ECE 471 – Embedded Systems
Lecture 27

Vince Weaver

https://web.eece.maine.edu/~vweaver

vincent.weaver@maine.edu

13 November 2026

Announcements

- Midterm exam on Wednesday the 18th
  Will go over today what's on it
- Homework #6 Finally graded, will try to grade more of them before the next midterm
- If you need to borrow project parts let me know. Sending an e-mail in advance can help if it's a part I have to search for.
- Computer Security/Bugs in the news: ongoing Microsoft Downtime

Midterm #2 Preview

- Can have 1 page of notes like last time
- Primarily material since the previous midterm exam

Midterm #2 – Booting on the Pi

- What a bootloader does
- Why boot firmware and early filesystems need to be simple
- Why Pi is unusual booting (GPU-based)

Midterm #2 – Firmware

- Pro/Con of Signed Firmware

Midterm #2 – Real Time

- Definitions
- Is this hard, soft, firm

Midterm #2 – Busses

- i2c/SPI/1-wire
- Know the tradeoffs between i2c, SPI, 1-wire
- Be able to follow the C code for them

Midterm #2 – Computer Security/Bugs

- Security
  - Buffer overrun, why it is bad
  - C string issues
  - Be at least familiar at a high level with the buggy code examples given, but I won't ask you details of specific incidents

Some Case Studies

When Computers and Embedded Systems Go Wrong

- Sometimes it's security related
- Sometimes it is poorly written code
- Sometimes it is bad user interface (UI)
- Sometimes it's just bad luck
- Often it's a combination of all of the above

Cars / CANbus

- Bus connecting all parts of your car
- No direct support on Raspberry Pi
- Maybe will have more details on it if we have extra time end of semester

Some CANbus features

- Broadcast... any device can send packets to any other
- Priority.. devices set own priority, can monopolize bus
- No authentication... any device can control any other
- Firmware Updates
  - Challenge-response.
  - Cars are supposed to block attempts to re-flash or enter debug mode without auth.
  - But, mostly 16-bits, and required to allow a try every 10s, so can brute force in a week.

CANbus security paper

- 2010 IEEE Symposium on Security and Privacy. *Experimental Security Analysis of a Modern Automobile* U of Washington and UCSD.
- Fuzzing/ARM/CANbus
- tested at airport. cord to yank laptop out of ODB-II

Cars / Effects

- Not supposed to disable CAN or reflash firmware while car moving, but on the cars tested they could.
- Probing – packet sniffing, fuzzing (easier as packet sizes small)
- controlled radio – display, sounds, chimes
- Instrument panel – set arbitrary speed, rpm, fuel, odometer, etc
- Body control – could lock/unlock (jam by holding down lock), pop trunk, blow horn, wipers on, lights off
- Engine... mess with timing. forge "airbag deployed" to stop engine

Cars / CANbus – More effects

- can control brakes (on / off suddenly)
- managed to lock brakes so bad even reboot and battery removal not fix, had to fuzz to find antidote
- heating, cooling, lights, instrument panel
- windows/locks Why? fewer wires if on a bus then direct-wired

Cars / CANbus – What Can be Affected

- electronic stability control, antilock, need info from each wheel
- roll stability control (affect braking, turning to avoid rollover)
- cruise control
- pre-crash detection (tighten seatbelts, charge brakes)
- can over-ride starter switch. wired-or

Cars / CANbus – More

- fancy/more dangerous attacks
  - Have speedometer read too high
  - Disable lights at night
  - "self-destruct" w countdown on dash, horn beeping as got closer, then engine disable.

Cars / CANbus – Bus Separation

- while it might be nice to have separate busses for important and unimportant, in practice they are bridged

CANBus – do you need local access?

- Locks– monitor buttons, also remote keyfob... but also disengage if airbag deploys
- OnStar – remotely monitor car, even remotely stop it (in case of theft) over wireless modem
- OBD-II port
  - cars after 2008 required to have canbus on OBD-II
  - Can get wireless connectors for it so you can monitor on your phone, etc

Stuxnet

- SCADA – supervisory control and data acquisition
- industrial control system
- STUXNET.. targets windows machines, but only activates if Siemens SCADA software installed. four zero-day vulnerabilities
  USB flash drives
  signed with stolen certificates

Stuxnet

- Interesting as this was a professional job.
- Possibly by US/Israel targeting very specific range of centrifuges reportedly used by Iran nuclear program.
- While reporting "everything OK" the software then spun fast then slow enough to ruin equipment.

Examples – JTag/hard-disk

- JTAG/Hard-disk takeover
- `http://spritesmods.com/?art=hddhack&page=8`
- Find JTAG
- 3 cores on hard-disk board, all ARM. One unused.
- Install custom Linux on third core. Then have it do things like intercept reads and change data that is read.

Places for More Info

- Embedded projects: `http://hackaday.com`
  They had a recent series on CAN-bus
- Computer Risks and Security Issues: The RISKS digest from comp.risks
  `http://www.risks.org`

Software Bugs

- Not all bugs are security issues
- Coding bugs can have disastrous effects
- User interface bugs also can have serious consequences

Automotive

- Bugs, Toyota firmware
- http://www.edn.com/design/automotive/4423428/2/Toyota-s-killer-firmware--Bad-design-and-its-consequences
- NPR station sent image files over radio w/o file extension, bricked 2014-2017 Mazda infotainment computer in a way that was unfixable and needed total replacement

  https://arstechnica.com/cars/2022/02/radio-station-snafu-in-seattle-bricks-some-mazda-infotainment-systems/
- All Honda, Nissan, Infiniti, Acura, cars could be started/accessed via SiriusXM only knowing the VIN

  https://twitter.com/samwcyo/status/1597792097175674880

Automotive

- Case study: car privacy / mozilla article from September 2023
- News: Tesla had recall because firmware update caused power steering to fail when go over bump
- October 2025 – some jeeps got over-the-air firmware update and stopped working

https://arstechnica.com/cars/2025/10/software-update-bricks-some-jeep-4xe-hybrids-over-the-weekend/

Airplanes/Aviation

- September 2026. Military airplane files flightplan in England with some unexpected data on it. Crashes the computers of all the airports in england for a day, stranding lots of passengers
- AA Flight 965. Autopilot to waypoint R. Re-entered it, two starting with R, so it helpfully picked one with highest frequency, did a semi-circle turn to east right into a mountain.
- Air France Flight 447, reliance on autopilot was so far out of range autopilot gave up, confused them by reporting errors when it kicked in again

Airplane

- Boeing 737MAX issues with MCAS system
  Changed the engine placement, tried to avoid expense of re-training pilots by using software to make the new planes handle like the old ones. Did not go well.
- Qantas flight 72 – equipment giving bad readings, had 3 redundant ones but the way the code was written the bad one caused the plane to make sudden dives injuring many

Airplane

- Case study: air traffic control crash in england August 2023 due to not well tested input https://jameshaydon.github.io/nats-fail/
- XL airways 888T – when repainting squirted angle-of-attach sensors with fire hose. Two of 3 then froze in flight, the computer voted, but the two wrong ones matched so assumed the working one was broken. (note: the computer did send an error but due to the confusion from the cascading failures the plane still crashed)

Military

- Patriot missile – clock drift slightly, but when on for hundreds of hours enough to affect missile tracking
- Yorktown smart ship – 1997 – Running Windows NT. Someone entered 0 in a field, divide by 0 error, crashed the ship. Database crash, crashed propulsion system. Rumors that it needed to be towed in, but no, only down for 2.75 hours.
- F-22s computers crashed when crossing 180 degrees longitude? Lost navigation and communication, had to follow tankers back to Hawaii.
- Possible similar story of jets flying low over Dead Sea

Military

- US Navy ship crash, $100m damage, mostly due to bad UI design/bad checkbox

https://adrian3.com/blog/2019/2019-09-28-The-US-Navys-100-million-dollar-checkbox.php

Spacecraft – Mariner 1

- Mariner 1 (1962) – rocket off course due to mis-transcribed specification into FORTRAN, missing overbar

Spacecraft – Apollo 11

- Apollo 11 (1969) – landing on moon (often considered first embedded system/real-time OS)
- 36k ROM (rope), 2k RAM, 70lbs, 55W, 5600 3-input NOR
- Processor normally loaded with 85% load. DELTAH program run which take 10%. But buggy radar device was stealing 13% even though in standby mode.
- Multiple 1202 overload alarms
- Mini real-time OS with priority killed low-priority tasks so things still worked.

Spacecraft – Ariane 5

- Ariane 5 Flight 501 (1996) – famous. $370 million.
- Old code copied from Ariane 4. Horizontal acceleration
- Could not trigger on Ariane 4 (accel never that large)
- Could trigger on more powerful Ariane 5
- Conversion from 64-bit float to 16-bit signed int overflowed. Trap
- Primary guidance computer crashed
- Secondary computer, but ran same code, crashed
- Sent debug messages after crash, autopilot read those as velocity data
- Destructed 37s after launch
- Written in ADA

Spacecraft – Mars

- NASA Mars Polar Lander (1999)
  - likely mistook turbulence vibrations for landing and shut off engine 40m above surface
- NASA Mars Climate Orbiter
  - ground software using lbf (pound/foot) units, craft expecting Newtons

Spacecraft – Mars

- NASA Mars Spirit rover (2004)
  - temporarily disabled due to too many files on flash drive
  - Constantly rebooting
  - Radio could understand some commands directly, could reboot with flash disabled.
  - Fixed when deleted some unneeded files.
  - Eventually reformat.
  - Issue is 90 day design period, lasted years (until 2010)

Spacecraft – Mars

- Phobos-Grunt (2012)
  - Bit flip in memory caused it to crash before firing rockets to Mars
  - Entered safe mode waiting for command
  - Antennas not deployed until after rocket firing
  - Could not receive command to leave safe mode.
- ExoMars Schiaparelli Lander (2016)
  - Bad data to inertial measurement unit for 1 second
  - thought this meant it was below ground level, released parachute when still 3.7km up.
  - Had valid data from radar

Spacecraft – General

- Boeing Starliner OTF-1 flight issues, lack of proper testing (2019)
  Didn't do full stack testing (tested parts individually) so when in real life rocket was on 11 hours before capsule it got the time wrong and fired engines at wrong time
- Japan moon lander (2023) – altitude radar report 3km change as went over cliff, lander thought this was too high and started filtering out altitude data

Medical – Therac-25

- Therac-25 radiation treatment machine, 1985-1987
- 6 accidents, patients given 100x dose. Three died
  High power beam activated w/o spreader too.
  Older machines had hardware interlock, this one in software. Race condition. If 8-bit counter overflow just as entering manual over-ride, it would happen.

Medical – Therac-25

- Triggering the bug
- To trigger, had to press X (mistake), up (to correct), E (to set proper) then "Enter" all within 8 seconds. This was considered an improbable series of keypresses.
- This missed during testing as it took a while for operators to get used to using machines enough to type that fast.
- Used increment rather than move to set flag, this meant sometimes it wrapped from 255 to 0, disabling safety checks
- Written in Assembly Language

Therac-25 (what went wrong)

- Software not independently reviewed
- No reliability modeling or risk management
- Something wrong: Printed “MALFUNCTION” and error number 1 to 64 which was not documented in manual. Press P to clear.
- Operators not believe complaints from patients.
- The setup was not tested until after it was installed at hospital.
- cut-and-pasted software from earlier model that had hardware interlocks
- Concurrent (parallel) operation with race conditions

Another Medical Example

- Devices like pacemakers, how does a doctor reprogram them?
- Are they password protected?

Financial

- Knight Capital. Upgrade 7 of 8 machines, missed last. Re-used a flag definition with new software. Caused massive selloff, $440 million

Security

- Crowdstrike (2024). Bad update to a windows security software program took down a large number of systems, this included many embedded systems

Power

- 2003 Blackout
  - Power plant fail. Cause more current down transmission lines in Ohio. Heat, expand, touch tree, short out.
  - Race condition in Unix XA/21 management system, so alarms not go off
  - Eventually primary system fail as too many alarms queue up
  - Backup server also fail
  - During failure, screens take 59s (instead of 1s) to update
  - Blackout of most of NY and a lot of north east.

Time / Calendar Issues

- Run out of room to store date
  - y2k issue, two digit years in 1999 to 2000
  - 2038 issue on UNIX/Linux, 32-bit time since epoch (1970) overflow
  - GPS satellite rollover 10-bit week (1024 weeks)

Time / Calendar Issues

- Daylight Savings issues
- Leap second issues
- Leap year issues (code fails if year has 366 days in it)
  2024 Eutelsat OneWeb down because of this

Time / Jiffy Wrapping

- Have a timer in OS that overflows in short term, on old Windows after 48 days, crash. Linux similar problem, in end to force people to handle it at boot the overflow was made to happen in a few minutes (rather than months later) so bugs caught quickly.

Other

- 2026 Smart TV botnets
- 2026 October Food Freezer attacks