Sistema de Monitoreo en Tiempo Real con APRS Trackdirect para la Gestión de Incendios Forestales

Patrick Hugo Nepveu Nelson, Lesmes Torres González
Escuela de Ingeniería Electrónica
Instituto Tecnológico de Costa Rica
Cartago, Costa Rica
Emails: patrick.nepveu@estudiantec.cr, lesmestorres@estudiantec.cr

***Resumen*—Este trabajo presenta la implementación progresiva de un sistema local basado en APRS Trackdirect para la recepción, procesamiento y visualización de datos en tiempo real sin depender de infraestructura externa. El proyecto integra tecnologías de radioaficionados (APRS) y comunicaciones de largo alcance y bajo consumo (LoRa), cumpliendo con la normativa costarricense del espectro radioeléctrico. La solución fue desarrollada inicialmente en un entorno virtualizado y posteriormente migrada a un servidor físico de bajos recursos, permitiendo evaluar su desempeño real en condiciones operativas. El sistema funciona incluso sin conexión a Internet, incrementando la resiliencia de los equipos de emergencia en situaciones críticas como incendios forestales.**

***Index Terms*—APRS, Trackdirect, LoRa, Monitoreo en tiempo real, Incendios forestales, Docker, Virtualización, Telecomunicaciones, PNAF.**

I. Introducción

Los incendios forestales representan uno de los desastres naturales más críticos debido a su rápida propagación, la dificultad de acceso y la complejidad de su control en zonas rurales y montañosas. La protección de brigadistas y la coordinación operativa requieren sistemas de monitoreo robustos capaces de funcionar aun sin conectividad a Internet.

Los sistemas de rastreo tradicionales dependen de servicios en la nube que requieren conectividad constante, lo cual no es viable en zonas afectadas por emergencias. Por ello, este proyecto implementa un sistema completamente local basado en APRS Trackdirect que permite:

- Recibir datos APRS generados por trackers en campo.
- Procesar y almacenar la información en un servidor local.
- Visualizar en tiempo real las posiciones en un mapa accesible desde un navegador web.

La solución se implementó inicialmente en una máquina virtual (VM) para desarrollo y pruebas controladas. Posteriormente, se migró a un servidor físico de bajo consumo, permitiendo validar su comportamiento real en condiciones similares a escenarios operativos como los de la Isla de Chira.

II. Fundamentos

II-A. APRS

APRS es un protocolo digital utilizado para transmitir información en tiempo real como posición, telemetría y mensajes. Opera sobre AX.25 y aprovecha estaciones iGate y digipeaters para extender la cobertura [1].

II-B. LoRa

LoRa es una tecnología de comunicación de largo alcance y bajo consumo basada en modulación Chirp Spread Spectrum (CSS), lo cual la hace resistente a interferencias y adecuada para entornos remotos [2][3].

II-C. Regulación del Espectro en Costa Rica

El Plan Nacional de Atribución de Frecuencias (PNAF) regula el uso de bandas ISM y las restricciones sobre modos digitales como APRS en VHF. LoRa opera en bandas libres, mientras que APRS requiere licencia de radioaficionado [4].

II-D. VirtualBox y Ubuntu Server

VirtualBox permite la virtualización de sistemas operativos aislados, facilitando pruebas sin riesgo para el sistema principal. Ubuntu Server proporciona estabilidad y ligereza para aplicaciones de red [5][6]. La configuración del la maquina virtual se puede apreciar en la Figura 1.

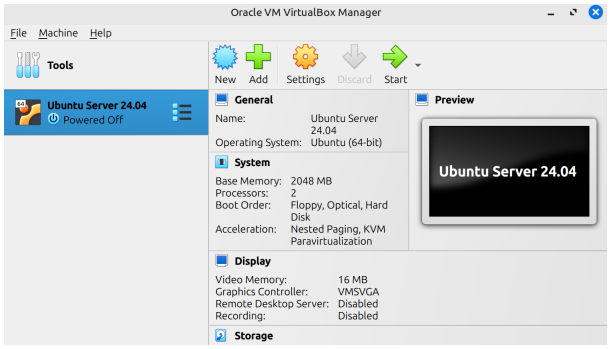

Figura 1. Máquina virtual Ubuntu Server en VirtualBox.

II-E. Docker y Contenedores

Docker encapsula aplicaciones junto con sus dependencias, facilitando el despliegue homogéneo del sistema tanto en entornos virtualizados como en servidores físicos. En este proyecto, la arquitectura se implementó mediante múltiples contenedores especializados, cada uno responsable de una función específica dentro del sistema APRS Trackdirect. Esta separación modular mejora la estabilidad, simplifica el mantenimiento y permite aislar fallos sin afectar el resto del sistema [7][8].

- **aprsc**: Implementa un servidor APRS local basado en *aprsc*. Recibe tramas APRS provenientes del receptor LoRa o VHF, las valida y las pone a disposición del colector mediante sockets TCP. Actúa como la "puerta de entrada" del sistema.
- **collector**: Es el módulo encargado de conectarse al servidor *aprsc*, decodificar las tramas APRS utilizando `aprslib` y almacenarlas en la base de datos PostgreSQL. Opera de forma continua y es el componente con mayor carga computacional, ya que procesa cada paquete recibido en tiempo real.
- **websocket**: Publica los datos recién almacenados en la base de datos a través de un servidor WebSocket. Permite la actualización en tiempo real del mapa sin recargar la página web.
- **cron**: Ejecuta tareas programadas, tales como limpieza periódica de tablas, regeneración de índices y mantenimiento del repositorio de dispositivos OGN. Contribuye a la estabilidad a largo plazo del sistema.
- **web**: Contiene Apache2 y la interfaz web de Trackdirect. Sirve la aplicación al usuario final, permitiendo la visualización de posiciones, rutas y telemetría directamente desde un navegador dentro de la red local.
- **db**: Implementa PostgreSQL junto con los scripts de inicialización, creación de tablas e índices utilizados por el sistema. Aloja todos los paquetes recibidos y actúa como núcleo del almacenamiento de datos.
- **reverseproxy**: Expone la aplicación hacia la red mediante un proxy inverso basado en Apache. Permite acceder al servidor web y al servicio WebSocket.

La Fig. 2 muestra la ejecución conjunta de los contenedores mediante el comando `docker-compose up`, donde se observa la estructura modular del sistema que permite que cada servicio funcione de manera independiente pero coordinada.

```
server@server:~/trackdirect$ docker-compose up -d
[+] Running 8/8
 ✔ Network trackdirect_default           Created
 ✔ Container trackdirect-db-1            Started
 ✔ Container trackdirect-aprsc-1         Started
 ✔ Container trackdirect-websocket-1     Started
 ✔ Container trackdirect-cron-1          Started
 ✔ Container trackdirect-web-1           Started
 ✔ Container trackdirect-collector-1     Started
 ✔ Container trackdirect-reverseproxy-1  Started
```

Figura 2. Contenedores desplegados mediante `docker-compose up`.

III. Problemática

La falta de conectividad en zonas rurales afecta el monitoreo en tiempo real durante incendios forestales. Regiones como la Isla de Chira carecen de infraestructura robusta, lo que compromete la seguridad del personal de emergencia al no contar con herramientas de rastreo local. La ubicación geográfica de la Isla de Chira se muestra en la Figura 3.

Figura 3. Mapa de la Isla de Chira.

IV. Objetivos

IV-A. General

Implementar un servidor local basado en APRS Trackdirect para la recepción y visualización en tiempo real de datos APRS

IV-B. Específicos

- Configurar APRS Trackdirect en un servidor Linux (VM y físico).
- Desarrollar una interfaz web de visualización accesible en red local.
- Documentar instalación, configuración y operación.

V. Metodología

La metodología del proyecto siguió dos etapas: primero en entorno virtualizado, donde se realizó la configuración inicial y luego en servidor físico de bajos recursos donde se migró.

V-A. Implementación en Máquina Virtual

- Se creó una máquina virtual en VirtualBox con Ubuntu Server.
- Se instalaron Webmin, Docker, Docker Compose y utilidades, como se muestra en la Figura 4
- Se desplegó Trackdirect en contenedores.
- Se realizaron pruebas de recepción, decodificación, estabilidad y monitoreo.

La VM permitió desarrollar y ajustar la arquitectura del servidor antes de migrarlo aun equipo físico.

VI. Implementación en Servidor Físico

VI-A. Hardware de bajos recursos utilizado

Se utilizó un Acer Aspire V5-123 con:

- CPU: AMD E1-2100 (2 hilos) @ 1.0 GHz.
- RAM: 1.4 GB disponible.
- GPU: Radeon HD 8210.
- OS: Ubuntu 24.04.3 LTS (kernel 6.8.0).

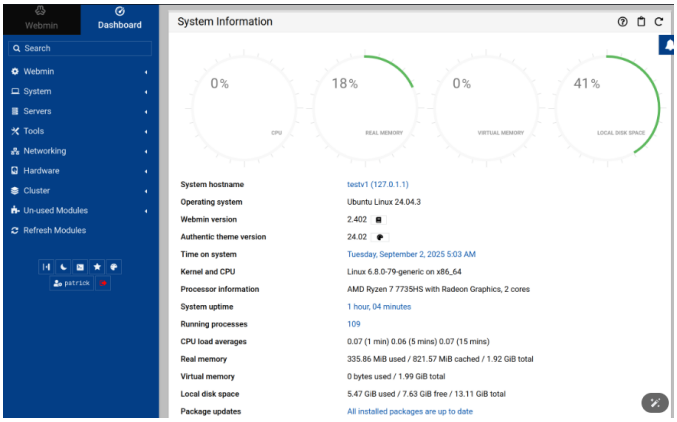

Figura 4. Webmin ejecutándose en el servidor virtual.

Figura 5. Servidor físico de bajos recursos, Acer Aspire V5-123

VI-B. Instalación de Webmin

Descargado desde el sitio oficial, verificado con SHA-256 e instalado con, se muestra la implementación en la FIgura 6:

```
dpkg -i webmin-current.deb
sudo apt-get -f install
```

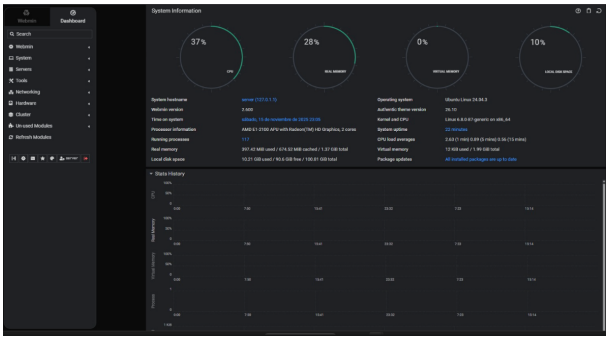

Figura 6. Webmin ejecutándose en el servidor físico.

VI-C. Despliegue de Trackdirect

La instalación fue idéntica a la VM:

- Clonado del repositorio.
- Edición de configuraciones.
- Ejecución con Docker Compose.

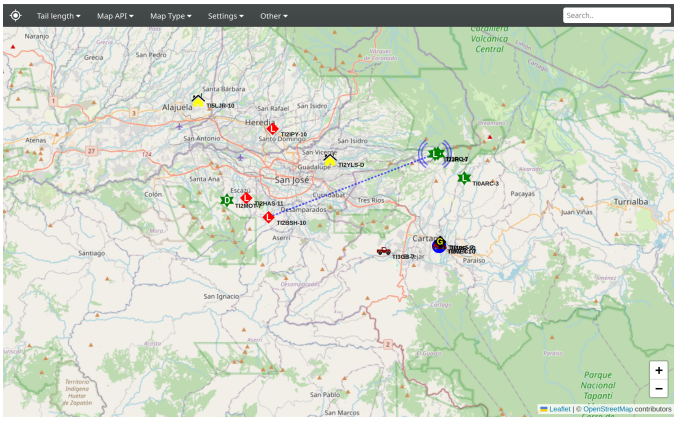

Figura 7. Interfaz de Trackdirect en servidor físico.

VII. Vista Funcional y Arquitectura

A nivel de la arquitectura del servidor, se implementó el esquema mostrado en la Figura 8.

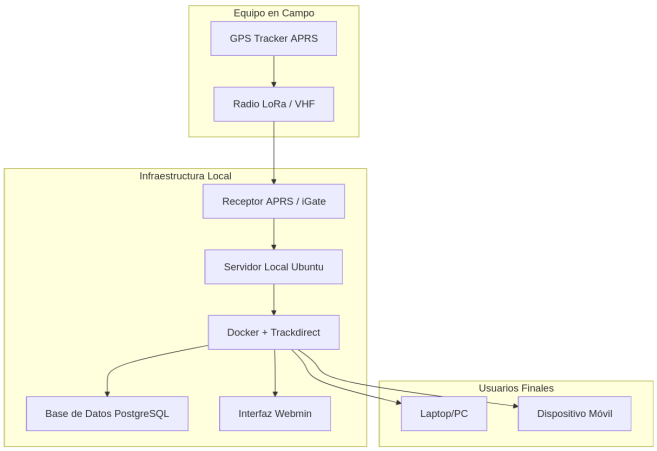

Figura 8. Arquitectura modular basada en contenedores.

Los dispositivos APRS transmiten vía LoRa/VHF, un receptor local envía las tramas al servidor y Trackdirect procesa, almacena y publica los datos en una interfaz web local.

VIII. Resultados

El sistema fue evaluado en dos entornos con características computacionales distintas: una máquina virtual configurada para desarrollo y un servidor físico de bajos recursos que simula las condiciones reales previstas para aplicaciones en campo. Las pruebas se centraron en latencia, estabilidad, carga del sistema, uso de memoria y capacidad de procesamiento continuo de paquetes APRS.

VIII-A. *Resultados en Máquina Virtual (VirtualBox)*

La plataforma de pruebas inicial consistió en una máquina virtual con 2 vCPUs y 4 GB de RAM. Este entorno permitió la validación progresiva de la arquitectura.

La latencia medida entre la recepción, procesamiento y publicación interna de datos se mantuvo estable alrededor de 5 ms, sin variaciones significativas durante la operación prolongada. El uso del CPU permaneció entre el 5–7 % y el consumo de memoria osciló entre 410–530 MB. Durante 8 horas de operación continua no se registraron reinicios de contenedores ni errores asociados a la base de datos.

VIII-B. *Resultados en Servidor Físico*

La migración al servidor físico permitió evaluar el sistema bajo las restricciones reales de hardware. La plataforma utilizada consistió en un procesador AMD E1-2100 de 1 GHz con 1.4 GB de RAM disponible y un disco duro mecánico.

Durante 24 horas continuas de operación, el sistema procesó un total de 1 815 468 paquetes APRS, verificado mediante el comando:

```
docker exec -it trackdirect-db-1 psql -U
trackdirect -c "SELECT count(*)
FROM public.packet
```

La tasa de decodificación se mantuvo en 99.72 %, con pérdidas asociadas únicamente a tramas incompletas o duplicadas provenientes de APRS-IS.

En cuanto al rendimiento del sistema, el uso del CPU osciló típicamente entre 60–70 %, con picos ocasionales que alcanzaron el 100 % durante momentos de sincronización inicial o inserciones masivas en la base de datos. Esto es consistente con las limitaciones del procesador de bajo consumo.

El uso de memoria alcanzó valores cercanos a 1.2 GB en operación sostenida, lo cual provocó actividad de *swap* en periodos de carga elevada. La utilización de memoria de intercambio no afectó la estabilidad del sistema, pero confirma que la arquitectura opera en el límite de la capacidad instalada.

La latencia extremo a extremo se mantuvo alrededor de 8 ms en promedio, con picos ocasionales de 15–20 ms durante procesos de alto uso del CPU. No se registraron caídas de servicios, interrupciones de WebSocket ni reinicios automáticos de contenedores.

VIII-C. *Comparación entre Entornos*

En la Tabla I se presenta un desglose ampliado:

Cuadro I
COMPARACIÓN DE RENDIMIENTO ENTRE ENTORNOS

Métrica	VirtualBox	Servidor Físico
Latencia	5 ms	8 ms
CPU uso	5–7 %	60–70 % (picos 100 %)
RAM uso	410–530 MB	1.2 GB + swap
Paquetes procesados	—	1 815 468
Decodificación	—	99.72 %
Disponibilidad	100 %	100 %

VIII-D. *Análisis General*

Los resultados demuestran que, pese a operar sobre hardware significativamente limitado, Trackdirect mantiene un desempeño estable y una capacidad de procesamiento adecuada para aplicaciones de monitoreo en tiempo real. El colector mantiene la inserción continua de paquetes sin pérdidas significativas; la base de datos PostgreSQL conserva integridad y tiempos de respuesta aceptables incluso con intercambio de memoria; el servidor WebSocket sostiene comunicación sin interrupciones en presencia de carga elevada; y la arquitectura basada en Docker permite un funcionamiento consistente y aislado entre módulos, independientemente del entorno de despliegue.

Estos resultados confirman que el sistema es viable para su uso en zonas remotas con infraestructura mínima, asegurando operación autónoma y resiliente ante escenarios de emergencia.

VIII-E. *Ajustes Estéticos y Personalizaciones*

Además de la configuración funcional del sistema, se realizaron varias modificaciones menores para adaptar la interfaz y la presentación del servidor a las necesidades del proyecto. Entre estos ajustes destaca la edición del archivo `about.php` dentro de la interfaz web de Trackdirect, donde se reemplazó el contenido predeterminado por información del equipo desarrollador, detalles del sistema y una descripción contextualizada del propósito del despliegue.

También se modificaron aspectos visuales de la interfaz, tales como títulos, textos descriptivos y algunos elementos de estilo, con el fin de ofrecer una presentación más coherente con el caso de uso del proyecto. Estas modificaciones no alteran el funcionamiento interno de Trackdirect, pero mejoran la claridad y la identificación del sistema cuando se utiliza en entornos de emergencia o demostración técnica. De la Figura 9 se puede observar la interfaz personalizada.

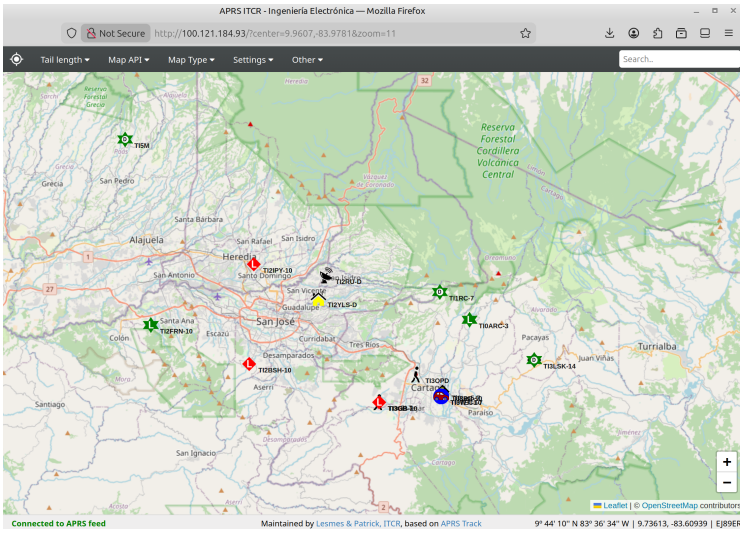

Figura 9. Interfaz de Trackdirect tras las modificaciones cosméticas.

IX. Recomendaciones

- Integración con Prometheus y Grafana.
- Automatización con Ansible.
- Redundancia mediante múltiples receptores APRS.

X. Conclusiones

El proyecto logró implementar un sistema de monitoreo completamente local, autónomo y resiliente. La migración de entorno virtual a servidor físico permitió validar la viabilidad real de Trackdirect incluso en hardware limitado. La arquitectura basada en Docker facilitó portabilidad, repetibilidad y estabilidad en ambos entornos, demostrando que la solución es adecuada para emergencias en zonas remotas.

Referencias

[1] APRS Foundation Inc., "Aprs: Automatic packet reporting system," https://www.aprs.org/, 2025, consultado: 12 agosto 2025.

[2] Semtech Corporation, "Lora technology overview," https://www.semtech.com/lora, 2025, consultado: 12 agosto 2025.

[3] L. Alliance, "Lorawan 1.0.4 specification," 2020.

[4] Superintendencia de Telecomunicaciones (SUTEL), "Plan nacional de atribución de frecuencias (pnaf)," Documento oficial, 2023, alcance N.º 99 a La Gaceta N.º 95, mayo 2023. [Online]. Available: https://sutel.go.cr

[5] Oracle, "Virtualbox user manual," https://www.virtualbox.org, [En línea].

[6] Canonical, "Ubuntu server documentation," https://ubuntu.com/server/docs, [En línea].

[7] C. Boettiger, "An introduction to docker for reproducible research," in *ACM SIGOPS Operating Systems Review*, vol. 49, no. 1, 2015, pp. 71–79.

[8] D. Inc., "Docker documentation," https://docs.docker.com, [En línea].