Χρήση του SIEM
(ELK Wazuh use Case)

Αναφορά Διπλωματικής Εργασίας

Κωνσταντίνος Σαρηκιοσές
mte1928

Abstract

Το cyber threat hunting είναι η διαδικασία της προληπτικής και επαναληπτικής αναζήτησης μέσω δικτύων για τον εντοπισμό και την απομόνωση προωθούμενων απειλών που αποφεύγουν τις υπάρχουσες ρυθμίσεις ασφαλείας. Για την ανάλυση των logs χρησιμοποιείται το Elasticsearch, το οποίο είναι μια κατανεμημένη και ανοιχτού κώδικα μηχανή ανάλυσης για όλους τους τύπους δεδομένων. Επιπροσθέτως γίνεται χρήση του Wazuh plugin, μια δωρεάν, ανοιχτού κώδικα και έτοιμη για επιχειρήσεις λύση παρακολούθησης ασφάλειας για ανίχνευση απειλών, παρακολούθηση ακεραιότητας, απόκριση συμβάντων και συμμόρφωση. Υλοποιήθηκαν οι ενέργειες εγκατάσταση Wazuh server και agents, τα σενάρια ελέγχου ύποπτων αρχείων μέσω VirusTotal, επίβλεψη USB συσκευών αποθήκευσης και επίβλεψη Docker.

Εισαγωγή Threat Hunting, IoCs και MITRE ATT&CK

Το cyber threat hunting είναι η διαδικασία της προληπτικής και επαναληπτικής αναζήτησης μέσω δικτύων για τον εντοπισμό και την απομόνωση προωθούμενων απειλών που αποφεύγουν τις υπάρχουσες ρυθμίσεις ασφαλείας.[1] Σε αντίθεση με το penetration testing, όπου ο ερευνητής ασφαλείας προσπαθεί να βρει ευπάθειες από το εξωτερικό του δικτύου, ο κυνηγός απειλών υποθέτει ότι το δίκτυο έχει ήδη παραβιαστεί, σχηματίζει μια υπόθεση και ξεκινά μια έρευνα.
Τα κύρια καθήκοντα ενός threat hunter περιστρέφονται γύρω από τα εξης:

- Προσδιορισμός τύπων επιθέσεων
- Καθορισμός, καθοδήγηση και ιεράρχηση επιχειρησιακών απαιτήσεων
- Κατανόηση της ικανότητας, τακτικής, τεχνικών και των διαδικασιών του threat actor
- Ανάπτυξη συστημάτων ανίχνευσης
- Ανάπτυξη αμυντικών στρατηγικών

Οι κύριες απειλές στον κυβερνοχώρο, σύμφωνα με τον ENISA, εμπίπτουν σε μία από τις ακόλουθες κατηγορίες: κακόβουλο λογισμικό, επιθέσεις μέσω διαδικτύου, επιθέσεις εφαρμογών ιστού, άρνηση υπηρεσίας, botnets, ηλεκτρονικό ψάρεμα, ανεπιθύμητο περιεχόμενο, Ransomware, εσωτερική απειλή, φυσική χειραγώγηση / ζημιά / κλοπή / απώλεια, κιτ εκμετάλλευσης, παραβιάσεις δεδομένων, κλοπή ταυτότητας, διαρροή πληροφοριών και μη διορθωμένο λογισμικό.[2]

Το threat hunting είναι μια προληπτική προσέγγιση για την επίλυση του ζητήματος «network intruder» και ταυτόχρονα βάζει τον οργανισμό στη διαδικασία της τακτικής διενέργειας ελέγχων στα συστήματα ασφάλειας και παρακολούθησης. Με αυτόν τον τρόπο εντοπίζονται εσφαλμένες διαμορφώσεις και καλύπτονται τυφλά σημεία ασφαλείας. Ο threat hunter τίθεται σε μια θέση όπου πρέπει να σκέφτεται ως επιτιθέμενος ή εισβολέας και εντοπίζει νέους φορείς επίθεσης για να προσομοιώσει τη συμπεριφορά τους. Καθ 'όλη τη διάρκεια της διαδικασίας, οι ενέργειες που εκτελούνται πρέπει να υποστηρίζονται από την αρχική υπόθεση, καθώς ακολουθώντας το ίχνος κάθε πιθανής προειδοποίησης που εμφανίζεται στα συστήματα παρακολούθησης, όπως το SIEM, μπορεί να εισαγάγει ζητήματα πέρα από το πεδίο της έρευνας και να οδηγήσει σε μια άκαρπη προσπάθεια.

Υπάρχουν τρεις βασικοί δρόμοι για την προσέγγιση μιας υπόθεσης threat hunting[3]:

- IoC Driven
- Analytics and machine learning Driven
- Situationally Driven

[1] Cyber Threat to Critical Infrastructure
[2] 2016 Threat Landscape Report, ENISA
[3] Cyber Threat Intelligence Understanding Fundamentals

IoC Driven

Κάθε οργανισμός έχει ρυθμίσει μια περίμετρο ασφαλείας για την αποτροπή επιθέσεων που στοχεύουν στο δίκτυο και τα περιουσιακά του στοιχεία. Για την προστασία ευαίσθητων δεδομένων από διαρροές ή την υπεράσπιση του δικτύου εφαρμόζονται συστήματα όπως firewalls, IDS ή IPS. Ο στόχος τους μπορεί να επιτευχθεί, αλλά ενεργώντας ως μεμονωμένοι κόμβοι δημιουργεί ορισμένα παράπλευρα ζητήματα.

Μια πολύ κοινή παράπλευρη ζημία είναι το ζήτημα των false positives. Τα false positives απαιτούν την προσοχή του security analyst ή ακόμα και του threat hunter, σε περίπτωση έρευνας υπόθεσης, για την καταστολή των προειδοποιήσεων που δημιουργούνται από τα συστήματα ασφαλείας. Οι άπειροι επαγγελματίες στον τομέα της ασφάλειας ενδέχεται επίσης να θέσουν σε σίγαση ή να αγνοήσουν όλες τις μελλοντικές προειδοποιήσεις του ίδιου είδους, κατατάσσοντας τις όλες ως false positives χωρίς περαιτέρω έρευνα. Αυτός ο τύπος ενέργειας θεωρείται κακή πρακτική καθώς το πλαίσιο της προειδοποίησης αγνοείται πλήρως. Χωρίς διερεύνηση των παραμέτρων, η πηγή και η πραγματική αιτία της προειδοποίησης κατα τη διάρκεια μιας πιθανής cyber επίθεσης αγνοείται πλήρως και το αρμόδιο τμήμα της εταιρείας δεν εφαρμόζει το προσχεδιασμένο πλάνο incident response.[4] Επιπλέον, δεν εξαρτάται μόνο από τις δεξιότητες και την κατάρτιση του προσωπικού πληροφορικής. Η ποσότητα των προειδοποιήσεων που παράγονται από μεμονωμένα συστήματα μπορεί συχνά να φτάσει σε μεγάλο αριθμό, καθιστώντας αδύνατη την ατομική έρευνα προειδοποίησης ή εξαιρετικά απαιτητικό σε πόρους και χρόνο.

Ένας τρόπος για να μειωθεί ο όγκος των προειδοποιήσεων, που δημιουργήθηκαν από τα συστήματα IDS και IPS και απαιτούν άμεση προσοχή ή ενέργειες από ένα φυσικό πρόσωπο είναι να γίνει χρήση περαιτέρω ανάλυσης αντί για την ίδια την προειδοποίηση. Για να ταξινομηθούν οι προειδοποιήσεις με βάση το επίπεδο σοβαρότητάς τους, κάθε προειδοποίηση συσχετίζεται με άλλους δείκτες Indicators of Compromise, ή IoC, έτσι ώστε να αποκαλυφθεί το κίνητρο του δυνητικού εισβολέα.

Indicators of Compromise (IOCs) are forensic artifacts that are used as signs that a system has been compromised by an attack or that it has been infected with a particular malicious software. [5]Τα παραπάνω ενδέχεται να περιλαμβάνουν hashes κακόβουλου λογισμικού, AV Signatures και blacklisted διευθύνσεις IP, hostnames, text strings, ονόματα διεργασιών, ονόματα αρχείων, διευθύνσεις email κ.λπ.

Ο threat hunter πρέπει στη συνέχεια να συσχετίσει τη σοβαρότητα κάθε συμβάντος με την προειδοποίηση και τα IoC που εντοπίστηκαν στο σύστημα και να κατευθύνει το προσωπικό πληροφορικής προς το πιο επείγον. Έχοντας κατηγοριοποιήσει την έξοδο πληροφοριών ασφαλείας, μειώνεται το ποσοστό των αγνοημένων ή ακόμη και η πλήρης σίγαση των

[4] Cyber Threat Intelligence from Honeypot Data Using Elasticsearch
[5] Automatic Extraction of Indicators ofCompromise for Web Applications

προειδοποιήσεων, παρέχει τη δυνατότητα δημιουργίας αυτοματοποιημένων ενεργειών για επαναλαμβανόμενα συμβάντα και, χρησιμοποιώντας τα δεδομένα που υπολογίζονται από τη συσχέτιση, μειώνεται δραστικά ο αριθμός των ψευδών θετικών.

Analytics and machine learning Driven

Μια άλλη προσέγγιση για το κυνήγι απειλών βασίζεται σε αναλυτικά στοιχεία που δημιουργούνται στο δίκτυο, που υπολογίζονται και παρουσιάζονται στον πίνακα ελέγχου του System Information and Event Management (SIEM). Η μηχανική μάθηση μπορεί να βοηθήσει πολύ όταν ο threat hunter σχηματίζει μια νέα υπόθεση και μπορεί να του δώσει τα εργαλεία για να συσχετίσει διαφορετικές ανωμαλίες και να εντοπίσει μια υποκείμενη επίθεση. Αυτές οι ανωμαλίες από μόνες τους δεν προκαλούν συνήθως προειδοποίηση ή, δεδομένης της έλλειψης περισσότερων αποδεικτικών στοιχείων που οδηγούν στην πραγματική πηγή, δεν θα ληφθούν ιδιαίτερα υπόψη.

Οι αλγόριθμοι δημιουργούν ενα baseline αναλύοντας ιστορικά και τρέχοντα δεδομένα που δημιουργούνται από κάθε κόμβο στο δίκτυο. Η εκμάθηση είναι συνεχής που σημαίνει ότι η περίοδος εκπαίδευσης δεν είναι μόνο ένα καθορισμένο χρονικό πλαίσιο μετά από κάθε εγκατάσταση των εξαρτημάτων του SIEM. Δημιουργούνται διαφορετικά μοτίβα βάσει πολλαπλών τάξεων μεγέθους του χρόνου. Αυτή η μέθοδος καθορίζει baselines που μπορούν να συσχετιστούν με ωριαία, ημερήσια, μηνιαία, ετήσια δεδομένα και ούτω καθεξής.

Η Elastic[6] χρησιμοποιεί εφαρμογές μηχανικής εκμάθησης για την αυτοματοποίηση της ανάλυσης δεδομένων χρονοσειρών με την καθιέρωση αξιόπιστων βασικών γραμμών φυσιολογικής δραστηριότητας δεδομένων και την ανίχνευση ανώμαλων τάσεων στα δεδομένα. Τα δεδομένα μπορούν να επεξεργαστούν σε παρτίδες ή σε συνεχείς ροές δεδομένων σε πραγματικό χρόνο.

Περιπτώσεις σχετικές με ανωμαλίες σε τιμές, μετρήσεις ή συχνότητες που σχετίζονται με χρονικές αποκλίσεις, στατιστική ευαισθησία και ασυνήθιστη συμπεριφορά, για έναν μόνο χρήστη ή ομάδες χρηστών αναγνωρίζονται, βαθμολογούνται και συνδέονται στα δεδομένα χρησιμοποιώντας αλγόριθμους μηχανικής εκμάθησης με σχετικά στατιστικά στοιχεία για την αντιστάθμιση των αποκλίσεων.

Χρησιμοποιώντας αυτές τις πρόσφατα διαθέσιμες πληροφορίες, ο threat hunter έχει πολλά περισσότερα στοιχεία που πρέπει να ληφθούν υπόψη ενώ αναλαμβάνει μια νέα υπόθεση. Ακόμη και προειδοποιήσεις που θεωρούνται άσχετες με την πρώτη ματιά, ή εκτός του πλαισίου χωρίς να ληφθούν υπόψη οι υπόλοιπες ανωμαλίες που εμφανίζονται, μπορούν να συμβάλουν στο υποκείμενο ζήτημα. Αυτά τα αναλυτικά στοιχεία είναι ένα ισχυρό εργαλείο για τον threat hunter καθώς παρουσιάζουν πληροφορίες που δεν ήταν διαθέσιμες στο παρελθόν λόγω του τεράστιου αριθμού των logs που αναλύθηκαν και της αφαίρεσης που παρέχουν οι αλγόριθμοι, αλλά το ανθρώπινο στοιχείο δεν μπορεί να αφαιρεθεί ακόμη από την εικόνα. Η κριτική σκέψη

[6] Elastic.co

του threat hunter είναι το αναπόσπαστο μέρος που διαμορφώνει την τελική υπόθεση και αποφασίζει ποιες από τις προειδοποιήσεις είναι σχετικές με το θέμα.

Situationally Driven

Όταν ξεκινά μια situationally driven υπόθεση, ή είναι γνωστή ως "crown jewel analysis", ο threat hunter λαμβάνει υπόψη τα πιο πολύτιμα περιουσιακά στοιχεία του οργανισμού και τα θέτει στο επίκεντρο της προσοχής. Είναι πολύ σημαντικό να γίνονται γνωστές οι αλλαγές στο περιβάλλον και να αναγνωρίζονται συνεχώς νέοι φορείς επίθεσης.

Για τον προσδιορισμό των πιο πολύτιμων περιουσιακών στοιχείων σε ένα επιχειρηματικό περιβάλλον, πρέπει να ληφθούν υπόψη οι ακόλουθοι παράγοντες: Προσδιορισμός της αποστολής της επιχείρησης, εντοπισμός των περιουσιακών στοιχείων και των πληροφοριών στις οποίες βασίζεται η επιχείρηση και κατοχή ενός ενημερωμένου χάρτη δικτύου και πόρων. Μόλις αυτές οι πληροφορίες είναι διαθέσιμες στον threat hunter, τότε ο ίδιος έχει τη δυνατότητα να στραφεί στη νοοτροπία του εισβολέα και να αρχίσει να κατασκευάζει γραφήματα επίθεσης.

Ως αποτέλεσμα, τα πιο πολύτιμα και ευάλωτα περιουσιακά στοιχεία είναι ο στόχος του εισβολέα. Αυτά τα περιουσιακά στοιχεία γίνονται προτεραιότητα όταν τεθούν σε εφαρμογή αμυντικά συστήματα δικτύου με σκοπό την ελαχιστοποίηση του κινδύνου και του αντίκτυπου σε μια πιθανή επίθεση. Είναι επίσης σημαντικό για τους threat hunters να είναι ανοιχτόμυαλοι και να αποφεύγουν υποθέσεις που δεν μπορούν να οδηγήσουν σε επιτυχές threat hunt.

Διαδικασία Threat Hunting

Το επίσημο threat hunting μοντέλο αποτελείται από έξι διαδοχικά στάδια: purpose, scope, equip, plan review, execute, και feedback. Το πρώτο στάδιο του threat hunting κύκλου, γνωστό ως στάδιο σκοπού, περιγράφει τους στόχους και τα αποτελέσματα του threat hunt. Το δεύτερο στάδιο είναι το πεδίο εφαρμογής και περιλαμβάνει την ανάπτυξη ενός λεπτομερούς σχεδίου για το πού να συλλέξετε δεδομένα, καθώς και την ανάπτυξη αναλυτικών ερωτημάτων, επίσης γνωστά ως υποθέσεις. Η πρώτη φάση του σταδίου προσδιορίζει την περιοχή που πραγματοποιείται ένα hunt και όλα τα σχετικά συστήματα και πρωτόκολλα. Η δεύτερη φάση του σταδίου εφαρμογής περιλαμβάνει την ανάπτυξη υποθέσεων που υποστηρίζουν το γενικό σκοπό.

Οι υποθέσεις αντιστοιχίζονται σε πηγές δεδομένων για να αποδείξουν ή να διαψεύσουν την ερώτηση. Είναι σημαντικό να σημειωθεί ότι η ανάπτυξη υποθέσεων πρέπει να συμβεί μετά τον καθορισμό των σταδίων του σκοπού και του πεδίου εφαρμογής. Η σειρά ανάπτυξης είναι ζωτικής σημασίας, διότι πρέπει να δημιουργηθούν υποθέσεις για να αποδειχθεί ή να διαψευστεί ένα ερώτημα. Χωρίς έναν αρχικά καθορισμένο σκοπό ή εύρος που οδηγεί τη δημιουργία ερωτήσεων, οι παραδοχές σχετικά με τις πηγές δεδομένων και τις υποθέσεις μπορεί να οδηγήσουν το hunt μακριά από το επιδιωκόμενο αποτέλεσμα και να εισαγάγουν περιττές γνωστικές προκαταλήψεις. Το στάδιο εξοπλισμού επικεντρώνεται στον προσδιορισμό των τεχνικών ανάλυσης και των εργαλείων που απαιτούνται για την επεξεργασία δεδομένων και την απόδειξη ή την απόρριψη αναπτυγμένων υποθέσεων. Μια ανασκόπηση του σχεδίου διασφαλίζει ότι οι αναπτυγμένες υποθέσεις και οι προσδιορισμένοι πόροι ανταποκρίνονται στον συνολικό σκοπό του κυνηγιού.

Μόλις εγκριθεί το σχέδιο, το στάδιο εκτέλεσης είναι όπου ένας threat hunter συλλέγει και αναλύει δεδομένα σύμφωνα με τις υποθέσεις που έχουν προσδιοριστεί. Τέλος, το στάδιο ανατροφοδότησης επιτρέπει την αναδρομή στην εκτέλεση κάθε προηγούμενου σταδίου του threat hunting μοντέλου και παρέχει την ευκαιρία να εντοπιστούν βελτιώσεις για μελλοντικά hunts. Ο κύκλος threat hunting είναι κυκλικός για να διασφαλίσει ότι τα αποτελέσματα και τα διδάγματα από προηγούμενα hunts θα επηρεάζουν τα μελλοντικά.[7]

[7] **A Practical Model for Conducting Cyber Threat Hunting**


PURPOSE
Why are you threat hunting?
SCOPE
Where are you hunting and what questions do you hope to answer?
EQUIP
What data sources and tools are you using?
PLAN REVIEW
Does the planned hunt meet the overall purpose?
EXECUTE
Conduct the hunt.
FEEDBACK
What went well? What could be done better?

MITRE ATT&CK Framework

Το MITER ATT&CK Framework, ή μοντέλο, είναι μια γνωσιακή βάση της εχθρικής συμπεριφοράς, που βασίζεται σε πραγματικές παρατηρήσεις. Το ATT&CK σημαίνει adversarial tactics, techniques, and common knowledge. Το μοντέλο μπορεί να χρησιμοποιηθεί για τον καλύτερο χαρακτηρισμό της εχθρικής συμπεριφοράς μετά την παραβίαση του δικτύου με στόχο την απόσταξη κοινών συμπεριφορών σε γνωστή δραστηριότητα εισβολής σε άτομα ή συνδυασμούς ενεργειών που μπορεί να λάβει ένας αντίπαλος για την επίτευξη των στόχων του.[8]

Η τακτική και οι τεχνικές είναι μια νέα προσέγγιση στις κυβερνοεπιθέσεις. Αντί να δουν τις συνέπειες μιας επίθεσης ή ενός Indicator of compromise (IoC), οι αναλυτές ασφαλείας θα πρέπει να εξετάσουν τις στρατηγικές και τις τεχνικές που υποδηλώνουν ότι μια επίθεση είναι σε εξέλιξη. Η τακτική είναι ο λόγος πίσω από τη στρατηγική μιας επίθεσης. Οι τεχνικές απεικονίζουν πώς ένας αντίπαλος επιτυγχάνει έναν τακτικό στόχο κάνοντας μια λειτουργία.

Κοινή γνώση είναι οι αντίπαλοι που καταγράφουν τη χρήση στρατηγικών και τεχνικών. Η καταγραφή των διαδικασιών είναι βασικά κοινή γνώση. Πιο γνωστό στον τομέα της ασφάλειας στον κυβερνοχώρο ως ο όρος "tactics, techniques , and procedures", ή TTP.

Ο στόχος του MITRE ATT&CK είναι να δημιουργήσει μια λεπτομερή λίστα γνωστών στρατηγικών και τεχνικών αντιπάλων που χρησιμοποιήθηκαν κατά τη διάρκεια μιας επίθεσης στον κυβερνοχώρο. Θα πρέπει να είναι σε θέση να συλλέξει ένα μεγάλο και ιδανικά εξαντλητικό φάσμα σταδίων επίθεσης και ακολουθιών που διατίθενται σε κυβερνητικούς, εκπαιδευτικούς και εμπορικούς οργανισμούς. Επομένως, ο στόχος της MITER ATT&CK είναι η καθιέρωση ενός τυπικού συστήματος ταξινόμησης που θα κάνει τις αλληλεπιδράσεις πιο ακριβείς μεταξύ των οργανισμών.

Το ATT&CK δεν είναι πανίσχυρο, παρ’ όλα τα πλεονεκτήματά του. Οι οργανισμοί που το αντιμετωπίζουν ως τέτοιο ενδέχεται να καταλήξουν σε μια λανθασμένη αίσθηση ασφάλειας και εσφαλμένη κατανομή πόρων. Κάθε framework έχει περιορισμούς, καθώς, όπως περιγράφεται παραπάνω, τα frameworks είναι απλουστευμένες πραγματικές συνθήκες.[9]

Οι κύριες περιπτώσεις χρήσης του ATT&CK περιλαμβάνουν:

- Adversary emulation

  Το ATT&CK μπορεί να χρησιμοποιηθεί για τη δημιουργία σεναρίων εξομοίωσης αντιπάλων ή για τον έλεγχο και την επικύρωση άμυνας εναντίον τυπικών τεχνικών αντιπάλων.

[8] Finding Cyber Threats with ATT&CK™-Based Analytics
[9] MITRE ATT&CK Framework Introduction

- Red Teaming

  Το ATT&CK μπορεί να χρησιμοποιηθεί για την οικοδόμηση red team στρατηγικών και για τον συντονισμό των λειτουργιών για την αποφυγή τυχόν προστατευτικών μέτρων που ενδέχεται να υπάρχουν εντός ενός δικτύου.
- Creation of behavioral analytics

  Το ATT&CK μπορεί να χρησιμοποιηθεί για την ανάπτυξη και την αξιολόγηση συμπεριφορικών αναλυτικών στοιχείων για τον εντοπισμό εχθρικής δραστηριότητας.
- Defensive Gap Evaluation

  Το ATT&CK μπορεί να χρησιμοποιηθεί ως μοντέλο εχθρικής συμπεριφοράς για την αξιολόγηση των πόρων, την παρακολούθηση και την αποφυγη των συστημάτων άμυνας εντός της επιχείρησης ενός οργανισμού.
- SOC Maturity Evaluation

  Το ATT&CK μπορεί να χρησιμοποιηθεί ως ένα μέτρο αξιολόγησης ως προς το πόσο αποτελεσματικό είναι ένα SOC στην ανίχνευση, την ανάλυση και αντίδραση στις εισβολές.
- Cyber Threat Intelligence Enrichment

  Το ATT&CK είναι χρήσιμο από τη σκοπιά των μοτίβων συμπεριφοράς για την αναγνώριση και καταγραφή προφίλ ομάδων αντιπάλων που είναι αγνωστικοί στους πόρους που μπορεί να χρησιμοποιήσει η ομάδα.

Εργαλεία Threat Hunting

- Logs
  Οι threat hunters απαιτούν δεδομένα. Στο ελάχιστο, είναι απαραίτητο να υπάρχουν αρχεία καταγραφής δεδομένων για κοσκίνισμα. Οι βασικές πηγές αυτών των δεδομένων περιλαμβάνουν logs τερματικών, logs συμβάντων των Windows, logs προστασίας από ιούς και logs proxy και firewall.

- Analytics
  Το machine learning και η ανάλυση δεδομένων είναι ένα πλεονέκτημα για οργανισμούς που μπορούν να τους αντέξουν οικονομικά λόγω της ικανότητάς τους να αυτοματοποιήσουν τον εντοπισμό απειλών στον κυβερνοχώρο και να αναγνωρίσουν μια απειλή συσχετίζοντας πληροφορίες από μεγάλη ποσότητα δεδομένων.

- SIEM
  Ένα κεντρικό σύστημα πληροφοριών ασφαλείας και διαχείρισης συμβάντων μπορεί να συσχετίσει όλα τα logs σας καλύτερα από τους ανθρώπους μόνο. Τα SIEM logs διευκολύνουν την ικανότητά του threat hunter να έχει πρόσβαση σε μεμονωμένα κομμάτια πληροφοριών αλλα και σε συνδέσμους και συσχετισμούς που αποκαλύπτουν την πραγματική απειλή.

Όσον αφορά τον προσδιορισμό της καλύτερης λύσης SIEM για έναν οργανισμό, υπάρχουν μερικοί παράγοντες που πρέπει να ληφθούν υπόψη, όπως το μέγεθος της εταιρείας, η πολυπλοκότητα της τεχνολογικής υποδομής, το κόστος και οι πόροι. Συχνά, η εμπορική λύση με την υψηλότερη πληρωμή δεν είναι πάντα η καλύτερη λύση, καθώς υπάρχουν πολλά SIEMS χαμηλού κόστους και ανοιχτού κώδικα που είναι εξίσου αποτελεσματικά εκτός από τον περιορισμό στην υποστήριξη πελατών.

Για οργανισμούς με περιορισμένο προϋπολογισμό, υπάρχει ένα πλήθος εξαιρετικών εργαλείων ανοιχτού κώδικα που διατίθενται για καταγραφή και ανάλυση logs, εγκληματολογία κεντρικών υπολογιστών και μνήμης, reverse engineering κακόβουλου λογισμικού και άλλα. Για παράδειγμα, μια οικονομικά αποδοτική εναλλακτική λύση SIEM είναι η υλοποίηση του Elastic Stack, το οποίο αποτελείται απο τα Elasticsearch, Logstash και Kibana.

Elastic Stack

Το Elastic Stack αναπτύχθηκε από την Elastic και είναι αναμφισβήτητα η πιο δημοφιλής πλατφόρμα διαχείρισης καταγραφής ανοιχτού κώδικα σήμερα. Αποτελείται από τα Elasticsearch, Logstash, Beats και Kibana.

Elasticsearch

Το Elasticsearch είναι μια κατανεμημένη και ανοιχτού κώδικα μηχανή ανάλυσης για όλους τους τύπους δεδομένων. Είναι γνωστό για την ταχύτητα και την επεκτασιμότητά του. Σε συνδυασμό με την ικανότητά του να δημιουργεί ευρετήρια πολλούς τύπους περιεχομένου, το Elasticsearch χρησιμοποιείται για πολλές περιπτώσεις χρήσης όπως αναζήτηση σε εφαρμογές, επιχειρήσεις, παρακολούθηση απόδοσης εφαρμογών και αναλυτικά στοιχεία ασφαλείας για να αναφέρουμε μερικά.

Μια σημαντική εφαρμογή ανάλυσης του Elasticsearch είναι η ανάλυση ασφάλειας. Τα logs πρόσβασης και παρόμοια logs σχετικά με την ασφάλεια του συστήματος μπορούν να αναλυθούν με το Elastic Stack, παρέχοντας μια πιο ολοκληρωμένη εικόνα του τι συμβαίνει στα συστήματά ενός οργανισμού σε πραγματικό χρόνο.

Logstash

Παραδοσιακά, το Logstash έχει χρησιμοποιηθεί για την επεξεργασία logs από εφαρμογές και την αποστολή τους στην Elasticsearch, εξ ου και το όνομα. Αυτή παραμένει μια δημοφιλής περίπτωση χρήσης, αλλά το Logstash έχει εξελιχθεί σε ένα εργαλείο γενικότερου σκοπού, που σημαίνει ότι είναι ένας pipeline επεξεργασίας δεδομένων. Τα δεδομένα που λαμβάνει το Logstash αντιμετωπίζονται ως events, τα οποία μπορεί να είναι οτιδήποτε. Θα μπορούσαν να είναι καταχωρήσεις logs, παραγγελίες ηλεκτρονικού εμπορίου, πελάτες, μηνύματα συνομιλίας κ.λ.π. Αυτά τα συμβάντα επεξεργάζονται στη συνέχεια από το Logstash και αποστέλλονται σε έναν ή περισσότερους προορισμούς.

Ένα Logstash pipeline αποτελείται από τρία μέρη, ή στάδια, εισόδους, φίλτρα και εξόδους. Κάθε στάδιο μπορεί να κάνει χρήση κάποιου plugin. Ένα plugin εισόδου θα μπορούσε να είναι ένα αρχείο, για παράδειγμα, που σημαίνει ότι το Logstash θα διαβάσει συμβάντα από ένα δοθέν αρχείο. Το Logstash θα μπορούσε επίσης να λαμβάνει συμβάντα μέσω HTTP ή να αναζητά πληροφορίες από μια βάση δεδομένων

Beats

Το Beats είναι μια συλλογή των λεγόμενων αποστολέων δεδομένων.
Είναι ελαφριοί agents με έναν μόνο σκοπό που εγκαθίστανται σε διακομιστές, οι οποίοι στη συνέχεια στέλνουν δεδομένα στο Logstash ή το Elasticsearch. Υπάρχουν αρκετά είδη

αποστολέων δεδομένων,που ονομάζονται beats, συλλέγουν διαφορετικά είδη δεδομένων και εξυπηρετούν διαφορετικούς σκοπούς. Για παράδειγμα, υπάρχει ένα beat που ονομάζεται Filebeat, το οποίο χρησιμοποιείται για τη συλλογή logs και την αποστολή τους είτε στο Logstash είτε στο Elasticsearch. Το Filebeat απο την εγκατάσταση περιλαμβάνει modules για κοινά logs, όπως το nginx, ο διακομιστής web Apache ή η MySQL.

Σύντομα τα beats θα αντικατασταθούν απο έναν συγκεντρωτικό agent ο οποίος θα ρυθμίζεται αναλόγως τις ανάγκες συλλογής logs. Ο Elastic Agent είναι ένας μεμονωμένος agent που μπορεί να τοποθετηθεί σε hosts ή containers για τη συλλογή δεδομένων και την αποστολή τους στο Elastic Stack. Στο παρασκήνιο, το Elastic Agent εκτελεί τους αποστολείς Beats ή το Elastic Endpoint που απαιτείται για την εκάστοτε διαμόρφωση. Κατα τη σύνταξη του παρόντος κειμένου, το Elastic Agent είναι σε beta έκδοση και αναμένεται να δοκιμαστεί περαιτέρω.

Kibana

Το Kibana είναι μια πλατφόρμα ανάλυσης και οπτικοποίησης που επιτρέπει σε κάποιον να οπτικοποιήσει δεδομένα από την elasticsearch και να τα αναλύσει και να τα κατανοήσει. Το Kibana είναι ένας πίνακας εργαλείων όπου μπορεί κανείς να δημιουργήσει οπτικοποιήσεις όπως γραφήματα. Παρέχει επίσης μια διεπαφή για τη διαχείριση ορισμένων τμημάτων του elasticsearch, όπως έλεγχος ταυτότητας και εξουσιοδότηση δικαιωμάτων.

Δυνατότητες Elastic SIEM

Το Elastic SIEM είναι ευέλικτο επειδή επιτρέπει στον χρήστη να αποφασίσει ποιες πηγές πληροφοριών θα πρέπει να χρησιμοποιηθούν ως είσοδος στο σύστημα παρακολούθησης και ανίχνευσης δραστηριότητας. Το ίδιο το εργαλείο είναι απλώς ένας διερμηνέας δεδομένων και δεν έχει μεγάλη χρησιμότητα χωρίς ρύθμιση και επεξεργασία των logs και λοιπών δεδομένων.

Threat Intelligence

Η πιο σημαντική απαίτηση ενός SIEM που δεν παρέχει το Elastic NV είναι το threat intelligence. Η λειτουργία προστίθεται με τη μορφή του MITRE ATT&CK framework. Το Elastic παρέχει επίσης κανόνες ανίχνευσης δωρεάν. Ωστόσο, πρέπει να σημειωθεί πως οι κανόνες αυτοί δεν ενημερώνονται αυτόματα και δεν ανταγωνίζονται πλήρως με τα εξελιγμένα ερευνητικά τμήματα που τροφοδοτούν αλληλουχίες επιθέσεων και threat intelligence πληροφοριών σε ανταγωνιστικά συστήματα SIEM.

Παρακολούθηση συμπεριφοράς χρηστών

Το Elastic SIEM υλοποιεί ανάλυση συμπεριφοράς χρήστη και οντότητας, user and entity behavior analysis, ή εν συντομία UEBA. Αυτό έχει γίνει ένα τυπικό στοιχείο ενός συστήματος SIEM. Παρακολουθεί την κανονική συμπεριφορά από κάθε χρήστη ή ομάδα χρηστών και προσαρμόζει ανάλογα τα επίπεδα επικινδυνότητας της ειδοποίησης. Αυτή η διαδικασία χρησιμοποιεί τεχνικές μηχανικής εκμάθησης που βασίζονται σε AI και βοηθά στη μείωση των ψευδώς θετικών αναφορών. Πριν από την εισαγωγή του UEBA, τα συστήματα SIEM έτειναν να είναι υπερευαίσθητα και να επισημαίνουν μια σειρά νόμιμης δραστηριότητας ως ύποπτη.

Επίβλεψη δικαιωμάτων πρόσβασης

Το Elastic SIEM αναγνωρίζει τα logs από συστήματα διαχείρισης δικαιωμάτων πρόσβασης, όπως το Active Directory για να ανακαλύψει τις προσπάθειες των εισβολέων να εισβάλουν με τεχνικές που περιλαμβάνουν brute force επιθέσεις. Η συλλογή logs από όλο το σύστημα επιτρέπει στον διαχειριστή δικτύου να εντοπίζει μοτίβα δραστηριότητας που δείχνουν ότι ένας πιθανός επιτιθέμενος προσπαθεί να βρει μια διαδρομή στο δίκτυο που θα οδηγήσει στην πρόσβαση χρήσιμων πληροφοριών εκμεταλλευόμενος τα δικαιώματα των ήδη υπαρχόντων χρηστών.

Ανίχνευση ανωμαλιών

Οι κανόνες περι threat intelligence που παρέχονται από το Elastic ενεργοποιούν ειδοποιήσεις που προσελκύουν την προσοχή του προσωπικού υποστήριξης δικτύου στο dashboard του SIEM όπου παρέχονται περισσότερες πληροφορίες. Επιπλέον χαρακτηριστικά ανάλυσης υποστηρίζουν χειροκίνητες ρυθμίσεις για τη διεξαγωγή threat hunting.

Wazuh

Το Wazuh είναι μια δωρεάν, ανοιχτού κώδικα και έτοιμη για επιχειρήσεις λύση παρακολούθησης ασφάλειας για ανίχνευση απειλών, παρακολούθηση ακεραιότητας, απόκριση συμβάντων και συμμόρφωση.

Forked από το OSSEC, το Wazuh είναι ένα framework παρακολούθησης που χρησιμοποιεί agents για τη συλλογή logs και φιλτράρισμα σύμφωνα με τα σύνολα κανόνων που ορίζει ο διαχειριστής. Με ένα καλό προκαθορισμένο σύνολο κανόνων και μεγάλη προσαρμοστικότητα, το Wazuh είναι ένα εργαλείο που μπορεί να χρησιμοποιηθεί ως τα θεμέλια προς ένα ώριμο μοντέλο ασφάλειας.

Wazuh Server

Ο Wazuh server βασίζεται σε μια σειρά εφαρμογών όπου κάθε εφαρμογή ή στοιχείο έχει σχεδιαστεί για να εκτελεί μια συγκεκριμένη εργασία. Αυτά τα στοιχεία συνεργάζονται για την ανάλυση δεδομένων που λαμβάνονται από διάφορα logs, την ενεργοποίηση ειδοποιήσεων όταν ένα συμβάν ταιριάζει με έναν κανόνα, την εγγραφή νέων agents και την αποστολή δεδομένων στον Elastic Stack server.

Αποτελείται απο τον wazuh manager, την υπηρεσία του registration, το Wazuh API και το Filebeat.

- **Wazuh Manager**
  Ο Wazuh Manager λαμβάνει και αναλύει δεδομένα από τους agents χρησιμοποιώντας αποκωδικοποιητές και κανόνες που έχουν δημιουργηθεί για την ενεργοποίηση ειδοποιήσεων ασφαλείας. Ο manager χρησιμοποιείται επίσης για τη διανομή configuration αρχείων στους agents, για την παρακολούθηση της κατάστασής τους και για την αποστολή μηνυμάτων ελέγχου για την ενεργοποίηση αυτόματων ενεργειών σε επίπεδο agent.

- **Registration service**
  Το registration service χρησιμοποιεί έναν ασφαλή μηχανισμό για την εγγραφή agents χωρίς παρέμβαση από την πλευρά του server.

- **Wazuh API**
  Το Wazuh API παρέχει μια διεπαφή για τη διαχείριση και παρακολούθηση του configuration του manager και των agents. Μπορεί να χρησιμοποιηθεί για την καταχώριση agents, τον έλεγχο των manager logs, τους αποκωδικοποιητές και τους κανόνες και την παροχή χρήσιμων πληροφοριών που σχετίζονται με τους agents, συμπεριλαμβανομένης της κατάστασής τους, των λεπτομερειών του λειτουργικού

συστήματος και των ειδοποιήσεων που σχετίζονται με την παρακολούθηση της ακεραιότητας αρχείων και των rootcheck.

- **Filebeat**
  Το Filebeat χρησιμοποιείται σε κατανεμημένες αρχιτεκτονικές, όπου ο διακομιστής Wazuh και το Elastic Stack είναι εγκατεστημένα σε διαφορετικά συστήματα, για την προώθηση δεδομένων ειδοποιήσεων στο Logstash. Αυτό το στοιχείο έχει αναπτυχθεί από την Elastic.

Wazuh Agents

Ο Wazuh agent εκτελείται στα συστήματα που χρήζουν παρακολούθησης εντός του δικτύου και είναι υπεύθυνος για τη συλλογή logs και συμβάντων. Το Rootcheck εκτελεί εντοπισμό rootkit και κακόβουλο λογισμικό σε κάθε σύστημα όπου είναι εγκατεστημένος ο agent. Η παρακολούθηση και ανάλυση log συλλέγει και αναλύει logs συστήματος και αναζητά οποιαδήποτε ύποπτη δραστηριότητα. Το Syscheck εκτελείται περιοδικά για τον έλεγχο αλλαγών σε οποιοδήποτε configuration αρχείο ή registry entry στα Windows. Το OpenSCAP έχει σχεδιαστεί για να ελέγχει για αδύναμες ή ευάλωτες εφαρμογές και configurations.

Registration Process

Η συλλογή δεδομένων συμβάντων ασφαλείας από τον Wazuh agent απαιτεί την ενεργοποίηση της επικοινωνίας με τον Wazuh manager. Ο Wazuh manager πρέπει να γνωρίζει ποιος Wazuh agent στέλνει τα συμβάντα ασφαλείας και εάν είναι εξουσιοδοτημένος. Αυτό το βήμα ονομάζεται εγγραφή Wazuh agent και μπορεί να γίνει χρησιμοποιώντας την υπηρεσία εγγραφής. Χρησιμοποιώντας τη θύρα 1515 και πρωτόκολλο TCP, ο Wazuh manager παρακολουθεί το αίτημα εγγραφής του Wazuh agent χρησιμοποιώντας μια σύνδεση TLS. Ο Wazuh agent θα αποκτήσει ένα μοναδικό κλειδί, που χρησιμοποιείται για την κρυπτογράφηση της κίνησης μεταξύ τους. Μόλις ολοκληρωθεί η εγγραφή, αυτή η επικοινωνία δεν θα χρησιμοποιείται πλέον, εκτός εάν ο Wazuh agent πρέπει να εγγραφεί σε έναν νέο Wazuh manager.

Μετά την εγγραφή, ο Wazuh agent πρέπει να ρυθμιστεί ώστε να υποδεικνύει τον προορισμό στον οποίο θα αποστέλλονται τα συλλεγόμενα συμβάντα ασφαλείας. Από προεπιλογή, ο Wazuh manager θα χρησιμοποιήσει ένα κανάλι επικοινωνίας μέσω της θύρας 1514 χρησιμοποιώντας πρωτόκολλο UDP, μέσω του οποίου ο Wazuh agent θα στείλει τα δεδομένα που συλλέχθηκαν.

Απλό Registration χωρίς επαλήθευση

Η διαδικασία που περιγράφεται παρακάτω δεν λαμβάνει μέτρα για τη προστασία του Wazuh manager, Wazuh agent και της μεταξύ τους επικοινωνίας. Δεν ενδείκνυται για χρήση σε περιβάλλον production, αλλα μόνο για δοκιμαστική λειτουργία.

Το παρόν δοκιμαστικό δίκτυο αποτελείται από:

- Wazuh manager CentOS host: `192.168.1.19`
- Wazuh agent Ubuntu host: `192.168.1.13`

Σε ένα terminal του Wazuh agent host εκτελούνται οι παρακάτω εντολές με δικαιώματα root χρήστη.

1. Για το registration του agent χρησιμοποιείται το βοηθητικό πρόγραμμα “agent-auth” με την IP διεύθυνση του manager.

```
# /var/ossec/bin/agent-auth -m 192.168.1.19
```

2. Για την ενεργοποίηση της επικοινωνίας με τον manager, επεξεργαζεται το αρχείο διαμόρφωσης του agent, το οποίο βρίσκεται στο directory “/var/ossec/etc/ossec.conf”. Στην ενότητα ”<client> <server>”, το *MANAGER_IP* πρέπει να αντικατασταθεί με τη διεύθυνση IP του manager ή το όνομα που έχει δηλωθεί στο DNS.

```
<client>
  <server>
    <address>192.168.1.19</address>
    ...
  </server>
</client>
```

3. Για να εφαρμοστούν οι παραπάνω αλλαγές επανεκκινείται ο agent.

```
# systemctl restart wazuh-agent
```

Registration με επαλήθευση Manager και Agent

Το παρόν δοκιμαστικό δίκτυο αποτελείται από:

- Wazuh manager CentOS host: `192.168.1.19`
- Wazuh agent Ubuntu host: `192.168.1.13`

Οι εντολές που παρατίθενται παρακάτω εκτελούνται με δικαιώματα root στον host του Wazuh manager.

Επαλήθευση Manager

Η επαλήθευση με πιστοποιητικό κλειδιού SSL διασφαλίζει ότι έχει δημιουργηθεί η σύνδεση μεταξύ του σωστού Wazuh agent και του σωστού Wazuh manager. Η υπηρεσία εγγραφής με πιστοποίηση SSL απαιτεί τη δημιουργία ενός Certificate Authority, ή CA εν συντομία, που χρησιμοποιείται για την υπογραφή πιστοποιητικών για τον Wazuh manager και τους Wazuh agents. Ο κάθε host θα λάβει ένα αντίγραφο αυτού του CA προκειμένου να επαληθεύσει το πιστοποιητικό του. Υπάρχουν δύο επιλογές για την εγγραφή του Wazuh agent χρησιμοποιώντας την επαλήθευση host, η επαλήθευση του Wazuh agent καθώς και η επαλήθευση του Wazuh manager. Σημειώνεται πως και οι δύο μπορούν να χρησιμοποιηθούν στη διαδικασία εγγραφής.

Για τη δημιουργία του CA πιστοποιητικού γίνεται χρήση της βιβλιοθήκης openssl

```
# openssl req -x509 -new -nodes -newkey rsa:4096 -keyout rootCA.key -out
rootCA.pem -batch -subj "/C=US/ST=CA/O=Manager"
```

Το αρχείο rootCA.key που δημιουργήθηκε πρόσφατα είναι το ιδιωτικό κλειδί του CA. Απαιτείται για την υπογραφή άλλων πιστοποιητικών και είναι πολύ σημαντικό να διατηρηθεί ασφαλές και να μην αντιγραφεί ποτέ σε άλλους hosts.

Για την επαλήθευση του Wazuh manager χρησιμοποιώντας SSL, δημιουργείται ένα πιστοποιητικό SSL και υπογράφεται χρησιμοποιώντας το κλειδί του CA. Αυτό θα επιτρέψει στους Wazuh agents να διασφαλίσουν ότι είναι συνδεδεμένοι με τον σωστό Wazuh manager κατά τη διάρκεια της υπηρεσίας εγγραφής.

Δημιουργείται το configuration αρχείο req.conf, και αντικαθίσταται το <manager_IP> πρέπει να αντικατασταθεί με τη διεύθυνση IP, ή το όνομα που έχει δηλωθεί στο DNS, του Wazuh manager όπου πρόκειται να εγγραφούν οι Wazuh agents. Το αρχείο διαμόρφωσης έχει ως εξής:

```
[req]
distinguished_name = req_distinguished_name
req_extensions = req_ext
prompt = no
[req_distinguished_name]
C = US
CN = 192.168.1.19
[req_ext]
subjectAltName = @alt_names
[alt_names]
DNS.1 = wazuh
DNS.2 = wazuh.com
```

Για την επαλήθευση του Wazuh manager χρησιμοποιώντας ένα πιστοποιητικό SSL, δημιουργείται ένα πιστοποιητικό SSL για τον Wazuh agent και υπογράφεται χρησιμοποιώντας το CA που δημιουργήθηκε.

```
# openssl req -new -nodes -newkey rsa:4096 -keyout sslmanager.key -out
sslmanager.csr -config req.conf
```

```
# openssl x509 -req -days 365 -in sslmanager.csr -CA rootCA.pem -CAkey
rootCA.key -out sslmanager.cert -CAcreateserial -extfile req.conf
-extensions req_ext
```

Το κλειδί και το πιστοποιητικό αντιγράφονται στο directory /var/ossec/etc/

```
# cp sslmanager.key sslmanager.cert /var/ossec/etc
```

Ο Wazuh Manager επανεκκινείται για να εφαρμοστούν οι παραπάνω αλλαγές

```
# systemctl restart wazuh-manager
```

Οι παρακάτω εντολές εκτελούνται στον host του agent με δικαιώματα root.

Το αρχείο rootCA.pem αντιγράφεται στον host του agent και τοποθετείται στο directory /var/ossec/etc/

```
# cp rootCA.pem /var/ossec/etc
```

Για το registration του agent χρησιμοποιείται το βοηθητικό πρόγραμμα “agent-auth” με την IP διεύθυνση του manager και την τοποθεσία του rootCA.pem

```
# /var/ossec/bin/agent-auth -m 192.168.1.19 -v /var/ossec/etc/rootCA.pem
```

Για την ενεργοποίηση της επικοινωνίας με τον Wazuh manager, επεξεργάζεται το configuration αρχείο του Wazuh agent που βρίσκεται στο directory /var/ossec/etc/ossec.conf

Για την ενεργοποίηση της επικοινωνίας με τον manager, επεξεργαζεται το αρχείο διαμόρφωσης του agent, το οποίο βρίσκεται στο directory “/var/ossec/etc/ossec.conf”.
Στην ενότητα ”<client> <server>”, το *MANAGER_IP* πρέπει να αντικατασταθεί με τη διεύθυνση IP του manager ή το όνομα που έχει δηλωθεί στο DNS

```
<client>
  <server>
    <address>192.168.1.19</address>
    ...
  </server>
</client>
```

Για να εφαρμοστούν οι παραπάνω αλλαγές επανεκκινείται ο agent.

```
      # systemctl restart wazuh-agent
```

Επαλήθευση Agent

Η διαδικασία επαλήθευσης του Wazuh agent θα επιτρέψει στον Wazuh manager να διασφαλίσει ότι ο σωστός Wazuh agent συνδέεται κατά τη διάρκεια της υπηρεσίας εγγραφής. Η επαλήθευση του Wazuh agent μπορεί να ολοκληρωθεί προαιρετικά επαληθεύοντας τον host στον οποίο είναι εγκατεστημένος ο agent. Όμως παρακάμπτοντας την επαλήθευση του host θα επιτραπεί η κοινή χρήση του ίδιου πιστοποιητικού σε παραπάνω από έναν Wazuh agent. Το υπογεγραμμένο πιστοποιητικό θα επαληθεύσει τον Wazuh agent. Η υπηρεσία εγγραφής για Wazuh agents όπου δεν υπάρχει το πιστοποιητικό στον host θα απορριφθεί.

Για την έκδοση και υπογραφή του πιστοποιητικού για τον agent εκτελώντας τις ακόλουθες εντολές στη θέση των αρχείων CA. Οι εντολές που παρατίθενται παρακάτω εκτελούνται με δικαιώματα root στον host του Wazuh manager.

```
# openssl req -new -nodes -newkey rsa:4096 -keyout sslagent.key -out
sslagent.csr -batch
```

Προαιρετικά και αν είναι επιθυμητή η επαλήθευση του host, η επιλογή -batch αντικαθίσταται απο την επιλογή

```
-subj '/C=US/CN=<agent_IP>'
```

```
# openssl x509 -req -days 365 -in sslagent.csr -CA rootCA.pem -CAkey
rootCA.key -out sslagent.cert -CAcreateserial
```

Το πιστοποιητικό αντιγράφονται στο directory /var/ossec/etc/

```
# cp rootCA.pem /var/ossec/etc
```

Γίνεται τροποποίηση του αρχείου /var/ossec/etc/ossec.conf για την ενεργοποίηση της επαλήθευσης του host. Αφαιρούνται τα σχόλια στην ενότητα <auth> <ssl_agent_ca> και προστίθεται η διαδρομή για το αρχείο CA.

```
<auth>
  ...
  <ssl_agent_ca>/var/ossec/etc/rootCA.pem</ssl_agent_ca>
  ...
</client>
```

Προαιρετικά και αν είναι επιθυμητή η επαλήθευση του host προστίθεται η παρακάτω γραμμή μετά το πεδίο <ssl_agent_ca></ssl_agent_ca>

```
 <ssl_verify_host>yes</ssl_verify_host>
```

Ο Wazuh Manager επανεκκινείται για να εφαρμοστούν οι παραπάνω αλλαγές

```
# systemctl restart wazuh-manager
```

Για το registration του agent χρησιμοποιώντας το πιστοποιητικό και το κλειδί και επιτρέποντας την επικοινωνία με τον διαχειριστή Wazuh εκτελούνται οι παρακάτω εντολές στον host του agent με δικαιώματα root.

Το πιστοποιητικό (αρχείο .cert) και το κλειδί του (αρχείο .key), που δημιουργήθηκε προηγουμένως στον manager, αντιγράφονται στο φάκελο /var/ossec/etc

```
# cp sslagent.cert sslagent.key /var/ossec/etc
```

Για το registration του agent χρησιμοποιείται το βοηθητικό πρόγραμμα “agent-auth” με την IP διεύθυνση του manager

```
# /var/ossec/bin/agent-auth -m <manager_IP> -x /var/ossec/etc/sslagent.cert
-k /var/ossec/etc/sslagent.key
```

Για την ενεργοποίηση της επικοινωνίας με τον manager, επεξεργαζεται το αρχείο διαμόρφωσης του agent, το οποίο βρίσκεται στο directory “/var/ossec/etc/ossec.conf”.
Στην ενότητα ”<client> <server>”, το *MANAGER_IP* πρέπει να αντικατασταθεί με τη διεύθυνση IP του manager ή το όνομα που έχει δηλωθεί στο DNS

```
<client>
  <server>
    <address>192.168.1.19</address>
    ...
  </server>
</client>
```

Για να εφαρμοστούν οι παραπάνω αλλαγές επανεκκινείται ο agent.

```
      # systemctl restart wazuh-agent
```

Απομακρυσμένη Ρύθμιση Wazuh Agent

Ο Wazuh manager έχει τη δυνατότητα να ρυθμίζει απομακρυσμένα τους Wazuh agents. Οι ρυθμίσεις αποθηκεύονται στο αρχείο agent.conf στον host του Wazuh manager και μετά την ανίχνευση αλλαγών στο αρχείο αυτό, οι νέες ρυθμίσεις προωθούνται στους agents. Οι agents μπορούν να ομαδοποιηθούν ώστε να τους αποστέλλονται ρυθμίσεις που είναι συγκεκριμένες για την εκάστοτε ομάδα. Κάθε agent μπορεί να ανήκει σε περισσότερες από μία ομάδες και εκτός εάν έχει διαμορφωθεί διαφορετικά η δομή των agents, όλοι ανήκουν σε μια ομάδα που ονομάζεται default.

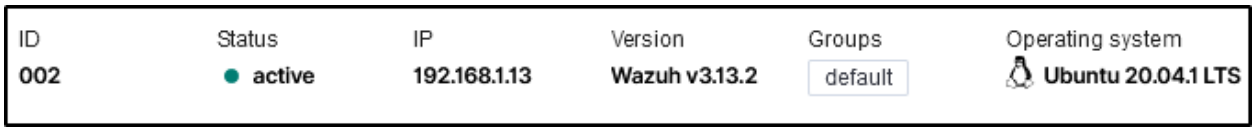
Κάθε Wazuh agent group έχει ξεχωριστό directory ρυθμίσεων στον host του Wazuh manager. Το directory των groups είναι το /var/ossec/etc/shared/ και κάθε σύνολο ρυθμίσεων αποθηκεύεται στο αντίστοιχο directory. Για παράδειγμα οι ρυθμίσεις του default group βρίσκονται στο directory /var/ossec/etc/shared/default. Σε περίπτωση που ένας agent έχει εκχωρηθεί σε πολλές ομάδες, όλα τα αρχεία που περιέχονται σε κάθε group directory θα συγχωνευτούν σε ένα και θα σταλούν στον agent. Σημειώνεται πως ο χρήστης ossec, ο οποίος δημιουργείται κατα την εγκατάσταση του wazuh, πρέπει να έχει δικαίωμα read στα αρχεία ρυθμίσεων.

Ο agent θα αποθηκεύσει τα κοινόχρηστα αρχεία στο /var/ossec/etc/shared και οχι σε ένα group directory.

Είναι σημαντικό να γίνει κατανοητό ποιο αρχείο ρυθμίσεων έχει προτεραιότητα μεταξύ του ossec.conf, το οποίο χρησιμοποιείται για ρυθμίσεις τοπικα στον host του agent, και του agent.conf όταν χρησιμοποιείται η κεντρική ρύθμιση των agents.Στη περίπτωση αυτή, η τοπική και η κοινόχρηστη διαμόρφωση συγχωνεύονται, ωστόσο, το αρχείο ossec.conf διαβάζεται πριν από το κοινόχρηστο agent.conf και η τελευταία διαμόρφωση οποιασδήποτε ρύθμισης θα αντικαταστήσει την προηγούμενη. Επίσης, εάν έχει οριστεί διαδρομή αρχείου για μια συγκεκριμένη ρύθμιση και στα δύο αρχεία διαμόρφωσης, και οι δύο διαδρομές θα συμπεριληφθούν στην τελική διαμόρφωση. Συνοπτικά, οι ρυθμίσεις στο agents.conf υπερισχύουν των ρυθμίσεων στο ossec.conf.

Use Case

Στο ακόλουθο παράδειγμα, παρουσιάζεται η ρύθμιση ενός Wazuh agent απομακρυσμένα μέσω του Wazuh manager. Οι εντολές εκτελούνται σε τερματικό του Wazuh manager με δικαιώματα root. Ο agent, στη προκειμένη περίπτωση, έχει ID 002, λειτουργικό σύστημα Ubuntu 20 και ανήκει στο default group.



ID	Status	IP	Version	Groups	Operating system
002	active	192.168.1.13	Wazuh v3.13.2	default	Ubuntu 20.04.1 LTS

Αρχικά, γίνεται ομαδοποίηση των agents στο αντίστοιχο group ανάλογα με το εγκατεστημένο λειτουργικό σύστημα. Για την δημιουργία του ubuntu group εκτελείται η παρακάτω εντολή

```
# /var/ossec/bin/agent_groups -a -g ubuntu
```

Έπειτα το νέο ubuntu group ανατίθεται στον agent.

```
# /var/ossec/bin/agent_groups -a -i 002 -g ubuntu
Do you want to add the group 'ubuntu' to the agent '002'? [y/N]: y
Group 'ubuntu' added to agent '002'.
```

Αφαιρείται το default group απο τον agent, ωστε να δέχεται configuration που αφορά αποκλειστικά το ubuntu group.

```
# /var/ossec/bin/agent_groups -r -i 002 -g default
Do you want to delete the group 'default' of agent '002'? [y/N]: y
Agent '002' removed from 'default'.
```

Στο παράδειγμα αυτό θα προωθηθεί configuration για rootkit detection στο ubuntu group. Στο directory του manager /var/ossec/etc/shared/ubuntu γίνεται η ακόλουθη αλλαγή στο αρχείο agent.conf

```
# vim agent.conf
```

```
<agent_config>

<rootcheck>
  <rootkit_files>/var/ossec/etc/shared/rootkit_files.txt</rootkit_files>
</rootcheck>

</agent_config>
```

Στον host του ubuntu agent έχουν προωθηθεί οι παραπάνω ρυθμίσεις και βρίσκονται στο αρχείο /var/ossec/etc/shared/agent.conf

```
root@victim-VirtualBox:/var/ossec/etc/shared# cat agent.conf
<agent_config>

<rootcheck>
  <rootkit_files>/var/ossec/etc/shared/rootkit_files.txt</rootkit_files>
</rootcheck>

</agent_config>
```

Έλεγχος Ακεραιότητας Αρχείων

Η μονάδα File Integrity Monitoring, ή FIM εν συντομία, βρίσκεται στον Wazuh agent. Εκτελεί περιοδικές σαρώσεις του συστήματος,αποθηκεύει τα hash ελέγχου και τα χαρακτηριστικά των αρχείων που παρακολουθούνται και τα κλειδιά μητρώου των Windows σε μια τοπική βάση δεδομένων FIM. Αναζητά τις τροποποιήσεις συγκρίνοντας τα hash ελέγχου των νέων αρχείων με τα παλιά hash ελέγχου. Όλες οι εντοπισμένες αλλαγές αναφέρονται στον Wazuh manager.

Με την εγκατάσταση εφαρμόζονται προεπιλεγμένες ρυθμίσεις και κάθε Wazuh agent έχει το syscheck ενεργοποιημένο και προκαθορισμένο. Το syscheck είναι η ενότητα ρυθμίσεων που αφορούν τον έλεγχο της ακεραιότητας αρχείων στο configuration file “ossec.conf”. Συνιστάται να ελέγχεται και να τροποποιείται η ρύθμιση του παρακολουθούμενου κόμβου ή τερματικού. Τα αποτελέσματα παρακολούθησης ακεραιότητας αρχείων για ολόκληρο το περιβάλλον μπορούν να παρατηρηθούν στον πίνακα ελέγχου της εφαρμογής Kibana.

Προτείνεται επίσης να προστεθούν οι παρακάτω ρυθμίσεις στο αρχείο /var/ossec/etc/ossec.conf για την βελτίωση του εύρους και της αμεσότητας παρακολούθησης:

```
<alert_new_files>yes</alert_new_files>
```

Σηματοδοτεί την δημιουργία security event στο Wazuh Dashboard κατα την δημιουργία ή αποθήκευση νέου αρχείου. Χωρίς αυτή τη προσθηκη τα events αφορούν μόνο τροποποίηση ήδη υπάρχοντος αρχείου.

```
<directories check_all="yes">...</directories>
```

Η επιλογή *check_all* επιτρέπει ελέγχους του μεγέθους του αρχείου, των δικαιωμάτων, του κατόχου, της τελευταίας ημερομηνίας τροποποίησης, του inode και όλων των hash (MD5, SHA1 και SHA256).

```
<directories realtime="yes">...</directories>
```

Η παρακολούθηση σε πραγματικό χρόνο ενεργοποιείται με την επιλογή *realtime*. Αυτή η επιλογή λειτουργεί μόνο με τα directories και όχι με τα μεμονωμένα αρχεία. Η ανίχνευση αλλαγών σε πραγματικό χρόνο διακόπτεται κατά τη διάρκεια περιοδικών σαρώσεων syscheck και επανενεργοποιείται μόλις ολοκληρωθούν.

Ενσωμάτωση VirusTotal

Το VirusTotal είναι μια διαδικτυακή υπηρεσία που αναλύει αρχεία και διευθύνσεις URL για την ανίχνευση ιών, worm, trojan και άλλων ειδών κακόβουλου περιεχομένου χρησιμοποιώντας μηχανές προστασίας από ιούς και σαρωτές ιστότοπων. Είναι μια δωρεάν υπηρεσία με πολλές χρήσιμες δυνατότητες με κύριες τις παρακάτω:

- Το VirusTotal αποθηκεύει όλες τις αναλύσεις που εκτελεί, οι οποίες επιτρέπουν την αναζήτηση hash ενός συγκεκριμένου αρχείου. Με την αναζήτηση του hash στο VirusTotal μπορεί να γίνει γνωστό εάν το συγκεκριμένο αρχείο έχει ήδη σαρωθεί από το VirusTotal στο παρελθόν.

- Το VirusTotal παρέχει επίσης ένα API που επιτρέπει την πρόσβαση στις πληροφορίες που δημιουργούνται από το VirusTotal χωρίς να χρειάζεται η χρήση browser. Σημειώνεται πως η δωρεάν έκδοση του API έχει τα όρια των 4 request ανα λεπτό και 500 request ανα ημέρα.

Ο έλεγχος ακεραιότητας αρχείων έχει τη δυνατότητα να συνδεθεί άμεσα με την υπηρεσία του VirusTotal. Αυτή η ενοποίηση χρησιμοποιεί το API του VirusTotal για τον εντοπισμό κακόβουλου περιεχομένου στα αρχεία που παρακολουθούνται από την παρακολούθηση ακεραιότητας αρχείων, File Integrity Monitoring ή FIM εν συντομία. Το FIM αναζητά οποιαδήποτε προσθήκη, αλλαγή ή διαγραφή αρχείων στους φακέλους που παρακολουθούνται. Αυτή η ενότητα αποθηκεύει το hash αυτών των αρχείων και ενεργοποιεί ειδοποιήσεις όταν παρατηρηθεί οποιαδήποτε αλλαγή. Η ενσωμάτωση VirusTotal ενεργοποιείται όταν εμφανίζεται μια ειδοποίηση FIM. Από αυτήν την ειδοποίηση εξάγεται το πεδίο hash του αρχείου.

Στη συνέχεια υποβάλλεται ένα αίτημα HTTP POST στη βάση δεδομένων VirusTotal χρησιμοποιώντας το VirusTotal API για σύγκριση μεταξύ του υποβεβλημένου hash και των πληροφοριών που περιέχονται στη βάση δεδομένων.

Στη συνέχεια λαμβάνεται μια απάντηση JSON που είναι το αποτέλεσμα αυτής της αναζήτησης που θα ενεργοποιήσει μία από τις ακόλουθες ειδοποιήσεις:

- Error: Public API request rate limit reached.
- Error: Check credentials.
- Alert: No records in VirusTotal database.
- Alert: No positives found.
- Alert: X engines detected this file.

Use Case

Στο παρακάτω σενάριο εξετάζεται η περίπτωση όπου ένας χρήστης λαμβάνει απο το internet και αποθηκεύει ένα κακόβουλο εκτελέσιμο τοπικά σε περιβάλλον Ubuntu 20.

Το εκτελέσιμο δημιουργήθηκε κάνοντας χρήση του mfsvenom και με την εκτέλεσή του δημιουργείται ένα reverse meterpreter shell στη συσκευή του επιτιθέμενου.

Στην παρακάτω εικόνα φαίνονται οι ειδοποιήσεις που δημιουργεί το FIM της συσκής του θύματος. Υπάρχει μια ειδοποίηση για την προσθήκη του κακόβουλου αρχείου στον φάκελο που παρακολουθείται και μια δεύτερη, η οποία δημιουργήθηκε όταν ο χρήστης άλλαξε τα δικαιώματα του αρχείου και δόθηκε το δικαίωμα της εκτέλεσης.

FIM: Recent events

Time	Path	Action	Rule description	Rule Level	Rule Id
2020/11/18 19:16:53	/home/victim/Downloads/shell64.elf	modified	Integrity checksum changed.	7	550
2020/11/18 19:16:46	/home/victim/Downloads/shell64.elf	modified	Integrity checksum changed.	7	550
2020/11/18 19:16:46	/home/victim/Downloads/shell64.elf	modified	Integrity checksum changed.	7	550
2020/11/18 19:16:46	/home/victim/Downloads/shell64.elf	modified	Integrity checksum changed.	7	550
2020/11/18 19:16:46	/home/victim/Downloads/shell64.elf	added	File added to the system.	5	554

Παρακάτω φαίνεται η απάντηση απο το API του VirusTotal στο request που στάλθηκε με το hash του εκτελέσιμου αρχείου.

```
** Alert 1605708643.998088: mail  - virustotal,gdpr_IV_35.7.d,
2020 Nov 18 14:10:43 (victim-VirtualBox) 192.168.1.13->virustotal
Rule: 87105 (level 12) -> 'VirusTotal: Alert - /home/victim/Downloads/shell64.elf - 13 engines detected this file'
{"virustotal": {"found": 1, "malicious": 1, "source": {"alert_id": "1605708635.995915", "file": "/home/victim/Downloads/shell64.elf",
virustotal.found: 1
virustotal.malicious: 1
```

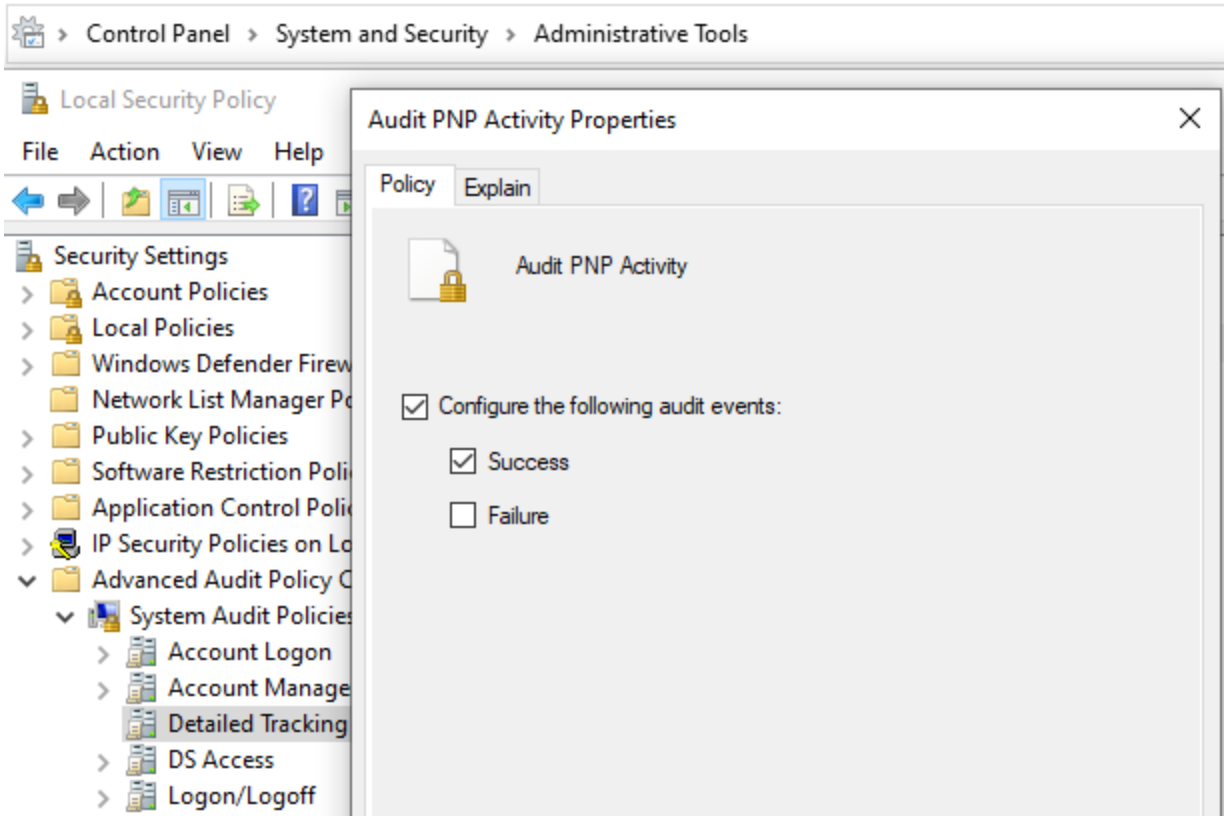
Το εκτελέσιμο αναγνωρίστηκε και επισημάνθηκε ως κακόβουλο.

Επίβλεψη USB σε Windows

Ο στόχος της παρακολούθησης USB είναι να δημιουργείται μια ειδοποίηση όταν μια συσκευή αποθήκευσης συνδέεται σε ένα σύστημα Windows που παρακολουθείται από Wazuh agent. Επιπλέον, είναι δυνατή η δημιουργία μιας λίστας με εξουσιοδοτημένες συσκευές, ώστε να εντοπίζεται μια μη εξουσιοδοτημένη εισβολή.

Οι ρυθμίσεις εφαρμόζονται σε όλες τις εκδόσεις των Windows. Στο παρακάτω παράδειγμα ο host του agent διαθέτει λειτουργικό σύστημα Windows 10.

Αρχικά, ενεργοποιείται στον host η δημιουργία event κατα την εισαγωγή συσκευής αποθήκευσης USB. Η ενεργοποίηση γίνεται στο παράθυρο Administrative Tools > Local Security Policy > Advanced Audit Policy Configuration > System Audit Policies > Detailed Tracking και επιλέγοντας το Audit PNP Activity, δηλαδή επιτήρηση συσκευών Plug and Play. Στο παράθυρο που εμφανίζεται, στην καρτέλα Policy, ενεργοποιείται η δημιουργία event κατα την επιτυχή εισαγωγή συσκευής USB.



Για την διάκριση μεταξύ εξουσιοδοτημένων και μη συσκευών είναι δυνατή η δημιουργία whitelist για έμπιστες αποθηκευτικές συσκευές USB. Έτσι, μόνο μη εξουσιοδοτημένες συσκευές θα δημιουργούν alert στο Wazuh dashboard. Για τον σκοπό αυτόν, δημιουργείται μια CDB λίστα η οποία περιέχει τα ID των εξουσιοδοτημένων συσκευών USB. Για την εύρεση του ID μιας συσκευής, αφού εισαχθεί επιτυχώς σε Windows host, εκτελείται η παρακάτω εντολή σε τερματικό Powershell με δικαιώματα διαχειριστή.

Τα δυο πρώτα αποτελέσματα της εικόνας αφορούν σκληρούς δίσκους του host, ενω το τρίτο, τη συσκευή USB που θα συμπεριληφθεί στη λίστα των εξουσιοδοτημένων συσκευών.

Αντίστοιχα σε Linux host, η παραπάνω εντολή αντικαθίσταται απο το

```
 # usb-devices
```

script, το οποίο εκτελείται σε τερματικό bash.

Δημιουργείται η λίστα επιτρεπόμενων συσκευών στον host του manager

```
# vim etc/lists/usb-devices
```

```
0415130000013165:Test_USB_Device
```

Στο πρώτο μέρος φαίνεται το ID της συσκευής, ενώ μετα την ‘:’ παρατίθεται προαιρετικό σχόλιο που αφορά την ίδια συσκευή.

Το directory της νέας λίστας πρέπει να συμπεριληφθεί στο αρχείο ossec.conf. Διαφορετικά, ο manager δεν θα μπορεί να το διαβάσει:

```
# vim /var/ossec/etc/ossec.conf
```

```
<ruleset>
    <!-- Default ruleset -->
    <decoder_dir>ruleset/decoders</decoder_dir>
    <rule_dir>ruleset/rules</rule_dir>
    <rule_exclude>0215-policy_rules.xml</rule_exclude>
    <list>etc/lists/audit-keys</list>
    <list>etc/lists/amazon/aws-eventnames</list>
    <list>etc/lists/security-eventchannel</list>

    <!-- User-defined ruleset -->
    <decoder_dir>etc/decoders</decoder_dir>
    <rule_dir>etc/rules</rule_dir>
    <list>etc/lists/usb-devices</list>
  </ruleset>
```

Το τελευταίο βήμα είναι να μεταγλωττιστεί το νέο αρχείο χρησιμοποιώντας το ossec-makelists. Εκτελείται η εντολή /var/ossec/bin/ossec-makelists με το ακόλουθο αποτέλεσμα

```
* File etc/lists/usb-devices.cdb needs to be updated
```

Έπειτα τροποποιείται το σύνολο κανόνων. Καταρχάς, πρέπει να προστεθεί ένας νέος αποκωδικοποιητής που επιτρέπει την εξαγωγή του απαιτούμενου σειριακού αριθμού όταν εντοπίζεται μια νέα συσκευή αποθήκευσης USB. Προστίθεται ο decoder που φαίνεται παρακάτω στο ακόλουθο αρχείο: 0380-windows_decoders.xml.

```
# vim /var/ossec/ruleset/decoders/0380-windows_decoders.xml
```

```
<decoder name="windows_fields">
  <type>windows</type>
  <parent>windows</parent>
  <regex>USBSTOR#Disk&Ven_(\S*)&Prod_(\S*)&Rev_(\.*)#(\S*)&0#\S*\s</regex>
  <order>usb.vendor, usb.product, usb.rev, usb.serial_number</order>
</decoder>
```

Τέλος, δημιουργούνται δύο διαφορετικοί κανόνες. Ένας για να δημιουργείται ειδοποίηση όταν εντοπίζεται μια εξουσιοδοτημένη συσκευή και ένας δεύτερος για τη δημιουργία ειδοποίησης μη εξουσιοδοτημένης συσκευής. Ο δεύτερος θα ελέγξει τη λίστα CDB με σκοπό να βρεθεί αντιστοιχία με το αναγνωριστικό της συσκευής, το οποίο είναι αποθηκευμένο στο πεδίο usb.serial_number. Και οι δύο κανόνες πρέπει να προστεθούν στο local_rules.xml.

```
 # vim /var/ossec/etc/rules/local_rules.xml
```

```
<rule id="100002" level="5">
  <if_sid>18104</if_sid>
  <id>^6416$</id>
  <description>Windows: Authorized PNP device connected.</description>
</rule>
<rule id="100003" level="7">
  <if_sid>18104</if_sid>
  <id>^6416$</id>
  <list field="usb.serial_number"
lookup="not_match_key">etc/lists/usb-devices</list>
  <description>Windows: Unauthorized PNP device connected.</description>
</rule>
```

Για να εφαρμοστούν οι παραπάνω αλλαγές επανεκκινείται ο agent.

```
      # systemctl restart wazuh-agent
```

Τα alerts στο αρχείο alerts.log έχουν τη παρακάτω μορφή.

```
** Alert 1495797025.37192: - local,syslog,sshd,
2017 May 26 11:10:25 (windows_agent) any->WinEvtLog
Rule: 100003 (level 7) -> 'Windows: Unauthorized PNP device connected.'
User: (no user)
2017 May 26 04:10:24 WinEvtLog: Security: AUDIT_SUCCESS(6416):
Microsoft-Windows-Security-Auditing: (no user): no domain: WIN-EDHF85L4G6H:
A new external device was recognized by the system. Subject: Security ID:
S-1-5-18 Account Name: WIN-EDHF85L4G6HAccount Domain: WORKGROUP Logon ID:
0x3E7 Device ID:
SWD\WPDBUSENUM\_??_USBSTOR#Disk&Ven_OTi6828&Prod_Flash_Disk&Rev_1.89#1B3D42
CB4E7400D4&0#{53f56307-b6bf-11d0-94f2-00a0c91efb8b} Device Name: 86JT19A1
Class ID: {EEC5AD98-8080-425F-922A-DABF3DE3F69A} Class Name: WPD Vendor
IDs: - Compatible IDs: wpdbusenum\fs SWD\Generic Location Information: -
account_name: WIN-EDHF85L4G6H$
account_domain: WORKGROUP
logon_id: 0x3E7
usb.vendor: OTi6828
usb.product: Flash_Disk
usb.rev: 1.89
usb.serial_number: 1B3D42CB4E7400D4
```

Επίβλεψη Docker

Η επίβλεψη του Docker πραγματοποιείται σε 2 επίπεδα. Το πρώτο επίπεδο αφορά τον host που λειτουργεί ως Docker server και το δεύτερο επίπεδο αφορά επίβλεψη σε επίπεδο container.

Επίβλεψη Docker Server

Η επίβλεψη του Docker server έχει σκοπό τη δημιουργία ειδοποιήσεων για κάθε αλλαγή που αφορά την κατάσταση των Docker containers, όπως start, stop, pause, unpause, κτλ. Η ανάλυση των γεγονότων επιτυγχάνεται μέσω του Wazuh agent, ή Wazuh server, που είναι εγκατεστημένος στον host του Docker server, ο οποίος συλλέγει και αναλύει τα γεγονότα.

Είναι απαραίτητη η εγκατάσταση της “Python” σε έκδοση 2.7, ή νεότερη, και της Python βιβλιοθήκης “Docker”, ωστε να είναι ομαλή η λειτουργία του wodle στον Docker server.
Η χειροκίνητη εγκατάσταση των παραπάνω είναι απαραίτητη μόνο αν ο Wazuh manager είναι παλαιότερος της έκδοσης 3.9.0.

Σε Wazuh agent ή Wazuh manager παλαιότερης έκδοσης απο 3.9.0 ακολουθώνται τα παρακάτω βήματα:

```
sudo apt install python2
sudo pip2 install docker
```

Στο αρχείο ρυθμίσεων /var/ossec/etc/ossec.conf του Docker server προστίθεται το παρακάτω:

```
<wodle name="docker-listener">
      <interval>3m</interval>
      <attempts>5</attempts>
      <run_on_start>yes</run_on_start>
      <disabled>no</disabled>
</wodle>
```

Και τέλος, γίνεται επανεκκίνηση του Wazuh agent για να γίνει εφαρμογή των νέων ρυθμίσεων.

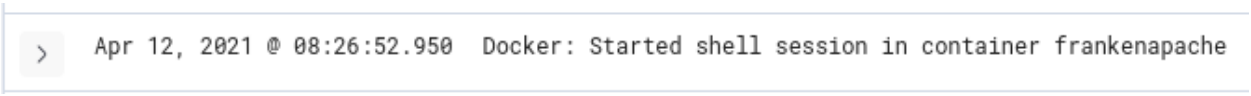
```
      # sudo systemctl restart wazuh-agent
```

Σε περίπτωση δυσλειτουργίας, περισσότερες πληροφορίες βρίσκονται στο log file του Wazuh agent στο παρακάτω directory

```
      /var/ossec/logs/ossec.log
```

Κάνοντας μία δοκιμή στον Docker server, ενα docker container ενός apache server δέχεται τις εντολές start και pause. Στο Dashboard του Wazuh manager εμφανίζονται τα παρακατω alerts:

Επίσης, με την δημιουργία διεργασίας shell στο container δημιουργείται το παρακάτω alert.



Επίβλεψη Docker Container

Έχοντας πρόσβαση σε logs και περιστατικά ασφαλείας στον Docker host, επεκτείνεται η λειτουργικότητα και η ορατότητα στα Docker containers. Για τον λόγο αυτό, είναι απαραίτητη η πρόσβαση στα logs που παράγουν. Ο τρόπος που είναι δομημένη η λειτουργία των Docker containers καθιστά την διατήρηση των αρχείων από ένα στιγμιότυπο στο επόμενο αδύνατη χωρίς να έχει εφαρμοστεί κάποια επιπρόσθετη ρύθμιση εκ των προτέρων. Στόχος είναι η επίλυση του παραπάνω προβλήματος και η διατήρηση των logs του service που εκτελείται ακόμα και αν γίνει διαγραφή και επανεκκίνηση του Docker container.

Δίνεται το παρακάτω σενάριο:
Δημιουργείται ένας nginx web server σε Docker container. Είναι απαραίτητη η καταγραφή, παρακολούθηση και ανάλυση των logs που δημιουργεί κατα τη λειτουργία του. Για την επίβλεψη του container χρησιμοποιείται ο Wazuh agent, ο οποίος είναι εγκατεστημένος στον Docker server και οχι σε καθε container.

Για τη δημιουργία του container, με βάση το επίσημο image του nginx, εκτελείται η παρακάτω εντολή:

```
docker run -it --rm -d -p 8080:80 --name web nginx
```

Το option --name web θέτει το όνομα του container σε “web” για διευκόλυνση της εκτέλεσης των επόμενων εντολών. Με το option -p αντιστοιχίζεται το port 80 του container με το port 8080 του host. Δοκιμάζοντας τοπικά στον browser του host τη διεύθυνση http://localhost:8080/ εμφανίζεται η παρακάτω σελίδα.

Παράλληλα, με την είσοδο στη σελίδα καταγράφεται στο access.log του nginx το request που έστειλε ο browser.

Σε περίπτωση που για κάποιο λόγο χρειαστεί να σταματήσει το container ή να γίνει επανεκκίνηση του service, τα logs αυτα δεν είναι πλέον διαθέσιμα. Τα logs που θα περιέχει το αρχείο, μετά την επανεκκίνηση του container, ανήκουν στο νέο container. Αρα, τα logs αυτα δεν μπορούν να χρησιμοποιηθούν ως πηγή πληροφορίας σε περίπτωση περιστατικού ασφαλείας.

Για την επίλυση του παραπάνω προβλήματος θα γίνει χρήση Docker volume με σκοπό την εξωτερική αποθήκευση των logs και την αποτροπή διαγραφής τους με την εκκίνηση νέου container.

Η εντολή εκκίνησης του container πραγματοποιείται ωστε τα logs να προωθούνται στο volume και να αποθηκεύονται.

```
docker run -it --rm -d
-p 8080:80
-v /var/log/serverlogs:/var/log/nginx
--name web nginx
```

Συγκεκριμένα, γίνεται χρήση λειτουργίας Docker volume η οποία ονομάζεται bind mount. Η επιλογή -v ορίζει με τη πρώτη παράμετρο το directory στο οποίο αποθηκεύονται τα logs στο container και με τη δεύτερη παράμετρο, το directory στο οποίο θα αποθηκεύονται τα logs στον host και θα είναι διαθέσιμα στον Wazuh agent.

Για την επαλήθευση της παραπάνω λειτουργίας, η σελίδα του nginx ανανεώνεται στον browser μερικές φορές ωστε να δημιουργηθούν logs στο αρχείο access.log και δίνεται εντολή διακοπής λειτουργίας και καταστροφής του Docker container. Έπειτα, γίνεται ξανα εκκίνηση του nginx container με τις ίδιες ρυθμίσεις.

```
docker stop web
```

```
docker run -it --rm -d
-p 8080:80
-v /var/log/serverlogs:/var/log/nginx
--name web nginx
```

Μέσω του browser στον host εκτελούνται GET requests προς τη σελίδα του nginx ώστε να δημιουργηθούν νέες καταγραφές στο αρχείο access.log.

```
172.17.0.1 - - [28/May/2021:09:45:14 +0000] "GET / HTTP/1.1" 304 0 "-" "Mozilla/5.0 (X11; Ubuntu;
172.17.0.1 - - [28/May/2021:09:45:14 +0000] "GET / HTTP/1.1" 304 0 "-" "Mozilla/5.0 (X11; Ubuntu;
172.17.0.1 - - [28/May/2021:09:45:14 +0000] "GET / HTTP/1.1" 304 0 "-" "Mozilla/5.0 (X11; Ubuntu;
172.17.0.1 - - [28/May/2021:10:53:34 +0000] "GET / HTTP/1.1" 304 0 "-" "Mozilla/5.0 (X11; Ubuntu;
172.17.0.1 - - [28/May/2021:10:54:39 +0000] "GET / HTTP/1.1" 304 0 "-" "Mozilla/5.0 (X11; Ubuntu;
172.17.0.1 - - [28/May/2021:10:54:39 +0000] "GET / HTTP/1.1" 304 0 "-" "Mozilla/5.0 (X11; Ubuntu;
172.17.0.1 - - [28/May/2021:10:54:39 +0000] "GET / HTTP/1.1" 304 0 "-" "Mozilla/5.0 (X11; Ubuntu;
172.17.0.1 - - [28/May/2021:10:54:39 +0000] "GET / HTTP/1.1" 304 0 "-" "Mozilla/5.0 (X11; Ubuntu;
172.17.0.1 - - [28/May/2021:10:54:39 +0000] "GET / HTTP/1.1" 304 0 "-" "Mozilla/5.0 (X11; Ubuntu;
```

Στη παραπάνω εικόνα φαίνονται logs του πρώτου container πριν τις 10.00 και logs του δεύτερου μετά τις 10.00. Συνεπώς, τα logs απο προηγούμενα containers δεν διαγράφονται όταν για κάποιο λόγο το container απο το οποίο δημιουργήθηκαν επανεκκινηθεί ή διαγραφεί και είναι διαθέσιμα στον Wazuh agent του Docker host.

Στις ρυθμίσεις του Docker host προστίθεται η τοποθεσία των νέων log αρχείων ωστε να γίνουν διαθέσιμα προς ανάγνωση στον Wazuh agent.

```
sudo vim /var/ossec/etc/ossec.conf
```

```
  <!-- Docker container logs -->
  <localfile>
      <location>/var/log/nginx/*.log</location>
      <log_format>json</log_format>
  </localfile>
```

Με την επανεκκίνηση του Wazuh agent service στον Docker host ελέγχονται οι πηγές των logs και επαληθεύεται η προσθήκη των νέων αρχείων.

Logs files

List of log files that will be analyzed

- /var/ossec/logs/active-respo...
- /var/log/auth.log
- /var/log/syslog
- /var/log/dpkg.log
- /var/log/kern.log
- /var/log/serverlogs/access.log
- /var/log/serverlogs/error.log

Drupal 8 Blue Whale Incident

Δόθηκε προς ανάλυση το περιστατικό “Blue Whale”, το οποίο αφορά την επίθεση προς Drupal 8 Server.

Ο Drupal Server είναι χωρισμένος σε container web server και container βάσης δεδομένων. Λόγω του version που χρησιμοποιείται στον web server, το σύστημα είναι ευπαθές σε επίθεση Remote Code Execution[10]. Η ευπάθεια αυτή επέτρεψε στον επιτιθέμενο να αποκτήσει πρόσβαση με shell στον web server με δικαιώματα του web χρήστη, χωρίς όμως να μπορεί να αποκτήσει πρόσβαση με περισσότερα δικαιώματα. Έπειτα για να αποκτήσει πρόσβαση στο root χρήστη εκμεταλλεύτηκε μια εσφαλμένη ρύθμιση η οποία επιτρέπει σε οποιονδήποτε έχει πρόσβαση μέσω του δικτύου να εκτελεί εντολές στον host ως ο docker χρήστης, χωρίς αυθεντικοποίηση. Έτσι, εγκατέστησε ένα νέο κακόβουλο ssh-server container το οποίο κάνει bind όλο το filesystem του host και έχει δικαίωμα να κάνει αλλαγές. Τέλος, ο επιτιθέμενος πρόσθεσε το RSA κλειδί του στον host και είχε πρόσβαση μέσω ssh στον host ως root. Έχοντας τη πρόσβαση, εγκατέστησε κακόβουλο λογισμικό για να διατηρήσει ένα σταθερό backdoor στον host.

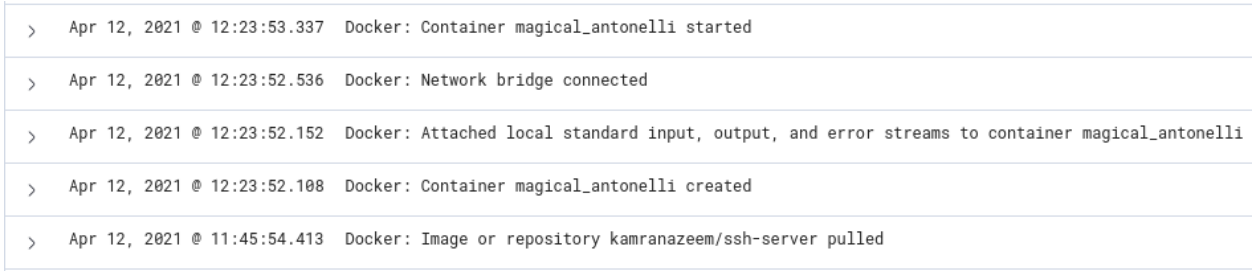
Με την εφαρμογή της επίβλεψης του docker server, μια αντίστοιχη συμπεριφορά επιτιθέμενου θα δημιουργήσει τα παρακάτω alerts στο Wazuh dashboard. Για τη δημιουργία ενός ssh-server εκτελέστηκαν οι εντολές

```
docker pull kamranazeem/ssh-server
```

για τη λήψη του νέου container και

```
docker run kamranazeem/ssh-server
```

για την εκκίνηση του container.



Συνεπώς, ο SOC Analyst που παρακολουθεί το wazuh dashboard είναι ενήμερος για το νέο κακόβουλο container και μπορεί να προβεί στις ανάλογες ενέργειες ωστε να αποτρέψει την πρόσβαση του κακόβουλου χρήστη, την εγκατάσταση του backdoor στον host και να διασφαλίσει το σύστημα.

[10] Drupal 8 REST RCE CVE-2019-6340

Δημιουργία Χρηστών και Διαχειριστών

Χρήστης Read-only

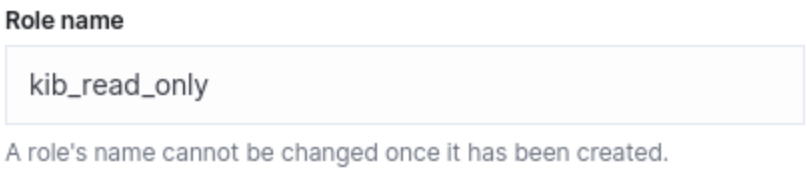
Για τη δημιουργία read-only χρήστη απαιτείται πρόσβαση μέσω browser στην IP διεύθυνση του Wazuh manager.

ElasticStack Configuration

Με το επιτυχημένο log in admin χρήστη, εμφανίζεται η εφαρμογή του Elastic. Στο μενού του Elastic επιλέγεται το “Stack Management”.

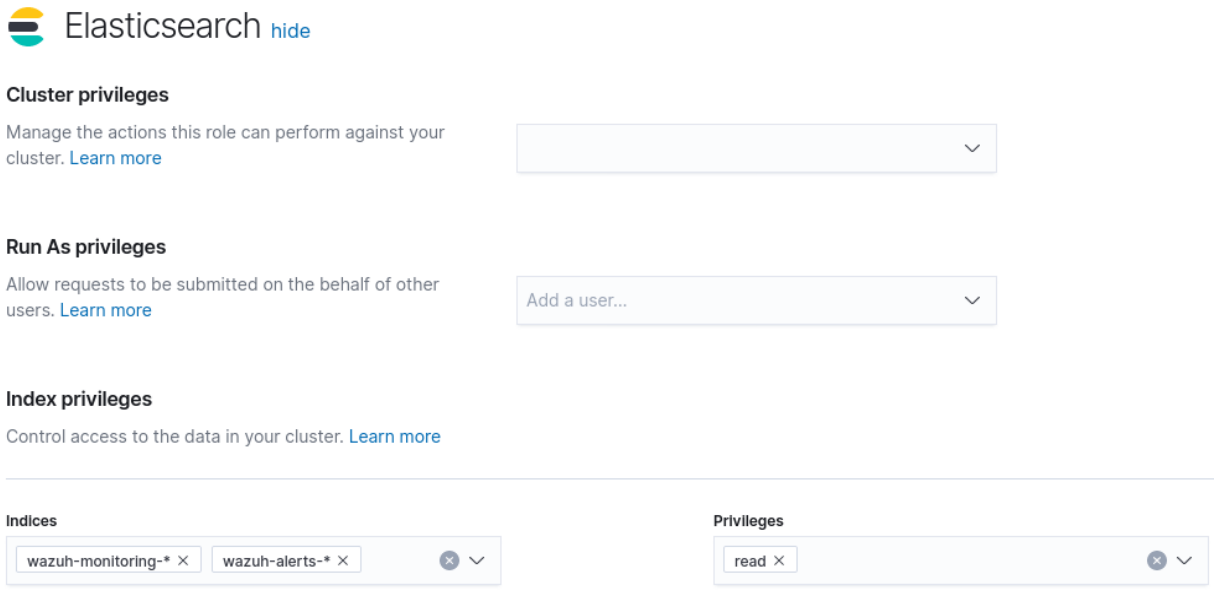
Στην κατηγορία “Security”, στην αριστερή στήλη, επιλέγεται η ενότητα “Roles”. Στην σελίδα που εμφανίζεται, επιλέγεται το “Create role”.

Συμπληρώνεται το όνομα του νέου ρόλου, στο παρόν παράδειγμα θα χρησιμοποιηθεί το “kib_read_only”:



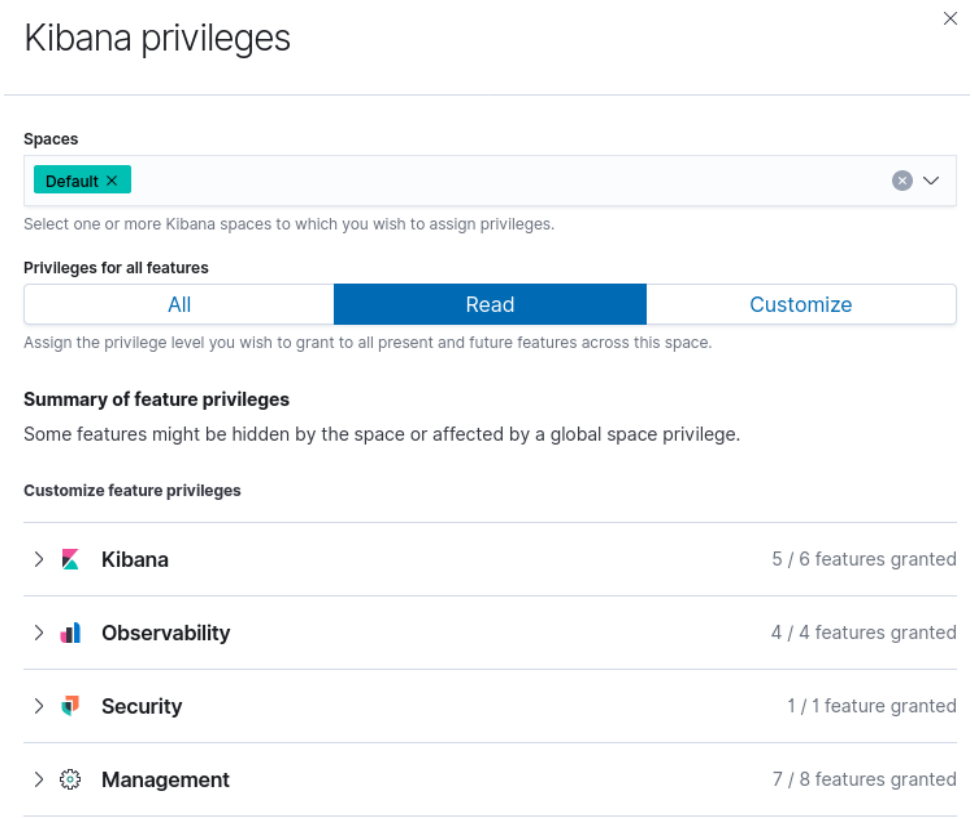
Στη συνέχεια, συμπληρώνονται τα δικαιώματα του νέου ρόλου οσον αφορά το Elasticsearch Cluster:

Indices: **wazuh-monitoring-***
wazuh-alerts-*
Privilleges: **read**



Και τα δικαιώματα του νέου ρόλου που αφορούν το Kibana:

Spaces: **Default**
Privilleges for all featues: **Read**



Για περισσότερες πληροφορίες σχετικά με τους ρόλους στο Kibana, επισυνάπτεται αναλυτικός πίνακας στο Documentation του Kibana.

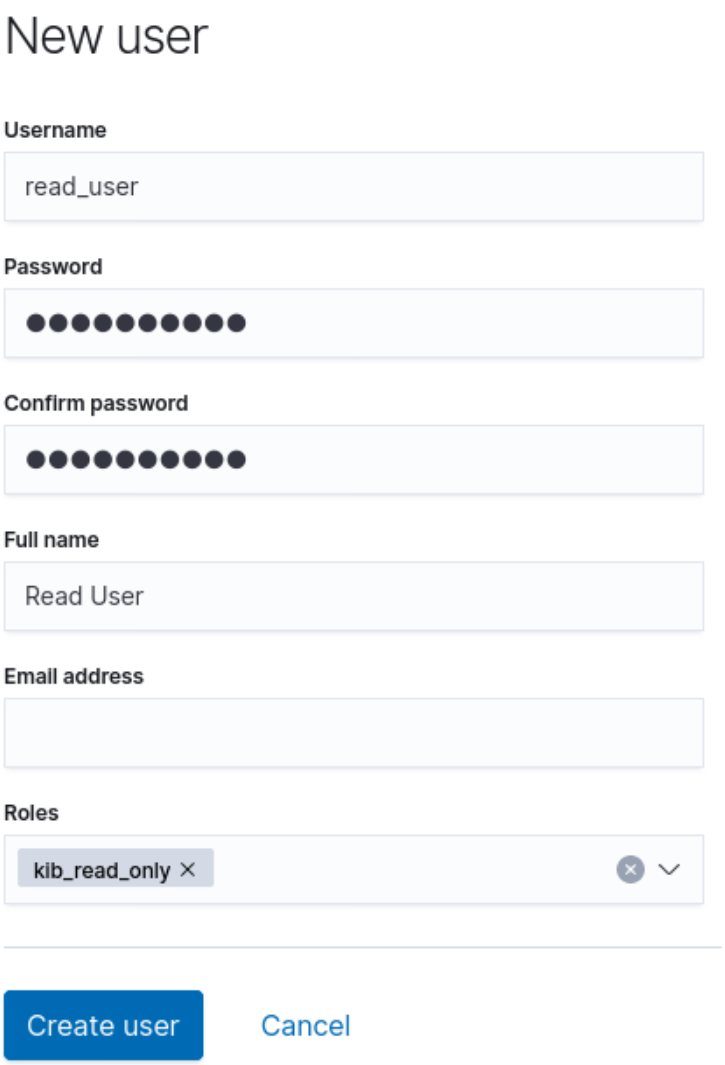
Στην αριστερή στήλη, στην κατηγορία “Security”, επιλέγεται η ενότητα “Users”. Στην σελίδα που εμφανίζεται, επιλέγεται το “Create user” και συμπληρώνονται υποχρεωτικά τα πεδία:

- Όνομα χρήστη, “read_user” για το παρόν παράδειγμα
- Κωδικός

Και προαιρετικά:

- Πλήρες όνομα χρήστη, “Read User” για το παρόν παράδειγμα
- e-mail

Στο τελευταίο πεδίο “Roles” ανατίθεται στον νέο χρήστη ο ρόλος που δημιουργήθηκε παραπάνω.

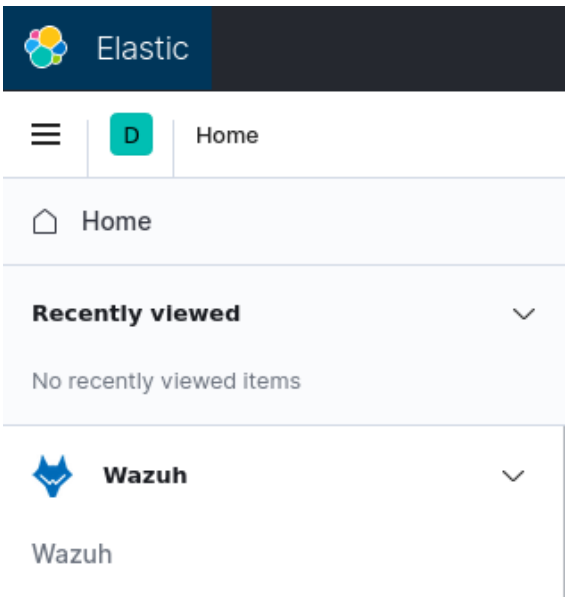


Υπάρχει επίσης η δυνατότητα ανάθεσης προκαθορισμένου ρόλου. Περισσότερες πληροφορίες για τους προκαθορισμένους ρόλους περιέχονται στο Documentation του Elastic.

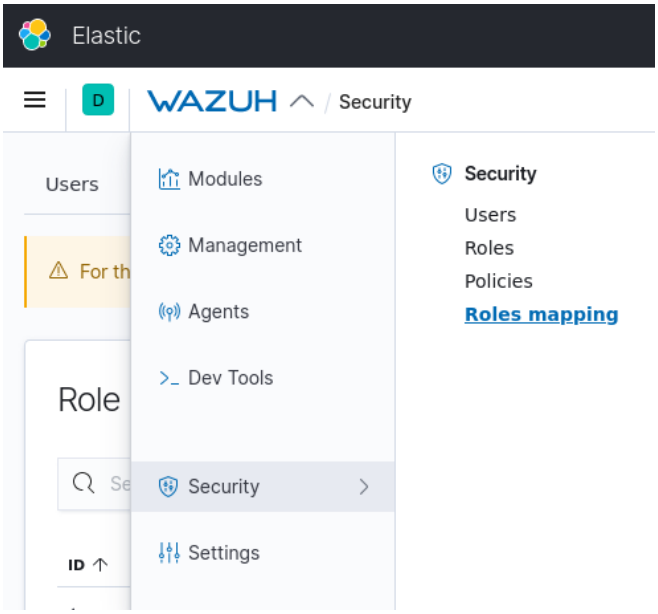
Wazuh Configuration

Στη συνέχεια, δημιουργείται νεός ρόλος στην εφαρμογή του Wazuh και αντιστοιχίζεται με τον χρήστη και τα δικαιώματα που μόλις δημιουργήθηκε στο Elastic.

Απο το μενού του Elastic επιλέγεται η εφαρμογή του Wazuh.



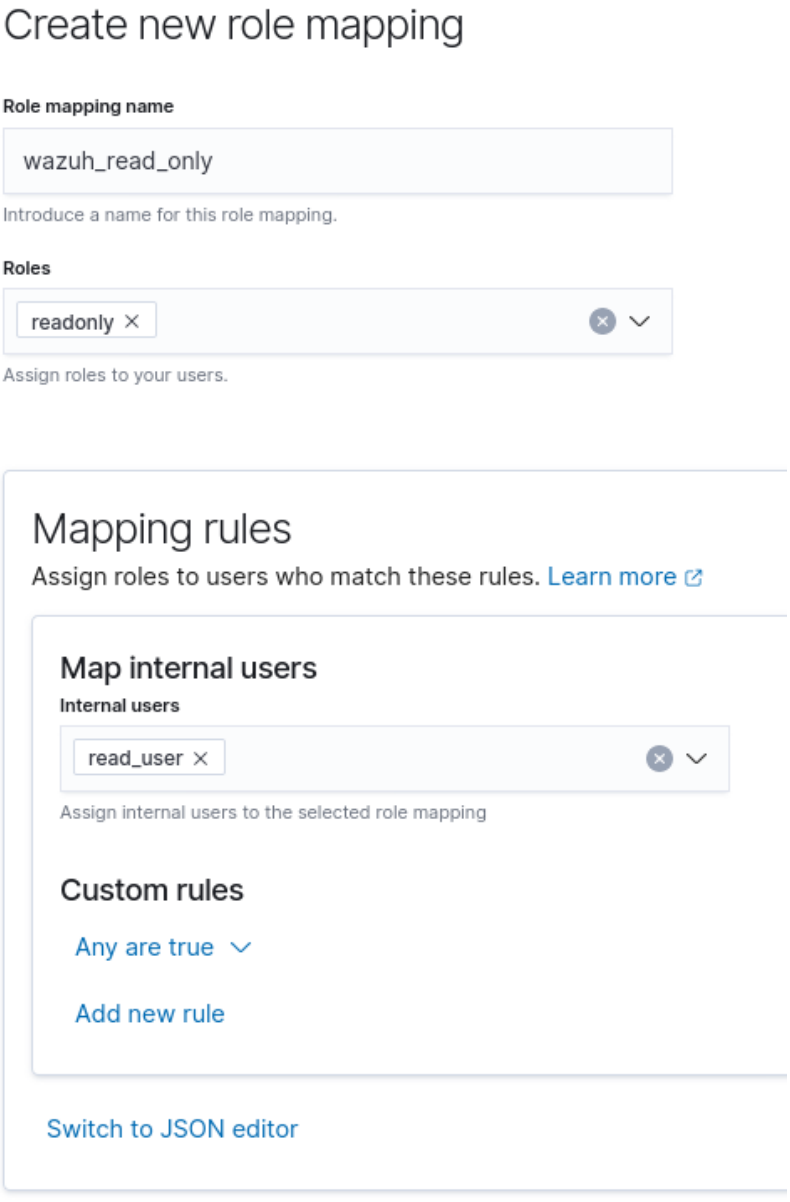
Στο μενού του Wazuh, επιλέγεται η ενότητα Security → Roles mapping



Πατώντας το “Create Role Mapping” εμφανίζεται η φόρμα για την αντιστοίχηση των ρόλων μεταξύ του Elastic και του Wazuh.

Συμπληρώνεται το όνομα του νέου ρόλου στο wazuh, τα δικαιώματα του στο Wazuh και ο αντίστοιχος χρήστης στο Elastic.

Role mapping name: “wazuh_read_only”, για το παρόν παράδειγμα
Roles: **readonly**
Internal users: “read_user”, για το παρόν παράδειγμα



Και τέλος επιλέγεται το “Save Role Mapping” για την αποθήκευση των αλλαγών. Περισσότερες πληροφορίες σχετικα με τα δικαιώματα περιλαμβάνονται στο Documentation του Wazuh.

Χρήστης Administrator

Elastic Configuration

Με το επιτυχημένο log in admin χρήστη, εμφανίζεται η εφαρμογή του Elastic. Στο μενού του Elastic επιλέγεται το “Stack Management”.

Στην κατηγορία “Security”, στην αριστερή στήλη, επιλέγεται η ενότητα “Users”. Στην σελίδα που εμφανίζεται, επιλέγεται το “Create user”.

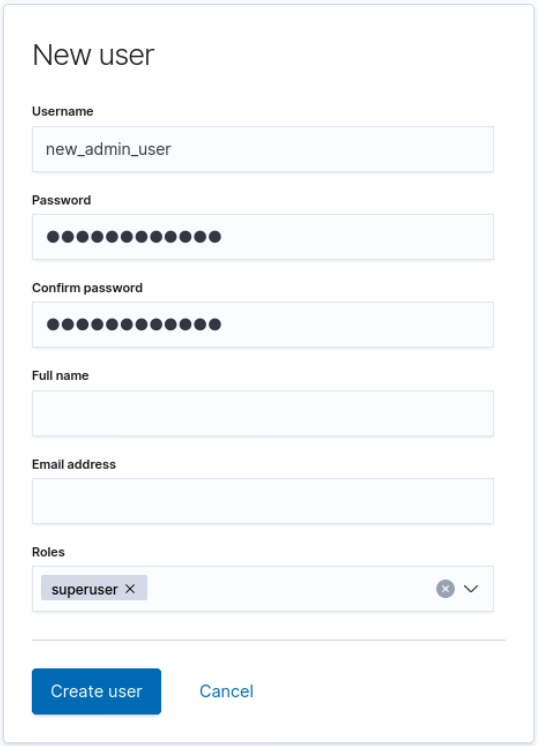
Συμπληρώνονται υποχρεωτικά τα πεδία:

- όνομα χρήστη, “new_admin_user” για το παρόν παράδειγμα
- κωδικός

Προαιρετικά τα πεδία:

- Πλήρες όνομα
- e-mail

Και στο πεδίο που αφορά τους ρόλους του χρήστη επιλέγεται το “superuser”.



References

1. Cyber Threat to Critical Infrastructure
2. 2016 Threat Landscape Report, ENISA
3. Cyber Threat Intelligence Understanding Fundamentals
4. Cyber Threat Intelligence from Honeypot Data Using Elasticsearch
5. Automatic Extraction of Indicators ofCompromise for Web Applications
6. Elastic.co
7. A Practical Model for Conducting Cyber Threat Hunting
8. Finding Cyber Threats with ATT&CK™-Based Analytics
9. MITRE ATT&CK Framework Introduction