



Tema	Grupo	Ayudante		Evaluación Grupal del TP
Z			Nota	
			Fecha	
			Firma	

Integrantes						
Padrón	Apellido	Nombre	E-mail	Asistencias		Evaluación Individual

Evaluación Grupal de la Entrega				
	Aprobada	Se Debe Rehacer	No Entrega	Observaciones
Presentación				
Hipótesis Planteadas				
Problemas Relevantes				
Comandos Solicitados				
Instalación				
Demonio				
Configuración				
Antifraude				
Comandos Auxiliares				
Archivos Auxiliares				
Funcionamiento General				



Índice

Condiciones de Resolución y Corrección	3
Introducción.....	3
Documentación a Presentar	3
Contenido de la Carpeta.....	3
Formato y contenido del Disquete.....	4
Observaciones a tener en cuenta en el desarrollo.....	4
Enunciado.....	5
AFINSTAL, Comando de Instalación	5
Descripción	5
Antes de empezar	5
Pasos sugeridos para AFINSTAL.....	6
AFIMONIO, Demonio de Llamadas.....	11
Descripción	11
Observaciones.....	11
AFICONF, Configurador de Umbrales.....	12
Archivos de Input.....	12
Archivos de Output.....	12
Descripción del Caso de Uso.....	12
Observaciones	13
ANTIFRAUDE, Generador de alarmas	14
Archivos de Input.....	14
Archivos de Output.....	14
Procesar Todos los archivos	14
Procesar Un archivo	14
Procesar Un registro.....	14
Requerimientos NO Funcionales.....	15
Observaciones	15
Archivos	16
Archivo de Log	16
Directorio	16
Estructura	16
Archivo de Instalación.....	16
Directorio	16
Estructura	16
Ejemplo para el comando antifraude:.....	16
Archivo de Orígenes/Destinos.....	16
Directorio	16
Estructura de oridesti.txt	16
Ejemplo	16
Archivo de Umbrales de Orígenes/Destinos Sospechosos	17
Directorio	17
Estructura de umbrales.param	17
Archivos de Llamadas	17
Directorios.....	17
Estructura de Llamadas.....	17
Archivo de Alarmas	18
Directorio	18
Estructura de alarmas.txt.....	18



Condiciones de Resolución y Corrección

Introducción

1. Se deberá tener en cuenta para la resolución TODAS las condiciones que se enuncian.
2. Se deben respetar los formatos de archivos especificados
3. Se debe respetar la estructura de directorios planteada
4. Cada comando deberá ser resuelto utilizando el lenguaje indicado.
5. El día de vencimiento del TP, cada ayudante convocará a los integrantes de un grupo e iniciará la corrección mediante una entrevista grupal. Es imprescindible la presencia de todos los integrantes del grupo el día de la corrección. El objetivo de esto es comprender la dinámica de trabajo del equipo y los roles desempeñados por cada uno.
6. Posterior a la entrega se podrá acordar entre el Ayudante y el Grupo el intercambio de correspondencia a través de la cuenta so7508@hotmail.com. Por este medio el ayudante podrá solicitar a los alumnos correcciones, mejoras en las instrucciones de ejecución, nuevos juego de datos, etc. Cabe aclarar que todos los mensajes a esta casilla deberán tener como asunto "GrupoX" (x es el número asignado al grupo) para poder redireccionar el mail al ayudante correspondiente.
7. Dentro de los ítems a chequear el ayudante evaluará **aspectos formales** (como ser la forma de presentación de la carpeta), **aspectos funcionales** (como ser que el TP funcione) y **aspectos operativos**: que el TP resuelva el problema planteado.

Documentación a Presentar

Se deberá hacer entrega de:

- Una Carpeta de Trabajo Práctico con el contenido solicitado mas adelante. En el pie de cada hoja se debe incluir: Número de Grupo y Tema (en el margen izquierdo) y Número de Hoja (en el margen derecho). TODAS las hojas deberán estar numeradas y enganchadas a una carpeta. Las hojas sueltas no se considerarán como parte de la misma.
- Un disquete de Trabajo Práctico con el formato y contenido solicitado mas adelante. El disquete debe tener Etiqueta con el Número de Grupo y Tema.

Contenido de la Carpeta

1. **Carátula:** La entregada en este mismo documento con los datos completos en 2 COPIAS.
2. **Índice del Contenido de la Carpeta.**
3. **Hipótesis y Aclaraciones:** todas las hipótesis que han sido consideradas en la resolución del TP y cualquier otra aclaración que se considere necesaria.
4. **Problemas relevantes:** Enumeración de los problemas relevantes que se les hayan presentado durante el desarrollo y prueba de los comandos y como los solucionaron.
5. **Archivo README: Instructivo de Instalación**
6. **Comandos Desarrollados:** para cada comando desarrollado incluir cuando corresponda:
 - a. **Nombre del Comando**
 - b. **Tipo de comando:** Solicitado o Auxiliar (los comandos auxiliares son aquellos que no forman parte del set de comandos solicitados.)
 - c. **Justificación de su uso** (solo para los comandos auxiliares)
 - d. **Archivos de Input**
 - e. **Archivos de Output**
 - f. **Parámetros**, si corresponde
 - g. **Opciones**, si corresponde
 - h. **Ejemplos de invocación** con distintos parámetros y valores, indicando en que oportunidad es usado y cuál es el resultado previsto.
 - i. **Listado del código** del comando: éste deberá incluir comentarios útiles para la comprensión del código.
7. **Archivos**
 - a. **Archivos del enunciado:** Por cada archivo del enunciado indicar:
 - i. Nombre del Archivo y Estructura.
 - b. **Archivos auxiliares:** Por cada archivo auxiliar que se utilice indicar:
 - i. Nombre del Archivo y Estructura



ii. Justificación de su uso

8. Apéndice A

- a. **Condiciones de resolución, entrega y corrección:** se obtiene del presente documento.
- b. **Enunciado:** se obtiene del presente documento.

Formato y contenido del Disquete

El disquete de Trabajo Práctico deberá estar en formato DOS y deberá tener un único archivo instalable en formato ".tgz" con todos los archivos y directorios empaquetados en un archivo "tar" y luego comprimido con "gzip".

MUY IMPORTANTE: cuando se efectúa el "tar" se debe usar la opción de "directorio relativo" para evitar problemas cuando se proceda a la instalación en la sala de máquinas de la facultad para su corrección.

El instalable deberá contener:

- Los scripts desarrollados (según se detallan mas adelante)
- Juego de datos de ejemplo y de prueba lo suficientemente variado como para contemplar todas las variantes posibles. Los archivos de prueba deben respetar las estructuras y directorios indicados en el enunciado ya que es uso y costumbre intercambiar datos entre los distintos grupos para enriquecer los testeos.
- Un archivo README con las instrucciones de instalación y cualquier otra indicación u observación que se quiera incorporar para que el ayudante pueda realizar los testeos
- Cualquier otro archivo que el grupo considere necesario (ej: resultado de las pruebas por Uds. Efectuadas, logs, etc.)

Observaciones a tener en cuenta en el desarrollo

1. Toda invocación desde un comando a otro debe devolver un código de retorno cero (0) si fue exitoso o distinto de cero si tuvo errores. Siempre al finalizar el comando se debe indicar si finalizó correctamente o con errores.

2. En la descripción de los comandos se sugieren los pasos de ejecución. Estos deben considerarse indicativos, ya que el programador puede alterar este orden según su conveniencia siempre que no afecte el resultado final esperado. Se dan a los efectos de ordenar la explicación. Si Ud. considera conveniente modificar algún paso, tanto sea en el orden de ejecución o en la forma de resolverlo, no hay inconveniente, siempre que lo aclare debidamente en las hipótesis.

3. Errores y Mensajes:

3.1. Cuando se describen los comandos se hace referencia a errores y mensajes informativos, estas referencias son sólo una guía para el desarrollador, pueden existir errores no descriptos en el enunciado que deban ser manejados por el programador.

3.2. Siempre se debe especificar el error con un mensaje amigable.

3.3. Es conveniente para el seguimiento y corrección de los comandos que a medida que se avanza en la ejecución del mismo se vayan mostrando por consola mensajes descriptivos indicando el punto de avance y resultados intermedios.

3.4. Es conveniente que trabajen usando un archivo centralizado de mensajes de error para que de esta manera se estandaricen. Esto hablará no sólo de la calidad del trabajo sino del grado de integración del grupo.

4. Archivo de Log:

4.1. Los comandos ANTIFRAUDE, AFIMONIO y AFINSTAL deben grabar un archivo de Log, por ejemplo antifraude.log, afimonio.log y afinstal.log.

4.2. En el archivo de log se debe grabar todos los mensajes que se muestran por la consola del operador (3.3) y los mensajes de los errores que se produzcan (3.2)

4.3. La escritura en el archivo de Log debe ser homogénea para todos los comandos que generan registros de log.

4.4. Cada línea de log debe contener:

Fecha y hora actual, Usuario, Nombre del comando y Mensaje



Enunciado

Una empresa de telefonía celular desea desarrollar un sistema software denominado ANTIFRAUDE que le permita ir procesando archivos que contienen registros de llamadas telefónicas (que detecta mediante un “demonio de llamadas”) y generar alarmas cuando las mismas son de un destino considerado sospechoso.

Los umbrales para determinar si un destino es sospechoso, está dado por un conjunto reglas definidas con ayuda de un “configurador de umbrales”

El sistema software ANTIFRAUDE deberá estar compuesto por:

- Un programa Perl denominado *Antifraude.pl* que procesa las llamadas y genera las alarmas cuando corresponde, también llamado “generador de alarmas”
- Un shell script denominado *Aficonf* que genera los umbrales de origen/destino sospechoso que necesita *antifraude.pl* para procesar las llamadas, también llamado “configurador de umbrales”
- Un shell script denominado *Afimonio* que detecta el arribo de archivos de llamadas a ser procesadas por *antifraude.pl*, también llamado “demonio de llamadas”
- Y finalmente un shell script denominado *Afinstal* que permite la instalación de lo anterior, también llamado “comando de instalación”

AFINSTAL, Comando de Instalación

Descripción

Este es el script para la instalación del paquete *Antifraude*, el cual está compuesto por los comandos:

- ANTIFRAUDE.PL
- AFICONF
- AFIMONIO

El script debe permitir instalar los 3 comandos al mismo tiempo, o de a uno por vez.

Cuando este script interactúa con el usuario, el valor por omisión propuesto (o valor default) se debe mostrar entre paréntesis ().

Siempre que este proceso cancele, mostrar el siguiente mensaje:

```
"Proceso de instalación cancelado"
```

Los siguientes 3 directorios generados por AFINSTAL pueden ser escogidos por el usuario en el momento de la instalación, sin embargo, AFINSTAL debe proponer el siguiente default:

- Para los scripts ejecutables (AFICONF, AFIMONIO y ANTIFRAUDE): `$grupoxx/Antifraude/bin`
- Para los archivos de datos: `$grupoxx/Antifraude/llamadas`
- Para los archivos de log: `$grupoxx/Antifraude/log`

Antes de empezar ...

Antes de iniciar la ejecución de AFINSTAL es necesario copiar desde un medio externo el instalable. Estos pasos deberán ser explicados dentro del instructivo de instalación (archivo README.txt), a modo de ejemplo podemos enumerar los siguientes pasos:

1. Crear el directorio `$grupoxx/tmp`
2. Copiar el archivo `*.tgz` en ese directorio
3. Descomprimir el `*.tgz` de manera de generar un `*.tar`
4. Extraer los archivos del tar.

Estos pasos deben generar AL MENOS la siguiente estructura de directorio con AL MENOS los siguientes archivos:

- `$grupoxx/Antifraude/conf` con las tablas de configuración
- `$grupoxx/Antifraude/inst` con los archivos relativos al paquete de instalación



Pasos sugeridos para AFINSTAL

Recordar lo indicado en el [punto 2](#) de "Observaciones a tener en cuenta en el desarrollo".

1. Detectar si Antifraude o alguno de sus componentes ya está instalado

El script de instalación se debe preparar de manera tal que detecte si el paquete Antifraude ya se encuentra instalado.

1.1. Si todo el paquete ya esta instalado, mostrar el siguiente mensaje:

```
*****
* Se encuentran instalados los siguientes componentes:      *
* Antifraude.pl instalado el ... (fecha) por ... (usuario)  *
* Aficonf instalado el ... por (usuario de instalación)     *
* Afimonio instalado el (fecha de instalación) por ...     *
*                                                            *
*          Copyright TPSistemasOp (c)2004                    *
*****
```

Proceso de Instalación Cancelado

1.2. Si alguno de sus componentes no esta instalado, entonces continuar con la instalación del componente faltante. Atención: en este punto se debe tener en cuenta la estructura de directorio ya generada (almacenada en \$grupo/antifraude/conf/afinstal.conf)

1.3. Si ningún componente Antifraude esta instalado, continuar

2. Chequear si es el SO correcto

Antifraude se comercializa en tres versiones: una compatible con Unix/Linux, otra para Windows y otra para Mac.

Aunque nosotros solo desarrollaremos los scripts compatibles con Unix/Linux, debemos tener una manera de chequear si es el sistema operativo correcto.

Para chequear esto, procederemos de la siguiente manera:

Si el script de instalación se llama AFINSTAL, diremos que es el script para Unix/Linux

Si el script de instalación se llama AFINSTAW, diremos que es el script para Windows

Si el script de instalación se llama AFINSMAC, diremos que es el script para Mac

De esta manera, con solo renombrar el script AFINSTAL podremos testear esta funcionalidad.

2.1. Mostrar un mensaje:

```
" Chequeando SO ..."
```

y obtener el SO en donde se esta ejecutando el script. Almacenar este valor en la variable SISOPTP.

ACLARACION: Toda vez que se solicita "Almacenar tal valor en tal variable" es para facilitar la explicación de los pasos subsiguientes.

2.2. Si el script de instalación se llama AFNSTAW, entonces mostrar el siguiente mensaje:

```
" ERROR!!!
Este proceso de instalación es para Windows
y su Sistema Operativo es $SISOPTP
Solicite a su proveedor la versión adecuada
e inicie nuevamente el proceso de instalación."
Cancelar el proceso.
```

2.3. Si el script de instalación se llama AFINSMAC, entonces mostrar el siguiente mensaje:

```
" ERROR!!!
Este proceso de instalación es para Mac
y su Sistema Operativo es $SISOPTP
Solicite a su proveedor la versión adecuada
```



e inicie nuevamente el proceso de instalación."
Cancelar el proceso.

- 2.4. Si el script de instalación se llama AFINSTAL, entonces mostrar el siguiente mensaje:
" Proceso de instalación para \$SISOPT
Presione ENTER para continuar"
Continuar

3. Aceptación de términos y condiciones

Mostrar en consola el siguiente mensaje:

```
*****
*
*      Proceso de Instalación de "Antifraude"      *
*      Copyright TPSistemasOp (c)2004             *
*
*****
A T E N C I O N: Al instalar Antifraude UD. expresa estar
en un todo de acuerdo con los términos y condiciones del
"ACUERDO DE LICENCIA DE SOFTWARE" incluido en este paquete.
```

Si desea cancelar este proceso oprima 'F' y luego ENTER.
Para continuar oprima solamente ENTER."

Si el usuario oprime F y luego ENTER, Cancelar el proceso, mostrando el siguiente mensaje:
"Proceso de instalación cancelado"

Si el usuario solo dio ENTER, Continuar

Si el usuario oprime cualquier otra cosa y luego ENTER, volver a Mostrar en consola el mensaje de términos y condiciones.

4. Chequear que Perl esté instalado (solo para antifraude.pl)

Prevía a la instalación de Antifraude es necesario chequear que Perl versión 5 o superior este instalado.

Si al chequear esto da error, entonces mostrar el siguiente mensaje:

```
"      ERROR !:

      Este proceso de instalación solo puede ser
      ejecutado si Perl 5 o superior ya esta instalado.

      Efectúe la instalación de Perl
      e inicie nuevamente el procedimiento de instalación."
```

Cancelar el proceso.

5. Mostrar el nombre del directorio de instalación

Determinar el directorio corriente y almacenar este valor en la variable CURRDIR.

Mostrar en pantalla el siguiente mensaje:

```
" Nombre del directorio de instalación: ($CURRDIR/inst) "
```

Almacenar este valor en la variable INSTDIR.

6. Mostrar el nombre del directorio de parámetros de configuración

Mostrar en pantalla el siguiente mensaje:

```
"Nombre del directorio de configuración: ($CURRDIR/conf) "
```

Almacenar este valor en la variable CONFDIR



7. Definir el directorio de ejecutables

Mostrar en pantalla el siguiente mensaje:

```
"Nombre del directorio de ejecutables: ($CURRDIR/bin) "
```

Si el usuario desea aceptar debe oprimir solamente ENTER, si lo desea cambiar, debe ingresar el path completo del directorio y luego ENTER. Almacenar este valor en la variable BINDIR.

8. Definir el directorio de datos (solo para antifraude.pl)

Mostrar en pantalla el siguiente mensaje:

```
"Nombre del directorio de datos: ($CURRDIR/llamadas) "
```

Si el usuario desea aceptar debe oprimir solamente ENTER, si lo desea cambiar, debe ingresar el path completo del directorio y luego ENTER. Almacenar este valor en la variable DATADIR

9. Definir el directorio de log, el nombre del archivo de Log y su tamaño máximo.

Mostrar en pantalla el siguiente mensaje:

```
"Nombre del directorio de log: ($CURRDIR/log) "
```

Si el usuario desea aceptar debe oprimir solamente ENTER, si lo desea cambiar, debe ingresar el path completo del directorio y luego ENTER. Almacenar este valor en la variable LOGDIR.

Luego, mostrar en pantalla el siguiente mensaje:

```
"Nombre del archivo de log: (nombre del comando.log) "
```

Si el usuario desea aceptar debe oprimir solamente ENTER, si lo desea cambiar, debe ingresar el nombre completo y luego ENTER. Almacenar este valor en la variable LOGNAME.

Luego, mostrar en pantalla el siguiente mensaje:

```
"Tamaño máximo para el archivo <LOGNAME> (en Mbytes): 200 Mb"
```

Si el usuario desea aceptar debe oprimir solamente ENTER, si lo desea cambiar, debe ingresar una cantidad que se interpreta como Mb y luego ENTER. Almacenar este valor en la variable LOGSIZE. Cuando el archivo de Log supere este tamaño, se debe truncar dejando solo las últimas 100 líneas. Esto es para evitar el crecimiento indefinido del archivo.

10. Mostrar, según corresponda a cada comando, la estructura de directorios resultante

Limpiar la pantalla y Mostrar en consola el siguiente mensaje:

```
"      PARAMETROS DE INSTALACION DE (nombre del comando) "  
"      ===== "  
"Nombre del directorio de instalación: $INSTDIR "  
"Nombre del directorio de ejecutables: $BINDIR "  
"Nombre del directorio de datos: $DATADIR "  
"Nombre del directorio de configuración: $CONFDIR "  
"Directorio y Nombre del archivo de Log: $LOGDIR/$LOGNAME "  
"Tamaño máximo para el archivo de Log: $LOGSIZE "  
"Si los datos ingresados son correctos oprima solamente ENTER, si desea  
modificar alguno de ellos oprima cualquier otra tecla y luego ENTER."
```

Si el usuario solo dio ENTER, continuar. Sino: limpiar la pantalla y volver al punto 5). En este caso, los valores default propuestos deben ser los contenidos en las variables: INSTDIR, BINDIR, DATADIR, CONFDIR, LOGDIR, LOGNAME, LOGSIZE, etc

11. Confirmar Inicio de Instalación

Mostrar en consola el siguiente mensaje:



"Si desea iniciar la instalación de <nombre del comando>, oprima solamente ENTER, si desea cancelar el proceso oprima cualquier otra tecla y luego ENTER."

Si el usuario solo dio ENTER, Continuar, sino, Cancelar el proceso

12. Crear la estructura de Directorio definida.

Crear la siguiente estructura y Mostrar en consola el siguiente mensaje:

```
/$BINDIR  
/$DATADIR  
/$DATADIR/aprocesar  
/$DATADIR/enproceso  
/$DATADIR/rechazadas  
/$DATADIR/procesadas  
/$LOGDIR/$LOGNAME
```

"Creando Estructura de Directorio....."

13. Mover archivos

Mostrar en consola el siguiente mensaje:

"Moviendo Archivos"

Mover los script ejecutables a BINDIR
Mover las tablas de configuración a CONFDIR
Mover los archivos de datos de prueba a DATADIR/aprocesar
Mover el resto de los archivos a INSTDIR

14. Instalar Sistema Antifraude.pl

Esto es solo un placebo de lo que sería una instalación real.

Mover antifraude.pl de INSTDIR al directorio BINDIR.

Mostrar en consola el siguiente mensaje:

" Instalación del componente Antifraude.pl completada"

15. Instalar Aficonf

Mover aficonf de INSTDIR al directorio BINDIR.

Mostrar en consola el siguiente mensaje:

" Instalación del componente Aficonf completada"

16. Instalar AFIMONIO

Mover afimONIO de INSTDIR al directorio BINDIR.

Mostrar en consola el siguiente mensaje:

" Instalación del componente AfimONIO completada"

17. Guardar la información de Instalación

Almacenar la siguiente información en el archivo ainstal.conf en el directorio CONFDIR..

Línea 01: INSTDIR = \$INSTDIR

Línea 02: CONFDIR = \$CONFDIR

Si es antifraude

Línea 10: Comando ANTIFRAUDE.PL

Línea 11: BINDIR = \$BINDIR

Línea 12: LOGDIR = \$LOGDIR

Línea 13: LOGNAME = \$LOGNAME

Línea 14: LOGSIZE = \$LOGSIZE

Línea 15: DATADIR = \$DATADIR

Línea 16: DATASIZE =

Línea 17: USERID = Usuario de la instalación



Línea 18: FECINS = fecha de instalación

Si es aficonf

Línea 20: Comando AFICONF
Línea 21: BINDIR = \$BINDIR
Línea 22: LOGDIR = \$LOGDIR
Línea 23: LOGNAME = \$LOGNAME
Línea 24: LOGSIZE = \$LOGSIZE
Línea 25: USERID = Usuario de la instalación
Línea 26: FECINS = fecha de instalación

Si es afimonio

Línea 30: Comando AFIMONIO
Línea 31: BINDIR = \$BINDIR
Línea 32: LOGDIR = \$LOGDIR
Línea 33: LOGNAME = \$LOGNAME
Línea 34: LOGSIZE = \$LOGSIZE
Línea 35: USERID = Usuario de la instalación
Línea 36: FECINS = fecha de instalación

Linea51 y siguientes: disponibles para su uso por parte del usuario

18. Borrar archivos temporarios, si los hubiese generado

19. Ofrecer arrancar el demonio AFIMONIO (si se encuentran instalados todos los demás componentes del paquete)

Mostrar el siguiente mensaje

"Desea efectuar la activación del demonio antifraude?"

"Si desea arrancar el proceso de antifraude, oprima solamente ENTER, si desea cancelar el proceso oprima cualquier otra tecla y luego ENTER."

Si arranca el demonio, mostrar mensaje indicando que esta activado y bajo que numero de proceso.

Si el usuario no desea arrancar el demonio, entonces Mostrar el siguiente mensaje:

"Para realizar la activación utilice el comando afimonio de la siguiente forma: <describir como invocar el comando>"

Continuar.

20. Mostrar mensajes indicando que fue lo que se instaló.

Ejemplo:

```
*****
* Proceso de Instalación Concluido *
* Se encuentran instalados los siguientes componentes: *
* Antifraude.pl instalado el ... (fecha) por ... (usuario) *
* Aficonf instalado el ... por (usuario de instalación) *
* Afimonio instalado el (fecha de instalación) por ... *
* *
* Copyright TPSistemasOp (c)2004 *
*****
Presione ENTER para finalizar
```



AFIMONIO, Demonio de Llamadas

Descripción

El propósito del proceso AFIMONIO es:

1. Chequear la existencia de archivos en el directorio \$DATADIR/aprocesar
2. Si existen archivos, entonces, por cada archivo que se detecta
 - 2.1. Verificar si es un archivo de llamadas: Los archivos de llamadas son los que tienen un nombre con el siguiente formato: **aaaammddhhMM.<central telefónica(4 dígitos)>**
 - 2.2. Si el archivo es de llamadas, entonces
 - 2.2.1. Mover el archivo a \$DATADIR/enproceso
 - 2.2.2. Grabar en el archivo de log: Fecha y hora actual, Usuario, Nombre del comando y "Se detecta archivo de llamadas: <nombre del archivo>"
 - 2.2.3. Arrancar el comando antifraude.pl, siempre que el mismo no esté corriendo
 - 2.3. Si el archivo NO es de llamadas, entonces
 - 2.3.1. Mover el archivo a \$DATADIR/rechazadas
 - 2.3.2. Grabar en el archivo de log: Fecha y hora actual, Usuario, Nombre del comando y "Se rechaza el archivo: <nombre del archivo>"
3. Dormir un tiempo x
4. Volver al punto 1

Queda a consideración de cada grupo el valor que se asigna a la variable x.

Observaciones

Un demonio es un proceso que corre en forma permanente, en background.

Suele armarse como un conjunto de sentencias dentro de un ciclo eterno.

Se utiliza generalmente para:

- monitorear otros procesos
- ejecutar procesos rutinarios



AFICONF, Configurador de Umbrales

Archivos de Input

Archivo de Origenes/Destinos Posibles: **\$CONFDIR/oridesti.txt**

Ver estructura en: [Estructura de oridesti.txt](#)

Archivos de Output

Archivo de Umbrales de Origenes/Destinos Sospechoso: **\$CONFDIR/umbrales.param**

Ver estructura en: [Estructura de umbrales.param](#)

Descripción del Caso de Uso

Recordar lo indicado en el [punto 2](#) de "Observaciones a tener en cuenta en el desarrollo".

Nombre del Caso de Uso: AB Destino Sospechoso	
Actor: Usuario de Gestión Antifraude	
Objetivo: Funcionalidad a través del cual se da de Alta o Baja umbrales de alarma de destinos/origenes sospechosos	
Curso Normal	Alternativas
<p>1. El actor selecciona la opción "Nuevo".</p>	<p>1.1. El actor ingresa un Número de Línea y selecciona "Buscar".</p> <p>1.2. El sistema realiza la búsqueda exitosamente de todos los registros Activos que cumplen con la condición 1.1 y muestra una lista para su selección con:</p> <ul style="list-style-type: none"> Id de Registro Número de línea Origen/Destino Tipo de Llamada Usuario Fecha y Hora <p>1.3. El actor selecciona el Id que desea dar de baja y selecciona "Borrar".</p> <p>1.4. El sistema realiza exitosamente la baja lógica del registro modificando: Estado = Inactivo, Usuario = usuario logueado, Fecha = fecha actual, Hora = hora actual.</p> <p>1.5. El sistema muestra al usuario un mensaje amigable de acción exitosa. Fin del Caso de Uso</p> <p>1.2.1. La búsqueda resulta fallida</p> <p>1.2.2. El sistema muestra un mensaje de error amigable. Continúa en el punto 1.1</p> <p>1.3.1. El actor selecciona "Salir". Fin del Caso de Uso</p> <p>1.4.1. El borrado lógico resulta fallido</p> <p>1.4.2. El sistema muestra un mensaje de error amigable. Continúa en el punto 1.1</p>
<p>2. El actor ingresa:</p> <ul style="list-style-type: none"> • Número de línea (números) • Tipo de Llamada (Entrante o Saliente) • Origen/Destino (descripción) 	
<p>3. El sistema verifica que los datos son correctos:</p>	<p>3.1 La verificación resulta fallida</p>



<ul style="list-style-type: none">• Numero de Línea es solo numérico• Tipo de Llamada es solo "E" o "S"• Origen/Destino existe en oridesti.txt y el control es exitoso	3.2. El sistema muestra un mensaje de error amigable. Continúa en el punto 2.
4. El sistema controla que no exista un umbral de alarma idéntico al que se quiere ingresar y el control es exitoso.	4.1 El control resulta fallido 4.2. El sistema muestra un mensaje de error amigable informándole que ya existe un umbral de alarma idéntico para esa línea. Continúa en el punto 2.
5. El sistema "Graba" los datos ingresados, mas: Id de Registro = numero, Estado = Activado, Usuario = usuario logueado, Fecha = fecha actual, Hora = hora actual Y la grabación resulta OK.	5.1 La grabación resulta fallida 5.2. El sistema muestra un mensaje de error amigable. Continúa en el punto 2.
6. Fin Caso uso.	

Observaciones

Origen/Destino: Si la llamada es entrante, el valor que contiene este campo representa un Origen. Si la llamada es Saliente, el valor que contiene este campo representa un Destino. El contenido de este campo se debe validar contra el campo descripción del archivo de Orígenes/Destinos:
\$CONFDIR/oridesti.txt

Si el actor selecciona "Salir", en cualquier momento del Caso de uso, el mismo comienza nuevamente.

Los eventos que se enuncian (Nuevo, Buscar, Borrar, Salir) pueden ser reemplazados por combinaciones de teclas, números, etc. Aclarar que método se usa en "hipótesis planteadas".

Recordar lo indicado en el [punto 3](#) de "Observaciones a tener en cuenta en el desarrollo".



ANTIFRAUDE, Generador de alarmas

Archivos de Input

Archivo de Llamadas En Proceso: **\$DATADIR/enproceso/<nombre del archivo>**

Archivo de Umbrales de Orígenes/Destinos Sospechoso: **\$CONFDIR/umbrales.param**

Archivos de Output

Archivo de Log **\$LOGDIR/\$LOGNAME**

Archivo de Alarmas Generadas: **\$DATADIR/alarmas/alarmas.txt**

Ver Estructura en: [Estructura de alarmas.txt](#)

Archivo de Llamadas Procesadas: **\$DATADIR/procesadas/<nombre del archivo>**

Procesar Todos los archivos

Este comando tiene por objetivo procesar todos los archivos de llamadas que arriban al directorio \$DATADIR/enproceso. Cuando procesa todos los archivos termina.

Debe procesar los archivos del mas antiguo al mas reciente. El archivo mas antiguo es aquel que tiene en su nombre la fecha-hora mas antigua.

Procesar Un archivo

Procesar un archivo es procesar todos los registros que contiene ese archivo. Cuando se termina de procesar un archivo, se lo debe mover al directorio: \$DATADIR/procesadas

Procesar Un registro

Procesar un registro es:

1. Indicar por Consola que numero de registro se esta procesando y de que archivo

2. Validar sus campos, para ello verificar:

2.1. Que cada registro contenga exactamente 8 campos

2.2. Que el numero de línea sea solo numérico

2.3. Que si llamada es de tipo "E" venga informado el campo Origen

2.4. Que si la llamada es tipo "S" venga informado el campo Destino

3. Si el registro no corresponde al formato especificado, entonces:

3.1. Indicar por Consola que el registro es rechazado

3.2. Descartar el registro grabando en el Archivo de Log lo siguiente:

Línea 1: Fecha y hora actual, Usuario, Nombre del comando y Mensaje = "El siguiente registro de llamada tiene formato inválido"

Línea 2: Contenido COMPLETO del registro de llamada que se descarta.

3.3. Continuar con el registro siguiente desde el paso 1

4. Si el registro corresponde al formato especificado, entonces:

4.1. Buscar si existe un umbral de alarma para esa llamada (\$CONFDIR/umbrales.param)

4.1.1. Opción "A"



-
- 4.1.1.1. Llamada.Numero de línea = Umbral.Numero de Línea
 - 4.1.1.2. Llamada.Destino = Umbral.Origen/Destino
 - 4.1.1.3. Llamada.Tipo de Llamada = Umbral.Tipo de Llamada = S
 - 4.1.1.4. Umbral.Estado = A
 - 4.1.2. Opción "B"
 - 4.1.2.1. Llamada.Numero de línea = Umbral.Numero de Línea
 - 4.1.2.2. Llamada.Origen = Umbral.Origen/Destino
 - 4.1.2.3. Llamada.Tipo de Llamada = Umbral.Tipo de Llamada = E
 - 4.1.2.4. Umbral.Estado = A
 - 4.2. Si existe un umbral de alarma para esa llamada (Se cumple "A" o "B"), entonces
 - 4.2.1. Indicar por Consola el id de registro del archivo de umbrales cuya regla se cumple.**
 - 4.2.2. Grabar un registro de alarma en el archivo \$DATADIR/alarmas/alarmas.txt**
 - 4.2.3. Continuar con el registro siguiente desde el paso 1
 - 4.3. Si No existe un umbral de alarma para esa llamada, entonces
 - 4.3.1. Continuar con el registro siguiente desde el paso 1

Requerimientos NO Funcionales

Por cuestiones de performance, se requiere cargar en alguna estructura en memoria TODOS los registros de umbrales.

Observaciones

Cuando se dice "Indicar por Consola", recordar lo indicado en el [punto 4.2](#) de "Observaciones a tener en cuenta en el desarrollo".



Archivos

Todos los archivos tienen como separador de campo el carácter “;” punto y coma.

Archivo de Log

Directorio

\$LOGDIR/\$LOGNAME

Estructura

Línea	N Caracteres
-------	--------------

Archivo de Instalación

Directorio

\$CONFDIR/afinstal.conf

Estructura

Línea	N Caracteres
-------	--------------

Ejemplo para el comando antifraude:

```

INSTDIR = /grupo08/antifraude/inst
CONFDIR = /grupo08/antifraude/conf
...
Comando ANTIFRAUDE.PL
BINDIR = /grupo08/antifraude/bin
LOGDIR = /grupo08/antifraude/log
LOGNAME = antifraude.log
LOGSIZE = 20 Mb
DATADIR = /grupo08/antifraude/llamadas
DATASIZE =
USERID = Grupo08
FECINS = 27/12/2004
  
```

Archivo de Orígenes/Destinos

Directorio

\$CONFDIR/oridesti.txt

Estructura de oridesti.txt

Código	2 caracteres
Descripción	N caracteres

Ejemplo

```

AF; AFRICA
AN; AMERICA NORTE
AP; ASIA PACIFICO
EC; EUROPA CENTRAL
EE; EUROPA DEL ESTE
EO; EUROPA OCCIDENTAL
LA; LATINOAMÉRICA
MO; MEDIO ORIENTE
  
```



Archivo de Umbrales de Orígenes/Destinos Sospechosos

Directorio

\$CONFDIR/umbrales.param

Estructura de umbrales.param

Id del registro	N caracteres, Numero secuencial ≥ 1 que identifica unívocamente a un registro de umbral. Se incrementa en uno
Número de Línea	N caracteres, Numérico
Origen/Destino	2 caracteres, valores posibles según archivo oridesti.txt
Tipo de Llamada	1 carácter, valores posibles: E = Entrante, S = Saliente
Estado	1 carácter, valores posibles: A = Activado, I = Inactivo
Usuario	N caracteres, siempre = login del usuario que graba el reg.
Fecha	8 caracteres, formato aaaammdd, siempre es la fecha actual
Hora	6 caracteres, formato hhMMdd, siempre es la hora actual

Archivos de Llamadas

Directorios

- Archivo de Llamadas A Procesar: \$DATADIR/aprocesar/<nombre del archivo>
- Archivo de Llamadas En Proceso: \$DATADIR/enproceso/aaaammddhhMM.DDDD, dónde DDDD es el código de la central telefónica, 4 caracteres numéricos
- Archivo de Llamadas Procesadas: \$DATADIR/procesadas/<nombre del archivo>
- Archivos Rechazados: \$DATADIR/rechazadas/<nombre del archivo>

Estructura de Llamadas

Número de Línea	N caracteres, Numérico
Cantidad de Minutos	N caracteres, Numérico
Fecha de Llamada	8 caracteres, formato aaaammdd
Hora de Llamada	6 caracteres, formato hhMMdd
Tipo de Llamada	1 carácter, valores posibles: E = Entrante, S = Saliente
Destino	2 caracteres, viene informado solo en las llamadas Salientes
Origen	2 caracteres, viene informado solo en las llamadas Entrantes
Prestadora	N caracteres



Archivo de Alarmas

Directorio

\$DATADIR/alarmas/alarmas.txt

Estructura de alarmas.txt

Prestadora	N caracteres, se obtiene a partir del "Prestadora" del archivo de llamadas
Central Telefónica	4 caracteres, se obtiene a partir del nombre del archivo de llamadas
Número de Línea	N caracteres, se obtiene a partir del "Numero de Linea" del archivo de llamadas
Cantidad de Minutos	N caracteres, se obtiene a partir del "Cantidad de Minutos" del archivo de llamadas
Regla aplicada	N caracteres, se obtiene a partir del "Id del registro" de Umbrales.param
Fecha de Llamada	8 caracteres, se obtiene a partir del "Fecha de llamada" del archivo de llamadas
Hora de Llamada	6 caracteres, se obtiene a partir del "Hora de llamada" del archivo de llamadas
Usuario	N caracteres, siempre es el login del usuario que graba el registro
Fecha	8 caracteres, formato aaaammdd, siempre es la fecha actual
Hora	6 caracteres, formato hhMMdd, siempre es la hora actual