

# Sommerprüfung 2025

## **Ausbildungsberuf**

Fachinformatiker/Fachinformatikerin (VO 2020) Fachrichtung:  
Anwendungsentwicklung

## **Prüfungsbezirk**

Fachinformatiker/-in Anwendungsentwicklung (AP T2V1)

Adam Rigely

Identnummer: [IDENTNUMMER]

E-Mail: [EMAIL], Telefon: [TELNUMMER]

Ausbildungsbetrieb: [AUSBILDUNGSBETRIEB]

Projektbetreuer: [NAME]

E-Mail: [EMAIL],

Telefon: [TELNUMMER]

## **Thema der Projektarbeit**

Die Entwicklung eines Wireshark-Dissectors für das  
BAOS-Kommunikationsprotokoll.

# 1 Thema der Projektarbeit

Die Entwicklung eines Wireshark-Dissectors für das BAOS-Kommunikationsprotokoll.

# 2 Geplanter Bearbeitungszeitraum

Beginn: 24.02.2025

Ende: 21.03.2025

# 3 Ausgangssituation

Das Unternehmen [FIRMENNAME] ist in der Gebäudeautomation tätig. Zu der Automatisierung von Gebäuden werden Zonenregler angeboten, die verschiedene Kommunikationsprotokolle, wie z.B. BACnet und Modbus, unterstützen. Um das Kommunikationsprotokoll KNX zu unterstützen, wird das von dem Unternehmen Weinzierl entwickelte, abstrahierte Protokoll BAOS (Bus Access and Object Server) und das Weinzierl KNX BAOS Modul in den Zonenreglern implementiert. Das BAOS-Modul wird als eine Art Dolmetscher zwischen dem Zonenregler und dem KNX-Bus fungieren. Der Zonenregler sendet die Befehle als BAOS-Telegramme an das BAOS-Modul, die von dem Modul interpretiert und anschließend als KNX-Telegramme an den KNX-Bus weitergeleitet werden. Die Kommunikation ist bidirektional, daher werden die an den Zonenregler adressierte KNX-Telegramme von dem BAOS-Modul an die entsprechende Adresse als BAOS-Telegramme weitergeleitet. Somit ist es nicht nötig, in dem Zonenregler das komplizierte KNX-Kommunikationsprotokoll zu implementieren. Damit Fehler und unerwünschte Situationen vermieden werden ist eine Lösung für die effektive Überwachung der Kommunikation über das BAOS-Protokoll gewünscht.

Das BAOS-Protokoll wird zwar nach Vorgaben in den eigenen Regler-Modulen implementiert, allerdings sind noch einige Anpassungen erforderlich, damit dieses Kommunikationsprotokoll in den Geräten verwendet werden kann. Zurzeit läuft das BAOS-Protokoll auf einem gewöhnlichen Entwickler-PC, unter dem Betriebssystem Windows 11, der mit einem Weinzierl KNX BAOS Modul über eine USB-Schnittstelle verbunden ist. Im Rahmen der Entwicklung werden einfache KNX-Fähige Geräte, wie z.B. Taster, von dem PC über das BAOS-Modul kontrolliert. Um die Kommunikation zu überwachen wird das Netzwerkanalyse-Tool Wireshark verwendet. Da die Interpretierung des BAOS-Protokolls in Wireshark zurzeit nicht unterstützt wird (Stand in der Version 4.4.2), wird der BAOS-Payload des USB-Paketes nicht zerlegt, sondern als eine Menge reiner, unstrukturierter hexadezimaler Werte dargestellt.

Es ist zwar möglich, während der Überwachung der Kommunikation die einzelnen Bytes der BAOS-Telegramme jedes Mal händisch zu zerlegen, dieses Vorgehen nimmt jedoch, selbstverständlich, unnötig viel Zeit in Anspruch. Um dies zu ändern ist es wünschenswert, Wireshark um entsprechende Fähigkeiten zu erweitern. In dem Projekt sollen die folgenden wesentlichen funktionalen Anforderungen erfüllt werden:

- 1) Wireshark soll in der Lage sein, ein BAOS-Telegramm als eingebetteten Payload in einem USB-Paket erkennen zu können.
- 2) Wireshark soll in der Lage sein, das erkannte BAOS-Telegramm zu zerlegen und die einzelnen wesentlichen Teile strukturiert verarbeiten zu können.
- 3) Wireshark soll in der Lage sein, das verarbeitete BAOS-Telegramm strukturiert und sinnvoll als eingekapselten Teil eines USB-Paketes, darstellen zu können.

## 4 Projektziel

Im Rahmen des innerbetrieblichen Projektes wird ein Wireshark Dissector entwickelt um Wireshark in die Lage zu versetzen, den oben genannten funktionalen Anforderungen entsprechen zu können. Die zwei wesentlichen Programmiersprachen, die von dem Wireshark-Projekt unterstützt werden, sind die Skriptsprache Lua, und die klassische Programmiersprache C. Nach Erwägung der Vorteile und Nachteile der beiden Programmiersprachen, wurde die Skriptsprache Lua für die Entwicklung des Dissectors gewählt. Die wichtigsten Faktoren, die bei der Auswahl in Betracht gezogen wurden, sind:

- 1) In Anbetracht der einfachen Syntax von Lua dauert die Entwicklung kürzer als im Falle von C.
- 2) Die einfachen Lua-Schnittstellen zu Wireshark verkürzen die Entwicklungsdauer.
- 3) Die Integration eines Lua-Plugins in Wireshark ist einfacher und nimmt weniger Aufwand in Anspruch als eines C-Plugins.
- 4) Da das Lua-Plugin nicht kompiliert wird, wird das iterative Testen während der Entwicklung schneller als eines C-Plugins.

Die effektive Entwicklung des Dissectors bedarf einer ordentlich eingerichteter und konfigurierter Entwicklungsumgebung. Auf dem Entwicklungs-PC, unter Windows 11, wird ein Lua-Interpreter konfiguriert, damit gewisse Lua-Funktionen, unabhängig von Wireshark, getestet werden können. Da für die Entwicklung des Dissectors eine vollständige IDE nicht nötig wird, bietet der Code-Editor Microsoft Visual Studio Code das beste Funktionalität-Leistungsverhältnis, und dementsprechend wird der bei der Entwicklung genutzt.

Die Wireshark-Dokumentation und das Lua-API-Referenzhandbuch werden studiert. Es wird recherchiert wie der Dissector an Wireshark angeschlossen wird und welche Lua-API-Funktionen sich anbieten. Ein Plan wird entworfen, wie der Dissector, nach der Kommunikationsprotokollbeschreibung in der BAOS-Dokumentation, entwickelt werden soll. Nach den Vorbereitungen wird die tatsächliche Entwicklung des Dissectors durchgeführt.

Anschließend wird getestet, ob die BAOS-Telegramme erwartungsgemäß erkannt, interpretiert, zerlegt und strukturiert dargestellt werden können.

Nach der Testphase wird Wireshark, mit dem neuen Dissector, innerbetrieblich eingesetzt. Zu dem Endprodukt und dem Projekt werden Dokumentationen geschrieben, und über die Erfahrungen und gewonnenen Kenntnissen wird reflektiert.

## 5 Zeitplanung

In der ersten Phase des Projektes werden Vorbereitungen für das Projekt durchgeführt. Als Erstes werden die Anforderungen analysiert. Danach wird eine angemessene Entwicklungsumgebung eingerichtet. Anschließend wird eine Recherche zur Entwicklung und Implementierung eines Wireshark-Dissectors durchgeführt. Diese Projektphase wird voraussichtlich 8 Stunden in Anspruch nehmen.

In der zweiten Phase des Projektes werden Konzepte entworfen. Als Erstes wird ein UML-Aktivitätsdiagramm für den Dissector erstellt. Anschließend wird ein Mockup ebenso gezeichnet, was darstellt, wie die zerlegten Telegramme in Wireshark präsentiert werden sollen. Diese Projektphase wird voraussichtlich 8 Stunden in Anspruch nehmen.

In der dritten Phase des Projektes wird die tatsächliche Entwicklung und Implementierung des Dissectors als ein Lua-Plugin durchgeführt. In dieser Projektphase wird die Zerlegung sowohl des FT 1.2-Frames, als auch des BAOS-Telegramms in diesem Frame programmiert. Bei der Programmierung der Zerlegung des BAOS-Telegramms werden alle BAOS-Subdienste berücksichtigt. Diese haben einzigartige Strukturen, die dementsprechend einzigartiger Zerlegung bedürfen. Diese Projektphase wird die meiste Zeit, voraussichtlich 40 Stunden in Anspruch nehmen.

In der vierten Phase des Projektes wird Wireshark mit dem neuen Dissector gründlich getestet. Es wird getestet ob die von dem BAOS-Modul an den PC, und ob die von dem PC an das BAOS-Modul gesendeten BAOS-Telegramme in Wireshark erwartungsgemäß erkannt, interpretiert, zerlegt und strukturiert dargestellt werden können. Diese Projektphase wird voraussichtlich 12 Stunden in Anspruch nehmen.

Schließlich, in der fünften Phase des Projektes werden die Produkt- und die Projektdokumentation geschrieben. Diese Projektphase wird voraussichtlich 12 Stunden in Anspruch nehmen.

Das Gesamtprojekt wird voraussichtlich 80 Stunden betragen.

## 6 Anlagen

keine

## 7 Präsentationsmittel

Für die Präsentation werden ein Laptop und ein Beamer nötig sein.

## 8 Hinweis!

Ich bestätige, dass der Projektantrag dem Ausbildungsbetrieb vorgelegt und vom Ausbildenden genehmigt wurde. Der Projektantrag enthält keine Betriebsgeheimnisse. Soweit diese für die Antragstellung notwendig sind, wurden nach Rücksprache mit dem Ausbildenden die entsprechenden Stellen unkenntlich gemacht.

Mit dem Absenden des Projektantrages bestätige ich weiterhin, dass der Antrag eigenständig von mir angefertigt wurde. Ferner sichere ich zu, dass im Projektantrag personenbezogene Daten (d. h. Daten über die eine Person identifizierbar oder bestimmbar ist) nur verwendet werden, wenn die betroffene Person hierin eingewilligt hat.

Bei meiner ersten Anmeldung im Online-Portal wurde ich darauf hingewiesen, dass meine Arbeit bei Täuschungshandlungen bzw. Ordnungsverstößen mit „null“ Punkten bewertet werden kann. Ich bin weiter darüber aufgeklärt worden, dass dies auch dann gilt, wenn festgestellt wird, dass meine Arbeit im Ganzen oder zu Teilen mit der eines anderen Prüfungsteilnehmers übereinstimmt. Es ist mir bewusst, dass Kontrollen durchgeführt werden.