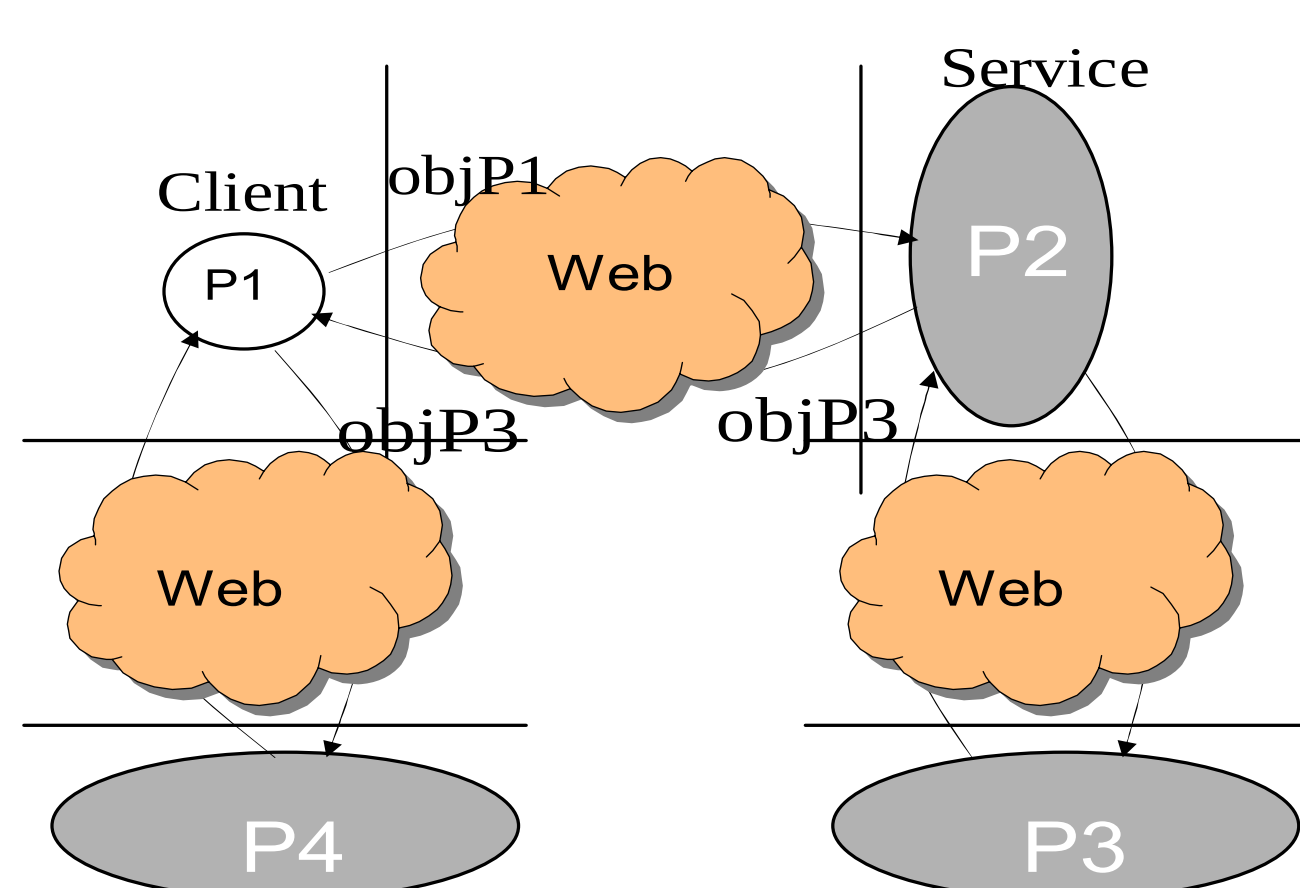


Information Leaks and Privacy in Web Services Computing

Ashish Kundu

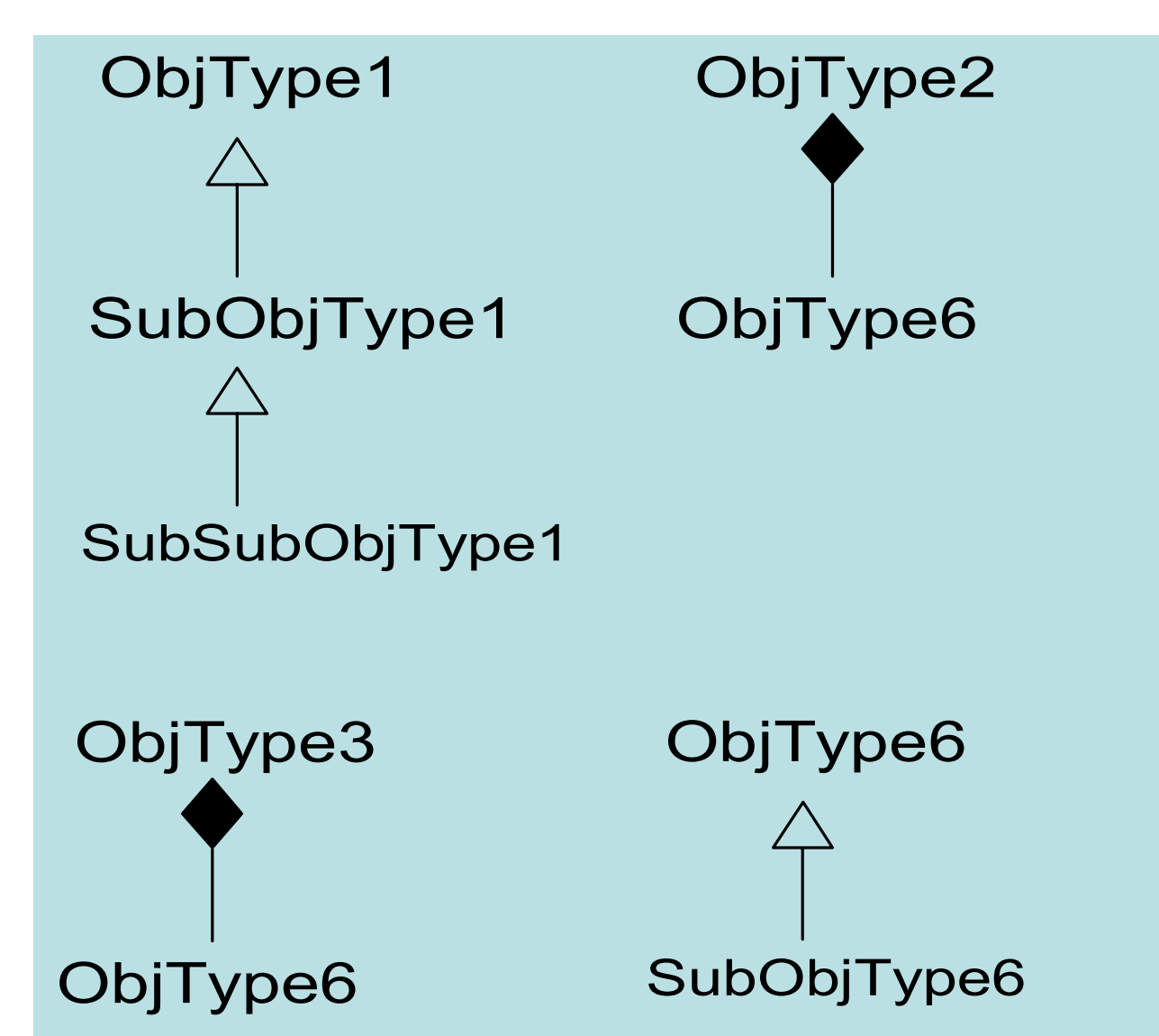
CERIAS & Dept. of Computer Science, Purdue University, ashishk@cs.purdue.edu

Web Services Interactions



Party	Return Type	Method	Parameter Type(s)
P ₁	void	callback ₁	ObjType ₃
P ₂	ObjType ₃	method ₂	ObjType ₁
P ₃	ObjType ₅	method ₃	ObjType ₂ , SubObjType ₁
P ₄	ObjType ₅	method ₄	ObjType ₃

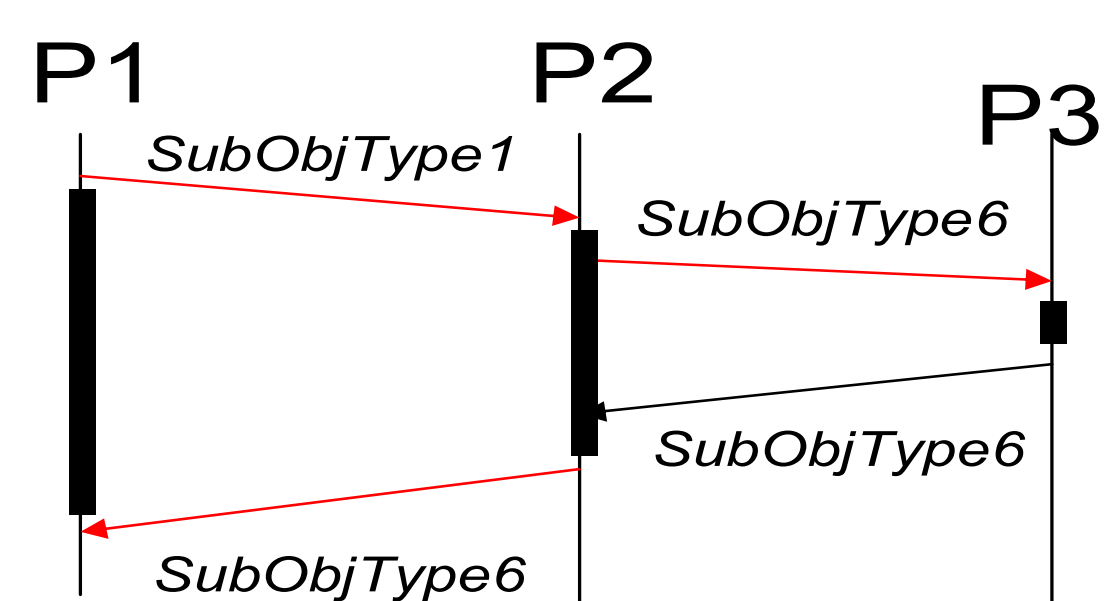
Type Hierarchy



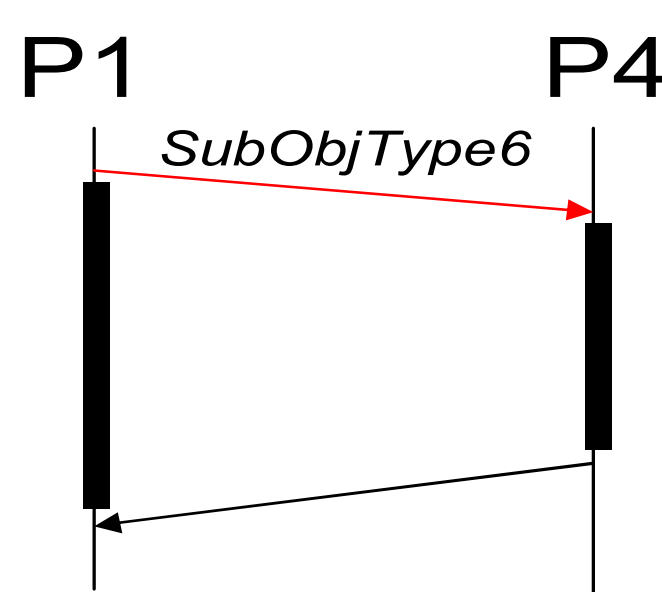
ObjType1	ObjType6
studentId: int getStudentId() setStudentId(sid: int)	studentId: int getStudentId() setStudentId(sid: int)
SubObjType1	SubObjType6
studentId: int DOB: java.util.Date getStudentId() setStudentId(sid: int) getDOB() setDOB(dob: java.util.Date)	studentId: int SSN: int getStudentId() setStudentId(sid: int) getSSN() setSSN(ssn: int)
SubSubObjType1	
studentId: int DOB: java.util.Date phone: int getStudentId() setStudentId(sid: int) getDOB() setDOB(dob: java.util.Date) setPhone(phone: int) getPhone(phone: int)	

Information Leaks

Compositional



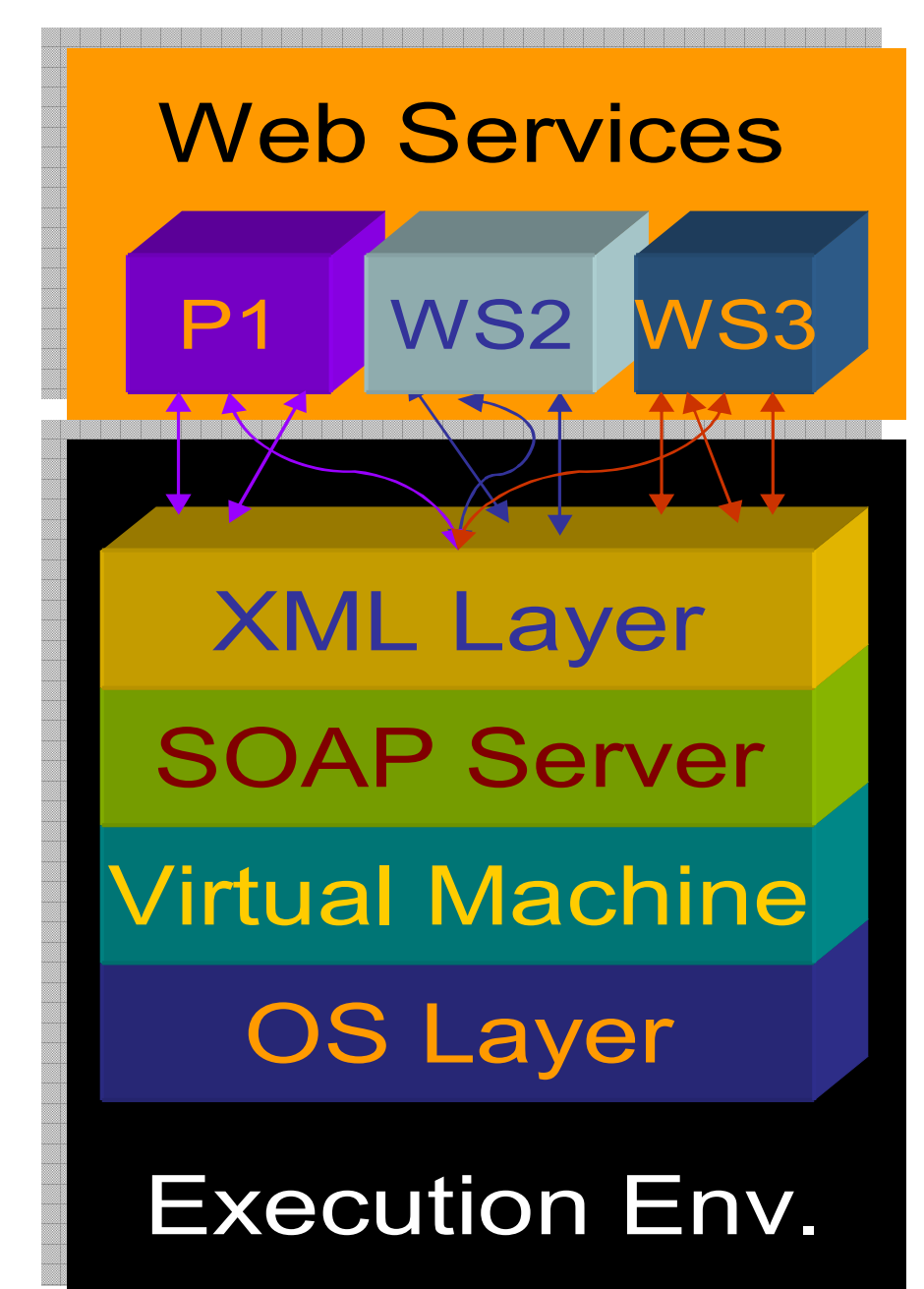
Elemental



- **Leak (P1,P2):** Instance of SubObjType1 for ObjType1
- **Leak (P2,P3):** Instance of SubObjType6 for ObjType6 in object of type ObjType2.
- **Leak (P2,P1):** Instance of SubObjType6 for ObjType6 in object of type ObjType3.
- **Leak (P1,P4):** Instance of SubObjType6 for ObjType6 in object of type ObjType3.

Privacy-Safe Web Services

- A web service is privacy-safe if it satisfies
 - **Leak-Source Property:** the party is not a source for any information leak
 - **Leak-Exploitation Property:** the party and its execution environment do not exploit any information leak.
- **Local Leak-Source Property**
 - **Development Guideline:** Strong type matching.



Transitive Leak-Source Property

Member-masking

```
for each specialized member variable X do {
    if there is a setter function // Java Bean
        invoke method setX(random or undefined value);
    else mask its value by X := random or undefined value;
}
```

Partial encryption

```
for each specialized member variable X do {
    encrypt value of X by
    X := encrypt value of X using key K;
}
```

Leak-Exploitation & KLIC

- **Leak-Exploitation Property**
 - Certification of Web Services
 - Versioning of Web Services
 - Static Program Analysis
 - Type Deception Tracking
- **Cost of Information Leaks (KLIC)**
 - Naive: $KLIC_{ij} = (\sum_{1 \leq k \leq m_{ij}} size(O_{kij}) - \sum_{1 \leq k \leq m_{ij}} size(O_{kij})) / \sum_{1 \leq k \leq m_{ij}} size(O_{kij})$
 - Weighted: $KLIC_{ij} = \sum_{1 \leq k \leq m_{ij}} \sum_{1 \leq l \leq n_k} (D_{lk} + V_{lk} + M_{lk}) * W_{lk}$

References

1. Ashish Kundu, *Information Leaks and Safe Web Services*, Submitted.