ENERGY LOGSERVER

SIEM **FOR BEGINNERS**

ВСЕ, ЩО ВИ ХОТІЛИ ЗНАТИ ПРО КЕРУВАННЯ ЛОГАМИ,
АЛЕ БОЯЛИСЯ ЗАПИТАТИ

www.energylogserver.com

Еволюція технологій SIM, SEM, SIEM

ENERGY LOGSERVER

SIEM – це клас рішень, який часто є недооціненим, залишаючись у всіх на вустах. Завдання **SIEM** включають в себе централізований збір і керування подіями безпеки з різних систем. Йому передував ряд рішень, які прагнули спростити і автоматизувати контроль засобів безпеки.

Два поширені типи таких систем:

- SIM (Security Information Management) – система керування логами
- SEM (Security Event Management) – система керування подіями безпеки

Хоча визначення і схожі, між ними є величезна різниця:

1. Це скоріше еволюція однієї ідеї для збору подій, ніж два варіанти її втілення.
2. SIM автоматизує збір журналів, SEM – будь-які заходи безпеки в принципі. Через це джерела інформації різні: у SIM це програмні та апаратні рішення, а у SEM – бази даних Oracle і MySQL.
3. SIM просто дозволяє спростити пошук логів за допомогою автоматичного збирання та сортування. У SEM є функції контекстного аналізу та кореляції зібраних даних – він шукає події зі спільним знаменником і може скласти з них висновок про загрозу чи проблему.

SIEM (Security Information and Event Management) поєднує функції двох зазначених систем, дозволяючи централізовано керувати інформацією та подіями, що генеруються іншими рішеннями безпеки.

Він збирає та аналізує записи журналів, які генерують ваші активи. Все, що записують ваші антивіруси, файрволи, системи запобігання витоку даних, захисти ендпоінтів, керування привілейованим доступом тощо, тепер зібрано в одному єдиному рішенні – **SIEM**.

Чим зумовлена поява SIEM та потреба у рішеннях цього класу?

ENERGY LOGSERVER

Зростання кількості інформації, яку мають аналізувати спеціалісти. У період диджиталізації компанії усвідомили, що типові завдання постійно ускладнюються. Для кожного з них тепер потрібно розуміти і пов'язувати стільки подій, що порівняти їх правильно і зробити висновок на ходу неможливо.

Атаки ускладнюються, а отже додається кількість маркерів, які їх видають. Сучасні загрози включають відразу кілька векторів – від банального фішингу до впровадження інтелектуального шкідливого ПЗ, а самі зловмисники якісно маскують свої дії і обходять системи захисту. Таким чином, жодне окреме рішення безпеки не дасть вам вичерпної інформації про те, що відбувається щось погане.

При цьому кожне окреме рішення має вузьку специфікацію та надає вам інформацію поза контекстом. Інструменти аналізують факти без прив'язки до ситуації, де вони відбуваються. Цього стає замало, бо жодна атака не відбувається у якихось ізольованих рамках. Атака – не подія, а їхня сукупність.

Порівняйте: співробітник у відпустці швидко перевіряє корпоративну пошту – хакер краде його дані й теж заходить у пошту. Для нас з вами це дві абсолютно різні ситуації, як у плані важливості, так і самої суті, не кажучи вже про наслідки та реакцію.

А для систем контролю доступу це просто два входи в обліковий запис, з різних IP- та MAC-адрес. Масштабувавши ситуацію на десятки співробітників та сотні процесів, отримаємо безпеку, повну непотрібної, а найчастіше й марної рутини. Щоб перевірити всі подібні підозрілі випадки, вам необхідно розібратися в подальших діях користувача та змін у системі: перевірити запити на доступ до файлів, завантаження підозрілого ПЗ, сканування мережі та спроби підвищити рівень доступу. Все це знаходиться в різних рішеннях, а отже, потрібно перевірити кожне з них.

Ваші рішення не можуть самостійно виявити атаку, але надають всю необхідну для цього інформацію. Тепер потрібно лише знайти її.

Розслідуємо інцидент разом із SIEM

ENERGY LOGSERVER

Пропонуємо розібрати на практиці, які процеси відбуваються всередині рішення, і як вони призводять до результату. Інформація нижче – покрокова демонстрація розв’язування завдання розслідування інциденту.

Ми відтворили реальну атаку на окремому хості у тестовій інфраструктурі. За задумом, все почалося з того, що з атакованого комп'ютера надійшло сповіщення про підозрілий інцидент. Наше завдання – розслідувати цей інцидент як ІБ-спеціаліст. Спробуємо дізнатися, чи є ця подія атакою. Якщо так – деталізуємо її до дрібниць, якщо ні – дізнаємося, що сталося та як уникнути таких тривог.

Ще одна важлива умова: вважатимемо, що прогалин у системі захисту немає і **SIEM** правильно інтегрована у всі рішення. Іншими словами, розв’язати поставлене завдання можливо, інформація в системі про це є, потрібно просто її знайти.

Приймаємо умови та переходимо безпосередньо до **SIEM**.

Збір логів та інформації з системи:

ENERGY LOGSERVER

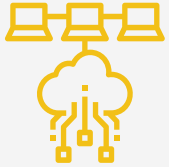
В основі будь-якої **SIEM** знаходиться Log Management – централізоване автоматизоване збирання записів журналів в одному місці. Ми вже з'ясували, що виявити атаку ми зможемо за сукупністю подій, а отже, першим етапом нам потрібні дані з логів кожного рішення.

Інструменти ІБ:

- файрволи (VPN-концентратори, вебфільтри
- IDS/IPS
- захист ендпоінтів (антивіруси, EDR та ін.)
- DLP
- PAM, WAF, MFA

Дані про активи:

- конфігурація
- розташування
- користувачі
- порти та протоколи
- звіти про вразливості
- інвентаризація ПЗ



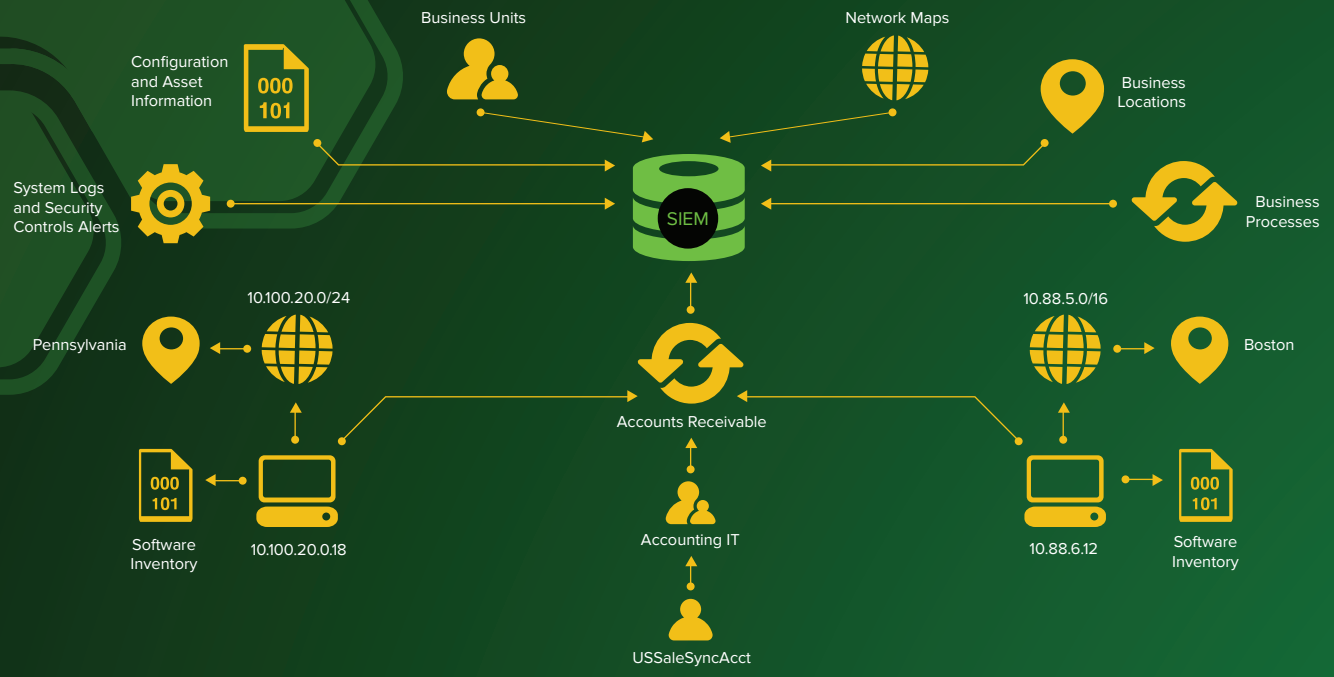
Інфраструктура:

- маршрутизатори, комутатори, точки доступу
- контролери доменів
- сервери додатків, корпоративні портали та сервіси
- бази даних
- журнали ОС та хмарних сервісів

Як генеруються логи у вашій мережі

ENERGY LOGSERVER

Кожне перераховане вище джерело створює десятки та сотні записів щодня, і цього не змінити. Як генеруються логи у вашій мережі – показано на схемі нижче.



10.100.20.18 ініціював копіювання бази даних з використанням облікових даних USSalesSyncAcct на віддаленому хості 10.88.6.12 - Status Code 0x44F8

А тепер уявіть, що схожі процеси повторюються тисячі разів на день, створюючи цілий масив інформації, у тому числі корисної. Без **SIEM** все це буде лише мертвим вантажем, у якому не розібратися вручну.

Як це працює?

За допомогою лог-менеджменту ми можемо дістати зі сховища і переглянути лог цієї події, який сформований на комп'ютері, що атакується.

Виглядає він таким чином:

5ssh:nottyroot192.168.202.49□b(15ssh:nottyroot192.168.202.49□b(15ssh:nottyroot192.168.202.49□b(15ssh:nottyroot192.168.202.49□b(15ssh:nottyroot192.168.202.49□b(15ssh:nottyroot192.168.202.49□b(15ssh:nottyroot192.168.202.49□b(15ssh:nottyroot192.168.202.49□
b(15ssh:nottyroot192.168.202.49□b(15ssh:nottyroot192.168.202.49□b(15ssh:nottyroot192.168.202.49□b(15ssh:nottyroot192.168.202.49□b(15ssh:nottyroot192.168.202.49□b(15ssh:nottyroot192.168.202.49□b(15ssh:nottyroot192.168.202.49□b(15ssh:nottyroot192.168.202.49□b(1
5ssh:nottyroot192.168.202.49□b(15ssh:nottyroot192.168.202.49□b(15ssh:nottyroot192.168.202.49□b(15ssh:nottyroot192.168.202.49□b(15ssh:nottyroot192.168.202.49□b(15ssh:nottyroot192.168.202.49□b(15ssh:nottyroot192.168.202.49ý¹b(15ssh:nottyroot192.168.202.49ý¹b(1¸5s
sh:nottyroot192.168.202.49ÿ¹b(1·5ssh:nottyroot192.168.202.49ÿ¹b(1½5ssh:nottyroot192.168.202.49ºb(1¿5ssh:nottyroot192.168.202.49ºb(1¸5ssh:nottyroot192.168.202.49ºb(1·5ssh:nottyroot192.168.202.49ºb(1 ¿5ssh:nottyroot192.168.202.49ºb(1½5ssh:nottyroot192.168.202.49º
b(1¸5ssh:nottyroot192.168.202.49ºb(1·5ssh:nottyroot192.168.202.49ºb(1¿5ssh:nottyroot192.168.202.49ºb(1½5ssh:nottyroot192.168.202.49 ºb(1¸5ssh:nottyroot192.168.202.49ºb(1·5ssh:nottyroot192.168.202.49ºb(1¿5ssh:nottyroot192.168.202.49ºb(1½5ssh:nottyroot192.168.202.
49ºb(1¸5ssh:nottyroot192.168.202.49ºb(1·5ssh:nottyroot192.168.202.49ºb(1¿5ssh:nottyroot192.168.202.4ºb(1½5ssh:nottyroot192.168.202.4ºb(1¸5ssh:nottyroot192.168.202.4ºb(1·5ssh:nottyroot192.168.202.4ºb(1¿5ssh:nottyroot192.168.202.49 ºb(1½5ssh:nottyroot192.168.202.49
ºb(15ssh:nottyroot192.168.202.49(ºb(15ssh:nottyroot192.168.202.49(ºb(15ssh:nottyroot192.168.202.49*ºb(15ssh:nottyroot192.168.202.49*ºb(15ssh:nottyroot192.168.202.49*ºb(15ssh:nottyroot192.168.202.49*ºb(15ssh:nottyroot192.168.202.49,ºb(15ssh:nottyroot192.168.202.49
,ºb(15ssh:nottyroot192.168.202.49,ºb(15ssh:nottyroot192.168.202.49,ºb(15ssh:nottyroot192.168.202.49.ºb(15ssh:nottyroot192.168.202.49.ºb(15ssh:nottyroot192.168.202.49.ºb(15ssh:nottyroot192.168.202.49.ºb(15ssh:nottyroot192.168.202.490ºb(15ssh:nottyroot192.168.202.4
90ºb(15ssh:nottyroot192.168.202.490ºb(15ssh:nottyroot192.168.202.490ºb(15ssh:nottyroot192.168.202.491ºb(15ssh:nottyroot192.168.202.491ºb(15ssh:nottyroot192.168.202.492ºb(15ssh:nottyroot192.168.202.492ºb(15ssh:nottyroot192.168.202.494ºb(15ssh:nottyroot192.168
.202.494ºb(16ssh:nottyroot192.168.202.49Rºb(16ssh:nottyroot192.168.202.49Rºb(16ssh:nottyroot192.168.202.49Sºb(16ssh:nottyroot192.168.202.49Sºb(16ssh:nottyroot192.168.202.49Tºb(1 6ssh:nottyroot192.168.202.49Tºb(16ssh:nottyroot192.168.202.49Vºb(1 6ssh:nottyroo
t192.168.202.49Vºb(16ssh:nottyroot192.168.202.49Vºb(16ssh:nottyroot192.168.202.49Vºb(1 6ssh:nottyroot192.168.202.49Wºb(16ssh:nottyroot192.168.202.49Wºb(16ssh:nottyroot192.168.202.49Xºb(16ssh:nottyroot192.168.202.49Xºb(1 6ssh:nottyroot192.168.202.49Yºb(1 6ss
h:nottyroot192.168.202.49Yºb(16ssh:nottyroot192.168.202.49Zºb(16ssh:nottyroot192.168.202.49Zºb(16ssh:nottyroot192.168.202.49[ºb(1 6ssh:nottyroot192.168.202.49[ºb(16ssh:nottyroot192.168.202.49\ºb(16ssh:nottyroot192.168.202.49\ºb(16ssh:nottyroot192.168.202.49\ºb(1
6ssh:nottyroot192.168.202.49\ºb(1>6ssh:nottyroot192.168.202.49|ºb(1?6ssh:nottyroot192.168.202.49|ºb(1D6ssh:nottyroot192.168.202.49}ºb(1F6ssh:nottyroot192.168.202.49}ºb(1>6ssh:nottyroot192.168.202.49}ºb(1?6ssh:nottyroot192.168.202.49}ºb(1D6ssh:nottyroot192.168.20
2.4ºb(1F6ssh:nottyroot192.168.202.4ºb(1>6ssh:nottyroot192.168.202.4ºb(1?6ssh:nottyroot192.168.202.4ºb(1D6ssh:nottyroot192.168.202.49ºb(1F6ssh:nottyroot192.168.202.49ºb(1>6ssh:nottyroot192.168.202.49ºb?6ssh:nottyroot192.168.202.49ºb(1¼Hssh:nottyroot192.168.20
2.49¿ýb(1½Hssh:nottyroot192.168.202.49¿ýb(1»Hssh:nottyroot192.168.202.49¿ýb(1¾Hssh:nottyroot192.168.202.49¿ýb(1¼Hssh:nottyroot192.168.202.49ëýb(1½Hssh:nottyroot192.168.202.49ëýb(1»Hssh:nottyroot192.168.202.49ëýb(1¾Hssh:nottyroot192.168.202.49ëýb(1 ¼Hss
h:nottyroot192.168.202.49ëýb(1½Hssh:nottyroot192.168.202.49ëýb(1»Hssh:nottyroot192.168.202.49ëýb(1¾Hssh:nottyroot192.168.202.49ëýb(1¼Hssh:nottyroot192.168.202.49ëýb(1½Hssh:nottyroot192.168.202.49ëýb(1»Hssh:nottyroot192.168.202.49ëýb(1¾Hssh:nottyroot192
.168.202.49ëýb(1¼Hssh:nottyroot192.168.202.49ëýb(1»Hssh:nottyroot192.168.202.49ëýb(1½Hssh:nottyroot192.168.202.49ëýb(1¾Hssh:nottyroot192.168.202.49ëýb(1¼Hssh:nottyroot192.168.202.49ëýb(1»Hssh:nottyroot192.168.202.49ëýb(1½Hssh:nottyroot192.168.202.49ëý
b(1¾Hssh:nottyroot192.168.202.49ëýb(1Hssh:nottyroot192.168.202.49ëýb(1Hssh:nottyroot192.168.202.49ëýb(1Hssh:nottyroot192.168.202.49ëýb(1Hssh:nottyroot192.168.202.49ëýb(1Hssh:nottyroot192.168.202.49ëýb(1Hssh:nottyroot192.168.202.49ëýb(1Hssh:nottyroot192.16
8.202.49ëýb(1Hssh:nottyroot192.168.202.49ëýb(1Hssh:nottyroot192.168.202.49ëýb(1Hssh:nottyroot192.168.202.49ëýb(1Hssh:nottyroot192.168.202.49ëýb(1Hssh:nottyroot192.168.202.49ëýb(1Hssh:nottyroot192.168.202.49ëýb(1Hssh:nottyroot192.168.202.49ëýb(1Hssh:nottyr
oot192.168.202.49ëýb(1Hssh:nottyroot192.168.202.49ëýb(1Hssh:nottyroot192.168.202.49ëýb(1Hssh:nottyroot192.168.202.49ëýb(1Hssh:nottyroot192.168.202.49ëýb(1Hssh:nottyroot192.168.202.49ëýb(1Hssh:nottyroot192.168.202.49ëýb(1Hssh:nottyroot192.168.202.49ëýb(1H

З цього прикладу стає ясно, що аналізувати логи самотужки марно. Тим паче, що в реальній компанії таких логів буде в кілька десятків-сотень-тисяч разів більше, і якщо один у теорії можна побороти, то працювати так постійно — на жаль, ні.

Висновок: Log Management допомагає нам знайти старі записи, але цього недостатньо для отримання цінної інформації. Добре, що з цього **SIEM** тільки починається.

Готуємо дані до аналізу

ENERGY LOGSERVER

Усі джерела логів зазвичай записують події по-своєму. Передбачається, що ці записи читатимуться людьми, а не машиною, тому різний формат логів не заведено вважати проблемою. Але в нашому випадку важливо отримати можливість обробляти записи інструментами **SIEM**, а значить рішення має спершу привести всі наявні логи до спільного знаменника.

Наприклад, в логі

> **“User Broberts Successfully Authenticated to 10.100.52.105 from client 10.10.8.22”**

необхідно виділити загальні для всіх логів елементи

> **“User [USERNAME] [STATUS] Authenticated to [DESTIP] from client [SOURCEIP]”**

Цей процес називається нормалізація. Завдяки їй **SIEM** обробляє величезну кількість записів на рівні математичних моделей та методів статистики. Простіше кажучи, фільтрує та сортує логи. Різниця між лог-менеджментом з нормалізацією і без в тому, що без нормалізації ми можемо обробляти лише логи з одного джерела. Відфільтрувати логи різних джерел за якимось загальним критерієм у нас не вийде.

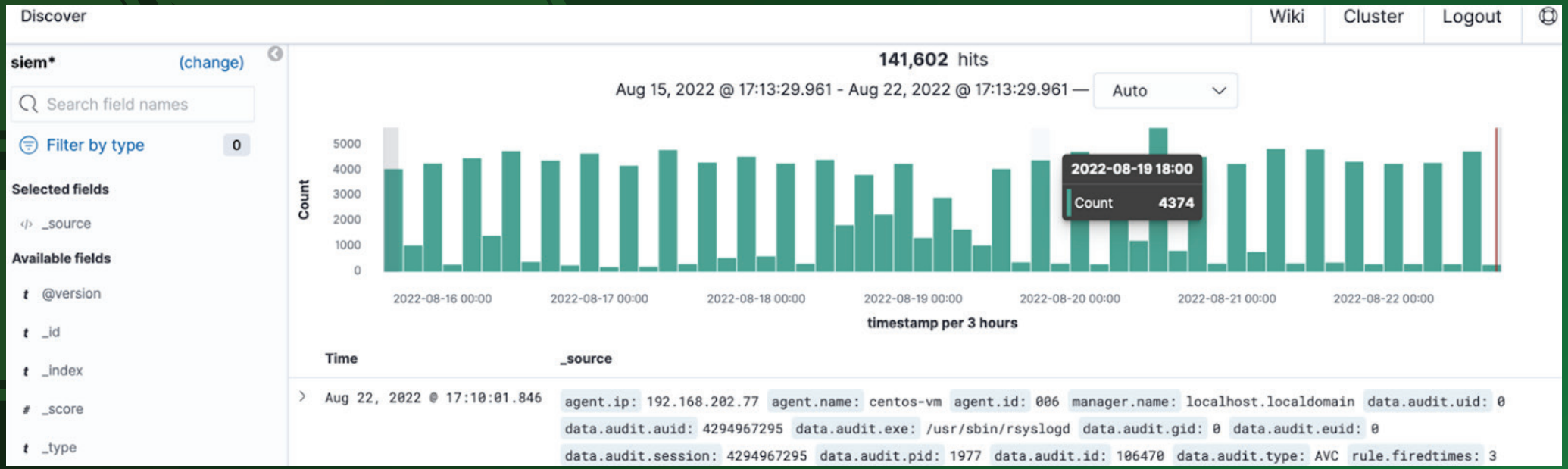
Це нам і потрібно для наступних дій, адже суть **SIEM** – доповнити інформацію з одних джерел даними з інших.

Як це працює?

ENERGY LOGSERVER

Отримавши можливість додати трохи статистики та фільтрувати дані, з'являється можливість структурувати лог і щось роздивитись.

Ось як це виглядає у консолі:



Ми бачимо спроби входу до облікового запису, які не пройшли через неправильно введені облікові дані. **SIEM** зафіксував час, IP-адреси та інші параметри запитів. Ми бачимо, що вони повторювалися багато разів, бачимо використані паролі, а також відзначаємо різницю в мілісекунди між запитами.

Це ознака машинного підбору пароля, але навіщо робити висновки самостійно, якщо у нас є **SIEM**? Розслідуємо далі.

Суть процесу кореляції

ENERGY LOGSERVER

Кореляція дозволяє знайти взаємозв'язок між різними подіями, зв'язати їх в один ланцюжок та порівняти зі станом норми. У результаті ми отримуємо класифіковані події за прийнятими нами критеріями (ризик-менеджмент, alerts, пріоритизація тощо). А якщо додати до рівняння інтеграцію з базами індикаторів компрометацій, матимемо можливість більш ефективного виявлення загроз за допомогою **SIEM**, зокрема атак нульового дня.

Правила детектування, інтегровані із системою оповіщення та реагування, дозволяють налаштовувати сценарії реакцій на події, які не повинні відбуватися у вашій системі. Правильно налаштована система зможе виокремлювати з усього різноманіття логів маркерні події і повідомляти вас про потенційну загрозу.

Якщо одна подія від конкретного рішення може бути як абсолютно нешкідливою, так і свідчити про загрозу, кілька таких свідчень з різних джерел вже говорять про факт атаки. Наявн ість інформації про нормальний стан системи створює контекст для подій, що аналізуються — система не оцінює події за принципом «добре це чи погано?», а знаходить відповідь на запитання
«Чи це має відбуватися в **поточних умовах?»**

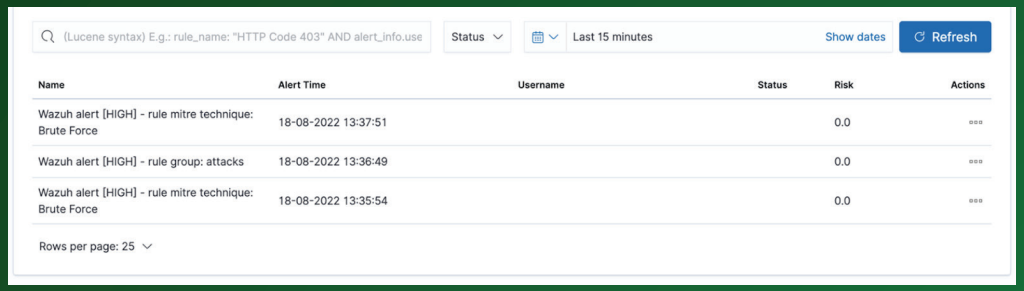
Як це працює?

ENERGY LOGSERVER

Нормалізація та кореляція дають нам можливість робити що завгодно з нашим поки що незрозумілим логом. Тож давайте зробимо щось корисне!

Включаємо оповіщення на тип атак Brute Force

Це одразу дає нам перелік спрацьовувань правила (з можливістю заглибитись у кожен інцидент):



— Стає простіше, чи не так?

Але ще не все – ми вже знайшли та виділили подію, точно бачимо, що у нас була атака типу Brute Force. Але нам цього замало.

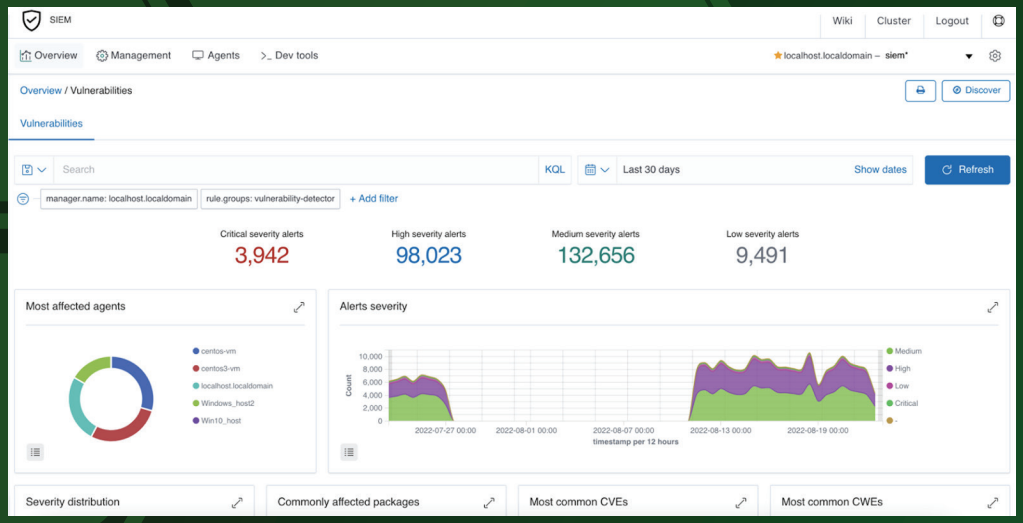
На секунду повернемося в загальну картину, подивимося, як справи загалом, і пірнемо назад в наш інцидент.

— Цього разу — глибше.

Як це працює?

ENERGY LOGSERVER

Загальний дашборд за всіма подіями: досить побіжного погляду, щоб оцінити ситуацію та отримати важливі метрики. Зрозуміло, все це можна налаштовувати та кастомізувати під свої завдання.



Виділяємо атаки на певний хост:

Як це працює?

ENERGY LOGSERVER

Далі клікаємо на нашу подію та вивчаємо деталі.

Наприклад, з якого джерела на який була зроблена атака...

Incident ID: -6Uer4IB9B1_si7K3kNm
Rule Name: Wazuh alert [HIGH] - rule mitre technique: Brute Force

alert_time	2022-08-18T04:03:32.019871Z
match_body.@src_ip	192.168.202.49
match_body.@timestamp	2022-08-18T04:02:33.319Z
match_body.@version	1
match_body.GeoLocation	{}
match_body._id	qaUdr4IB9B1_si7K-0Cu
match_body._index	siem-2022.08
match_body._type	_doc
match_body.agent.id	007
match_body.agent.ip	192.168.202.79

...тут – інформацію про те, що паролі були помилковими, тобто атака не пройшла...

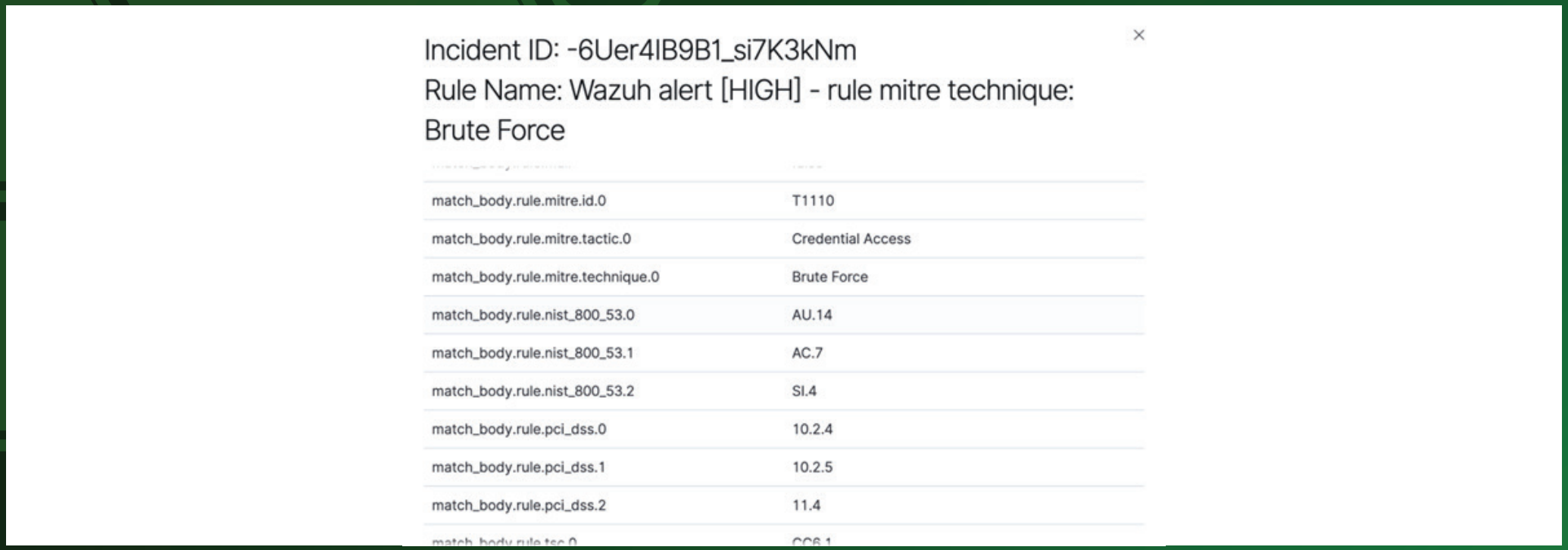
Incident ID: -6Uer4IB9B1_si7K3kNm
Rule Name: Wazuh alert [HIGH] - rule mitre technique: Brute Force

match_body.predecoder.timestamp	Aug 18 00:02:32
match_body.previous_output	Aug 18 00:02:32 centos3-vm sshd[13707]: Failed password for root from 192.168.202.49 port 37026 ssh2 Aug 18 00:02:31 centos3-vm sshd[13708]: Failed password for root from 192.168.202.49 port 37030 ssh2 Aug 18 00:02:31 centos3-vm sshd[13705]: Failed password for root from 192.168.202.49 port 37024 ssh2 Aug 18 00:02:30 centos3-vm sshd[13707]: Failed password for root from 192.168.202.49 port 37026 ssh2 Aug 18 00:02:29 centos3-vm sshd[13708]: Failed password for root from 192.168.202.49 port 37030 ssh2 Aug 18 00:02:29 centos3-vm sshd[13706]: Failed password for root from 192.168.202.49 port 37028 ssh2 Aug 18 00:02:29 centos3-vm sshd[13705]: Failed password for root from 192.168.202.49 port 37024 ssh2

Як це працює?

ENERGY LOGSERVER

...ну а тут – який метод атаки був використаний:



Incident ID: -6Uer4IB9B1_si7K3kNm
Rule Name: Wazuh alert [HIGH] - rule mitre technique: Brute Force

match_body.rule.mitre.id.0	T1110
match_body.rule.mitre.tactic.0	Credential Access
match_body.rule.mitre.technique.0	Brute Force
match_body.rule.nist_800_53.0	AU.14
match_body.rule.nist_800_53.1	AC.7
match_body.rule.nist_800_53.2	SI.4
match_body.rule.pci_dss.0	10.2.4
match_body.rule.pci_dss.1	10.2.5
match_body.rule.pci_dss.2	11.4

У відверто незрозумілому наборі символів, який ми отримали з хоста, ми виявили та розглянули атаку до найдрібніших деталей.

Тепер дайте відповідь на запитання — чи легко було б досягти цього ж результату без **SIEM**?

Чим це корисно?

ENERGY LOGSERVER

Уявіть, що компанія генерує сотні ГБ текстових логів щодня. У всьому цьому масиві ховається саме та інформація, яка вкаже на атаку чи критичну вразливість. Але тепер ви бачите самі, що обробити навіть один лог — це серйозне завдання. Навіть якщо найняти цілу армію фахівців, єдиним завданням якої буде лише читання логів – людський фактор рано чи пізно вплине на процес.

До речі, саме розширенням відділів безпеки іноді намагаються компенсувати нестачу можливостей щодо аналізу наявних подій. Більше людей зроблять більше роботи – це факт. Але з урахуванням кількості подібних завдань, як у нашому прикладі, таке використання людського ресурсу все рівно залишається неефективним.

SIEM-система – це інструмент, який позбавить фахівців від рутини та звільнить час на стратегію, аналіз та реагування. **SIEM** за лічені секунди опрацьовує тисячі таких випадків, як у нашому прикладі. Так команда отримає час та сили для оптимізації інфраструктури, реагування на атаки та впровадження того, що сприятиме покращенням. Альтернатива – розбиратися днями в купі подій та хибнопозитивних спрацьовувань, що в результаті призведе лише до підтримки поточної норми.

SIEM зробить ефективнішою роботу всіх рішень, оскільки важливі події, виявлені ними, а також інша корисна інформація не пройдуть повз фахівців. У той же час самі фахівці будуть продуктивнішими, адже тепер їхньої уваги вистачить на все, що дійсно потребує їхнього втручання.

Energy Logserver — SIEM нового покоління

ENERGY LOGSERVER

SIEM від Energy Logserver вдалося стерти верхню межу можливостей. Рішення доступне у трьох базових конфігураціях на вибір, кожна з яких має певний набір функцій для вирішення задач різного рівня. Цю основу можна масштабувати та розширювати як завгодно без будь-яких обмежень.

Log Management — все про управління журналами

- Рольова модель доступу
- Багатовимірна система сповіщень і звітності
- Інтеграція з LDAP, AD, Radius, SSO
- Багаторівнева система архівації даних
- Масштабована архітектура та кластеризація
- Гнучкий конструктор дашбордів та готові шаблони
- Сотні готових парсерів, гнучкий конструктор

SIEM — розширене управління безпекою

- Поведінковий аналіз користувачів і пристроїв
- Динамічна інтеграція з базами IoC, TTP, Threat Intellegence, MITRE ATT&CK
- GDPR, NIST, CIS, PCI DSS, HIPAA compliance
- File Integrity Monitoring (FIM)
- Сканер вразливостей
- 1000+ правил детектування та кореляцій
- Допомога у виявленні підозрілої поведінки на базі штучного інтелекту
- Playbooks
- Моніторинг роботи додатків та сервісів
- Інтегрована система керування ризиками
- Система керування інцидентами

Network Probe — мережевий аналіз

- Детальний аналіз мережевого трафіку
- Аналіз Netflow (v5, v9, IPFIX, sflow. iflow. NetStrean)
- Продуктивність від 10 Gbps
- Від 100 000 FPS
- Візуалізація трафіку на рівнях L2-L7
- Кореляція журналів та даних з трафіку
- Перевірка трафіку згідно репутаційних база та IoC
- Виявлення атак нульового дня
- Аналіз активності користувачів у мережі
- Моніторинг SRT, RTT, Delay, Jitter, аномалій мережі

Контакти

ENERGY LOGSERVER©

Energy Logserver – це інструмент, який не змушує вас йти на компроміси та зменшувати вимоги. Навпаки, він розширює ваші можливості та додає контролю там, де, начебто, контроль неможливий.

Щоб замовити демоверсію Energy Logserver, напишіть на електронну пошту

els@bakotech.com.

www.energylogserver.com