

Flurin Arquint

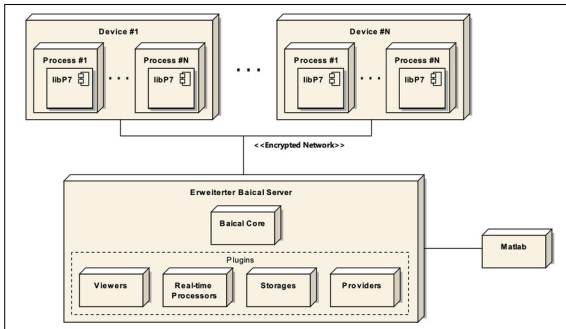


Jan Brupbacher

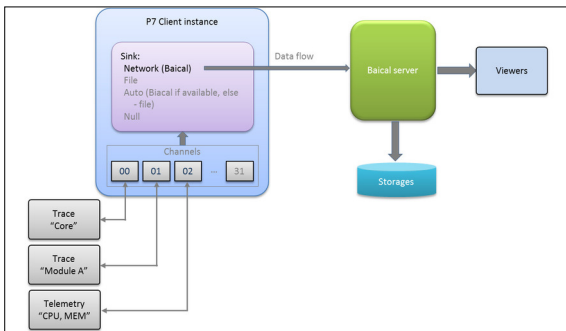
Studenten	Flurin Arquint, Jan Brupbacher
Examinator	Prof. Reto Bonderer
Themengebiet	Embedded Software Engineering

Erweiterter Baical Server

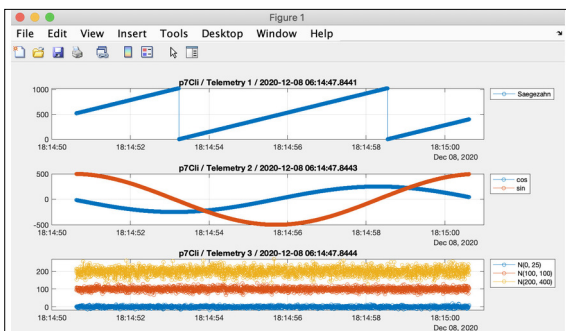
Baical Logging System erweitern mit Matlab-Endpoint und Verschlüsselung



Mögliche Verteilung der Komponenten des erweiterten Baical Systems
Eigene Darstellung



Überblick über die Komponenten der P7 Bibliothek und dem Datenfluss
<http://baical.net/p7.html>



Mit Erweiterung eingeleasene Telemetrien dargestellt in Matlab R2020b
Eigene Darstellung

Einleitung: Das Baical System ist ein hochleistungsfähiges Toolset, mit welchem man Telemetrien (z.B. Sensordaten) und Trace (Log) Daten senden und empfangen kann. Es besteht aus dem Baical Server und der P7 Bibliothek, wobei beide Open Source Softwarekomponenten sind. Dieses bestehende System galt es in dieser Arbeit mit folgenden nützlichen Erweiterungen auszubauen:

- Verschlüsselung der Daten, welche von einer Applikation mittels der P7 Bibliothek an den Baical Server gesendet werden.
- Anbindung an Matlab, damit die an den Server gesendeten Telemetrien direkt in Matlab verarbeitet werden können.
- Automatisches Reconnect nach Verbindungsunterbruch und Zwischenspeichern der Daten auf Seite des Senders.
- Synchronisation der Telemetrien, damit diese einheitliche Zeitstempel aufweisen.

Vorgehen: Zu Beginn der Arbeit stand eine umfassende Analyse des bestehenden Baical Systems und die Erstellung eines Pflichtenheftes sowie eines Projektplans an. Folgend wurden unbekannte Bestandteile von Netzwerk, Matlab und Verschlüsselung näher betrachtet und mögliche Konzepte für deren Verwendung evaluiert und ausprobiert. Weiter wurden die ausgearbeiteten Konzepte implementiert und in sich getestet.

Durch die relativ starken Unabhängigkeiten der einzelnen Erweiterungen war eine saubere Aufteilung der Arbeit gut möglich. Daher wurden diese alle getrennt entwickelt und am Schluss zusammengeführt.

Ergebnis: Bei der Verschlüsselung entschied man sich für die Verwendung vom DTLs Protokoll, da dies eine komfortable Basis für eine Verschlüsselung über das verbindungslose Transportprotokoll UDP (welches vom Baical verwendet wird) bietet. AES wurde dabei als Algorithmus gewählt und alles mithilfe der Embedded Library wolfSSL implementiert.

Für die Matlab-Anbindung wurden Matlab Executables (MEX) geschrieben, welche über TCP und UDP mit einem selbst entwickelten Server-Plugin kommunizieren. Diese ermöglichen in Matlab den direkten Zugriff auf Telemetrien, welche auf dem Baical Server vorhanden sind. Werden zur Laufzeit weitere Telemetriedaten an den Server gesendet, können diese auch für eine Zeiddauer aufgenommen oder live in Matlab dargestellt werden.

Mittels Matlab Timetables konnte die Zeitsynchronisation von Telemetrien elegant ermöglicht werden. Diese erlauben es, verschiedene Telemetrien mit den gängigen Interpolationsalgorithmen zu synchronisieren.