

Vulnerabilidades en funciones

Insegura	Segura
<p><code>Strcpy (char *destination, const char *source)</code></p> <p>Copia la cadena en source a destination.</p>	<p><code>strncpy(char *destination, const char *source, size_t size);</code></p> <p>Copia la cadena en source a destination revisando siempre el tamaño para evitar overflow.</p>
<ul style="list-style-type: none"> • <code>printf(const char *format, ...)</code> • <code>sprintf (char *str, const char *format, ...)</code> <p>Imprime el contenido que se le pasa como argumento tomando en cuenta el formato dado.</p>	<ul style="list-style-type: none"> • <code>snprintf(char *str, size_t size, const char * restrict format, ...)</code> • <code>asprintf(char **ret, const char *format, ...)</code> <p>Ambas son versiones mejoradas de printf y sprintf, sin embargo, la primera toma en cuenta el tamaño de la cadena para evitar overflow y la segunda toma en cuenta el reservar la memoria también para evitar esto último.</p>
<p><code>strcmp(const char *s1, const char *s2)</code></p> <p>Regresa 0 cuando las cadenas son iguales y cualquier otro de lo contrario (negativo si $s1 < s2$ y positivo si $s1 > s2$)</p>	<p><code>strncmp (const char * s1, const char * s2, size_t num)</code></p> <p>Hace lo mismo que strcmp, pero no compara más del número que se le pasa como argumento.</p>
<p><code>strcat(char *dest, const char *src)</code></p> <p>Concatena la segunda cadena al final de la primera.</p>	<p><code>strncat(char *destination, const char *source, size_t size);</code></p> <p>De la misma forma que strcpy, solo concatena el número de bytes que se le pasa como argumento.</p>
<p><code>vsprintf(char *str, const char *format, va_list arg)</code></p> <p>Le da un formato a una cadena usando una lista de parámetros</p>	<p><code>vsnprintf(char *str, size_t size, const char *format, va_list ap);</code></p> <p>Hace lo mismo que vsprintf, pero toma en cuenta solo la cadena con el tamaño que se le pasa como argumento</p>

Referencias:

https://www.owasp.org/index.php/Reviewing_Code_for_Buffer_Overruns_and_Overflows

<https://linux.die.net/man>