**مفهرس Wazuh**

مفهرس Wazuh هو محرك بحث وتحليل نصي كامل يعمل في الوقت الفعلي لبيانات الأمان. يتم تحليل بيانات السجلات المُدخلة إلى خادم Wazuh وإعادة توجيهها إلى المفهرس للفهرسة والتخزين. ثم يتم الاستعلام عن هذه الأحداث على لوحة معلومات Wazuh.

يقوم مفهرس Wazuh بتخزين البيانات كمستندات JSON. يربط كل مستند مجموعة من المفاتيح أو أسماء الحقول أو السمات بقيمها المقابلة، والتي يمكن أن تكون أحرفًا أو أرقامًا أو قيمًا منطقية أو تواريخ أو مصفوفات من القيم أو مواقع جغرافية أو أنواع أخرى من البيانات.

يمكن تكوين مفهرس Wazuh كعقدة واحدة أو كمجموعة متعددة العقد، مما يوفر قابلية التوسع والتوافر العالي. يقوم بتوزيع المستندات عبر حاويات مختلفة تُعرف باسم القطع (shards). وبدوره، يوزع هذه القطع عبر عقد المجموعة. من خلال توزيع المستندات عبر قطع متعددة وتوزيع تلك القطع عبر عقد متعددة، يضمن مفهرس Wazuh التكرار الاحتياطي. يضمن التكرار الاحتياطي توافر مفهرس Wazuh في حالة حدوث عطل ويعزز قدرة الاستعلام عبر عقد المجموعة.

**المحتويات**

* فهارس مفهرس Wazuh
* إعادة الفهرسة
* ضبط مفهرس Wazuh
* ترحيل فهارس Wazuh
* تكوين مفهرس Wazuh على نقاط النهاية المحصنة

# فهارس مفهرس Wazuh

**فهارس مفهرس Wazuh - الأنماط والأمثلة**

الفهرس هو مجموعة من المستندات المرتبطة ببعضها البعض. يستخدم مفهرس Wazuh الفهارس لتخزين وتنظيم بيانات الأمان من أجل الاسترجاع السريع.

**1. wazuh-alerts-\***

**الوصف:** نمط الفهرس للتنبيهات التي يولدها خادم Wazuh عند مطابقة الأحداث لقواعد الاكتشاف.

**مثال على البيانات المخزنة:**

{

{

"timestamp": "2024-03-15T10:23:45.123Z",

"rule": {

"id": "5710",

"level": 5,

"description": "محاولة تسجيل دخول SSH فاشلة"

},

"agent": {

"id": "001",

"name": "web-server-01",

"ip": "192.168.1.100"

},

"data": {

"srcip": "203.0.113.45",

"srcuser": "admin",

"dstport": "22"

},

"location": "/var/log/auth.log",

"decoder": {

"name": "sshd"

}

}

**حالات الاستخدام:**

* اكتشاف محاولات الاختراق
* تتبع الأنشطة المشبوهة
* مراقبة أحداث الأمان الحرجة

**2. wazuh-archives-\***

**الوصف:** نمط الفهرس لجميع الأحداث المرسلة إلى خادم Wazuh، بما في ذلك الأحداث التي لم تُولد تنبيهات.

**مثال على البيانات المخزنة:**

{

json

{

"timestamp": "2024-03-15T10:25:12.456Z",

"agent": {

"id": "002",

"name": "db-server-01"

},

"full\_log": "Mar 15 10:25:12 db-server sshd[1234]: Accepted publickey for user1 from 192.168.1.50",

"location": "/var/log/secure",

"decoder": {

"name": "sshd"

}

}

**حالات الاستخدام:**

* التحليل الجنائي الشامل
* متطلبات الامتثال والتدقيق
* التحقيقات التاريخية

**3. wazuh-monitoring-\***

**الوصف:** نمط الفهرس لحالة اتصال وكلاء Wazuh (نشط، غير متصل، قيد الانتظار، لم يتصل مطلقًا).

**مثال على البيانات المخزنة:**

{

"timestamp": "2024-03-15T10:30:00.000Z",

"agent": {

"id": "003",

"name": "app-server-02",

"ip": "192.168.1.120",

"status": "active",

"version": "4.7.0"

},

"manager": {

"name": "wazuh-manager-01"

},

"last\_keep\_alive": "2024-03-15T10:29:55.000Z"

}

**حالات الاستخدام:**

* مراقبة صحة الوكلاء
* اكتشاف انقطاع الاتصال
* تتبع توافر الأنظمة

**4. wazuh-statistics-\***

**الوصف:** نمط الفهرس لمقاييس أداء خادم Wazuh )الأحداث المستلمة، المعالجة، المُسقطة(.

**مثال على البيانات المخزنة:**

json

{

"timestamp": "2024-03-15T10:35:00.000Z",

"manager": {

"name": "wazuh-manager-01"

},

"events": {

"received": 15432,

"decoded": 15420,

"dropped": 12

},

"bytes": {

"received": 2457600

},

"tcp\_sessions": 45,

"queue\_usage": 12.5

}

**حالات الاستخدام:**

* تحليل أداء النظام
* تحديد الاختناقات
* تخطيط السعة

**5. wazuh-states-vulnerabilities-\***

**الوصف:** نمط الفهرس للثغرات الأمنية المكتشفة في نقاط النهاية المراقبة.

**مثال على البيانات المخزنة:**

{

"timestamp": "2024-03-15T11:00:00.000Z",

"agent": {

"id": "004",

"name": "web-server-02"

},

"vulnerability": {

"cve": "CVE-2024-1234",

"title": "ثغرة في Apache HTTP Server",

"severity": "high",

"cvss": {

"base\_score": 7.5

},

"published": "2024-02-10",

"affected\_package": {

"name": "apache2",

"version": "2.4.41"

},

"reference": "https://nvd.nist.gov/vuln/detail/CVE-2024-1234"

},

"status": "open"

}

**حالات الاستخدام:**

* إدارة الثغرات الأمنية
* تحديد أولويات التصحيح
* تقييم المخاطر الأمنية

**6. wazuh-states-inventory-hardware-\***

**الوصف:** نمط الفهرس للمعلومات الأساسية عن مكونات الأجهزة.

**مثال على البيانات المخزنة:**

{

"timestamp": "2024-03-15T09:00:00.000Z",

"agent": {

"id": "005",

"name": "app-server-03"

},

"hardware": {

"cpu": {

"name": "Intel Core i7-9700K",

"cores": 8,

"speed": "3.6 GHz"

},

"ram": {

"total": 16384,

"unit": "MB"

},

"board": {

"manufacturer": "Dell Inc.",

"serial": "ABC123XYZ789"

}

}

}

**حالات الاستخدام:**

* إدارة أصول الأجهزة
* اكتشاف تغييرات الأجهزة غير المصرح بها
* التخطيط للترقيات

**7. wazuh-states-inventory-hotfixes-\***

**الوصف:** نمط الفهرس للتحديثات المثبتة على نقاط نهاية Windows.

**مثال على البيانات المخزنة:**

### 🔗 مثال متقدم: ربط الإصلاحات مع الثغرات المُصلحة

عندما تقوم وحدة اكتشاف الثغرات في Wazuh بمقارنة الإصلاحات المثبتة مع قاعدة بيانات الثغرات، يتم إنشاء بيانات مدمجة توضح العلاقة بين التحديثات والثغرات المُصلحة:

**مثال 1: إصلاح يعالج ثغرات متعددة**

json

{

"timestamp": "2024-03-15T08:30:00.000Z",

"agent": {

"id": "006",

"name": "win-desktop-01",

"ip": "192.168.1.85"

},

"hotfix": {

"id": "KB5034441",

"description": "2024-01 Cumulative Update for Windows 11",

"installed\_on": "2024-01-15",

"installed\_by": "NT AUTHORITY\\SYSTEM"

},

"vulnerabilities\_patched": [

{

"cve": "CVE-2024-20656",

"title": "Windows Hyper-V Security Feature Bypass Vulnerability",

"severity": "important",

"cvss\_score": 6.7,

"status": "patched",

"patched\_date": "2024-01-15"

},

{

"cve": "CVE-2024-20698",

"title": "Windows Kernel Elevation of Privilege Vulnerability",

"severity": "important",

"cvss\_score": 7.8,

"status": "patched",

"patched\_date": "2024-01-15"

},

{

"cve": "CVE-2024-21305",

"title": "Windows Kernel Security Feature Bypass Vulnerability",

"severity": "important",

"cvss\_score": 6.8,

"status": "patched",

"patched\_date": "2024-01-15"

}

],

"patch\_coverage": {

"total\_cves\_addressed": 3,

"critical": 0,

"high": 0,

"important": 3,

"moderate": 0

}

}

**مثال 2: تحديث أمني حرج**

json

{

"timestamp": "2024-03-10T14:22:00.000Z",

"agent": {

"id": "015",

"name": "win-server-2019-prod"

},

"hotfix": {

"id": "KB5035857",

"description": "Security Update for Windows Server 2019",

"installed\_on": "2024-03-10",

"installed\_by": "DOMAIN\\AdminUser",

"size": 457000000,

"type": "Security Update"

},

"vulnerabilities\_patched": [

{

"cve": "CVE-2024-21410",

"title": "Microsoft Exchange Server Elevation of Privilege Vulnerability",

"severity": "critical",

"cvss\_score": 9.8,

"cvss\_vector": "CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H",

"status": "patched",

"patched\_date": "2024-03-10",

"exploit\_available": true,

"exploit\_maturity": "functional",

"published\_date": "2024-02-13",

"days\_to\_patch": 25,

"reference\_urls": [

"https://msrc.microsoft.com/update-guide/vulnerability/CVE-2024-21410",

"https://nvd.nist.gov/vuln/detail/CVE-2024-21410"

]

}

],

"patch\_impact": {

"reboot\_required": true,

"service\_restart\_required": ["W3SVC", "MSExchangeServiceHost"],

"downtime\_window": "2024-03-10T14:00:00Z to 2024-03-10T14:30:00Z"

}

}

**مثال 3: حالة ثغرة غير مُصلحة (للمقارنة)**

يمكن أيضاً رؤية الثغرات التي لم يتم إصلاحها بعد عند مقارنتها بالتحديثات الموجودة:

json

{

"timestamp": "2024-03-15T09:00:00.000Z",

"agent": {

"id": "007",

"name": "win-legacy-server"

},

"scan\_result": {

"installed\_hotfixes": [

{

"id": "KB5032198",

"installed\_on": "2023-11-20"

},

{

"id": "KB5033909",

"installed\_on": "2023-12-15"

}

],

"missing\_patches": [

{

"hotfix\_id": "KB5034441",

"release\_date": "2024-01-09",

"days\_overdue": 65

}

],

"unpatched\_vulnerabilities": [

{

"cve": "CVE-2024-20656",

"title": "Windows Hyper-V Security Feature Bypass Vulnerability",

"severity": "important",

"cvss\_score": 6.7,

"status": "open",

"fix\_available": true,

"required\_patch": "KB5034441",

"threat\_level": "medium",

"exploitability": "low",

"discovered\_date": "2024-03-15",

"risk\_score": 65

},

{

"cve": "CVE-2024-21305",

"title": "Windows Kernel Security Feature Bypass Vulnerability",

"severity": "important",

"cvss\_score": 6.8,

"status": "open",

"fix\_available": true,

"required\_patch": "KB5034441",

"threat\_level": "medium",

"exploitability": "high",

"discovered\_date": "2024-03-15",

"risk\_score": 72

}

],

"compliance\_status": {

"patch\_level": "non-compliant",

"missing\_critical\_patches": 0,

"missing\_important\_patches": 1,

"last\_update": "2023-12-15",

"days\_since\_last\_update": 90

}

}

}

**مثال 4: تقرير شامل لحالة التصحيح**

json

{

"timestamp": "2024-03-15T10:00:00.000Z",

"agent": {

"id": "008",

"name": "win-workstation-finance"

},

"patch\_status\_report": {

"os\_version": "Windows 10 Enterprise 22H2",

"build\_number": "19045.4046",

"total\_hotfixes\_installed": 47,

"recent\_hotfixes": [

{

"id": "KB5034763",

"installed\_on": "2024-01-25",

"cves\_patched": ["CVE-2024-21351", "CVE-2024-21334", "CVE-2024-21335"]

},

{

"id": "KB5034441",

"installed\_on": "2024-01-15",

"cves\_patched": ["CVE-2024-20656", "CVE-2024-20698", "CVE-2024-21305"]

}

],

"vulnerability\_summary": {

"total\_vulnerabilities\_found": 8,

"patched": 6,

"unpatched": 2,

"critical\_patched": 1,

"high\_patched": 2,

"medium\_patched": 3,

"low\_patched": 0

},

"patched\_vulnerabilities": [

{

"cve": "CVE-2024-21351",

"severity": "critical",

"cvss\_score": 9.1,

"patch\_id": "KB5034763",

"patched\_on": "2024-01-25",

"time\_to\_patch\_days": 12

},

{

"cve": "CVE-2024-20656",

"severity": "high",

"cvss\_score": 7.8,

"patch\_id": "KB5034441",

"patched\_on": "2024-01-15",

"time\_to\_patch\_days": 6

}

],

"compliance\_score": 75.0,

"risk\_level": "low",

"next\_scan\_scheduled": "2024-03-22T10:00:00.000Z"

}

}

### 📊 كيفية عمل آلية المقارنة

1. **جمع بيانات الإصلاحات** - Wazuh يجمع قائمة بجميع التحديثات المثبتة (KB)
2. **مسح قاعدة البيانات** - يقارن مع قواعد بيانات الثغرات (NVD, Microsoft Security Response Center)
3. **تحديد الثغرات المُصلحة** - يطابق كل KB مع الثغرات التي يعالجها
4. **تقييم المخاطر** - يحسب مستوى التهديد المتبقي بناءً على التحديثات المفقودة
5. **إنشاء التنبيهات** - ينبه عن الثغرات الحرجة غير المُصلحة

**حالات الاستخدام:**

* التحقق من امتثال التصحيحات
* تحديد الثغرات المُصلحة بدقة
* تدقيق سجل التحديثات
* تقييم فعالية إدارة التصحيحات
* تحديد أولويات التحديثات المطلوبة
* إنشاء تقارير الامتثال الأمني

**8. wazuh-states-inventory-interfaces-\***

**الوصف:** نمط الفهرس لمعلومات واجهات الشبكة (حالة التشغيل، إحصائيات نقل البيانات).

**مثال على البيانات المخزنة:**

{

"timestamp": "2024-03-15T10:15:00.000Z",

"agent": {

"id": "007",

"name": "firewall-01"

},

"interface": {

"name": "eth0",

"mac": "00:0c:29:3f:4a:5b",

"state": "up",

"mtu": 1500,

"type": "ethernet",

"stats": {

"tx\_packets": 1523456,

"rx\_packets": 2341567,

"tx\_bytes": 987654321,

"rx\_bytes": 1234567890,

"tx\_errors": 0,

"rx\_errors": 2

}

}

}

**حالات الاستخدام:**

* مراقبة أداء الشبكة
* اكتشاف الواجهات غير المصرح بها
* تحليل أنماط حركة المرور

**9. wazuh-states-inventory-networks-\***

**الوصف:** نمط الفهرس لعناوين IPv4 و IPv6 المرتبطة بواجهات الشبكة.

**مثال على البيانات المخزنة:**

{

"timestamp": "2024-03-15T10:20:00.000Z",

"agent": {

"id": "008",

"name": "db-server-02"

},

"network": {

"interface": "eth0",

"ipv4": {

"address": "192.168.1.150",

"netmask": "255.255.255.0",

"broadcast": "192.168.1.255",

"gateway": "192.168.1.1"

},

"ipv6": {

"address": "fe80::20c:29ff:fe3f:4a5b",

"prefix": "64"

}

}

}

**حالات الاستخدام:**

* تتبع تعيينات عناوين IP
* اكتشاف تغييرات التكوين
* التحقيق في حوادث الشبكة

**10. wazuh-states-inventory-packages-\***

**الوصف:** نمط الفهرس لحزم البرامج المثبتة على نقاط النهاية.

**مثال على البيانات المخزنة:**

{

"timestamp": "2024-03-15T09:45:00.000Z",

"agent": {

"id": "009",

"name": "web-server-03"

},

"package": {

"name": "nginx",

"version": "1.18.0-6ubuntu14.4",

"architecture": "amd64",

"vendor": "Ubuntu",

"install\_time": "2023-11-20T14:30:00.000Z",

"size": 1234567,

"description": "خادم ويب عالي الأداء"

}

}

**حالات الاستخدام:**

* إدارة جرد البرامج
* اكتشاف البرامج غير المصرح بها
* التحقق من إصدارات الحزم للثغرات

**11. wazuh-states-inventory-ports-\***

**الوصف:** نمط الفهرس لمنافذ الشبكة المفتوحة على نقاط النهاية.

**مثال على البيانات المخزنة:**

{

"timestamp": "2024-03-15T11:10:00.000Z",

"agent": {

"id": "010",

"name": "app-server-04"

},

"port": {

"protocol": "tcp",

"local\_ip": "0.0.0.0",

"local\_port": 443,

"remote\_ip": "0.0.0.0",

"remote\_port": 0,

"state": "listening",

"process": "nginx",

"pid": 1234

}

}

**حالات الاستخدام:**

* اكتشاف الخدمات غير المصرح بها
* تقليل سطح الهجوم
* التحقق من امتثال سياسات الشبكة

**12. wazuh-states-inventory-processes-\***

**الوصف:** نمط الفهرس لعمليات النظام العاملة على نقاط النهاية.

**مثال على البيانات المخزنة:**

{

"timestamp": "2024-03-15T11:15:00.000Z",

"agent": {

"id": "011",

"name": "workstation-01"

},

"process": {

"pid": 5678,

"name": "chrome",

"state": "running",

"ppid": 1234,

"user": "john.doe",

"cmd": "/usr/bin/google-chrome --type=renderer",

"priority": 20,

"nice": 0,

"vm\_size": 524288,

"resident": 204800,

"start\_time": "2024-03-15T08:00:00.000Z"

}

}

**حالات الاستخدام:**

* اكتشاف البرامج الضارة
* مراقبة العمليات المشبوهة
* التحقيقات الجنائية

**13. wazuh-states-inventory-protocols-\***

**الوصف:** نمط الفهرس لتفاصيل تكوين توجيه الشبكة والبروتوكولات.

**مثال على البيانات المخزنة:**

{

"timestamp": "2024-03-15T10:00:00.000Z",

"agent": {

"id": "012",

"name": "router-01"

},

"protocol": {

"interface": "eth0",

"type": "ipv4",

"gateway": "192.168.1.1",

"dhcp": "enabled",

"metric": 100

},

"route": {

"destination": "0.0.0.0",

"gateway": "192.168.1.1",

"mask": "0.0.0.0",

"interface": "eth0",

"metric": 100

}

}

**حالات الاستخدام:**

* اكتشاف تغييرات التوجيه غير المصرح بها
* التحقق من تكوينات الشبكة
* اكتشاف محاولات إعادة توجيه حركة المرور

**14. wazuh-states-inventory-system-\***

**الوصف:** نمط الفهرس للمعلومات الأساسية عن نظام التشغيل والبنية.

**مثال على البيانات المخزنة:**

{

"timestamp": "2024-03-15T09:00:00.000Z",

"agent": {

"id": "013",

"name": "mail-server-01"

},

"system": {

"hostname": "mail-server-01.company.local",

"os": {

"name": "Ubuntu",

"version": "22.04.3 LTS",

"codename": "Jammy Jellyfish",

"platform": "ubuntu",

"kernel\_name": "Linux",

"kernel\_version": "5.15.0-91-generic",

"kernel\_release": "5.15.0-91"

},

"architecture": "x86\_64",

"release": "22.04",

"scan\_time": "2024-03-15T09:00:00.000Z"

}

}

**حالات الاستخدام:**

* إدارة جرد الأصول
* التحقق من امتثال إصدارات نظام التشغيل
* تجميع التنبيهات حسب نوع النظام

**ملخص الفوائد**

تتيح هذه الفهارس مجتمعة لـ Wazuh:

✅ **المراقبة الشاملة** - تغطية كاملة لجميع جوانب الأمان والبنية التحتية

✅ **الاستجابة السريعة** - استرجاع فوري للبيانات الحرجة عند وقوع حادث

✅ **الامتثال والتدقيق** - سجلات تاريخية مفصلة لجميع الأنشطة

✅ **اكتشاف التهديدات** - تحليل متقدم للسلوكيات الشاذة

✅ **إدارة الثغرات** - تتبع دقيق للثغرات الأمنية والتصحيحات

## إنشاء نمط فهرس مخصص

يصف هذا القسم كيفية إنشاء نمط فهرس مخصص، على سبيل المثال، **my-custom-alerts-**، جنبًا إلى جنب مع النمط الافتراضي، ***wazuh-alerts-***. انتقل إلى المستخدم الجذر (root) ونفذ الخطوات أدناه.

**1.** أوقف خدمة Filebeat:

systemctl stop filebeat

**2.** قم بتنزيل قالب Wazuh واحفظه في ملف (على سبيل المثال، template.json):

curl -so template.json https://raw.githubusercontent.com/wazuh/wazuh/v4.13.1/extensions/elasticsearch/7.x/wazuh-template.json

**3.** افتح ملف القالب وحدد موقع هذا السطر في بداية الملف:

"index\_patterns": [

"wazuh-alerts-4.x-\*",

"wazuh-archives-4.x-\*"

],

أضف النمط المخصص الخاص بك ليبدو كما يلي:

"index\_patterns": [

"wazuh-alerts-4.x-\*",

"wazuh-archives-4.x-\*",

"my-custom-alerts-\*"

],

حرف النجمة (\*) في أنماط الفهرس مهم لأن Filebeat سينشئ فهارس باستخدام اسم يتبع هذا النمط، وهو أمر ضروري لتطبيق التنسيق المناسب لتصور التنبيهات على لوحة معلومات Wazuh.

**4.** احفظ التعديلات وأدخل القالب الجديد في مفهرس Wazuh. سيؤدي هذا إلى استبدال القالب الموجود:

curl -XPUT -k -u <INDEXER\_USERNAME>:<INDEXER\_PASSWORD> 'https://<INDEXER\_IP\_ADDRESS>:9200/\_template/wazuh' -H 'Content-Type: application/json' -d @template.json

استبدل:

* <INDEXER\_IP\_ADDRESS> بعنوان IP الخاص بمفهرس Wazuh الخاص بك
* <INDEXER\_USERNAME> و <INDEXER\_PASSWORD> باسم المستخدم وكلمة المرور الخاصة بمفهرس Wazuh. يمكنك الحصول على بيانات اعتماد مفهرس Wazuh للنشر الجديد باستخدام الأمر:

**ملاحظة:** إذا كنت تستخدم Wazuh OVA، استخدم بيانات الاعتماد الافتراضية admin:admin أو راجع قسم إدارة كلمات المرور.

tar -axf wazuh-install-files.tar wazuh-install-files/wazuh-passwords.txt -O | grep -P "\'admin\'" -A 1

**المخرجات:**

{"acknowledged":true}

**ملاحظة:** {"acknowledged":true} يشير إلى أن القالب تم إدراجه بشكل صحيح.

**تحذير:** قم بتنفيذ الخطوة 5 فقط إذا كنت تريد استبدال نمط فهرس التنبيه الافتراضي **wazuh-alerts-**\* و/أو نمط فهرس الأرشيف الافتراضي **wazuh-archives-**\* بـ **my-custom-alerts-**\*.

**5.** افتح ملف تكوين تنبيهات Wazuh /usr/share/filebeat/module/wazuh/alerts/manifest.yml واختياريًا ملف الأرشيفات /usr/share/filebeat/module/wazuh/archives/manifest.yml واستبدل اسم الفهرس.

على سبيل المثال، من:

- name: index\_prefix

default: wazuh-alerts-

إلى هذا:

- name: index\_prefix

default: my-custom-alerts-

**ملاحظة:** يجب ألا يحتوي اسم الفهرس على الأحرف #, , /, \*, ?, ", <, >, |, , ويجب ألا يبدأ بـ \_, -, أو +. أيضًا، يجب أن تكون جميع الحروف صغيرة.

**6.** (اختياري) إذا كنت تريد استخدام نمط الفهرس الجديد بشكل افتراضي، افتح ملف /usr/share/wazuh-dashboard/data/wazuh/config/wazuh.yml وأضف التكوين أدناه:

pattern: my-custom-alerts-\*

سيجعل هذا خادم Wazuh ينشئ و/أو يختار نمط الفهرس الجديد تلقائيًا.

**7.** أعد تشغيل Filebeat ومكونات خادم Wazuh:

systemctl restart filebeat

systemctl restart wazuh-manager

systemctl restart wazuh-indexer

systemctl restart wazuh-dashboard

**تحذير:** إذا كان لديك بالفعل فهارس تم إنشاؤها بالاسم السابق، فلن يتم تغييرها. يمكنك الاستمرار في التغيير إلى نمط الفهرس السابق لرؤيتها، أو يمكنك إجراء إعادة فهرسة لإعادة تسمية الفهارس الموجودة.

## التحقق من معلومات الفهارس

يمكنك التحقق من معلومات حول فهارس Wazuh بطريقتين:

1. استخدام واجهة مستخدم الويب.
2. تقديم طلب إلى واجهة برمجة تطبيقات (API) مفهرس Wazuh.

### استخدام واجهة مستخدم الويب

**1.** في القائمة اليسرى العليا للوحة معلومات Wazuh ☰، انتقل إلى **Indexer management > Index Management**.

**2.** انقر على **Indices**.

**3.** إذا لم يكن النمط موجودًا في لوحة معلومات Wazuh، فقم بإنشاء نمط جديد باستخدام نمط الفهرس المستخدم في القالب **my-custom-alerts-**\*، وتأكد من استخدام **timestamp** كاسم حقل مرشح الوقت (Time Filter field).

### استخدام واجهة برمجة تطبيقات مفهرس Wazuh

يمكنك الاستعلام عن معلومات الفهارس باستخدام واجهة برمجة تطبيقات مفهرس Wazuh من لوحة معلومات Wazuh أو خادم Wazuh.

#### لوحة معلومات Wazuh

انتقل إلى ☰ > **Indexer management > Dev Tools**:

GET /\_cat/indices/wazuh-\*?v

#### واجهة سطر الأوامر

**1.** احصل على اسم المستخدم وكلمة المرور لمفهرس Wazuh للنشر الجديد باستخدام الأمر أدناه:

tar -axf wazuh-install-files.tar wazuh-install-files/wazuh-passwords.txt -O | grep -P "\'admin\'" -A 1

**ملاحظة:** إذا كنت تستخدم Wazuh OVA، استخدم بيانات الاعتماد الافتراضية admin:admin أو راجع قسم إدارة كلمات المرور.

**2.** قم بتشغيل الأمر التالي للاستعلام عن حالة الفهرس الخاص بك. استبدل <INDEXER\_USERNAME> و <INDEXER\_PASSWORD> باسم المستخدم وكلمة المرور التي حصلت عليها. استبدل <INDEXER\_IP\_ADDRESS> بعنوان IP الخاص بمفهرس Wazuh أو FQDN. يمكنك استبدال wazuh-\* بنمط أكثر تحديدًا لاستعلامك، مثل wazuh-alerts-\*.

curl -k -u <INDEXER\_USERNAME>:<INDEXER\_PASSWORD> https://<INDEXER\_IP\_ADDRESS>:9200/\_cat/indices/wazuh-\*?v

**المخرجات:**

health status index uuid pri rep docs.count docs.deleted store.size pri.store.size

green open wazuh-statistics-2023.30w xtHZtGqBR0WNJWbs5sjrnQ 1 0 2394 0 1.2mb 1.2mb

green open wazuh-alerts-4.x-2023.07.28 VbBfAasJTsiqw3lwRhY5sg 3 0 513 0 1.9mb 1.9mb

green open wazuh-alerts-4.x-2023.07.27 7s2x8INqRVmtz5uqMDuA7Q 3 0 515 0 2mb 2mb

...

## فهارس wazuh-alerts-\*

يقوم خادم Wazuh بتحليل الأحداث المستلمة من نقاط النهاية المراقبة ويولد تنبيهات عندما تطابق الأحداث قاعدة اكتشاف. يتم حفظ هذه التنبيهات باستخدام فهارس **wazuh-alerts-**\*.

يسجل خادم Wazuh بيانات التنبيه في ملفات /var/ossec/logs/alerts/alerts.json و /var/ossec/logs/alerts/alerts.log بشكل افتراضي. بمجرد الحفظ في ملف /var/ossec/logs/alerts/alerts.json، يقوم بإعادة توجيه مستند تنبيه JSON إلى واجهة برمجة تطبيقات مفهرس Wazuh للفهرسة. يتم تخزين الملفات المفهرسة في دليل /var/lib/wazuh-indexer/nodes/0/indices الخاص بمفهرس Wazuh.

عند إعادة توجيه التنبيهات إلى مفهرس Wazuh، يقوم خادم Wazuh بتنسيق التاريخ الحالي في اسم فهرس. على سبيل المثال، سيحدد خادم Wazuh أسماء الفهارس **wazuh-alerts-4.x-2023.03.17** و **wazuh-alerts-4.x-2023.03.18** لتنبيهات 17 و 18 مارس، على التوالي. يقوم مفهرس Wazuh بعد ذلك بإنشاء فهارس التنبيه باستخدام أسماء فهارس **wazuh-alerts-**\* المحددة.

يمكنك تعديل اسم الفهرس الافتراضي في ملف /usr/share/filebeat/module/wazuh/alerts/ingest/pipeline.json الخاص بخادم Wazuh. للقيام بذلك، انتقل إلى حقل **date\_index\_name** ومفتاح **date\_rounding** لتغيير تنسيق اسم الفهرس الافتراضي في ملف /usr/share/filebeat/module/wazuh/alerts/ingest/pipeline.json:

{

"description": "Wazuh alerts pipeline",

"processors": [

{ "json" : { "field" : "message", "add\_to\_root": true } },

...

{

"date\_index\_name": {

"field": "timestamp",

"date\_rounding": "d",

"index\_name\_prefix": "{{fields.index\_prefix}}",

"index\_name\_format": "yyyy.MM.dd",

"ignore\_failure": false

}

},

...

],

"on\_failure" : [{

"drop" : { }

}]

}

حيث القيم:

* **M** - تمثل الشهر
* **w** - تمثل الأسبوع
* **d** - تمثل اليوم

## فهارس wazuh-archives-\*

بالإضافة إلى تسجيل التنبيهات في ملفات /var/ossec/logs/alerts/alerts.json و /var/ossec/logs/alerts/alerts.log، يمكنك تمكين أرشيفات Wazuh لتسجيل وفهرسة جميع الأحداث التي يتلقاها خادم Wazuh. يتضمن ذلك الأحداث التي يتم تحليلها بواسطة Wazuh والأحداث التي لا تؤدي إلى تنبيهات.

قد يكون تخزين وفهرسة جميع الأحداث مفيدًا للتحليل اللاحق ومتطلبات الامتثال. ومع ذلك، يجب أن تأخذ في الاعتبار أن تمكين تسجيل وفهرسة جميع الأحداث سيزيد من متطلبات التخزين على خادم Wazuh.

بشكل افتراضي، ينشئ مفهرس Wazuh فهارس الأحداث لكل يوم فريد. يمكنك تعديل اسم الفهرس الافتراضي في ملف /usr/share/filebeat/module/wazuh/archives/ingest/pipeline.json الخاص بخادم Wazuh. للقيام بذلك:

1. انتقل إلى حقل **date\_index\_name**.
2. حدد موقع مفتاح **date\_rounding** وقم بتغيير تنسيق اسم الفهرس الافتراضي في ملف /usr/share/filebeat/module/wazuh/archives/ingest/pipeline.json.

توفر الأقسام أدناه تفاصيل حول كيفية تمكين أرشيفات Wazuh وإعداد فهارس **wazuh-archives-**\*.

### تمكين أرشيفات Wazuh

**1.** قم بتحرير /var/ossec/etc/ossec.conf على خادم Wazuh وقم بتعيين سطر <logall\_json> إلى yes. هذا يمكّن التسجيل في archives.json لجميع الأحداث. تتطلب إعادة التوجيه إلى مفهرس Wazuh تسجيل جميع الأحداث بتنسيق JSON.

<logall\_json>yes</logall\_json>

**2.** أعد تشغيل مدير Wazuh لجعل التغيير فعالاً.

systemctl restart wazuh-manager

أو

service wazuh-manager restart

**3.** قم بتحرير /etc/filebeat/filebeat.yml وقم بتغيير enabled إلى true في تعيين الأرشيفات. هذا يمكّن إعادة توجيه الأحداث إلى مفهرس Wazuh.

filebeat.modules:

- module: wazuh

alerts:

enabled: true

archives:

enabled: true

**4.** أعد تشغيل خدمة Filebeat لتطبيق التغيير:

systemctl restart filebeat

1. اختبر أن خدمة Filebeat تعمل بشكل صحيح:

filebeat test output

**المخرجات:**

elasticsearch: https://127.0.0.1:9200...

parse url... OK

connection...

parse host... OK

dns lookup... OK

addresses: 127.0.0.1

dial up... OK

TLS...

security: server's certificate chain verification is enabled

handshake... OK

TLS version: TLSv1.2

dial up... OK

talk to server... OK

version: 7.10.2

### تحديد نمط الفهرس

**1.** في القائمة اليسرى العليا للوحة معلومات Wazuh ☰، انتقل إلى **Dashboard management > Dashboard Management** وانقر على **Index Patterns**.

**2.** انقر على **Create index pattern**.

**3.** قم بتعيين **wazuh-archives-**\* كاسم نمط الفهرس (Index pattern name). هذا يحدد نمط الفهرس لمطابقة الأحداث التي يتم إعادة توجيهها وفهرستها. انقر على **Next step**.

**4.** حدد **timestamp** لحقل الوقت (Time field).

**ملاحظة:** كن حذرًا لاختيار timestamp بدلاً من @timestamp.

**5.** انقر على **Create index pattern**.

### عرض نمط الفهرس

**1.** انقر على **Explore** في القائمة اليسرى العليا ☰، ثم انقر على **Discover**.

**2.** حدد **wazuh-archives-**\* لعرض الأحداث.

## فهارس wazuh-monitoring-\*

حالة اتصال وكيل Wazuh المُسجل في أي لحظة هي واحدة مما يلي:

* نشط (Active)
* غير متصل (Disconnected)
* قيد الانتظار (Pending)
* لم يتصل مطلقًا (Never connected)

يخزن Wazuh سجلًا لحالة اتصال جميع وكلائه. بشكل افتراضي، يقوم بفهرسة حالة اتصال الوكيل باستخدام فهارس **wazuh-monitoring-**\*. ينشئ مفهرس Wazuh أحد هذه الفهارس في الأسبوع بشكل افتراضي. راجع الوثائق حول فترات الإنشاء المخصصة. تخزن هذه الفهارس حالة اتصال جميع الوكلاء كل 15 دقيقة بشكل افتراضي. راجع الوثائق حول تكرار طلبات API.

تتطلب لوحة معلومات Wazuh هذه الفهارس لعرض معلومات حول حالة الوكيل. على سبيل المثال، بالنقر على ☰ > **Agents management > Summary**، يمكنك رؤية معلومات مثل حالة اتصال وكيل Wazuh والتطور التاريخي ضمن أطر زمنية محددة.

في ملف تكوين لوحة معلومات Wazuh، يمكنك تغيير الإعدادات للقيام بما يلي:

* تعطيل إدراج وإظهار بيانات حالة الاتصال للوكلاء. قم بتغيير **wazuh.monitoring.enabled** لتحقيق ذلك.
* تغيير تكرار إدراج بيانات حالة الاتصال للوكلاء. قم بتغيير **wazuh.monitoring.frequency** لتحقيق ذلك.

## فهارس wazuh-statistics-\*

تستخدم لوحة معلومات Wazuh فهارس **wazuh-statistics-**\* لعرض إحصائيات حول استخدام وأداء خادم Wazuh. تتضمن المعلومات المعروضة عدد الأحداث المفككة والبايتات المستلمة وجلسات TCP.

تقوم لوحة معلومات Wazuh بتشغيل طلبات إلى واجهة برمجة تطبيقات مدير Wazuh للاستعلام عن المعلومات المتعلقة بالاستخدام. تقوم بإدراج البيانات في فهارس **wazuh-statistics-**\* من المعلومات المجمعة. ينشئ مفهرس Wazuh فهرس **wazuh-statistics-**\* في الأسبوع بشكل افتراضي. راجع الوثائق حول فترة إنشاء الإحصائيات. تخزن هذه الفهارس إحصائيات خادم Wazuh كل 5 دقائق بشكل افتراضي. راجع الوثائق حول تكرار تنفيذ المهام.

لتصور هذه المعلومات في لوحة معلومات Wazuh، انتقل إلى **Server management > Statistics**.

## فهارس wazuh-states-vulnerabilities-\*

يُستخدم فهرس **wazuh-states-vulnerabilities-**\* في Wazuh لتخزين البيانات المتعلقة بحالة الثغرات الأمنية للأصول المراقبة. يحتوي هذا الفهرس عادةً على معلومات حول الثغرات الأمنية المكتشفة في نقاط النهاية المراقبة، بما في ذلك تفاصيل مثل الخطورة والحالة والبرامج المتأثرة ومرجع الثغرة الأمنية. يسمح الرمز \* في نهاية نمط الفهرس بإنشاء فهارس متعددة بأسماء مشابهة، مقسمة حسب الوقت أو عوامل أخرى. هذا يتيح التخزين والاسترجاع الفعال لبيانات الثغرات الأمنية عبر الزمن.

لتصور هذه المعلومات في لوحة معلومات Wazuh، انقر على **Vulnerability Detection** من الصفحة الرئيسية للوحة معلومات Wazuh.

## فهارس wazuh-states-inventory-hardware-\*

يحتوي فهرس **wazuh-states-inventory-hardware-**\* على جرد الأجهزة الأساسي المجمع من نقاط النهاية المراقبة. يمثل كل مستند في هذا الفهرس تفاصيل حول مكونات أجهزة نقطة النهاية، مثل وحدة المعالجة المركزية والذاكرة.

يوفر هذا الفهرس لفرق الأمان والمديرين رؤية للأجهزة الأساسية لكل نقطة نهاية. من خلال تتبع هذه المعلومات، يجعل Wazuh من الممكن اكتشاف تغييرات الأجهزة والتحقق من تكوينات الأصول ودعم جهود الامتثال أو التدقيق.

نظرًا لأن تفاصيل الأجهزة نادرًا ما تتغير في الظروف العادية، يمكن أن تكون التعديلات غير المتوقعة الملتقطة في هذا الفهرس إشارة إلى حالات شاذة. على سبيل المثال، إذا انتقلت إلى **IT Hygiene > System > Hardware**، فسترى معلومات تتعلق بالأجهزة.

## فهارس wazuh-states-inventory-hotfixes-\*

يخزن فهرس **wazuh-states-inventory-hotfixes-**\* معلومات حول تحديثات Windows (الإصلاحات السريعة) المثبتة على نقاط النهاية المراقبة. يتوافق كل إدخال في هذا الفهرس مع تحديث معين، بما في ذلك تفاصيل مثل معرّف الإصلاح السريع والوصف وتاريخ التثبيت.

يرتبط هذا الفهرس ارتباطًا وثيقًا بوحدة اكتشاف الثغرات الأمنية في Wazuh. من خلال المرجعية المتقاطعة للإصلاحات السريعة المثبتة مع قواعد بيانات الثغرات الأمنية المعروفة، يمكن لـ Wazuh تحديد الثغرات الأمنية التي تم إصلاحها بالفعل وأيها لا تزال غير مُصلحة على نقطة النهاية.

بالإضافة إلى إدارة الثغرات الأمنية، يساعد هذا الفهرس أيضًا المديرين على التحقق من امتثال تحديثات النظام، وتدقيق سجل التصحيحات، والتأكد من تطبيق التحديثات الحرجة باستمرار عبر بيئتهم.

للعثور على معلومات تتعلق بالإصلاحات السريعة على نقاط نهاية Windows، انتقل إلى **IT Hygiene > Software > Windows KBs**.

## فهارس wazuh-states-inventory-interfaces-\*

يخزن فهرس **wazuh-states-inventory-interfaces-**\* معلومات تفصيلية حول واجهات الشبكة على نقاط النهاية المراقبة. يسجل كل مستند سمات مثل حالة الواجهة (تعمل أو متوقفة)، وعنوان MAC، وإحصائيات نقل الحزم.

يمنح هذا الفهرس المديرين رؤية لواجهات الشبكة على أنظمتهم. من خلال تتبع نشاط الواجهة وتغييرات التكوين، نكتشف السلوك غير المعتاد مثل توقف الواجهات بشكل غير متوقع، أو ظهور واجهات جديدة، أو أنماط نقل حزم غير طبيعية.

بالإضافة إلى المساعدة في التحقيقات الأمنية، يُعد هذا الفهرس مفيدًا أيضًا للمراقبة التشغيلية وتخطيط السعة والتحقق من أن تكوينات الشبكة تظل متسقة مع المعايير التنظيمية.

للعثور على معلومات تتعلق بواجهة شبكة نقطة النهاية، انتقل إلى **IT Hygiene > Network > Interfaces**.

## فهارس wazuh-states-inventory-networks-\*

يحتوي فهرس **wazuh-states-inventory-networks-**\* على معلومات حول عناوين IPv4 و IPv6 المخصصة لواجهات الشبكة على نقاط النهاية المراقبة. يعرض كل سجل تفاصيل حول تعيين الواجهة لعنوان IP، مما يتيح الرؤية لكيفية اتصال نقطة النهاية بالشبكة.

هذا الفهرس قيّم لتتبع التغييرات في تكوينات الشبكة، مثل تعيين عناوين IP جديدة وإزالة القديمة. يمكن أن تشير مثل هذه التغييرات إلى إعادة تكوين مشروعة، ولكنها قد تشير أيضًا إلى تكوينات خاطئة أو نشاط مشبوه.

من خلال الحفاظ على هذا الجرد، يساعد Wazuh المديرين على التحقق من إعدادات الشبكة، ودعم متطلبات الامتثال، والتحقيق في حوادث الأمان حيث يكون سجل عناوين IP وتعييناتها مطلوبًا.

للعثور على معلومات تتعلق بتعيين عنوان IP لنقطة النهاية، انتقل إلى **IT Hygiene > Network > Addresses**.

## فهارس wazuh-states-inventory-packages-\*

يخزن فهرس **wazuh-states-inventory-packages-**\* معلومات حول حزم البرامج المثبتة حاليًا على نقاط النهاية المراقبة. يوضح كل سجل تفاصيل الحزمة ويتضمن تفاصيل مثل اسم الحزمة والإصدار والمورد.

يوفر هذا الفهرس الأساس لإدارة جرد البرامج داخل Wazuh. كما يمكّن المديرين من تتبع التغييرات في مجموعة البرامج، والتحقق من الامتثال للسياسات التنظيمية، واكتشاف وجود تطبيقات غير مصرح بها أو قديمة.

يُعد هذا الفهرس مكونًا حاسمًا في اكتشاف الثغرات الأمنية في Wazuh، والذي يقوم بالمرجعية المتقاطعة لإصدارات الحزم مع الثغرات الأمنية المعروفة لتحديد نقاط النهاية التي قد تكون معرضة للخطر. بهذه الطريقة، لا يساعد الفهرس في إدارة الأصول فحسب، بل يلعب أيضًا دورًا مباشرًا في تحسين الوضع الأمني.

للعثور على معلومات تتعلق بحزم البرامج لنقطة النهاية، انتقل إلى **IT Hygiene > Software > Packages**.

## فهارس wazuh-states-inventory-ports-\*

يسجل فهرس **wazuh-states-inventory-ports-**\* منافذ الشبكة المفتوحة المكتشفة على نقاط النهاية المراقبة. يتضمن تفاصيل مثل أرقام المنافذ والخدمات المرتبطة وحالات الاستماع.

من خلال الحفاظ على الرؤية للمنافذ المكشوفة، يساعد هذا الفهرس المديرين على تحديد الخدمات غير المصرح بها، وتتبع التغييرات في تعرض النظام، وتقليل سطح الهجوم. يمكن أن تكون المنافذ المفتوحة غير المصرح بها علامة مبكرة على الاختراق أو التكوين الخاطئ، مما يجعل هذه البيانات ضرورية لكل من المراقبة الأمنية وعمليات تدقيق الامتثال.

للعثور على معلومات تتعلق بالمنافذ على نقطة نهاية مراقبة، انتقل إلى **IT Hygiene > Network > Traffic**.

## فهارس wazuh-states-inventory-processes-\*

يحتوي فهرس **wazuh-states-inventory-processes-**\* على معلومات حول العمليات العاملة على نقاط النهاية المراقبة. يصف كل إدخال سمات مثل اسم العملية ومعرّف العملية (PID) والمستخدم المرتبط.

يتيح تتبع العمليات الجارية لـ Wazuh اكتشاف تنفيذ البرامج المشبوهة أو غير المصرح بها. هذه الرؤية حاسمة لاكتشاف البرامج الضارة وآليات الاستمرارية أو العمليات المارقة التي قد تتهرب من الدفاعات التقليدية. كما توفر للمديرين مرجعًا تاريخيًا لنشاط النظام، مما يدعم التحقيقات الجنائية.

للعثور على معلومات تتعلق بالعمليات على نقطة نهاية مراقبة، انتقل إلى **IT Hygiene > Processes**.

## فهارس wazuh-states-inventory-protocols-\*

يخزن فهرس **wazuh-states-inventory-protocols-**\* تفاصيل حول تكوين توجيه الشبكة والبروتوكولات المدعومة لكل واجهة شبكة على نقاط النهاية المراقبة. يتضمن ذلك أنواع البروتوكولات وجداول التوجيه وارتباطات الواجهة.

تمكّن مراقبة هذه المعلومات المؤسسات من التأكد من أن تكوينات الشبكة تتماشى مع الخطوط الأساسية المتوقعة. يمكن أن تشير تغييرات البروتوكول غير المتوقعة أو إدخالات التوجيه إلى تكوينات خاطئة أو نشاط ضار، مثل إعادة توجيه حركة المرور أو محاولات النفق.

للعثور على معلومات تتعلق بالبروتوكولات على نقطة نهاية مراقبة، انتقل إلى **IT Hygiene > Network > Protocols**.

## فهارس wazuh-states-inventory-system-\*

يوفر فهرس **wazuh-states-inventory-system-**\* تفاصيل على مستوى النظام حول كل نقطة نهاية مراقبة، بما في ذلك نظام التشغيل والإصدار واسم المضيف والبنية.

يعمل هذا الفهرس كسجل رئيسي لسمات النظام الأساسية، مما يدعم إدارة الجرد وتتبع الامتثال والتحقيقات الأمنية. كما يسمح للمديرين بتجميع التنبيهات وربطها بناءً على اسم النظام ونوع نظام التشغيل والبنية.

للعثور على معلومات على مستوى النظام تتعلق بنقاط النهاية المراقبة، انتقل إلى **IT Hygiene > System > OS**.

# إعادة الفهرسة

عندما يتم إجراء تغييرات على مخطط بيانات الفهرس، يصبح من الضروري إعادة فهرسة البيانات لتعكس هذه التغييرات. قد لا تتطابق البيانات الموجودة مع المخطط المحدث دون إعادة الفهرسة، مما يؤدي إلى تناقضات في البيانات أو أخطاء أثناء الاستعلامات. تتيح لك إعادة الفهرسة نسخ كل أو جزء من بياناتك من فهرس مصدر إلى فهرس وجهة.

لإعادة فهرسة فهرس موجود، قم بتنفيذ الخطوات التالية إما على لوحة تحكم Wazuh أو على خادم Wazuh.

## لوحة تحكم Wazuh

1. انقر على القائمة العلوية اليسرى ☰ وانتقل إلى **Indexer management** ثم **Dev Tools**.
2. أدخل استدعاء API التالي، مع استبدال my-source-index بنمط الفهرس المصدر و my-destination-index بنمط فهرس الوجهة.

POST /\_reindex

{

"source":{

"index":"my-source-index"

},

"dest":{

"index":"my-destination-index"

}

}

```

\*\*على سبيل المثال:\*\*

```

POST /\_reindex

{

"source":{

"index":"wazuh-alerts-\*"

},

"dest":{

"index":"example-alerts"

}

}

```

\*\*الناتج:\*\*

```

{

"took": 23655,

"timed\_out": false,

"total": 26849,

"updated": 0,

"created": 26849,

"deleted": 0,

"batches": 27,

"version\_conflicts": 0,

"noops": 0,

"retries": {

"bulk": 0,

"search": 0

},

"throttled\_millis": 0,

"requests\_per\_second": -1,

"throttled\_until\_millis": 0,

"failures": []

}

## واجهة سطر الأوامر

قم بتشغيل الأمر التالي على أي مكون مركزي من مكونات Wazuh المسموح له بالمصادقة على Wazuh API. استبدل <INDEXER\_USERNAME> و <INDEXER\_PASSWORD> باسم المستخدم وكلمة المرور الخاصة بالفهرس:

bash

curl -k -u "<INDEXER\_USERNAME>:<INDEXER\_PASSWORD>" -XPOST "https://<INDEXER\_IP\_ADDRESS>:9200/\_reindex" -H 'Content-Type: application/json' -d'

{

"source":{

"index":"my-source-index"

},

"dest":{

"index":"my-destination-index"

}

}'

**على سبيل المثال:**

bash

root@wazuh-server:~$ curl -k -u "INDEXER\_USERNAME:INDEXER\_PASSWORD" -XPOST "https://<INDEXER\_IP\_ADDRESS>:9200/\_reindex" -H 'Content-Type: application/json' -d'

{

"source":{

"index":"wazuh-alerts-\*"

},

"dest":{

"index":"example-alerts"

}

}'

**الناتج:**

json

{"took":18025,"timed\_out":false,"total":26854,"updated":26854,"created":0,"deleted":0

# ضبط Wazuh indexer

يوضح هذا الدليل كيفية تغيير الإعدادات لتحسين أداء Wazuh indexer. لتغيير كلمة مرور Wazuh indexer، راجع قسم إدارة كلمات المرور.

## قفل الذاكرة

## Shards و Replicas

### قفل الذاكرة

عندما يقوم النظام بمبادلة الذاكرة (swapping)، قد لا يعمل Wazuh indexer كما هو متوقع. لذلك، من المهم لسلامة عقدة Wazuh indexer ألا يتم مبادلة أي جزء من Java Virtual Machine (JVM) إلى القرص. لمنع مبادلة أي ذاكرة من Wazuh indexer، قم بتكوين Wazuh indexer لقفل مساحة عنوان العملية في RAM كما يلي.

**ملاحظة:** تحتاج إلى صلاحيات مستخدم root لتشغيل الأوامر الموضحة أدناه.

1. أضف السطر التالي إلى ملف التكوين /etc/wazuh-indexer/opensearch.yml على Wazuh indexer لتمكين قفل الذاكرة:

bootstrap.memory\_lock: true

1. قم بتعديل حد موارد النظام. يعتمد تكوين إعدادات النظام على نظام التشغيل الخاص بتثبيت Wazuh indexer.

**Systemd | SysVinit**

أنشئ دليلاً جديداً للملف الذي يحدد حدود النظام:

bash

mkdir -p /etc/systemd/system/wazuh-indexer.service.d/

قم بتشغيل الأمر التالي لإنشاء ملف wazuh-indexer.conf في الدليل الذي تم إنشاؤه حديثاً مع إضافة حد النظام الجديد:

bash

cat > /etc/systemd/system/wazuh-indexer.service.d/wazuh-indexer.conf << EOF

[Service]

LimitMEMLOCK=infinity

EOF

```

3. قم بتحرير ملف `/etc/wazuh-indexer/jvm.options` وغيّر أعلام JVM. قم بتعيين قيمة حجم heap لـ Wazuh indexer للحد من استخدام الذاكرة. تمنع حدود JVM heap استثناء OutOfMemory إذا حاول Wazuh indexer تخصيص ذاكرة أكثر من المتاحة بسبب التكوين في الخطوة السابقة. القيمة الموصى بها هي نصف ذاكرة RAM للنظام. على سبيل المثال، قم بتعيين الحجم كما يلي لنظام يحتوي على 8 GB من RAM.

```

-Xms4g

-Xmx4g

حيث إجمالي مساحة heap:

* -Xms4g - الحجم الأولي محدد بـ 4GB من RAM.
* -Xmx4g - الحجم الأقصى محدد بـ 4GB من RAM.

**تحذير:** لمنع تدهور الأداء بسبب تغيير حجم JVM heap أثناء التشغيل، يجب أن تكون قيم الحد الأدنى (Xms) والحد الأقصى (Xmx) متساوية.

1. أعد تشغيل خدمة Wazuh indexer:

bash

systemctl daemon-reload

systemctl restart wazuh-indexer

1. تحقق من أن الإعداد تم تغييره بنجاح، من خلال تشغيل الأمر التالي للتحقق من أن قيمة mlockall معينة على true:

bash

curl -k -u <INDEXER\_USERNAME>:<INDEXER\_PASSWORD> "https://<INDEXER\_IP\_ADDRESS>:9200/\_nodes?filter\_path=\*\*.mlockall&pretty"

**الناتج:**

json

{

"nodes" : {

"sRuGbIQRRfC54wzwIHjJWQ" : {

"process" : {

"mlockall" : true

}

}

}

}

```

إذا كان الناتج false، فقد فشل الطلب، ويظهر السطر التالي في ملف `/var/log/wazuh-indexer/wazuh-indexer.log`:

```

Unable to lock JVM Memory

### Shards و Replicas

يوفر Wazuh indexer إمكانية تقسيم index إلى أجزاء متعددة تسمى shards. كل shard هو "index" مستقل وظيفياً بالكامل يمكن استضافته على أي عقدة في مجموعة Wazuh indexer. التقسيم مهم لسببين رئيسيين:

* التوسع الأفقي.
* توزيع العمليات وتوازيها عبر shards، مما يزيد من الأداء والإنتاجية.

بالإضافة إلى ذلك، يسمح Wazuh indexer للمستخدمين بعمل نسخة واحدة أو أكثر من index shards فيما يسمى replica shards، أو replicas باختصار. النسخ المتماثل مهم لسببين:

* يوفر توافراً عالياً في حالة فشل shard أو عقدة.
* يسمح بتوسيع حجم البحث والإنتاجية حيث يمكن تنفيذ عمليات البحث على جميع replicas بالتوازي.

#### عدد Shards لـ index

قبل إنشاء أول index، ضع في اعتبارك بعناية عدد shards التي ستكون مطلوبة. لا يمكن تغيير عدد shards دون إعادة الفهرسة.

يعتمد عدد shards المطلوبة للحصول على أداء مثالي على عدد العقد في مجموعة Wazuh indexer. كقاعدة عامة، يجب أن يكون عدد shards مساوياً لعدد العقد. على سبيل المثال، يجب أن تحتوي مجموعة بها ثلاث عقد على ثلاثة shards، بينما تحتاج مجموعة بعقدة واحدة فقط إلى واحد.

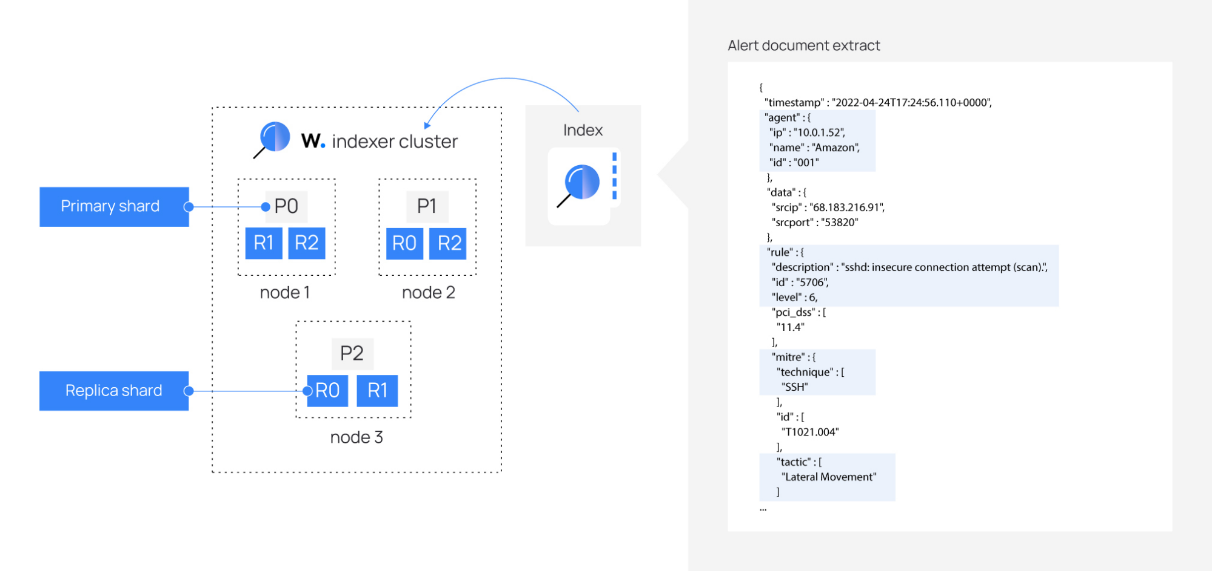
#### عدد Replicas لـ index

يعتمد عدد replicas على مساحة التخزين المتاحة للـ indices. فيما يلي مثال على كيفية إعداد مجموعة Wazuh indexer بثلاث عقد وثلاثة shards.

* **لا يوجد replica:** كل عقدة لديها shard واحد. إذا تعطلت عقدة واحدة، يتوفر index غير مكتمل يحتوي على shard اثنين فقط.
* **Replica واحد:** كل عقدة لديها shard واحد و replica واحد. إذا تعطلت عقدة واحدة، لا يزال index كامل متاحاً.
* **Replica اثنان:** كل عقدة لديها index كامل مع shard واحد و replica اثنان. مع هذا الإعداد، تستمر المجموعة في العمل حتى لو تعطلت عقدتان. على الرغم من أن هذا يبدو أفضل حل، إلا أنه يزيد من متطلبات التخزين.

توضح الصورة أدناه مجموعة Wazuh indexer بثلاث عقد، كل منها يحتوي على primary shard و replica shards اثنين.

**رسم توضيحي لمجموعة Wazuh indexer مع shards و replicas**



#### تعيين عدد Shards

**تحذير:** يتم تحديد عدد shards و replicas لكل index عند إنشاء index. بمجرد إنشاء index، على الرغم من إمكانية تغيير عدد replicas ديناميكياً، لا يمكن تغيير عدد shards دون إعادة الفهرسة.

ينشئ التثبيت الافتراضي لعقدة Wazuh indexer كل index بثلاثة primary shards ولا يوجد replicas. يمكنك تعديل عدد primary shards و replicas عن طريق تحميل template جديد باستخدام Wazuh API.

في المثال التالي، نقوم بتعيين عدد shards لـ Wazuh indexer أحادي العقدة إلى

1. قم بتنفيذ الخطوات التالية على عقدة Wazuh indexer أو أي مكون مركزي مسموح له بالمصادقة باستخدام Wazuh API.

1. قم بتنزيل template لـ Wazuh indexer:

bash

curl https://raw.githubusercontent.com/wazuh/wazuh/v4.13.1/extensions/elasticsearch/7.x/wazuh-template.json -o w-indexer-template.json

1. قم بتحرير w-indexer-template.json لتعيين index.number\_of\_shards إلى 1. لتجنب قيام Filebeat بالكتابة فوق template الموجود، قم بتعيين order إلى 1. تؤدي templates متطابقة متعددة بنفس الترتيب إلى ترتيب دمج غير حتمي.

json

{

"order": 1,

"index\_patterns": [

"wazuh-alerts-4.x-\*",

"wazuh-archives-4.x-\*"

],

"settings": {

"index.refresh\_interval": "5s",

"index.number\_of\_shards": "1",

"index.number\_of\_replicas": "0",

"index.auto\_expand\_replicas": "0-1",

"index.mapping.total\_fields.limit": 10000,

...

1. قم بتحميل الإعدادات الجديدة:

bash

curl -X PUT "https://<INDEXER\_IP\_ADDRESS>:9200/\_template/wazuh-custom" -H 'Content-Type: application/json' -d @w-indexer-template.json -k -u <INDEXER\_USERNAME>:<INDEXER\_PASSWORD>

**الناتج:**

json

{"acknowledged":true}

1. تأكد من تحديث التكوين بنجاح:

bash

curl "https://<INDEXER\_IP\_ADDRESS>:9200/\_template/wazuh-custom?pretty&filter\_path=wazuh-custom.settings" -k -u <INDEXER\_USERNAME>:<INDEXER\_PASSWORD>

**الناتج:**

json

{

"wazuh-custom" : {

"settings" : {

"index" : {

"mapping" : {

"total\_fields" : {

"limit" : "10000"

}

},

"refresh\_interval" : "5s",

"number\_of\_shards" : "1",

"auto\_expand\_replicas" : "0-1",

"number\_of\_replicas" : "0",

...

إذا كان index قد تم إنشاؤه بالفعل، فيجب إعادة فهرسته.

#### تعيين عدد Replicas

يمكن تغيير عدد replicas ديناميكياً باستخدام Wazuh indexer API. في مجموعة أحادية العقدة، يجب تعيين عدد replicas إلى صفر. يتم ذلك عن طريق تشغيل الأمر التالي على عقدة Wazuh indexer أو أي مكون مركزي مسموح له بالمصادقة باستخدام Wazuh API:

bash

curl -k -u "<INDEXER\_USERNAME>:<INDEXER\_PASSWORD>" -XPUT "https://<INDEXER\_IP\_ADDRESS>:9200/wazuh-alerts-\*/\_settings" -H 'Content-Type: application/json' -d'

{

"settings": {

"index": {

"number\_of\_replicas": 0

}

}

}'

# ترحيل Wazuh indices

في هذا القسم، نركز على ترحيل Wazuh indices باستخدام snapshots. يساعد هذا في استعادة التنبيهات من مجموعة Wazuh indexer إلى أخرى دون فقدان الطابع الزمني الأصلي.

## إعداد نظام الملفات المشترك

نوصي باستخدام Network File System (NFS) لإنشاء نظام ملفات مشترك لمستودع snapshot.

### خادم NFS

قم بتنفيذ الخطوات التالية لإعداد NFS على خادم مخصص:

1. أنشئ دليلاً مستهدفاً لمستودع snapshot في دليل /mnt:

bash

mkdir /mnt/snapshots

1. قم بتثبيت NFS عن طريق تشغيل الأوامر التالية:

**Yum | APT**

bash

yum update

yum install -y nfs-utils

yum install exportfs

systemctl enable nfs-server

systemctl start nfs-server

1. أضف دليل /mnt/snapshots إلى ملف /etc/exports باستخدام الأمر أدناه. استبدل متغير <NETWORK\_ADDRESS/CIDR> بعنوان شبكتك.

bash

echo "/mnt/snapshots <NETWORK\_ADDRESS/CIDR>(rw,sync,no\_root\_squash,no\_subtree\_check)" | sudo tee -a /etc/exports

حيث:

* **rw** - يسمح بالوصول للقراءة والكتابة إلى الدليل المشترك.
* **sync** - يجبر خادم NFS على كتابة التغييرات على القرص فوراً، مما يجعل نظام الملفات متزامناً.
* **no\_root\_squash** - يسمح لمستخدم "root" على نظام عميل NFS بالحصول على وصول كامل وغير مقيد إلى الملفات على خادم NFS.
* **no\_subtree\_check** - يعطل فحص الشجرة الفرعية، مما يمكن أن يحسن الأداء لأشجار الدلائل الكبيرة.

1. قم بتطبيق تكوين NFS:

bash

exportfs -a

### Wazuh indexer

قم بتنفيذ الخطوات التالية على عقدة (عقد) Wazuh indexer لإكمال إعداد نظام الملفات المشترك.

1. أنشئ دليلاً مستهدفاً لمستودع snapshot في دليل /mnt:

bash

mkdir /mnt/snapshots

1. قم بتثبيت عميل NFS:

**Yum | APT**

bash

yum -y install nfs-utils

1. قم بتحميل الدليل المشترك /mnt/snapshots على عقدة (عقد) Wazuh indexer. استبدل متغير <NFS\_SERVER\_IP> بعنوان IP لخادم NFS:

bash

mount -t nfs <NFS\_SERVER\_IP>:/mnt/snapshots /mnt/snapshots

1. امنح مستخدم wazuh-indexer ملكية دليل /mnt/snapshots:

bash

chown wazuh-indexer:wazuh-indexer /mnt/snapshots

1. أضف التكوين: path.repo: /mnt/snapshots إلى ملف /etc/wazuh-indexer/opensearch.yml لتحديد مسار المستودع:

yaml

network.host: "127.0.0.1"

node.name: "node-1"

cluster.initial\_master\_nodes:

- "node-1"

cluster.name: "wazuh-cluster"

node.max\_local\_storage\_nodes: "3"

path.data: /var/lib/wazuh-indexer

path.logs: /var/log/wazuh-indexer

path.repo: /mnt/snapshots

plugins.security.ssl.http.pemcert\_filepath: /etc/wazuh-indexer/certs/wazuh-indexer.pem

plugins.security.ssl.http.pemkey\_filepath: /etc/wazuh-indexer/certs/wazuh-indexer-key.pem

plugins.security.ssl.http.pemtrustedcas\_filepath: /etc/wazuh-indexer/certs/root-ca.pem

plugins.security.ssl.transport.pemcert\_filepath: /etc/wazuh-indexer/certs/wazuh-indexer.pem

plugins.security.ssl.transport.pemkey\_filepath: /etc/wazuh-indexer/certs/wazuh-indexer-key.pem

plugins.security.ssl.transport.pemtrustedcas\_filepath: /etc/wazuh-indexer/certs/root-ca.pem

plugins.security.ssl.http.enabled: true

plugins.security.ssl.transport.enforce\_hostname\_verification: false

plugins.security.ssl.transport.resolve\_hostname: false

plugins.security.ssl.http.enabled\_ciphers:

- "TLS\_ECDHE\_RSA\_WITH\_AES\_128\_GCM\_SHA256"

- "TLS\_ECDHE\_RSA\_WITH\_AES\_256\_GCM\_SHA384"

- "TLS\_ECDHE\_ECDSA\_WITH\_AES\_128\_GCM\_SHA256"

- "TLS\_ECDHE\_ECDSA\_WITH\_AES\_256\_GCM\_SHA384"

plugins.security.ssl.http.enabled\_protocols:

- "TLSv1.2"

plugins.security.authcz.admin\_dn:

- "CN=admin,OU=Wazuh,O=Wazuh,L=California,C=US"

plugins.security.check\_snapshot\_restore\_write\_privileges: true

plugins.security.enable\_snapshot\_restore\_privilege: true

plugins.security.nodes\_dn:

- "CN=indexer,OU=Wazuh,O=Wazuh,L=California,C=US"

plugins.security.restapi.roles\_enabled:

- "all\_access"

- "security\_rest\_api\_access"

plugins.security.system\_indices.enabled: true

plugins.security.system\_indices.indices: [".opendistro-alerting-config", ".opendistro-alerting-alert\*", ".opendistro-anomaly-results\*", ".opendistro-anomaly-detector\*", ".opendistro-anomaly-checkpoints", ".opendistro-anomaly-detection-state", ".opendistro-reports-\*", ".opendistro-notifications-\*", ".opendistro-notebooks", ".opendistro-asynchronous-search-response\*"]

*### خيار للسماح لـ Filebeat-oss 7.10.2 بالعمل ###*

compatibility.override\_main\_response\_version: true

1. أعد تشغيل Wazuh indexer لتطبيق تغييرات التكوين:

bash

systemctl restart wazuh-indexer

**تحذير:** تأكد من تأكيد أن دليل /mnt/snapshots يحتوي على ملكية wazuh-indexer:wazuh-indexer على عقد Wazuh indexer باستخدام أداة ll.

كرر خطوات **إعداد نظام الملفات المشترك > Wazuh indexer** على Wazuh indexer الوجهة لاستخدام دليل مشاركة NFS، /mnt/snapshots، كمستودع snapshot الخاص به.

## إعداد مستودع snapshot

على لوحة تحكم Wazuh، قم بتنفيذ الخطوات التالية:

1. انقر على القائمة العلوية اليسرى ☰، انتقل إلى **Indexer management > Snapshot Management > Repositories**، وحدد **Create repository** لإنشاء مستودع snapshot جديد.
2. أدخل اسم مستودع، حدد نوع المستودع **Shared file system**، أدخل موقع المستودع /mnt/snapshots، وحدد **Add** لتسجيل المستودع الجديد.

**إنشاء مستودع snapshot**

كرر الخطوات أعلاه على مجموعة Wazuh الوجهة لإعداد مستودع snapshot مماثل.

## أخذ snapshots

1. انقر على القائمة العلوية اليسرى ☰، وانتقل إلى **Indexer management > Snapshot Management > Snapshots**.
2. حدد **Take snapshot**، وأدخل **اسم Snapshot**.
3. حدد أو أدخل أنماط index المصدر.
4. حدد المستودع الذي تم إنشاؤه مسبقاً لتخزين snapshots.
5. حدد **Advanced options** وضع علامة على خيار **Include cluster state in snapshots**.

**خيار تضمين حالة cluster في snapshots**

1. حدد **Add** لإنشاء snapshot جديد.

يتم حفظ ملفات snapshot في موقع المستودع /mnt/snapshots.

**ملف Snapshot المحفوظ**

## استعادة snapshots

لإكمال خطوات ترحيل Wazuh indices، قم باستعادة snapshots المأخوذة من Wazuh indexers القديمة إلى Wazuh indexers الوجهة. قم بتنفيذ الخطوات التالية على Wazuh indexer الوجهة.

**ملاحظة:** من الضروري أن تكون قد نفذت الخطوات في قسمي **إعداد نظام الملفات المشترك** و**إعداد مستودع snapshot** على مجموعة Wazuh الوجهة قبل المتابعة إلى **استعادة snapshots**.

1. أعد تشغيل عقد Wazuh indexer في مجموعة Wazuh الوجهة لتحميل ملفات snapshot باستخدام الأمر:

bash

systemctl restart wazuh-indexer

1. انقر على القائمة العلوية اليسرى ☰، انتقل إلى **Indexer management > Snapshot Management > Snapshots**، وقم بتحديث صفحة Snapshots. ستظهر snapshots الموجودة في موقع المستودع /mnt/snapshots على لوحة تحكم مجموعة Wazuh الوجهة.
2. حدد snapshot وانقر على **Restore**. احذف بادئة restored\_ لاستعادة indices بأسمائها الأصلية. توجد بادئة restored\_ لتجنب تعارض أسماء index.
3. حدد **Advanced options** وتأكد من إلغاء تحديد جميع الخيارات.

**خيارات استعادة snapshot المتقدمة**

1. حدد **Restore snapshot** لإكمال عملية الترحيل.

**استعادة snapshot**

# تكوين Wazuh indexer على نقاط النهاية المحصنة

يستخدم Wazuh indexer مكتبة Java Native Access (JNA) لتنفيذ بعض التعليمات البرمجية الأصلية المعتمدة على النظام الأساسي. على Linux، يتم استخراج التعليمات البرمجية الأصلية الداعمة لهذه المكتبات في وقت التشغيل إلى دليل مؤقت ثم تعيينها في صفحات قابلة للتنفيذ في مساحة عنوان indexer. يتطلب هذا ألا تكون الملفات الأساسية على نظام ملفات مثبت بخيار noexec.

بشكل افتراضي، سينشئ Wazuh indexer دليله المؤقت داخل /tmp. ومع ذلك، تقوم بعض تثبيتات Linux المحصنة بتثبيت /tmp بخيار noexec بشكل افتراضي. هذا يمنع JNA من العمل بشكل صحيح.

## ضمان أذونات التنفيذ في دليل temp لـ Wazuh indexer

لحل هذه المشكلة، إما قم بإزالة خيار noexec من نظام ملفات /tmp الخاص بك أو قم بتكوين Wazuh indexer لاستخدام موقع مختلف. اتبع الخطوات أدناه لتغيير الدليل المؤقت لـ Wazuh indexer عن طريق تعيين متغير البيئة $OPENSEARCH\_TMPDIR.

**ملاحظة:** تحتاج إلى صلاحيات مستخدم root لتشغيل جميع الأوامر الموضحة أدناه.

1. أنشئ الدليل المؤقت وقم بتعيين الأذونات المناسبة:

bash

mkdir /var/lib/wazuh-indexer/tmp

bash

chown wazuh-indexer:wazuh-indexer /var/lib/wazuh-indexer/tmp

```

2. أضف `Environment=OPENSEARCH\_TMPDIR=/var/lib/wazuh-indexer/tmp` إلى ملف `/lib/systemd/system/wazuh-indexer.service`، وهو ملف تكوين systemd لـ Wazuh indexer. يجب أن يكون ملف التكوين مشابهاً لما يلي:

```

[Service]

Type=notify

RuntimeDirectory=wazuh-indexer

PrivateTmp=true

Environment=OPENSEARCH\_HOME=/usr/share/wazuh-indexer

Environment=OPENSEARCH\_TMPDIR=/var/lib/wazuh-indexer/tmp

Environment=OPENSEARCH\_PATH\_CONF=/etc/wazuh-indexer

Environment=PID\_DIR=/run/wazuh-indexer

Environment=OPENSEARCH\_SD\_NOTIFY=true

EnvironmentFile=-/etc/default/wazuh-indexer

1. أعد تشغيل خدمة Wazuh indexer لتطبيق التغييرات:

bash

systemctl restart wazuh-indexer

## معالجة عمليات إعادة التشغيل غير المرغوب فيها لـ Wazuh indexer على Ubuntu

يؤدي تعديل دليل Java المؤقت لـ Wazuh indexer على بعض نقاط نهاية Ubuntu إلى اكتشاف needrestart لعمليات Java العادية في الدليل على أنها تغييرات في المكتبات. ونتيجة لذلك، يقوم needrestart بشكل غير صحيح بوضع علامة على خدمة Wazuh indexer على أنها تستخدم مكتبات قديمة وإما يطالب بإعادة التشغيل أو يعيد تشغيل خدمة Wazuh indexer تلقائياً. يحدث هذا حتى عندما تكون تحديثات حزم النظام غير مرتبطة بـ Wazuh indexer. الحل البديل هو استبعاد خدمة Wazuh indexer من فحوصات needrestart باستخدام الأمر أدناه:

bash

echo '$nrconf{blacklist\_rc} = [ qr(^wazuh-indexer) ];' > "/etc/needrestart/conf.d/wazuh-indexer.conf"

**ملاحظة:** سيجعل هذا الإعداد needrestart يتجاهل دائماً خدمة Wazuh indexer، حتى في الحالات التي تكون فيها إعادة التشغيل مشروعة. لذلك، قد يختار المستخدمون تطبيقه وفقاً لتقديرهم الخاص.