Floppy Forensics

Why you need a Ph.D. from USC 20 years later

Fred Cohen
Fred Cohen & Associates
The University of New Haven

Outline

- Background
  - Of your speaker
  - Of the matter at hand
- Arrives a floppy disk
  - What's a floppy worth?
  - How did it get to me?
- First get the data
  - Start with 100 floppy drives
  - Add some old systems
  - Toss in your own Linux CD
  - Out comes the data
- Then explain it...
  - How does a floppy work?
  - OK – how does it break?
- Could I be wrong?
  - What are the chances?
  - What are they really?
- Summary & Conclusions
  - You never know when your Ph.D. in EE will come in handy.
  - Questions & Comments

My (forensics) background

- B.S. EE 1977, C-MU
- M.S. Information Science, 1981, Pitt
- Ph.D. EE 1986, USC
- 150+ professional papers
- Several technical books
- “Challenges to Digital Forensic Evidence”
- California POST certified instructor
- Developed “ForensiX”
- Developed “ForWord”
- Invited speaker at digital forensics conferences
- Developed and teach graduate courses in digital forensics (UNH)
- Guest instructor at FLETC and SEARCH
- Ran a 35+ person research group doing forensics at Sandia

"High Fees – No Guarantees"

- Built forensic tools used for LE
- Built parallel digital forensic analysis platforms
- Consulting in digital forensics for many major corporations
- Developed "White Glove" Linux for digital forensics
- Testified in federal court about digital forensics matters
- Testified to congress about related matters
- Have assisted LE in many digital forensics cases
- And that's not my day job...
- And my sales pitch...

On to the matter at hand

- 2005-06-10
  - Agreed to work on the matter at hand
- 2005-06-21
  - Received one 3.25” floppy disk in FedEx
  - 3M Write-locked double-sided 135 TPI light gray with identifying marks
  - Apparently nobody else could read it

- For one reason or another, this was the one and only piece of original evidence in a case, and the disk was unreadable
- The last time the disk was used (other than to try to extract its contents for this case) was apparently some time in 1989

Outline

- Background
  - Of your speaker
  - Of the matter at hand
- Arrives a floppy disk
  - What's a floppy worth?
  - How did it get to me?
- First get the data
  - Start with 100 floppy drives
  - Add some old systems
  - Toss in your own Linux CD
  - Out comes the data
- Then explain it...
  - How does a floppy work?
  - OK – how does it break?
- Could I be wrong?
  - What are the chances?
  - What are they really?
- Summary & Conclusions
  - You never know when your Ph.D. in EE will come in handy.
  - Questions & Comments

The disk

- How much is a floppy disk worth?
  - $16M in case 1
  - More cases going
  - This is the one and only disk with the key evidence
  - The key evidence is in a corrupted part of the floppy
  - It came to me because they were desperate
  - High fees, no ...

How did I get it?

- The grape vine...
  - In the field of digital forensics there are essentially no Ph.D. EEs with my kind of background and who don't work for companies that prohibit the work
  - The lawyer asked his best expert who asked around and found me (they were in Florida, I was in CA)
  - They had already tried the “experts” at the commercial companies but the data did not come
  - The lawyer had a call with me, got my background and decided he wanted an expert who could stand up to the serious challenges they might face from major corporations who stood a lot to lose.
  - “We won't give up till you run out of money.”

Floppy recovery approaches

- Physically assemble, read disk (win?) OR
- Read relevant sectors from disk (win?) OR
- Alter drive alignment and read (win?) OR
- Read files repeatedly till success (win?) OR
- NO STANDARD APPROACH IN HERE
- Modify drive to change thresholds (win?) OR
- Read analog (storage scope), interpret (win?) OR
- Use real fine colored ferrite, epoxy, microscope
- Use electron microscope with EM attachment
- Expensive OR Destructive

Outline

- Background
  - Of your speaker
  - Of the matter at hand
- Arrives a floppy disk
  - What's a floppy worth?
  - How did it get to me?
- First get the data
  - Start with 100 floppy drives
  - Add some old systems
  - Toss in your own Linux CD
  - Out comes the data
- Then explain it...
  - How does a floppy work?
  - OK – how does it break?
- Could I be wrong?
  - What are the chances?
  - What are they really?
- Summary & Conclusions
  - You never know when your Ph.D. in EE will come in handy.
  - Questions & Comments

100 Floppies (300 Trojans?)

- Sometimes, a floppy is written “off-track”
  - You can read it if you realign the floppy drive head to also be off track by the same amount
  - And if you have the right floppy drive with the right head, etc.
  - Where do you get the right floppy drive?
- I go to Alaska
  - Charles Preston
- Charles goes to a swap meet in Anchorage
  - 100 floppy drives for $100 (no guarantees)
  - We start taking them apart to find ones we can realign
  - We use alignment tools (circa 1990)
    - The spiral disk is the best one for this)
    - Careful with that screw driver...

Some vintage computers

- To run the old floppies and the old software tools
  - You need vintage computers
  - Pack rats to the rescue
  - Computer museums have real value
  - Charles has his own... and so do I...
- I made $20,000 last year because I had old stuff
- When do you throw things out?
  - The day before someone calls with a lot of money if you have them
  - Which is why we cache our garbage
- But vintage computers don't run the new software tools that help you get at the data!!!

Toss in a White Glove

- So I have my own tool...
  - White Glove Linux has the tools we need and software to let me program it
  - And it runs on lots of old systems with little memory and no disk
  - And it has some of my special tools built up over the years
  - They just have to have a CD so we can boot them up into our Linux
- For forensics it must be...
  - It's forensically sound (changes nothing unless I make it do so)
  - And I can testify to its construction, calibration, and operation as a tool for presentation in court
  - And we can make it do all sorts of special and weird things because that's what we built it for
  - Garrett Gee, the CCDs

The data on silver platter (CD)

- So after a while
  - We found the best floppy drive to read it with
  - We turned off hardware error handling
  - We read the bits that were available
  - But that didn't quite do it
- There is only one disk and it's original evidence
  - Handle it as little as possible!
  - File rereads retry a lot!
- So I tried an old trick
  - Read it a block at a time with unlimited retries (low-level HW reads)
  - Each block goes into its own file, but the ones that fail get reread till they come back with a valid data length and checksum
  - Reassemble the blocks to recreate the floppy (even if you have to give up on some blocks)

Why does this work?

- If a file has N blocks AND
- P(bad block) = 25%
- THEN P(read works) = 0.75^N
- Reading a 20 block (~10K) file
  - For 20 blocks = 0.75^{20} ~= .00317
  - So it will take about 315 20-block read attempts
  - Or an average of about 3150 1-block reads
- Reading one block at a time
  - Average number of reads = 1.33...
  - For 20 blocks = 1.33*20 or 27 1-block read attempts
- Forensic disk image 720K = 1440 blocks...

Outline

- Background
  - Of your speaker
  - Of the matter at hand
- Arrives a floppy disk
  - What's a floppy worth?
  - How did it get to me?
- First get the data
  - Start with 100 floppy drives
  - Add some old systems
  - Toss in your own Linux CD
  - Out comes the data
- Then explain it...
  - How does a floppy work?
  - OK – how does it break?
- Could I be wrong?
  - What are the chances?
  - What are they really?
- Summary & Conclusions
  - You never know when your Ph.D. in EE will come in handy.
  - Questions & Comments

Explaining it – the hard part

- At the end of the day, I have to be able to demonstrate that this yields
  - Real data – not just fantasy data I created
  - That can be relied upon for legal purposes
  - The reflects that original unaltered contents
  - Even in the presence of hardware errors
- Legal processes are adversarial processes
  - They have experts on the other side
  - They stand to make or lose millions of dollars
  - They will try to invalidate it if they can

How does a floppy work?

- We all know – you stick it in and it reads and writes – but what are the error modes and how does it fail?
- A floppy disk is a ferro-magnetic coated piece of plastic or flexible metal that rotates under a read-write head (electromagnet)
- In “read” the magnetized ferro-magnetic material passes by the wire loop thus inducing current in the wire as the flux density changes
- The induced current (if over threshold) is interpreted by the drive as a “Transition” (T) in Modified Frequency Modulation (MFM) mode

How floppies work

- "Recording Codes for Digital Magnetic Storage"
  - Paul H. Suegel – IEEE Trans Magnetics Mag-21#5
  - September, 1985
- MFM encoding:
  - Designed to make the FSM simple

State Data	A	B	Code	Data
0	10/A	00/A	N0	0
1	01/B	01/B	N1	1

Fig. 8. Encoder and decoder tables for MFM

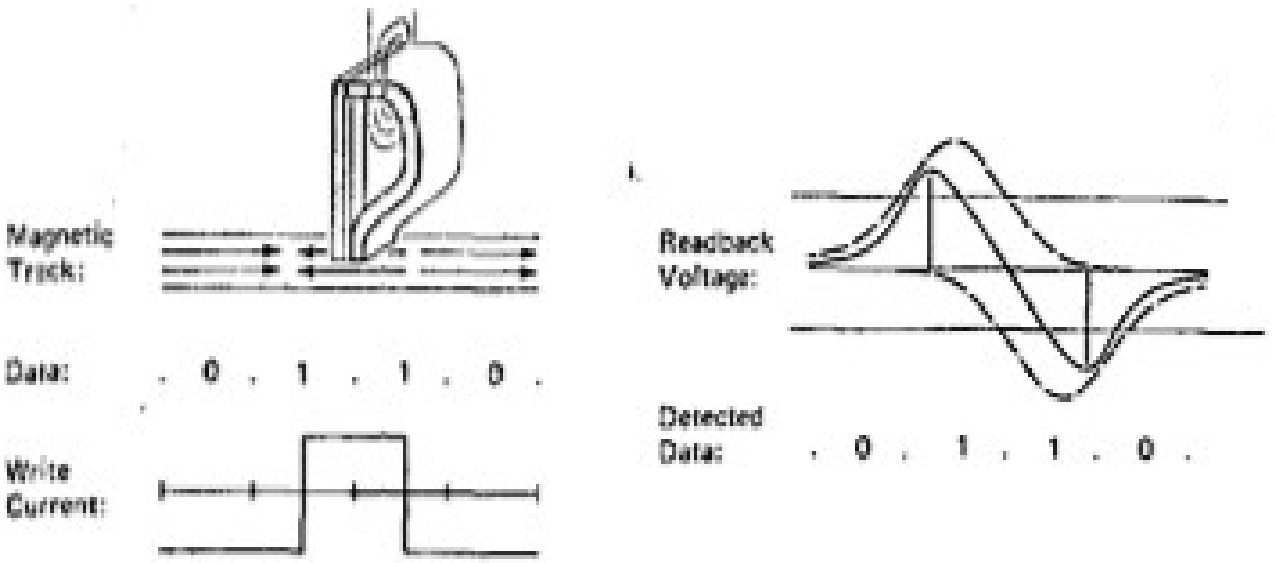
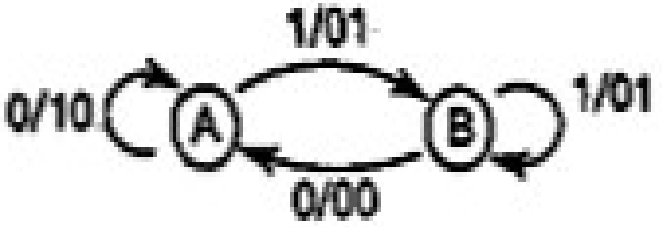
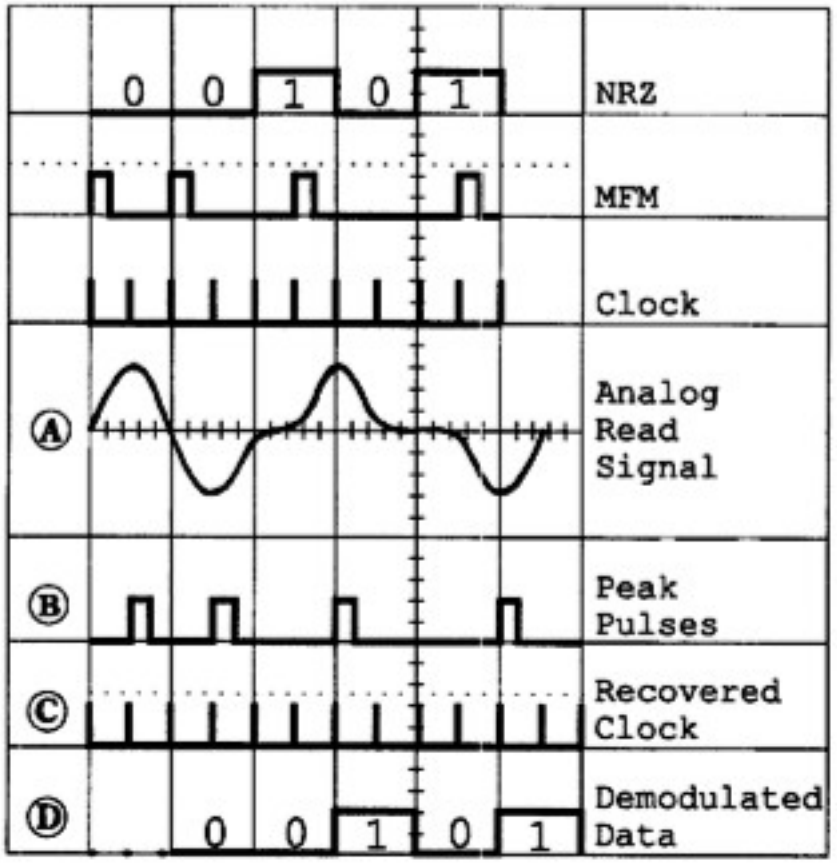


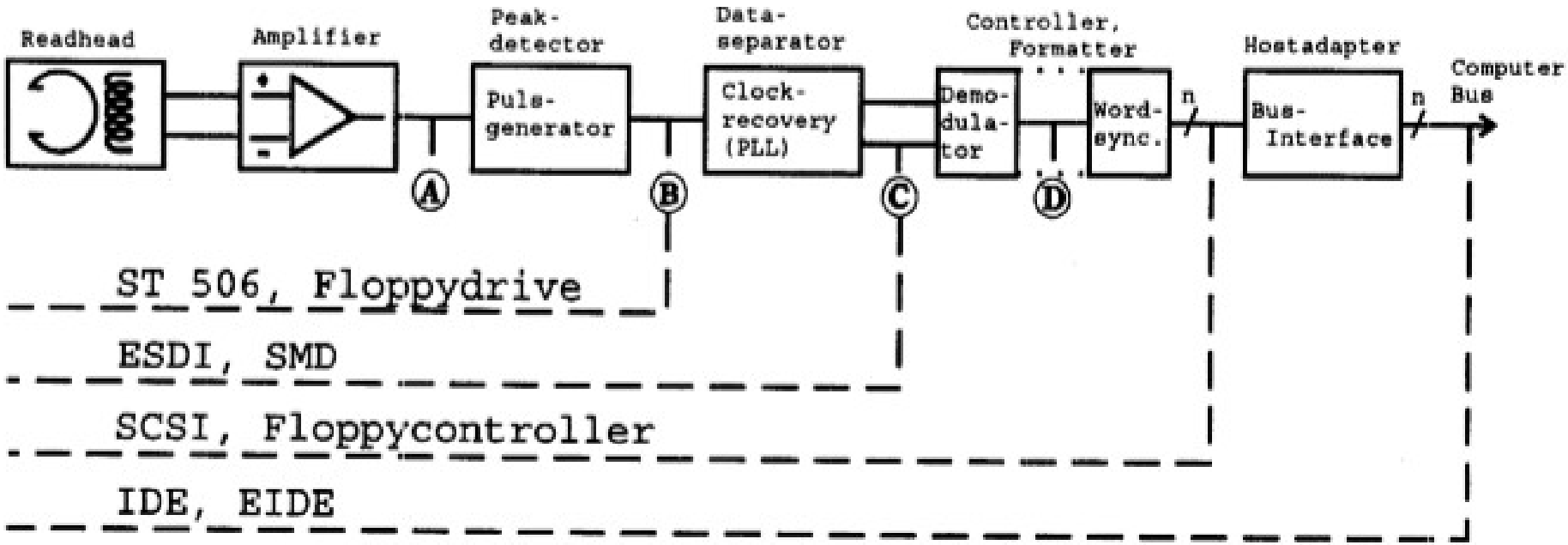
Fig. 1. Digital magnetic recording channel



Floppy electronics

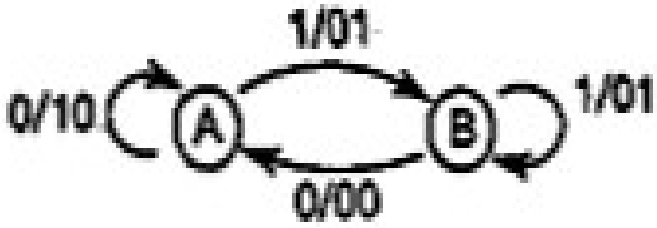
- "Computer Evidence
  - Analysis and Recovery of Magnetic Storage Media"
  - Hans-Joachim Leimkuller
  - 0-7803-2627-X/95 IEEE 1995





MFM mode faults

- Current direction is ignored by the drive
  - Changes in flux density produce transitions (T)
  - Non-changes produce no transition (N)
  - Direction of transition is ignored
  - What about A:0/11, A:0/00, B:1/00 and B:1/11?
    - Oops... Nobody noticed this...
    - Typically produce read errors?



- Houston – we have a problem
  - When the flux density is reduced:
    - we don't always get Ts
    - we can get T-N (T becomes an N)
    - we cannot get N-T (N becomes a T)

Floppy physicality

- 720K floppy
  - 3.5” outer diameter – 1” inner diameter
  - $3.14*(1.75^2) = 9.6$ sq-in – $3.14*(0.5^2) = 0.78$
  - OR about 9 sq in / 720Kbytes = 7.2M bits (CRC, etc.)
  - $25.4mmx25.4mm*1.25*10^{-6} = 8*10^{-4}$ sq mm/bit
  - Bits are not packed at maximum density - they are in tracks that take about ½ or less of the actual space
  - Transitions are across that space and are at somewhat smaller sizes within those areas
  - Density is different for inner tracks and outer tracks

Floppy drive physicality

- The drives have different properties
  - Different heads, motors, tolerances, thresholds
  - Different electronics (no circuit diagrams available)
  - Different calibrations and tolerances and errors
- This leads to failure modes at the edges
  - Of tracks (two drives offset can be incompatible)
  - Of flux densities (weaker field strength drive writes)
  - Of speeds (faster is not always better)
- And is used by defenders and attackers
  - Weak bit writes for copy protection
  - Weird formats and offset heads for criminals

MFM floppy failure modes

- You can get:
  - False transitions
    - From excessive charge on media induced by what?
    - External EMF – but it tends to be over large areas
      - field strength at distances tends not to be from near point sources
      - Inside the cover, the distance is set at several mm or more
      - How do you get a point change in at the size of 10^{-4} sq mm?
    - Not caused by media failures because it has to slew from max to min or min to max – zero to either does not transition
  - Missed transitions
    - Scratches, heat, bending, chemicals, biologicals, wear
  - PLL timing loss
    - Enough lost transitions

How can T-N impact 1,0?

- "Weak" bits:
  - Reduced field strength causes loss of transitions
  - Typically caused by wear, time, temperature changes
  - Lost transitions lead to PLL failure and bad timing
    - Produces hardware level failure signals and incomplete blocks read into memory (partial blocks)
  - Lost transitions lead to what else? (Nobody knew)
    - NT = 1 T-N > NN: 1-NN = 0, 0-NN = HW Error
    - 0-TN = 0 T-N > NN: 0-NN = HW Error
    - 1-NN = 0 No T, No T-N (NTNN > NNNN)
  - Weak bits lead to HW Error – OR – 1-0
  - They can never produce 1s!

The full table

- All of the data outputs that can result from T loss
  - Single T losses in table 111-101 and 11 into 10
    - I will write this as 11-[10/01] for now
    - Red circles impossible 0-1 for previous bits AND no change!
  - All multiple T losses lead to NNNN: invalid

Data	*Originally*	*Can turn into*	*Result*
[11]	NTNT	NNNT	1[01]
	NTNT	NTNN	[10]
	NTNT	NNNN	invalid
0[00]	TNTN	NNTN	1[00]
	TNTN	TNNN	invalid
	TNTN	NNNN	invalid
1[00]	NNTN	NNNN	invalid
[10]	NTNN	NNNN	invalid
0[01]	TNNT	NNNT	1[01]
	TNNT	TNNN	invalid
1[01]	NNNT	NNNN	invalid

Did I mention CRC codes?

- Floppies have 16 bit CRC codes for each 512 byte block
  - CRC codes do not cover all errors
  - 100% detection of single bit errors
  - 100% detection of all adjacent double bit errors
  - 100% detection of error windows spanning up to 16 bits
  - 100% detection of 2-bit errors not separated by 2^{16}-1
  - For arbitrary multiple errors spanning more than 16 bits
    - At worst 1 in 2^{16} failures (99.995% detection rate)
- Which means it takes more than one bit flip to cause a valid read with wrong data
- And many if adjacent bits flip

Outline

- Background
  - Of your speaker
  - Of the matter at hand
- Arrives a floppy disk
  - What's a floppy worth?
  - How did it get to me?
- First get the data
  - Start with 100 floppy drives
  - Add some old systems
  - Toss in your own Linux CD
  - Out comes the data
- Then explain it...
  - How does a floppy work?
  - OK – how does it break?
- Could I be wrong?
  - What are the chances?
  - What are they really?
- Summary & Conclusions
  - You never know when your Ph.D. in EE will come in handy.
  - Questions & Comments

So what does this mean?

- If weak fields are the cause of the failures
  - They can only produce T-N and 11-[10/01] or HW faults
  - If no HW faults, no 1s added to the final result
  - P(T-N -> valid code) = 2/22 = 1/11
  - P(incorrect read with valid CRC) ~= 1 in 2^{16} = 1/65536
  - IF MFM and CRC are not correlated, 1/720896 reads
- Unless we do tens of thousands of rereads the odds are very good that the data is valid – right?
  - But no 0-1 transitions are possible
    - How does a CRC work for only 11-[10/01] transitions?
  - P(1) != P(0) in typical data - ~ 75% of contents are 0s
    - 512 bytes * 8 bits /4 = 1024 1s per block – effects on CRC?

And what is the cause exactly?

- Remember – we are re-reading blocks that fail
  - Are some T-N errors at or below detection threshold?
    - If they are near the threshold
    - There is a probability they will show up on a given read
    - There is a probability the reads will work with multiple T-Ns
  - Are the errors caused by alignment differences?
    - The seeks sometimes go a bit off track and get correct data
    - Some failed reads may pick up different lengths before failures
  - Are the errors caused by timing faults?
    - If so, rereads will pick up different lengths before failures
    - This implies missing substantial numbers of Ts
- Which one is it?

Which one?

- Head alignment problems
  - Should be detected by realignment performance changes
  - The one and only piece of original evidence wears further every time you use it
  - So we were unwilling to test it a lot of times
  - Head realignment did not yield substantial improvements on the whole disk
  - We didn't test sector after sector (destroying the disk)
  - Other evidence: success or failure rates right after seeks
- Timing problems cause HW failures and partial reads
  - So do head alignment faults
  - We detected many partial reads (<512 bytes)

Which one?

- Typical rereads observed were:
  - 3 retries per seek (HW)
  - Groupings of retries from 1 to 80 seeks (=240 retries)
  - ~200 blocks require 40 retries average = 8000 retries
  - Often good read on first read after seek for low retries
  - Correlation between harder to read tracks and locations on disk drive (one sector after another with high counts)
- From this I conclude:
  - Some worn or failing areas (tracks) of the disk
  - Some weak flux areas near threshold (small reread #s)
  - No 1's are “created” by recovery (all T-N > 11-[10/01])
  - Some blocks may have 11-[10/01] residuals P(x) small

Did I mention the $s?

- There is a lot of money involved in this case
  - People will lie and cheat for a lot of money
  - How do we show that the “original” is real?
- Defeating the forgery claim
  - Assertions with no facts about the possibility that the floppy was a forgery
  - As a preemptive strike:
    - Make a preliminary determination
    - Identify how it could be made more solid
  - How can a floppy fail and how can this lead to dating of the data?

Floppy failure modes revisited

- Can the physical failure mode lead to dating?
  - Scratches – oxides form over time as F(environment)
    - Can tell me minimum time since the scratch (none detected)
  - Loss of field strength by random reorientation
    - Very likely not a very recent write (several years likely)
  - Bending – mechanical breakdown of location
    - Unless it leaves a scratch... (no bends or other residuals found)
  - Chemicals – effects instantaneous but residuals can last
    - Some decays with time (no evidence found)
  - Biologicals – growth over time leads to aging
    - While at USC summer internship – visited Ft. Wachuka
    - Biologicals grow/decay with time and environment (none found)
  - Wear – happens at the time of use (no evidence found)

More physical evidence

- Without destructive testing little more can be done
- If degradation effects desired content
  - Content was written before degradation
    - Otherwise errors in writing would have prevented writing it during read-back
  - Sets limit on latest time of write
    - Depending on results and specifics of the situation
    - Examination and analysis may lead to bounds on age of bits
  - Unlikely something the forger would do
    - It degrades the very part of the content they are trying to get into evidence
  - Degradation found was inconsistent with typical available intentional “weak bit” writing

More physical stuff

- Flux levels of drives
  - Technical changes in that time frame increased depth of electromagnetic penetration (we didn't do analog work)
- Tightness of write areas
  - Different penetration and patterns from different drives at different times (alignment indicated older technology)
- Speed of writing
  - Faster drives came later and you can tell by shapes of the electromagnetic patterns (no read failures on slow drives)
- Serial and batch numbers on disks
  - Manufacturing runs and other printed stamps (some OK)
- Material composition of disk (didn't look)

How is software built?

- Compiler and versions
  - Some didn't exist at that time (checked all versions)
- Copyrights and internal version numbers
  - Actual dates may show earliest possible dates
  - Version numbers internally may yield results
- How code is assembled
  - Methods change with time, new algorithms pop up...
- Libraries used and bug fixes
  - Patches and patch levels indicate dates (binary exam)
- Instruction sets provided in code
  - Different instructions are added over time for new CPUs

More on software

- File dates and layouts
  - File system as you see it
  - OLE files and internal object dates (ForWord)
  - Methods for generating dates and times (lots of details)
  - Other software from the period with its errors
- Residual data in deleted and overwritten areas
  - If the disk was used at a later time, even if overwritten, deleted data areas may reveal prior content (hex edit)
  - The prior content has the same potential for showing newer dates and indicators.
- We checked many of these things and found no obvious indications of forgery

An interesting project to do

- Faults are 11-[10/01] faults and languages are known
  - We should be able to systematically reconstruct valid values reflective of actual originals
    - Examine all 11-[10/01] changes in each reread block
    - Identify those that form valid parts of the code space
      - 'This' $=_8$ 124 150 151 163 (note initial '0' bit stops intraword effects)
      - 124_8 = 01 010 100_2 No valid weak bit errors
      - 150_8 = 01 101 000_2 -> 00 101 000_2 = 50_8 = '(' = T(is
      - 163_8 = 01 110 011_2 -> 01 010 011_2 = 123_8 = 'S' = ThiS
      - 163_8 = 01 110 011_2 -> 01 100 011_2 = 143_8 = 'c' = Thic
      - 163_8 = 01 110 011_2 -> 01 110 010_2 = 162_8 = 'r' = Thir
      - and 01 100 010_2 and 01 010 010_2 = 142_8 and 122_8 = 'b' and 'R' Thib/ThiR
    - Start with valid outputs, test faults, do linguistic analysis
- Some things like this appeared to exist on recovery

An alternative validation

- Why not just reread the floppy and compare?
  - In the actual case we were limited by time and did not want to destroy the original data by many rereads
- Did I mention White Glove and firewalls?
  - Luck favors the prepared
  - WG firewall floppy disks fail – so I have golden copies
  - I can do repetitions of recoveries on them to validate
  - Every few years (last week...) - low retries = golden
- Legal cases are hurry up and wait affairs
  - The total time available was about a week
  - 10 FTE days total for this effort

How to forge and win

- To get past this examination you would need:
  - Vintage hardware in good operating condition
  - Vintage software and tools and knowledge of how to use them
  - Vintage media that had not been used since then
  - Creation of all necessary content including deletions, old versions, and overwritten areas, including areas at the ends of blocks and deleted directory areas
  - The proper set of errors in the content under inspection
  - You would have to have multiples of these to create some of the different seek behaviors generating the error modes detected
  - After creation, you would have to have subjected it to degradation that typically occurs over many uses and several years
- We concluded this was most likely authentic and unaltered from the represented time frame

Outline

- Background
  - Of your speaker
  - Of the matter at hand
- Arrives a floppy disk
  - What's a floppy worth?
  - How did it get to me?
- First get the data
  - Start with 100 floppy drives
  - Add some old systems
  - Toss in your own Linux CD
  - Out comes the data
- Then explain it...
  - How does a floppy work?
  - OK – how does it break?
- Could I be wrong?
  - What are the chances?
  - What are they really?
- Summary & Conclusions
  - You never know when your Ph.D. in EE will come in handy.
  - Questions & Comments

Forensics is a team sport

- It takes a village (or something like that)
  - Nobody can know enough to do all of this stuff – serious digital forensics on tough cases needs a team approach
  - Nobody has all of the things you need
    - You need collaboration between laboratories
    - Find what you need through a networked approach
  - Kevin Manson and TUS and Cybercops and more
    - Online groups are creating communities of experts necessary to solve complex crimes and address complex issues
    - There are very few Ph.D. level people in the field
    - Only a few cases require them – for now...
    - This is changing as the courts realize it's not a slam dunk
  - Internal adversarial views (a good partner helps a lot)
    - Without Charles/Chet it fails

Forensics is a skills game

- You need to be able to read and understand a broad range of technical material
  - Electromagnetics, some physics, math skills
  - Electronics, electromechanics, chem/bio/misc facts
  - Coding, I/O systems, OS mechanisms, libraries, SW, data and representations, languages, complexity
- You need to generate new results on the fly
  - Nobody checked the error modes of MFM codings
  - Nobody else did systematic rereads this way
  - Nobody figured out if/how these rereads worked
  - Nobody (Still) has analyzed CRC under MFM faults
- We think it's normal – but we are EEs...

Forensics is experimental

- You need to do experiments on many things
  - But not on the original evidence!
  - Test on a dummy first
    - Every time we touched the evidence disk we did exactly the same thing to a test disk along with verifying non-faults first
    - Test error modes (verify write-protection with the identical HW and SW on the dummy first) to protect original evidence
  - You need repeatable results – but don't repeat them
    - The evidence wears out, so you can't just keep trying it
    - Repetitions on dummy versions first
- Original evidence has to be treated very carefully
  - Chain of custody, non-alterations, document use, don't risk unless/until you have to, need to get court permission for each activity

Why does it take Ph.D. EEs?

- Because there are very few people with
  - The proper range of skills and knowledge
  - Who can combine science, physics, and mathematics
  - To address complex previously unsolved computer-related problems
  - With on-demand experimental validation
  - And can credibly testify in court with serious opposition
- I have used parts of every course I took while at USC in digital forensic cases – it takes a Ph.D. EE
- But I don't get the day-to-day cases
  - High fees – no guarantees
  - I won't stop trying till you run out of money...

Thank You

Questions?
Discussion?!

Dr.Cohen at Mac.Com
http://all.net/