“The only truly secure system is one that is powered off, cast in a block of concrete and sealed in a lead-lined room with armed guards.”

– Gene Spafford
Security Vulnerabilities

- Anti-Patterns for vulnerabilities
  - Ignoring vulnerabilities until attacked
  - Assuming vulnerabilities won’t be exploited:
    - Unsecure embedded networks
    - Reverse engineering of devices
    - Hidden functionality
  - Assuming passwords will be secure

- Vulnerability: a point in the system susceptible to attack
  - Includes HW, SW, network, people, infrastructure, organization
  - Exploit: a method of converting a vulnerability to a security breach
  - Attack: someone uses an exploit to breach system security
Even Simple Devices Are Targets

DON'T BREW THAT CUPPA! Your kettle could be a SPAMBOT
https://goo.gl/zxGH4S

Russian report says Chinese appliances hide Wifi slurping spam-spreaders 29 Oct 2013 at 07:03, Simon Sharwood

Russian authorities have claimed that household appliances imported from China contain tiny computers that seek out open WiFi networks and then get to work sending spam and distributing malware.

St Petersburg news outlet Rosbalt reported last week that local authorities had examined kettles and irons and found “20 to 30 pieces of Chinese home appliance ‘spy’ microchips” that “sends some data to the foreign server”.

A bit of digging suggests it is legitimate. One source the story mentions, Gleb Pavlov of customs broker Panimport can be found at the link we've popped in on the company's name. We've also been able to find this link to an appliances company called “Sable Ltd”, the very name translation engines say is the employer of one Innokenty Fedorov whose company found the bugged appliances.
Weak or Master Passwords

- Weak passwords are bad
  - 1234, 777 (US), 888888 (China)
  - password, iloveyou, qwerty

- Factory master passwords are worse!
  - Once one user knows, everyone will know

- Don’t use the same key in all systems
  - Keeloq car remote broken due to using same manufacturer key in all units

- Use long enough crypto keys
  - Every year safe key size gets a little longer
  - E.g.: 256 bit symmetric key
  - 3072 bit public key
Avoid Default Passwords

Investigative cybercrime journalist Brian Krebs reported on Tuesday that his website, KrebsOnSecurity.com, was hit by a massive distributed denial-of-service (DDoS) attack that could be the largest in history.

According to Krebs, his site was targeted with various types of DDoS attacks, including SYN and HTTP floods. The attack peaked at 665 Gbps and 143 Mpps (million packets per second), but it was successfully mitigated by Akamai, the company that provides DDoS protection services for KrebsOnSecurity.

These 60 dumb passwords can hijack over 500,000 IoT devices into the Mirai botnet

Always change your device’s default password.
Mistakes Using Cryptography

- Attackers go after implementation mistakes
  - Usually you don’t have to break the cryptography

- Typical mistakes
  - Sending initial passwords or secrets without encrypting
  - Using known flawed protocols (e.g., flawed secret key exchange, flawed software)
  - Implementing your own crypto from books
  - Permitting weak passwords
  - Not applying security patches
“Proprietary protocol” does not provide much protection

- Automotive CAN with proprietary messaging

Figure 4. Screenshot of the CARSHARK interface. CARSHARK is being used to sniff the CAN bus. Values that have been recently updated are in yellow. The left panel lists all recognized nodes on high and low speed subnets of the CAN bus and has some action buttons. The demo panel on the right provides some proof-of-concept demos.

Physical Access to System

How easy is it for someone to steal your design?
- Hardware design
- Software design

Chip peels are no big deal
- Can recover hardware schematics from silicon
- Can recover software from memory
- “Tamper resistant” is a good way to slow down attacks – but does not stop them

Hidden Functionality

- Assume that any “secret” functionality will be revealed
  - Factory test modes
  - Factory service modes
  - “Easter eggs”

- This includes:
  - Service technician master password
  - Ability to reset system
  - Default administrative accounts

- Potential better approach:
  - “Factory test” jumper on internal board
  - Factory test mode warning on screen

https://goo.gl/Ty9aYJ
Counterfeit Systems

How do you know components are legitimate?

- Often chips/boards fail to meet specifications, but are superficially the same function
  - Rejects that failed non-functional testing
  - Salvaged used components
  - “Clone” hardware without safety mechanisms
- What if fake shows up in a critical application?
- US Customs seizes ~1-2 million fake ICs per year

What if someone wants to clone your whole product?

- “Tamper-proofing” may help, but not if attack is lucrative
- Clones might be built in part by scavaging authentic components
- Will need to have some way to authenticate and track serial numbers

Cloud Connected Devices

- Embedded meets Internet Security
  - Need good practices for IT security
  - Need good practices for embedded
  - IT penetration can cause safety issues via embedded device(!)

- Questions to ask in design
  - How does Cloud know it is a legitimate device?
    - Deploy each device with a unique public key signed by factory
  - How does user securely connect smart phone to device?
    - Print unique WPA (etc.) key on sticker inside unit
  - What if user forgets password?
    - Provide “factory reset” ability; NOT a shared master factory password
  - How will you do secure update? Factory Key revocation?

More Embedded-Specific Security Issues

- **Resources are scarce**
  - Consider a Smart Card chip (TPM) for keys & crypto

- **Embedded networks are generally insecure**
  - Short network messages, no built-in security
  - Power, memory, CPU constrain security resources

- **Power drain attacks**
  - Attacks designed to deplete batteries

- **Real time operation attacks**
  - Only a slight overload might cause real time schedule problems

- **Tamper resistance & evidence for critical properties**
  - How can you prove someone didn’t alter your safety critical system? (Even the owner?)

- **Ensuring updates are authentic & are installed**
  - How can you ensure only certified configurations will run?
  - How do you ensure installation of required updates with intermittent external connectivity?
Be realistic with vulnerabilities
- Users won’t change default passwords
- Weak passwords will be used
- Counterfeit systems will be built
- All network systems will be attacked

Pitfalls:
- Assuming users will practice excellent security hygiene
- Using a master password
- Assuming attackers can’t extract secrets from at least one device
  - Using a given symmetric key in more than one device instance
Row Hammer: (CMU, 2014)

The Meltdown and Spectre exploits use "speculative execution." What's that?

You know the trolley problem? Well, for a while now, CPUs have basically been sending trolleys down both paths, quantum-style, while awaiting your choice. Then the unneeded "phantom" trolley disappears.

The phantom trolley isn't supposed to touch anyone. But it turns out you can still use it to do stuff. And it can drive through walls.

That sounds bad. Honestly, I've been assuming we were doomed ever since I learned about Rowhammer.

What's that? If you toggle a row of memory cells on and off really fast, you can use electrical interference to flip nearby bits and—do we just suck at...computers?

Yup. Especially shared ones.

So you're saying the cloud is full of phantom trolleys armed with hammers. ...yes. That is exactly right.

Okay. I'll, uh...install updates? Good idea.

Rapid row activations (yellow rows) may change the values of bits stored in victim row (purple row).[12]:2

https://en.wikipedia.org/wiki/Row_hammer

https://xkcd.com/1938/