Danilo Bargen

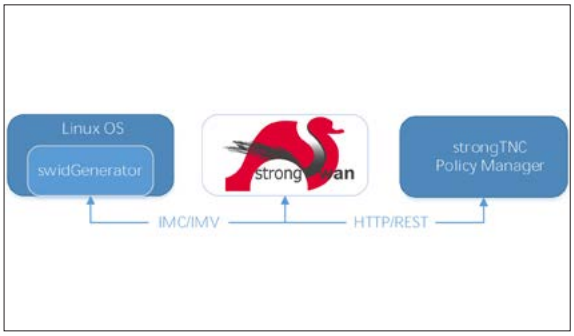
Christian Fässler

Jonas Furrer

Diplomanden	Danilo Bargen, Christian Fässler, Jonas Furrer
Examinator	Prof. Dr. Andreas Steffen
Experte	Dr. Ralf Hauser, PrivaSphere AG, Zürich
Themengebiet	Sicherheit

Endpoint Compliance Monitoring based on Software Identification Tags

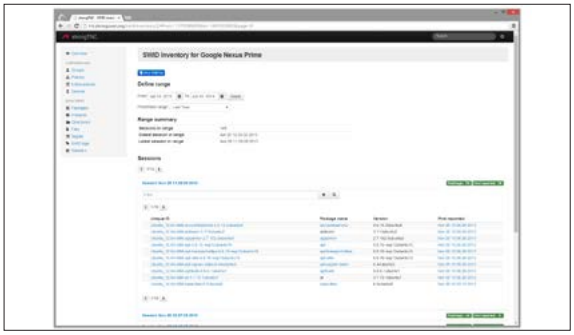
Integration von Software Identification Tags in den strongTNC Policy Manager und Entwicklung einer Clientkomponente zur Generierung von SWID Tags auf Linux-Systemen



Zusammenspiel der entwickelten Komponenten

```
<?xml version="1.0" encoding="UTF-8"?>
<SoftwareIdentity name="strongswan"
        uniqueId="debian_7.4-x86_64-strongswan-4.5.2-1.5+deb7u3"
        version="4.5.2-1.5+deb7u3" versionScheme="alphanumeric"
        xmlns="http://standards.iso.org/iso/19770/-2/2014/schema.xsd">
  <Entity name="strongSwan" regid="regid.2004-03.org.strongswan" role="tagCreator"/>
  <Entity name="HSR" regid="regid.2004-03.org.strongswan" role="publisher"/>
  <Payload>
    <File location="/usr/sbin" name="charon-cmd"/>
    <File location="/usr/sbin" name="strongswan"/>
    <File location="/usr/lib/systemd/system" name="strongswan.service"/>
    <File location="/usr/lib64/strongswan" name="libcharon.so.0"/>
    <File location="/usr/lib64/strongswan" name="libcharon.so.0.0.0"/>
    <File location="/usr/lib64/strongswan" name="libstrongswan.so.0"/>
    <File location="/usr/lib64/strongswan" name="libstrongswan.so.0.0.0"/>
    <File location="/usr/lib64/strongswan" name="libtls.so.0"/>
    <File location="/usr/lib64/strongswan" name="libtls.so.0.0.0"/>
    <File location="/usr/share/doc/strongswan-5.1.2" name="README"/>
  </Payload>
</SoftwareIdentity>
```

Beispiel eines SWID XML Tags



Ansicht des SWID-Tag-Inventars eines Gerätes im strongTNC Policy Manager

Ausgangslage: Das amerikanische National Cybersecurity Center of Excel lence schlägt in einem Entwurfsdokument vor, durch eine kontinuierliche Überwachung der installierten Software auf Client-Systemen (Desktop PCs, Laptops, Tablets, Smartphones etc.) die Gefahr von Cyberattacken zu minimieren. Das Software Asset Management soll über Software Identification (SWID) Tags erfolgen, welche durch die Norm ISO/IEC 19770-2: 2009 international standardisiert sind. Die Trusted Computing Group (TCG) hat ein offenes Framework für das aktive Monitoring von Endgeräten mit dem Namen Trusted Network Connect (TNC) entwickelt. In einer Diplomarbeit an der HSR wurde bereits eine erste Version eines TNC-Policy-Managers implementiert, dieser soll nun für das Software Asset-Management mittels SWID Tags erweitert werden. Zusätzlich soll eine Client-Komponente für die Linux-Distributionen Debian und Ubuntu entwickelt werden, welche aus den Informationen der Paketverwaltung SWID Tags generiert.

Vorgehen/Technologien: Der SWID-Generator wurde vollständig in Python entwickelt. Es sollen keinerlei Abhängigkeiten zu externen Bibliotheken bestehen. So ist sicherzustellen, dass der Client problemlos zusammen mit dem strongSwan VPN Client eingesetzt werden kann. Die Analyse des bestehenden strongTNC Policy Manager hat gezeigt, dass eine enge Kopplung mit der strongSwan-VPN-Lösung besteht. Zudem weist der bestehende Code Qualitätsmängel auf. Diese beiden Punkte wurden neben der Integration der SWID-Tag-Verwaltung zum zentralen Bestandteil dieser Arbeit.

Ergebnis: Der SWID-Generator implementiert die Unterstützung für die drei weit verbreiteten Paketverwaltungssysteme DPKG, RPM und Pacman. Die Architektur wurde modular ausgelegt, so dass die Unterstützung für weitere Paketverwaltungen einfach hinzugefügt werden kann. Um die Verteilung auf Clientsystemen zu erleichtern, wurde der Generator in den Python-Package-Index aufgenommen und kann so mit einem einzigen Kommandozeilenbefehl installiert werden. Durch Refactoring, statische Codeanalyse, Continuous Integration und zusätzliche Integrations- und Unittests konnte die Codequalität von strongTNC messbar verbessert werden. Zur Entkopplung des strongTNC Policy Manager von den Umsystemen wurde ein Konzept ausgearbeitet, welches eine serviceorientierte Architektur mit einer REST API vorsieht. Dieses wurde bereits erfolgreich für die neu integrierten Komponenten umgesetzt. Das API-Konzept soll nun als Vorschlag zur Umsetzung einer generischen TNC-Schnittstelle der TCG unterbreitet werden.