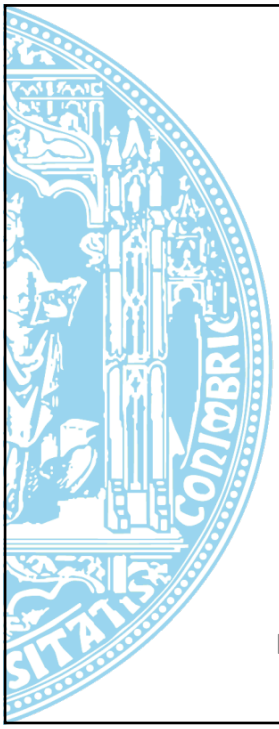


PERSPECTIVES ON DEPENDABILITY AND SECURITY BENCHMARKING OR TO BENCHMARK OR NOT TO BENCHMARK

IFIP WG 10.4
71st Meeting
Queenstown, New Zealand
Jan. 27th, 2017

Marco Vieira
mvieira@dei.uc.pt
Department of Informatics Engineering
University of Coimbra - Portugal

BENCHMARKING CONTEXT

Assessing and comparing computer systems and/or components according to specific quality attributes

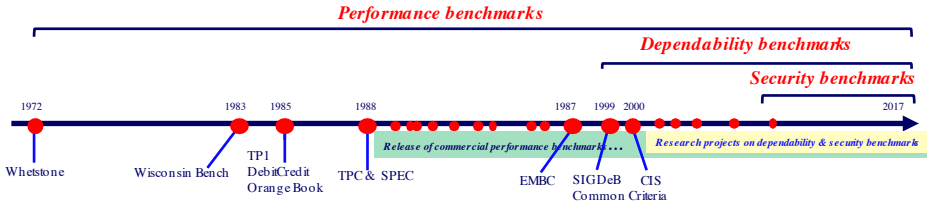
- Performance benchmarking
  - Well established both in terms of research and application
  - Supported by organizations like TPC and SPEC
  - Mostly for marketing
- Dependability benchmarking
  - Well established from a research perspective
  - No endorsement from the industry

BENCHMARKING CONTEXT

Assessing and comparing computer systems and/or components according to specific quality attributes

- Security benchmarking
  - Several works can be found
  - No common approach available yet
- Towards the future:
  - Resilience benchmarking
  - Trustworthiness benchmarking

BENCHMARKS TIMELINE



OUTLINE

- Past perspective
  - Performance, dependability, security...
- What is wrong?
  - and an approach...
- Future perspective... or not...
  - Resilience, trustworthiness...
  - Industry/users acceptance

PERFORMANCE BENCHMARKING

Assessing and comparing computer systems and/or components in terms of performance

PERFORMANCE BENCHMARKING

- Workload:
  - Set of representative operations
- Metrics:
  - Throughput
  - Response time
  - Latency
  - …

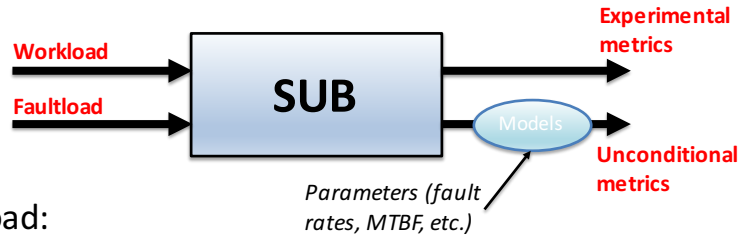
TPC-C (1992)

- Workload:
  - Database transactions
- *Although some integrity tests are performed, it assumes that nothing fails*
  - Transaction rate (tpmC)
  - Price per transaction ($/tpmC)

DEPENDABILITY BENCHMARKING

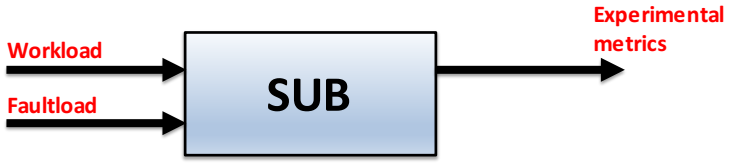
Assessing and comparing computer systems and/or components considering dependability attributes

DEPENDABILITY BENCHMARKING



- Faultload:
  - Set of representative faults, injected into the system
- Metrics:
  - Performance and/or dependability
    - Both baseline and in the presence of faults
  - Unconditional and/or direct

DBENCH-OLTP (2005)



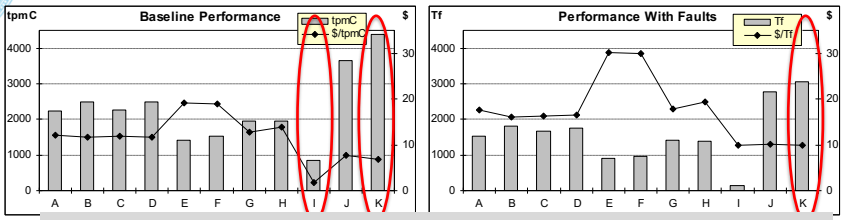
- Workload:
  - TPC-C transactions
- Faultload:
  - Operator faults + Software faults + HW component failures
- Metrics:
  - Performance: tpmC, $/tpmC, Tf, $/Tf
  - Dependability: Ne, AvtS, AvtC

DBENCH-OLTP (2005)

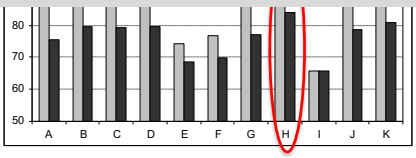
System	Operating System	DBMS	DBMS Config.	Hardware
A	Windows 2K Prof. SP 3	Oracle 8i R2 (8.1.7)	Config. A	*Processor*: Intel Pentium III 800 MHz *Memory*: 256MB *Hard Disks*: Four 20GB/7200 rpm *Network*: Fast Ethernet
B	Windows 2K Prof. SP 3	Oracle 9i R2 (9.0.2)	Config. A	
C	Windows Xp Prof. SP 1	Oracle 8i R2 (8.1.7)	Config. A	
D	Windows Xp Prof. SP 1	Oracle 9i R2 (9.0.2)	Config. A	
E	Windows 2K Prof. SP 3	Oracle 8i R2 (8.1.7)	Config. B	
F	Windows 2K Prof. SP 3	Oracle 9i R2 (9.0.2)	Config. B	
G	SuSE Linux 7.3	Oracle 8i R2 (8.1.7)	Config. A	
H	SuSE Linux 7.3	Oracle 9i R2 (9.0.2)	Config. A	
I	SuSE Linux 7.3	PostgreSQL 7.3	-	
J	Windows 2K Prof. SP 3	Oracle 8i R2 (8.1.7)	Config. A	*Processor*: Intel Pentium IV 2GHz *Memory*: 512MB *Hard Disks*: Four 20GB/7200 rpm *Network*: Fast Ethernet
K	Windows 2K Prof. SP 3	Oracle 9i R2 (9.0.2)	Config. A	

Faultload: Operator faults

DBENCH-OLTP (2005)



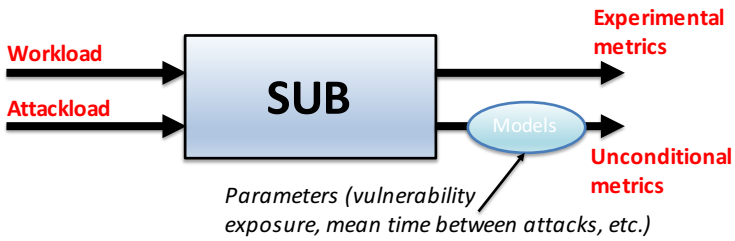
Does not take into account malicious behaviors (faults = vulnerability + attack)



SECURITY BENCHMARKING

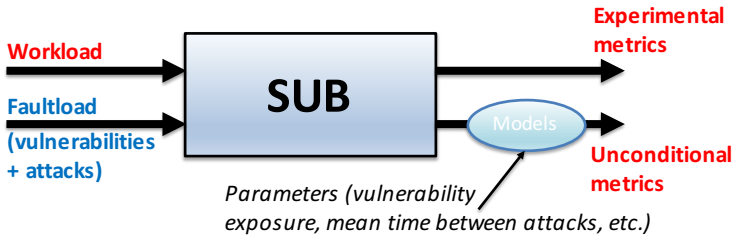
Assessing and comparing computer systems and/or components considering security aspects

SECURITY BENCHMARKING



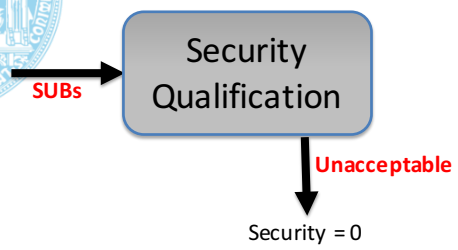
- *Attacking what? Do we know the vulnerabilities? What are representative attacks?*
- Metrics:
  - Performance + dependability ?
  - Security (e.g., number vulnerabilities, attack detection)

SECURITY BENCHMARKING



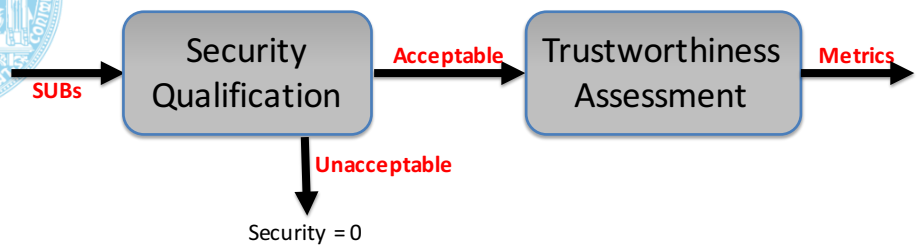
- *Does not work if one wants to benchmark how secure different systems are!*
- *e.g. does the number of vulnerabilities of a system represent anything?*
  - e.g. intrusion detectors

A DIFFERENT APPROACH...



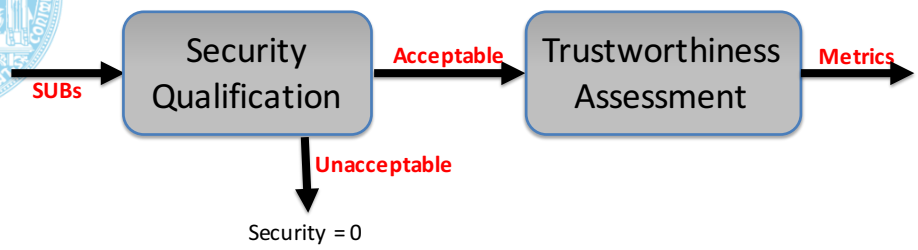
- Security Qualification:
  - Apply state-of-the-art techniques and tools to detect vulnerabilities
  - SUBs with vulnerabilities are:
    - Disqualified!
    - Or vulnerabilities are fixed...

A DIFFERENT APPROACH...



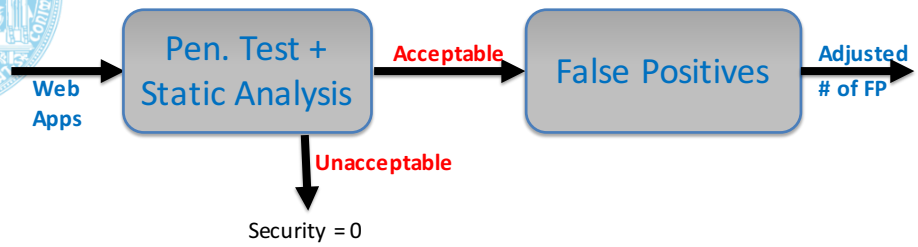
- Trustworthiness Assessment:
  - Gather evidences on how much one can trust
  - e.g., best coding practices, development process, bad smells

A DIFFERENT APPROACH…



- Metrics:
  - Portray trust from a human perspective
  - Dynamic: may change over time
  - Depend on the type of evidences gathered
  - Different metrics for different attack vectors

SBENCH-WEB (2012)



- Security Qualification:
  - Penetration testing + static analysis
- Trustworthiness Assessment:
  - Multiple static analysis tools

SBENCH-WEB (2012)

Metrics:

$$TM = \frac{\text{No. Lines of Code}/100}{F*0.93 + Y*0.64 + I*0.33}$$

F: Findbugs; Y: Yasca; I:IntelliJ Idea

- Example:

		Lines of	Trustworthiness
4	Yazd 3	56255	17,7
5	JavaBB 0.99	23807	10,2
6	mvnForum 1.2	76774	10,2
7	JSForum 0.0.2	1693	0,4

False positives are not a good evidence!

Explore behavioral aspects (e.g. memory and CPU)

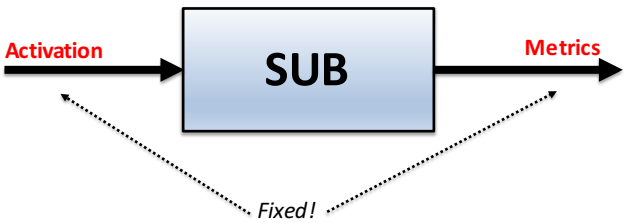
WHAT IS WRONG?

Established benchmarks are mostly for marketing!

- Strict benchmarking conditions
  - Fixed workload & faultload + Small set of metrics
- Workload & faultload:
  - May not be representative of the user scenario
- Metrics:
  - Fixed! May not satisfy the user needs
  - Decision based on several metrics is difficult!

No dependability or security benchmark endorsed by any organization or industry

FIXED!

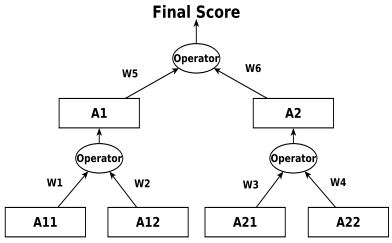


- Example:
  - Benchmarking vulnerability detection tools
  - Typical metric: F-Measure
  - Is this good in all scenarios?
    - Business critical: recall
    - Best effort: F-Measure
    - Minimum effort: Markedness

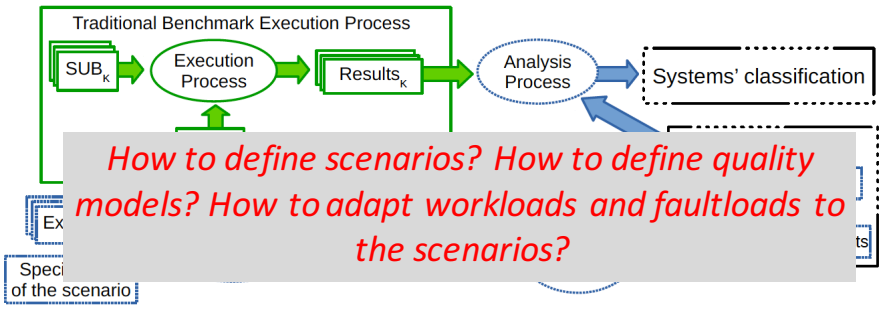
A POTENTIAL APPROACH...

Benchmarking conditions adaptable to the user needs

- Include multiple usage scenarios:
  - Metrics depend on the scenario
  - Adaptable workload and faultload
- Use quality models instead of independent metrics
  - Quality models should also adapt to the scenario



SCENARIOS AND QUALITY MODELS



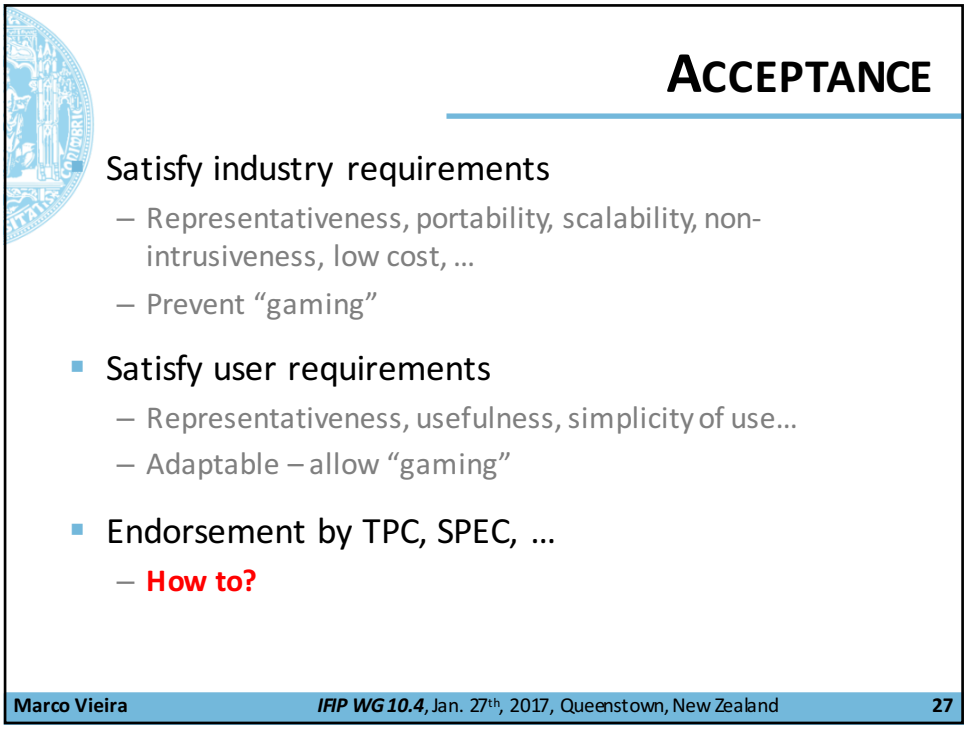
How to define scenarios? How to define quality models? How to adapt workloads and faultloads to the scenarios?

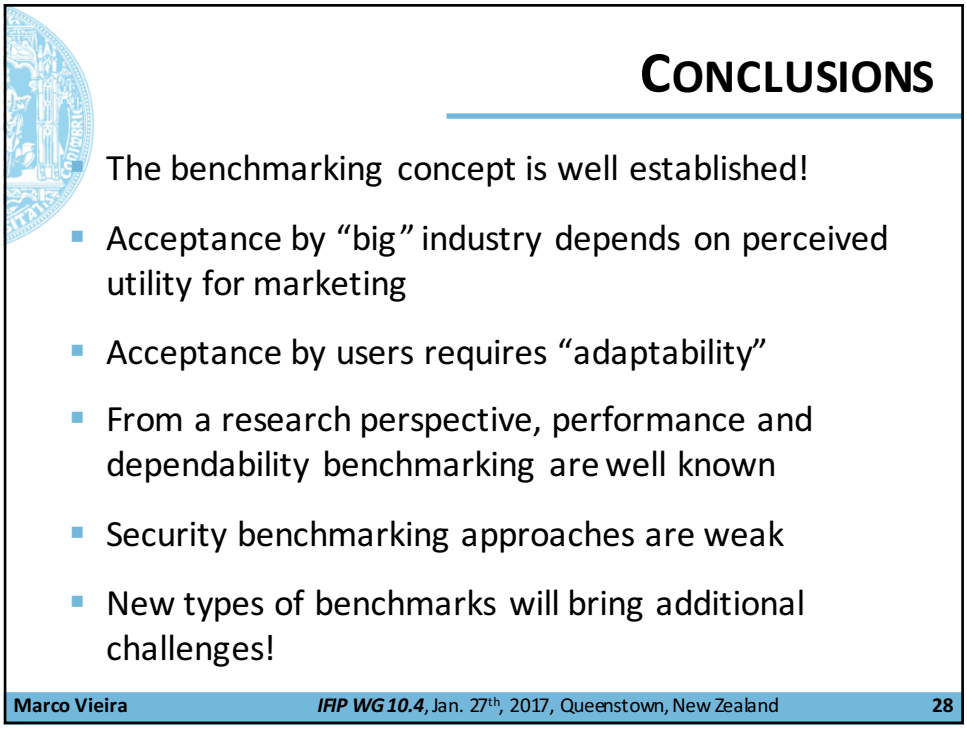
IS THERE A FUTURE?

Resilience Benchmarking

- Assess and compare the behavior of components and computer systems when subjected to changes
- Which resilience metrics?
  - Comparable, consistent, understandable, meaningful, …
- Changeloads:
  - Representative, practical, portable, …

- Trustworthiness Benchmarking
  - What evidences to collect?
  - What metrics?
  - Dynamicity of perception… social trust…


ACCEPTANCE
Satisfy industry requirements
– Representativeness, portability, scalability, non-intrusiveness, low cost, …
– Prevent "gaming"
Satisfy user requirements
– Representativeness, usefulness, simplicity of use…
– Adaptable – allow "gaming"
Endorsement by TPC, SPEC, …
– How to?
Marco Vieira
IFIP WG 10.4, Jan. 27th, 2017, Queenstown, New Zealand
27


CONCLUSIONS
The benchmarking concept is well established!
Acceptance by "big" industry depends on perceived utility for marketing
Acceptance by users requires "adaptability"
From a research perspective, performance and dependability benchmarking are well known
Security benchmarking approaches are weak
New types of benchmarks will bring additional challenges!
Marco Vieira
IFIP WG 10.4, Jan. 27th, 2017, Queenstown, New Zealand
28

QUESTIONS?

Marco Vieira
Department of Informatics Engineering
University of Coimbra
mvieira@dei.uc.pt
http://eden.dei.uc.pt/~mvieira