TECNOLOGÍA | PRUEBAS DE SEGURIDAD

**El “Bug Bounty” en la seguridad informática   
de las grandes empresas**

*El Bug Bounty se puede definir como una modalidad de seguridad ofensiva en el que una organización ofrece recompensa a aquellos hackers que encuentren diferentes vulnerabilidades en su infraestructura.*

**Por Julián M. Zappia.  
Diplomatic Week.**

A través de un modelo de seguridad basado en el Crowdsourcing, las compañías logran atraer a grandes talentos de todo el mundo para que verifiquen si sus sistemas están seguros.

Para los hackers, se presenta como un reto atractivo poder descubrir vulnerabilidades en los sistemas de compañías de la talla de Facebook, Google, Twitter, y otras grandes empresas, y ganar dinero por ello.

Por su parte, a las compañías les resulta interesante tener a una gran cantidad de investigadores, muchos de ellos con un alto nivel de conocimientos, auditando la seguridad de sus sistemas.

Además, es común, que los hackers dediquen varios meses consecutivos investigando en búsqueda de vulnerabilidades, lo cual lleva a conseguir un amplio conocimiento en profundidad de los sistemas de la compañía.

El lado interesante del bug bounty, con sus ventajas y desventajas, es que no se paga por tiempo dedicado, sino por resultados.

Esto conlleva una parte llamativa en la que el hacker no está ligado de ninguna forma a la organización.

No hay horarios, ni plazos, ni otro tipo de vinculación; pero a su vez trae el riesgo de que solo facturará si se encuentran vulnerabilidades.

De una forma esquematizada, la mecánica es la siguiente:

* Una organización publica un programa en la que invita a los hackers a encontrar vulnerabilidades a cambio de recompensa. Cada programa tiene sus políticas y condiciones, en las que se indicará lo que se puede hacer y lo que no.
* Los hackers tendrán que emplear su conocimiento y habilidades para descubrir vulnerabilidades. Hay diferentes metodologías, cada hacker emplea la que mejor se le adapte.
* Cuando el hacker encuentra una vulnerabilidad, se lo debe comunicar a la empresa a través de un reporte en el que explicará en que consiste la vulnerabilidad, su impacto y pasos para reproducirla
* Los triagers (personas encargadas en verificar y categorizar las vulnerabilidades) reproducirán la vulnerabilidad reportada y verificarán si cumple con las condiciones descritas en el programa; en ese caso, el hacker recibirá una recompensa. Hay que tener en cuenta que solo será el primer hacker en reportar la vulnerabilidad el que se lleve el dinero.

Las compañías invitan a los hackers a auditar sus infraestructuras a través de los programas. En estos, se presentan las condiciones y requisitos necesarios para poder ser recompensado.

Es fundamental leer con atención las condiciones descriptas en la política del programa, porque es un error muy habitual lanzarse a investigar sin leer demasiado e ignorar los aspectos descritos en las políticas que son fundamentales para que todo el trabajo realizado por el investigador sea bien logrado.

Hay que leer detenidamente y ver que activos están en el “proyecto”, que vulnerabilidades se aceptan, instrucciones concretas para facilitar las labores de los equipos de seguridad, confidencialidad, etc.

Las condiciones, además de ser un requisito básico para poder ser recompensado, también pueden servir para decantarse por un programa u otro.

Así, se puede elegir en función de si interesa un proyecto reducido o amplio, si es buscar vulnerabilidades en determinados activos de diferentes tipos.

Además, también cuenta con distintas métricas para conocer el estado del programa. También aquí se determina que un programa es seguro, es decir que cumple sus condiciones y no hay problemas legales

A continuación, se detallan aquellos aspectos básicos y comunes a tener en cuenta a la hora de leer las políticas de los programas:

**Scope.**

Activos en los cuales se podrá investigar en búsqueda de vulnerabilidades. También se puede excluir algún activo en concreto. Solo serán validos los reportes que sean sobre activos dentro del Scope.

Aquí puede haber grandes variaciones de un programa a otro: algunos tendrán un scope amplio (muchos activos) y otros tendrán un scope más pequeño (algún programa puede centrarse exclusivamente en su web principal por ejemplo). También se pueden especificar activos IoT, Mobile Hacking, API o análisis de código; por lo tanto cada hacker podrá elegir en función de sus habilidades.

**Vulnerabilidades permitidas.**

De la misma forma que para los activos, también habrá algunas vulnerabilidades que no se acepten y técnicas que no se puedan utilizar.

Por ejemplo: uso de ingeniería social, uso de Nessus, y otras aplicaciones suelen estar prohibidas. En muchos casos a cada tipo de vulnerabilidad se le asigna una recompensa

Recompensas: relación entre criticidad de la vulnerabilidad y dinero con el que es recompensado.

**Estadísticas de un programa.**

Los puntos vistos suelen ser habituales a todos los programas, pero cada organización podrá poner en su programa aquello que considere necesario y que aporte para mejorar la experiencia de reporte de vulnerabilidades.

Esto incluye métricas relativas, como gestiona la organización los reportes, cuantos reportes se han recibido, cuantas vulnerabilidades solucionadas, tiempo medio de respuesta al reporte, entre otros requerimientos.

**Tipos de programas.**

En función de las posibilidades de participación encontramos los siguientes programas:

* Programa público: cualquier usuario puede acceder a las condiciones del programa e investigar en búsqueda de vulnerabilidades.
* Programa privado: solo se podrá participar con una invitación. Estas invitaciones suelen llegar luego de conseguir descubrir las vulnerabilidades en otros programas dentro de una misma plataforma.

Hay otro tipo de programas, sin compensación económica, los denominados VDP (Vulnerability Disclosure Program), en los cuales la recompensa es por puntuación de reputación para la plataforma. (con los cuales se puede acceder a programas privados). Los VDP, también pueden ser públicos o privados.

**Plataformas.**

Cuando una organización decide lanzar un programa de Bug Bounty puede hacerlo de diferentes formas:

* Programa Self-Hosted: las organizaciones gestionan su propio programa de bug bounty única y exclusivamente. El personal de ciberseguridad de la organización será el responsable de comunicarse con los hackers y verificar las vulnerabilidades recibidas. Microsoft, Meta y Apple, son algunas de las organizaciones que gestionan su propio programa.
* Plataforma: son entornos que albergan varios programas de diferentes organizaciones. Estos programas aparecen organizados en un directorio, únicamente público, acompañados de estadísticas y características del programa. Las plataformas sirven de intermediarias entre el hacker y el equipo de seguridad, facilitando tareas como la comunicación y el pago de la recompensa. Estos entornos, además, usan la gamificación para hacer más atractivo el proceso y que haya competencia, fomentando, por supuesto, la creación de una comunidad. Por último, también son unas fuentes de consulta de información muy interesante, ya que en las plataformas se podrán encontrar artículos, tutoriales, etc.

**Reportes.**

Cuando un hacker encuentre una vulnerabilidad, para comunicarse con la empresa en cuestión, deberá escribir un reporte. Un reporte es un informe, con carácter profesional, en el que de una forma clara y concisa se explicará cual es la vulnerabilidad, que impacto tiene y los pasos a seguir para explotarla.

Un buen reporte facilitará el trabajo a los triagers, y hará que el proceso de resolución sea más eficiente. Ya se puede haber descubierto una vulnerabilidad espectacular, que sin un buen reporte se quedará en nada. Es fundamental saber transmitir el hallazgo y permitir así validar la vulnerabilidad y dejar constancia de su hallazgo.

Por lo general, las plataformas de Bug Bounty tienen un formulario o plantilla que ayudará a guiar la información que se debe proporcionar cuando se encuentre una vulnerabilidad.

.Por último, es una buena técnica no dar nada por asumido, para que, de esta forma, el reporte sea autocontenido y proporcione todos los detalles necesarios para reproducir la vulnerabilidad y ser consciente de las consecuencias de explotarla.

**Estados de un reporte.**

Una vez enviado y analizado, los triagers darán un estado al reporte:

* Resolved: el reporte es válido y no se necesita más información. La vulnerabilidad será parcheada y el hacker recompensado. Este será el objetivo de todo bug hunter.
* Duplicate: la vulnerabilidad descubierta es válida pero ya está pendiente de ser parcheada (solucionada) porque fue previamente descubierta por otro hacker. En el Bug Bounty se lleva la recompensa el que primero encuentre la vulnerabilidad.
* Informative: el reporte contiene información útil para la organización, pero simplemente con carácter informativo ya que carece de impacto contra su seguridad.
* N/A: el reporte no aplica para llevarse recompensa ya que no es válido. Esto puede deberse a que no cumple algo de lo indicado en las condiciones del programa.
* Triaged: el reporte está siendo evaluado, pero todavía no está resuelto. Por lo general es buena señal, ya que se está investigando como parchear (solucionar) la vulnerabilidad.
* Need more information: el triager necesita que el hacker aporte más información para poder determinar un estado final para el reporte.

Tras enviar un reporte, es habitual que haya un hilo de comunicación con los triagers para así poder validar el reporte de una forma eficiente. A menudo es necesario conocer más detalles para poder replicar la vulnerabilidad con éxito.

**La importancia del impacto.**

Será un aspecto fundamental para que el reporte sea, ya no solo sea cerrado como “Resolved”, sino también recompensado de acorde a la criticidad del hallazgo. Cuando se habla de impacto, se hace referencia a las consecuencias que puede tener para la organización que la vulnerabilidad sea explotada.

Para cuantificar el impacto, se tendrá en cuenta la triada CIA (Confidencialidad, Integridad y Disponibilidad (Availability)) que determina los tres principios de la ciberseguridad.

* Confidencialidad: asegura que la información solo sea accesible por usuarios autorizados.
* Integridad: asegura que la información sea correcta y no sufra modificaciones.
* Disponibilidad: asegura que la información y los sistemas siempre puedan ser utilizados sin sufrir interrupciones.

Cuando se encuentre una vulnerabilidad, habrá que cuestionarse si su explotación altera de alguna forma a alguno de estos tres principios mencionados.

Además, también habrá que tener en cuenta el scope del programa, ya que ahí se puede ver la prioridad que se otorga a los distintos activos (una misma vulnerabilidad tendrá mayor recompensa en un activo principal para la organización)

**Conclusiones.**

Con el Bug Bounty se puede ganar dinero, pero no hay que confundirse, no es dinero fácil. Hay que tener muy claro que el éxito llegará con dedicación, esfuerzo y por supuesto disfrutar de este proceso. Tener la idea de que el Bug Bounty es sentarse un momento delante de la computadora y ejecutar una herramienta para que lleguen las vulnerabilidades y billetes solos de forma fácil es un planteamiento que llevará al burnout y la frustración tarde o temprano.

A nivel psicológico, el bug bounty puede ser muy exigente. Una mala gestión de las expectativas, poner mucha presión en uno mismo, una mala racha de hallazgos, etc. Pueden llevar al famoso burnout. Por eso, hay que prestar especial atención a la salud mental y llevar una vida equilibrada es muy recomendable.

El Bug Bounty es una actividad muy atractiva para los amantes del hacking. El hecho de poder investigar en los activos de las compañías y poder ser recompensado por poder ayudar a que el mundo sea más ciberseguro es algo muy satisfactorio.

Además, para poder conseguir este objetivo se hace imprescindible tener conocimientos, los cuales llegan dedicando tiempo a experimentar, aprender, probar herramientas, leer artículos y preguntar a la comunidad.

En definitiva, una serie de tareas que llevan a trabajar en progresar y ser mejor día a día. Y por supuesto disfrutando de este proceso, porque como decía Aristóteles: “El placer en el trabajo lleva a la perfección de la obra”.