



## Seguridad Informática

# Programación con Perl

Cervantes Varela Juan Manuel  
Vallejo Fernández Rafael Alejandro

# Perl en Windows (Strawberry Perl)

Perl no viene instalado por defecto en Windows y hay dos opciones disponibles en la página oficial de Perl.

Para este curso utilizaremos **Strawberry Perl** debido a que está diseñado para ser lo más cercano a un ambiente de perl de un sistema UNIX.

Está disponible para instalarse o ejecutarse en modo portable.

# Perl en Windows (Strawberry Perl)

Para descargarlo debemos visitar el sitio oficial:

<http://strawberryperl.com/>

**Nota:** Bajar la **versión 5.30.3.1** ya que utilizaremos otro programa adicional que por el momento no es compatible con la última versión de Strawberry Perl 5.32.0.1.

**Versión instalable:**

<http://strawberryperl.com/download/5.30.3.1/strawberry-perl-5.30.3.1-64bit.msi>

**Versión portable:**

<http://strawberryperl.com/download/5.30.3.1/strawberry-perl-5.30.3.1-64bit-portable.zip>

# Perl en Windows (Perl2Exe)

El otro programa que utilizaremos es Perl2Exe que permite convertir scripts de Perl en archivos ejecutables para sistemas Windows o Unix.

Este programa es comercial y tiene una versión gratuita que podemos bajar desde la página oficial.

<https://www.indigostar.com/perl2exe/>

La última versión disponible es la **30.10** y es compatible con versiones de Perl 5.30.x, y es la razón por la que bajamos Strawberry Perl 5.30.3.1.

**Para descargar Perl2Exe (no requerirá instalación):**

<https://www.indigostar.com/download/perl2exe-30.10-win.zip>

# Perl en Windows

Si se está utilizando la versión portable de Strawberry Perl ejecutar **portableshell.bat** que nos abrirá una terminal que puede ejecutar los scripts de Perl.

Para ejecutarlos simplemente se debe hacer:

```
perl "path/to/script/script.pl"
```

# Perl en Windows

Para generar el ejecutable podemos hacerlo desde la misma terminal que nos abrió Strawberry Perl, solamente debemos buscar Perl2Exe y ejecutarlo mediante la línea de comandos.

```
perl2exe-30.10\perl2exe path\to\script.pl
```

Esto nos generará en el directorio actual el ejecutable:

```
script.exe
```

# Ejercicio

- **Ejercicio 19**

Crear un script en Perl y su ejecutable en Windows que obtenga datos sobre el sistema operativo, por ejemplo:

Nombre de usuario, directorio del usuario, dirección IP del equipo (No Vmnet##), RAM, CPU, etcétera.

**Nota:** Mínimo 6 datos sobre el sistema

# Función pack y unpack

Son dos funciones para transformar datos de acuerdo con una plantilla definida por el usuario.

- Pack

Empaqueta en una sola cadena binaria, un conjunto de valores con el formato de la plantilla proporcionada.

- Unpack

Descomprime una cadena binaria o bloque de bytes usando el formato especificado en una plantilla, devolviendo una lista de los valores desempaquetados.

```
1  pack PLANTILLA, VALOR_1, ... , VALOR_N
```

```
1  unpack PLANTILLA, CADENA_BYTES
```



# Funcion pack y unpack

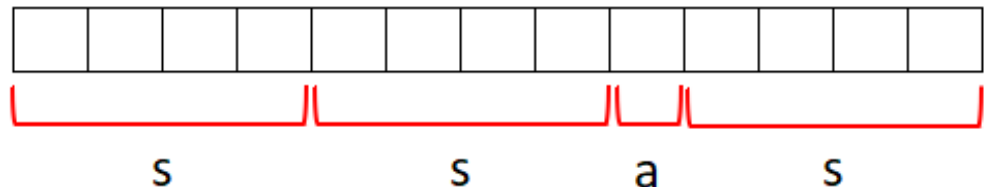
```
1 my $ps = pack( 'ssas', 1, 20302, 65, 1024);
2 print $ps, " ", length($pa), "\n";      # □NOA□
```

\$ps =



```
1 my ($val1, $val2, $val3, $val4) = unpack "ssas", $ps;
2 print "$val1, $val2, $val3, $val4\n";    # 1, 20302, A, 1024
```

my (\$var1, \$var2, \$var3, \$var4) =



# Funcion pack y unpack - plantillas

La plantilla se define con una serie de caracteres que definen el formato del valor y el tamaño de bytes requeridos, a continuación se muestran algunos y su significado:

```
1  a  ASCII character string padded with null characters
2  c  A signed character (range usually -128 to 127)
3  d  A double-precision floating-point number
4  f  A single-precision floating-point number
5  i  A signed integer
6  I  An unsigned integer
7  l  A signed long integer
8  L  An unsigned long integer
9  s  A signed short (16-bit) value.
10 S  An unsigned short value.
11 x  A null byte
12
13 !  Forces native (short, long, int) sizes instead of fixed (16-/32-bit) sizes.
```

El numero de caracteres de la plantilla, debe coincidir con el numero de valores a empaquetar/dempaquetar.

# Keylogger

Ej. Keylogger

## Práctica 10

Programar un keylogger en perl.

Debe recibir como argumento el nombre de un archivo, en el cual se irán guardando todas las teclas presionadas mientras se esté ejecutando el keylogger.