

Security in the Context of Dependability

Tacksoo Im
John D. McGregor
School of Computing
Clemson University
ORNL Presentation

1

Security in Context

- ❑ Security is not a directly measurable quantity.
- ❑ The level of security is usually described in terms of levels of availability, integrity and confidentiality.
- ❑ The level of security is used to help define the level of dependability.

2

Security in Software Architecture

- ❑ Security should be considered from the earliest point of the development process.
- ❑ Desired level of security is stated as a non-functional requirement.
- ❑ Trade-off between security and other non-functional qualities should be considered.
- ❑ How does an architectural design decision (tactic) influence security and other qualities?

3

What is an Architectural Tactic?

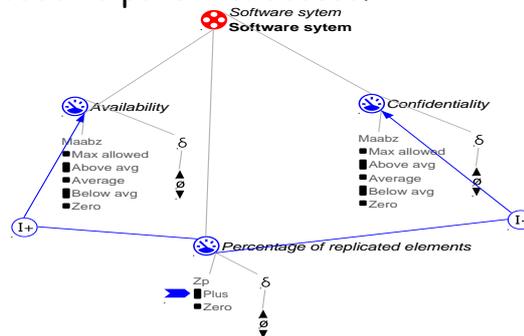
- ❑ An architectural tactic is a design decision that results in a desired change in the quality under consideration.
- ❑ Some examples are reducing the computational overhead and hiding information.
- ❑ Architectural tactics often influence one or more qualities.

4

Security Tradeoff 1

Availability vs. Confidentiality

- When availability is increased, confidentiality decreases because of prolonged exposure.
- The longer the system is available, the longer it is exposed to potential access.

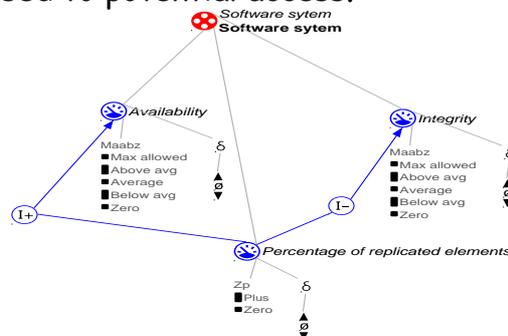


5

Security Tradeoff 2

Availability vs. Integrity

- When availability is increased, integrity decreases because of prolonged exposure.
- The longer the system is available, the longer it is exposed to potential access.

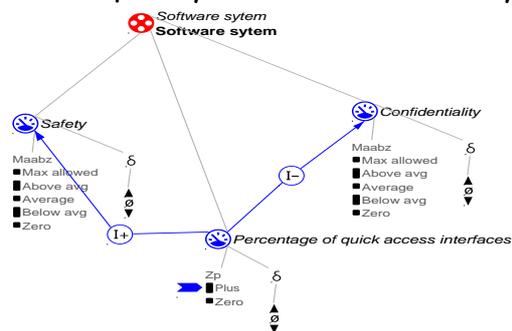


6

Security Tradeoff 3

□ Safety vs. Confidentiality

- In some cases, safety has to be relaxed to increase confidentiality and vice versa.
- Disabling confidentiality measures to ensure the data can be quickly accessed for safety.

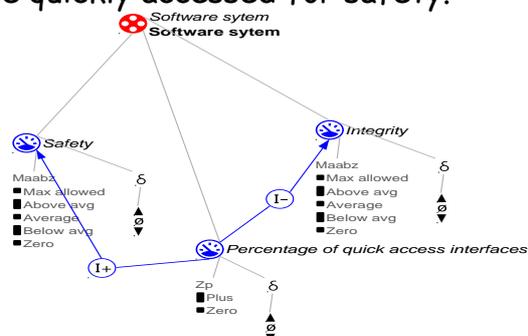


7

Security Tradeoff 4

□ Safety vs. Integrity

- In some cases, safety has to be relaxed to increase integrity and vice versa.
- Disabling integrity measures to ensure the data can be quickly accessed for safety.



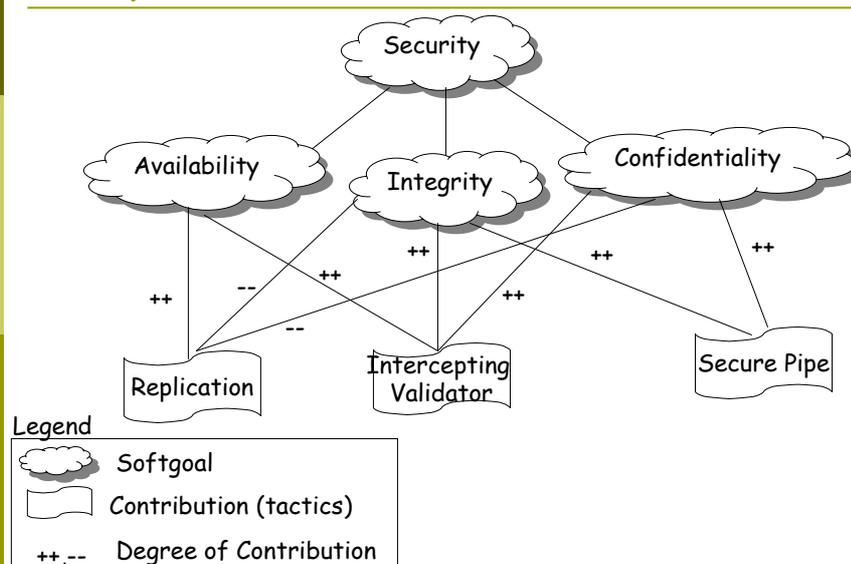
8

Security as a Non-functional Requirement

- Non-functional requirements such as Security can be seen as a *softgoal*.
- Softgoals do not have a clear definition and a criteria for satisfaction.
- A softgoal interdependency graph captures the interdependencies of softgoals.
- Softgoals are *satisfied*. (achieved within satisfactory boundaries)
- An architectural tactic can contribute positively, negatively, fully or partially to satisficing softgoals.

9

A Softgoal Interdependency Graph



10

What is Qualitative Reasoning?

- Qualitative Reasoning is reasoning with imprecise data.
- Often used to model tacit (implicit) knowledge.
- **Influences** model processes that cause changes within a model.
- **Proportionalities** propagate the effects of a process.
- **Model Fragments** describe the structure and behavior of the system in a general way.

11

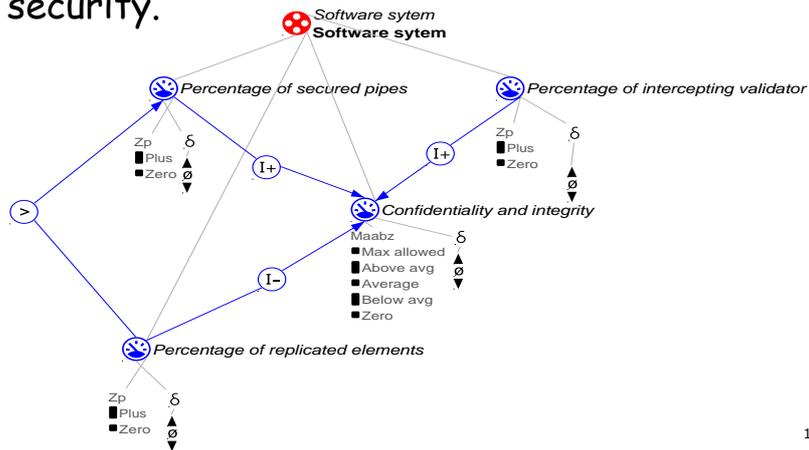
Qualitative Reasoning about Security

- Softgoals can be mapped to a qualitative scale. (i.e. Max, Exceeds goal, meets goal, does not meet goal, Min)
- QR can be used to determine if a softgoal is satisfied.
- Positive, negative, full and partial contributions to the softgoal can be seen as *influences*.

12

Qualitative Reasoning about Security : An Example

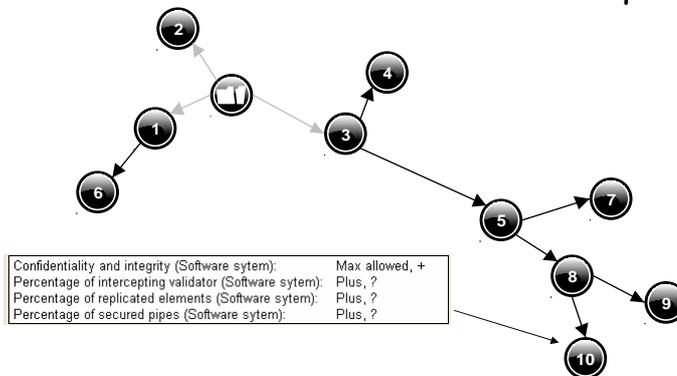
- Model Fragment from a QR model for security.



13

Qualitative Reasoning about Security : The Results

- Garp3 (a workbench for QR modeling) gives the result of the application of the tactics.
- The circles shows the state of the qualities.



14

Satisficing Security Requirements

- Combining Garp3 model fragments will help us reason about the result of applying a set of tactics.
- How does a change in security influence overall system dependability?

15

Conclusion

- Qualitative Reasoning can be used to find out the overall effects of an architectural tactic on software security.
- QR model of security can be a part of model that covers dependability.

16