

BOEING 737 MAX TRIBUNA

¿Qué causó la tragedia del Boeing 737 MAX de Ethiopian Airlines?

Todo indica que un sensor envió información errónea al software

PAQUI LUCIO

23 ABR 2019 - 12:30 CEST



Imagen de un Boeing 737 MAX 8, en la planta de Boeing en Renton, Washington , en 2015. **MATT MCKNIGHT (REUTERS)**

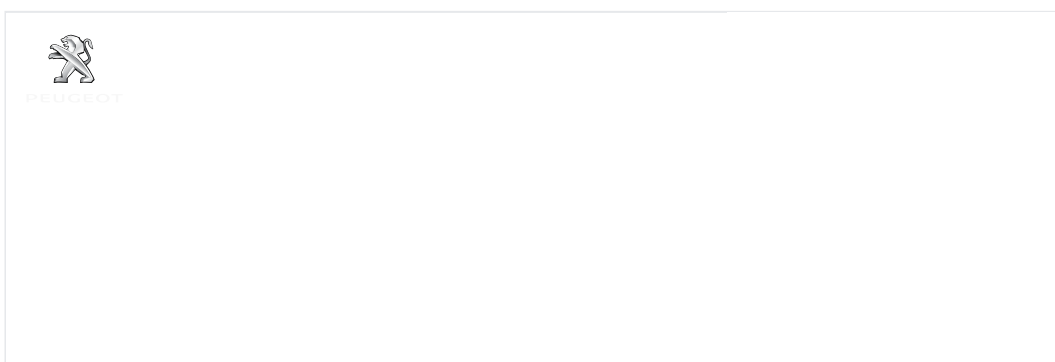
Según un informe oficial del pasado 4 de abril, no se ha detectado ningún fallo humano entre las causas del [accidente de aviación que, el pasado 10 de marzo, se cobró 157 vidas, al caer en picado a los pocos minutos de despegar](#). El informe detalla que los pilotos trataron de desactivar el software conocido como MCAS (siglas en inglés de "Sistema de Aumento de las Características de Maniobra") hasta en cuatro ocasiones, sin conseguirlo. El [modelo 737 MAX](#) fue diseñado para mejorar la potencia y eficiencia del Boeing 737. Los nuevos motores son más pesados que sus predecesores, y fueron

colocados en distinta posición, lo que modificó el centro de gravedad. El software MCAS se diseñó para que actuara automáticamente cuando un sensor de "ángulo de ataque" detectase que la inclinación del morro es muy alta, forzando a inclinarlo hacia abajo. Todo indica que un sensor envió información errónea al software MCAS. Aunque el fallo de un sensor puede considerarse un fallo técnico, el software que actúa en base a la información que le transmiten dispositivos externos, como sensores, debe tener mecanismos de recuperación de fallos de dichos dispositivos o, como mínimo, permitir su propia desactivación. Por tanto, todo indica que lo que se produjo fue un fatal error de software que impidió la desactivación del MCAS.

El software MCAS del B737MAX es altamente crítico y, como tal, se le debería exigir un elevado nivel de fiabilidad

Nuestro quehacer diario depende cada día más del software, y dicho software es cada día más complejo. Cada vez es más común que las televisiones, frigoríficos, calderas, aspiradores, coches, trenes, etc.-- tengan numerosos sistemas software empotrados, que les permitan funcionar autónomamente. En unos casos esta dependencia puede resultar crítica: un error de software puede causar daños irreparables a personas, bienes o al medio ambiente. Se suele hablar de software cuya seguridad o fiabilidad es crítica o, simplemente, de software crítico. Por ejemplo, es crítico el software utilizado para controlar el funcionamiento de cualquier tipo de medio de transporte (aéreo o terrestre), dispositivos médicos, transacciones bancarias y financieras, etc. Del software no-crítico los usuarios requerimos cierto grado de disponibilidad y fiabilidad que a menudo es difícil de conseguir, lo que, aunque no cause una tragedia, puede resultar bastante molesto. Para mejorar nuestra vida cotidiana, y sobre todo hacerla más segura, es cada vez más importante que cualquier software, y más aún si es crítico, cuente con ciertas garantías de que funciona correctamente.

PUBLICIDAD





Días sobre ruedas Peugeot.

Probar un Peugeot tiene más ventajas que nunca. Lo quiero.

Inspired by  invibes

En el caso de que el software dependa de un entorno regulado por sensores u otros dispositivos, debería garantizarse una reacción segura y controlada, incluso a cambios inesperados del entorno. De hecho, existen estándares que deben ser aplicados en los procesos de desarrollo de software para que la calidad de los sistemas resultantes pueda ser certificada por las entidades oficiales competentes. Los estándares utilizan diferentes definiciones de niveles de criticidad, de acuerdo a los cuales establecen los procedimientos y requisitos a satisfacer. Obviamente, cuanto más crítico es un software, más estrictos son los requisitos de certificación.

Tardaremos en saber por qué el Boeing 737 MAX volaba usando software defectuoso y si se aplicaron o no los estándares requeridos en su desarrollo

La normativa correspondiente depende del área de aplicación: aeronáutica (DO-178B), sistemas ferroviarios (UNE-EN 50128), automoción (UNE-EN 26262-6), aplicaciones espaciales (NASA-GB-8719.8), etcétera. Para niveles altos de criticidad, los distintos estándares requieren (parafraseando el DO-178B) "el uso de técnicas basadas en Matemáticas, con la intención de que se puedan llevar a cabo los adecuados análisis matemáticos que puedan contribuir a establecer la corrección del software diseñado." Es verdad que cualquiera, sin conocimientos matemáticos, puede aprender a programar y desarrollar aplicaciones que realicen ciertas funciones, pero es igualmente cierto que, sin fundamentos y técnicas matemáticas, no se pueden llevar a cabo los análisis necesarios para garantizar la fiabilidad del software desarrollado.

El software MCAS del B737MAX es altamente crítico y, como tal, se le debería exigir un elevado nivel de fiabilidad. Esta tragedia, [junto con otra \(de Lion Air\) de similares características ocurrida seis meses antes](#), han puesto en entredicho la reputación en

seguridad no solo de la compañía aeronáutica Boeing, sino también de la Administración Federal de Aviación (FAA) estadounidense, encargada de certificar el software de control a bordo de los aviones. El pasado 3 de abril los responsables de Boeing anunciaron que la actualización del MCAS recibirá datos de dos sensores, en lugar de uno y, si al compararlos hay un desacuerdo entre ellos de más de 5,5 grados, no se activará y alertará a los pilotos a través de un indicador en el panel. Está suficientemente claro que el software MCAS fue el causante de la tragedia, pero tardaremos en saber por qué el Boeing 737 MAX volaba usando software defectuoso y si se aplicaron o no los estándares requeridos en su desarrollo. Se están investigando varias demandas contra la FAA por negligencia en la certificación del B737MAX.

Paqui Lucio es profesora titular del Departamento de Lenguajes y Sistemas Informáticos. Universidad del País Vasco.

Crónicas del Intangible es un espacio de divulgación sobre las ciencias de la computación, coordinado por la sociedad académica SISTEDES (Sociedad de Ingeniería de Software y de Tecnologías de Desarrollo de Software). El intangible es la parte no material de los sistemas informáticos (es decir, el *software*), y aquí se relatan su historia y su devenir. Los autores son profesores de las universidades españolas, coordinados por Ricardo Peña Marí (catedrático de la Universidad Complutense de Madrid) y Macario Polo Usaola (profesor titular de la Universidad de Castilla-La Mancha).

Se adhiere a los criterios de

[Más información >](#)

ARCHIVADO EN:

Boeing-McDonnell Douglas · Aviones · Transporte aéreo · Software · Empresas · Informática · Economía · Transporte · Industria

CONTENIDO PATROCINADO

Ecoplan Citroën. Nuevo Berlingo desde 16.690€. Aprovecha la oferta

CITROËN

¿Qué autonomía tiene un coche eléctrico?

KIA ES

El precio de las alarmas de hogar en Lasarte no es el que crees

ALARMAS HOGAR - ENLACES PATROCINADOS

Y ADEMÁS...

Revuelo en Twitter con el aviso sobre el DNI que ha hecho la Policía

HUFFINGTON POST

Khabib sobre McGregor y su agresión a un anciano: "El dinero le..."

TIKITAKAS

David Bisbal enloquece Twitter al interpretar el himno de España...

EPIK

recomendado por

NEWSLETTER

Recibe la mejor información en tu bandeja de entrada

