School of
Information Systems

SMU
SINGAPORE MANAGEMENT UNIVERSITY

On the Effectiveness of Software Diversity:

A Systematic Study on Real-World Vulnerabilities

Jin HAN, Debin GAO, and Robert H. DENG

School of Information Systems
Singapore Management University

Outline

- Motivation
- Concrete Research Questions
- Data Source and Preliminary Analysis
- Application Software Vulnerabilities
  - Software substitutes
  - Software on multiple OS
- Conclusion

Outline

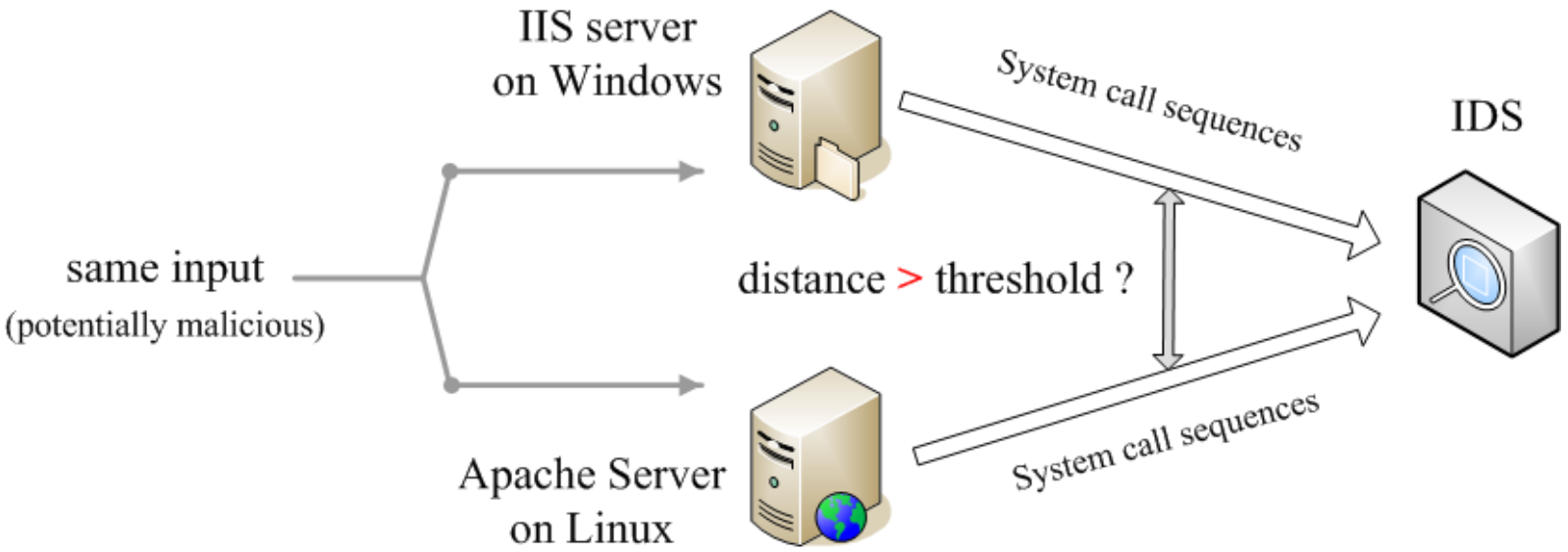
- **Motivation**
- Concrete Research Questions
- Data Source and Preliminary Analysis
- Application Software Vulnerabilities
  - Software substitutes
  - Software on multiple OS
- Conclusion

Software Diversity

- Compared to software mono-culture, software diversity can be adopted at various levels to enhance system security:
  - **System-level**:
    - Instruction-set Randomization [Barrantes(CCS'03), Gaurav(CCS'03)]
    - Address Space Randomization [Bhatkar, USENIX Security '03]
  - **Application-level**:
    - N-version programming [Chen et al., 1978]
    - N-variant systems [Cox et al., USENIX Security '06]
    - Behavioral Distance [Gao et al., RAID '05, RAID '06]

Software Diversity Application

- An example of **Behavioral Distance** [Gao et al., RAID05, RAID06]:



- Another example:
  Utilizing diverse software in network to decrease the virulence of worms and the effectiveness of single attacks to repeated applications. [O'Donnell et al, CCS 04]

The assumption

- These systems which utilize diverse off-the-shelf software usually

  assume that these software products are diverse enough
  not to be compromised simultaneously with the same exploit

- Is such an assumption valid?
- How accurate is this assumption?
- What is the effectiveness of utilizing diverse software in these applications?

Outline

- Motivation
- **Concrete Research Questions**
- Data Source and Preliminary Analysis
- Application Software Vulnerabilities
  - Software substitutes
  - Software on multiple OS
- Conclusion

Two ways for app. diversity

- Different software with same functionalities (Software substitutes)
  - E.g.  vs   Foxit Reader
  - Alternativeto: http://alternativeto.net/desktop
- Same software running on multiple OS

 on   

Research Questions

- Software substitutes:
  - What is the percentage of software that has potential substitutes with the same functionality?
  - For those that are software substitutes of one another, do they have the same vulnerability?
  - Can they be exploited with the same attack?
- Software on multiple OS:
  - How many software products can run on multiple OS?
  - Do vulnerabilities of the software on one OS propagate to the same software on a different OS?
  - If so, can they be exploited by the same attack when running on different OS?

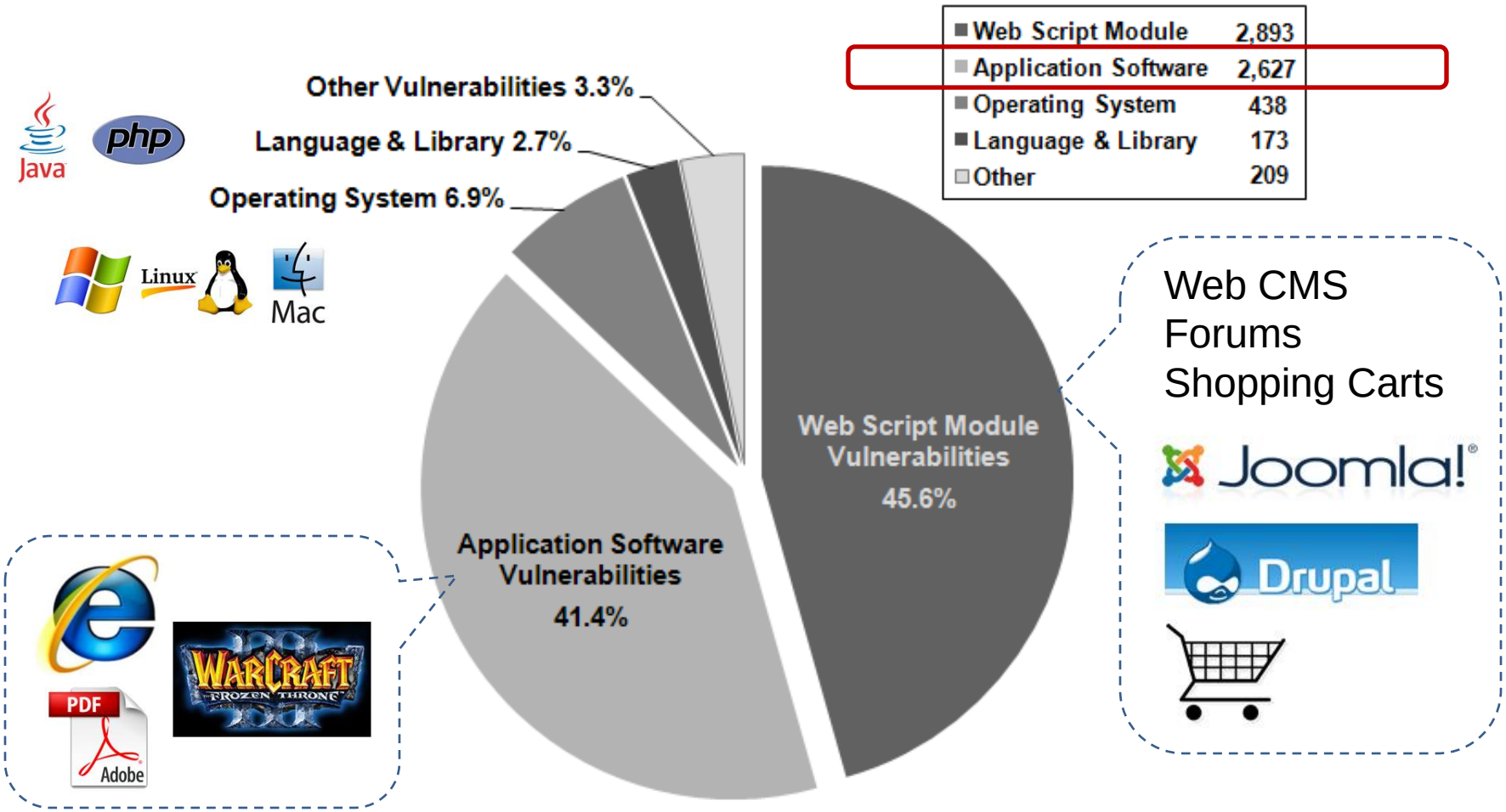
Outline

- Motivation
- Concrete Research Questions
- **Data Source and Preliminary Analysis**
- Application Software Vulnerabilities
  - Software substitutes
  - Software on multiple OS
- Conclusion

Source of Information

- The main source:
  - 6,427 software vulnerabilities in 2007, in NVD/CVE (National Vulnerability Database/Common Vulnerability and Exposures)
- Other sources utilized:
  - SecurityFocus, FrSIRT, CERT, Milw0rm, Secunia, OSVDB, IBM X-Force, and also the bug lists from the software vendors.

Preliminary Analysis



Vulnerabilities in different software categories (2007)

Outline

- Motivation
- Concrete Research Questions
- Data Source and Preliminary Analysis
- **Application Software Vulnerabilities**
  - Software substitutes
  - Software on multiple OS
- Conclusion

Most of app. have substitutes

- 2,627 application software vulnerabilities correspond to 1,825 distinct software products.
- Only 1.4% (25 out of 1,825) don't have substitutes

Vendor	Product	CVE entry
ATI	Display driver	CVE-2007-4315
NVIDIA	Video driver	CVE-2007-3532
Intel	2200BG Wireless driver	CVE-2007-0686
HP	Help and Support Center	CVE-2007-3180
HP	Quick Launch Button	CVE-2007-6331
Alibaba	Alipay ActiveX control	CVE-2007-0827
Microgaming	Download Helper ActiveX	CVE-2007-2177

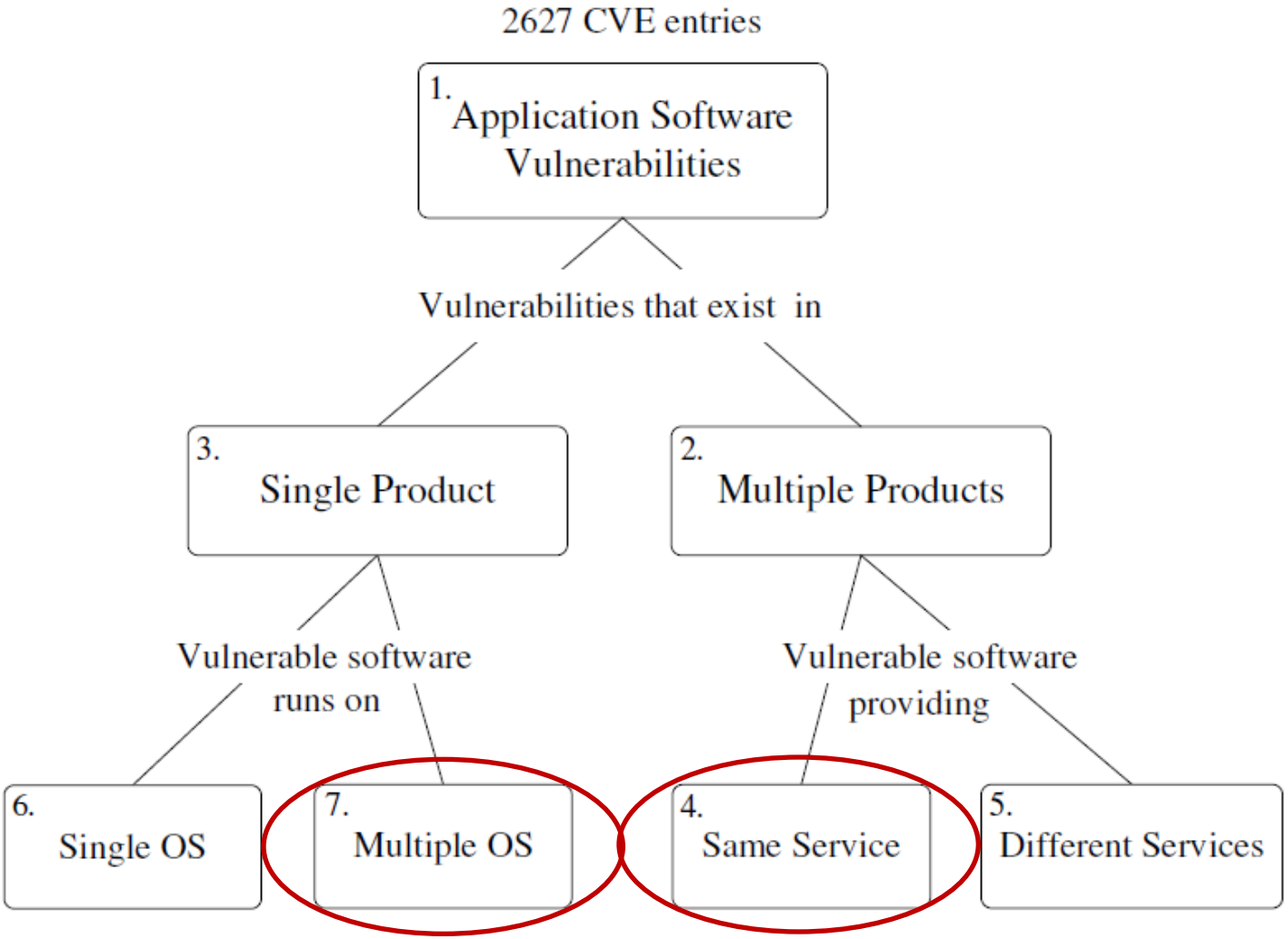
Hardware drivers

Specific software

Specific plug-in

Examples of software products without substitutes

Analysis Tree



Outline

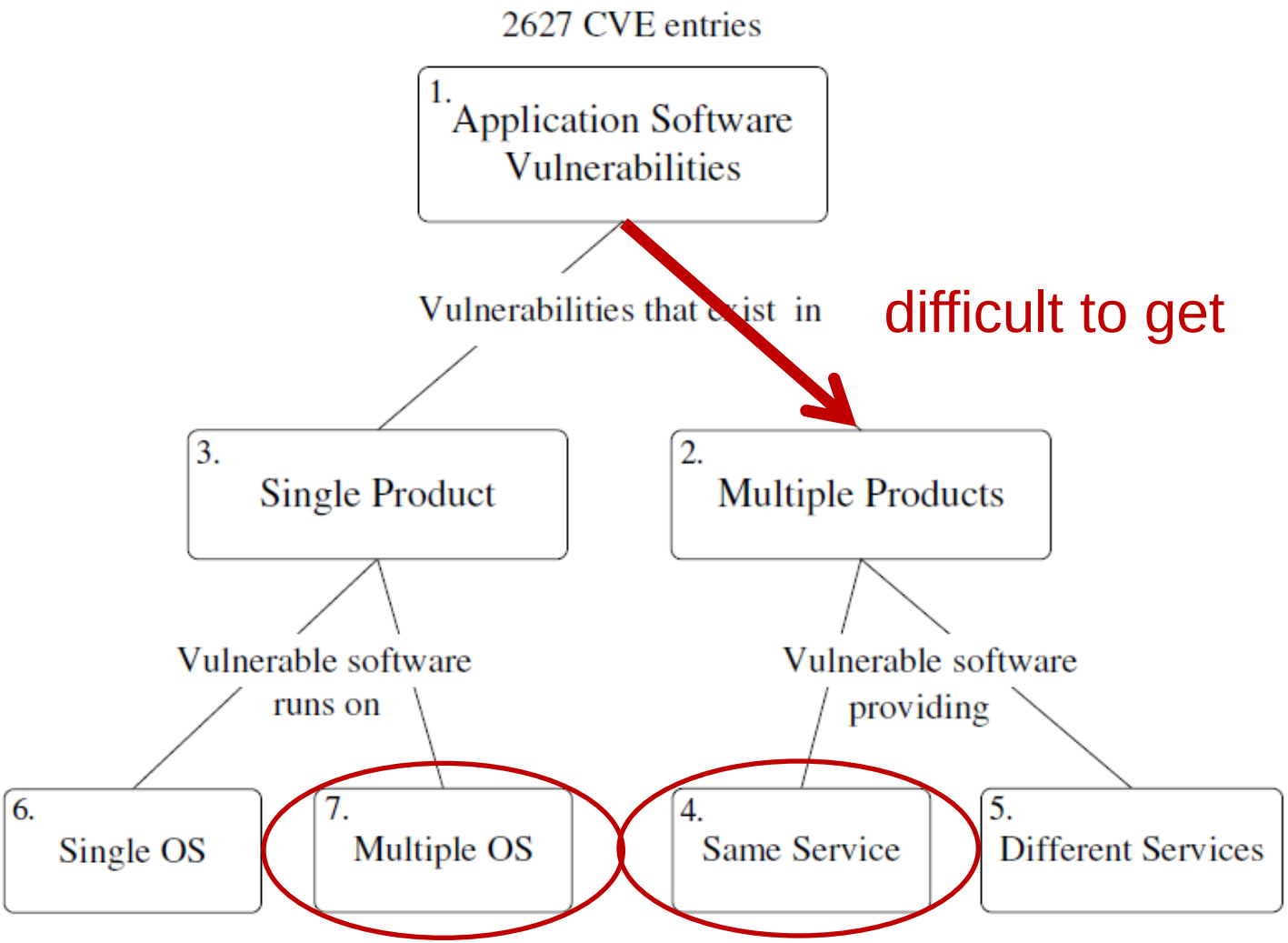
- Motivation
- Concrete Research Questions
- Data Source and Preliminary Analysis
- Application Software Vulnerabilities
  - **Software substitutes**
  - **Software on multiple OS**
- Conclusion

Difficulties in analysis

- An interesting observation is that the same vulnerability may be represented in multiple entries in the CVE database.

CVE Entry	Description
CVE-2007-2761	Stack-based buffer overflow in MagicISO 5.4 build 239 and earlier allows remote attackers to execute arbitrary code via a long filename in a .cue file.
CVE-2007-2888	Stack-based buffer overflow in UltraISO 8.6.2.2011 and earlier allows user-assisted remote attackers to execute arbitrary code via a long FILE string (filename) in a .cue file.

Analysis Tree



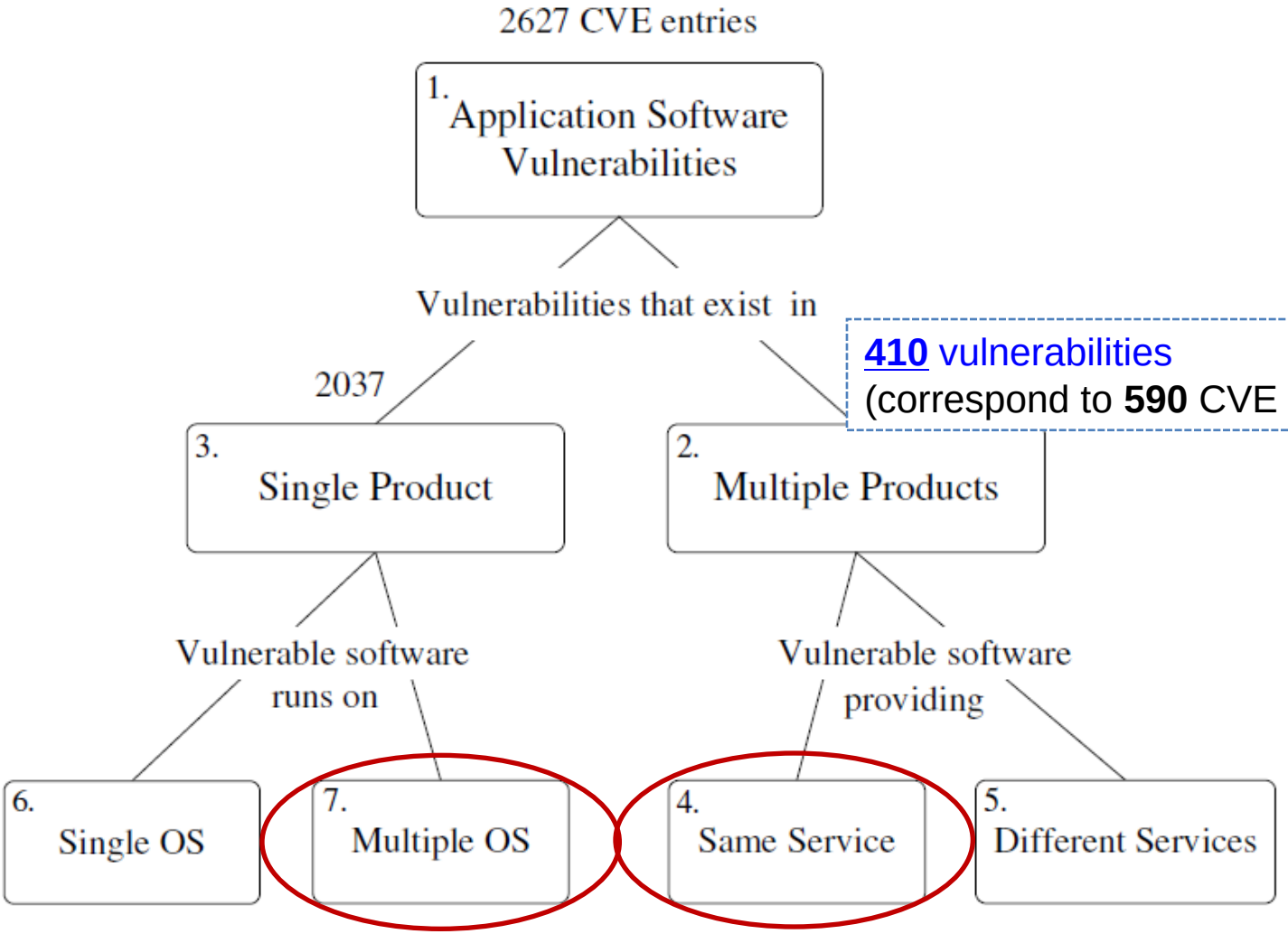
Speed up the analysis

- Different CVE entries that refer to the same vulnerability usually have similar descriptions.
- Vector Space Model

$$\text{sim}(d_1, d_2) = \frac{\vec{d_1} \cdot \vec{d_2}}{|\vec{d_1}| \times |\vec{d_2}|} = \frac{\sum_{i=1}^{t} w_{i,1} \times w_{i,2}}{\sqrt{\sum_{i=1}^{t} w_{i,1}^2} \times \sqrt{\sum_{i=1}^{t} w_{i,2}^2}}$$

- The result: 410 vulnerabilities exist in multiple software products

Analysis Tree



410 vulnerabilities
(correspond to **590** CVE entries)

Vulnerabilities in software substitutes

- 29 out of 410 vulnerabilities exist in **software substitutes**

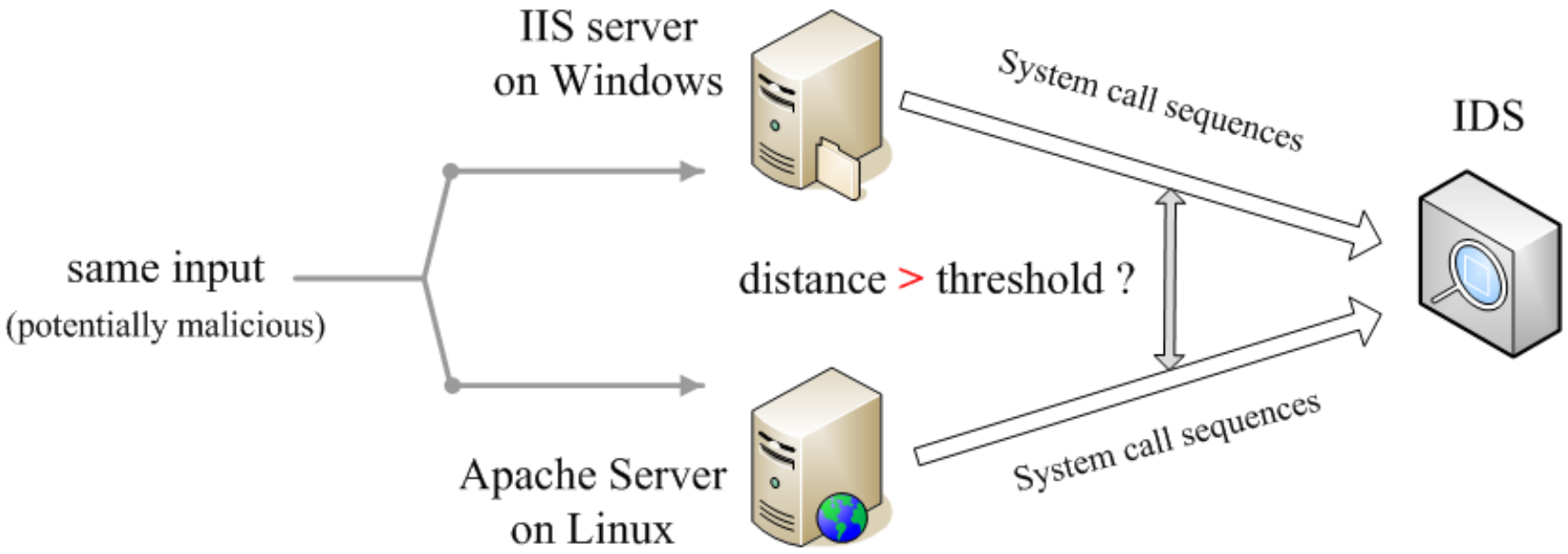
CVE Entry	Description
CVE-2007-0548	KarjaSoft Sami HTTP Server 2.0.1 allows remote attackers to cause a denial of service (daemon hang) via a large number of requests for nonexistent objects.
CVE-2007-3340	BugHunter HTTP Server (httpsv.exe) 1.6.2 allows remote attackers to cause a denial of service (application crash) via a large number of requests for nonexistent pages.
CVE-2007-3398	LiteWEB 2.7 allows remote attackers to cause a denial of service (hang) via a large number of requests for nonexistent pages.

Effectiveness of using software substitutes

- Only 1.4% (25 out of 1,825) of the app. products don't have substitutes
- 16.8% (410 out of 2,447) vulnerabilities exists in multiple software
- 7.1% (29 out of 410) vul. exists in software substitutes
- 70% (14 out of 20), can be exploited with the same code on multiple products
- Approximately, 0.83% (16.8% * 7.1% * 70%) fail to detect

Software Diversity Application

- An example of **Behavioral Distance** [Gao et al., RAID05, RAID06]:

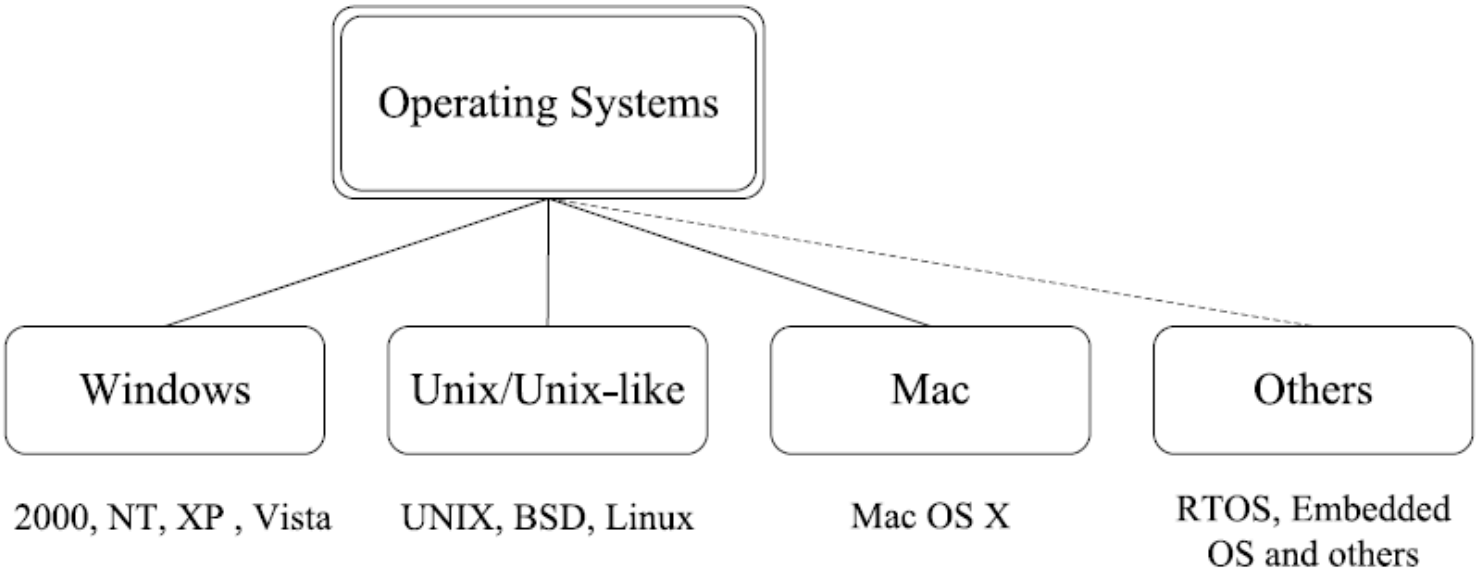


Outline

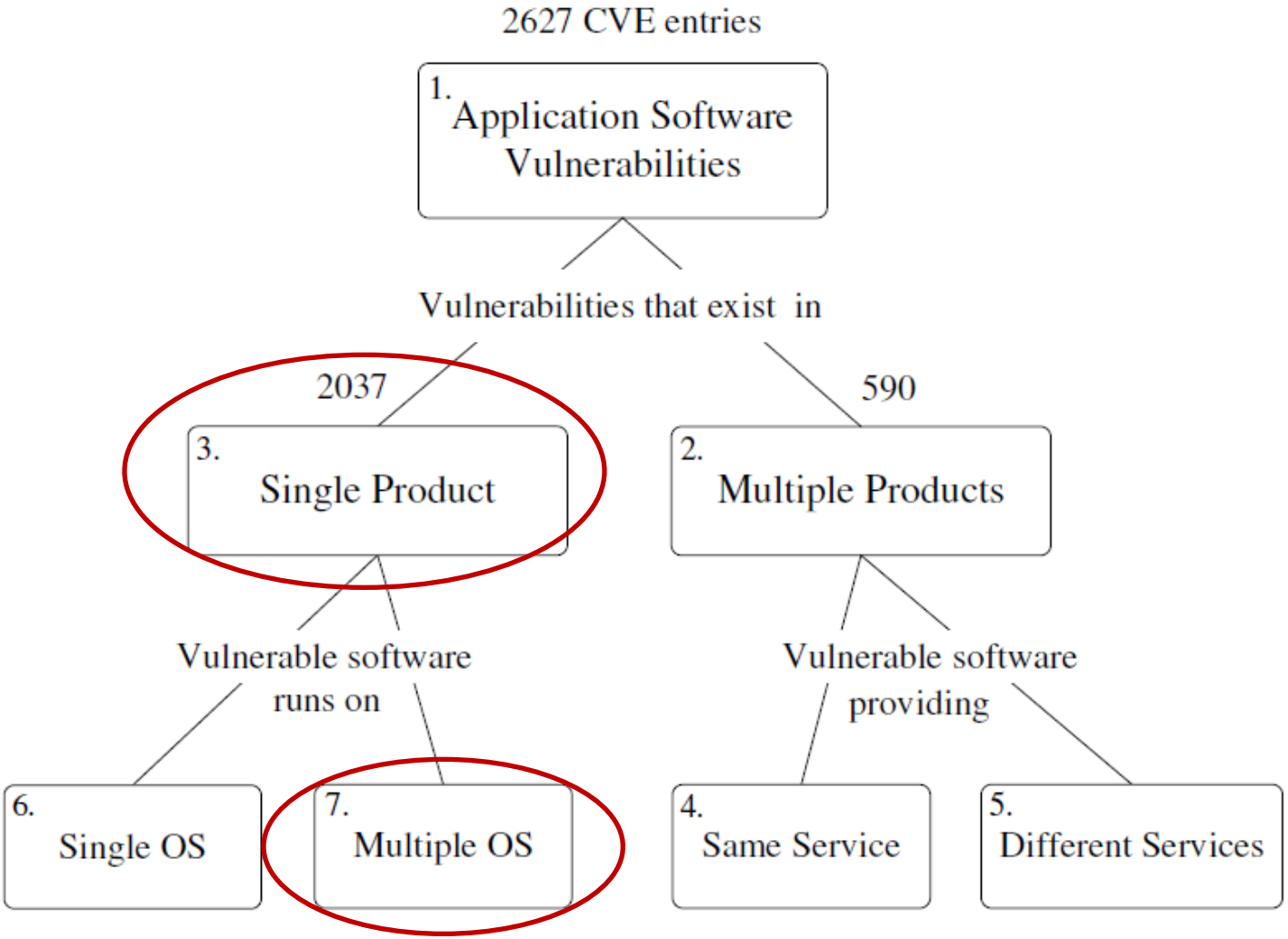
- Motivation
- Concrete Research Questions
- Data Source and Preliminary Analysis
- Application Software Vulnerabilities
  - Software substitutes
  - **Software on multiple OS**
- Conclusion

Different OS

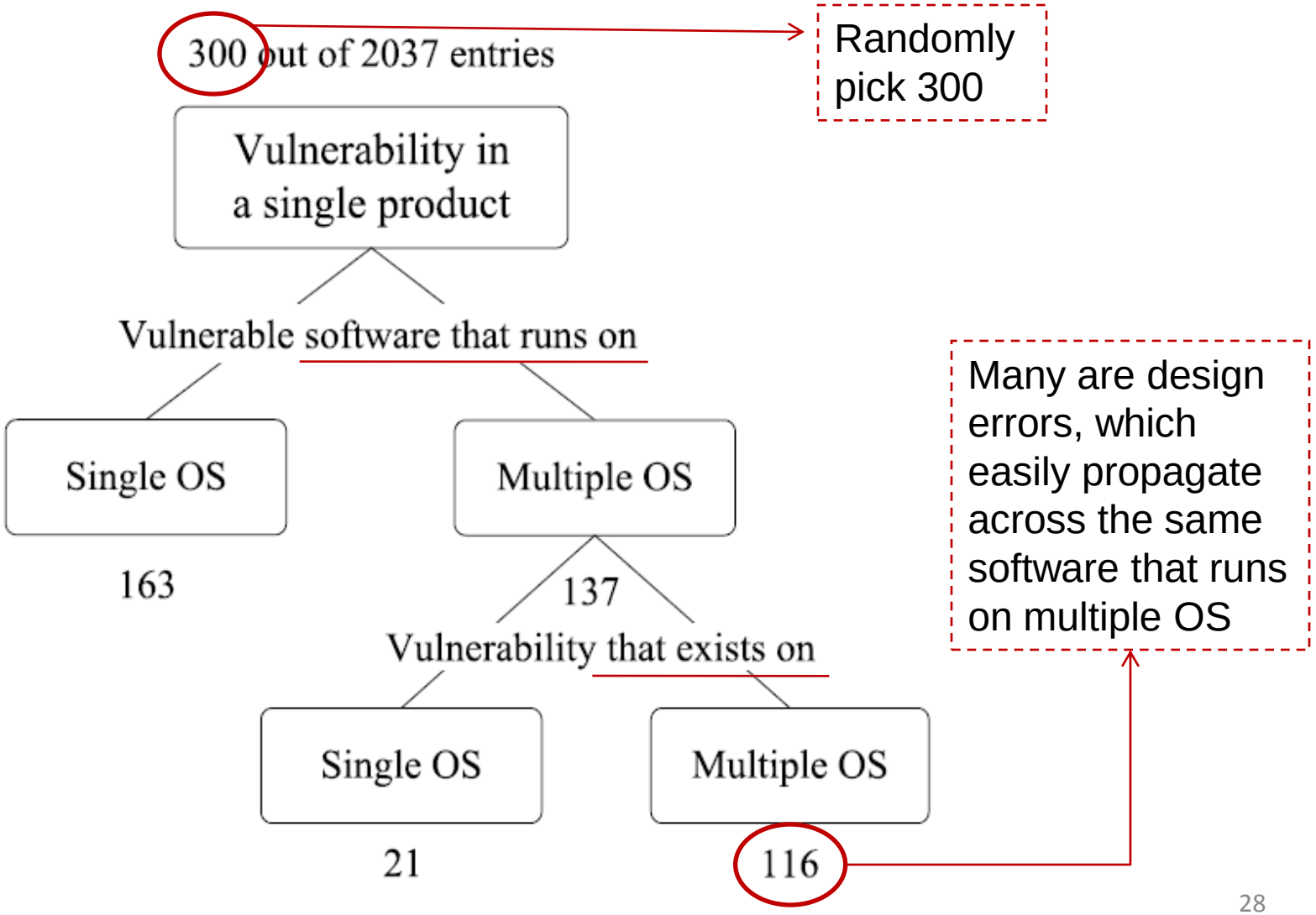
- Different kernels
  - NT for Win; Solaris, BSD, Linux kernel for UNIX-like and XNU for Mac OS X
- Different binary executable formats:
  - PE for Win, ELF for UNIX and Mach-O for Mac OS X



Analysis Tree



Software on Multiple OS



To exploit software that runs on different OS

- 84.7% (116 out of 137) of the vulnerability exist in the same software on different OS
  - Does this mean it is not effective to utilize software on different OS?
- Exploit code is unlikely to be the same:
  - Same software on different OS, usually have different source code
  - Even if source code is the same, attack codes are different due to different API and system calls across different OS

Outline

- Motivation
- Concrete Research Questions
- Data Source and Preliminary Analysis
- Application Software Vulnerabilities
  - Software substitutes
  - Software on multiple OS
- **Conclusion**

Conclusion

- Analyzed the vulnerabilities published in 2007 and corresponding software
- Two ways of introducing software diversity utilizing off-the-shelf software:
  - Software substitutes & Software on multi-OS
- The results show:
  - more than 98.5% have substitutes, the chance to be compromised by the same attack is very low.
  - Nearly half of the application software are officially supported to run on multi OS, their exploit code is quite different.

Contents not covered

- Vulnerabilities in other software categories
  - Web script modules
  - Operating systems
  - languages and libraries

Vulnerability Types	Number of entries	Percentage
Cross-site scripting	714	24.7%
SQL injection	669	23.1%
PHP remote file inclusion	634	21.9%
Directory/Path traversal	267	9.2%
Cross-site request forgery	50	1.7%
Others	559	19.3%
Total	2893	100%

Vulnerabilities in web script modules

Q & A

Thanks

Additional Slide 1

- The example for different exploit code for the same vulnerability (6 of the 20 vulnerabilities).
  - CVE-2007-4734 OTSTurntables 1_00 (m3u File) local buffer overflow exploit
  - CVE-2007-4735 Virtual DJ 5_0 (m3u File) Local buffer overflow exploit

Potential attack strategy to evade IDS

- Algorithm to evade detection:

```
os_ret ← os_test();
if is_win(os_ret) then
    win_attack_code();
else if is_unix(os_ret) then
    unix_attack_code();
else if is_mac(os_ret) then
    mac_attack_code();
end if
```

- Different from OS fingerprinting
- Two difficulties to implement

1. Speak slowly
2. Use laser pointer
3. Admit to the face, try to limit the scope of its effect