GPLMT: A Lightweight Response Execution Framework

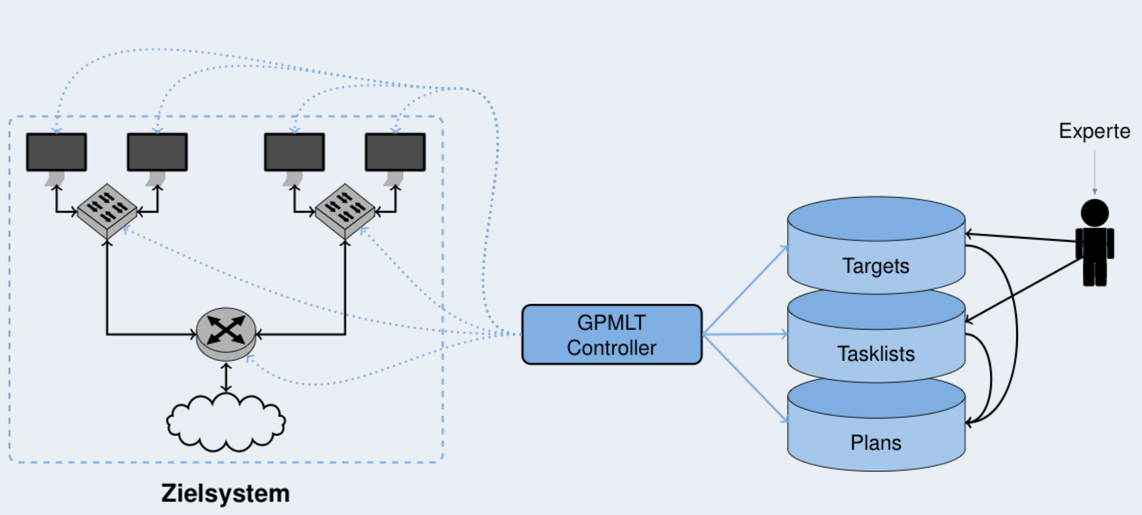
GEFÖRDERT VOM Bundesministerium für Bildung und Forschung

Technische Universität München

SURF – Systemic Security for Critical Infrastructures

Abschlussmeeting und Demoday für das Projekt SURF
08.12.2016, Infineon Technologies AG, Am Campeon 1-12, Neubiberg

System Architektur - Überblick



Datenbeschreibung

Infrastrukturinformationen werden in sog. *Targets* abgebildet. Ein Target beschreibt hierbei eine Infrastrukturkomponente, welche Reaktionen ausführen kann.

Reaktionen werden durch sog. *Tasklisten* modelliert. Eine Taskliste enthält hierbei Anweisungen für eine konkrete Aktion.

Sowohl Targets als auch Tasklisten werden vom Administrator vorab angelegt.
Zur Laufzeit werden aus diesen Informationen Reaktionspläne in Abhängigkeit von der Auswahl der Reaktionen generiert.

Kommunikation und Steuerung

Die Steuerung des Kontrollflusses wird vom *GMPLT Controller* übernommen.
Dieser liest den spezifizierten Ausführungsplan ein und plant die Ausführung der einzelnen Aktionen.

Auf den zu steuernden Komponenten muss keine zusätzliche Software installiert werden, da der Controller auf einen direkten Kommunikationskanal, wie z.B. *ssh*, aufsetzt.

Der Controller kann zudem als zentrales *Aktions-Repository* verwendet werden.
Von diesem Repository können Aktionen direkt auf die entsprechenden Komponenten verteilt werden.

Beispiel für einen Ausführungsplan

```
1 ...
2 <steps>
3 <step tasklist="tasklist1" targets="node1" />
4 <step tasklist="tasklist2" targets="node2" />
5 <register-teardown tasklist="teardown-tasklist"
   targets="node1" />
6 <synchronize targets="node1" />
7 <step tasklist="tasklist3" targets="node1 node2" />
8 <synchronize />
9 <step tasklist="tasklist4" targets="node3" />
10 </steps>
11 …
```

Möglichkeiten der Ausführungssteuerung

Eine parallele Ausführung kann mittels `steps` Definitionen erreicht werden.
Eine sequentielle Ausführung wird durch ein explizites `synchronize` Statement erreicht.

Schleifen werden mithilfe der `repeat` Definitionen umgesetzt.
Hierbei sind folgende Varianten unterstützt:

- **Anzahl:** `iterations`
- **Zeitbasiert:** `during` **und** `until`
- **Variablen:** `listings`

Nachgelagerte Aktionen können mittels des `register-teardown` Statement angelegt werden.
Eine weitere Möglichkeit zur Zeitsteuerung sind die Angabe von `start` und `stop` Zeiten.

Dokumentation und Code

Download GPLMT:
https://github.com/docmalloc/gplmt/

Dokumentation:
http://gplmt.readthedocs.io/en/latest/

Veröffentlichung:
Wachs, Matthias, et al. "GPLMT: A Lightweight Experimentation and Testbed Management Framework." International Conference on Passive and Active Network Measurement. Springer International Publishing, 2016.

Projekteckdaten

Laufzeit 09/2014 – 08/2016 (12/2016)

Gesamtvolumen ca. 4,19 Mio. €

Bereich IT-Sicherheit – Kritische Infrastrukturen

Ziel Entwicklung einer Ganzheitlichen Lösung zur Verbesserung der Schutzsysteme für KRITIS

Konsortium

AIRBUS GROUP, infineon, Fraunhofer SIT, HIRSCHMANN A BELDEN BRAND, Technische Universität München, TECHNISCHE HOCHSCHULE DEGGENDORF

Kontaktinformationen - TUM

Webseite https://www.net.in.tum.de/html/surf/

Verbundkoordination Infineon Technologies AG

Kontaktadressen

Prof. Dr.-Ing. Georg Carle
+49 89 289 18030
carle@Net.in.tum.de

Dr. Holger Kinkelin
+49 89 289 18006
kinkelin@net.in.tum.de