PacketMancer

Juan Felipe Orozco Cortes

October 2025

PacketMancer

Del `.pcap` al diagnóstico accionable en un solo comando.

PacketMancer es una herramienta de código abierto para el análisis de red, escrita en Rust. Nace de la frustración de pasar horas buscando la aguja en el pajar digital que son las capturas de paquetes. Su misión es automatizar el primer nivel de diagnóstico, permitiendo a los ingenieros enfocarse en resolver problemas, no en encontrarlos.

Este proyecto se está construiguiendo en público. Puedes seguir el viaje, los desafíos técnicos y las decisiones de diseño en mi blog: La Verdad en los Paquetes (Substack).

Características Principales (MVP Actual)

- **Análisis de Estado de TCP**: Identifica problemas de salud en conversaciones TCP, incluyendo:
  - Retransmisiones
  - Paquetes Duplicados y Fuera de Orden
  - Eventos de Ventana Cero
  - Eventos de ACK Duplicado (indicador de pérdida de paquetes)
- **Motor de Análisis Modular**: Construido sobre un `Engine` que permite añadir nuevos detectores (DNS, HTTP, etc.) en el futuro.
- **Procesamiento Eficiente**: Lee archivos `.pcap` y `.pcapng` en modo streaming, permitiendo analizar capturas de varios gigabytes sin agotar la memoria.
- **Salida Dual**: Ofrece un reporte legible para humanos en la consola y una salida estructurada en formato JSON para la integración con otros scripts y herramientas.

Empezando

Prerrequisitos

PacketMancer está construido en Rust y depende de `libpcap`.

1. **Instalar Rust**: Si aún no lo tienes, instálalo a través de `rustup`:

   ```
   curl --proto '=https' --tlsv1.2 -sSf https://sh.rustup.rs | sh
   ```

2. **Instalar** `libpcap` (dependencia de sistema):
   - En Debian/Ubuntu:

     ```
     sudo apt-get update && sudo apt-get install -y libpcap-dev
     ```

   - En Fedora/CentOS:

     ```
     sudo dnf install -y libpcap-devel
     ```

Instalación y Compilación

1. **Clona el repositorio**:

```
git clone https://github.com/topassky3/packetmancer.git
cd packetmancer
```

2. **Compila el proyecto**:

```
cargo build --release
```

   El binario final se encontrará en `target/release/packetmancer`.

Uso

Ejecuta PacketMancer desde la línea de comandos, pasándole la ruta a un archivo de captura.

Comando Básico:

```
cargo run --release -- --file /ruta/a/tu/captura.pcap
```

Ejemplo de Ejecución

```
cargo run --release -- --file captures/tcp-ecn-sample.pcap --top 2
```

Salida en Consola:

```
Iniciando analisis del archivo: captures/tcp-ecn-sample.pcap

--- Reporte del Detector de Salud TCP ---
Se encontraron 1 conversaciones TCP distintas.

Top 2 conversaciones por volumen de paquetes:
 - Flujo: 1.1.23.3:46557 <-> 1.1.12.1:80/TCP
   -> C->S: Paquetes: 309, Retrans.: 1, Fuera de Orden: 0,
      Ventana0: 0, ACKs Dup. (eventos>=3): 0
   <- S->C: Paquetes: 170, Retrans.: 0, Fuera de Orden: 0,
      Ventana0: 0, ACKs Dup. (eventos>=3): 0

--- ANALISIS COMPLETADO ---
```

Contribuyendo

¡Este es un proyecto de código abierto y las contribuciones son bienvenidas!

- **Reportar Bugs**: Si encuentras un problema, por favor abre un *issue* con la mayor cantidad de detalles posible.
- **Sugerir Funcionalidades**: ¿Tienes una idea para un nuevo detector o una mejora? ¡Abre un *issue*!
- **Pull Requests**: Si quieres contribuir con código, ¡excelente! Por favor, abre un *issue* primero para discutir el cambio.

Licencia

Este proyecto está bajo la Licencia MIT.