]
Memory Tools

Graph based approach
Using road map to **navigate** data
- Crash/gdb utility (Linux) [USENIX 2005]
- LiveKD/WinDBG (Windows)
- Volatility
- KOP [CCS’09]

Value-invariant based approach
Using constant value (or range) to look for data
- Value-invariant [CCS’09]
- ColdBoot [USENIX Security’07]
- ...
Disk Tools

- **Sleuthkit** [http://www.sleuthkit.org/](http://www.sleuthkit.org/)
Disk Tools

- **Sleuthkit** [http://www.sleuthkit.org/](http://www.sleuthkit.org/)
- **foremost** [http://foremost.sf.net/](http://foremost.sf.net/), Linux based file carving program
Disk Tools

- **Sleuthkit** [http://www.sleuthkit.org/](http://www.sleuthkit.org/)
- **foremost** [http://foremost.sf.net/](http://foremost.sf.net/), Linux based file carving program
- **FTimes** [http://ftimes.sourceforge.net/FTimes/index.shtml](http://ftimes.sourceforge.net/FTimes/index.shtml), a system baselining and evidence collection tool
Disk Tools

- **Sleuthkit** [http://www.sleuthkit.org/](http://www.sleuthkit.org/)
- **foremost** [http://foremost.sf.net/](http://foremost.sf.net/), Linux based file carving program
- **FTimes** [http://ftimes.sourceforge.net/FTimes/index.shtml](http://ftimes.sourceforge.net/FTimes/index.shtml), a system baselining and evidence collection tool
Disk Tools

- **Sleuthkit** [http://www.sleuthkit.org/](http://www.sleuthkit.org/)
- **foremost** [http://foremost.sf.net/](http://foremost.sf.net/), Linux based file carving program
- **FTimes** [http://ftimes.sourceforge.net/FTimes/index.shtml](http://ftimes.sourceforge.net/FTimes/index.shtml), a system baselining and evidence collection tool
Disk Tools

- **Sleuthkit** [http://www.sleuthkit.org/](http://www.sleuthkit.org/)
- **foremost** [http://foremost.sf.net/](http://foremost.sf.net/), Linux based file carving program
- **FTimes** [http://ftimes.sourceforge.net/FTimes/index.shtml](http://ftimes.sourceforge.net/FTimes/index.shtml), a system baselining and evidence collection tool
How to extract the analyzing data

Memory

- Crash Dumps
How to extract the analyzing data

Memory
- Crash Dumps
- LiveKd Dumps
How to extract the analyzing data

Memory

- Crash Dumps
- LiveKd Dumps
- Hibernation Files
How to extract the analyzing data

Memory

- Crash Dumps
- LiveKd Dumps
- Hibernation Files
- Virtual Machine Imaging
How to extract the analyzing data

Memory

- Crash Dumps
- LiveKd Dumps
- Hibernation Files
- Virtual Machine Imaging
- `dd (/dev/mem or /dev/kmem), Win32dd`
How to extract the analyzing data

**Memory**

- Crash Dumps
- LiveKd Dumps
- Hibernation Files
- Virtual Machine Imaging
- `dd (/dev/mem or /dev/kmem), Win32dd`
- `fmem`, a kernel module (/dev/fmem)
How to extract the analyzing data

Memory

- Crash Dumps
- LiveKd Dumps
- Hibernation Files
- Virtual Machine Imaging
- `dd (/dev/mem or /dev/kmem), Win32dd`
- `fmem`, a kernel module (/dev/fmem)
- ...

Disk Data is persistent. Directly fed to the analysis system.
How to extract the analyzing data

Memory
- Crash Dumps
- LiveKd Dumps
- Hibernation Files
- Virtual Machine Imaging
- `dd (/dev/mem or /dev/kmem), Win32dd`
- `fmem`, a kernel module (`/dev/fmem`)
- ...

Disk
How to extract the analyzing data

**Memory**
- Crash Dumps
- LiveKd Dumps
- Hibernation Files
- Virtual Machine Imaging
- `dd (/dev/mem or /dev/kmem), Win32dd`
- `fmem`, a kernel module (`/dev/fmem`)
- ...

**Disk**
Data is persistent. Directly fed to the analysis system.
Outline

1. Overview
   - Course Goals
   - Course Style

2. Course Content
   - Foundations
   - Techniques
   - Tools
   - Applications

3. Course Project

4. Course Policy

5. Summary
Digital Forensics

Recovering evidence from digital data to support or refute a hypothesis before a criminal court.

- Computer forensics
- Network forensics
- Database forensics
- Mobile device forensics
Digital Forensics

Recovering evidence from digital data to support or refute a hypothesis before a criminal court.

- Computer forensics
- Network forensics
- Database forensics
- Mobile device forensics
DEFCON 2011 CTF F100 Challenge

INPUT
PNG image, 19025 x 1, 8-bit/color RGBA, non-interlaced
DEFCON 2011 CTF F100 Challenge

INPUT
PNG image, 19025 x 1, 8-bit/color RGBA, non-interlaced

ANALYSIS
A long row with pixels, let’s rearrange to blocks
DEFCON 2011 CTF F100 Challenge

INPUT

PNG image, 19025 x 1, 8-bit/color RGBA, non-interlaced

ANALYSIS

A long row with pixels, let's rearrange to blocks

#!/usr/bin/python
import Image
def create_image_file(len):
    newImg = Image.new("RGB", (len, 800), "BLACK")
    oldImg = Image.open(‘f100.png’)
    for pixel, value in enumerate(oldImg.getdata()):
        x, y = (pixel \% len, pixel / len)
        newImg.putpixel((x,y),value)
    newImg.save(‘f100\_'+str(len)+’.png’)

def main():
    for i in range(25, 139):
        create_image_file(i)

OUTPUT
DEFCON 2011 CTF F100 Challenge

INPUT
PNG image, 19025 x 1, 8-bit/color RGBA, non-interlaced

ANALYSIS
A long row with pixels, let's rearrange to blocks

```python
#!/usr/bin/python
import Image
def create_image_file(len):
    newImg = Image.new("RGB", (len, 800), "BLACK")
    oldImg = Image.open(’f100.png’)
    for pixel, value in enumerate(oldImg.getdata()):
        x, y = (pixel \% len, pixel / len)
        newImage.putpixel((x,y),value)
    newImage.save(’f100\_’+str(len)+’’.png’)
def main():
    for i in range(25, 139):
        create_image_file(i)
```

OUTPUT
thankYouSirPleasemayIhaveAnother
Kernel Rootkit Defense

Kernel level malware which hides kernel object and enables a continued privileged access
Kernel Rootkit Defense

Kernel level malware which hides kernel object and enables a continued privileged access
Kernel level malware which hides kernel object and enables a continued privileged access
Kernel Rootkit Defense

Kernel level malware which hides kernel object and enables a continued privileged access.
Malware Analysis

Malicious software

- Consists of programming (code, scripts, active content, and other software) designed to disrupt or deny operation
Malware Analysis

Malicious software

- Consists of programming (code, scripts, active content, and other software) designed to disrupt or deny operation
- Gathers information that leads to loss of privacy or exploitation
Malware Analysis

Malicious software

- Consists of programming (code, scripts, active content, and other software) designed to disrupt or deny operation
- Gathers information that leads to loss of privacy or exploitation
- Gains unauthorized access to system resources and other abusive behavior
Malware Analysis

Malicious software

- Consists of programming (code, scripts, active content, and other software) designed to disrupt or deny operation
- Gathers information that leads to loss of privacy or exploitation
- Gains unauthorized access to system resources and other abusive behavior
Malware Analysis

Malicious software

- Consists of programming (code, scripts, active content, and other software) designed to disrupt or deny operation
- Gathers information that leads to loss of privacy or exploitation
- Gains unauthorized access to system resources and other abusive behavior
Game Hacking/Cheating

Modification of a platform’s system memory during game play, or modification of files that comprise a game, to achieve a desired effect during game play.
Game Hacking/Cheating

Modification of a platform’s system memory during game play, or modification of files that comprise a game, to achieve a desired effect during game play.

- Building a bot
Game Hacking/Cheating

Modification of a platform’s system memory during game play, or modification of files that comprise a game, to achieve a desired effect during game play.

- Building a bot
- Manipulating memory
Game Hacking/Cheating

Modification of a platform’s system memory during game play, or modification of files that comprise a game, to achieve a desired effect during game play.

- Building a bot
- Manipulating memory
- Operating a proxy
Game Hacking/Cheating

Modification of a platform’s system memory during game play, or modification of files that comprise a game, to achieve a desired effect during game play.

- Building a bot
- Manipulating memory
- Operating a proxy
- Map hacking [IEEE S&P 2011]
Information Leakage in Cloud Computing

Virtual machine shares the hardware, and data could be leaked from one VM to another.
Information Leakage in Cloud Computing

Virtual machine shares the hardware, and data could be leaked from one VM to another.
Information Leakage in Cloud Computing

Virtual machine shares the hardware, and data could be leaked from one VM to another.
Program Analysis

- Data flow integrity [OSDI’06]
Program Analysis

- Data flow integrity [OSDI’06]
- Value-invariant discovery (e.g., DAIKON)
Program Analysis

- Data flow integrity [OSDI’06]
- Value-invariant discovery (e.g., DAIKON)
- Value-based profiler
Program Analysis

- Data flow integrity [OSDI’06]
- Value-invariant discovery (e.g., DAIKON)
- Value-based profiler
- Software-piracy detection using values [ICSE’11]
Outline

1. Overview
   - Course Goals
   - Course Style

2. Course Content
   - Foundations
   - Techniques
   - Tools
   - Applications

3. Course Project

4. Course Policy

5. Summary
## Course Projects

TBA
Outline

1. Overview
   - Course Goals
   - Course Style

2. Course Content
   - Foundations
   - Techniques
   - Tools
   - Applications

3. Course Project

4. Course Policy

5. Summary
Grading Policy

- 40% In-Class Presentations
Grading Policy

- 40% In-Class Presentations
- 10% Class participation
Grading Policy

- 40% In-Class Presentations
- 10% Class participation
- 10% Paper review/summary
Grading Policy

- 40% In-Class Presentations
- 10% Class participation
- 10% Paper review/summary
- 40% Class Project
Grading Policy

- 40% In-Class Presentations
- 10% Class participation
- 10% Paper review/summary
- 40% Class Project
- **Exceptional work** will be rewarded appropriately
Other Policy

Late Policy

No late submission
Other Policy

**Late Policy**
No late submission

**Collaboration Policy**
Encouraged, but limit the team member to at most two students.

Cheating Policy
Strictly follow the university policy on cheating and plagiarism
Other Policy

Late Policy
No late submission

Collaboration Policy
Encouraged, but limit the team member to at most two students.

Cheating Policy
Strictly follow the university policy on cheating and plagiarism
<table>
<thead>
<tr>
<th>Policy</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Late Policy</td>
<td>No late submission</td>
</tr>
<tr>
<td>Collaboration Policy</td>
<td>Encouraged, but limit the team member to at most two students.</td>
</tr>
<tr>
<td>Cheating Policy</td>
<td>Strictly follow the university policy on cheating and plagiarism</td>
</tr>
</tbody>
</table>
Outline

1 Overview
   • Course Goals
   • Course Style

2 Course Content
   • Foundations
   • Techniques
   • Tools
   • Applications

3 Course Project

4 Course Policy

5 Summary
Summary

- **Data** is a critical aspect of computer system.
**Summary**

- **Data** is a critical aspect of computer system.
- **Reverse Engineering** of data is possible.
Data is a critical aspect of computer system.

Reverse Engineering of data is possible.

Reverse Engineered data has a wide impact on many applications.
Summary

- **Data** is a critical aspect of computer system.
- **Reverse Engineering** of data is possible.
- **Reverse Engineered** data has a wide impact on many applications.

Next Lecture

Overview of operating system (manage memory data), and file system (manage disk data).