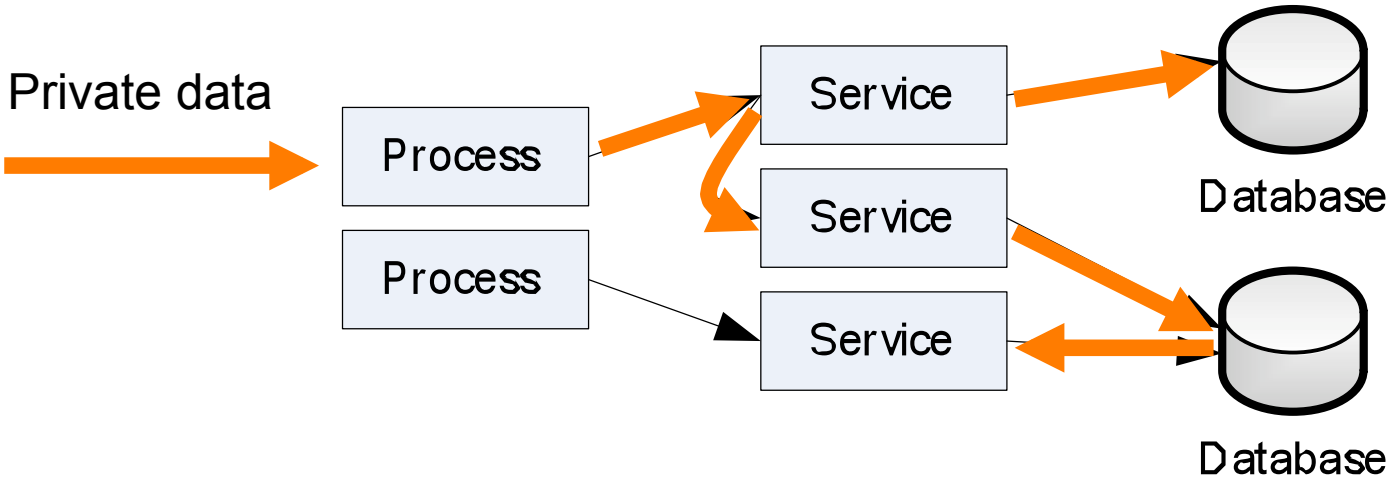
Providence: A Framework for Private Data Propagation Control in Service-Oriented Systems

Roman Khazankin

Vienna University of Technology

Problem statement



SOA

- How is the private information propagated throughout the system?
- For which purposes is it used?

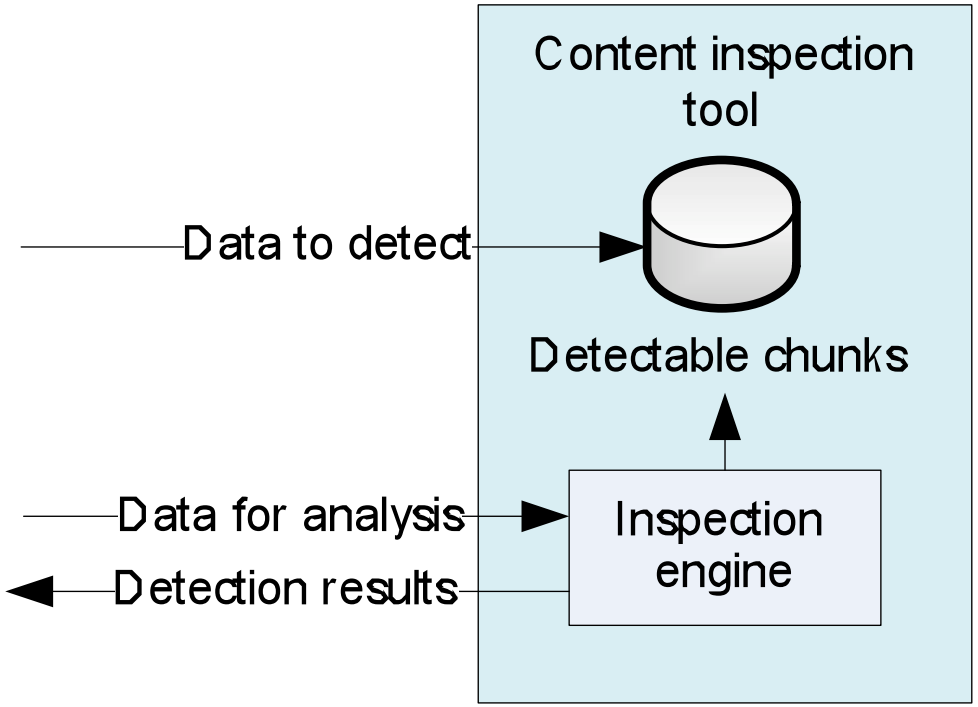
Related work

- Private information is guarded in a single source
- Privacy issues are considered only within particular process
- Approach requires interference with business logic of services
- **No general case, practical solutions**

Content Inspection

- Well-developed algorithms and tools for detecting pre-loaded information in network transmissions.
- Successfully applied in DLP (Data Loss Prevention) solutions

Content Inspection

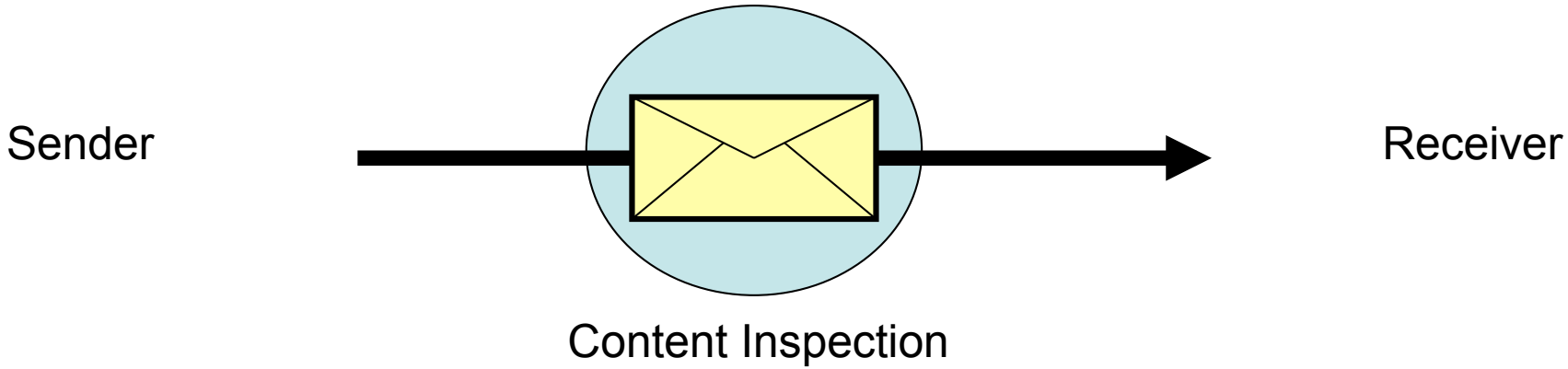


Message exchange monitoring

Context

- Time
- Application
- Process
- Credentials

...



Private data disclosures

Private information:

Name: John Johnson
Address : 1040 Example st. 2/3
Loan: 250 000 $
Date: 01.01.2010

Primitives:
(used by content inspection)

Name(“John Johnson”)
Address(“1040 Example st. 2/3”)
Amount(250000)
Date(01.01.2010)

Disclosure specification:

(Name(“John Johnson”) *OR*
Address(“1040 Example st. 2/3”))
AND
Amount(250000) *AND*
Date(01.01.2010)

Possible detectable form

```
……
<entry when=“1/1/10”>
        <n>Johnson J.</n>
        <sum>250,000</sum>
</entry>
……
```

Contexts

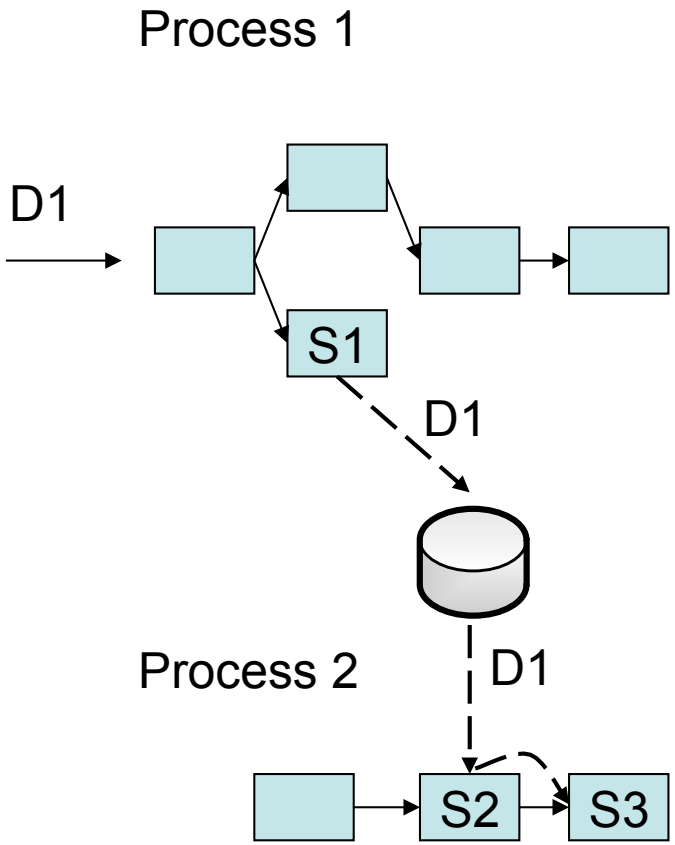
- A context can be a **subcontext** of another context.
- If a disclosure occurs in context C1 which is subcontext of context C2, then it also occurs in C2

Example.
Context = {Process A, Receiver = Endpoint1}
 is *subcontext* of
Context = {Process A}

Privacy policies and promises

- A privacy **promise** may be assigned to a **context**
- A privacy **policy** may be assigned to a **disclosure**
- Policies and promises are comparable (we can check if a promise satisfies a policy)
- So if a disclosure occurs in a context, we can check the promise against the policy

Example



Context = {Process 1}
Promise = {Only for system administration}

Disclosure **D1**
Policy = {System administration, research and development}

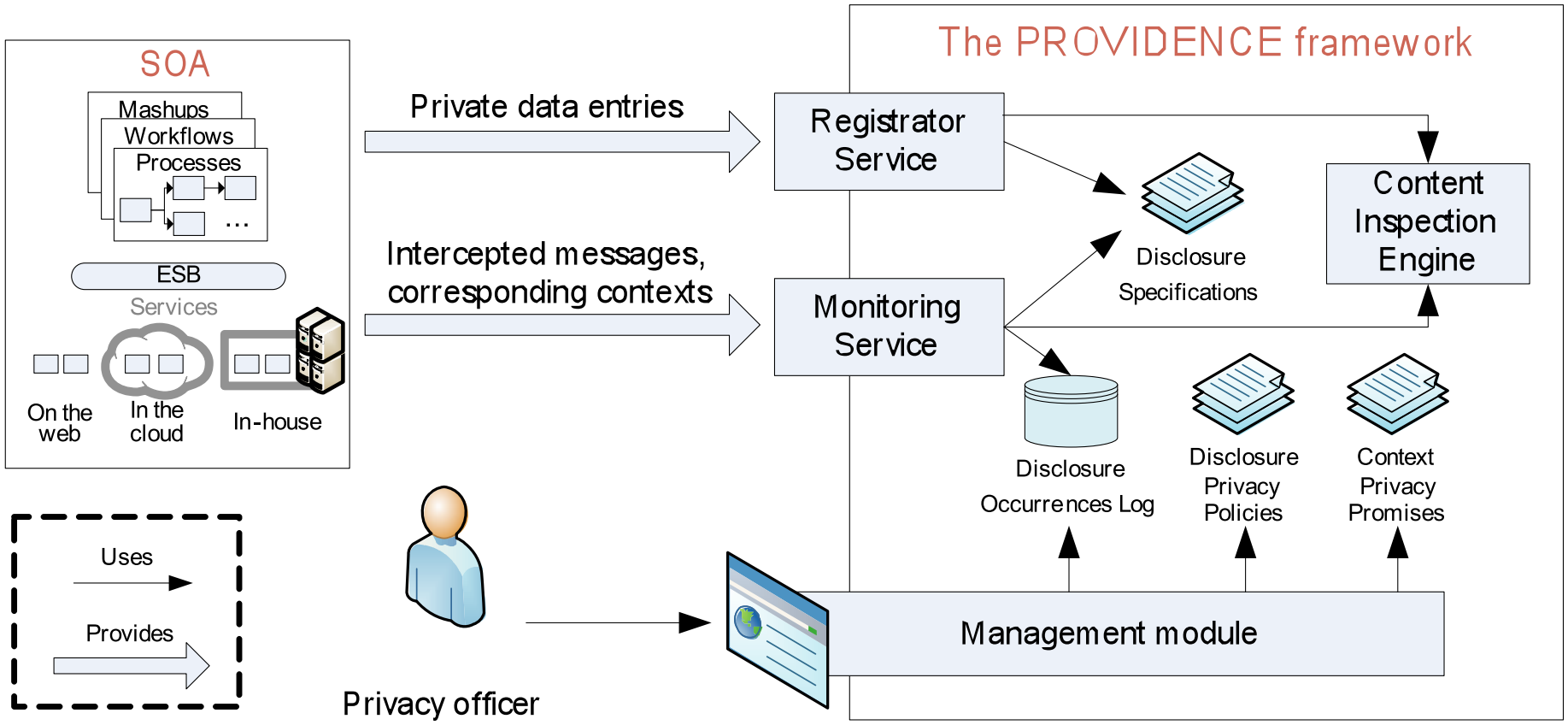
Context = {Process 2}
Promise = {System administration, Marketing}

Logging disclosure occurrences

Disclosure occurrences log enables for more functionality:

- Which disclosures occur in specified context?
- In which contexts disclosure of specified type occurs?
- What promise is enough for specified context to keep compliant with current private data usage practices?
- How is the private data of specified type actually used?
- What if we want to set another policy for private data or context, what violations will it produce for the current environment?

Architecture



Conclusion

- A framework which allows to control the private data propagation in SOA
- Loose coupling with the system (can be deployed, e.g. at ESB level)
- Different specifications can be used for policies and contexts

Thank you for attention.

Authors

- Roman Khazankin, TU Vienna
- Schahram Dustdar, TU Vienna