Security Models & Covert Channels

ECE568 – Lecture 18

Courtney Gibson, P.Eng.
University of Toronto ECE

Outline

Security Policies

- The Bell-LaPadula Security Model
- Biba Integrity Model

Side-Channels Covert Channels

- Types of side-channels
- Covert Channels vs. Side-Channels
- Non-interference

Security Policies

Security Policies

Security policies govern how a system handles information

- Their goal is to ensure that a system maintains some security property
- They are mainly used by the military, government, etc.

Security policies are based on **security models** that are generally quite intuitive, and yet rigorous enough that they stand up to proofs for security

- These models help check if a system is designed securely
- The system will be secure if there are no bugs

Security policies are implemented as mandatory access control (MAC) policies

- They work with or trump any user-specified DAC policies

Types of Policies

Confidentiality Policies

- Confidentiality defines who is authorized to access information or resources
- These policies help protect information from leaking
- Historically, they are used for military applications
  - Protect secret information from being leaked to the enemy
  - Companies can protect customer privacy or corporate secrets

Integrity policies

- Integrity defines the trustworthiness or reliability of information
- These policies help prevent corruption of information
- Normally, they are used by companies
  - Protect destruction, defacement of company data

Bell-La Padula (BLP) Model

The Bell-La Padula Model or Multi-level Security Model (MLS) is used to build confidentiality policies

- It has two types of **elements** in the system
  - **Subjects** are active participants in the system
  - **Objects** are data or resources that need to be protected
- There are various **classification levels** defined by the system, ordered from highest to lowest
  - *e.g.*, Top Secret (TS), Secret (S), Classified (C) and Unclassified (UC)
- There are a number of **categories** in the system
  - *e.g.*, FBI, NSA, President (POTUS)

Bell-La Padula (BLP) Model

Each object is associated with a **security level** of the form (classification level, set of categories)

- *e.g.*, {TS, {NSA, POTUS}} means this object contains top secret information pertaining to NSA and the President

- Each subject has a **security clearance** up to a given classification level, and their maximum security level is defined as (security clearance, set of categories)
  - *e.g.*, a subject may be assigned the {S, {NSA}} security level

Category Set Lattice

The set of categories forms a **lattice** (a partially ordered set) with a **dominance** relation

If element E_1 has a higher security clearance than element E_2, and E_1's category set contains E_2's category set, then E_1 is said to **dominate** E_2

- *e.g.*, {TS, {NSA,POTUS}} dominates {S,{NSA,POTUS}, {TS, {NSA}}, etc.

Bell-La Padula Security Model

Bell-La Padula (BLP) defines two rules:

- **Simple Security Property:** A subject **s** can read an object **o** iff the security level of **s** dominates security level of **o**, and **s** has discretionary read access to **o**
- ***-Property (Star Property):** **s** can write **o** iff the security level of **o** dominates security level of **s**, and **s** has discretionary write access to **o**

- These rules say you can only read from things less secure than you (*i.e.*, no read up) and write to things more secure than you (*i.e.*, no write down)
- Information can **never** flow from a more secure (confidential) source to a less secure one
- In practice, exceptions require "trusted subjects"

Biba Integrity Policy

Bell-La Padula does not provide any integrity, because a low security level subject can modify high security level objects

- The Biba Model is used to build integrity policies
  - It prevents low integrity information from affecting higher integrity information
- Biba is the mathematical dual of Bell-LaPadula
  - Each Subject and Object has an associated integrity level
  - Also, subjects can execute other subjects (*i.e.*, programs)

Biba Integrity Policy

Biba has three rules:

- **Simple Integrity Property: s** is permitted to read an object **o** iff the integrity level of **s** is lower than the integrity level of **o**, i(s) ≤ i(o) (*i.e.*, no read down)
- ***-Integrity Property:** A subject **s** is permitted to write to an object **o** iff the integrity level of **o** is lower than the integrity level of **s**, i(o) ≤ i(s) (*i.e.*, no write up)
- **s1** can execute **s2** iff i(s2) ≤ i(s1)

Side Channels & Covert Channels

Unexpected Consequences

- We have, thus far, been examining potential problems that can manifest with ***intentional*** inputs and outputs.
- Your system likely has a number of ***unintentional*** I/O channels that can potentially be exploited by an attacker.

Side Channels

A **side-channel** allows **unintentional** information flow out of (or into) your system.

- User activity
- Sensor data
- Cryptographic keys

Channels can come in many unexpected forms:

- R/F leakage
- Power usage
- Event timing
- …

Side Channels

While some side-channels can be remotely exploited, risks increase considerably when the attacker has local / physical access.

- Smart cards
- IoT devices
- Mobile devices
- Cloud

Timing Analysis

Observation: The execution time of algorithms can sometimes leak important security information.

```
int
strcmp(const char * s1, const char * s2)
{
  register const unsigned char *string1 = (const unsigned char *)s1;
  register const unsigned char *string2 = (const unsigned char *)s2;
  register unsigned char c1;
  register unsigned char c2;

  do {

    c1 = (unsigned char) *string1++;
    c2 = (unsigned char) *string2++;
    if (c1 == '\0') return (c1 - c2);

  } while (c1 == c2);

  return(c1 - c2);
}
```

Runtime proportionate to the number of **matching** characters.

Timing Analysis

Timing Analysis

Observation: When measuring remotely, it can require a reasonable amount of filtering, due to jitter in the communication channel.

strcmp(“hello”, “a”);	180ns
strcmp(“hello”, “b”);	176ns
strcmp(“hello”, “c”);	184ns
…	…
strcmp(“hello”, “h”);	192ns
strcmp(“hello”, “i”);	183ns

Demo

Password-Cracking with a Stopwatch

Test Device
ESP32
microcontroller

"Victim"
ATMega328P
microcontroller

BOOT
EN
3V3 GND D15 D2 D4 RX2 TX2 D5 D18 D19 D21 D3 TX0 D22 D23
VIN GND D13 D12 D14 D27 D26 D25 D33 D32 D35 D34 VN VP EN
ESP-WROOM-32
ATMEGA328P
TIMING ANALYSIS
s3cr3tP4
3.3V/8MHz/28800-N-8-1
GND VCC SCK SDA

Activities
Applications
Terminator
Mar 7 21:46:55
100 %
cgibson@waves: ~
No candidate found: checking again
Character found: "s3c" (baseTime: 13715, thisCharAverage: 13895, delta: 180)
Character found: "s3cr" (baseTime: 14244, thisCharAverage: 14491, delta: 247)
Character found: "s3cr3" (baseTime: 14742, thisCharAverage: 14913, delta: 171)
CTRL-A Z for help | 115200 8N1 | NOR | Minicom 2.8 | VT102 | Offline | ttyUSB0

TIMING ANALYSIS
s3cr3tP4sswOrd&:-)@#
3.3V/8MHz/28800-N-8-1
GND VCC SCK SDA
D21 RX0 TX0 D22 D23
D35 D34 VN VP EN
WiFi ESP-WROOM-32
TEXTOOL

Timing Analysis

Recommendation: Security-sensitive algorithms should either have constant or randomized runtime, independent of input.

Remote Timing Analysis

Fig. 4. Dependency between number of leading zero bits and wall clock execution time of the signature operation.

Reference: *Remote Timing Attacks are Still Possible*, Brumley et al., ESORICS 2011
http://packetstorm.linuxsecurity.com/papers/cryptography/timing-attacks.pdf

Electromagnetic Side-Channel

- Wim van Eck published a seminal paper in 1985, examining the possibility of using stray signals from monitors, in order to reconstruct confidential video signals

Electromagnetic Side-Channel

- Original work was done on CRT (Cathode Ray Tube) screens, but current works shows that this is still possible, using timing channel of modern LCD monitors

Reference: *Synchronization Clock Frequency Modulation Technique for Compromising Emanations Security*, Watanabe et al., EMC'09/Kyoto
http://www.ieice.org/proceedings/EMC09/pdf/21P1-4.pdf

Electromagnetic Side-Channel

- Same principle can be applied to other I/O devices (*e.g.*, keyboards)

Reference: *Compromising Electromagnetic Emanations of Wired and Wireless Keyboards*, Vuagnoux et al., USENIX 2009
http://www.usenix.org/events/sec09/tech/full_papers/sec09_attacks.pdf

Power Analysis

Observation: The power *input* to our circuit is often also an unintended *output* that can leak considerable information about the system's operation

- IoT Devices
- Smart Cards
- Security Tokens
- …

Demo

Safe-Cracking with an Oscilloscope

MAXIMUM
SECURITY SAFES

1Q
2ABC
3DEF
4GHI
5JKL
6MNO
7PRS
8TUV
9WXY
*
0Z
#
WARNING
WARRANTY VOID IF COVER REMOVED

6120-305
04 / 2016
SARGENT & GREENLEAF INC.
Nicholasville, KY USA
www.sargentandgreenleaf.com
SECURITY
LISTED
HIGH SECURITY ELECTRONIC LOCK
TYPE 1 ANTL
23930410
CE
PROCELL
DURACELL
9V
Pile alcaline professionnelle
Professional Alkaline Battery

ALIGN RIDGE WITH GLOV
61HP-101
D2
D3
J3
Z1

Idea: Put a small "current sense" resistor between the battery and the device under test. The voltage across the resistor will give us a "fingerprint" of the device's power consumption.

VEGA
6130
6120V204
REV

MICROCHIP

93AA46A/B/C, 93LC46A/B/C, 93C46A/B/C

1K Microwire Compatible Serial EEPROM

Device Selection Table

Part Number	Vcc Range	ORG Pin	Word Size	Temp Ranges	Packages
93AA46A	1.8-5.5	No	8-bit	I	P, SN, ST, MS, OT, MC, MN
93AA46B	1.8-5-5	No	16-bit	I	P, SN, ST, MS, OT, MC, MN
93LC46A	2.5-5.5	No	8-bit	I, E	P, SN, ST, MS, OT, MC, MN
93LC46B	2.5-5.5	No	16-bit	I, E	P, SN, ST, MS, OT, MC, MN
93C46A	4.5-5.5	No	8-bit	I, E	P, SN, ST, MS, OT, MC, MN
93C46B	4.5-5.5	No	16-bit	I, E	P, SN, ST, MS, OT, MC, MN
93AA46C	1.8-5.5	Yes	8- or 16-bit	I	P, SN, ST, MS, MC, MN
93LC46C	2.5-5.5	Yes	8- or 16-bit	I, E	P, SN, ST, MS, MC, MN
93C46C	4.5-5.5	Yes	8- or 16-bit	I, E	P, SN, ST, MS, MC, MN

Pin Function Table

Name	Function
CS	Chip Select

Features:

- Low-Power CMOS Technology
- ORG Pin to Select Word Size for '46C' Version

PDIP/SOIC (P, SN)

Name	Function
CS	Chip Select
CLK	Serial Data Clock
DI	Serial Data Input
DO	Serial Data Output
Vss	Ground
NC	No internal connection
ORG	Memory Configuration
Vcc	Power Supply

6130
6120V204
Made in China
RoHS
Q4
R2
R8
R9
R7
U2
Y1
C10
R11
D2
C2
Q3
R4
C4
R14
Q5
R3

1Q
2ABC
3DEF
4GHI
5JKL
6MNO
7PRS
8TUV
9WXY
*
0Z
#

MSO5104 Sun January 26 23:13:27 2020
RIGOL
STOP
H
1.00ms
200MSa/s
20Mpts
Measure
STOP/RUN
D
31.724ms
T
1.40V
A
CLK
TX
1
200mV B
+344mV
2
100mV
0.00V
3
100mV
0.00V
4
100mV
0.00V
L
GI
GII
23:13

MSO5104 Sun January 26 23:13:27 2020
RIGOL
STOP
H 1.00ms
200MSa/s
20Mpts
Measure
STOP/RUN
D 31.724ms
T 1.40V A
CLK
TX
1 200mV B
+344mV
2 100mV
0.00V
3 100mV
0.00V
4 100mV
0.00V
L
GI
GII
23:13

MSO5104 Sun January 26 23:13:27 2020
RIGOL
STOP
H 1.00ms
200MSa/s
20Mpts
Measure
STOP/RUN
D 31.724ms
T 1.40V A
CLK
TX
1
0
1 200mV B
+344mV
2 100mV
0.00V
3 100mV
0.00V
4 100mV
0.00V
L
GI
GII
23:13

0 0 1 1 0 1 0 0

MSO5104 Sun January 26 23:13:27 2020
RIGOL
STOP
H
1.00ms
200MSa/s
20Mpts
Measure
STOP/RUN
D
31.724ms
T
1.40V
A
CLK
TX
1
200mV B
+344mV
2
100mV
0.00V
3
100mV
0.00V
4
0.00V
L
GI
GII
23:13

MSO5104 Sun January 26 23:14:36 2020
RIGOL
STOP
H 5.00ms
200MSa/s
20Mpts
Measure
STOP/RUN
D 31.724ms
T 1.40V A
CLK
TX
1 200mV B +352mV
2 100mV 0.00V
3 100mV 0.00V
4 100mV 0.00V
L
GI
GII
23:14

MSO5104 Sun January 26 23:14:36 2020
RIGOL
STOP
H 5.00ms
200MSa/s
20Mpts
Measure
STOP/RUN
D 31.724ms
T 1.40V A
CLK
TX
1 2
3 4
5 6
1 200mV B
+352mV
2 100mV
0.00V
3 100mV
0.00V
4 100mV
0.00V
L
GI
GII
23:14

Activities
Applications
Terminator
Mar 7 21:51:02
100 %
cgibson@waves: ~/src/cgibson/SG6120
cgibson@waves:~/src/cgibson/SG6120$

1Q
2ABC
4GHI
5JKL
6MNO
7PRS
8TUV
9WXY
*
0Z
#

Differential Power Analysis

Reference: *Differential Power Analysis*, Kocher et al., CRYPTO'99
ftp://amusing.mit.edu/afs/net/dev/admin/www/root/org/a/auto-id/foley/Reference/kocher99diffpowatk.ps

Power Analysis

Fig. 7: Different power dissipation measurements for the AES-128 running at (a) 2MHz, (b) 9.6MHz, (c) 12MHz and (d) 16MHz, exhibit more switching noise generated at the higher frequency, hence the peaks are undistinguishable

Reference: *Pammu, Ali Akbar, et al. "A Highly Efficient Side Channel Attack with Profiling through Relevance-Learning on Physical Leakage Information." IEEE Transactions on Dependable and Secure Computing (2018).*

Power Analysis

Fig. 6. Noise current generation circuit using 2-level stochastic 10-bit LFSR, followed by a 5-bit current steering DAC.

Reference: *Das, Debayan, et al. "ASNI: Attenuated Signature Noise Injection for Low-Overhead Power Side-Channel Attack Immunity." IEEE Transactions on Circuits and Systems I: Regular Papers (2018).*

Power Analysis

Fig. 1. (a) Overall system architecture for RFVD, (b) supply and frequency dependence of correlation power analysis (CPA), (c) SCA leakage suppression of an AES core with voltage dithering, and (d) RFVD for fixed throughput encryption core.

Reference: *Singh, Arvind, et al. "Improved power side channel attack resistance of a 128-bit AES engine with random fast voltage dithering." ESSCIRC 2017-43rd IEEE European Solid State Circuits Conference. IEEE, 2017.*

Power Analysis

Fig. 3. RO placements on the FPGA fabric indicated by diamond marks.

Reference: *Zhao, Mark, and G. Edward Suh. "FPGA-based remote power side-channel attacks." 2018 IEEE Symposium on Security and Privacy (SP). IEEE, 2018.*

Audio Analysis

Reference: *Keyboard Acoustic Emanations Revisited*, Zhuang et al., TISSEC 2009
http://www.tygar.net/papers/Keyboard_Acoustic_Emanations_Revisited/ccs.pdf

Video Attacks

Reference: *ClearShot: Eavesdropping on Keyboard Input from Video*, Balzarotti et al., IEEE Conf on Security and Privacy 2008
http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.207.4423&rep=rep1&type=pdf

Reference: *iSpy: Automatic Reconstruction of Typed Input from Compromising Reflections*, Raguram et al., ACM Computer and Communications Security 2011
http://wwwx.cs.unc.edu/~amw/resources/ispy.pdf

Reference: *MAPS: Machine-based Automatic Phone Surveillance*, Goswami et al., DCS Technical Report, UNC Chapel Hill, 2011
ftp://ftp.cs.unc.edu/pub/techreports/11-004.pdf

Video Attacks

Reference: *Compromising Reflections – or – How to Read LCD Monitors Around the Corner*, Backes et al.., IEEE Symp. on Security & Privacy 2008
http://gauss.ececs.uc.edu/Courses/c653/extra/reflections.pdf

Covert Channels

Your system will have a security model that specifies how information **should** flow in the system when communicating **explicitly** (*e.g.*, read and write)

- A **covert channel** is one that an attacker creates intentionally, to allows **implicit** information flow, outside of the specified policy
  - *e.g.*, suppose two processes share a lock on the file, then one process can transmit one bit of information to the other process, by holding or not holding the lock
  - *e.g.*, a process can vary the amount of time it takes to execute by causing many page faults, thus transmitting information to another party

Covert vs. Side Channels

- Covert channels usually imply that information is sent intentionally, and the sender wishes to remain undetected
- With **side channels**, information is leaked unintentionally

Non-Interference

Covert channels are generally very hard to detect

- They exist whenever the actions of one process **affect** the actions of another process in some way, even though there is no explicit communication

Non-interference allows analyzing convert channels

- A system has the non-interference property if and only if any sequence of inputs to a process will produce the same outputs, regardless of inputs to another process

For example:

- Say a user with a 'low' clearance is working on a machine
- The machine will respond in the same way whether or not a user with a 'high' clearance is working on the machine
- Thus, the low user will not be able to acquire any information about the activities of the high user

Non-Interference

Non-interference is a very strict property and it is difficult to make a system satisfy this property

- However, every covert channel identified using non-inference analysis is not necessarily usable
  - The sender side must be able to control what is transmitted into the channel, and the receiver must be able to interpret the information that is sent
  - The channel must have sufficient **capacity** to be useful, *e.g.*, a channel may not be useful if only one bit of information can be transmitted every hour

Questions?