Analyzing Tradeoffs between Software Security and Performance

PhD student
Catia Trubiani
catia.trubiani@univaq.it

Advisor
Vittorio Cortellessa
vittorio.cortellessa@univaq.it

Project PRIN PaCo (Performability-aware Computing)
Camerino, 15th September 2010

Outline

» Motivation

» Our approach
  - Security Library
  - Enabling Security

» Experimental validation

» Conclusions

V.Cortellessa, C.Trubiani, L.Mostarda, N.Dulay
@ISARCS - International Symposium on ARchitecting Critical Systems, CompArch 2010
Motivation

» Trade-off analysis for critical systems

Problem statement

» How much the security solutions degrade performance?
Our approach

A process for the analysis of security/performance tradeoffs

A vision of our Security Library

Dependencies between Mechanisms
Security Library (1/3)

» Some preliminary operations

- setKeyType()
- setKeyLength()
- generatePKey(type, length)
- generateSKey(type, length)

"generateKeys"

- reqCertificate(compInfo, P(C))
- sendCertificate(compCertificate)

- opt
- [trusted]

Security Library (2/3)

» Basic Mechanisms

- alt
  - [asymEncrypt]
    - setAlgorithmType()
    - setAlgorithmMode()
    - encryptAlg(msg, P(R))
    - encryptAlg(msg, K(K(S), K(R)))
  - [symEncrypt]
    - setAlgorithmType()
    - setPaddingScheme()
    - encryptAlg(msg, P(R))

Digital Signature

- setHashFunction()
- generateDigest()
- encryptAlg(digest, S(C))
Security Library (3/3)

» Composed Mechanisms

Our approach at work!

» More details of the approach by means of a driving case study, i.e. the CUSPIS system:

- Cultural asset authentication (CAA)
  "GeoDataGeneration" scenario
- Cultural asset transportation (CAT)
**Application Model**

» The Application Model is a static and dynamic representation of a software architecture

```java
<<component>>
Qualified Organization
geoData: gdType
gdType genGeoData () (return this.geoData);

void store(gdType geoDataName)

<<component>>
Database
```

**Security-Annotated App.Model(s)**

» A Security-Annotated model is obtained by introducing security annotations

```java
<<component>>
Qualified Organization
geoData: gdType
gdType genGeoData () (return this.geoData);

void store(gdType geoDataName)

Data Origin Authentication

<<component>>
Database
```

* System Configuration $SC_1$, i.e. the required security settings (e.g. Data Origin Authentication)
A Security-Annotated model is obtained by introducing security annotations. A Security-Annotated model is obtained by introducing security annotations.

Enabling Security

Operational steps:
1. Interpretation of security annotations
   - **Key-aspect**: composability of models
     (i) Entry points unambiguously defined
     (ii) Security models easily composable

2. Evaluation of security mechanisms at the application level
   - **Key-aspect**: application-independent parameters are specified in the Security Library
     (i) Implementation options unambiguously defined
     (ii) Estimation of application-dependent Security Mechanisms
A Security-Enabled model is obtained by embedding the appropriate security mechanisms.
A Performance model is obtained by transforming a software model into a performance model.

**System Configuration SC₁**

```
setHashFunction()
{HashAlgorithm={SHAwithRSA, MD5, ...},
KeySize={1024, 2048, 4096, ...}}
```

**System Configuration SC₂**

```
setAlgorithmType()
{Algorithm={AES, 3DES, RSA, ...},
Mode={CBC, ECB, ...}}

setKeyLength()
{KeyLength={256, 512, ...}}
```
Validation of the case study

» Experimental results (1/2)

System Configuration $SC_1$

<table>
<thead>
<tr>
<th>KeySize (byte)</th>
<th>Model Solution Results (tags/sec)</th>
<th>Implementation Monitoring Data (tags/sec)</th>
<th>Model Prediction Error (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Platform 1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2048</td>
<td>9.92</td>
<td>9.11</td>
<td>2.85</td>
</tr>
<tr>
<td>4096</td>
<td>1.98</td>
<td>1.92</td>
<td>3.83</td>
</tr>
<tr>
<td>Platform 2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2048</td>
<td>8.45</td>
<td>8.11</td>
<td>4.02</td>
</tr>
<tr>
<td>4096</td>
<td>1.85</td>
<td>1.78</td>
<td>3.78</td>
</tr>
</tbody>
</table>

Platform 1 - Intel(R) Core2, 2.0GHz with 2GB RAM, Windows Vista
Platform 2 - Intel Pentium4, 3.4GHz with 2GB RAM, Windows XP

Validation of the case study

» Experimental results (2/2)

System Configuration $SC_2$

<table>
<thead>
<tr>
<th>KeySize (byte)</th>
<th>Model Solution Results (tags/sec)</th>
<th>Implementation Monitoring Data (tags/sec)</th>
<th>Model Prediction Error (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Platform 1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2048</td>
<td>3.43</td>
<td>3.29</td>
<td>4.08</td>
</tr>
<tr>
<td>4096</td>
<td>1.35</td>
<td>1.33</td>
<td>1.48</td>
</tr>
<tr>
<td>Platform 2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2048</td>
<td>3.33</td>
<td>3.2</td>
<td>3.90</td>
</tr>
<tr>
<td>4096</td>
<td>1.38</td>
<td>1.34</td>
<td>2.90</td>
</tr>
</tbody>
</table>

Platform 1 - Intel(R) Core2, 2.0GHz with 2GB RAM, Windows Vista
Platform 2 - Intel Pentium4, 3.4GHz with 2GB RAM, Windows XP
A broader analysis

» What happens while varying the system workload across the SC₁ and SC₂ configurations?

Afterthoughts

» Experimentation:
- Our models provide promising results (i.e. the worst model prediction error is 4.08%)
- The analysis of the workload provides interesting insights

» Limitations:
- Security Mechanisms: encryption and digital signature
- Enabling security implies the usage of the mechanisms at the application level, thus they can be influenced by application-dependent characteristics
Conclusions

» Contributions:
- A framework to support the analysis of software architecture (i.e., performance degradation while varying security solutions)
- Introduction of models for basic security mechanisms

» Future works:
- Introduction of costs for security solutions
- Trade-off analysis between security and other non-functional attributes, e.g. availability

Thank you!