



Universidad Simón Bolívar
Departamento de Computación y Tecnología de la Información
Curso de Redes I CI-4835
Trimestre Septiembre - Diciembre 2017

Taller Cerrado #3

Objetivo Principal del Taller:

Que los alumnos diseñen y construyan una aplicación final que funcione apropiadamente en la red usando la API de sockets de Berkeley.

Introducción:

La banca moderna ha recurrido al uso de la Internet y de las nuevas tecnologías de la información como elementos claves para mejorar su funcionamiento. Así que muchos de su servicios son ofrecidos con apoyo de computadores y redes de comunicaciones electrónicas, tal como muestra la Figura 1.



Figura 1: Imagen que simula el servicio de un sistema telecajero automático (ATM)

De ese modo se disminuyen costos, se aumenta la productividad y se refuerza la competitividad por los clientes. Sin embargo, si no se refuerza la confiabilidad de los sistemas, este soporte se convierte en algo contraproducente. Por eso se agregan distintos mecanismos e instrumentos, para hacer que los sistemas interactúen entre sí en tiempo real y detecten cualquier anomalía. Se disminuyen costos, se aumenta la productividad y se acrecienta la competitividad por atender satisfactoriamente los clientes. La Figura 2 muestra un posible esquema de interacción de un ATM con sus servidores en la sede principal de un banco.

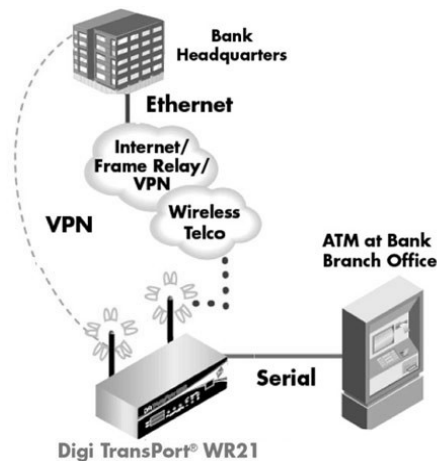


Figura 2: Esquema de interacción entre un ATM y los computadores centrales de un banco

Por otra parte, siempre hay cabida para agregar nuevos mecanismos de supervisión y detección de delitos o intentos de cometerlos. El banco “RedBanc” ha decidido probar el funcionamiento de un prototipo de un sistema de vigilancia redundante (SVR) para sus equipos ATM con una organización lógica, tal como muestra la Figura 3.



Figura 3: Organización lógica del prototipo SVR

En esa figura se muestra como los cajeros automáticos seguirán interconectados a los servidores centrales del banco y como además estarán en comunicación con un equipo especial que ejecutará el SVR. El banco decidió además contratarle para que desarrolle el prototipo de software de modo que se puedan hacer todas las pruebas básicas del sistema.

Descripción de la operatividad del SVR:

El objetivo principal del SVR es ofrecer un sistema en tiempo real para la monitorización del funcionamiento de los ATM. No tiene intervención alguna para autorizar o negar ninguna operación financiera del ATM. Únicamente opera para establecer alarmas inmediatas, en caso de que se detecte que está ocurriendo algún tipo de anomalía; la tipificación de posibles incidentes le ha sido participado previamente. En caso de que deban enviarse, las alarmas serán procesadas por correo electrónico a los supervisores que se le haya indicado durante la etapa de configuración.

Para hacer eso posible, cada ATM deberá disponer de un programa que se comunice con el SVR y le enviará copia completa de cada operación bancaria que en tiempo real haga el ATM. El módulo central del SVR recibirá cada comunicación y examinará su contenido en búsqueda de algún patrón, que se le haya señalado que debe detectar y advertir. En caso de que la información recibida contenga algún patrón de los que busca, emitirá la alarma de inmediato y almacenará toda la información como parte de sus bitácoras. Si la información no contiene ninguna actividad que deba reconocer, únicamente la guardará.

Toda entrada de una bitácora deberá contener al menos los datos de la siguiente tupla:

(serial, fecha, hora, identificación del ATM, código del evento, patrón reconocido, información recibida)

Para que el SVR opere con “confiabilidad”, debe ser capaz de:

- 1.- Procesar hasta 1000 equipos ATM simultáneamente.
- 2.- Recuperarse ante cualquier falla de conectividad y en caso de que esta sea permanente (más de 5 minutos sin intercambio alguno de datos), debe notificar de inmediato como una alarma más.
- 3.- Estar en la capacidad de registrar todas las operaciones o intentos de ellas, y no perder dicha información en caso de que sufra algún percance.

Por otra parte, los tipos de mensajes que puede recibir el SVR de un ATM son como el que seguidamente se muestra:

```

003:28:2012:10:38:200004 The peripheral device is busy
003:28:2012:10:45:260004 Response of device not completely received
003:28:2012:10:52:420004 Device asked for termination
003:28:2012:13:39:250004 Application started
003:28:2012:13:41:000004 Ports Verification
003:28:2012:13:41:390004 Communication Offline
...
004:03:2012:12:15:000004 Device Status Change
004:03:2012:12:16:000004 Low Cash alert
004:03:2012:12:20:000004 Running Out of notes in cassette 50
004:03:2012:12:20:000004 Cassette 50 empty
...
004:03:2012:14:50:250004 Application started
004:04:2012:15:35:200004 Service mode entered
004:04:2012:15:48:000004 Safe door opened
004:04:2012:15:49:190004 TOP CASSETTE removed
004:04:2012:15:49:530004 TOP CASSETTE inserted
004:04:2012:15:51:000004 SECOND CASSETTE removed
004:04:2012:15:51:050004 SECOND CASSETTE inserted
004:04:2012:15:51:380004 THIRD CASSETTE removed
004:04:2012:15:52:060004 THIRD CASSETTE inserted
004:04:2012:15:52:060004 BOTTOM CASSETTE removed
004:04:2012:15:52:250004 BOTTOM CASSETTE inserted
004:04:2012:15:52:350004 Safe door closed

```

Figura 4: Ejemplo de data que suministrará el ATM

Adicionalmente, para el prototipo a instrumentar los patrones de texto que el SVR debe identificar son los siguientes:

- 1.- “Communication Offline”
- 2.- “Communication error”
- 3.- “Low Cash alert”
- 4.- “Running Out of notes in cassette”
- 5.- “empty”
- 6.- “Service mode entered”
- 7.- “Service mode left”
- 8.- “device did not answer as expected”
- 9.- “The protocol was cancelled”
- 10.- “Low Paper warning”
- 11.- “Printer Error”
- 12.- “Paper-out condition”

Trabajo a desarrollar:

El software que usted debe desarrollar deberá estar conformado en varias partes, un módulo que residirá en los ATM, que únicamente transmitirá copia de los datos de las operaciones realizadas en ese equipo, y otro que operará en el SVR. El módulo que centraliza toda la

información en el SVR será el que realizará el examen en busca de los patrones y emitirá las medidas requeridas. La aplicación que se ejecutará en el ATM se denominará “svr_c” y debe leer sus datos de la *entrada estándar de datos*, mientras que la parte que se ejecutará en el módulo central se denominará “svr_s” y recibirá sus datos de las transmisiones de red.

Como el prototipo debe ser transportable y ejecutarse en modo rápido, el banco ha solicitado que se desarrolle en lenguaje C. Adicionalmente, cada módulo deberá poder ser ejecutado desde la *línea de comandos* con una sintaxis similar a la de Linux®. Esta será más adelante descrita en modo exacto.

Por otra parte, algunas de las consideraciones adicionales que usted debe asumir en su trabajo y que se darán como completamente válidas son:

- 1.- Los canales de comunicación de la red bancaria son seguros. Esto quiere decir, que no requiere cifrar ninguna información y que los medios de transmisión no serán interrumpidos como parte de ningún accidente o sabotaje.
- 2.- El “hardware” donde corre el código del SVR es totalmente confiable y no puede ser manipulado indebidamente.
- 3.- Todos los equipos operarán bajo TCP/IP versión 4 y cada sistema contiene un nombre de dominio y una dirección IP única. E.g. (*atm1.redbanc.com.ve / 190.23.45.27*)

Sintaxis de las invocación de comandos:

La sintaxis para la invocación de la ejecución del módulo en el ATM será la siguiente:

```
svr_c -d <nombre_módulo_central> -p <puerto_svr_s> [-l <puerto_local>]
```

Donde:

<nombre_módulo_central>: Es el nombre de dominio o la dirección IP (versión 4) del equipo en donde se deberá ejecutar el módulo central del SVR.

<puerto_svr_s>: Es el número de puerto remoto en que el módulo central atenderá la llamada.

<puerto_svr_local>: Es el número de puerto local que el software utilizará para comunicarse con el módulo central.

Por su parte, la sintaxis para invocar el software que recibe la información de los ATM es:

```
svr_s -l <puerto_svr_s> -b <archivo_bitácora>
```

Donde:

<puerto_svr_s> Es el número de puerto local en que el módulo central atenderá la llamada.

<archivo_bitácora> Es el nombre y dirección relativa o absoluta de un archivo de texto, a modo de bitácora, que refleja las operaciones que se hicieron y registra además las alarmas emitidas. Cada línea del archivo, es una entrada que refleja alguna operación que se efectuó o que intentó llevarse a cabo. Por simplicidad del trabajo, toda la atención de los ATM se almacenará en un única bitácora.

Condiciones de la entrega.

El trabajo se desarrollará en equipos de tres (3) estudiantes, pero todos los integrantes deben estar en capacidad de explicar y desarrollar cualquier parte del trabajo, aún cuando originalmente no la haya realizado. El plazo de entrega será informado por los canales oficiales de información del curso.

Entrega del Trabajo (Informe y Código):

Al finalizar su trabajo, cada equipo debe entregar un informe impreso y el código fuente de su desarrollo. El documento deberá con incluir al menos los siguientes contenidos:

- .- Descripción general de la solución instrumentada.
- .- Explicación detallada de cómo ejecutar el sistema. Debe cubrirse el proceso de compilación, ejecución y finalización. El sistema debe contener un *makefile* que permita la correcta compilación de todos sus componentes en modo simple al invocar el comando *make*. Para ejecutar los programas se debe seguir de forma estricta la sintaxis especificada en el enunciado, de no ser así se considerará que el proyecto no funciona y no será corregido.
- .- Aspectos del proyecto que funcionan según el enunciado y cuáles no.
- .- Código debidamente documentado de todos los programas, siguiendo los estándares de documentación de Javadoc, con el programa y los encabezados de las funciones implementadas.

Adicionalmente, todos sus programas deben seguir las buenas prácticas de estilo de programación en C y todas las llamadas al sistema deben ser correctamente manejadas.

Criterio de Evaluación

- .- Las calificaciones del proyecto podrá arrojar notas distintas para cada integrante del equipo.
- .- El equipo evaluador se reservará la construcción de un baremo y un conjunto de programas automatizados para comprobar el resultado de su trabajo.

.- También es posible que el equipo evaluador distribuya otro documento que complemente la información o detalle cualquier aspecto que sea necesario aclarar.

Plataforma donde será evaluado su trabajo

.- Los proyectos serán evaluados en los equipos del Laboratorio de Computación (LDC).

.- Para las pruebas de su proyecto utilice como número de puertos un número de 5 cifras que comience por 2 que las siguientes 4 cifras coincidan con las últimas cuatro cifras del número de carnet de alguno de los integrantes del equipo. Si se requiere más de un puerto, repita el procedimiento pero con el número de carnet de otro integrante.

Grupo Docente de Redes de Computadores I (GDRC I)