OSY.SSI[2019][6]

In the last episode...

Internet vs the Classical Theory

In the news...

- US Attorney General William Barr will present an open letter to Facebook and its CEO, Mark Zuckerberg, cosigned by British and Australian officials, asking the company not to implement end-to-end encryption protections [src]
- HTTPS-snooping malware 'patches' random number generator, decodes traffic and installs rogue certificates [src]
- CVE-2019-2215: Local privilege escalation vulnerability affecting Samsung, Xiaomi, Huawei, Google Pixel amongst others. Exploited in the wild. [src]
- CVE-2019-11932: WhatsApp RCE triggered by sending a GIF [src]
- Former Yahoo engineer pleads guilty to hacking thousands of accounts in hunt for nudes [src]
- ANU Vice Chancellor releases detailed report of cyberattack [src]
- Egyptian Hackers Blamed For Cyberattack On Android Devices—But It's Complicated [src]
- DHS and FDA warn about much broader impact of Urgent/11 vulnerabilities [src]
- A recent attack aimed at a U.S.-based oil, gas and chemical supplier leverages the company's use of the enterprise-class Asterisk open-source PBX software, used for VoIP services. [src]
- EA Games Leaks Personal Data of 1600 FIFA 20 Competitors [src]

Mise en bouche

Easy Print 42 times the letter B.

Mise en bouche

- Easy Print 42 times the letter `B`.
- Easy Create a function `addme(x, y)` that returns `x + y`

Mise en bouche

- Easy Print 42 times the letter `B`.
- Easy Create a function `addme(x, y)` that returns `x + y`
- Easy Test your function with `x = 42` and `y = 69`. Does it work as intended?

Mise en bouche

- Easy Print 42 times the letter `B`.
- Easy Create a function `addme(x, y)` that returns `x + y`
- Easy Test your function with `x = 42` and `y = 69`. Does it work as intended?
- Medium Choose `x > 0` and `y > 0` so that `x + y = x`.

Mise en bouche

- Easy Print 42 times the letter `B`.
- Easy Create a function `addme(x, y)` that returns `x + y`
- Easy Test your function with `x = 42` and `y = 69`. Does it work as intended?
- Medium Choose `x > 0` and `y > 0` so that `x + y = x`.
- Medium Choose `x` $\neq$ `0` and `y` $\neq$ `0` so that `x/y` $= \infty$.

Mise en bouche

- Easy Print 42 times the letter `B`.
- Easy Create a function `addme(x, y)` that returns `x + y`
- Easy Test your function with `x = 42` and `y = 69`. Does it work as intended?
- Medium Choose `x > 0` and `y > 0` so that `x + y = x`.
- Medium Choose `x` $\neq$ `0` and `y` $\neq$ `0` so that `x/y` = ∞.
- Easy Choose `x` so that `x` $\neq$ `x`.

Mise en bouche

Easy Print 42 times the letter `B`.

Easy Create a function `addme(x, y)` that returns `x + y`

Easy Test your function with `x = 42` and `y = 69`. Does it work as intended?

Medium Choose `x > 0` and `y > 0` so that `x + y = x`.

Medium Choose `x` $\neq$ `0` and `y` $\neq$ `0` so that `x/y` = ∞.

Easy Choose `x` so that `x` $\neq$ `x`.

Clearly, some basic mathematical properties of usual numbers are not satisfied.

Mise en bouche

Easy Print 42 times the letter `B`.

Easy Create a function `addme(x, y)` that returns `x + y`

Easy Test your function with `x = 42` and `y = 69`. Does it work as intended?

Medium Choose `x > 0` and `y > 0` so that `x + y = x`.

Medium Choose `x` $\neq$ `0` and `y` $\neq$ `0` so that `x/y` = ∞.

Easy Choose `x` so that `x` $\neq$ `x`.

Clearly, some basic mathematical properties of usual numbers are not satisfied.

(Also: $x + (y + z) \neq (x + y) + z$, $x(yz) \neq (xy)z$, $x(y + z) \neq xy + xz$, etc.)

Mise en bouche

Easy Print 42 times the letter `B`.

Easy Create a function `addme(x, y)` that returns `x + y`

Easy Test your function with `x = 42` and `y = 69`. Does it work as intended?

Medium Choose `x > 0` and `y > 0` so that `x + y = x`.

Medium Choose `x` $\neq$ `0` and `y` $\neq$ `0` so that `x/y` = ∞.

Easy Choose `x` so that `x` $\neq$ `x`.

Clearly, some basic mathematical properties of usual numbers are not satisfied.

(Also: $x + (y + z) \neq (x + y) + z$, $x(yz) \neq (xy)z$, $x(y + z) \neq xy + xz$, etc.)

Do programmers know this and its consequences when they design and implement programs?

Do you know what a rhetorical question is?

Floating point numbers can kill

Do you know what a rhetorical question is?

Floating point numbers can kill

- Patriot Missile incident (Dharan, Saudi Arabia, 1991): 1/10

Do you know what a rhetorical question is?

Floating point numbers can kill

- Patriot Missile incident (Dharan, Saudi Arabia, 1991): 1/10
  - 1 Scud missed by 0.5 km. 28 dead, 1000 injured.
- Ariane 5 maiden flight 501 (Kourou, French Guyana, 1996): 64-bit to 16-bit

Do you know what a rhetorical question is?

Floating point numbers can kill

- Patriot Missile incident (Dharan, Saudi Arabia, 1991): 1/10
  - 1 Scud missed by 0.5 km. 28 dead, 1000 injured.
- Ariane 5 maiden flight 501 (Kourou, French Guyana, 1996): 64-bit to 16-bit
  - 1 launcher exploding. USD 370 millions, satellites destroyed, market shift to Soyuz-U/Fregat.
- USS Yorktown (international waters, 1997): x/0.

Do you know what a rhetorical question is?

Floating point numbers can kill

- Patriot Missile incident (Dharan, Saudi Arabia, 1991): 1/10
  - 1 Scud missed by 0.5 km. 28 dead, 1000 injured.
- Ariane 5 maiden flight 501 (Kourou, French Guyana, 1996): 64-bit to 16-bit
  - 1 launcher exploding. USD 370 millions, satellites destroyed, market shift to Soyuz-U/Fregat.
- USS Yorktown (international waters, 1997): x/0.
  - Computer system down. Propulsion system down. In the middle of nowhere.
- And various major vulnerabilities based on number play
  - WebKit/Chrome (CVE-2009-2195)
  - Ruby (CVE-2013-4164)
  - Adobe Flash (CVE-2015-3077, CVE-2014-0502, etc.)
  - Mozilla Thunderbird (CVE-2017-5407): Pixel and history stealing via floating-point timing side-channel!

The overarching principle

It is obvious, but precisely for this reason it requires insisting

Mathematical model $\neq$ Program $\neq$ What the computer should do $\neq$ What it does

While it is relatively rare, it *does* happen that the CPU returns the *wrong* answer to an arithmetic question (as was famously the case for the first Intel Pentium with `FDIV` and `F00F`).

Many programmers forget that.

The overarching principle

It is obvious, but precisely for this reason it requires insisting

Mathematical model $\neq$ Program $\neq$ What the computer should do $\neq$ What it does

While it is relatively rare, it *does* happen that the CPU returns the *wrong* answer to an arithmetic question (as was famously the case for the first Intel Pentium with `FDIV` and `F00F`).

Many programmers forget that. Please, don't.

Example: Intel Pentium `F00F`, Reloaded

"Oh it's just an old bug. Who uses Pentiums anymore? Plus it's easily mitigated and it just forces a reboot. No big deal."

Example: Intel Pentium F00F, Reloaded

"Oh it's just an old bug. Who uses Pentiums anymore? Plus it's easily mitigated and it just forces a reboot. No big deal."

(BlackHat, 2015) Exploit F00F to install a rootkit in the System Management Mode (SMM).

Example: Intel Pentium `F00F`, Reloaded

"Oh it's just an old bug. Who uses Pentiums anymore? Plus it's easily mitigated and it just forces a reboot. No big deal."

(BlackHat, 2015) Exploit `F00F` to install a rootkit in the System Management Mode (SMM).

Can wipe the UEFI, re-infect the OS after a clean install, bypass "Secure Boot" sequences, etc.

Part II
Physics, Caches, Injection, Beyond the Classical Theory

Cache-cache

A cache is some fast-and-expensive memory, that is used to avoid fetching slow-and-cheap memory. It's a form of 'look-up table'.

HDD	DRAM	L3	L2	L1	registers
1 ms	100 ns	80 cy	10 cy	4 cy	1 cy
1 TB	16 GB	8 MB	256 kB	32 kB	8 B

- If we've been clever and pre-loaded the right stuff, data is ready and we can proceed: it's a cache hit
- Otherwise, we have to ask the next cache in line, it's a cache miss

The difference is small, but measurable: at the very least we found a timing channel.

Cache-cache

More specifically, we can have

- A cold-start miss, if the requested data is missing from the cache,
- A capacity miss, if the requested data is larger than the cache,
- A conflict miss, if the requested data could fit, but won't due to e.g. alignment constrains

Since most architectures have only few registers, the vast majority of programs use caches at some point.

Reset, initialisation, microarch

- Reset attack: Cache memory is volatile, and often easy to erase (cold boot). It is also possible to fill it with crap.
- Initialisation attack: if we can run a program on the same system, it will fill the caches in a controlled way.
- Microarchitectural attack: the status of the cache is known and containst everything needed for the targeted program.

- Flush + Reload / Evict + Reload
- Flush + Flush (why ?)
- Prime + Probe

What do I need to run a cache attack?

Does it work in practice?

Does it work in practice?

It depends.

Does it work in practice?

It depends.
Yes. Maybe. Definitely!

First demonstrated in 2005

- OpenSSL, full secret AES key extracted in 13 ms
- Android dm-crypt, full secret AES key in 65 ms

But wait, there's more!

Spooky Cache-cache

Most CPUs do not execute a program as written, for instance when meeting a conditional jump, they execute both branches (why?)

Spooky Cache-cache

Most CPUs do not execute a program as written, for instance when meeting a conditional jump, they execute both branches (why?)

This speculative execution mechanism may however alter caches. So it leaks information.

Spooky Cache-cache

Most CPUs do not execute a program as written, for instance when meeting a conditional jump, they execute both branches (why?)

This speculative execution mechanism may however alter caches. So it leaks information.

Building on this observation since 2017 about two dozen attacks have been described targeting modern CPUs:

- Spectre (1, 1.2, 2, NG, NG1.1, NG3a, NG4, RSB)
- Meltdown
- SWAPGS
- MDS: Fallout, RIDL, ZombieLoad (MSBDS, MLPDS, MFBDS, MDSUM)
- Foreshadow/L1TF

How do you detect/mitigate such an attack?

Virtual Cache-cache

On a laptop or a phone, you have your own cache.

Virtual Cache-cache

On a laptop or a phone, you have your own cache.

But in a cloud environment, you share your cache with many people.

Virtual Cache-cache

On a laptop or a phone, you have your own cache.

But in a cloud environment, you share your cache with many people.

If you're lucky you can read/write in other people's memories. Why would you do that?

Virtual Cache-cache

On a laptop or a phone, you have your own cache.

But in a cloud environment, you share your cache with many people.

If you're lucky you can read/write in other people's memories. Why would you do that?
On Amazon EC2:

- Keystroke recovery (SWT01)
- Stealing secret keys (ZJRR12, IGIES15, IIES14)

No amount of sandbox or virtual machine would change the hardware.

Bonus: Clémentine Maurice's [Intro to cache attacks]

Ain't got no time for that!

Ok so we just discussed caches and how they give us time. But what is time anyway?

Ain't got no time for that!

Ok so we just discussed caches and how they give us time. But what is time anyway?

Computers are not theoretical devices.

They are physical devices, and the usual (software-oriented) engineer makes the confusion between model and reality.

All models are lies, some are useful

Motivating, textbook example

Exercice: write a program that takes as input k and outputs g^k. The program must not reveal k.

Motivating, textbook example

Exercice: write a program that takes as input k and outputs g^k. The program must not reveal k.

Ok so now you already realise you can't just compute $g \times \cdots \times g$. (tell me why)

Motivating, textbook example

Exercice: write a program that takes as input k and outputs g^k. The program must not reveal k.

Ok so now you already realise you can't just compute $g \times \cdots \times g$. (tell me why)

But you also realise that the operation must somehow depend on k.

The general idea

- The system depends on some unknown k
- Predict what the system would do if $k = x$
- Compute correlation between prediction and reality

If it doesn't work: wrong hypothesis, try another x. Otherwise...

Example (demo?): password finding with timing information.

A better algorithm for g^x

The square-and-multiply algorithm (`x_bin` is the binary representation of x)

```
def power(g,x_bin):
    result = 1
    while i = x_bin.pop() and i != None:
        if i == 1:
            result = result * g
        g = g*g
    return result
```

Side note: this operation is essential to many cryptographic algorithms.

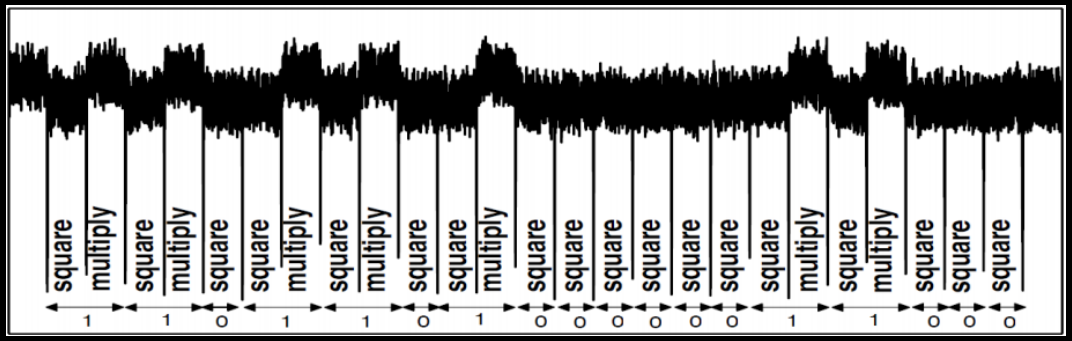
Look ma'! No timing channel!

At first glance this is a better algorithm than the naive one:

- Much faster: complexity $O(\log x)$ vs $O(x)$
- No timing channel: if x has a fixed size we are fine

So why am I talking about it?

Simple power analysis: square and mutiply



We literally read the secret bits from the power consumption trace!

SPA: A few comments

- Here 'correlation' really is obvious (sometimes, real math needed)
- Especially effective against embedded devices
- How do you prevent this?

Photons?

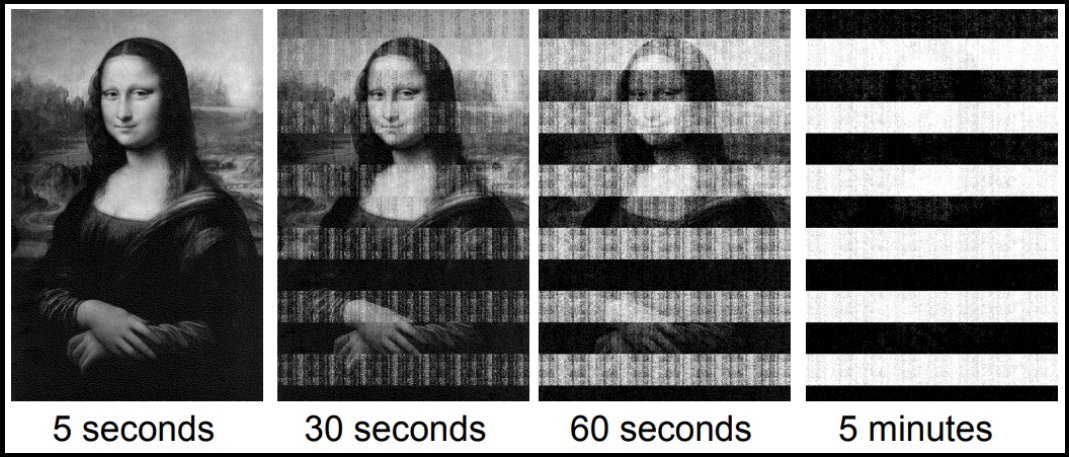
Transistors emit photons when they switch

- 10^2 to 10^4 photons per switch with peak in NIR region (900–1200 nm)
- Can be detected with photomultipliers and CCD cameras
- Comes from area close to the transistor drain (mostly NMOS side)

So we can literally see transistors switching.

Data remanence

Data remanence



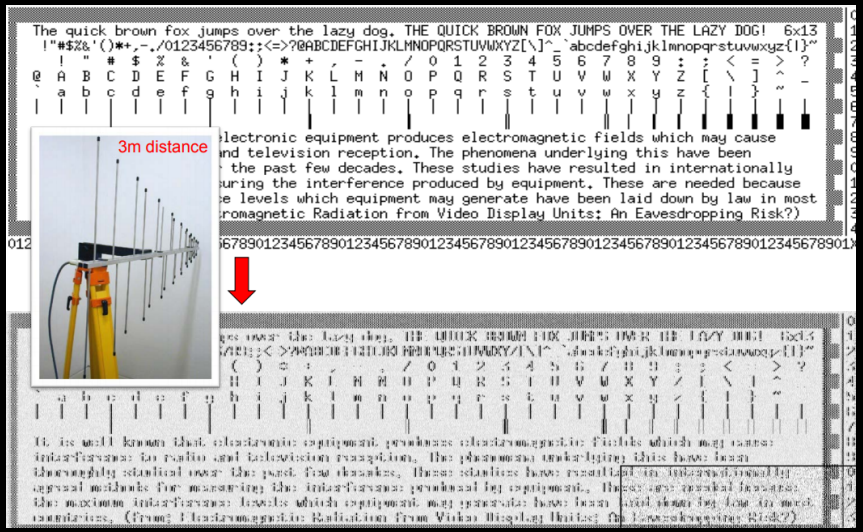
At -50 degrees C, memory remains unchanged for about 10 hours $\Rightarrow$ Cold boot attacks

Reading screens from 8 km away

How do you call a wire with varying current?

Reading screens from 8 km away

How do you call a wire with varying current?



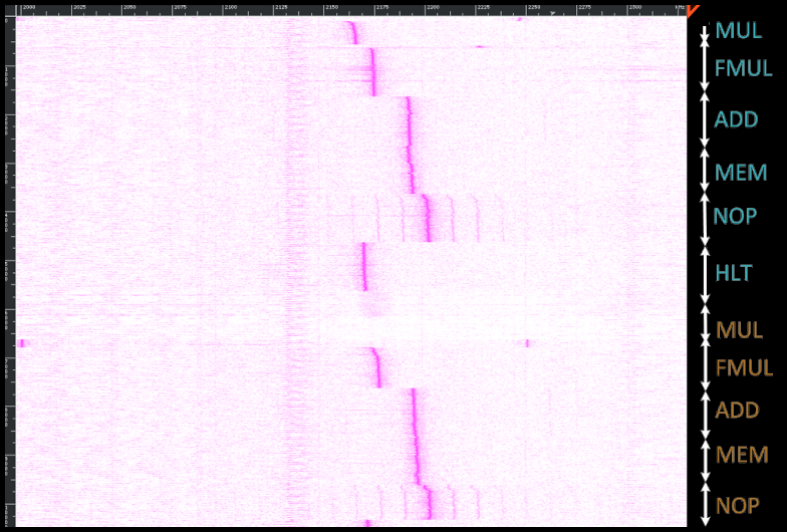
Video?

Reading keystrokes from 25 m away

How do you call a free USB charger/HDMI cable/Power outlet?

Reading keystrokes from 25 m away

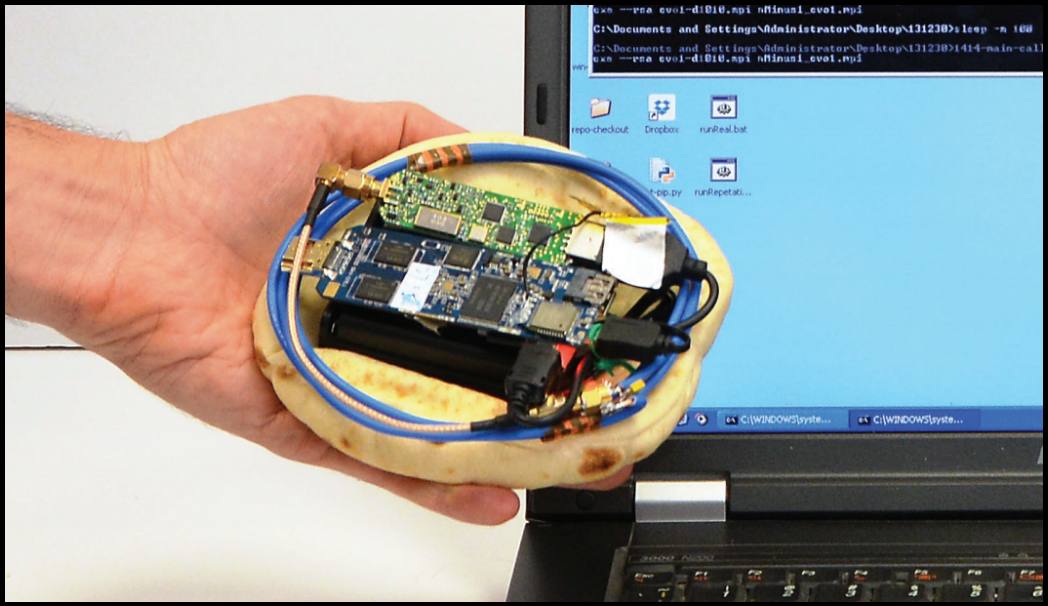
How do you call a free USB charger/HDMI cable/Power outlet?



The “ground” leaks information up to several meters! Even through contact.

Have some hummus?

Portable Instrument for Trace Acquisition – PITA

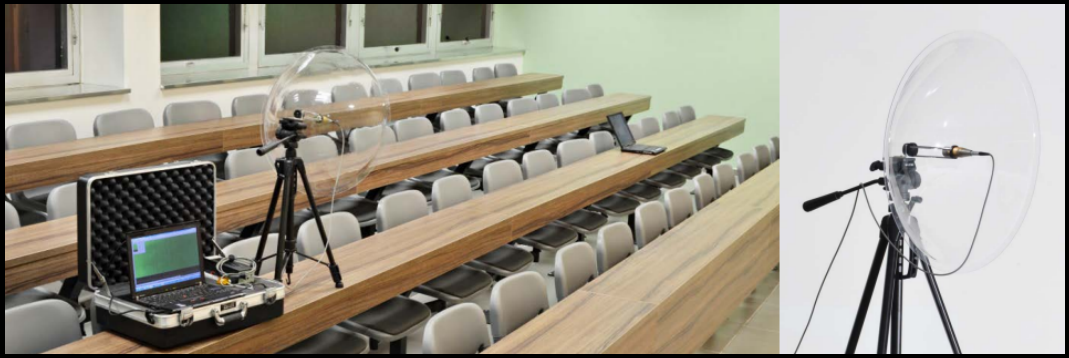


Acoustic side-channels

Have you ever noticed that your computer makes noise?

Acoustic side-channels

Have you ever noticed that your computer makes noise?



RSA 4096-bit key extraction from 10 meters away using a parabolic microphone.

Hardware trojan

Sometimes there isn't a readily usable side or covert channel.

So maybe we can create one?

This is easier if we are / have access to the manufacturer.

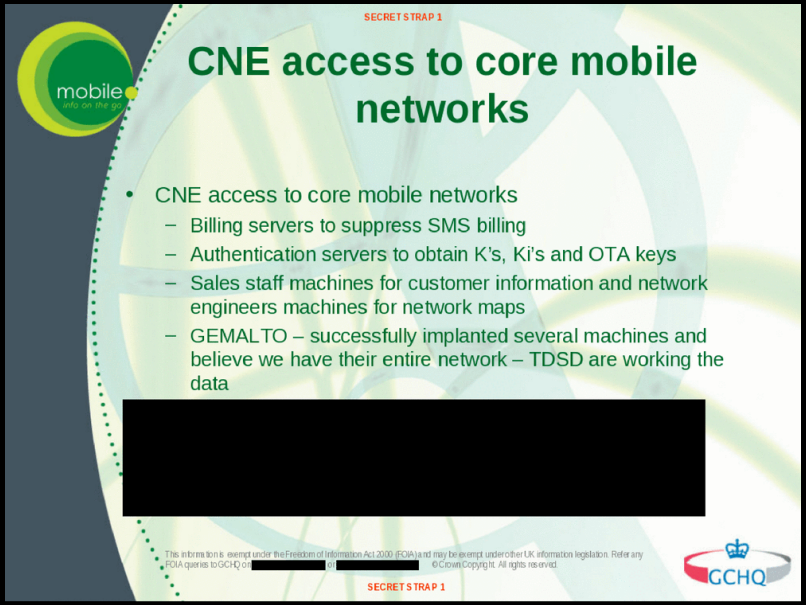
The Clipper Chip 1993

Hardware trojans (cont'd) / DAPINO GAMMA 2010

February 19 2015, 8:25 p.m.

MERICAN AND BRITISH spies hacked into the internal computer network of the largest manufacturer of SIM cards in the world, stealing encryption keys used to protect the privacy of cellphone communications across the globe, according to top-secret documents provided to *The Intercept* by National Security Agency whistleblower Edward Snowden.

The hack was perpetrated by a joint unit consisting of operatives from the NSA and its British counterpart Government Communications Headquarters, or GCHQ. The breach, detailed in a secret 2010 GCHQ document, gave the surveillance agencies the potential to secretly monitor a large portion of the world's cellular communications, including both voice and data.


SECRET STRAP 1
mobile
info on the go
CNE access to core mobile networks
CNE access to core mobile networks
Billing servers to suppress SMS billing
Authentication servers to obtain K's, Ki's and OTA keys
Sales staff machines for customer information and network engineers machines for network maps
GEMALTO – successfully implanted several machines and believe we have their entire network – TDSD are working the data
[redacted]
This information is exempt under the Freedom of Information Act 2000 (FOIA) and may be exempt under other UK information legislation. Refer any FOIA queries to GCHQ on [redacted]. ©Crown Copyright. All rights reserved.
GCHQ
SECRET STRAP 1

Hardware trojans (cont'd)

- 3500 counterfeit Cisco network components were discovered in the US (FBI)
- Backdoor in a military grade FPGA device (Skorobogatov 2012)
- Mobile phones developed by Chinese device manufacturer ZTE have been found to carry a backdoor to instantly gain root access (Alperovitch 2012)

China anti-terrorism bill, passed at the end of 2015, took effect on the first day of 2016. Amongst the provisions, is the requirement that every manufacturer provides the Chinese government with "backdoor" access to their products.

About 80% of all electronic devices in circulation are manufactured in China.

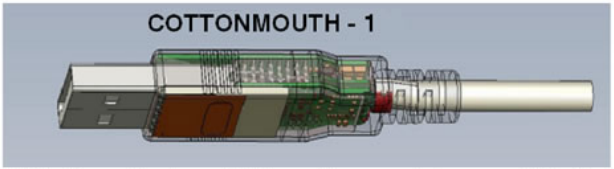
TOP SECRET//COMINT//REL TO USA, FVEY

COTTONMOUTH-I

ANT Product Data

(TS//SI//REL) COTTONMOUTH-I (CM-I) is a Universal Serial Bus (USB) hardware implant which will provide a wireless bridge into a target network as well as the ability to load exploit software onto target PCs.

08/05/08



(TS//SI//REL) CM-I will provide air-gap bridging, software persistence capability, "in-field" re-programmability, and covert communications with a host software implant over the USB. The RF link will enable command and data infiltration and exfiltration. CM-I will also communicate with Data Network Technologies (DNT) software (STRAITBIZARRE) through a covert channel implemented on the USB, using this communication channel to pass commands and data between hardware and software implants. CM-I will be a GENIE-compliant implant based on CHIMNEYPOOL.

Così fan tutte

To what extent can we trust hardware components?

What techniques and countermeasures can be used to prevent, detect, or circumvent side-channels/covert channels/hardware trojans?

While you think about it

Let's have a break