Just-In-Time Security: Sicherheit im Entwicklungsprozess

Mike Wiesner
SpringSource Germany

Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

Über mich

- Senior Consultant bei SpringSource Germany
- Spring-/Security-Consulting
- Trainings
- IT-Security Consulting / Reviews
- mike.wiesner@springsource.com

Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

Agenda

- Was ist Application Security?
- Requirements & Prozesse
- Risikoanalyse
- Testing

Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

Was ist Application Security?

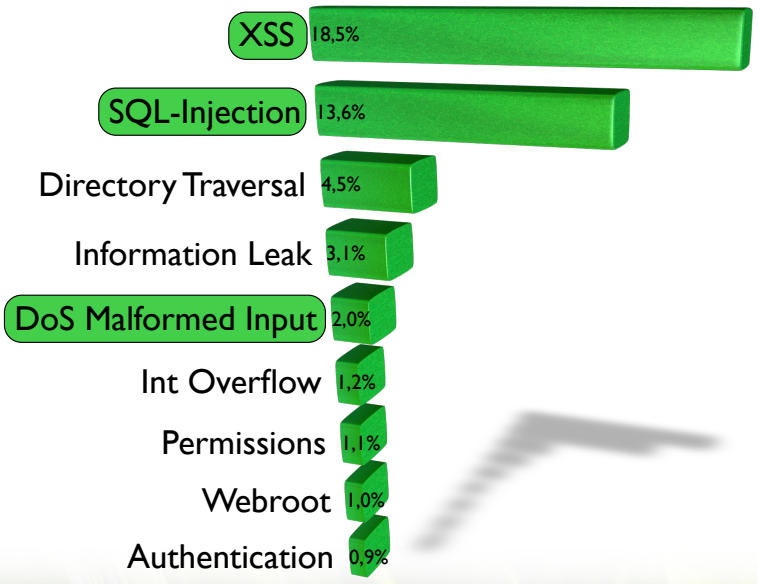
- Ziel: Eine Anwendung „in sich" sicher zu machen
- Also keine:
  - (Web App) Firewalls
  - IDS
  - Proxy-Server
  - Betriebssystem Security

Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

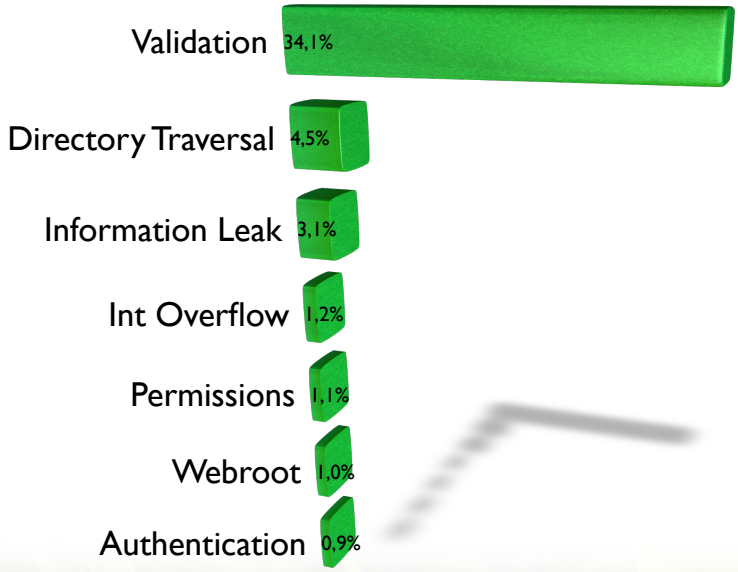
Security Merkmale

- Authentifizierung
  - Identitätsfeststellung
- Autorisierung
  - Prüfung von Rechten
- Überwachung
  - Protokollierungen (Audit-Trails)

Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.



Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.



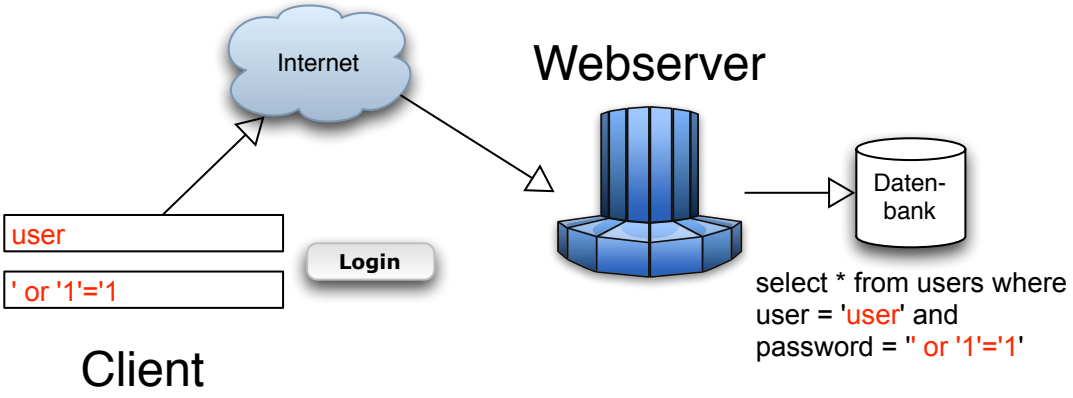
Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

Security Hardening

- Validierungen
  - SQL-Injection / Code-Injection
  - Cross-Site-Scripting
  - ...
- Defense in Depth
  - Prüfungen in mehreren Schichten
  - Verschiedene Technologien
- Secure Coding
  - Fehlerbehandlung, Trust-Zones, ...

Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

SQL Injection



Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

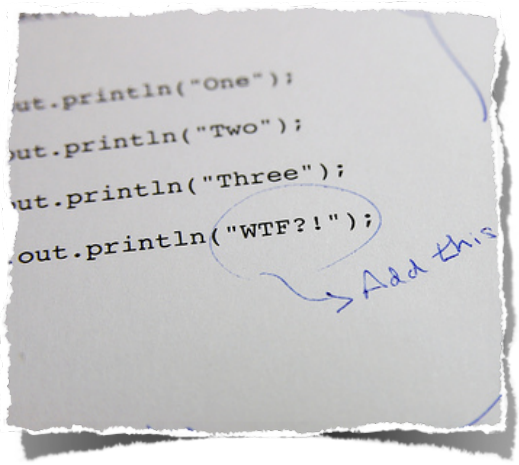
Defense in Depth

- 100 protzentigen Schutz gibt es nicht
- Prüfung in mehreren Schichten
- Mit verschiedenen Technologien

Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

Secure Coding

- Fehlerbehandlung
- Trust-Zones
- Generische Schnittstellen



Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

Agenda

- Was ist Application Security?
- Requirements & Prozesse
- Risikoanalyse
- Testing

Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

Requirements / Prozesse

- Ist Application Security ein Requirement oder ein Prozess?
- Beides!
- Requirements:
  - Authentifizierung, Autorisierung, Logging
- Prozess:
  - Defense in Depth, Secure Coding, (Validierung)

Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

Security Requirements

- Müssen nicht von Anfang an berücksichtigt werden
- Späteres hinzufügen mit AOP möglich
  - Evolutionäres Design
- Zum Teil Non-Functional Requirements

Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

Eingabe-Validierung

- „Die Maske muss ein Textfeld enthalten“
- Und welche Zeichen sind erlaubt?
  - UTF-8, UTF-16, Alphanumerisch, ...
- Validierung fängt mit der Anforderung an!
- Immer Whitelist hinterfragen
- Für alle Dateneingänge
  - Webservices, OCR-Scans, Batch-Imports

Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

Ausgabe-Validierung

- „Der Benutzer darf HTML-Tags verwenden“
- Welche davon?
  - <javascript>, <frame>, ...
- Auch hier: Whitelist in der Anforderung

Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

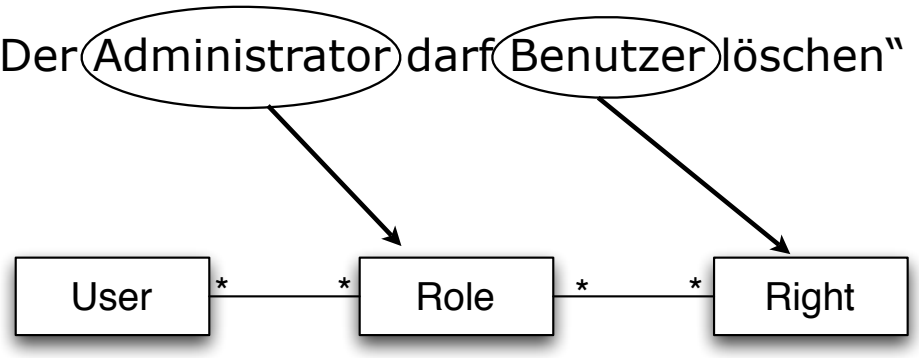
Rollen / Rechte

- „Der Administrator darf die Methode deleteAll des User-Objekts aufrufen“
- Schon zu spezifisch
- Rollen werden direkt mit Methoden/URLs verbunden
- Eine Abstraktation mehr einführen:
  - Rechte

Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

Rollen / Rechte

- „Der Administrator darf Benutzer löschen“



- Grobe Regel:
  - pro Use-Case ein Recht
  - pro Akteur ein Rolle

Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

Security Requirements

- Security Requirements fangen bei der Spezifikation an
- Zu allgemeine Beschreibungen führen zu unsicherem Code
- Fachabteilung muss darauf sensibilisiert werden
- Projektleiter muss auf die Einhaltung achten

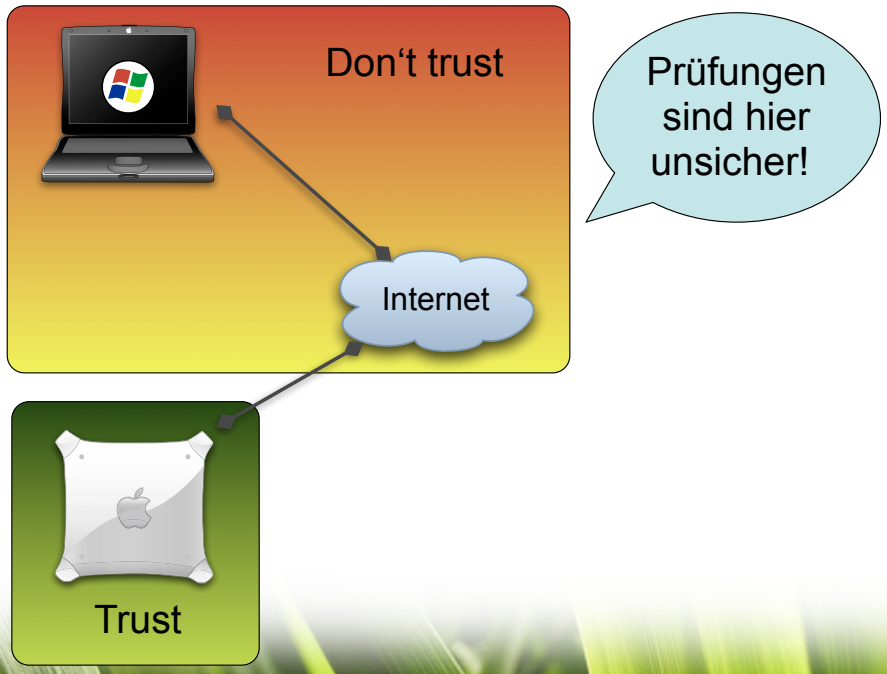
Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

Security als Prozess

- Security ist mehr als nur Validierung, Rechteprüfung und Authentifizierung
- Unüberwindbare Prüfungen
- Secure Coding
- Einsatz von Security Patterns

Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

Unüberwindbare Prüfungen



Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

Trust-Zones

- Entwickler müssen mit Trust-Zones vertraut sein
- Requirements alleine wäre auch im „Don‘t Trust“-Bereich erfüllt
- Anwendung ist aber nicht sicher!

- Sicherstellung durch Architektur-Constraints möglich
  - AspectJ, SonarJ, ...

Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

Secure Coding

```
public interface OrderService {

  public List<Order> getUserOrders(String subject);

  public List<Order> getAllOrders();

}
```

- getUserOrders soll per AJAX verwendet werden
  - Export via DWR als JavaScript-Proxy
- Jetzt steht aber auch getAllOrders zur Verfügung!

Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

Code Injection

```
public interface Calculator {

  public long calculate(String function);

}
```

- „function" ist JRuby-Code
- Wird vom Benutzer eingegeben
- Wie kontrolliere ich ob es nur Formeln sind?

Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

Security Patterns

- Etliche Lösungsansätze vorhanden
- Vorteil:
  - Problem, Lösung und auch Nachteile sind detailliert beschreiben
  - Guter Startpunkt
- Nachteil:
  - Es gibt viele davon
  - Umsetzung immer noch selbst notwendig

Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

Security als Prozess

- Zuordnung zu Requirements oft schwer möglich
- Ist Teil nahezu jedem Requirements
- Lösung:
  - Security Trainings
  - Reviews (Audits, Pair Programming)
  - Penetration Tests
  - Zum Teil: Code-Scanner (pmd, findBug)
- Team Know-How ist hier der Schlüssel zum Erfolg!

Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

Agenda

- Was ist Application Security?
- Requirements & Prozesse
- Risikoanalyse
- Testing

Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

„Was kann die Anwendung?“

Nichts, aber das ist dafür 100% sicher!

Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

Wieviel Security und Wann?

- Security ist wichtig
- Aber Funktionen sind es auch
- Wie finde ich also die richtige Balance?

Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

Risikoanalyse

- Sicherheitsrisiken lösen bedeutet Aufwand
- Aufwand = Zeit = Nicht vorhanden ;-)
- Macht es Sinn jedes theoretische Risiko zu behandeln?

Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

Risikoanalyse

Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

Risikoanalyse

- Technische Risiken auf Geschäftsrisiken abbilden
- Was bedeutet es für das Business wenn diese Lücke ausgenutzt wird?
- Ist der Aufwand zum Schließen höher wie der maximale Verlust?

Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

Maximaler Verlust

- Nicht nur direkter Verlust
  - Serverausfall
  - Datenverlust
- sonder auch
  - Imageverlust
  - Vertrauensverlust bei Kunden
  - Schadenersatzansprüche

Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

Agenda

- Was ist Application Security?
- Requirements & Prozesse
- Risikoanalyse
- Testing

Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

Security Tests

- Benutzer erstellen keine Bug-Reports wenn Sie „zu viel" dürfen
- Security-Bugs müssen während der Entwicklung gefunden werden
- Manuelles Testen ist sehr Aufwendig
- Automatisiertes Testen teilweise aber auch
- Es kommt auf die richtigen Tools an

Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

Berechtigungen

- 5 Rollen, 60 Rechte
  - 5 x 60 = 300 JUnit Test Cases
- Besser mit FIT (Framework for Integration Tests):

de.secpod.acegiaop.AcegiAOPFitLoginTest		
benutzername	**passwort**	**isValidLogin()**
admin	pwadmin	1
user	pwuser	1
user		
falsc		

de.secpod.acegiaop.AcegiAOPFitRoleTest		
rolle	**adminOp()**	**userOp()**
ADMIN	1	1
USER	0 *expected* 1 *actual*	1

Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

fit2.htm

Login

Das Login ist nur mit korrektem Benutzernamen und Passwort erlaubt.

In der folgenden Tabelle sind diverse Kombinationen aus Benutzernamen und Passwörter hinterlegt. In der Spalte isValidLogin() ist angegeben ob diese Kombination erlaubt ist.

de.secpod.acegiaop.AcegiAOPFitLoginTest		
benutzername	**passwort**	**isValidLogin()**
admin	pwadmin	1
user	pwuser	1
user	falschesPasswort	0
falscherUser	pwuser	0

Rollen

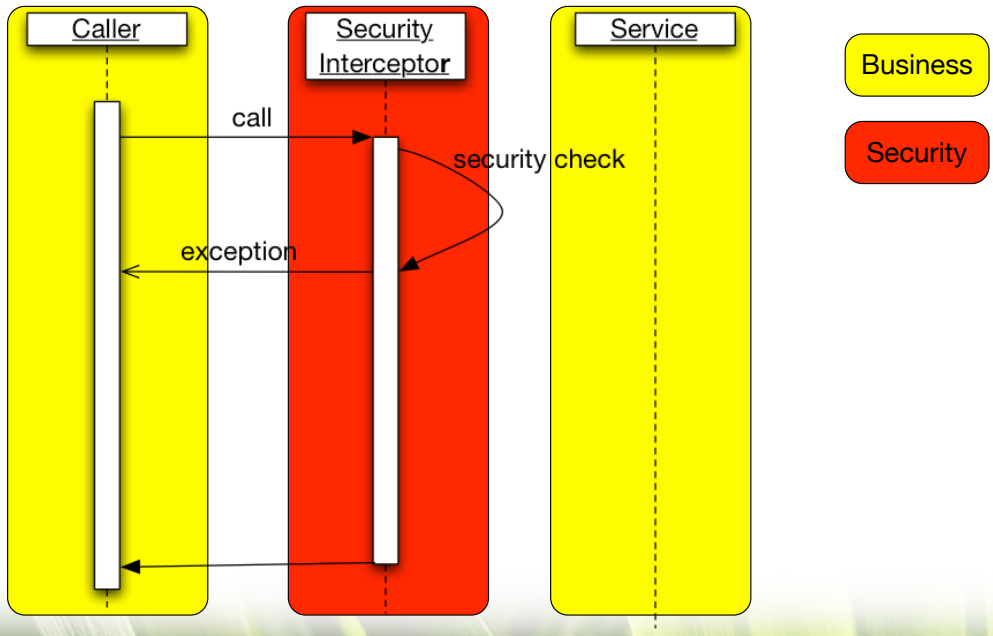
In der Anwendung sind diversen Rollen mit jeweils verschiedenen Rechten vorhanden. In der ersten Spalte ist jeweils die Rolle und in den Nachfolgenden Spalten die Rechte.

1 = Zugriff erlaubt
0 = Zugriff verweigert

de.secpod.acegiaop.AcegiAOPFitRoleTest		
rolle	**adminOp()**	**userOp()**
ADMIN	1	1
USER	0 *expected* 1 *actual*	1

S 1 Ab 1 1/1 Bei 0,5 cm Ze Sp 1 0/145 MAK ÄND ERW ÜB

Security Tests mit Spring Security (AOP)



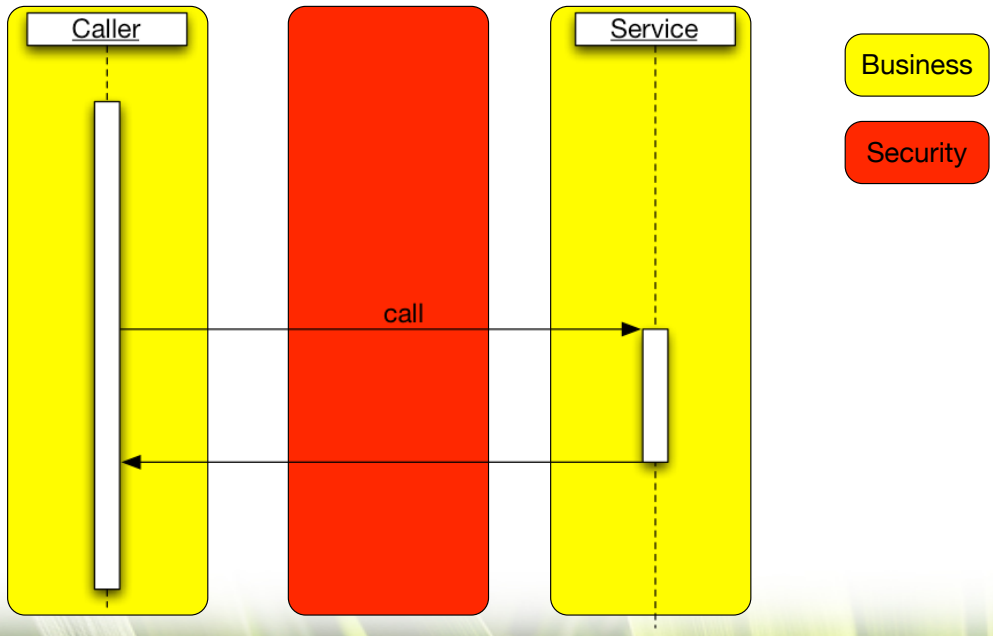
Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

Business Tests

- Keine Beeinflussung der Business Tests durch Security
- Deswegen: Security abschaltbar machen
- Auch hier: Tools können helfen

Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

Business Tests mit Spring Security (AOP)



Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

Coding Tests

- Trust-Zones mit AspectJ festlegen
  - Security-Check im „Don't Trust"-Bereich führt zu Compiler Warnungen/Fehler
- Analyse-Tools wie PMD, FindBugs, ...
- Alles aber nur begrenzt möglich
- Beste Möglichkeit: Reviews

Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

Abschluss-Tests

- In regelmäßigen Abstanden (z.B. nach Milestones):
  - Security Audits (White-Box)
  - Penetration Tests (Black-Box)
- Nicht erst 2 Wochen vor Go-Live!

Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

FAZIT

Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

Fazit

- Alle Akteure sind gefragt: Anforderer, Manager, Entwickler
- Tools können helfen
- Jeder Entwickler benötigt aber Security Know-How
- Security ist nicht nur ein Requirement
- Security kann niemals als erledigt gekennzeichnet werden
  - Neue Funktionen = Neue Security
  - Neue Angriffsmöglichkeiten, ...

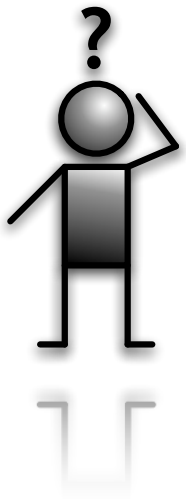
Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

Fragen?

Mike Wiesner
SpringSource Germany

mike.wiesner@springsource.com
Skype: mikewiesner

http://www.springsource.com/de
http://www.mwiesner.com



Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.

Credits

In diesem Vortrag wurde Fotos von folgenden Usern verwendet:

http://www.flickr.com/photos/rocketraccoon
http://www.flickr.com/photos/cambodia4kidsorg
http://www.flickr.com/photos/mukluk
http://www.flickr.com/photos/azrainman
http://www.flickr.com/photos/scottfeldstein

Copyright 2008 SpringSource. Copying, publishing or distributing without express written permission is prohibited.