What is Kubernetes node impersonation?

Kubernetes node impersonation is authenticating to the cluster API as a node the attacker does not own. On EKS this is done by wrapping stolen AWS caller identity into a k8s-aws-v1 bearer token via the AWS IAM authenticator, so the attacker is treated as that EC2 node. A node identity can read secrets projected to its pods and expand access from there. Defend with IMDS hardening, IAM least privilege, and node-authorization controls.

HOW IT WORKS

01 How it works and example

The attacker first steals EC2 instance-role credentials, usually from the metadata service. They then build a presigned AWS caller-identity request and wrap it as a k8s-aws-v1 bearer token, which the cluster accepts as the node's identity. Authenticated as the node, they read the secrets and tokens projected into that node's pods and pivot deeper, as seen in a 2026 cluster intrusion.

SOURCES

[1] MITRE ATT&CK Cloud Matrix

[2] HuggingFace, Agent intrusion technical timeline

[3] Amazon EKS cluster authentication

Find the cloud and Kubernetes attack paths before someone else does.

securelayer7.net/learn/cloud-security/kubernetes-node-impersonation

Open online