Hardware is the New Software

Joe Grand aka $\text{hlnphn}$, Grand Idea Studio, Inc.
/me

๏ Electrical engineer
๏ Hardware hacker
๏ Product designer
๏ Member of the L0pht hacker think-tank in 1990s
๏ Co-host of Prototype This on Discovery Channel
๏ Security work includes breaking smart parking meters, authentication tokens, and early PDAs
We Are Controlled By Technology

- Electronics are embedded into nearly everything we use on a daily basis
- Often taken for granted and inherently trusted
  - H/W is not voodoo, but people treat it that way
- Hardware has largely been ignored in the security field
  - Many products susceptible to compromise via simple, practical classes of attack
  - Vendors mostly respond to security problems by blowing them off (like S/W in the 90s!)
    * ...or it is blown completely out of proportion
The Time is Now...

- The tools are available
- The information is available
- All you need is the confidence to approach the problem...
Why Hardware Hacking? For Good?

- Security competency
  - Test hardware security schemes for failures/weaknesses

- Consumer protection
  - I don't trust glossy marketing materials...do you?

- Military intelligence
  - What is that hardware? How was it designed? By whom?

- Education and curiosity
  - To simply see how things work
  - Do something new, novel, and/or unique
Why Hardware Hacking? For Evil?

- **Theft of service**
  - Obtaining a service for free that normally costs $$$

- **Competition/cloning**
  - Specific theft of information/data/IP to gain a marketplace advantage

- **User authentication/spoofing**
  - Forging a user's identity to gain access to a system
Easy Access to Tools

- Cost of entry can be less than setting up a software development environment!
- Pre-made, entry-level packages available
Easy Access to Tools 2

Soldering Iron

- From a simple stick iron to a full-fledged rework station (~$10 to $5k)
- Fine tip, 700 degree F, > 50W soldering stick iron is recommended
- Ex.: Weller WP25 or W60P Controlled-Output, $67.95
Easy Access to Tools 3

Soldering accessories

- Solder: No-clean flux, thin gauge (0.032" or 0.025" diameter), ~60/40 Rosin core or Lead-free

- Desoldering Tool ("Solder Sucker"): Manual vacuum device that pulls up hot solder, useful for removing components from circuit boards (Radio Shack #64-2098, $7.99)

- Desoldering Braid: Wicks up hot solder (Radio Shack #64-2090, $3.99)

- IC Extraction Tool: Helps lift ICs from the board during removal/desoldering (Radio Shack #276-1581, $8.39)
Easy Access to Tools 4

Soldering accessories (continued)

- ChipQuik SMD Removal Kit: Allows the quick and easy removal of surface mount components
- Tip cleaner: Helps to keep the solder tip clean for even heat distribution. Ex.: Sponge, lead-free tip tinner
Easy Access to Tools 5

- **Multimeter**
  - Provide a number of precision measurement functions: AC/DC voltage, resistance, capacitance, current, and continuity
  - Ex.: Fluke Model 115, $129.00
Easy Access to Tools 6

Oscilloscope

- Provides a visual display of electrical signals and how they change over time
- Available in analog, digital, and mixed-mode versions
- Approximate price range $100 (used) - $20k US
- Ex.: USBee, $295-$1495, www.usbee.com
- Ex.: PicoScope, $250-$1500, www.pico-usa.com
Microscope

- Useful for careful inspection of circuit boards, reading small part numbers, etc.
- Human hands have more resolution than the naked eye can resolve
  - Greatly aids in soldering surface mount devices
  - You'll be amazed at what fine-pitch components you can solder when using a decent microscope!
- Approximate price range $100 - $5k US
- Ex.: Vision Engineering, www.visioneng.com
- Ex.: AmScope/Precision World, http://stores.ebay.com/Precision-World
Easy Access to Tools 8

PCB Design

- Many low-cost, open source, or captive solutions
- Ex.: EAGLE, www.cadsoftusa.com
- Ex.: gEDA, http://geda.seul.org
- Ex.: Kicad, www.lis.inpg.fr/realise_au_lis/kicad
Easy Access to Manufacturing

PCB Fabrication
- Can get professional prototype PCBs for ~$20 US each
- Many production houses available online
- Ex.: Advanced Circuits, www.4pcb.com
- Ex.: BatchPCB, www.batchpcb.com

PCB Assembly
- Have someone else build your complicated surface-mount boards
- Ex.: Advanced Assembly, www.aapcb.com
- Ex.: Screaming Circuits, www.screamingcircuits.com
Easy Access to Manufacturing 2

- Rapid Prototyping
  - Laser cutter
  - CNC
  - PCB prototype machine
    - Ex.: T-Tech, LPKF
  - 3D printing
    - Open-source solutions now exist
    - Ex.: MakerBot, www.makerbot.com
    - Ex.: RepRap, www.reprap.org
    - Ex.: Fab@home, www.fabathome.org
Easy Access to Information

- Open source hardware and DIY sites becoming commonplace
- People are publishing their new work daily
  - Pictures, videos, source code, schematics, Gerber plots
- G00gle & YouTube
- hack a day, www.hackaday.com
- Instructables, www.instructables.com
- Adafruit Industries, www.adafruit.com
Easy Access to Other People

๏ You don't have to live in a bubble anymore (if you don't want to)

๏ Can outsource tasks to people with specific/specialized skills

๏ Hackerspaces
  • Local venues for sharing equipment and resources
  • Much different than the hacker groups of the 80s and 90s that paved the way
  • Hundreds exist all over the world
  • Ex.: HackerspaceWiki, http://hackerspaces.org
  • Ex.: HacDC, www.hacdc.org
  • Ex.: Noisebridge (SF), www.noisebridge.net
Easy Access to Other People 2

๏ Workshops
  • Public, membership-based organizations (like a health club)
  • Classes and training available
  • Like hackerspaces, but more focused/directed to serve a specific purpose
  • Ex.: Techshop, www.techshop.ws
  • Ex.: The Crucible, www.thecrucible.org

๏ Various Forums & Cons
  • Black Hat, DEFCON, ToorCon, HOPE, ShmooCon, CCC, HAR, Hack in the Box, etc.
Hardware Hacking Methodology

There's never only *one* correct process

**Major subsystems:**
- Information gathering
- Hardware teardown
- External interface analysis
- Silicon die analysis
- Firmware reversing
Hardware Hacking Methodology 2

General guidelines:
1. Research the product
2. Obtain the product
3. Examine product for external attack areas
4. Open the product
5. Reverse engineer circuitry, silicon, and/or firmware
6. Identify potential attack areas
7. Perform attack
8. If not successful, repeat steps 6-7
Information Gathering

- **Crawling the Internet for specific information**
  - Product specifications, design documents, marketing materials
  - Check forums, blogs, Twitter, Facebook, etc.

- **Acquire target hardware**
  - Purchase, borrow, rent, steal, or ask the vendor
  - Ex.: eBay, surplus

- **Dumpster diving**

- **Social engineering**
Hardware Teardown

- Hardware and electronics disassembly and reverse engineering
- Get access to the circuitry
- Component and subsystem identification
- Gives clues about design techniques, potential attacks, and system functionality
- Typically there are similarities between older and newer designs
  - Even between competing products
External Interface Analysis

- Communications monitoring
- Protocol decoding and/or emulation
- Ex.: Smartcard, Serial, USB, JTAG, I2C, SPI, Ethernet, CAN

- Any interface accessible to the outside world may be an avenue for attack
  - Especially program/debug connections: If a legitimate designer has access to the interface, so do we

- Using oscilloscope, logic analyzer, dedicated sniffers, software tools, etc.
  - Ex.: Bus Pirate, http://buspirate.com
Silicon Die Analysis

- Supremely useful depending on attack goals
  - Simple imaging to gather clues
  - Key/algorithm extraction from ICs
  - Retrieve contents of Flash, ROM, FPGAs, other non-volatile devices
  - Cutting or repairing silicon structures (security fuses, traces, etc.)

- Like reversing circuitry, but at a microscopic level
"Real" equipment still fairly expensive, but can find in academic environment, get from surplus, or go low-tech:

- Fuming Nitric Acid (HNO3)
- Acetone
- Microscope
- Micropositioner w/ sewing needle
Silicon Die Analysis 3

☐ Required reading/viewing:


• Chris Tarnovsky/Flylogic Engineering's Analytical Blog, [www.flylogic.net/blog](http://www.flylogic.net/blog)


• "Hardware Reverse Engineering," Karsten Nohl, 25C3, [http://tinyurl.com/ya3s56r](http://tinyurl.com/ya3s56r)

Firmware Reversing

- Extract program code/data from on-board memory devices
  - Using off-the-shelf device programmer or product-specific tool
  - You'll end up with a binary or hex dump
  - Ex.: Flash, ROM, RAM, EEPROM, FPGA

- Quick run through w/ strings and hex editor to pick most interesting area to begin with

- Gives clues to possible entry/access points to administrative menus or ideas of further attacks
Firmware Reversing 2

- Disassembly and reverse engineering using IDA, etc.
- Modify, recompile, and reprogram device, if desired
- Now pure software hackers can get into the game
  - Using tools and techniques they are already familiar with
  - Electronic/embedded systems are typically nothing more than a general purpose computer programmed to perform a specific task
Common Themes

- Most product design engineers not familiar with security
- Many products based on publicly available reference designs provided by chip vendors
- Components easy to access, identify, and probe
- Engineers and manufacturers want easy access to product for testing and debugging
- Even the simplest attacks can have huge repercussions
Lots of High Profile Attacks

- **e-Voting Machines**
  - Massive security problems with devices around the world
  - Casting multiple votes, tampering with election configurations and data, easily changing firmware, remote detection of voting via TEMPEST monitoring
  - Ex.: [www.eff.org/issues/e-voting/](http://www.eff.org/issues/e-voting/)
  - Ex.: [www.avirubin.com/vote/](http://www.avirubin.com/vote/)
  - Ex.: [http://wijvertrouwenstemcomputersniet.nl/English/](http://wijvertrouwenstemcomputersniet.nl/English/)

- **ATM "cash dispensing" bug (pulled from Black Hat US 2009)**
Lots of High Profile Attacks 2

Smart power meters

- Wireless and peer-to-peer capabilities, no authentication for in-the-field firmware updates, can sever customer from power grid


Source unknown
Boston MBTA Fare Collection

- Stored value and/or time-based pass (unlimited rides during a given time period)
- CharlieTicket: Magnetic stripe, can be rewritten for value up to $655.36 by changing 16-bits corresponding to value
- CharlieCard: RFID-based smartcard using MIFARE Classic
  - Weak encryption leading to key recovery and full access to card
  - MIFARE Classic proprietary Crypto-1 algorithm previously broken by Karsten Nohl, et. al. 2007-2008
- MBTA launched assault on researchers to try and squelch release of information (only temporarily successful)
- Ex.: http://tech.mit.edu/V128/N30/subway.html
- Ex.: www.eff.org/cases/mbta-v-anderson
Smart Parking Meters

- Parking industry generates $28 billion annually worldwide
- Where there's money, there's risk for fraud and abuse
- Attacks/breaches can have serious fiscal, legal, and social implications
- Collaboration w/ Jake Appelbaum and Chris Tarnovsky to analyze San Francisco implementation
- Full details at www.grandideastudio.com/portfolio/smart-parking-meters/
Parking Meter Technology

- Pure mechanical replaced with hybrid electromechanical in early 1990s
  - Mechanical coin slot
  - Minimal electronics used for timekeeping and administrator access (audit, debug, programming?)

- Now, we're seeing pure electronic "smart" systems
  - Microprocessor, memory, user interface
  - US is late to the game, other countries have been doing this for years
Parking Meter Technology 2

- **User Interfaces**
  - Coin
  - Smartcard
  - Credit card

- **Administrator Interfaces**
  - Coin
  - Smartcard
  - Infrared
  - Wireless (RF, GPRS)
  - Other (Serial via key, etc.)
Prior Problems and/or Failures

- New York City reset via infrared (universal remote control), 2001, http://tinyurl.com/mae3g8
- San Diego stored value card by H1kari, 2004, www.uninformed.org/?v=1&a=6&t=txt
- Chicago multi-space failures, June 2009
  - Firmware bug or intentional social disobedience?
  - http://tinyurl.com/nt7g19
- Lots of other smartcard hacking has been done in the past
  - Ex.: Dutch phone cards (Hack-Tic), FedEx/Kinko's, satellite TV (DirecTV/DISH)
San Francisco MTA

- Part of a $35 million pilot program to replace 23,000 mechanical meters with "smart" parking meters in 2003
- Infrastructure currently comprised of MacKay Guardian XLE meters
- Stored value smart card
  - $20 or $50 quantities
  - Can purchase online with credit card or in cash from selected locations
Easy to replay transaction with modified data to obtain unlimited parking

- Determined solely by looking at oscilloscope captures of smartcard transactions
- Succeeded in three days
Meter Disassembly: MacKay Guardian
Meter Disassembly:
MacKay Guardian 2
Meter Disassembly: MacKay Guardian 3
Meter Disassembly: MacKay Guardian 4
Meter Disassembly: MacKay Guardian 5
Information Gathering

- A chance encounter w/ Department of Parking & Transportation technician on the streets of SF
  - Ask smart, but technically awkward questions to elicit corrections

- Crawling the Internet for specific information
  - Product specifications, design documents, etc.
  - What is the core business competency?
  - Do they have technical troubles?
I am learning how to use CVS and as part of this process I set up a test repository to 'play' with.

D:\src\working\epurse\cvstest>cygcheck -s -v -r -h

Cygnus Win95/NT Configuration Diagnostics

Win9X Ver 4.10 build 67766446  A
Path: /cygdrive/c/NOVELL/CLIENT32
     /cygdrive/c/WINDOWS
     /cygdrive/c/WINDOWS/COMMAND
     /usr/bin
     /cygdrive/c/JJMACKAY/MET_TALK
     /cygdrive/c/JJMACKAY/UTILITY

GEMPLUS_LIB_PATH = `C:\WINDOWS\GEMPLUS'

Found: C:\cygwin\bin\gcc.exe
Found: C:\cygwin\bin\gdb.exe

xxx, Sr. Software Designer
Smartcard Die Analysis

- Purchased and decapsulated multiple cards to look for clues of manufacturer and functionality

- Decapsulation process for smartcards
  1. Remove plastic surrounding the die (usually w/ acetone)
  2. Throw die into small Pyrex of heated Fuming Nitric Acid (HNO3)
  3. Rinse in acetone
  4. Glue die into a ceramic DIP package (for probing)
  5. If part is for analysis, prevent scratching!
Visually identified that two different smartcard types exist
  • Gemplus GemClub-Memo (ASIC)
  • 8051 microcontroller *emulating* GemClub-Memo

Dependent on card serial number
  • Older cards are ASIC, newer cards are MCU

Microcontroller has potential for hidden/undocumented commands
  • One could retrieve the code from the card and reverse engineer (we didn't)
Smartcard Die Analysis 3
Smartcard Communications Monitoring

- Used "shim" between smartcard and meter
  - Unpopulated Season 2 Interface
- Monitored I/O transaction w/ digital oscilloscope
- Asynchronous serial data @ 9600, 8E1 captured and decoded
  - Correct baud rate determined by measuring bit width on scope
Smartcard
Communications Monitoring 2

Data to/from smartcard/meter

Bit width

ATR (Answer-to-Reset)
Smartcard Protocol Decoding

- Captured multiple transactions to gather clues on operation
  - Different valued cards
  - Different serial numbers

- Based on what values changed per transaction & per card, could narrow down what data meant what

- Decoded transaction functionality by hand, no computer needed!
Initialization

**Meter**
- Reset
- Read Address 0
- Read Address 1
- Read Address 2
- Read Address 3
- Read Address 4

**Card**
- [4 byte responses unless noted]
  - ATR
  - Manufacturer ID
  - Serial #
  - Constant
  - Unknown (8)
    [Used for meter to calculate CSC1 password?]
**Initialization 2**

**Meter**
- Read CSC1
  - Ratification Counter
- CSC1 Password
  - [Password calculated by meter and sent to card for authentication]
- Read Address 14
- Read CTC1
  - Card Transaction Counter

**Card**
- [4 byte responses unless noted]
  - 0
  - Password OK (2)
  - 0
  - CTC1 [value varies]
Initialization 3

**Meter**
- Read Balance 2
- Read CTC1
  - Card Transaction Counter

**Card**
- [4 byte responses unless noted]
  - Maximum Card Value
    - Ex.: 0xFF FF F0 AF = $20
    - Ex.: 0xFF FF F1 27 = $50
- CTC1 [value varies]
By updating the Balance 1 Value (8 bytes), CTC1 automatically increments.

CTC1 is the only value that changes during the entire transaction!
Computation of Card Value

- Maximum card value = (Balance 2 - 95d)
  - Ex.: $0AF (175d) - 95d = 80 units
    - $80 * 0.25 = $20
  - Ex.: $127 (295d) - 95d = 200 units
    - $200 * 0.25 = $50
First attempt to replay exact transaction captured w/ scope
- Microchip PIC16F648A
- Written in C using MPLAB + CCS PIC-C
- Challenge for code to be fast enough and incorporate required short delays while still be readable/useful C
```
#include "card.h"

void main (void)
{
    port_b_pullups(FALSE); // disable port B pull-ups

    atr();
    manufacturer();
    issuer();
    current_value();
    while(1)
    {
        issuer();
        deposit_coin();
    }
}

void atr(void)
{
    delay_ms(1);
    putc(0x38);delay_us(170); // guard time
    putc(0x02);delay_us(170);
    putc(0x53);delay_us(170);
    putc(0x01);
}

void manufacturer(void)
{
    output_fccid(510);
    while (getc() != 0x00);
    while (getc() != 0xBE);
    while (getc() != 0x88);
    while (getc() != 0x90);
    while (getc() != 0x04);
    delay_us(380);
    putc(0xBE);delay_us(170); // guard time
    putc(0x7A);delay_us(170);
    putc(0x11);delay_us(170);
    putc(0x11);delay_us(170);
    putc(0x11);delay_us(170);
    putc(0x00);delay_us(170);
    putc(0x90);
}
```
Protocol Emulation 3

Then, modified code to change various values until success

• Knowing how "remaining value" is computed, what happens if we change Balance 2 to $FFF?
  • Ex.: $FFF (4095d) - 95d = 4000 units?
• Meter believes card has the maximum possible value
• Could also have the code never increment CTC1 so stored value never decreases
Ported code to Silver Card (PIC16F877-based smart card)

- PIC-based smartcards have been popular for satellite TV hackers for years, so required equipment is readily available
  - Ex.: http://interesting-devices.com
Hardware Evolution

1) Custom PCB + shim

2) MM2 card w/ external PIC

3) Silver Card PIC16F877 smartcard
San Francisco MTA Results

![Image of a parking meter with a digital display showing $999.99]
Final Thoughts

- Hardware is now more accessible to hackers than ever before
- The line is now blurred between HW & SW
- Simplest attacks known for decades still work
- New skills and techniques continually being developed and shared
- The time is right to get involved
- The media likes it, too!
Q & A

[ joe@grandideastudio.com ]