

**SOLUCIÓN PARA EL MEJORAMIENTO DEL SERVICIO DE CORREO ELECTRÓNICO
EN LA RED DE DATOS DE LA UNIVERSIDAD DEL CAUCA.**

Trabajo de desarrollo

**DAVID FERNANDO ANDRADE SOLANO
VICTOR ANDRES CASTRO DUEÑAS**



**ANEXO G
ROBUSTEZ DE CONTRASEÑAS**

Director: Guefry Agredo Méndez M.Sc.

**UNIVERSIDAD DEL CAUCA
FACULTAD DE INGENIERÍA ELECTRÓNICA Y TELECOMUNICACIONES
DEPARTAMENTO DE TELECOMUNICACIONES
Grupo I+D Nuevas Tecnologías en Telecomunicaciones - GNTT
POPAYÁN, 2012**

TABLA DE CONTENIDO

Introducción	1
1. Módulo de robustez de contraseñas	1
2. Generador automático de contraseñas	3
REFERENCIAS BIBLIOGRÁFICAS	6

ANEXO G

ROBUSTEZ DE CONTRASEÑAS

Introducción

Para proteger la información de autenticación de cada usuario, se hace necesario crear una contraseña robusta para que esta no sea fácil de interpretar por parte de terceros que quieran utilizarla para realizar operaciones inapropiadas como el envío de *spam*.

En este anexo se detalla un código realizado en **Javascript** para la verificación de robustez de contraseñas y otro para generar contraseñas seguras de forma automática.

1. Módulo de robustez de contraseñas

A continuación se evidencia el script [1], que permite realizar la verificación de robustez de contraseñas:

```
function calcular_seguridad(clave,formulario){
    Contador_seg=passwd_seg(clave);
}

function passwd_seg(clave){
    var seg = 0;
    if (clave.length!=0){
        if (numeros_pwd(clave) && letras_pwd(clave)){
            seg += 10;
        }
        if (letras_pwd(clave) && mayusculas_pwd(clave)){
            seg += 10;
        }
        if (CaracEspeciales_pwd(clave) && letras_pwd(clave)){
            seg += 10;
        }
        if (CaracEspeciales_pwd(clave) && mayusculas_pwd(clave)){
            seg += 10;
        }
        if (CaracEspeciales_pwd(clave) && numeros_pwd(clave)){
            seg += 10;
        }
        if (numeros_pwd(clave) && numeros_pwd(clave)){
            seg += 10;
        }
    }

    if (clave.length >= 4 && clave.length <= 5){
        seg += 10;
    }else{
        if (clave.length >= 6 && clave.length <= 8){
```

```

        seg += 30;
    }else{
        if (clave.length > 8){
            seg += 40;
        }
    }
}
}
return seg
}

var numeros="0123456789";

function numeros_pwd(texto){
    for(i=0; i<texto.length; i++){
        if (numeros.indexOf(texto.charAt(i),0)!=-1){
            return 1;
        }
    }
    return 0;
}

var letras="abcdefghijklmnopqrstuvwxyz";
function letras_pwd(texto){
    texto = texto.toLowerCase();
    for(i=0; i<texto.length; i++){
        if (letras.indexOf(texto.charAt(i),0)!=-1){
            return 1;
        }
    }
    return 0;
}

var letras_mayusculas="ABCDEFGHIJKLMNOPQRSTUVWXYZ";
function mayusculas_pwd(texto){
    for(i=0; i<texto.length; i++){
        if (letras_mayusculas.indexOf(texto.charAt(i),0)!=-1){
            return 1;
        }
    }
    return 0;
}

var letras_especiales="!#$%&/()=?;:'<>|°~^`{}[]@";
function CaracEspeciales_pwd(texto){
    for (i=0; i<texto.length; i++){
        if (letras_especiales.indexOf(texto.charAt(i),0)!=-1){
            return 1;
        }
    }
    return 0;
}

```

Para la realización de este código fue necesaria la utilización de tres funciones básicas de **Javascript**, estas funciones se describen a continuación:

- **Length**: Esta función calcula la longitud de una cadena de texto.
- **CharAt (posicion)**: Con esta función se obtiene el carácter que se encuentra en la posición indicada.
- **indexOf (Caracter)**: Esta función calcula la posición en la que se encuentra el carácter indicado dentro de la cadena de texto. Si el carácter se incluye varias veces dentro de la cadena, se devuelve a la primera posición partiendo de izquierda a derecha, si la cadena no contiene el carácter, la función devuelve el valor de "-1".

Inicialmente, la función `calcular_seguridad`, es la que recibe el password proveniente de un formulario de HTML, esta función se encarga de pasar dicha palabra a la función `passwd_seg` que es la encargada de asignar los valores numéricos a la variable dependiendo de la combinación y número de caracteres que esta palabra tenga para realizar este trabajo de asignación de valores. La función `passwd_seg` necesita de las funciones `numeros_pwd`, `letras_pwd`, `mayusculas_pwd`, `CaracEspeciales_pwd`, estas son las encargadas de verificar que la contraseña contenga cada uno de los caracteres necesarios para que sea robusta.

Ahora se describe como es el formulario HTML, el cual hace la llamada a la función de `calcular_seguridad` de **Javascript**.

```
<form>
<input type="password" size=15 name="clave"
onkeyup="calcular_seguridad(this.value, this.form)">
</form>
```

El tipo del formulario es `password`, esto permite que lo que se escriba en el será observado con formas de asterisco (*). En la parte `onkeyup`, es donde se hace el llamado a la función `calcular_seguridad`, este llamado se realiza cada vez que se oprima una tecla para ingresar un carácter, el comando `this` indica que se está trabajando sobre ese formulario y que se envíen los valores que este contiene.

2. Generador automático de contraseñas

Para generar las contraseñas aleatorias se utiliza el siguiente código [2]:

```
var caracter="1234567890";
caracter+="QWERTYUIOPASDFGHJKLÑZXCVCBNM";
caracter+="qwertyuiopñlkjhçgfdasazxcvbnm";
caracter+="!#$%&/()=?;ıç'<>|°~^`{}[]@";
```

```

var total=caracter.length;

function azar(form){
var numero_caracteres= form.LongitudCampo.value;
var clave="";
if(form.LongitudCampo.value>=8){

for(a=0;a<numero_caracteres;a++){
clave+=caracter.charAt(parseInt(total*Math.random(1)));
}
form.clave2.value=clave;
}
}
}

```

Las funciones de **Javascript** utilizadas para realizar este código son las siguientes:

- **Random**: Esta función genera un número aleatorio comprendido entre 0 y 1 se le puede pasar un parámetro numérico el cual indica que al número aleatorio obtenido se le sume ese valor.
- **Length**: Esta función calcula la longitud de una cadena de texto.
- **CharAt**: Obtiene un carácter que se encuentre en una posición indicada.
- **ParseInt**: Devuelve un número entero de un valor decimal.

Inicialmente se crea un formulario en el cual el usuario podrá especificar la cantidad de caracteres con que desea que la contraseña sea creada, el formulario es el siguiente:

```

<form>
<label>Longitud password:</label>
<input type="text" name="LongitudCampo" style="text-align:center"
size="18" value="8">
<label>Clave:</label>
<input type="text" readonly="readonly" size="18" style="text-
align:center" name="clave2" id="clave2" /><br>
<input type="button" value="Generar Password" style="text-
align:center" onclick="azar(this.form)" />
</form>

```

La llamada a la función que genera el password aleatorio se realiza a través del formulario tipo "button" llamado "generar password".

Para este programa es necesaria la utilización de la función **Random** que pertenece a la clase **math** de **JavaScript**, esta función genera un número aleatorio que esta comprendido entre 0 y 1. También se define una variable llamada "caracter", esta tendrá definida todas las letras minúsculas, letras mayúsculas, números y caracteres

especiales necesarios para formar la contraseña, a esta función se le pasa como parámetro el valor numérico "1". Con la función `Length` se obtiene la longitud de la variable "character" y se guarda en una variable llamada total esta se multiplica por el resultado que bote la función `Random` y después a ese valor se le aplica la función `ParseInt`, al realizar esta acción se esta generando un número aleatorio cuyo rango es hasta la longitud máxima de la variable `character`, se utiliza la función `charAt` para obtener la letra respectiva de la variable "character" utilizando el número aleatorio creado, este proceso se realiza la cantidad de veces que sea necesaria para formar la contraseña con el número de caracteres que el usuario definió en el formulario.

REFERENCIAS BIBLIOGRÁFICAS

- [1] «desarrolloweb.com,» 2008. [En línea]. Available: <http://www.desarrolloweb.com/articulos/script-informar-seguridad-clave-javascript.html>.
- [2] [En línea]. Available: <http://programadorasp.com/generar-password-y-contrasenas-aleatorias-con-javascript/>.