COSTO DE LOS ERRORES ENSEGURIDAD DEL SOFTWARE

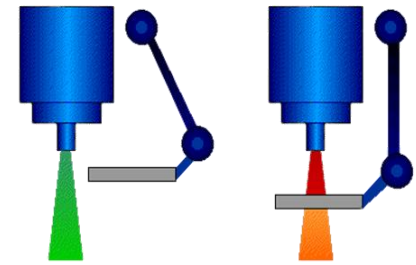
Según el departamento de estado de los Estados Unidos

- Los errores de software son tan comunes y graves que cuestan alrededor de $59.5 billones de dólares anualmente o el 0.6% del producto interno bruto.
- Más de la tercera parte de este costo, $22.2 billones, se pueden eliminar con una infraestructura de pruebas que permita la identificación y remoción de los defectos de software más rápida y efectivamente.
- Estos ahorros se asocian con encontrar los errores más cerca de las etapas de desarrollo en que se introdujeron. Normalmente, se encuentran muy al final del proceso o después de la venta.

CASOS DE ERRORES FAMOSOS

THERAC-25

- Era una máquina empleada en terapia de radiación, producida por Atomic Energy of Canada Limited.
- Causó al menos seis accidentes entre 1985 y 1987, y le costó la vida al menos a cinco personas.
- El problema estaba en la interfaz gráfica que permitía proporcionar dosis de radiaciones mortales a los pacientes.
- El software no detectaba la rotación generada en los electrodos y no notaba que el paciente estaba recibiendo una dosis de raciación letal.



RAZONES

- El diseño no incluía ningún bloqueo del hardware para prevenir que se llegara a ese alto nivel de energía sin que estuviera todo en posición.
- Se reutilizó el software de otros modelos que sí tenían el bloqueo anterior y no eran vulnerables a este problema.
- El hardware no proveía al software un modo para verificar que todo estuviera funcionando correctamente.

- La tarea de control del equipo no se sincronizaba con la del operador. Las condiciones del problema se dieron cuando el operador cambiaba la configuración muy rápido. Esto no se daba durante las pruebas porque apenas estaban aprendiendo.

PATRIOT MISSILE SYSTEM

- En Febrero 25, 1991, el misil patriot estuvo en operación por 100 horas. En este momento el reloj interno se había movido un tercio de segundo.
- Para un objetivo que se movía tan rápido el error en posición era de 600 mt.
- El sistema de radar detectó el misil tipo Scud y predijo dónde iba a estar. Por el error de reloj, el sistema buscó el misil y no lo encontró en el momento por lo que asumió que había sido un error de detección.
- Dado que no se interceptó el misil este impacto y murieron 28 militares.

MARS POLAR LANDER

- La Mars Polar Lander (MPL) era una sonda espacial estadounidense lanzada por un cohete Delta II 7425 el 3 de enero de 1999, que llegó a Marte el 3 de diciembre de 1999 tras un viaje de nueve meses.
- La sonda llegó en buen estado a Marte.
- El 3 de diciembre de 1999, diez minutos antes de aterrizar, se perdió el contacto.
- La causa de la pérdida de comunicación se desconoce.
- Una posible causa es que durante el descenso la apertura de las patas de aterrizaje era tan brusca que podría haber activado los sensores que indicaban que se había tocado suelo, responsables de parar el motor. Así, la sonda habría parado su motor en pleno vuelo, estrellándose fatalmente contra el suelo.

ARIANE 5 ROCKET

- El Ariane 5 es un cohete de un sólo uso diseñado para colocar satélites en órbita geoestacionaria y para enviar cargas a órbitas bajas.

- En Junio 4 de 1996 fue el vuelo de prueba para el sistema de lanzamiento del Ariane 5.
- El cohete se destruyó 37 segundos después del lanzamiento convirtiendo la falla en uno de los errores de software más costosos de la historia.
- El software reutilizó especificaciones del Ariane 4. Pero la ruta era muy distinta y fuera del rango para el que se diseñó el anterior software.
- Específicamente, el Ariane 5 tenía 5 veces más aceleración y esto causó que los computadores fallaran.
- Las pruebas no se realizaron sobre las condiciones de vuelo del Ariane 5.
- Por la diferencia de ruta se ocasionó un error de conversión lo que llevó a una cascada de problemas culminando en la destrucción del vuelo.

2003 APAGÓN

- Se encontró que FirstEnergy no tomó ninguna acción o alertó a otros centros de control hasta que era demasiado tarde.
- Un error de software del sistema de administración de energía de General Electric prevenía que las alarmas se mostraran en el sistema de control.
- Todas las alarmas y eventos se represaron y el servidor principal falló en 30 minutos. El servidor secundario también falló por la misma razón y todas las aplicaciones dejaron de funcionar.

Ciudades afectadas	
Ciudades	**Personas afectadas**
New York City and surrounding areas	14,300,000
Greater Toronto Area (Golden Horseshoe)	8,100,000
Newark, New Jersey and surrounding counties and suburbs	6,980,000
Detroit and Surrounding Areas	5,400,000
Cleveland and Greater Cleveland	2,900,000
Ottawa	780,000 of 1,120,000*
Buffalo and Surrounding Areas	1,100,000
Rochester	1,050,000
Baltimore and Surrounding Counties	710,000
London, ON and Surrounding Areas	475,000
Toledo	310,000
Windsor	208,000
Estimated Total[120]	55,000,000

TALLER PRÁCTICO DEL SEGURIDAD DEL SOFTWARE

1. Mencione algunos ejemplos (3) positivos y negativos que indiquen el impacto del software en la sociedad actual.

2. A medida que la presencia del software se vuelve más generalizada, los riesgos al público (debido a las fallas en los programas) representan una preocupación significativa y creciente. Desarrollar un escenario catastrófico realista en el que la falla de un programa de computadora podría producir un gran daño (ya sea económico o humano). EXPLIQUE.

3. ¿Qué haría usted para reducir el deterioro del software?

4. Mencione algunas posibles fallas del hardware y posibles soluciones para evitar estas fallas (3)

5. Cree usted que una vez que el programa (software) ha sido terminado y puesto a funcionar EL TRABAJO ESTÁ TERMINADO. Si – No. ¿Porqué? EXPLIQUE

6. El desarrollo de software se ve constantemente impedido por la lentitud en la creación de componentes hardware y mecanismos que servirán para que extienda su potencial. (está de acuerdo: ¿si – no) PORQUÉ?