SOA Sicherheit

Dr. Bruce Sams
OPTIMAbit GmbH
bruce.sams@optimabit.com
+49 (8165) 65095

OWASP

Frankfurt, 25.11.08

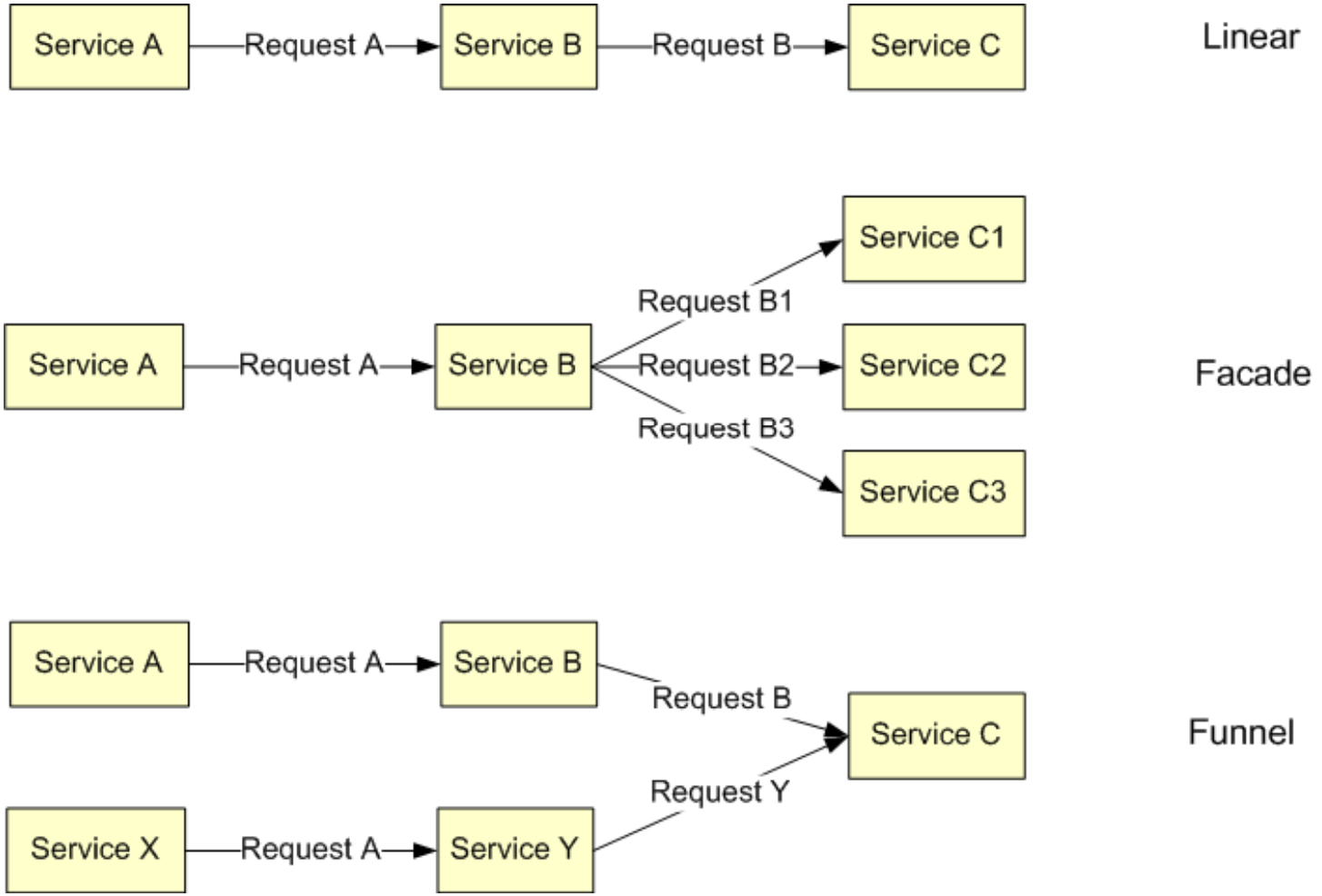
Agenda

- Zusammenfassung des Problems
- Standards für Sicherheit
- Architektur (Security als Service)
- Ausblick

Die Herausforderung von SOA Sicherheit

- Die große Herausforderung für Unternehmen ist es, umfassende Sicherheit für SOA mit einfacher Integration und Verwendung zu vereinen.
  - Lösungen, die auf "traditionellen" Techniken (z.B. Firewalls, SSL) basieren, sind unzulänglich.
  - Um erfolgreich zu sein, muss die Sicherheit auch als Service betrachtet werden.
  - Betrachtung der Macro- und Mikroskala

SOA = verbundene Services



SSL und Web Services

- SSL bietet keine Lösung für:
  - Weiterleitung einer Nachricht
  - Verschlüsselung der Nachricht
- SSL ist OK, wenn Sie keine Multi-Schritt Dienste haben und Identitäten nicht propagieren müssen.
  - Nachrichtensicherheit ist besser: die Nachrichten selber sind verschlüsselt.
- *Ein SOA braucht Nachrichtensicherheit, NICHT nur Transportsicherheit.*

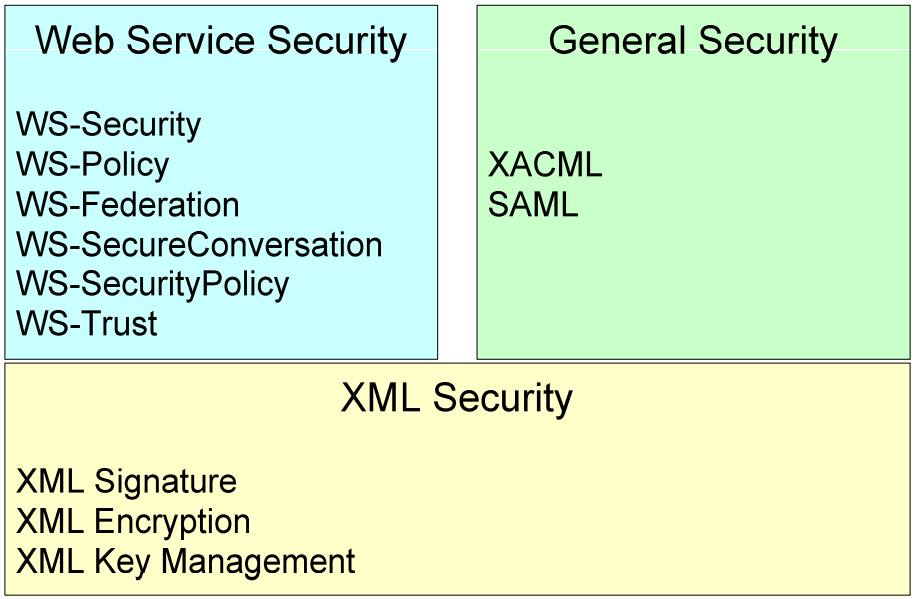
Wichtige Aspekte der SOA Sicherheit

- Multiple Schritte benötigen Propagierung der Identität
- Viele Protokolle (HTTP, JMS, FTP) bedeutet, dass die Sicherheit vom Protokol getrennt wird (Kein SSL).
- Interoperabilität braucht viele Standards!
- Arbeiten über verschiedenen Domänen braucht Identitätsmanagement & Föderation

STANDARDS

Keine Sicherheit in der WS Standard

- Die Web Services Spezifikation kümmert sich um die Sicherheit überhaupt nicht.
  - Weitere Standards vom W3C, OASIS, Microsoft, IBM und andere gehen spezifisch auf die Sicherheit.



Unverschlüsseltes XML

- Ohne Verschlüsselung

```
<PaymentInfo>
  <Name>John Smith</Name>
  <CreditCard Limit="5000" Currency="EUR">
    <Number>4019 2445 0277 5567</Number>
    <Issuer>Bank of the Internet</Issuer>
    <Expiration>200504</Expiration>
  </CreditCard>
</PaymentInfo>
```

Kompletter Inhalt verschlüsselt

```
<EncryptedData
        xmlns="http://www.w3.org/2001/04/xmlenc#"
        Type="http://www.optimabit.de/xml">
  <CipherData>
    <CipherValue>A23B45C56</CipherValue>
  </CipherData>
</EncryptedData>
```

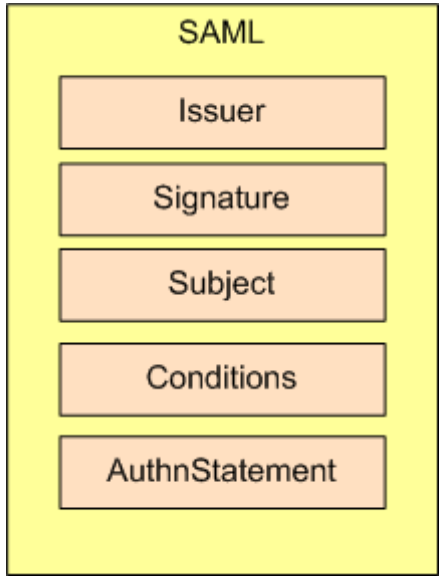
Verschlüsseltes XML

- Nur die Nummer wird verschlüsselt

```
<PaymentInfo>
  <Name>John Smith</Name>
  <CreditCard Limit="5000" Currency="EUR">
    <Number><EncryptedData>...</EncryptedData></Number>
    <Issuer>Bank of the Internet</Issuer>
    <Expiration>200504</Expiration>
  </CreditCard>
 </PaymentInfo>
```

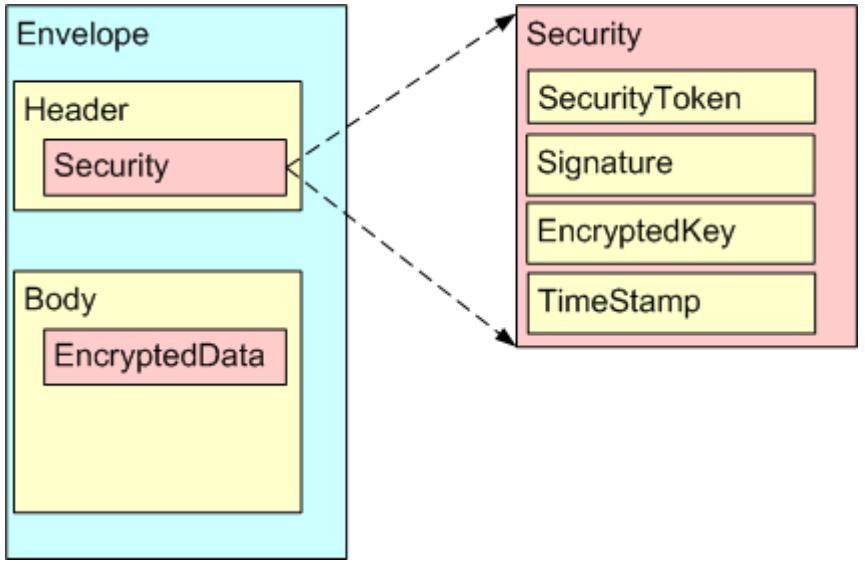
SAML Overview

- Security Assertion Markup Language (SAML) defines
  - a standard for the format and embedding of security information in an XML file
  - protocols for token exchange
- The most important uses of SAML is for authentication, authorization and SSO.
  - SAML tokens contain a collection of assertions about a subject (e.g. a user, his identity and his rights).



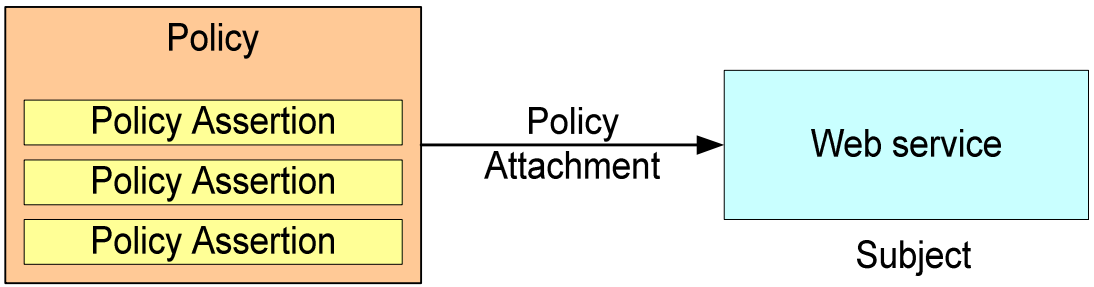
WS-Security ist ein "Meta" Standard

- Der OASIS WS-Security Standard vereint verschiedene bestehende XML and WS Standards, unter einem "Schirm".
- Die WS-Security Spezifikation funktioniert mit SOAP Version 1.1 +.
- SAML Tokens werden oft verwendet.



WS-Policy Diagram

- Eine WS-Policy ist eine Anzahl von Behauptungen, die an einem Web Service angebunden wird.
  - Das Attachment findet am WSDL oder UDDI statt, so dass der Service selbst nicht modifiziert werden muss.
  - Es bietet eine flexible Konfiguration, z.B. der Server kann mehrere Optionen akzeptieren, hat aber eine Präferenz.



Sicherheitsbedenken beim WS-Policy

- Angriffe
  - Fingerprinting
  - Downgrading
  - Denial of Service
- Optionen:
  - Authentifizeirung für den Client, um die Policy lesen zu dürfen
  - Keine sicherheitsrelevante Information in der Policy
  - Verschlüsselte Übertragung.
  - Sichere Konfiguration

Chained Policy Bomb

```
<Policy wsu:Id="p1">
  <PolicyReference URI="#p2"/ >
  <PolicyReference URI="#p2"/>
</Policy>

<Policy wsu:Id="p2" >
  <PolicyReference URI="#p3"/>
  <PolicyReference URI="#p3"/>
</Policy>

<Policy wsu:Id="p3" >
  <PolicyReference URI="#p4"/>
  <PolicyReference URI="#p4"/>
</Policy>
<!-- Policy Id p4 through p101 -->
```

```
<Policy wsu:Id="p101" >
  <wsa:UsingAddressing />
</Policy>
```

This call results in 2^100 policy statements…

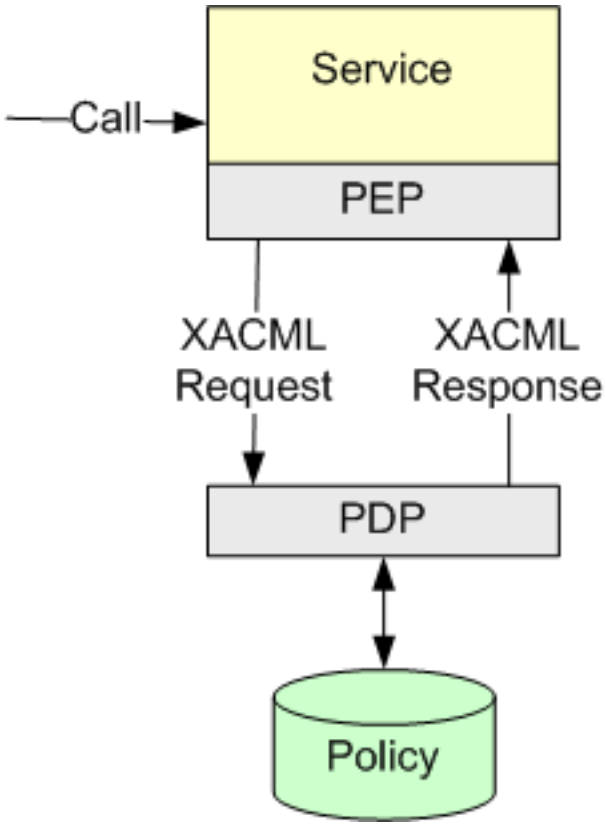
ARCHITEKTUR

Autorisierung

- Um die Identität des Ausrufers zu propagieren, muss sie in der Nachricht selbst eingebunden sein.
  - Der Empfänger kann die Information nutzen, um Entscheidungen über Authentifizierung bzw. Autorisierung zu treffen.
  - Die Security Assertion Markup Language (SAML) beitet Tokens mit Authentifizierungsinformation an.

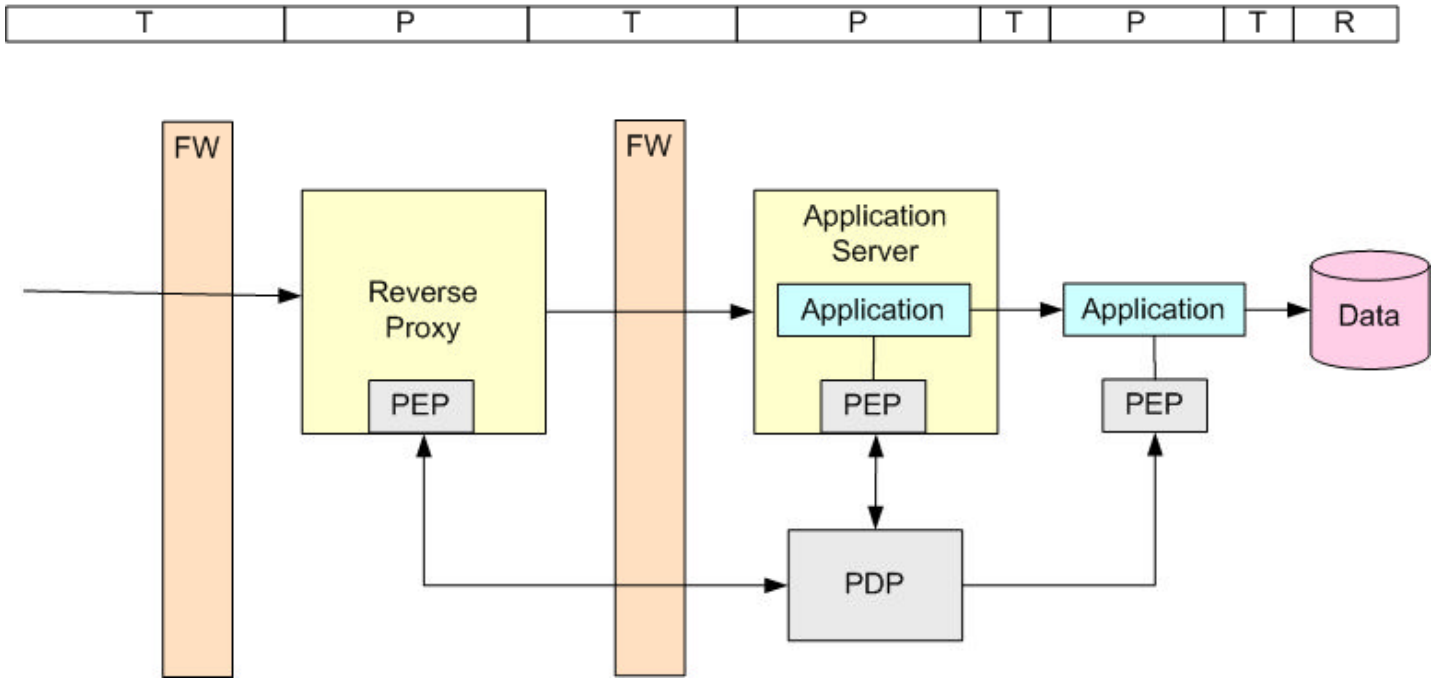
Sicherheit als Service

- Ein Policy Decision Point (PDP) ist für Entscheidungen über Zugriffskontrolle verantwortlich.
- Ein Policy Enforcement Point (PEP) ist für die Durchsetzung einer PDP-Entscheidung verantwortlich.
- XACML ist ein Standard für den Austausch.

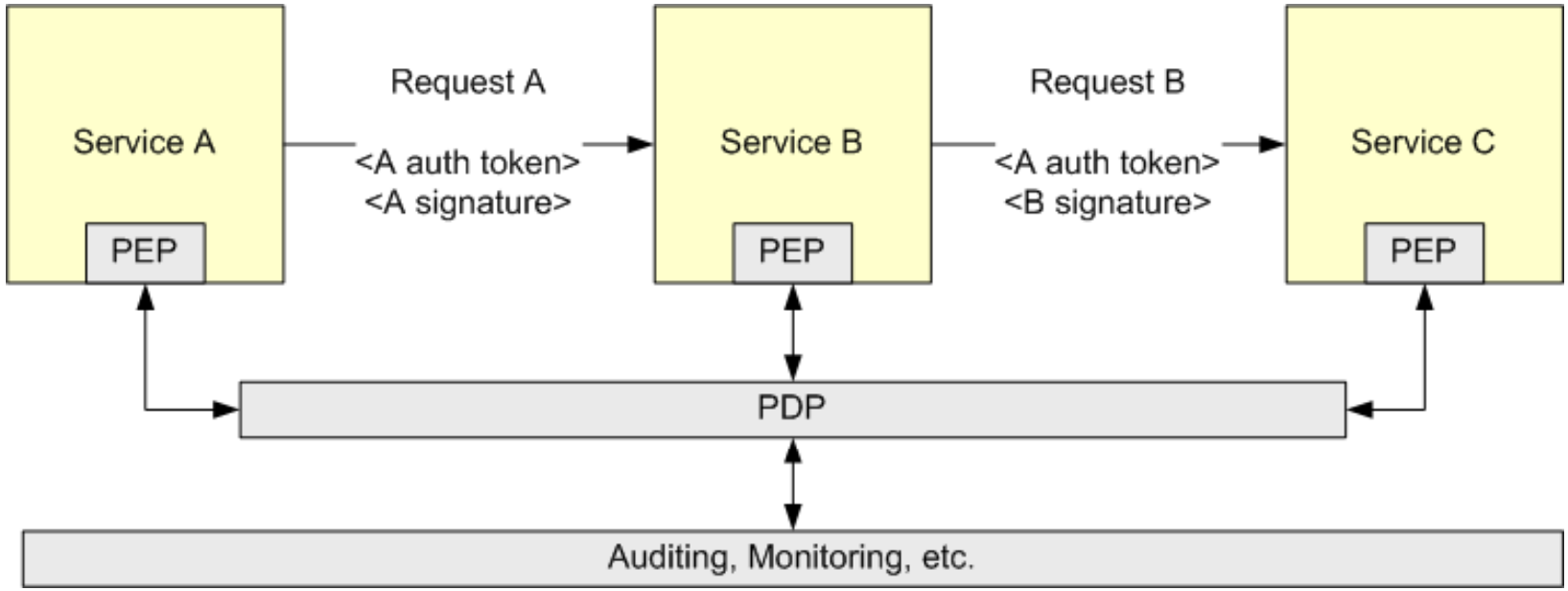


Die Prinzipien angewandt

- So sieht eine Anwendung mit PDP und PEP aus.
  - PEP wo Daten von T=>P wechseln

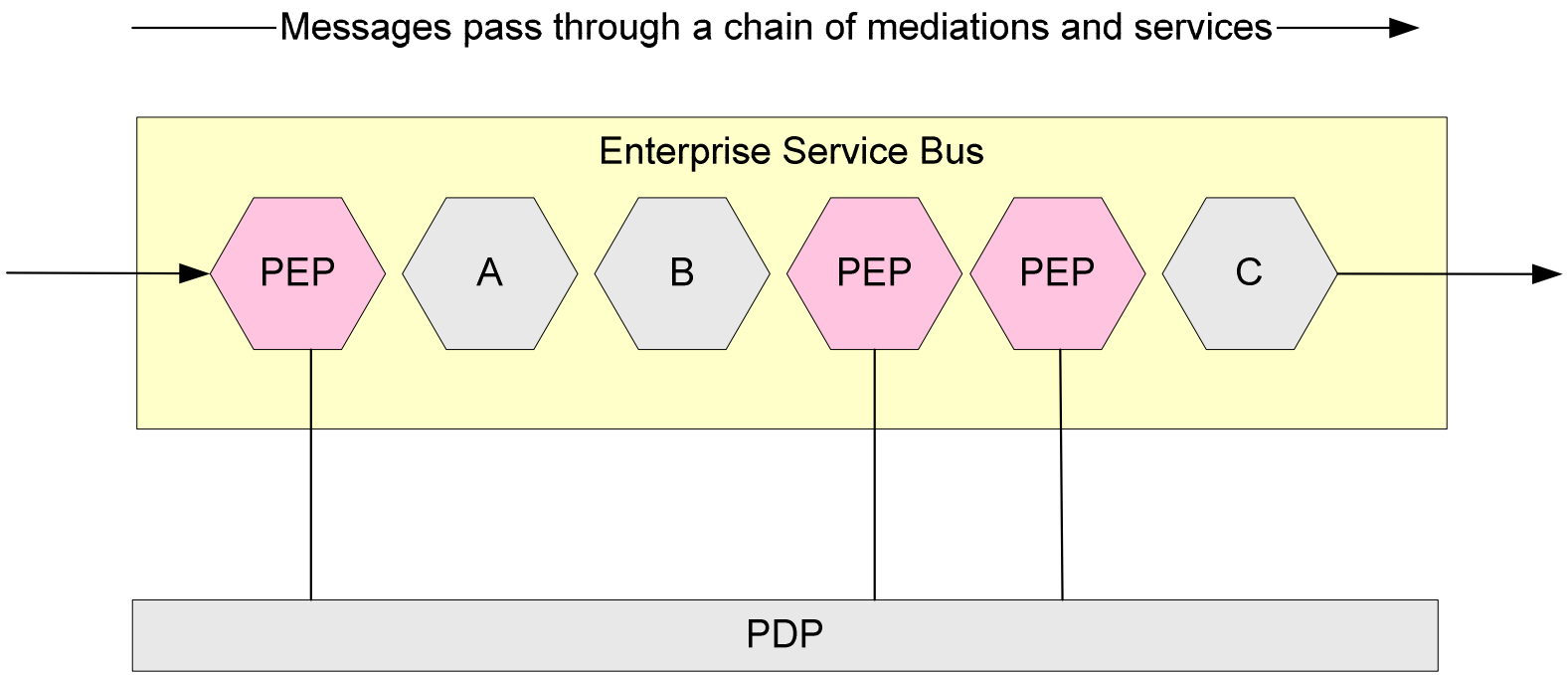


Sicherheit als Service: Diagram



Using the "Security as a Service" paradigm

Sicherheit als Service: Diagram



Using the "Security as a Service" paradigm

Zugriffskontrolle: grob => fein

- Eine Progression von groben zu feinen Entscheidungen hat einige Vorteile:
  - Unautorisierte Requests können weit entfernt von den geschützten Ressourcen blockiert werden.
  - Die Anzahl von unautorisierten Requests, die das Ziel erreichen, wird minimiert.
  - Die Performanceeinbußen von Sicherheitslogik in der Anwendungsschicht werden minimiert.
  - Das Kombinieren von Services in einer Kette wird vereinfacht.

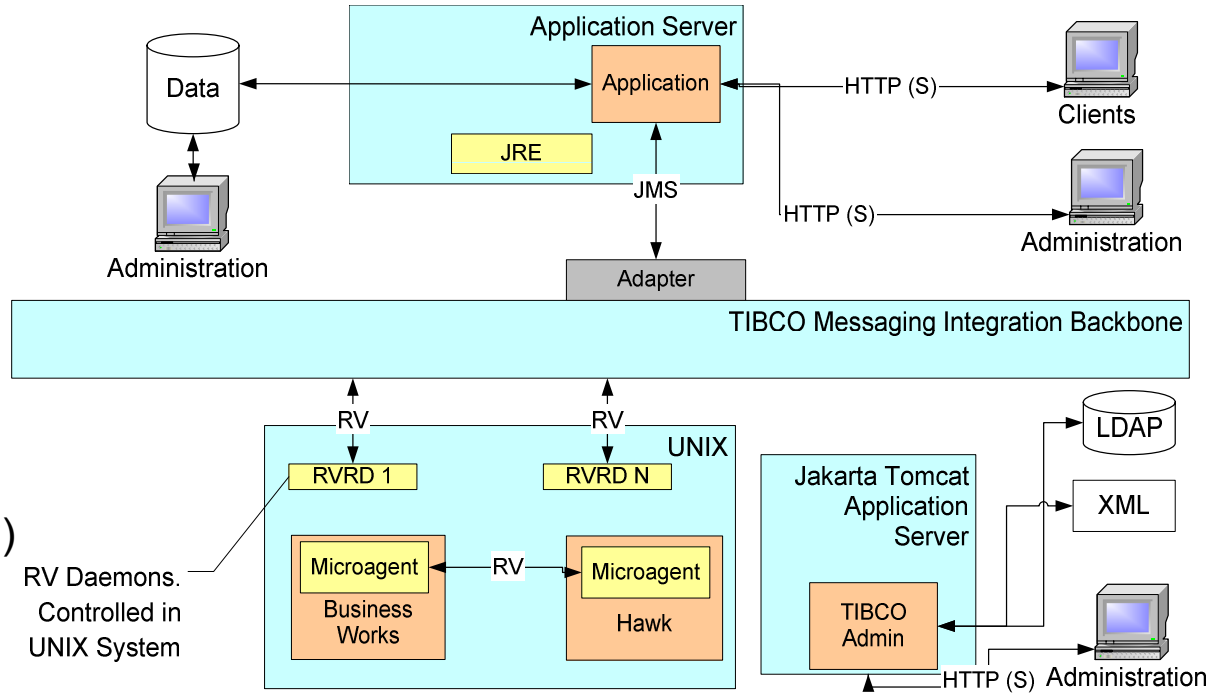
AUSBLICK

Macro and Micro Scales

- Sichere Konfiguration und Installation!
  - Beispiel TIBCO.

Some Issues

- RV Multicast Protocol
- Databases
- Passwords (admin)
- Passwords (configuration)
- Adapter authentication
- Application server
- Daemons
- Identity Management (LDAP)
- ...



Angriffsarten

- Heute
  - SOA haben dasselbe Probleme wie WebApps (Injection, Parameter Manipulation, usw).
  - Viel XML bedueten potentielle Probleme mit Entity Bomben
- Voraussage für die nächste 3 Jahre
  - Konfigurationsprobleme
  - Inkonsistene Zugriffskontrolle
  - Implementationsprobleme
  - Neue, unbekannte

Google SOA/SSO Schwachstelle

- T. Groß (IBM) analysiert SAML SSO
  - Einige "Probleme" im Protokol/Bindings festgestellt.

- Armando, etal (Uni Genova) analysiert Google SAML/SSO.
  - Oktober 2008 => Google Schwachstelle basierend auf Rogue Service Provider.

Andere SOA/WS Optionen

- Enterprise Service Bus
  - Hohe Performance, Transaktionen
  - Keine ESB-spezifische Sicherheitsstandards

- REST Web Services
  - Einfache Handhabung
  - Keine Verwendung von WS-*, erfordert Eigenlösungen (Tokens/Replay, A&A).

Zusammenfassung

- Standards und Implementierungen existieren, um WS-Security zu verwenden, aber der "volle" Einsatz ist nicht weit verbreitet.
- Neue Standards verkomplizieren die Implementierung (z.B. WS-Policy).
- Neue, bisher unerforschte Schwachstellen werden entdeckt.
- Unachtsame Programmierung wird ein Problem sein

Danke!

- Danke für Ihre Aufmerksamkeit!
  - Dr. Bruce Sams
  - OPTIMA Business Information Technology
  - bruce.sams@optimabit.com