CYBER DEFENSE AFRICA : ANNONCE DE SÉCURITÉ **Date** : 25/03/2025

Type de Contenu	Description
Titre	***Vulnérabilités critiques du contrôleur NGINX, permettant une Exécution de Code à Distance (RCE) sans authentification***
ID	*202503/AS/17*
Code TLP	*Green*
Niveau de Risque	***Critique***
Application et service affectés	*Ingress NGINX Controller*
Version / Edition	*Ingress NGINX Controller < 1.12.1, 1.11.5 et 1.10.7*
Plateforme	*Kubernetes*
Mise à jour	*Ingress NGINX Controller versions **1.12.1, 1.11.5 et 1.10.7***
Résumé	*Un ensemble de cinq vulnérabilités critiques ont été identifiées dans le contrôleur **Ingress NGINX de Kubernetes**, mettant **plusieurs clusters** en danger immédiat.*
Impacts	• ***Exécution de Code à Distance (RCE) Sans Authentification*** • ***Prise de contrôle complète du système*** • ***Compromission du Cluster & Fuite de Données*** • ***Déplacement Latéral & Élévation de Privilèges*** • ***Perturbation des Services & Attaque par Déni de Service (DoS)*** • ***Déploiement de Malwares ou de Cryptomineurs***
Description Détaillée	*Une liste de cinq (05) vulnérabilités critiques identifiées dans Ingress NGINX et connues sous le nom de **« IngressNightmare vulnerabilities »** avec un score CVSS de 9.8/10 permettent l'exécution de code à distance non authentifiée, l'élévation de privilèges, la prise de contrôle du système, la compromission et la fuite de données.* *Il s'agit de :* • ***CVE-2025-24513*** *- Une vulnérabilité de validation d'entrée incorrecte qui pourrait entraîner une traversée de répertoire dans le conteneur, conduisant à un déni de service ou à une divulgation limitée des accès secrets du cluster lorsqu'elle est combinée avec d'autres vulnérabilités.* • ***CVE-2025-24514 (CVSS score : 8.8)*** *- La balise **auth-url Ingress** peut être utilisée pour injecter une configuration dans **NGINX**, ce qui entraîne l'exécution de code arbitraire sur le contrôleur ingress-nginx et la divulgation des accès confidentiels au niveau de ce dernier.* • ***CVE-2025-1097 (CVSS score : 8.8)*** *- La **balise auth-tls-match-cn Ingress** peut être aussi utilisée pour injecter une configuration dans **NGINX**, ce qui entraîne l'exécution de code arbitraire dans le contrôleur ingress-nginx et la divulgation des accès secrets au contrôleur.*

	• ***CVE-2025-1098 (CVSS score : 8.8)*** *- Les balises* ***« Ingress mirror-target et mirror-host »*** *peuvent être utilisées pour injecter une configuration arbitraire dans* ***NGINX****, ce qui entraîne aussi l'exécution de code arbitraire au sein du contrôleur ingress-nginx et la divulgation des accès secrets au contrôleur.*
	• ***CVE-2025-1974 (CVSS score : 9.8)*** *- Un attaquant non authentifié ayant accès au réseau pod peut réaliser une exécution de code arbitraire sur le contrôleur ingress-nginx sous certaines conditions :*
	  • *Exécuter du code à distance et envoyer des charges utiles malveillantes*
	  • *Injecter une configuration malveillante et l'utiliser pour lire des fichiers sensibles et exécuter un code arbitraire.*
	  • *Abuser d'un compte de service à privilèges fort afin de lire les secrets de Kubernetes et finalement faciliter la prise de contrôle totale du cluster.*
	
	Toutes les vulnérabilités, à l'exception de la CVE-2025-1974, concernent des améliorations de la façon dont Ingress NGINX traite certains paramètres de configuration. La CVE-2025-1974, par contre, peut être combinée avec les autres failles pour faciliter la prise de contrôle du cluster sans nécessiter d'informations d'identification ou d'accès administratif.

Recommandations	***Dans un bulletin de sécurité, Kubernetes affirme avoir corrigé toutes ces vulnérabilités dans les versions 1.12.1, 1.11.5 et 1.10.7 d'Ingress NGINX Controller.***
	Il est recommandé aux utilisateurs de mettre à jour vers la dernière version dès que possible et de s'assurer que le point de terminaison du webhook d'admission n'est pas exposé à l'extérieur.
	Comme mesures d'atténuation, il est conseillé de limiter l'accès au contrôleur d'admission au serveur API de Kubernetes et de désactiver temporairement le composant du contrôleur d'admission s'il n'est pas nécessaire.
	
	Etapes pour la détection et la mitigation :
	1. ***Déterminer si vos clusters utilisent ingress-nginx***
	Dans la plupart des cas, vous pouvez vérifier cela en exécutant la commande suivante avec des permissions d'administrateur de cluster :
	Bash : ***kubectl get pods --all-namespaces--selector app.kubernetes.io/name=ingress-nginx***
	
	2. ***Mise à jour et patch des vulnérabilités***
	Toutes ces vulnérabilités sont corrigées dans les versions ***1.12.1*** *et* ***1.11.5*** *du contrôleur Ingress NGINX. Il est fortement recommandé aux administrateurs de :*
	  • *Mettre à jour vers la dernière version du* ***contrôleur Ingress NGINX.***
	  • *S'assurer que le* ***point d'accès du webhook d'admission n'est pas exposé à l'extérieur (internet).***

	3. ***Mitigations en cas d'impossibilité de mise à jour immédiate*** *Si vous ne pouvez pas mettre à jour immédiatement, envisagez ces mesures suivantes :* • *Appliquer des politiques réseaux stricts pour que seul **le serveur API Kubernetes** puisse accéder au **contrôleur d'admission**.* • *Désactiver temporairement le composant de **contrôleur d'admission d'Ingress-NGINX** si la mise à jour n'est pas possible immédiatement.* • *Si **ingress-nginx** a été installé avec « **Helm** », réinstallez-le avec l'option suivante: Bash : **helm install** <nom_du_release> **ingress-nginx --set controller.admissionWebhooks.enabled=false*** • *Si **ingress-nginx** a été installé manuellement :* ➢ *Supprimez la configuration de webhook de validation nommée **ingress-nginx-admission**.* ➢ *Retirez l'argument **--validating-webhook** du conteneur **ingress-nginx-controller** dans son **Deployment** ou **DaemonSet**.* 4. ***Note :*** *Après la mise à jour, pensez à **réactiver** le contrôleur d'admission de validation, car il joue un rôle essentiel dans la sécurisation de vos configurations Ingress.*
Indicateurs de Compromission	*N/A*
Liens pour plus de détails	*https://kubernetes.io/blog/2025/03/24/ingress-nginx-cve-2025-1974/* *https://www.wiz.io/blog/ingress-nginx-kubernetes-vulnerabilities*