Universidad de Cienfuegos “Carlos Rafael Rodríguez”

Facultad de Informática

Carrera de Ingeniería Informática

Sistema para la centralización, análisis y procesamiento de trazas de servicios telemáticos para la Red UCf.

Trabajo de diploma para optar por el título de Ingeniería en Informática

Autor:

Jorge Luis Rivero Pérez.

Tutor:

MSc. Alexis Gómez Domínguez. PDVCupetSA “Refinería de Petróleo de Cienfuegos”.

Consultantes:

MSc. Daimarelys Acevedo Cardoso. Universidad de Cienfuegos.

MSc. Vladimir Carrera Martínez. Universidad de Cienfuegos.

Cienfuegos, Cuba

Curso 2008 – 2009

Agradecimientos

Gracias a mi mamá, a mi papá, y a mi hermana por su apoyo, preocupación y por ser constantes guías de mi vida.

Gracias a mi novia Miriam por apoyarme, ayudarme y estar siempre a mi lado.

Gracias a mi tutor Alexis por compartir todos sus conocimientos y por su ayuda.

Gracias a todos mis compañeros de estos 5 años de estudio, en especial a mis amigos: Emmanuel, Anel, Reinier, Enmanuel, Aldo, Daniel, Oscarito, Alina, Víctor y Yeikel.

Gracias a todos mis profesores, en especial a Dailyn, Vladimir, Daimarelys y Chou por ayudar a la conformación de este trabajo.

Gracias a Dios.

Dedicatoria

Dedico este trabajo a mis padres y a mi hermana, por tanto apoyo, constancia y dedicación. A ellos debo todo lo que soy.

Resumen

El presente trabajo está orientado al desarrollo de un sistema para la centralización, análisis y procesamiento de trazas de servicios telemáticos para la red de la Universidad de Cienfuegos, con el fin de mantener un control más estricto sobre dichos servicios, a partir de la generación de reportes y detección de fallas o intrusiones. El mismo surge debido a que el análisis de las trazas es una actividad determinante para lograr un correcto funcionamiento en las redes de área local, pero esto no se realiza con la periodicidad necesaria, debido a que consume mucho tiempo. Actualmente el análisis y procesamiento de la información contenida en las trazas de los servicios de la red de la Universidad de Cienfuegos, es realizado de dos maneras: manualmente y mediante el uso de algunas herramientas que por ser muy específicas para un tipo de servicio se encuentran muy descentralizadas, por tanto no cumplen con todas las expectativas, lo que propició la realización de este trabajo.

Índice de figuras

Índice de tablas

Introducción

Con el crecimiento de Internet y las capacidades de las nuevas redes de datos, las vulnerabilidades y amenazas informáticas a los sistemas de cómputo y de comunicaciones han aumentado. Es por ello que el análisis de las trazas generadas por los servicios telemáticos y los equipos de la red, se ha convertido en una herramienta muy útil a la hora de prevenir y evitar muchos de estos problemas, permitiendo mantener un control más estricto sobre dichos sistemas, a partir de la generación de reportes, detección de fallas o anomalías y la posibilidad de avisos mediante alarmas. Por otra parte este análisis es una de las tareas que más tiempo consume y por consiguiente que menos se hace, pues en ocasiones resulta ser tediosa. Así mismo dada la cantidad de mensajes informativos que se generan en una red telemática y que se almacenan en los diferentes equipos, lograr detectar en forma manual los problemas o anomalías es muy dificultoso y con una alta probabilidad de error debido a la gran cantidad de falsos positivos[1]. Esto se ve aumentado cuando se usa un sistema de almacenamiento centralizado, donde la información proviene de varias fuentes distintas, sin embargo la centralización brinda beneficios desde el punto de vista de la seguridad informática y facilidades en el proceso de análisis que se concentra en un único lugar. Un método comúnmente utilizado para explotar las posibilidades de la centralización es el uso de herramientas informáticas para el análisis y procesamiento de trazas, en realidad existen muchas soluciones de monitoreo y la mayoría se basan en sumarizar la información de las trazas de días previos siendo esto es muy útil para la generación de reportes y análisis posterior a una falla o intrusión, pero no tanto para la resolución de problemas debido a que un administrador no puede actuar en forma proactiva, previamente a que el error ocurra siendo esta una debilidad de dichos sistemas pues muchas fallas o accesos no autorizados se ven precedidos por mensajes que de haber sido detectados, podrían haber permitido tomar acciones preventivas. Otro aspecto negativo de muchas de las herramientas

[1] Falsos positivos: Un falso positivo es un error por el cual un Software reporta que el resultado de una operación es “alarmante”, cuando en realidad no lo es.

existentes es el hecho de ser Software privativo lo que en el caso de nuestro país se convierte en un problema debido a las dificultades de acceder al producto legalmente.

Situación del problema:

En la red telemática de la Universidad de Cienfuegos se generan diariamente una gran cantidad de mensajes informativos asociados a los servicios telemáticos existentes, los cuales son almacenados en los mismos servidores donde se encuentran los servicios que los generan, a excepción de las trazas relacionadas con el servicio de Proxy (Squid) que se almacena en una base de datos externa. Esta situación hace que el análisis diario de los eventos ocurridos en la red sea una tarea incómoda y en ocasiones complicada que requiere una gran cantidad de tiempo, lo que hace que no se realice con la periodicidad necesaria. Otras de las deficiencias existentes en la red son sin lugar a dudas la imposibilidad de poder correlacionar los eventos de servicios diferentes, pero que interactúan entre sí y la no generación de alarmas que informen la ocurrencia de situaciones críticas.

Actualmente el análisis y procesamiento de la información contenida en las trazas de los servicios de la red de la UCf, es realizado de dos maneras: manualmente (inspección visual) y mediante el uso de algunas herramientas de Software que por ser muy específicas para un tipo de servicio, se encuentran instaladas en el mismo lugar donde se genera la traza.

Problema:

Ante esta situación y por la importancia que representa para cualquier centro contar con un sistema que facilite y potencie el análisis y procesamiento de la información contenida en las trazas de los distintos servicios de su red; queda definido como problema, la ausencia de un sistema para la centralización, análisis y procesamiento de las trazas de los principales servicios telemáticos de la Red UCf.

Por estas razones nos planteamos las siguientes interrogantes:

¿Cuáles son las diferentes tecnologías que pueden utilizarse para almacenar de forma centralizada las trazas generadas por los principales servicios en la red telemática de la Universidad de Cienfuegos?

¿Qué herramienta de Software desarrollar para realizar el procesamiento y análisis de las trazas almacenadas centralmente?

Debido a ello se decidió desarrollar un sistema para la centralización, análisis y procesamiento de trazas; encaminado a eliminar las deficiencias del modelo actual, capaz de elevar los niveles de seguridad, estabilidad, disponibilidad, confiabilidad y no repudio de la misma, garantizando además minimizar las incomodidades y la ocurrencia de errores durante el procesamiento, objeto del presente trabajo.

Objeto de estudio y campo de acción:

La presente investigación es desarrollada en la Red UCf y se considera como objeto de estudio de la misma los mensajes generados por los dispositivos de red y por el servicio Web, así como la centralización de los mismos.

Idea a defender:

Con el sistema propuesto serán mejoradas las tareas de supervisión y control sobre los servicios telemáticos, garantizando así, la detección de cualquier problema sobre los mismos.

Justificación de la investigación:

Con esta investigación se logra prevenir y dar solución a las vulnerabilidades de los sistemas de cómputo y de comunicaciones, debido a que al tener centralizada la información contenida en las trazas de los diferentes servicios telemáticos de la red, se ven aumentadas las opciones de monitoreo. Ya no será solamente sumarizar la información de las trazas de días previos, para la generación de reportes y análisis posteriores a una falla o intrusión, sino que permitirá correlacionar los eventos de servicios diferentes, pero que interactúan entre sí y la generación de alarmas que informan la ocurrencia de situaciones críticas. Con

esto queda evidenciada la posibilidad de emprender acciones proactivas, encaminadas a garantizar la estabilidad, disponibilidad y seguridad de la red.

Objetivo general:

Desarrollar una herramienta de Software para el análisis y procesamiento de las trazas centralizadas de los principales servicios telemáticos de la Red UCf, mediante el empleo de Software Libre.

Objetivos específicos:

- Seleccionar las tecnologías de procesamiento de trazas y los modelos de diseños de almacenamiento de trazas centralizadas que permitan alcanzar los niveles de seguridad esperados.
- Implementar un sistema para la centralización y almacenamiento de las trazas de los principales servicios telemáticos de la Red UCf.
- Analizar los elementos del sistema a automatizar.
- Investigar sistemas automatizados existentes vinculados al objeto de estudio.
- Validar la implementación final, mediante el empleo de herramientas de Software y el análisis del comportamiento del sistema.

Metodología de trabajo:

Para el desarrollo de la investigación se utilizarán diferentes métodos y técnicas que permitirán enfrentar el problema. Estos métodos y técnicas favorecerán el cumplimiento de las siguientes tareas:

- Se realizará una revisión de la bibliografía técnico-especializada para la construcción del marco teórico que permitirá conocer el estado del arte referente a las tecnologías empleadas en los sistemas de análisis y procesamiento de trazas, principio de funcionamiento, ventajas y desventajas asociadas.

- Se entrevistará al personal capacitado para conocer más a fondo conceptos que se deben tener en cuenta para la investigación.
- Se seleccionarán las tecnologías de procesamiento de trazas y los modelos de diseños de almacenamiento de trazas centralizadas más efectivos que permitan alcanzar los niveles de seguridad esperados.
- Se realizarán pruebas prácticas y se utilizarán aplicaciones que permitirán reunir elementos para la definición de las técnicas y mecanismos de seguridad a emplear, así como la simulación de situaciones reales en aplicaciones virtuales y en escenarios de pruebas.
- Se validará la implementación final mediante el empleo de herramientas de Software y el análisis del comportamiento del sistema.

Para la conformación y mejor comprensión, este trabajo se ha separado en cinco capítulos:

Capítulo I.- "Fundamentación teórica": El contenido de este capítulo es la base de la fundamentación teórica del tema que se va a desarrollar. Se expone una descripción de bs conceptos relacionados con el análisis y procesamiento de trazas y sus implementaciones. Se describe el objeto de estudio, el flujo actual de los procesos, así como un análisis y descripción de soluciones existentes, describiendo la solución propuesta.

Capítulo II.- "Modelo del dominio": El modelo del dominio ayuda a modelar y describir la solución propuesta. Este capítulo presenta una descripción detallada del diagrama del modelo de objeto del dominio, su enunciado y descripción de las reglas del dominio para garantizar las restricciones que existen en el mismo.

Capítulo III.- "Requisitos": En este capítulo se describe y analiza el modelo de sistema de acuerdo con las exigencias de la metodología RUP. Se identifican los requerimientos funcionales y no funcionales, se definen los actores y casos de uso del sistema, se enumeran una serie de diagramas que sirven de guía para la implementación del modelo de sistema, como es: el diagrama de casos de uso del sistema.

Capítulo IV.- "Construcción de la solución propuesta": En este capítulo se describe la solución propuesta a través de los diagramas de clases, de los modelos lógico y físico de los datos, y del diagrama de implementación, así como otros principios de diseño del sistema, como son: diseño de la interfaz entrada y salida, tratamiento de errores y concepción de la ayuda.

Capítulo V.- "Análisis de prueba y estudio de factibilidad": En este capítulo se describen los resultados obtenidos al someter a prueba al sistema, así como sus respectivas validaciones. Además se detalla el estudio de factibilidad económica del sistema mediante el análisis de puntos de casos de uso, teniendo en cuenta la estimación del esfuerzo, el cálculo de costos, los beneficios tangibles e intangibles y el análisis de costos y beneficios.

Capítulo I. Fundamentación teórica

El contenido de este capítulo es la base de la fundamentación teórica del tema que se va a desarrollar. En él se expone una descripción de los conceptos asociados al problema. Se brindan análisis comparativos con otras soluciones existentes describiendo además la solución propuesta. Además, se presentan definiciones, ventajas, utilidades, forma correcta de aplicación, y características específicas que reflejan las potencialidades del uso y la importancia de los procesos en que intervienen.

1.1 Conceptos asociados al dominio del problema

Servidor de red:

"Un servidor de red es una computadora que maneja peticiones de datos, correo electrónico, transferencia de archivos de otras computadoras (clientes) y variados servicios telemáticos. También puede referirse a un Software específico, como lo es el servidor WWW. Una computadora puede tener distintas aplicaciones de Software del tipo servidor, proporcionando muchos servidores a clientes en la red. Por ejemplo, las computadoras que contienen sitios Web se llaman servidores ya que "sirven" recursos de Web para aplicaciones cliente como los navegadores o Browsers."[1]

Dispositivo de red:

Los dispositivos de red son aquellos que permiten la interconexión de los Hosts que participan de una red, los mismos se describen generalmente en función de las operaciones que realizan según el modelo de referencia OSI o la arquitectura TCP/IP como dispositivos de nivel 1, 2, 3, 4 o superior, también suelen llamarse como dispositivos de nivel físico, de nivel de enlace, de nivel de red, de nivel de transporte o de nivel de aplicación haciendo alusión al nombre de cada uno de dichos niveles.

Servicios telemáticos:

Los servicios telemáticos están muy relacionados con los usuarios finales, son el elemento fundamental de una red de computadoras pues los enlaces, servidores

de red y equipos de interconexión no tienen utilidad práctica si no existen servicios telemáticos soportados en la red. Ejemplos clásicos de ellos son:

- Correo electrónico
- Acceso a páginas Web.
- Bases de Datos de información.
- Mensajería instantánea
- Video Conferencias.
- Foros de discusión.
- Laboratorios virtuales.
- Noticias

Trazas:

"Las trazas son el primer registro de la actividad en los sistemas y redes. Ellos contienen el registro de acciones y eventos que tienen lugar en un sistema. Las trazas de un sistema son una parte primaria de la seguridad y pueden ser usados en la detección de ataques e intrusiones, así como en el análisis de fallas de Hardware y Software." [2]

Syslog centralizado:

"El programa Syslog, es una interfase que provee un Framework estándar para que tanto programas como el mismo sistema operativo puedan generar mensajes que pueden ser almacenados localmente o ser enviados a un Host remoto. Originalmente escrito para Unix, se convirtió en un estándar que se usa en muchos sistemas operativos y dispositivos de red. En un sistema de Syslog centralizado, un servidor común recibe todos los mensajes de Syslog de todos los sistemas de la red. Esto incluye trazas de los servidores Unix/Linux/Windows etc., cortafuegos, y dispositivos de red (Routers, Switches, etc.)." [2]

Syslog-ng

"Si bien el demonio de Syslog estándar de Linux (*syslogd*) es adecuado para concentrar información de otros Syslog, se decidió utilizar *syslogng (syslog new generation)* que es un reemplazo directo de Syslog.

El syslog original permite que los mensajes sean organizados basándose en los pares *priority/facility*; *syslogng* agrega la posibilidad de filtrar basándose en el contenido de los mensajes usando expresiones regulares.

Syslogng permite guardar los mensajes en una base de datos y también usar TCP en lugar de UDP para enviar los mensajes a otro servidor." [2]

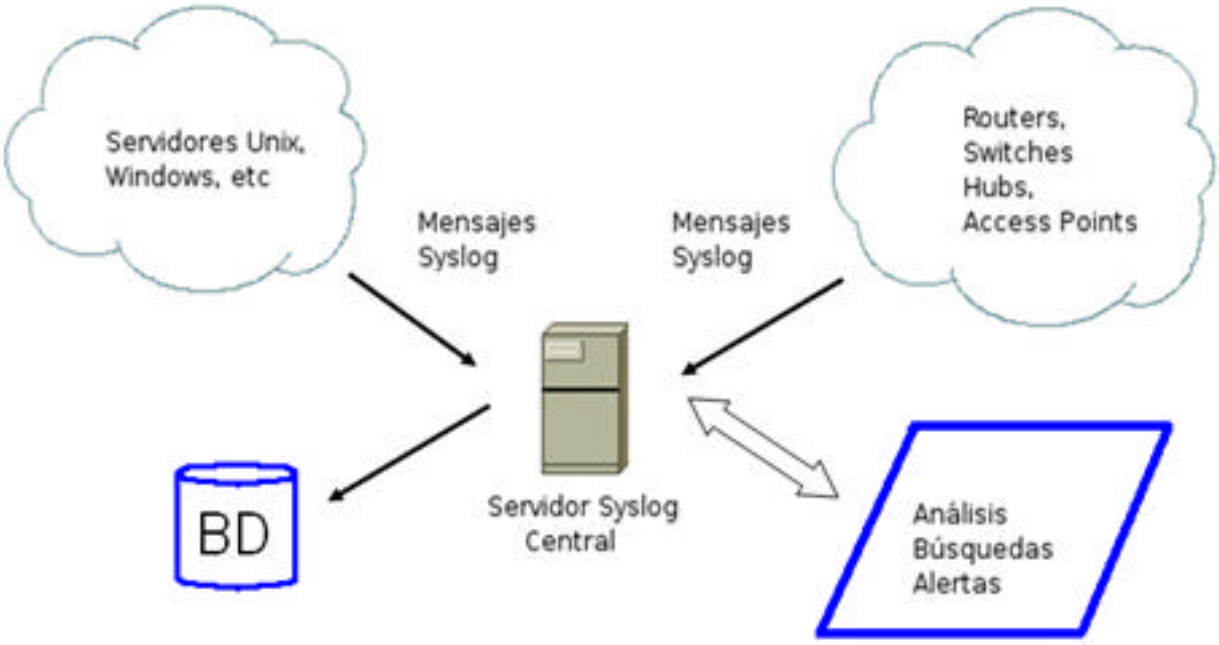


Figura 1. Syslog Centralizado

Script:

Un Script es un conjunto de instrucciones que permiten la automatización de tareas, creando pequeñas utilidades. Es muy utilizado para la administración de sistemas UNIX. Son ejecutados por un intérprete de línea de órdenes y usualmente son archivos de texto.

1.2 Descripción del objeto de estudio

1.2.1 Flujo actual de los procesos y análisis crítico de la ejecución de estos

La red de la Universidad de Cienfuegos (UCf) cuenta aproximadamente con 14 servidores y varios equipos de red, los cuales brindan una gran variedad de servicios telemáticos que se encuentran distribuidos en dichos servidores. Cada uno de estos servicios genera una o varias trazas (registros de informaciones). De estos servicios se benefician aproximadamente 3000 usuarios entre estudiantes, trabajadores docentes y no docentes. La red interconecta cerca de 500 computadoras entre laboratorios de estudiantes, locales de profesores o trabajadores, la residencia estudiantil y sedes universitarias municipales (SUMs).

Las tareas relacionadas con el procesamiento y análisis de las trazas de los servicios telemáticos en la red de la Universidad de Cienfuegos se realiza actualmente de forma manual (inspección visual), no existe como se explicó anteriormente, un sistema automatizado que permita realizar este proceso, sólo para las trazas relacionadas con el acceso a Internet existen algunos módulos del SICC-IP que permiten conocer algunos datos relacionados con la utilización de dicho servicio. El resto de los servicios genera sus propias trazas y las almacenan de manera local y en los formatos propios de cada uno de ellos y con las configuraciones generadas por defecto, además en las trazas almacenadas existen referencias continuas a números de códigos y de errores estandarizados y publicados en los RFC pero que por la gran cantidad de códigos existentes no se pueden memorizar completamente y tener que estar revisando un listado constantemente se convierte en un trabajo lento y tedioso. Tampoco existen algoritmos de análisis para determinar situaciones anómalas en la red ni mecanismos de generación de alarmas ante estas situaciones. Si bien es cierto que existe un elevado control de los eventos que ocurren y es posible determinar los detalles de la ocurrencia de los mismos, su realización se vuelve muy compleja requiriendo largas jornadas de trabajo y una participación total del administrador y de su nivel de preparación y experiencia para saber que trazas revisar e

interpretar correctamente los resultados, existiendo por consiguiente un margen de error elevado.

1.3 Sistemas automatizados existentes vinculados al campo de acción

Existe una gran variedad de sistemas informáticos desarrollados para realizar el análisis y procesamiento de las trazas que generan los servicios telemáticos en una red de computadoras, cada uno de ellos tiene características específicas y permiten controlar uno o varios servicios. A continuación se describen algunos de estos sistemas que por sus características y utilización se consideran importantes.

1.3.1 GFI EventsManager 8

Este es un sistema propietario, hecho para sistemas Windows, lo que lo descarta de ser una solución al problema; pero a pesar de esto, **"**es una solución de procesamiento de registros que proporciona control y administración en toda la red de registros de sucesos Windows, registros W3C y eventos Syslog generados por sus recursos de red. Es por ello que con independencia de su no utilización como solución al problema, se analizo a fin de tomar experiencias en el manejo de las trazas de los sistemas. GFI EventManager incluye un procesamiento inteligente de sucesos que procesa los registros y presenta la información en informes centralizados, sencillos y amigables.

Posee características que lo hacen novedoso y robusto, como son:

1. Registro de sucesos centralizado.

2. Motor de escaneo de alto rendimiento.

3. Alertas en tiempo real.

4. Soporte extendido de registros de sucesos.

5. Recoge datos de sucesos distribuidos por una WAN en una base de datos central.

6. Gestión de registros de sucesos basados en reglas.

7. Perfiles de análisis de registros de sucesos.

8. Facilidad para crear o personalizar informes incluyendo informes estándar tales como:

- Informes de uso de cuentas.
- Informes administrativos de cuentas.
- Informes de cambios de directiva.
- Informes de acceso a objetos.
- Informes administrativos de aplicaciones.
- Informes de servidor de impresión.
- Informes de sistema del registro de sucesos Windows.
- Informes de tendencias de eventos.” [3]

La versión 8 soporta Windows Server 2008 y Vista events, SNMP Traps generado por fuentes de red heterogéneas y SQL Server Audits. GFI EventsManager ha alcanzado también el estatus “Certified for Microsoft’s Windows Server 2008”. La nueva plataforma estará disponible a finales de este mes de febrero.

GFI EventsManager 8 es una solución que cubre dos funciones principales: monitorización, y administración y archivo de sucesos. La primera, ayuda a los administradores a monitorizar el estado y seguridad de toda la red, mejorando la disponibilidad, mientras que la función de administración y archivo permite al administrador centralizar los sucesos de múltiples orígenes en varios formatos de manera que sea más sencillo identificar deficiencias, proporcionar trazas detalladas de auditoría y cumplir diversas regulaciones. La disponibilidad de la información, que se encuentra y analiza fácilmente, ayuda a la investigación forense, reduciendo la complejidad del volumen de registros de sucesos, así como el coste y tiempo necesarios para analizar la información.

1.3.2 Bitacora 4.0

“Bitacora 4.0, presentada por S21sec como la nueva versión de su plataforma de

gestión de trazas; es una plataforma centralizada de trazas destinada a gestionar y explotar los eventos de una organización mediante alarmas e indicadores de seguridad. Permite gestionar la seguridad a través de los eventos, además de controlar el cumplimiento de las normativas de seguridad y calidad.

Bitacora recolecta y almacena las trazas de los sistemas de una organización en formato nativo para facilitar la gestión y análisis de cualquier tipo de evento, dotando a las trazas de carácter judicial. Reporta datos de seguridad, y además ofrece indicadores que ayudan en la toma de decisiones de las diferentes unidades de negocio. De esta manera, actúa como un Centro Estructural de Análisis y Alarmas mediante un Cuadro de Mandos que facilita la gestión empresarial"[4].

Esta herramienta es pues una solución válida para adaptarse al cumplimiento del estándar PCI DSS, ya que se encuentra en posición de cumplir los siete requisitos contemplados en el requerimiento 10 de dicho estándar, 'Revisión y Monitorización de todos los accesos a los recursos de red y a los datos de titulares de tarjetas'. Para empezar, Bitacora es capaz de relacionar todos los accesos a los componentes del sistema (aquellos que almacenan, procesan o transmiten datos de titulares de tarjetas) con un individuo en concreto, así como registrar todos los eventos exigidos por PCI DSS (accesos de los usuarios a la información, intentos de acceso no autorizados, acciones realizadas por los administradores de los sistemas, etc.) siempre que el dispositivo genere el correspondiente log.

Por otra parte, en relación a la seguridad de los registros, Bitácora permite proteger los archivos de auditoría de posibles modificaciones no autorizadas (mediante firma digital y sellado de tiempo), realizar una copia de respaldo de las trazas de forma inmediata y establecer diferentes perfiles de acceso, con el fin de limitar la visualización de los registros de auditoría a aquellos usuarios que lo necesiten para el ejercicio de sus funciones.

Existen dos últimas exigencias que han sido satisfechas gracias a las nuevas funcionalidades de Bitacora 4.0: la revisión de trazas de todos los componentes del sistema al menos una vez al día, mediante la creación de un conjunto mínimo

de alertas que permita monitorizar estos eventos, y el mantenimiento de un histórico de los registros de auditoría durante al menos un año, con un mínimo de tres meses de disponibilidad online.

Debido a ello, se puede afirmar que Bitacora constituye la herramienta idónea para apoyar la implantación de los requerimientos establecidos en el estándar PCI DSS en relación a la monitorización de los sistemas, favoreciendo y facilitando, al mismo tiempo, el cumplimiento de normativas y estándares aplicables a nivel internacional en materia de gestión de seguridad de la información.

Las herramientas IPtables Log Analyzer, Wallfire Wflogs y FWlogwatch, a pesar de presentar novedosas funcionalidades son específicas para cortafuegos, no pueden ser utilizadas para analizar las trazas de otros servicios telemáticos de una red.

Conclusiones sobre Bitacora 4.0:

Esta es descartada de ser una solución debido a que es una herramienta privativa, y está orientada fundamentalmente a auditorias, lo que la convierte en una herramienta muy específica para un tipo de análisis, no satisfaciendo así las necesidades actuales de la Red UCf.

1.3.3 IPtables Log Analyzer

"IPtables Log Analyzer, es una herramienta para Linux 2.4 ó 2.6 destinada al análisis de las trazas de los cortafuegos. Almacena los eventos en una base de datos; los administradores pueden usar el interfaz Web para acceder a la base de datos. Realiza un análisis basado en reglas, que deben ser definidas por el administrador."[5]

1.3.4 Wallfire Wflogs

"Wallfire Wflogs, es una herramienta de análisis que pertenece al proyecto Wallfire, aunque puede usarse independientemente; el módulo del programa analiza y procesa Netfilter, Ipchains, IPfilter, Cisco PIX, Cisco IOS y ficheros de registro Snort, mostrando los resultados como texto, HTML, XML o en tiempo real. WFlogs no tiene soporte de base de datos, pero puede convertir formatos de ficheros de registros entre Netfilter, IPchains e IPfilter. WFlogs puede procesar los

registros del cortafuego de forma online u offline. Los administradores pueden usar un shell para modificar interactivamente el comportamiento de WFlogs, así como las opciones de filtrado.” [5]

1.3.5 FWlogwatch

“FWlogwatch, herramienta desarrollada por Boris Wesslowski para RUS-CERT en la Universidad de Stuttgart, Alemania. FWlogwatch tiene tres modos de operación:

1. Modo Log Summary:

En este modo el programa genera texto o páginas HTML con el resumen del análisis de los ficheros de registros del cortafuegos.

2. Modo Interactive Report:

En este modo el programa genera automáticamente informes de incidencias que los administradores pueden reenviar a quienes hayan sido afectados por el incidente.

3. Modo Realtime Response:

En este modo FWlogwatch responde a ataques ejecutando Scripts, enviando mensajes de correo o automáticamente modificando las reglas del cortafuego.”[5]

Conclusiones sobre IPtables Log Analyzer, Wallfire Wflogs y FWlogwatch:

IPtables Log Analyzer, Wallfire Wflogs, FWlogwatch son herramientas para analizar las trazas de cortafuegos, lo que las descarta de ser soluciones debido a que son específicas para ese tipo de servicio.

1.3.6 OSSEC

“OSSEC es un Sistema de Descubrimiento de Intrusión, escalable, multiplataforma, y de código abierto, que posee poderosas características de correlación y análisis, incluyendo análisis de trazas, chequeo de integridad de ficheros, monitoreo de los registros de Windows, alertas en tiempo real. El Software sigue una arquitectura cliente/servidor en el que el elemento servidor se ubica en una localización central y la tecnología agente se distribuye sobre

múltiples dispositivos gestionados. Monitoriza las modificaciones de directorio y ficheros, almacena información sobre autenticación y dispara alertas de usuario cuando se producen fallos de autenticación o las entradas de los usuarios resultan sospechosas.

OSSEC funciona con la mayoría de los sistemas operativos, incluidos Linux, OpenBSD, MacOS, Solaris y Windows. Para utilizarlo, los usuarios deben instalar el Software sobre un servidor y después desplegar el agente sobre las máquinas cliente utilizando un asistente de instalación Windows. "La comunicación entre servidor y agente está encriptada. Lo que garantiza la privacidad de los datos log en tránsito."[6]**.**

Conclusiones sobre OSSEC:

A pesar contar con novedosas funcionalidades, OSSEC es un sistema detector de intrusos, lo que lo hace muy específico para un tipo de análisis, además a pesar de funcionar en múltiples sistemas operativos, OSSEC requiere un asistente de instalación para los clientes, que sólo funciona en plataformas Windows. Otra deficiencia encontrada es que con grandes cantidades de datos, puede saturar el ancho de banda.

1.3.7 Turbo Syslog o Syslog Turbo.

"Con una base de datos SQL, Turbo Syslog; colecta, almacena y analiza los mensajes de las trazas. Ahora con soporte para redes IPv4 e IPv6, y disponible para sistemas Microsoft Windows y GNU/Linux.

Beneficios:

1. Almacena los mensajes de las trazas en una base de datos centralizada, para un posterior análisis.
2. Posee un analizador basado en reglas capaz de realizar acciones diferenciadas, según van almacenándose los mensajes de las trazas.
3. Crea poderosas vistas SQL para analizar la red.
4. Brinda la posibilidad de archivar los mensajes que llevan más tiempo almacenados en el sistema, para que puedan ser usados en auditorias." [7]

Syslog Turbo, incorpora novedosas características para el análisis de trazas, como la centralización de las mismas, y el almacenamiento de los mensajes en una base de datos.

1.3.8 Php-Syslog-ng

Pensado para hacer consultas en una base de datos creada directamente por Syslog-ng, se instala y configura rápidamente. Maneja fácilmente registros de muchos anfitriones y ofrece búsquedas personalizadas para requisitos particulares basadas en el anfitrión, la facilidad, la prioridad, el date/time, y el contenido de los mensajes Syslog. También presenta un modo con filtros personalizados para requisitos específicos, que permiten supervisar los sistemas casi en tiempo real.

Conclusiones sobre Turbo Syslog y Php-Syslog-ng:

Las herramientas Turbo Syslog y Php-Syslog-ng, son un poco rígidas a la hora de mostrar los reportes, mostrando lo que está definido en el sistema, no permiten definir criterios a partir de los cuales pudieran generarse reportes y alarmas, evitando así los falsos positivos.

El sistema que se obtiene con la presente investigación brinda la posibilidad y facilidad de trabajar con las trazas generadas de forma que con los conocimientos que tengan los administradores en materia de seguridad informática y protocolos TCP/IP, pueda detectar anomalías y errores que no estaban previstos en la configuración del sistema, además permite evitar o comprobar en mayor medida los falsos positivos, satisfaciendo así las necesidades de los administradores de la Red UCf.

1.4 Tendencias, metodologías y/o tecnologías actuales

Para poder garantizar calidad en el desarrollo de una herramienta de Software es necesario seguir las indicaciones de alguna metodología. Es necesario e importante, antes de llevar a cabo el proceso de desarrollo, hacer un estudio de cuales son las tecnologías actuales, conocidas o no, con el fin de seleccionar y utilizar la más conveniente. En este capítulo se exponen las características fundamentales de la metodología escogida y más conocidas por los autores de

este trabajo, para el desarrollo de Software, el Proceso Unificado de Desarrollo. Se justifican las razones por las cuales ha sido seleccionada para guiar el proceso de desarrollo de la herramienta propuesta.

1.4.1Tecnologías Web

Tecnologías Cliente

Navegador Web

- Mozilla Firefox.

Tecnologías de programación

- HTML.
- Hojas de estilo en cascada (CSS).
- JavaScript.

Tecnologías Servidor

Servidor Web

- Apache.

Servidor de Trazas

- Syslog-ng

Servidor de Base de Datos

- PostgreSQL 8.3

Tecnologías de Programación

- PHP.
- Python

1.4.2 Fundamentación del lenguaje, gestor de bases de datos y de otros Software utilizados

Python

Se escogió el lenguaje Python, debido a las grandes funcionalidades que brinda para el trabajo con ficheros de texto, siendo este el caso de las trazas y por los módulos y paquetes que tiene incorporado para las operaciones en el campo de las redes. Además, Python es un idioma de programación dinámico que es usado en una gran variedad dominios de la aplicación. Python se compara a menudo a Tcl, Perl, Ruby, o Java. Algunos de sus rasgos más importantes son:

1. Posee una sintaxis muy clara y legible.
2. Fuertes capacidades de introspección.
3. Orientado a objetos.
4. Sus expresiones son naturales, similares al lenguaje procedural.
5. Completamente modular, con soporte para herencia de paquetes.
6. Fuerte tratamiento a las excepciones.
7. Tipos de datos dinámicos de alto nivel.
8. Extensiones y módulos fácilmente escritos en C, C++ (o Java para Jython, o idiomas de .NET para IronPython).

Python posee cientos de bibliotecas de apoyo que lo convierten en la solución para casi cualquier tipo de problemas. Permite escribir el código necesario rápidamente.

Python esta disponible para sistemas operativos como: Windows, Linux/Unix, OS/2, Mac, entre otros. [8]

HTML

El sistema tendrá una interfase Web, la cual brindará todas las funcionalidades de gestión, y análisis. Por tal motivo se decidió utilizar HTML como lenguaje, ya que es un lenguaje de marcación diseñado para estructurar textos y presentarlos en

forma de hipertexto, que es el formato estándar de las páginas Web, además permite aglutinar textos, sonidos e imágenes y así como la combinación de los mismos y nos permite la introducción de referencias a otras páginas por medio de los enlaces hipertexto.[9]

CSS

En el sistema se utiliza CSS debido a que como el mismo posee una interfase Web, con la utilización de CSS se logra separar la presentación de la lógica y permite concentrar en un único lugar, la definición de estilos de todos los elementos similares. Mediante un bloque de código situado dentro de la misma página o en un archivo separado, es posible controlar el estilo de ciertos elementos en lugar de dar formato a cada elemento en forma individual. De esta forma, si se decide realizar algún cambio en el diseño, bastará con hacer las modificaciones en un solo lugar, en vez de modificar cada elemento HTML por separado. [10]

PHP

Se escogió como lenguaje PHP para implementar las funcionalidades del sistema, debido a que PHP puede ser desplegado en la mayoría de los servidores Web y en casi todos los sistemas operativos y plataformas sin costo alguno. Cuenta con un conjunto de funciones empaquetadas en bibliotecas que dan la posibilidad de acceder a un servidor de bases de datos y trabajar con él mediante la utilización de las funciones adecuadas en dependencia del servidor; además posee características como: velocidad, estabilidad, seguridad y simplicidad que son pruebas de sus beneficios. Otras de las causas de su uso, es que es un módulo del servidor Web Apache, siendo este el servidor que utilizado en la Red UCf para este tipo de aplicaciones.

El código escrito utilizando este lenguaje puede ser interpretado en sistemas operativos como las diferentes versiones de Unix, Windows (95, 98, NT, ME, 2000, XP,…) y Mac. [11]

JavaScript

JavaScript es un lenguaje de secuencias de comandos basado en objetos e interpretado. Aunque tiene menos capacidades que los lenguajes orientados a objetos de altas prestaciones como C++ y Java, JavaScript es más que suficientemente eficiente para los propósitos para los que está creado, razón por la cual es utilizado en el sistema para realizar algunas tareas, como validaciones de formularios, para de esta manera liberar al servidor Web de estas funciones. JavaScript no es una versión reducida de cualquier otro lenguaje (sólo está relacionado, distante e indirectamente, con Java, por ejemplo), tampoco es una simplificación de ningún lenguaje. Sin embargo, es un lenguaje limitado. Por ejemplo, no es posible escribir aplicaciones independientes en JavaScript y la capacidad de lectura y escritura de archivos es mínima. Más aún, las secuencias de comandos de JavaScript sólo pueden ejecutarse con un intérprete, que bien puede estar en un servidor Web o en un explorador de Web.[12]

PostgreSQL

Una de las principales funcionalidades del sistema a implementar es la capacidad de almacenar toda la información contenida en las trazas de los diferentes servicios en una base de datos. Se escogió PostgreSQL como sistema gestor de base de datos debido a que a pesar de no ser tan rápido en sus consultas, como es el caso de MySQL, posee características como:

Alta concurrencia:

Mediante un sistema denominado MVC (Acceso concurrente multiversión) PostgreSQL permite que mientras un proceso escribe en una tabla, otros accedan a la misma tabla sin necesidad de bloqueos. Cada usuario obtiene una visión consistente de lo último a lo que se le hizo commit.

Amplia variedad de tipos nativos:

PostgreSQL provee nativamente soporte para:

- Números de precisión arbitraria.
- Texto de largo ilimitado.

- Figuras geométricas (con una variedad de funciones asociadas)
- Direcciones IP (IPv4 e IPv6).
- Bloques de direcciones estilo CIDR.
- Direcciones MAC.
- Arrays.

Otras características:

- Claves ajenas también denominadas Llaves ajenas o Llaves Foráneas.
- Disparadores.
- Vistas.
- Integridad transaccional.
- Herencia de tablas.
- Tipos de datos y operaciones geométricas.[13]

Apache2

En la Red de la UCf se utiliza Apache2 como servidor HTTP. El servidor HTTP Apache (en partícular Apache2) es un Software (libre) de código abierto para plataformas Unix (BSD, GNU/Linux, etc.), Windows, Macintosh y otras, que implementa el protocolo HTTP/1.1

Apache presenta entre otras características mensajes de error altamente configurables, bases de datos de autenticación y negociado de contenido, pero fué criticado por la falta de una interfaz gráfica que ayude en su configuración.

Sus características más relevantes son:

- Modular
- Open source
- Multi-plataforma

- Extensible
- Popular (fácil conseguir ayuda/soporte)
- Gratuito

La arquitectura del servidor Apache es muy modular. El servidor consta de una sección *core* y diversos módulos que aportan mucha de la funcionalidad que podría considerarse básica para un servidor Web.[14]

Syslog-ng

Syslog permite a los administradores obtener información de registro en sus sistemas de manera uniforme para toda la red. Realizando la tarea de guardar, analizar y procesar las trazas fácilmente. Se utiliza Syslog-ng debido a que elimina las deficiencias de Syslog:

- Falta de métodos de autenticación.
- Mensajes en texto claro.
- Configuración poco flexible.
- Uso inconsistente de orígenes y prioridades.
- No registra el origen de la fuente.
- Uso de transferencia de mensajes no orientado a la conexión (UDP).

Syslog-ng mantiene la compatibilidad con Syslog, y garantiza la centralización de los mensajes informativos de los servicios telemáticos.

1.4.3 Herramientas para el desarrollo

Quanta Plus

Para el diseño de la aplicación Web se utilizará Quanta, por ser este un editor HTML libre, que permite la administración de proyectos, vistas previas dinámicas, completamiento de código, plantillas, barras de herramientas, brinda una estructura de árbol al documento.[15]

GIMP

Se utiliza Gimp (*GNU Image Manipulation Program*) por ser un programa de edición de imágenes, tanto dibujos como fotografías. Es un programa libre y

gratuito, englobado en el proyecto GNU y disponible bajo la licencia Licencia pública general de GNU. Se le puede considerar como la alternativa más firme para Photoshop, aunque posee una interfaz muy diferente.[16]

1.4.4 Fundamentación de la metodología utilizada

Lenguaje de Modelación Unificado (UML)

"El Lenguaje de Modelado Unificado (UML - Unified Modeling Language) permite modelar, construir y documentar los elementos que forman un producto de Software que responde a un enfoque orientado a objetos. Este lenguaje fue creado por un grupo de estudiosos de la Ingeniería de Software formado por: Ivar Jacobson, Grady Booch y James Rumbaugh en el año 1995. Desde entonces, se ha convertido en el estándar internacional para definir, organizar y visualizar los elementos que configuran la arquitectura de una aplicación orientada a objetos. Con este lenguaje, se pretende unificar las experiencias acumuladas sobre técnicas de modelado e incorporar las mejores prácticas actuales en un acercamiento estándar.

UML no es un lenguaje de programación sino un lenguaje de propósito general para el modelado orientado a objetos y también puede considerarse como un lenguaje de modelado visual que permite una abstracción del sistema y sus componentes." [17]

Entre sus objetivos fundamentales se encuentran:[18]

1. Ser tan simple como sea posible, pero manteniendo la capacidad de modelar toda la gama de sistemas que se necesita construir.

2. Necesita ser lo suficientemente expresivo para manejar todos los conceptos que se originan en un sistema moderno, tales como la concurrencia y distribución, así como también los mecanismos de la ingeniería de Software, como son el encapsulamiento y el uso de componentes.

3. Debe ser un lenguaje universal, como cualquier lenguaje de propósito general.

4. Imponer un estándar mundial.

Proceso Unificado de Desarrollo (RUP)

El Proceso Unificado de Desarrollo (RUP, por su denominación en inglés Racional Unified Process), fue creado por el mismo grupo de expertos que crearon UML, Ivar Jacobson, Grady Booch y James Rumbaugh en el año 1998. El objetivo que se perseguía con esta metodología era producir Software de alta calidad, es decir, que cumpla con los requerimientos de los usuarios dentro de una planificación y presupuesto establecidos. Esta metodología concibió desde sus inicios el uso de UML como lenguaje de modelado. Es un proceso dirigido por casos de uso, este avanza a través de una serie de flujos de trabajo, está centrado en la arquitectura y es iterativo e incremental. Además cubre el ciclo de vida de desarrollo de un proyecto y toma en cuenta las mejores prácticas a utilizar en el modelo de desarrollo de Software.

A continuación se muestran estas prácticas.[19]

- Desarrollo de Software en forma iterativa.
- Manejo de requerimientos.
- Utiliza arquitectura basada en componentes.
- Modelación del Software visualmente
- Verifica la calidad del Software.
- Controla los cambios.

1.5 Conclusiones.

El estudio realizado en este capítulo expone la importancia que tiene la utilización de un sistema capaz de centralizar, analizar y procesar las trazas de los servicios telemáticos en la Red UCf, demostrando las deficiencias asociadas a los métodos tradicionales utilizados para este fin. Además se analizan críticamente las soluciones utilizadas en la Red UCf, determinándose la necesidad de desarrollar e implementar un sistema que solucione los problemas actuales, exponiéndose

finalmente las metodologías, herramientas y tecnologías a emplear en el desarrollo de la solución propuesta.

Capítulo II. Modelo del dominio

En el presente capítulo se describe el análisis de la solución propuesta, utilizando UML como lenguaje de modelado y RUP como metodología, empleando el Modelo del Dominio para la captura de objetos; de esta forma se logra un mejor entendimiento del problema en cuestión.

2.1 Descripción del modelo del dominio

"El modelo del domino captura los tipos más importantes de objetos en el contexto del sistema. Los objetos del dominio representan las "cosas" que existen o los eventos que suceden en el entorno en el que trabaja el sistema. Muchos de los objetos del dominio o clases pueden obtenerse de una especificación de requisitos o mediante la entrevista con los expertos del dominio. El objetivo del modelado del dominio es comprender y describir las clases más importantes dentro del contexto del sistema." [20]

Para una mayor comprensión del contexto en que se desarrolla el sistema se definen los principales conceptos relacionados con el entorno del problema.

2.2 Definición de las entidades y los conceptos principales

En el modelo de dominio se definen las siguientes clases: Servidor, Servicio, Traza, Script, Datos Almacenados, Criterios de Configuración, Administrador, Resultado.

Servidor: computadora que maneja peticiones de datos, correo electrónico, servicios de redes y transferencia de archivos brindando así, servicios a otras computadoras (clientes).

Servicio: Utilidad brindada por los servidores (servicio Web, servicio Proxy, etc.). Sus eventos y acciones quedan almacenados en las trazas.

Traza: Son el primer registro de la actividad en los sistemas y redes. Contienen el registro de acciones y eventos que tienen lugar en un sistema. Son una parte primaria de la seguridad y pueden ser usados en la detección de ataques e

intrusiones, así como en el análisis de fallas de hardware y Software.

Script: Utilidad o programa ejecutado por un intérprete que permite la automatización de tareas. Es el encargado de analizar las trazas y emitir los datos almacenados.

Datos Almacenados: Es toda la información que el Script extrae de las trazas y almacena en una base de datos.

Criterios de Configuración: Conjunto de patrones definidos por el Administrador para analizar los datos.

Administrador: Es quien analiza los Datos Almacenados y según los Criterios de Configuración definidos con anterioridad por el mismo emite los resultados.

Resultado: muestra las acciones y el estado de los Servicios que tienen lugar en la red LAN.

2.3 Reglas del negocio a considerar

Un Servidor puede ofrecer uno o más Servicios, según sus características. Atendiendo a las especificidades de la red puede existir más de un Servidor brindando Servicios del mismo tipo.

- En una red LAN puede haber tantos Servicios como requieran los usuarios de la misma, los Servicios serán ofrecidos por los Servidores.
- Las trazas sólo serán emitidos por los Servicios y sólo serán analizados por los Scripts.
- Los Scripts sólo analizarán el contenido de las trazas generadas por los Servicios sin alterarlo; para de esta forma emitir los Datos Almacenados.
- Los Datos Almacenados no pueden ser alterados.
- El Administrador sólo analizará la información contenida en los Datos Almacenados.
- Los Criterios de Configuración son patrones definidos únicamente por el Administrador.

- El Resultado sólo será emitido por el Administrador.

2.4 Representación del modelo del dominio

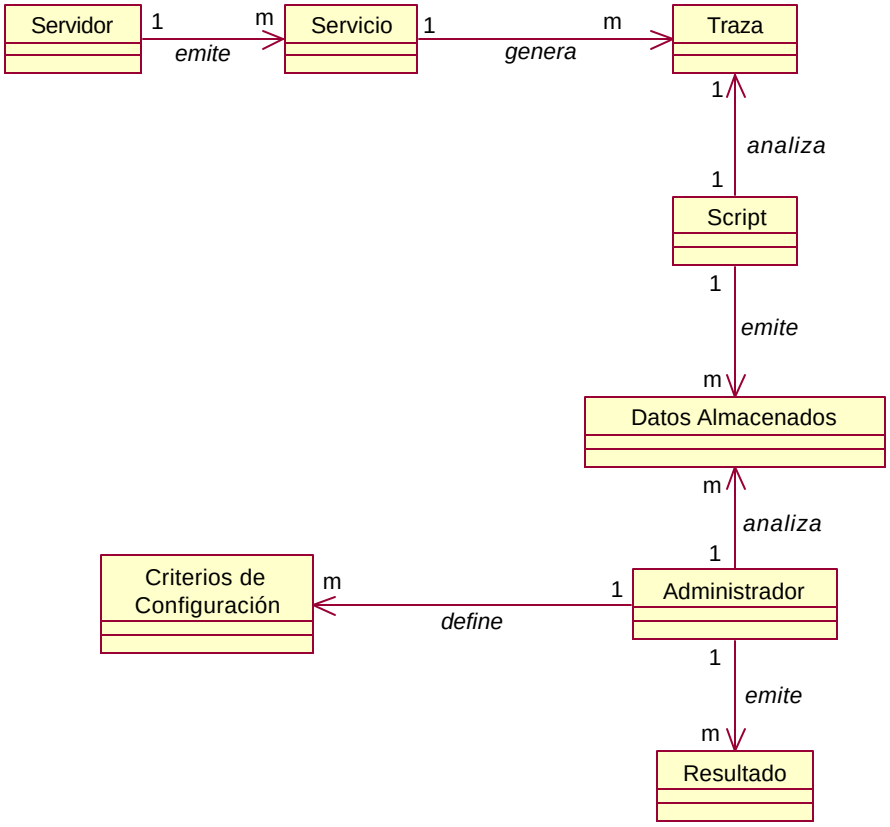


Figura 2. Diagrama de clases del modelo de objetos del dominio.

2.5 Conclusiones

Con el desarrollo de este capítulo se describió la solución propuesta, utilizando UML como lenguaje de modelado y RUP como metodología lográndose un mejor entendimiento de la solución propuesta al trabajo a desarrollar.

Capítulo III: Requisitos

En el presente capítulo se exponen los requerimientos funcionales y no funcionales del sistema, la descripción general del producto que se propone, así como el modelo de casos de uso del sistema y la descripción de cada uno de estos.

3.1 Descripción del sistema propuesto

En este epígrafe será detallado el sistema, describiendo su concepción, así como la instalación y configuración del mismo.

3.1.1 Concepción general del sistema

El sistema para la centralización, análisis y procesamiento de trazas de servicios telemáticos para la Red UCf, surge por la necesidad de contar con un sistema que a partir de la información contenida en las trazas de los servicios telemáticos de una red LAN permita conocer el estado y uso de la misma, mediante el uso de reportes y alarmas, y que ayude a la toma de acciones proactivas y reactivas encaminadas a garantizar los niveles de funcionalidad, operatividad y seguridad esperados.

El sistema estará compuesto por un módulo de captura de la información contenida en las trazas, que se encargará de filtrar las trazas, almacenar su contenido en una base de datos y generar alarmas ante la ocurrencia de alguno de los criterios definidos previamente por el administrador, y por un módulo de pos procesado que se encargará de la generación de reportes, a partir de los datos almacenados en la base de datos, y de la correlación de eventos, mostrando así el funcionamiento de cada servicio de la red y de la red en general. Con este sistema serán eliminadas las deficiencias existentes en los métodos utilizados actualmente.

3.1.2 Instalación y configuración del sistema.

3.1.2.1 Centralización de las trazas.

Esta etapa constituye el pilar del sistema, de ahí la importancia de una correcta

instalación y configuración de la misma. Para esto se debe:

1. Instalar servidor para centralizar las trazas de los servicios telemáticos.

En el caso de Ubuntu GNU/Linux distribución Hardy empleamos el siguiente comando:

apt-get install syslog-ng

Una vez instalado el Syslog-ng, se procede a la configuración del mismo.

2. Configuración del servidor de trazas centralizadas.

Para configurar el Syslog-ng se debe editar el fichero /etc/syslog-ng.conf, esta configuración cuenta con varios parámetros como son:

i) Options: Define las propiedades globales del fichero de configuración.

ii) Source: Define la fuente de donde provienen los mensajes, syslog-ng acepta mensajes a través de la red, utilizando los protocolos TCP o UPD.

El formato es:

source <sourcename> { sourcedriver params; sourcedriver params; ... };

donde *sourcename* es un identificador y *sourcedriver* es el método usado para recibir estos mensajes.

iii) Destination: Define donde serán escritos los mensajes.

El formato es:

destination <destname> { destdriver params; destdriver params; ... ; };

donde *destname* es un identificador y *destdriver* es el método usado para guardar los mensajes.

iv) filtering rules: reglas de filtrado (palabra clave: *filter*)

El formato es:

filter <filtername> { expression; };

Donde *filtername* es el nombre del filtro y *expression* es una expresión lógica

simple. Se puede usar "and", "or" y "not" para conectar las funciones incorporadas.

Las funciones pueden ser:

- **facility:** una lista separada por comas de nombres de facilities.
- **level:** una lista separada por comas o un rango separado por "." de nombres de priority.
- **program:** una expresión regular para buscar coincidencias con nombres de programas.
- **host:** una expresión regular para buscar coincidencias con nombres de Host.
- **match:** una expresión regular que busca coincidencias en el mensaje de Syslog en sí.

v) Log: Conecta los campos de Source y Destination. Une la fuente y el destino de las trazas. (Log = Source + Destination).

El formato es:

log { source S1; source S2; ... filter F1; filter F2; ... destination D1; destination D2; ... };

Donde Sx se refiere a uno de los orígenes de log declarados, Fx a uno de los filtros y Dx a uno de los destinos.

Ejemplo:

source net { udp(ip(0.0.0.0) port(514)); }; Escucha en todos las interfaces del servidor.

destination d_switch01 { file("/var/log/cisco/cisco3550.log"); }; Indica el camino del fichero destino para almacenar las trazas.

filter f_switch01 { host("192.168.0.1"); }; Filtra todos los paquetes que en el campo host tienen la dirección IP 192.168.0.1

log { source(net); filter(f_switch01); destination(d_switch01); }; Crea la traza que

se forma con los paquetes recibidos por cualquier interface y el puerto 514 del servidor de Syslog, que además en el campo Host del paquete Syslog tienen la IP 192.168.0.1 y ellos se almacenarán en el fichero cisco3550.log indicado en destination.

Una vez configurado el servidor syslog-ng, se procede a configurar los clientes.

3. Configuración de los clientes Syslog-ng.

Se denomina clientes de Syslog a los Hosts y dispositivos de red que utilizando este protocolo enviarán los mensajes al servidor central de trazas. En este caso fueron configurados los enrutadores y Switch Cisco y el servicio Web Apache2.

Ejemplo de equipos con tecnología Cisco:

Configurando el Switch Cisco Catalyst 3550

En el caso de Cisco tenemos que configurar tres opciones básicas que posteriormente pueden perfeccionarse en función de las necesidades. Estas opciones se configuran desde el modo ***configure terminal*** del Cisco IOS.

Las opciones son:

1. Router(config)# **logging on**
2. Router(config)# **logging 192.168.0.1**
3. Router(config)# **logging interface fastethernet 0/1**
4. Router(config)# **logging logging facility local5**

1. Habilita el equipo para que genere trazas de los eventos que se especifiquen.

2. Indica la dirección IP del servidor de Syslog al que se enviarán las trazas.

3. Indica la interfaz por donde saldrán las trazas generadas en el Switch.

4. Indica el tipo de prioridad (facility) según el formato de los paquetes Syslog, en este caso se habilita local5 lo que permite conocer los eventos importantes del sistema y las trazas generadas por las ACLs que se indiquen con la opción *log* o *log-input*

Ejemplo de configuración para Apache2

En el fichero httpd.conf:

CustomLog "| /usr/bin/logger –p local1.info" common

Y en el fichero /etc/syslog-ng.conf:

local1. @syslog-nghost*

Donde *@syslog-nghost* se refiere a la dirección IP del servidor de trazas.

3.1.2.2 Módulo de captura de la información almacenada en las trazas.

1. Implementación de los Scripts de captura

Los Scripts de captura fueron desarrollados utilizando Python como lenguaje de programación. Los mismos establecen una conexión con el servidor de base de datos y leen línea a línea el contenido de las trazas y van realizando operaciones como: el filtrado y almacenamiento de la información en la base de datos, la obtención de la dirección de red desde donde vienen y van los paquetes. Ver anexo 30.

2. Configuración del Cron

Al fichero /etc/crontab[2] se le adicionan los Scripts encargados de filtrar y almacenar el contenido de las trazas. Para ello es empleada la siguiente configuración:

Minuto Hora DiaDelMes Mes DiaDeLaSemana Intérprete Script

3.1.2.3 Módulo de pos procesado.

1. Copiar la herramienta Web al directorio /var/www del servidor Web y configurar el fichero globalvars.php.

2. Entrar al sistema a través de un navegador Web y añadir los servidores y

[2] Crontab: es una aplicación cuya finalidad es ejecutar un programa a determinada hora.

dispositivos de red, así como los servicios telemáticos cuyas trazas van a ser analizadas. Empleando para ello los criterios de configuración.

3. Agregar los criterios de configuración, como son: puertos, direcciones IP, patrones, URL, etc.

3.1.3 Requerimientos funcionales

Los requerimientos funcionales especifican acciones que el sistema debe ser capaz de realizar, sin tomar en consideración ningún tipo de restricción física. Por lo general se describen mejor a través del modelo de Casos de uso y los Casos de uso como tal. Por lo tanto los requerimientos funcionales especifican el comportamiento de entrada y salida del sistema y surgen de la razón fundamental de la existencia del producto.

Para la determinación de varios de los requerimientos funcionales que aquí se plantean, se realizaron entrevistas a los administradores de red de la Universidad de Cienfuegos y de otros centros, a fin de tener una visión, lo más cercana a la realidad posible con respecto a los principales problemas e insuficiencias existentes en el análisis y procesamiento de las trazas. Además de las sugerencias aportadas por los mismos en cuanto a posibles soluciones desde el punto de vista teórico, las que se tuvieron en cuenta a la hora automatizar las funciones.

Los requerimientos funcionales del sistema propuesto son los siguientes:

1. Autenticarse.
2. Cambiar contraseña del administrador.
3. Cambiar la cuenta del administrador.
4. Filtrar y almacenar el contenido de las trazas de los dispositivos de red.
5. Filtrar y almacenar el contenido de las trazas de los servicios Web Apache2.
6. Generar alarma por paquetes provenientes de puertos sospechosos.
7. Generar alarma por paquetes hacia puertos sospechosos.

8. Generar alarma por paquetes provenientes de IP sospechosas.
9. Generar alarma por paquetes hacia IP sospechosas.
10. Generar alarma por paquetes con patrones sospechosos en URL.
11. Generar alarma por paquetes con extensiones de ficheros peligrosas.
12. Generar alarma por paquetes con URL sospechosas.
13. Generar alarma por accesos desde IP sospechosas.
14. Agregar datos de servidor o dispositivo de red.
15. Modificar o datos de servidor dispositivo de red.
16. Eliminar datos de servidor o dispositivo de red.
17. Listar los datos de todos los servidores o dispositivo de red.
18. Agregar datos de servicio telemático.
19. Eliminar datos de servicio telemático.
20. Listar los datos de todos los servicios telemáticos.
21. Agregar datos de puerto a chequear.
22. Modificar datos de puerto a chequear.
23. Eliminar datos de puerto a chequear.
24. Listar los datos de todos los puertos.
25. Agregar datos de dirección IP a chequear.
26. Modificar datos de dirección IP a chequear.
27. Eliminar datos de dirección IP a chequear.
28. Listar los datos de todas las direcciones IP.
29. Agregar datos de URL a chequear.
30. Modificar datos de URL a chequear.
31. Eliminar datos de URL a chequear.

32. Listar los datos de todas las URL.

33. Agregar datos de patrón de URL.

34. Modificar datos de patrón de URL.

35. Eliminar datos de patrón de URL.

36. Listar datos de patrón de URL.

37. Agregar datos de extensión de fichero a chequear.

38. Modificar datos de extensión de fichero a chequear.

39. Eliminar datos de extensión de fichero a chequear.

40. Listar los datos de todas las extensiones de ficheros.

41. Agregar datos de VLAN a chequear.

42. Modificar datos de VLAN a chequear.

43. Eliminar datos de VLAN a chequear.

44. Listar los datos de todas las VLAN.

45. Agregar datos de umbral.

46. Modificar datos de umbral.

47. Listar alarmas.

48. Eliminar alarmas.

49. Determinar cantidad de paquetes permisibles para una misma IP y un mismo puerto.

50. Determinar cantidad de paquetes permisibles para diferentes IP y un mismo puerto.

51. Determinar cantidad de paquetes permisibles para una misma IP y diferentes puertos.

52. Determinar cantidad de paquetes permisibles desde una misma IP y un mismo puerto.

53. Determinar cantidad de paquetes permisibles desde diferentes IP y un mismo puerto para una misma IP y un mismo puerto.
54. Determinar cantidad de paquetes permisibles para o desde los puertos que se consideran peligrosos.
55. Determinar paquetes con direcciones IP de origen prohibidas o peligrosas.
56. Determinar paquetes con direcciones de IP de destino prohibidas o peligrosas.
57. Determinar paquetes con puertos de origen prohibido o peligroso.
58. Determinar paquetes con puertos de destino prohibido o peligroso.
59. Determinar cantidad de paquetes permisibles desde una misma IP.
60. Determinar cantidad de paquetes permisibles desde una misma IP para un mismo puerto.
61. Determinar cantidad de paquetes permisibles desde una misma IP para una misma IP.
62. Determinar cantidad de paquetes permisibles desde un mismo puerto.
63. Determinar cantidad de paquetes permisibles desde un mismo puerto, para una misma IP.
64. Determinar cantidad de paquetes permisibles desde un mismo puerto, para un mismo puerto.
65. Determinar cantidad de paquetes permisibles desde una misma IP, para un mismo puerto y una misma IP.
66. Determinar cantidad de paquetes permisibles desde una misma IP y un mismo puerto, para un mismo puerto y una misma IP.
67. Determinar cantidad de paquetes permisibles desde una misma IP y un mismo puerto, para una misma IP.
68. Determinar cantidad de paquetes permisibles desde una misma IP y un mismo puerto, para un mismo puerto.

69. Determinar paquetes HTTP con URL peligrosas.
70. Determinar paquetes HTTP con extensiones de ficheros peligrosas.
71. Determinar paquetes HTTP con patrones peligrosos.
72. Determinar paquetes HTTP provenientes de direcciones IP consideradas peligrosas.
73. Determinar paquetes HTTP con determinado protocolo.
74. Determinar paquetes HTTP con determinado método.
75. Determinar paquetes HTTP con determinado código de estado.
76. Determinar peticiones desde una IP específica al servidor Apache.
77. Determinar paquetes HTTP con determinado protocolo y método.
78. Determinar paquetes HTTP con determinado protocolo y código de estado.
79. Determinar paquetes HTTP con determinado protocolo que provienen de una Ip específica.
80. Determinar paquetes HTTP con determinado protocolo, método y código de estado.
81. Determinar paquetes HTTP con determinado protocolo, método, código de estado, provenientes desde una IP específica
82. Determinar paquetes HTTP con determinado método y código de estado.
83. Determinar paquetes HTTP con determinado método, código de estado, provenientes de una IP específica.
84. Determinar paquetes HTTP con determinado código de estado, provenientes de una IP específica.
85. Mostrar descripción de códigos de estado.
86. Determinar paquetes HTTP con determinado método, provenientes de una IP específica.
87. Determinar paquetes HTTP con determinado código de estado y protocolo,

provenientes de una IP específica.

88. Determinar paquetes HTTP con determinado método y protocolo, provenientes de una IP específica.
89. Determinar tráficos de la red por puertos.
90. Determinar tráficos de la red por VLAN.
91. Determinar tráficos de la red por protocolos.
92. Determinar tráficos de la red por estados.
93. Cerrar sesión.
94. Consultar ayuda del sistema.

3.1.4 Requerimientos no funcionales

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener (entiéndase por propiedades las características que hacen al producto atractivo, usable, rápido o confiable) como restricciones del entorno o de la implementación, rendimiento, dependencias de la plataforma, facilidad de mantenimiento, extensibilidad y fiabilidad.

Los requerimientos no funcionales del sistema propuesto son los siguientes:

Requerimientos de apariencia o interfaz externa

El sistema debe tener una interfaz sencilla, legible y simple de usar, el producto debe ser profesional, autoritario e interactivo para que el usuario se sienta confiado. Se deben brindar opciones de navegabilidad al usuario, como información sobre donde está y todo lo que puede hacer desde ese lugar, a través de señales de navegación. El contenido debe ser mostrado de manera comprensible para el usuario.

Requerimientos de Usabilidad

El sistema estará dirigido a administradores y a operadores de redes para facilitarles el estudio de las características del tráfico cursado por sus redes, ayudándoles a conocer el uso que se está dando a sus infraestructuras e identificando las fuentes de los problemas de rendimiento que aparecen. Por lo tanto el sistema está diseñado para ser utilizado por personas con conocimientos en el manejo de redes de computadora y el ambiente Web en sentido general.

Requerimientos de Rendimiento

El sistema deberá responder de forma eficaz y eficiente ante las solicitudes de los usuarios. Deberá ser preciso y exacto a la hora de realizar los cálculos. La velocidad de respuesta a las peticiones de reportes debe ser normal. La eficiencia de la aplicación estará determinada en gran medida por el aprovechamiento de los recursos que se disponen en el modelo Cliente/Servidor, y la velocidad de las consultas a la base de datos. Se realizará la validación de los datos y la

manipulación de eventos en el cliente y en el servidor aquellas que por cuestiones de seguridad, o de acceso a los datos lo requieran. Lográndose así un tiempo de respuesta más rápido, una mayor velocidad de procesamiento y un mayor aprovechamiento de los recursos.

Requerimientos de Soporte

Una vez terminado el desarrollo del sistema se realizaran pruebas en distintas redes Lan-IP para observar su comportamiento. La aplicación será documentada para garantizar el soporte de la herramienta. Se realizará mantenimiento al sistema a fin de aumentar las funcionalidades del mismo a través de versiones posteriores. Los servicios de instalación y mantenimiento del mismo serán responsabilidad del administrador del sistema en la entidad que sea utilizado.

Requerimientos de Portabilidad

El producto podrá ser usado bajo los sistemas operativos *GNU/Linux y Windows 95* o superior.

Requerimientos Políticos-culturales

La aplicación debe cumplir con las regulaciones políticas de la entidad que utilice el sistema.

Requerimientos de Confiabilidad

El sistema deberá ser robusto, no presentar fallos frecuentes y/o severos, y en caso de ocurrir deberá recuperarse rápidamente. Las operaciones a realizar deben ser transaccionales.

Requerimientos de Ayudas y Documentación en línea.

El sistema tendrá una ayuda donde se realizarán aclaraciones sobre las opciones del sistema, que garantizarán el buen desempeño de los usuarios a la hora de interactuar con el mismo.

Requerimientos de Software

La aplicación debe poderse ejecutar en entornos *Unix y GNU/Linux.* (Multiplataforma), para su ejecución del lado del servidor necesita preferentemente GNU/Linux como sistema operativo, PostgreSQL como sistema gestor de base de datos, Apache como servidor Web y Python para ejecutar los Scripts del módulo de captura de la información contenida en las trazas. Los servicios telemáticos deben estar configurados para garantizar la centralización de las trazas a través del Syslog-ng. Del lado del cliente cualquiera de los navegadores Web existentes en el mercado con la opción de Java Script activada.

Requerimientos de Hardware

Se requiere de una computadora como servidor de base de datos con PostgreSQL como gestor de base de datos, con una capacidad de memoria RAM no menor de 384 Mbyte, capacidad de almacenamiento proporcional a la cantidad de trazas que se generan y el tiempo de almacenamiento exigido (se recomienda como mínimo 80 GByte de almacenamiento), microprocesador Pentium III a 700 MHz como mínimo. Las terminales clientes sólo requerirán de una computadora conectada a la red, para poder ejecutar los navegadores Web, las mismas deben cumplir los requisitos mínimos de hardware del sistema operativo que se emplee.

Requerimientos de Seguridad

La información estará protegida contra accesos no autorizados utilizando mecanismos de autenticación y autorización que puedan garantizar el cumplimiento de esto: cuenta, contraseña; de manera que sólo el administrador pueda acceder a la información brindada por el sistema, garantizando así la confidencialidad. Además para reforzar la seguridad en el acceso se configurarán reglas en el cortafuego (Firewall) del servidor donde se instale el sistema, permitiéndole acceder al mismo solo a las estaciones de trabajo de los administradores de la red que lo utilizarán.

Se usarán mecanismos de encriptación de los datos que por cuestiones de seguridad no deben viajar al servidor en texto claro, para ello se empleará SSL en

la comunicación cliente servidor. Los datos sensibles se almacenarán de forma encriptada en la base de datos utilizando para ello MD5 como algoritmo de encriptación.

Se implementarán sesiones para evitar el acceso al sistema sin pasar por el mecanismo de validación o autenticación.

Se harán validaciones de la información tanto en el cliente como en el servidor.

No obstante, el administrador accederá de manera rápida y operativa al sistema sin que los requerimientos de seguridad se conviertan en un retardo para el.

3.2 Modelo de casos de uso del sistema

3.2.1 Actores del modelo del sistema

Un actor no es más que un conjunto de roles que los usuarios de Casos de Uso desempeñan cuando interaccionan con estos Casos de Uso. Los actores representan a terceros fuera del sistema que colaboran con el mismo. Una vez identificado los actores del sistema, queda identificado el entorno externo del sistema.

Nombre del Actor	Descripción
Administrador	Es quien utiliza el sistema para conocer parámetros como estabilidad, disponibilidad y seguridad de la red. Realiza tareas como la gestión de servidores y/o dispositivos de red, de los servicios telemáticos, del formato de salida de los reportes; además procesa el contenido de las trazas de los servicios, en busca de situaciones críticas. Define los criterios de configuración y genera los reportes.
Syslog-ng	Es el Software encargado de centralizar en un servidor, los mensajes generados por los diferentes servicios telemáticos y dispositivos de red.
Cron del Sistema Operativo	Es la utilidad del Sistema Operativo, encargada de ejecutar los Scripts que almacenan el contenido de las trazas, en la base

	de datos del sistema.

Tabla 1. Actores del Sistema.

3.2.2 Paquetes y sus relaciones

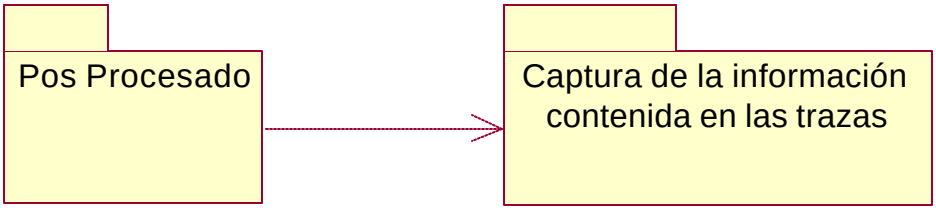


Figura 3. Paquetes y sus relaciones.

3.2.3 Casos de uso del sistema

"La forma en que los actores usan el sistema se representa con un caso de uso. Los casos de uso son "fragmentos" de funcionalidad que el sistema ofrece para aportar un resultado de valor para sus actores. De manera más precisa, un caso de uso especifica una secuencia de acciones que el sistema puede llevar a cabo interactuando con sus actores, incluyendo alternativas dentro de la secuencia." [20]

Casos de uso del paquete de captura de la información contenida en las trazas.

1. Filtrar y almacenar la información de las trazas.
2. Generar alarmas de dispositivos de red.
3. Generar alarmas de Apache2.

Casos de uso del paquete de pos procesado

1. Autenticarse.
2. Gestionar datos del administrador.
3. Gestionar criterios de configuración.
4. Gestionar los datos de servidores y/o dispositivos de red.
5. Gestionar los datos de los servicios telemáticos.
6. Procesar los datos almacenados de las trazas de los dispositivos de red.
7. Procesar los datos almacenados de las trazas del servicio Web Apache2.
8. Mostrar estado de la red por VLAN.

9. Mostrar estado de la red por puertos.
10. Mostrar estado de la red por protocolos.
11. Mostrar estado de la red por estados.
12. Gestionar alarmas.
13. Cerrar sesión.
14. Consultar ayuda del sistema.

3.2.4 Diagramas de casos de uso del sistema

"EL modelo de casos de uso permite que los desarrolladores de Software y los clientes lleguen a un acuerdo sobre los requisitos, es decir, sobre las condiciones y posibilidades que debe cumplir el sistema. El modelo de casos de uso sirve como acuerdo entre clientes y desarrolladores, y proporciona la entrada fundamental para el análisis, el diseño y las pruebas." [20]

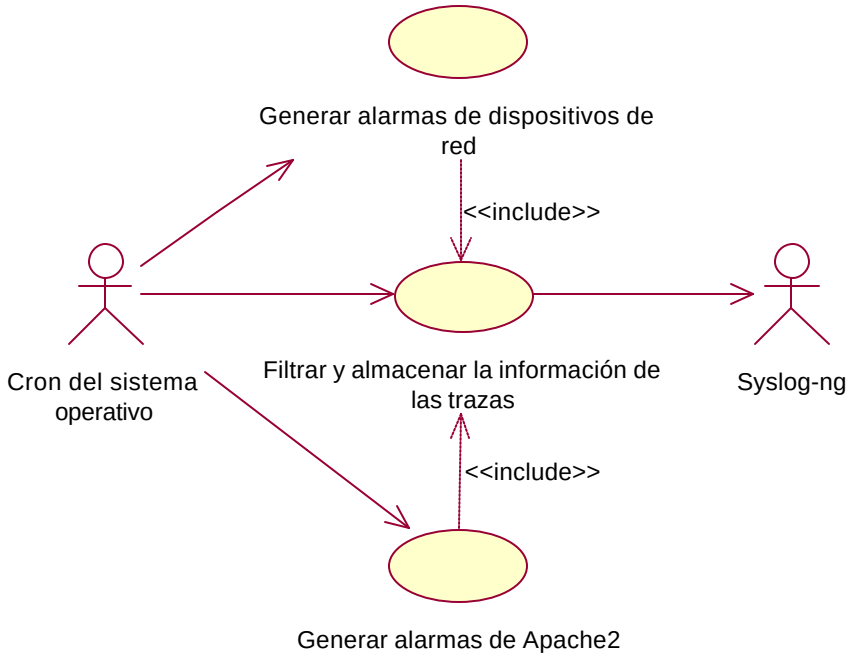


Figura 4. Diagrama de casos de uso del paquete de captura de la información contenida en las trazas.

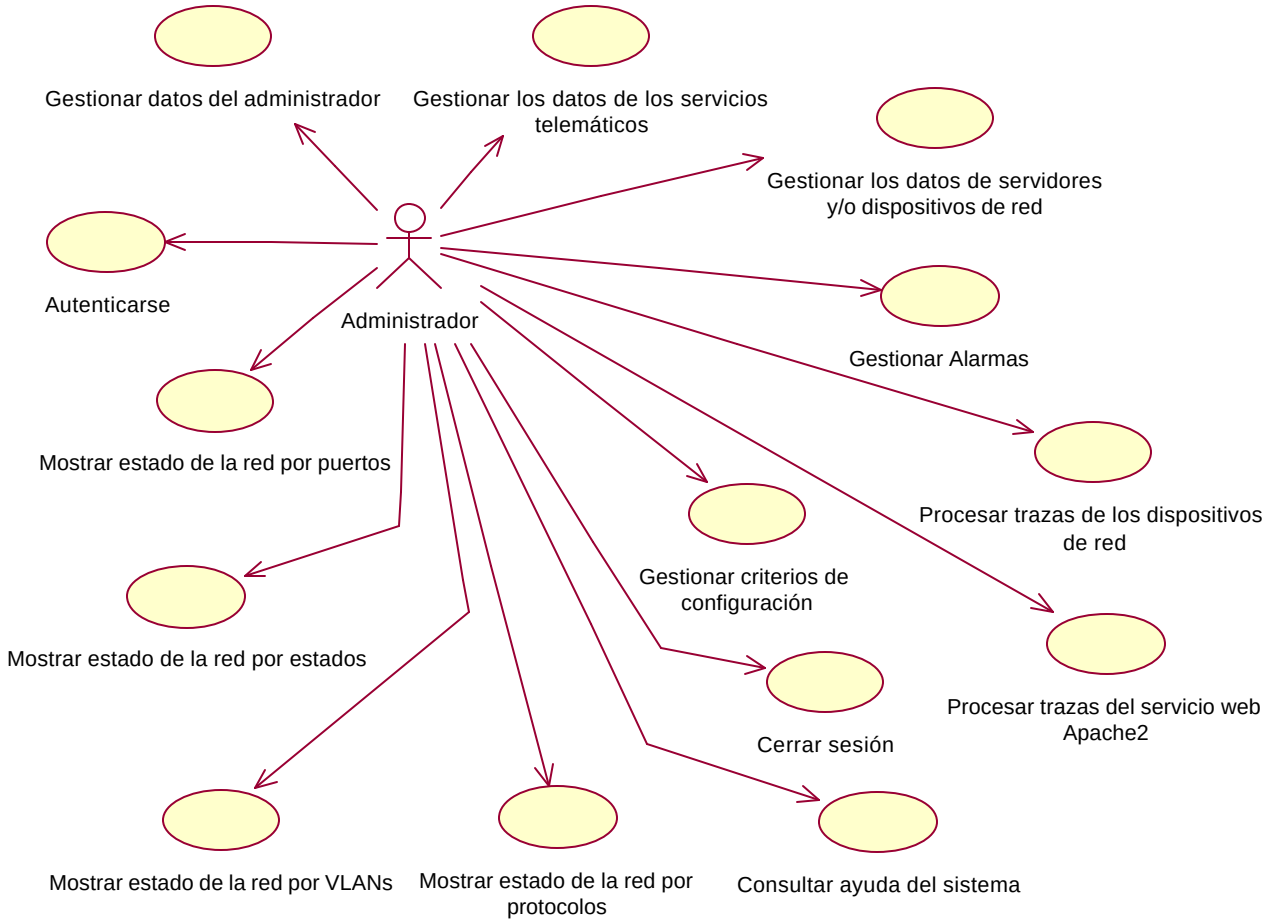


Figura 5. Diagrama de casos de uso del paquete de pos procesado.

3.2.5 Descripción de los casos de uso del sistema

3.2.5.1 Descripción de los casos de uso del módulo de pos procesado

Caso de uso:	Autenticarse.
Actores:	Administrador (inicia).
Propósito:	Acceder a las funcionalidades del sistema.
Resumen: El caso de uso se inicia cuando el administrador desea acceder a información y/o funcionalidades brindadas por el sistema. Para esto debe introducir su cuenta y contraseña, el sistema verifica los parámetros antes mencionados, y además la dirección IP desde donde se está accediendo, con los datos de los usuarios del sistema que están almacenados en la base de datos. Si los mismos son correctos, es permitido el acceso al sistema, en caso contrario, es negado el acceso; culminando así el caso de uso.	
Referencias:	R1.
Precondiciones:	Debe existir un administrador con cuenta y contraseña correcta; y una dirección IP permitida.
Poscondiciones:	El administrador puede acceder a la información y a las funcionalidades brindadas por el sistema.
Requisitos especiales:	-
Prototipo:	Anexo 1

Tabla 2. Descripción del caso de uso de sistema: Autenticarse.

Caso de uso:	Gestionar datos del administrador.
Actores:	Administrador (inicia).
Propósito:	El administrador pueda cambiar la contraseña de acceso al sistema, la cuenta o la dirección IP de acceso.
Resumen: El caso de uso se inicia cuando el administrador desea modificar los datos de acceso al sistema. Una vez dentro del sistema el administrador cambia su cuenta, su contraseña o la dirección IP desde la que accede; para cada una de estas tareas se verifica su contraseña de acceso al sistema. De esta forma culmina el caso de uso.	
Referencias:	R2, R3.
Precondiciones:	El administrador debe estar haber ingresado al sistema.
Poscondiciones:	Quedan modificados los parámetros de acceso al sistema.
Requisitos especiales:	-
Prototipo:	Anexo 2

Tabla 3. Descripción del caso de uso de sistema: Gestionar datos del administrador.

Caso de uso:	Gestionar criterios de configuración.
Actores:	Administrador (inicia).
Propósito:	Definir los criterios que le van a indicar al sistema como tiene que clasificar y procesar la información capturada de la red.
Resumen: El caso de uso se inicia cuando el administrador define, modifica o elimina los criterios por los cuales se procesan y clasifican los datos almacenados en la base de datos del sistema sobre el estado de los servicios telemáticos, dispositivos de red y/o servidores; culminando así el caso de uso. Los criterios a gestionar son: puertos, direcciones IP, URL, patrones de URL, extensiones de ficheros, VLAN, umbral para el tráfico de paquetes por puertos. Para cada criterio, se introduce al sistema el valor del mismo así como una breve descripción. El sistema realiza la búsqueda de estos patrones en los datos almacenados para generar reportes y alarmas en caso de encontrar alguna irregularidad.	
Referencias:	R21, R22, R23, R24, R25, R26, R27, R28, R29, R30, R31, R32, R33, R34, R35, R36, R37, R38, R39, R40, R41, R42, R43, R44, R45, R46.
Precondiciones:	Deben existir criterios mediante los cuales se pueda procesar el contenido de las trazas.
Poscondiciones:	Quedan definidos los criterios por los cuales se va a procesar el la información obtenida de las trazas.
Requisitos especiales:	-
Prototipo:	Anexo 3

Tabla 4. Descripción del caso de uso de sistema: Gestionar criterios de configuración.

Caso de uso:	Gestionar los datos de servidores y/o dispositivos de red.
Actores:	Administrador (inicia).
Propósito:	Gestionar la información de los servidores y/o dispositivos de red.
Resumen: El caso de uso se inicia cuando el administrador desea agregar, eliminar, o modificar los datos de algún servidor y/o dispositivo de red. Para el caso de agregar datos, son introducidos al sistema datos, como: el nombre DNS y direcciones IP de los dispositivos, luego de brindarle esta información al sistema, la misma queda almacenada. Para el caso de modificar los datos, se especifica el dispositivo cuyos datos van a ser modificados, y luego estos quedan modificados. En el caso de eliminar los datos de uno o más dispositivos se especifican cual o cuales son los dispositivos y sus datos quedan eliminados. El caso de uso culmina con la realización de alguna de estas tres acciones.	
Referencias:	R14, R15, R16, R17.
Precondiciones:	Deben existir físicamente los servidores y/o los dispositivos, cuyos datos se desea que queden almacenados.
Poscondiciones:	Quedan almacenados, modificados o eliminados los datos de los servidores y/o dispositivos de red en la base de datos del sistema.
Requisitos especiales:	-
Prototipo:	Anexo 4

Tabla 5. Descripción del caso de uso de sistema: Gestionar servidores y/o dispositivos de red.

Caso de uso:	Gestionar los datos de los servicios telemáticos.
Actores:	Administrador (inicia).
Propósito:	Gestionar los datos de los servicios telemáticos, que se quieren controlar.
Resumen: El caso de uso se inicia cuando el administrador desea agregar, eliminar, o modificar los datos de algún servicio telemático. Para el caso de agregar los datos, se introduce al sistema información, como: el nombre del servicio, cantidad de trazas, el servidor que estará a cargo del mismo. Luego de brindarle esta información al sistema, la misma queda almacenada. En el caso de eliminar los datos de uno o más servicios, se especifican cual o cuales son los servicios, y sus datos quedan eliminados. El caso de uso culmina con la realización de alguna de estas tres acciones.	
Referencias:	R18, R19, R20.
Precondiciones:	Deben estar instalados en la red los servidores y servicios cuyos datos se desea que queden almacenados; y deben existir Scripts capaces de filtrar el contenido de sus trazas y almacenarlas en la base de datos del sistema.
Poscondiciones:	Quedan almacenados, modificados o eliminados los datos de los servicios telemáticos, en la base de datos del sistema.
Requisitos especiales:	-
Prototipo:	Anexo 5

Tabla 6. Descripción del caso de uso de sistema: Gestionar los datos de los servicios telemáticos.

Caso de uso:	Procesar los datos almacenados de las trazas de los dispositivos de red.
Actores:	Administrador (inicia).
Propósito:	Procesar y analizar los datos almacenados provenientes de las trazas de los dispositivos de red.
Resumen: El caso de uso se inicia cuando el administrador analiza los datos almacenados provenientes de las trazas de los dispositivos de red, en busca de información como: la cantidad de paquetes por segundo permisibles, cantidad de paquetes permisibles para diferentes IP y un mismo puerto, cantidad de paquetes permisibles para una misma IP y diferentes puertos, o otros valores como: la cantidad de paquetes permisibles para o desde los puertos que se consideran peligrosos, paquetes con direcciones IP de origen prohibidas o peligrosas, paquetes con direcciones de IP de destino prohibidas o peligrosas, determinar tráficos de la red por puertos, tráficos de la red por direcciones IP. Para la realización de alguna de estas tareas, es seleccionada la opción correspondiente y se acota la búsqueda por fechas. El caso de uso culmina cuando es mostrado el reporte con los resultados obtenidos.	
Referencias:	R49, R50, R51, R52, R53, R54, R55, R56, R57, R58, R59, R60, R61, R62, R63, R64, R65, R66, R67, R68.
Precondiciones:	Deben estar almacenados los datos provenientes de las trazas de los dispositivos de red en el sistema, así como los criterios de configuración necesarios para el análisis de la información referente al servicio.
Poscondiciones: **Requisitos especiales:** **Prototipo:**	Son mostrados los valores antes mencionados. - Anexo 6

Tabla 7. Descripción del caso de uso de sistema: Procesar los datos almacenados de las trazas de los dispositivos de red.

Caso de uso:	Procesar los datos almacenados de las trazas del servicio Web Apache2.
Actores:	Administrador (inicia).
Propósito:	Procesar los datos almacenados provenientes de las trazas del servicio Web Apache2, en busca de información de interés.
Resumen: El caso de uso se inicia cuando el administrador analiza los datos almacenados provenientes de las trazas de los servicios Web Apache2, en busca de información como: paquetes con dominios de origen peligrosos, paquetes con URL peligrosas, paquetes con extensiones de ficheros peligrosas, paquetes con URL con patrones registrados como peligrosos, y otro tipo de información como: determinar paquetes HTTP con determinado protocolo, o con determinado método, o con determinado código de estado, o la combinación de cualquiera de estas opciones. Para la realización de alguna de estas tareas, es seleccionada la opción correspondiente y se acota la búsqueda por fechas. El caso de uso culmina cuando es mostrado el reporte con los resultados obtenidos.	
Referencias:	R69, R70, R71, R72, R73, R74, R75, R76, R77, R78, R79, R80, R81, R82, R83, R84, R85, R86, R87, R88.
Precondiciones:	Deben estar almacenados los datos provenientes de los servicios Web Apache2, en el sistema, así como los criterios de configuración necesarios para el análisis de la información referente al servicio.
Poscondiciones: **Requisitos especiales:** **Prototipo:**	Son mostrados los valores antes mencionados. - Anexo 7

Tabla 8. Descripción del caso de uso de sistema: Procesar los datos almacenados de las trazas del servicio Web Apache2.

Caso de uso:	Mostrar estado de la red por VLAN.
Actores:	Administrador (inicia).
Propósito:	Este caso de uso tiene como propósito mostrar como se comporta el tráfico de paquetes por VLAN en la red.
Resumen: El caso de uso se inicia cuando el administrador desea ver como se está comportando el tráfico de paquetes por VLAN en la red, o sea visualizar la cantidad de paquetes que van o vienen de las VLAN definidas en los criterios de configuración. Para esto el administrador puede seleccionar un servicio, o todos los servicios de tráfico en la red, luego decide si va a mostrar los paquetes que van hacia las VLAN, o los que vienen de las mismas. Después acota la búsqueda por fechas. Con la representación del estado del tráfico de paquetes en la red por VLAN, finaliza el caso de uso.	
Referencias:	R90
Precondiciones:	Deben estar almacenados los datos provenientes de las trazas de los dispositivos de red en el sistema, así como los criterios de configuración sobre las VLAN, para el análisis de la información referente a los servicios.
Poscondiciones:	Es mostrada la información que describe el comportamiento del tráfico de paquetes en la red por VLAN.
Requisitos especiales:	-
Prototipo:	Anexo 8

Tabla 9. Descripción del caso de uso de sistema: Mostrar estado de la red por VLAN.

Caso de uso:	Mostrar estado de la red por puertos.
Actores:	Administrador (inicia).
Propósito:	Este caso de uso tiene como propósito mostrar como se comporta el tráfico de paquetes por puertos en la red.
Resumen: El caso de uso se inicia cuando el administrador desea ver como se está comportando el tráfico de paquetes por puertos en la red, o sea visualizar la cantidad de paquetes que van o vienen de los puertos. Para esto el administrador puede seleccionar un servicio, o todos los servicios de tráfico en la red, luego decide si va a mostrar los paquetes que van hacia los puertos, o los que vienen de los mismos, luego decide si desea usar el umbral para el tráfico de puertos definido en los criterios de configuración. Después acota la búsqueda por fechas. Con la representación del estado del tráfico de paquetes en la red por puertos, finaliza el caso de uso.	
Referencias:	R89
Precondiciones:	Deben estar almacenados los datos provenientes de las trazas de los dispositivos de red en el sistema, así como los criterios de configuración sobre los umbrales de tráfico, para el análisis de la información referente a los servicios.
Poscondiciones:	Es mostrada la información que describe el comportamiento del tráfico de paquetes en la red por puertos.
Requisitos especiales:	-
Prototipo:	Anexo 9

Tabla 10. Descripción del caso de uso de sistema: Mostrar estado de la red por puertos.

Caso de uso:	Mostrar estado de la red por protocolos.
Actores:	Administrador (inicia).
Propósito:	Este caso de uso tiene como propósito mostrar como se comporta el tráfico de paquetes por los diferentes protocolos.
Resumen: El caso de uso se inicia cuando el administrador desea ver como se está comportando el tráfico de paquetes según los protocolos, o sea visualizar la cantidad de paquetes que van o vienen por cada uno de los protocolos. Para esto el administrador puede seleccionar un servicio, o todos los servicios de tráfico en la red, acota la búsqueda por fechas. Con la representación del estado del tráfico de paquetes en la red por protocolos, finaliza el caso de uso.	
Referencias:	R91
Precondiciones:	Deben estar almacenados los datos provenientes de las trazas de los dispositivos de red en el sistema.
Poscondiciones:	Es mostrada la información que describe el comportamiento del tráfico de paquetes en la red por protocolos.
Requisitos especiales:	-
Prototipo:	Anexo 10

Tabla 11. Descripción del caso de uso de sistema: Mostrar estado de la red por protocolos.

Caso de uso:	Mostrar estado de la red por estados.
Actores:	Administrador (inicia).
Propósito:	Este caso de uso tiene como propósito mostrar como se comporta el tráfico de paquetes según los paquetes que han sido permitidos y/o denegados.
Resumen: El caso de uso se inicia cuando el administrador desea ver la cantidad de paquetes que han sido permitidos y/o denegados. Para esto el administrador puede seleccionar un servicio, o todos los servicios de tráfico en la red, acota la búsqueda por fechas. Con la representación del estado del tráfico de paquetes en la red por estados, finaliza el caso de uso.	
Referencias:	R92
Precondiciones:	Deben estar almacenados los datos provenientes de las trazas de los dispositivos de red en el sistema.
Poscondiciones:	Es mostrada la información que describe el comportamiento del tráfico de paquetes en la red por estados.
Requisitos especiales:	-
Prototipo:	Anexo 11

Tabla 12. Descripción del caso de uso de sistema: Mostrar estado de la red por estados.

Caso de uso:	Gestionar Alarmas.
Actores:	Administrador (inicia).
Propósito:	Mostrar todas las situaciones alarmantes ocurridas hasta un momento dado y permitir su eliminación.
Resumen: El caso de uso se inicia cuando el administrador desea mostrar todas las alarmas que han ocurrido, aparece un listado con las mismas, brindando la posibilidad de eliminar una o varias de estas alarmas, culminando así el caso de uso.	
Referencias:	R47, R48
Precondiciones:	En la base de datos de alarmas, deben existir almacenadas las situaciones alarmantes encontradas.
Poscondiciones:	Quedan mostradas y/o almacenadas, todas las alarmas encontradas.
Requisitos especiales:	-
Prototipo:	Anexo 12

Tabla 13. Descripción del caso de uso de sistema: Gestionar alarmas.

Caso de uso:	Cerrar sesión
Actores:	Administrador (inicia).
Propósito:	Cerrar la sesión para salir del sistema.
Resumen: El caso de uso se inicia cuando el Administrador desea salir del sistema. Se brinda la opción de cerrar la sesión Administrador. Terminando así el caso de uso.	
Referencias:	R93.
Precondiciones:	El Administrador debe estar registrado en el sistema.
Poscondiciones: **Requisitos especiales:** **Prototipo:**	- - Anexo 13

Tabla 14. Descripción del caso de uso de sistema: Cerrar sesión.

Caso de uso:	Consultar ayuda del sistema
Actores:	Administrador (inicia).
Propósito:	Proporcionar información de utilidad al Administrador respecto al uso y funcionamiento del sistema.
Resumen: El caso de uso se inicia cuando el Administrador solicita la ayuda de algún tópico relacionado con el modo de uso del sistema. En dependencia con la necesidad del Administrador el sistema brindará la información que se ajuste adecuadamente a sus necesidades. El caso de uso finaliza una vez que el Administrador realice la consulta de la temática deseada de forma satisfactoria.	
Referencias:	R94.
Precondiciones:	-
Poscondiciones: **Requisitos especiales:** **Prototipo:**	- - Anexo 14

Tabla 15. Descripción del caso de uso de sistema: Consultar ayuda del sistema.

3.2.5.2 Descripción de los casos de uso del módulo de captura de la información contenida en las trazas.

Caso de uso:	Filtrar y Almacenar la información contenida en las trazas.
Actores:	Cron del Sistema Operativo (inicia).
Propósito:	Capturar la información contenida en las trazas de los diferentes servicios.
Resumen: El caso de uso se inicia, cuando el Cron del sistema operativo ejecuta, con la periodicidad establecida por el administrador, los Scripts encargados de filtrar la información contenida en las trazas de los servicios telemáticos, como son las del servicio Web Apache2, y las de los dispositivos de red. Para esto primero se abre la traza, se identifica a que servicio pertenece, y se lee línea a línea su contenido, separando el mismo según su tipo; (Ej: fecha, puerto de origen, Ip de origen, puerto de destino, IP de destino, protocolo, IP de acceso, código de estado, cantidad de paquetes, etc.). Esta información es almacenada en la base de datos del sistema. La fecha que aparece en la primera línea de la traza es escrita en el fichero de configuración, para ser usada por el caso de uso: Gestionar Alarmas. Culminando así el caso de uso.	
Referencias:	R4, R5
Precondiciones:	Deben estar ejecutándose los servicios telemáticos, y centralizadas sus trazas en el servidor de trazas.
Poscondiciones:	Queda filtrada y almacenada en la base de datos del sistema, la información que estaba contenida en las trazas de los diferentes servicios telemáticos.
Requisitos especiales:	-
Prototipo:	-

Tabla 16. Descripción del caso de uso de sistema: Filtrar y Almacenar la información contenida en las trazas.

Caso de uso:	Generar Alarmas de Apache2
Actores:	Cron del Sistema Operativo (inicia).
Propósito:	Buscar dentro de la información almacenada, de las trazas del servicio Web Apache2, la ocurrencia de algunos accesos considerados como peligrosos.
Resumen: El caso de uso se inicia, cuando es ejecutado el Script encargado de buscar situaciones alarmantes dentro de la información almacenada de las trazas del servicio Web Apache2. Se contabilizan las ocurrencias de parámetros que han sido definidos en los criterios de configuración; para este servicio específicamente son: URL, patrones de URL, extensiones de ficheros y accesos desde IP definidas previamente. Luego se inserta en la base de datos de las alarmas, las cantidades encontradas, la primera fecha de su aparición, así como una breve descripción. Culminando así el caso de uso.	
Referencias:	R10, R11, R12, R13.
Precondiciones:	Debe estar almacenada la última fecha en el fichero de configuración, y deben estar almacenados en la base de datos del sistema los criterios de configuración del servicio Web Apache2, como son: direcciones IP sospechosas, URL, patrones de URL, y extensiones de ficheros.
Poscondiciones:	Quedan almacenados los últimos accesos alarmantes encontrados en la información almacenada de Apache2.
Requisitos especiales:	-
Prototipo:	-

Tabla 17. Descripción del caso de uso de sistema: Generar alarmas de Apache2.

Caso de uso:	Generar alarmas de dispositivos de red
Actores:	Cron del Sistema Operativo (inicia).
Propósito:	Buscar dentro de la información almacenada, de las trazas de los dispositivos de red, la ocurrencia de algunos accesos considerados como peligrosos.
Resumen: El caso de uso se inicia, cuando es ejecutado el Script encargado de buscar situaciones alarmantes dentro de la información almacenada de las trazas de los dispositivos de red. Se contabilizan las ocurrencias de parámetros que han sido definidos en los criterios de configuración; para este servicio específicamente son: puertos y direcciones IP definidas previamente. Luego se inserta en la base de datos de las alarmas, las cantidades encontradas, la primera fecha de su aparición, así como una breve descripción. Culminando así el caso de uso.	
Referencias:	R6, R7, R8, R9.
Precondiciones:	Debe estar almacenada la última fecha en el fichero de configuración, y deben estar almacenados en la base de datos del sistema los criterios de configuración de los dispositivos de red, como son: direcciones IP sospechosas y puertos.
Poscondiciones:	Quedan almacenados los últimos accesos alarmantes encontrados en la información almacenada de los dispositivos de red.
Requisitos especiales:	-
Prototipo:	-

Tabla 18. Descripción del caso de uso de sistema: Generar alarmas de dispositivos de red.

3.3 Conclusiones

Con el desarrollo de este capítulo ha quedado descrito el sistema propuesto, así como los requerimientos funcionales y no funcionales del mismo. Además se realizó una descripción detallada del modelo de sistema a través de la definición de sus actores, paquetes y casos de usos, mostrándose para estos los diagramas de casos de uso.

Capítulo IV: Construcción de la solución propuesta

En el presente capítulo se realiza una descripción de la construcción de la solución propuesta, para ello se han utilizado los diagramas de clases Web del diseño y el modelo lógico y físico de la base de datos. Se describen los principios de diseño utilizados, mostrando ejemplos de cómo son presentados al usuario y la concepción general de la ayuda. Para describir los elementos fundamentales de la implementación se muestra el diagrama de implementación.

4.1 Diagrama de clases del diseño

El diagrama de clases del diseño describe gráficamente las especificaciones de las clases de Software, de las interfaces, así como sus relaciones en una aplicación. En el caso de las aplicaciones Web, el diagrama de clases representa las colaboraciones que ocurren entre las páginas, donde cada página lógica puede ser representada como una clase.

A continuación se presentan los diagramas de clases Web que describen los casos de uso del sistema.

Caso de Uso	Diagrama de Clases
Filtrar y almacenar la información de las trazas.	Anexo 15
Generar alarmas de Apache2	Anexo 15
Generar alarmas de dispositivos de red	Anexo 15

Tabla 19. Diagramas de clases Web. Paquete de captura de la información contenida en las trazas.

Caso de Uso	Diagrama de Clases
Autenticarse.	Anexo 16
Gestionar datos del administrador.	Anexo 17
Gestionar los datos de servidores y/o dispositivos de red.	Anexo 18

Gestionar los datos de los servicios telemáticos.	Anexo 19
Gestionar criterios de configuración.	Anexo 20
Mostrar estado de la red por VLAN	Anexo 21
Mostrar estado de la red por puertos	Anexo 22
Mostrar estado de la red por protocolos	Anexo 23
Mostrar estado de la red por estados	Anexo 24
Procesar trazas de los dispositivos de red	Anexo 25
Procesar trazas del servicio Web Apache2	Anexo 26
Gestionar alarmas	Anexo 27
Cerrar sesión	Anexo 28
Consultar ayuda del sistema	Anexo 29

Tabla 20. Diagramas de clases Web. Paquete de pos procesado.

4.2 Diseño de la base de datos

4.2.1 Modelo lógico de datos

El objetivo fundamental del modelo lógico de datos es brindar una vista de las entidades lógicas de datos y sus relaciones con independencia de la plataforma de base de datos a utilizar. Por lo general el modelo se construye en 3FN y es una representación gráfica del diagrama de clases persistentes.

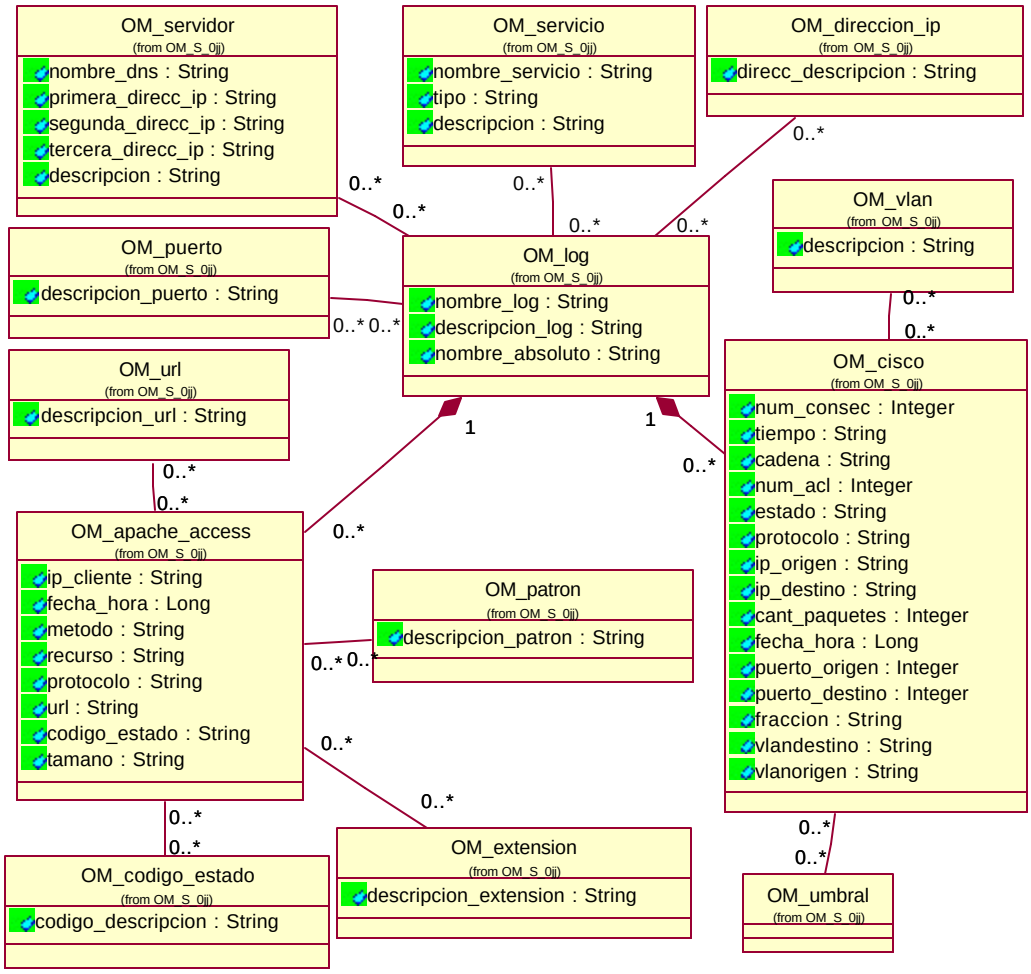


Figura 6. Diagrama de clases persistentes para la base de datos de servicios.

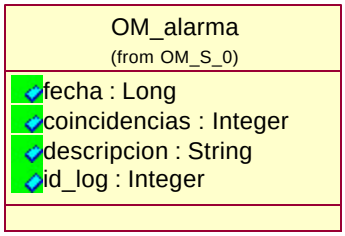


Figura 7. Diagrama de clases persistentes para la base de datos de alarmas.

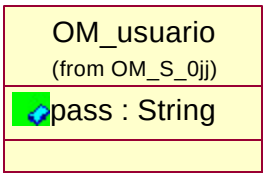


Figura 8. Diagrama de clases persistentes para la base de datos de usuarios.

4.2.2 Modelo físico de datos

Describe la representación lógica y física de los datos persistentes (la representación de las tablas que conforman la base de datos del sistema, obtenido a partir del diagrama de clases persistentes). Es el artefacto resultante de la actividad de diseño de base de datos.

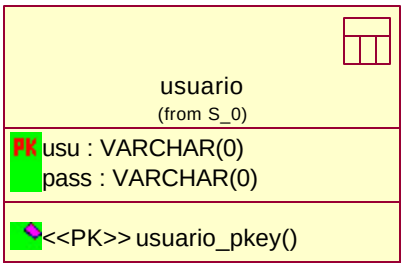


Figura 9. Diagrama del modelo físico de datos para la base de datos de los usuarios.

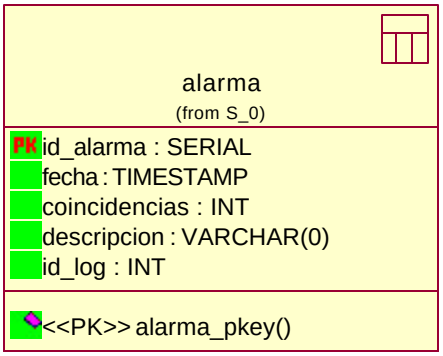


Figura 10. Diagrama del modelo físico de datos para la base de datos de las alarmas.

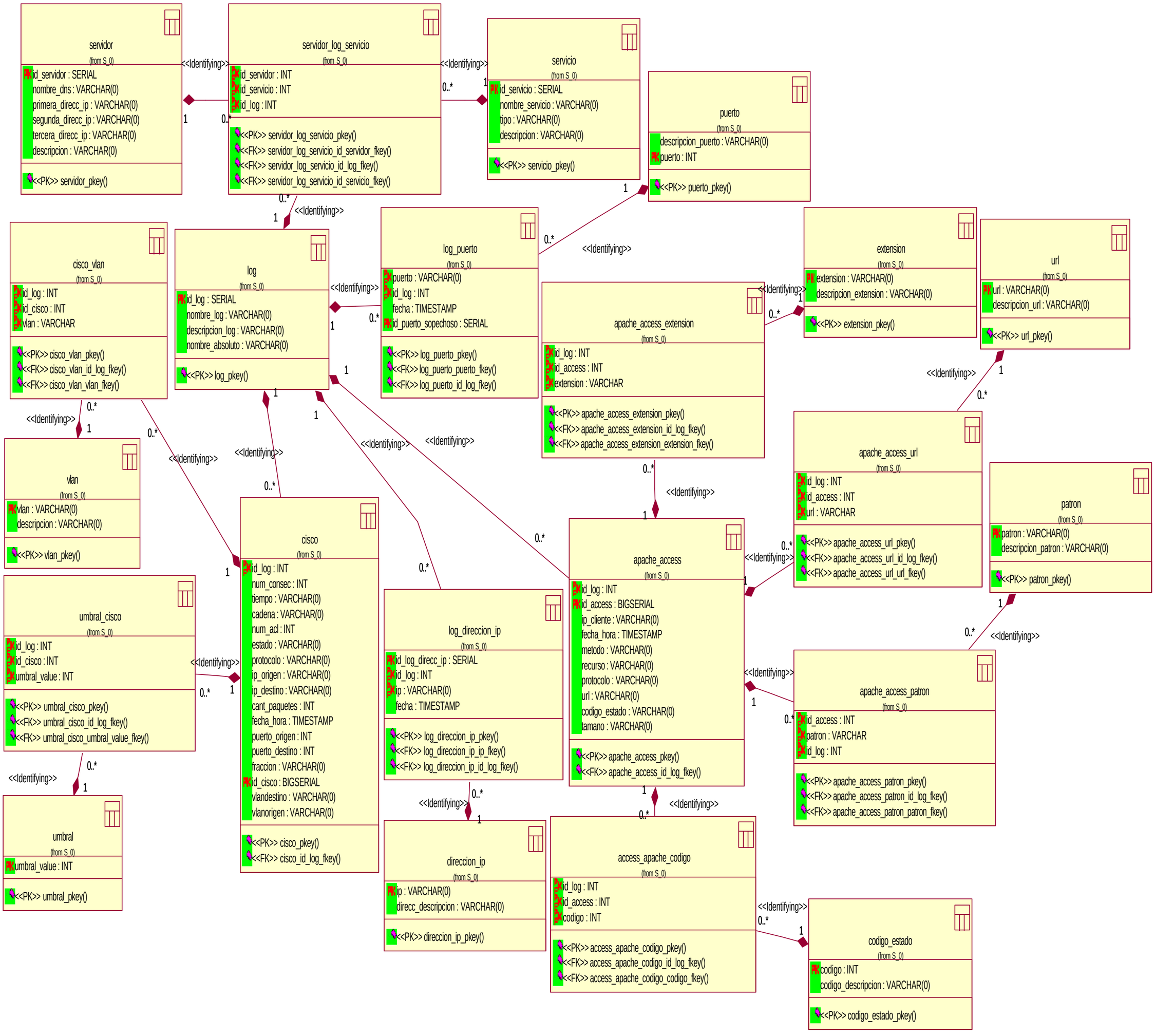


Figura 11. Diagrama del modelo físico de datos para la base de datos de servicios.

4.3 Diagrama de implementación

Con el diagrama de implementación se describen los componentes y la organización de acuerdo a los nodos, así como las dependencias entre ellos, por lo que en una sola vista se pueden apreciar los Diagramas de Componente y de Despliegue. Aunque se pueden construir por separado estos diagramas, en ellos no queda clara la relación directa que puede darse entre componentes ubicados en nodos diferentes.

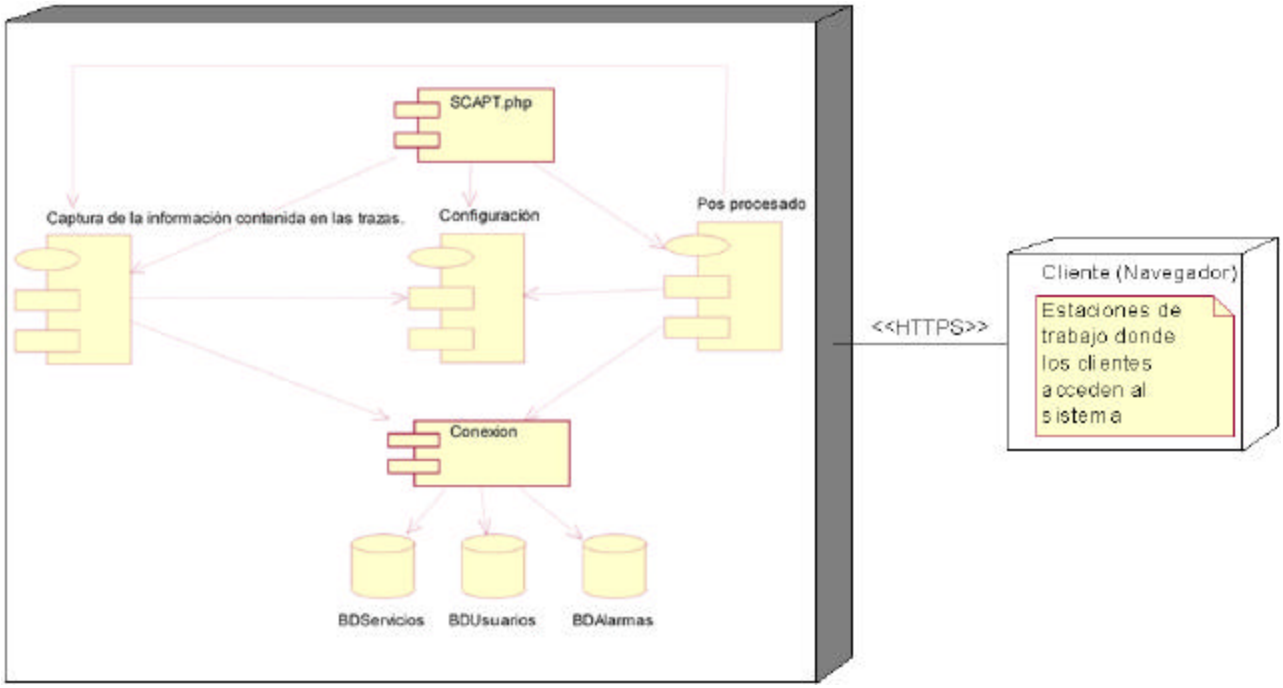


Figura 12. Diagrama de implementación.

4.4 Principios de diseño del sistema

A continuación se describen los principios de diseño seguidos para el desarrollo del sistema. Es importante mencionar que el mismo está orientado, a facilitar la rapidez y eficiencia en su utilización y en el caso de los módulos administrativos a garantizar su funcionalidad y comodidad en la gestión y administración descentralizada y remota.

4.4.1 Estándares en la interfaz de la aplicación

Para el caso del módulo de pos procesado, en el sistema se contrastan colores con tonalidad gris oscuro, gris claro y negro (negro y gris oscuro para la imagen empleada como Banner, las líneas y las letras utilizadas en todo el sistema, son de color azul oscuro, y para el fondo se utiliza el gris claro), además la letra debe ser legible y sencilla (se utiliza el tipo de letra Arial) para que unido al color utilizado contrasten con el fondo y sean refrescantes a la vista del usuario, se evita el uso excesivo de imágenes, animaciones y efectos visuales, en beneficio de una interfaz sencilla y rápida. Se mantuvo un diseño único en todo el sistema que proporciona un menú en la parte izquierda del sitio el cual está presente en todas las páginas del sistema, permitiendo que los administradores naveguen por el sistema manteniendo la posibilidad de acceso a todas las opciones en todo momento.

4.4.2 Tratamiento de errores

El sistema será programado de manera tal que se minimicen los errores aplicando técnicas de validación de datos y mediante la cuidadosa confección de la interfaz de entrada salida. Los mensajes de error que emite el sistema se mostrarán el un lenguaje claro y de fácil comprensión a personas sin conocimientos avanzados de computación. Además de esto se pedirá la confirmación por parte del usuario de operaciones que impliquen riesgos de pérdida de datos, tales como abrir o crear nuevas presentaciones y plantillas.

4.4.3 Concepción general de la ayuda

El sistema cuenta con una ayuda general la cual se encuentra disponible al administrador en cada momento, permitiéndole navegar por sus tópicos y consultar cualquier duda sobre el sistema, lo cual permite que el administrador pueda hacer un uso eficiente de la aplicación, la ayuda mantiene los mismos principios de diseño pero se muestra en una página independiente. Además existen ayudas específicas para diferentes opciones del sistema, a la que se puede acceder desde un enlace indicado por un signo de interrogación ubicado a

continuación de la opción que lo requiera y mostrándose la ayuda en una pequeña ventana independiente.

4.5 Conclusiones

En el presente capítulo se ha descrito la solución propuesta a través de la representación gráfica de los diagramas de clases del modelo de sistema, de los modelos lógico y físico de datos, y del diagrama de implementación. Fueron descritos además, los principios de diseño del sistema, específicamente, los temas de estándares de la interfaz, concepción del tratamiento de errores, sistema de ayuda.

Capítulo V: Análisis de prueba y estudio de factibilidad

En este capítulo se describen primeramente las pruebas y validaciones realizadas al sistema y posteriormente se realiza un análisis de la factibilidad para la aplicación desarrollada, utilizando como metodología COCOMO II.

5.1 Pruebas y validaciones realizadas

A fin de comprobar el funcionamiento del sistema desarrollado se realizaron pruebas prácticas simulando posibles situaciones que afectarían la estabilidad, operatividad y seguridad en la red telemática. Además se emplearon herramientas de Software tanto para las pruebas antes mencionadas como para validar algunos resultados.

5.1.2 Pruebas realizadas

5.1.2.1 Modificación de las trazas en los equipos que las generan

Esta prueba consistió en eliminar las evidencias locales de acceso desde una IP restringida al servidor Web, de forma que quedara registrado el acceso a otra hora y desde una IP autorizada.

En este caso se comprobó que como el sistema almacena una copia de las trazas que se le envían directamente según se generan, se pueden detectar con prontitud situaciones similares y en la medida de la existencia de alarmas previamente configuradas se puede conocer rápidamente la ocurrencia de una violación.

5.1.2.2 Simulación de propagación de virus conocidos

Esta prueba consistió en 2 acciones diferentes:

Primero. Desde un Host fue empleado un Shell Script[3] para enviar paquetes desde ese Host a diferentes Hosts pero al mismo puerto, empleando un puerto que está registrado como peligros por ser empleado en la propagación de virus, según los informes de los CERT y sitios antivirus. El Script se basó en una

[3] Shell Script: Se refiere a programas escritos para el shell de UNIX/LINUX.

modificación del uso del protocolo Telnet y se empleó TCPDUMP para validar el comportamiento esperado.

Segundo. Fue simulada la posible contaminación del servidor Web, para ello se ubicaron en el mismo archivos con patrones registrados en la base de datos como peligrosos, a partir de los informes de los CERT. Posteriormente se procede al acceso a través del protocolo HTTP a los archivos contaminados.

Como resultado en ambos casos se obtuvo una alarma informando la ocurrencia del evento.

5.1.2.3 Simulación de intentos de accesos no autorizados

Para esta prueba se intento acceder desde varios Hosts no autorizados y de forma consecutiva a un servidor empleando el protocolo SSH, lo que generó una alarma producto de la detección de un número de peticiones denegadas de forma consecutiva mayor o igual que 5 (incluye desde un solo Host) y de 10 de forma no consecutiva.

También se pudo comprobar el empleo de las opciones de inspección visual de las trazas, demostrando ser una herramienta efectiva que agiliza este trabajo.

5.1.2.4 Simulación de ataque de DDoS

Para esta prueba fueron simulados accesos desde varios Hosts de forma concurrente a un mismo servidor y un mismo servicio (Web puerto 80). Obteniéndose el resultado esperado al generarse una alarma al sobre pasar 50 peticiones en total.

5.1.2.5 Simulación de intentos de Spoofing

Se utilizó una herramienta de construcción de paquetes, incluida en la distribución de seguridad (GNU/Linux) Backtrack, con la cual se modificaron los datagramas IP incorporándole en la IP de origen una dirección de la VLAN administrativa procediéndose a enviarlos desde diferentes VLAN. Empleándose la herramienta para determinar esta anomalía mediante la comprobación de la alarma generada por el sistema.

5.1.2.6 Simulación de monopolización del canal o el Backbone por un Host

Empleando los reporte de tráfico y gráficas de VLAN se pudo definir primeramente que VLAN tiene más tráfico y después con los reportes de tráfico de la herramienta se pudo definir cual o cuales Hosts de esa VLAN estaban generando o recibiendo el tráfico.

5.1.2.7 Simulación de empleo de herramientas de Scanner de vulnerabilidades desde un Host no autorizado

Para esta prueba fueron verificadas a través de los reportes de tráfico, las peticiones desde un mismo Host a otro Host y diferentes puertos, fundamentalmente los puertos que se encuentran en el rango de 1 a 1023. Los resultados obtenidos se lograron mediante el empleo de herramientas como Retina y Nmap.

5.1.2.8 Simulación de violación de privilegios en el servicio Web

Para esta prueba se simuló el acceso al servidor Web desde un Host con dirección IP registrada como peligrosa, lo cual provocó una alarma informando la ocurrencia del evento. También fueron chequeados los mensajes con códigos de estado 401 el cual indica un acceso no autorizado al servidor.

5.1.3 Herramientas de Software utilizadas para validar los resultados

Para validar los resultados obtenidos se utilizaron varias herramientas, entre las que se destacan:

Retina es una herramienta que ofrece la posibilidad de identificar eficazmente las vulnerabilidades que dejan a las redes expuestas a ataques maliciosos, revisando los servicios existentes en una máquina y proponiendo soluciones.

Nmap es un programa de código abierto que sirve para efectuar rastreo de puertos. Se usa para evaluar la seguridad de sistemas informáticos, así como para descubrir servicios o servidores en una red informática.

Tcpdump es una herramienta en línea de comandos, del tipo Sniffer, cuya utilidad principal es analizar el tráfico que circula por la red. Permite al usuario capturar y

mostrar a tiempo real los paquetes transmitidos y recibidos en la red a la cual el ordenador está conectado.

Backtrack es una distribución GNU/Linux en formato LiveCD pensada y diseñada para la auditoría de seguridad y relacionada con la seguridad informática en general. Incluye una larga lista de herramientas de seguridad listas para usar, entre las que destacan numerosos scanners de puertos y vulnerabilidades, archivos de exploits, sniffers, herramientas de análisis forense y herramientas para la auditoría Wireless.

Telnet y **SSH** son protocolos utilizados para acceder a máquinas remotas a través de una red. Pudiendo ser empleados para comprobar algunos servicios.

5.2 Estudio de factibilidad

Para llevar a cabo un buen proyecto de desarrollo de Software, debemos comprender el ámbito del trabajo a realizar, los recursos requeridos, las tareas a ejecutar, las referencias a tener en cuenta, el esfuerzo (COSTE) a emplear y la agenda a seguir.

Para determinar si la construcción de un Software es factible o no se han introducido en la Ingeniería del Software una serie de técnicas utilizadas dentro de las tareas de planificación, que ayudan a planificar y controlar el esfuerzo y el tiempo necesario de desarrollo:

- Técnicas de estimación del esfuerzo (costo) de desarrollo. Dentro de las cuales se sitúa el Modelo Constructivo de Costes (Constructive Cost Model (COCOMO)).

- Técnicas de planificación y seguimiento de proyectos.

La estimación es una de las primeras actividades de la gestión de proyectos informáticos. Se le define como la predicción del personal, del esfuerzo, de los costos y del tiempo que se requerirán para realizar todas las actividades y construir todos los productos asociados con el proyecto. Su objetivo es conocer en etapas tempranas y de manera aproximada, el costo, la duración y los recursos necesarios para el desarrollo de proyectos de Software. Se trata de una apreciación del futuro y la exactitud con la que ésta se realice, depende la mayoría de las veces de una buena herramienta de estimación, de la experiencia del estimador y del acceso a una base de información histórica de los proyectos.

5.2.1 Planificación por casos de usos

La estimación mediante el análisis de Puntos de Casos de Uso es un método propuesto originalmente por Gustav Karner, y posteriormente refinado por muchos otros autores. Se trata de un método de estimación del tiempo de desarrollo de un proyecto mediante la asignación de "pesos" a un cierto número de factores que lo afectan, para finalmente, contabilizar el tiempo total estimado para el proyecto a partir de esos factores.

5.2.1.1 Cálculo de Puntos de Casos de Uso sin Ajustar

UUCP = UAW + UUCW

Donde:

UUCP: Puntos de Casos de Uso sin ajustar.

UAW: Factor de Peso de los Actores sin ajustar.

UUCW: Factor de Peso de los Casos de Uso sin ajustar.

Cálculo de UAW:

Actor	Factor de peso
Administrador	3
Syslog-ng	2
Cron	2

Tabla 21. Asignación de peso a los actores.

UAW = (Cantidad de actores) * Peso

UAW = 2*2+1*3

UAW = 7

Cálculo de UUCW:

Caso de uso	Tipo
Filtrar y almacenar la información de las trazas	simple
Generar alarmas de Apache2	medio
Generar alarmas de dispositivos de red	medio
Autenticarse	simple
Gestionar datos del administrador	simple
Gestionar criterios de configuración	complejo
Gestionar los datos de servidores y/o dispositivos de red	medio
Gestionar los datos de los servicios telemáticos	simple
Procesar los datos almacenados de las trazas de los dispositivos de red	complejo
Procesar los datos almacenados de las trazas del servicio Web Apache2	complejo
Mostrar estado de la red por VLAN	simple
Mostrar estado de la red por puertos	simple
Mostrar estado de la red por protocolos	simple
Mostrar estado de la red por estados	simple
Gestionar alarmas	simple
Cerrar sesión	simple
Consultar ayuda del sistema	simple

Tabla 22. Factor de peso de los casos de uso sin ajustar.

UUCW = 11*5 + 3*10 + 3*15

UUCW = 130

Los puntos de casos de uso sin ajustar resultan:

UUCP = UAW + UUCW

UUCP = 7 + 130

UUCP = 137

5.2.1.2 Cálculo de los Puntos de Casos de Uso Ajustados

UCP = UUCP x TCF x EF

Donde:

UCP: Puntos de Casos de Uso ajustados.

TCF: Factor de complejidad técnica.

EF: Factor de ambiente.

Cálculo de TCF:

Factor	Descripción	Peso	Val. Asig.	Porque	Total
T1	Sistema Distribuido	2	5	Cuenta con un módulo que es una aplicación Web, y el otro módulo está conformado por Scripts que son ejecutados por el Cron, además cuenta con un servidor de trazas.	10
T2	Obj. de performance o tiempo de respuesta	1	4	La velocidad de respuesta es directamente proporcional a las entradas provistas por el administrador.	4
T3	Eficiencia del usuario final	1	5	No existen restricciones de eficiencia para el administrador.	5
T4	Procesamiento interno complejo	1	5	Se realizan operaciones complejas en ambos módulos.	5
T5	El código debe ser reutilizable	1	5	Cada módulo es implementado pensando en la incorporación de nuevos servicios, lo que fomentó que el código escrito fuera cien	5

				por ciento reutilizable.	
T6	Facilidad de intalación	0.5	2	Al existir dos módulos, diferentes en cuanto al uso de tecnologías de programación, y al necesitar de otros sistemas como el Syslog-ng, con su respectiva configuración, se hace un poco difícil la instalación.	1
T7	Facilidad de uso	0.5	4	El sistema una vez instalado es fácil de usar.	2
T8	Portabilidad	2	4	El sistema es muy portable.	8
T9	Facilidad de cambio	1	4	El sistema ha sido concebido pensando en la incorporación futura de nuevos servicios, y en cambios en las operaciones sobre los ya existentes.	4
T10	Concurrencia	1	4	Buena concurrencia	4
T11	Incluye objetivos especiales de seguridad	1	4	Debido a la importancia que poseen los datos que maneja el sistema, se hacen necesarios objetivos especiales de seguridad.	4
T12	Provee acceso directo a terceras partes	1	3	Cuenta con accesos a partes más importantes.	3
T13	Se requieren facilidades especiales de entrenamiento a usuarios	1	3	Para administradores y operadores de red, no se requieren muchos entrenamientos para el uso del sistema.	3

Tabla 23. Cálculo del factor de complejidad técnica.

$TCF = 0.6 + 0.01 * S(Peso_i * Valor\ asignado_i)$

$TCF = 0.6 + 0.01 * (10 + 4 + 5 + 5 + 5 + 1 + 2 + 8 + 4 + 4 + 4 + 3 + 3)$

$TCF = 0.6 + 0.01 * 58$

$TCF = 0.6 + 0.58$

$TCF = 1.18$

Cálculo de EF

Factor	Descripción	Peso	Val. Asg.	Porque	Total
E1	Familiaridad con el modelo de proyecto utilizado	1.5	4	Existe familiarización con este tipo de proyectos.	6
E2	Experiencia con la aplicación	0.5	4	Se han realizados aplicaciones de este tipo.	2
E3	Experiencia en orientación a objetos	1	4	Los paradigmas de la programación orientada a objetos han sido aplicados en los sistemas que han sido implementados anteriormente.	4
E4	Capacidad del analista líder	0.5	4	Experiencia media	2
E5	Motivación	1	5	Alta motivación	5
E6	Estabilidad de los requerimientos	2	4	Abierto a cambios y mejoras	8
E7	Personal part-time	-1	0	El proyecto lo realiza una sola persona.	0
E8	Dificultad del lenguaje de programación	-1	2	Se utilizan varios lenguajes de programación, pero el que más se usa es PHP.	-2

Tabla 24. Cálculo del factor ambiente.

$EF = 1.4 - 0.03 * S(Peso_i * Valor\ asignado_i)$

EF = 1.4 - 0.03 * (6+2+4+2+5+8+0-2)

EF = 1.4 - 0.03 * 25

EF = 0.65

Los puntos de casos de uso ajustados resultan:

UCP= UUCP * TCF * EF

UCP=137 * 1.18 * 0.65

UCP=105.08

5.2.2 Estimación del esfuerzo

CF: Factor de Conversión

CF= 20 Horas-Hombre

El esfuerzo en horas /hombre está dado por:

E= UCP * CF

E= 105.08 * 20 = 2101.6 horas-hombre

Duración:

Trabajando 25 días al mes y 12 horas diarias como promedio, se tiene que:

Duración (días)= Total de Horas /Hombre entre 12 horas al día = 5254/12 =437.8 días

Duración (meses)=Total de días / 30 días por mes = 437.8 / 25 = 17.512 ˜ 18 meses

Actividad	Porcentaje	Valor
Análisis	10.00 %	525.4
Diseño	20.00 %	1050.8
Programación	40.00 %	2101.6
Pruebas	15.00 %	788.1
Sobrecarga (otras actividades)	15.00 %	788.1
Total de horas	100 %	5254

Tabla 25. Criterios de distribución de esfuerzos.

5.2.3 Cálculo de costos

Tomando como salario promedio mensual $275.00

Costo = 18 meses * $275.00 = $4950.00

5.2.4 Beneficios tangibles e intangibles

El desarrollo de este sistema se logra obtener datos que permiten tomar acciones proactivas encaminadas a garantizar la estabilidad, disponibilidad y seguridad de la red, garantizando así el correcto funcionamiento de la misma. Además se logra que las tareas de supervisión y control de la información contenida en las trazas

de los servicios telemáticos resulten tediosos, y sean realizados con la periodicidad necesaria. De esta manera se logra que los esfuerzos empleados en el desarrollo del sistema estén encaminados al cumplimiento de los objetivos planteados.

5.2.5 Análisis de costos y beneficios

Anteriormente las tareas de supervisión y control de la información contenida en las trazas de los diferentes servicios telemáticos de la Red UCf, se realizaban de dos maneras: mediante el uso de herramientas que por ser muy específicas para un tipo de servicio, dejaban de ser una buena solución; y manualmente lo que provocaba un consumo elevado de tiempo.

Con la explotación del sistema antes descrito se logran mejorar estas tareas, con la ayuda de reportes y generación de alarmas.

Con el análisis del costo del sistema, los beneficios que brinda el mismo y dando cumplimiento los objetivos planteados, se concluye que la aplicación es factible.

5.3 Conclusiones

A lo largo de este capítulo se describen los resultados obtenidos al someter el sistema a pruebas, validando los mismos con herramientas a fines, obteniéndose en cada prueba los resultados deseados. Además se realizó el estudio de factibilidad económica del desarrollo del sistema, llegando a la conclusión de que con un hombre que desarrolle todas las etapas requeridas como son análisis, diseño e implementación, trabajando 12 horas diarias, 25 días al mes durante 18 meses, es suficiente. El desarrollo del sistema tendrá un costo de $4950.00.

Conclusiones

Como resultado de la presente investigación se puede concluir que con la utilización del sistema desarrollado se logran mejorar las tareas de supervisión y control sobre los servicios telemáticos de la Red UCf. Contribuyendo de esta manera a elevar los niveles de seguridad existentes a partir de poder contar con un conocimiento más detallado de los eventos que ocurren en la red, y poder dar una respuesta más oportuna.

De igual forma podemos afirmar lo siguiente:

- Se desarrolló una herramienta de Software para el análisis y procesamiento de las trazas centralizadas de los principales servicios telemáticos de la Red UCf, empleándose para ello Software Libre.
- Se implementó un sistema para la centralización y almacenamiento de las trazas de los principales servicios telemáticos de la Red UCf.
- Fueron seleccionadas las tecnologías de procesamiento de trazas y los modelos de diseños de almacenamiento de trazas centralizadas, que permitieron alcanzar los niveles de seguridad esperados.
- Fueron analizados los elementos del sistema a automatizar.
- Se investigaron sistemas automatizados existentes asociados al problema.
- Se validó la implementación final, mediante el empleo de herramientas de Software y el análisis del comportamiento del sistema.

Recomendaciones

- Realizar estudio que permita la incorporación de las trazas de otros servicios telemáticos al sistema, con el fin de lograr la integración del análisis de trazas de todos los servicios de la Red UCf.
- Incorporar un módulo de gestión de alarmas a fin de aportarle facilidad y flexibilidad en la generación y configuración de las mismas.
- Continuar trabajando para culminar los módulos de correlación de eventos, análisis estadístico y procesamiento en línea así como incorporar técnicas de inteligencia artificial que permitan mejorar los análisis proactivos, en busca de lograr una estrategia de red autodefensiva. Acciones en las cuales se trabajan actualmente.

Referencias Bibliográficas

[1]“Glosario Informática e Internet,” Ene. 2008; http://glosario.panamacom.com/letras.html.

[2] N. Altalef, “Syslog Centralizado con detección de eventos.,” Ene. 2008; http://www.redklee.com.ar.

[3]“What’s new in GFI EventsManager 8?. ,” Oct. 2008; http://kbase.gfi.com/showarticle.asp?id=KBID003285.

[4]“Financial Tech Magazine.,” Nov. 2006; http://www.financialtech-mag.com/000_estructura/index.php?id=24&idb=101&ntt=6729&sec=8&vn=1.

[5]“Herramientas de análisis para ficheros de registros de cortafuegos. ,” Mar. 2008; http:// www.linux-magazine.es/issue/03/Analyzer.pdf .

[6]“OSSEC,” May. 2008; http://www.ossec.net/main/about/.

[7]“Weird Solutions,” Oct. 2008; http://corporate.weird-solutions.com/products/syslog-turbo.

[8]“Python,” Ene. 2009, pág. http://www.python.org.

[9]“Manual de HTML,” Ene. 2009; http://www-app.etsit.upm.es/~alvaro/manual/manual.html.

[10]“Introducción a CSS,” Dic. 2008; http://www.maestrosdelweb.com/editorial/introcss/.

[11]“PHP,” Ene. 2009; http://www.maestrosdelweb.com/editorial/desarrollo-web/php/.

[12]“Introducción a JavaScript,” Ene. 2009; http://www.maestrosdelweb.com/editorial/%C2%BFque-es-javascript/.

[13]“PostgreSQL,” Ene. 2009; http://www.postgresql.org.

[14]“Apache2,” Ene. 2009; http://www.apache2.com.

[15]“QuantaPlus,” Ene. 2009; http://docs.kde.org/kde3/es/kdewebdev/quanta/introduction-3-2.html.

[16]“Gimp,” Ene. 2009; http://www.maestrosdelweb.com/editorial/%C2%BFque-es-javascript/.

[17] P. Letelier Torres, “Desarrollo de Software Orientado a Objeto usando UML,” May. 2008; http://www.creangel.com/uml/intro.php.

[18] X. Ferrá Grau, “Desarrollo orientado a objetos con UML,” May. 2008; http://www.clikear.com/manuales/uml/introduccion.asp.

[19] M.G. Díaz Antón, “Propuesta de una metodología de desarrollo de Software educativo bajo un enfoque de calidad sistemática,” May. 2008;

http://www.academiainteractiva.com/ise.pdf.

[20] I. Jacobson, *El Proceso Unificado de Desarrollo de Software*, La Habana: Editorial Félix Varela, 2008.

Bibliografía

[1]“Apache2,” Ene. 2009; http://www.apache2.com.

[2]“CERT de Cuba”; www.cucert.cu.

[3]“Cert del MES,” Feb. 2009; www.cert-mes.mes.edu.cu.

[4]“Como usar el cron,” Jun. 2009; http://blogdrake.net/node/2171.

[5]“Como usar Syslog,” Jun. 2009; www.linux-magazine.es/issue/01/Syslog.pdf.

[6]“Configuración de crontab,” Jun. 2009; http://www.fabrega.info/documentos/cron/x32.html.

[7] P. Letelier Torres, “Desarrollo de Software Orientado a Objeto usando UML,” May. 2008; http://www.creangel.com/uml/intro.php.

[8] X. Ferrá Grau, “Desarrollo orientado a objetos con UML,” May. 2008; http://www.clikear.com/manuales/uml/introduccion.asp.

[9] I. Jacobson, *El Proceso Unificado de Desarrollo de Software*, La Habana: Editorial Félix Varela, 2008.

[10]“Falso positivo,” Jun. 2009; http://www.perantivirus.com/sosvirus/pregunta/falsopos.htm.

[11]“Financial Tech Magazine.,” Nov. 2006; http://www.financialtech-mag.com/000_estructura/index.php?id=24&idb=101&ntt=6729&sec=8&vn=1.

[12]“Gimp,” Ene. 2009; http://www.maestrosdelweb.com/editorial/%C2%BFque-es-javascript/.

[13]"Glosario Informática e Internet," Ene. 2008; http://glosario.panamacom.com/letra-s.html.

[14]"Herramientas de análisis para ficheros de registros de cortafuegos. ," Mar. 2008; http:// www.linux-magazine.es/issue/03/Analyzer.pdf .

[15]"Herramientas de análisis para ficheros de registros de cortafuegos. ," Mar. 2008; http:// www.linux-magazine.es/issue/03/Analyzer.pdf .

[16]"Introducción a CSS," Dic. 2008; http://www.maestrosdelweb.com/editorial/introcss/.

[17]"Introducción a JavaScript," Ene. 2009; http://www.maestrosdelweb.com/editorial/%C2%BFque-es-javascript/.

[18]"Introducción a NMap," Jun. 2009; http://www.maestrosdelweb.com/editorial/nmap/.

[19]"Manual de HTML," Ene. 2009; http://www-app.etsit.upm.es/~alvaro/manual/manual.html.

[20]"Mejorar el rendimiento del servidor PostgreSQL," Feb. 2009; http://www.cybercanibal.com/index.php/Tunning-Postgres.html.

[21]"Nueva actualización del IDS de código abierto OSSEC," Jun. 2009; http://www.idg.es/cio/Nueva_actualizaci%C3%B3n_del_IDS_de_c%C3%B3digo_abierto_OSSEC/doc54908-seguridad.htm.

[22]"OSSEC," May. 2008; http://www.ossec.net/main/about/.

[23]“PHP,” Ene. 2009; http://www.maestrosdelweb.com/editorial/desarrollo-web/php/.

[24]“PostgreSQL,” Ene. 2009; http://www.postgresql.org.

[25]“Programando tareas con Crontab,” Jun. 2009; http://www.encuentroalternativo.com/programando-tareas-con-crontab.

[26] M.G. Díaz Antón, “Propuesta de una metodología de desarrollo de software educativo bajo un enfoque de calidad sistemática,” May. 2008; http://www.academiainteractiva.com/ise.pdf.

[27]“Protocolos de comunicaciones,” Jun. 2009; http://www.telefonicaonline.com/on/io/es/atencion/tutoriales_articulos/firewall/A1.htm.

[28]“Python,” Ene. 2009, pág. http://www.python.org.

[29]“QuantaPlus,” Ene. 2009; http://docs.kde.org/kde3/es/kdewebdev/quanta/introduction-3-2.html.

[30]“Retina,” Jun. 2009; http://www.amodio.biz/jorge/inetar/docs/Pagina12-Retina-940316.pdf.

[31]“Retina_Datasheet,” Jun. 2009; www.eeye.com/html/assets/pdf/datasheet_retina_ES.pdf.

[32]“Sans,” Jun. 2009; http://www.sans.org/.

[33] N. Altalef, “Syslog Centralizado con detección de eventos.,” Ene. 2008; http://www.redklee.com.ar.

[34]“Syslog-ng La caja negra,” Jun. 2009; issuu.com/linuxmagazinespain/docs/lmes_01/62.

[35]“UNAM-CERT,” Jun. 2009; http://www.cert.org.mx/boletin/?vulne=4906.

[36]“Ventajas de python,” Jun. 2009; http://www.lawebdelprogramador.com/news/mostrar_new.php?id=79&texto=Python&n1=11498&n2=3&n3=0&n4=0&n5=0&n6=0&n7=0&n8=0&n9=0&n0=0.

[37]“Weird Solutions,” Oct. 2008; http://corporate.weird-solutions.com/products/syslog-turbo.

[38]“What’s new in GFI EventsManager 8?. ,” Oct. 2008; http://kbase.gfi.com/showarticle.asp?id=KBID003285.

Glosario de Términos

Backbone: Se refiere a las principales conexiones troncales de una red.

CERT: El término CERT proviene de las siglas en inglés *Computer Emergency Response Team* y viene a definir a un equipo de personas dedicado a la implantación y gestión de medidas tecnológicas con el objetivo de mitigar el riesgo de ataques contra los sistemas de la comunidad a la que se proporciona el servicio. También es conocido por las siglas CSIRT (Computer Security and Incidente Response Team) y ofrece servicios de respuesta ante incidentes y de gestión de seguridad.

Cron: En el sistema operativo Unix, cron es un administrador regular de procesos en segundo plano (*demonio*) que ejecuta procesos o Scripts a intervalos regulares (por ejemplo, cada minuto, día, semana o mes).

CSS: Las hojas de estilo en cascada son un lenguaje formal usado para definir la presentación de un documento estructurado escrito en HTML o XML.

DDOS: Denegación distribuida de servicios.

GNU/Linux: es el término empleado para referirse al sistema operativo similar a Unix que utiliza como base las herramientas de sistema de GNU y el núcleo Linux.

IDS: Sistema detector de intrusos.

IP: Internet Protocol (Protocolo de Internet). Uno de los protocolos más representativos del estándar TCP/IP, es el responsable del esquema de direccionamiento utilizado en Internet y define la estructura y el intercambio de los datagramas IP entre redes distantes, definido en el RFC 791.

LAN: Local Area Network (Red de área local). Red de datos para dar servicio a un área geográfica máxima de unos pocos kilómetros cuadrados, por lo cual pueden optimizarse los protocolos de señal de la red para llegar a velocidades de transmisión de Gbps (gigabits por segundo).

MD5: En criptografía, MD5 (abreviatura de Algoritmo de Resumen del Mensaje 5) es un algoritmo de reducción criptográfico de 128 bits ampliamente usado.

NMap: Network Mapper.

HTTP: HyperText Transfer Protocol (Protocolo de Transferencia de Hipertexto) Protocolo usado para la transferencia de documentos WWW.

Open Source: En la actualidad Open Source es utilizado para definir un movimiento nuevo de Software, diferente al movimiento del Software Libre, aunque no completamente incompatible con este, de modo que es posible (como de hecho ocurre) que ambos movimientos trabajen juntos en el desarrollo práctico de proyectos.

PHP: PHP (acrónimo recursivo de "PHP: Hypertext Preprocessor", originado inicialmente del nombre PHP Tools, o Personal Home Page Tools) es un lenguaje de programación interpretado. Se utiliza entre otras cosas para la programación de páginas Web activas, y se destaca por su capacidad de mezclarse con el código HTML.

Python: es un lenguaje de programación interpretado creado por Guido van Rossum en el año 1991.

RFC: Acrónimo inglés de Request For Comments "Petición de comentarios".Conjunto de archivos de carácter técnico donde se describen los estándares o recomendaciones. Resultado y proceso de creación de un estándar en Internet. Los nuevos estándares se proponen y publican en Internet como RFC. El grupo de trabajo de ingeniería de Internet (IETF) es un cuerpo de opinión que admite discusión a través de comentarios, en los que se establece un nuevo estándar.

SANS: SysAdmin Audit, Networking and Security Institute.

Script: En informática, un Script es un guión o conjunto de instrucciones. Permiten la automatización de tareas creando pequeñas utilidades. Es muy utilizado para la administración de sistemas UNIX. Son ejecutados por un intérprete de línea de órdenes y usualmente son archivos de texto.

Shell: Intérprete de comandos.

Spoofing: En términos de seguridad de redes hace referencia al uso de técnicas

de suplantación de identidad generalmente con usos maliciosos.

SSH: Acrónimo en inglés de "**S**ecure **SH**ell" en español: intérprete de órdenes seguro. Es un protocolo que sirve para acceder a máquinas remotas a través de una red.

SSL: Protocolo de Capa de Conexión Segura, son protocolos criptográficos que proporcionan comunicaciones seguras por una red, comúnmente Internet.

Syslog-ng: es una interfaz que provee un Framework estándar para que tanto programas como el mismo sistema operativo puedan generar mensajes que pueden ser almacenados localmente o ser enviados a un Host remoto.

Telnet: Nombre de un protocolo de red que sirve para acceder mediante una red a otra máquina.

TCP: Transmission Control Protocol (Protocolo de Control de Transmisión). Protocolo de nivel de transporte estándar de la internet definido en STD 7, RFC 793.

TCP/IP: Transmission Control Protocol/Internet Protocol (Protocolo de Control de Transmisión/Protocolo Internet). Convenio de los protocolos de aplicación y transporte que se utiliza en el IP. Incluye FTP, TELNET, SMTP y UDP.

UCf: Universidad de Cienfuegos.

URL: localizador uniforme de recurso. Es una secuencia de caracteres, de acuerdo a un formato estándar, que se usa para nombrar recursos, como documentos e imágenes en Internet, por su localización.

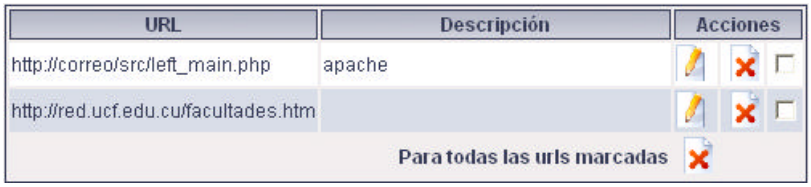
VLAN: (acrónimo de 'red de área local virtual') es un método de crear redes lógicamente independientes dentro de una misma red física.

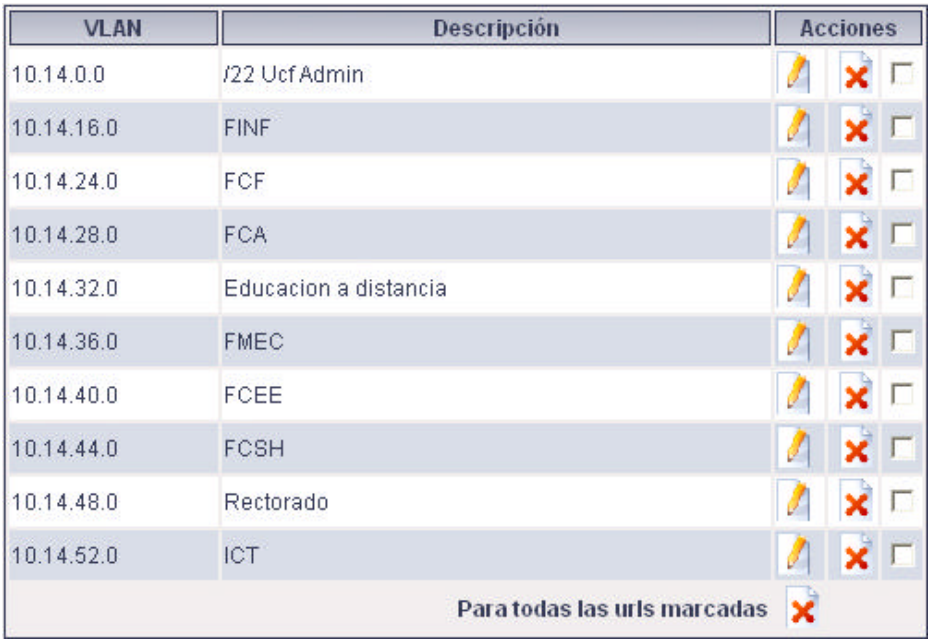
Anexos

Anexo 1: Prototipo del caso de uso Autenticarse.

Anexo 2: Prototipo del caso de uso Gestionar datos del administrador.

Anexo 3: Prototipo del caso de uso Gestionar criterios de configuración.





VLAN	Descripción	Acciones
10.14.0.0	/22 Ucf Admin	
10.14.16.0	FINF	
10.14.24.0	FCF	
10.14.28.0	FCA	
10.14.32.0	Educacion a distancia	
10.14.36.0	FMEC	
10.14.40.0	FCEE	
10.14.44.0	FCSH	
10.14.48.0	Rectorado	
10.14.52.0	ICT	

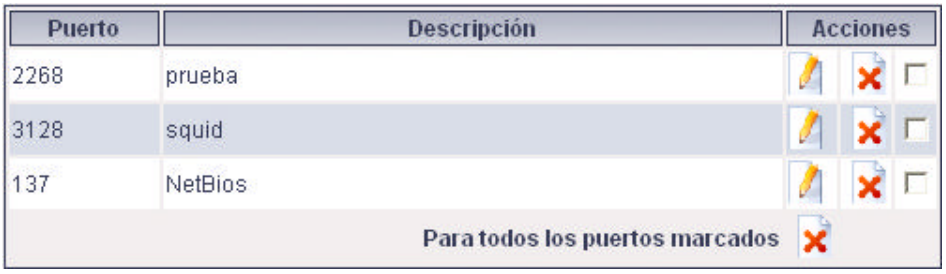
Para todas las urls marcadas

Agregar Puerto

Puerto:

Descripción:

Aceptar Cancelar



Puerto	Descripción	Acciones
2268	prueba	
3128	squid	
137	NetBios	

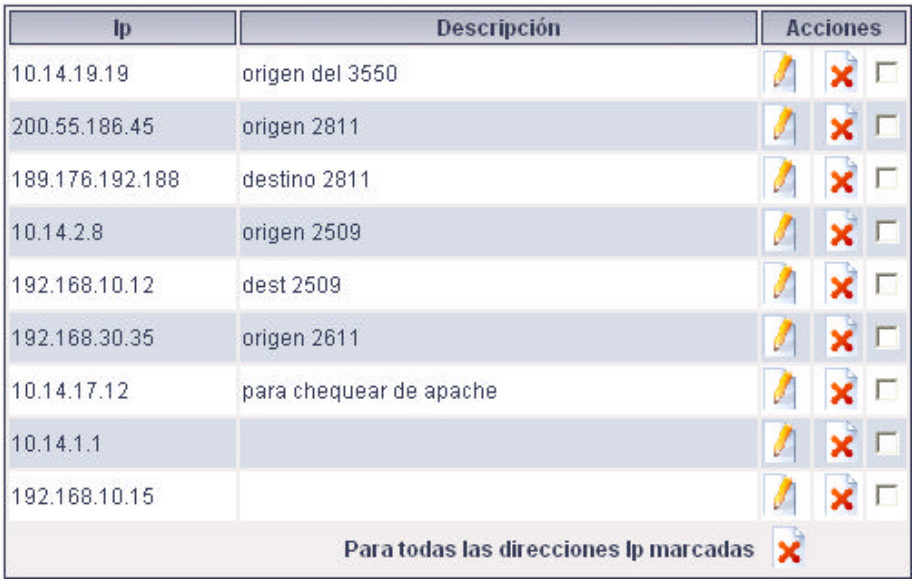
Para todos los puertos marcados

Agregar Dirección Ip

Dirección:

Descripción:

Aceptar Cancelar



Ip	Descripción	Acciones
10.14.19.19	origen del 3550	
200.55.186.45	origen 2811	
189.176.192.188	destino 2811	
10.14.2.8	origen 2509	
192.168.10.12	dest 2509	
192.168.30.35	origen 2611	
10.14.17.12	para chequear de apache	
10.14.1.1		
192.168.10.15		

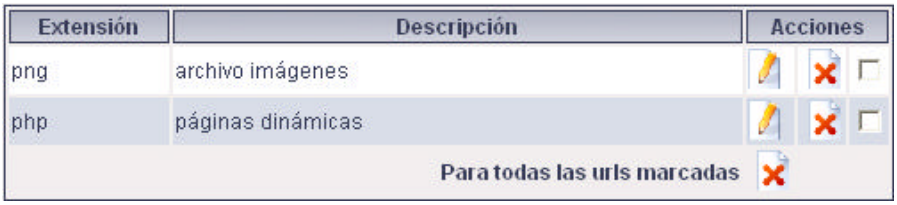
Para todas las direcciones Ip marcadas

Agregar Extensión de Fichero

Extensión:

Descripción:

Aceptar Cancelar



Extensión	Descripción	Acciones
png	archivo imágenes	
php	páginas dinámicas	

Para todas las urls marcadas

Agregar Patrón

Patrón:

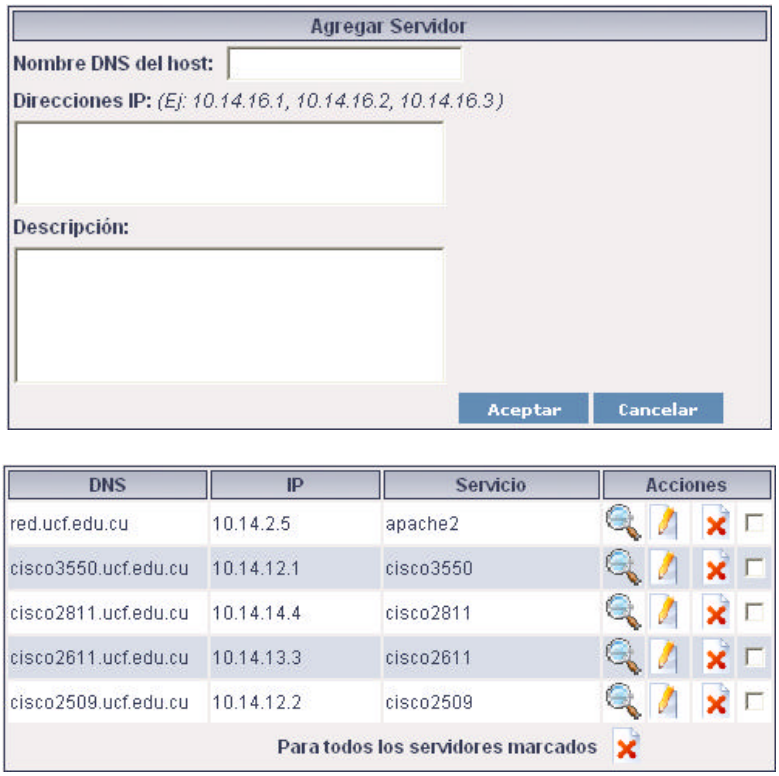
Descripción:

Aceptar Cancelar

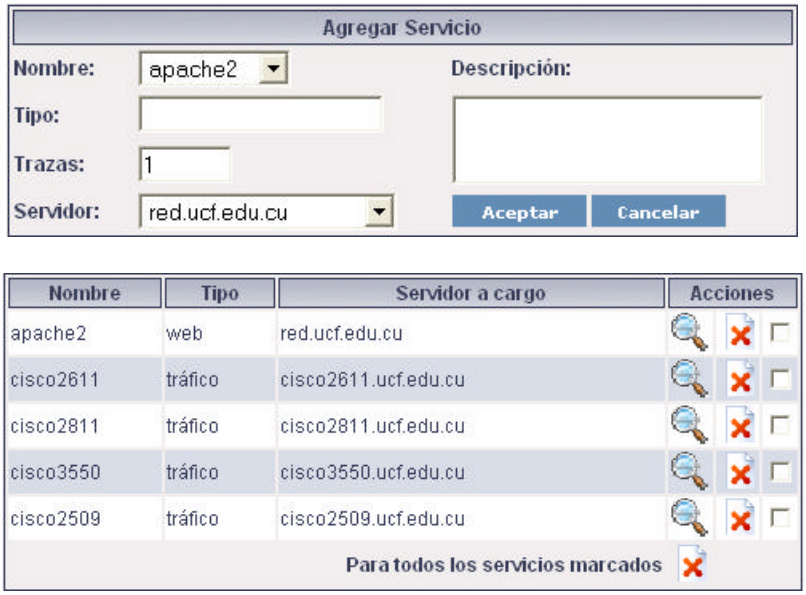
Patrón	Descripción	Acciones
welcome.jpg.1	18-11-2008 Gusano Windows 80	
ecard.exe	28-10-2008 Spyware Windows 80/TCP	
Postal17102008.exe	24-10-2008 Troyano Windows 80/TCP	

Para todos los puertos marcados

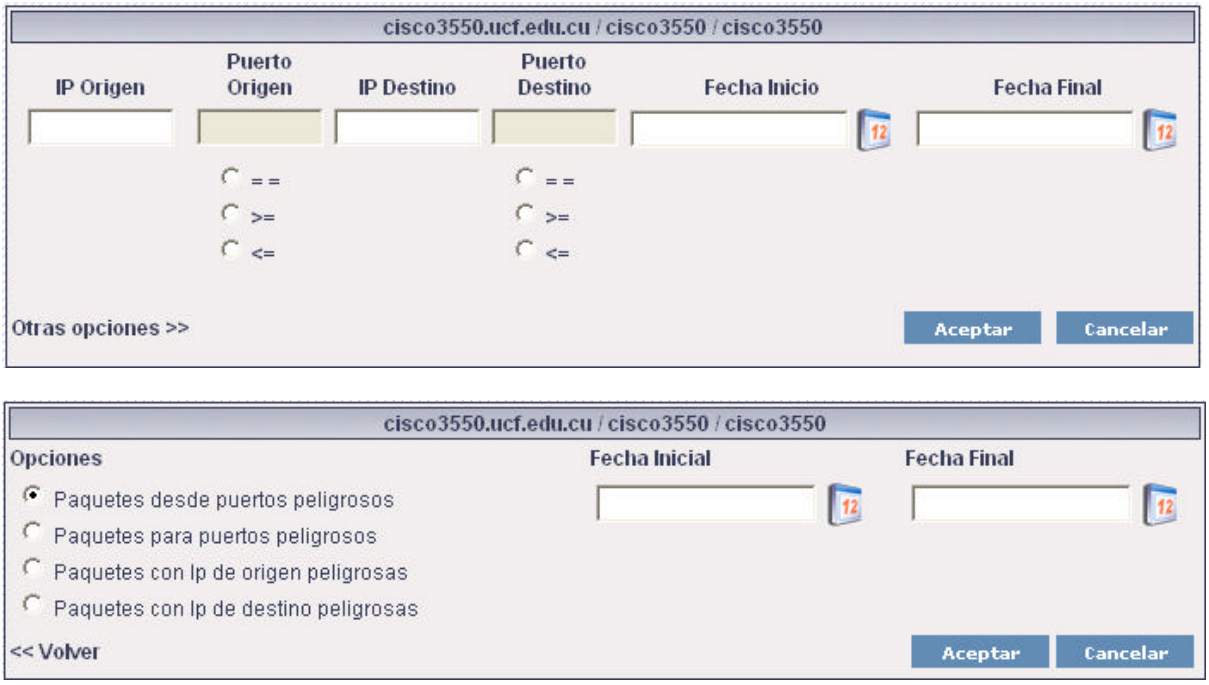
Anexo 4: Prototipo del caso de uso Gestionar los datos de servidores y/o dispositivos de red.



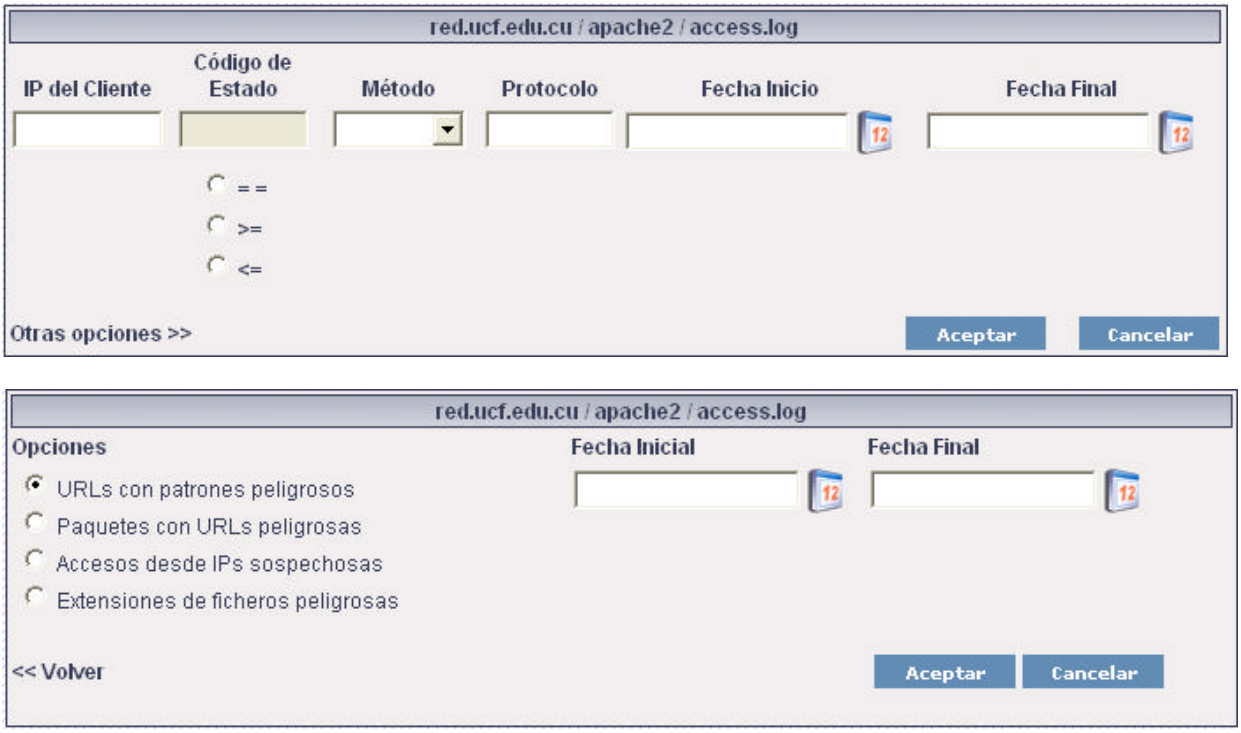
Anexo 5: Prototipo del caso de uso Gestionar los datos de los servicios telemáticos.



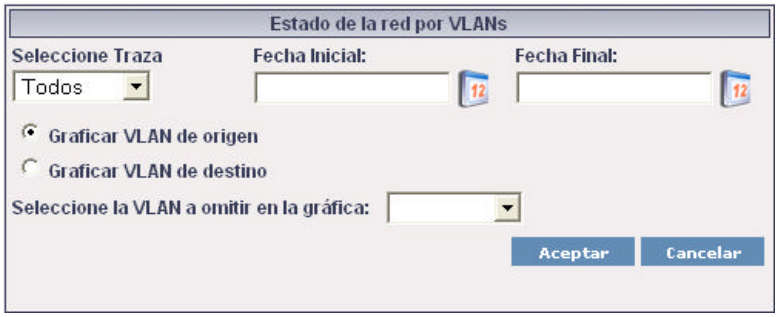
Anexo 6: Prototipo del caso de uso Procesar los datos almacenados de las trazas de los dispositivos de red.



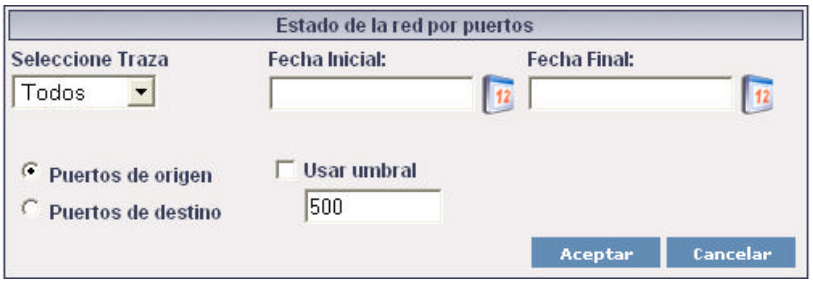
Anexo 7: Prototipo del caso de uso Procesar los datos almacenados de las trazas del servicio Web Apache2.



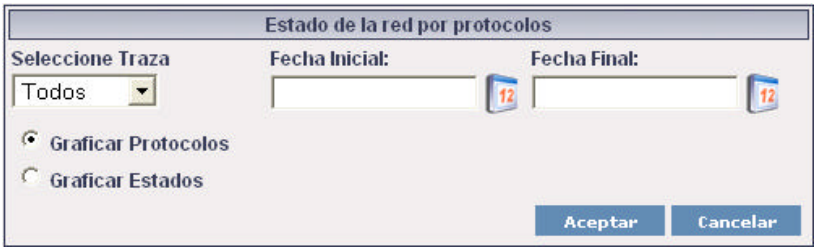
Anexo 8: Prototipo del caso de uso Mostrar estado de la red por VLAN.



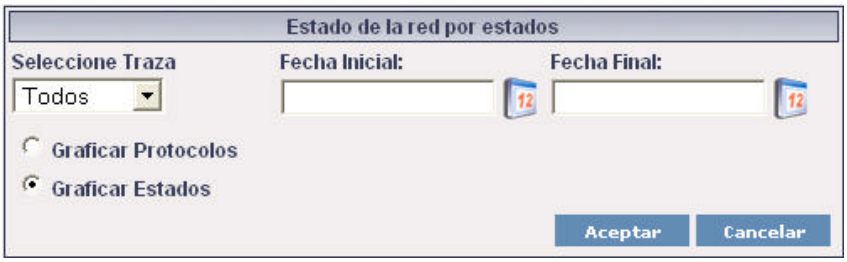
Anexo 9: Prototipo del caso de uso Mostrar estado de la red por puertos.



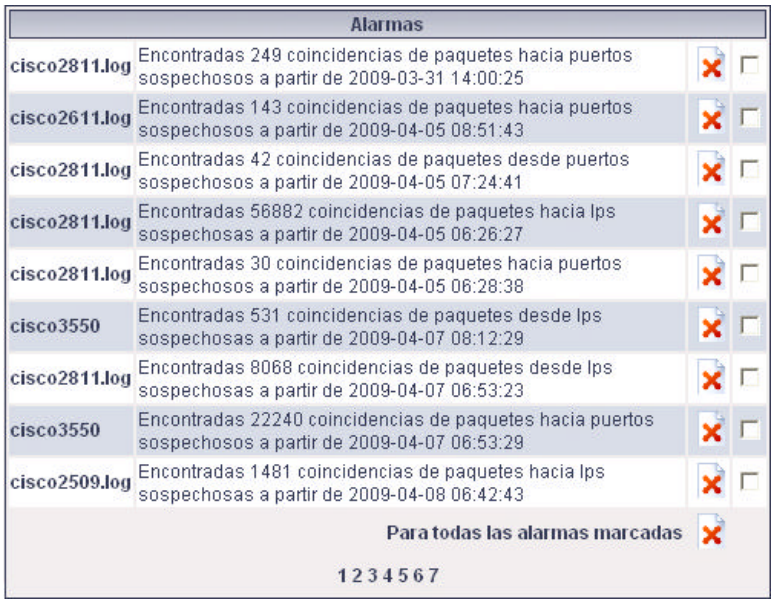
Anexo 10: Prototipo del caso de uso Mostrar estado de la red por protocolos.



Anexo 11: Prototipo del caso de uso Mostrar estado de la red por estados.



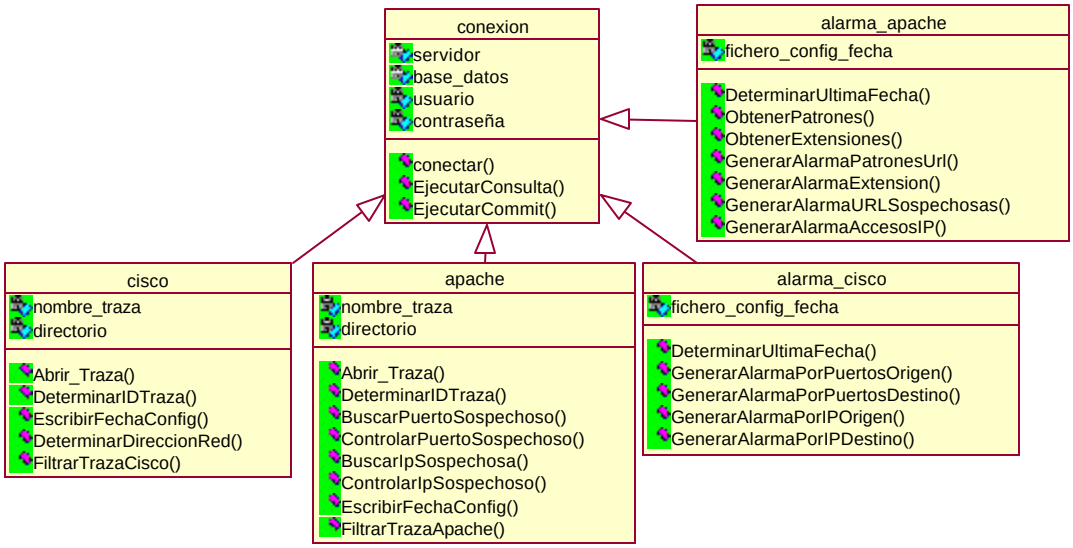
Anexo 12: Prototipo del caso de uso Gestionar Alarmas.



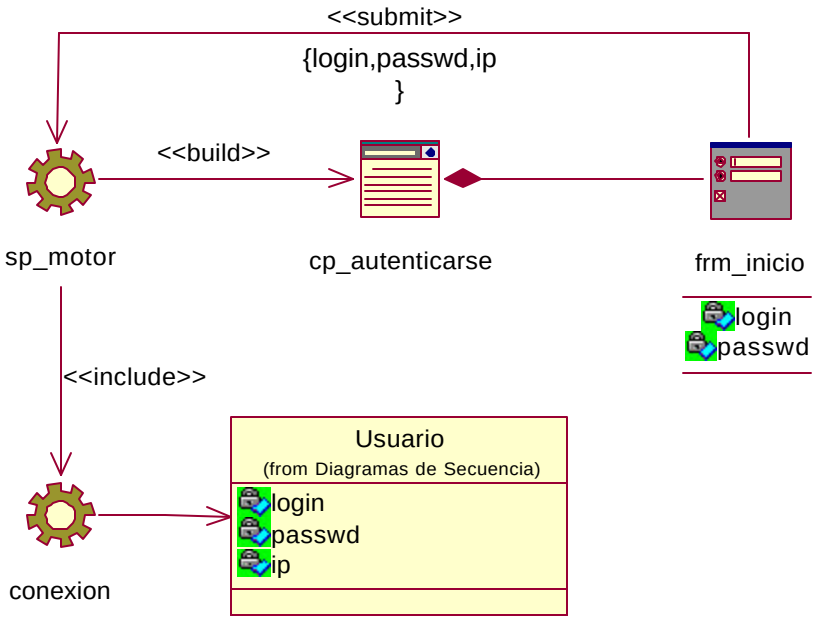
Anexo 13: Prototipo del caso de uso Cerrar sesión.

Anexo 14: Prototipo del caso de uso Consultar ayuda del sistema.

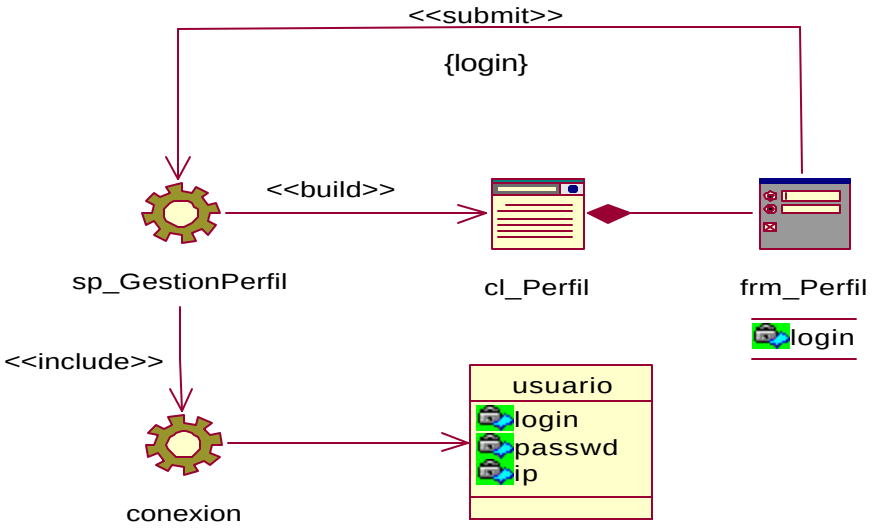
Anexo 15: Diagrama de clases del módulo de captura de la información contenida en las trazas.



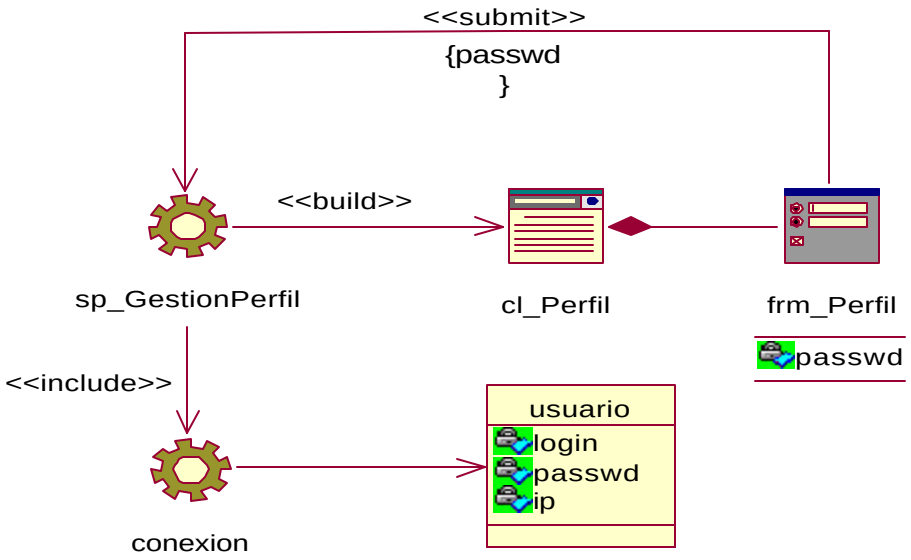
Anexo 16: Diagrama de clases Web. Autenticarse.



Anexo 17: Diagrama de clases Web. Gestionar datos del administrador.

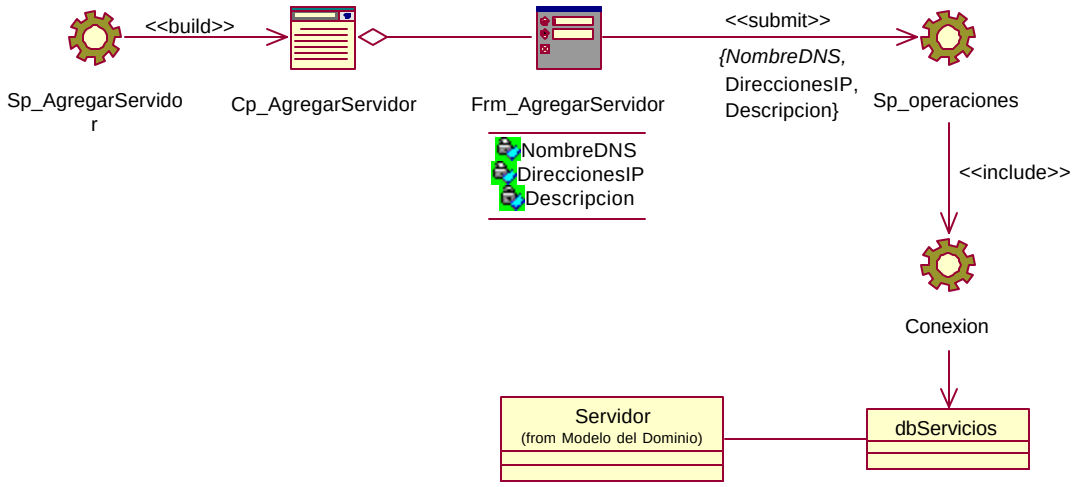


Cambiar Cuenta

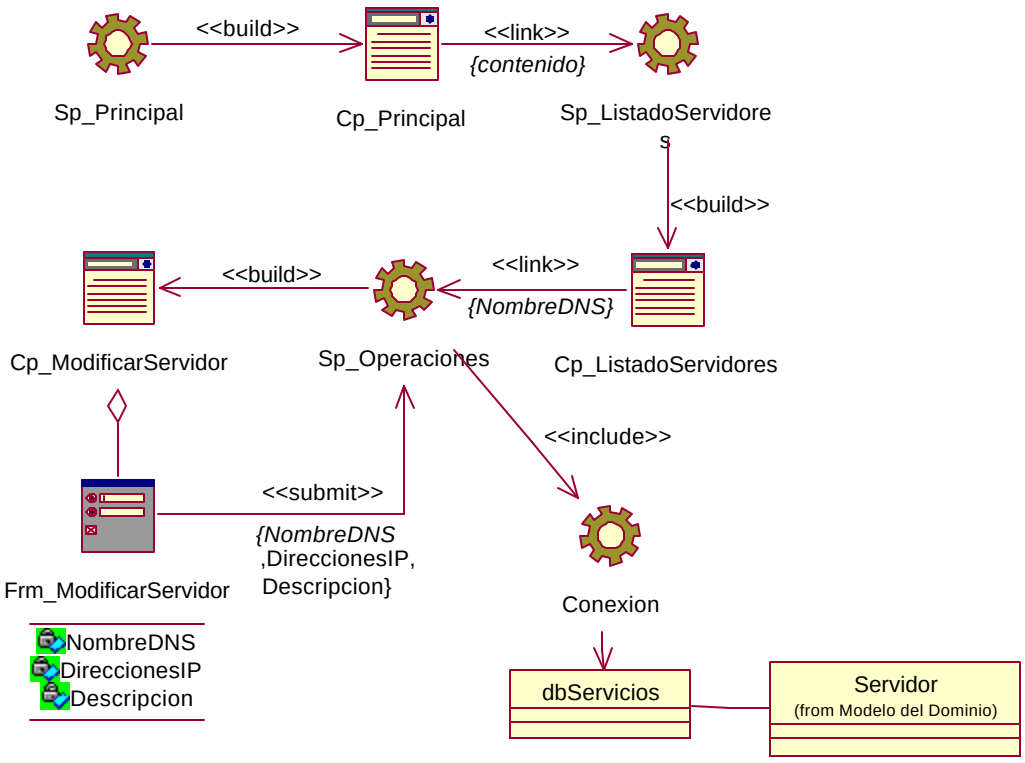


Cambiar contraseña

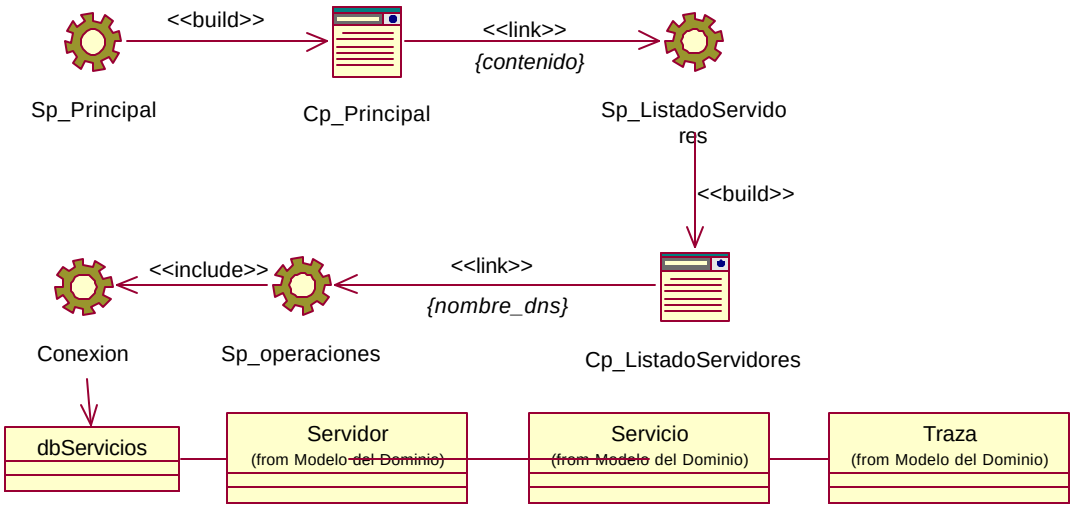
Anexo 18: Diagrama de clases Web. Gestionar los datos de servidores y/o dispositivos de red.



Agregar Servidor o dispositivo de red.

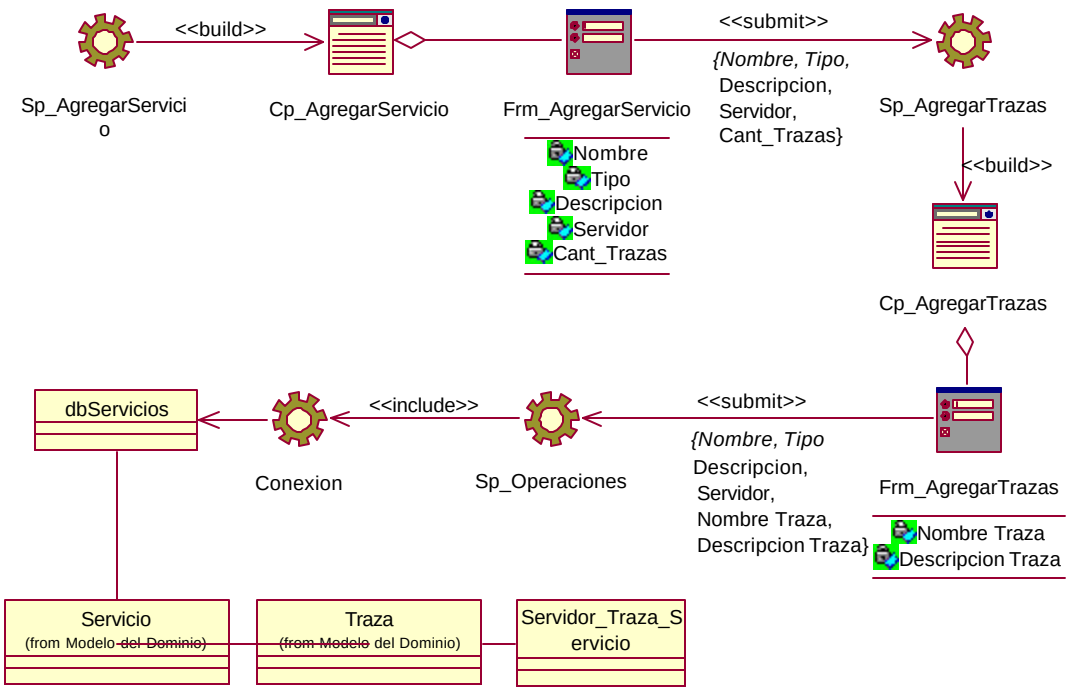


Modificar datos de Servidor o dispositivo de red.

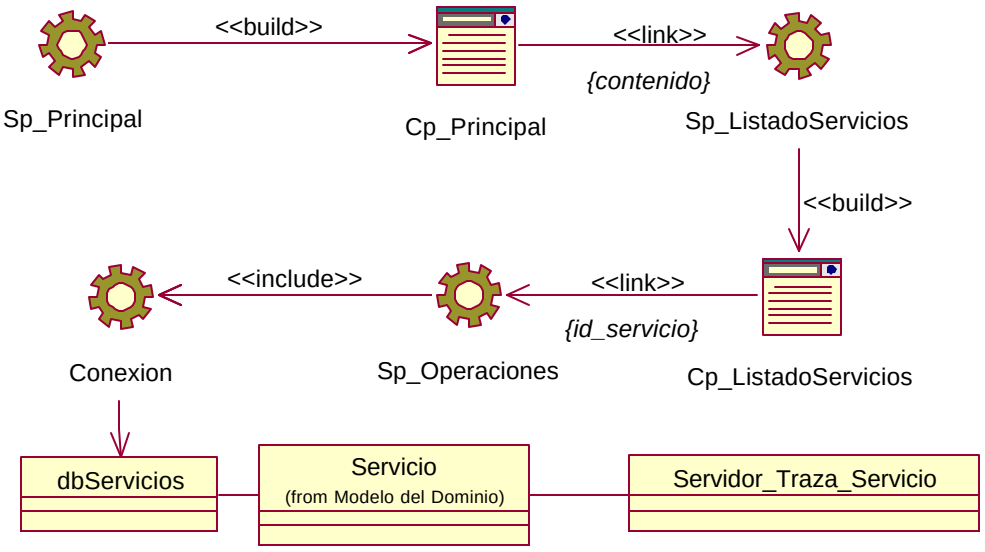


Eliminar datos de Servidor o Dispositivo de Red.

Anexo 19: Diagrama de clases Web. Gestionar los datos de los servicios telemáticos.

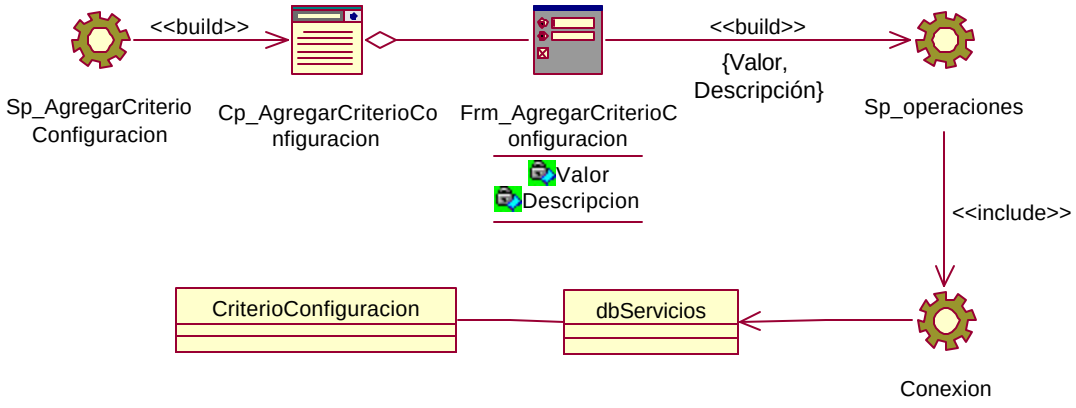


Agregar Servicio Telemático.

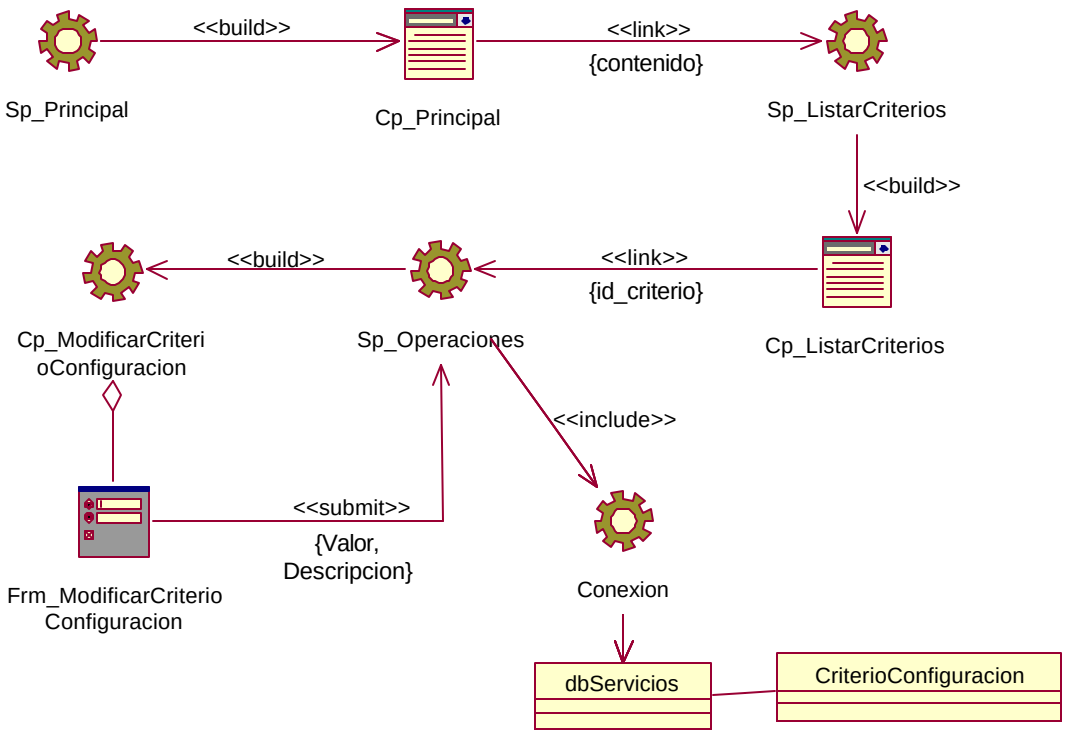


Eliminar Servicio Telemático.

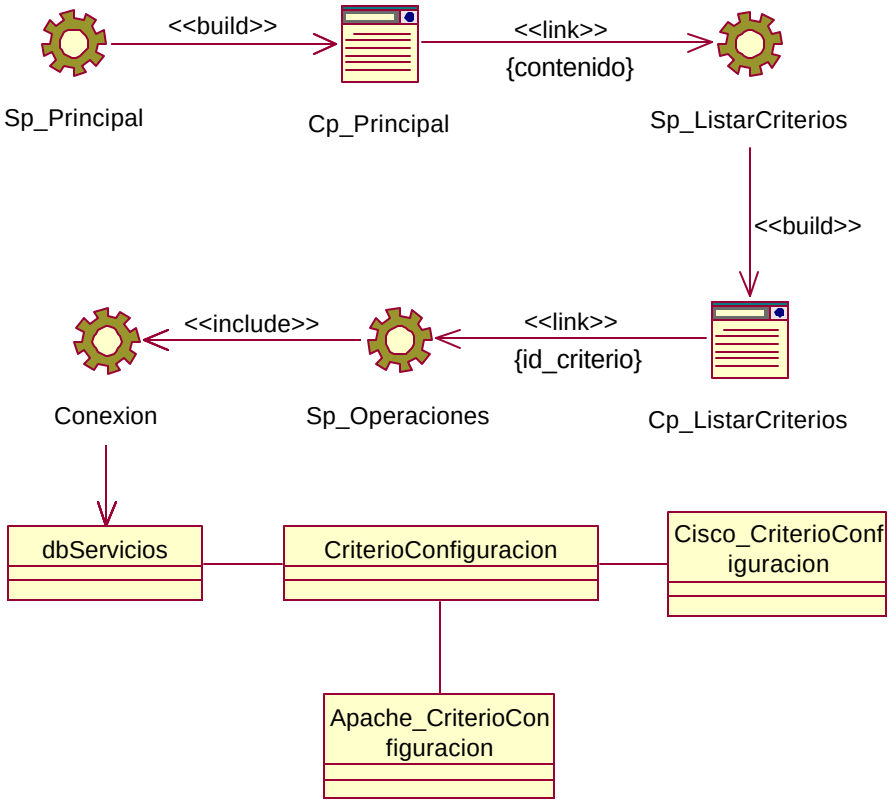
Anexo 20: Diagrama de clases Web. Gestionar Criterios de Configuración.



Agregar Criterio de Configuración

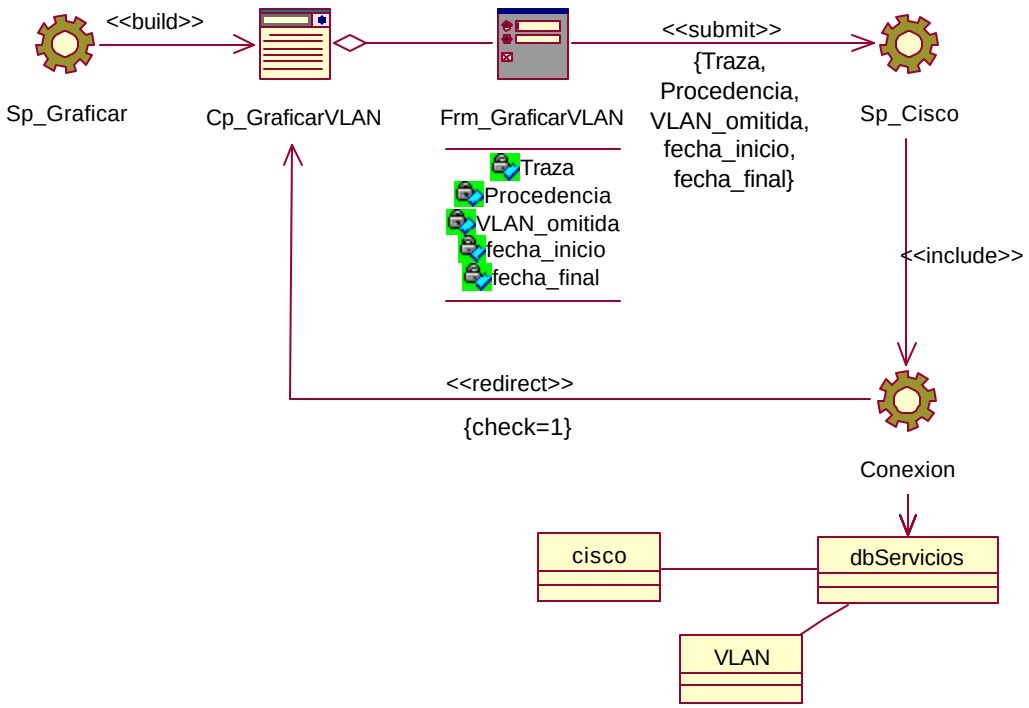


Modificar Criterio de Configuración

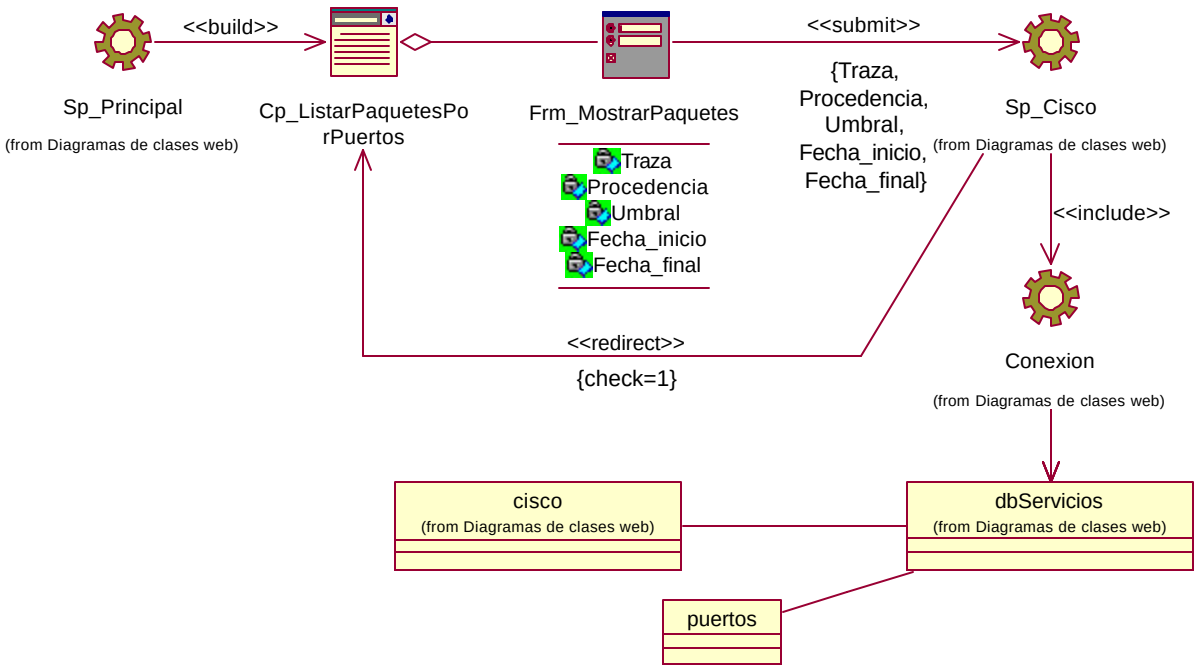


Eliminar Criterio de Configuración

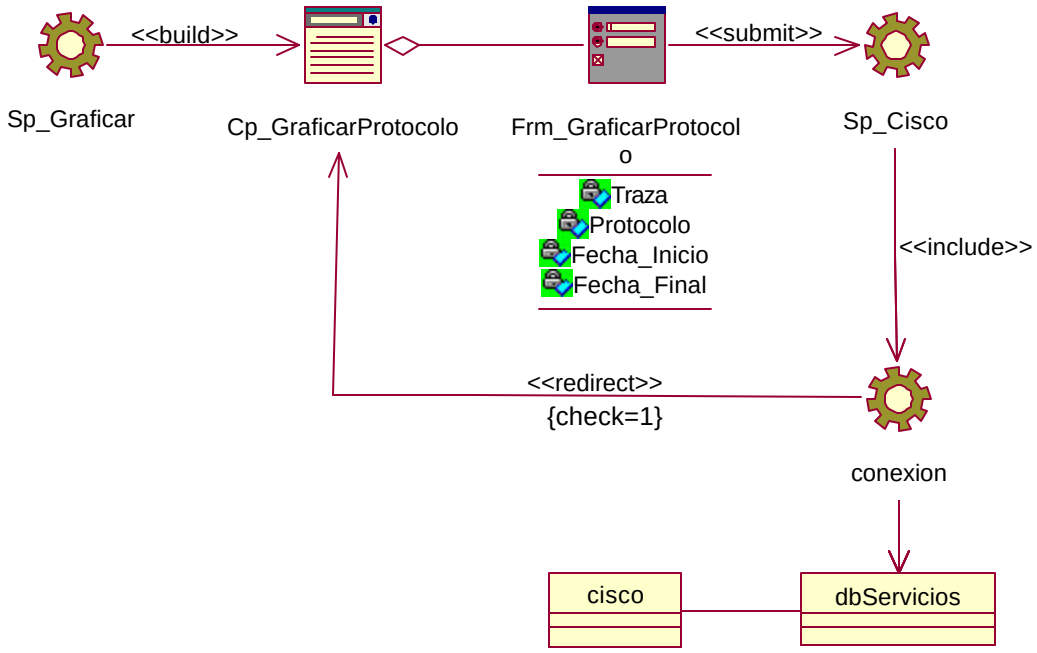
Anexo 21: Diagrama de clases Web. Mostrar Estado de la red por VLAN.



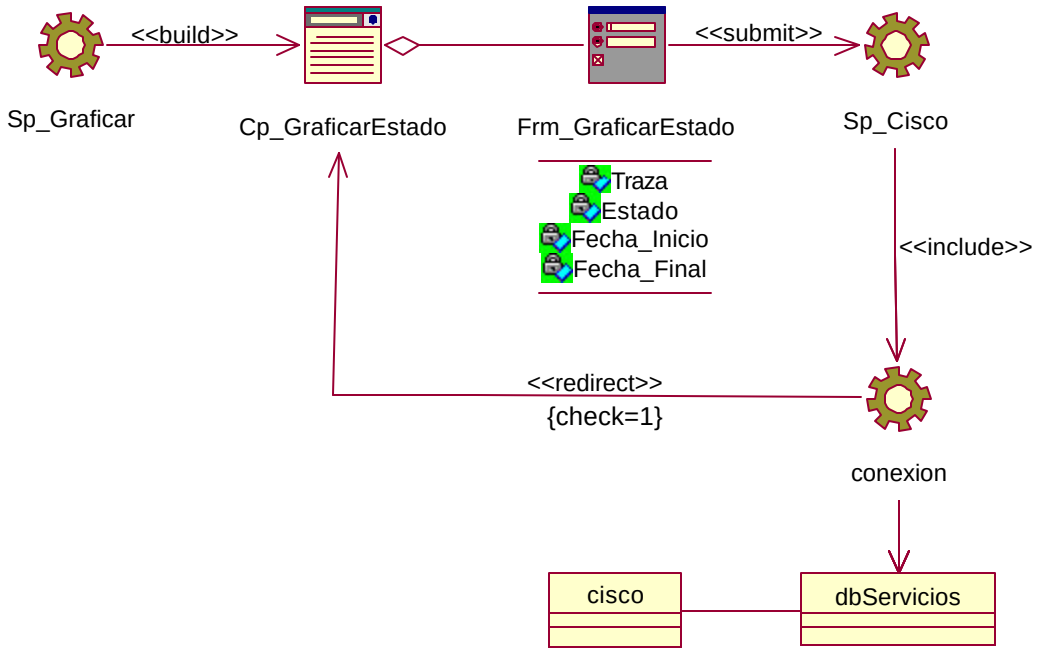
Anexo 22: Diagrama de clases Web. Mostrar estado de la red por puertos.



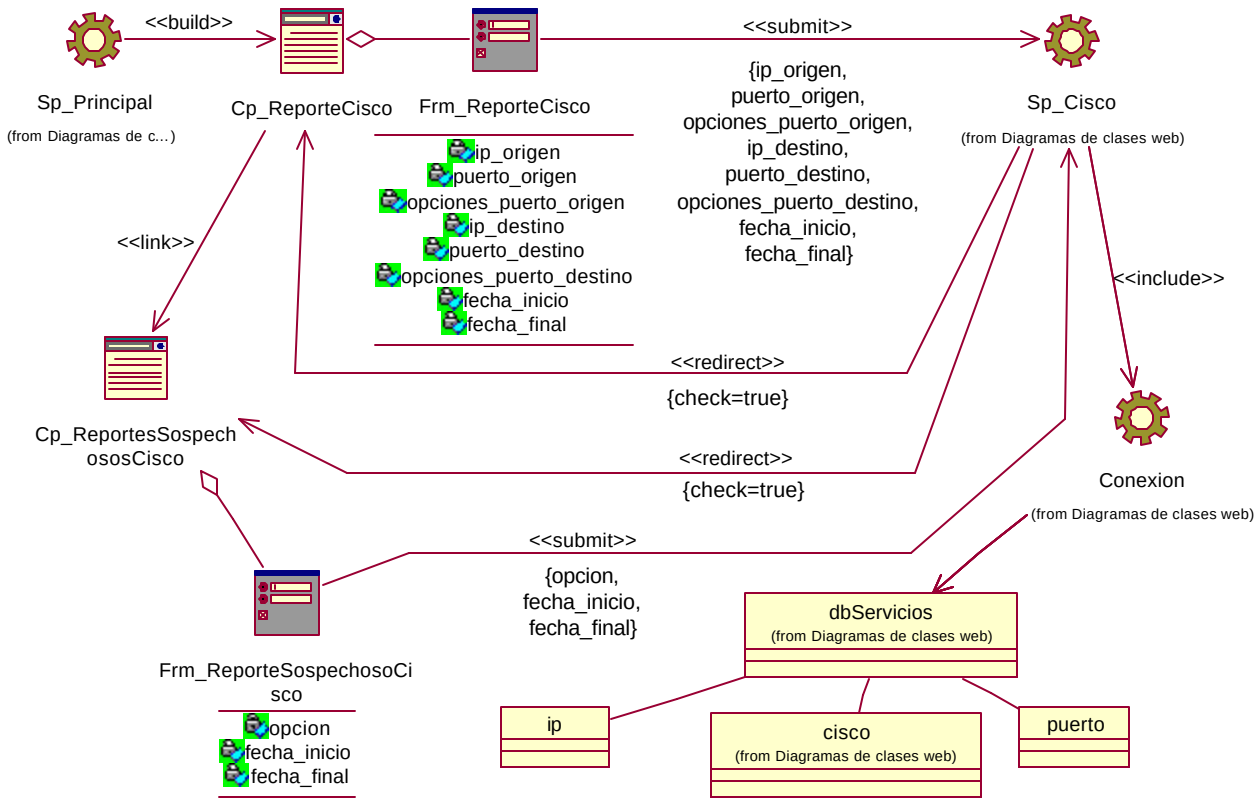
Anexo 23: Diagrama de clases Web. Mostrar estado de la red por protocolos.



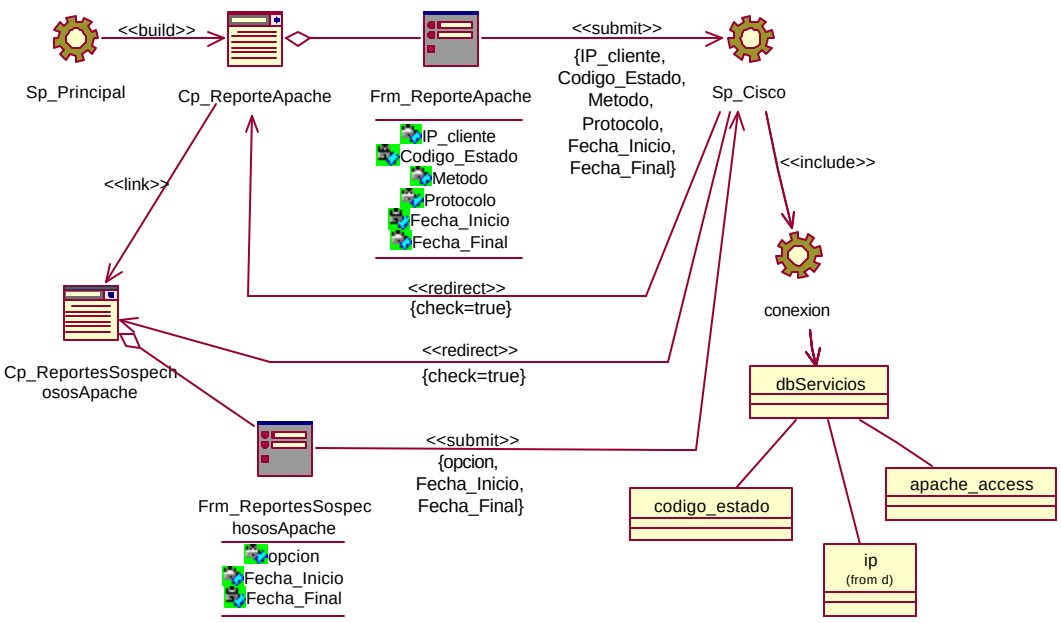
Anexo 24: Diagrama de clases Web. Mostrar estado de la red por estados.



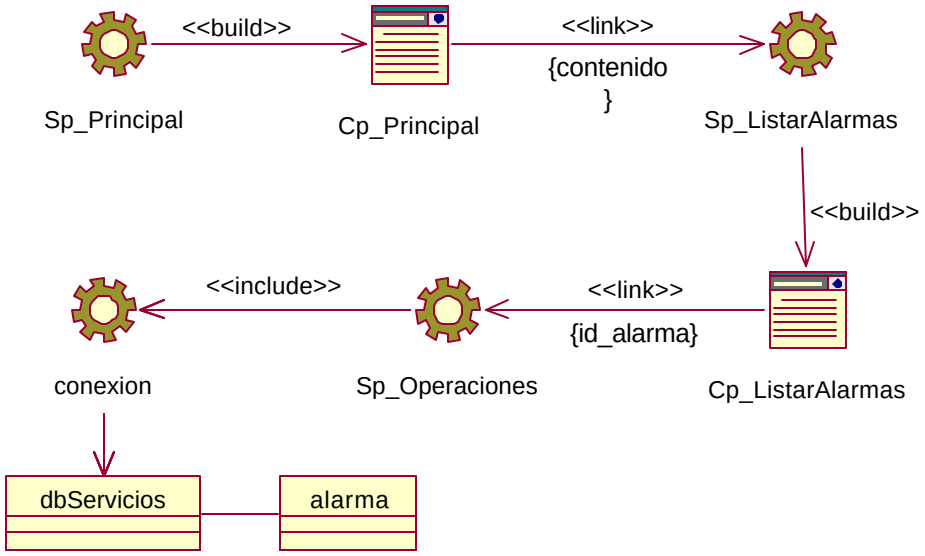
Anexo 25: Diagrama de clases Web. Procesar trazas de dispositivos de red.



Anexo 26: Obtención de la dirección de red desde donde vienen y van los paquetes.

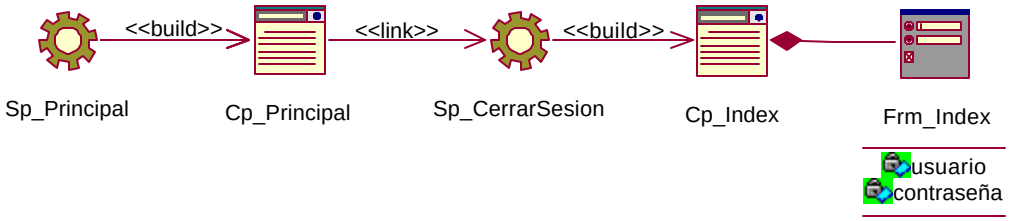


Anexo 27: Diagrama de clases Web. Gestionar alarmas.

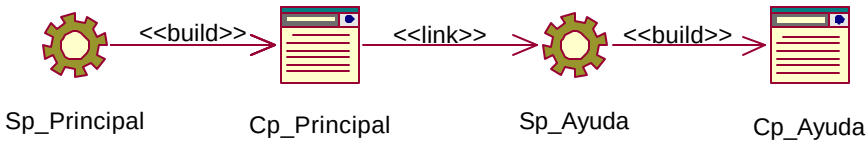


Eliminar alarmas

Anexo 28: Diagrama de clases Web. Cerrar sesión.



Anexo 29: Diagrama de clases Web. Consultar ayuda del sistema.



Anexo 30: Obtención de la dirección de red de origen y destino de los paquetes.