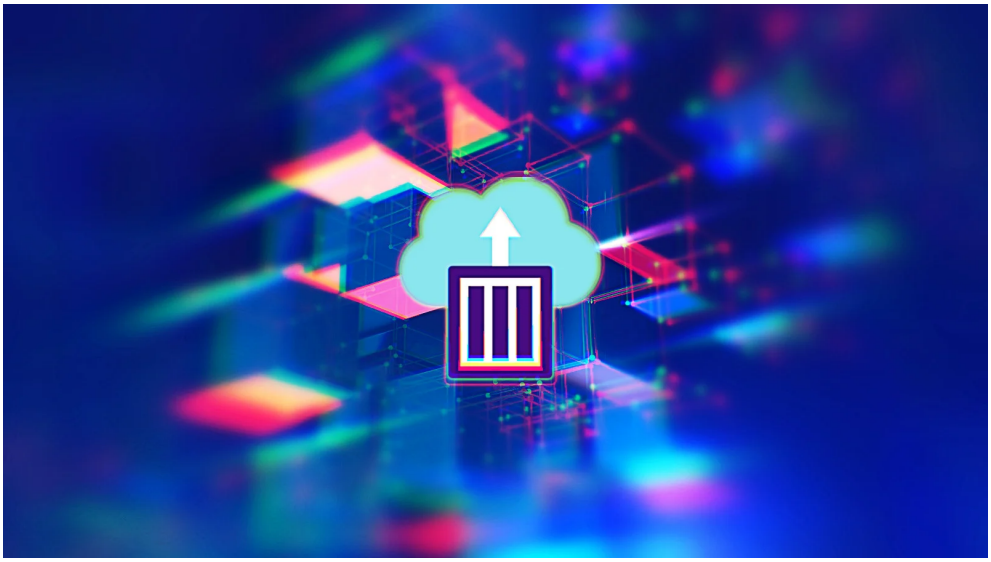
Microsoft fixes bug letting hackers take over Azure containers

By **Ionut Ilascu (https://www.bleepingcomputer.com/author/ionut-ilascu/)** September 9, 2021 11:08 AM **0**



Microsoft has fixed a vulnerability in Azure Container Instances called Azuresc
belongi

An adve
users' containers and gain access to all their data deployed to the platform, the researchers say.

Customer data at risk

Microsoft has notified customers that were potentially impacted by Azurescape to change privileged credentials for containers deployed to the platform before August 31

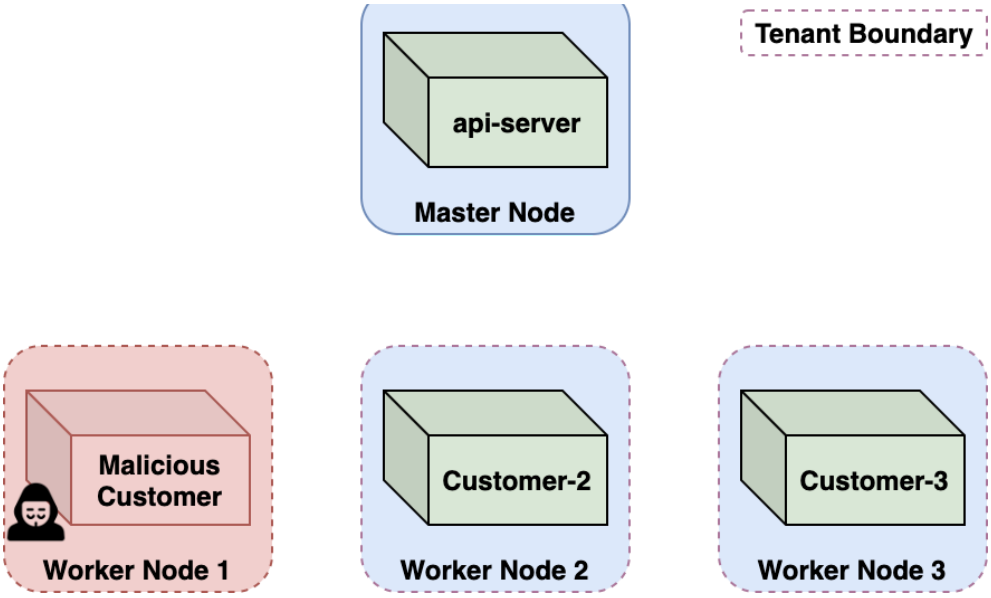
The company says that it sent the alerts out of an abundance of caution because it found no indication of an attack that leveraged the vulnerability to access customer data.

> “If you did not receive a Service Health Notification, no action is required. The vulnerability is fixed and our investigation surfaced no unauthorized access in other clusters” - Microsoft (https://msrc-blog.microsoft.com/2021/09/08/coordinated-disclosure-of-vulnerability-in-azure-container-instances-service/)

Microsoft's Azure Container Instances (ACI) is a cloud-based service that allows companies to deploy packaged applications (containers) on the cloud.

For those not familiar with containers, they have all the executables, dependencies, and files necessary to run a particular application, but are stored in a single package for easy distribution and deployment.

When containers are deployed, ACI will isolate them from other running contain
with ea



Container isolation - source: Palo Alto Networks (https://unit42.paloaltonetworks.com/azure-container-instances/)

Blame it on outdated code

Researchers at Palo Alto Networks found and reported Azurescape to Microsoft. In a report today, the company's Yuval Avrahami provides technical details (https://unit42.paloaltonetworks.com/azure-container-instances/) about the vulnerability, noting that it “allowed malicious users to compromise the multitenant Kubernetes clusters hosting ACI.”

Avrahami says that finding the issue started when with finding that ACI used code released almost five years ago, that was vulnerable to container escaping bugs.

```
ubuntu@ip-172-31-92-81:~/whoc/csp_stash$ ./aci_container_runtime  -v
runc version 1.0.0-rc2
commit: 9df8b306d01f59d3a8029be411de015b7304dd8f
spec: 1.0.0-rc2-dev
```

Outdated container runtime used in ACI - source Palo Alto Networks (https://unit42.paloaltonetworks.com/azure-container-instances/)

"RunC v1.0.0-rc2 was released on Oct. 1, 2016, and was vulnerable to at least two container breakout CVEs. Back in 2019, we analyzed one of these vulnerabilities, CVE-2019-5736," the researcher explains.

Exploiting CVE-2019-5736 was sufficient to break out of the container and get code execution with elevated privileges on the underlying host, a Kubernetes node.

The researcher summarized the next steps for getting unauthorized access to other containers as follows:

- On the node, monitor traffic on the Kubelet port, port 10250, and wait for a request that includes a JWT token in the *Authorization* header
- Issue *az container exec* to run a command on the uploaded container. The bridge pod will now send an exec request to the Kubelet on the compromised node
- On the node, extract the bridge token from the request's *Authorization* header and use it to pop a shell on the API-server.

To demonstrate the attack, Palo Alto Networks published a video showing how an attacker could have broken out of their container to get administrator privileges for the entire cluster.