



# ANRC

2014 - 2015



ANRC is an industry-leading firm focused on delivering Advanced Cyber Security Training, Enterprise Threat Assessments, and Innovative Security Solutions. ANRC draws upon decades of experience obtained at the front lines of today's cyber conflicts to develop its progressive and comprehensive security solutions for the defense of private enterprises.

## OUR MISSION

ANRC was formed with two visions in mind. The first was to provide the very latest as well as the very best computer security education possible to help security professionals and administrators defend their information systems. The second vision was to administer our education through a revolutionary new approach, an approach that gives our customers knowledge that will be truly usable, valuable, and retainable. Over the years we've delivered training to thousands of students worldwide, while developing a world-class portfolio of courses that address today's most challenging cyber security topics.

Through our focused execution and vigilance to serve as an innovator in the cyber security industry, our vision and mission has broadened, resulting in additional service and product offerings that further enable our customers to protect their enterprises from advanced threats.






## ADVANCED CYBER SECURITY TRAINING

In our comprehensive, lab-intensive courses students will master not just new tools and tricks, but also a repeatable methodology they can employ for years to come. Our goal is for students to leave with honed, cutting-edge skills they can immediately apply in their daily work.

Students attending an ANRC course experience a progressive and energetic atmosphere, and should prepare to be challenged throughout the entire 5-day session.

- **70%+ Lab Environment**
- **Vendor-Neutral / Hands-On Experiences**
- **Real-World Demonstrations**
- **Interactive Classroom Discussions**
- **All courses are led by the industry's most experienced Subject Matter Experts**

### Course Key

-  **Ebook Included**
-  **College Credit Available**
-  **Course Material Downloads Available**
-  **Certification Track**
-  **Supplemental Book Included**

# TABLE OF CONTENTS

## CERTIFICATION TRACKS

---

Malware Triage Analyst	4
Malware Reverse Engineer	5
Penetration Testing	6
Kernel Security Developer	7
Mobile Security Analyst	8
Network Security Analyst	9

## CYBER SECURITY

---

Introduction to Cyber Security	10
--------------------------------	----

## PROGRAMMING

---

Introduction to C Programming	11
Introduction to Python	12
Windows System Programming	13

## OPERATING SYSTEMS

---

Understanding Operating Systems	14
Windows Internals	15
Operating System Intrusion Analysis	16
Red Hat Linux Kernel Internals	17
Windows Kernel Internals and Debugging	18
Windows Kernel Programming and Dump Analysis	19
Windows Kernel Rootkits	20

## MOBILE SECURITY

---

iPhone Development, Exploitation and Reversing	21
Android Development, Exploitation and Reversing	22

## REVERSE ENGINEERING

---

Behavioral Malware Analysis	23
Assembly for Reverse Engineers	24
Introduction to Malware Reverse Engineering	25
Advanced Malware Reverse Engineering	26

## HACKING

---

Hacking With Python	27
Penetration Testing	28

## NETWORK THREAT ANALYSIS

---

Network Traffic Analysis	29
Advanced Network Traffic Analysis	30
Malicious Network Traffic Analysis	31
Cyber Threats Detection and Mitigation	32

# MALWARE TRIAGE ANALYST

Introduction to Cyber Security  
CS-100 Page 10



Understanding Operating Systems  
OS-150 Page 14



Introduction to Python  
PR-130 Page 12



Behavioral Malware Analysis  
RE-100 Page 23

# MALWARE REVERSE ENGINEER

Understanding Operating Systems  
OS-150 Page 14



Behavioral Malware Analysis  
RE-100 Page 23



Operating System Intrusion Analysis  
OS-250 Page 16



Introduction to C Programming  
PR-110 Page 11



Assembly For Reverse Engineers  
RE-150 Page 24



Introduction to Malware Reverse Engineering  
RE-201 Page 25



Advanced Malware Reverse Engineering  
RE-300 Page 26

# PENETRATION TESTING

Understanding Operating Systems  
OS-150 Page 14



Behavioral Malware Analysis  
RE-100 Page 23



Operating System Intrusion Analysis  
OS-250 Page 16



Introduction to Python  
PR-130 Page 12



Hacking With Python  
HK-200 Page 27



Penetration Testing  
HK-210 Page 28

# KERNEL SECURITY DEVELOPER

Windows Internals  
OS-210 Page 15



Windows System Programming  
PR-210 Page 13



Red Hat Linux Internals  
OS-300 Page 17



Windows Kernel Internals and Debugging  
OS-310 Page 18



Windows Kernel Programming and Dump Analysis  
OS-400 Page 19



Windows Kernel Rootkits  
OS-450 Page 20

# MOBILE MALWARE ANALYST

Introduction to C Programming  
PR-110 Page 11



Introduction to Python  
PR-130 Page 12



Behavioral Malware Analysis  
RE-100 Page 23



iPhone Development, Exploitation and Reversing  
MS-301 Page 21



Android Development, Exploitation and Reversing  
MS-300 Page 22



# NETWORK SECURITY ANALYST

Network + or Equivalent  
(See index for suggested vendor courses)



Introduction to Cyber Security  
CS-100 Page 10



Network Traffic Analysis  
NT-100 Page 29



Advanced Network Traffic Analysis  
NT-200 Page 30



Introduction to Python  
PR-130 Page 12



Malicious Network Traffic Analysis  
NT-300 Page 31



Cyber Threats Detection and Mitigation  
NT-301 Page 32

# INTRODUCTION TO CYBER SECURITY

In 2014 the world has continued to watch as breach after breach results in millions of credit card and personal information records being posted on the Internet. The Internet Storm Center reports an average of over 700,000 detected intrusion attempts daily – and that’s only the events they catch! There is no question that Cyber Security is a necessity and an increasing global concern, the challenge is where to start the daunting task of securing your infrastructure, training your end users and preparing your organization to face the year ahead.

Introduction to Cyber Security is **the** foundational training for all users whether management, IT, end user or programmer. Equip your team with the up to date knowledge of threats we all face and the hands-on skills to address them. With information culminated from the most trusted sources; CERT, NIST, DHS and others, this course presents an objective, complete, and cutting edge view of our current environment as well as a vision of the near future of Cyber Security.

## ATTENDING STUDENTS WILL LEARN:

- Overview of the Hacking Cycle
- Phases of Network Reconnaissance
- Use and Methodology of Network Scanning Tools
- DNS Analysis and Manipulation
- Malware Types
- Defensive Postures
- Security Appliance Types and Uses
- Defense in Depth Model
- Access Control Mechanisms
- Authentication Protocol Types and Uses
- Encryption Protocol Types and Uses
- VPN Protocol Types and Uses

## WHO SHOULD ATTEND:

- IT Administrators seeking an understanding of security threats and basic mitigation controls
- Database Administrators desiring an increased security awareness
- Managers of network resources who want an understanding of the current threat landscape
- End Users needing a heightened awareness of Cyber Security

## PREREQUISITES:

- There are no prerequisites, however a basic understanding of computer and network
- Terminology is recommended

## COURSES THAT FOLLOW:

- Network Traffic Analysis - Page 29
- Advanced Network Traffic Analysis - Page 30
- Introduction to Python - Page 12
- Malicious Network Traffic Analysis - Page 31
- Cyber Threats Detection and Mitigation - Page 32

# INTRODUCTION TO C PROGRAMMING

For reverse engineers, exploit developers, and vulnerability researchers it all starts here. With all the advancement of operating systems and hardware devices, C code is still the dominant language around. Understanding how to write, read, and debug C programs are essential to a successful career in computer security.

This course covers the basics of understanding the C programming language. As with many of our "Introduction" courses we start off light and slow to give students the practical skills necessary to build on during and after class. Upon course completion students will have a firm grasp on how to design, implement, and debug console applications written in C.

## ATTENDING STUDENTS WILL LEARN:

- Variable usage
- Repetition (loop) statements
- Decision Branches
- File Input and Output
- C Standard Libraries
- Creating Custom Libraries and Functions
- An Understanding of Computational Expressions
- Structures
- Pointers
- Advanced Debugging Techniques

## WHO SHOULD ATTEND:

- Application Programmers
- Forensic Analysts who need to identify malicious software
- Reverse Engineers

## PREREQUISITES:

Students should possess a strong understanding of computer systems. Completion of Understanding Operating Systems is recommended.

## COURSES THAT FOLLOW:

- Behavioral Malware Analysis - Page 23
- Assembly for Reverse Engineers - Page 24
- Introduction to Malware Reverse Engineering - Page 25
- Advanced Malware Reverse Engineering - Page 26
- iPhone Development, Exploitation and Reversing - Page 21
- Android Development, Exploitation and Reversing - Page 22

# INTRODUCTION TO PYTHON

Python is one of the latest and greatest scripting languages in use today. Because of its flexibility and integration with other existing programming languages as well as its powerful object oriented design, Python is taking over by storm in the security analyst arena. This introductory course will give you a great foundation in how to use Python to build powerful scripts. A course designed for novice programmers, or those wanting to adopt yet another programming language into their arsenal, we guarantee you'll receive more hands-on practical experience from this course than any other available in the industry today. Come see for yourself how we can turn a basic user into a programmer in 5 days!

## ATTENDING STUDENTS WILL LEARN:

- Python Object Types
- Numeric Types
- Strings
- Lists and Dictionaries
- Python Statements
- Assignments, Expressions, and Prints
- if Tests and Syntax Rules
- Repetition Statements
- Functions
- Modules
- Classes

## WHO SHOULD ATTEND:

- Network Security Analysts seeking to automate traffic analysis
- Penetration Testers
- Reverse Engineers
- Incidence Response Team Members

## PREREQUISITES:

- There are no prerequisites, however a basic understanding of computer and network terminology is recommended

## COURSES THAT FOLLOW:

- iPhone Development, Exploitation and Reversing - Page 21
- Android Development, Exploitation and Reversing - Page 22
- Behavioral Malware Analysis - Page 23
- Malicious Network Traffic Analysis - Page 31
- Cyber Threats Detection and Mitigation - Page 32

# WINDOWS SYSTEM PROGRAMMING

Where the Windows Internals course explains how the Windows Operating System works this course will show you how to go about writing code to interface with it. Using the Windows System Programming book written by Johnson M. Hart along with customized additions by ANRC course development teams this course will teach you how to use the Windows API to perform practical application development and Windows system development. Additionally there is a heavy emphasis on the skills and techniques necessary to be an effective Malware Analyst, Penetration Tester or Exploitation Developer.

## ATTENDING STUDENTS WILL LEARN:

- Writing C code that uses the Windows API
- Creating Files, Processes and Threads
- Memory Management (Memory Mapped Files, DLL's, Virtual Memory, Heap Memory)
- Dynamic Linked Libraries (DLLs)
- Input / Output
- Exception Handling
- Hooks
- Windows Registry
- Windows Security
- Process Management
- Windows Sockets
- Malicious Application Development

## WHO SHOULD ATTEND:

- Penetration Testers
- Reverse Engineers
- Incidence Response Team Members

## PREREQUISITES:

- Students should possess a solid understanding of the C Programming Language.  
The Introduction to C Programming course is an excellent preparation for attendance.

## COURSES THAT FOLLOW:

- Red Hat Linux Kernel Internals - Page 17
- Windows Kernel Internals and Debugging- Page 18
- Windows Kernel Programming and Dump Analysis - Page 19
- Windows Kernel Rootkits - Page 20

# UNDERSTANDING OPERATING SYSTEMS

## LEARNING HOW MODERN OPERATING SYSTEMS WORK

Have you ever wondered what's under the hood of a modern operating system? What components are vulnerable to attack? These questions and more are answered in our Understanding Operating Systems class, combining theory and principles with hands on practice with the world's most popular operating systems.

This course covers the principles of process, memory and I/O management that underpin all modern operating systems, including a hands-on look at how they are implemented in Microsoft Windows, Linux, Mac OS X and Solaris. Learn how Windows and Linux operating systems deal differently with areas including network configuration, security and user/group management. After attending this course students will understand how the components of operating systems work and interact - providing an excellent foundation for courses in malware analysis, intrusion analysis and penetration testing.

### ATTENDING STUDENTS WILL LEARN:

- Computer Architecture Basics
- Process, Memory and I/O Management
- Networking Configuration
- Command Shell Tools
- Security Mechanisms
- User & Group Management

### WHO SHOULD ATTEND:

- Security Analysts who are going on to study intrusion analysis and related fields
- Malware Analysts needing an understanding of how operating systems work
- Programmers who need to understand issues that affect software development

### PREREQUISITES:

- Attending students should have a basic familiarity with one or more common operating systems. Experience with VMware is an advantage, but not necessary.

### COURSES THAT FOLLOW:

- Introduction to C Programming - Page 11
- Assembly for Reverse Engineers - Page 24
- Introduction to Malware Reverse Engineering - Page 25
- Advanced Malware Reverse Engineering - Page 26

# WINDOWS INTERNALS

As Information Technology Professionals strive to broaden their security skills to an advanced level, having an in-depth understanding of the Microsoft Operating System becomes essential. Whether you're monitoring a system for real-time anomalies, or performing an analysis from a forensics perspective, knowing how the Microsoft Operating System functions is imperative.

In this course you'll learn about the core components of the modern Windows Operating System, and how they inter-operate with one another. Using the internationally renowned Windows Internals book, our qualified instructors will walk you through and explain over 30 labs designed to give you insight into the world of the Windows Operating System.

## ATTENDING STUDENTS WILL LEARN:

- Core system and management mechanisms
- Services, Processes, Threads, and Jobs
- Virtual Memory
- Objects and Handles
- Registry Functions
- Analysis Tools
- Internal data structures with a kernel debugger (liveKD)
- Windows Security Model and Authorization
- Process Internals
- Thread Internals
- I/O Processing and Management
- Device Drivers
- Windows Network Stack
- Windows API
- Windows Driver Framework

## WHO SHOULD ATTEND:

- Security Analysts who are going on to study intrusion analysis and related fields
- Malware Analysts needing an understanding of how operating systems work
- Programmers who need to understand issues that affect software development
- Reverse Engineers seeking insight into the Windows API

## PREREQUISITES:

- Understanding Operating Systems is an excellent preparatory course for Windows Internals, but is not required. Previous C programming experience would also be beneficial.

## COURSES THAT FOLLOW:

- Windows System Programming - Page 13
- Red Hat Linux Kernel Internals - Page 17
- Windows Kernel Internals and Debugging- Page 18
- Windows Kernel Programming and Dump Analysis - Page 19
- Windows Kernel Rootkits - Page 20

 Microsoft Windows Internals, 6th Edition

 Course Material Downloads Available

# OPERATING SYSTEM INTRUSION ANALYSIS

## INVESTIGATING SYSTEM INTRUSIONS AND PROTECTING AGAINST THEM

While the security industry and professionals continue to seek the silver bullet solution to all Intrusion Detection System pitfalls, there is no completely effective solution that eliminates human decision making as part of the analysis process. Investing solely in technical solutions without making a comparable investment in an organization's analytical personnel leaves assets exposed and has landed some of the largest firms squarely in the front-page news.

Discovering exactly how an attacker has infiltrated a system can be difficult. This course teaches students how to correctly create a baseline of an operating system and then use it to detect unwanted activities when they occur. In this course students will learn the most useful commands and tools that can be employed during investigation to reveal the significant indicators of infiltration and compromise. Both the Windows and Linux Operating Systems are covered in this course.

### ATTENDING STUDENTS WILL LEARN:

- Proactive Auditing / Monitoring
- Establishing a Baseline
- Looking for Signs of Intrusions
- Evidence of Rootkits
- Examining Log Files
- Examining User and Group Accounts
- Auditing Services and Daemons
- MD5, SHA1 Hashing
- Digital Signature Verification

### WHO SHOULD ATTEND:

- Incident Responders who need to quickly address a security breach
- Forensic Investigators who need to identify malicious intrusions
- Exploitation Analysts needing operating system knowledge
- Malware Analysts requiring a thorough understanding of operating system intrusions

### PREREQUISITES:

- Microsoft Windows and Linux Command Line Experience
- VMWare or other Virtualization Software recommended
- The Operating Systems Fundamentals and Basic Malware Analysis Courses are highly recommended

### COURSES THAT FOLLOW:

- Introduction to C Programming - Page 11
- Assembly for Reverse Engineers - Page 24
- Introduction to Malware Reverse Engineering - Page 25
- Advanced Malware Reverse Engineering - Page 26



# RED HAT LINUX KERNEL INTERNALS

## LEARN HOW THE LINUX KERNEL FUNCTIONS AND START DEVELOPING MODULES

Red Hat Linux internals teaches you all the fundamental requirements necessary to understand and start developing for the Linux kernel. Get deep into the internals of the Linux OS and begin to develop kernel modules for the latest Red Hat distribution. From kernel module implementation through memory and process management, which include I/O, debugging, and file system topics, this introductory course is all encompassing. Red Hat Linux Internals is designed to provide a solid foundation for those looking to start low level kernel development for this popular operating system. With the majority of the class being hands-on each student will be issued a laptop with all the necessary tools to learn the skills and essential methodologies required to be a Red Hat Linux developer.

### ATTENDING STUDENTS WILL LEARN:

- How to setup a development environment for Red Hat Linux
- How the Red Hat Linux kernel functions
- How to develop kernel modules for the Red Hat Linux operating system which interact with I/O, memory, processes, file systems, and networking
- Obfuscation methods used by attackers to escape detection
- Configuring, compiling and building the Linux Kernel using RedHat 7.0 Distribution.
- Kernel subsystems responsible for the core internals of the OS including the Process and Memory Management, CFS Scheduler, Virtual File Subsystem, Block I/O Layer,
- Page Cache, and System Call Interface
- Debugging a kernel vmcore using the crash utility, gdb, exec and kdump.
- SystemTap and developing user land administration scripts using stap.
- Writing, compiling and debugging Loadable Kernel Modules (LKM)

### WHO SHOULD ATTEND:

- Developers looking to get a grasp on Red Hat Linux kernel internals and development
- individuals who have a solid understanding of low level Windows system programming and are looking to branch into the Linux OS

### PREREQUISITES:

- Experience in C programming
- Knowledge of systems programming in a UNIX or Linux environment (register-level hardware programming knowledge is recommended but not required)
- Familiarity with basic tools, such as vi, Emacs, and file utilities
- Familiarity with Unix development tools such as gcc and make

### COURSES THAT FOLLOW:

- Introduction to C Programming - Page 11
- Assembly for Reverse Engineers - Page 24
- Introduction to Malware Reverse Engineering - Page 25
- Advanced Malware Reverse Engineering - Page 26

# WINDOWS KERNEL INTERNALS AND DEBUGGING

This course provides an excellent foundation in the Windows kernel environment. Students will learn to work with the kernel debugger and how to harness its power to examine data structures, call stacks and variables. Learn to automate debugging and reverse engineering tasks using scripts. Take a deep dive into key components of the kernel, showing their internal operation with the help of kernel debugger extensions while highlighting potential indicators of compromise by rootkits.

## ATTENDING STUDENTS WILL LEARN:

- Kernel Architecture
- Execution Environment
- Synchronization
- Memory Management
- I/O Management
- Object Management
- Navigating Call Stacks
- Navigating Data Structures
- Kernel Debugging Tools and Extensions
- Anti-Malware Operations
- Malware Kernel Interactions and Exploitation

## WHO SHOULD ATTEND:

- Exploitation Analysts needing operating system knowledge
- Malware Analysts requiring a thorough understanding of operating system intrusions
- Programmers who need to understand issues that affect software development
- Reverse Engineers seeking insight into the Windows API

## PREREQUISITES:

- This entry level course requires attendees to have a solid understanding of operating system concepts, working knowledge of Windows and familiarity with C/C++. The
- Introduction to C Programming, Windows Internals and Windows System Programming Courses are recommended prior to attendance.

## COURSES THAT FOLLOW:

- Window Kernel Programming and Dump Analysis - Page 19
- Windows Kernel Rootkits - Page 20

# WINDOWS KERNEL PROGRAMMING AND DUMP ANALYSIS

Most security software on Windows is implemented in the kernel. This course starts with the basics of kernel mode software development and then progressively dives into the APIs and advanced programming techniques required to implement kernel mode software. The dump analysis part of this course is not about running “!analyze -v”, but using that as a starting point to get to the root cause of problems while applying kernel internals knowledge and debugging techniques along the way.

The hands-on lab exercises involve extensive kernel mode programming and debugging. This course does NOT cover development or debugging of user mode (Win32) applications or drivers for hardware devices like PCI and USB.

## ATTENDING STUDENTS WILL LEARN:

- Driver Development Environment
- Driver Programming Basics
- Asynchronous Execution
- Locking & Serialization
- Advanced Driver Programming
- Filter Drivers
- Memory Dumps
- Debugging Hangs
- Debugging Crashes
- Live Kernel Debugging

## WHO SHOULD ATTEND:

- Malware Analysts requiring a thorough understanding of operating system intrusions
- Programmers who need to understand issues that affect software development
- Reverse Engineers seeking insight into the Windows API

## PREREQUISITES:

- This is an intermediate level course and requires attendees to be proficient in C/C++ programming, have good working knowledge of the Windows kernel and WinDBG.
- The Introduction to C Programming, Windows Internal, Windows System Programming Courses and The Windows Kernel Internals and Debugging courses prepare attendees for this intermediate level course.

## COURSES THAT FOLLOW:

- Windows Kernel Rootkits - Page 20

# WINDOWS KERNEL ROOTKITS

To achieve maximum stealth and obtain unabated access to the system, rootkits execute in kernel mode. This course discusses the complete end-to-end modus-operandi of rootkits. Vulnerabilities in kernel mode drivers and how they are exploited to perform local privilege escalation and gain kernel mode execution is covered. Security enhancements that have been progressively added from Windows 7 to the latest update of Windows 8.1 are discussed along with some circumvention techniques. Finally the kernel interfaces, data structures and mechanisms leveraged by rootkits to perform post-exploitation steps are also covered.

The hands-on lab exercises involve extensive kernel mode reverse engineering, programming and debugging. Attendees will implement key components of a rootkit and test them on 64-bit Windows systems.

## ATTENDING STUDENTS WILL LEARN:

- Kernel Architecture
- Kernel Vulnerabilities
- Kernel Security Mitigations
- Kernel Security Bypass
- Driver Exploitation
- Hooking Techniques
- Filtering Mechanisms
- Covert Communications
- Stealth Behavior
- Detection Tools & Case Studies

## WHO SHOULD ATTEND:

- Forensic Analysts
- Malware Analysts requiring a thorough understanding of Kernel Rootkits
- Programmers who need to understand issues that affect software development
- Reverse Engineers seeking insight into Malware Behavior

## PREREQUISITES:

- This is an advanced level course and requires attendees to be fluent in C/C++ programming,
- have good working knowledge of the Windows Kernel internals and APIs and be able to use
- the kernel debugger to debug drivers. The Windows Kernel Programming and Dump Analysis
- courses prepare attendees for this course.

## COURSES THAT FOLLOW:

- Advanced Malware Reverse Engineering - Page 26

# IPHONE DEVELOPMENT, EXPLOITATION AND REVERSING

This course covers everything from iPhone development and application security to hacking the iOS and its applications. iPhone Development, Exploitation and Reversing is a laboratory intensive programming course designed for students who need a working knowledge of iPhone development and hacking. In this course we will explain how iOS works internally and discover key locations where data is stored and how to extract it. Students will also learn to use the tools needed to discover security vulnerabilities. Leave with the ability to deploy, execute and test your own developed programs using an iOS debugger both in an emulation environment and on an actual handset.

## ATTENDING STUDENTS WILL LEARN:

- iPhone Architecture and Design
- Basic iOS Application Design with Objective C
- ARM Assembly used in iOS
- Decompiling and Reverse Engineering iOS Binaries
- Identifying Controller, Libraries, Variables and Method Names used by iOS Apps
- Harvesting Geo-Location Data
- iOS Forensics 101
- Jailbreaking
- Remote Data Mining of iOS Devices

## WHO SHOULD ATTEND:

- Security Analysts seeking to understand Mobile exploitation
- Network Security Engineers supporting a BYOD environment
- Mobile Application Developers
- Penetration Testers
- Incident Response Team Members
- Reverse Engineers
- Forensic Analysts

## PREREQUISITES:

- This is fast-paced course designed for developers and security professionals. Previous
- experience with Apple products as well as programming in C and scripting skills would be beneficial. Introduction to C Programming, Introduction to Python or Introduction to Perl
- are good foundations for this course.

## COURSES THAT FOLLOW:

- Android Development, Exploitation and Reversing - Page 22

# ANDROID DEVELOPMENT, EXPLOITATION AND REVERSING

Android Development, Exploitation and Reversing is a hands-on, intensive programming course designed to teach the fundamentals of software development for the Android platform. Through a combination of instructor-led demonstrations and laboratory programming assignments and challenges, students will build and enhance their practical knowledge of software development for the Android Operating System. Learn to write, execute and troubleshoot software both in an Android emulator and on Android devices.

After learning the basics, students will advanced in this course to reverse engineering malware, understanding the ARM architecture as well as modifying applications for their desired intent and mitigations against this. The class will conclude with creating an application that can remotely mine data from Android devices.

## ATTENDING STUDENTS WILL LEARN:

- Android Architecture and Design
- Android SDK
- Decompiling Reverse Engineering Android Applications
- Modifying Android Applications
- Forensics and Investigating Application Permission Integrity
- Jailbreaking Android Phones
- Remote Data Mining of Android Devices

## WHO SHOULD ATTEND:

- Security Analysts seeking to understand Mobile exploitation
- Network Security Engineers supporting a BYOD environment
- Mobile Application Developers
- Penetration Testers
- Incident Response Team Members
- Reverse Engineers
- Forensic Analysts

## PREREQUISITES:

- The ideal student should have C programming as well as Python, Perl or Java experience.
- Recommended prerequisites include: Introduction to C Programming, Introduction to Python,
- and the Introduction to Perl or Introduction to Java courses.

## COURSES THAT FOLLOW:

- iPhone Development, Exploitation and Reversing - Page 21

# BEHAVIORAL MALWARE ANALYSIS

## LEARN HOW TO PERFORM DYNAMIC MALWARE ANALYSIS

Basic Malware Analysis teaches you all the fundamental requirements necessary to analyze malicious software from a behavioral perspective. Students will gain the skills to set up a controlled sandbox environment and use system monitoring tools to quickly analyze a software sample and its malicious affects to the system. From simple keyloggers to massive botnets, this class introduces students to a wide variety of current, real-world samples. Learn the tools and essential methodologies required to become an effective malware analyst!

### ATTENDING STUDENTS WILL LEARN:

- How to identify malware and discover it's capabilities
- How to setup a secure lab environment to analyze malicious software
- How to use open source tools to characterize malware samples quickly
- Obfuscation methods used by attackers to escape detection

### WHO SHOULD ATTEND:

- Threat operation analysts seeking to have a better understanding of malware
- Incident responders who need to quickly address a system security breach
- Forensic investigators who need to identify malicious software
- Individuals who have experimented with malware analysis and want to expand their malware analysis techniques and methodologies

### PREREQUISITES:

- Attending students should have a thorough understanding of Microsoft Windows, the
- Understanding Operating Systems course provides an excellent foundation for Malware
- Analysts to build on. Experience with VMWare software, although not required, would be
- beneficial. Knowledge of networking protocols and Wireshark filtering is recommended but not required.

### COURSES THAT FOLLOW:

- Introduction to Malware Reverse Engineering - Page 25
- Advanced Malware Reverse Engineering - Page 26



College Credit Available



Windows Registry Guide



Course Material Downloads Available

# ASSEMBLY FOR REVERSE ENGINEERS

## MAXIMIZE YOUR REVERSING SKILLS

Many employed as Analysts or Programmers may not have had the time to learn the assembly language- a skill that will save them precious time when effective analysis is needed most. Designed for Malware Analysts and Code Developers alike, Assembly for Reverse Engineers will equip students with the know-how to effectively read assembly, understand statements, and reverse machine code back to its higher-level equivalent.

Discover how a compiler transforms high level code into machine code, learn common assembly statements you'll see used in the real world, and then be introduced to writing your own code during this week long, lab-intensive course.

### ATTENDING STUDENTS WILL LEARN:

- Data Representation (Bits, Bytes, 2's Complement, number conversion)
- Stack Memory, Heap Memory, Stack Tracing
- Essential Assembly Instructions (most frequently used)
- X86 Memory Addressing Modes
- Repetition, Branching, Function Calls and Call Stack Tracing
- XOR Encryption and ASSM Obfuscation
- How to Combat Anti-Reversing
- How Compilers Work
- How to Reverse Malware

### WHO SHOULD ATTEND:

- Forensic Investigators who need to identify and examine malicious code on systems
- Exploitation Analysts needing reverse engineering skills
- Penetration Testers who want to develop their own tools
- Malware Analysts requiring a thorough understanding malicious code

### PREREQUISITES:

- Knowledge of the C Programming Language is recommended. The Understanding Operating Systems and the Introduction to C Programming courses are highly recommended.

### COURSES THAT FOLLOW:

- Introduction to Malware Reverse Engineering - Page 25
- Advanced Malware Reverse Engineering - Page 26



# INTRODUCTION TO MALWARE

## MASTER STATIC MALWARE ANALYSIS

Equipped with the dynamic malware analysis techniques from the Behavioral Malware Analysis course you're ready to venture into this more advanced course. During this five day class, students will learn Static Malware Analysis using a debugger and disassembler.

Using the OllyDbg Debugger and IDA Pro Disassembler in a controlled environment to identify exactly what the malware specimen does and how it's doing it. After you've mastered the evaluation portion of the class, you will learn how to patch a specimen to make sections inactive or crack the program to allow full access to areas that have been hidden or encrypted by the malware developer.

### ATTENDING STUDENTS WILL LEARN:

- Assembly language debugging fundamentals including:
- Conversion methodology from source code to assembly code
- Intel CPU memory management and structures
- CPU control flows and order of operations
- Olly Debugger including:
- Tool Features
- Stepping, Stepping Over and Running code
- Useful Plug-ins and Add-ons
- Breakpoint fundamentals and usage
- Patching and assembling executables
- Decrypting and decoding packed executables
- Windows PE Header and Import Address Table fundamentals
- Reversing DLL Malware
- DLL malware run as an application, DLL malware installed as a service, and
- DLL malware ran through the services controller

### WHO SHOULD ATTEND:

- Security analysts seeking to have a better understanding of malware
- Incident responders who need to quickly address a system security breach
- Forensic investigators who need to identify malicious software
- Individuals who have experimented with malware analysis and want to expand their malware analysis techniques and methodologies

### PREREQUISITES:

- Attending students should have a thorough understanding of Microsoft Windows.
- Students should have a basic understanding and be able to read Intel X86 Assembly.
- C Programming skills are recommended and PERL or Python scripting skills would be beneficial.
- Knowledge of networking protocols and Wireshark filtering is also recommended.

### COURSES THAT FOLLOW:

- Advanced Malware Reverse Engineering - Page 26

 Course Material Downloads Available

# ADVANCED MALWARE REVERSE ENGINEERING

Serving as the final class in our malware series Advance Malware Analysis will challenge you more than ever. Using the latest malware samples that are the hardest to reverse engineer we push our students to use every means necessary to defeat all defensive measures employed by Malware authors to wreak havoc across the Internet. Each malware sample analyzed in class will require first unpacking the sample and removing any software armoring or protection put in place to thwart the security analyst.

After the student successfully removes armoring agents they'll have to navigate past several anti-debugging techniques employed by the most elite malware samples today. Finally each sample will require skillful knowledge and usage of OllyDbg or IDA Pro tools with scripting abilities to reverse engineer the destructive code and determine exactly what the malware does.

## ATTENDING STUDENTS WILL LEARN:

- Malicious document analysis
- Extracting and analyzing embedded shell script from documents
- Manually unpacking obfuscated malware
- Methods for Analyzing and Defeating Armored Malware
- Advanced Rootkits, DLL's and Windows Services
- Advanced Anti-Reversing Malware
- Configuring Visual Studio for IDA Pro plugin Development
- Using IDA Pro SDK for writing custom plugins.
- How to manually unpack modern Executable packers used to obfuscate hardened malware
- To Reverse engineer encryption routines and replication using Python
- To manually "carve and drop" document embedded malware (MSOffice, Adobe PDF)
- Advanced root kits tactics and techniques, including debugging strategies

## WHO SHOULD ATTEND:

- Reverse Engineers seeking to take their skill set to the next level
- Software Engineers who need a complete understanding of malware

## PREREQUISITES:

- \*\*Please note this course requires extensive skills and programming knowledge. It is recommended that the student first attend Introduction to Python , Assembly for Reverse Engineers, and Introduction to Malware Reverse Engineering before attending this course.
- Students should also have C Programming experience.

# HACKING WITH PYTHON

In today's rapid development environment, the Python scripting language's ability to merge scripting and object oriented programming together has made it an essential tool to master.

This course teaches you how to use Python to build powerful scripts and to use those scripts to push the limits of system security. Designed to be used for Gray Hat hacking, the course will detail code that can be used to ethically hack into applications and networks to test security. Reverse Engineers will benefit from the ability to automate Malware Analysis that the Python language affords. Python's ability to quickly automate analysis tasks in IDAPro and OllyDbg has made it the new scripting language of choice. The world's best hackers are using Python to do their handiwork. Shouldn't you?

## ATTENDING STUDENTS WILL LEARN:

- Automate tedious reversing and security tasks
- Design and program your own debugger
- Learn how to fuzz Windows drivers and create powerful fuzzers from scratch
- Code and library injection
- Soft and hard hooking techniques
- Sniff secure traffic out of an encrypted web browser session
- Use PyDBG, Immunity Debugger, Sulley, IDAPython, PyEMU, and more

## WHO SHOULD ATTEND:

- Any IT or Security specialist who'd like to automate routine analysis tasks including:
- Penetration Testers
- Reverse Engineers
- Incident Responders
- Security Operations Center Staff
- Network Security Analysts

## PREREQUISITES:

- The Introduction to Python course.

## COURSES THAT FOLLOW:

- Penetration Testing - Page 28



# PENETRATION TESTING

The ability of an organization to proactively test its own defenses is quickly becoming more of a requirement than a luxury. Penetration testing (Pentesting) is a method of evaluating the security of a computer system or network by simulating an attack by a malicious user. The process involves an active analysis of the system for any potential vulnerabilities that may result from poor or improper system configuration, and known and/or unknown hardware or software flaws. This analysis is carried out from the position of a potential attacker, and can involve active exploitation of security vulnerabilities. Any security issues that are found will be presented to the system owner together with an assessment of their impact and often with a proposal for mitigation or a technical solution. The intent of a penetration test is to determine feasibility of an attack and the amount of business impact of a successful exploit, if discovered.

## ATTENDING STUDENTS WILL LEARN:

- How to analyze their organization's network as a potential intruder would
- How to scan and launch attacks against encountered vulnerabilities
- How to gain access and escalate privileges without being detected
- How to recover from a successful intrusion
- How to use ANRC's Linux Security Auditor Distro to audit their networks

## WHO SHOULD ATTEND:

- Network Analysts seeking to develop security related skills
- Incident Responders needing to quickly address system security breaches
- Penetration Testers looking to reduce their detectability
- Threat Operations Analysts seeking a better understanding of network intrusions
- Network Administrators needing a better understanding of network security

## PREREQUISITES:

??

# NETWORK TRAFFIC ANALYSIS

## GROW YOUR ANALYTIC INTELLIGENCE

Network Traffic Analysis will enable students to differentiate between normal and abnormal network traffic. The course focuses on research, filtering and comparative analysis to identify the different types of activity on a network and attribute their source.

A subject matter expert will teach you security-related tactics, techniques and procedures for performing network analysis in today's ever-changing threat landscape. You'll learn to follow conversations through redirection as well as how to develop custom filters for non-dissected protocols. After attending this course, students will be able to hone in on the key events in a traffic capture and reconstruct the event time line.

### ATTENDING STUDENTS WILL LEARN:

- Internet Based Open Source Research
- Wireshark Protocol Analyzer
- Effective Capture and Display Filtering
- Decoding Protocols Lacking Dissector Support
- Tracing System, Service and User Transactions
- Recognizing Encoding Types
- Base-64 and URL Encoding
- Non-Dissected Protocol Analysis
- HTTP Header Analytics (User-Agents, Referrers, Accept Lines, etc)
- Cookie Tracking

### WHO SHOULD ATTEND:

- Network Analysts seeking to develop security related skills
- Incident Responders needing to quickly address system security breaches
- Penetration Testers looking to reduce their detectability
- Threat Operations Analysts seeking a better understanding of network intrusions
- Network Administrators needing a better understanding of network security

### PREREQUISITES:

- A Broad Understanding of TCP/IP and associated Protocols as well as knowledge of network hardware and segment Types is required. Previous exposure to Wireshark or other protocol analyzer(s) is also recommended but not required.

### COURSES THAT FOLLOW:

- Advanced Network Traffic Analysis - Page 30
- Malicious Network Traffic Analysis - Page 31
- Cyber Threats Detection and Mitigation - Page 32

# ADVANCED NETWORK TRAFFIC ANALYSIS

After mastering the basics of the Network Traffic Analysis course, students will gain a deeper understanding of global networking and large WAN based operations. This course is designed to hone analytic skills and prepare attendees for the rigors of traffic analysis on a global scale.

## ATTENDING STUDENTS WILL LEARN:

- Mapping Network Structures through Passive Analysis-
  - Identifying Firewalls
  - Identifying Routers
  - Identifying DNS Servers
  - Identifying Web Servers
  - Identifying DB's
  - Identifying DMZ's
  - Using open source tools for information gathering
  - IANA
  - Whois
  - Ping
  - O/S Identification
  - Network Taps / Sensor Fundamentals
  - Advanced analysis tools and techniques
- 
- WHO SHOULD ATTEND:  
Network administrators seeking a better understanding of their networks
  - Threat operation analysts seeking to have a better understanding of network intrusions
  - Incident responders who need to quickly address a system/network security breach
  - Penetration Testers looking for new ways to discover networks and their details

## PREREQUISITES:

- A firm understanding of networking concepts and terminology is required to be successful in this course. Network +, CCNA, or equivalent knowledge is recommended. Additionally, students should have completed the Network Traffic Analysis course prior to attending.

## COURSES THAT FOLLOW:

- Malicious Network Traffic Analysis - Page 31
- Cyber Threats Detection and Mitigation - Page 32

# MALICIOUS NETWORK TRAFFIC ANALYSIS

## UNCOVER SYSTEM INTRUSIONS BY IDENTIFYING MALICIOUS NETWORK ACTIVITY

There are a tremendous amount of network based attacks to be aware of on the Internet today and the number is increasing rapidly. In order to defend against these lethal network attacks they must be understood at the packet level. This course teaches you how to analyze, detect and understand common network based attacks used today in modern network warfare.

By learning to identify statistical patterns students will gain the skills to perform critical, real-time analysis in a production environment. Malicious Network Traffic Analysis employs several traffic analysis tools including Wireshark, RSA's NetWitness Investigator, and ColaSoft's Capsa 7 alongside custom tools developed by ANRC networking experts to show you how to detect these network attacks and be prepared to handle them.

### ATTENDING STUDENTS WILL LEARN:

- Strategic, Tactical, and Operational Analysis
- Situational Awareness
- Current Networking Trends in Malware
- IDS / IPS evasion techniques
- Flow and Statistical Analysis to help identify malicious behavior
- Coordinated Attacks
- Botnets
- Browser Attacks (Javascript, Obfuscation)
- Drive-By-Downloads
- OSI Layer 2,3,4,5,6,7 Attacks
- Social Engineering and Phishing Attacks
- Tunneling and Covert Channels

### WHO SHOULD ATTEND:

- Security analysts seeking to have a better understanding of network based malware and attacks
- Incident responders who need to quickly address a system security breach
- Forensic investigators who need to identify malicious network attacks
- Penetration Testers who want to learn how to obscure their traffic

### PREREQUISITES:

- A firm understanding of networking concepts and terminology is required to be successful in this course. Network +, CCNA, or equivalent knowledge is recommended. Additionally, students should have completed the Network Traffic Analysis course prior to attending.

### COURSES THAT FOLLOW:

- Cyber Threats Detection and Mitigation - Page 32

# CYBER THREATS DETECTION AND MITIGATION

## INVESTIGATING NETWORK INTRUSIONS AND PROTECTING AGAINST THEM

Cyber threats are increasing at an alarming rate every year and the ability for organizations to defend against full-scale, distributed attacks quickly and effectively has become much more difficult. An Intrusion Detection system affords security administrators the ability to automate the process of identifying attacks amongst the thousands of TCP and UDP conversations on their network, provided the IDS' signatures are well written.

Taught by leaders in network defense who work in the computer security industry, this course demonstrates how to defend large scale network infrastructure by building and maintaining intrusion detection systems and mastering advanced signature writing techniques. With Intrusion Detection Systems and trained network security auditors, organizations have a reliable means to prioritize and isolate the most critical threats in real time.

### ATTENDING STUDENTS WILL LEARN:

- IDS Types and Features
- Sensor Placement
- Sensor Configuration
- Signature Writing Basics
- IDS Evasion Techniques
- TCP and UDP Conversation Reassembly
- Signature Tuning
- Sensor Tuning
- Event Filtering and Post Detection Event Analysis
- Attacks on IDS Sensors and Mitigation Techniques

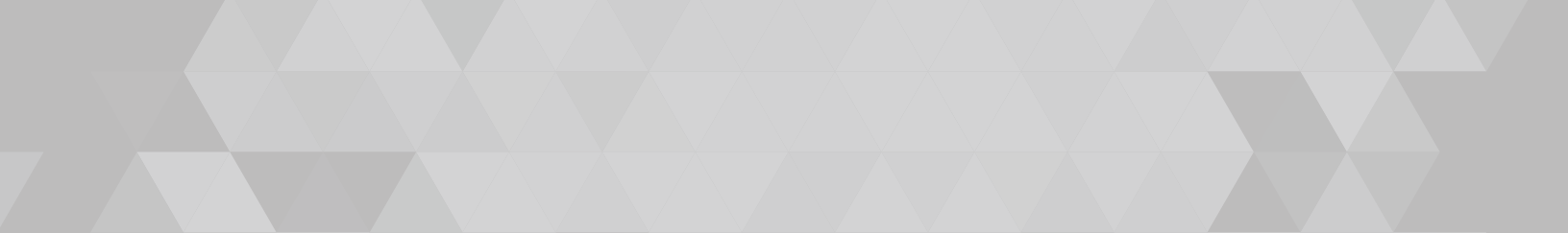
### WHO SHOULD ATTEND:

- Incident Responders who need to understand and react to IDS alerts
- Network Defenders seeking to automate threat detection
- Security Managers who desire to improve their defensive model
- IDS administrators who wish to improve their signature writing skills
- Security Operations Center Staff seeking to automate traffic analysis
- Penetration Testers looking to reduce their network visibility

### PREREQUISITES:

- A Firm understanding of TCP/IP
- Network + or Equivalent Knowledge or Background
- Both the Network Traffic Analysis and Malicious Network Traffic Analysis courses are recommended prior to attending





index and key to go here



**ANRC**

[www.anrc-services.com](http://www.anrc-services.com)  
800-742-7931  
[training@anrc-services.com](mailto:training@anrc-services.com)