

INFORME ACCELERATE

State of DevOps 2022



Patrocinado por



Índice

01	Resumen ejecutivo	03	05	Sorpresas	55
02	Comparación con otras empresas	08	06	Datos demográficos y organizativos	59
03	¿Cómo lo puedes mejorar?		07	Conceptos finales	67
	Introducción	19	08	Agradecimientos	68
	Cloud	21	09	Autores	69
	SRE y DevOps	26	10	Metodología	73
	Capacidades técnicas de DevOps	29	11	Lecturas adicionales	76
	Cultura	37			
04	Por qué es importante la seguridad de la cadena de suministro	42			

01

Resumen ejecutivo



Derek DeBellis



Claire Peters

Durante los últimos ocho años, llevamos elaborando el informe Accelerate State of DevOps con la colaboración de 33,000 profesionales durante el proceso. Nuestra investigación se centra en examinar cómo las capacidades y las prácticas predicen los resultados que consideramos fundamentales para DevOps:

- Rendimiento de la entrega de software: las cuatro métricas clave del rendimiento de la entrega de software, que son la frecuencia de implementación, el plazo de espera para los cambios, la tasa de errores de cambio y el tiempo de restablecimiento del servicio.
- Rendimiento operativo: la quinta métrica clave, la confiabilidad.
- Rendimiento organizativo: en qué medida tu organización cumple los objetivos de rendimiento y rentabilidad.

También nos enfocamos en los factores que subyacen a otros resultados, como el agotamiento y la probabilidad de que los empleados recomienden a sus equipos.



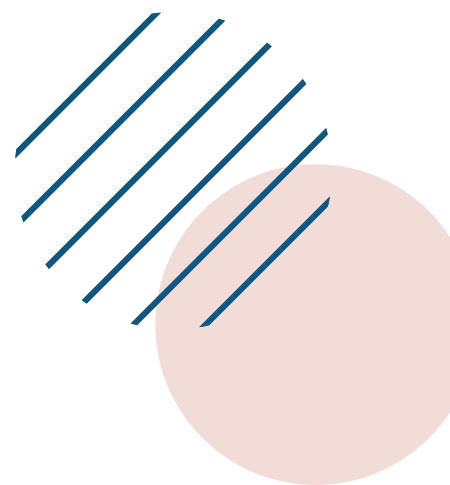
Protege la cadena de suministro de software

En 2021, nos dimos cuenta de que la seguridad de la cadena de suministro de software es esencial para alcanzar muchos resultados importantes.

Este año profundizamos en la seguridad de la cadena de suministro de software, por eso la convertimos en un punto principal del informe y la encuesta. Aprovechamos el framework de [Supply Chain Levels for Secure Artifacts \(SLSA\)](#) a fin de explorar las prácticas técnicas que apoyan el desarrollo de la seguridad de la cadena de suministro de software. También usamos el [Framework de desarrollo de seguridad de software \(SSDF, Secure Software Development Framework\)](#) del Instituto Nacional de Normas y Tecnología (NIST, National Institute for Standards and Technology) para explorar las actitudes, los procesos y las prácticas no técnicas relacionadas con la seguridad de la cadena de suministro de software.

Llegamos a la conclusión de que el mayor predictor de las prácticas de seguridad en el desarrollo de aplicaciones de una organización era cultural, no técnico. Las culturas con alta confianza, baja culpabilización y centradas en el rendimiento tenían una probabilidad 1.6 veces mayor de adoptar prácticas de seguridad emergentes, en comparación con las culturas con baja confianza, alta culpabilización y centradas en el poder o las reglas. También encontramos pruebas que sugieren que el análisis de seguridad previo a la implementación es eficaz para encontrar dependencias vulnerables, lo que da como resultado menos vulnerabilidades en el código de producción.

La adopción de buenas prácticas de seguridad en el desarrollo de aplicaciones se correlacionó con beneficios adicionales. Notamos que los equipos que se enfocan en establecer estas prácticas de seguridad redujeron el agotamiento de los desarrolladores. Los equipos con niveles bajos de prácticas de seguridad tienen una probabilidad 1.4 veces mayor de tener niveles altos de agotamiento que los equipos con niveles altos de seguridad.¹ Los equipos que se enfocan en establecer prácticas de seguridad son más propensos a recomendar su equipo a otra persona. Además, las prácticas de seguridad relacionadas con SLSA predicen de forma positiva el rendimiento organizativo y el rendimiento de la entrega de software, pero para que tengan plena efectividad se necesita una sólida capacidad de implementación continua.



¹ En esta estadística, consideramos “alto” a una desviación estándar ≥ 1 en la puntuación (por ejemplo, seguridad) y “bajo” a una desviación estándar ≤ -1 en la puntuación.



Factores que impulsan el rendimiento organizativo

Además de las prácticas de seguridad mencionadas con anterioridad, las variables clave que influyen en el rendimiento organizativo tienden a entrar en las siguientes categorías:

Cultura de la organización y del equipo

Según la definición de [Westrum](#), las culturas de alta confianza y baja culpabilización tienden a tener un mayor rendimiento organizativo. Del mismo modo, las organizaciones con equipos que se sienten apoyadas a través de la financiación y el patrocinio del liderazgo tienden a obtener un mayor rendimiento organizativo. Lo mismo ocurre cuando el equipo es estable y los integrantes lo perciben de manera positiva (probabilidad de recomendarlo). Por último, las empresas que ofrecen acuerdos de trabajo flexibles suelen mostrar altos niveles de rendimiento organizativo.

Confiabilidad

Las prácticas que asociamos con la ingeniería de confiabilidad (p. ej., objetivos de confiabilidad claros, métricas de confiabilidad destacadas, etc.) y el grado en que las personas informan que cumplen con sus expectativas de confiabilidad son predictores poderosos de altos niveles de rendimiento organizativo.

Cloud

Descubrimos que el uso de la nube predice el rendimiento organizativo. Las empresas cuyo software se compiló primero en la nube y para ella suelen tener un mayor rendimiento organizativo. El uso de nubes privadas, públicas, híbridas o una combinación de estas se corresponde con un mayor rendimiento organizativo que solo el uso de servidores locales. Quienes usan varias nubes públicas tienen 1.4 veces más probabilidades de tener un rendimiento organizativo por encima de la media que quienes no lo hacen.

El uso de la nube también parece tener un impacto en el rendimiento organizativo a través de otros factores en nuestro conjunto de datos. Un ejemplo es la seguridad de la cadena de suministro, en la que descubrimos que las organizaciones que usaban nubes públicas también eran más propensas a implementar prácticas de SLSA. Esto se debe, quizás, a que los proveedores de servicios en la nube fomentan y proporcionan componentes básicos para muchas prácticas de SLSA, como la automatización de las compilaciones y las implementaciones.^{2,3} El punto es que el uso de plataformas en la nube permite a un equipo heredar muchas capacidades y prácticas que con el tiempo se traducen en un mayor rendimiento organizativo.

²Jung, Sun Jae. "Introduction to Mediation Analysis and Examples of Its Application to Real-world Data." *Journal of preventive medicine and public health = Yebang Uihakhoe chi* vol. 54,3 (2021): 166-172. doi:10.3961/jpmph.21.069


³ Carrión, Gabriel Cepeda, Christian Nitzl, y José L. Roldán. "Mediation analyses in partial least squares structural equation modeling: Guidelines and empirical examples." *Partial least squares path modeling*. Springer, Cham, 2017. 173-195.

El contexto es importante

Por mucho tiempo, el DORA tuvo en cuenta que los efectos dependen del contexto más amplio del equipo. Creemos que es importante comprender las características de un equipo (los procesos, los puntos fuertes, las limitaciones y los objetivos) y el entorno en el que se desarrolla el trabajo. Por ejemplo, una capacidad técnica que resulta beneficiosa en un contexto puede ser perjudicial en otro. Este año nos enfocamos en elaborar modelos explícitos de estas condiciones hipotéticas en forma de interacciones, muchas de estas hipótesis están respaldadas por los datos de este año:

- El alto rendimiento de la entrega de software solo es beneficioso para el rendimiento organizativo cuando el rendimiento operativo también es alto. La rapidez en la entrega tal vez no importe si tu servicio no puede cumplir con las expectativas de confiabilidad de los usuarios.
- Implementar controles de seguridad en la cadena de suministro de software, como los recomendados por el framework SLSA, tiene un efecto positivo en el rendimiento de la entrega de software si la integración continua está bien establecida. Si no se dispone de capacidades de integración continua, el rendimiento de la entrega de software y los controles de seguridad podrían entrar en conflicto.
- El impacto de las prácticas de ingeniería de confiabilidad de sitios (SRE) en la capacidad de un equipo para alcanzar los objetivos de confiabilidad no es lineal. La práctica de la SRE no afecta de forma positiva la confiabilidad hasta que un equipo logra un determinado nivel de madurez de la SRE. Antes de que un equipo alcance este nivel de madurez de la SRE, no se detecta una relación entre la SRE y el alcance de los objetivos de confiabilidad. Sin embargo, a medida que crece la adopción de la SRE de un equipo, se llega a un punto de inflexión en el que el uso de la SRE comienza a predecir una confiabilidad sólida. La mejora de la confiabilidad repercute en el rendimiento organizativo.
- Las capacidades técnicas se apoyan unas en otras. La entrega continua y el control de la versión se amplían entre sí para promover altos niveles de rendimiento en la entrega de software. La combinación de la entrega continua, la arquitectura con bajo acoplamiento, el control de la versión y la integración continua fomentan el rendimiento de la entrega de software que es mayor que la suma de sus partes.





Las condiciones de las que depende la entrega, y la necesidad de comprender el contexto más amplio de un equipo, nos llevan a una conclusión similar a esta idea de 2021:

“Para realizar mejoras significativas, los equipos deben adoptar una filosofía de mejora continua. Usa las comparativas a fin de medir tu estado actual, identificar restricciones en función de las capacidades que estudia la investigación y experimentar con mejoras para aliviar dichas restricciones. En la experimentación habrá una combinación de victorias y fracasos, pero en ambas situaciones los equipos pueden tomar medidas importantes como resultado de las lecciones aprendidas”.

En realidad, este año encontramos un efecto que está muy alineado con esta filosofía general: los equipos que reconocen la necesidad de mejorar de forma continua tienden a tener un mayor rendimiento organizativo que los que no lo hacen.

En resumen, los equipos deben adaptarse siempre y experimentar con las prácticas de desarrollo de software.

Lo sabemos porque, en general, los equipos que lo hacen tienen un mayor rendimiento organizativo. Esto no es siempre así (lo que funciona para una organización no tiene por qué funcionar en otra), pero sí la mayoría de las veces. A medida que experimentes con las prácticas de DevOps, no te preocupes por los fracasos ocasionales mientras perfeccionas lo que funciona para tu equipo.

Este año también descubrimos una gran cantidad de [sorpresas](#) en los datos, pero tienes que seguir leyendo para saber cuáles son.

Los equipos que reconocen la necesidad de mejorar de forma continua tienden a tener un mayor rendimiento organizativo que los que no lo hacen.

02

Comparación con otras empresas



Derek DeBellis

¿Te interesa saber cómo se compara tu equipo con otros del sector? En esta sección, se incluye la evaluación comparativa más reciente del rendimiento de DevOps. Examinamos cómo los equipos desarrollan, entregan y operan los sistemas de software y, a continuación, segmentamos a los encuestados en grupos que recogen las combinaciones más comunes de rendimiento de DevOps.

Este año incluimos dos enfoques de agrupamiento en clústeres que son diferentes. El primero se basa en los antecedentes históricos. Este enfoque de agrupamiento en clústeres se enfoca en la creación de clústeres basados en cuatro métricas que capturan el rendimiento de la entrega de software: el plazo de entrega, la frecuencia de implementación, el tiempo para restablecer el servicio y la tasa de errores de cambio. A continuación, resumimos cada una de ellas. El objetivo de este enfoque es ayudarte a cuantificar el rendimiento actual de tu equipo para que puedas compararlo con el de otros equipos.

El segundo enfoque de agrupamiento en clústeres incluye una quinta métrica, la confiabilidad, que usamos para comprender el rendimiento operativo. ¿Por qué agregar una nueva métrica al análisis de clústeres? Porque siempre advertimos la importancia de esta métrica. En realidad, tenemos pruebas que sugieren que el rendimiento de la entrega puede ser perjudicial para el rendimiento organizativo si no se acompaña de un sólido rendimiento operativo. A diferencia de nuestro enfoque tradicional de agrupamiento, este es un ejercicio descriptivo que intenta describir una imagen de las formas comunes de rendimiento de los equipos en cuanto a la entrega y el rendimiento operativo. Por lo tanto, no siempre es obvio qué clúster es mejor.

En primer lugar, veamos una breve descripción general de las cinco medidas que usamos para entender la entrega de software y el rendimiento operativo.

Entrega de software y rendimiento operativo

Para satisfacer las demandas de los sectores en constante cambio, las organizaciones deben entregar y emplear el software con rapidez y confianza. Cuanto más rápido los equipos puedan realizar cambios en el software, más rápido podrás generar valor para tus clientes, realizar experimentos y recibir comentarios valiosos. Tras ocho años de recopilación de datos y de investigación, desarrollamos y validamos cuatro métricas que miden el rendimiento de la entrega de software. Desde 2018, incluimos una quinta métrica para capturar las capacidades operativas.

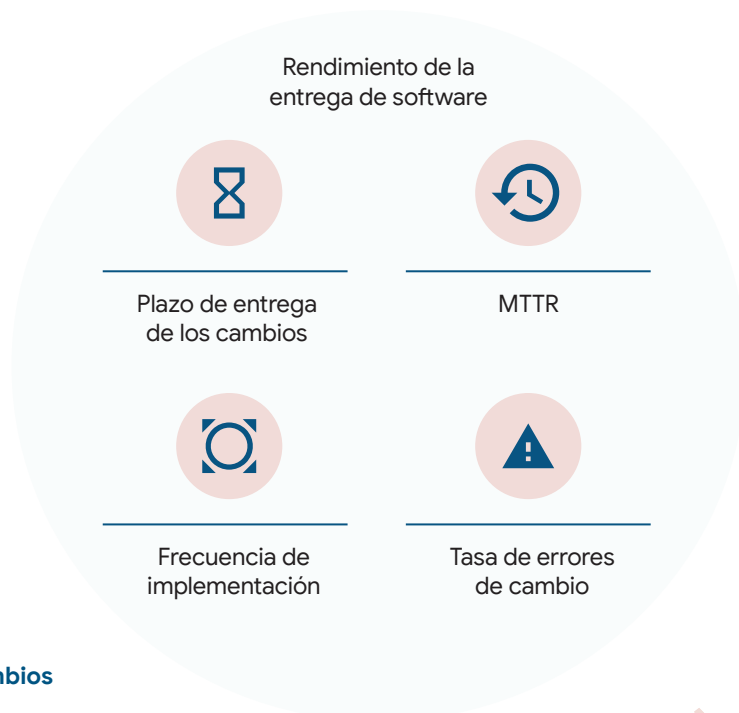
Los equipos que se destacan en las cinco mediciones tienen un rendimiento organizativo extraordinario. Llamamos a estas cinco medidas **rendimiento de la entrega de software y operativo (software delivery and operational, SDO)**. Ten en cuenta que estas métricas se enfocan en los resultados a nivel de sistema, lo que ayuda a evitar los problemas comunes de seguimiento de las métricas de software, que pueden provocar que las funciones se enfrenten entre sí y que se realicen optimizaciones locales a costa de los resultados generales.

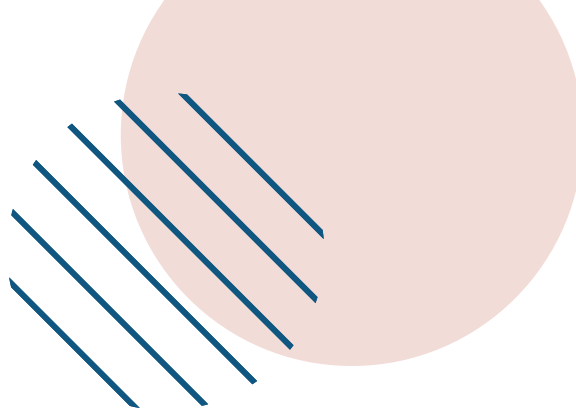


Cinco métricas de entrega y rendimiento operativo

Se pueden considerar cuatro métricas de rendimiento de la entrega de software en términos de capacidad de procesamiento y estabilidad. Medimos la capacidad de procesamiento mediante el **plazo de entrega de los cambios de código** (es decir, el tiempo desde la confirmación del código hasta el lanzamiento en producción) y la **frecuencia de implementación**. Medimos la estabilidad con el **tiempo de restablecimiento del servicio** después de un incidente y la **tasa de errores de cambio**.

La quinta métrica representa el **rendimiento operativo** y es una medida de las prácticas operativas modernas. Basamos el **rendimiento operativo** en la **confiabilidad**, que es la forma en que tus servicios cumplen con las expectativas de los usuarios, como la disponibilidad y el rendimiento. Por lo general, medíamos la disponibilidad en lugar de la confiabilidad, pero debido a que la disponibilidad es un enfoque específico de la ingeniería de la confiabilidad, incluimos la confiabilidad en 2021 para que la disponibilidad, la latencia, el rendimiento y la escalabilidad tuvieran una representación más amplia. En concreto, le pedimos a los encuestados que calificaran su capacidad para cumplir o superar sus objetivos de confiabilidad. Descubrimos que los equipos con diferentes grados de rendimiento de entrega obtienen mejores resultados (por ejemplo, menos agotamiento) cuando también priorizan el rendimiento operativo.





Enfoque histórico del agrupamiento en clústeres: rendimiento de la entrega del agrupamiento en clústeres

Este año, cuando se evaluó la solución de los cuatro clústeres que usamos desde 2018, nos dimos cuenta de que los datos mostraban un clúster de bajo rendimiento y otro con un alto rendimiento. Sin embargo, los dos clústeres que antes usábamos a fin de delimitar el rendimiento medio y el alto no se diferenciaban lo suficiente como para garantizar una división. Además, los distintos índices que usamos para seleccionar la solución de clústeres adecuada sugerían

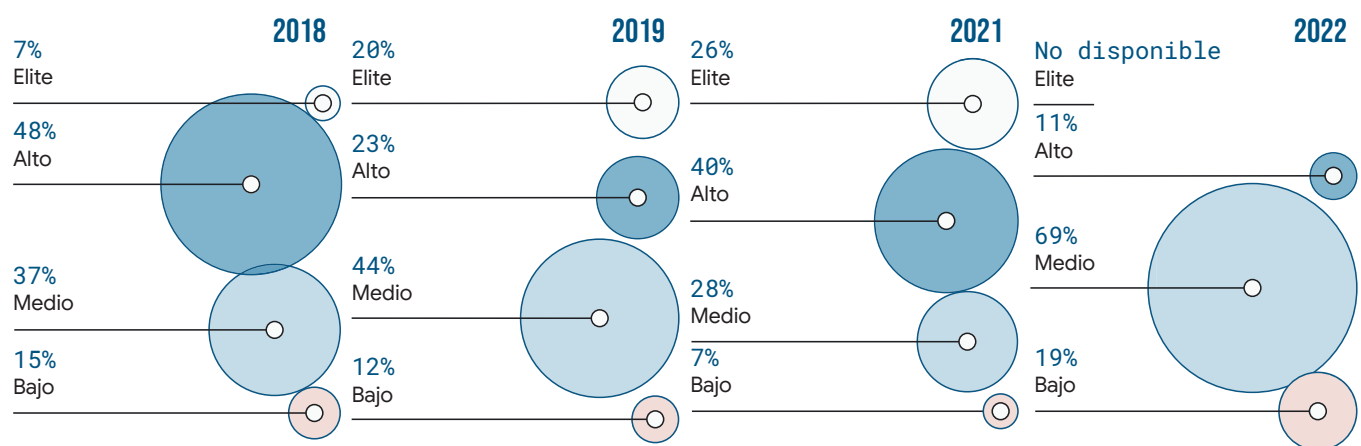
en todo momento que los tres clústeres eran los que mejor capturaban los datos, sin importar las técnicas aplicadas de agrupamiento en clústeres. En la siguiente tabla, se describen las características de rendimiento de las entregas de cada clúster.

La diferencia más llamativa con relación al del año pasado es que este año no consideramos que ningún clúster sea de élite. El clúster alto de este año es una combinación de los clústeres altos y de élite del año pasado. Decidimos omitir un clúster de élite, ya que el clúster de rendimiento más alto no indica suficientes características del clúster de élite del año pasado.

Métrica de rendimiento de la entrega de software	Baja	Media	Alta
Frecuencia de implementación En la aplicación o el servicio principal en el que trabajas, ¿con qué frecuencia tu organización implementa código en la producción o lo lanza para los usuarios finales?	Entre una vez al mes y una vez cada 6 meses	Entre una vez por semana y una vez por mes	A pedido (varias implementaciones por día)
Plazo de entrega de los cambios En la aplicación o servicio principal en el que trabajas, ¿cuál es tu plazo de entrega de los cambios (es decir, cuánto tiempo se necesita desde la confirmación del código hasta la ejecución con éxito en la producción)?	Entre un mes y seis meses	Entre una semana y un mes	Entre un día y una semana
Tiempo de restablecimiento del servicio En la aplicación o servicio principal en el que trabajas, ¿cuánto tiempo se necesita por lo general para restablecer el servicio cuando ocurre un incidente o un defecto que afecta a los usuarios (p. ej., una interrupción no planificada o una inhabilitación del servicio)?	Entre una semana y un mes	Entre un día y una semana	Menos de un día
Tasa de errores de cambio En la aplicación o servicio principal en el que trabajas, ¿qué porcentaje de cambios en la producción o en el lanzamiento a los usuarios genera un servicio degradado (p. ej., provocan una interrupción en el servicio o lo inhabilitan) y, en consecuencia, necesita una corrección (p. ej., que se requiera aplicar un parche rápido, realizar reversión, implementar una corrección o aplicar un parche)?	46%-60%	16%-30%	0%-15%

Sugiere que esta muestra no representa a los equipos o las organizaciones cuyos empleados sienten que avanzaron. Una hipótesis posible, que en este momento no podemos fundamentar con datos, es que el desarrollo de software registró una reducción de la innovación en términos de prácticas, intercambio de información y herramientas. Esto podría ser el resultado de la actual pandemia que obstaculiza la capacidad de compartir conocimientos y prácticas entre equipos y organizaciones. Es posible que hubiera menos oportunidades de reunirse y aprender unos de otros, lo que, a su vez, pudo haber frenado la innovación. Esperamos investigar con más profundidad para analizar las causas de este hallazgo.

Dicho esto, si se comparan los clústeres bajos, medios y altos de este año con los del año pasado, se verá que hay un cambio hacia un rendimiento de entrega un poco más alto. Parece que los clústeres de este año están entre dos de los del año pasado. El clúster alto de 2022 está entre el alto de 2021 y el de élite. El clúster bajo de 2022 parece estar entre el bajo y el medio de 2021. El incremento del clúster de bajo rendimiento sugiere que, si bien se redujo el límite superior del rendimiento de las entregas, se elevó el límite inferior.

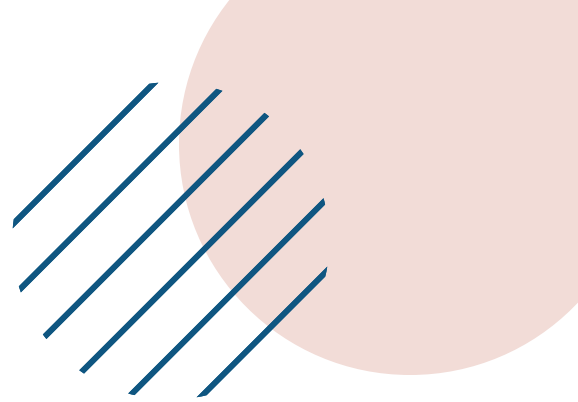


Los desgloses de porcentajes del cuadro anterior muestran que el porcentaje de personas con alto rendimiento se encuentra en el nivel más bajo de los últimos 4 años, mientras que el porcentaje de personas con bajo rendimiento aumentó de forma drástica: del 7% en 2021 al 19% este año. Más de dos tercios de los encuestados de este año pertenecen al clúster medio. La evidente caída en los equipos con alto rendimiento y de élite podría sugerir que muchos de los encuestados de este año se encuentran en organizaciones o equipos que no establecieron o están en proceso de establecer una cultura de DevOps que vemos surgir en muchos equipos modernos.

Puede que nos enfoquemos demasiado en las diferencias entre 2021 y 2022, en lugar de resaltar las similitudes. Los clústeres de 2021 y 2022 comparten numerosas características, incluida una gran separación entre los de alto rendimiento y los de bajo rendimiento. Por ejemplo, se estima que las personas con alto rendimiento¹ tienen 417 más implementaciones que los de bajo rendimiento.

¹ Consulta la sección "Metodología" para conocer lo que hay detrás de esta estimación





Agrupamiento en clústeres del rendimiento de la entrega y el rendimiento operativo

Decidimos hacer un análisis del clúster en las tres categorías que representan las cinco métricas: **capacidad de procesamiento** (un compuesto de plazo de entrega de cambios de código y frecuencia de implementación), **estabilidad** (tiempo para restablecer un servicio y cambiar la tasa de errores) y **rendimiento operativo** (confiabilidad). La razón es el rol fundamental que tiene el rendimiento operativo en nuestros modelos.

Para las organizaciones que no muestran un rendimiento operativo sólido, la capacidad de procesamiento y la estabilidad tienen menos impacto en el rendimiento de la organización. Creemos que describir el ámbito del rendimiento de DevOps sin tener en cuenta el operativo excluye una parte fundamental del proceso.

Explorar los datos nos llevó a una solución de cuatro clústeres. A continuación, se desglosan los cuatro clústeres y sus nombres:

Clúster	Estabilidad		Rendimiento operativo	Capacidad de procesamiento		Porcentaje de encuestados
	Tiempo de restablecimiento del servicio	Tasa de errores de cambio	Confiabilidad	Tiempo de entrega	Frecuencia de implementación	
Iniciándose	Entre un día y una semana	31%-45%	A veces se cumplen las expectativas	Entre una semana y un mes	Entre una vez por semana y una vez por mes	28%
En flujo	Menos de una hora	0%-15%	Por lo general, cumplen las expectativas	Menos de un día	A pedido (varias implementaciones por día)	17%
Lento	Menos de un día	0%-15%	Por lo general, cumplen las expectativas	Entre una semana y un mes	Entre una vez por semana y una vez por mes	34%
Retirar	Entre un mes y seis meses	46%-60%	Por lo general, cumplen las expectativas	Entre un mes y seis meses	Entre una vez al mes y una vez cada 6 meses	21%

Sin embargo, si se visitaran dos equipos del mismo clúster, lo más probable es que serían muy diferentes, y nuestra historia podría no captar lo que se ve. La historia que presentamos para cada clúster es un intento de aprovechar nuestra experiencia de trabajo con varios equipos a fin de hacer inteligibles estos patrones en los datos. Además, si visitaste el mismo equipo en dos momentos diferentes, puede que no hayan permanecido en el mismo clúster. Una de las razones posibles podría ser que el equipo haya mejorado o se haya degradado. Otra posibilidad es que el equipo haya pasado a un patrón de implementación más apropiado para el estado actual de su aplicación o servicio. Por ejemplo, en un punto anterior del desarrollo de una aplicación o servicio, un equipo podría haberse enfocado en la exploración (clúster inicial), pero a medida que empiezan a encontrar su nicho, pueden cambiar su enfoque hacia la confiabilidad (clúster en flujo o clúster lento).

Cada clúster tiene características únicas y representa una proporción sustancial de las respuestas.

El clúster **inicial** no obtiene buenos ni malos resultados en ninguna de nuestras dimensiones. Tal vez está en las primeras etapas de desarrollo del producto, función o servicio. Puede que se centren menos en la confiabilidad porque se concentran en obtener comentarios, comprender la adaptación de los productos al mercado y, en general, en explorar.

El clúster **en flujo** funciona bien en todas las características: alta confiabilidad, alta estabilidad y alta capacidad de procesamiento. Solo el 17% de los encuestados se encuentra en este estado de flujo.

Los encuestados del clúster **lento** no se implementan con demasiada frecuencia, pero cuando lo hacen, es probable que lo hagan bien. Más de un tercio de las respuestas corresponden a este clúster, por lo que esta muestra es la más amplia y representativa. Este patrón es probable que sea típico (aunque lejos de ser exclusivo) de un equipo que mejora de forma gradual, pero ellos y sus clientes están en su mayoría contentos con el estado actual de su aplicación o producto.

Y, por último, el clúster de **retiro** parece un equipo que trabaja en un servicio o aplicación que sigue siendo valioso para ellos y sus clientes, pero que ya no está en desarrollo activo.

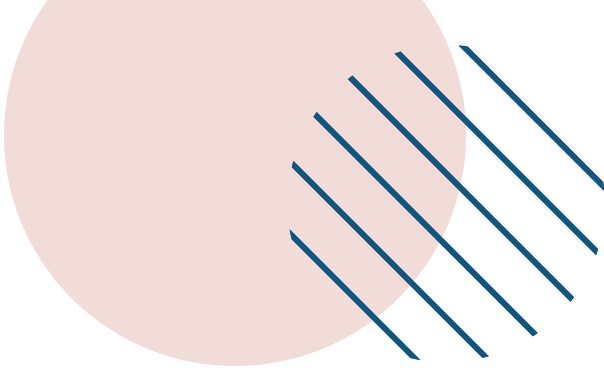
Estos clústeres difieren mucho en sus prácticas y capacidades técnicas. Dados los elevados niveles de entrega de software y rendimiento operativo que demuestra el clúster **en flujo** decidimos examinar en qué se diferencian de los demás, en cuanto a sus prácticas y capacidades técnicas. Descubrimos que, en relación con otros, el clúster **en flujo** se centra más en los siguientes aspectos:

- Arquitecturas con acoplamiento bajo: se refiere a la capacidad de los equipos a fin de realizar cambios a gran escala en el diseño de su sistema sin depender de otros equipos para realizar esos cambios.
- Proporcionar flexibilidad: se refiere a la flexibilidad de la empresa en cuanto a la organización del trabajo de los empleados.
- Control de versión: se refiere a cómo se gestionan los cambios en el código de la aplicación, la configuración del sistema, la configuración de la aplicación, etcétera.
- Integración continua (CI): se refiere a la frecuencia con la que las ramas se integran en el enlace troncal.
- Entrega continua (CD): se refiere a las capacidades enfocadas en lograr que los cambios entren en producción de forma segura, sostenible y eficiente.

Como dato curioso, el clúster **en flujo** tiende a centrarse menos en la documentación. El año pasado comprobamos que las prácticas de documentación son fundamentales para el rendimiento de la entrega y el rendimiento operativo (SDO). ¿Cómo es que el clúster **en flujo** tiene un sólido rendimiento de SDO sin un fuerte enfoque en la documentación? Por una parte, hay muchas formas de conseguir un buen rendimiento del SDO fuera de la documentación. Además, tal vez el clúster **en flujo** esté refactorizando de forma continua su código para crear un proceso más autodocumentado y, por lo tanto, tenga menos necesidad de documentos como los que describimos en la encuesta.

El clúster **lento**, que representa la mayor proporción de nuestros encuestados, suele estar formado por encuestados de organizaciones más grandes que tienden a ser menos nativos de la nube que otros clústeres. Imagina una empresa muy desarrollada con algunos procesos sólidos que, a fin de cuentas, sigue proporcionando a los usuarios finales experiencias estables y confiables (y valiosas). Este clúster muestra una cultura generativa orientada al rendimiento.² Una de las características más interesantes del clúster lento es que tiene una baja capacidad de procesamiento y una alta cultura de trabajo positiva (una cultura de trabajo “generativa”, según Westrum). Se trata de una combinación poco común. Es más habitual ver que la capacidad de procesamiento y la cultura del trabajo sean proporcionales (alta/alta o baja/baja). Esperamos realizar más investigaciones para comprender mejor la relación entre la capacidad de procesamiento y la cultura.

² (Westrum)



También analizamos cómo se comparan estos diferentes clústeres en tres resultados: el agotamiento, el rendimiento de la organización y el trabajo no planificado. Lo que descubrimos superó nuestras expectativas. El clúster de **retiro** superó a los demás en cuanto a rendimiento organizativo. Si se observan las características de este clúster (estabilidad y capacidad de procesamiento deficientes), esto parece no coincidir con la mayoría de los resultados anteriores de DORA. Pero en lugar de culpar al azar por crear una anomalía (muy probable), queremos explorar algunas explicaciones posibles.

Cuando se trata de desentrañar estos hallazgos, hay que tener en cuenta un resultado complementario importante. El clúster de **retiro** logra un gran rendimiento organizativo a un gran costo: sus equipos tienen las tasas más altas de agotamiento, parecen ser más propensos a cometer errores y están cargados con mayor cantidad de trabajo no planificado. En forma conjunta, estos resultados sugieren que la confiabilidad puede ser suficiente para lograr un alto rendimiento organizativo, pero sin velocidad y estabilidad, tu equipo pagará el costo del agotamiento y el trabajo no planificado.

Tenemos otras hipótesis para explicar por qué el clúster de **retiro** tiene un mayor rendimiento organizativo que el resto de los clústeres, en particular el clúster **en flujo**. A continuación, una lista rápida.

- Existen características que subyacen a estos cuatro clústeres que pueden no estar en nuestros datos. Por ejemplo, el tamaño de la organización puede ser un buen indicador de madurez. El clúster **en flujo** tiende a ser de empresas más pequeñas, lo que puede indicar que sus productos están en etapas más formativas.
- Los miembros del clúster **en flujo** tienden a ser de empresas más pequeñas, que pueden estar menos vinculadas a las infraestructuras y procesos históricos y, en consecuencia, cuentan con procesos DevOps más sofisticados. Dicho esto, los datos muestran que el tamaño de una organización está correlacionado de forma positiva con el rendimiento de esta por razones que pueden no estar relacionadas con la tecnología.
- El clúster **en flujo** solía ignorar los principios descritos en la cultura generativa de Westrum. Con frecuencia, se ha observado que esto va en detrimento del rendimiento de la organización.
- Las organizaciones de cada clúster pueden variar en cuanto a lo que entienden por expectativas de confiabilidad y cómo las supervisan. Lo mismo ocurre con la forma en que definen sus objetivos de rendimiento organizativo.

- El clúster de **retiro** puede tener un alto rendimiento organizativo a corto plazo, pero nos preguntamos cómo será su rendimiento a largo plazo. ¿El agotamiento provocará la rotación de personal? ¿Podrán escalar sus procesos?
- Nos hacemos estas preguntas en diferentes niveles. Las capacidades técnicas (por ejemplo, la arquitectura con acoplamiento bajo) se plantean a nivel de equipo. Las preguntas sobre el rendimiento organizativo se plantean a nivel de la organización. Las organizaciones suelen tener numerosos equipos, y es posible que el encuestado reconozca que, aunque su organización funciona bien, el equipo en el que trabaja no lo hace.

Además, el clúster **en flujo** obtiene la segunda puntuación más alta en rendimiento organizativo, por detrás del clúster de **retiro** y presenta algunos de los niveles más bajos de agotamiento y trabajo no planificado. Según demuestran los clústeres **en flujo** y **lento**, la filosofía de DevOps es más eficaz cuando existe confiabilidad.

Nos entusiasma seguir explorando nuevas formas de describir las variaciones del sector. A partir de ahora, queremos seguir incluyendo el rendimiento operativo como una dimensión relevante para comprender estas variaciones. También queremos evitar los clústeres altamente prescriptivos y evaluativos (p. ej., el de Elite), y concentrarnos en cambio en el ejercicio descriptivo de identificar constelaciones comunes de rendimiento de SDO.



03

¿Cómo lo puedes mejorar?



Derek DeBellis

¿Cómo mejorar en una diversidad de resultados?

El informe State of DevOps está destinado a proporcionar una orientación basada en pruebas a fin de ayudar a tu equipo a enfocarse en las prácticas y capacidades de DevOps que obtienen los resultados que te interesan. Este año ampliamos nuestra investigación sobre la seguridad y el conjunto de resultados que desean los equipos. Antes, nos enfocábamos en la entrega de software, el rendimiento operativo (SDO) y los resultados de este. Seguimos haciéndolo, pero también queríamos explorar el agotamiento, la probabilidad de recomendar el equipo, el trabajo no planificado y la tendencia a cometer errores, no solo como medio para mejorar el rendimiento de SDO y de la organización, sino también como fines en sí mismos. Por consiguiente, este año nos aseguramos de destacar las prácticas y capacidades que parecen influir en estos resultados.





El modelo de investigación cambió este año a fin de reflejar mejor una teoría DORA subyacente: no existe un enfoque único para DevOps. En la práctica, comprobamos que hacer recomendaciones implica comprender el contexto más amplio de un equipo. Una práctica que es beneficiosa para un equipo puede resultar negativa para otro. Por ejemplo, hace tiempo se planteó la hipótesis de que las capacidades técnicas (como la arquitectura con acoplamiento bajo, el desarrollo basado en troncos, el control de versión y la integración continua) tienen un impacto positivo más acentuado en el rendimiento de la entrega de software cuando se aplica la entrega continua. Este año modelamos de forma explícita esta y otras interacciones. El objetivo es mejorar nuestra comprensión, desde el simple “¿qué tiene un efecto sobre qué?” hasta incluir “¿bajo qué condiciones estos efectos existen, se amplifican o se atenúan?” Comprender toda esta condicionalidad resultó ser una tarea complicada y emocionante, pero nos entusiasma compartir con ustedes algunos de estos primeros hallazgos.

Puedes encontrar los modelos de investigación de este año y de años anteriores en línea en [nuestro sitio web](#).

Más allá de las cuatro claves

¿Cómo mejoran las métricas de DORA el rendimiento del desarrollo y las operaciones? Un equipo multifuncional de ingenieros de software de Liberty Mutual Insurance revisa con regularidad el rendimiento mediante las “[cuatro métricas clave](#)” de DORA. Por ejemplo, Jenna Dailey, facilitadora de proyectos sénior en Liberty Mutual, nos contó que un equipo aprovechó la investigación de DORA para ayudar al equipo a identificar un cuello de botella, pasar a un enfoque de desarrollo basado en pruebas y lograr mejoras en su rendimiento general.

Obtén más información sobre el enfoque de Liberty Mutual a la hora de aprovechar los datos y las métricas de DORA para mejorar la calidad y la entrega del software en sus recientes [Tomorrow Talks](#).



Eric Maxwell

Cloud

Basándose en el impulso que observamos en los últimos años, el uso de la computación en la nube sigue acelerándose. En realidad, el porcentaje de personas que informan el uso de la nube pública, incluidas las nubes múltiples, es ahora del 76%, frente al 56% de 2021. La cantidad de personas que afirman no usar la nube en absoluto, incluidas las que no usan la nube privada, se redujo a solo un 10.5%, en comparación con un 21% el año pasado. El uso de múltiples nubes públicas aumentó de un 21% a un 26%, y el uso de la nube híbrida sube un 25% hasta el 42.5%. También observamos un pequeño aumento en el uso de la nube privada hasta un 32.5%, en comparación con el 29% que se informó el año pasado.

Aumento del
36%
en el uso de la nube
pública

Aumento del
25%
en el uso de la nube
híbrida



El uso de la computación en la nube tiene un impacto positivo en el rendimiento general de la organización. Los encuestados que usan la nube tienen un **14% más de probabilidades de superar** los objetivos de rendimiento de la organización que sus compañeros que no usan la nube.

Como demostramos en años anteriores, y seguimos validando en este informe, el uso de la computación en la nube tiene un impacto positivo en el rendimiento general de la organización. Los encuestados que usan la nube tienen un **14% más de probabilidades de superar** los objetivos de rendimiento de la organización que sus compañeros que no usan la nube. Nuestra investigación muestra que la computación en la nube permite a los equipos sobresalir en aspectos como la seguridad y la confiabilidad de la cadena de suministro de software, que a su vez contribuyen al rendimiento de la organización.

Resulta sorprendente que los usuarios de todos los tipos de nube (pública, privada, híbrida y múltiple) mostraran una asociación negativa con la tasa de errores de cambios, lo que significa un aumento de la tasa de errores de cambio.

Esto garantiza una mayor investigación. En lugar de especular sobre las razones de esta situación, profundizaremos en futuras investigaciones. Sin embargo, salvo en algunas excepciones, el uso de aplicaciones nativas de la nube (aplicaciones que se diseñaron para la nube) se destacó con indicadores positivos en todos los aspectos que encuestamos.

El uso de cualquier plataforma de computación en la nube, pública o privada, contribuye de forma positiva a los resultados de la cultura y el entorno de trabajo (por ejemplo, cultura generativa, menos agotamiento, más estabilidad y mayor satisfacción de los empleados). Los usuarios de la nube obtuvieron el 16% de puntuación más alto en estos resultados culturales.

El uso de la nube sigue acelerando su crecimiento interanual

	2022	Porcentaje de cambio durante el año 2021
Nube híbrida	42.47%	25%
Nubes públicas o múltiples nubes públicas	76.08%	36%
Nube privada	32.55%	12%
Sin Cloud	10.55%	-50%

El uso de nubes híbridas y múltiples (y privadas) parece tener un impacto **negativo** en los indicadores de rendimiento de la entrega de software (MTTR, plazo de entrega y frecuencia de implementación) **a menos que** los encuestados tengan altos niveles de **confiabilidad**.

Las nubes híbridas y múltiples impulsan el rendimiento de la organización

Se siguen observando fuertes indicadores de que el uso de la nube híbrida y de múltiples nubes públicas tiene un impacto positivo en las organizaciones. Los profesionales que usaron varias nubes mostraron un rendimiento organizativo 1.4 veces mayor en comparación con los que no usaron la nube. Sin embargo, el uso de la nube híbrida y múltiples nubes (así como la nube privada) parece tener un impacto negativo en varios indicadores de rendimiento de la entrega de software (MTTR, tiempo de entrega y frecuencia de implementación) a menos que los encuestados también tengan altos niveles de confiabilidad. Este hallazgo demuestra la importancia de

una práctica sólida de SRE y el rol que desempeña la confiabilidad en la entrega de software.

En 2021, les pedimos a los encuestados que nos dijeran la razón *principal* por la que usaban múltiples nubes públicas, mientras que en 2022 les pedimos a los participantes que nos indicaran todos los beneficios que obtuvieron con el uso de múltiples proveedores de servicios en la nube. La disponibilidad fue la primera ventaja que se mencionó, lo que coincide con la atención y el enfoque que vimos en el sector en torno a la confiabilidad: no se pueden tener servicios confiables, a menos que estén disponibles. Más del 50% de los profesionales afirmaron que aprovechaban las ventajas exclusivas de los distintos proveedores de servicios en la nube.

Beneficios que se obtienen con la incorporación de múltiples proveedores de servicios en la nube

Disponibilidad	62.61%
Aprovechar las ventajas únicas de cada proveedor	51.59%
La confianza se expande entre varios proveedores	47.54%
Recuperación ante desastres	43.48%
Cumplimiento legal	37.97%
Táctica de negociación o requisito de adquisición	19.13%
Otro	4.06%

> El **50%**

de los encuestados afirma que usa varios proveedores de servicios en la nube.

El uso de las cinco características de la computación en la nube es un comienzo fundamental de una larga cadena causal que conduce al rendimiento de la organización.

Las cinco características de la computación en la nube

De acuerdo con nuestro enfoque de investigación anterior, no solo queríamos saber si los participantes usaban tecnologías de computación en la nube, sino cómo las usaban. Para ello, nos preguntamos por las cinco características esenciales de la computación en la nube, según la definición del Instituto Nacional de Estándares y Tecnología (NIST).

Autoservicio a pedido: los consumidores pueden aprovisionar recursos de procesamiento según sea necesario, de forma automática, sin la interacción humana del proveedor.

Acceso amplio a la red: están disponibles de forma masiva y los consumidores pueden acceder a ellas a través de varios clientes, como teléfonos celulares, tablets, laptops y estaciones de trabajo.

Agrupación de recursos: los recursos del proveedor se agrupan en un modelo de multiusuarios, con recursos físicos y virtuales asignados de forma dinámica a pedido.

Por lo general, el cliente no tiene control directo sobre la ubicación exacta de los recursos proporcionados, pero puede especificar la ubicación en un nivel más alto de abstracción, como país, estado o centro de datos.

Elasticidad rápida: Las funciones se pueden aprovisionar de forma elástica y se lanzan para escalar con rapidez hacia afuera o hacia adentro a pedido. Las capacidades del consumidor disponibles para el aprovisionamiento parecerían ser ilimitadas y se pueden asignar en cualquier cantidad y en cualquier momento.

Servicio medido: los sistemas en la nube controlan y optimizan de forma automática el uso de recursos mediante la capacidad de medición en un nivel de abstracción adecuado para el tipo de servicio, como almacenamiento, procesamiento, ancho de banda y cuentas de usuario activas. El uso de los recursos se puede supervisar, controlar y notificar para que sea transparente.

Este informe valida los tres años anteriores de investigación de DORA, y concluye que la presencia de estas cinco características en una organización afecta de forma positiva a la entrega de software y al rendimiento de las operaciones. También comprobamos que estas características permiten mejorar el rendimiento de la organización, ya que ponen en marcha procesos que afectan a la organización de forma positiva. Mostrar las cinco características de la computación en la nube es el primer paso de un largo viaje que conduce a un mayor rendimiento organizativo.

En 2022, se observa que los equipos aprovechan cada vez más los diferenciadores de la computación en la nube. Por cuarto año consecutivo, observamos una creciente incorporación de las cinco características de la computación en la nube. La agrupación de recursos registró el mayor incremento, del 14%, y la elasticidad rápida, que fue la segunda característica que más se usó el año pasado, registró el menor aumento, del 5%.

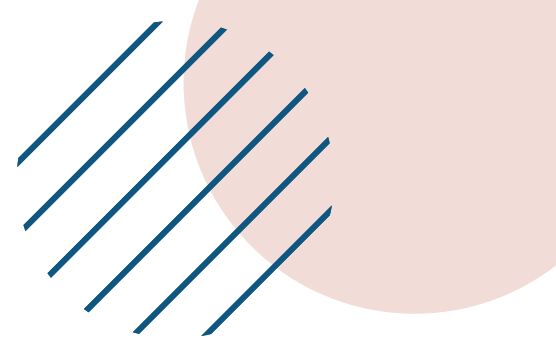
NIST	2021	2022	Cambio porcentual
Acceso amplio a la red	74%	80%	8
Elasticidad rápida	77%	81%	5
Autoservicio a pedido	73%	78%	7
Servicio medido	78%	83%	7
Agrupación de recursos	73%	83%	14



Aumento del
14%
en la agrupación
de recursos



Dave Stanke



SRE y DevOps

Un equipo tecnológico competente contribuye más a su organización que el envío del código, incluso más que el envío del código de calidad. También garantizan que los servicios que entregan se mantengan disponibles, tengan un buen rendimiento y cumplan con las expectativas de los usuarios en el tiempo. La confiabilidad es una medida multifacética de lo bien que un equipo mantiene estos compromisos, y este año continuamos nuestras exploraciones sobre la confiabilidad como un factor en la entrega y las operaciones de software.

La ingeniería de confiabilidad de sitios (SRE) es un enfoque influyente de las operaciones que se originó en Google y ahora se practica en muchas organizaciones. La SRE prioriza el aprendizaje empírico, la colaboración multifuncional, la amplia dependencia de la automatización y el uso de técnicas de medición, entre ellas los objetivos de nivel de servicio (SLO). Otras prácticas operativas modernas emplean métodos similares, no obstante, aplican convenciones de nombres diferentes. Por lo tanto, para evaluar el alcance de estas prácticas de la forma más objetiva posible, nuestra encuesta procura emplear un lenguaje neutro y descriptivo en el texto que presentamos a los encuestados. También recopilamos datos de los

resultados de la ingeniería de confiabilidad: la medida en que los equipos pueden lograr sus objetivos de confiabilidad. Las entradas y salidas (prácticas de SRE y resultados de confiabilidad) se reflejan en nuestro modelo predictivo junto con otras capacidades de DevOps.

La confiabilidad es esencial

El uso de la SRE es una práctica muy común entre los equipos encuestados: la mayoría de los encuestados usa una o más de las prácticas que preguntamos. En toda esta diversidad de equipos, los datos revelan una relación sutil entre la confiabilidad, la entrega de software y los resultados: cuando la confiabilidad es pobre, el rendimiento de la entrega de software no predice el éxito de la organización. Sin embargo, con una mayor confiabilidad, comenzamos a ver la influencia positiva de la entrega de software en el éxito empresarial.

Sin confiabilidad, el rendimiento de la entrega de software no predice el éxito de la organización.

Invertir en la SRE produce mejoras en la confiabilidad, pero solo una vez que se alcanzó un umbral de aceptación.

Este fenómeno es coherente con el uso del framework de “porcentaje de error aceptable” de la SRE: cuando un servicio no es confiable, los usuarios no se beneficiarán de enviar códigos más rápido en ese contexto frágil.

Como afirman desde hace tiempo los ingenieros de confiabilidad de sitios, la confiabilidad es la “característica” más importante de cualquier producto. Nuestra investigación respalda la observación de que mantener las promesas a los usuarios es una condición necesaria para que la mejora de la entrega de software beneficie a la organización.

Confirma la curva J

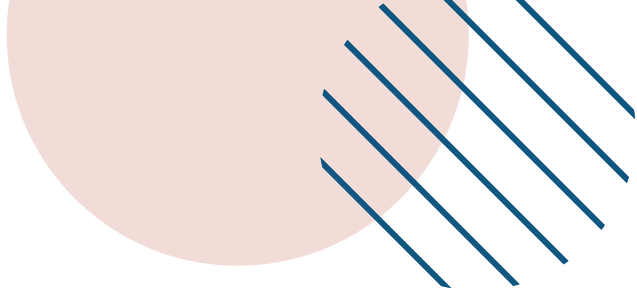
¿Qué desafíos te esperan en el camino hacia la confiabilidad? En la publicación de O’Reilly “Hoja de ruta de Enterprise para el informe de SRE”¹, los colaboradores de la encuesta DORA, James Brookbank y Steve McGhee, reflexionan sobre sus experiencias de implementación de SRE en organizaciones establecidas y recomiendan “confirmar la curva J del cambio”. Descrito con anterioridad en el Informe State of DevOps de 2018, la “Curva J” es un fenómeno en el que las transformaciones organizativas suelen exhibir un éxito temprano, seguido de períodos de disminución de los rendimientos, o incluso regresiones.

¹ <https://sre.google/resources/practices-and-processes/enterprise-roadmap-to-sre/>

Sin embargo, quienes persisten a través de estos desafíos, a menudo experimentan niveles renovados y sostenidos de logros elevados.

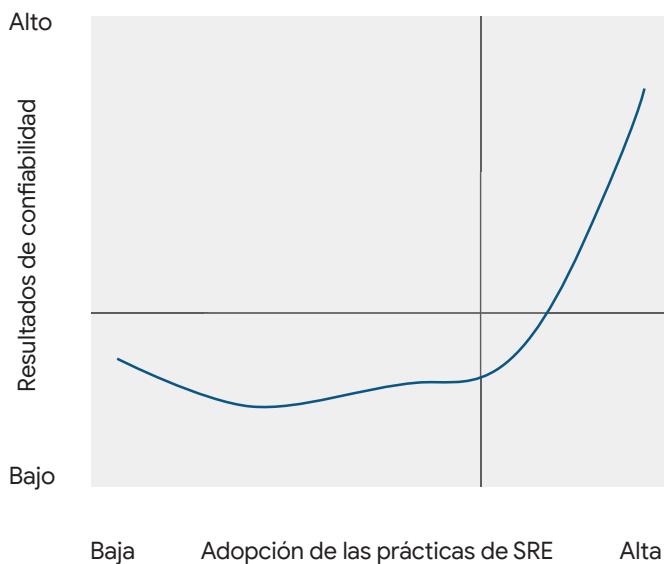
La investigación de este año revela un patrón de la Curva J en los equipos tecnológicos que estudiamos: cuando los equipos participan en menos prácticas de ingeniería de confiabilidad (lo que sugiere que están en una etapa más temprana de la adopción de la SRE), estas prácticas no predicen mejores resultados de confiabilidad. Sin embargo, a medida que los equipos adoptan más la SRE, llegan a un punto de inflexión en el que el uso de la SRE comienza a predecir con firmeza la confiabilidad y, a la vez, el rendimiento de la organización.





Los equipos confiables hacen servicios dependientes: la cultura de los equipos generativos predice una mayor confiabilidad.

Los equipos que están en las primeras etapas de un recorrido hacia una práctica de la SRE deben estar preparados para los contratiempos en el camino. Puede ser un largo recorrido, ya que la cultura, el proceso y las herramientas se alinean con los nuevos principios básicos. Pero pueden estar seguros de que, con el tiempo y una inversión continua, el éxito es probable.



Los equipos que perseveran tras los pasos iniciales de la adopción de la SRE notan una mejora creciente en los resultados de confiabilidad

Invierte en personas, procesos y herramientas

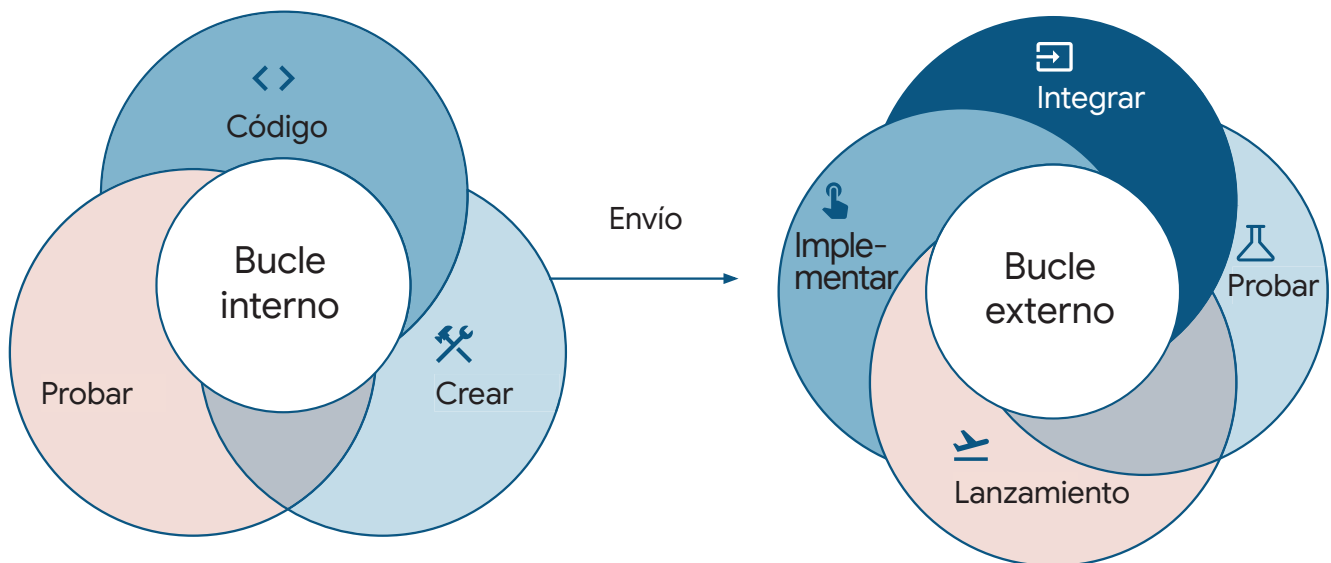
La confiabilidad es un esfuerzo humano, y en muchos sentidos el enfoque de la SRE lo ejemplifica. Uno de los principios básicos de la SRE es que la percepción del usuario, a diferencia de los datos de supervisión interna, es la medida real de la confiabilidad. Por tanto, no es de extrañar que la confiabilidad se vea impulsada por una dinámica de equipo positiva. Descubrimos que los equipos con una cultura “generativa”, que exhibe confianza y colaboración, son más propensos a practicar la SRE, y más propensos a lograr buenos resultados de confiabilidad. Los equipos estables, cuya composición es coherente a lo largo del tiempo, también ofrecen servicios para el usuario con una mayor confiabilidad. Además, al igual que con DevOps en su totalidad, los esfuerzos de ingeniería de confiabilidad se benefician del aumento de los esfuerzos humanos con procesos y herramientas. Prácticas como el uso de la computación en la nube y la integración continua predicen mejores resultados de confiabilidad.

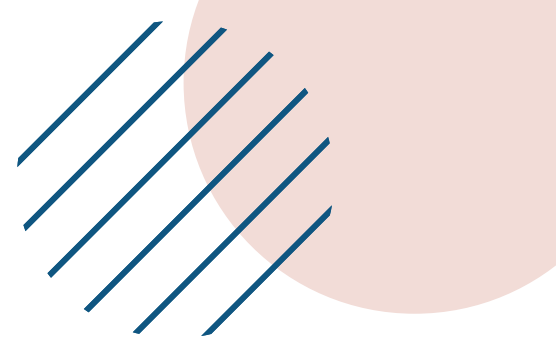


Eric Maxwell

Capacidades técnicas de DevOps

Este año, analizamos una variedad de capacidades técnicas para comprender los resultados que impulsan las diferentes prácticas técnicas. Consideramos dos fases amplias del desarrollo de software: el “bucle interno”, que comprende tareas de los desarrolladores como la programación, las pruebas y el envío al control de versión, y el “bucle externo”, que incluye actividades como la combinación de códigos, la revisión automatizada de código, la ejecución de prueba, la implementación y el lanzamiento.





Quienes alcanzan los objetivos de confiabilidad son **1.4 veces más propensos a usar CI.**

Nuestra investigación muestra que las empresas que se destacan en el desarrollo de bucles internos y externos pueden enviar el código más rápido y con mayores niveles de confiabilidad. Las capacidades que contribuyen en mayor medida al alto rendimiento son el control de versión, la integración continua, la entrega continua y la arquitectura con bajo acoplamiento.

A continuación, se detalla quienes alcanzan los objetivos de confiabilidad:

33% más probabilidades de usar el control de versión

39% más probabilidades de aplicar la integración continua

46% más probabilidades de aplicar la entrega continua

40% más probabilidades de tener sistemas basados en una arquitectura con acoplamiento bajo

De hecho, los encuestados que hacen un uso más alto que el promedio de todas las capacidades anteriores tienen un **rendimiento organizativo 3.8 veces mayor** que los que no usan estas capacidades técnicas.

Integración continua

La integración continua, también conocida como CI, es una parte del proceso de desarrollo de bucle externo que compila de forma automática un artefacto y ejecuta una serie de pruebas automatizadas para cada confirmación de código en un esfuerzo por evaluar si el código está listo a fin de ser implementado. Este proceso ofrece comentarios rápidos y automatizados al desarrollador, lo que les permite operar con mayores niveles de confianza. CI es una parte clave para llevar el código desde la estación de trabajo de un desarrollador hasta la producción. Al igual que en años anteriores, se demuestra que la CI impulsa el rendimiento de entrega. **Quienes alcanzan los objetivos de confiabilidad son 1.4 veces más propensos a usar CI que otros.**

Este año profundizamos un poco más en la otra parte del proceso de desarrollo del bucle externo: la entrega continua, que describiremos en un capítulo más adelante. Pero antes, veamos un componente complementario de la integración continua: el desarrollo basado en troncales.

Desarrollo basado en troncales

El desarrollo basado en troncales es la práctica de combinar de forma continua el código en la troncal y evitar las ramas de características de larga duración. Esta práctica se considera un complemento de la integración continua y se demostró durante años que acelera la velocidad de entrega del software.

Debido al cambio en los datos demográficos de este año en cuanto a los años de experiencia en el trabajo, podemos ver que la experiencia importa a la hora de aplicar el desarrollo basado en troncales. El año pasado, el 40% de los encuestados afirmaron tener más de 16 años de antigüedad en el trabajo, mientras que este año esa categoría solo representa el 13%. Para seguir validando nuestro tema “dependencias del envío”, vemos que las personas con menos experiencia en general obtienen resultados menos positivos en relación con el desarrollo basado en troncales y vemos lo siguiente:

- ▼ **Disminución** del rendimiento general de la entrega de software
- ▲ **Aumento** de la cantidad de trabajo no planificado
- ▲ **Aumento** de probabilidad de errores
- ▲ **Aumento** de la tasa de errores de cambio

Las personas con más de 16 años de experiencia que usan el desarrollo basado en troncales son conscientes de los beneficios de la práctica y comprueban lo siguiente:

- ▲ **Aumento** del rendimiento general de la entrega de software
- ▼ **Disminución** de la cantidad de trabajo no planificado
- ▼ **Disminución** de probabilidad de errores
- ▼ **Disminución** de la tasa de errores de cambio

Es probable que esto se deba a las prácticas adicionales que se requieren para implementar con éxito el desarrollo basado en troncales. Los equipos que no tienen reglas estrictas para no abandonar nunca una troncal dañada o que no usan ramas de código con acceso restringido y revierten de forma automática el código que daña la troncal, sin duda sufrirán cuando traten de desarrollar en la troncal.

Sin embargo, la presencia del desarrollo basado en troncales muestra un impacto positivo en el rendimiento general de la organización.



Frank Xu

Entrega continua

La entrega continua (CD) es una práctica de desarrollo de software con los siguientes beneficios:

1. Permite al equipo implementar el software a la producción o a los usuarios finales en cualquier momento
2. Garantiza que el software esté en un estado que permita la implementación durante todo su ciclo de vida, incluso cuando se trabaja en nuevas características
3. Establece un bucle de comentarios rápidos que permite al equipo comprobar la calidad y la capacidad de implementación del sistema, y otorga prioridad a la solución de problemas que bloquean la implementación.

Ten en cuenta que la entrega continua no implica siempre el despliegue continuo, la práctica en la que cada compilación de software se implementa de forma automática. La entrega continua solo requiere que una compilación de software pueda implementarse en cualquier momento.

El año pasado, examinamos las capacidades técnicas de DevOps que predicen la probabilidad de que un equipo emplee la CD, y descubrimos que los factores como la arquitectura con acoplamiento bajo y las pruebas y las integraciones continuas estaban entre las predicciones más fuertes. Este año, además de examinar los factores que impulsan el uso de la CD, identificamos y analizamos los efectos de la CD por sí misma, así como sus interacciones con otras capacidades de DevOps en los resultados del desarrollo.

La CD impulsa el rendimiento de la entrega de software

Al igual que los hallazgos de años anteriores, el uso de CD es un indicador de un mayor rendimiento en la entrega de software, por separado y en combinación con otras capacidades de DevOps. Los equipos que obtuvieron una mayor puntuación en CD son más propensos a tener una mayor frecuencia de implementación de código en producción y un menor plazo de entrega para los cambios y el restablecimiento del servicio.

Los equipos que combinan el control de versión y la entrega continua tienen 2.5 veces más probabilidades de tener un alto rendimiento en la entrega de software que los equipos que solo se enfocan en uno.

Además, los encuestados son 2.5 veces más propensos a informar sobre un mayor rendimiento en la entrega de software cuando su equipo también adopta prácticas de control de versión.

La CD puede incrementar el trabajo no planificado

Los datos sugieren que la entrega continua permite a los desarrolladores dedicar más tiempo a la repetición de tareas o al trabajo no planificado. Una hipótesis de este hallazgo es que los desarrolladores tienen más posibilidades de compilar aplicaciones de forma iterativa cuando existen ciclos de reacción más estrechos. En consecuencia, pueden considerar algunos cambios iterativos como trabajo no planificado en la misma parte del sistema. Este trabajo puede ser al mismo tiempo no planificado y, además impulsado por los comentarios de una implementación anterior.

Prácticas técnicas y CD

Nuestra investigación demuestra con regularidad que un amplio conjunto de capacidades técnicas apoyan la CD. Este año, exploramos lo que ocurre cuando algunas de estas capacidades individuales se usan junto con la CD. Descubrimos que el desarrollo basado en troncales y la arquitectura con acoplamiento bajo, junto con la CD, pueden tener un impacto negativo en el rendimiento de un equipo. Por ejemplo, observamos que los equipos que adoptan de forma conjunta arquitecturas con acoplamiento bajo y CD tienen un 43% más de probabilidades de anticipar una probabilidad de error superior al promedio (es decir, interrupciones de producto, vulnerabilidades de seguridad y una degradación significativa del rendimiento de sus servicios), en comparación con los equipos que solo adoptaron la CD. Estos efectos precisan de una mayor investigación y apuntan a una posible fricción para los equipos que mejoran. Esta fricción puede estar relacionada con la curva J de la transformación, en la que los equipos consiguen mejoras anticipadas, pero luego se tambalean a medida que avanzan tras resultados inmediatos. Se requiere un compromiso de mejora a fin de desarrollar todo su potencial. Para mejorar cualquier capacidad, como la CD, hay que asegurarse de observar los efectos en el equipo y en el rendimiento general.



David Farley

Arquitectura con acoplamiento bajo

Los sistemas con acoplamiento bajo son importantes para la eficacia de los equipos y las organizaciones. Esto no solo se aplica a los sistemas basados en la nube o en los microservicios, sino que está relacionado con la capacidad de cambio de una organización. La facilidad con la que una organización puede cambiar su software con seguridad y confianza es un indicador de la calidad del software.

Con una arquitectura con acoplamiento bajo los equipos pueden hacer lo siguiente:

- Realizar cambios a gran escala en el diseño de su sistema sin tener que depender de otros equipos que realicen cambios en sus sistemas
- Obtener comentarios más rápidos a través de pruebas independientes y a la carta con menores costos de coordinación
- Implementar el código con un tiempo de inactividad mínimo

En el informe de este año, les pedimos a los encuestados que describieran si el software que construyen se basa en una arquitectura con acoplamiento bajo o no. Los resultados fueron interesantes y mostraron una variedad de asociaciones, en su mayoría positivas, entre la presencia de una arquitectura con acoplamiento bajo y el rendimiento de los equipos en múltiples dimensiones.



Los beneficios de una arquitectura con acoplamiento bajo

Los equipos que se enfocan en la compilación de software con arquitecturas con acoplamiento bajo están en mejor posición para desempeñarse con solidez en cuanto a estabilidad, confiabilidad y capacidad de procesamiento. Estos equipos también son más propensos a recomendar su lugar de trabajo a un amigo o colega.

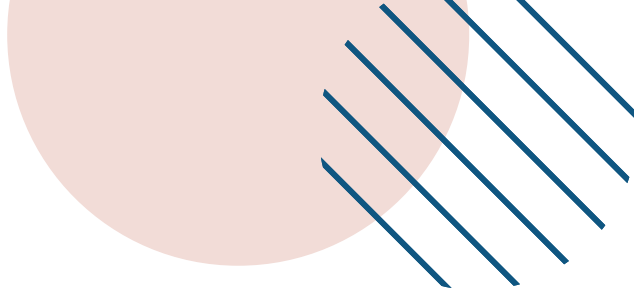
Es muy común ver el software que usa una arquitectura con acoplamiento bajo de los equipos que se implementan en la nube, adoptan un enfoque de arquitectura de microservicios y

Los equipos que se enfocan en la compilación de software con arquitecturas con acoplamiento bajo están en mejor posición para desempeñarse con solidez en cuanto a estabilidad, confiabilidad y capacidad de procesamiento.

administran cientos de servicios. Sin embargo, el acoplamiento bajo es más que una simple medida de la cantidad de servicios en un sistema. Los componentes de una arquitectura con acoplamiento bajo pueden implementarse de forma independiente. Esta independencia les permite a los equipos desarrollar, probar e implementar sus servicios sin una sobrecarga de coordinación costosa entre los equipos.

En el mundo real, el acoplamiento bajo no se limita a un estilo arquitectónico. En esencia, es la habilidad de hacer un cambio en una parte del sistema, sin que ese cambio afecte a otras partes. De esta forma, las organizaciones pueden dividir su trabajo para que los equipos individuales puedan avanzar sin tener que coordinarse con otros equipos.

De acuerdo con nuestra experiencia, los equipos que requieren pruebas de integración profundas con otros servicios como una forma de crear confianza en su software antes de implementarlo no lograron todavía un acoplamiento bajo. Para ello, estos equipos se beneficiarían de la mejora de las interfaces y el aislamiento entre los sistemas. Una forma eficaz de mejorar las interfaces y el aislamiento es mejorar la “Capacidad de realizar pruebas” de los servicios y componentes. Si el diseño te permite probar el servicio de forma aislada, entonces su interfaz está, por definición, con acoplamiento bajo.



También descubrimos que los equipos cohesivos y estables que usan una arquitectura con acoplamiento bajo tienen más probabilidades de usar prácticas de desarrollo de software que fomentan y apoyan la mejora continua. Por ejemplo, las prácticas de SRE, como establecer objetivos de confiabilidad para priorizar el trabajo o realizar revisiones periódicas a fin de revisar los objetivos de confiabilidad sobre la base de la evidencia, son compatibles con la arquitectura con acoplamiento bajo.

Las arquitecturas con acoplamiento bajo también permiten a la organización incorporar empleados con más facilidad, ya que los equipos independientes que no necesitan coordinarse con otros equipos tienen más libertad para aumentar el tamaño de sus equipos de forma independiente.

En resumen, el acoplamiento bajo de los servicios de software tiene un mayor impacto que el técnico. También afecta los aspectos sociotécnicos del desarrollo de software. El acoplamiento está en la raíz de la Ley de Conway: la idea de que los sistemas de diseño de una organización reflejan su propia estructura de comunicación. Más sistemas con acoplamiento bajo significan más organizaciones con acoplamiento bajo con un enfoque de desarrollo más distribuido y escalable.



Resultados sorprendentes

La investigación de este año reveló que la arquitectura con acoplamiento bajo podría contribuir al agotamiento de los equipos. Este es un resultado sorprendente que contradice los resultados de años anteriores. Nuestro análisis demuestra que los equipos estables en los que la información fluye sin problemas tienen niveles más bajos de agotamiento. La cultura generativa de Westrum y la estabilidad del equipo respaldan la arquitectura con acoplamiento bajo y disminuyen el agotamiento, por lo que esto es una clara contradicción. Se necesitan más investigaciones antes de poder sacar conclusiones definitivas.

Al mismo tiempo, cuando los requisitos de seguridad son definidos y controlados a través de una organización de seguridad consolidada, puede ser más difícil para los equipos separar su software de otros equipos. Así se demuestra aún más el beneficio de trasladar las preocupaciones de seguridad al equipo más responsable de la aplicación (consulta también este artículo: [Por qué es importante la seguridad de la cadena de suministro](#)). Esta es una de las formas más sutiles de acoplamiento en las organizaciones, y aunque recopilamos datos sobre seguridad, es probable que esto sea igual de cierto en otras funciones centralizadas. Permitir que los equipos tomen sus propias decisiones en cuanto a la seguridad, y otras funciones centralizadas con frecuencia, es una forma de avanzar hacia el aprovechamiento de las ventajas que el uso de la arquitectura con acoplamiento bajo puede aportar a tu organización.



Daniella Villalba

Cultura

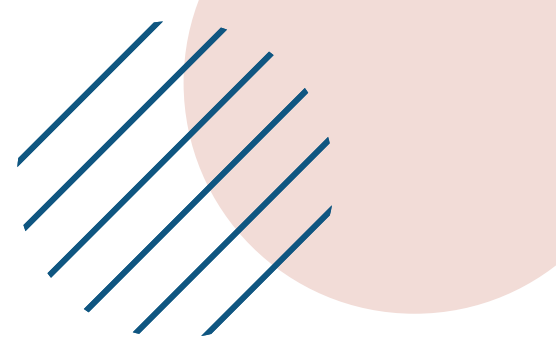
“Bien, así es como se hacen las cosas por aquí”.

Es probable que se haya pronunciado esta frase en innumerables ocasiones y en una amplia gama de sectores para describir el enfoque de su organización ante los desafíos y las oportunidades.

Cada organización tiene su propia cultura, y nuestra investigación demostró de forma coherente que la cultura es fundamental para el éxito de una organización y el bienestar de sus empleados.

La cultura también es un aspecto necesario de DevOps, ya que en el nivel más básico, DevOps tiene que ver con las herramientas, las prácticas y **la forma en que las personas trabajan juntas** para desarrollar y entregar el software con rapidez, confianza y seguridad. Comprender los factores que influyen en la cultura de una organización puede ayudar a los líderes a afrontar los desafíos relacionados con la cultura. Es por ello que fomentar una cultura saludable debe ser una prioridad para las organizaciones. Si no se atienden, estos desafíos relacionados con la cultura pueden impedir que las prácticas de DevOps se apliquen.





Este año seguimos usando la tipología organizativa de Westrum para medir la salud de la cultura de una organización. Además, ampliamos nuestra comprensión de la cultura con la medición de la deserción del equipo, los acuerdos de trabajo flexibles, la percepción de la participación de la organización y el agotamiento.

Los datos de la investigación de este año respaldan los resultados anteriores que indican que el **rendimiento de la organización** se ve afectado por el tipo de cultura que existe en ella. De forma específica, una **cultura generativa** se asocia con niveles más altos de rendimiento organizativo en comparación con las organizaciones que se caracterizan por una cultura burocrática o patológica. Los empleados de las

organizaciones con una cultura generativa tienen más probabilidades de pertenecer a equipos estables, producir documentación de mayor calidad y dedicar la mayor parte de su tiempo a un trabajo valioso.

Deserción del equipo

Investigamos la deserción de equipos y descubrimos que los **equipos estables** (equipos con una composición que no cambió mucho en los últimos 12 meses) tienen más probabilidades de existir en las organizaciones de alto rendimiento. La deserción constante puede afectar a la productividad y a la moral, ya que los nuevos miembros del equipo necesitan tiempo para integrarse. Y aquellos que

Cultura organizativa de Westrum

Patológica <i>Orientada al poder</i>	Burocrática <i>Orientada a las reglas</i>	Generativa <i>Orientada al rendimiento</i>
Colaboración baja	Colaboración moderada	Colaboración alta
Se culpa al mensajero	Se ignora al mensajero	Se capacita al mensajero
Responsabilidades eludidas	Limitación de responsabilidades	Se comparten las responsabilidades
Se desaconsejan las conciliaciones	Se toleran las conciliaciones	Se incentivan las conciliaciones
El fracaso genera chivos expiatorios	El fracaso conduce a la justicia	El fracaso genera preguntas
No se permite la innovación	La innovación genera problemas	Se implementa la innovación

permanecen tendrían que adaptarse a los cambios en su carga de trabajo y en la dinámica del equipo. Además, nuestra investigación demostró que los equipos estables tenían más probabilidades de informar la producción de documentación de calidad en comparación con los equipos que experimentaron más deserción. Un equipo que afronta los cambios de forma constante puede tener más dificultades para mantenerse al día con las prácticas que conducen a una documentación de calidad.

Las organizaciones de alto rendimiento tienen más probabilidades de contar con acuerdos de trabajo flexibles.

Acuerdos de trabajo flexibles

Debido al cambio de los acuerdos de trabajo flexibles que muchas organizaciones adoptaron desde el inicio de la pandemia del COVID-19, investigamos si dar a los empleados la libertad de elegir entre opciones remotas, presenciales

o híbridas se asociaba con un mayor rendimiento de la organización. Los resultados mostraron que las organizaciones con niveles más altos de **flexibilidad de empleados** tienen un mayor rendimiento organizativo en comparación con las organizaciones con acuerdos de trabajo más rígidos. Estos resultados demuestran que ofrecer la libertad de modificar sus acuerdos laborales a los empleados según sus necesidades conlleva beneficios tangibles y directos para una organización.

Agotamiento

El agotamiento es un sentimiento de temor, apatía y cinismo en torno al trabajo. Cuando las personas sufren agotamiento, no solo están desmotivadas y agotadas, sino que también es muy probable que tengan una menor satisfacción en el trabajo, lo que puede aumentar la rotación de los empleados. El agotamiento se vincula a un amplio rango de resultados de salud psicológica y física, como el aumento del riesgo de depresión y ansiedad, enfermedades cardíacas y pensamientos suicidas¹.

El año pasado medimos el agotamiento en el contexto de la pandemia del COVID-19 y comprobamos que una cultura generativa estaba asociada a índices más bajos de agotamiento de los empleados.

¹ Maslach C, Leiter MP. Understanding the burnout experience: recent research and its implications for psychiatry. World Psychiatry. Junio de 2016; 15(2):103-11. doi: 10.1002/wps.20311. PMID: 27265691; PMCID: PMC4911781.

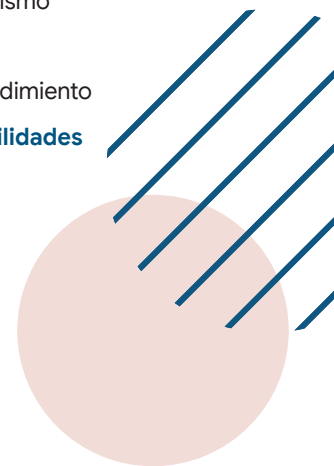
Los modelos de trabajo flexibles se asocian a la disminución del agotamiento de los empleados y al aumento de la probabilidad de que recomienden a su equipo como un buen lugar para trabajar.

Este año repetimos este resultado y ampliamos nuestra comprensión del agotamiento con la demostración de que los equipos estables y los acuerdos de trabajo flexibles también están asociados a un menor agotamiento. Además, este año medimos el Net Promoter Score (NPS) del equipo, que indica si los empleados recomendarían su equipo a un amigo o colega. Comprobamos que el NPS del equipo estaba asociado a la percepción de la aceptación del liderazgo. Asimismo, al igual que los resultados sobre el agotamiento, comprobamos que una cultura generativa, un equipo estable y un acuerdo de trabajo flexible se asocian con la posibilidad de que las personas recomienden su equipo a otros.

Cómo perciben los empleados a su organización

Por último, investigamos la percepción de la aceptación del liderazgo pidiéndoles que predijeran cuánta asistencia esperaban que recibiera su equipo en los próximos 12 meses. Los resultados mostraron que una mayor aceptación del liderazgo percibida (por ejemplo, más apoyo financiero, más asignación de recursos, patrocinios) se asociaba con las organizaciones de alto rendimiento.

También les pedimos que predijeran la probabilidad de que se produjera una violación de la seguridad o una interrupción completa en los próximos 12 meses. Los resultados demostraron que las personas que trabajaban en organizaciones de alto rendimiento tenían menos probabilidades de esperar que se produjera un error grave: tenían una visión más positiva de su organización. Del mismo modo, comprobamos que las personas que trabajaban en organizaciones con un alto rendimiento de software y entrega tenían **menos probabilidades** de considerar que era necesario cambiar sus prácticas actuales para mejorar los resultados empresariales.



Algunos comentarios sobre la representación

Los resultados indicaron que los empleados de grupos infrarrepresentados tenían más probabilidades de informar que dedicaban más tiempo al trabajo no planificado, sin importar que pertenecieran a organizaciones de alto o bajo rendimiento. También comprobamos que los empleados de los grupos infrarrepresentados informaron mayores niveles de agotamiento en comparación con los empleados que no pertenecen a estos grupos. Los líderes de equipo deben tener en cuenta el riesgo de desequilibrio de la carga de trabajo y asegurarse de que el trabajo se asigna de forma equitativa entre los miembros del equipo.

Juntos, estos resultados ponen de manifiesto la importancia de crear un entorno inclusivo y saludable para los empleados, tanto a nivel de organización como de equipo.

Aunque seguimos haciendo hincapié en la importancia de la cultura, reconocemos que cambiar o incluso mejorar la cultura de una organización no es tarea fácil.

Recomendamos que las organizaciones traten de entender primero las experiencias de sus empleados y, en consecuencia, inviertan recursos a fin de abordar los problemas relacionados con la cultura como parte de los esfuerzos de transformación de DevOps.



04

Por qué es importante la seguridad de la cadena de suministro

En noviembre de 2020, eran muy pocos los profesionales de la tecnología que sospechaban que se avecinaba una crisis de seguridad en la cadena de suministro de software. La [Open Source Security Foundation](#), una organización sucesora de los esfuerzos anteriores, se fundó a fin de enfocarse en la seguridad del software de código abierto, y aunque hubo algunos [puntos brillantes](#) para abordar este problema, el tema no estaba en la primera página de los principales periódicos. Un ataque importante, [SolarWinds](#), cambió todo eso.

Cuando los atacantes pueden penetrar en silencio en miles de grandes empresas y redes gubernamentales gracias a las actualizaciones de software de troyanos, los tiempos cambian rápido.

En la actualidad, el tema de la seguridad de la cadena de suministro de software es un asunto urgente, tal vez no durante la cena familiar, pero sí en la sala de juntas.



John Speed Meyers



Todd Kulesza



Existen numerosas iniciativas y gran parte de la industria del software está comprometida con la reforma de sus propias prácticas de seguridad en la cadena de suministro de software y con la mejora de la seguridad de los bienes comunes de código abierto.

En este capítulo, nos enfocamos en dos iniciativas: Niveles de la cadena de suministro para artefactos seguros ([SLSA](#), que se pronuncia “salsa”), y el Framework de Desarrollo Seguro de Software NIST ([SSDF](#)). Cada uno ofrece una serie de medidas de defensa para asegurarse de que los atacantes no puedan manipular los procesos de producción de software y evitar a los defensores de la red mediante actualizaciones de software maliciosas.

Pero ¿en qué medida se usan las prácticas de seguridad de cadena de suministro del software asociadas con SLSA y SSDF? ¿Qué prácticas necesitan ayuda para impulsar su incorporación y cuáles ya se usan de forma generalizada? Hasta la fecha, no había respuestas sistemáticas a estas preguntas. Gracias a una encuesta realizada a cientos de profesionales del sector del software sobre el uso que hacen de las prácticas asociadas a la seguridad de la cadena de suministro, ofrecemos algunas respuestas iniciales. En concreto, se destacan cuatro resultados principales:

01 La incorporación ya comenzó: Las prácticas de seguridad de la cadena de suministro de software plasmadas en los SLSA y el SSDF ya cuentan con una modesta incorporación, pero hay un amplio espacio para más.

02 Las culturas más sanas tienen ventaja: La cultura organizativa es un elemento fundamental de las prácticas de seguridad en el desarrollo de software, ya que las culturas de mayor confianza y “sin culpa” tienen más probabilidades de establecer prácticas de los SLSA y el SSDF que las culturas organizativas de menor confianza.

03 Existe un punto de integración clave: La adopción de los aspectos técnicos de la seguridad de la cadena de suministro de software parece depender del uso de CI/CD, que a menudo proporciona la plataforma de integración para muchas prácticas de seguridad de la cadena de suministro.

04 Ofrece beneficios inesperados: Además de una reducción de los riesgos de seguridad, mejores prácticas de seguridad conllevan ventajas adicionales, como la reducción del agotamiento.

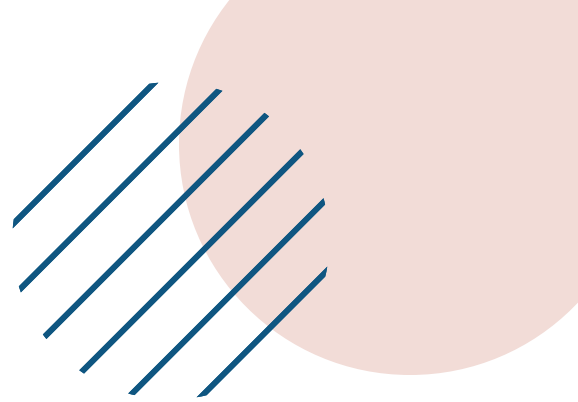
Lo que las empresas hacen en la actualidad para evitar las vulnerabilidades de seguridad

A fin de comprender mejor lo que las organizaciones hacen hoy para identificar y resolver las vulnerabilidades de seguridad en el software que compilan, agregamos más de 24 preguntas a la encuesta de este año. Estas preguntas se dividen en dos categorías:

- Preguntas en las que se pedía a los encuestados que estuvieran de acuerdo o en desacuerdo con una afirmación (por ejemplo, “Mi organización tiene un método eficaz para hacer frente a las amenazas a la seguridad” o “Tengo acceso a las herramientas necesarias para ejecutar las pruebas de seguridad”).
- Preguntas en las que se pedía a los encuestados que indicaran cómo están establecidas o no las prácticas de seguridad en su organización (por ejemplo, “Las compilaciones se definen a través de secuencias de comando compiladas y nada más” o “Las versiones de producción se compilan mediante un sistema centralizado de CI/CD, nunca en la estación de trabajo de un desarrollador”). Usamos la escala “establecido/no establecido” porque en las primeras pruebas se comprobó que los encuestados estaban de acuerdo con algunas preguntas relacionadas con la seguridad. Sin embargo, las preguntas sobre el SSDF estaban redactadas de forma más natural con la escala de respuesta de acuerdo/desacuerdo.

El **framework SLSA**, en su versión 0.1 al momento de redactar este documento, describe una serie de prácticas de integridad de la cadena de suministro de software asociadas con “niveles” SLSA, en los que los mayores niveles corresponden a niveles más altos de garantía de seguridad de la cadena de suministro de software. Preguntamos a los encuestados sobre varias de las prácticas particulares asociadas a los SLSA. En concreto, la encuesta preguntaba: “¿En qué medida están establecidas las siguientes prácticas para la aplicación o servicio principal en el que trabajas?” La tabla 1 enumera la redacción de las prácticas relacionadas con los SLSA que se contemplan en la encuesta.

El **SSDF**, que en la actualidad se encuentra en la versión 1.1, se enfoca en las prácticas a fin de ayudar a las organizaciones a enviar el software con menos vulnerabilidades, y para minimizar el impacto potencial de las vulnerabilidades restantes. En lugar de los “niveles” de los SLSA, las prácticas del SSDF se agrupan en cuatro categorías: preparación de la organización, protección del software que se desarrolla, producción de software bien protegido y respuesta eficaz ante las vulnerabilidades descubiertas. En la encuesta, se preguntaba a los encuestados en qué medida estaban de acuerdo (o en desacuerdo) con las afirmaciones que describían varias prácticas del SSDF; estas preguntas se resumen en la tabla 2.



Práctica de SLSA	Definición de la encuesta
CI/CD centralizada	Las versiones de producción se compilan mediante un sistema centralizado de CI/CD, nunca en la estación de trabajo de un desarrollador
Conservación del historial	Las revisiones y el historial de cambios se conservan por tiempo indefinido
Secuencia de comandos de compilación	Las compilaciones se definen en su totalidad a través de la secuencia de comandos de compilación, pero nada más
Aislado	Las compilaciones están aisladas; no pueden interferir en las compilaciones simultáneas o posteriores
Compilación de archivos de texto	Las definiciones y las opciones de configuración se definen en archivos de texto almacenados en un sistema de control de versión
Metadatos de parámetros	Los metadatos compilados (p. ej.: dependencias, procesos de compilación, entornos de compilación) sobre un artefacto incluyen todos los parámetros
Metadatos de dependencias	Los metadatos compilados (p. ej.: dependencias, procesos de compilación, entornos de compilación) sobre un artefacto documentan todas las dependencias
Metadatos generados	Los metadatos compilados (p. ej.: dependencias, procesos de compilación, entornos de compilación) se generan mediante el servicio de compilación o mediante un generador de metadatos compilados que lee el servicio de compilación
Evitar las entradas	Cuando se ejecutan las compilaciones, se evita que los pasos de la compilación carguen las entradas de compilación de forma dinámica (es decir, todas las fuentes y dependencias necesarias se obtienen por adelantado)
Sin edición de los usuarios	Los usuarios de los servicios de compilación no pueden editar los metadatos compilados (p. ej.: dependencias, procesos de compilación, entornos de compilación) de un artefacto
Metadatos disponibles	Los metadatos compilados (p. ej. dependencias, procesos de compilación, entornos de compilación) están disponibles para aquellos que los necesitan (p. ej., a través de una base de datos central), y se entregan en un formato que acepten
Revisión de dos personas	Cada cambio en el historial de una revisión se debe revisar y aprobar de forma individual por dos personas de confianza antes de su presentación
Metadatos firmados	Los metadatos compilados (p. ej.: dependencias, procesos de compilación, entornos de compilación) sobre cómo se produjo un artefacto se firman con mi servicio de compilación

Tabla 1. Preguntas de la encuesta relacionadas con SLSA

Nota: Los encuestados tenían cinco posibles respuestas a cada pregunta: no establecida en absoluto, algo establecida, moderadamente establecida, muy establecida y completamente establecida.

Práctica SSDF	Definición de la encuesta
Revisiones de seguridad	Se realiza una revisión de seguridad para todas las características principales de las aplicaciones en las que trabajo
Análisis y pruebas continuas de código	Realizamos de forma continua análisis y pruebas de código automatizadas o manuales para todas las actualizaciones compatibles con el fin de identificar o confirmar la presencia de vulnerabilidades no detectadas antes
Pruebas de seguridad iniciales	Las pruebas de seguridad se ejecutan al inicio del proceso de desarrollo del software, ya sea por mí o por otro equipo
Abordar las amenazas con eficacia	Mi organización cuenta con un método eficaz para afrontar las amenazas a la seguridad
Integrado al equipo de desarrollo	Los roles de seguridad están integrados en nuestro equipo de desarrollo de software
Requisitos de los documentos	Nuestra organización dispone de procesos para identificar y documentar todos los requisitos de seguridad del software que nuestra organización desarrolla o adquiere (incluidos los de terceros y los de código abierto)
Revisar los requisitos con regularidad	Los requisitos de seguridad se revisan a intervalos regulares (anuales, o antes si es necesario)
Metadatos generados	Los metadatos compilados (p. ej.: dependencias, procesos de compilación, entornos de compilación) se generan mediante el servicio de compilación o mediante un generador de metadatos compilados que lee el servicio de compilación
Integrado en el ciclo de desarrollo	En mi empresa, el protocolo de seguridad del software está integrado sin problema en nuestro proceso de desarrollo
Proceso estándar en los proyectos	En mi empresa, disponemos de un proceso estandarizado para abordar la seguridad del software en todos los proyectos
Supervisar los informes de seguridad	Nos esforzamos por supervisar la información que proviene de fuentes públicas sobre posibles vulnerabilidades en el software que usamos y sus componentes de terceros
Contar con las herramientas necesarias	Tengo acceso a las herramientas necesarias para ejecutar pruebas de seguridad

Tabla 2. Preguntas de la encuesta relacionadas con el SSDF

Nota: Los encuestados tenían siete respuestas posibles para cada pregunta: muy en desacuerdo, en desacuerdo, bastante en desacuerdo, ni de acuerdo ni en desacuerdo, bastante de acuerdo, de acuerdo y muy de acuerdo.

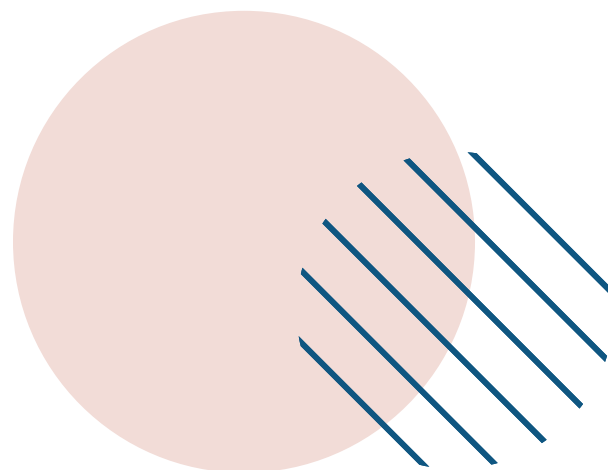


En general, encontramos una incorporación bastante amplia de prácticas del sector emergentes, aunque con mucho margen para que se consoliden. Por ejemplo, mientras que el 66% de los encuestados está de acuerdo con la afirmación “En mi empresa, el protocolo de seguridad del software está integrado sin problema en nuestro proceso de desarrollo”, solo el 18% está muy de acuerdo. Las figuras 1 y 2 resumen las respuestas de los participantes a nuestras preguntas sobre la seguridad.

Comprobamos que el uso de sistemas de integración continua/entrega continua (CI/CD) para las actualizaciones de producción era la práctica más habitual establecida, ya que el 63% de los encuestados dijo que estaba “muy” o “completamente” establecida. El hecho de que CI/CD lidere esta lista se alinea con una investigación previa sobre la seguridad, que descubrió que [la mayoría de las organizaciones implementan el análisis de seguridad a nivel de aplicación como parte de su proceso de CI/CD](#). Además, un conjunto separado de entrevistas cualitativas enfocadas a la seguridad sugirió que la mayoría de los desarrolladores no podían ejecutar tales herramientas a nivel local durante el desarrollo. El framework SLSA también se basa en los sistemas de CI como punto central de integración para la seguridad de la cadena de suministro. Nuestro análisis del modelo, descrito en la siguiente sección, reveló que la presencia de CI en una organización era un indicador

de la madurez de sus prácticas de seguridad. Por lo tanto, creemos que sin esta pieza fundamental de la infraestructura, es muy difícil para una organización garantizar un conjunto coherente de escáneres, linters y pruebas que se ejecutan en los artefactos de software que crean.

Además de CI/CD, otras prácticas habituales establecidas incluyen la conservación indefinida del historial de código (60%), las compilaciones que se definen solo a través de secuencias de comandos (58%), el mantenimiento de las compilaciones aisladas entre sí (57%) y el almacenamiento de las definiciones de las compilaciones en el control de origen (56%). En el extremo inferior, entre las dos prácticas menos establecidas se encuentran la exigencia de que dos o más revisores aprueben cada cambio de código (45%) y la firma de los metadatos compilados para evitar y detectar manipulaciones (41%).



Además de las preguntas sobre las prácticas establecidas, también se les pidió a los participantes que estuvieran de acuerdo o en desacuerdo con una serie de afirmaciones sobre la seguridad en su organización. La afirmación con mayor nivel de acuerdo fue: “Nos esforzamos de forma constante por supervisar la información procedente de fuentes públicas en relación con posibles vulnerabilidades en el software que usamos y los componentes de terceros”, con un 81% de encuestados de acuerdo. Por el contrario, la afirmación con menor proporción de acuerdo hacía referencia a las repercusiones negativas de las prácticas de seguridad en el desarrollo de software: el 56% de los encuestados estaba de acuerdo en que “los procesos de seguridad del software que existen en mi empresa ralentizan el proceso de desarrollo de las aplicaciones en las que trabajo”.

Si bien es motivador que esto tenga el nivel más bajo de acuerdo de los encuestados, el hecho de que la mayoría de ellos diga que los procesos de seguridad actuales ralentizan el desarrollo sugiere que existe un gran margen de mejora en las herramientas y enfoques de seguridad. Nuestro análisis del modelo también respalda esta interpretación y muestra efectos mixtos (aunque menores) en el rendimiento de la entrega de software.



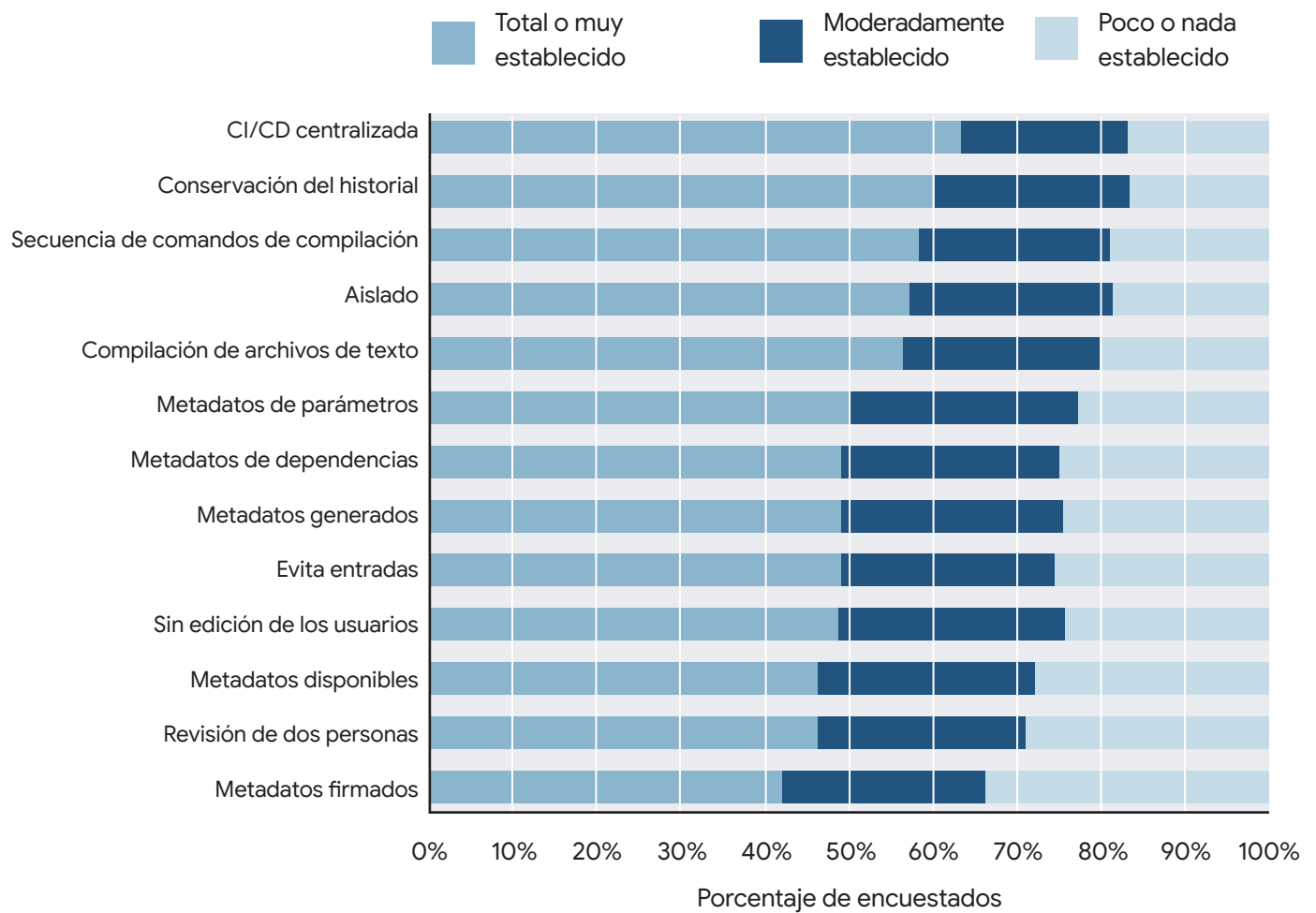


Figura 1. Establecimiento de prácticas de SLSA

Respuestas a la encuesta sobre el establecimiento de prácticas de SLSA. La mayoría de los encuestados indicaron que se habían establecido de alguna manera todas estas prácticas, pero fueron muy pocos los que dijeron que ya estaban establecidas “por completo”.

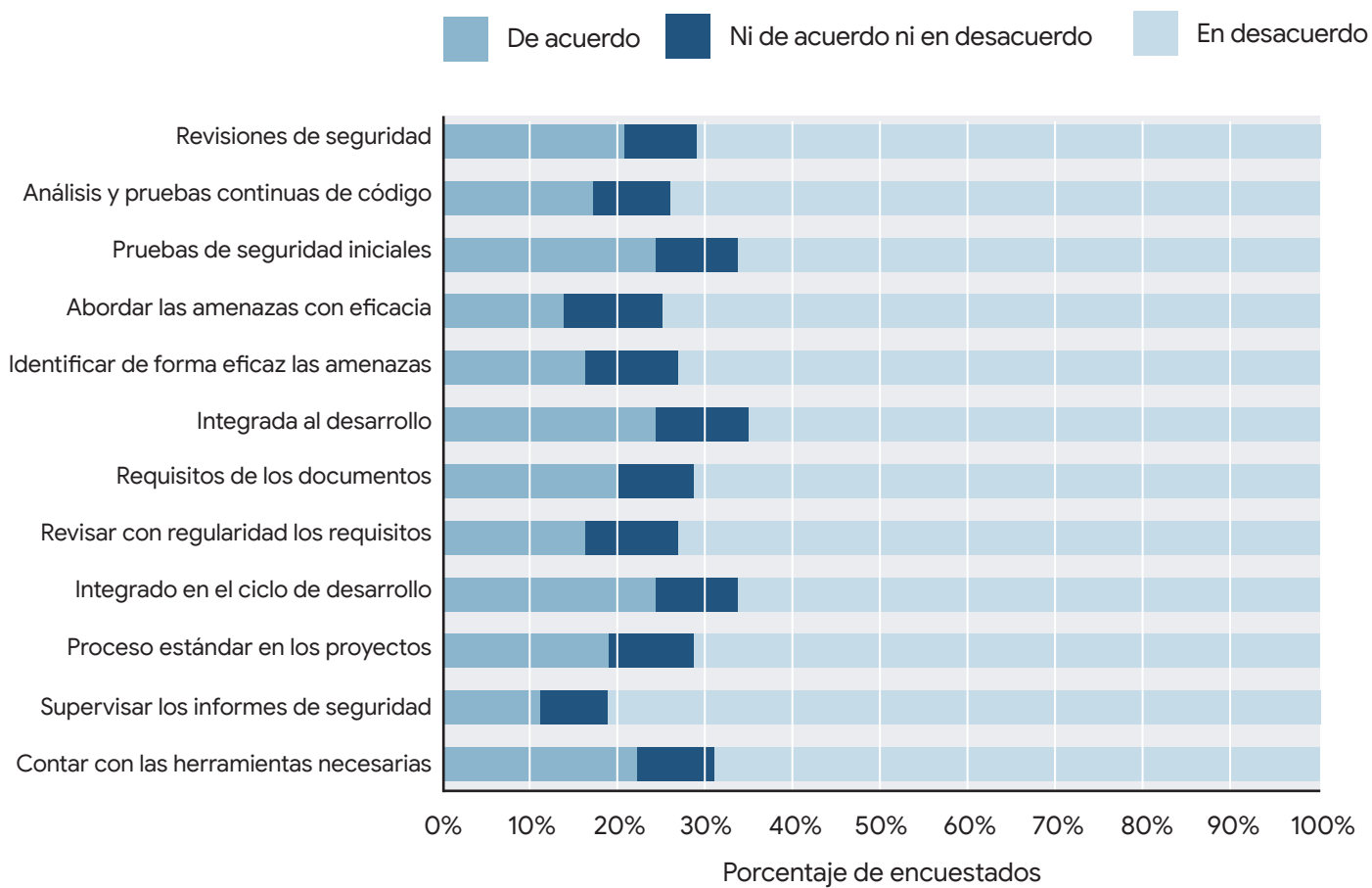


Figura 2. Establecimiento de prácticas de SSDF

Respuestas a la encuesta sobre el establecimiento de prácticas de SSDF. Al igual que el SLISA, la mayoría de los encuestados coincidieron en que su organización seguía todas estas prácticas.

¿Qué es lo que ayuda a las empresas a seguir buenas prácticas de seguridad?

La seguridad para aplicaciones es solo un aspecto del desarrollo de software y, por tanto, una de las numerosas demandas que compiten por el tiempo y la atención de los desarrolladores. Los enfoques de alta fricción de seguridad pueden ser frustrantes para los desarrolladores y poco efectivos en general, ya que se trata de evitar los puntos de fricción. Por ejemplo, un conjunto de entrevistas de investigación con ingenieros de software profesionales reveló que sus puntos de contacto con los equipos de seguridad se limitaban al inicio o al final de un proyecto, y que podía ser difícil relacionarse con los equipos. Según uno de los participantes: “Tenemos un equipo de seguridad para aplicaciones, pero nunca revisaron mi código... soy como la mayoría de los ingenieros, por lo general, los evito”.

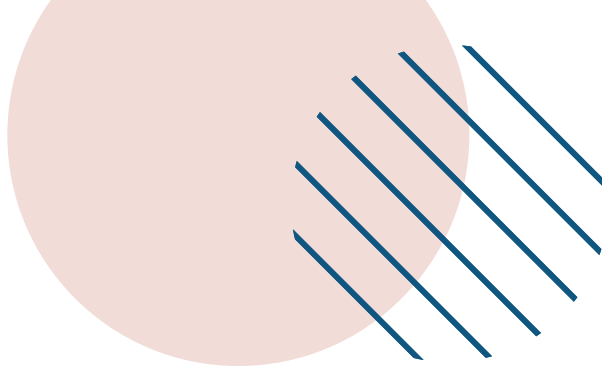
Un enfoque para mejorar la seguridad del software es reducir las barreras a fin de seguir las prácticas de seguridad. Los desarrolladores con los que hablamos querían hacer lo correcto, y a menudo comentaban su frustración porque el envío de características o correcciones tenía siempre prioridad sobre los posibles problemas de seguridad. Por ejemplo, uno de los participantes en una encuesta relacionada con la seguridad describió su mayor desafío en materia de seguridad como “hacer que sea una prioridad en primer lugar. No es atractiva, no vende más productos, [y] no es un problema hasta que se convierte en uno”.

Los datos de nuestra encuesta sugieren que existen varios factores que facilitan a los desarrolladores “hacer el trabajo seguro”.

El factor más importante que encontramos no era en absoluto técnico, sino cultural: **las organizaciones más cercanas al grupo de cultura “generativa” de Westrum tenían muchas más probabilidades de decir que disponían de prácticas de seguridad bien establecidas**, según la definición del framework SLSA¹. Los aspectos de las culturas generativas incluyen ser muy cooperativos, compartir riesgos y responsabilidades, y aprender de los errores del pasado. Nuestra hipótesis es que estas características se manifiestan en prácticas de seguridad más saludables de diversas maneras, como alentar a los ingenieros de software a ser más proactivos con respecto a la seguridad de la cadena de suministro, recompensar a las personas por sus esfuerzos en torno a la seguridad, sin importar su puesto, o reducir los riesgos percibidos de informar sobre posibles problemas de seguridad.

Desde el punto de vista tecnológico, tres de los factores más importantes que impulsan la seguridad están relacionados con la infraestructura. Esto tiene sentido de forma intuitiva: si tu infraestructura facilita la realización de tareas como el análisis de vulnerabilidades o la revisión manual del código, es más probable que tus ingenieros las usen. En concreto, comprobamos que **disponer de sistemas para el control de origen, la integración continua y la entrega continua**

¹ Resulta interesante que estos mismos encuestados no fueran más propensos a estar de acuerdo con las preguntas de NIST SSDF. Aunque SLSA y SSDF analizan diferentes aspectos de la seguridad en el desarrollo de aplicaciones, esperábamos encontrar superposición entre estos conjuntos de preguntas. Como se mencionó antes, es posible que la escala de respuesta de SSDF estuviese favorecida por las respuestas de “acuerdo”, lo que explicaría esta diferencia.



estaba relacionado con tener también prácticas de

SLSA mejor establecidas. Una parte fundamental de esto es probable que sea el momento en que los problemas de seguridad reciben la atención de los desarrolladores, lo cual, según una encuesta independiente, se produce sobre todo durante la CI. Por lo general, la CI precede de forma directa a las revisiones de código, y es cuando se ejecutan los análisis de vulnerabilidad y otras herramientas de análisis de código, ya que garantiza que todas las confirmaciones de código están sujetas a los mismos requisitos de seguridad. La falta de un sistema de compilación centralizado dificulta mucho más el análisis coherente y, la falta de control de código de origen dificulta la existencia de un sistema de compilación centralizado en primer lugar.

Sin embargo, el análisis de seguridad como parte de la CI/CD puede no presentarse lo más rápido posible para los ingenieros de software. En una serie de entrevistas relacionadas con la seguridad con desarrolladores de aplicaciones, nos comentaron en repetidas ocasiones que el análisis de seguridad en su estación de trabajo de desarrollo les ayudaría a ahorrar tiempo y esfuerzo. Se citaron con frecuencia dos situaciones: 1) quieren saber de antemano si compilan sobre una dependencia con vulnerabilidades conocidas, a fin de poder replantearse el uso de esa dependencia antes de compilar sobre ella, y 2) quieren evitar los largos tiempos de espera de CI, a veces medidos en horas, solo para confirmar si sus cambios actuales resuelven un problema de seguridad. En ambos casos, los ingenieros de

software afirmaron que, si bien era necesario un “respaldo” de CI, la capacidad de ejecutar las mismas herramientas de seguridad a nivel local les ayudaría a trabajar con mayor rapidez y eficacia.

Los factores culturales y tecnológicos antes mencionados fueron los mayores impulsores de la seguridad, pero no los únicos. Otros factores destacables incluyen los siguientes:

- Flexibilidad en los acuerdos de trabajo (por ejemplo, ¿apoya la organización el trabajo desde casa?)
- Uso de la nube (pública o privada)
- Trabajar en un servicio o aplicación “nativa de la nube”
- Sentir que la empresa invierte y valora a tu equipo
- Baja rotación de equipo
- Tamaño de la organización, con organizaciones más grandes que registran puntuaciones de seguridad más altas

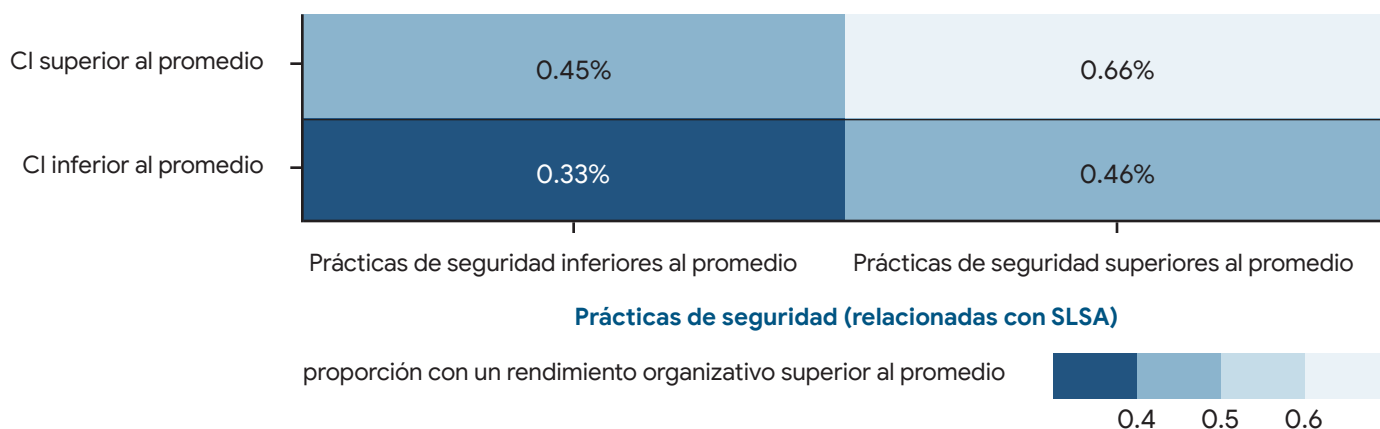
Sin embargo, estos factores parecen correlacionarse en su mayoría con la cultura generativa de Westrum (por ejemplo, obtener acuerdos de trabajo flexibles, sentirse valorado por la organización o tener una rotación baja de equipos) o con el uso de CI/CD (por ejemplo, trabajar en una aplicación nativa de la nube o trabajar en una organización grande). Estos datos nos hacen pensar que la cultura organizativa y los procesos de desarrollo modernos (como la integración continua) son los mayores impulsores de la seguridad en el desarrollo de aplicaciones de una organización, y el mejor punto de partida para las organizaciones que desean aumentar su posición de seguridad.

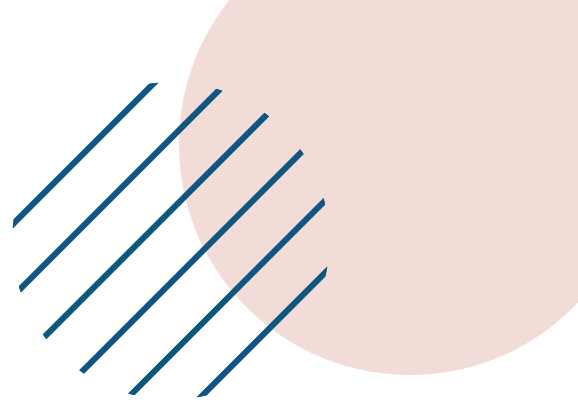
¿A qué resultados conducen las buenas prácticas de seguridad?

A medida que las organizaciones mejoran sus prácticas de seguridad en torno al desarrollo de software, ¿qué beneficios podrían esperar? Los datos de nuestra encuesta confirman que los participantes anticipan una **menor probabilidad de violaciones de la seguridad, interrupciones del servicio y disminución del rendimiento a medida que las empresas aumentan el establecimiento de prácticas de seguridad en la cadena de suministro**. Del mismo modo, en el primer semestre de 2022 realizamos una investigación independiente en la que descubrimos que la ejecución de herramientas como los análisis de vulnerabilidades durante la CI aumentaba en gran medida la probabilidad de identificar vulnerabilidades en las dependencias del software: los encuestados que usaban esas herramientas tenían casi el doble de probabilidades de informar que identificaban una vulnerabilidad de seguridad en su propio código o en una de sus dependencias. En resumen: Las prácticas de SLSA y SSDF parecen funcionar según lo previsto. No pretendemos que

puedan eliminar las amenazas de seguridad, pero nuestras pruebas sugieren que reducen el riesgo de seguridad de una organización.

Las prácticas de seguridad también pueden influir de forma positiva en los resultados basados en el rendimiento, pero existe un detalle: la CI desempeña un rol fundamental. Cuando no se implementa la CI, las prácticas de seguridad no muestran ningún efecto en el rendimiento de la entrega de software. No obstante, cuando se implementa la CI, las prácticas de seguridad producen un fuerte efecto positivo en el rendimiento de la entrega de software. Esto significa en esencia que la CI es necesaria para que las prácticas de seguridad tengan un impacto positivo en el rendimiento de la entrega de software. Además, las prácticas de seguridad suelen tener un efecto positivo en el rendimiento de la organización, y cuando la CI está bien establecida, este efecto se amplía. El siguiente gráfico intenta visualizar este efecto.





Los encuestados también informaron, junto con la reducción de los riesgos de seguridad percibidos, un **menor agotamiento** entre los miembros del equipo y una mayor disposición a recomendar su organización como **un excelente lugar de trabajo**. Ambos resultados demuestran que la seguridad es una tarea más para los ingenieros de software. Las herramientas y los procesos que los ayudan a incorporar prácticas seguras en su flujo de trabajo de desarrollo existente, a diferencia del trabajo no planificado o las “simulacros” cuando se descubre una amenaza, proporcionan un mecanismo para reducir los riesgos de seguridad y aumentar la satisfacción de los desarrolladores.

En conjunto, nuestros datos sugieren que los equipos **saludables y de alto rendimiento también acostumbra a tener buenas prácticas de seguridad** bien establecidas (aunque, como se señaló antes, todavía queda mucho por mejorar). Seguir enfoques como los framework SLSA o SSDF podría no mejorar por sí solo todas las métricas de cultura y rendimiento que medimos, pero es evidente que la seguridad no tiene por qué ir en detrimento de otras prioridades de desarrollo.

Las herramientas y los procesos que los ayudan a incorporar prácticas seguras en su flujo de trabajo de desarrollo existente, en contraposición al trabajo no planificado o a las “medidas extraordinarias” cuando se descubre una amenaza, proporcionan un mecanismo para reducir los riesgos de seguridad y aumentar la satisfacción de los desarrolladores.

05

Sorpresas



Derek DeBellis

Aunque el informe de cada año se centra en las respuestas de la encuesta del año correspondiente, hacemos todo lo posible por comprender estos resultados en el contexto de todo el catálogo de informes sobre State of DevOps y de la investigación complementaria (por ejemplo, la investigación sobre el agotamiento y la cultura). Comprobar la confiabilidad de estos efectos a través de esfuerzos de replicación es un principio básico del programa de investigación. Esto nos permite ajustar nuestras creencias para que se ajusten a los datos y comprender la evolución o las tendencias emergentes.

Este año nos encontramos con algunas sorpresas. Existen múltiples razones que las explican. Primero, la muestra cambió este año para incluir a más personas en las primeras etapas de su carrera que en los informes anteriores. Una de las interpretaciones es que escuchamos más a las personas que son responsables directos de la aplicación de las prácticas y capacidades técnicas, en oposición a las personas que

podrían ser responsables de supervisar o dirigir la implementación de estas prácticas. Otra posibilidad es que exista un cambio en el sector o en el mundo. Ya no se puede garantizar que lo que funcionaba funcionará mañana. Las fuerzas macroeconómicas, y otro año en gran medida a la sombra de la pandemia del COVID-19, por ejemplo, pueden haber cambiado la física de los DevOps. Por último, cambios sutiles en lo que se incluye en nuestro modelo pueden modificar las relaciones entre las variables.¹



¹ “El libro del porqué” de Judea Pearl y “Statistical Rethinking” de Robert McElreath nos presentan increíbles ejemplos de cómo lo que se incluye y lo que no se incluye en los modelos estadísticos puede influir en el resultado del modelo.

Un resultado inesperado o no hipotético pone a los investigadores en una situación difícil a la hora de redactar el informe. Ante el riesgo de que el resultado sea falso o, como mínimo, que aún no se establezca (o incluso se contradiga) la evidencia empírica de varios estudios, lo responsable es realizar una investigación de seguimiento para intentar replicar los resultados y comprender su causa.² Si se enfoca en las sorpresas, un investigador también corre el riesgo de restar importancia a cuántos efectos surgieron de manera confiable a lo largo de años de investigación. Exploramos desde el punto de vista estadístico más de cien rutas de aprendizaje de cada informe State of DevOps. De este modo, se corre el riesgo de que se produzcan hallazgos falsos por simple casualidad. Intentamos contrarrestarlo con esfuerzos de replicación anuales.

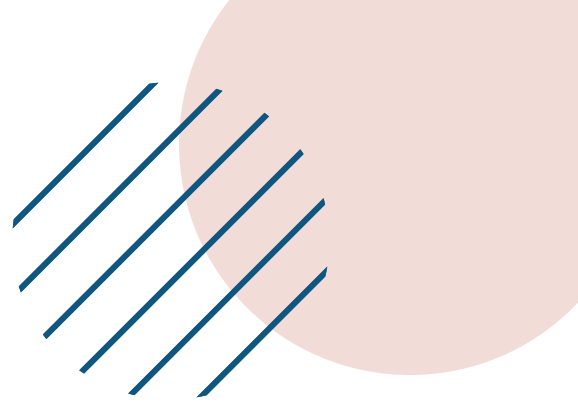
Por otro lado, si no se informa de los resultados, se corre el riesgo de crear un efecto de “archivado”³, lo que hace que lo esperado o agradable se convierta en algo conocido y lo inesperado o difícil de digerir quede oculto. Buscamos un equilibrio: no queremos hacer sensacionalismo de los nuevos hallazgos, pero también creemos que es importante compartirlos. Estas son las cosas que más nos sorprendieron, y lo que pensamos que significan:

01 Observamos con regularidad que las prácticas de desarrollo basadas en troncales tienen un impacto positivo en el rendimiento de la entrega de software. De hecho, esto se ha observado en todos los años del estudio desde 2014. Las capacidades de desarrollo basadas en troncales se comportaron de forma poco habitual este año. Por un lado, las capacidades troncales tuvieron un impacto negativo en el rendimiento de la entrega de software. En los informes de investigación anteriores ocurría lo contrario. Debido a lo aberrante de este resultado, estamos ansiosos por ver si se replica en próximas investigaciones y por saber si la comunidad tiene alguna explicación.

02 Solo comprobamos que el rendimiento de la entrega de software es beneficioso para el rendimiento de la organización cuando el rendimiento operativo también es alto, y muchos encuestados no tenían un alto rendimiento operativo. Esto contradice las iteraciones anteriores de nuestra investigación, en las que la conexión entre el rendimiento de la entrega de software y el rendimiento de la organización era bastante más clara.

² Kerr, N. L. (1998). HARKing: Hypothesizing after the results are known. *Personality and social psychology review*, 2(3), 196-217.

³ Rosenthal, R. (1979). The file drawer problem and tolerance for null results. *Psychological Bulletin*, 86(3), 638.



03 Las prácticas de documentación repercutieron de forma negativa en el rendimiento de la entrega de software. Esto se contradice con los informes anteriores. Una hipótesis es que la documentación se convierte en una práctica cada vez más automatizada, sobre todo entre los equipos de alto rendimiento. Mientras no se recopilen datos adicionales, tenemos pocas pruebas para apoyar o refutar esta creencia.

04 Algunas capacidades tecnológicas (es decir, el desarrollo basado en troncales, la arquitectura con acoplamiento bajo, la CI, la CD) parecían predecir el agotamiento. Como se mencionó antes, muchos de los encuestados de esta muestra se encontraban en una etapa profesional mucho más temprana que los participantes de las muestras de años anteriores. Por lo tanto, es posible que nos dirijamos a las personas responsables de implementar la capacidad en vez de los responsables de crear o supervisar la iniciativa. El proceso de implementación puede ser mucho más difícil que su supervisión. Queremos seguir investigando para entender mejor este resultado.

05 Las prácticas de ingeniería de confiabilidad tuvieron un impacto negativo en el rendimiento de la entrega de software. Una de las explicaciones es que no están relacionadas necesariamente con la causa. Este año notamos en un nuevo análisis de agrupamiento en clústeres (consulta “¿Cómo lo puedes comparar?”) que un subconjunto de clústeres parecía enfocarse en la confiabilidad mientras ignoraba el rendimiento de la entrega de software. Creemos que están separados, en el sentido de que se puede hacer uno sin hacer el otro, pero en última instancia, para que el rendimiento de la entrega de software cuente en términos de rendimiento de la organización, es necesario que exista la confiabilidad.

06 Agregamos prácticas relacionadas con SLSA para comprender si los equipos adoptan estos enfoques a fin de mantener una cadena de suministro de software segura. Aunque esperábamos que hubiera cierta asociación entre la implementación de las prácticas de seguridad y el rendimiento (p. ej., el uso de las capacidades técnicas, el mejor rendimiento de la entrega de software y el mejor rendimiento de la organización), nos sorprendió ver que las prácticas de seguridad eran en realidad el mecanismo a través del que las capacidades técnicas afectaban al rendimiento de la entrega de software y al rendimiento de la organización.

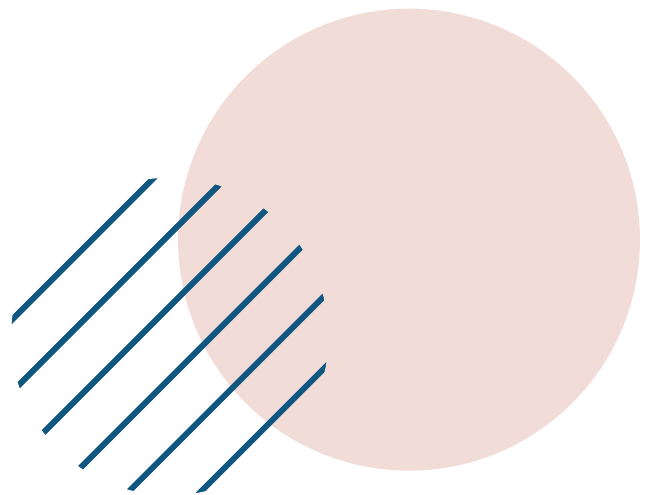


La inclusión de prácticas relacionadas con SLSA parece explicar gran parte del efecto de la integración continua, el control de la versión y la entrega continua en el rendimiento de la entrega de software y el rendimiento de la organización. En otras palabras, hay una cadena causal que se detecta en los datos en los que varias capacidades técnicas tienen un impacto positivo en las prácticas relacionadas con el SLSA y, a través de este impacto positivo en las prácticas relacionadas con el SLSA, tienen un impacto positivo en el rendimiento de la entrega de software y en el rendimiento de la organización. Usamos el análisis de mediación para detectar este resultado.⁴

⁵ Esto nos obliga a explorar si nuestra medida de las prácticas relacionadas con SLSA sigue otras características del equipo (p. ej., el rendimiento general) y de qué manera las prácticas de seguridad producen un mejor rendimiento en la entrega de software y el rendimiento de la organización.

Esperamos volver a estudiar estos efectos el año que viene, para ver si podemos reproducir y explicar estos nuevos patrones, o si debemos inclinarnos por descartarlos como valores atípicos (que también deberíamos intentar explicar). Como siempre, agradecemos los comentarios de la comunidad.

¡Únete a la [Comunidad DORA](http://dora.community) (<http://dora.community>) para continuar el debate sobre estas sorpresas y otras respuestas del informe de este año!



⁴ Jung, Sun Jae. "Introduction to Mediation Analysis and Examples of Its Application to Real-world Data." Journal of preventive medicine and public health = Yebang Uihakhoe chi vol. 54,3 (2021): 166-172. doi:10.3961/jpmph.21.069

⁵ Carrión, Gabriel Cepeda, Christian Nitzl, and José L. Roldán. "Mediation analyses in partial least squares structural equation modeling: Guidelines and empirical examples." Partial least squares path modeling. Springer, Cham, 2017. 173-195.

06

Datos demográficos y organizativos

Gracias por tus contribuciones a nuestra investigación y a nuestro sector.

¿Quién respondió la encuesta?

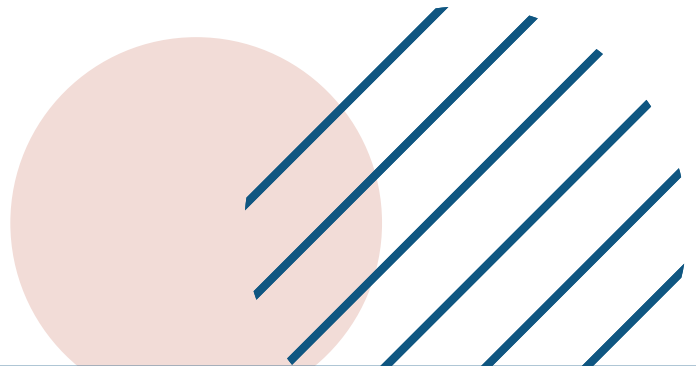
Con ocho años de investigación y más de 33,000 encuestas de profesionales de la industria, el Informe State of DevOps muestra las prácticas de desarrollo de software y DevOps que hacen que los equipos y las organizaciones sean más exitosos. Este año, 1,350 profesionales de una variedad de industrias de todo el mundo compartieron sus experiencias para ayudarnos a comprender los factores que impulsan un mayor rendimiento. Gracias por tus contribuciones a nuestra investigación y a nuestro sector. Para resumir, la representación en las medidas demográficas y organizativas se mantuvo muy coherente.



Derek DeBellis

Al igual que en años anteriores, recopilamos la información demográfica de cada encuestado. Las categorías incluyen género, discapacidad y grupos poco representados.

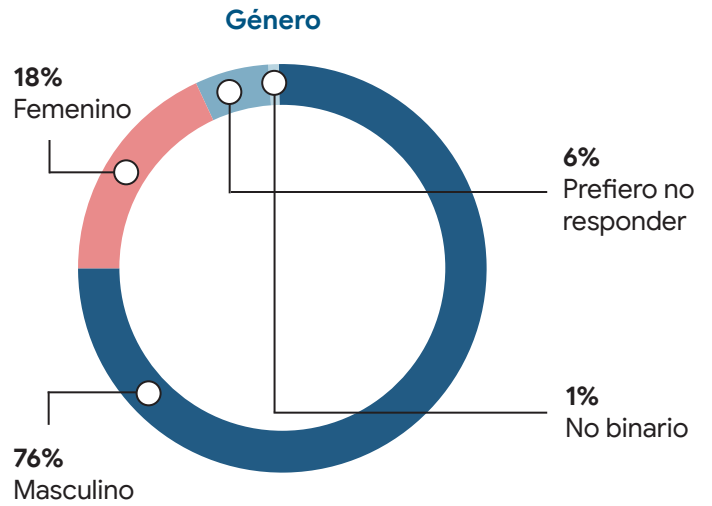
Este año, observamos una representación coherente con los informes anteriores en categorías organizativas, como el tamaño de la empresa, la industria y la región. Una vez más, más del 60% de los encuestados trabaja como ingenieros o administradores, y un tercio trabaja en la industria tecnológica. Además, observamos la representación de empresas de los sectores de servicios financieros, venta minorista, industria y fabricación.



Datos demográficos

Género

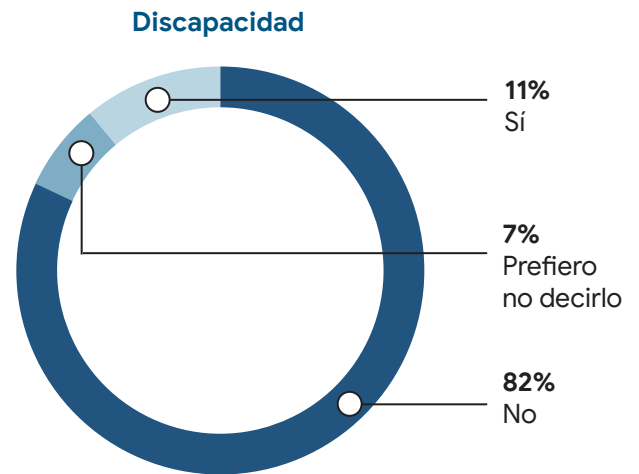
En relación con 2021, la muestra de este año tenía una proporción mayor de mujeres encuestadas (18% en comparación con el 12%). La proporción de hombres encuestados (76%) fue inferior a la de 2021 (83%). Los encuestados afirmaron que las mujeres representan el 25% de sus equipos, lo que coincide con el año 2021 (25%).



Porcentaje de mujeres: **25% Mediana**

Discapacidad

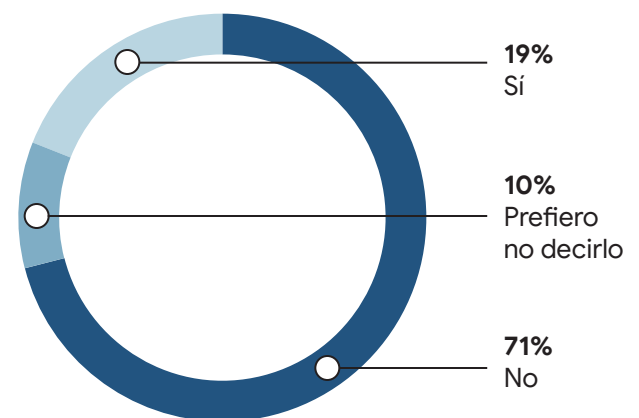
Identificamos la discapacidad en función de seis dimensiones que siguen los lineamientos de la [Washington Group Short Set](#) (Lista breve de preguntas del Grupo de Washington). Este es el cuarto año que preguntamos sobre la discapacidad. El porcentaje de personas con discapacidad fue coherente con nuestro informe de 2021, con un 11%.



Grupo infrarrepresentado

La identificación como miembro de un grupo poco representado puede referirse al origen étnico, el género o alguna otra característica. Este es el cuarto año que preguntamos sobre la poca representación. El porcentaje de personas que se identifican como poco representadas aumentó un poco del 17% en 2021 al 19% en 2022.

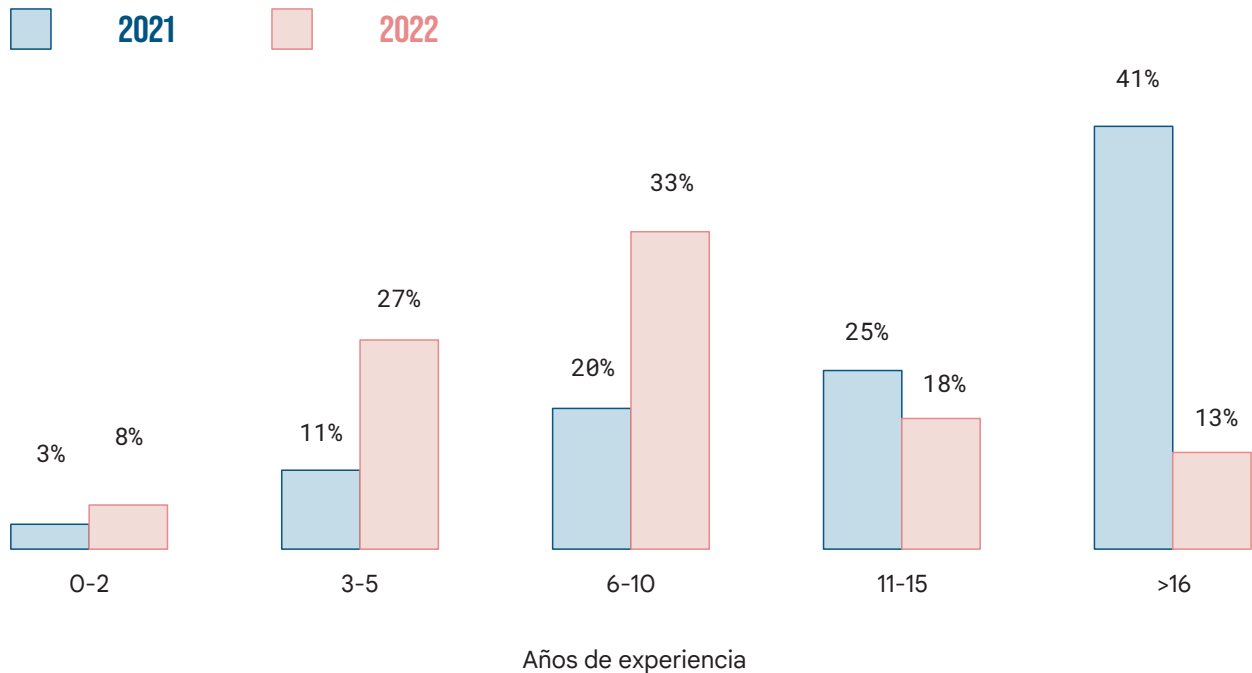
Grupo infrarrepresentado

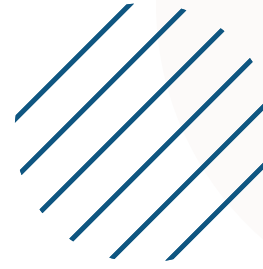


Años de experiencia

Este año hay más encuestados con cinco años de experiencia o menos (35%) que en 2021 (14%). No es sorprendente, entonces, que la proporción de encuestados con más de 16 años de experiencia (13%) sea una fracción de 2021 (41%).

Este cambio puede explicar algunos patrones que aparecen en los datos, y creemos que es importante tenerlo en cuenta cuando se interpretan los resultados, en especial al compararlos con los del año pasado.

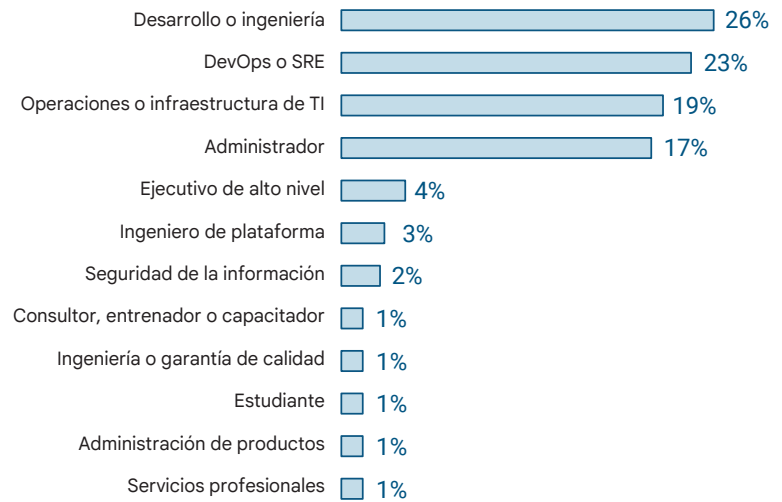




Datos organizativos

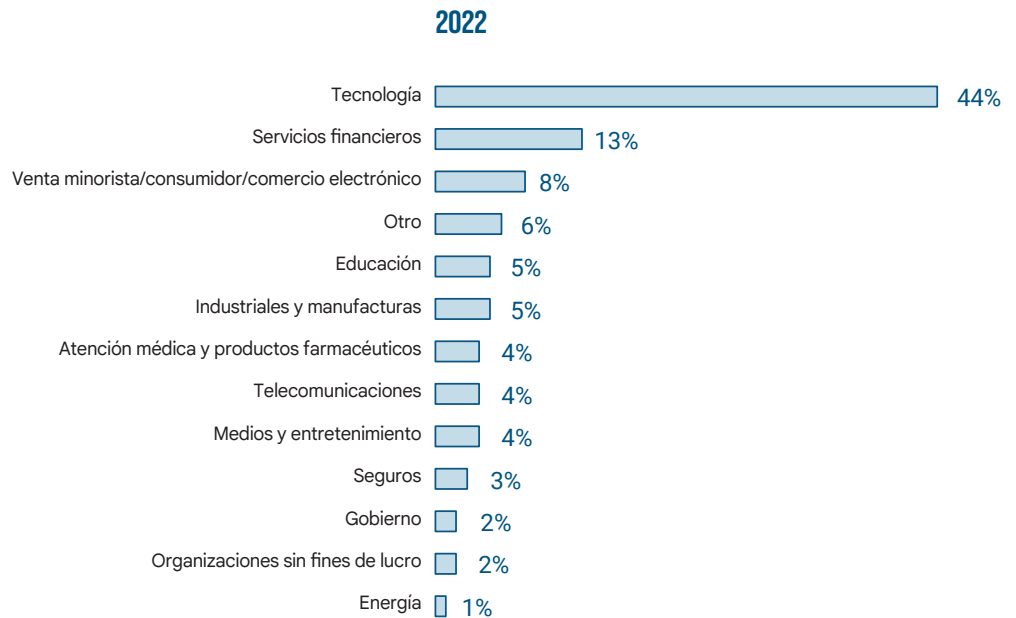
Rol

El 85% de los encuestados son personas que trabajan en equipos de desarrollo o ingeniería (26%), trabajan en equipos de DevOps o SRE (23%), trabajan en equipos de operaciones de TI o infraestructura (19%), o son gerentes (17%). La proporción de encuestados que trabajan en equipos de operaciones o infraestructuras de TI (19%) duplicó con creces la proporción del año pasado (9%). Los ejecutivos de alto nivel (9% en 2021 hasta el 4%), y los servicios profesionales (4% en 2021 hasta el 1%) son dos de las disminuciones más pronunciadas en relación con el año pasado.



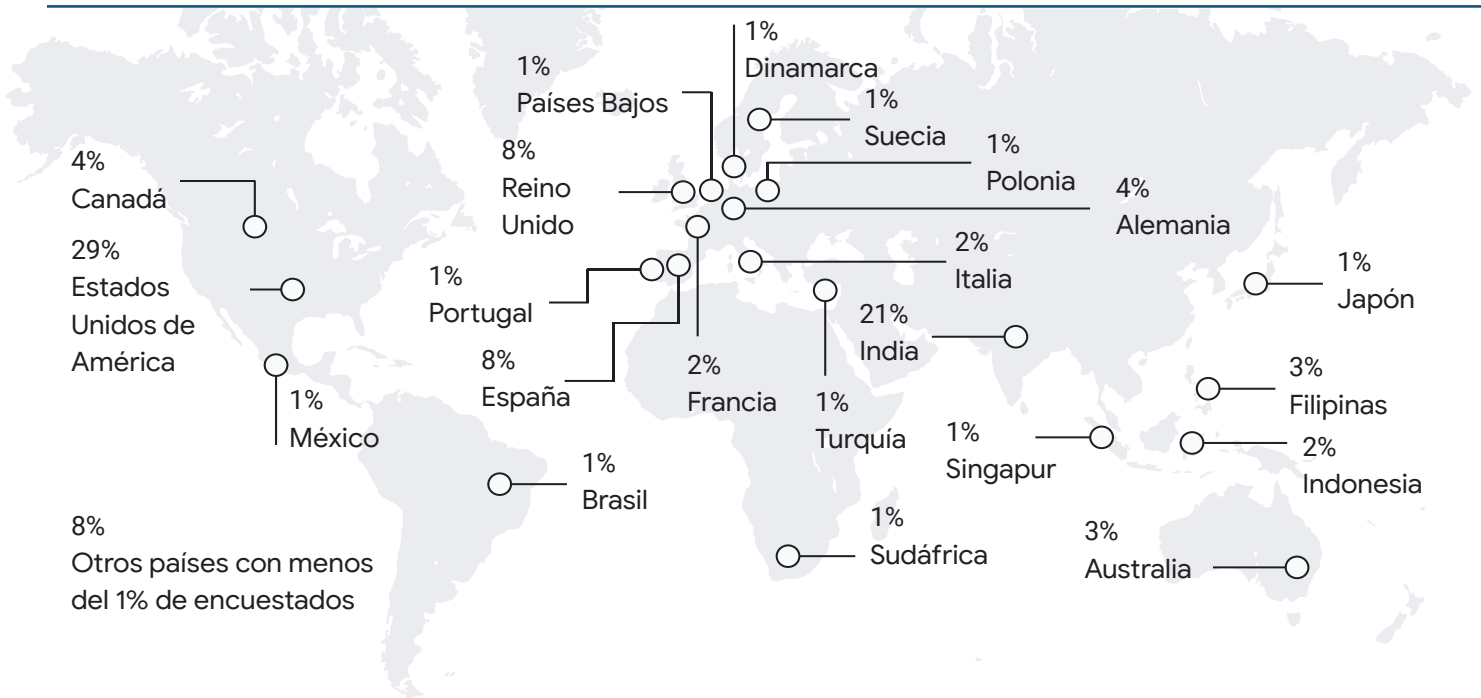
Sector

Al igual que en anteriores informes State of DevOps, se observa que la mayoría de los encuestados trabajan en el sector tecnológico, seguido de los servicios financieros, otros y la venta minorista.



Región

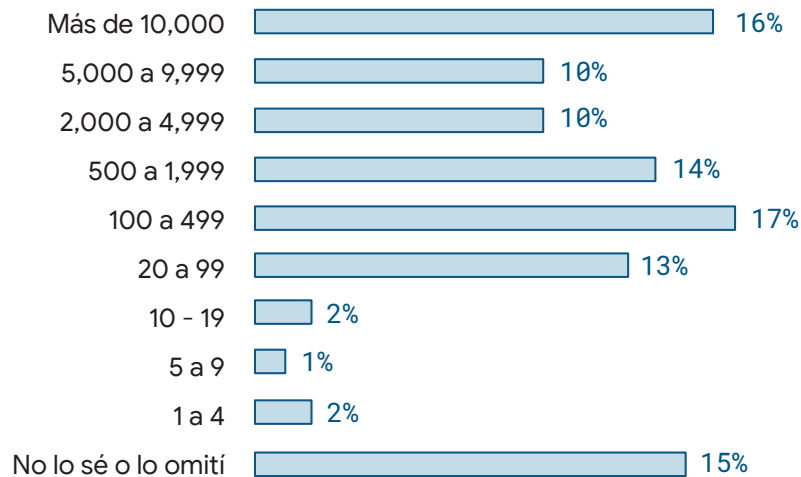
Este año pedimos a los encuestados que seleccionaran el país del que procedían en lugar de la región. La región, que con frecuencia estaba representada por un continente, parecía demasiado tosca para comprender la composición de nuestros encuestados. Recibimos respuestas de participantes de más de 70 países. El 89% de los encuestados provenían de 22 países.

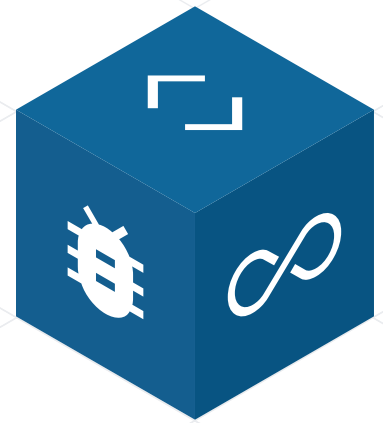


Cantidad de empleados

En coherencia con anteriores encuestas State of DevOps, los encuestados proceden de organizaciones de diversos tamaños. El 22% de los encuestados se encuentra en empresas con más de 10,000 empleados y el 7% en empresas con entre 5,000 y 9,999 empleados. Otro 15% de los encuestados pertenecen a organizaciones de entre 2,000

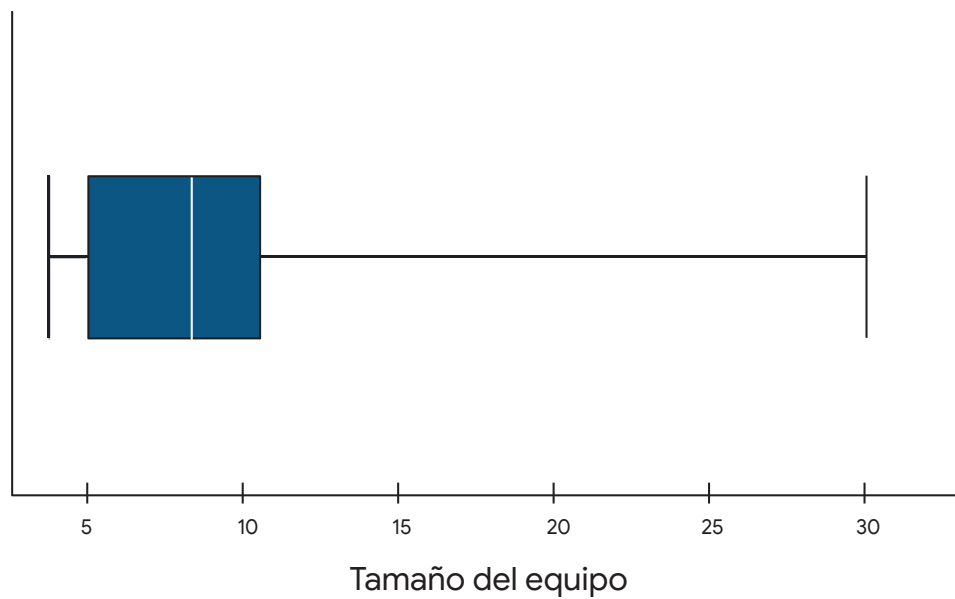
y 4,999 empleados. También observamos una buena representación de encuestados (13%) de organizaciones de entre 500 y 1,999 empleados, un 15% de entre 100 y 499 empleados y, por último, un 15% de entre 20 y 99 empleados. Este año, además, se permitió a los encuestados seleccionar “no sé” sobre el tamaño de su organización. El 15% de los encuestados dijo no saber o, también omitió la pregunta.





Tamaño del equipo

Este año les pedimos a los participantes que indicaran la cantidad aproximada de personas en su equipo. El 25% de los encuestados trabaja en equipos de hasta 5 personas. El 50% de los encuestados trabaja en equipos de hasta 8 personas. El 75% de los encuestados trabaja en equipos de hasta 12 personas.

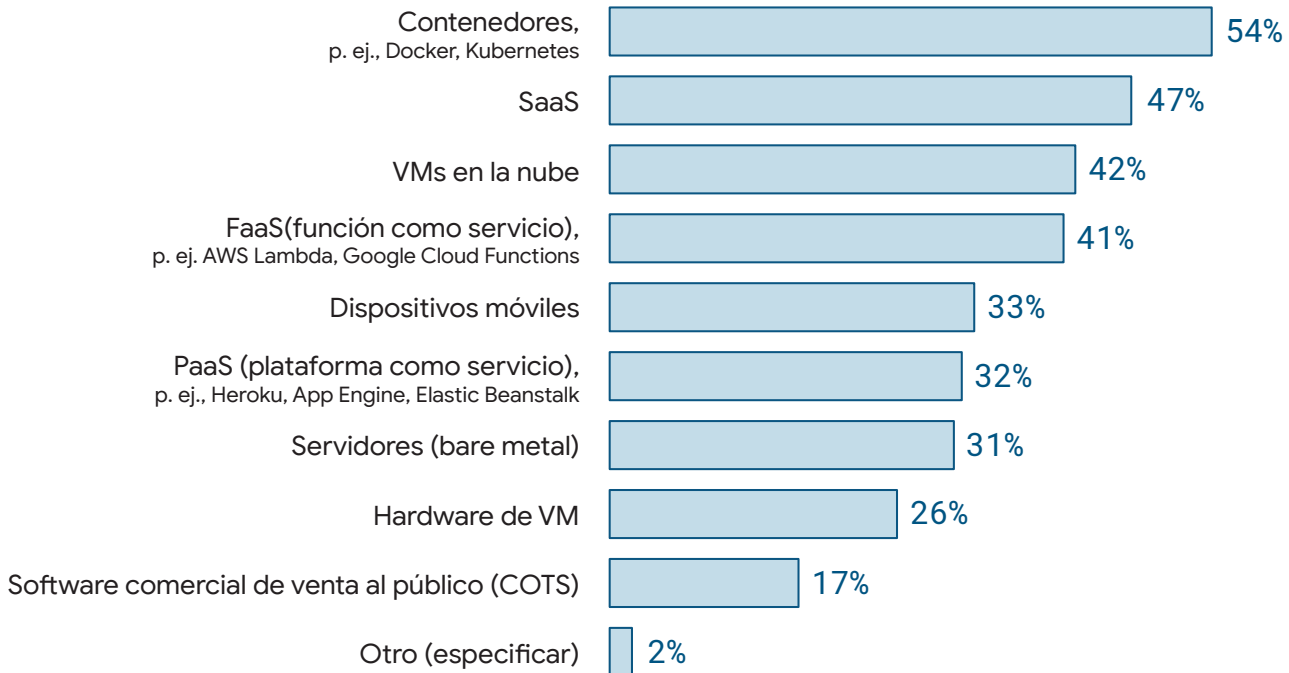


Destino de implementación

El año 2021 fue el primero en el que decidimos analizar dónde implementan los encuestados el principal servicio o aplicación en la que trabajan. Para nuestra sorpresa, el objetivo de implementación principal fueron los

contenedores. Este año no fue diferente, aunque en una proporción menor (54%) que el año pasado (64%).

También agregamos más opciones con la esperanza de ofrecer a los encuestados más formas de reflejar el lugar donde realizan las implementaciones.



07

Conceptos finales



Derek DeBellis

Todos los años que elaboramos este informe, nos esforzamos por ofrecer una descripción rigurosa de cómo las prácticas y las capacidades generan resultados críticos para el negocio, como el rendimiento de la organización. Evaluamos la replicación de muchos efectos en informes anteriores, y ampliamos el alcance de nuestro trabajo para considerar las prioridades emergentes en el espacio de DevOps. Este año diseñamos nuestra encuesta y análisis para profundizar en las prácticas de seguridad y modificamos nuestro enfoque de modelos estadísticos a fin de explorar la condicionalidad o las dependencias de ciertos efectos. También exploramos nuevas formas de describir el entorno de la entrega de software y el rendimiento operativo.

En muchos sentidos, la narrativa que se materializó este año es un eco de los años anteriores: las capacidades técnicas se compilan para crear un mejor rendimiento. Existen muchos beneficios inherentes al uso de la nube, la cultura del lugar de trabajo y la flexibilidad conducen a un mejor rendimiento de la organización y el agotamiento de los empleados impide que las organizaciones alcancen sus objetivos. El análisis de las interacciones que agregamos de forma explícita al modelo nos ayudó a comprender las condiciones en las que pueden producirse determinados efectos.

Por ejemplo, el rendimiento de la entrega de software solo parece tener un impacto positivo en el rendimiento de la organización cuando el rendimiento operativo (confiabilidad) es alto, lo que lleva a la conclusión de que se necesitan ambos para prosperar como organización. También hubo algunas sorpresas, que destacamos en una sección específica.

Agradecemos a todas las personas que participaron en la encuesta de este año. Esperamos que nuestra investigación los ayude a ti y a tu organización a crear mejores equipos y mejor software, al mismo tiempo que mantienen el equilibrio entre la vida laboral y personal.

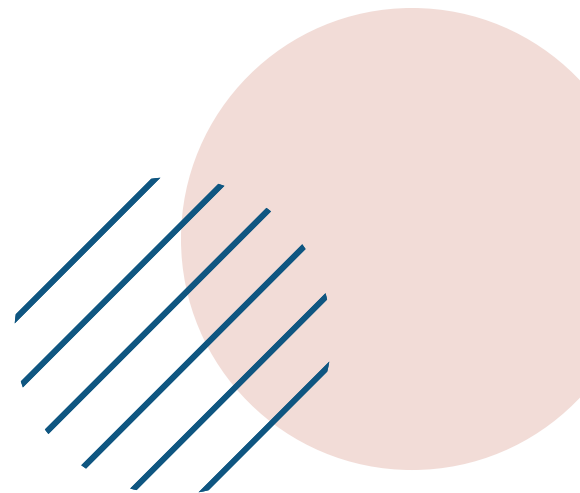


08

Agradecimientos

Una gran familia de colaboradores apasionados hizo posible el informe de este año. El diseño de las preguntas de la encuesta, el análisis, la redacción, la edición y el diseño del informe son solo algunas de las formas en las que nuestros colegas nos ayudaron a llevar a cabo este gran esfuerzo. Los autores desean agradecer a todas estas personas sus aportes y orientación en el informe de este año. Todos los agradecimientos se muestran en orden alfabético.

Scott Aucoin	Eric Maxwell
Alex Barrett	John Speed Meyers
James Brookbank	Steve McGhee
Kim Castillo	Jacinda Mein
Lolly Chessie	Alison Milligan
Jenna Dailey	Pablo Pérez Villanueva
Derek DeBellis	Claire Peters
Rob Edwards	Connor Poske
Dave Farley	Dave Stanke
Christopher Grant	Dustin Smith
Mahshad Haeri	Seth Vargo
Nathen Harvey	Daniella Villalba
Damith Karunaratne	Brenna Washington
Todd Kulesza	Kaiyuan “Frank” Xu
Amanda Lewis	Nicola Yap
Ian Lewis	



09

Autores



Claire Peters

Claire Peters es investigadora de experiencia del usuario en Google. Su investigación abarca diversos aspectos y expresiones de DORA en entornos aplicados. Estudia el Programa de modernización de aplicaciones en la nube de Google Cloud (CAMP), los acuerdos con los clientes basados en DORA y las herramientas relacionadas con las Cuatro Claves, con el objetivo de ayudar a los equipos y a las personas a aplicar los principios de DORA de forma más eficaz en su trabajo diario. Claire también es miembro del equipo principal de investigación de DORA, construye la encuesta anual de DORA y el informe State of DevOps. Tiene una maestría en Análisis Cultural Aplicado de University of Copenhagen.



Dave Farley

Dave Farley es el director general y fundador de Continuous Delivery Ltd y creador del canal de YouTube de Continuous Delivery. Dave es coautor del exitoso libro Continuous Delivery. También es el autor más vendido de Modern Software Engineering: Doing What Works to Build Better Software Faster (Hacer lo que funciona para construir un mejor software más rápido). Es coautor del Manifiesto Reactivo y ganador del Premio Duke por el proyecto de código abierto LMAX Disruptor. Dave es el pionero de la entrega continua, líder de pensamiento y experto en CD, DevOps, TDD y diseño de software, además de tener una larga trayectoria en la creación de equipos de alto rendimiento, la configuración de organizaciones para el éxito y la creación de software excepcional. Encuentra más información sobre Dave en su [Twitter](#), [YouTube](#), [blog](#), y [sitio web](#).



Daniella Villalba

Daniella Villalba es investigadora de la experiencia del usuario dedicada al proyecto DORA. Se enfoca en comprender qué factores favorecen la satisfacción y productividad de los desarrolladores. Antes de Google, Daniella estudió los beneficios de la capacitación en meditación, los factores psicosociales que afectan las experiencias de los alumnos universitarios, la memoria de los testigos presenciales y las confesiones falsas. Obtuvo un doctorado en psicología experimental de Florida State University.



Dave Stanke

Dave Stanke es ingeniero de relaciones con desarrolladores en Google, donde asesora a los clientes sobre las prácticas recomendadas para adoptar DevOps y SRE. Durante su carrera, ocupó una serie de cargos muy variada, que incluye los de director de tecnología para startups, gerente de producto, agente de asistencia al cliente, desarrollador de software, administrador de sistemas y diseñador gráfico. Tiene una maestría en administración de tecnología de Columbia University.



Derek DeBellis

Derek DeBellis es investigador cuantitativo de la experiencia del usuario en Google. En Google, Derek se encarga de la investigación de encuestas, el análisis de registros y la búsqueda de formas de medir conceptos fundamentales para el desarrollo de productos. Derek acaba de publicar sobre la interacción entre los humanos y la IA, el impacto del COVID-19 en el abandono del tabaquismo, el diseño de los errores de la PLN y el rol de la experiencia del usuario en las discusiones sobre la privacidad.



Eric Maxwell

Eric Maxwell encabeza la práctica de transformación digital DevOps de Google, en la que asesora a las mejores empresas del mundo sobre cómo ser aún mejores, un poco a la vez, de forma continua. Eric dedicó la primera mitad de su carrera a ser ingeniero, en el terreno, automatizó todas las tareas y generó empatía con otros profesionales. Eric fue el cofundador del Programa de modernización de aplicaciones en la nube de Google Cloud (CAMP), es miembro del equipo principal de DORA y autor de la guía DevOps Enterprise. Antes de Google, Eric se dedicó a preparar cosas increíbles con otros colegas de Chef Software.



John Speed Meyers

John Speed Meyers es un científico de datos de seguridad en Chainguard, una startup de seguridad de la cadena de suministro de software. Los proyectos de investigación de John incluían temas como la seguridad de la cadena de suministro de software, la seguridad del software de código abierto y la respuesta del mundo al creciente poder militar chino. Antes, John trabajó en In-Q-Tel, RAND Corporation y Center for Strategic and Budgetary Assessments. John tiene un doctorado en análisis político de Pardee RAND Graduate School, un máster en asuntos públicos (MPA) de School of Public and International Affairs de Princeton y una licenciatura en relaciones internacionales de Tufts University.



Kaiyuan “Frank” Xu

Kaiyuan “Frank” Xu es investigador cuantitativo de la experiencia del usuario en Google. Analiza los datos de los registros y de las encuestas a fin de comprender los patrones de uso y los comentarios de los usuarios de los productos de Google Cloud con el fin de mejorar la excelencia en el uso de los productos para los desarrolladores. Antes de Google, Kaiyuan realizó investigaciones cualitativas y cuantitativas de usuarios durante años en Microsoft para los productos Azure y Power Platform. Obtuvo un máster en Ingeniería y Diseño centrado en el ser humano en University of Washington.



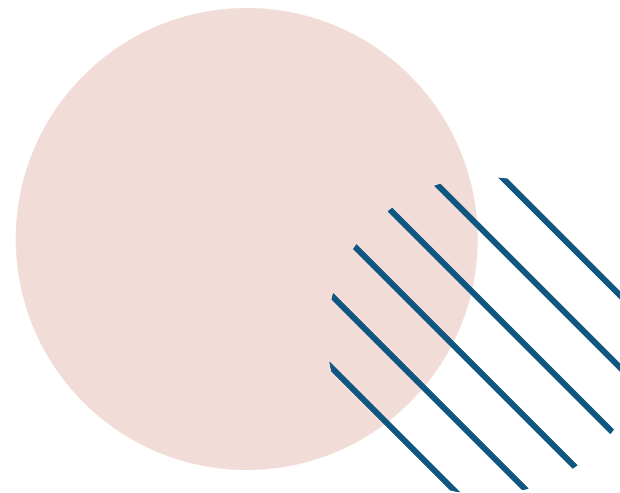
Nathen Harvey

Nathen Harvey es ingeniero de relaciones con desarrolladores en Google y desarrolló su carrera sobre la base de ayudar a los equipos a alcanzar su potencial mientras alinean la tecnología con los resultados comerciales. Nathen tuvo el privilegio de trabajar con algunos de los mejores equipos y comunidades de código abierto para ayudarlos a aplicar los principios y las prácticas de DevOps y SRE. Nathen coeditó y contribuyó a la publicación de *“97 Things All Cloud Engineer Should Know”* (97 cosas que todo ingeniero de la nube debe saber), O'Reilly, 2020.



Todd Kulesza

Todd Kulesza es investigador de la experiencia del usuario en Google, donde estudia cómo trabajan los ingenieros de software en la actualidad y cómo podrían trabajar mejor en el futuro. Tiene un doctorado en Informática de Oregon State University.



10

Metodología

Diseño de la investigación

Este estudio emplea un diseño transversal, basado en la teoría, conocido como **predictivo inferencial**, uno de los tipos más comunes de diseños realizados en la investigación empresarial y tecnológica actual. El diseño predictivo inferencial se emplea cuando el diseño totalmente experimental no es práctico o es imposible.

Población objetivo y muestreo

La población de segmentación de esta encuesta eran profesionales y líderes que trabajaban en la tecnología y las transformaciones, o en campos muy relacionados, y, sobre todo, profesionales con conocimientos de DevOps. La encuesta se promocionó mediante listas de direcciones de correo electrónico, promociones en línea, un panel en línea y redes sociales, y pedimos a las personas que compartieran la encuesta con sus redes (es decir, muestreo de bola de nieve).

Creación de construcciones latentes

Formulamos nuestras hipótesis y construcciones con construcciones validadas con anterioridad siempre que fuera posible. Desarrollamos construcciones nuevas basadas en teoría, definiciones y aportes de expertos. Luego, tomamos medidas adicionales para aclarar la intención a fin de garantizar que los datos recopilados en la encuesta tuvieran una alta probabilidad de ser confiables y válidos.¹

Cálculo de las diferencias entre los que tienen un rendimiento bajo y los que tienen un rendimiento alto

En la sección “¿Cómo lo puedes comparar?” comparamos a los que tienen un rendimiento bajo y a los que tienen un rendimiento alto en las cuatro métricas de rendimiento de la entrega. El método es sencillo. Tomemos como ejemplo la frecuencia de una implementación. El clúster alto se implementa a pedido (es decir, varias veces por día). Si realizan un promedio de cuatro implementaciones al día, esto supone 1,460 implementaciones al año (4 x 365). Por el contrario, los que tienen un rendimiento más bajo se implementan entre una vez al mes y una vez cada seis meses, con una frecuencia media de implementación de una vez cada 3.5 meses, y una frecuencia media de implementación de aproximadamente 3.4 implementaciones al año (12/3.5). Tomamos la proporción (1,460 implementaciones altas / 3.4 implementaciones bajas) y obtenemos 417 veces. Este enfoque se generaliza para las demás métricas de rendimiento del desarrollo.

¹ Churchill Jr, G. A. “A paradigm for developing better measures of marketing constructs”, Journal of Marketing Research 16:1, (1979), 64–73.

Métodos de análisis estadístico

Análisis de clústeres

Para las dos soluciones de agrupamiento de clústeres descritas en la sección “¿Cómo lo puedes comparar?”, usamos el agrupamiento en clústeres jerárquico con una versión del método de aglomeración de Ward² a fin de evaluar lo bien que se adaptan a los datos las distintas soluciones de clúster.

En los primeros resultados de agrupamiento en clústeres que presentamos, buscamos clústeres de respuestas a través de la frecuencia de implementación, el plazo de entrega, el tiempo para restablecer un servicio y la tasa de errores de los cambios. Este año encontramos tres clústeres después de evaluar 14 soluciones diferentes de agrupamiento en clústeres jerárquico con 30 índices diferentes para determinar la cantidad de clústeres.³

El segundo análisis de clústeres que presentamos seguía la misma metodología que el primero, pero se implementó sobre diferentes dimensiones de los datos. Queríamos

buscar patrones de respuesta comunes (es decir, clústeres) en la capacidad de procesamiento (un compuesto de la frecuencia de implementación y el plazo de entrega), el rendimiento operativo (confiabilidad) y la estabilidad (un compuesto del tiempo para restablecer el servicio y la tasa de errores de cambio). También exploramos diferentes algoritmos de agrupamiento en clústeres para ver la sensibilidad de los resultados a nuestro enfoque.

Si bien no existe una forma establecida de cuantificar esa sensibilidad (que conozcamos), los agrupamientos en clústeres que surgieron tendieron a tener características similares.

Modelo de medición

Antes de realizar el análisis, identificamos las construcciones mediante un análisis de factores exploratorios con un análisis de componentes principales con rotación varimax.⁴ Confirmamos las pruebas estadísticas de validez convergente y divergente y la confiabilidad mediante la variación promedio extraída (AVE), la correlación, el alfa de Cronbach⁵, ρA^6_A , la proporción entre heterorasgo y monorasgo^{5,7}, y la confiabilidad compuesta.

² Murtagh, Fionn, y Pierre Legendre. “Ward’s hierarchical agglomerative clustering method: which algorithms implement Ward’s criterion?” (El método de agrupación aglomerativa jerárquica de Ward: ¿qué algoritmos aplican el criterio de Ward?). *Journal of classification* 31.3 (2014): 274-295.

³ Charrad M., Ghazzali N., Boiteau V., Niknafs A. (2014). “NbClust: An R Package for Determining the Relevant Number of Clusters in a Data Set.” (NBCLust: Un paquete de R para determinar el número relevante de clústeres en un conjunto de datos), *Journal of Statistical Software*, 61(6), 1-36.”, “URL <http://www.jstatsoft.org/v61/i06/>”

⁴ Straub, D., Boudreau, M. C., y Gefen, D. (2004). Validation guidelines for IS positivist research (Lineamientos de validación para la investigación positivista de IS). *Communications of the Association for Information systems*, 13(1), 24.

⁵ Nunnally, J.C. *Psychometric Theory*. Nueva York: McGraw-Hill, 1978

⁶ Hair Jr, Joseph F., et al. “Partial least squares structural equation modeling (PLS-SEM) using R: A workbook.” (Modelo de ecuaciones estructurales por el método de mínimos cuadrados parciales (PLS-SEM) con R: una hoja de cálculo) (2021): 197.

⁷ Brown, Timothy A., and Michael T. Moore. “Confirmatory factor analysis” (Análisis factores de confirmación). *Manual de modelos de ecuaciones estructurales* 361 (2012): 379.

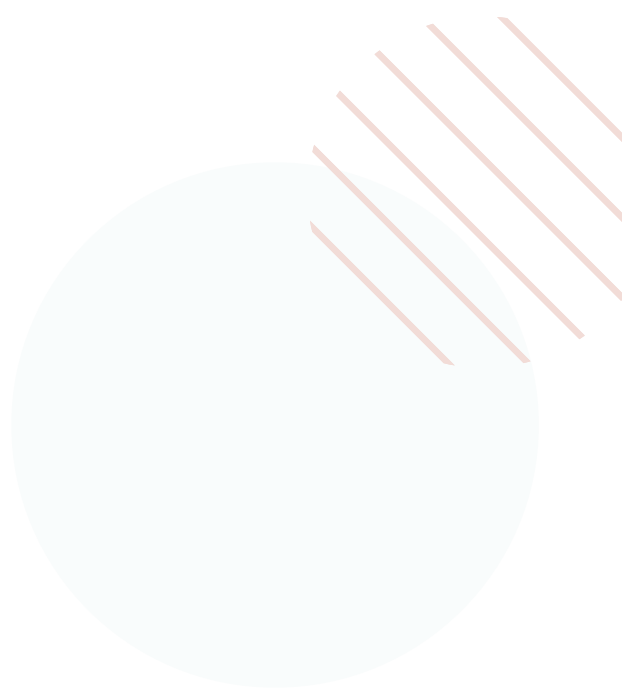


Modelados de ecuación estructural

Probamos los modelos de ecuaciones estructurales (SEM) mediante el análisis de mínimos cuadrados parciales (PLS)⁸, que es un modelo de ecuaciones estructurales basado en la correlación.

Análisis del segundo modelo de clústeres

Para entender qué predice la membresía de un clúster, empleamos la regresión logística multinomial.⁹ Usamos este enfoque porque estábamos tratando de predecir la membresía de un clúster, que, en este caso, son datos categóricos no ordenados con más de dos niveles. A fin de comprender los resultados que predecía la pertenencia a un clúster, usamos una regresión lineal para cada resultado (agotamiento, trabajo no planificado y rendimiento organizativo).



⁸ Hair Jr, J. F., Hult, G. T. M. Ringle, C. M. y Sarstedt, M. (2021). "A primer on partial least squares structural equation modeling (PLS-SEM)" (Manual de modelos de ecuaciones estructurales por mínimos cuadrados (PLS-SEM)). Sage publications.

⁹ Ripley, Brian, William Venables y Maintainer Brian Ripley. "Package 'nnet'". Versión del paquete R 7.3-12 (2016): 700.

11

Lecturas adicionales

Únete a DORA.community para debatir, aprender y colaborar a fin de mejorar la entrega de software y el rendimiento de las operaciones.

<http://dora.community>

Obtén más información sobre las cuatro métricas clave

<https://goo.gle/four-keys>

Obtén más información sobre las capacidades de DevOps.

<https://goo.gle/devops-capabilities>

Obtén más información sobre cómo puedes implementar las prácticas DORA en tu organización, a partir de nuestra guía para empresas

<https://goo.gle/enterprise-guidebook>

Encuentra recursos sobre la ingeniería de confiabilidad de sitios (SRE)

<https://sre.google>

<https://goo.gle/enterprise-roadmap-sre>

Realiza la Verificación rápida de DevOps

<https://goo.gle/devops-quickcheck>

Explora el programa de investigación de DevOps

<https://goo.gle/devops-research>

Conoce a otras empresas que implementaron las prácticas DORA mediante la lectura del libro electrónico ganador del premio Google Cloud DevOps Award

<https://goo.gle/devops-awards>

Obtén más información sobre el Programa de modernización de aplicaciones en la nube de Google Cloud o CAMP.


<https://goo.gle/3daLa9s>

Aprovechar los datos y las métricas DORA para transformar los procesos tecnológicos

<https://goo.gle/3Doh8Km>

Leer el informe: “El ROI de la transformación de DevOps: Cómo cuantificar el impacto de tus iniciativas de modernización” de Forsgren, N., Humble, J., y Kim, G. (2018).

<https://goo.gle/3qECllh>



Leer el libro: *Accelerate: The science behind devops: Building and scaling high performing technology organizations*. (La ciencia detrás de devops: compilar y escalar organizaciones tecnológicas de alto rendimiento). IT Revolution.
<https://itrevolution.com/book/accelerate>

Obtén más información sobre el framework de niveles de la cadena de suministro para artefactos seguros (SLSA)
<https://slsa.dev>

Obtén más información sobre Framework de Desarrollo Seguro de Software (NIST SSDF)
<https://goo.gle/3qBXLWk>

Obtén más información sobre la cultura de DevOps: Cultura organizativa de Westrum
<https://goo.gle/3xq7KBV>

Obtén más información sobre Open Source Security Foundation <https://openssf.org/>

Obtén más información sobre in-toto
<https://in-toto.io/>

Obtén más información sobre la lista de materiales de software de NTIA.gov <https://www.ntia.gov/SBOM>

Seguridad cibernética: Respuesta federal a los incidentes de SolarWinds y Microsoft Exchange
<https://www.gao.gov/products/gao-22-104746>

Obtén más información sobre el análisis de seguridad a nivel de aplicación como parte de su proceso de CI/CD
<https://go.dev/blog/survey2022-q2-results#security>

Por último, aunque no menos importante, consulta los informes anteriores de State of DevOps. Todos ellos se encuentran en <https://goo.gle/dora-sodrs>:

[Informe Accelerate State of DevOps de 2014](#)

[Informe Accelerate State of DevOps de 2015](#)

[Informe Accelerate State of DevOps de 2016](#)

[Informe Accelerate State of DevOps de 2017](#)

[Informe Accelerate State of DevOps de 2018](#)

[Informe Accelerate State of DevOps de 2019](#)

[Informe Accelerate State of DevOps de 2021](#)

