



AutoInformations- und Konfigurationservice

Motivation

Heutige Rechnernetze beinhalten heterogene Geräte mit diversen uneinheitlichen Schnittstellen. Aus der Architektur des Internet ergibt sich, dass *Informationen* in den Endgeräten gespeichert sind und üblicher Weise nicht über das Netz propagiert werden. *Umkonfigurationen* müssen ebenfalls lokal vorgenommen werden.

Es steckt also viel Wissen in den Endsystemen und wenig Wissen im Netz.

Bei der heute vorhandenen Netzbandbreite und vor allem auch aus Usability-Gesichtspunkten ist es wünschenswert, Informationen über im Netz befindliche Komponenten sammeln zu können und die Netzkomponenten auf sichere Weise fernkonfigurieren zu können.

Aufgabenstellung

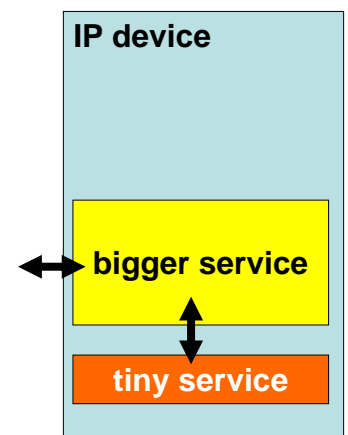
In dieser Arbeit sollen für ein Linuxsystem die Komponenten entwickelt werden, die benötigt werden, um relevante Informationen über das System zu erlangen. Diese Komponenten sollen ebenfalls ermöglichen, das System aus der Ferne zu veranlassen, sich umzukonfigurieren (bspw. IPSec-Tunnel aufzubauen, anderen WLAN-Accesspoints beizutreten etc.).

Die Softwarearchitektur soll dabei aus zwei Teilen bestehen:

- Einem lokal laufenden Service in C, der die konkreten Informationen über das System liefert und Umkonfigurationen umsetzt („tiny service“).
- Einem Java-Service, der Wissensinstanzen im Netz Informationen über das lokale Gerät und dessen Konfiguration kommuniziert sowie (Um-)Konfigurationsanweisungen durchsetzt.

In der Arbeit ist zu recherchieren, welche Verfahren es zum Erlangen des Zieles schon gibt und welche Vor- und Nachteile die jeweiligen Ansätze haben.

Es ist ein geeignetes Protokoll zum Austausch der Wissensinformation zu identifizieren oder zu spezifizieren.



Für den „tiny service“ ist zu identifizieren, welche Informationen wie erlangt werden können und welche Informationen relevant sind. Der Service ist mitsamt seiner Schnittstellen geeignet zu spezifizieren.

Der „bigger service“ sollte vom konkreten Betriebssystem abstrahieren und eine einheitliche Schnittstelle zum Netzwerk zur Verfügung stellen.

Voraussetzungen

Erfahrung in Java und C sowie mit Linux.
Spaß an eigener Recherche und Teamarbeit im größeren Gesamtprojekt.

Stichworte

Knowledge Plane, Autokonfiguration, Linux

