Forensic Impact according to the Firmware Manipulation

proneer

proneer@gmail.com

http://forensic-proof.com

Security is a people problem…

개요

1. 메인보드/주변장치 펌웨어 소개

2. 메인보드/주변장치 펌웨어 조작

3. 저장매체 펌웨어 소개

4. 저장매체 펌웨어 조작

개요

- 참고
  - **Advance and Development of Computer Firmware Security Research**, ISIP'09
  - **Firmware Manipulation and Forensic Impact and Current Best Practice**, ADFSL'10
  - **Malware and Steganography in hard disk firmware**, Springer-Verlag France 2011

Mainboard/Peripheral Firmware

펌웨어

- **펌웨어는 일반적으로 ROM(Read-Only Memory)에 기록**
  - PROM, EPROM, EEPROM
- **메인보드 펌웨어**
  - BIOS (Basic Input/Output System) ➔ **ROM BIOS**
  - UEFI (Unified Extensible Firmware Interface) ➔ **EFI BIOS**
- **주변장치 펌웨어; OPROM (Option ROM)**
  - LAN 카드
  - PCI 카드
  - 그래픽 카드
  - … …

펌웨어 역할

- **메인보드 펌웨어**
  - 장치 초기화
  - POST (Power On Self Test)
  - 부팅 작업
  - OS 로드

- **주변장치 펌웨어**
  - 메인보드 펌웨어에서 지원하지 않는 확장 기능 지원
    - ✓ 주변 장치 제어 및 관리

펌웨어 악성코드

- **CIH 바이러스 (1998 – 2002)**
  - BIOS 펌웨어 파괴

- **ACPI BIOS Rootkit**
  - John Heasman, Implementing and Detecting an ACPI BIOS Rootkit, Blackhat DC, 2006.
  - ACPI 펌웨어에 루트킷 설치

- **PCI Rootkit**
  - John Heasman, Implementing and Detecting an PCI Rootkit, Blackhat DC, 2007.
  - PCI 확장 카드 펌웨어에 루트킷 설치

전통적인 BIOS

- **1980’ ~ 1990’**
  - CPU, 칩셋, 버스, 컨트롤러와 통신하기 위한 인터페이스 역할
  - 저장매체로부터 운영체제 로드

- **1990’ ~**
  - 기본 기능 이외에 추가적으로 많은 기능 추가
    - ✓ APM (Advanced Power Management)
    - ✓ PnP (Plug and Play)
    - ✓ ACIP (Advanced Configuration and Power Interface)
    - ✓ U-Key (MSI 메인보드만 지원, 암호가 저장된 USB 디스크가 연결되야만 부팅)
    - ✓ TPM (Trusted Platform Module)
    - ✓ … …

전통적인 BIOS

- **특징**
  - **제한된 동작 환경**
    - ✓ 16비트 리얼모드(real mode), 32비트 플랫 모드(flat mode)
    - ✓ IA-32, IA-64에서 동작 불가
  - **패치와 업데이트의 어려움**
    - ✓ 정해진 환경에서만 가능
    - ✓ 어셈블리 언어를 이용한 프로그래밍
  - **독립된 플랫폼 환경**
    - ✓ 공개된 표준화된 환경의 부족
  - **보안적인 고려 미비**

새로운 펌웨어

- **UEFI (통합 확장형 펌웨어 인터페이스)**
  - 인텔에서 BIOS를 대체할 수단으로 EFI (Extensible Firmware Interface) 표준 제안
  - 이후 통합적인 논의를 위해 Unified EFI 포럼 조직

- **CSS (Core System Software) BIOS**
  - 포이닉스(Phoenix)와 마이크로소프트가 주도하여 계획
  - UEFI 에 패배

새로운 펌웨어

- **UEFI (통합 확장형 펌웨어 인터페이스) 특징**
  - 32비트 보호모드에서 동작
  - GUI 인터페이스
  - MBR (Master Boot Record) ➔ GPT (GUID Partition Table) 로 변화
    - ✓ MBR 한계 해결
  - C 언어 프로그래밍 가능
  - 네트워크 통신 가능
  - 추가적인 응용프로그램 로드 가능
  - … …

BIOS vs. UEFI

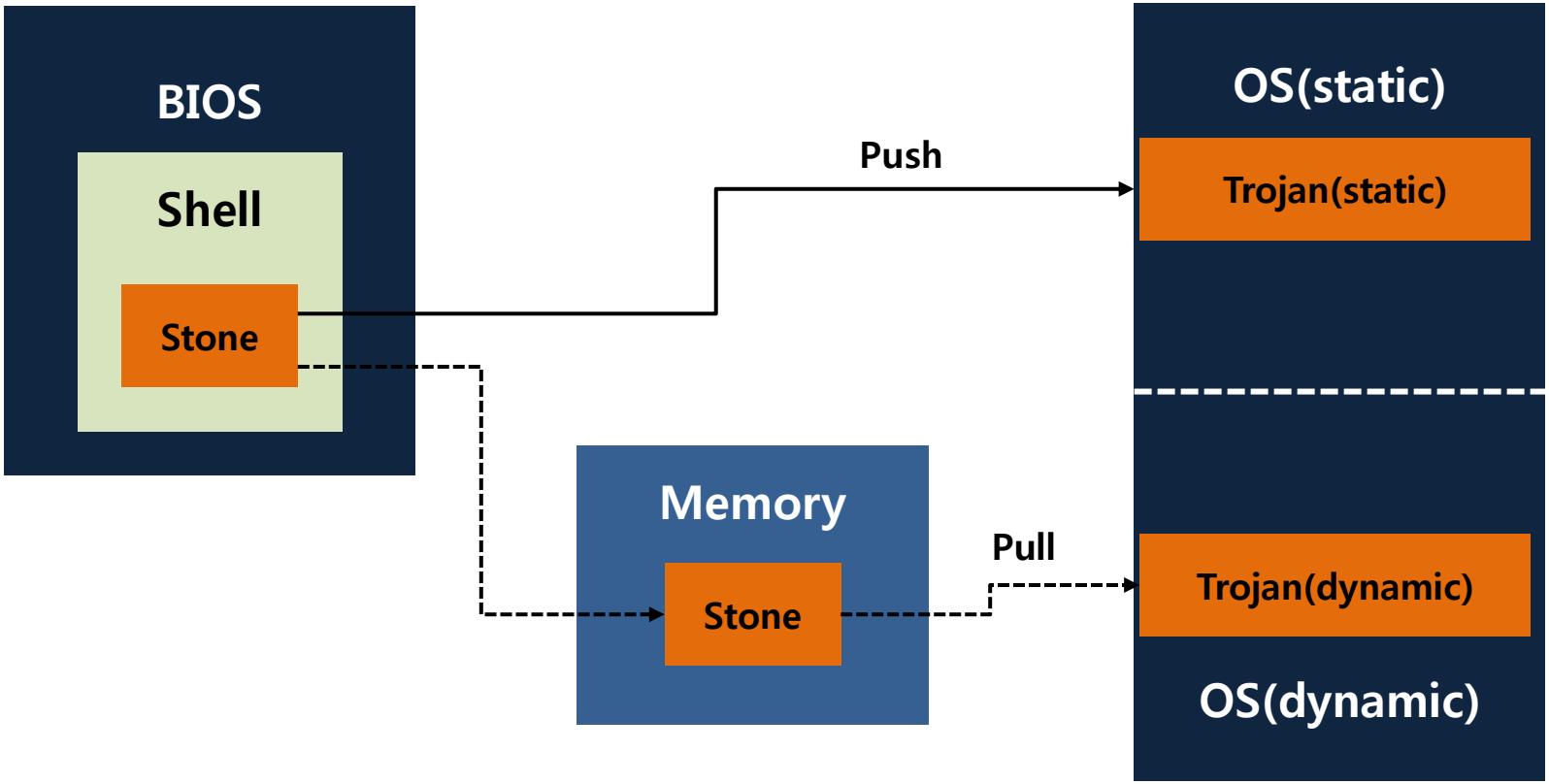
특징	Legacy BIOS	EFI BIOS
실행 모드	16비트 리얼 모드	32비트 보호 모드
사용자 인터페이스	텍스트 기반	그래픽 기반
프로그래밍 언어	어셈블리 언어	C 언어
드라이버 언어	어셈블리 언어	C 언어, EFI 바이트 코드
TCP/IP 지원 여부	미지원	지원
보안 설계	미지원	부분 지원
외부 드라이버 로드 지원 여부	미지원	지원, EFI 드라이버
외부 응용프로그램 로드 지원 여부	미지원	지원, EFI 응용프로그램

Mainboard/Peripheral Firmware Manipulation

펌웨어 위협

- **펌웨어 취약점**
  - 펌웨어의 취약점 악용
- **펌웨어에 악성코드 삽입**
  - 펌웨어 플래시 메모리에 악성코드 삽입
  - ACPI rootkit, PIC rootkit, BIOS rootkit
- **외부의 공격**
  - DoS (Denial-Of-Service)
  - 펌웨어 파괴 (예, CIH)

펌웨어 위협


BIOS
Shell
Stone
Push
OS(static)
Trojan(static)
Memory
Stone
Pull
Trojan(dynamic)
OS(dynamic)

펌웨어 위협 방지 방안

- **전통적인 BIOS 방지 방안**
  - AEGIS
  - ECC(Efficient Code Certification)
  - TPM(Trusted Platform Module)에 추가 ➔ 패치 및 업데이트 제한
- **신뢰적인 펌웨어 연구**
  - CTRM(Core Root of Trust for Measurement)
  - UTBIOS
- **펌웨어 조작 탐지 방안**

결론

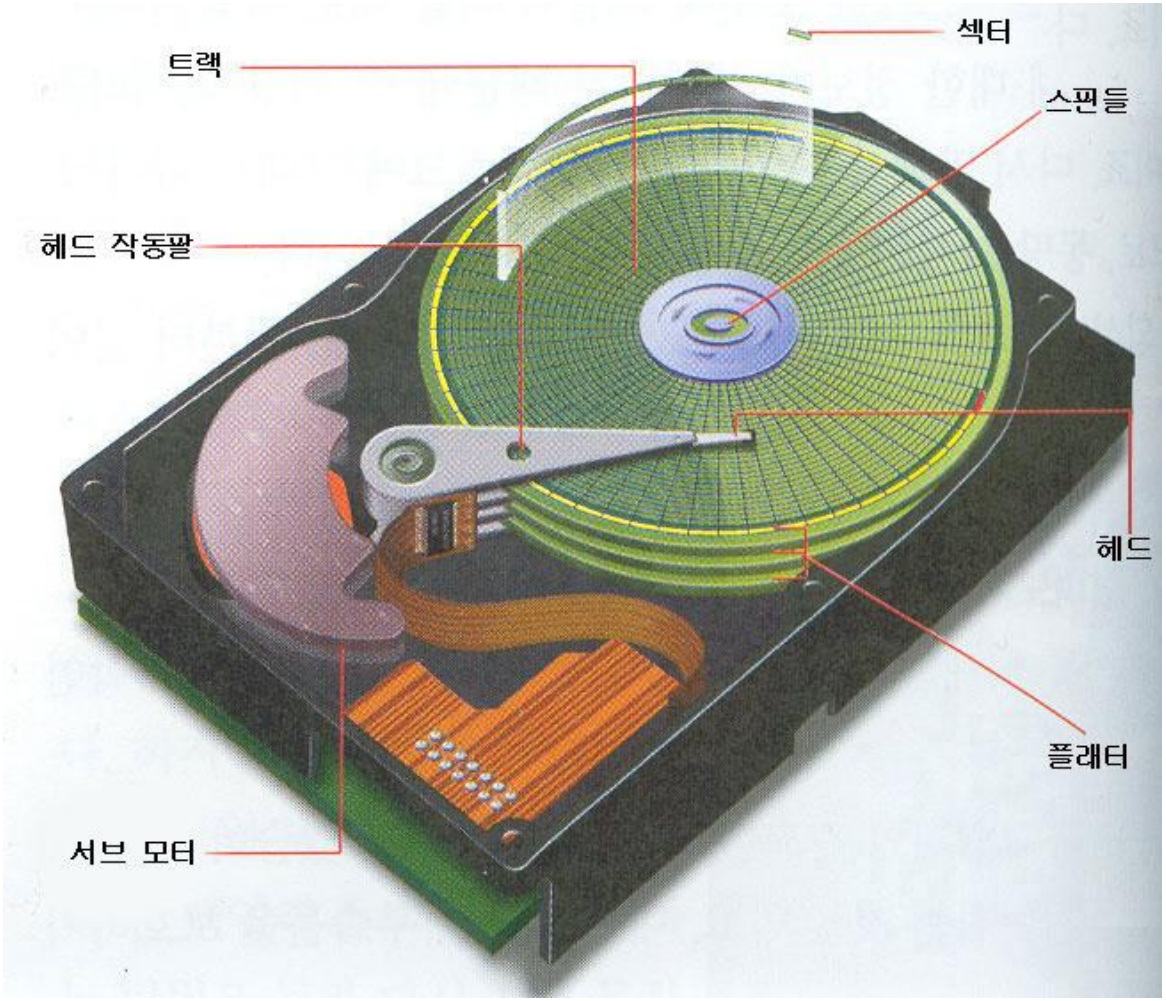
- **펌웨어 조작에 대한 큰 위협**
  - OS 시작 전에 동작
  - 쉬운 접근과 탐지가 어려움
  - 각 벤더마다 다른 형태의 구조 및 코드 사용

- **완벽한 방안이 있는가?**
  - BIOS에 대한 방안은 어느정도 연구
  - UEFI에 대한 방안은 아직 미흡 ➔ UEFI의 기능으로 BIOS 보다 더 큰 위협이 될 수 있음

- **그렇다면?**
  - **BIOS** – OS 로드 전에 탐지 및 치료 방안 연구, 신뢰적인 펌웨어 개발
  - **UEFI –** EFI 드라이버나 응용프로그램을 개발하여 보호, 신뢰적인 펌웨어 개발

Storage Firmware

디스크 구조

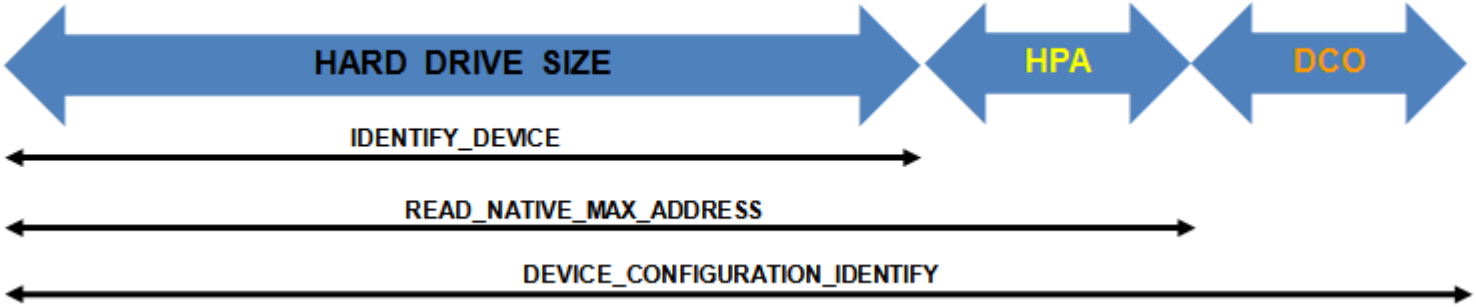


http://dsk114.com.ne.kr/hdd/hdd.html

디스크 구조

- **주소 지정 방식**
  - CHS (Cylinder, Head, Sector)
  - LBA (Logical Block Address)

- **HPA (Host Protected Area), DCO (Device Configuration Overlay)**



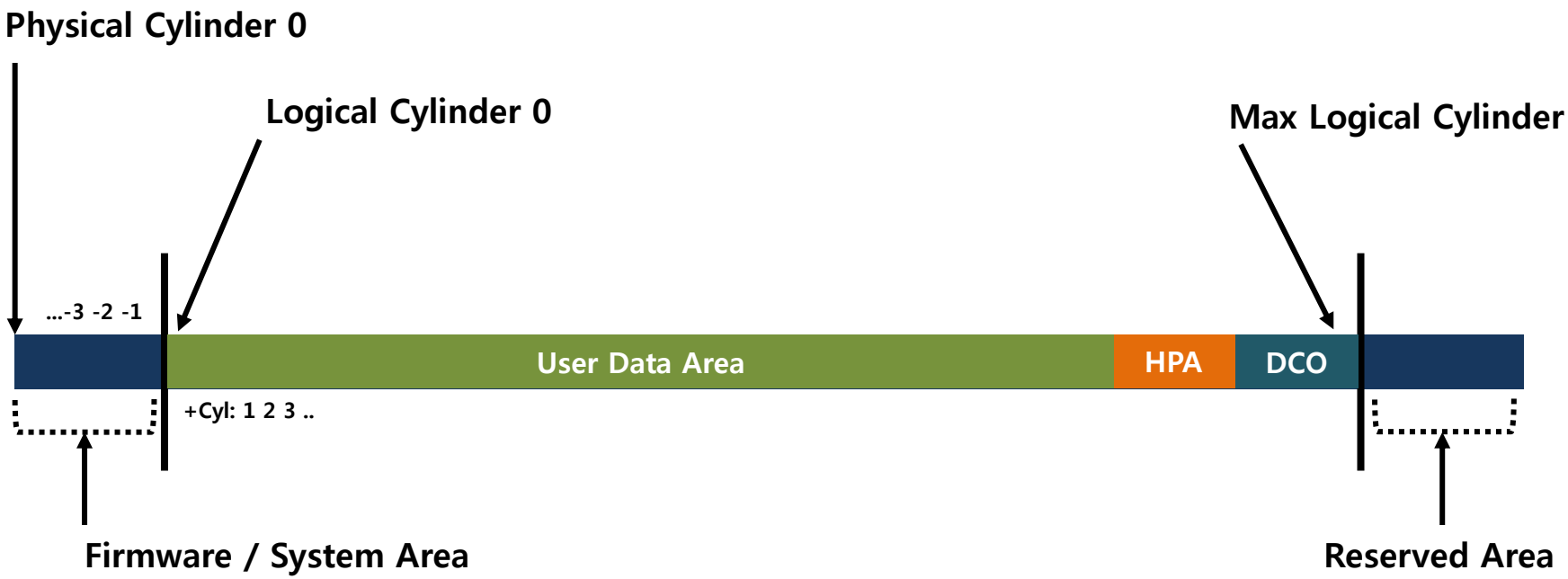
펌웨어

- **PCB 컨트롤러 보드의 펌웨어**
  - 플래터 펌웨어/시스템영역 로드

- **디스크 펌웨어**
  - 디스크 내부 동작 제어
  - 부팅 시 디스크 초기화
  - 자가 진단
  - 스핀들 스핀 업
  - 서보(Servo) 타이밍 조절

디스크 구조

- 데이터 저장 영역 구조



시스템 영역(System Area)

- S.M.A.R.T. Data
- System Logs
- Serial Number, Model Number
- P-List (Primary Defects List)
- G-List (Grown Defects List)
- U-List (Translator Data)
- Program Overlays – Firmware, Executable Code
- Zone Table
- Servo Parameters
- Test Routines
- Factory Defaults Tables
- Security Data Password for drive

시스템 영역(System Area)

- **S.M.A.R.T. (Self-Monitoring Analysis and Reporting Technology)**
  - 하드디스크 진단 기술로 이상 여부를 알려줌

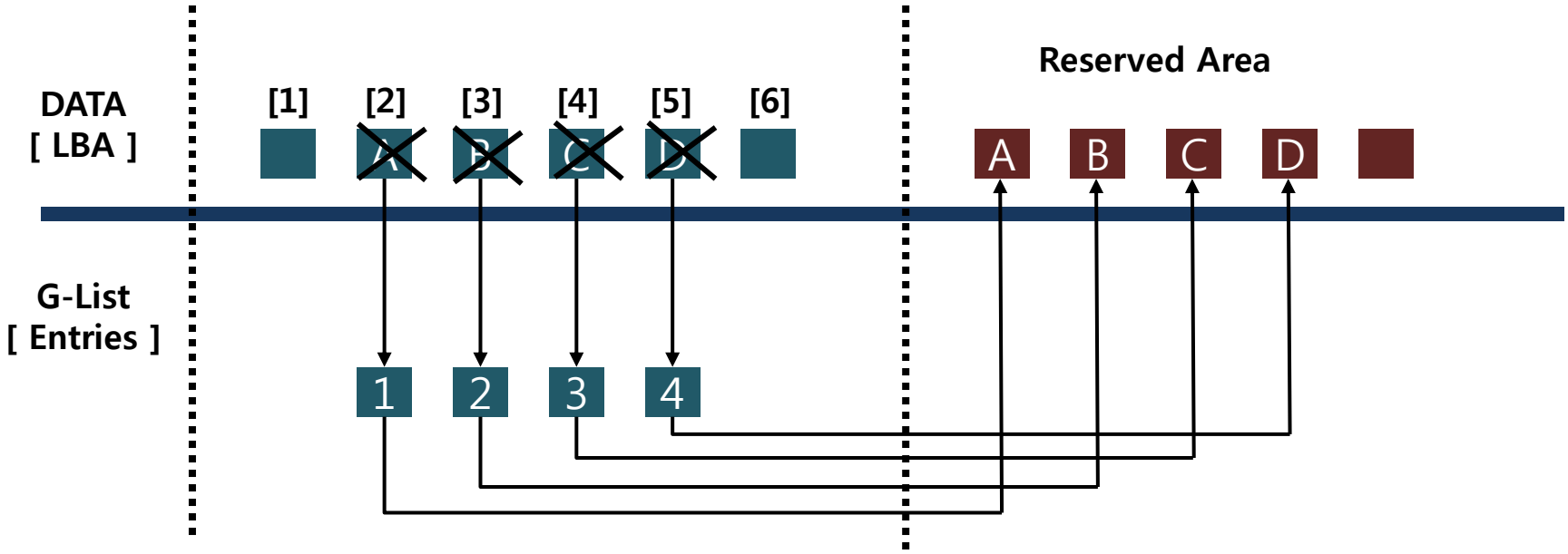
- **P-List (Primary Defects List)**
  - 디스크 제조시에 발생한 결함 섹터 목록

- **G-List (Grown Defects List)**
  - 디스크를 사용하면서 발생한 결함 섹터 목록

- **P-List와 G-List의 섹터는 자동적으로 우회**

시스템 영역(System Area)

- P-List와 G-List의 재매핑



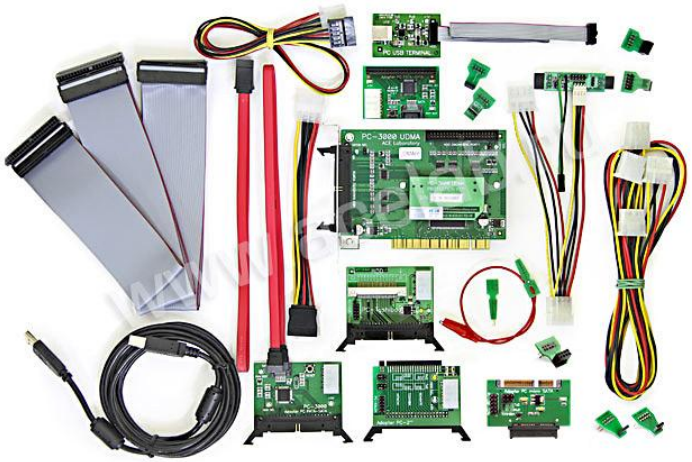
Storage Firmware Manipulation

펌웨어/시스템 영역 조작 도구

- **무료 도구**
  - HDD Firmware Serial Number Source Code
  - 제한된 기능 지원

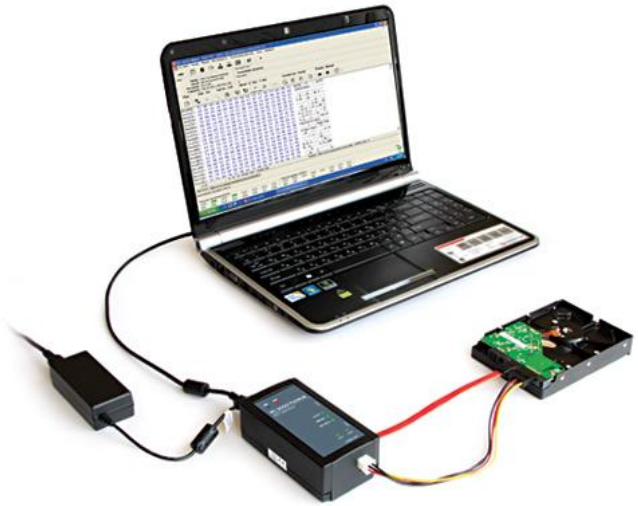
- **상용 도구**
  - ACE Laboratory – PC-3000
    - ✓ PC-3000 for Windows UDMA
    - ✓ PC-3000 for SCSI
    - ✓ PC-3000 Portable
    - ✓ PC-3000 Flash SSD Edition

PC-3000

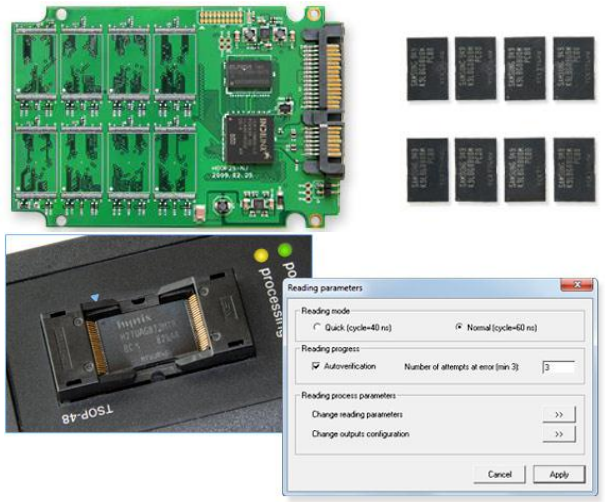


PC-3000 for Windows UDMA

PC-3000 for SCSI



PC-3000 for Portable



PC-3000 Flask SSD Edition

펌웨어 위협

- **특정 섹터 접근 차단**
  - P-List, G-List를 이용해 원하는 섹터 ➔ 배드 섹터 ➔ 데이터(악성코드, 중요데이터 등) 은닉
    - ✓ NTFS의 $BadClus를 이용해 속이는 것은?
- **포렌식 장비 동작 방해**
  - 펌웨어 조작으로 이미징 작업을 못하도록 방해
- **전용 도구(PC-3000 등)가 없이 일반 포렌식 도구에서 조작 여부 판단의 어려움**
- **그렇다면 현실적으로 이런 공격이 가능한가?**

결론

- **펌웨어 조작의 위협**
  - 펌웨어 조작으로 이상 동작
  - SA 변경은 하드웨어 장비 없이는 거의 탐지 불가능
- **실제 위협 가능성**
  - 대상 디스크의 제조사와 모델을 미리 알아야 함
  - 하드웨어 장비로 공격이 가능한 환경
- **연구**
  - 디스크 펌웨어의 변경 여부를 탐지할 수 있는 방안은?
  - 에러 목록(P-List, G-List)의 조작을 탐지할 수 있는 방안은?
  - 디스크 펌웨어, SA 정보를 확인할 수 있는 도구는?
  - PC-3000 도구 이외의 도구는? 더 간편한 방법은?

질문 및 답변