GitHub clarifie sa politique d'accueil des malwares

Quel sort pour les *malwares*, exploits et autres outils malveillants hébergés sur GitHub ? La plate-forme vient de clarifier sa politique en la matière, après avoir consulté ses utilisateurs.

En toile de fond, une initiative qui avait alerté les chercheurs en sécurité. En l'occurrence, la suppression, en mars, d'un PoC qui permettait d'exploiter la faille ProxyLogon. Laquelle pouvait occasionner la compromission de serveurs Exchange. Raison invoquée : au vu du niveau de risque (vulnérabilité activement exploitée), ledit PoC n'entrait pas dans les usages « acceptables ».

Pourquoi celui-ci et pas d'autres ? s'était-on demandé dans la communauté *infosec*. Non sans rappeler qu'Exchange est un produit de Microsoft… maison mère de GitHub.

À l'issue du processus de consultation, deux documents ont fait l'objet de modifications. D'un côté, celui qui liste les usages « acceptables ». De l'autre, les « lignes directrices » à destination de la communauté. Principal objectif : préciser la définition des contenus interdits… ainsi que de ceux autorisés.

Pour ce qui est des interdits, il s'agit, texto, de ceux qui « portent directement des attaques illégales actives ou des campagnes de *malware* entraînant des dommages techniques ». On nous donne, comme exemples de ces dommages, la surconsommation et l'indisponibilité de ressources, le déni de service et la perte de données. Tout en mentionnant « l'utilisation de la plate-forme pour délivrer des exécutables malveillants ou en tant qu'infrastructure d'attaque ».

GitHub : une doctrine de la bonne foi

D'après les termes qu'emploie GitHub, le caractère abusif n'est constatable qu'*a posteriori*. Sera considéré comme non acceptable un contenu qui n'aura « pas eu un objectif [d'aide à la recherche en sécurité], implicite ou explicite, avant que surviennent les abus ».

Sont autorisés, au contraire, les contenus qui relèvent de la bonne foi. GitHub affirme qu'il supposera par défaut cet état de fait. Mais invite quiconque publie de tels contenus à effectuer deux démarches :

- Identifier et décrire, dans le README.md du projet ou dans le code source, tout élément potentiellement sensible pour la cybersécurité
- Fournir, dans un fichier SECURITY.md, une méthode de contact

Dans la pratique, GitHub laisse un vaste champ d'interprétation. Il explique tout de même se réserver des droits en « de rares cas d'abus important ». En premier lieu, de restreindre l'accès à l'instance d'un contenu utilisée pour une attaque en cours ou une campagne. Dans la plupart des cas, cela consiste à imposer une authentification. On nous affirme néanmoins qu'en dernier recours, il pourrait falloir couper l'accès à la ressource… voire, lorsque cette coupure n'est pas possible, la supprimer totalement. Ces mesures sont temporaires dans la mesure du possible. Et il

existe un mécanisme d’appel des décisions de GitHub (*via* le support).

Photo d’illustration © DASPRiD / CC BY 2.0