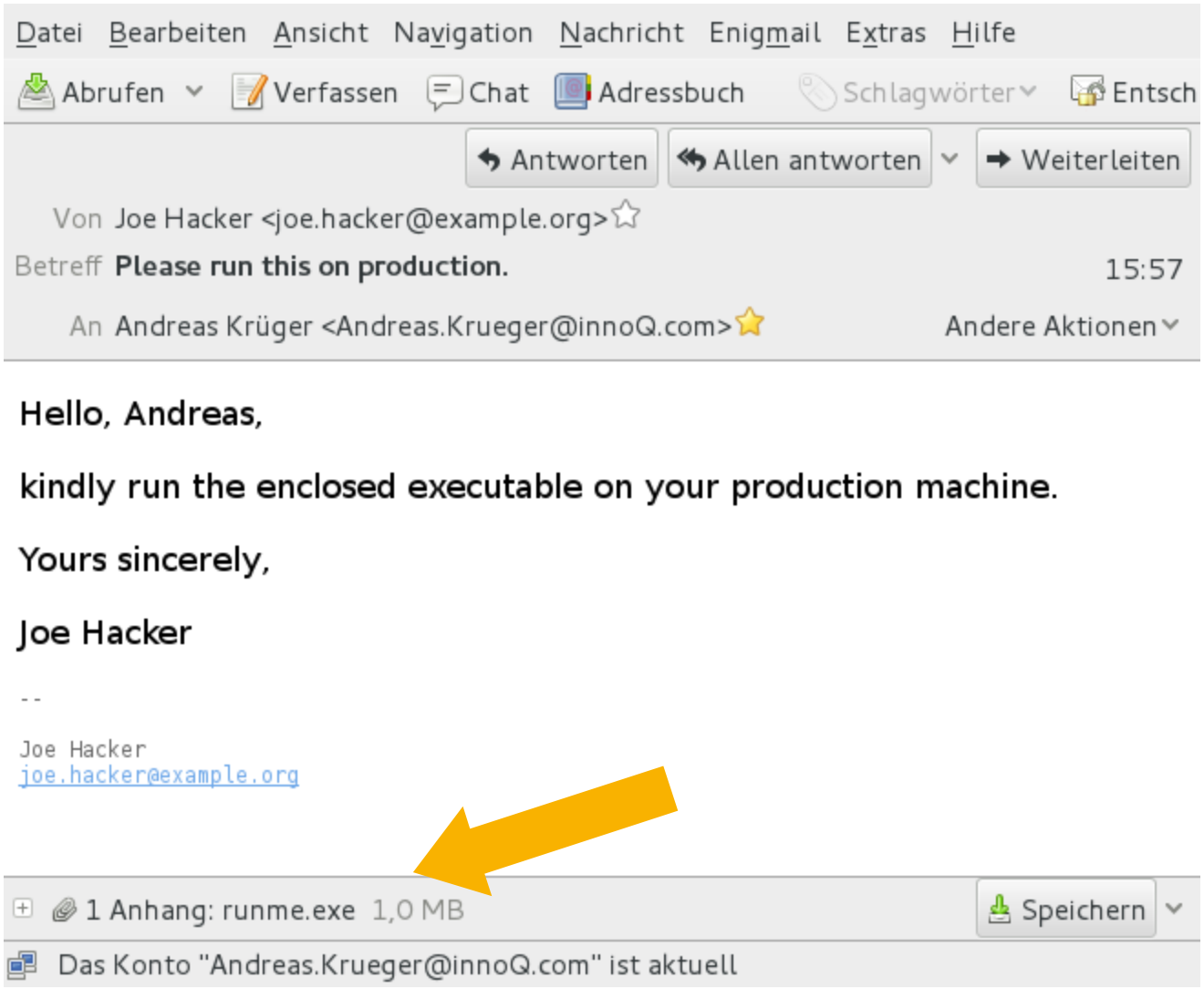
Sicherheitsprobleme üblicher Build- und Deploymentumgebungen

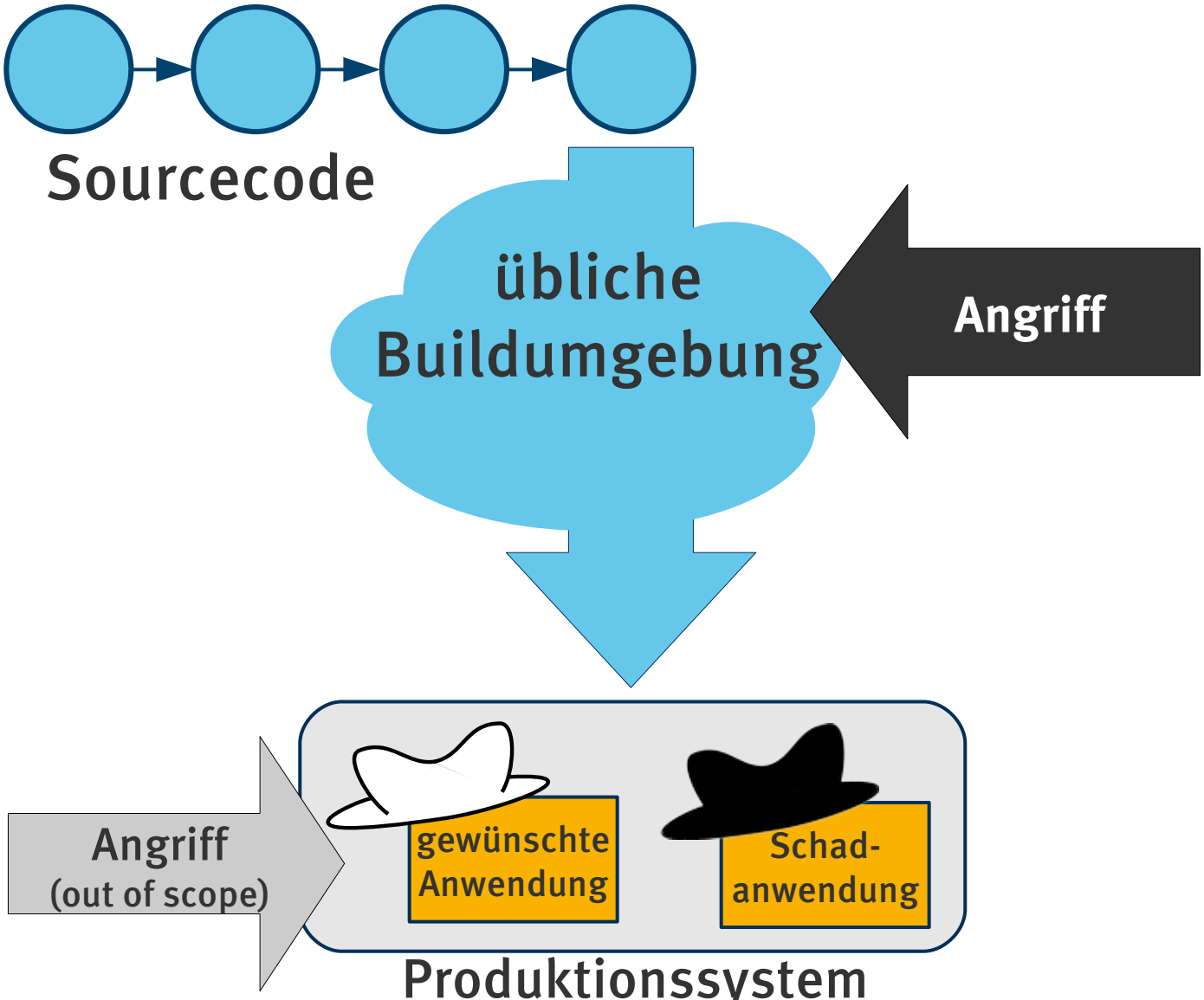
Dr. Andreas Krüger
Senior Consultant bei innoQ Deutschland GmbH
für Continous Lifecycle 2015-11-10

innoQ

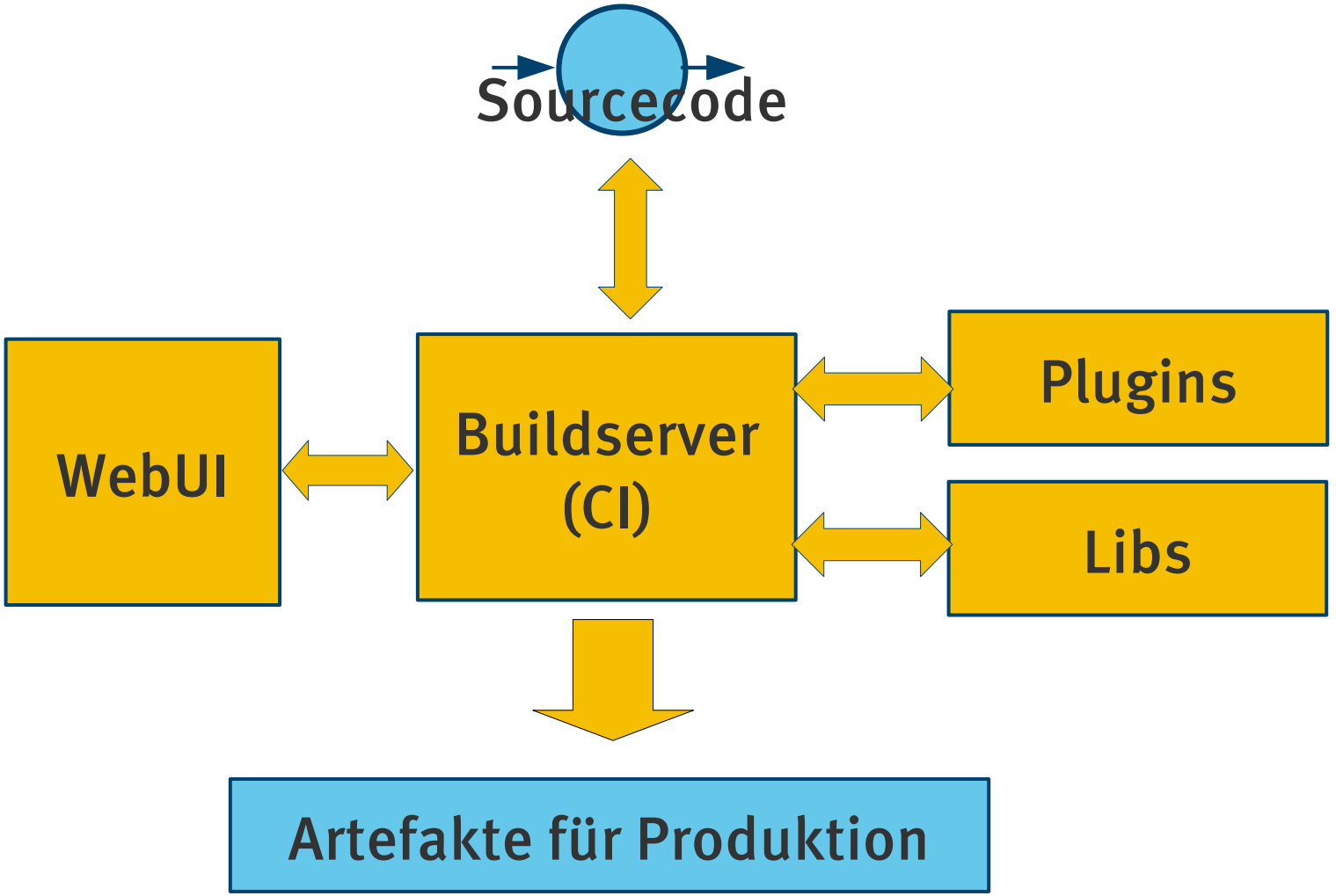
Bitte einmal kopfschütteln



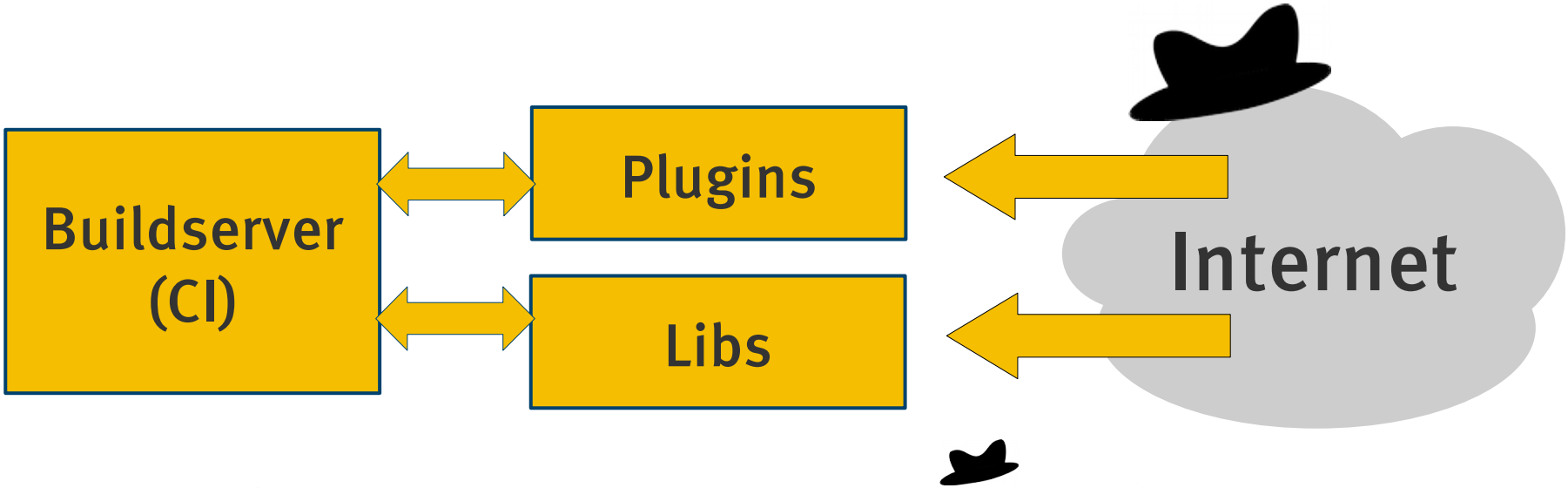
Thema hier



übliche Buildumgebung

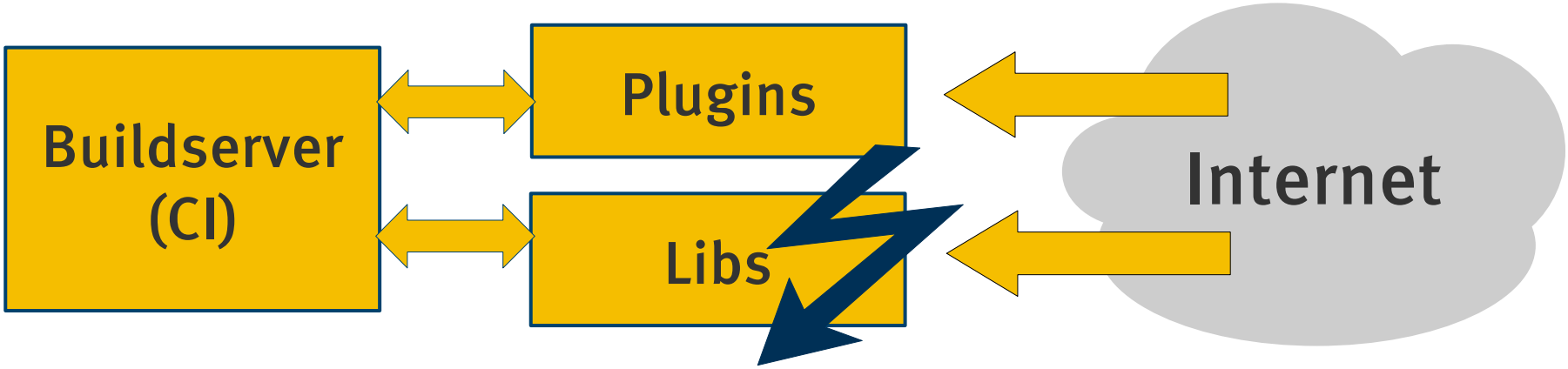


Plugins und Libs aus dem Internet



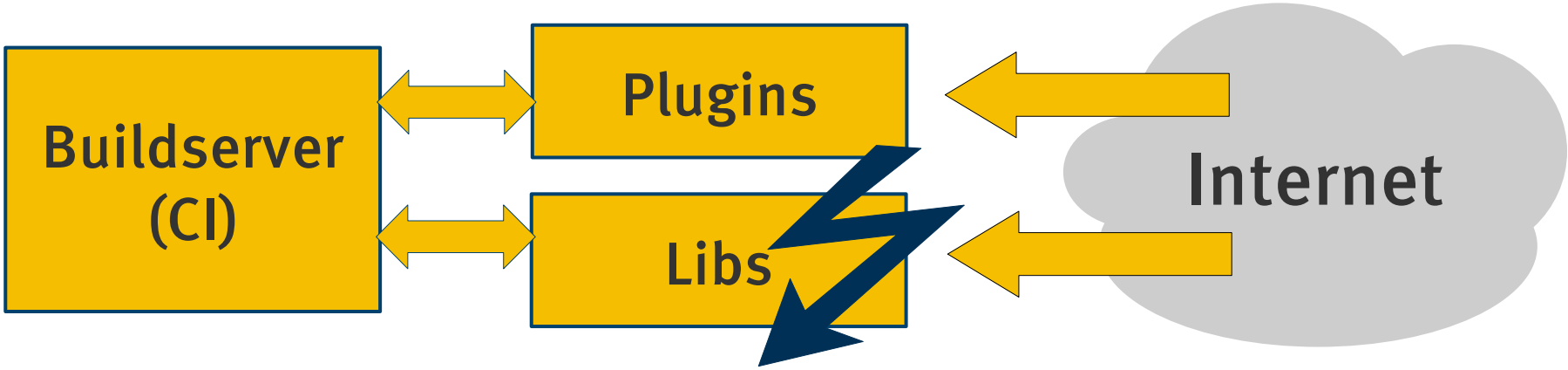
- häufig kein HTTPS, und HTTP ist fälschbar
- keine Überprüfung
- (Angriffe auf den fernen Server?)

Wahre Geschichte

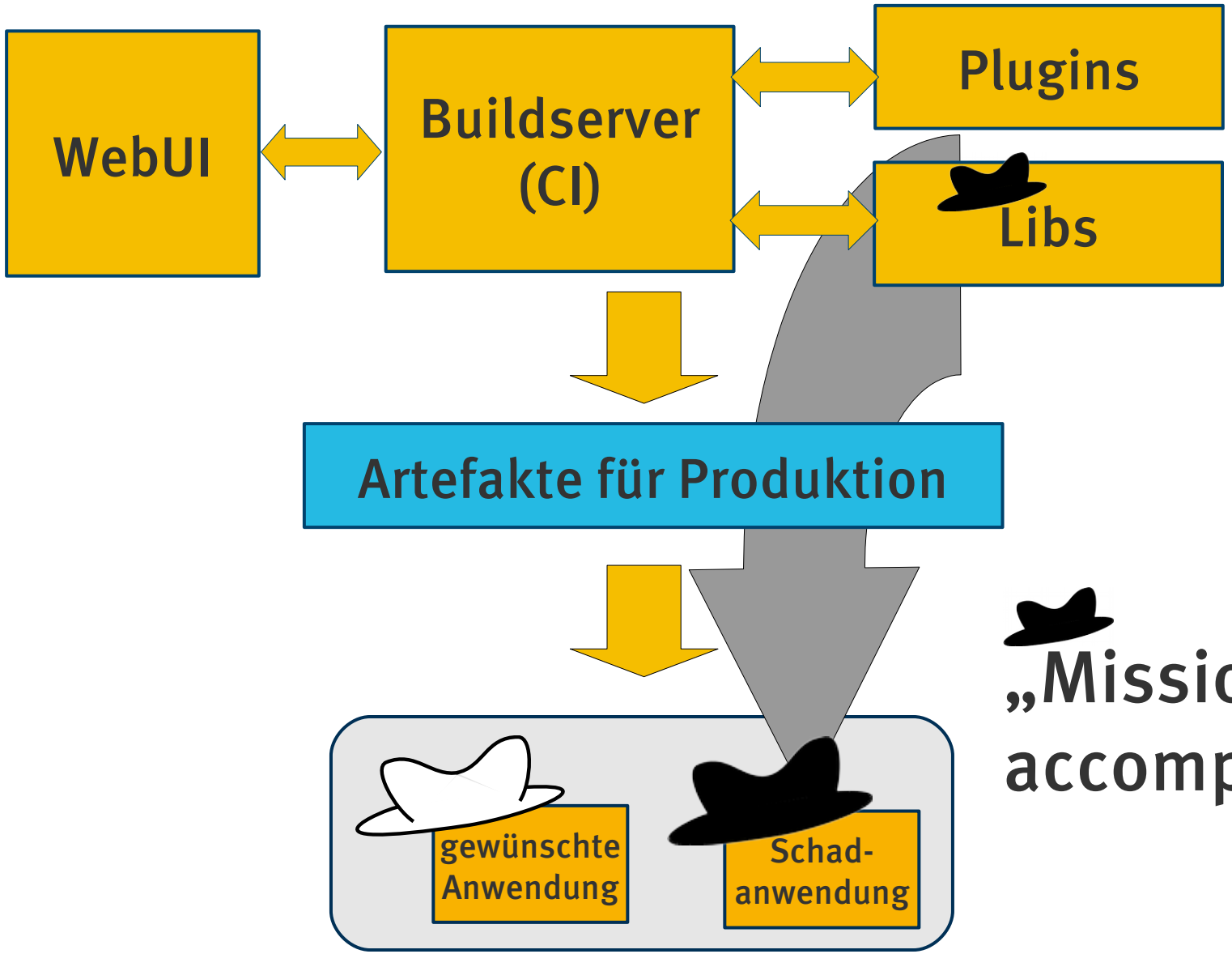


- Build kracht
- soll: Eine JS-Datei

Wahre Geschichte

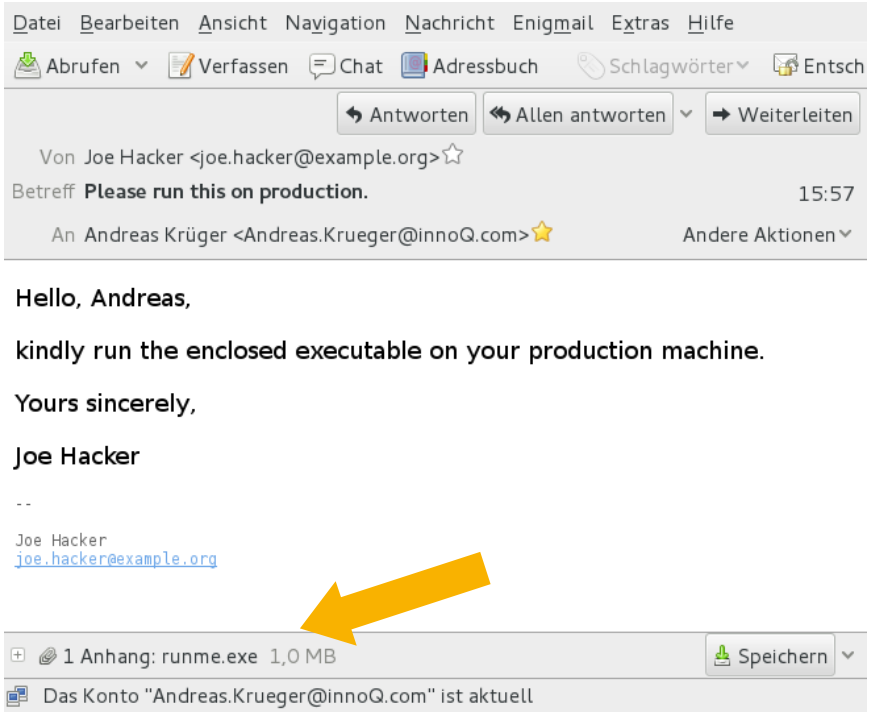


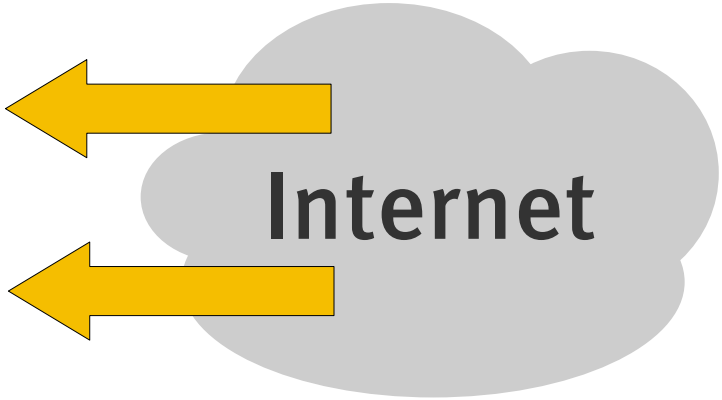
- Build kracht
- soll: Eine JS-Datei
- ist: HTML “Domain zu verkaufen”.



„Mission accomplished!"

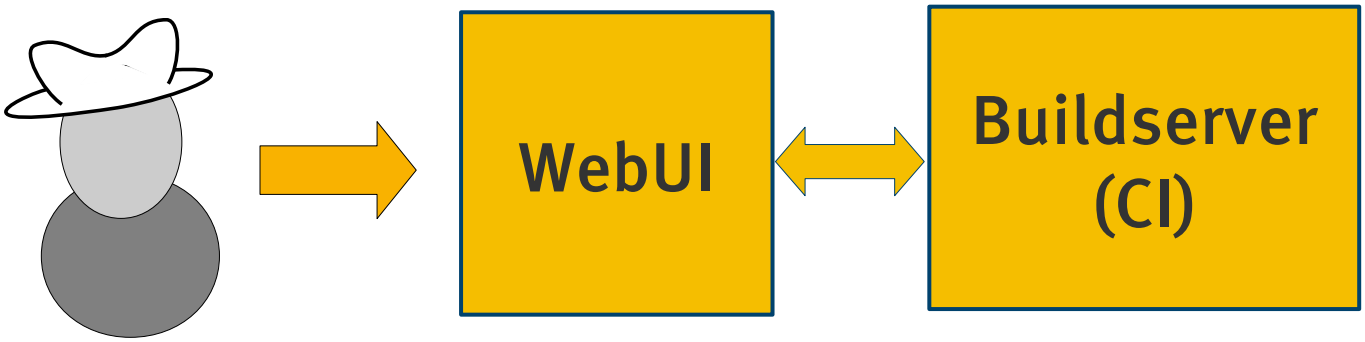
ziemlich





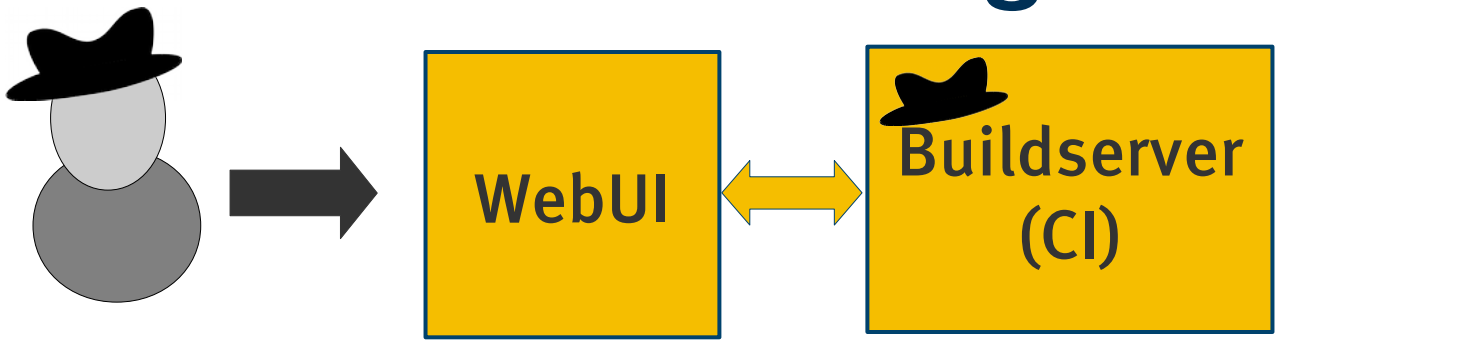
ähnlich...

Buildserverkonfiguration



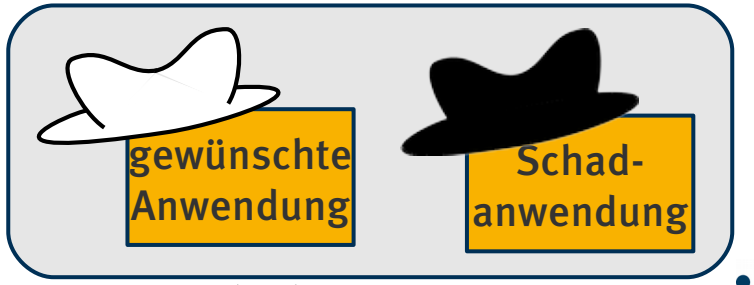
- Jeder aus dem Entwicklungsteam darf. (Hoffentlich!)
- Nachvollziehbar, wer wann was?
- Konfigurationsreviews?

Buildserverkonfiguration



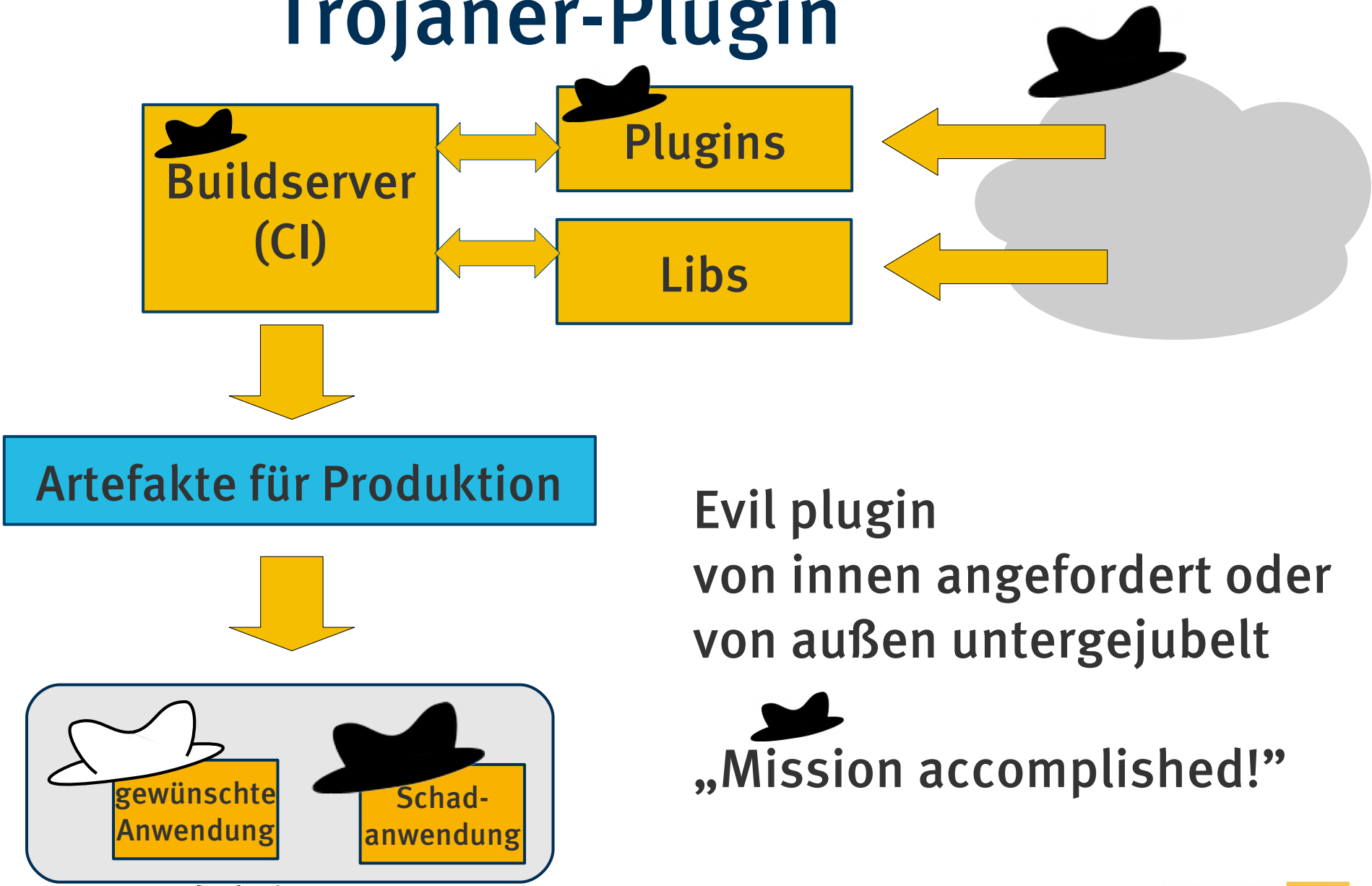
Der Buildserver kann beliebige Artefakte produzieren.

„Mission accomplished!”



Produktionssystem

Trojaner-Plugin



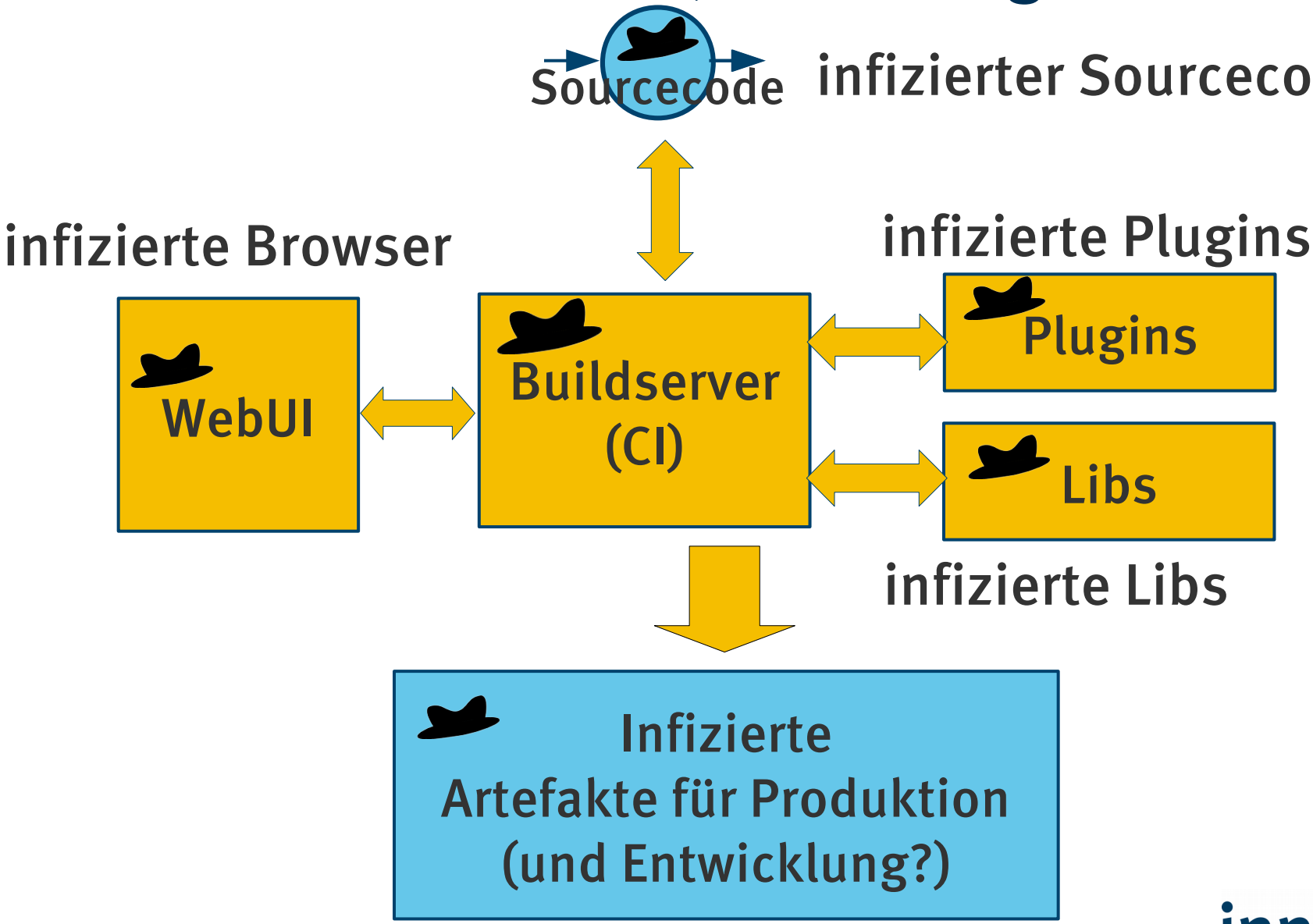
Evil plugin
von innen angefordert oder
von außen untergejubelt

„Mission accomplished!”

Wer einmalig ein
Trojaner-Plugin
im Buildsystem
untergebracht hat,
hat dort Kontrolle
für lange Zeit.

innoQ

Eroberter Buildserver, viele Möglichkeiten.



Repositories

Beispiele:

- Maven Repositories
  mit Libraries und Plugins.
- RPM Repositories
  mit Software für die Produktion.
- Puppet-Konfiguration
  auf dem zentralen Puppet-Server.

Repositories

- Der Buildserver schreibt viele davon.
- Wer den Buildserver kontrolliert,
  hat Schreibrechte.
- Ist Repoinhalt signiert,
  so hat der Buildserver den Schlüssel.
- Gefälschte Artefakt im Repo
  sind unauffälliger als
  Hintertüren im Sourcecode.

Passen Sie auf ihre Repos auf!

innoQ

Was kann helfen?

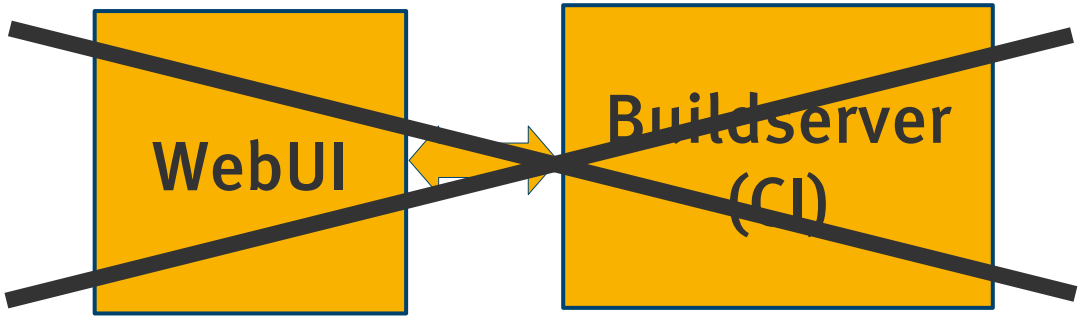
innoQ

Was kann helfen?

- Nachvollziehbarkeit
- Kryptografische Signaturen

innoQ

Nachvollziehbarer Buildserver



Buildserverkonfiguration ist Software und gehört ins Versionsmanagement.

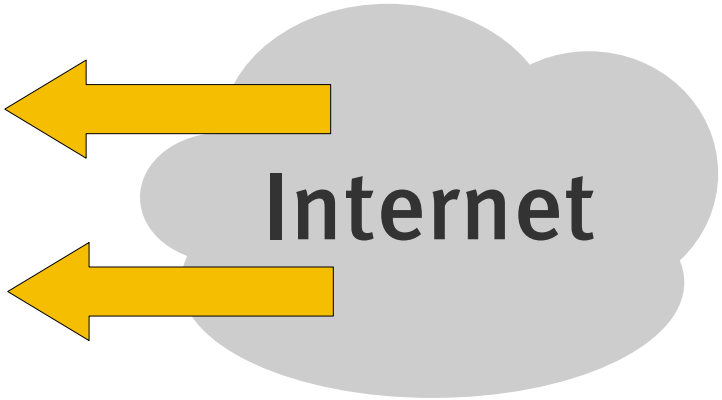
Ich möchte meinen Buildserver bauen können!

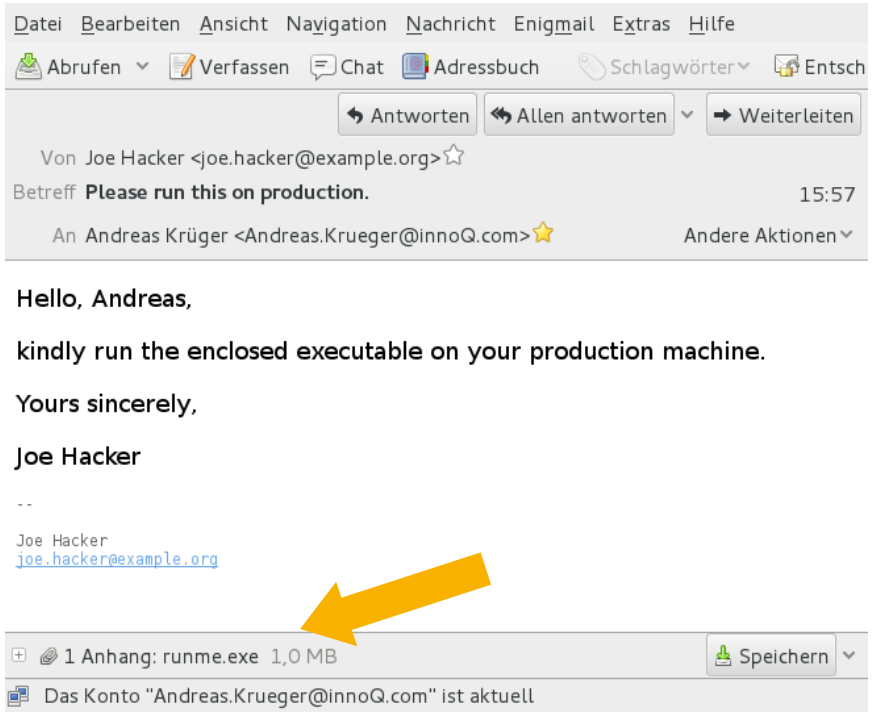
innoQ

Nachvollziehbare Repositories

- Bei „append only” - Repositories
  jede Hinzufügungszeit protokollieren
- Repositories mit sich änderndem Inhalt
  versionieren
- „Das Internet”
  ist kein nachvollziehbares Repository
  (Ausnahmen bei Verträgen)

Signierte Artefakte



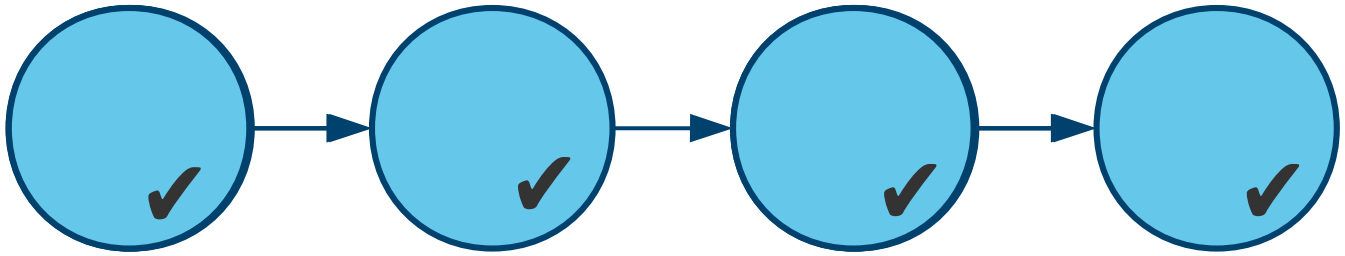


Fälschungssicherheit durch Kryptographie.

schwer

innoQ

Signierter Code



Checkins signieren

Manipulationsschutz durch Kryptographie.

einfach

innoQ

Vielen Dank!

Dr. Andreas Krüger

andreas.krueger@innoQ.com

GPG-Key Fingerabdruck
9FA7 EDDE 320B 3B35 006C 2075 BC0D 1E82 2DC1 BE5B

Bonusmaterial

innoQ

Checkins signieren mit git und Signaturen überprüfen

- anfangs (fast) nicht dokumentiert

```
git commit -S
git log --show-signature
git show -s –show-signature
```

- Überprüfung automatisieren!

```
git verify-commit
```

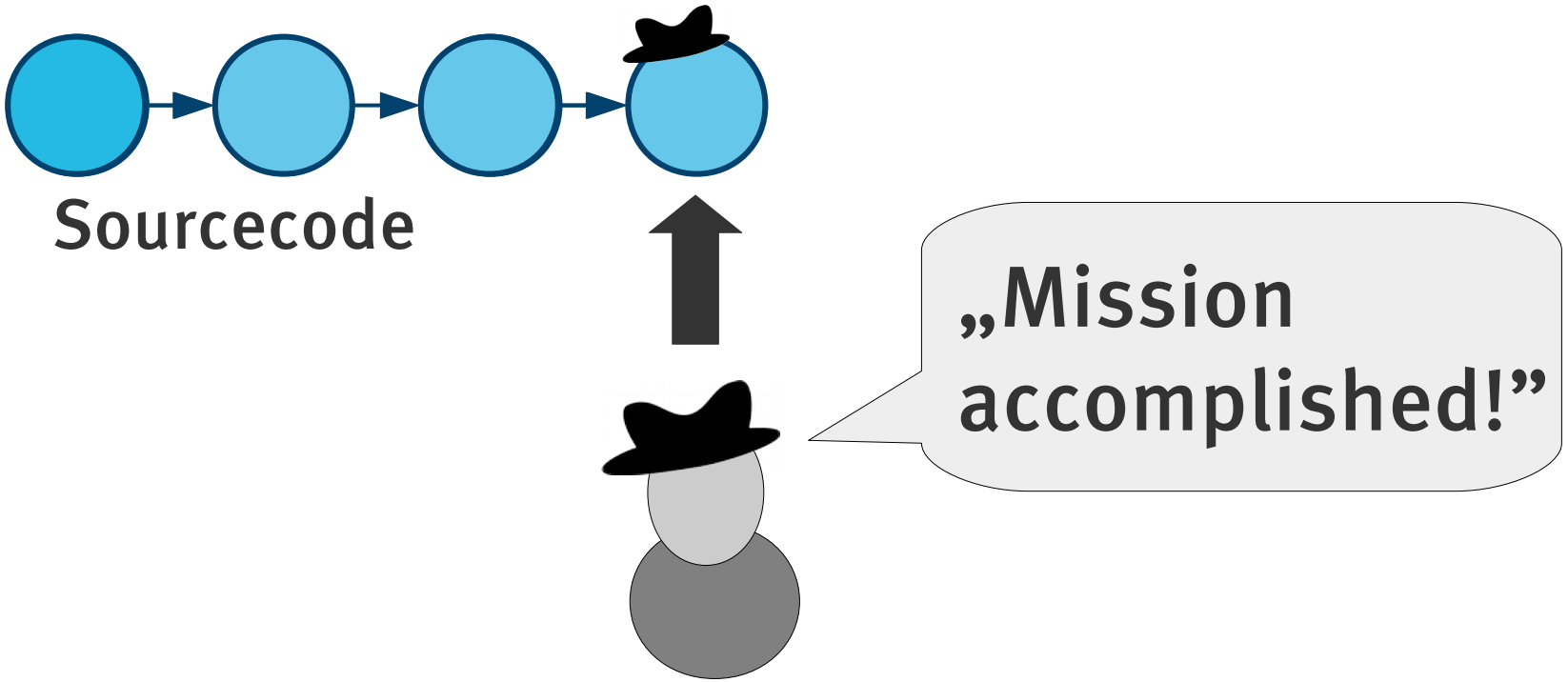
Pro: http://mikegerwitz.com/papers/git-horror-story

(Veraltetes?) Contra von Linus, „besser Tags unterschreiben”:

http://git.661346.n2.nabble.com/GPG-signing-for-git-commit-td2582986.html#a2583316

innoQ

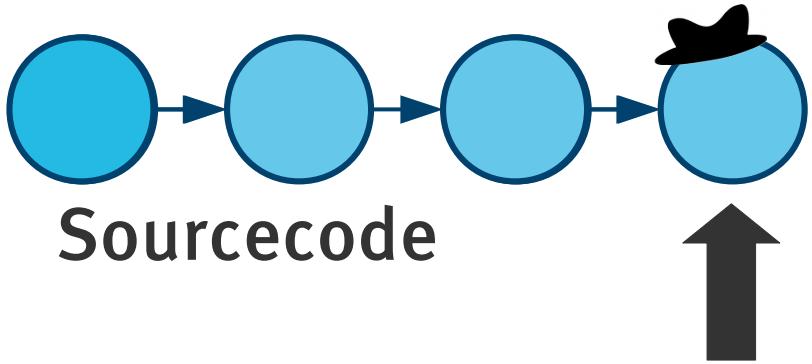
Hintertür im Code



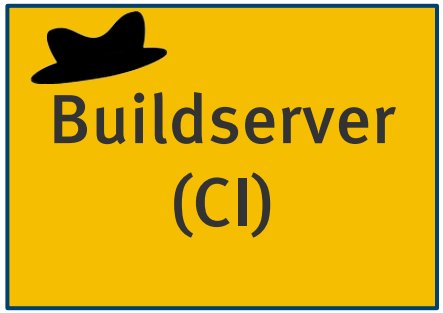
- Signed Check-Ins, automatisiert prüfen
- Code Reviews flächendeckend
- (“Rewriting of history” einschränken)

innoQ

Hintertür in den Code via Buildserver



Funktionsuser für Tags?



Besser: Buildserver liest den Sourcecode nur.
Versionsinfo in die Artefakte einbetten statt Tags.

innoQ