Remote code execution in Wazuh server

Critical **davidjiglesias** published **GHSA-hcrc-79hj-m3qh** yesterday

Package	Affected versions	Patched versions
wazuh-manager	>= 4.4.0	>= 4.9.1

Description

Summary

An unsafe deserialization vulnerability allows for remote code execution on Wazuh servers.
The vulnerability can be triggered by anybody with API access (compromised dashboard or Wazuh servers in the cluster) or, in certain configurations, even by a compromised agent.

Details

DistributedAPI parameters are a serialized as JSON and deserialized using `as_wazuh_object` (in `framework/wazuh/core/cluster/common.py`). If an attacker manages to inject an unsanitized dictionary in DAPI request/response, they can forge an unhandled exception (`__unhandled_exc__`) to evaluate arbitrary python code.

Using the server API, it quite easy to trigger. For example, using the `run_as` endpoint (implemented by `run_as_login` in `api/api/controllers/security_controller.py`): the `auth_context` argument is completely controlled by the attacker, and is forwarded to the master server to handle. By sending a malicious `run_as` request to a worker server, it is possible to execute code on the master server.

Severity

Critical 9.9 / 10

CVSS v3 base metrics

Attack vector	Network
Attack complexity	Low
Privileges required	Low
User interaction	None
Scope	Changed
Confidentiality	Low
Integrity	High
Availability	High

Learn more about base metrics

CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:C/C:L/I:H/A:H

CVE ID

CVE-2025-24016

Weaknesses

CWE-502

Credits

DanielFi Reporter

GGP1 Remediation developer

It is also possible to exploit the bug as a compromised agent, in certain configurations.
A compromised agent can respond to a `getconfig` request with a malicious JSON object (containing a serialized unhandled exception). If the `getconfig` request was caused because of a server API request to `/agents/{agent_id}/config/{component}/{configuration}` (`api.controllers.agent_controller.get_agent_config`), and the agent is managed by a server other than the one that received the server API request, the unsafe deserialization will occur on the server that received the original server API request.

```
user          server A              server B         agent
  |               |                     |             |
  | -get-config-> |                     |             |
  |               | --get-config-dapi-> |             |
  |               |                     | --getconf-> |
  |               |                     | <-payload-- |
  |               X <-----payload------ |             |
  |               |                     |             |
```

It is likely that there are more ways to reach the unsafe deserialization function (`as_wazuh_object`), some of them might even be accessible from different contexts (without credentials, or initiated by a compromised agent). I suggest fixing the root cause instead of attempting to sanitize inputs that reach it. Note that there are multiple other ways to execute arbitrary code in `as_wazuh_object` , easier by using a `__callable__` , or potentially abusing callable gadgets in `exception` , `wresults` or `Wazuh` .

PoC

To trigger using the server API (assuming default credentials):

```
curl -X POST -k -u "wazuh-wui:MyS3cr37P450r.*-" -H "Content-Type: p
```

this will shut down the master server.

Impact

This is a remote code execution on Wazuh server, affecting the latest version (v4.9.0 at this time)