50.020 Security
Lecture 18 - Side Channel attacks

This lecture

- Side-Channel Attacks
  - Physical-layer Attacks
    - ATM/Credit Card protocols
    - Skimming Attacks
    - Relay Attacks
    - Other Physical-Layer Attacks
  - Side-Channel Attacks
    - Side-Channel Attack Example 1: Power-monitoring Attack
    - Side-Channel Example 2: Cache-based attack (Spectre Attack is one example)

Physical-Layer Attacks

History

- Cheques
  - You fill the amount and target account, sign
  - Cheque will be deposited at bank, and money will be transferred
- Why assumed to be secure?
  - Signature
  - Basic forgery protection in cheque paper
  - Serial number on cheque
- Early Credit cards (starting 1934) used embossing
  - Physical imprint of the number on card was used
  - Phone calls were required for manual verifications

Overview Credit/Debit Cards

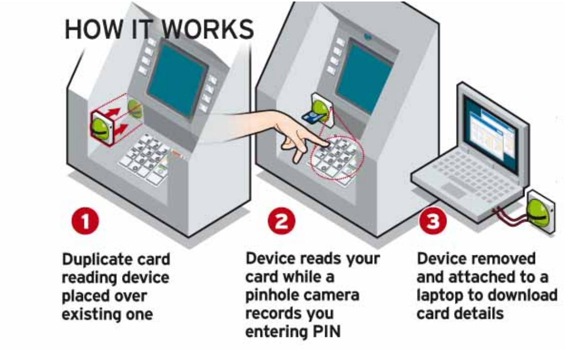
- Two-factor authentication
  - Physical Token (card with ID)
  - Password (PIN number for Signature)
- ID can be stored on magnetic stripe or microchip
- Although microchips are used everywhere in Singapore, not the same elsewhere
  - US, for example, seems to still widely use magnetic stripe
  - Even Singaporean cards have the stripe, but it is de-activated

Skimming Attacks

`https://www.youtube.com/watch?v=DTOoIip-zaQ`
`https://www.youtube.com/watch?v=MOoiYKhJ5NM`

Skimming Attacks: Magstripe

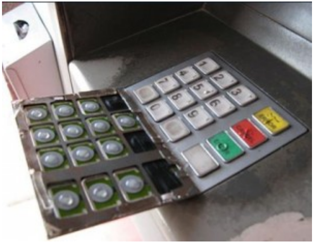
- Magnetic stripe information can be copied by *skimming devices*
- Device is located in front of ATM slot
- Card details are used to create copy of card


HOW IT WORKS
1
Duplicate card reading device placed over existing one
2
Device reads your card while a pinhole camera records you entering PIN
3
Device removed and attached to a laptop to download card details

Source: www.antiskimmingeye.com

Skimming Attacks: Stealing the Pin

- PIN required in addition to copied card
- Can be obtained by
  - Hidden camera (e.g. in skimmer)
  - Additional capturing membrane around keys
  - Additional keypad on top of original keypad

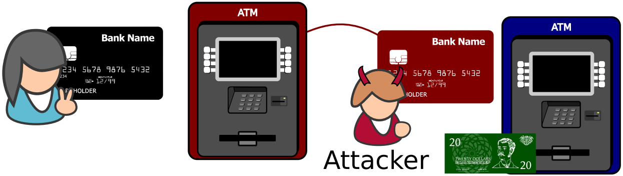


Impact of Skimming

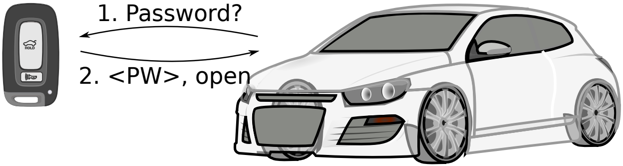
- The European ATM Security Team (EAST) reported
  - 5,822 attacks in 2013, 201 Euros million
  - 5,631 attacks in 2014, 238 Euros million
  - ˜4000 attacks in 2015
  - ˜3000 attacks in 2016
- By now, countermeasures have been implemented in many countries
- Card reader in supermarkets have also been infected with malware in the past

ATM Relay Attacks

- Theoretically, relay attacks are possible on chip-based cards
- The attacker sets up a fake ATM, that forwards the communication
- Real money is given out at attacker
- Not really seen in practise yet



Wireless car keys

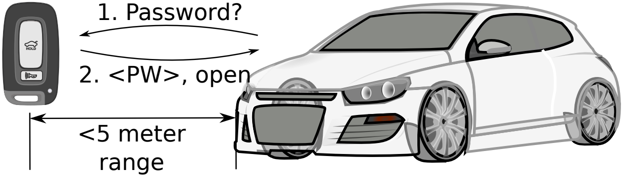


1. Password?

2. <PW>, open

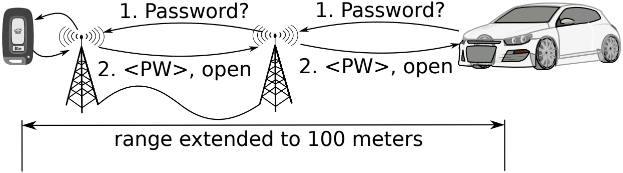
- Modern cars can be unlocked wirelessly
- Without touch of a button on remote! Can even start engine . . .
- Convenient! But: is this secure?

Wireless car keys



- Lets assume protocol is secure
- Radio signal range is low, <5m
- So key owner is always close. Is this secure?

Wireless car keys



- Attacker can forward messages to extend range
- Forwarding only requires two antennas and a cable
- Range extended to $>$ 100m in experiments

Relay Attack Analysis

- Worked on all tested cars with hand-free wireless opener
- More expensive models more susceptible . . .
- On data layer, everything is fine (car was opened by key)
  - Attacker provided a service by signal relay . . .
- So, where is the actual problem, how to protect?

Relay Attack Analysis

- Worked on all tested cars with hand-free wireless opener
- More expensive models more susceptible . . .
- On data layer, everything is fine (car was opened by key)
  - Attacker provided a service by signal relay . . .
- So, where is the actual problem, how to protect?

- From Information Security perspective, original protocol broken
  - With unlimited connectivity, car would constantly unlock
- But customers want this feature, how to fix?

Physical-layer Distance Bounding

- *Distance Bounding* protocols:
  - A and B exchange messages, record timing of messages
  - DB provides an upper bound on distance between A and B
  - Distance of responding partner, not forwarder
- DB protocols solve our car key problem!
  - Only open if key is close ($<$5m)
- But: implementation a BIG problem
  - Timing measurement needs to be in *ns* scale
  - No commercial product yet
  - Product could also be used for localization etc.
- Modeling/ proofs of this also a hard problem

Other Physical-Layer Attacks

What about wireless communications?

- Wireless communications
  - Jamming, Anti-Jamming
  - Long-distance eavesdropping
  - Radio fingerprinting
  - Others
- Localization protocols
  - GPS (time-of-arrival)
  - Wifi-based localization (RSSI)
  - Any other wireless localization. . .

Side-Channel Attacks

- Side-channel Attack Types
- Side-Channel Attack Example 1: Power-Monitoring Attack
- Side-Channel Attack Example 2: Cache-based (Spectre Attack is one exmple)

Side-Channel Attacks

A side-channel attack is any attack based on information gained from the implementation of a computer system, rather than weaknesses in the implemented algorithm itself. Possible types of side-channel attacks:

- Power Monitoring Attack
- Timing Attack
  - attacks based on measuring how much time various computations take to perform
- Electromagnetic Attack
  - Read screens through walls
  - Eavesdrop on USB keyboard keystrokes
  - `https://www.youtube.com/watch?v=AFWgIAgMtiA`
- Audio side-channels
- Optical side-channels
  - Reflections on objects
- Other types
- All of these are practically exploited!

Side-Channel Example 1: Power-Monitoring Attack

Side-Channel Example 1: Power-Monitoring Attack

- Smartcards are used to securely store information
  - Contains protected memory
  - Well-defined API to get or set data in memory from outside
- Can be used to produce signatures using secret keys
  - Signatures are used to certify authenticity of data
  - Key is required to produce signature
  - API will never disclose key to outside world
- So signatures can only be made by smartcard
- Sounds perfect? It was until '96 . . .

Attack scenario

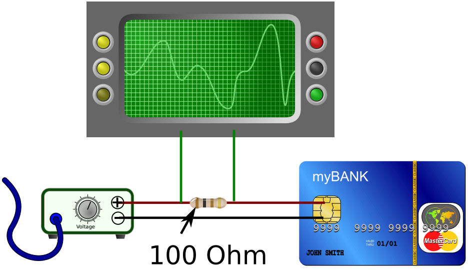
- Attacker has physical access to smartcard
    - Smartcard uses RSA + square and multiply
- Attacker wants to learn secret key stored on card
    - For example, to clone the card (PayTV)
- Attacker can trigger a signature operation
- How can this be used by an attacker?

Power Consumption based Side-Channel Analysis

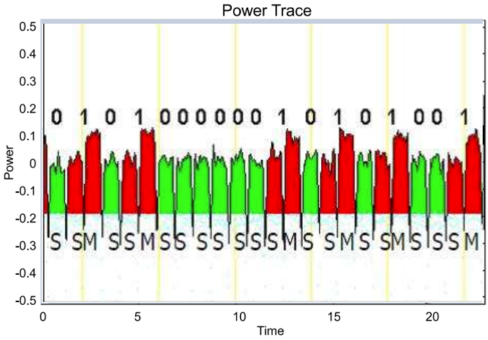
- Attacker can measure power consumption
  - Power supplied by attacker
  - Power consumption varies with operation
- Operations depend on the key. . .
- Try to find the exponent (i.e., key) based on power consumed

Power Consumption based Side-Channel Analysis

- Victim: Sensor node running RSA Enc+Dec after boot
    - Encryption key is public, known
    - Decryption key is private, goal of attacker
- An oscilloscope is used to capture the power consumption



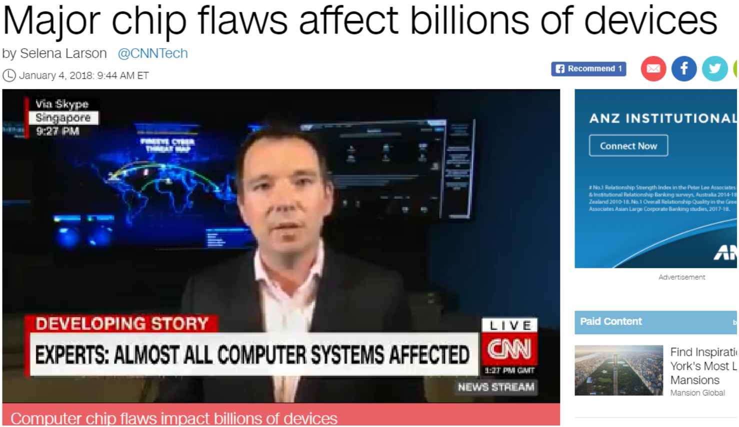
Power consumtion of Mult/Square



- In many practical implementation, power consumption of Mult/Square will differ
- Because squaring needs only one operand, it is faster
- As result, power trace of multiplication and squaring is different

Side-Channel Example 1: Cache-based Attack

Side-channel Attack Example 2: Spectre and Meltdown Attacks



Major chip flaws affect billions of devices

by Selena Larson @CNNTech

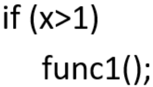
January 4, 2018: 9:44 AM ET

Computer chip flaws impact billions of devices

Speculative execution

Instead of idling, CPUs can guess likely program path and do speculative execution

For example, given following code:

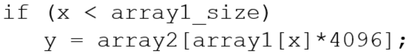


```
if (x>1)
    func1();
```

- Branch predictor: if() will probably be true (based on prior history)
- CPU starts func1() speculatively, but does not commit changes
- When value arrives from memory, if() can be evaluated definitively. check if guess was correct:
  - Correct: Commit speculative work and gain performance
  - Incorrect: Discard speculative work

Spectre Attack: Example

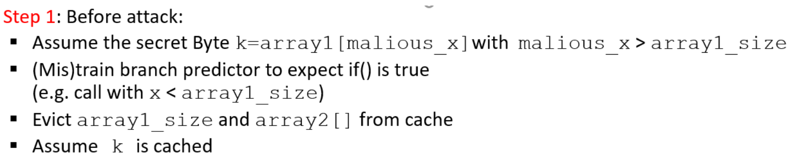
- Consider the case where the following code is part of a function (e.g., a system call or a library) receiving an unsigned integer x from an untrusted source.



```
if (x < array1_size)
   y = array2[array1[x]*4096];
```

- Execution without speculation is safe.
- What about with speculative execution?

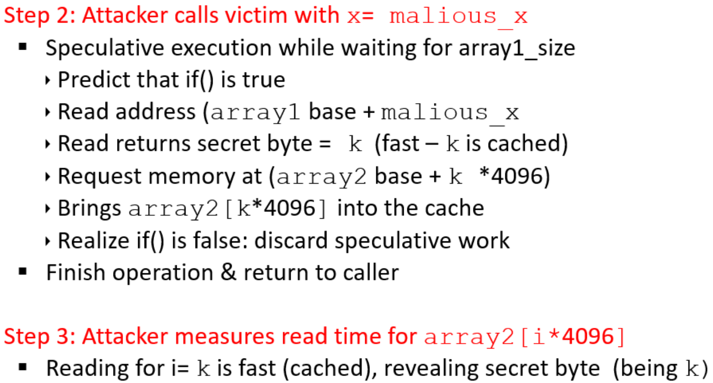
Spectre Attack Example: How to Attack



Step 1: Before attack:

- Assume the secret Byte `k=array1[malious_x]` with `malious_x` > `array1_size`
- (Mis)train branch predictor to expect if() is true (e.g. call with `x` < `array1_size`)
- Evict `array1_size` and `array2[]` from cache
- Assume `k` is cached

Spectre Attack Example: How to Attack



Step 2: Attacker calls victim with x= malious_x

- Speculative execution while waiting for array1_size
  - Predict that if() is true
  - Read address (array1 base + malious_x
  - Read returns secret byte = k (fast – k is cached)
  - Request memory at (array2 base + k *4096)
  - Brings array2[k*4096] into the cache
  - Realize if() is false: discard speculative work
- Finish operation & return to caller

Step 3: Attacker measures read time for array2[i*4096]

- Reading for i= k is fast (cached), revealing secret byte (being k)

If interested, you can read the following reference for details. `https://spectreattack.com/spectre.pdf`

Conclusion

- Low-level physical attacks are possible in many cases
- Even with trusted devices (e.g. chips, smartcards) relay attacks are possible
- Theoretical attacks for ATMs and similar card reader devices
- Practical attacks for wireless car keys
- Side-Channel attacks might recover secret data
- Possible countermeasures against side-channel attacks: improve design
  - Reduce EM radiation through shielding
  - Make timing worst case always
  - Try to achive data-independent power consumption