SAVONIA

THIS IS THE WAY TO CHANGE THE WORLD.

THESIS – BACHELOR'S DEGREE
TECHNOLOGY, COMMUNICATION AND TRANSPORT

ATAK INTEGRATION IN COREMOTE TACTICS

AUTHOR/S Nhan Vo

SAVONIA UNIVERSITY OF APPLIED SCIENCES

THESIS
Abstract

Field of Study Technology, Communication and Transport	
Degree Programme Degree Programme in Incormation Technology, Internet of Things	
Author Nhan Vo	
Title of Thesis ATAK integration in COREMOTE Tactics	
Date 06.12.2024	Pages/Appendices 52
Client Organisation /Partners Mentura Group Oy	

Abstract

This thesis, commissioned by Mentura Group Oy, explored the integration of the Android Team Awareness Kit (ATAK) into the COREMOTE Tactics platform, a comprehensive solution for situational awareness, automatic vehicle location management, fleet management, and dispatching for organizations managing mobile field forces. ATAK was chosen because of its intensive features in critical communication, including real-time positioning, video feed sharing, encrypted data package sharing and messages, its active ecosystem that has been used globally. This study highlighted ATAK's ability to integrate real-time tracking and detailed mapping within a single platform.

The thesis focused on researching and analyzing existing interfaces of both ATAK and Tactics, so that an ATAK integration in COREMOTE Tactics was implemented, which was a real-time location stream from TAK server. This integration is a new software component of Tactics which enhances the efficiency and effectiveness of tactical operations.

The thesis project employed a detailed evaluation of the TAK and Tactics ecosystem, a possibility of integrating TAK server in COREMOTE Tactics, and the usage of several technologies and tools such as OpenLayers, MapServer, MapCache, FeatureServer, and two servers running within two separate Virtual Machines in a secure local network, with firewall configurations managed via Proxmox. Additionally, two adapters were built, following the Tactics component adapter architecture, leveraging Kafka as the event streaming platform to facilitate the connection of the real-time location stream from TAK server to Tactics. This development project not only provided a new software component but also illustrated on going and significant contribution of Mentura in the tactical operations management field.

Keywords

ATAK, COREMOTE Tactics, Kafka, real-time location sharing, situational awareness solutions, automatic vehicle location management

CONTENTS

1 INTRODUCTION 7
2 THESIS GOALS AND DETAILED PLAN 8
3 ATAK 9
3.1 Overview 9
3.2 Ecosystem 9
3.3 Features 10
3.3.1 Placement 10
3.3.2 Range tools 11
3.3.3 Compass interaction 12
3.3.4 Maps and favorites 13
3.3.5 Overlay manager 14
3.3.6 Radio controls 15
3.3.7 Video player 16
3.3.8 Contacts 17
3.3.9 Data package 18
3.4 Development and usage 19
4 COREMOTE TACTICS 20
4.1 Coremote server system architecture 20
4.2 Basic operation 20
4.3 User interface overview 21
4.3.1 Maps 21
4.3.2 Units and fleet management 23
4.3.3 Point of interests 24
4.3.4 Messaging center 26
5 COMPONENT DESIGN REQUIREMENTS 28
5.1 Research questions 28
5.2 Integration basic operation 28
5.3 Integration architecture 29
5.4 Security concepts 33
5.5 Tools and technologies 33
6 INTEGRATION DEVELOPMENT PROCESS 35
6.1 Implementation process 35

6.2 Design and architecture 35
6.3 Development environment setup 37
6.4 Integration development 38
6.4.1 Add integrated server dialog 38
6.4.2 TAK adapters as Tactics components 39
6.4.3 Data visualization 42
6.4.4 TAK fleet and its units 43
6.4.5 Simulated critical mission involving TAK fleet and Tactics fleet 44
6.5 Testing and review 45
6.6 Documentation and training 45
7 FUTURE DIRECTIONS 47
8 DISCUSSION 49
9 CONCLUSION 50
REFERENCES 51

LIST OF FIGURES

Figure 1. Active TAK users on WebTAK map (Vo 2024) 9
Figure 2. CoT events in TAK real-time location stream (Vo 2024) 10
Figure 3. Self-marker overview (Vo 2024) 11
Figure 4. Bullseye example (Vo 2024) 11
Figure 5. Dynamic line and static line example (Vo 2024) 12
Figure 6. First person view direction support (Vo 2024) 12
Figure 7. Kuopio local map layer (Vo 2024) 13
Figure 8. Overlay manager overview (Vo 2024) 14
Figure 9. Elevation data of Nordic (Vo 2024) 15
Figure 10. Default radio channels in ATAK (Vo 2024) 15
Figure 11. TAK ICU overview (Vo 2024) 16
Figure 12. TAK ICU with ATAK (Vo 2024) 16
Figure 13. TAK chat (Vo 2024) 17
Figure 14. Pre-defined messages (Vo 2024) 18
Figure 15. Available data packages on TAK server (Vo 2024) 18
Figure 16. Typical COREMOTE Tactics configuration (Mentura 2020, 10) 20
Figure 17. Tactics viewpoint (Mentura 2023, 11) 21
Figure 18. Tactics UI overview (Mentura 2024b) 21
Figure 19. Base maps with split-screen mode (Mentura 2023, 15) 22
Figure 20. Custom map (Mentura 2024b) 23
Figure 21. Units and fleets branch tree (Mentura 2023, 26) 23
Figure 22. Unit info window (Mentura 2023, 30) 24
Figure 23. POI list (Mentura 2023, 60) 24
Figure 24. POI search (Mentura 2024b) 25
Figure 25. Tactics messaging center (Mentura 2023) 26
Figure 26. ATAK integration configuration (Vo 2024) 28
Figure 27. Tactics server components (Vo 2024) 29
Figure 28. TAK adapters architecture diagram (Vo 2024) 30
Figure 29. Kafka TAK Adapter project structure (Vo 2024) 31
Figure 30. Real time positioning JSON fields (Vo 2024) 32
Figure 31. Integration security concept (Vo 2024) 33
Figure 32. TAK name and server address (Vo 2024) 35
Figure 33. Truststore and client certificate (Vo 2024) 36
Figure 34. Configured TAK server (Vo 2024) 36

Figure 35. Build.xml example (Apache n.d.) ... 37
Figure 36. Add integrated server dialog (Vo 2024) ... 38
Figure 37. Web service method for integrated server creation (Vo 2024) ... 38
Figure 38. Read CoT event (Vo 2024) ... 39
Figure 39. Config properties of Kafka TAK Adapter (Vo 2024) ... 40
Figure 40. Kafka TAK Adapter thread (Vo 2024) ... 40
Figure 41. Config properties of Tactics Kafka Adapter (Vo 2024) ... 41
Figure 42. Method to send request to Tactics backend (Vo 2024) ... 41
Figure 43. Method to send location update to Tactics backend (Vo 2024) ... 42
Figure 44. JSON data stored in Kafka (Vo 2024) ... 43
Figure 45. TAK fleet (Vo 2024) ... 43
Figure 46. Active TAK users on Tactics map (Vo 2024) ... 44
Figure 47. TAK users and Tactics users on the Tactics map (Vo 2024) ... 44
Figure 48. Google API key (Vo 2024) ... 45
Figure 49. Future ATAK integration architecture diagram (Vo 2024) ... 47

1 INTRODUCTION

In an increasingly interconnected and data-driven world, the demand for advanced software solutions that improve situational awareness and tactical operation efficiency has been growing rapidly since the beginning of the 21st century. Before the Covid-19 pandemic, there had been various companies in different regions of the world participating in this market. However, when the pandemic broke out, it is a kind of test to critical communication due to social distancing policies, lockdowns, and mandatory quarantines, which could also be seen as a revolution in critical communication since there is an increase of companies from different regions participating in the market. Organizations involved in critical operations, such as military forces, law enforcement agencies, and emergency services, rely heavily on those complex tools to not only ensure of the effectiveness and efficiency of operations but also minimize human loss in some emergency cases. The Android Team Awareness Kit (ATAK) is one of the tools that provides a powerful geospatial infrastructure and situational awareness platform that is widely used by various militaries and emergency services throughout the world. This Android app is a part of the TAK ecosystem having several products such as WebTAK, WinTAK, and ITAK.

Mentura Group Oy, a leader in providing intelligent software solutions for critical and tactical operations, has developed COREMOTE Tactics, which is a comprehensive platform for situational awareness, tactics operation management solutions (Mentura 2024a). This thesis explores the compatible existing features of both ATAK and Tactics to implement an integration of ATAK into COREMOTE Tactics. The combination of Tactics and ATAK is promising because of their variety of benefits for end users, such as more secure communication, detailed geospatial information of two platforms' users within a single map, a full history of users and groups for management purposes, more flexibility of different maps in different environments including indoor and outdoor, and a generic way for data exchange between TAK server and Tactics server.

This thesis demonstrates how to integrate a third-party server into an existing critical communication platform—COREMOTE Tactics. It specializes in integrating a real-time location stream from the TAK server into COREMOTE Tactics. This thesis includes all research analysis, architecture, security concepts design, implementation, and documentation of the integration.

2 THESIS GOALS AND DETAILED PLAN

The primary objective of this thesis was to research and implement the integration of a TAK server into COREMOTE Tactics. This goal originally started from the discovery of ATAK's intensive features within its Android mobile application, which is called ATAK. Depending on its use, civilian or military, ATAK is referred to as the Android Team Awareness Kit or Advanced Tactical Assault Kit, respectively. Its advanced capabilities, such as real-time positioning, encrypted communication, secured data packages, and situational awareness tools, make it a suitable choice for third-party server integration with COREMOTE Tactics.

Regarding that, several questions were listed as a research base and working base for this software component integration:

- What is a TAK server and what is its ecosystem?

Understand the technical framework and architecture of the TAK server.

- Who are ATAK end users?

Identify user groups, focusing on civilians, who utilize ATAK and find out if there is a business model based on it.

- What is COREMOTE Tactics?

Analyze the COREMOTE Tactics, its core functionalities, and its potential to benefit from ATAK integration.

- How can a Java client program be made to access a real-time positioning data stream?

Develop a technical capability to connect and process live location data from the TAK server.

- How can we connect the TAK server and the COREMOTE Tactics server?

Define a secure and reliable connection between the two systems to enable data flow, specializing in real-time positioning stream.

- How can we store TAK-integrated servers in COREMOTE Tactics?

Design a robust data model in Tactics to manage TAK server configuration, which should have CRUD operations (get, create, update, and delete).

The integration development was managed via Jira, a project management tool that allows developers and managers to keep track of bugs, issues, and priorities in the project. Several Jira tickets were created to manage the workflow of this integration development process:

- Design overall integration with TAK server
- Design data model to store a third-party server integration detail
- Implement Tactics map UI to add TAK servers
- TAK location message handling in backend/TAK adapter
- Implement UI to show TAK users in map UI tree view

3 ATAK

3.1 Overview

ATAK is a part of the TAK ecosystem, which was first released by the Air Force Research Laboratory in 2010 and is now maintained by the TAK Product Center (CivTAK, n.d.). It is an open resource platform published on GitHub (United States Department of Defense 2020) for development and on their own website *tak.gov*. Additionally, ATAK has a plugin architecture allowing developers to add their custom functionalities, which is also seen as a potential market for companies building their own business models. There is a variety of plugins released such as GeoTakCam, DataSync, TAK ICU, Compass Nav, etc.

In general, a TAK client does not require a server if all clients are all within the same subnet. TAK clients use multicast to discover clients within the same subnet. All clients in that network can see each other and share information.

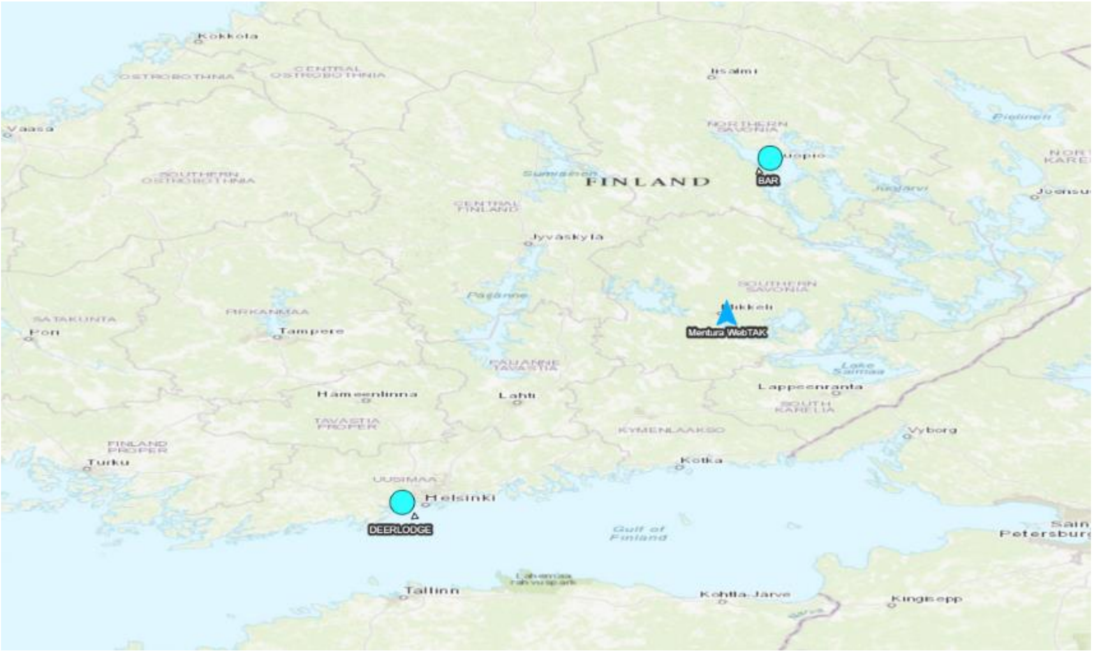

Figure 1. Active TAK users on WebTAK map (Vo 2024)

3.2 Ecosystem

The TAK product family consists of comprehensive tools designed to enhance situational awareness, communication, and coordination across various operational management. The core of this ecosystem is the TAK server, which serves as a central hub for all actions not only between users and devices but also between admins and users. All these TAK-based products have two versions: military and civilian. This thesis focuses on civilian use only.

WebTAK is a browser-based web interface designed to share awareness across multiple end-users (Riskaware 2024). In other words, the operations administrators have an overview of various users in different operations at the same time within a single map, manage all secure data exchange between users and TAK server, create a user’s group for some specific missions, etc.

WinTAK is a Windows-based application that provides full TAK functionalities on the desktop and offers a platform for command centers and operations.

ITAK is a subset of ATAK used in iOS devices (Riskaware 2024). Due to ATAK's subset, it does not support all existing TAK functionalities as ATAK does, but it has most of the core features such as maps and imagery, overlays, chat, video, and situational awareness. Connecting TAK server in ITAK is a bit different from that in ATAK, which only allows an SSL connection while ATAK allows several streaming protocols such as TCP, SSL, and QUIC.

ATAK is an Android app that has all TAK functionalities, and its original purpose was to be built for the military. Users can easily test this out by downloading it from Google Play. Due to the default server configuration setting, an SSL connection under the port of 8089 is used for the Cursor on the Target (CoT) stream, HTTPS 8443 is for browser-web-based interfaces and HTTPS 8446 is for unsecured user credentials. If a network has its own firewall configuration, opening these mentioned ports is extremely crucial for end-user devices connecting to a TAK server.

TAK-X is a specialized version of the TAK application designed to support cross-domain operations, ensuring secure communication and coordination throughout different security domains. It is the culmination of the two most well-known Windows-based command and control platforms: WinTAK and RaptorX. (Riskaware 2024).

TAK Tracker is a lightweight, standalone android application used for sending geospatial information to a TAK server. It also supports sending chat and defined emergency messages. (Riskaware 2024).

3.3 Features

ATAK has various powerful features, varying from different fields, such as location sharing, data sharing, map sharing, etc.

The core feature is its real-time users' locations stream. This stream provides detailed information about an event that occurred by a user, such as an event type encrypted using XML format, EUD (end-user device) information, user's geospatial location, user group, user callsign name, TAK application that is used, and TAK server version. ATAK also supports more secure encryption, which is Google Protobuff (United States Department of Defense 2020) that ensures smaller message size due to its binary format.

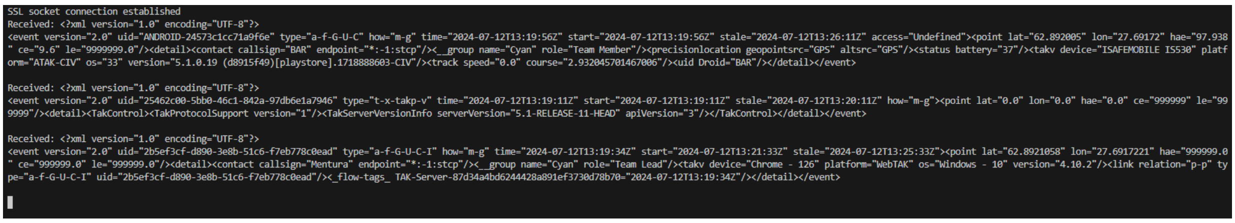

Figure 2. CoT events in TAK real-time location stream (Vo 2024)

3.3.1 Placement

Placement is one of the features of location sharing, which supports users in marking the location of interests on the map. This feature uses a point dropper tool with default missions such as marking a point, observing a point, and adding sensors. Self-marker is the client's location on the map, retrieved from GPS (Global Positioning System), which has several usages, and the most common usage is to measure the distance between the user and point droppers. The distance will be displayed in miles or metric units depending on app settings in the end-user device.

Figure 3. Self-marker overview (Vo 2024)

3.3.2 Range tools

Range tools comprise of all drawing object tools on the map for measuring direction and distance: R&B circle, dynamic measure line, static measure line, and bullseye (TAK Product Center 2024).

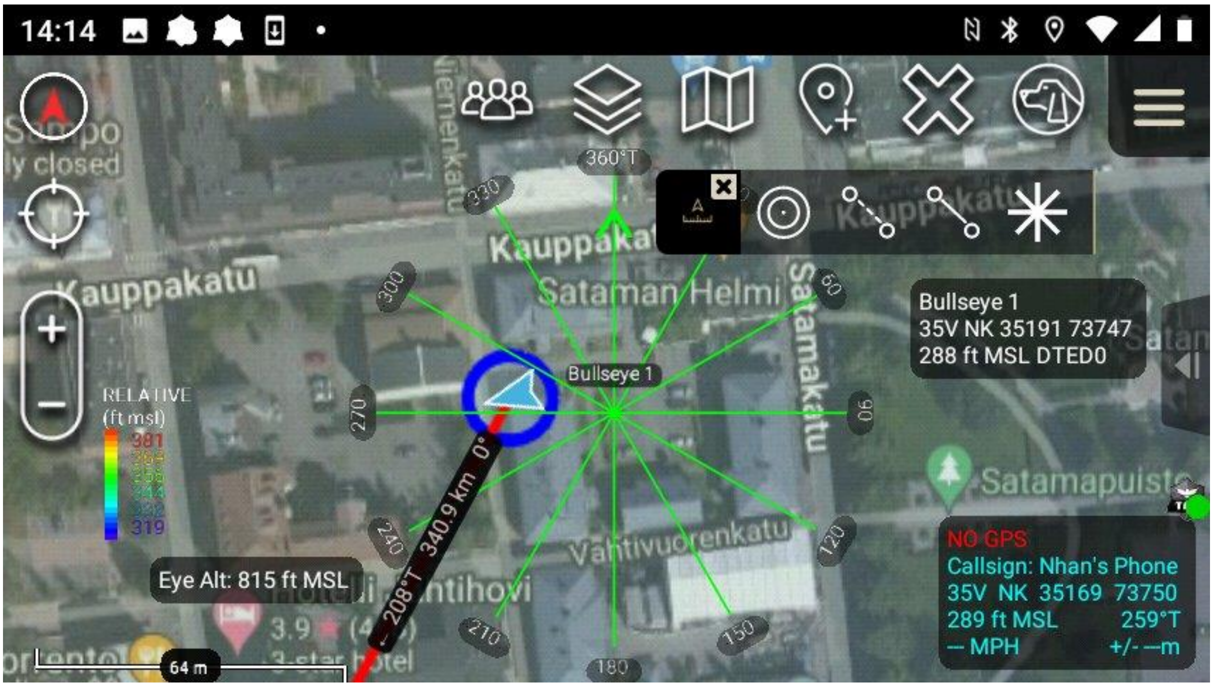

Figure 4. Bullseye example (Vo 2024)

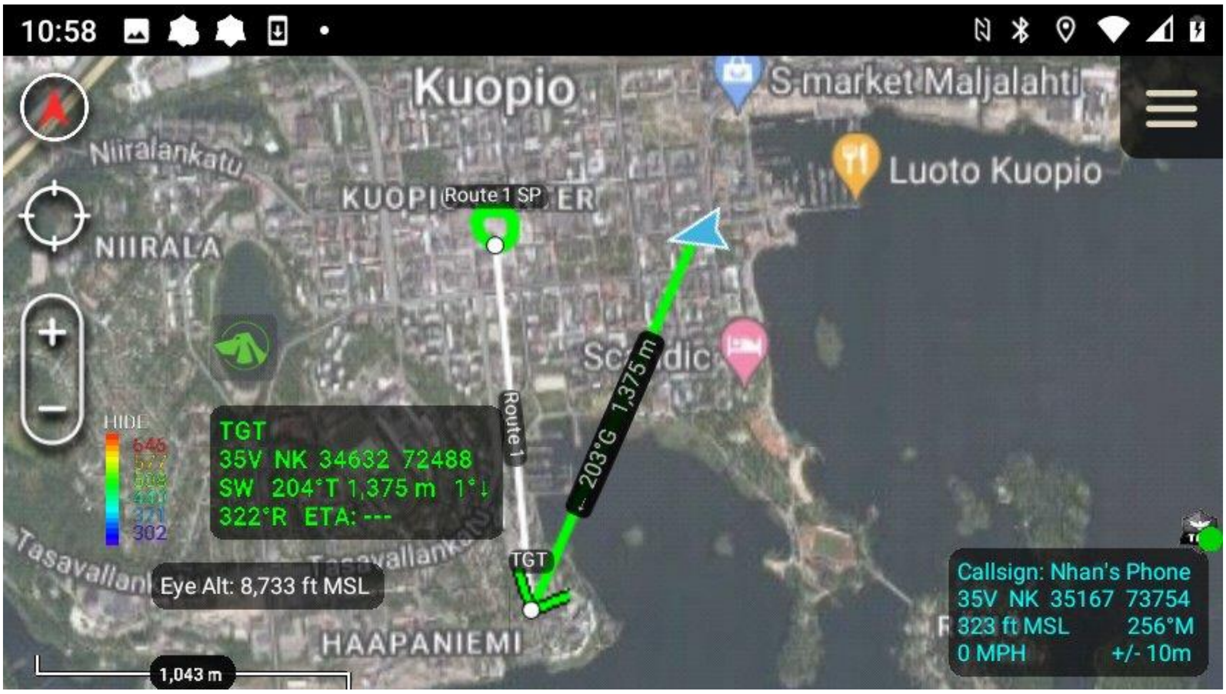

Figure 5. Dynamic line and static line example (Vo 2024)

3.3.3 Compass interaction

Compass interaction is used to control map orientation. ATAK features 3D viewing of terrain and map items (TAK Product Center 2024, 7). First person view is a map view from a first-person perspective simulating the view of a user looking straight ahead towards the horizon rather than from default over-head map view perspective. There are two control buttons for adjusting camera view and subterranean view.

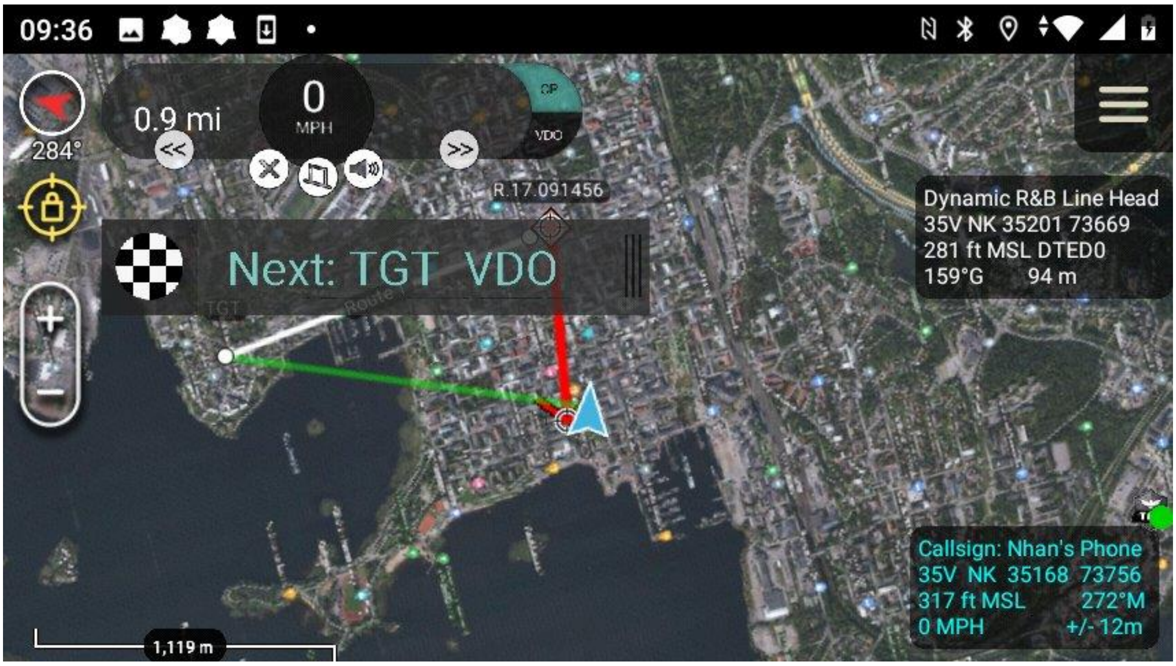

Figure 6. First person view direction support (Vo 2024)

Figure 6 illustrates an overview of first-person view support with several features such as distance measurement between the user's location and route marker location, estimated time to reach the destination, speed visibility, and Google speaker support for navigation purposes. Route is easily imported from a file in different formats such as KML, KMZ, GPX, and Shapefile (TAK Product Center, 2024, 51).

3.3.4 Maps and favorites

This feature provides the behaviors related to maps and favorites within the system, focusing on the available tools for selecting, managing, and utilizing geographic data effectively. It covers a variety of map selection methods and favorites impact in user experience enhancement.

ATAK offers four methods for selecting map areas:

- Rectangle: users define a rectangular area by dragging across the map. This is ideal for quickly selecting large areas.
- Freeform: allows users to draw any shapes on the map, providing flexibility for area selection that do not shape in standard geometric shapes.
- Lasso: enables users to circle an area
- Map select: allows users to select an existing shape or route as their desired download area. It is particularly useful for operations that need to focus on predefined routes, ensuring that users can download relevant map data.

This feature also has Web Feature Service support (TAK Product Center 2024), which represents a change in the way geographic information is created, modified, and exchanged on the internet. Rather than sharing geographic information and the file level in File Transport Protocol (FTP), this offers direct fine-grained access to geographic information.

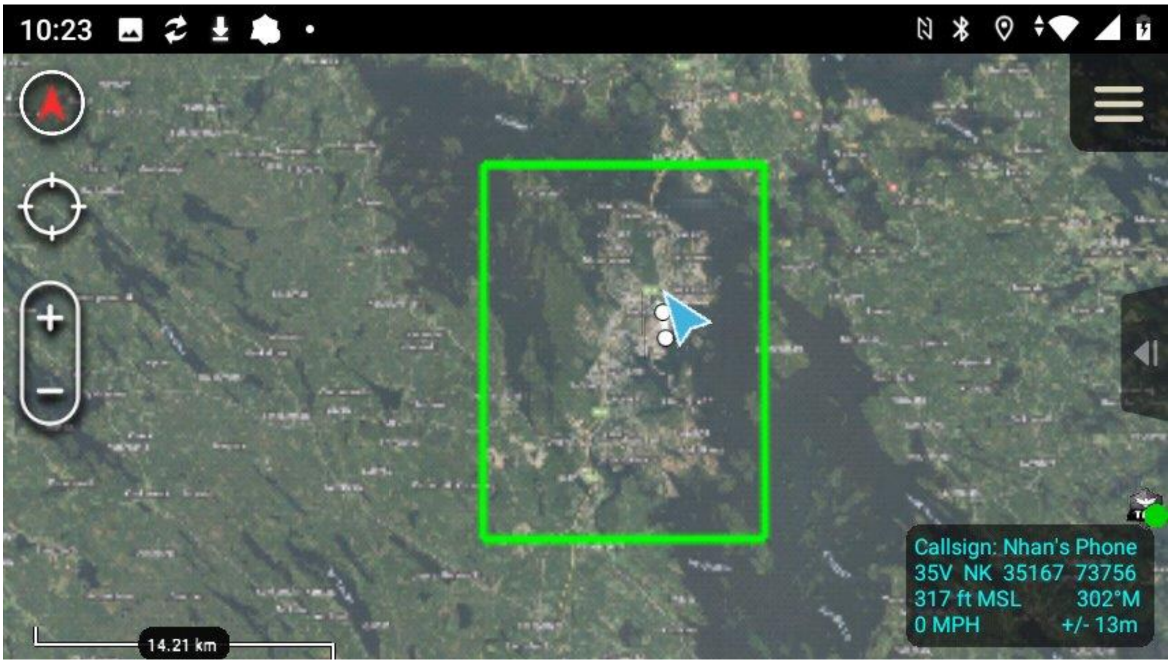

Figure 7. Kuopio local map layer (Vo 2024)

The local map layer feature offers significant advantages, particularly due to its extensive flexibility when internet connection is unavailable. By utilizing this, users can maintain clear visibility of maps within downloaded map layers in some specific areas, ensuring continuous access to critical geographic information even in offline conditions.

3.3.5 Overlay manager

This sorts maps and objects into categories and subcategories. The visibility can be toggled on or off based on needs. (TAK Product Center 2024)

When ATAK receives a 3D model that has been edited by WinTAK, the overlay manager displays the metadata associated with the model along with information included, such as callsign, and terrain model employed (TAK Product Center 2024, 30).

Figure 8. Overlay manager overview (Vo 2024)

The above figure depicts all overlay manager features including managing data packages on the TAK server, navigation to existing routes, market, users, geofences management, map controls, other overlays such as center designator, eye altitude, grid lines, and heatmap overlay.

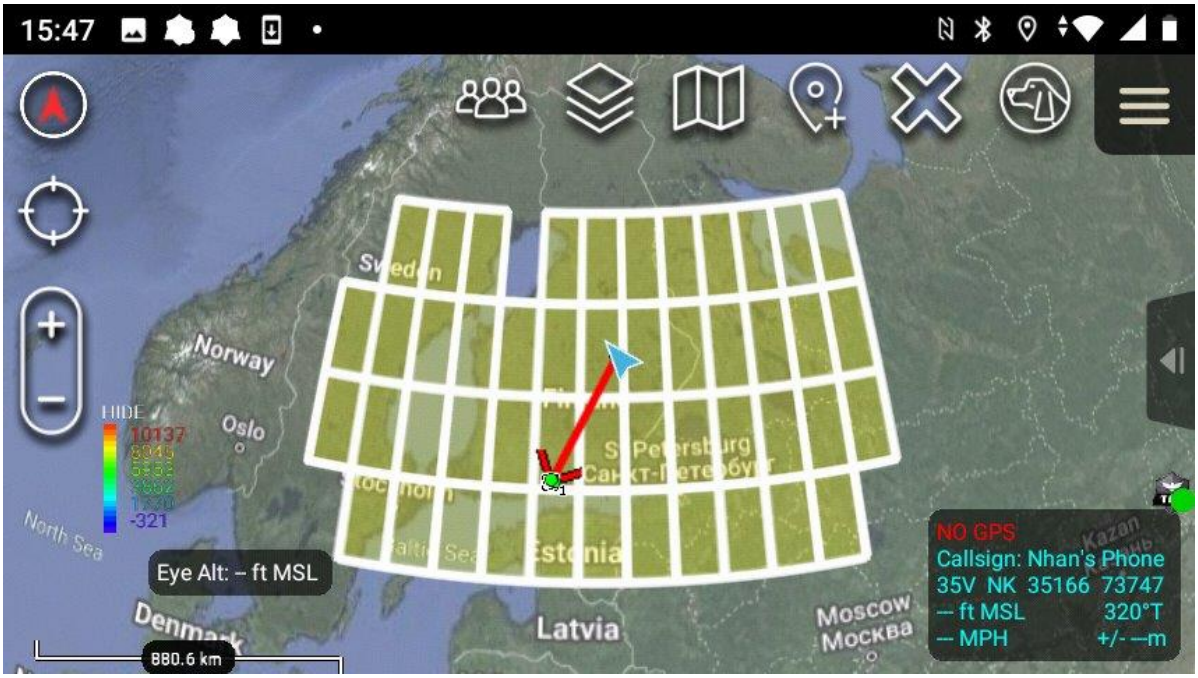

Figure 9. Elevation data of Nordic (Vo 2024)

3.3.6 Radio controls

This feature was initially built for law enforcement and emergency services since they require to have a secure communication method, which usually results in the use of TETRA radios due to its secure voice and data transmission, excellent functionality, high availability, and large capacity. ATAK has supported a wide range of waveforms for radio communications including L-band, S-band, C-band low, C-band high, Ku-band low, Ku-band low 2, and Ku-band high (TAK Product Center 2024, 32). Frequency is also adjustable manually depending on user needs.

Once a connection is established, point-to-point protocol communication is available and is indicated on the Radio Controls menu. Other radios on the same network will also appear on the map once squelched (TAK Product Center 2024, 31). Additionally, users can ping radio, initiate test video, and engage with raw video recorders.

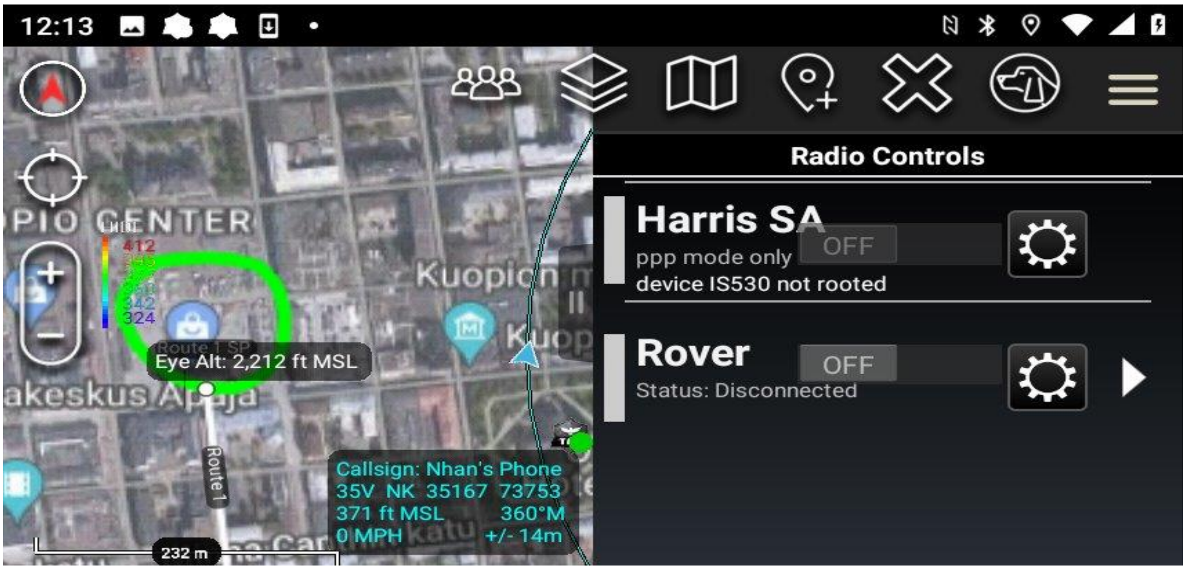

Figure 10. Default radio channels in ATAK (Vo 2024)

3.3.7 Video player

This assists in playing video streams from IP cameras, Rover 5, and H.264 encoders (TAK Product Center 2024, 40). The IP camera is most used in civilian use. This IP camera, an Internet Protocol camera, is a type of digital camera that is managed via an IP network. There is an existing plugin demonstrating all capabilities of Video Player which is TAK ICU. There are various supported network protocols such as UDP, RTSP, RTMP, RTMPS, TCP, RTP, HTTP, HTTPS, etc.

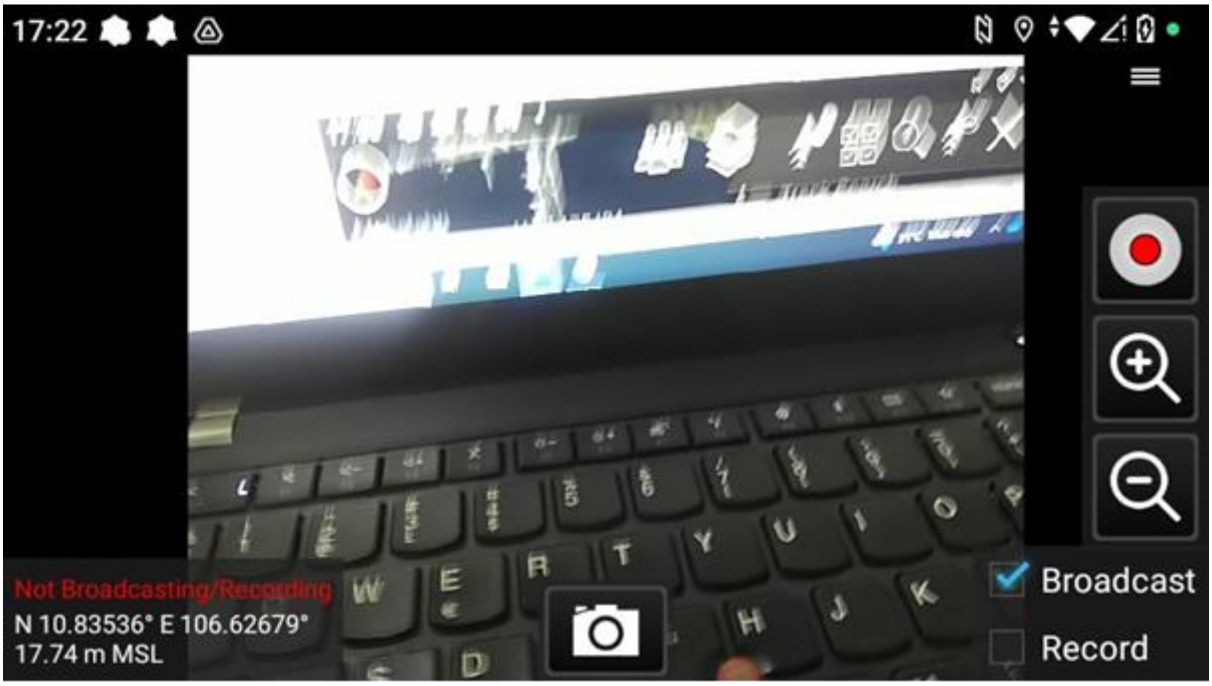

Figure 11. TAK ICU overview (Vo 2024)

Figure 12. TAK ICU with ATAK (Vo 2024)

The above figure depicts a TAK client successfully accessing a video stream from TAK ICU. Once configured, two modes are available for viewing the video stream: broadcast and record. These modes are designed to support both operational management and data maintenance, ensuring that video content can be effectively monitored and preserved as needed.

3.3.8 Contacts

There are multiple ways within the Contact list where users can communicate with other users on the local network or TAK server. By default, a TAK user has the rights to chat with all users on the server in “All chat rooms”, users assigned to the same group or team. Each communication group is extremely secure, which prevents unauthorized users from accessing to. In other words, users in each group are provided access to their assigned groups or teams. Enhanced security for all communications can be configured on a mesh network in ATAK. An AES-256 encryption key is created on one EUD and is then shared with other EUDs (TAK Product Center 2024, 39). Once enabled, encrypted devices can securely communicate and exchange data such as their locations, chat, data packages, etc (TAK Product Center 2024, 39).

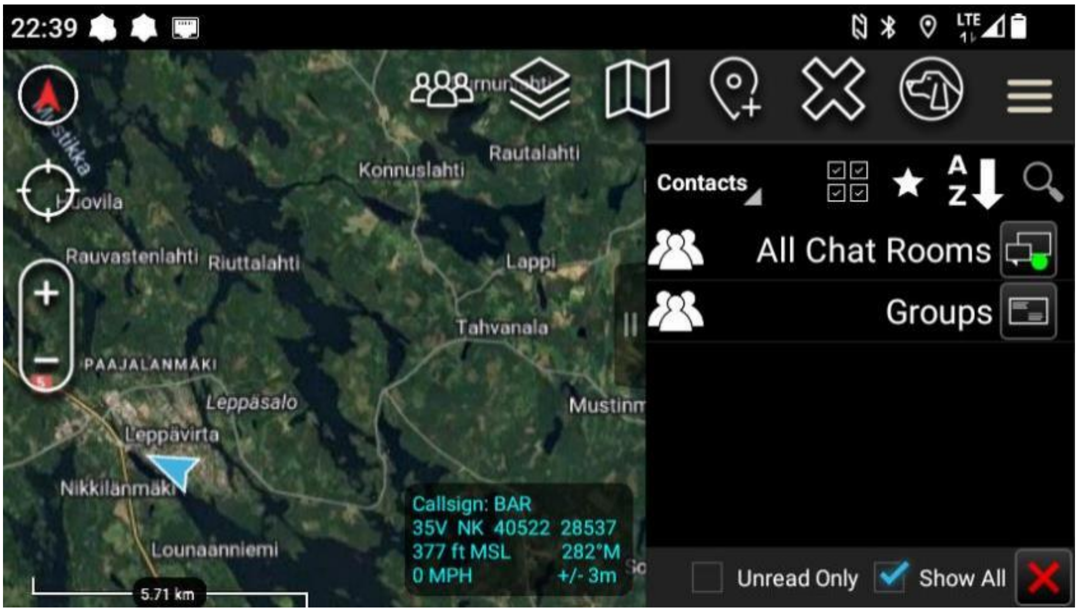

Figure 13. TAK chat (Vo 2024)

Due to its critical use in crucial operations, sending messages within a short amount of time is a must, leading to the creation of default messages. These pre-defined messages present an easy way to transmit a brief message to other network members concerning position and/or other important communication.

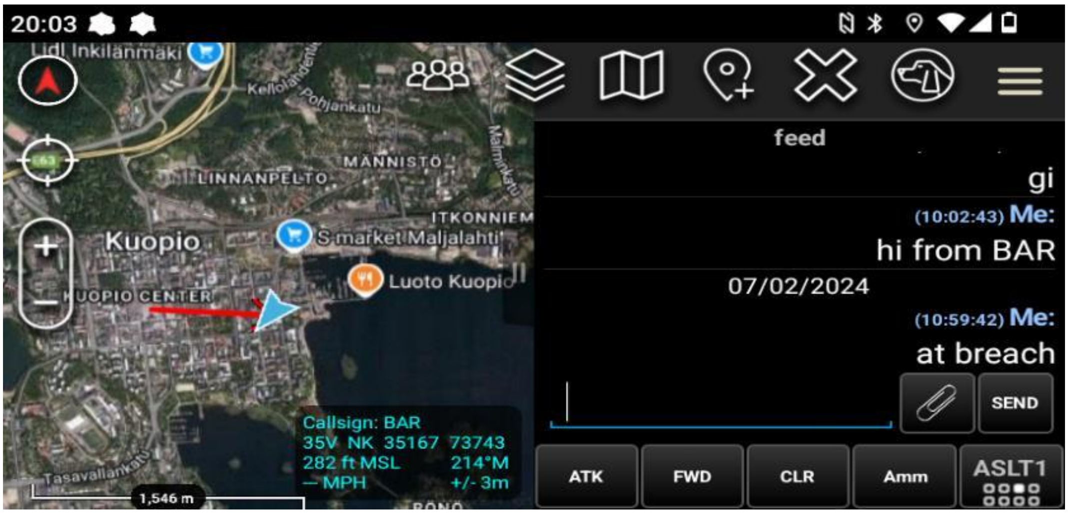

Figure 14. Pre-defined messages (Vo 2024)

3.3.9 Data package

In ATAK, a data package refers to a collection of data that has been compiled and added by TAK users. Its content varies significantly depending on the sources and the users' operational requirements. Typically, these packages include a wide range of information, such as photographs captured in the field, encrypted messages exchanged between group or team members in missions, points of interest (POIs) that highlight critical locations on the map, and video recordings that document real-time events. These data elements are crucial for supporting decision-making processes, enabling users to have an overview of the operational environment and ensuring that all team members have access to the latest information regarding their missions.

All created data packages are stored on the TAK server, which is accessible to all users. These can be downloaded through the TAK server protected by SSL under the port of 8089. The following figure shows all available packages fetched from a TAK server. Under the data package name, information is provided such as the user's callsign and the timestamp when the package was created.

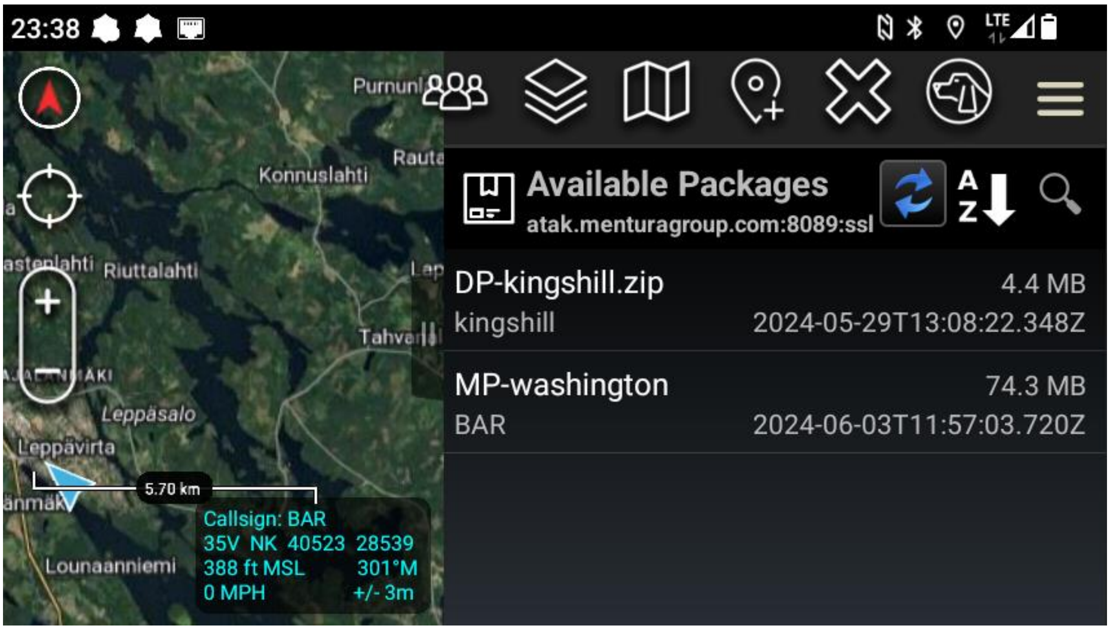

Figure 15. Available data packages on TAK server (Vo 2024)

3.4 Development and usage

ATAK was delivered to the US military in 2010, where the phone was deployed with a chest mount for greater usability on the move during missions. It has since been tested in so many real-world scenarios. (Riskaware 2024). There is a variety of US-based companies consulting this open source software platform, which has caught the attention of various European companies entering the market such as AIMS Finland, JTB Defense, Amalgame Consult, Carbon 01, etc. (JTB Defense 2024; Amalgame Consult 2024; Carbon 2024). Based on those companies, there have been plenty of custom plugins released on the market, which have played vitally in critical operations in the US and other parts of the world. This TAK is currently used by American Law Enforcement and Emergency Services, Canadian Police, British Army, Philippine National Police, along with others (Jr. 2018; Largs&Millport 2020; Royal Canadian Mounted Police 2019). However, its use is kept confidential.

4 COREMOTE TACTICS

4.1 Coremote server system architecture

The COREMOTE Tactics Server system is a Java Enterprise Edition-based server system allowing configuration, and management of system components, network adapters, and event handlers (Mentura 2020). The system provides the ability to support the operations of police, traffic control, and emergency services by monitoring not only stationary objects but also moving objects, which are usually vehicles and law enforcement (Mentura 2020). Data transmission is done over a data communication network, which supports several network protocols such as GSM, 3G, LTE, Tetra network, etc. Furthermore, it uses the PostgreSQL relational database system and its geospatial extension PostGIS. The below figure describes an overview of a typical COREMOTE Tactics system configuration.

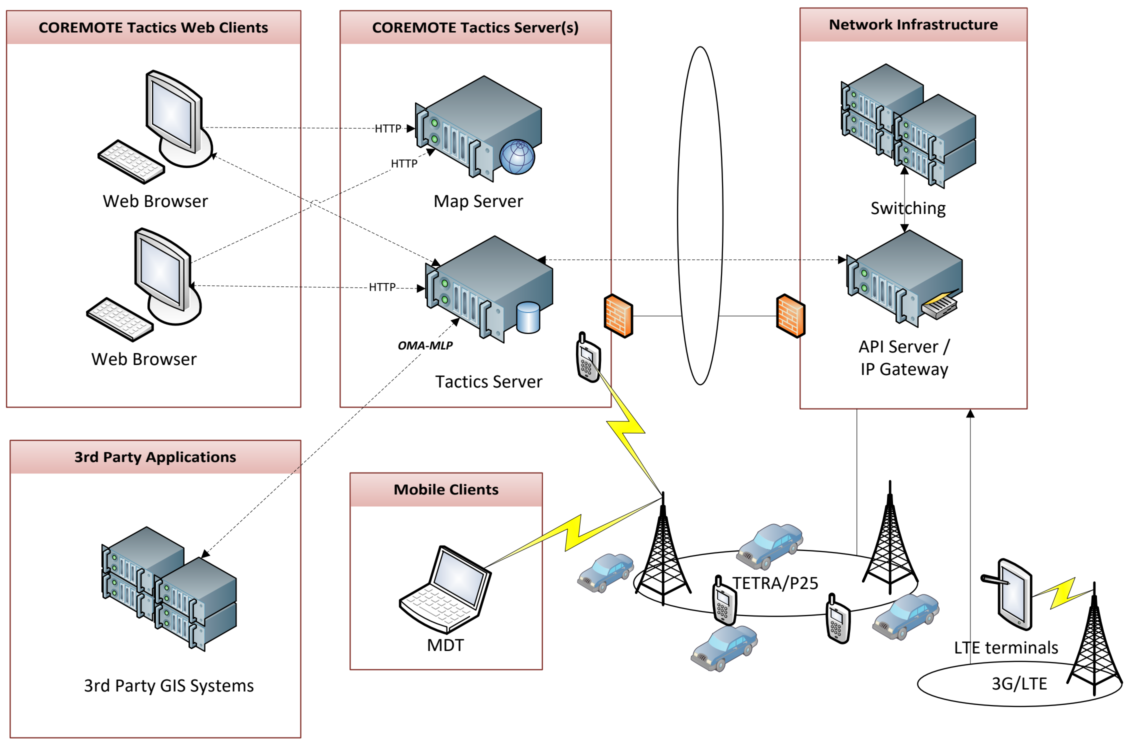

Figure 16. Typical COREMOTE Tactics configuration (Mentura 2020, 10)

4.2 Basic operation

The system organizes different management tasks into layers, which are easily configured into various customer systems. It supports a wide range of operational activities for control room and office environments, including real-time situational awareness about fleet geospatial locations and statuses, data analysis for reporting and management, and indoor positioning based on equipped buildings. It collects context information from different sources like sensors, cameras, or detectors for dispatchers. Additionally, it also integrates data from Automatic Vehicle Location (AVL) and Geographic Information System (GIS), ensuring fast response from law enforcement for dispatching incidents or events to field units over messages and voice. For field units, it facilitates the connection between radio and mobile clients, enabling easy access to data and status updates, and manages the automatic location reporting from mobile devices (Tetra radios, smartphones, tracking devices) using available networks (Mentura, 2024).

Different users are served with different types of data, according to their rights with different views and capabilities.

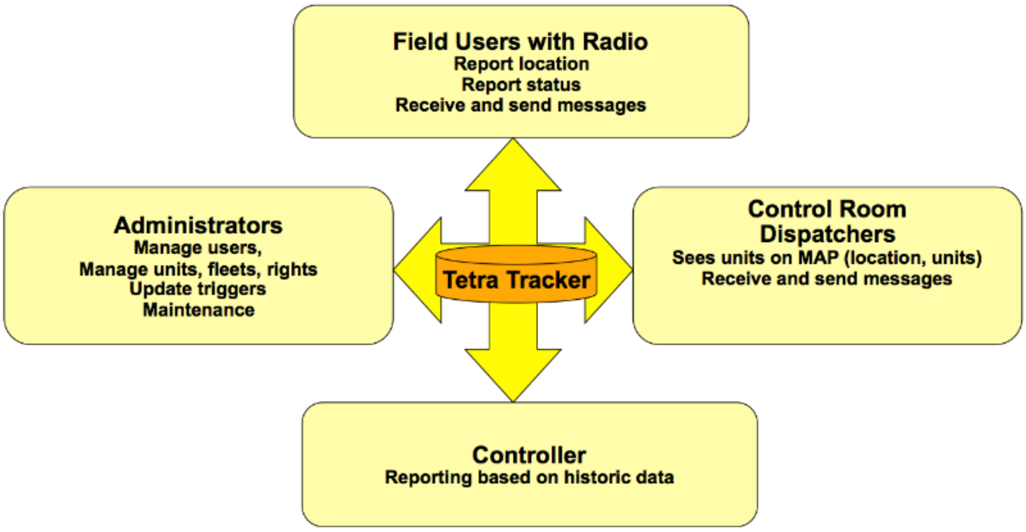

Figure 17. Tactics viewpoint (Mentura 2023, 11)

4.3 User interface overview

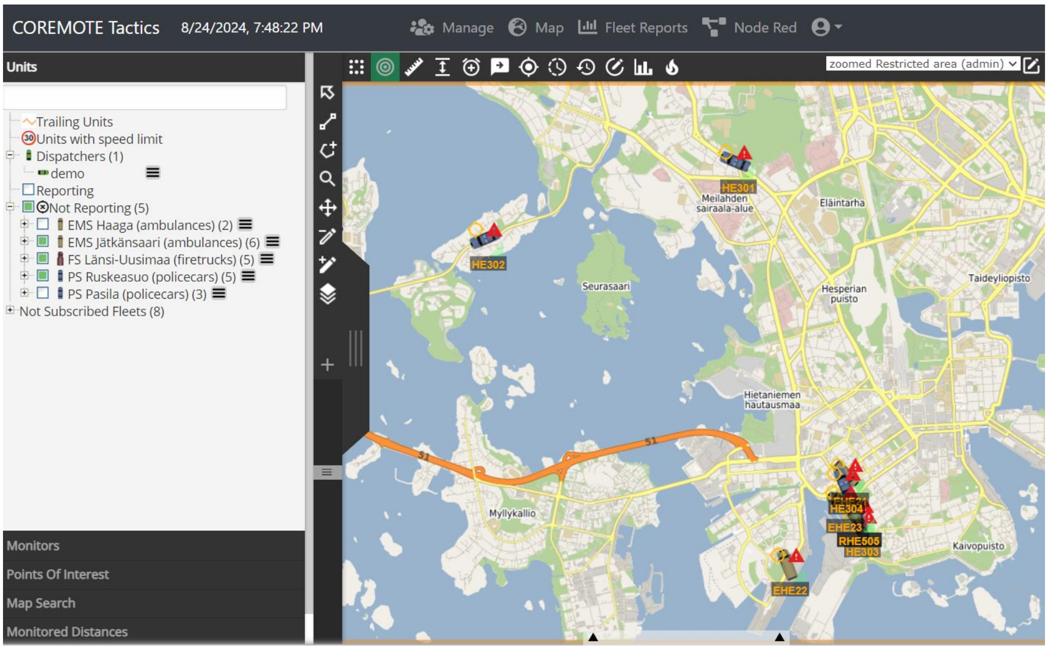

Figure 18. Tactics UI overview (Mentura 2024b)

The above figure shows an admin overview of Tactics after login. With administrative privileges, the user gains access to all administrative features such as managing units, fleets, points of interest, server configurations, along with various map tools. Essentially, an admin user has full control over all aspects of management.

4.3.1 Maps

This system has a variety of map layer support, including Google Satellite, Google Hybrid, Google Street, Google Physical, OSM, and custom maps. Google maps are configured based on a generated

Google API key in server configuration (Mentura 2020, 17-19). In case of an unavailable internet connection, the Map-Server is responsible for producing map images and POI information for browser clients along with custom maps. The map material from the Map Server is transformed into different formats, such as ESRI shape files, GEO-Tiff raster images, etc. Depending on the source material type, a conversion is required to be compatible with the COREMOTE Tactics map server system and customer requirements. The Map-Server generates mapping data in various standard protocols (Mentura 2020, 11):

- WMS (Web Map Service)
- WFS (Web Feature Service)
- WCS (Web Coverage Service)

The following figure briefly illustrates different base maps:

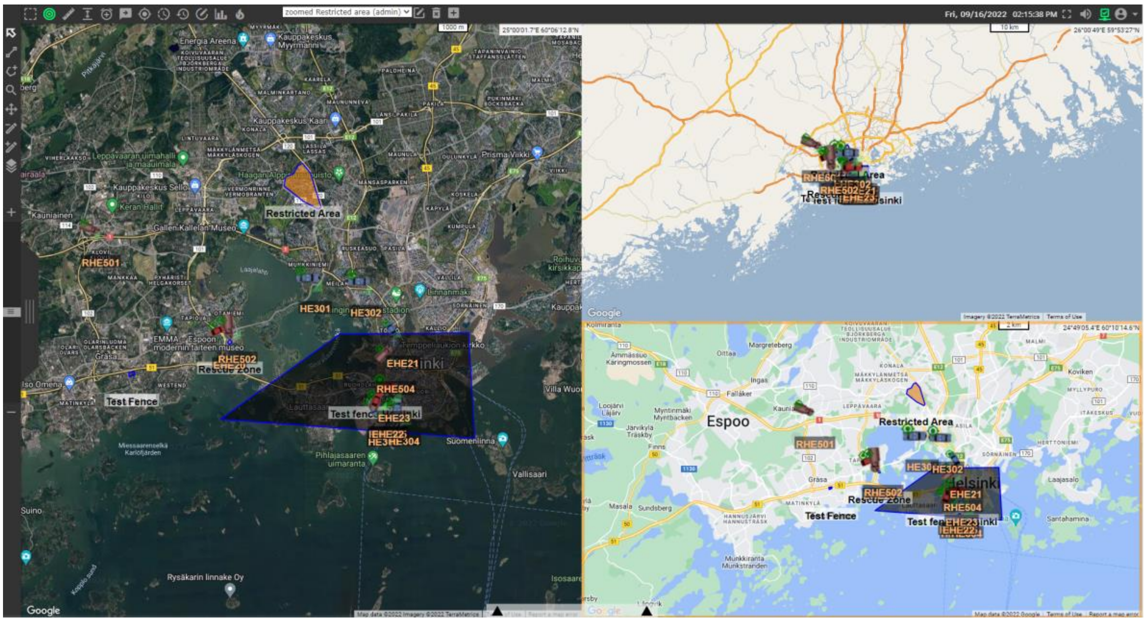

Figure 19. Base maps with split-screen mode (Mentura 2023, 15)

Custom maps are valuable in scenarios where traditional global maps may fail to reach the requirements of providing accurate real-time location data. More specifically, indoor spaces are places where standard mapping services struggle due to limited GPS signal availability and the complexity of layout. In addition, custom maps become a powerful tool under unavailable internet connection, ensuring uninterrupted access to critical location information. The below figure describes an example of a custom map with a variety of complex layouts:

Figure 20. Custom map (Mentura 2024b)

4.3.2 Units and fleet management

In Tactics, organizational structure is composed of units and fleets. A unit is an entity reporting location and status information. A fleet is a collection of units grouped for operational or organizational purposes

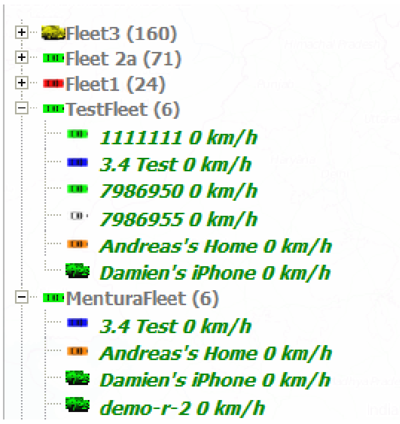

Figure 21. Units and fleets branch tree (Mentura 2023, 26)

The color of the unit label under the unit icon represents the unit reporting status, which is defined below. The status colors are customized by administrators. (Mentura 2023, 26)

Icon	Status	Description
	Not reporting	Unit is not reporting according to Triggers
	No GPS Fix	Unit is reporting according to Triggers but has NO GPS Fix

	**OK**	Unit is reporting according to Trigger and has a GPS Fix

When a client user is added to the system, it will be added to a user group. The rights for user group to see units on the map are set separately based on fleet. A unit or a user can be in multiple fleets. A fleet can have several units or users. Once connected to a fleet, that user gains access to receive up-to-date geospatial information and moving units on the map.

An admin can monitor a unit, which displays all information in a unit information window. That window contains information regarding the fleet it belongs to, reporting status, age of the last location update, speed in km/h, and Tetra status (operative status of the unit), e.g: on the route, the mission started on site, free, coming back to central, etc.

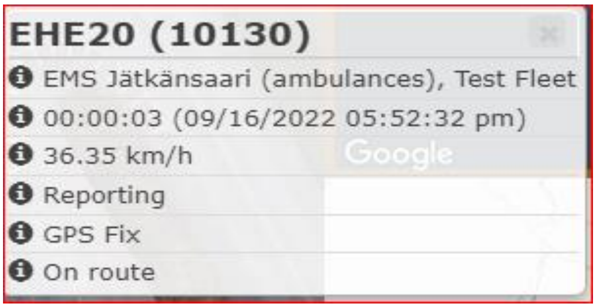

Figure 22. Unit info window (Mentura 2023, 30)

4.3.3 Point of interests

A point of interest refers to a location on the map created, edited, or deleted by a user. Its main purpose is to be shared among users. The list below shows all current POIs on server:

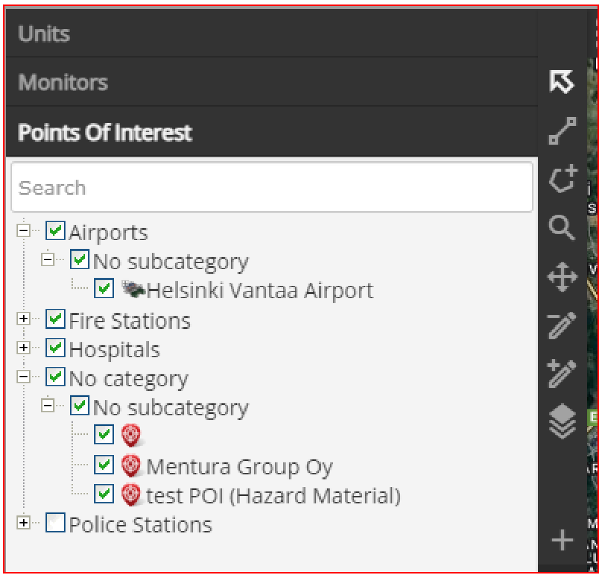

Figure 23. POI list (Mentura 2023, 60)

The POI category search is configured per map, which the user can have multiple maps with different category searches and shift in between the maps. Once result is complete, user can do the following:

- Select single/multiple categories by clicking on the checkboxes to see all POI's under that category to appear on the map.
- Click on a POI under a category to center the map on that POI.
- Deselect single/multiple categories to hide all POI's under that category to disappear from the map.

Key in the name in the "search" field and search for the POI.

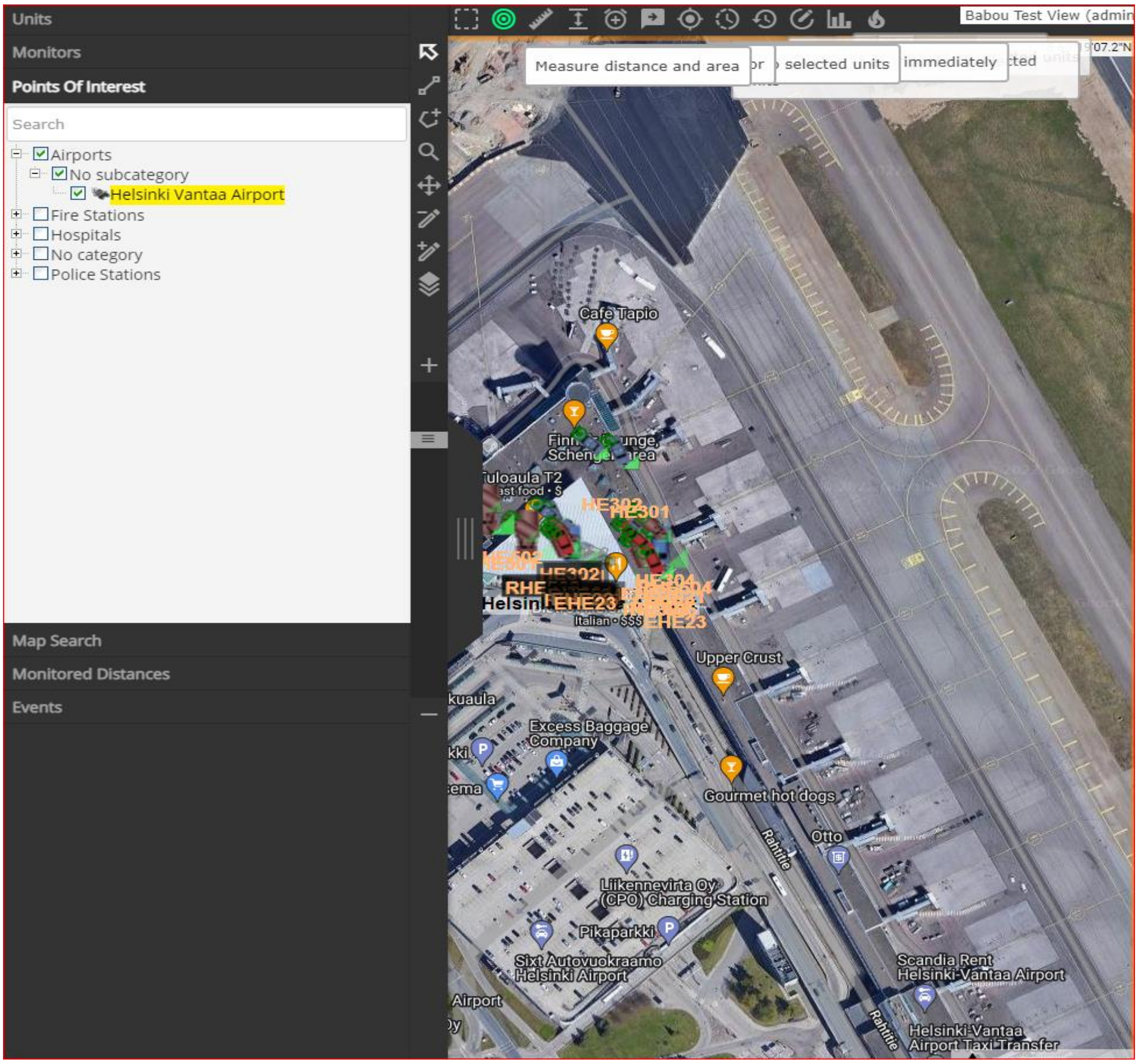

Figure 24. POI search (Mentura 2024b)

POI can also cooperate with other features to make operations more effective, such as geofence. Geofence is a virtual geographic boundary that allows users to set up triggers when an object enters or leaves that geographic area. In tactics, a combination of geofence and point of interest is critical, ensuring all critical events are delivered to users on time, and all events will be stored in the database, which is also beneficial for data analysis later.

4.3.4 Messaging center

This is a set of features for this system that enables users to manage messages. Messaging center is a place where logged-in users can manage messages in different types of mailboxes: inbox, outbox, and sent. A message informs about several information, including its subject, its sender, its timestamp, and its body. A message body usually has a default status value, which is set based on customer requirements; for example, status +50 is on route. The figure below briefly describes an overview of the Tactics messaging center:

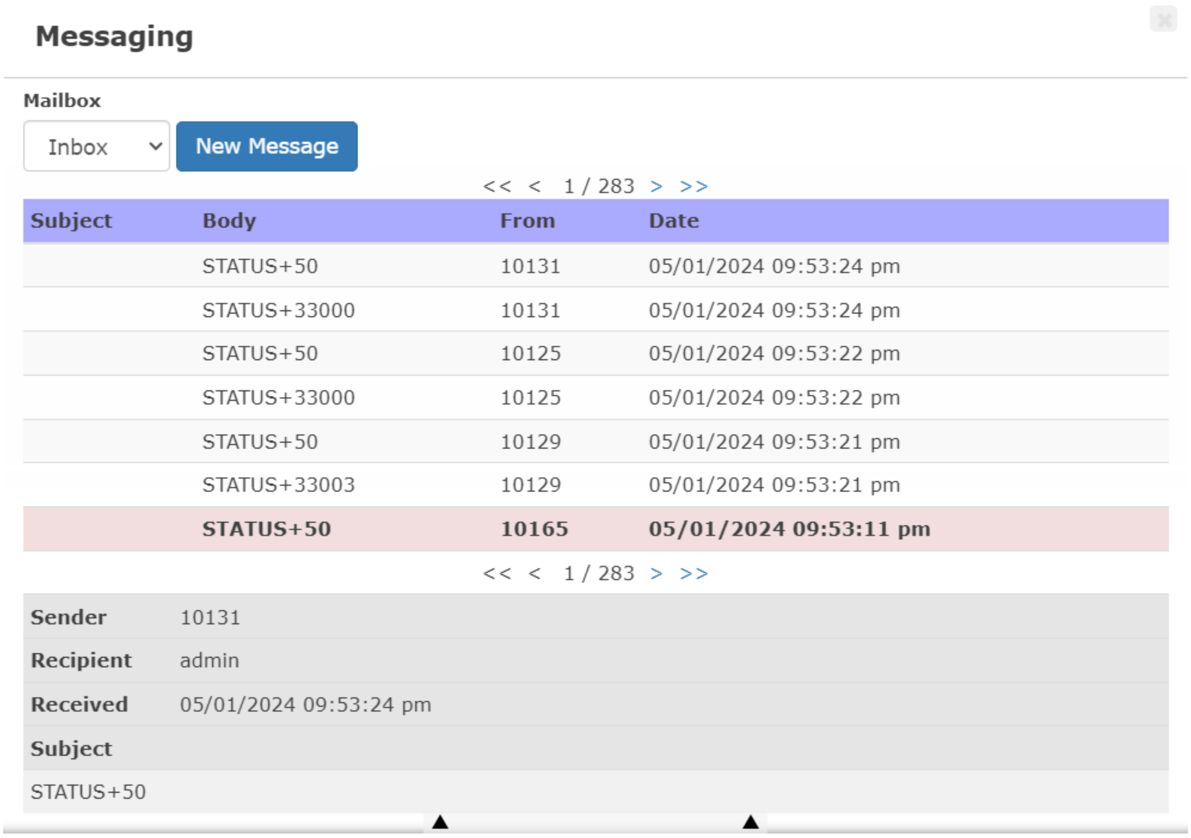

Figure 25. Tactics messaging center (Mentura 2023)

Terrestrial Trunked Radio (TETRA) is a digital trunked mobile radio standard, developed to meet the needs of traditional Professional Mobile Radio (PMR) user organizations such as public safety, transportation, government, commercial & industry, etc. TETRA offers a wide range of benefits, including communication traffic is handled effectively, message and call encryption, priority call handling, direct mode operation, full duplex voice for PABX and PSTN telephony communication (ETSI 2024). Due to its initial design for mission-critical communication, COREMOTE Tactics must integrate TETRA as a key component to manage message routing and logging within the networks that the user operates. However, it is important to note that all TETRA features must be acquired separately.

TETRA locations can be shared with ATAK clients in a common server – Tactics adapter server configuration. One potential approach is to set up a hybrid environment for TETRA phones and ATAK clients. There are several options to do that:

- Carry two separate devices: a TETRA phone and an ATAK client.

- Carry two connected devices: a remote speaker microphone (RSM) connected to a TETRA phone and an ATAK client. The RSM lets the user operate both devices simultaneously, or an ATAK client inside a cover or sleeve connected to a TETRA phone helps the user operate both devices.
- Carry one device: a hybrid device that has the functionality of a TETRA phone and a smartphone in one, such as Tactilon Dabat from Airbus.

5 COMPONENT DESIGN REQUIREMENTS

5.1 Research questions

Several questions have been outlined to help conceptualize what the ATAK integration will look like in COREMOTE Tactics. Additionally, they also provide a comprehensive overview of the TAK community, the massive ecosystem development throughout the world, and business models or commercial products based on TAK.

- How can we share TETRA locations with ATAK clients?
- Could we use ATAK clients as Tactics clients?

5.2 Integration basic operation

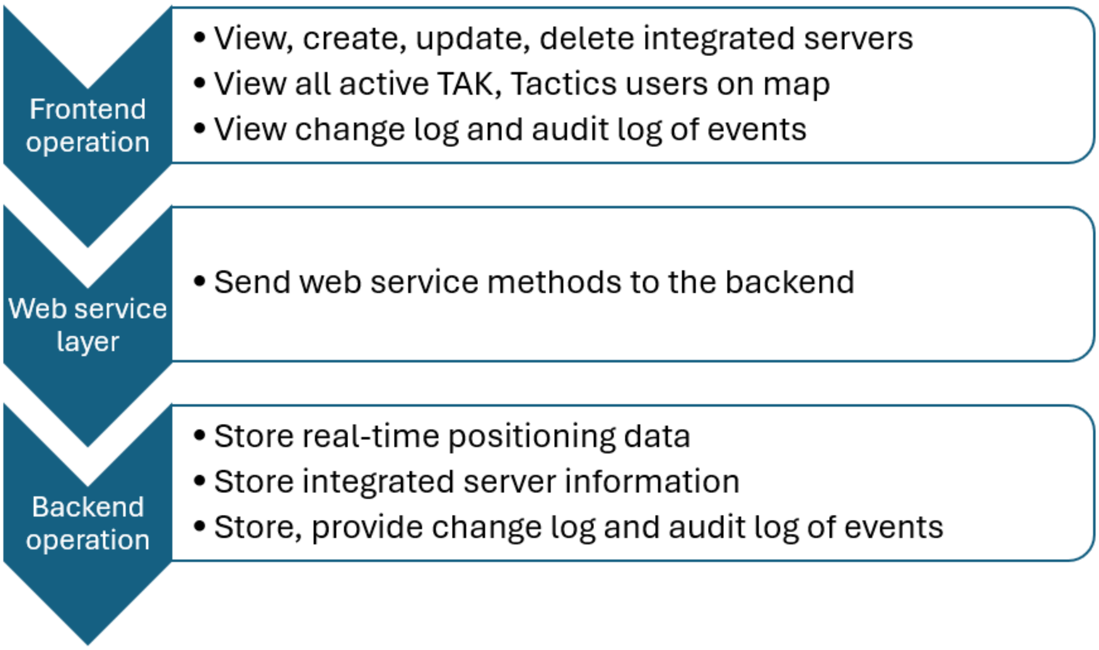

Figure 26. ATAK integration configuration (Vo 2024)

This ATAK integration consists of several configured components from different modules of the Tactics system. Frontend operation mainly operates in JavaScript, having Jshint to check if JavaScript source code compiles with coding rules. The web service layer was built in Java, and it strictly follows the Apache Torque concept, meaning that all data coming from the front end is parsed into value objects based on the XML schema and generated by Torque. Additionally, it also generates a corresponding change log and audit log for actions, such as getting, creating, updating, and deleting an entity, ensuring that all events are logged in the system, which is extremely useful for troubleshooting bugs. The backend uses PostgreSQL 16 as the database. Regarding this integration, the backend stores data coming from the web service layer into corresponding tables:

- Integrated server information: name, hostname, protocol, port, certificates, certificate password
- Real-time positioning data: Cursor on Target (CoT)
- Change log and audit log of web service method execution events

5.3 Integration architecture

When integrating TAK server into Tactics, the TAK server is a client connection which requires the public CA certificate and an issued client certificate for secure connections using SSL protocol under the port of 8089. Additionally, this port provides a real-time CoT (Cursor on Target) stream showing the real-time geospatial location of TAK users.

Because the TAK server is extremely secure, it requires a range of ports open in the firewall system to connect to different services, which the table (JayR 2024) below will briefly describe it:

SERVICE	**PROTOCOL**	**PORT**	**SOURCE**	**DESTINATION**	**DIRECTION**
TAK signaling	TCP/S	8089	Client	Server	In
API Web UI WebTAK	TCP/S	8443	Client	Server	In
Certificate provisioning	TCP/S	8446	Client	Server	In

TAK servers can be added or removed generically. In COREMOTE Tactics, a typical system usually has various adapters for different purposes. In other words, when a new service is designed and integrated, a new Tactics adapter is built to satisfy the need to integrate that service into Tactics. Regarding the Tactics' architecture, configured TAK servers will be adapters on Tactics server. Based on that, the TAK server connection can be easily configured in the Tactics system. There are several configured actions for Tactics components, such as activation, deactivation, and shutdown. The Tactics components below are managed by the Tactics backend, providing a corresponding status of whether it is running properly, or something has gone wrong.

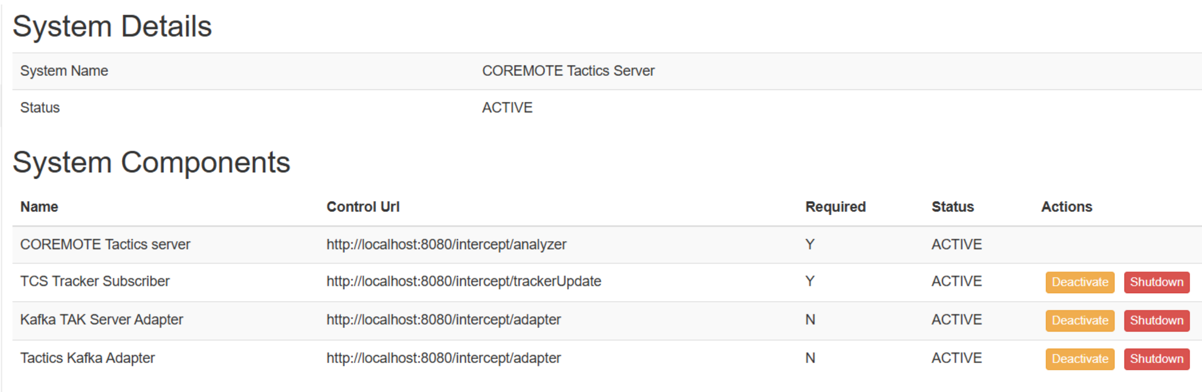

Figure 27. Tactics server components (Vo 2024)

TAK users are also seen as Tactics users, so both users from TAK and Tactics cannot only see each other on the overview map of Tactics but also do either Tactics features or those of TAK together. Each TAK server's user will be added to the same Tactics fleet, in which they can cooperate with other fleets in tactical operations. Figure 27 illustrates all Tactics components with the ATAK integration. Regarding big data that TAK provides, Kafka appeared to be the most suitable distributed system for transferring data from TAK to COREMOTE Tactics due to its high performance, scalability, reliable data distribution, and powerful fault-tolerance mechanisms.

Kafka TAK Adapter and Tactics Kafka Adapter

The following figure demonstrates the TAK adapter architecture in detail. In this architecture, the process begins when the TAK server serializes data from XML to JSON format, the most common data format to push to Kafka. After that, the Kafka topic is connected to a specific network protocol and port configuration, which must be properly set up in the firewall system to ensure uninterrupted data flow.

Currently, this architecture is a point-to-point network connection, but this can be scaled up by adding more producers and consumers for other data streams from a single TAK server, including data package feed and video feed. Moreover, the architecture supports integrating multiple TAK servers into COREMOTE Tactics or multiple COREMOTE Tactics retrieving data from a single TAK server.

Adding producers and consumers facilitates data integration from multiple TAK servers, creating a more extensive data pipeline. For example, each TAK server could serve a specific unit. From COREMOTE Tactics' perspective, an overview map can display all TAK users across servers, enabling unified situational awareness, which is crucial for operational management. However, from the TAK end user perspective, end users can only see users within their server, ensuring operational security and confidentiality.

Similarly, having several consumers obtain data from a single TAK server offers organizations the ability to deploy region-specific or mission-specific platforms. This architecture would be extremely useful in scenarios where various operational units are involved, but it also requires shared situational awareness and coordination for overall mission success.

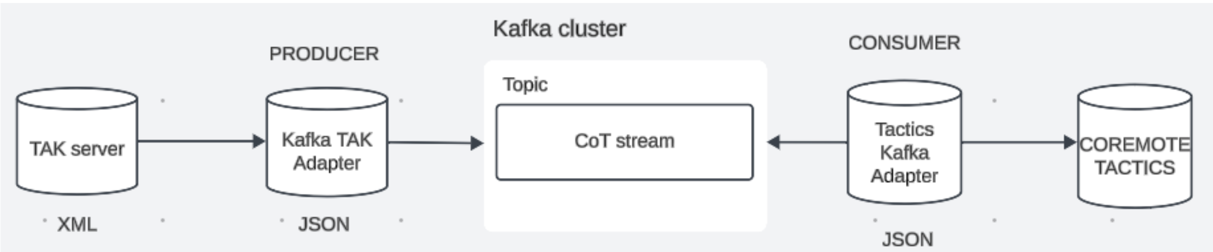

Figure 28. TAK adapters architecture diagram (Vo 2024)

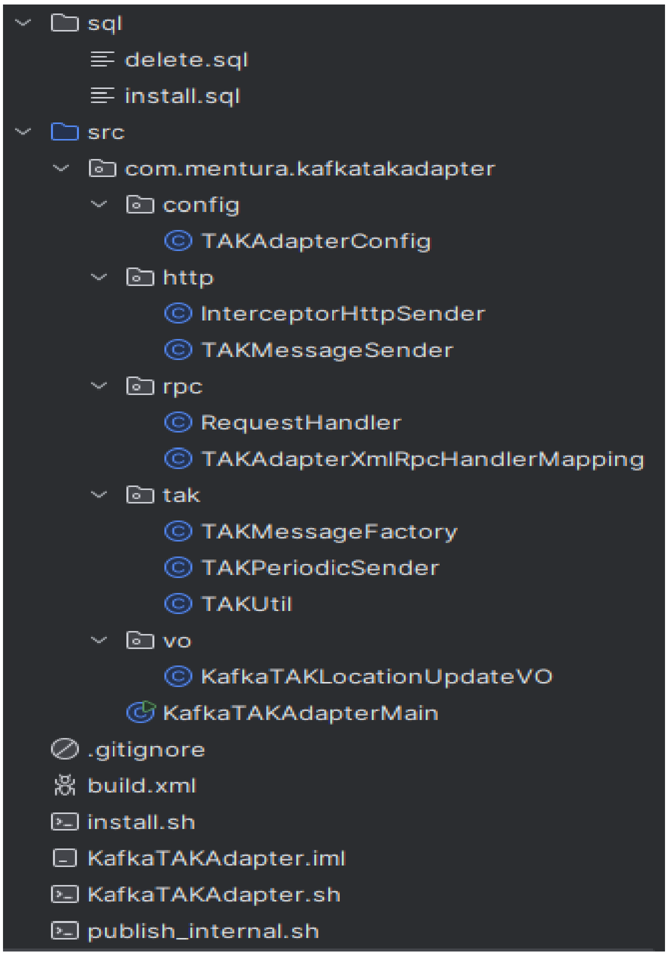

Figure 29. Kafka TAK Adapter project structure (Vo 2024)

The above figure shows an overview of how the Kafka TAK adapter is implemented. The SQL folder contains two SQL scripts for adding and removing the Kafka TAK adapter into COREMOTE Tactics as a component. In other words, the TAK server integrates into Tactics via this Tactics component. which configuration file named TAKAdapterConfig plays a vital role in defining the settings that connect the TAK server to Kafka. This configuration file contains key parameters such as server addresses, ports, and protocol details for establishing the connection between TAK and this adapter (producer). In other words, this file translates XML into JSON, so Kafka can easily interpret the real-time positioning data from the TAK server. KafkaTAKLocationUpdateVO represents the JSON format of real-time positioning data, which has three main sections: event, location, and user.

```
private String uid;  3 usages
private String eventType;  3 usages
private Date timestamp;  3 usages
private Date startedTs;  3 usages

private Float longitude;  3 usages
private Float latitude;  3 usages
private Float hae;  3 usages
private Float ce;  3 usages
private Float le;  3 usages
private Float speed;  3 usages
private Float course;  3 usages

private String callsign;  3 usages
private String group;  3 usages
private String role;  3 usages
```

Figure 30. Real time positioning JSON fields (Vo 2024)

UID is a globally unique name for this information on this event; event type is encrypted in Google Protocol Buffer (a-f-G-U-C is CoT event), timestamp describes when the event was generated, and startedTs describes when the event should be considered valid. Three fields hae, ce, le are specified for the CoT event, which are height above the WGS ellipsoid in meters, circular 1-sigma or a circular area about the point in meters, linear 1-sigma error or an altitude range about the point in meters. Callsign is a unique identifier name of the end user. (Michael J. Kristan & Jeffrey T. Hamalainen & Douglas P. Robbins & Patrick J. Newell 2009).

Within the Kafka TAK adapter structure, there is an RPC folder containing essential files for handling incoming requests and mapping data from the producer to the CoT stream topic. These files translate requests into the format that the Kafka system can process, and later transmit to the consumer. Moreover, the core functions are implemented in RequestHandler, including activating, deactivating, and shutting down the adapter. This approach enhances efficient management and control over the adapter's life cycle and its interaction with other components in COREMOTE Tactics.

The TAK folder includes all necessary files for pushing JSON data to Kafka, focusing on the CoT (Cursor on Target) stream topic. It also includes a utility file that provides various helper functions related to data handling and processing. All client methods and server methods for this adapter architecture are implemented here, ensuring that both client-side and server-side interactions are efficiently managed.

5.4 Security concepts

Access to a TAK server requires to have a server trust store and a client certificate, which are generated by a script in the TAK server Debian package stored in the Virtual Machine, to make an SSL handshake connection. Therefore, those two certificates also need to be configured in the adapter, which is generated for this ATAK integration.

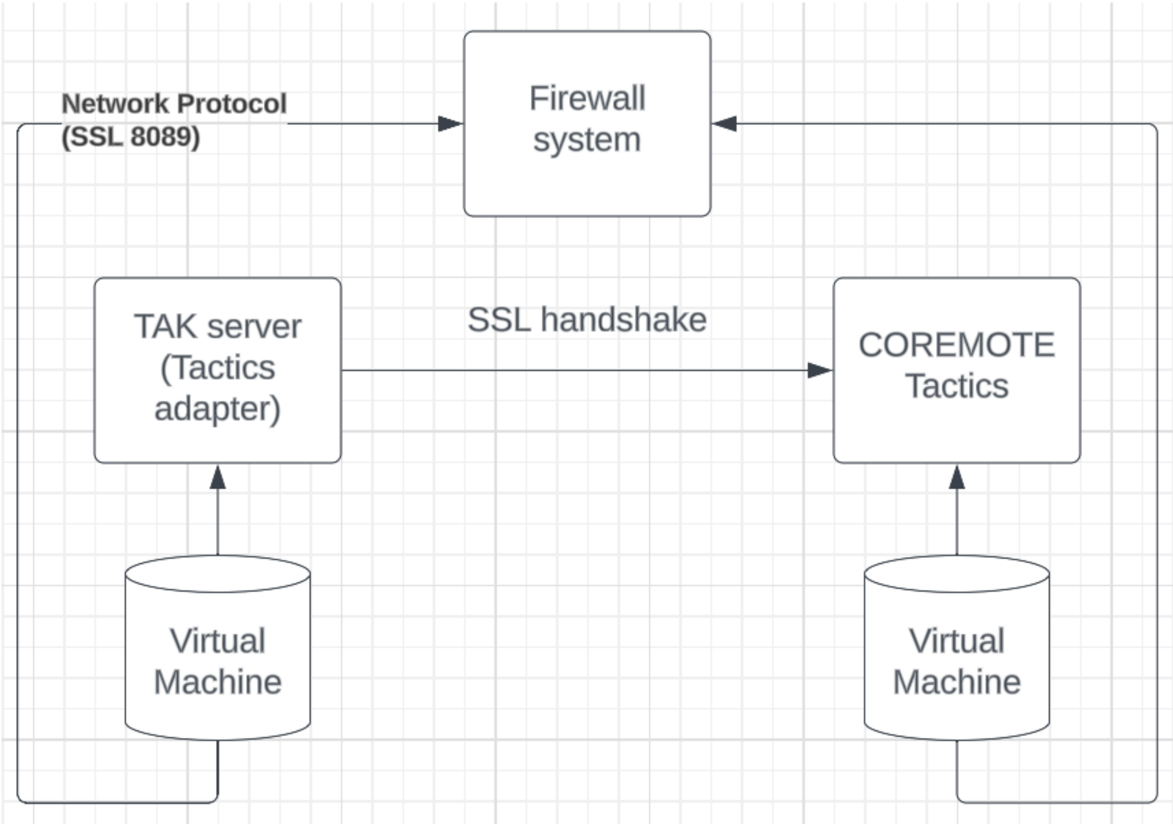

Figure 31. Integration security concept (Vo 2024)

Figure 31 shows a single TAK server integration into COREMOTE Tactics. COREMOTE Tactics has its own virtual machine for database management and other server configurations, including all adapters. TAK server also has its own virtual machine, where a Debian package is installed with a single TAK server having a single PostgreSQL 15 database. Due to secure server configurations of both ATAK and Tactics, a common network protocol is a must to communicate with each other via that protocol, in this case is SSL 8089.

5.5 Tools and technologies

This integration involved the use of multiple programming languages and technical skills. Java and JavaScript are the primary languages utilized in this process. Java was employed to develop a client program for fetching real-time CoT stream, which is crucial for the operational integration of ATAK's data into Tactics. JavaScript was used to implement the TAK configuration dialog, providing a user-friendly interface for configuring and managing the integration settings.

In addition to programming, a strong understanding of database management was crucial. Specifically, knowledge of maintaining a PostgreSQL database in Linux was required since the TAK server relies on this setup. This PostgreSQL serves as a warehouse for data storage, ensuring that all information is efficiently managed.

Besides technical aspects, research also played a pivotal role in the success of this thesis. Integrating ATAK into COREMOTE Tactics required a comprehensive knowledge of ATAK functionalities, protocols, and use cases. The understanding was obtained from extensive research, which involves reading and extracting information from various sources, including official documentation, the TAK community, technical research papers, and other relevant literature. This research ensures that integration is technically useful and compatible with the latest developments and standards in the field, resulting in a more effective solution.

6 INTEGRATION DEVELOPMENT PROCESS

6.1 Implementation process

The thesis work implementation was phased as follows:

- Analysis of existing interfaces for both ATAK and COREMOTE Tactics (05.2024 – 06.2024)
- Research and implementation of a client program for real-time location sharing (07.2024)
- Design of integration architecture and security concepts (08.2024):
    - Data model and UI dialog design for integrated server: 01.08.2024 – 11.08.2024
    - Design data model of integrated server: 12.08.2024 – 31.08.2024
- Implementation of integration software component (09.2024 – 12.2024):
    - ServerManagementBean implementation: 01.09.2024 – 10.09.2024
    - Web service methods (CRUD) implementation: 10.09.2024 – 20.09.2024
    - Add integrated server dialog implementation: 20.09.2024 – 20.10.2024
    - Kafka TAK Adapter and Tactics Kafka Adapter implementation: 01.11.2024 – 25.11.2024
    - OSM map import: 26.11.2024 – 27.11.2024
    - TAK fleet and its units implementation (03.12.2024 – 05.12.2024)
- Documentation and training (12.2024)

6.2 Design and architecture

The TAK server configuration dialog originally took the idea of an existing dialog in ATAK for configuring the TAK server in ATAK. The two figures below will briefly describe what is needed to implement this configuration:

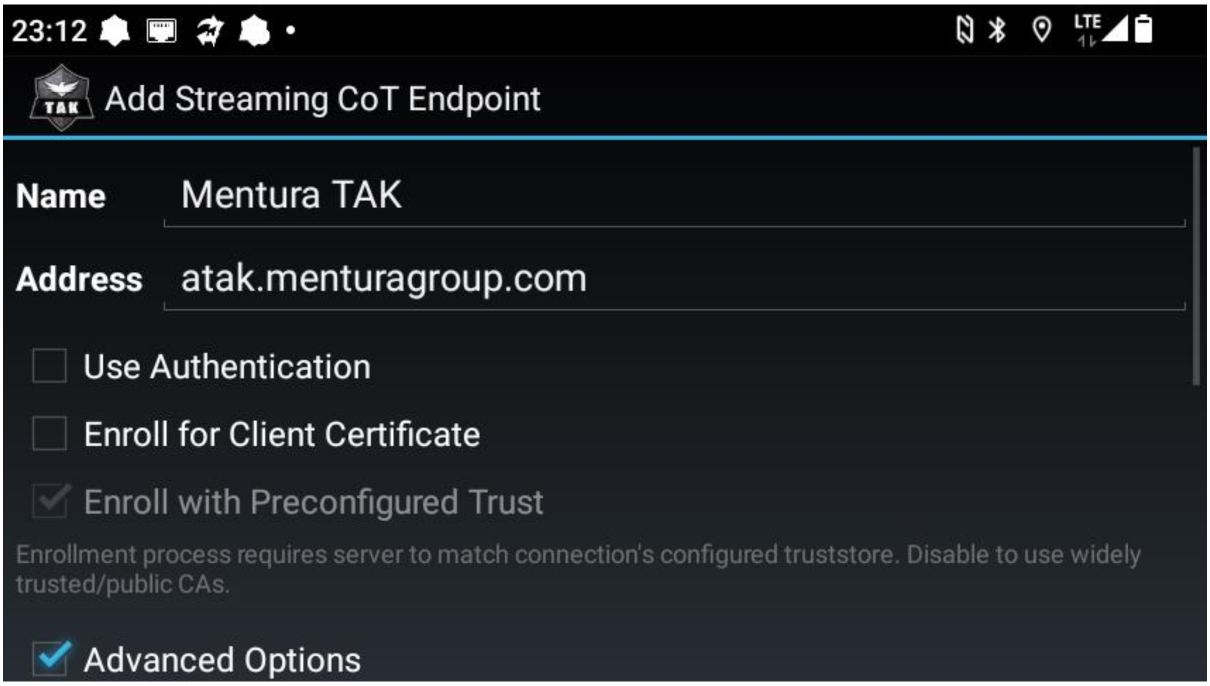

Figure 32. TAK name and server address (Vo 2024)

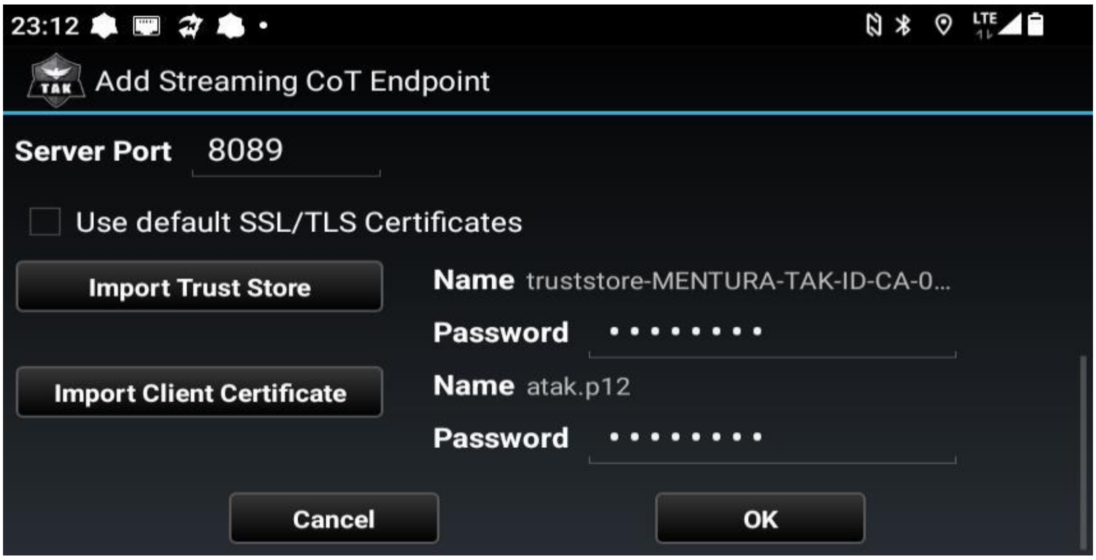

Figure 33. Truststore and client certificate (Vo 2024)

Figure 32 depicts that a name and host address are required, which name is a department or unit (take Mentura TAK as an example), and the host address is the server's Fully Qualified Domain Name (FQDN) for external access – atak.menturagroup.com. Figure 33 shows the advanced options for adding the streaming protocol, truststore, and client certificate. There are three supported streaming protocols, which are TCP, SSL, and QUIC; in this case, a standard SSL connection under port 8089 is chosen. Regarding the truststore and client certificate, those certificates must be created by executing the script in our TAK server's virtual machine, which are *./makeCert.sh <truststore's name>* and *./makeCert.sh <client's name>* respectively.

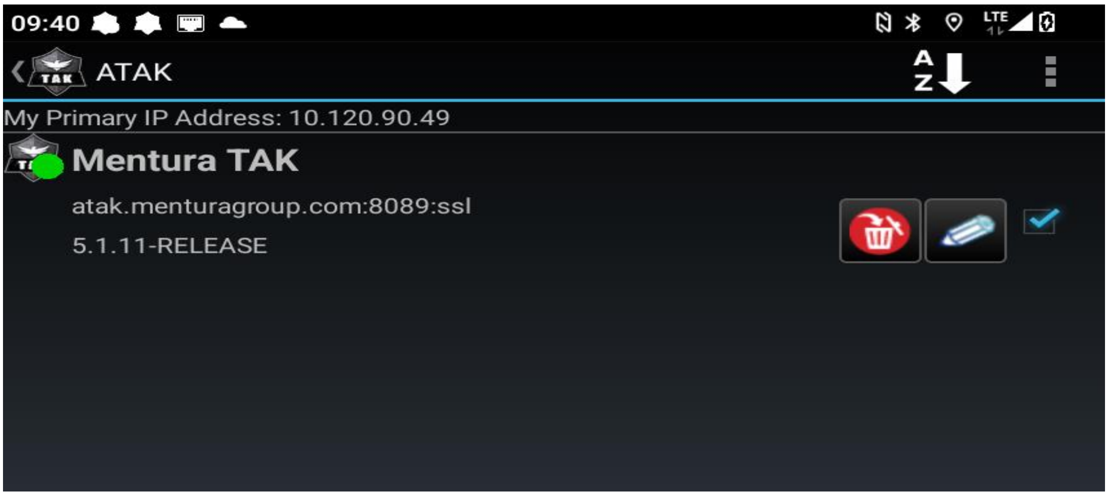

Figure 34. Configured TAK server (Vo 2024)

The original idea was to have a button for showing up the TAK configuration dialog on the menu controls. Once the configuration is complete, all configured TAK servers will be shown as a list having necessary information of the servers, including server's URL and its version. Expanding a configured TAK server provides a view of all TAK users on that server. Those servers can be easily removed by removing the server certificate, the client certificate, and closing the connection port in the firewall system.

6.3 Development environment setup

IntelliJ IDEA was chosen as an environment to implement two Tactics adapters to make the connection between the TAK server and the COREMOTE Tactics server due to several benefits. Firstly, it is a common Java IDE, ensuring that problems regarding setting up the project will be easily resolved. Secondly, it has a built-in remote debugging mode, providing a command line to debug without worrying about the syntax for each JDK. Lastly, this IDE is also known as a friendly-user programming environment for junior developers, which might help resolve problems faster due to its intensive and accurate automatic completion function.

Apache Ant is a Java library and command-line tool used for automating software build processes such as compiling, running, testing, and assembling Java applications (Apache 2024b), which in this case is data management in the backend. For the two TAK adapters, each adapter project has a build file (build.xml), containing task elements, including a description, properties, and targets. The following figure defines a build file example from Apache:

```
<project name="MyProject" default="dist" basedir=".">
  <description>
    simple example build file
  </description>
  <!-- set global properties for this build -->
  <property name="src" location="src"/>
  <property name="build" location="build"/>
  <property name="dist" location="dist"/>

  <target name="init">
    <!-- Create the time stamp -->
    <tstamp/>
    <!-- Create the build directory structure used by compile -->
    <mkdir dir="${build}"/>
  </target>

  <target name="compile" depends="init"
        description="compile the source">
    <!-- Compile the Java code from ${src} into ${build} -->
    <javac srcdir="${src}" destdir="${build}"/>
  </target>

  <target name="dist" depends="compile"
        description="generate the distribution">
    <!-- Create the distribution directory -->
    <mkdir dir="${dist}/lib"/>

    <!-- Put everything in ${build} into the MyProject-${DSTAMP}.jar file -->
    <jar jarfile="${dist}/lib/MyProject-${DSTAMP}.jar" basedir="${build}"/>
  </target>
```

Figure 35. Build.xml example (Apache n.d.)

On the other hand, since both TAK and Tactics server operate in a Linux environment, the decision was made to utilize PuTTY as the SSH client for remote access. PuTTY supports a wide range of protocols, including SSH, Telnet, login, Serial, and RAW, which makes it an ideal choice for developers who need to connect to remote servers securely. During the software component integration process, SSH was employed as the primary protocol due to its powerful security features. SSH not only encrypts the data between server and client, but it also provides secure authentication method.

6.4 Integration development

6.4.1 Add integrated server dialog

In general, all integrated servers will be added to the database via the following dialog. This dialog implementation mainly focused on integrating third party servers using TCP/S protocol as a security feature, requiring a truststore (a collection of all public certificates) and a client certificate.

Figure 36. Add integrated server dialog (Vo 2024)

The following figure shows how the data is sent to the backend via a web service method. This web service method follows a Java approach using an object-relational mapper called Apache Torque. All necessary classes (including Data Objects) were generated from an XML schema, which describes the database layout (Apache 2024a). The following code snippet shows a web service method for creating an integrated server:

```java
public ServerIntegrationGenericResponseVO createServer() {
    ServerIntegrationGenericResponseVO ret = new ServerIntegrationGenericResponseVO();
    ret.setReqAction(ServerIntegrationConstants.ACTION_CREATE_SERVER);
    ret.setErrorCode(0);

    ServerIntegrationBean resServerBean = null;
    UserSession session = getUserSession();
    logger.pushNDC( name: "TrackerWS.createServer." + seqnum++);
    UserHelper.logWebService( service: "TrackerWS", ServerIntegrationConstants.ACTION_CREATE_SERVER, session);
    try {
        logger.debug("Create integrated Server calling req parser!");
        ServerIntegrationBean reqServerBean = ServerManagementUtils.parseServerIntegrationBean(getRequestParameters());
        logger.debug("Create integrated Server calling server mgmt bean!");
        resServerBean = ServerManagementUtil.getLocalHome().create().createServer(session, reqServerBean);
        ret.setServerId(resServerBean.getId());
        logger.debug("Server details created successfully.");
    }
    catch (InterceptorException e) {
        logger.error("Failed to create an integrated server: " + e.getMessage());
        ret.setErrorCode(e.getErrorCode());
    }
    catch (Exception e) {
        logger.error("Failed to create server: " + e.getMessage());
    }
```

Figure 37. Web service method for integrated server creation (Vo 2024)

6.4.2 TAK adapters as Tactics components

As mentioned in the integration architecture, the TAK server was integrated into Tactics via components, which are the Kafka TAK adapter and the Tactics Kafka adapter. The Kafka TAK adapter is the producer, and the Tactics Kafka adapter is the consumer. The following code snippet illustrates how the producer adapter reads CoT events from the TAK server via an SSL socket connection under the port 8089:

```java
/**
 * Reads CoT events from the SSL connection
 */
public KafkaTAKLocationUpdateVO readCoTEvent() {
    KafkaTAKLocationUpdateVO locUpdate = null;
    StringBuilder messageBuilder = new StringBuilder();

    try {
        InputStream inputStream = sslSocket.getInputStream();
        byte[] buffer = new byte[1024];
        int bytesRead;

        bytesRead = inputStream.read(buffer);
        if(bytesRead != -1) {
            String response = new String(buffer, 0, bytesRead, StandardCharsets.UTF_8);
            logger.info(response);

            messageBuilder.append(response);

            String message = messageBuilder.toString();
            int endIndex = message.indexOf("</event>");
            if (endIndex != -1) {
                endIndex += "</event>".length();
                String completeMessage = message.substring(0, endIndex);

                locUpdate = parseMessageToLocationUpdate(completeMessage);

                messageBuilder.delete(0, endIndex);
            }
        }
    } catch (Exception e) {
        logger.error("Error reading CoT event from SSL socket: " + e.getMessage(), e);
    }
    return locUpdate;
}
```

Figure 38. Read CoT event (Vo 2024)

This method is used in a thread that generates a Kafka producer to send CoT events to the “cot-locations” topic. The Kafka producer is created by config stored in the backend, and the following figures show config properties for the producer adapter and the producer adapter thread:

```sql
DELETE FROM component_config where componentid=606;
DELETE FROM system_component where componentid=606;

INSERT INTO system_component ("componentid", "name", "description", "statuschgtime", "lastrefreshed", "status", "url", "required", "type", "control_host", "control_port", "in_use", "sys
 VALUES (606, 'Kafka TAK Server Adapter', 'Push CoT stream from TAK to Kafka', '2024-09-10 00:00:00', NULL, 'UNKNOWN', 'http://localhost:8080/intercept/adapter', 'N', 'PA', 'localhost'

-- register an adapter with default settings
INSERT INTO component_config VALUES (60600, 'port', '80', 606);
INSERT INTO component_config VALUES (60601, 'workingDir', '/home/mentura/apps/KafkaTAKAdapter/', 606);
INSERT INTO component_config VALUES (60602, 'appPath', '/home/mentura/apps/KafkaTAKAdapter/KafkaTAKAdapter.sh', 606);
INSERT INTO component_config VALUES (60603, 'loglevel', '4', 606);
INSERT INTO component_config VALUES (60604, 'backend', 'localhost', 606);
INSERT INTO component_config VALUES (60605, 'takServer', 'atak.menturagroup.com', 606);
INSERT INTO component_config VALUES (60606, 'takPort', '8089', 606);
INSERT INTO component_config VALUES (60607, 'clientCertPath', '/home/mentura/apps/KafkaTAKAdapter/atak.p12', 606);
INSERT INTO component_config VALUES (60608, 'serverCertPath', '/home/mentura/apps/KafkaTAKAdapter/truststore-MENTURA-TAK-ID-CA-01.p12', 606);
INSERT INTO component_config VALUES (60609, 'certPassword', 'atakatak', 606);
INSERT INTO component_config VALUES (60611, 'reconnectInterval', '30', 606);
INSERT INTO component_config VALUES (60612, 'reconnectAttempts', '3', 606);
INSERT INTO component_config VALUES (60613, 'kafkaHost', '', 606);
INSERT INTO component_config VALUES (60614, 'kafkaCoTTopic', 'cot-locations', 606);
```

Figure 39. Config properties of Kafka TAK Adapter (Vo 2024)

```java
while(true) {
    logger.debug("run() in outer while loop.");
    try
    {
        try
        {
            logger.debug("run() Instantiating Kafka producer.");
            // create a new producer
            producer = new KafkaProducer<String, String>(props);
            logger.debug("run() Kafka producer created.");
        }
        catch(Exception e)
        {
            logger.error("Failed to connect to kafka: " + e.getMessage());
            // we wait before trying again
            Thread.sleep(this.sleepTime);
        }
        catch(Throwable t)
        {
            logger.error("Failed to create Kafka producer:" + t.getMessage() );
        }

        // we have a producer start sending loop
        if(producer != null)
        {
            while(true)
            {
                logger.debug("run() calling sendPeriodicUpdate()");
                sendPeriodicUpdate(producer);

                Thread.sleep(this.sleepTime);
            }
        }
    }
```

Figure 40. Kafka TAK Adapter thread (Vo 2024)

Like the producer adapter, the consumer adapter needs to be configured by a SQL script, creating necessary config settings to the database. This SQL script is slightly different from the producer's script, the main difference comes from that this consumer script also defines a tracking protocol for all configured TAK users to connect to the Tactics server.

```sql
DELETE FROM component_config where componentid=607;
DELETE FROM system_component where componentid=607;

INSERT INTO system_component ("componentid", "name", "description", "statuschgtime", "lastrefreshed", "status", "url", "required", "type", "control_host", "control_port", "in_use", "sys
 VALUES (607, 'Tactics Kafka Adapter', 'Forward CoT stream from Kafka to Tactics', '2024-09-10 00:00:00', NULL, 'UNKNOWN', 'http://localhost:8080/intercept/adapter', 'N', 'PA', 'localho

-- register an adapter with default settings
INSERT INTO component_config VALUES (60700, 'port', '8080', 607);
INSERT INTO component_config VALUES (60701, 'workingDir', '/home/mentura/apps/TacticsKafkaAdapter/', 607);
INSERT INTO component_config VALUES (60702, 'appPath', '/home/mentura/apps/TacticsKafkaAdapter/TacticsKafkaAdapter.sh', 607);
INSERT INTO component_config VALUES (60703, 'loglevel', '4', 607);
INSERT INTO component_config VALUES (60704, 'backend', 'localhost', 607);
INSERT INTO component_config VALUES (60706, 'reconnectInterval', '30', 607);
INSERT INTO component_config VALUES (60707, 'reconnectInterval', '30', 607);
INSERT INTO component_config VALUES (60708, 'reconnectAttempts', '3', 607);
INSERT INTO component_config VALUES (60709, 'kafkaHost', 'localhost:9092', 607);
INSERT INTO component_config VALUES (60710, 'kafkaCoTTopic', 'cot-locations', 607);
INSERT INTO component_config VALUES (60711, 'httpSenderThreadCount', '5', 607);
INSERT INTO component_config VALUES (60712, 'locationServerPort', '%locationport%', 607);

begin;

insert into tracking_protocol (protocol_id, name, description, enabled) values (nextval('tracking_protocol_seq'), 'TAK', 'SSL based tracking protocol for TAK users', 1);

insert into tracking_element (element_id, max_units, component_id) values (nextval('tracking_element_seq'), 1000, 607);

insert into protocol_element (pe_id, protocol_id, element_id) values (nextval('protocol_element_seq'), currval('tracking_protocol_seq'), currval('tracking_element_seq'));

commit;
```

Figure 41. Config properties of Tactics Kafka Adapter (Vo 2024)

Data starts flowing from the TAK server to the Tactics server when both adapters are in active state. Once the data is successfully pushed to Kafka, the consumer adapter, Tactics Kafka Adapter, starts fetching data from Kafka, parsing crucial fields from that CoT event value object to send to the Tactics backend, including callsign, longitude, latitude, hae, speed, and course.

```java
JSONParser parser = new JSONParser();
JSONObject jsonMessage = (JSONObject) parser.parse(message);

String timestamp = jsonMessage.get("Timestamp").toString();
try {
    Date date = new Date();
    SimpleDateFormat df = new SimpleDateFormat("yyyyMMddHHmmss");
    df.setTimeZone(TimeZone.getTimeZone("GMT+0"));
    timestamp = df.format(date);
}
catch (Exception e) {
    logger.error("Failed to parse timestamp: "+e.getMessage(), e);
}
String currentId = config.getCurrentId();
String issi = jsonMessage.get("Callsign").toString();
String lon = jsonMessage.get("Longitude").toString();
String lat = jsonMessage.get("Latitude").toString();
String alt = jsonMessage.get("HAE").toString();
String velo = jsonMessage.get("Speed").toString();
String veloMs = "0";
if (velo == null || "".equals(velo)) {
    float kmph = 0;
    try {
        kmph = Float.parseFloat(veloMs) * 3600 / 1000;
        velo = String.format("%.2f", kmph);
    } catch (NumberFormatException nfe) {
        logger.error("CGI parameter ms is not a float: " + veloMs);
    } catch (Exception ex) {
        logger.error("CGI parameter ms is null.");
    }
}
String dir = jsonMessage.get("Course").toString();
String rad = "0";

String resultCode = ihs.sendLocationUpdate(currentId, "", issi, timestamp, lon, lat, alt, dir, velo, rad);
logger.debug("sendLocationUpdate result: " + resultCode);
```

Figure 42. Method to send request to Tactics backend (Vo 2024)

Those CoT value object fields are parsed to corresponding arguments of the method that sends a HTTP POST request to Tactics backend, which is *localhost:8080*

```java
public String sendLocationUpdate(String currentId, String reqId, String issi, String timestamp, String lon,
                                 String lat, String alt, String dir, String velo, String rad)
        throws IOException {

    Msg request = new Msg();

    request.setField(Msg.C_FIELD_MESSAGE_TYPE, "mlplocation");
    request.setField(Msg.C_FIELD_ID, currentId);
    request.setField("req_id", reqId);
    request.setField("method", "update");
    request.setField("handler", "mlplocation");

    request.setField("receiveTimestamp", timestamp); // TrackingMessageConstants.RECEIVE_TIMESTAMP;

    request.setField("issi", issi); // TrackingMessageConstants.ISSI
    if (lon != null)
        request.setField("locationLongitude", lon); // TrackingMessageConstants.LOCATION_LONGITUDE
    if (lat != null)
        request.setField("locationLatitude", lat); // TrackingMessageConstants.LOCATION_LATITUDE
    if (alt != null)
        request.setField("locationAltitude", alt); // TrackingMessageConstants.LOCATION_ALTITUDE

    if (dir != null)
        request.setField("directionOfTravel", toRadians(dir)); // LOCATION_DIRECTION_OF_TRAVEL
    if (velo != null)
        request.setField("locationHorizontalVelocity", velo); // TrackingMessageConstants.LOCATION_HORIZONTAL_VELOCITY
    if (rad != null)
        request.setField("radius", rad); // TrackingMessageConstants.RADIUS


    Vector res = mySender.sendRequest(request);
    logger.info("sendLocationUpdate(): URL: " + mySender.getUrl() + ", Sent: " + request + ", got back: " + res);
    return getResultCode(res);
}
```

Figure 43. Method to send location update to Tactics backend (Vo 2024)

6.4.3 Data visualization

The TAK server provides a wide range of data, including real-time positioning, data packages, video feed, POI, geofences, etc. This thesis work focused on visualizing real-time positioning data on the Tactics map, building a connection between TAK clients and Tactics clients, and providing a possibility of integrating a third-party server into Tactics. Kafka is a data streaming connection for the two mentioned servers. To do that, a Kafka topic and a bootstrap server were finalized, which are *cot-locations* and *localhost:9092,* respectively. The *cot-locations* topic supplies critical information from the TAK CoT stream, including event, location, and user details, to Kafka as a value object.

A value object is essential in the context of data management, particularly when dealing with systems like Kafka, as it encapsulates and provides a structured representation of the data being stored. Therefore, developers can create a clear and consistent data model, which serves several critical purposes, such as data clarity enhancement, data-pushing related problem troubleshooting, metadata providing additional context of stored data, and codebase maintainability. The following figure illustrates some sample data stored in the mentioned Kafka topic:

```
{"UID": "ANDROID-24573c1cc71a9f6e","EventType": "a-f-G-U-C","Timestamp": "2024-11-22T19:21:44.000Z","StartedTs": "2024-11-22T19:
27:56.000Z","Longitude": 27.691751,"Latitude": 62.892044,"HAE": 9999999.0,"CE": 16.1,"LE": 9999999.0,"Callsign": "BAR","Group":
"Cyan","Role": "Team Member"}
{"UID": "ANDROID-24573c1cc71a9f6e","EventType": "a-f-G-U-C","Timestamp": "2024-11-22T19:21:48.000Z","StartedTs": "2024-11-22T19:
28:00.000Z","Longitude": 27.691748,"Latitude": 62.892044,"HAE": 105.338,"CE": 13.9,"LE": 9999999.0,"Callsign": "BAR","Group": "C
yan","Role": "Team Member"}
{"UID": "ANDROID-24573c1cc71a9f6e","EventType": "a-f-G-U-C","Timestamp": "2024-11-22T19:21:50.000Z","StartedTs": "2024-11-22T19:
28:02.000Z","Longitude": 27.691748,"Latitude": 62.892044,"HAE": 105.538,"CE": 11.8,"LE": 9999999.0,"Callsign": "BAR","Group": "C
yan","Role": "Team Member"}
{"UID": "ANDROID-24573c1cc71a9f6e","EventType": "a-f-G-U-C","Timestamp": "2024-11-22T19:22:08.000Z","StartedTs": "2024-11-22T19:
28:20.000Z","Longitude": 27.691816,"Latitude": 62.891983,"HAE": 104.038,"CE": 8.6,"LE": 9999999.0,"Callsign": "BAR","Group": "Cy
an","Role": "Team Member"}
{"UID": "ANDROID-24573c1cc71a9f6e","EventType": "a-f-G-U-C","Timestamp": "2024-11-22T19:22:27.000Z","StartedTs": "2024-11-22T19:
28:39.000Z","Longitude": 27.691835,"Latitude": 62.89205,"HAE": 107.338,"CE": 10.7,"LE": 9999999.0,"Callsign": "BAR","Group": "Cy
an","Role": "Team Member"}
{"UID": "ANDROID-24573c1cc71a9f6e","EventType": "a-f-G-U-C","Timestamp": "2024-11-22T19:22:47.000Z","StartedTs": "2024-11-22T19:
28:59.000Z","Longitude": 27.69175,"Latitude": 62.892143,"HAE": 107.338,"CE": 13.9,"LE": 9999999.0,"Callsign": "BAR","Group": "Cy
an","Role": "Team Member"}
{"UID": "ANDROID-24573c1cc71a9f6e","EventType": "a-f-G-U-C","Timestamp": "2024-11-22T19:23:07.000Z","StartedTs": "2024-11-22T19:
29:19.000Z","Longitude": 27.691608,"Latitude": 62.89195,"HAE": 108.438,"CE": 10.7,"LE": 9999999.0,"Callsign": "BAR","Group": "Cy
an","Role": "Team Member"}
{"UID": "ANDROID-24573c1cc71a9f6e","EventType": "a-f-G-U-C","Timestamp": "2024-11-22T19:23:51.000Z","StartedTs": "2024-11-22T19:
30:03.000Z","Longitude": 27.691586,"Latitude": 62.89197,"HAE": 113.938,"CE": 10.7,"LE": 9999999.0,"Callsign": "BAR","Group": "Cy
an","Role": "Team Member"}
{"UID": "f09865aa-4b2e-44ae-aeb7-f8cefcc97a70","EventType": "t-x-d-d","Timestamp": "2024-11-22T19:25:44.000Z","StartedTs": "2024
-11-22T19:25:44.000Z","Longitude": 0.0,"Latitude": 0.0,"HAE": 0.0,"CE": 9999999.0,"LE": 9999999.0,"Callsign": "null","Group": "n
ull","Role": "null"}
```


Figure 44. JSON data stored in Kafka (Vo 2024)

6.4.4 TAK fleet and its units

As mentioned previously, all TAK-integrated server users belong to a Tactics fleet, referred to as TAK, which was created via the Tactics administrative user interface. The TAK fleet has all members of TAK users on the TAK-integrated server. Currently, the process of adding TAK users into the fleet is conducted manually via the administrative interface. This manual configuration, while functional, presents several challenges, including the potential for human errors, delays in user deployment, low efficiency, and limited scalability. Therefore, future developments are being explored to facilitate a dynamic configuration of TAK users. This innovative approach aims to automate the server integration process, enhancing the overall responsiveness of the fleet to changing operational requirements. The following figures illustrate the representation of the TAK fleet within the fleet tree and all active TAK users on the Tactics map:

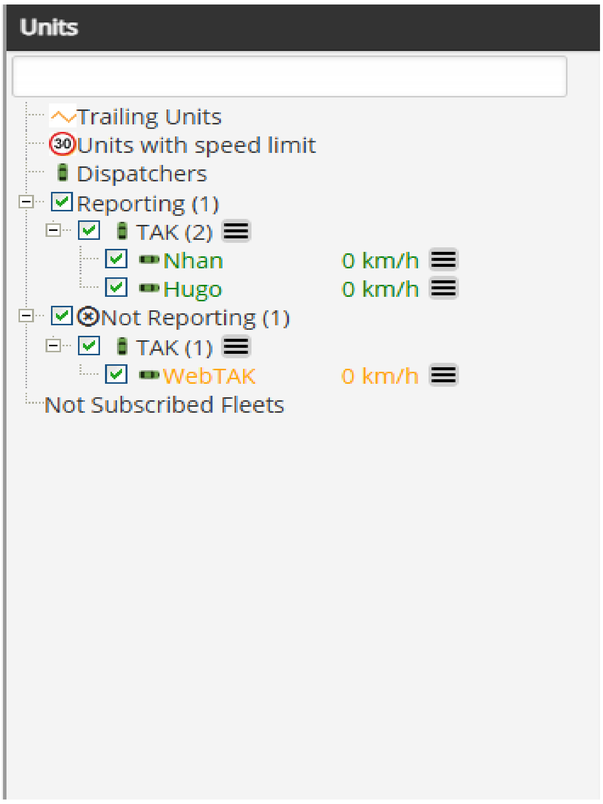

Figure 45. TAK fleet (Vo 2024)

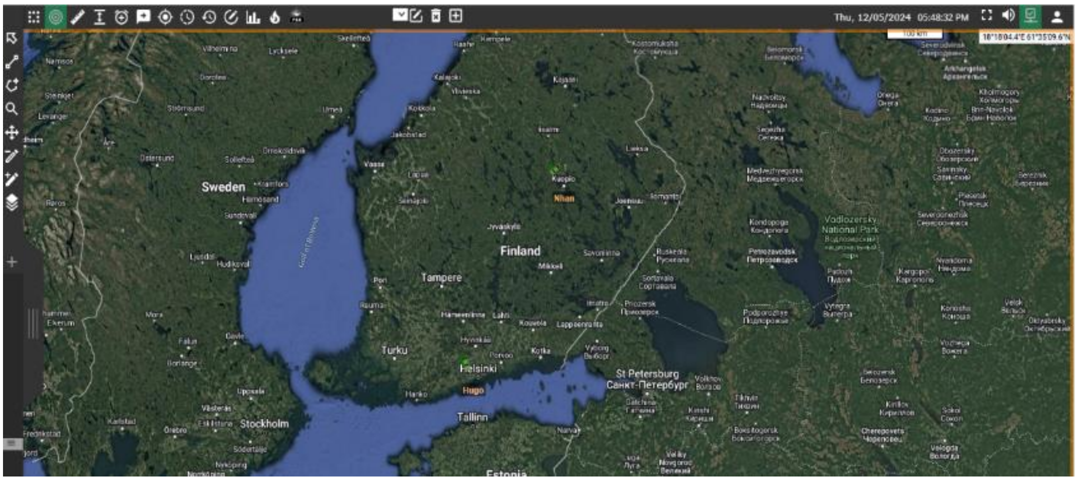

Figure 46. Active TAK users on Tactics map (Vo 2024)

6.4.5 Simulated critical mission involving TAK fleet and Tactics fleet

All fleets in a mission will be monitored through the administrative user interface, serving a centralized command and control hub. In other words, the administrator configures the system, manages users, assigns units and fleets to missions, and monitors the system. Users subscribe to unit locations, manage geo-fences, messages, and statuses, and report to the admin. This is designed to facilitate real-time oversight of all operational units, enabling administrators to issue commands, track progress, and ensure effective coordination among diverse teams.

The following figure demonstrates the admin view of a simulation when the TAK fleet and the FS Länsi-Uusimaa fleet involve in a critical mission:

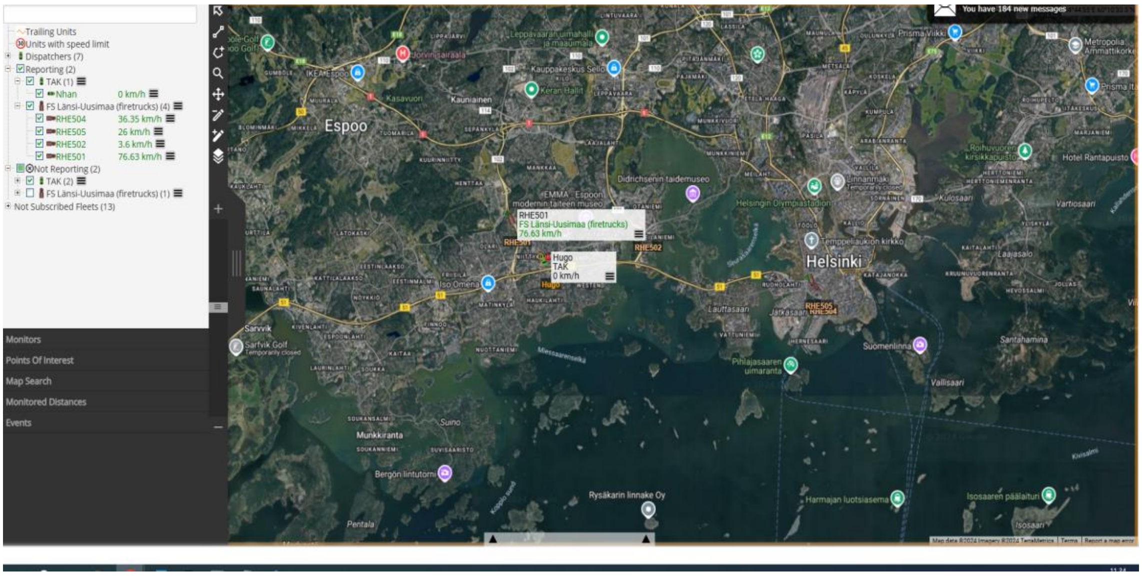

Figure 47. TAK users and Tactics users on the Tactics map (Vo 2024)

From the end user perspective, however, overview visibility is restricted. Users are only permitted to view their team members within the same fleet, unless specific settings are set by the admin. This design serves several crucial purposes, particularly in the context of sensitive operations. Limiting visibility to team members enhances operational security. The risk of sensitive data being leaked is minimized, which is crucial in missions where actions and locations of units must be confidential to prevent adversaries from gaining insights. Moreover, this design helps to reduce overload for end users. In complex missions, the large amount of information can be overwhelming, so focusing on users' attention allows them to concentrate on their tasks and responsibilities without being distracted by activities of other units in different fleets.

This simulated mission outlines the admin perspective and the end-user perspective of the TAK fleet and Tactics fleet. By understanding the roles and visibility restrictions, stakeholders can operate effectively to achieve the outcome of critical missions.

6.5 Testing and review

Testing is a crucial step in any software development process, so various tests were conducted to ensure all components work as intended.

Firstly, several tests were conducted in the integrated TAK server's CRUD operations to see whether all get, create, update, and delete functions worked properly. Those tests mainly focused on reviewing web service methods built in the web service layer of COREMOTE Tactics.

Secondly, the Kafka TAK Adapter and Tactics Kafka Adapter were reviewed. Those two adapters are the core components for this ATAK integration, as they act as a connection between the TAK server and the Tactics server. They were tested not only in normal scenarios but also several unprecedented scenarios to check if something happened unexpectedly. The adapters' logs would provide informative logs to troubleshoot the bugs.

Lastly, testing user visualization on the map configured to Tactics was considered as the most crucial testing criteria, as the integration would not be effective if end users of Tactics could not view users in real time on the map. Checking if the map was correctly configured to Tactics was the first thing, followed by data from the Kafka topic. The Kafka topic was sent successfully to Tactic's backend. A Google Map API key was installed and stored in the AVL configuration of the Tactics Manager UI, which is shown below:

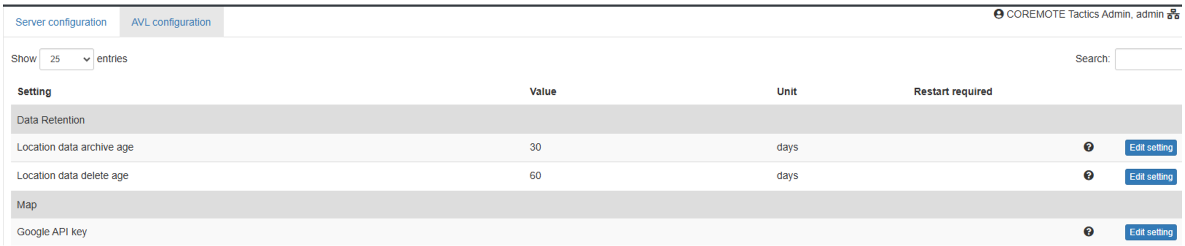

Figure 48. Google API key (Vo 2024)

This structured testing plan enhanced the final product's quality and promoted accountability and transparency throughout the development lifecycle. By following this process, the thesis was a success that delivers solutions that align with the strategic goals of Mentura.

6.6 Documentation and training

Documentation development is a critical process since it involves describing the design and architecture of the integration and each TAK adapter. Comprehensive documentation serves various purposes, making it an invaluable asset for both current and future development efforts. Maintaining up-to-date documentation is essential for ensuring that the system can be effectively managed and enhanced over time due to several benefits, such as allowing developers to understand the existing design and architecture faster, decreasing the risks of errors, and reducing unnecessary costs because of misunderstanding the concept. Additionally, an explanation of the current architecture should be included to minimize the time spent on understanding the system in the beginning. This not only improves productivity but also fosters knowledge-sharing within the team.

Documentation includes creating a user guide on deployment and an operation and maintenance guide, which should cover all crucial steps and be based on the end-user perspective. Every UI view and the functionality (search/create/modify/delete) need to be included. Moreover, the description should be explained precisely and concisely, following the user-centered approach. Understanding the users' needs, technical proficiency, and potential challenges they may face are keys to creating an effective document. Discussing with potential end users during the documentation process can provide valuable insights, enhancing the quality and relevance of the documentation.

To conclude, comprehensive documentation that includes user guides for deployment, operation, and maintenance is essential so that users can effectively interact with the system. Focusing on the end user perspective makes the documentation useful for users in operating and maintaining the system properly.

7 FUTURE DIRECTIONS

The current ATAK integration only developed the CoT stream, which makes the integration become promising when more ATAK features are available under the same roof as COREMOTE Tactics. Using Kafka makes the integration scalable, having the possibility of adding data package feed and video feed from the TAK server. Based on the below diagram, other ATAK features can be integrated by making a producer adapter and a consumer adapter.

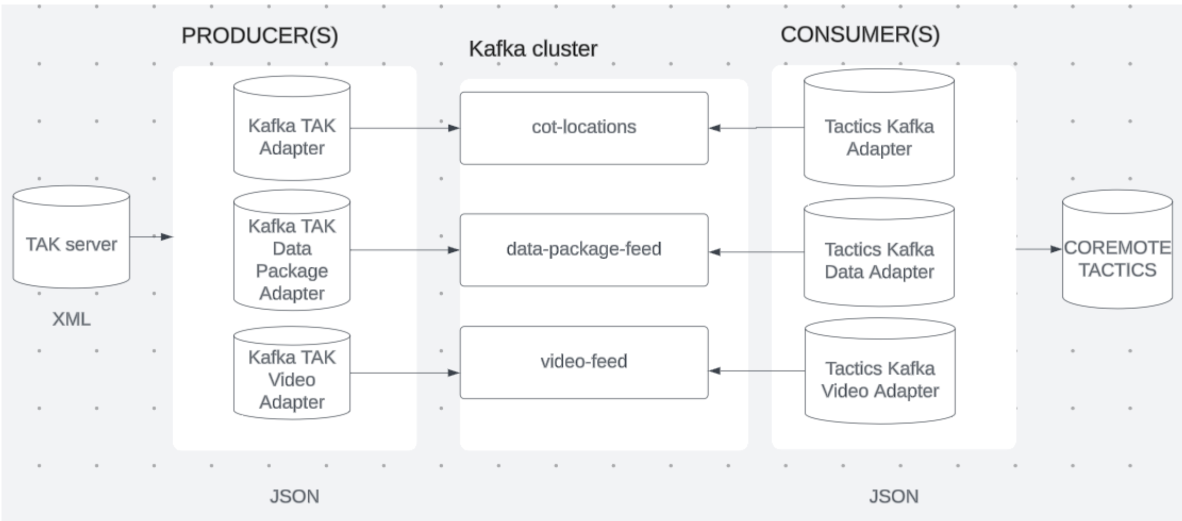

Figure 49. Future ATAK integration architecture diagram (Vo 2024)

For future enhancements, diving deeper into the TAK related resources and documentation is essential, providing a deeper understanding of other existing business models and products to improve the current ATAK integration, which only has CoT stream integrated. This requires a careful analysis of the resources, testing results, feedbacks and insights from conferences relevant to the TAK ecosystem. On the other hand, making connections with organizations in the critical communication market is a must, as it provides deeper insights into how they do business with this open-source platform. The unsuccessful attempts during the development process allowed to understand what technologies need to be carefully considered and learnt when building Tactics components for ATAK features. Another question raised is whether there is a better event streaming platform than Kafka, which could lessen the unnecessary complex of using Zookeeper when multiple topics are created in Kafka.

As the critical communication market has been more acquired, the demand of improving current technologies has also been increased, leading to a bigger market for not only big organizations but also for small companies eager to impact the world through innovative systems. The growing recognition of the critical communication market is driven by several factors. The need for reliable and secure communication systems has become higher. In an era having natural disasters, public health emergencies, and security threats, the ability to communicate swiftly and effectively can mean the difference between life and death. Take the TIG Series of remote access solutions as an example. Plugging Trellisware Multinet Bridge Radio into a TIG, connecting various remote sites and teams into a unified network via failover WAN bearers (Xi Systems Consultancy Ltd 2024).

Organizations responsible for public safety, law enforcement, and disaster management are seeking advanced critical communication technologies that can provide real-time information sharing and co-ordination. The benefits of critical communication platforms are extremely crucial for those organizations. For instance, in healthcare, swift communication without disruption can enhance patient outcomes during emergencies, while in public safety and law enforcement, effective coordination can improve safety for police officers, fire fighters, etc.

In summary, the growing reputation of the critical communication market is driving an urgent need for technological improvements, enhancing public safety and supporting critical missions across various sectors. More and more organizations entering the market creates a dynamic environment for advancements that can save lives and improve operational effectiveness. This upcoming trend promises to deliver significant benefits, contributing to a safer and more secure world.

8 DISCUSSION

This part aims at providing more generic information about TAK ecosystem and how it could be utilized to make a greater impact on the current critical communication market.

Since the TAK ecosystem was published in 2010, it has changed the way critical communication platforms are developed. Its impact is critical not only for military usage but also for civilian usage. A consistent CoT stream is one of the core technology stacks from ATAK, which is useful for critical communication platforms like COREMOTE Tactics. Some potential civilian use cases were raised while working on this thesis work, which are considered promising improvements for COREMOTE Tactics:

- Search and rescue in hazardous areas
- Hiking and trekking
- Traffic Camera Monitoring
- Emergency button connects straight to public safety

Living in an era when the Covid-19 pandemic has just finished, it is crucial to recognize the advanced technological improvements in critical communication platforms worldwide for a safer world. Due to the current chaotic situation with the increase of natural disasters and geopolitical tensions between nations, there has been a rise in the demand for technologies that can maintain national security and operational effectiveness. Military and defense organizations focus on developing secure communication channels that can withstand potential disruptions. The ability to communicate effectively in high-disruption environments is vital for strategic decision-making and coordination among various agencies.

9 CONCLUSION

This thesis explored the development and implementation of the ATAK (Android Team Awareness Kit) integration in COREMOTE Tactics, focusing on enhancing situational awareness and operational efficiency in critical missions. Several technologies used in the project, JavaScript for the frontend, Java with Apache Torque for web service layer, a Java backend with a PostgreSQL database, and Kafka as the event streaming platform, have made the integration become a robust and scalable solution for tactical operations. The work carried out here discovered ways to utilize ATAK features in COREMOTE Tactics, which would provide a solid working base for further developments that integrate other technologies.

The core objective of this thesis was to integrate a TAK server in COREMOTE Tactics, specifically in the context of existing Tactics architecture. This ATAK integration is considered as a working base for other potential third-party servers in the future, which has several phases such as requirement gathering, analysis, design, development, and release. Even though the thesis objective was achieved, several questions raised during the development process reveal that there can be improvements for this ATAK integration. Future research could address those questions by integrating more ATAK features and building custom plugins. Furthermore, this integration provides a roadmap not just for understanding TAK ecosystem, but also for significant advancement of the current COREMOTE Tactics system.

To conclude, the most crucial thesis objective was achieved, which is to integrate TAK server in COREMOTE Tactics, highlighting the contributions of Mentura in the critical communication market with this intensive research project.

REFERENCES

Artificial intelligence has been used in the work as follows:

ChatGPT 2024. OpenAI. GPT-3.5. Accessed for language check, November 2024. https://chat.openai.com.

Amalgame Consult. 2024. Portfolio. Amalgame Consult. https://www.amalgame-consult.com/index.php/portfolio/. Accessed on May 20, 2024.

Apache. 2024a. What is Apache Torque? Apache. https://db.apache.org/torque/torque-6.0/index.html. Updated on August 08, 2024. Accessed on November 23, 2024.

Apache. 2024b. Apache Ant. Apache. https://ant.apache.org/. Accessed on November 23, 2024.

Apache. n.d. Using Apache Ant. Apache. https://ant.apache.org/manual/using.html. Accessed on November 23, 2024.

Carbon. 2024. Development Updates. LinkedIn. https://www.linkedin.com/feed/update/urn:li:activity:7169355298908942336/. Accessed on May 20, 2024.

CivTAK. n.d. CivTAK/ATAK Platform. CivTAK. https://www.civtak.org/. Accessed on May 1, 2024.

ETSI. 2024. TETRA Introduction. ETSI. https://www.etsi.org/technologies/tetra. Accessed on August 18, 2024.

JayR. 2024. Building a TAK Server. mytecknet.com. https://mytecknet.com/lets-build-a-tak-server/. Accessed on May 1, 2024.

JTB Defense. 2024. TAK Development Resources. JTB Defense. https://jtbdefense.com/tak-dev. Accessed on May 20, 2024.

Kristan, Michael J., Jeffrey T. Hamalainen, Douglas P. Robbins, Patrick J. Newell. 2009. Cursor-on-Target Message Router User’s Guide. MITRE. https://www.mitre.org/sites/default/files/pdf/09_4937.pdf. Accessed on November 23, 2024.

Largs & Millport. 2020. Britain's Technological Advantage on the Battlefield. Largs & Millport News. https://www.largsandmillportnews.com/news/national-news/18805945.britain-race-technological-advantage-battlefield-says-wallace/. Accessed on November 23, 2024.

Mentura. 2020. OMG_COREMOTE Tactics_2020. Mentura Group Oy. Espoo, Finland. Accessed on April 26, 2024.

Mentura. 2023. UG_COREMOTE Tactics_2023. Mentura Group Oy. Espoo, Finland. Accessed on April 26, 2024.

Mentura. 2024a. Corporate Website. Mentura Group Oy. https://www.menturagroup.com/. Accessed on August 18, 2024.

Mentura. 2024b. CTS Demo Platform. Mentura Group Oy. https://ctsdemo.menturagroup.com/. Accessed on May 2, 2024.

Royal Canadian Mounted Police. 2019. App gives real-time view of operations. Royal Canadian Mounted Police. https://www.rcmp-grc.gc.ca/en/gazette/are-here. Accessed on November 23, 2024.

Riskaware. 2024. The Development of the ATAK App and TAK. Riskaware. https://www.riskaware.co.uk/insight/the-development-of-the-atak-app-and-tak/. Accessed on August 15, 2024.

TAK Product Center. 2024. ATAK 5.1 Software User Manual. TAK Product Center. https://tak.gov/products/atak-civ. Accessed on May 2, 2024.

United States Department of Defense. 2020. TAK Prototype. GitHub. https://github.com/deptofdefense/AndroidTacticalAssaultKit-CIV/tree/main/takproto. Accessed on July 12, 2024.

Vo, Nhan. 2024. Mentura Group Oy. Kuopio, Finland: Mentura Group Oy.

Xi Systems Consultancy Ltd. 2024. Updates from Xi Systems Consultancy. LinkedIn. https://www.linkedin.com/company/xi-systems-consultancy-ltd/posts/?feedView=all. Accessed on November 23, 2024.