Grayhash Training

- 트레이닝 전체 목록 -

강의명	교육일	강사	교육비	주기
Car Hacking Training - 몽이와 함께하는 자동차해킹	3일	정구홍	200만원	1~2개월
해커를 위한 임베디드 장비 개발 실습	3일	이원,정구홍	200만원	1~2개월
Windows Reverse Engineering	3일	강흥수	200만원	1~2개월
Web Browser 0-day Hunting	3일	현성원	200만원	2~3개월
Hardware(Embedded, IoT) Hacking	3일	정구홍	200만원	1~2개월
공유기 해킹 실습(ARM Exploitation)	2일	정구홍	150만원	2~3개월

- **강의 시간 :** 매일 오전 10시 ~ 오후 6시
- **강의 장소 :** 경기 성남시 분당구 정자동 정자역프라자 빌딩 GrayHash 트레이닝룸 **(5층)**
- **수강 정원 :** 10명 (최소 인원 5명)
- **중식 및 다과 제공**
- **개인 노트북 지참 필수**
- **교육비 VAT 별도**

Car Hacking Training -몽이와 함께하는 자동차해킹

- 강의 소개 -

최근 스마트카, 커넥티드카, 자율주행차와 같은 키워드들과 함께 차량에 대한 관심이 나날이 증가되고 있습니다. 하지만 이처럼 자동차의 기능이 확장되고 특히 인터넷과 주변기기에 연결될 수록 해커로부터 공격을 당할 수 있는 접점은 더욱 다양해지며, 자동차에 대한 해킹 공격은 자칫 인명사고로도 이어질 수 있기 때문에 보안에 대한 이해가 더욱 중요한 분야라고 할 수 있습니다.

본 트레이닝에선 자동차에 대한 다양한 공격 벡터들을 살펴보고, 이들 중 특히 CANbus, Bluetooth, USB, 그리고 SMS 공격 방식에 대하여 실습들과 함께 자세히 배워봅니다. 저희 GrayHash는 2015년부터 10종 이상의 다양한 차량용 장비들에 대한 보안 컨설팅을 수행해 왔으며, 이 경험을 기반으로 자동차 보안에 대한 Insight를 넓혀 드립니다.

- 주요 교육 내용 -

- CANbus Hacking
- Bluetooth based Car Hacking
- USB based Car Hacking
- SMS based Car Hacking
- 주요 자동차 해킹 사례 분석

- 상세 교육 내용 -

[1일차 - CANbus Hacking]

- CAN(Controller Area Network) 통신의 이해
- ECU(Electronic control unit)의 이해
- CANbus lines 연결하기 실습
- CANbus Hacking Tool 개발 실습
- CAN Message Sniffing 실습
- CAN Message 송신(Injection) 실습
- CAN Message 포맷의 구조 이해
- CANbus Controller와 Transceiver의 이해
- CAN Network 종류들의 이해

- 실제 차량의 OBD-II 포트와 연결하기 실습
- 주요 자동차 해킹 사례 및 취약점 발생 원리 분석
  - 2010 - Experimental Security Analysis of a Modern Automobile
  - 2011 - Comprehensive Experimental Analyses of Automotive Attack Surfaces
  - 2013 - Adventures in Automotive Networks and Control Units
  - 2014 - A Survey of Remote Automotive Attack Surfaces
  - 2015 - Remote Exploitation of an Unaltered Passenger Vehicle
  - 2015 - How to Hack a Tesla Model S
  - 2015 - Broadcasting Your Attack: Security Testing DAB Radio In Cars
  - 2015 - Drive it like you Hacked it: New Attacks and Tools to Wireles
  - 2016 - Advanced CAN injection technique for vehicle networks
  - 2016 - Car Hacking Research: Remote Attack Tesla Motors
  - 2017 - The CIA may be hacking cars
- 자동차 Attack Vector의 이해
  - Hardware Vulnerabilities Analysis (UART/JTAG/OBD-II)
  - Audio/Video/Navigation Device Vulnerabilities Analysis
  - Smart Device Vulnerabilities Analysis
  - Telematics Device Vulnerabilities Analysis
  - ETC : Web Browser, Smartphone APP, Radio Data System, Cloud Server

[2일차 - Infotainment System Hacking]

- Bluetooth based Car Hacking
  - Bluetooth 기초 (master/slave, piconet, bluetooth stack)
  - Bluetooth Packet 송수신 실습
  - Bluetooth Packet sniffing 실습
  - Bluetooth Packet의 포맷 분석
  - Bluetooth Stack 분석(Radio, LC, LMP, HCI, L2CAP, RFCOMM, SDP, OBEX)
  - Bluetooth Profile의 이해
  - Bluetooth Profile Scanning 실습
  - 차량용 디바이스의 주요 Profiles
  - 차량의 주요 Bluetooth 관련 기능들
  - 차량의 Bluetooth Packet Sniffing 및 분석
  - 차량 Bluetooth의 주요 Attack Vectors
  - Bluetooth Packet 변조 실습
  - Bluetooth Packet Fuzzing
- USB based Car Hacking
  - 차량 USB의 주요 Attack Vectors
  - USB(Universal Serial Bus) 기초
  - Beagle USB 480을 이용한 USB Packet Sniffing

- 주요 USB Packets 분석(Device, Configuration, Interface, Endpoint, String Descriptor)
- USB Packet의 세부 구조 분석(Transfer, Transaction, Token, PID, Address, Endpoint, CRC)
- USB Packet Fuzzing 환경 구축
- OrangePi를 이용한 USB fuzzer 개발
- Linux USB Gadget의 이해
- USB Stack Fuzzing
- USB based OS File system Fuzzing
- USB based Multi-media file Fuzzing

[3일차 - Telematics System Hacking]

- TCU(Telematics Control Unit)의 이해
- SMS PDU 포맷의 이해
- PDU 포맷 분석 실습
- PDU 데이터를 이용한 SMS 전송 실습
- Modem device를 이용한 SMS 전송 실습
- 보내는 SMS와 받은 SMS의 PDU 필드 값 비교
- MMS(Multimedia Messaging Service) 전송하기
- PDU-Header 및 TP-UDHI 필드의 이해
- IEI(Identity Element Identifie)와 Application Port의 이해
- SMS Hacking 사례 분석 : iPhone, Galaxy S5
- SMS Fuzzing 환경 구축
- Android telephony stack의 이해
- SMS 수신 시의 데이터 흐름 경로 분석 및 Fuzzing 지점 선정
- SMS Fuzzer 개발 방법의 이해
- Fake BTS를 이용한 SMS attack (USRP + OpenBTS)

.

해커를 위한 임베디드 장비 개발 실습

- 강의 소개 -

스마트플랫폼, 사무기기, 가전제품 등의 다양한 임베디드 기기들이 대중화 되고, 본격적인 사물인터넷(IoT)의 시대가 도래함에 따라 하드웨어 해킹의 위협 또한 증가되고 있습니다. 본 트레이닝에선 Linux 기반의 임베디드 장비 개발 실습을 통하여 임베디드 시스템의 내부 구조를 이해하고, 이를 통해 대상 기기에 대한 취약점 분석/해킹 능력을 향상시키는 것을 목표로 하고 있습니다.

참가자들은 본 트레이닝을 통해 임베디드 시스템의 하드웨어적인 구성 및 펌웨어의 구조와 세부적인 작동 원리를 파악할 수 있게 됩니다.

- 주요 교육 내용 -

- ARM 프로세서 기반 임베디드 시스템의 구조 이해
- 임베디드 시스템 개발 환경 구축 및 개발 실습
- 부트로더 및 운영체제의 이해
- 하드웨어 통신 프로토콜의 이해
- 주변장치 제어 방법의 이해

- 상세 교육 내용 -

[1일차 - 기초 펌웨어 개발]

- 실습 장비 Grayhash DanbiBoard-BASIC 소개 및 Spec 설명
- Firmware, Bootloader, OS, Filesystem의 이해
- CPU 입장에서 본 부팅 과정의 이해
- 실습 : 펌웨어 개발 환경 구축
- 펌웨어 코드 작성 이론 설명
- 실습 : 기본 펌웨어 개발 및 퓨징
- 펌웨어 코드 분석 및 GPIO의 이해
- UART를 이용한 DEBUG 메시지 출력
- 해커의 관점에서 본 UART 프로토콜
- 실습 : LED 등 주변장치 제어
- 하드웨어 통신 프로토콜 I2C의 이해
- 마무리 실습 : Character LCD 제어

[2일차 - 리눅스 시스템 개발]

- 실습 장비 Grayhash DanbiBoard-EDU 소개 및 Spec 설명
- 부트로더(Bootloader)의 이해
- 메모리, 메모리 컨트롤러 및 메모리 맵의 이해
- 해커의 관점에서 본 메모리 맵
- DDR2 RAM 초기화 방법에 대한 이해
- 실습 : RAM 초기화 및 RAM 테스트
- U-BOOT의 구조 및 커널 로딩 과정의 이해
- 실습 : U-BOOT 포팅 및 퓨징
- 부트로더를 이용한 펌웨어 업그레이드
- 해커의 관점에서 본 부트로더 (부트로더 보안체계 우회 사례들)
- 실습 : Linux Kernel 컴파일 및 퓨징
- Root File System의 이해
- 실습 : Root File System 빌드 및 퓨징
- 해커의 관점에서 본 Root File System
- 리눅스 부팅 과정 및 파티션 구조의 이해
- 마무리 실습 : 리눅스 시스템 부팅

[3일차 - 개발보드의 활용]

- 실습 장비 Grayhash DanbiBoard-PRO 소개 및 Spec 설명
- 디바이스 드라이버의 이해
- 실습 : 디바이스 드라이버를 통한 주변장치 제어
- 하드웨어 통신 프로토콜 SPI의 이해
- 해커의 관점에서 본 SPI 프로토콜
- TFT-LCD 디스플레이 연결
- Frame Buffer의 이해
- 실습 : Frame Buffer 장치를 이용한 TFT-LCD 제어
- 실습 : mp3 음원 및 동영상 재생
- USB 시스템 작동 구조의 이해
- 실습 : USB 키보드 및 랜카드 연결
- Ethernet 시스템 작동 구조의 이해
- 실습 : 네트워크 연결
- 마무리 실습 : 게임 에뮬레이터 실행하기

Windows Reverse Engineering

- 강의 소개 -

우리가 사용하는 데스크탑 및 노트북 PC는 가장 많은 악성코드와 해킹사건이 일어나고 있는 환경입니다. 세계의 대기업과 기관들은 자신들을 공격하는 해커들의 공격코드를 분석하기 위해 많은 노력을 기울이고 있습니다.

본 트레이닝은 Windows용 프로그램을 소스코드 없이 분석하는 방법을 익히는 것을 목적으로 하고 있으며, 이 과정에서 컴파일러가 생성하는 코드와 Windows 운영체제의 내부 구조를 배우고, 악성코드가 백신 진단을 회피하기 위해 어떻게 난독화를 하는지, 또한 이를 어떻게 해제하는지 역시 배우게 됩니다.

- 주요 교육 내용 -

- Reverse Engineering의 개념 및 방법 이해
- Windows PE 포맷의 이해
- x86 아키텍쳐 이해
- 악성코드 분석 실습
- 난독화 코드 분석 실습

- 상세 교육 내용 -

[1일차 - Reverse Engineering Basic]

- Disassembling/Decompiling의 이해
- 주요 Debugger 소개 (ollydbg/immdbg/windbg/gdb)
- Hooking/filtering/capturing tools 소개 (Sysinternal suite)
- x86 architecture의 이해 (Ring0/3, instruction, registers, memory 등)
- C/ASM 이해 (x86 Assembly, 함수 Stack frame, calling convention 등)
- ASM으로 함수 만들기 실습
- C/C++ Compiled codes 분석 (Debug/Release mode, 조건 분기, 루프 코드, Struct/class/OOP 등)
- Decompling(ASM -> C언어 변환) 실습

[2일차 - Windows Binary Reversing]

- Windows PE 포맷의 이해
- Windows에서의 process의 이해
- PE Parser 제작 실습
- Code Disassembling 기능 추가
- Game Reverse Engineering 실습
- 안전한 악성코드 분석 환경 준비
- 악성코드 및 Exploit 분석 실습
- DLL/Code Injection의 이해
- Code Injection tool 제작 실습

[3일차 - Analysing Packer and Obfuscator]

- Packer/Obfuscator의 이해 (Packers, protectors, custom packers)
- 리버싱을 방해하는 방법들 (anti-debug, anti-analysis)
- 유용한 Ollydbg Plugin들 소개
- 악성코드 Unpacking 실습
- Runtime Code 패치 실습
- Crackme 제작 및 분석 방해 기법 적용 실습
- 악성코드가 사용하는 난독화 도구의 목적과 기능 이해
- 코드 난독화(obfuscation) 분석 실습
- 상용 Protector 분석 실습

Web Browser 0-day Hunting

- 강의 소개 -

본 강의는 이미 오래전부터 현재까지도 심각한 보안 위협이 되어오고 있는 웹 브라우저 해킹에 대해 다루고 있습니다. 웹 브라우저의 다양한 취약점 유형들과 발생 원리에 대해 알아보고, 특정 취약점을 분석하고 공격하는 것을 직접 실습해 봄으로써 웹 브라우저 해킹 과정에 대해 구체적으로 이해할 수 있게 됩니다.

기본적인 이해를 마친 후에는 0-day 헌팅 실습을 통해 추후 웹 브라우저 버그 헌터가 되기 위한 기초를 닦는 것을 목표로 하고 있습니다.

- 주요 교육 내용 -

- 웹 브라우저 해킹 기초 설명
- 과거 주요 공격 사례 소개
- 웹 브라우저 취약점 발생 유형 설명
- 웹 브라우저 해킹 실습
- 웹 브라우저의 작동 원리 이해
- 웹 브라우저 Fuzzer 제작 실습
- 웹 브라우저 보호체계 무력화
- 그 외 웹 브라우저 해킹 관련 이슈들 소개

- 상세 교육 내용 -

[1일차 - Web Browser Hacking (Basic)]

< OT >

- 트레이닝 소개
- 웹 브라우저 해킹 연구 시작 계기

< 기초 설명 >

- 웹 브라우저 해킹 관련 기본 용어 소개
- 과거 주요 공격 사례 소개
- 웹 브라우저 취약점 발생 유형 설명

- Buffer OverFlow
- Out-Of-Bound Access
- Use-After-Free
- Type Confusion
- 그 외 (race condition, uninitialized memory, integer overflow)

• 웹 브라우저 취약점 공략 방법 설명
• 공개 웹 브라우저 취약점 찾아보기
- Exploit DB, Chromium, Project Zero, ETC

< 웹 브라우저 해킹 실습 >

• Windows XP + IE8 취약점 공격 실습
- 취약점 발생 원리 설명
- gflags.exe – full page heap, stack trace의 이해
- Heap Spray의 이해
- windbg를 이용한 디버깅 실습
- 쉘 코드 및 공격코드 작성 실습

• Windows XP + IE8 메모리 보호기법 우회 실습
- 메모리 보호기법(DEP) 우회법 설명 및 실습
- ROP(Return Oriented Programming) 공격 실습

[2일차 - Web Browser Hacking (0-day Hunting)]

< 기반 지식 설명 >

• HTML, 자바스크립트 기초
• 웹 브라우저의 구조, DOM의 구조, Element의 이해
• IE의 객체 생성 및 해제 구조, LFH에 대한 이해

< Fuzzer 제작 실습 >

• Fuzzing 원리 설명
- 파일 생성형 Fuzzer
- Javascript 실행형 Fuzzer

• 공개 웹 브라우저 Fuzzer 작동원리 분석
- 온라인에 공개된 최신 Fuzzer들 소개
- Grinder 소개
- Fuzzer 사용 실습

• Crash 분류하기
- Crash 분류하기
- Null-pointer dereferece Crash 이슈
- 다양한 Crash Case 소개
- Reproduce Issue 설명

• Fuzzer 제작 팁
  - Fuzzer Server (testcase 생성) 동작 설계
    1) 기본적인 틀 만들어보기
    2) testcase 생성 아이디어
• Fuzzer Client 동작 설계
  - Gflags - PageHeap, Memory Protection Issue

[3일차 - Web Browser Hacking (Advanced)]

< 웹 브라우저 보호체계 무력화 >

• IE Memory Protection & Bypass
  - VTGuard 설명
  - VTGuard 우회법 설명
  - Isolated Heap 설명
  - Isolated Heap 우회법 설명
  - Protected Free 설명
  - Isolated Heap / Protected Free 우회 실습
• 다양한 Memory Leak 방법들
  - Memory Leak 취약점 Case Study
  - Memory Leak 취약점을 이용한 Exploit Case Study
  - OOB Read/Write 취약점 Case Study
  - Flash 등 3rd party 취약점 Case Study
• Sandbox Escape에 대하여
  - Kernel Vulnerability 취약점 Case Study
  - Sandbox Vulnerability 취약점 Case Study

< Case Study >

• PWN2OWN에서 사용된 취약점들 분석

< Bug Bounty 프로그램 >

• Bug Bounty 소개
  - HP ZDI 제보 방법 및 절차
  - MSRC 제보 방법 및 절차
  - 그외 버그바운티 프로그램 소개

Hardware(Embedded, IoT) Hacking

- 강의 소개 -

스마트플랫폼, 사무기기, 가전제품 등의 다양한 임베디드 기기들이 대중화 되고, 본격적인 사물인터넷(IoT)의 시대가 도래함에 따라 하드웨어 해킹의 위협 또한 증가되고 있습니다. 본 트레이닝의 목적은 하드웨어 해킹의 원리를 이해하고, 단순히 하드웨어 해킹 툴을 이용하는 수준을 넘어서 자신이 원하는 하드웨어 기반의 해킹 툴을 직접 만들 수 있는 능력을 키우는 것에 초점을 맞추고 있습니다.

- 주요 교육 내용 -

- AVR 마이크로 컨트롤러 Programming 실습
- UART Hacking
- JTAG Hacking
- Flash Memory Dump

- 상세 교육 내용 –

[1일차 - AVR 프로그래밍]

- MCU(Micro Controller Unit)의 이해
- Atmega128A MCU 소개
- 개발 도구(Atmel studio 6.2) 설치
- ISP(In-System Programming)의 이해
- 범용/특수 입출력 포트의 이해
- LED, 모터, 스피커 제어 실습
- 7-Segment 및 Dot Matrix 제어 실습
- 온도 센서 제어를 통한 ADC의 이해
- 디지털 온도계 제작 실습
- UART 통신 실습
- 트랜지스터를 이용한 증폭 작용 실습
- 스위치를 이용한 입력 핀 사용 실습
- 인터럽트와 타이머의 이해

- PWM(Pulse Width Modulation)의 이해
- IrDA(적외선)을 이용한 무선 통신 실습
- AVR Firmware 추출하기
- AVR과 Arduino의 관계 이해
- Arduino 프로그래밍 실습

[2일차 - UART&JTAG Hacking]

[UART Hacking]

- 하드웨어 레벨 프로토콜의 이해 : UART, I2C(2-wire, TWI), SPI
- UART Programming의 이해
- Logic Analyzer를 이용한 UART 프로토콜 분석 실습
- UART Pin 찾기 실습
- 유무선 공유기 UART 연결 실습
- Baudrate 분석 실습
- 스마트폰(갤럭시S) UART 연결 실습
- UART 쉘을 이용한 바이너리 추출 실습
- 각종 기기들에 대한 UART 연결 시연 (스마트 TV, 홈 네트워크 시스템, CCTV, NAS 장비)
- UART 해킹 case by case

[JTAG Hacking]

- JTAG의 개념 및 작동 원리 이해하기 (TDI, TDO, TCK, TMS, TRST)
- TAPC(Test Access Port Controller) 상태도 이해하기
- JTAG 실전 활용 예제
- 상용 JTAG 툴 소개 : Riffbox Jtag, H-JTAG, AD-JTAG
- JTAG Pin Scanning : Jtagulator, JTAGenum
- JTAG을 이용한 펌웨어 획득 실습
- JTAG을 이용한 동적 디버깅 실습

[3일차 - Flash Memory 덤프 실습]

- Flash Memory 기초
- Winbond W25Q16BV Flash Memory의 이해
- 마이크로 프로세서 GPIO Handling 실습
- SPI 프로토콜의 이해
- Flash Memory Dumper 개발 실습
- Desoldering 실습
- Winbond W25Q16BV Serial Flash Memory Dump 실습
- 8-pin Test-clip 활용 실습
- Dump된 Firmware Image 분석 및 바이너리 파일 추출 실습

하드웨어 해킹-공유기 해킹 실습(ARM Exploitation)

- 강의 소개 -

스마트플랫폼, 사무기기, 가전제품 등의 다양한 임베디드 기기들이 대중화 되고, 본격적인 사물인터넷(IoT)의 시대가 도래함에 따라 하드웨어 해킹의 위협 또한 증가되고 있습니다. 본 트레이닝의 목적은 하드웨어 해킹의 원리를 이해하고, 단순히 하드웨어 해킹 툴을 이용하는 수준을 넘어서 자신이 원하는 하드웨어 기반의 해킹 툴을 직접 만들 수 있는 능력을 키우는 것에 초점을 맞추고 있습니다.

본 트레이닝은 공유기의 취약점을 찾아내고 exploit하는 방법에 대해 배우게 됩니다.

- 주요 교육 내용 -

- 공유기 펌웨어 추출 및 분석
- ARM Reversing & Exploitation

- 상세 교육 내용 -

[1일차 - 공유기 취약점 분석]

- 공유기의 구조 이해
- 타겟 디바이스의 펌웨어 획득 방법
- 공유기 펌웨어 이미지 구조 분석
- 파일 시스템 이미지 분석 및 바이너리 추출
- 임베디드 리눅스의 구조 이해
- 공유기의 취약점 유형들
- 공유기 취약점 탐지 전략
- 바이너리 정적/동적 분석 실습
- Buffer Overflow 취약점 탐지 및 분석
- 공유기 Remote Exploitation 공격 시연

[2일차 - 공유기 Remote Exploitation]

- ARM 실습 환경 구축
- ARM Assembly 문법 설명
- ARM Reverse Engineering 실습
- ARM 환경에서의 Buffer Overflow Attack

- ARM Shellcode 개발
- ARM Exploit 개발
- gdb를 이용한 디버깅 실습
- uPnP를 이용한 원격 해킹
- 공유기 Remote Exploitation 공격 실습
- 공유기 취약점 악용 시나리오
- 공유기 해킹 방지 방법