

# REPORTE VULNERACIONES FINANZAS 2

Fuente: /var/www/html/qareportes/app/Http/Controllers/Cloud/ReportController.php

## 1) Nombre del reporte

- Nombre: Reporte Vulneraciones Finanzas 2
- Código / Alias: report\_vulneraciones\_finanzas2\_YYYY\_MM
- Responsable: Equipo de Reportes Cloud
- Propósito (1 línea):

## 2) Alcance temporal

- Campo(s) de fecha utilizados: fecha (varios orígenes: emp\_code\_attemp.fecha, emp\_cambio\_precio.fecha, Solicitud de Pagos.fecha)
- Tipo de rango (diario / mensual / personalizado): Mensual (por año y mes, dinámico según fecha actual).
- Inclusión de horas (sí/no): Sí → 00:00:00 - 23:59:59.]
- Zona horaria: La del servidor

## 3) Origen de datos (tablas/APIs)

### 3.1 Conexión / Base: CENTOS

- Tabla / Recurso: emp\_code\_attemp
- Alias (si aplica): No aplica
- Campos utilizados (clave → descripción):
  - ter\_id → Terminal
  - user\_attempt → Usuario
  - fecha → Fecha del intento

- id → Identificador único (usado en COUNT)
- Condiciones (WHERE) aplicadas: Rango de fechas entre fecha inicio y fecha fin.
- Tipo de unión con otras fuentes (JOIN y llave): No aplica.
- Observaciones (ej. particiones, índices): Se usa groupBy(ter\_id, user\_attempt, DATE(fecha)).

### 3.2 Conexión / Base: CAPACITACION

- Tabla / Recurso: Solicitud de Pagos (API → resource/Solicitud de Pagos)
- Alias (si aplica): sp
- Campos utilizados (clave → descripción):
  - name → Identificador del registro
  - id\_de\_sucursal → Sucursal asociada
  - usuario\_que\_registro\_el\_pago → Usuario que registró el pago
  - fecha → Fecha de registro
- Condiciones (WHERE) aplicadas:
  - concepto = "Transacción / Compensaciones"
  - fecha >= fecha\_inicio
  - fecha <= fecha\_fin
- docstatus != 2
- Tipo de unión con otras fuentes (JOIN y llave): No aplica (consulta directa a recurso vía API).
- Observaciones: usuario\_que\_registro\_el\_pago se limpia quitando el dominio (@xxx).

### 3.3 Conexión / Base: ERP EMPRESARIAL

- Tabla / Recurso: emp\_cambio\_precio
- Alias (si aplica): No aplica
- Campos utilizados (clave → descripción):

- usuariocambia → Usuario que realiza el cambio de precio
  - fecha → Fecha del cambio
  - id → Identificador único (usado en COUNT)
- Condiciones (WHERE) aplicadas: Rango de fechas entre fecha inicio y fecha fin.
  - Tipo de unión con otras fuentes (JOIN y llave): No aplica.
  - Observaciones: Se agrupa por usuario y fecha.

### 3.4 Conexión / Base: EMPRESARIAL

- Tabla / Recurso: emp\_terminal
  - Alias (si aplica): No aplica
  - Campos utilizados (clave → descripción):
    - ter\_id → Identificador del terminal
    - ter\_nombre → Nombre del terminal
- 
- Condiciones (WHERE) aplicadas: Ninguna.
  - Tipo de unión con otras fuentes (JOIN y llave): No aplica.
  - Observaciones: Sirve para enriquecer con nombre de terminal en los reportes.

### 3.5 Conexión / Base: CAPACITACION

- Tabla / Recurso: Branch (API → resource/Branch)
  - Alias (si aplica): No aplica
  - Campos utilizados (clave → descripción):
    - ideentificador → ID de la sucursal
    - name → Nombre de la sucursal
    - agente → Indicador si es agente
    - concesionario → Indicador si es concesionario
- 
- Condiciones (WHERE) aplicadas: Ninguna (trae todos).
  - Tipo de unión con otras fuentes (JOIN y llave): No aplica.
  - Observaciones: Se usa para distinguir si la sucursal es agente o concesionario.

### 3.6 Conexión / Base: EMPRESARIAL

- Tabla / Recurso: emp\_usuario
- Alias (si aplica): No aplica
- Campos utilizados (clave → descripción):
  - usr\_terminalid → Terminal asignado al usuario
  - usr\_rol → Rol del usuario
  - usr\_id → Identificador del usuario
  - usr\_alias → Nombre o alias del usuario
- Condiciones (WHERE) aplicadas: usr\_id IN (usuarios encontrados en las otras consultas).
- Tipo de unión con otras fuentes (JOIN y llave): No aplica (pero conceptualmente enriquece con info de usuario).
- Observaciones: Se usa para obtener detalles adicionales del usuario que aparecen en el reporte final.

## 4) Filtros globales del reporte

- Inclusiones (must-have): Registros dentro de rango de fechas.
- Exclusiones (reglas de descarte): docstatus != 2 en Solicitud de Pagos.
- Reglas por estado / habilitado: Se guarda status = 1 (ok) o 0 (error).
- Parámetros externos (si el usuario puede filtrarlo): No expuestos al usuario (definidos internamente).

## 5) Transformaciones y Reglas de negocio

- Derivaciones de campos (cómo se calculan):
  - usuario\_que\_registro\_el\_pago → antes del @ (si aplica).
  - Mapeo de usuario ↔ terminal desde varias fuentes.
- Mapeos de estado:
  - tipo = cambio\_precio | compensaciones\_transacciones | error\_clave.

- Reglas de validación (p.ej., sólo registros validados):
  - Verifica existencia de tabla destino (verifyExistTable).
  - Si no existe, se crea con índices (fecha, created\_date, status).
- Reglas condicionales (si X entonces Y): Si usuario no está en array\_usuario\_agencia, se asignan valores vacíos.

## 6) Estructura de salida

- Tabla/archivo destino: report\_vulneraciones\_finanzas2\_YYYY\_MM
- Particionado / sufijo: Por año y mes (YYYY\_MM).
- Clave(s) primaria(s) o únicas: id (autoincrement).
- Columnas del reporte (nombre → tipo → descripción):
  - id → INT → PK autoincrement
  - fecha → DATE → fecha de registro
  - terminal → VARCHAR(20) → id terminal
  - terminal\_nombre → VARCHAR(255) → nombre terminal
  - usuario → VARCHAR(255) → id usuario
  - nombre\_completo → VARCHAR(255) → alias
  - cargo → VARCHAR(255) → rol usuario
  - tipo → VARCHAR(255) → categoría del evento
  - cantidad → VARCHAR(20) → ocurrencias
  - created\_date → DATETIME → fecha inserción
  - status → tinyint(4) → 1=activo
- Ordenamiento por defecto: No explícito (se asume fecha).

## 7) Frecuencia y operación

- Frecuencia de actualización: PENDIENTE (Esperando info de Eduardo)

- Ventana que recalcula (días/meses): Últimos 2-3 meses (dependiendo del mes actual).
- Tamaño de lote / paginado (chunking): Inserciones en lotes de 900 registros.
- Tolerancia a fallos / reintentos: No especificado, pero registra en emp\_reportes\_validation.
- Tiempos de ejecución esperados: PENDIENTE (Esperando info de Eduardo)

## 8) Calidad y controles

- Validaciones previas/post:
  - Validación de existencia de tabla.
  - Inserción de estado en emp\_reportes\_validation.
- Controles de duplicados: Truncate si la tabla existe antes de insertar.
- Métricas de completitud (qué se monitorea): cantidad por usuario/terminal/fecha.

## 9) Dependencias externas

- APIs / servicios y endpoints:
  - Apis:
    - APICAPACITACION/resource/Solicitud de Pagos
    - APICAPACITACION/resource/Branch
- Autenticación / headers: Token fijo → token fc60669957e6bdc:b207d472e96c9df
- Mapeos y contratos de datos: Respuesta esperada con campo data

## 10) Historial de cambios

---

Revisión #1

Creado 2025-10-24 15:30:16 -05 por Elian

Actualizado 2025-10-24 15:30:22 -05 por Elian