

Enrico RUSSO

Andrea VALENZA

Università di Genova

Weaknesses



<https://cybersecnatlab.it>

License & Disclaimer

2

License Information

This presentation is licensed under the
Creative Commons BY-NC License



To view a copy of the license, visit:

<http://creativecommons.org/licenses/by-nc/3.0/legalcode>

Disclaimer

- We disclaim any warranties or representations as to the accuracy or completeness of this material.
- Materials are provided “as is” without warranty of any kind, either express or implied, including without limitation, warranties of merchantability, fitness for a particular purpose, and non-infringement.
- Under no circumstances shall we be liable for any loss, damage, liability or expense incurred or suffered which is claimed to have resulted from use of this material.

Outline

3

- Confused Deputy
 - SetUID/SetGID
- Race Conditions
 - TOCTOU Race Conditions

Outline

4

- Confused Deputy
 - SetUID/SetGID
- Race Conditions
 - TOCTOU Race Conditions

Confused Deputy

5

- A specific type of Privilege escalation
- Exploits the Access Control List (ACL) model
 - Capability based systems are not affected

Original example

6

The Confused Deputy **(or why capabilities might have been invented)**

Norm Hardy
Senior Architect

<https://zoo.cs.yale.edu/classes/cs422/2010/bib/hardy88confused.pdf>

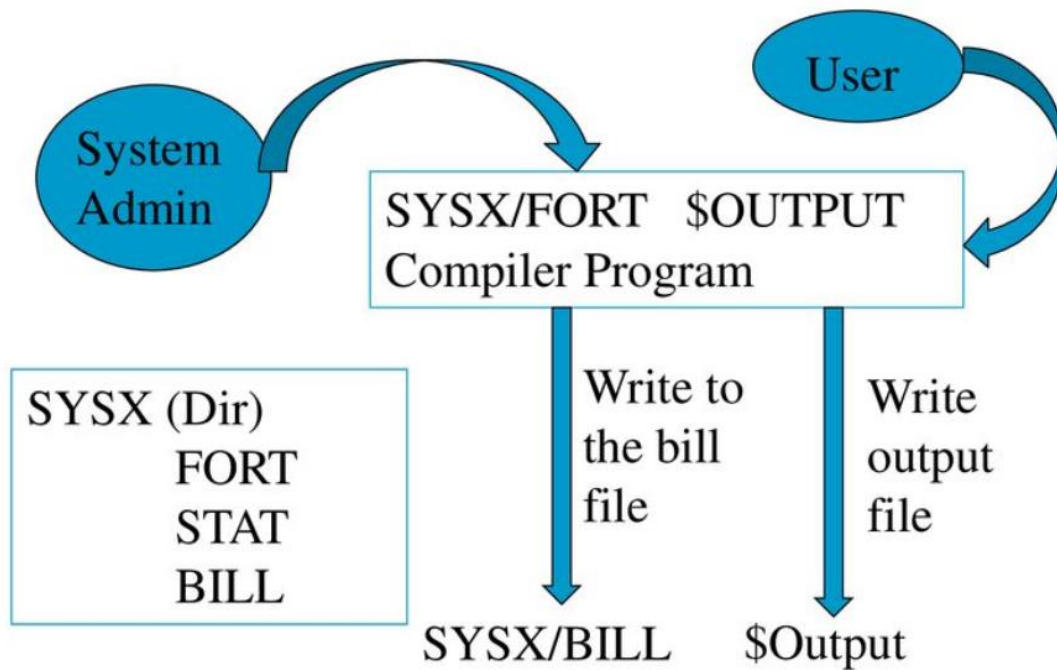
Original example

7

- A program provides compilation services to other programs
- The compiler service is pay-per-use: the compiler service stores billing information a BILL file
- Only the program has access to BILL.

Original example

8



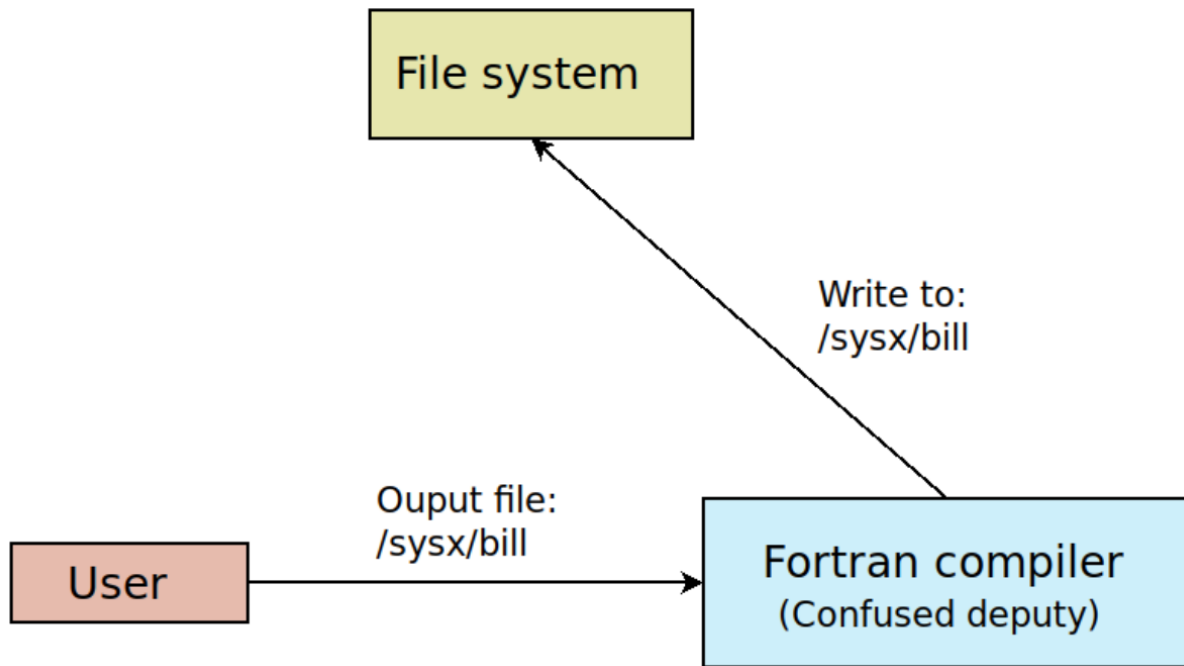
Original example

9

- Clients can compile and set the output file name
- A malicious client calls the output BILL
- Clients cannot open BILL, but the program can
- The program overwrites the BILL file with the compilation output

Original example

10



Protection against Confused Deputy

11

- Use capability based systems
 - Access Control List based systems are ineffective

Outline

12

- Confused Deputy
 - SetUID/SetGID
- Race Conditions
 - TOCTOU Race Conditions

SetUID/SetGID

13

- Open SetUID programs let users run as the owner
- If the program is vulnerable, users can run arbitrary commands as the owner
- The program should only offer the right capabilities
 - E.g., only running predefined commands

Outline

14

- Confused Deputy
 - SetUID/SetGID
- **Race Conditions**
 - TOCTOU Race Conditions

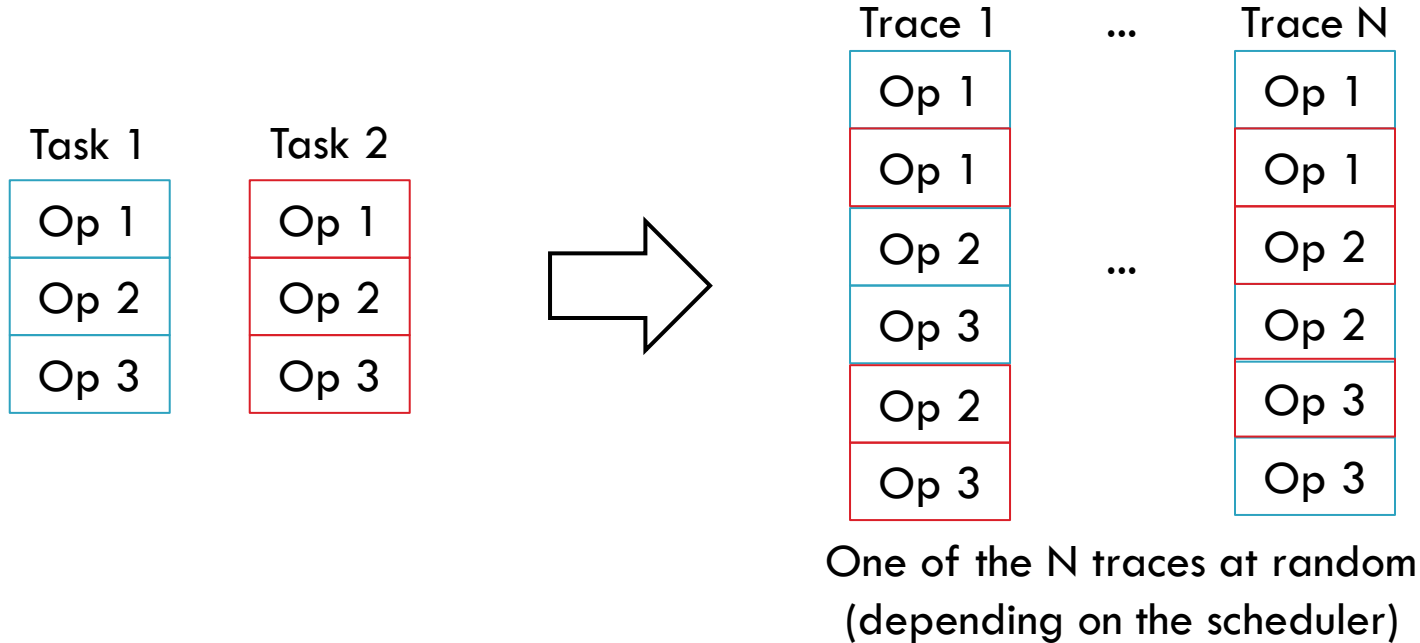
Race Conditions

15

- Concurrency
 - Execution of multiple control flows (e.g., threads)
 - Can lead to non-deterministic behavior
- Race conditions
 - Software can change the intended instruction order
 - E.g., two processes try to write the same file

Race Conditions

16



Properties for race conditions

17

- Concurrency property
 - Two (or more) concurrent control flows
- Shared object property
 - Concurrent control flows access a shared *race object*
- Change state property
 - At least one flow alters the state of the *race object*

Race window

18

- Code segment that accesses the race object in an insecure way
 - Also called *critical section*
- Traditional approach
 - Never overlap critical sections (mutual exclusion)
 - Synchronization primitives (SP)
- Misusing SPs may lead to deadlock

Outline

19

- Confused Deputy
 - SetUID/SetGID
- Race Conditions
 - Time of Check, Time of Use

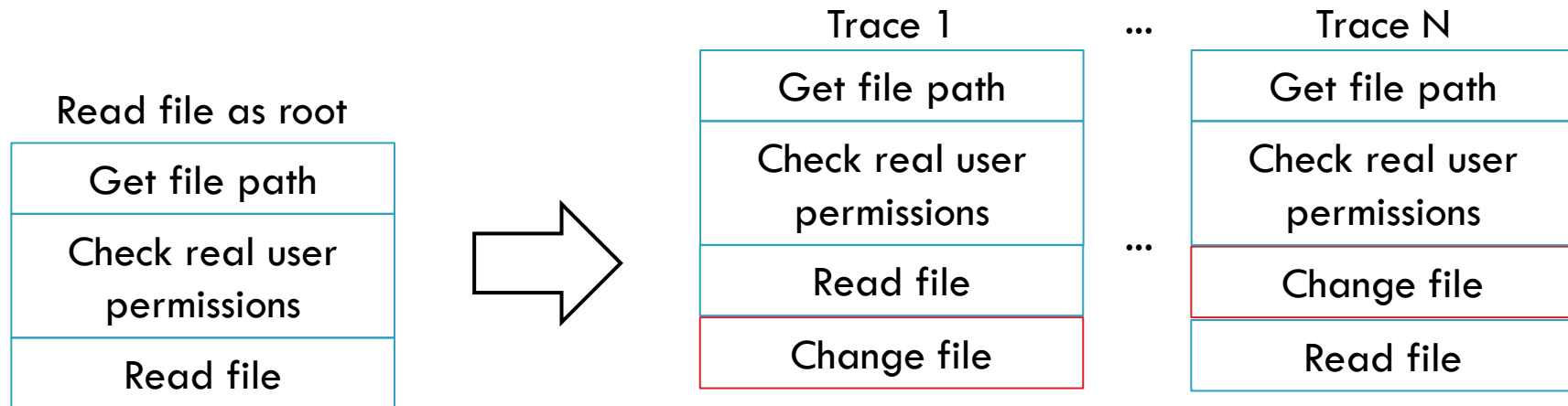
Time of Check, Time of Use

20

- Creates a race window by
 - First *checking* the race object
 - Then *using* the race object
- Concurrent operations can fit between checking and using

Time of Check, Time of Use

21



Run multiple times concurrently to get the trace you want (i.e., the one that triggers the vulnerability)

TOCTOU

22

- How can I change a file? “Get file path” passed already
 - Create a symbolic link to a valid file A
 - Make the program read the symbolic link path
 - Change the symbolic link to point to file B
 - The program returns the content of file B

Enrico RUSSO

Andrea VALENZA

Università di Genova

Weaknesses

