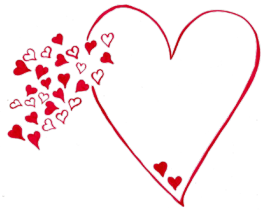
WAZUH

Alessandro De Salvo

Alessandro.DeSalvo@roma1.infn.it

Mini Workshop INFN Security – 14/02/2023



WAZUH

- Wazuh è un software della categoria SIEM (Security Info and Event Management) ed unisce sia le caratteristiche tipiche dei SIEM a quelle degli XDR (Extended Detection and Response)
  - OpenSource
  - Supporto esteso a pagamento disponibile (premium support)
  - https://wazuh.com

- Originariamente creato come fork di OSSEC
  - O meglio di OSSEC+, che integra più funzionalità (supporto fornito da Atomicorp, che ne fornisce una versione modificata a pagamento)
  - https://www.ossec.net

- Wazuh rispetto ad OSSEC è stato esteso sia in termini di funzionalità, di interoperabilità e documentazione

FUNZIONALITÀ PRINCIPALI

- Wazuh non si sostituisce a sistemi come firewall, vulnerability scanners o similari, ma piuttosto ne integra le funzionalità con sistemi di riconoscimento dei problemi di sicurezza e protezione attiva avanzata dei problemi rilevati
- Le sue funzionalità principali sono
  - HIDS (Host-Based Intrusion Detection System)
  - IPS (Intrusion Prevention System) – incident response
  - Security Analytics
  - Log Data Analysis
  - FIM (File Integrity Monitoring)
  - Vulnerability Detection (installed software)
  - SCA (Security Configuration Assessment)
  - Regulatory Compliance
- Wazuh può sia fornire i risultati delle analisi, come anche notificarle tramite svariati canali di comunicazione e intraprendere azioni automatiche (active response)

ANALISI DEGLI EVENTI [1]

- Wazuh fornisce le analisi di numerosi aspetti di sicurezza, fra i quali
  - Security Information Management
  - Monitoring di auditing e policy
  - Rilevamento di minacce e vulnerabilità
  - Compliance alle regolamentazioni

- I risultati possono essere consultati in tempo reale, sotto forma di report oppure tramite notifiche
- Nel caso di vulnerabilità o rilevamento di minacce il sistema è anche in grado di reagire in modo autonomo bloccando la minaccia ad esempio con regole di firewall

ANALISI DEGLI EVENTI [2]

SECURITY INFORMATION MANAGEMENT

Security events

Browse through your security alerts, identifying issues and threats in your environment.

Integrity monitoring

Alerts related to file changes, including permissions, content, ownership and attributes.

AUDITING AND POLICY MONITORING

Policy monitoring

Verify that your systems are configured according to your security policies baseline.

System auditing

Audit users behavior, monitoring command execution and alerting on access to critical files.

Security configuration assessment

Scan your assets as part of a configuration assessment audit.

THREAT DETECTION AND RESPONSE

Vulnerabilities

Discover what applications in your environment are affected by well-known vulnerabilities.

MITRE ATT&CK

Security events from the knowledge base of adversary tactics and techniques based on real-world observations

REGULATORY COMPLIANCE

PCI DSS

Global security standard for entities that process, store or transmit payment cardholder data.

NIST 800-53

National Institute of Standards and Technology Special Publication 800-53 (NIST 800-53) sets guidelines for federal information systems.

TSC

Trust Services Criteria for Security, Availability, Processing Integrity, Confidentiality, and Privacy

GDPR

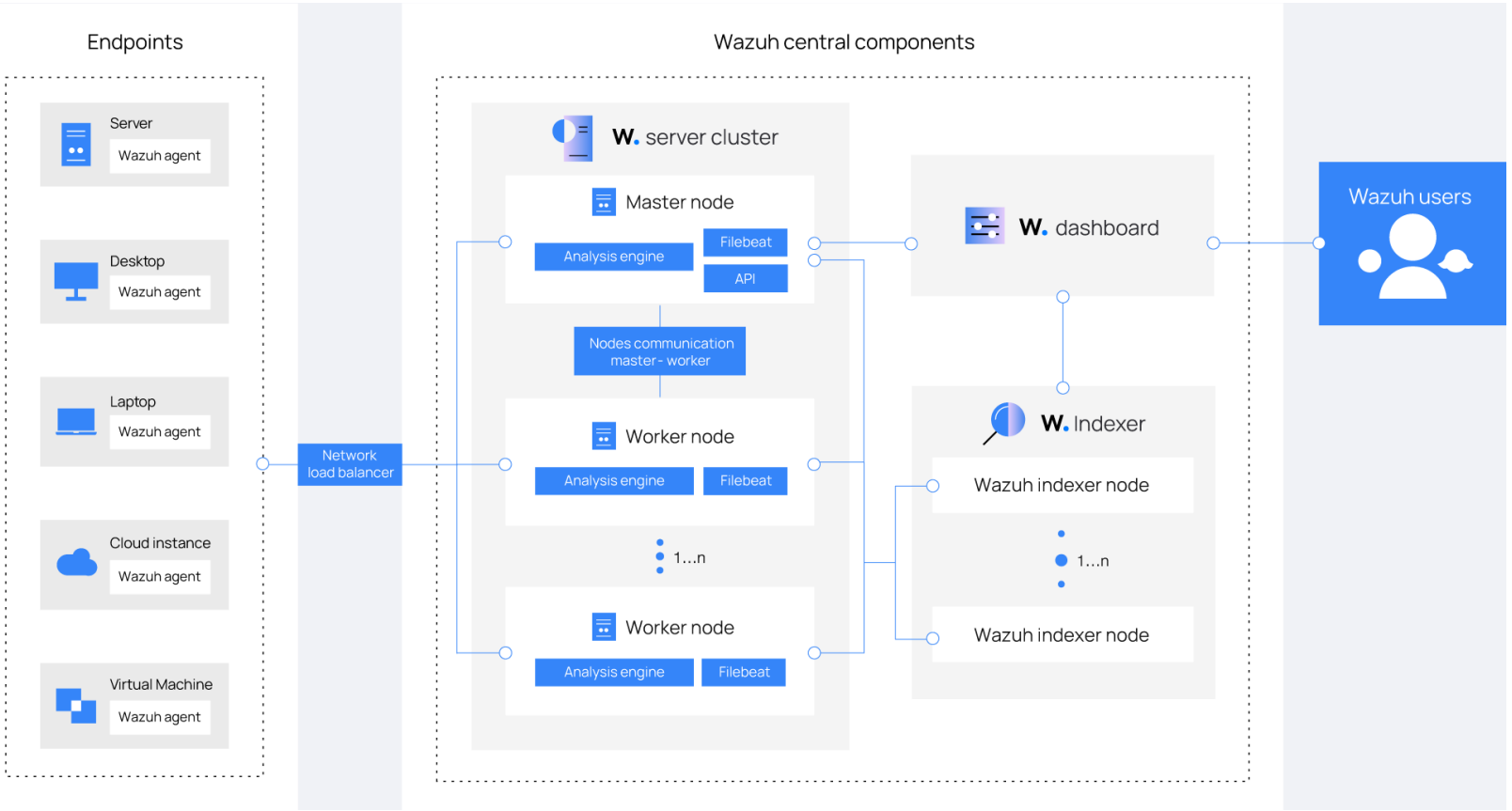
General Data Protection Regulation (GDPR) sets guidelines for processing of personal data.

HIPAA

Health Insurance Portability and Accountability Act of 1996 (HIPAA) provides data privacy and security provisions for safeguarding medical information.

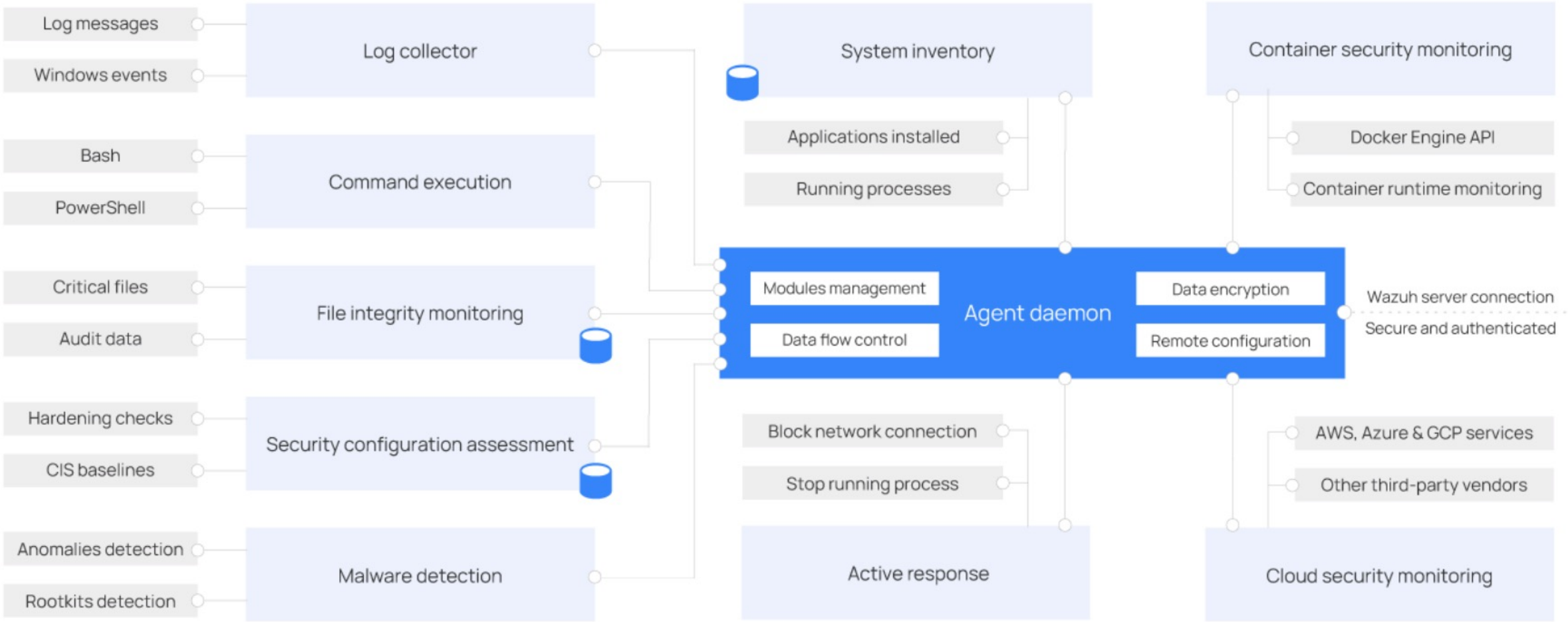
ARCHITETTURA

- I componenti principali dell'architettura di Wazuh sono
  - Server
  - Indexer
  - Dashboard
  - Agents



- Agent e server sono servizi specifici di Wazuh, mentre indexer e Dashboard sono servizi che utilizzano industry standard (ElasticSearch, Kibana, Beats, …)

ARCHITETTURA DEGLI AGENT



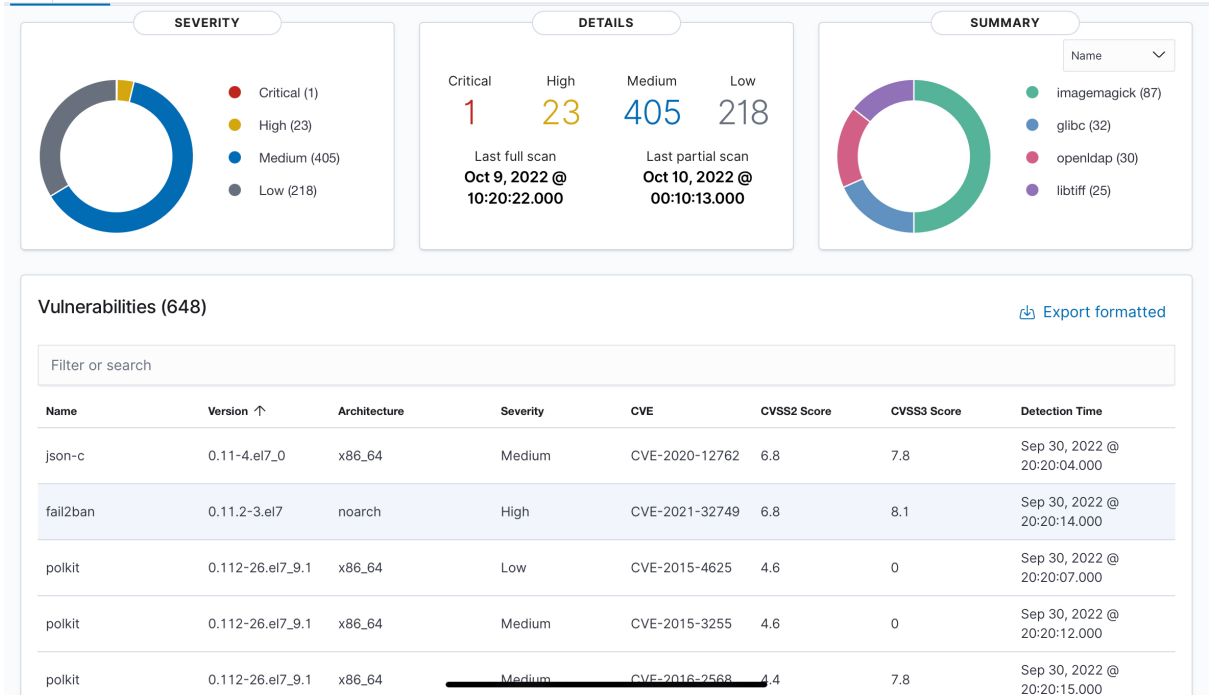
CONFIGURAZIONE CENTRALIZZATA DEGLI AGENT E GRUPPI

- Gli agent di Wazuh sono generalmente configurati con i file agent.conf, ma il server è in grado di inviare a ciascun agent una configurazione centralizzata periodicamente
  - Invio programmato per default ogni 10s, se la configurazione è cambiata
  - Repository centrale delle configurazioni centralizzate sul manager

- Gli agenti possono essere raggruppati in categorie omogenee per quel che riguarda la configurazione
  - I file di configurazione centralizzati possono essere suddivisi in gruppi
  - In fase di registrazione iniziale gli agent vengono assegnati automaticamente al gruppo default
  - Successivamente, tramite CLI o API gli agent possono essere assegnati a uno o più gruppi diversi
  - Ad esempio un host può essere in un gruppo DB e in un gruppo di un sito particolare

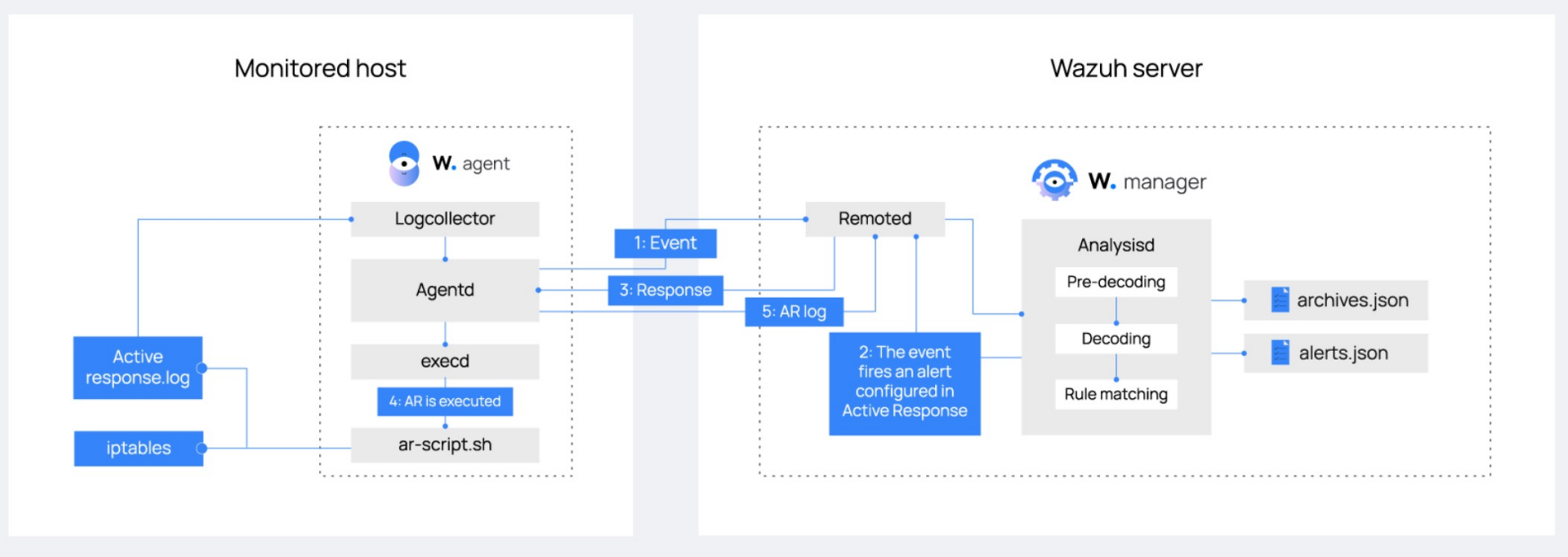
ANALISI DELLE VULNERABILITÀ

- Gli agent di Wazuh non si limitano solo ad analizzare i comportamenti anomali e le minacce, ma sono in grado anche di analizzare i software presenti negli host per identificare se siano già notoriamente vulnerabili, anche se il fix non è ancora disponibile
- Il server può indicare agli agent di effettuare periodicamente lo scan del software presente negli host e riportare negli alert la situazione globale e per ogni singolo agent in funzione
- Opzionalmente si possono ricevere notifiche e report sulla situazione dei software negli host



ACTIVE RESPONSE

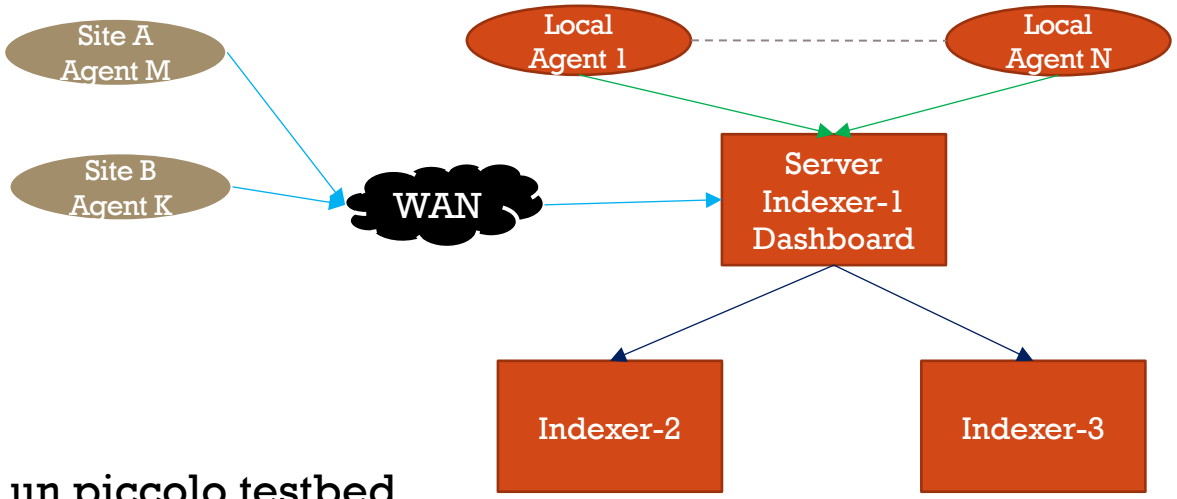
- Wazuh è in grado di reagire a condizioni rilevate con azioni correttive o di prevenzione
  - https://documentation.wazuh.com/current/user-manual/capabilities/active-response/how-it-works.html
- Le azioni sono attuate tramite uno script (e.g. Python) o un eseguibile configurato per essere attivato a seguito di una alert specifica, un livello di alert o un gruppo di regole
- Le active response sono definite a livello del server e iniziate da esso, e possono essere di due tipi
  - Stateful: script configurati per eseguire e poi invertire l'effetto un'azione, dopo un determinato lasso di tempo (ad esempio l'introduzione di una regola di firewall per bloccare temporaneamente una minaccia)
  - Stateless: script che eseguono un'azione singola senza invertirne l'effetto successivamente



INTEGRAZIONE CON SURICATA

- Suricata è un software NIDS (Network Intrusion Detection System) Open Source che può essere installato su hardware dedicato oppure direttamente sugli host da monitorare
  - https://suricata.io
- Suricata genera alert basati su regole del traffico identificato e le scrive sotto forma di log in formato JSON
- Tali log possono essere letti da Wazuh per poi essere analizzati con regole dedicate sul server, integrando di fatto le funzionalità native di Wazuh con le regole di analisi del traffico di Suricata
- Per abilitare l'utilizzo di Suricata in Wazuh è necessario
  - Installare Suricata sugli host da monitorare
  - Abilitare la lettura dei log locali di Suricata negli agent, tramite configurazione locale o centralizzata
- Una volta abilitato Suricata il server analizzerà automaticamente gli eventi da lui prodotti e li utilizzerà nelle Dashboard delle minacce, inoltre potrà bloccare le intrusioni tramite active response
- Suricata è disponibile per la maggior parte delle piattaforme software e in particolare per Linux/Mac/FreeBSD/UNIX/Windows

TESTBED (POC E R&D)



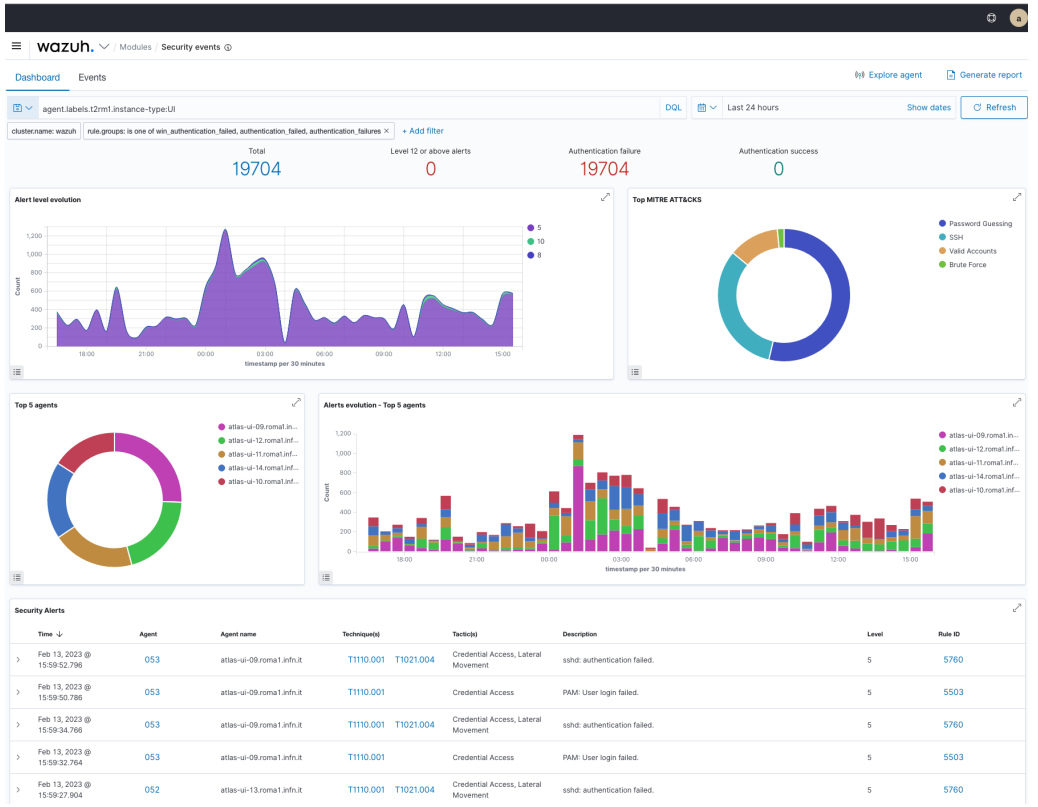
- A scopo di Proof Of Concept e R&D è stato creato un piccolo testbed
- Il testbed era originariamente formato da un singolo host con funzionalità di server+indexer+dashboard, allo scopo di testarne le funzionalità di base e la facilità di deployment
- Il testbed è stato ulteriormente espanso successivamente per aggiungere le seguenti funzionalità e facilities
  - Separazione in microservizi, multi-host
    - Per ragioni di disponibilità di risorse i servizi sono stati separati in multi-host e in particolare 1 server/API + indexer + dashboard e 2 indexer (**totale 3 host**)
    - E' molto facile aumentare la segmentazione (configurazione tramite ansible o puppet) e in particolare aggiungere host per server (ingester + API) e indexer, nonché spostare la dashboard su host dedicato, garantendo sia HA che incremento di capacità: il minimo descritto nella documentazione è 5 host, ma come ho descritto qui si riesce anche a contrarre tutto in 3 host con micro-servizi
  - Collezionamento di dati multi-sito con sistema centralizzato, anche multi-livello (encrypted data transfer over WAN)
  - Collezionamento di dati anche di syslog di host non direttamente supportati dagli agent (es. switch/firewall)
  - Scale-up del numero di agenti, in configurazione multi sito (attualmente ~100 agent, la maggior parte dei quali proveninenti dal Tier2 di ATLAS a Roma)

TESTBED (POC E R&D): OSSERVAZIONI [1]

- Stabilità
  - Il sistema è estremamente stabile, anche se in caso di aggiornamenti dei moduli di kibana è un po' complesso allineare le versioni nel browser, in quanto di norma vengono tenute in cache e devono essere esplicitamente eliminate per lavorare agevolmente (non un problema di wazuh, ma generale di Kibana)
- Resilienza
  - Il sistema si avvale di tutti i servizi di ElasticSearch (OpenSearch), inclusi quelli non specificamente disegnati per Wazuh, pertanto è estremamente versatile e resiliente, scalabile sia in orizzontale che verticale
- Configurabilità
  - Il numero di parametri configurabili è molto elevato, ma la possibilità di creare disastri è bassa
  - Praticamente ogni aspetto di wazuh è configurabile anche in modalità custom (active response, regole di alert, parsing di log, etc)
  - La documentazione è molto dettagliata e ben fatta
- Active-Response
  - Reazioni del sistema sempre puntuali e prevedibili (sia in banning che un-banning), da testare in modo più approfondito con script custom per sistemi non direttamente pilotabili dagli agent nativi di Wazuh

TESTBED (POC E R&D): OSSERVAZIONI [2]

- Utilizzo delle risorse
  - Come tutte le applicazioni in stile bigdata le risorse sono sempre il collo di bottiglia
  - In particolare lo spazio disco nel server
    - log di alert: con ~100 agent in media servono almeno 200GB al giorno, che però poi possono essere gettati una volta indicizzati
    - Data file di elastisearch: con ~100 agent in media servono circa 150 GB per indexer (3 indexer, 2 copie) per tenere un mese di dati, si possono definire policy di retention
  - Anche la CPU è importante, seppur in misura inferiore
    - Sia per non avere backlog nella fase di ingest che per avere una risposta ragionevole nelle interrogazioni
    - 3 VM con soli 4 core e 8GB di RAM di indexer/server riescono a tenere il carico di ~100 agent
- Segmentabilità e multi-tenancy
  - Gli agent possono avere label associate, che poi possono essere utilizzate per selezionare gruppi (siti, tipi di facilities, applicazioni specifiche, etc…) di host
  - Questa caratteristica è dinamicamente modificabile a livello centrale sul server verso determinati agent, tramite la configurazione centralizzata
  - Tali label sono molto utili in quanto possono essere usate per selezionare siti diversi o analizzare gli eventi di sicurezza o gli asset specifici da parte degli operatori centrali o locali
  - Gli utenti della dashboard possono avere livelli di privilegi diversi, diversificabili sia per privilegi centrali che locali di sito



DOCUMENTAZIONE E REFERENZE

- Uno dei punti di forza di Wazuh è l'estensiva documentazione, che copre anche molti aspetti del suo predecessore OSSEC
  - https://documentation.wazuh.com/current

- Supporto
  - https://wazuh.com/professional-services/#support

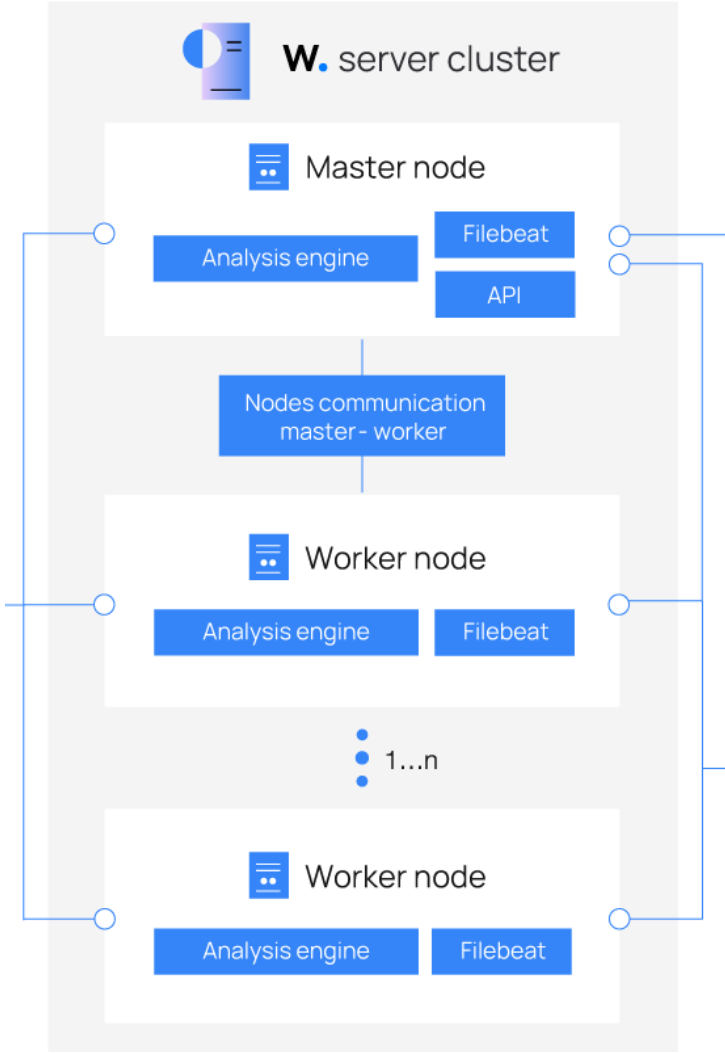
- Blog
  - https://wazuh.com/blog/

- Community
  - https://wazuh.com/community/

BACKUP SLIDES

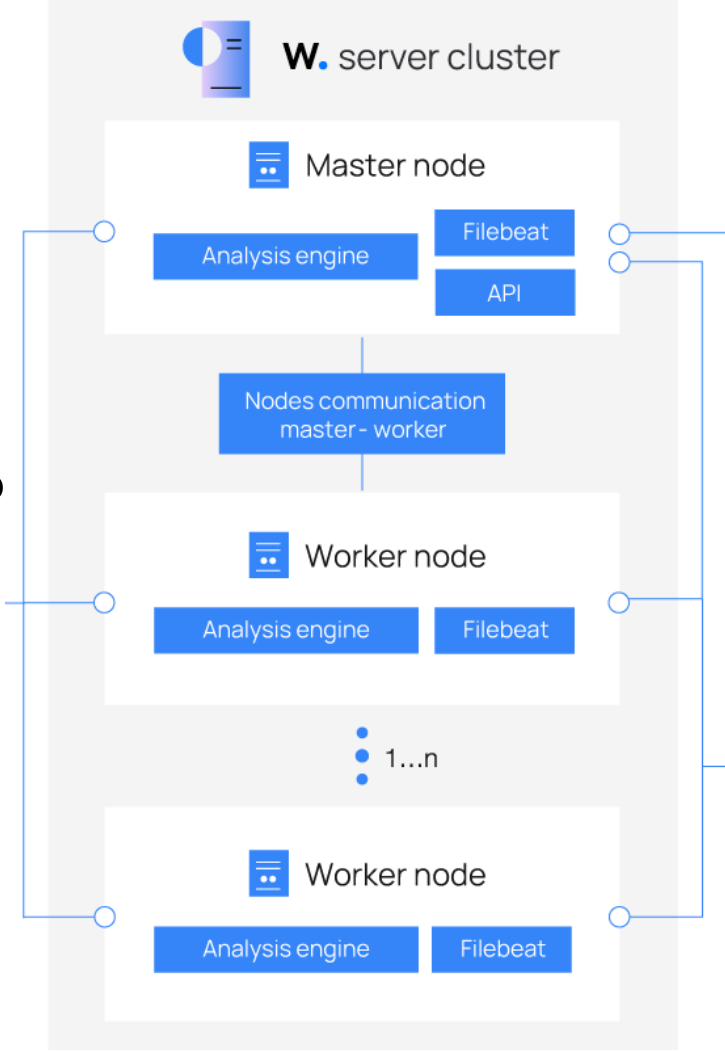
SERVER [1]

- Il server di Wazuh è composto almeno da un master node, che espone le porte di comunicazione per gli agenti e le API, estendibile con più worker in cluster per load balancing e capacity increase
- Gli agent si connettono al master per la registrazione e iniziale e successivamente al master stesso o ad uno dei worker per inviare i dati che vengono raccolti
- Il master e i worker ricevono i dati degli agent
  - li arricchiscono con metadati notevoli (derivati dall'analisi dell'evento e dalle regole predefinite in wazuh, nonché da dati presi da eventuali sistemi esterni come poor reputation ip list)
  - li inviano via filebeat agli indexer, eventualmente aggiungendo altri metadati come GeoIP
  - Notificano eventualmente eventi notevoli, over previsto dalla severity degli eventi stessi



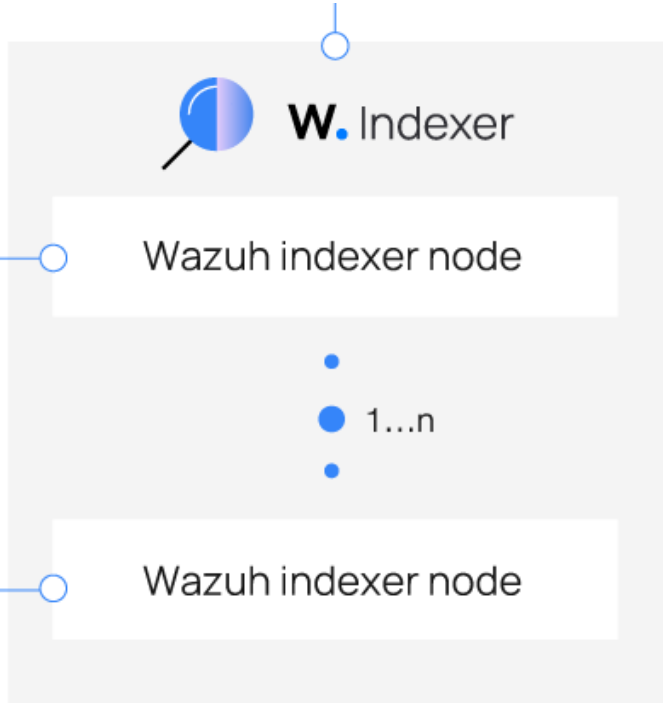
SERVER [2]

- Il master comunica con i worker per default tramite la porta 1516/tcp, che deve essere quindi accessibile solo tra i nodi del cluster
- Gli agent comunicano per default con il master sulla porta 1515/tcp per la registrazione iniziale e con master o worker sulla porta 1514/tcp
  - Tali porte devono essere accessibili a tutti gli agent
  - Nel caso di utilizzo di worker è necessario usare un load balancer (nginx/haproxy/…)
- Le API sono fornite dal master, per default su porta 55000/tcp
  - La Dashboard e qualsiasi altro servizio che voglia utilizzare le API devono avere accesso a questa porta



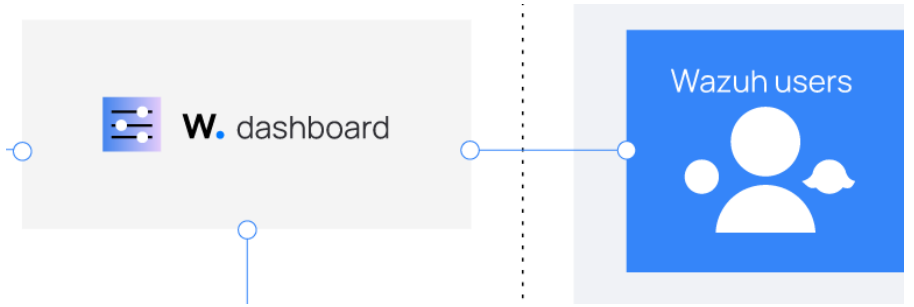
INDEXER

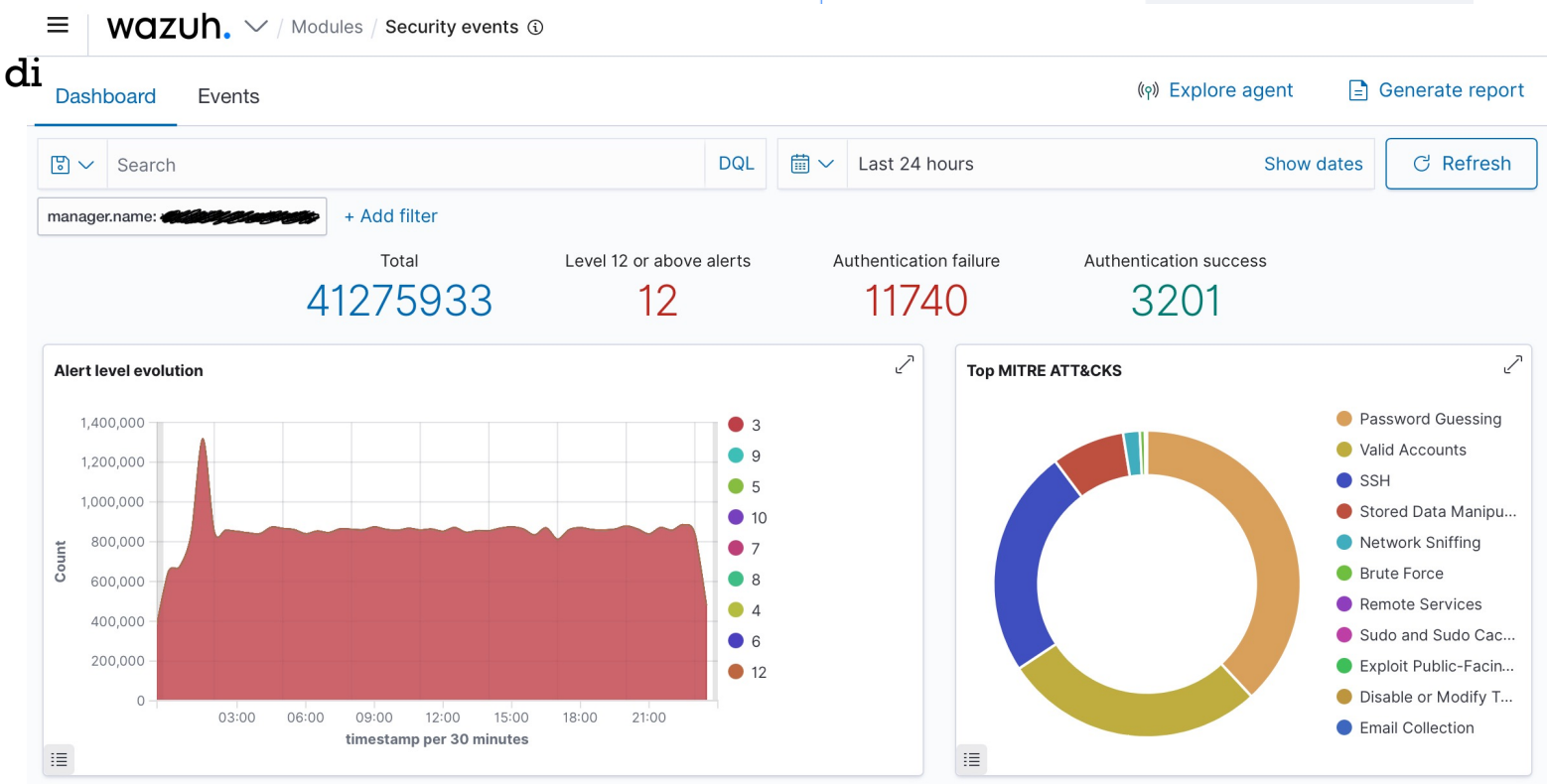
- Il nodi indexer di Wazuh sono di fatto dei nodi indexer di ElasticSearch
- La versione minimale dell'infrastruttura di Wazuh (ed ElasticSearch) prevede almeno un indexer, tuttavia per un sistema di produzione è consigliabile avere almeno 3 nodi
- L'indexer è il motore di ricerca full-text e analisi dati di Wazuh
- Gli indici registrano gli alert generati dal server di Wazuh, rendono possibile la ricerca in tempo reale e le analisi di correlazione dei dati
- Gli indici creati possono anche essere analizzati da strumenti esterni a Wazuh, essendo di fatto basati su formati standard (attualmente basati su ES 8.X)



DASHBOARD

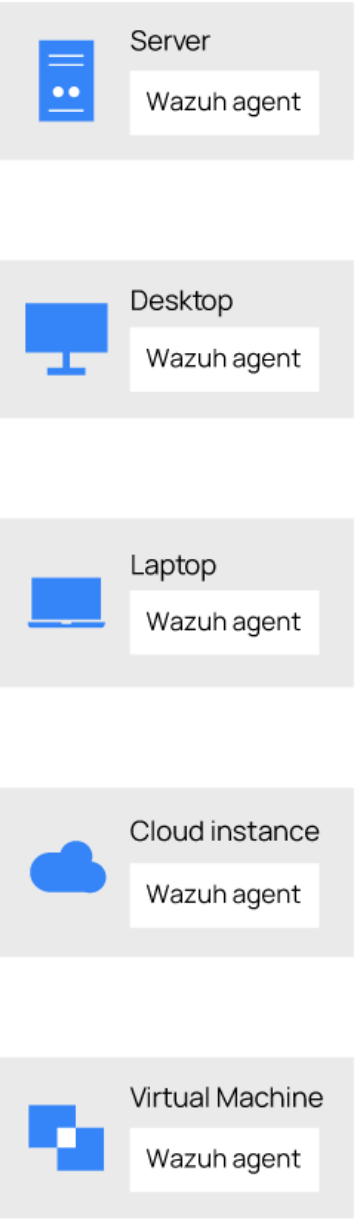
- La Dashboard di Wazuh è basata su Kibana
- Il modulo di Wazuh fornisce un set predefinito di Dashboard per le operazioni più comuni
  - Security Events
  - Vulnerabilità
  - File Integrity Monitoring data
  - Security Configuraztiin Assessment
  - Cloud Infrastructure Monitoring
  - Regulatory Compliance (PCI DSS, GDPR, TSC, HIPAA, NIST 800-53)
- È inoltre possibile aggiungere Dashboard personalizzate così come aggregazioni particolari di dati, con associata allarmistica





AGENT

- Gli agent di Wazuh sono il core del sistema di raccolta dati negli host
  - Generalmente sono eseguibili, sonde attivate in modo permanente negli host, che si occupano di inviare i dati importanti al server per le analisi di sicurezza (log, informazioni sui sw installati, etc)
  - Nel caso in cui non sia possibile installare software negli host, come ad esempio nel caso di switch, router o firewall, è possibile utilizzare il sistema chiamato **Agentless Monitoring**, tramite il quale il server può utilizzare connessioni ssh per entrare negli host ed estrarre le informazioni necessarie all'analisi di sicurezza
  - Tutte le trasmissioni dati tra gli agent e il server sono autenticate e criptate
- Gli agent e il server non devono necessariamente essere sulla stessa LAN
  - Questa caratteristica fa si che si possa anche avere un server centrale per più siti (ad esempio un SOC centralizzato)
  - Unica richiesta minima per avere un server multi-sito è di avere le porte pubbliche dei master/worker aperte alle subnet dei siti, ovvero 1514/tcp e 1515/tcp per default
  - Non è possibile collegare un singolo agent a più server, ma è possibile replicare off-site gli indexer tramite i meccanismi standard di replica di ElasticSearch (es. https://www.elastic.co/guide/en/elasticsearch/reference/current/xpack-ccr.html)



PIATTAFORME DEGLI AGENT

- Gli agent di Wazuh sono disponibili per le maggiori piattaforme software
  - https://documentation.wazuh.com/current/installation-guide/packages-list.html
  - Linux (Amazon, CentOS, Debian, Fedora, Oracle, SUSE, Ubuntu, Raspian, …)
  - Windows (>= XP)
  - MacOS
  - AIX
  - Solaris
  - HP-UX

- Altre piattaforme sono supportate tramite l'Agentless Monitoring tramite accesso SSH
- È inoltre possibile inviare syslog remoti a collector collegati a Wazuh: in questo modo si possono integrare la maggior parte degli apparati come router e firewall, quali i Fortigate o similari

DEPLOYMENT

- Tutti i componenti di Wazuh possono essere installati con diversi sistemi
  - https://documentation.wazuh.com/current/deployment-options/index.html
  - Cloud/container
    - OVA (Virtual Machine Appliance)
    - AMI (Amazon Machine Images)
    - Docker
    - Kubernetes
  - Offline
    - Wizard (ad-hoc script)
    - Ansible
    - Puppet
    - Source code
    - Installazione manuale

- Sono inoltre anche possibili installazioni di tipo commerciale, come ad esempio
  - Installazione con Elastic Stack basic license
  - Installazione con Splunk Enterprise

OPENSEARCH

- Wazuh è basato su OpenSearch
  - https://github.com/opensearch-project/OpenSearch
- OpenSearch è un fork community, open source, di ElasticSearch e Kibana, creato a seguito del cambio di tipo di licenza nel 2021
- OpenSearch offre le stesse caratteristiche di ElasticSearch più alcune caratteristiche normalmente vendute nei pack enterprise di ElasticSearch, come ad esempio il Machine Learning
- Gli indexer e la Dashboard di Wazuh sono compatibili con ES, ma nativamente integrati in OpenSearch, che ne estende le capacità
- I dati di Wazuh possono essere analizzati esternamente da qualsiasi infrastruttura basata su ES

ANOMALY DETECTION

- Una caratteristica interessante di OpenSearch è quella di mettere a disposizione come OpenSource l'infrastruttura di Machine Learning e Anomaly Detection già presenti nei pack enterprise di ElasticSearch
  - https://opensearch.org/docs/2.1/monitoring-plugins/ad/index/
- Un'anomalia in OpenSearch corrisponde ad qualsiasi comportamento anomalo delle serie temporali di dati
  - OpenSearch si avvale dell'algoritmo **RCF** (**Random Cut Forest**), ovvero un algoritmo non supervisionato che ha lo scopo di calcolare un grado di anomalia e un livello di confidenza per ogni punto di dati in tempo reale
  - I valori calcolati sono poi utilizzabili per evidenziare anomalie nelle serie di dati in ingresso
  - https://api.semanticscholar.org/CorpusID:927435
- Nel caso di Wazuh è possibile definire dei rilevatori di anomalia (detectors) in OpenSearch che usino gli indici di creati dagli indexer, anche applicando filtri specifici su regole particolari
  - Ad esempio è possibile creare un rilevatore di anomalia sulle login effettuate con successo nei nodi
- Le anomalie possono infine essere notificate dal sistema integrato di allarmistica
- Il plugin di anomaly detection di OpenSearch non è disponibile nella Dashboard per default, ma può essere facilmente installato, ad esempio con il comando seguente
  - sudo -u wazuh-dashboard /usr/share/wazuh-dashboard/bin/opensearch-dashboards-plugin install https://github.com/opensearch-project/anomaly-detection-dashboards-plugin/releases/download/1.2.0.0/anomaly-detection-dashboards-1.2.0.0.zip