ΑΛΕΞΑΝΔΡΕΙΟ Τ.Ε.Ι. ΘΕΣΣΑΛΟΝΙΚΗΣ
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Ευφυές σύστημα διαχείρισης πληροφοριών και περιστατικών ασφαλείας (SIEM)

Του φοιτητή

Σταυρίδη Κυριάκου

Αρ. Μητρώου: 144244

Επιβλέπων καθηγητής

Ηλιούδης Χρήστος

Θεσσαλονίκη 2019

ABSTRACT

Η διαρκώς αυξανόμενη πολυπλοκότητα των πληροφοριακών συστημάτων αυξάνει καθημερινά τα πιθανά κενά ασφαλείας τους. Όσο μεγαλύτερα γίνονται τα συστήματα που εξυπηρετούν τον άνθρωπο τόσο αυξάνονται και οι απαιτήσεις του ανθρώπου από αυτά, τόσο οι τεχνικές, όσο και οι λογικές. Λύση για την καθολική επίβλεψη και την όσο δυνατότερο ορθή λειτουργία αυτών των συστημάτων έχουν προσφέρουν τα Security Information and Event Management (SIEM) συστήματα. Οι ορίζοντες του πεδίου της ασφάλειας στην επιστήμη της πληροφορίας μεγάλωσαν πολύ με την εκκίνηση της ανάπτυξης των μηχανισμών SIEM. Έχουν σχεδιαστεί και αναπτύσσονται καθημερινά με στόχο την εύκολη διεκπεραίωση συνεργασίας με άλλους μηχανισμούς ώστε να μεγιστοποιηθεί το επίπεδο ασφάλειας ενός οργανισμού.

ΕΥΧΑΡΙΣΤΙΕΣ

Ευχαριστώ την οικογένεια μου η οποία πάντα με στήριζε κατά την διάρκεια της εκπόνησης της πτυχιακής μου εργασίας και φρόντιζε ώστε να έχω το κατάλληλο περιβάλλον για να αφοσιωθώ σε αυτή. Η κατανόηση που έδειξε με βοήθησε πολύ να συγκεντρωθώ στο έργο που είχα ξεκινήσει.

Ευχαριστώ τον Χρήστο Ηλιούδη ο οποίος με τις γνώσεις του ως καθηγητής και ερευνητής με βοήθησε να εκπονήσω ορθά την πτυχιακή μου εργασία. Η σημαντικότερη βοήθεια που μου έδωσε ήταν να μου μάθει να διαχειρίζομαι τον χρόνο που απαιτεί ένα μεγάλο ερευνητικό έργο και να μάθω κι εγώ ο ίδιος να ερευνώ και να διευρύνω τους ορίζοντες μου.

Περιεχόμενα

Ευρετήριο σχημάτων

Ευρετήριο πινάκων

Ευρετήριο εικόνων

ΚΕΦΑΛΑΙΟ 1

Εισαγωγή

1.1 Περιοχή έρευνας

Η τεχνολογική περιοχή της ασφάλειας έχει εξελιχθεί πάρα πολύ μέσα στα τελευταία 15 χρόνια. Ως αποτέλεσμα, αυτό φέρνει τη δυνατότητα δημιουργίας πολλών περισσοτέρων ειδών λογισμικού, συσκευών αλλά και πολλών καινούργιων και πρωτοπόρων σχεδίων επιχειρηματικής στρατηγικής που παλαιότερα δε θα ήταν καρποφόρα. Έχοντας όλα τα προαναφερόμενα υπόψιν, σταδιακά άρχισε να εμφανίζεται η ανάγκη για την ύπαρξη μιας λύσης που προσφέρει τη δυνατότητα της παρακολούθησης και της εποπτείας διαφόρων σημείων ενός επιχειρησιακού συστήματος ταυτόχρονα.

Η λύση που είχε τεθεί σε εφαρμογή αρχικά στην παραπάνω ανάγκη ήταν η ύπαρξη ανθρώπινου δυναμικού που ήταν και καταλλήλως εξειδικευμένο ώστε να υπάρχει συνεχής επιτήρηση και συντήρηση των συστημάτων αυτών. Αυτό φυσικά είχε πολύ μεγάλο κόστος, καθώς όπως προαναφέρθηκε αυτοί οι άνθρωποι έπρεπε να είναι ειδικοί στο κομμάτι που εποπτεύουν και να έχουν τις απαραίτητες γνώσεις ώστε όχι απλά να αντιληφθούν έγκαιρα την ύπαρξη του κινδύνου ή ανωμαλίας που είχε συμβεί, αλλά και να την περιορίσουν όσο το δυνατόν πιο πολύ.

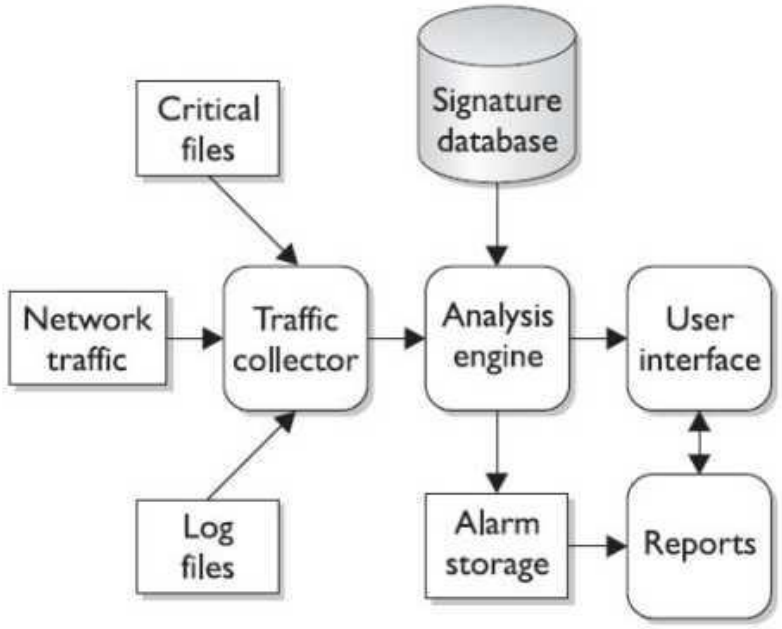
Για λίγο καιρό το ανθρώπινο δυναμικό αν και με μεγάλο κόστος κάλυπτε σωστά τις ανάγκες παρακολούθησης και εποπτείας ενός συστήματος, τις περισσότερες φορές μάλιστα, οι άνθρωποι υπεύθυνοι για το τελευταίο ήταν και αυτοί που συντηρούσαν το ίδιο το σύστημα καθημερινά. Με αυτόν τον τρόπο η εποπτεία αλλά και η συντήρηση γινόταν όλο ένα και καλύτερη καθώς ο συντηρητής ενός μηχανισμού τον επόπτευε κιόλας, μαθαίνοντας καλύτερα τον εκάστοτε μηχανισμό.

Όσο περνούσαν τα χρόνια και όσο τα επιχειρηματικά σχέδια και οι ανάγκες των οργανισμών γινόντουσαν πιο περίπλοκες και μεγαλύτερης έκτασης, οι μηχανισμοί λογισμικού και οι μηχανισμοί δικτύων που τις πραγματοποιούσαν επίσης απαιτούσαν μεγαλύτερη φροντίδα. Πολλοί πιστεύουν ότι όσο μεγαλύτερο είναι ένα πληροφορικό σύστημα, τόσο περισσότερος είναι και ο χρόνος που απαιτείται ώστε να δημιουργηθεί. Το παραπάνω αν και αληθεύει, σαν σκέψη δεν πρέπει να σταματάει εκεί. Η πραγματικότητα είναι ότι εκτός του χρόνου σκέψης, σχεδίασης και υλοποίησης ενός συστήματος, όποιο και να είναι αυτό, χρειάζεται και

ο απαραίτητος χρόνος αφού δημιουργηθεί και είναι σε λειτουργία ώστε να κρατιέται «εν ζωή» και να φέρνει εις πέρας τα καθήκοντα του χωρίς ανωμαλίες.

Σταδιακά, το κόστος της ύπαρξης του απαραίτητου ανθρώπινου δυναμικού για την κάλυψη των μέχρι τώρα αναφερομένων αναγκών σταμάτησε να είναι βιώσιμο. Έτσι η αγορά κινήθηκε προς την αυτοματοποίηση και άρχισε να κινείται γύρω από εργαλεία και λογισμικά που είναι υπεύθυνα για διάφορες ενέργειες εποπτείας και συντήρησης που μέχρι τώρα έκαναν άνθρωποι.

Η ύπαρξη πολλών συστατικών (components) σε ένα πληροφοριακό σύστημα έχει ως αποτέλεσμα να αυξάνεται ακόμα περισσότερο η πολυπλοκότητα του που συνεπάγεται την δυσκολία επίβλεψης του.



Σχήμα 1: Παράδειγμα ενός multi-component συστήματος

Αξιοσημείωτο είναι επίσης ότι για ένα πληροφοριακό σύστημα που εκτελεί σημαντικές λειτουργίες εικοσιτέσσερις ώρες το εικοσιτετράωρο, δεν μπορούν να υπάρχουν ανοχές ύπαρξης προβλημάτων σχετικά με την ακεραιότητα και την διαθεσιμότητα των υπηρεσιών του. Αυτό συνεπάγεται την απαίτηση να υπάρχει άμεση απόκριση σε σενάρια εμφάνισης κάποιας δυσλειτουργίας ή και απειλής. Προβλήματα και δυσλειτουργίες σε τέτοιας σημαντικότητας συστήματα μπορούν να αποτελέσουν την ρίζα μιας πολύ βλαβερής σειράς γεγονότων για τον οργανισμό

που αντιπροσωπεύουν, σε κάποιες περιπτώσεις μπορούν να αποτελέσουν και τον λόγο ανθρώπινων τραυματισμών ή χαμένων ζωών. Με βάση τα παραπάνω η επίβλεψη ενός συστήματος αποτελεί μια από τις σημαντικότερες ενέργειες που πρέπει ένας οργανισμός να συντελεί.

Αναμφισβήτητα, η επίβλεψη ενός πληροφοριακού συστήματος που εξυπηρετεί ορισμένους σκοπούς και ανάγκες είναι πολύ σημαντική. Μέσω της επίβλεψης ενός συστήματος οργανώνονται πιο εύκολα διαδικασίες όπως η συντήρηση του και η συλλογή δεδομένων για την καλύτερη περαιτέρω ανάπτυξη του. Το σημαντικότερο αποτέλεσμα που προκύπτει από την καλή επίβλεψη ενός συστήματος όμως είναι η ασφάλεια του. Η ύπαρξη κάποιου ανθρώπου ή ένα σύνολο ανθρώπων αφοσιωμένο στην επίβλεψη ενός συστήματος έχει τρομερή επίδραση στην προστασία του ίδιου από εξωτερικές απειλές όπως επίσης και στην καλύτερη κάλυψη των επιχειρηματικών απαιτήσεων ενός οργανισμού από ένα πληροφοριακό σύστημα. Είναι ευνόητο ότι κάτι τέτοιο ανεξάρτητα από τα θετικά χαρακτηριστικά που προσφέρει είναι και αρκετά κοστοβόρο για την επιχείρηση που χτίζει και υλοποιεί το έργο που επιβλέπεται. Συχνά η επίβλεψη και η διαχείριση ενός πληροφοριακού συστήματος γίνονται από το ίδιο άτομο ή την ίδια ομάδα ατόμων. Τα δύο προηγουμένως αναφερόμενα καθήκοντα δεν είναι σωστό να αναμοχλεύονται. Η διαχείριση ενός συστήματος αποτελεί πολύ σημαντικό και χρονοβόρο κομμάτι για τη λειτουργία του αλλά ταυτόχρονα και για την ανάπτυξη του. Η επίβλεψη από την άλλη μεριά είναι διαδικασία συνεχόμενη που ιδανικά είναι αφοσιωμένη στην καταγραφή γεγονότων, καταστάσεων, συμπεριφορών και αποτελεσμάτων που θα χρησιμοποιηθούν ώστε να ανυψώσουν την αποτελεσματικότητα του συστήματος με τον έναν ή τον άλλο τρόπο μετά από τις αλλαγές που θα προκύψουν ως συμπέρασμα. Την αξιοπιστία ενός συστήματος επηρεάζει άμεσα και το ανθρώπινο σφάλμα, το οποίο εμφανίζεται όλο και συχνότερα λόγω της σταθερής αύξησης της πολυπλοκότητας των συστημάτων όσο προοδεύει η τεχνολογία και οι απαιτήσεις του ανθρώπου από αυτή.

Το παραπάνω σενάριο του διαχωρισμού των καθηκόντων της επίβλεψης και της διαχείρισης ενός συστήματος, καθώς ουτοπικό, δημιουργεί ένα πρόβλημα, την ανάγκη ύπαρξης του απαραίτητου ανθρώπινου δυναμικού και την αφιέρωση των κατάλληλων εργατοωρών από αυτό. Η όσο το δυνατό αυτοματοποίηση της επίβλεψης και διαχείρισης ενός συστήματος θα επιφέρει σίγουρα τη μείωση εργατικού δυναμικού που χρειάζεται ένας οργανισμός για το κομμάτι της ασφάλειας, που συνεπάγεται τη μεγάλη μείωση των εξόδων του οργανισμού για την παραγωγή του έργου καθώς και για τη μελλοντική σωστή λειτουργία του.

Ως λύση στο παραπάνω έχουν δημιουργηθεί κατάλληλοι μηχανισμοί οι οποίοι είναι υπεύθυνοι και βοηθούν στην συνεχή επίβλεψη ενός πληροφοριακού συστήματος, χωρίς την ασταμάτητη ανθρώπινη συμβολή. Οι Μηχανισμοί αυτοί ονομάζονται μηχανισμοί διαχείρισης πληροφοριών και γεγονότων ασφαλείας (Security Information and Event Management - SIEM). Οι προγενέστεροι, λειτουργούν ελέγχοντας σε πραγματικό χρόνο τα γεγονότα που συμβαίνουν σε ένα σύστημα που παρακολουθούν. Καταγράφουν διάφορα γεγονότα που κρίνουν οι ίδιοι αξιοσημείωτα και ενημερώνουν κατάλληλα κάποιο άλλο υποσύστημα ή έναν άνθρωπο που είναι υπεύθυνος για την άμεση αντιμετώπιση τέτοιων ζητημάτων.

1.2 Στόχοι που τέθηκαν στη διπλωματική

Η διπλωματική εργασία στοχεύει στην πλήρη κατανόηση και μελέτη της δομής και της αρχιτεκτονικής των συστημάτων Security Information and Event Management (SIEM).

Επιπρόσθετα, στοχεύει στην έρευνα των υπερσύγχρονων προτύπων και τεχνολογιών που συμβάλλουν στην ανάπτυξη των προαναφερόμενων μηχανισμών και γενικότερα στον τομέα της ασφάλειας της πληροφορίας όπως και στην έρευνα των ιδανικών κριτηρίων σύγκρισης των μηχανισμών SIEM αλλά και την έμπρακτη σύγκριση των δημοφιλέστερων SIEM στην αγορά του σήμερα.

Τέλος, πέραν της ανάλυσης των συστημάτων SIEM, η διπλωματική στοχεύει και στην υλοποίηση ενός μηχανισμού SIEM από τα πρώτα του στάδια και την πετυχημένη δοκιμασία του μέσα από σενάρια ανώμαλων συμβάντων ασφάλειας.

1.3 Επιτεύγματα της διπλωματικής

Η διπλωματική εργασία εμβαθύνει σε μεγάλο βαθμό στην λειτουργία και στους μηχανισμούς ενός συστήματος Security Information and Event Management (SIEM). Αναλύει και περιγράφει όλες τις απαραίτητες δυνατότητες που οφείλει να έχει ο κάθε μηχανισμός και αναλύει επίσης δυνατότητες που θεωρούνται πιο εξελιγμένες βάσει του τρέχοντος επιπέδου τεχνολογικής ανάπτυξης γύρω από το οικοσύστημα της ασφάλειας της πληροφορίας.

Η διπλωματική πέραν της ανάλυσης των μηχανισμών SIEM, ερευνά και αναλύει τα γνωστότερα πρότυπα και τεχνολογίες που υπάρχουν σήμερα και χρησιμοποιούνται από πολλά προϊόντα και μηχανισμούς ασφάλειας.

Επιπρόσθετα, η διπλωματική εργασία ερευνά και καταγράφει μια δομημένη ροή ενεργειών ενός οργανισμού ώστε να γίνει ο προσδιορισμός των αναγκών και

των απαιτήσεων που έχει από τα εργαλεία ασφάλειας που χρησιμοποιεί ή θα αγοράσει. Έπειτα, θέτει κριτήρια που είναι ικανά να αξιολογήσουν ένα μηχανισμό SIEM με ακρίβεια και τα αναλύει. Μετά αυτών, κάνει μια συγκριτική αξιολόγηση των προϊόντων SIEM που είναι πολύ δημοφιλή στη σημερινή αγορά.

Τέλος, εκτελεί μια μεγάλη περίπτωση χρήσης (case study), όπου στήνει έναν μηχανισμό SIEM από την αρχή και τον αναπτύσσει ώστε να ανιχνεύσει δύο σημαντικά περιστατικά ασφαλείας.

1.4 Διάρθρωση της διπλωματικής

Το δεύτερο κεφάλαιο της διπλωματικής μελετά την δομή και την αρχιτεκτονική των μηχανισμών Security information and Event Management (SIEM). Το τρίτο κεφάλαιο εξετάζει μερικά από τα υπάρχον πρότυπα και τεχνολογίες που υπάρχουν και συμβάλουν στην καθημερινή ανάπτυξη του τομέα της ασφάλειας στην επιστήμη της πληροφορίας. Το τέταρτο κεφάλαιο περιγράφει τη διαδικασία του προσδιορισμού των αναγκών και απαιτήσεων ενός οργανισμού και συγκρίνει τα γνωστότερα προϊόντα SIEM που υπάρχουν στην αγορά σήμερα. Το πέμπτο κεφάλαιο περιλαμβάνει ένα case study σχετικά με τη σύνθεση ενός συστήματος SIEM από την αρχή έως και την ανίχνευση ανώμαλων συμβάντων. Στο έκτο κεφάλαιο αναφέρονται μελλοντικές επεκτάσεις που θα προσέδιδαν ακόμη περισσότερο στην αποδοτικότητα των μηχανισμών SIEM.

ΚΕΦΑΛΑΙΟ 2

Security Information and Event Management (SIEM)

ΕΙΣΑΓΩΓΗ

Τα τελευταία χρόνια στον τομέα της ασφάλειας την πληροφορίας έχουν αναπτυχθεί πολύ οι μηχανισμοί Security Information and Event Management (SIEM). Τα τελευταία, είναι ικανά να παρέχουν μια συνολική εποπτεία όλων των τεχνολογικών συστημάτων ενός οργανισμού και με τη βοήθεια τους μειώνονται σημαντικά οι εργατοώρες που αφιερώνονται στην ασφάλεια του οργανισμού από κυβερνοεπιθέσεις αλλά και στην συντήρηση του οργανισμού. Τα SIEM συστήματα συμβάλλουν επίσης πολύ και στις εργασίες συντήρησης των τεχνολογικών υποδομών του οργανισμού που εποπτεύουν.

2.1 Περιστατικά ασφάλειας

Πολλοί όταν ακούν περιστατικά ασφάλειας συνδέουν απευθείας τη σκέψη τους με περιστατικά κακόβουλης δραστηριότητας και διάφορες μορφές κυβερνοεπιθέσεων. Ωστόσο κάτι τέτοιο δεν είναι ακριβές.

Περιστατικό ασφαλείας μπορεί να θεωρηθεί οποιοδήποτε συμβάν που παίρνει μέρος και είναι ικανό να δημιουργήσει οποιασδήποτε μορφής ανωμαλία σε ένα σύστημα. Άσχετα από την προέλευση του και τις επιπτώσεις του ένα περιστατικό ασφαλείας μπορεί να κατηγοριοποιηθεί σε δύο γενικές κατηγορίες, τα περιστατικά από κακόβουλες ενέργειες και τα περιστατικά από λανθασμένες ενέργειες.

2.1.1 Περιστατικά από κακόβουλες ενέργειες

2.1.1.1 Περιστατικά με στόχο το κέρδος

Καθώς προχωρά και αναπτύσσεται η επιστήμη της τεχνολογίας και της πληροφορίας, παρουσιάζονται και νέοι τρόποι να κινηθεί η παγκόσμια οικονομία

μέσα από αυτήν. Δυστυχώς ή ευτυχώς ολοένα και περισσότερες επιχειρήσεις βασίζουν ένα κομμάτι της λειτουργίας τους στην τεχνολογία και στην επιστήμη της πληροφορικής. Αυτό ως αποτέλεσμα έχει να αυξάνονται με τον ίδιο ρυθμό και οι τρόποι να κάνει κάποιος διάφορες κακόβουλες ενέργειες με στόχο να έχει κάποιο οικονομικό κέρδος.

Ένα από τα μεγαλύτερα και πιο πρόσφατα περιστατικά εγκλήματος στον κυβερνοχώρο είναι το κρυπτογραφικό κακόβουλο λογισμικό (ransomware) WannaCry. Το WannaCry ήρθε στο παρασκήνιο τον Μάιο του 2017 και παραμένει μέχρι και σήμερα ένα από τα πιο επιτυχημένα στο είδος του. Όπως τα περισσότερα ransomware το WannaCry κρυπτογραφούσε τον δίσκο που το φιλοξενούσε και στη συνέχεια ζητούσε από το θύμα ένα ποσό χρημάτων σε μορφή κρυπτονομίσματος ώστε να αποκρυπτογραφηθούν τα δεδομένα του και να είναι χρησιμοποιήσιμα και πάλι. Το συνολικό κέρδος που είχαν οι δημιουργοί του υπολογίζεται να είναι κοντά στα 386,905$. Δε μοιάζουν πολλά, παρόλα αυτά η συνολική ζημία που επέφερε ο συγκεκριμένος ιός παγκοσμίως υπολογίζεται να είναι περίπου 4 δισεκατομμύρια, που είναι ένα εξωφρενικό ποσό.

2.1.1.2 Περιστατικά με στόχο την συναισθηματική ικανοποίηση

Δεν είναι λίγες οι φορές που ένας κακόβουλος χρήστης θα κυνηγήσει την συναισθηματική ικανοποίηση που προσφέρει η επιτυχής εισβολή και ζημιά σε ένα μεγάλο ή και μικρό σύστημα. Χωρίς να είναι ο κύριος στόχος το κέρδος αυτή τη φορά, πολύ συνηθισμένη είναι και η εμφάνιση κακόβουλων περιστατικών ασφαλείας που ο μόνος τους σκοπός είναι απλά να παρέχουν στον δράστη την ικανοποίηση μέσω της επιτυχίας. Πολλές φορές οι κυβερνοεπιθέσεις αυτού του είδους είναι οι πιο επικίνδυνες και με τις μεγαλύτερες συνέπειες καθώς ο δράστης μπορεί να φτάσει πολύ πιο μακριά και να τολμήσει να σχεδιάσει πολύ πιο καταστροφικές επιθέσεις. Εφόσον τα χρήματα πλέον δεν είναι παράγοντας ο επιτιθέμενος έχει μικρότερο κίνδυνο να ανακαλυφθεί.

Μία από τις μεγαλύτερες επιτυχημένες κυβερνοεπιθέσεις αυτού του είδους είναι η εισβολή στα δεδομένα του Γραφείου του ανθρώπινου δυναμικού των Ηνωμένων Πολιτειών (Office of Personnel Management), ή αλλιώς το ευρέος

γνωστό OPM Hack. Διέρρευσαν τα στοιχεία περισσότερων από τέσσερα εκατομμύρια ανθρώπων στο κοινό, κάνοντας το OPM Hack ένα από τα μεγαλύτερα επιτυχημένα σχέδια κυβερνοεγκλήματος δημοσιοποίησης πληροφοριών (dox hacks) που έγιναν ποτέ. Η συνολική οικονομική ζημία που δημιουργήθηκε υπολογίζεται περίπου στα 21,5$ εκατομμύρια.

2.1.2 Περιστατικά από λανθασμένες ενέργειες

Είναι λογικό ότι κατά την κατασκευή και δημιουργία κάποιου μηχανισμού που είναι κομμάτι ενός ολοκληρωμένου υπολογιστικού συστήματος μπορεί να υπάρξουν διαφόρων ειδών λάθη που θα επηρεάσουν άμεσα ή και μελλοντικά τη λειτουργία του. Αυτά τα λάθη δημιουργούνται συνήθως από τους μηχανικούς που κατασκεύασαν το πληροφοριακό σύστημα ή και από τους ανθρώπους που είναι υπεύθυνοι να συντηρούν τη λειτουργία του κατά τη διάρκεια του χρόνου.

Αυτά τα περιστατικά είναι ιδιαίτερα δύσκολο να εντοπιστούν και να διορθωθούν καθώς δεν έχουν πάντα εμφανή αίτια ύπαρξης. Τα συγκεκριμένα περιστατικά μπορεί να δημιουργηθούν από λάθη α) κατά τη σχεδίαση της αρχιτεκτονικής του συστήματος, β) κατά την υλοποίηση και δημιουργία των κομματιών που συντελούν το σύστημα, γ) κατά την εγκαθίδρυση και προετοιμασία του φυσικού ή και του νοητού χώρου, δ) κατά τη συντήρηση του συστήματος, ε) κατά την αλλαγή κομματιών / αναβάθμιση όλου του συστήματος ή κάποιον κομματιών του και τέλος ζ) από πιθανή μελλοντικά υπερβολική αύξηση φόρτου εργασίας στο σύστημα.

2.2 Security Event Management (SEM)

Ένα πολύ βασικό συστατικό της επίβλεψης και της παρακολούθησης ενός συστήματος είναι το να υπάρχει η ικανότητα να γίνεται και σε πραγματικό χρόνο. Τα Security Event Management (SEM) εργαλεία επιτυγχάνουν αυτόν τον σκοπό.

Τα συστήματα SEM είναι υπεύθυνα να παρακολουθούν διάφορα άλλα συστήματα συνήθως χρησιμοποιώντας agents και να ελέγχουν τα γεγονότα που συμβαίνουν σε αυτά. Έπειτα ένας agent SEM είναι ικανός να μειώσει αρκετά την πληροφορία αυτή που κατατάσσεται χρήσιμη για τον SEM manager ώστε να στείλει στον προηγούμενο μόνο τα απαραίτητα ή θεμιτά συμβάντα που αφορούν κάποια

συγκεκριμένη υπηρεσία ή λειτουργία του συστήματος που επιβλέπεται. Τα συμβάντα και τα δεδομένα που αντλούν οι agents από τα συστήματα που επιβλέπουν έχουν την μορφή των logs, τα οποία θα δούμε αναλυτικά αργότερα.

Τα δυνατά σημεία των Security Event Management συστημάτων είναι τα εξής [2]:

- Μπορούν να διαχειριστούν πολύ μεγάλους αριθμούς δεδομένων σε πραγματικό χρόνο
- Αν και όχι μεγάλες, προσφέρουν σε πραγματικό χρόνο δυνατότητες συσχετισμού διαφόρων συμβάντων μεταξύ τους με την πιθανότητα να προκύψει μια καινούρια ένδειξη ανωμαλίας.
- Παρέχουν μηχανισμούς αυτόματης ειδοποίησης στο κατάλληλο άτομο ή και σε περισσότερους ανθρώπους αν κρίνεται απαραίτητο.
- Όταν βρουν ένα συμβάν που αντιπροσωπεύει μια ανώμαλη συμπεριφορά, παράγουν ένα συμβάν σε κανονικοποιημένη μορφή που ακολουθεί το IDMEF [19] (Intrusion Detection Message Exchange Format).

Έχοντας υπόψιν τα παραπάνω μπορούμε να συμπεράνουμε ότι οι Security Event Management (SEM) μηχανισμοί βοηθούν τον ή τους διαχειριστές ενός μεγάλου συστήματος να έχουν άμεση ενημέρωση σε περίπτωση που συμβεί κάτι που μπορεί να είναι ανωμαλία με βάση τα δικά τους κριτήρια.

Ωστόσο αποδεικνύεται ότι το να είναι έτοιμο ένα σύστημα ενός οργανισμού ώστε να χρησιμοποιηθεί σωστά ένας μηχανισμός Security Event Management (SEM) και να το επιβλέψει με τα επιθυμητά αποτελέσματα είναι αρκετά περίπλοκο. Όσο μεγαλώνει το μέγεθος ενός οργανισμού και των συστημάτων του τόσο αυξάνεται και η πολυπλοκότητα του συστήματος που χρειάζεται για να γίνει η σωστή επίβλεψη του. Τα πιο μεγάλα εμπόδια που αντιμετωπίζονται κατά τις εγκαταστάσεις και την γενικότερη χρησιμοποίηση διαφόρων μηχανισμών Security Event Management (SEM) είναι τα εξής [6]:

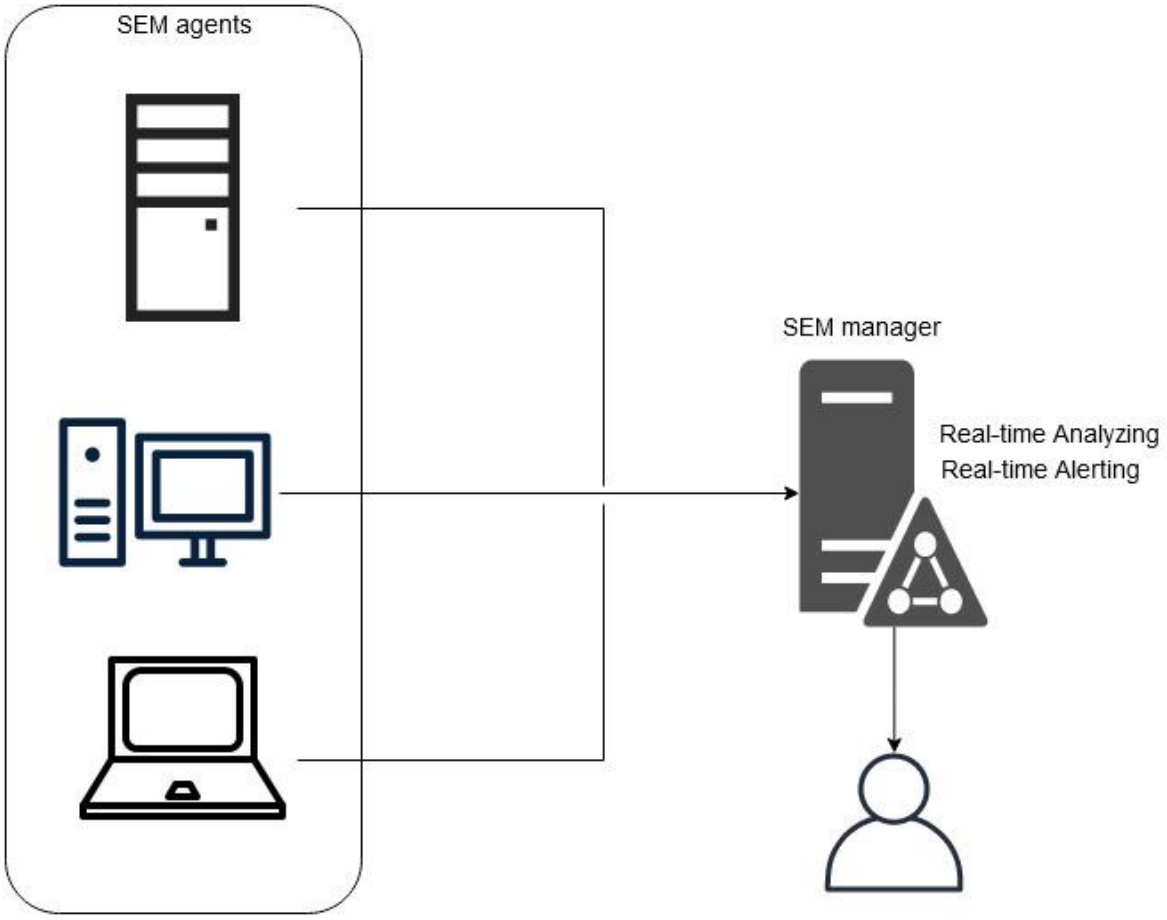
- Πολλά false positive (ψευδώς αληθινά) συμβάντα:
  - ❖ Πολλές φορές ο συσχετισμός των διαφόρων γεγονότων μεταξύ τους είναι τόσο περίπλοκος που κατά τον σχεδιασμό των κανόνων του SEM θα υπάρχουν διπλά καλυπτόμενα συμβάντα ή και συσχετισμοί που δεν στέκουν λογικά. Έτσι δημιουργούνται συχνά γεγονότα τα οποία στην πραγματικότητα δεν αποτελούν κάποια ανώμαλη συμπεριφορά που έπρεπε να ανιχνεύσει ο μηχανισμός.

- Περιορισμένη πληροφορία:

- ❖ Πολλές φορές η πληροφορία που παράγουν οι μηχανισμοί Security Event Management (SEM) για ένα συμβάν που μετά από την επεξεργασία του σημαδεύτηκε ως ανωμαλία, δεν είναι αρκετή. Αυτό συμβαίνει διότι όπως προαναφέρθηκε η πληροφορία για τα γεγονότα που παίρνουν μέρος σε ένα σύστημα ανακτάται μέσω των logs. Τα Logs βασίζονται εξολοκλήρου στην υπηρεσία που τα τυπώνει, οπότε η κακή διαχείριση και καταγραφή του εαυτού κάποιας υπηρεσίας συνεπάγεται στην ελλιπή πληροφορία προς επεξεργασία από κάποιον μηχανισμό παρακολούθησης.

- Η δημιουργία κανόνων είναι χειροκίνητη:
  - ❖ Το ζητούμενο αυτό απασχολεί χρόνια πολλούς οργανισμούς ασφαλείας και ιδιαίτερα αυτούς που χτίζουν έναν μηχανισμό SEM. Οι κανόνες που είναι υπεύθυνοι για την ανίχνευση των ανωμαλιών σε ένα σύστημα μπορούν να δημιουργηθούν μόνο χειροκίνητα από κάποιους μηχανικούς που πρέπει μάλιστα να έχουν ιδιαίτερα καλή γνώση του συστήματος SEM που χειρίζονται. Αυτό έχει ως αποτέλεσμα η διαδικασία της εγκατάστασης ενός SEM μηχανισμού και η προσαρμογή του για να καλύπτει έναν οργανισμό με επιτυχία χρειάζεται πολύ χρόνο και πολύ μελέτη από εξιδεικευμένους τεχνικούς. Επομένως, η δημιουργία κανόνων χειροκίνητα είναι ένα πρόβλημα που έχει πολύ μεγάλο κόστος.

Περισσότερα προβλήματα, εμπόδια και λύσεις θα αναλύσουμε και στη συνέχεια με βάση τα Security Incident and Event Management συστήματα.



Σχήμα 2: Απλουστευμένη απεικόνιση λειτουργίας SEM

2.3 Security Information Management (SIM)

Ο σκοπός των συστημάτων Security Information Management (SIM) είναι να συλλέγουν ιστορικά δεδομένα από συστήματα και να τα διατηρούν για αρκετό χρόνο, συνήθως έξι ή δώδεκα ή σε σπάνιες περιπτώσεις έως και είκοσι τέσσερις μήνες. Τα δεδομένα για τα συμβάντα συνήθως συλλέγονται σε κεντρικές αποθήκες δεδομένων ώστε να είναι εύκολη η πρόσβαση τους από διάφορους άλλους μηχανισμούς που θα κάνουν μαζικές αναλύσεις σε μη πραγματικό χρόνο. [7]

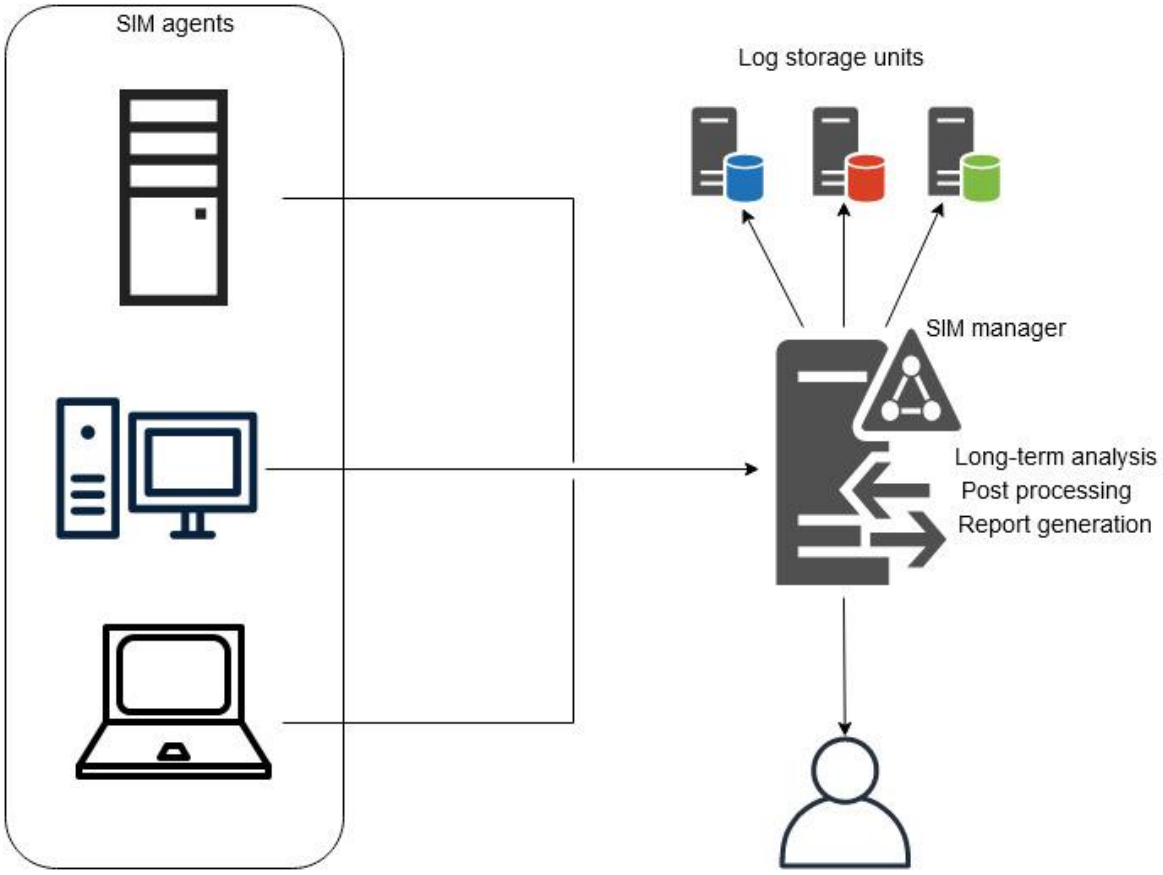
Με βάση την έρευνα και τεκμηρίωση του οργανισμού NIST [8], η δομή της μαζικής αποθήκευσης και διαχείρισης των δεδομένων σε μορφή logs αποτελείται από τρία στάδια:

- Συνήθως το πρώτο στάδιο είναι η υποχρέωση και καθήκον του επιβλέποντος συστήματος να παραχωρεί τα δεδομένα του στους ειδικούς εξυπηρετητές (servers). Η διαδικασία αυτή μπορεί να γίνεται και σε πραγματικό χρόνο αλλά και ασύγχρονα.

- Το δεύτερο στάδιο της δομής αποθήκευσης δεδομένων των συστημάτων SIM είναι η αποδοχή και αποθήκευση των δεδομένων που έρχονται από τα επιβλεπόμενα συστήματα από τους εξυπηρετητές (servers). Τώρα η διαδικασία αρχίζει να γίνεται τεχνολογικά πιο σύνθετη καθώς σε αυτό το σημείο οι servers είναι υπεύθυνοι να κανονικοποιήσουν την πληροφορία που λαμβάνουν σε μορφή logs από τα άλλα συστήματα. Η πληροφορία που φτάνει στους log storage servers όντως βρίσκεται σε μορφή logs αλλά τα logs τα ίδια έχουν πολλές διαφορετικές μορφές (syslog, NTsyslog, Apache log, WinEvent log και άλλα πολλά), αναλυτικότερα θα εξετάσουμε τα logs και τις μορφές τους όταν αναλύσουμε τους αποκωδικοποιητές (decoders). Οι log storage servers καλούνται να γενικεύσουν τα παραπάνω δημιουργώντας έτσι μιας γενικής μορφής log που συνήθως είναι το syslog που είναι και το πιο συνηθισμένο log format που χρησιμοποιούν οι εμπορικές υπηρεσίες σήμερα.

  Τέλος πρέπει να σημειωθεί ότι οι log storage servers έχουν τη δυνατότητα να κάνουν μια αίτηση προς τα συστήματα που εξυπηρετούν ώστε να λάβουν πολλά δεδομένα μαζί κατά απαίτηση (on demand) ώστε να γεμίσουν χρονικά κενά που ενδεχομένως να έχουν ή να διεξάγουν μια περαιτέρω δευτερεύουσα έρευνα μελλοντικά.

- Το τρίτο και τελευταίο στάδιο της λειτουργικότητας ενός SIM είναι το κομμάτι της παραγωγής αναφορών και της οπτικοποίησης των αποτελεσμάτων των ερευνών που διεξάγουν. Οποιαδήποτε λειτουργικότητα που αφορά την μετάδοση πληροφορίας προς τον χρήστη ανήκει στο τρίτο επίπεδο.



Σχήμα 3: Απλουστευμένη απεικόνιση λειτουργίας SIM

2.4 Security Information and Event Management (SIEM)

Η ένωση των μηχανισμών Security Event Management (SEM) και Security Information Management (SIM) σε ένα κυρίαρχο, πολυμήχανο σύστημα, αποτελεί το βήμα για την ύπαρξη ενός Security Information and Event Management System [2].

Τα τελευταία χρόνια οι μηχανισμοί SIEM έχουν κυριαρχήσει στο είδος τους και πολλοί πιστεύουν ότι είναι το μέλλον στον κεντρικό έλεγχο πολλών συστημάτων από μεγάλους οργανισμούς [10].

Τα συστήματα SIEM πλέον θεωρούνται από τα σημαντικότερα κομμάτια της διαχείρισης των δικτυακών υπηρεσιών, των λογισμικών υπηρεσιών αλλά και των φυσικών εγκαταστάσεων ενός μεγάλου οργανισμού. Τα συστήματα αυτά

επιτρέπουν την αποσαφήνιση και την εξέταση οποιουδήποτε γεγονότος που συμβαίνει ανά πάσα ώρα και στιγμή σε πολλά συστήματα ταυτόχρονα [11]. Αν αναγκαζόμασταν να ορίσουμε ένα σύστημα Security Information and Event Management (SIEM) θα λέγαμε ότι είναι ένα σύστημα ικανό να συλλέξει, αναλύσει και να παρουσιάσει χρήσιμες πληροφορίες από διάφορες συσκευές ή υπηρεσίες ασχέτως αν είναι δικτυακές ή όχι [3].

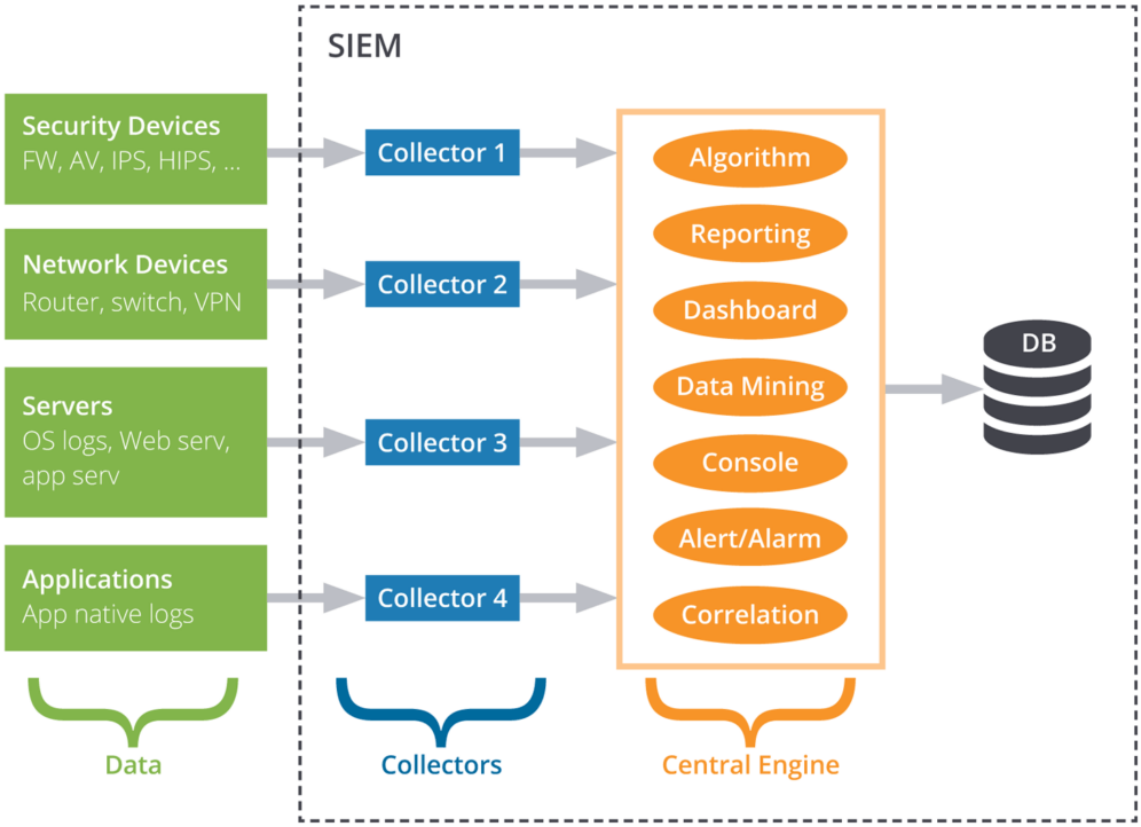
Έχοντας τις δυνατότητες ενός συστήματος SIM αλλά και ενός συστήματος SEM τα SIEM οφείλουν να προσφέρουν τις εξής δυνατότητες [11]:

- Συλλογή δεδομένων:
  - ❖ Η ζωτική λειτουργία ενός συστήματος SIEM είναι να συλλέγει τα απαραίτητα δεδομένα ώστε να εποπτεύει και σε δεύτερο χρόνο να προστατεύει τα συστήματα που εξυπηρετεί. Χωρίς να υπάρχει σωστή και έμπιστη συλλογή δεδομένων δεν μπορεί κανένας μηχανισμός να κάνει σωστά τη δουλειά του, ασχέτως το ποια είναι αυτή. Στην περίπτωση των μηχανισμών SIEM όμως η σωστή και πλήρης συλλογή δεδομένων είναι μια αρκετά περίπλοκη ανάγκη.

    Αρχίζοντας από τη δικτυακή άποψη του θέματος μας, είναι γνωστό ότι κάθε οργανισμός διαφέρει. Ένα Security Information and Event Management σύστημα θα πρέπει πάντα να είναι σε θέση να λαμβάνει ότι του στέλνουν οι agents του. Αυτό σημαίνει ότι πρέπει να είναι δικτυακά ορατό και προσβάσιμο από οποιοδήποτε σημείο του οργανισμού που καλύπτει. Για να γίνει αυτό θα πρέπει να είναι σωστά ρυθμισμένοι πολλοί παράγοντες του εκάστοτε οργανισμού όπως τα firewalls, τα routers, τα antiviruses και γενικά οποιαδήποτε υπηρεσία μπορεί να παίξει ρόλο στο τι κινείται και τι όχι σε ένα δίκτυο πληροφοριών.

    Επιπρόσθετα, εκτός από διαφορετική δικτυακή δομή, κάθε οργανισμός έχει και διαφορετικές υπηρεσίες, λογισμικά και λειτουργικά συστήματα στην καθημερινή λειτουργία του. Ένας μηχανισμός SIEM πρέπει να είναι σε θέση να συλλέξει επιτυχώς δεδομένα από διάφορες πηγές, ασχέτως αν αυτή η πηγή είναι ένα router με FreeBSD λειτουργικό σύστημα, ή είναι ένας desktop υπολογιστής με λειτουργικό σύστημα Windows. Ο κανόνας λέει ότι όποια συσκευή και λογισμικό μπορεί να παράγει logs και να προωθεί

δεδομένα στο δίκτυο θα πρέπει να υποστηρίζεται πλήρως από ένα σύστημα SIEM.

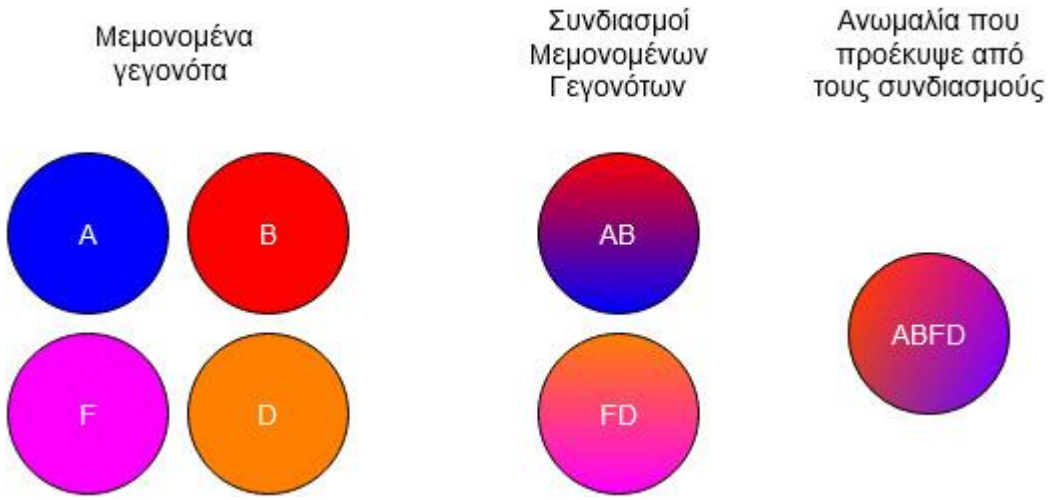


Σχήμα 4: Συλλογή δεδομένων από πολλές διαφορετικές πηγές

- Συσχέτιση διαφόρων συμβάντων μεταξύ τους:
  - ❖ Για να χαρακτηρίζεται δικαίως ένα σύστημα ως ένα Security Information and Event Management System σίγουρα θα πρέπει να είναι σε θέση να συσχετίσει γεγονότα μεταξύ τους που συνέβησαν σε διαφορετικές χρονικές στιγμές. Ένα γεγονός από μόνο του μπορεί να μην είναι αρκετό ώστε να φανεί μια ανωμαλία σε ένα σύστημα. Παίρνοντας πολλά από αυτά όμως και συσχετίζοντας το ένα με το άλλο μπορεί να ανακαλυφθούν ανωμαλίες όλων των ειδών.

    Χαρακτηριστικό παράδειγμα μιας ανωμαλίας που μπορεί να προκύψει μόνο από την συσχέτιση πολλών μη ανώμαλων γεγονότων μαζί είναι μια κλασσική και ευρέως γνωστή πλέον Denial of Service (DOS) ή μια

Distributed Denial of Service (DDOS) κυβερνοεπίθεση. Κανένας μηχανισμός ασφάλειας σε ένα πληροφοριακό σύστημα δε μπορεί να είναι σε θέση να αναγνωρίσει μια επίθεση DOS/DDOS μόνο από ένα γεγονός που την αποτελεί. Παραδείγματος χάρη, αν ένα συμβάν φτάσει στο σύστημα προστασίας ενός οργανισμού (ασχέτως τι είδους σύστημα προστασίας είναι αυτό) και μέσα αναφέρεται η πληροφορία ότι ο χρήστης X απέτυχε να κάνει σύνδεση στη βάση δεδομένων επειδή έβαλε λάθος κωδικό, τότε το σύστημα προστασίας απλά θα υποθέσει ότι ο υπάλληλος X έγραψε λάθος τον κωδικό του και θα προσπαθήσει ξανά, όλα καλά. Παρόλα αυτά, αν φτάσουν στο σύστημα προστασίας ίδιου οργανισμού 100 συμβάντα που όλα αναφέρουν το προηγούμενο μας παράδειγμα μέσα σε ένα μικρό χρονικό διάστημα (εξαρτάται από την πολιτική του οργανισμού), ο μηχανισμός προστασίας θα πρέπει να συμπεράνει ότι όλα αυτά τα γεγονότα μαζί αποτελούν μια επίθεση DOS/DDOS.



Σχήμα 5: Οπτικοποίηση ανακάλυψης ανώμαλου συμβάντος μέσα από την συσχέτιση άλλων

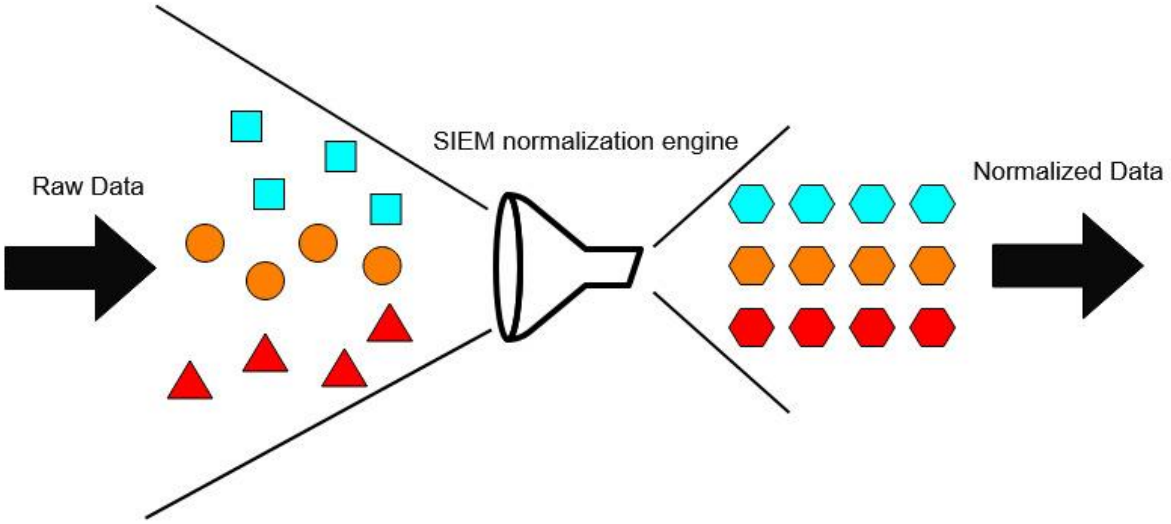
- Κανονικοποίηση και αποδοτική αποθήκευση της πληροφορίας που λαμβάνουν:
  - ❖ Μέχρι τώρα ξέρουμε ότι έχοντας τις ικανότητες ενός SIM, ένα SIEM οφείλει να είναι σε θέση να δέχεται και να λαμβάνει όλων των ειδών τις πληροφορίες, ώστε στη συνέχεια να τις αποθηκεύει όλες μαζί σε μια μαζική αποθήκη αυτών των δεδομένων. Αυτό όμως δεν αρκεί. Ένα SIEM για να κάνει την αποθήκευση των δεδομένων που λαμβάνει

αποδοτική θα πρέπει να κάνει μιας μορφής προ επεξεργασία στα δεδομένα εισόδου (input data) του ώστε να περιορίζει διάφορες διπλές εγγραφές και διπλές αναφορές γεγονότων, αλλά και σε ορισμένες περιπτώσεις να μειώνει τα δεδομένα που αποθηκεύει. Θα μπορούσαμε να πούμε ότι «ξεσκαρτάρει» κάποια άχρηστα events που λαμβάνει, αλλά ο όρος αυτός δεν είναι ακριβής καθώς ούτως ή άλλως η επεξεργασία τους γίνεται.

Τα συστήματα SIEM με τον παραπάνω τρόπο αποδοτικής αποθήκευσης των δεδομένων που δέχονται πετυχαίνουν δύο πολύ σημαντικά πράγματα, να αποθηκεύουν λιγότερο όγκο δεδομένων σε βάση χρόνου και κυριότερα να γίνεται η αναζήτηση μέσα σε αυτά τα δεδομένα πιο γρήγορη και αποτελεσματική.

Επιπλέον πολύ σημαντικό ρόλο για την ορθή λειτουργία ενός συστήματος SIEM έχει και η κανονικοποίηση των δεδομένων εισόδου. Η κανονικοποίηση των δεδομένων είναι η διαδικασία που χρειάζονται οι πληροφορίες ώστε ενώ έρχονται από πολλές διαφορετικές πηγές και έχοντας πολλές διαφορετικές μορφές να μπορούν να συνυπάρχουν σε μια κοινή δομημένη βάση πληροφοριών αλλά να διατηρούν και την πληροφορία που κουβαλούσαν πριν αποθηκευτούν [12].

Είπαμε όμως ότι τα συστήματα SIEM έχουν τις δυνατότητες και των συστημάτων SEM αλλά και των SIM, άρα η πληροφορία που έρχεται εκτός από το να αλλάζει και να αποθηκεύεται κανονικοποιημένη ώστε να γίνεται και ο έλεγχος της, θα πρέπει να υπάρχει και η δυνατότητα της αποθήκευσης της στην μορφή που ήρθε για να καλύψει την ανάγκη της μακροχρόνιας διατήρησης των δεδομένων. Έτσι στους περισσότερους μηχανισμούς SIEM που αναπτύσσονται σήμερα, προστίθεται και η δυνατότητα να αποθηκεύονται τα logs στη μορφή που έρχονται παράλληλα με όλες τις υπόλοιπες λειτουργίες τους. Αυτή η αρχιτεκτονική εξασφαλίζει ασφάλεια και σιγουριά στους προστατευόμενους από έναν μηχανισμό Security Information and Event Management καθώς μπορεί να χρησιμοποιηθεί μελλοντικά σε πιθανές έρευνες μετά από κάποια κυβερνοεπίθεση ή και για να είναι συμβατοί οι οργανισμοί με διάφορα πρότυπα της Ευρωπαϊκής Ένωσης.



Σχήμα 6: Απλοποιημένη οπτικοποίηση της λειτουργίας της κανονικοποίησης

- Παραγωγή συγκεντρωτικών αναφορών:
  - ❖ Σίγουρα από τις πιο χρήσιμες λειτουργίες ενός συστήματος SIEM είναι η δυνατότητα να παράγει συγκεντρωτικές αναφορές ώστε να κρατά τους χρήστες του ενήμερους. Αυτός άλλωστε είναι και ο κύριος σκοπός ύπαρξης ενός μηχανισμού SIEM, όπως είπε ο S. Torino [13], «Ένας μηχανισμός SIEM δεν είναι ένας μηχανισμός επικεντρωμένος γύρω από την ασφάλεια, αλλά ένας μηχανισμός επικεντρωμένος στην παραγωγή γνώσης». Η αρχική ανάγκη ύπαρξης της ιδέας για τους μηχανισμούς που ξέρουμε σήμερα ως Security Information and Event Management systems (SIEMs) ήταν να αναλάβουν την εποπτεία των τεχνολογικών συστημάτων και εγκαταστάσεων ενός οργανισμού, διότι είχαν αναπτυχθεί και είχαν γίνει τόσο περίπλοκοι, που δεν μπορούσε το ανθρώπινο δυναμικό να είναι αποδοτικό πλέον. Αφού το SIEM κάνει όλες τους τις ζωτικές λειτουργίες (συλλογή, κανονικοποίηση, αποθήκευση, ενημέρωση κτλ.), πρέπει να είναι σε θέση να παράγει αναφορές που περιγράφουν τι αποτελέσματα είχε από όλες αυτές τις ενέργειες που έκανε και την τρέχουσα κατάσταση του οργανισμού συμπυκνωμένα.

    Ανέκαθεν από την αρχή της ανάπτυξης τους και την εμφάνιση τους στην αγορά οι μηχανισμοί SIEM είχαν κάποιας μορφής παραγωγή συγκεντρωτικών αναφορών υλοποιημένη. Η τότε συνηθισμένη

υλοποίηση όμως ήταν σε πολύ πρώιμο στάδιο ακόμα και όχι πολύ φιλική προς τον χρήστη. Κάποιοι μάλιστα έχουν αναφερθεί στο παρελθόν στις συγκεντρωτικές αναφορές ως «κουραστικές» [14]. Κάπου εδώ έρχονται να συμβάλουν στην προκαθορισμένη προσπάθεια παραγωγής αναφορών των SIEM διάφοροι εξωτερικευμένοι μηχανισμοί που αναλαμβάνουν ή να κάνουν την παραγωγή της αναφοράς πλήρως μόνοι, ή και απλά να εμπλουτίσουν την αναφορά που έχει ήδη παραχθεί από τον υπάρχον μηχανισμό SIEM. Όπως και να έχει, ο τελικός σκοπός οποιασδήποτε προσπάθειας και αλλαγής γίνεται με επίκεντρο τις συγκεντρωτικές αναφορές που παράγονται από συστήματα SIEM, είναι να είναι πλήρη στην πληροφορία που αναδεικνύουν αλλά και φιλική προς τον χρήστη που ενημερώνεται από αυτές.

Σε εξελιγμένα συστήματα SIEM όμως τα τελευταία χρόνια υπάρχει η τάση παραγωγής πολλών διαφορετικών συγκεντρωτικών αναφορών που η κάθε μια προορίζεται για διαφορετική ομάδα ανθρώπων. Όπως έχει αναφερθεί αρκετές φορές ως τώρα, οι οργανισμοί γίνονται όλο ένα και μεγαλύτεροι, αυτό συνεπάγεται την αύξηση των δραστηριοτήτων τους, που αυτό με τη σειρά του συνεπάγεται την αύξηση της πολυπλοκότητας των δεδομένων που διαχειρίζονται, και κυρίως πληθαίνουν οι ευαισθησίες σε αυτά τα δεδομένα και ο τρόπος που πρέπει να γίνεται η διαχείριση τους. Από τα παραπάνω προκύπτει ότι στη σημερινή τάση της τεχνολογίας και στη τάση λειτουργίας των πλέον μεγάλων οργανισμών, σίγουρα υπάρχουν δεδομένα και αναλύσεις που κάποια άτομα επιτρέπεται να τα γνωρίζουν και κάποια άλλα όχι, ακόμα και αν εργάζονται στον ίδιο χώρο.

- Άμεση ειδοποίηση για τρέχοντα συμβάντα:
  - ❖ Τα συστήματα SIEM χωρίς τη δυνατότητα να παρέχουν άμεσες ειδοποιήσεις σίγουρα δε θα ήταν τόσο χρήσιμα συστήματα όπως πολλοί τα χαρακτηρίζουν [1]. Όπως έχει προαναφερθεί τα SIEM είναι πολύ αποτελεσματικές μηχανές παραγωγής γνώσης. Το να μετατραπεί αυτή η γνώση σε πλήρη ενημέρωση ήδη καλύπτεται από τους δευτερεύοντες μηχανισμούς που παράγουν συγκεντρωτικές αναφορές, η πραγματική αξία όμως των συστημάτων SIEM και αυτό μάλιστα που τους έχει κατατάξει στην περιοχή της ασφάλειας της πληροφορίας και όχι απλά στην αποθήκευση της είναι οι άμεσες

ειδοποιήσεις που παράγουν και προωθούν στους κατάλληλους ανθρώπους για τα συμβάντα που πήραν μέρος την εκάστοτε στιγμή.

Ο τομέας ανάπτυξης της παροχής άμεσων ειδοποιήσεων είναι πολύ ενδιαφέρον θέμα διότι εμπλέκεται άμεσα με τη διαδικασία αυτή και η εμπειρία του χρήστη από έναν μηχανισμό SIEM που τον εξυπηρετεί. Δεν πρέπει να ξεχνάμε ότι μια ειδοποίηση είναι ένας τρόπος να αλληλοεπιδράσει το κάθε σύστημα SIEM με έναν από τους χρήστες του. Αυτό σημαίνει ότι το είδος της ειδοποίησης και η μορφή της παίζουν πολύ σημαντικό ρόλο στην απόδοση ενός SIEM καθώς σχετίζονται έμπρακτα με τον τρόπο που θα αντιδράσει ο παραλήπτης της τελευταίας και πόσο καλά θα κατανοήσει κάποιο τυχόν συμβάν, ώστε να είναι καλά έτοιμος κατά την περίπτωση που θα χρειαστεί να εμπλακεί στην κατάσταση που δημιούργησε ένα συμβάν.

Οι πιο συνηθισμένες μορφές ειδοποιήσεων που χρησιμοποιούνται σήμερα στις αρχιτεκτονικές των SIEM είναι οι εξής:

- Email: Η πιο συνηθισμένη μορφή ειδοποίησης για αρκετά συμβάντα.

- SMS: Τα SMS μηνύματα χρησιμοποιούνται για γεγονότα που απαιτούν λίγο περισσότερη προσοχή.

- Τηλεφωνικά: Μια τηλεφωνική κλήση είναι και πιθανά να παραμείνει η πιο επείγουσα μορφή ειδοποίησης που μπορεί να φτάσει σε έναν χρήστη. Χρησιμοποιείται για γεγονότα που χρειάζονται άμεσα προσοχή.

- Διαδραστικά: Με την εξέλιξη των λειτουργικών συστημάτων και των διεπαφών χρήστη έχουν γίνει πολύ συνηθισμένες οι ειδοποιήσεις με την μορφή ενός pop-up στον υπολογιστή ή στο κινητό των χρηστών.

- SNMP: Κάθε SIEM εκτός απ’ το να είναι το ίδιο manager για άλλες συσκευές, μπορεί να είναι και ο agent για κάποιον SNMP manager που υπάρχει στο δίκτυο. Έτσι έχει τη δυνατότητα να

ειδοποιεί και ένα άλλο σύστημα εκτός από τους χρήστες του ίδιου.

Πιο εξελιγμένα SIEM μπορεί να είναι σε θέση να προσφέρουν και [11]:

- Πιθανή συμμόρφωση ενός οργανισμού σε διάφορα ευρωπαϊκά πρότυπα [16]:
  - ❖ Στον μοντέρνο κόσμο της πληροφορικής η γεωγραφική τοποθεσία ενός οργανισμού έχει μέγιστη σημασία στον τρόπο λειτουργίας των πληροφοριακών τού συστημάτων. Πλέον υπάρχουν πάρα πολλά πρότυπα από εγκεκριμένους οργανισμούς και ενώσεις χωρών τα οποία καθοδηγούν μια επιχείρηση ή έναν οργανισμό ώστε να αξιοποιεί σωστά τα τεχνολογικά του αποθέματα.

    Ένα σύστημα Security Information and Event Management μπορεί να βοηθήσει πάρα πολύ έναν οργανισμό να πετύχει την απαραίτητη συμμόρφωση που ορίζεται από την πιστοποίηση που θέλει να αποκτήσει. Εδώ και λίγα χρόνια, σε κάποια κράτη μέσα στην ευρωζώνη, οι οργανισμοί που διατηρούν πληροφορίες για τους πελάτες τους είναι πλέον νομικά υποχρεωμένοι να είναι συμμορφωμένοι με τους κανόνες που έχει θεσπίσει ο Γενικός Κανονισμός για την Προστασία Δεδομένων (General Data Protection Regulation, GDPR).

    Ένας οργανισμός που θέλει να συμμορφωθεί στους κανονισμούς του GDPR, χρησιμοποιώντας ένα μηχανισμό SIEM μπορεί να πετύχει τα εξής πολύ σημαντικά [17]:

    - Forensic Analysis: Το άρθρο 33 του GDPR [18] αναφέρεται στην δυνατότητα των Forensic Analytics. Από τους πιο αυστηρούς κανόνες του GDPR είναι η δυνατότητα άμεσης ενεργοποίησης forensic πρωτοκόλλων ώστε να αντιμετωπιστεί κάποια διαρροή ή υποκλοπή δεδομένων. Για να γίνει αυτό με επιτυχία, τα δεδομένα του οργανισμού πρέπει να κρατιούνται σαν ιστορικό και να αποθηκεύονται με ασφάλεια σε όλη τη διάρκεια της λειτουργίας του.

- User Behavior Analytics: Το άρθρο 32 του κανονισμού GDPR [18] αναφέρεται στο user behavior analytics. Με αυτόν τον τρόπο το GDPR θέλει να πετύχει την ακεραιότητα των προσωπικών δεδομένων των χρηστών στο κάθε σύστημα που υπάρχουν. Αυτό για έναν απλό διαχειριστή δικτύου σημαίνει ότι πρέπει να εποπτεύεται η πρόσβαση που παρέχεται και σε ποιο επίπεδο, σε κάθε χρήστη ενός πληροφοριακού συστήματος. Με απλά λόγια, πρέπει να είναι καταγεγραμμένο ανά πάσα ώρα και στιγμή το ποιος μπορεί να κάνει τι και σε ποια δεδομένα μέσα στον εκάστοτε οργανισμό. Οι μηχανισμοί SIEM μπορούν μέσα από γεγονότα όπως σύνδεση / αποσύνδεση, αλλαγές δικαιωμάτων, αλλαγές σε αρχεία και λοιπά, να παρέχουν μια εικόνα του ιστορικού χρήσης που κάνει ο κάθε χρήστης σε ένα εποπτευόμενο σύστημα εις βάθος.

- Real-time security monitoring: Επίσης με βάση το άρθρο 32 του κανονισμού GDPR [18], για να είναι ένας οργανισμός συμμορφωμένος, θα πρέπει να επικυρώνει ότι όσο χρησιμοποιεί ένα πληροφοριακό σύστημα που χειρίζεται ευαίσθητα δεδομένα, είναι αναγκασμένος να παρέχει εμπιστευτικότητα, την ακεραιότητα και την διαθεσιμότητα των δεδομένων αυτών στους χρήστες που τους ανήκουν. Για να επιτευχθεί ο παραπάνω δύσκολος στόχος, ένα σύστημα SIEM είναι σε θέση να επιβλέπει και να επιτηρεί συνεχώς τα γεγονότα ασφάλειας και μη που συμβαίνουν στον εκάστοτε οργανισμό. Έτσι αποτρέπονται οι ενέργειες που διαβάλουν τους 3 πυλώνες της ασφάλειας της πληροφορίας μέχρι σήμερα (confidentiality, integrity, availability) [20], και αν δεν αποτραπούν, σίγουρα υπάρχει ο εντοπισμός τους για μετέπειτα μετρίαση του συμβάντος που υπήρξε.

- Identity Mapping:
  - Μια ιδέα που ακόμα βρίσκεται στο βρεφικό της στάδιο. Το σκεπτικό του Identity Mapping [11] βασίζεται στο ότι θα καταγράφονται συγκεκριμένες δικτυακές πληροφορίες ή hardware specific πληροφορίες και θα συνδέονται με φυσικά πρόσωπα που χρησιμοποιούν το εποπτευόμενο σύστημα. Παραδείγματος χάρη θα

μπορούσε να γίνει σύνδεση κάθε υπαλλήλου μιας εταιρίας με τη MAC address της κάρτας δικτύου του ή με την IP address που είναι αφιερωμένη στον εργασιακό του ή προσωπικό του υπολογιστή. Αυτό προσφέρει σίγουρα καλύτερο και πιο αποτελεσματικό εντοπισμό σε πιθανές ανωμαλίες που θα ανιχνευθούν στο σύστημα υπό εποπτεία αλλά δυστυχώς εδώ και αρκετό καιρό η συγκεκριμένη ιδέα δεν έχει προχωρήσει πολύ, ή μάλλον όσο θα μπορούσε συγκριτικά με τις υπόλοιπες αναπτύξεις των SIEM, διότι η αλλαγή ή το ‘’μασκάρεμα’’ των δικτυακών χαρακτηριστικών ή των μοναδικών χαρακτηριστικών του υλικού μιας συσκευής, είναι εξαιρετικά εύκολο.

- Risk Determination:
  - ❖ Η ιδέα πίσω από το Risk Determination [13] είναι πολύ πρωτοπόρα και ενδιαφέρον σαν έργο ανάπτυξης. Ένα SIEM που έστω κοντεύει να φτάσει την πληρότητα στην απόδοση του ώστε να παρέχει κάτι τέτοιο, θα είναι σε θέση να υπολογίσει ποιος είναι ο κίνδυνος που υπάρχει για τον οργανισμό από ένα γεγονός που πήρε μέρος, βάζοντας στο παιχνίδι και μια ακόμα μεταβλητή που ως τώρα δεν έχει συζητηθεί, τη σημαντικότητα του μηχανήματος / υπηρεσίας που αφορά το γεγονός αυτό. Τα πατροπαράδοτα συστήματα SIEM υπολογίζουν το ρίσκο χρησιμοποιώντας 2 βασικές πληροφορίες που κατά την εμφάνιση της ανωμαλίας υπάρχουν ήδη και είναι ικανά να τις ανακτήσουν άμεσα, την σημαντικότητα του γεγονότος, και την εμπιστευτικότητα του μηχανισμού μέσω του οποίου εντοπίστηκε. Παραδείγματος χάρη ένα DDOS συμβάν θα μπορούσε να έχει επίπεδο σημαντικότητας 3 (critical) και επίπεδο εμπιστευτικότητας 3 (μικρό false positive ποσοστό κανόνα). Με την τρίτη μεταβλητή πλέον, την αξία του μηχανήματος / υπηρεσίας που έλαβε μέρος το συμβάν ως προς τον οργανισμό, ανοίγουν πολλοί νέοι ορίζοντες στη δυνατότητα ανάλυσης των συστημάτων SIEM και η γνώση που παράγουν μπορεί να γίνει εξαιρετικά πιο χρήσιμη.

- Application Programming Interface (API):
  - ❖ Μια πολύ μοντέρνα ιδέα που έρχεται όλο και περισσότερο στο παρασκήνιο σε νέα προϊόντα SIEM είναι η ύπαρξη API [11]. Προσφέροντας το δικό τους API τα συστήματα SIEM ανοίγουν τις πόρτες τους σε νέους μηχανισμούς, οι οποίοι έτσι θα μπορούν να ‘κουμπώσουν’ και να συνεργαστούν μαζί τους πολύ πιο εύκολα. Εδώ

θα υπάρξει σίγουρα πολύ γρήγορα η υιοθέτηση της παραπάνω τεχνικής καθώς η δυνατότητα οποιουδήποτε προϊόντος να συνεργαστεί εύκολα με άλλα χρήσιμα προϊόντα γύρω του, ανεβάζει την αξία του στην αγορά κατακόρυφα.

- Role based Access Control (RBAC):
  - ❖ Το να υπάρχει δυναμικό Role-Based Access Control [22] και να είναι προσφερόμενο από έναν μηχανισμό SIEM ενός οργανισμού σημαίνει ότι εκτός από το να φαίνεται η μεγάλη εικόνα της κατάστασης του οργανισμού και να είναι καταγεγραμμένα όλα τα γεγονότα που λαμβάνουν μέρος, να υπάρχουν και κανόνες που ελέγχουν αυτά τα προαναφερόμενα ώστε να διατηρείται εν τάξη η εξατομικευμένη πολιτική ασφάλειας και ρόλων χρήστη που έχει ορίσει ο ίδιος ο οργανισμός βάσει των δικών του αναγκών και τον δικό του ξεχωριστό τρόπο λειτουργίας. Έτσι, ο μηχανισμός SIEM που τον εποπτεύει, εκτός από τον έλεγχο στα γεγονότα που συμβαίνουν, μπορεί να ασκεί έλεγχο και στην πηγή από όπου προέρχονται. Διατηρώντας έτσι μια πολιτική RBAC και εξασφαλίζοντας για το εποπτευόμενο σύστημα την επιθυμητή οργάνωση.

- Παραγωγή συγκεντρωτικών αναφορών για μαζικά συμβάντα στο παρελθόν on demand:
  - ❖ Η συγκεντρωτικές αναφορές ανά συγκεκριμένα χρονικά διαστήματα που ορίζονται από τις προτιμήσεις των χρηστών του SIEM, είναι απαραίτητο να παρέχονται από οποιοδήποτε προϊόν που κατηγοριοποιεί τον εαυτό του σαν ένα Security Information and Event Management σύστημα. Η πολυπλοκότητα ανεβαίνει όμως όταν αυτές οι συγκεντρωτικές αναφορές επιθυμείται να παράγονται τη στιγμή που επιθυμεί ο χρήστης και να πηγαίνουν πίσω στο χρόνο όσο αυτός αρέσκεται. Όπως έχουμε πει τα δεδομένα των γεγονότων που παίρνουν μέρος σε έναν οργανισμό ο οποίος εποπτεύεται από ένα μηχανισμό SIEM, αποθηκεύονται για ένα σχετικά μεγάλο χρονικό διάστημα. Αυτό βέβαια δε σημαίνει ότι η ανάκτηση τους και η επεξεργασία τους είναι μια απλή διαδικασία. Οι σωστοί μηχανισμοί SIEM φροντίζοντας να διατηρούν τα δεδομένα που αποθηκεύουν με αποδοτικό τρόπο και κρατώντας μια σωστή οργάνωση στην αποθήκη logs τους, μπορούν να καλύψουν ανάγκες όπως η παραγωγή συγκεντρωτικών αναφορών για μαζικά συμβάντα στο παρελθόν.

Βέβαια αυτή η επέκταση των μοντέρνων SIEM δεν βρίσκεται στο παρασκήνιο τώρα, καθώς θεωρείται λιγότερο σημαντική από άλλες όπως το Risk Determination και το Identity Mapping που αναλύσαμε προηγουμένως.

2.5 Αποσαφήνιση πληροφοριών μηχανής

Η αποσαφήνιση πληροφορίας είναι ένας από τους βασικούς πυλώνες που κάνουν την ύπαρξη των Security Information and Event Management μηχανισμών δυνατή.

Κάτι που δεν είναι ευνόητο για όλους τους μηχανικούς της επιστήμης της πληροφορικής όταν ξεκινούν είναι ότι η μηχανές δεν είναι φτιαγμένες με στόχο να κατανοούν και να επεξεργάζονται την ανθρώπινη μορφή επικοινωνίας, ή τουλάχιστον δεν είναι ακόμα [23].

Η μηχανές μιλούν μεταξύ τους με πολλούς τρόπους, αλλά στην προκειμένη περίπτωση δε θα εστιάσουμε στο πως οι μηχανισμοί SIEM διαχειρίζονται την επικοινωνία με τα εποπτευόμενα μηχανήματα γύρω τους, αλλά με το πώς κάνουν τη διάσπαση των 'ωμών' ή ανεπεξέργαστων δεδομένων που λαμβάνουν ώστε τελικά να είναι σε θέση να αποσπάσουν όσο περισσότερη χρήσιμη πληροφορία μπορούν για να επιτύχουν με τον καλύτερο δυνατό τρόπο την ανάλυση και ανίχνευση γεγονότων που έχουν οριστεί ως συμβάντα ανωμαλιών.

Συνεχίζοντας, θα κάνουμε μια μικρή προσομοίωση προσεγγίζοντας σε μικρό βαθμό τον τρόπο λειτουργίας και ανάλυσης ενός παραδοσιακού SIEM μηχανισμού.

Log source: [24]

sshd[8813]: Accepted password for root from 192.168.10.1 port 1066 ssh2

Στην παραπάνω εγγραφή log είναι πολύ εύκολο να αντλήσουμε πληροφορία απλά διαβάζοντας το. Έγινε μια επιτυχημένη απόπειρα να συνδεθεί κάποιος χρήστης ως root στο σύστημα. Το process με το οποίο εκτελέστηκε η ενέργεια είναι ο sshd daemon και το process ID (PID) που του παραχωρήθηκε είναι το 8813. Η επιτυχημένη σύνδεση έγινε από την IP address 192.168.10.1 και μέσω του port 1066. Τέλος η σύνδεση έγινε χρησιμοποιώντας την version 2 του πρωτοκόλλου SSH.

Ένας μηχανισμός SIEM όμως δεν μπορεί να σκεφτεί όπως σκέφτονται οι άνθρωποι και να αναλύσει την πληροφορία όπως την αναλύουν αβίαστα αυτοί. Όπως έχουμε καταλάβει ήδη το παραπάνω log έχει ένα συγκεκριμένο format που δημιουργήθηκε ώστε να είναι κατανοητό από ανθρώπους.

Η υπηρεσία καταγράφει σε ένα αντίστοιχο αρχείο καταγραφής συμβάντων (log file) για κάθε συμβάν μια εγγραφή που ακολουθεί το πρότυπο του δημιουργού του λογισμικού sshd της και είναι εύκολο να καταλάβουμε ότι μοιάζει κάπως έτσι:

sshd[X]: Y password for A from B port C D

Εδώ οι πληροφορίες που αλλάζουν λοιπόν είναι οι εξής:

- X: Process ID (αναγνωριστικό διεργασίας)
- Y: State (κατάσταση)
- A: User (χρήστης)
- B: IP address (διεύθυνση πρωτοκόλλου IP)
- C: Port (θύρα δικτύου)
- D: SSHv2 (πρωτόκολλο)

Η πρώτη δουλεία του μηχανισμού SIEM σε οποιαδήποτε εγγραφή log που λαμβάνει είναι να καταλάβει σε ποιο πρόγραμμα αναφέρεται. Έτσι θα μπορέσει να χρησιμοποιήσει τους σωστούς decoders ώστε να συνεχίσει με την ανάλυση της πληροφορίας έχοντας μια ιδέα πλέον για το που βρίσκεται η χρήσιμη πληροφορία μέσα στην log εγγραφή. Αυτό το πετυχαίνει ανατρέχοντας το κομμάτι του log sshd[X].

Γνωρίζοντας πλέον ότι πρόκειται για ένα log entry δημιουργημένο από το process sshd, μπορεί να αρχίσει να αναλύει και να σπάει την εγγραφή που έλαβε χωρίζοντας την σε πεδία όπως τα έχει ορίσει ο μηχανικός που έγραψε και τον decoder για το sshd process. Έτσι πολύ εύκολα πλέον ακολουθεί τις οδηγίες του decoder και αντλεί τις πληροφορίες που αντλήσαμε και εμείς στην ανάλυση που κάναμε παραπάνω με το μάτι.

Πιο αναλυτικά οι decoders θα παρουσιαστούν στο κεφάλαιο 5.

ΚΕΦΑΛΑΙΟ 3

ΠΡΟΤΥΠΑ ΚΑΙ ΤΕΧΝΟΛΟΓΙΕΣ

ΕΙΣΑΓΩΓΗ

Υπάρχει πολύ μεγάλο πλήθος προτύπων και τεχνολογιών που συμβάλουν μαζί με τους μηχανισμούς SIEM στην διεύρυνση των οριζόντων του τομέα της ασφάλειας στην επιστήμη της πληροφορίας. Αυτά τα πρότυπα και οι τεχνολογίες αναπτύσσονται καθημερινά και ως τώρα έχουν παίξει πολύ σημαντικό ρόλο στην αποδοτικότητα πολλών ειδών εργαλείων που είναι φτιαγμένα με σκοπό την επίβλεψη και την προστασία ενός οργανισμού ή ενός νοικοκυριού.

3.1 Rule Standards

Ένας ακόμα πυλώνας της ιδεολογίας των μηχανισμών Security Information and Event Management (SIEM) είναι οι κανόνες που χρησιμοποιούν ώστε να πάρουν τις αποφάσεις τους και να υποδείξουν ότι ένα γεγονός που συνέβη σε ένα σύστημα υπό την εποπτεία τους αποτελεί μη φυσιολογική συμπεριφορά που χρήζει ίσως ιδιαίτερη μεταχείριση.

Χωρίς αυτή τη δυνατότητα τα συστήματα SIEM δε θα είχαν μεγάλη διαφορά από τις παραδοσιακές βάσεις δεδομένων που όλοι γνωρίζουμε, απλά θα είχαν την ικανότητα να αποθηκεύουν δεδομένα ώστε να είναι προσβάσιμα όποτε και αν χρειαστούν μελλοντικά.

Χρησιμοποιώντας λοιπόν τις χρήσιμες πληροφορίες που συγκέντρωσαν από την διαδικασία της αποσαφήνισης (βλ. 2.5 Αποσαφήνιση πληροφοριών μηχανής), οι μηχανισμοί SIEM αποφασίζουν με διάφορα κριτήρια αν το συμβάν που ελέγχεται κάθε φορά πρόκειται για μια ανωμαλία στο σύστημα ή όχι.

3.1.1 SIGMA

Δεδομένου ότι όταν μιλάμε για SIEM αναφερόμαστε ακόμα σε μια σχετικά νέα τεχνολογία στον τομέα της επιστήμης της πληροφορίας. Είναι λογικό τα διαφορετικά προϊόντα που έχουν αναπτυχθεί από τις εν ασχολία με το αντικείμενο εταιρείες, να ακολουθούν ένα δικό τους ιδιαίτερο και ξεχωριστό format συγγραφής κανόνων και τρόπο επεξεργασίας των δεδομένων που λαμβάνουν, ώστε να σχηματιστεί η νοητή αλυσίδα λειτουργιών που χαρακτηρίζει ένα προϊόν ως ένα μηχανισμό ασφάλειας SIEM. Επιπρόσθετα, αν σκεφτούμε ότι οι τάσεις της αγοράς γύρω από τον τομέα της ασφάλειας στην πληροφορική ολοένα και αυξάνονται [25], θα βρούμε έναν ακόμη λόγο η κάθε εταιρία που χτίζει έναν δικό της μηχανισμό SIEM, να κρατά την τεχνογνωσία της καθαρά και μόνο για την ίδια, βάζοντας τα δυνατά της να κυριαρχήσει έναντι των ανταγωνιστών της.

Ο Florian Roth και ο Thomas Patzke από την Nextron Systems είδαν την παραπάνω κατάσταση στην περιοχή έρευνας γύρω από τα SIEM ως μια ευκαιρία να δημιουργήσουν ένα πρωτοπόρο open standard format που θα ένωνε πολλούς μηχανισμούς SIEM και θα επέτρεπε την εύκολη μεταφορά και διαμοιρασμό γνώσης από πλατφόρμα σε πλατφόρμα, το SIGMA. Σύμφωνα με τον F. Almeida [26], τα open standards και οι open source τεχνολογίες βοηθούν την επιστήμη της πληροφορικής να φτάσει σε νέα ύψη και σε ανεξερεύνητα μονοπάτια.

Το SIGMA λοιπόν, είναι ένα είναι generic και open signature format που προσφέρει την συγγραφή κανόνων που περιγράφουν μια ανωμαλία με έναν ευθύ και απλό τρόπο. Το φορμάτ της γλώσσας κανόνων SIGMA είναι πολύ ευέλικτο και συμβατό με όλων των ειδών τις εγγραφές logs. Είναι δημιουργημένο με τέτοιο τρόπο ώστε να είναι εύκολο να αποτυπωθεί σε έναν κανόνα SIGMA ακριβώς το γεγονός που χρήζει η συγκεκριμένη ανάγκη ή ανάγκες του κάθε οργανισμού. Βέβαια ο σκοπός ύπαρξης της γλώσσας SIGMA δεν είναι αυτός. Αυτό που πετυχαίνει κάποιος αποτυπώνοντας συμβάντα ασφάλειας σε SIGMA κανόνες είναι να κάνει αυτούς του κανόνες διαθέσιμους για χρησιμοποίηση και αξιοποίηση από τα περισσότερα SIEM στην αγορά. Με λίγα λόγια, υπάρχει πλέον ένα open standard πρότυπο ώστε να είναι δυνατός ο διαμοιρασμός γνώσης κανόνων και συμβάντων για τα συστήματα SIEM.

3.1.1.1 Δομή των SIGMA κανόνων

Οι κανόνες SIGMA βασίζονται σε τέσσερις βασικές κατηγορίες πεδίων που τους αποτελούν:

- Τα *Metadata* είναι οι πληροφορίες που μας βοηθούν να περιγράψουμε το γεγονός από άνθρωπο σε άνθρωπο.

- Το *Log Source* είναι η πηγή που προέρχεται το συγκεκριμένο log entry ώστε να θεωρηθεί ότι ανήκει σε αυτόν τον κανόνα και ότι το περιγράφει σωστά

- Το *Detection* είναι ίσως και το σημαντικότερο εκ των τεσσάρων καθώς περιγράφει τα χνάρια που αφήνει πίσω του αυτό το γεγονός σε γλώσσα μηχανής, ώστε να είναι ανιχνεύσιμο πλέον το συμβάν από τα SIEM analysis engines.

- Τέλος το *Condition* είναι αυτό που καθορίζει ποια από τα Detection περιγραφόμενα χαρακτηριστικά πρέπει να αληθεύουν ώστε να παρθεί η τελική απόφαση ότι το συγκεκριμένο γεγονός υπό ανάλυση ταιριάζει αρκετά με την ανώμαλη συμπεριφορά που περιγράφεται από αυτόν τον κανόνα.

Πιο αναλυτικά, τα πιο συνηθισμένα πεδία που μπορεί να αποτελούν ένα κανόνα SIGMA μπορεί να είναι τα εξής: (έντονο κόκκινο = υποχρεωτικό πεδίο)

- **title**
  - status
  - description
  - author
  - reference
  - …
- **logsource**
  - category
  - product
  - service
  - definition
  - …
- **detection**

        - `{search-identifier}`
            - `{string X}`
            - `{fieldA: value}`
        - `{search-identifier}`
            - `{string Y}`
            - `{fieldB: value}`
        - `timeframe`
- **`condition`**
- `falsepositives`
- `level`
- …

Επεξήγηση πεδίων:

- ***Title***: Μια σύντομη περιγραφή της ανωμαλίας που στοχεύει ο κανόνας αυτός.

- Status: Η κατάσταση ανάπτυξης του κανόνα.
    - stable: Δοκιμασμένος και αποτελεσματικός
    - test: Δοκιμασμένος αλλά ίσως χρειαστεί λίγη ρύθμιση
    - experimental: Ακόμη υπό δοκιμές

- Description: Συμπληρωματική περιγραφή για το συμβάν που περιγράφεται με αυτόν τον κανόνα.

- Author: Ο συγγραφέας / δημιουργός του κανόνα.

- Reference: Η πηγή προέλευσης του κανόνα, παραδείγματος χάρη blogs, papers, presentations κτλ.

- ***Logsource***: Το logsource είναι μια κατηγορία πεδίων που περιγράφει τον τύπο ή προέλευση των δεδομένων log που αυτός ο κανόνας αναφέρεται.
    - Category: Η κατηγορία των τεχνολογιών που ανήκει ο κανόνας, π.χ.
        - Firewall
        - Web
        - antivirus

  - Product: Το συγκεκριμένο προϊόν ή υπηρεσία που παράγει αυτού του τύπου τα logs, π.χ.
    - Windows
    - Apache
    - pfSense
  - Service: Το συγκεκριμένο service name που αναγράφεται στο log entry, π.χ.
    - sshd
    - pam
    - mysqld
  - Definition: Χρήσιμο για άλλους μηχανικούς που διαβάζουν τον κανόνα, μπορεί να περιγράψει το σκεπτικό για τις προηγούμενες επιλογές όσον αφορά το log source.

- ***Detection***: Ένα σετ από στοιχεία που αφήνει πίσω η ανωμαλία που στοχεύουμε.
  - Search-identifiers: Τα search identifiers είτε θα είναι λίστες (keywords) είτε θα είναι Maps (selection).
    - Keywords: Τα keywords ορίζουν strings τύπου regular expression τα οποία θα κάνουν match με οποιοδήποτε κομμάτι στο πλήρες log entry που ελέγχεται.
    - Selection: Το selection είναι ζεύγη κλειδιών με τις τιμές τους. Κάθε search identifier selection μπορεί να περιέχει ένα ή περισσότερα ζεύγη και μεταξύ τους να υπάρχουν σχέσεις AND ή OR.
  - Timeframe: Το χρονικό διάστημα από τη στιγμή του πρώτου τέτοιου τύπου συμβάντος ώστε να δημιουργηθεί chain με τα επόμενα. Προσοχή στο timeframe, καθώς τις περισσότερες περιπτώσεις περιγράφεται και ξεχωριστά από τον κάθε μηχανισμό SIEM που χρησιμοποιείται.

- ***Condition***: Είναι το πιο περίπλοκο πεδίο του SIGMA κανόνα, και δέχεται συνέχεια αλλαγές με κάθε version του SIGMA format που κυκλοφορεί. Είναι αυτό που ορίζει ποια από τα detection fields θα πρέπει να ικανοποιηθούν ώστε να θεωρηθεί το ελεγχόμενο event ανωμαλία. Μερικά από τα arguments που δέχεται είναι τα εξής:
  - Logical AND/OR: search-identifier1 OR search-identifier2
  - 1 : all of the detection field requirements must be met

  - Negotiation with NOT: search-identifier1 and NOT search-identifier2
  - Brackets { } : Με τα brackets μπορούμε να συνδυάσουμε με OR ή AND keywords και selections

- False Positives: Εδώ μπορούμε να αναφέρουμε πιθανόν συμβάντα που εν γνώσει μας ίσως δημιουργήσουν κάποιες false positive ανωμαλίες. Αυτό το πεδίο χρησιμοποιείται για να προειδοποιήσουμε άλλους μηχανικούς ή σαν προσωπικό documentation.

- Level: Καθορίζει το επίπεδο επικινδυνότητας μιας ανωμαλίας.
  - low
  - medium
  - high
  - critical

3.1.1.2 Παράδειγμα χρήσης SIGMA

Ένας απλός οδηγός χρήσης για την αξιοποίηση του SIGMA open signature format είναι ο εξής:

- Φτιάχνουμε τον κανόνα SIGMA περιγράφοντας το γεγονός που επιθυμούμε.

- Φτιάχνουμε τον ‘χάρτη’ ή οδηγό που αντιστοιχίζει τις μεταβλητές του open standard format SIGMA σε μεταβλητές που χρησιμοποιεί η δική μας πλατφόρμα SIEM.

- Εκτελούμε τον μετατροπέα ‘sigmac’ που αντιστοιχεί εκάστοτε SIEM.

- Τέλος, απλά παίρνουμε τα αποτελέσματα του μετατροπέα, που είναι απλοί κανόνες με τη μορφή που χρησιμοποιεί το δικό μας SIEM σύστημα, και τις εισάγουμε στον μηχανισμό μας.

Στη συνέχεια θα εξετάσουμε ένα απλό παράδειγμα παραγωγής και ολοκληρωμένης χρήσης ενός SIGMA κανόνα για την γνωστή πλατφόρμα Splunk [28].

Ο στόχος μας είναι να περιγράψουμε μια προσπάθεια telnet σύνδεσης σε έναν internal server που έχει ένας οργανισμός. Ο internal server έχει την IP address 203.0.113.2 (reserved experimental address space for documentation [29])

Το πρώτο μας βήμα είναι να γράψουμε τον SIGMA κανόνα, π.χ. telnet_lookout.yml

```
title: Telnet connection attempt on restricted device
logsource:
category: firewall
     product: ufw
detection:
     selection:
          dst_port: 23
          dst_ip: 203.0.113.2
     condition: selection
```

Έπειτα θα φτιάξουμε το αρχείο για να κάνουμε το field mapping με τις μεταβλητές που χρησιμοποιεί το Splunk. Χρειαζόμαστε τις index, το destination_port και το source_port μεταβλητές, π.χ. telnetonly-splunk-sigma.yml.

```
logsources:
     ufw:
          category: firewall
          product: ufw
          index: main
     fieldmappings:
          destination_port: dst_port
          destination_ip: dst_ip
     defaultindex: main
```

Στη συνέχεια απλά εκτελούμε τον sigmac μετατροπέα ώστε να μας δώσει σαν έξοδο τον splunk κανόνα που επιθυμούμε.

```
./sigmac --target splunk --config telnetonly_splunk_sigma.yml
telnet_lookout.yml
```

Και θα παραχθεί ο κανόνας splunk έτοιμος να μπει στην πλατφόρμα μας:

```
{index=”main” destination_port=”23” destination_ip=”203.0.113.2”}
```

Τελειώνοντας, πρέπει να αναφερθεί μια πολύ ενδιαφέρουσα ιδέα για την χρήση του προτύπου SIGMA από έναν από τους δύο δημιουργούς του [30]. Ο Florian Roth θέτει το ερώτημα, «εφόσον τα SIGMA rules είναι εύκολο να κατασκευαστούν και να δημιουργηθούν μέσα σε λίγα λεπτά, γιατί παραμένει δουλειά των SOC engineers και όχι των developers που φτιάχνουν τα apps τους; Ποιος μπορεί να περιγράψει καλύτερα μια πιθανή ανωμαλία από τον άνθρωπο που έφτιαξε το σύστημα που αφορά;» Όπως και να έχει, το πρότυπο SIGMA βρίσκεται στην κεντρική σκηνή της τεχνολογικής ανάπτυξης των μηχανισμών SIEM αυτόν τον καιρό, και θα μείνει για πολύ ακόμα εκεί.

3.1.2 YARA

Η YARA είναι ένα εργαλείο που χρησιμοποιεί ένα δικό του πρότυπο σαν γλώσσα και σαν κύριο στόχο έχει να βοηθά τους ερευνητές των κακόβουλων λογισμικών (malware researchers) να ανιχνεύουν ευκολότερα και πιο αποδοτικά τα malwares που κινούνται σε ένα δίκτυο. Εκτός από την ανίχνευση, η YARA βοηθά στην κατηγοριοποίηση και ταξινόμηση των malwares. Δίνει έναν τρόπο στους ερευνητές να συνθέσουν σε ένα συμπιεσμένο κείμενο, την περιγραφή, τα σημάδια ανίχνευσης αλλά και τις συνθήκες ανίχνευσης για το malware που περιγράφεται.

3.1.2.1 Δομή των YARA κανόνων

Τα τρία βασικά κομμάτια πληροφορίας που περιέχονται σε μια περιγραφή YARA ή αλλιώς έναν YARA κανόνα είναι τα εξής [47]:

- Τίτλος: Ο κάθε κανόνας πρέπει να έχει έναν τίτλο, που τον ξεχωρίζει από τους υπόλοιπους.

- ***meta***: Τα meta αποτελούν τα μεταδεδομένα ενός YARA κανόνα και χρησιμοποιούνται για να χαρακτηρίσουν το malware που περιγράφεται στον κανόνα αυτόν. Μερικά γνωστά metadata είναι:
  - description: Η λεκτική περιγραφή του malware ή οποιαδήποτε πληροφορία θέλει να μεταφέρει ο συγγραφέας του σε κάποιον αναγνώστη.
  - threat_level: Το επίπεδο κινδύνου του συγκεκριμένου malware (τονίζεται ότι τα επίπεδα κινδύνου για τα malware που χαρακτηρίζονται δεν είναι συγκεκριμένα και ορίζονται από τις ανάγκες του συστήματος που χρησιμοποιείται το tool YARA.
  - in_the_wild: Περιγράφει αν το malware είναι σε ενεργή κατάσταση και υπάρχει πιθανότητα να μολύνει κάποιον.

- ***strings***: Τα σημάδια ανίχνευσης των malware, αποτυπώνονται με strings. Η YARA προσφέρει τρία είδη strings προς το παρόν:
  - hex_string: Τα δεκαεξαδικά strings που θα ανιχνευτούν μέσα σε πακέτα είναι από τους πιο συνηθισμένους τρόπους ανίχνευσης κακόβουλου λογισμικού.
  - text_string: Ο πιο απλός τρόπος ανίχνευσης ενός συγκεκριμένου πακέτου ώστε να βρεθεί ένα malware.
  - xor_string: Από τους πιο περίπλοκους τρόπους ανίχνευσης malware, το κάθε string κάνοντας μια λογική πράξη XOR, αλλάζει πλήρως δομή. Ο σκοπός των xor_strings είναι να ανακαλύπτουν τα κρυμμένα strings πίσω από τα strings που έχουν κωδικοποιηθεί με αυτόν τον τρόπο.

- ***condition***:
  - Οι συνθήκες ενός κανόνας είναι αυτό που ορίζει πότε θα χτυπήσει ο κανόνας σε ένα συμβάν που έχει ανιχνευθεί. Είναι απλές Boolean εκφράσεις (π.χ. AND, OR, “<”, “>”, “==”)που περιγράφουν την σχέση μεταξύ των σημαδιών ανίχνευσης.

3.1.2.2 Παράδειγμα χρήσης YARA

Ένα απλό παράδειγμα ενός κανόνα YARA είναι το εξής:

```
rule silent_banker : banker
{    meta:
        description = "Πρόκειται για τον ιό crazymalware"
        threat_level = 3
        in_the_wild = true

    strings:
        $a = {6A 40 68 00 30 00 00 6A 14 8D 91}
        $b = {8D 4D B0 2B C1 83 C0 27 99 6A 4E 59 F7 F9}
        $c = "UVODFRYSIHLNWPEJXQZAKCBGMT"

    condition:
        $a or $b or $c
}
```

Ο παραπάνω YARA κανόνας περιγράφει το malware “crazymalware” λέγοντας ότι έχει επίπεδο επικινδυνότητας 3 και είναι αυτή τη στιγμή ενεργός. Αναφέρει τρία strings ως σημάδια ανίχνευσης και θέτει στις συνθήκες ένα λογικό OR ανάμεσα στα σημάδια ώστε αν έστω και ένα βρεθούν, ο κανόνας κάνει trigger.

Ένα παράδειγμα κανόνα YARA με xor_strings [47]:

```
rule XorExample1
{
    strings:
        $xor_string = "This program cannot" xor

    condition:
        $xor_string
}
```

Ο παραπάνω κανόνας, περιέχοντας τη λογική πράξη XOR μετά τη δήλωση του string του αυτό που πετυχαίνει είναι να γίνονται όλες οι πιθανές πράξης XOR πάνω στο κάθε byte του πακέτου. Αν γραφόταν ένας κανόνας με την ίδια λειτουργικότητα αλλά χωρίς την πράξη XOR της YARA θα έμοιαζε κάπως έτσι:

```
rule XorExample2
{
    strings:
        $xor_string_00 = "This program cannot"
        $xor_string_01 = "Uihr!qsnfs`l!b`oonu"
        $xor_string_02 = "Vjkq\"rpmepco\"acllmv"
        // Repeat for every single byte XOR
    condition:
      any of them
}
```

3.2 Threat Intelligence Exchange

Ο διαμοιρασμός γνώσης ήταν και θα είναι πάντα ένα από τα πιο σημαντικά κομμάτια που βοηθούν όλους τους τομείς της επιστήμης να προχωρούν μπροστά. Η επιστήμη της πληροφορικής δεν αποτελεί την εξαίρεση στον κανόνα και ειδικά μετά από την απότομη εμφάνιση του παγκόσμιου διαδικτύου στην καθημερινότητα των περισσοτέρων πλέον ανθρώπων, όλο και περισσότερη γνώση και πληροφορία μοιράζεται μεταξύ όλων χωρίς ιδιαίτερη προσπάθεια.

Η παραγωγή και διαμοιρασμός γνώσης που αφορούν τα εγκλήματα στον κυβερνοχώρο (Cyber Threat Intelligence, CTI), είναι μια ιδέα και έννοια που γέννησε πλήθος καινούργιων ευκαιριών για διάφορες τεχνολογίες. Σύμφωνα με την Gartner [31], “το Cyber Threat Intelligence είναι ακόμη ένας πολύ ευέλικτος όρος” και σύμφωνα με τον D. Shackleford [32] “Υπάρχει ακόμη πολύ σύγχυση γύρω από την έννοια του Threat Intelligence και τον τρόπο που μεταφέρετε και αξιοποιείται”.

Ένας ορισμός που προσεγγίζει την λειτουργικότητα και χρησιμότητα του Cyber Threat Intelligence είναι ότι αποτελεί γνώση σχετικά με ανώμαλες και κακόβουλες συμπεριφορές και συμβάντα που έχουν παρατηρηθεί στον κυβερνοχώρο. Κάποιοι από τους λόγους που το CTI είναι στο παρασκήνιο πολλών ερευνητικών ομάδων και οργανισμών είναι οι εξής:

- Βοηθά στην καλύτερη προετοιμασία από κυβερνοεπιθέσεις.
- Παρέχει γνώση για νέες απειλές που μπορεί να δεχτούμε.

- Βοηθά στον εντοπισμό των ήδη υπαρχόντων απειλών.
- Συνεισφέρει στην καλύτερη κατανόηση των επιθέσεων.
- Με βάση τα παραπάνω, βοηθά στην ανάπτυξη πιο αποτελεσματικών αμυντικών στρατηγικών ενάντια σε επιθέσεις.

Το σημαντικότερο χαρακτηριστικό που πρέπει να περιέχει η γνώση που μεταφέρεται ως Cyber Threat Intelligence είναι να υποδεικνύει τον λόγο ύπαρξης της. Δηλαδή να μπορείς με βάση αυτή να πάρεις μια απόφαση σε περίπτωση που χρειαστεί, ασχέτως αν αυτή η απόφαση έχει να κάνει με τον τρόπο άμυνας από μια απειλή ή τον τρόπο ανίχνευσης της [33][34]. Αυτό που τεχνολογικά χρειάζεται για την αξιοποίηση όλης αυτής την 'έννοιας' τους Cyber Threat Intelligence, είναι ένας τρόπος να εκφράζεται σε γλώσσα μηχανής αποδοτικά και να μεταφέρεται από οργανισμό σε οργανισμό.

3.2.1 STIX (Structured Threat Information Expression)

Το Structured Threat Information eXpression (STIX) είναι ένα open standard format που ορίζει έναν τρόπο για να αποτυπώνεται δομημένα το Cyber Threat Intelligence σε γλώσσα που είναι κατανοήσιμη από τη μηχανή [36].

Ο κύριος στόχος του STIX είναι να αποτελεί την λύση στην προσπάθεια για αυτοματοποιημένη μεταφορά υψηλής ποιότητας CTI από οργανισμό σε οργανισμό. Το STIX παρότι ανήκει στον MITRE, είναι αποτέλεσμα από προσπάθειες που έγιναν από όλο το community του CTI και σύμφωνα με αυτό [37] η γνώση που μοιράζεται για τις απειλές στον κυβερνοχώρο μπορεί να είναι ευέλικτη, εύκολα αναγνώσιμη, εύκολο να αυτοματοποιηθεί, πολύ εκφραστική και τέλος επεκτάσιμη.

Το STIX χρησιμοποιεί τα STIX αντικείμενα (STIX objects) σαν γενικό όρο για όλων των ειδών τα τις οντότητες που θέλει να αποτυπώσει και τα συσχετίζει μεταξύ τους ώστε να περιέχει όλη την απαραίτητη πληροφορία που χρειάζεται ένα πλήρες πακέτο Cyber Threat Intelligence. Αυτή τη στιγμή η ενεργή γενιά του STIX που χρησιμοποιείται είναι η STIX 2.1 που προσφάτως άλλαξε από την 2.0. Επίσης αρκετά συχνά εμφανίζεται ο όρος TTP (tactics, techniques, procedures) που χρησιμοποιείται για να περιγράψει πως οι δράστες κυβερνοεπιθέσεων οργανώνουν και φέρνουν εις πέρας τις επιθέσεις τους.

Πιο αναλυτικά τα STIX objects μπορεί να περιέχουν STIX Domain Objects (SDOs) και STIX Relationship Objects (SROs).

Τα STIX Domain Objects (SDOs) που αποτελούν το STIX 2.1 είναι:

Attack Pattern: Ενός είδους TTP που περιγράφει τον τρόπο ή τρόπους που οι δράστες προσπαθούν να επιτεθούν στους στόχους.

Campaign: Μια σύνθεση από επιθετικές συμπεριφορές που περιγράφουν ένα κύμα από κακόβουλες ενέργειες μέσα σε ένα συγκεκριμένο χρονικό διάστημα.

Course of Action: Μια πρόταση από έναν παραγωγό Cyber Threat Intelligence προς τον αποδέκτη της, περιγράφοντας τι ενέργειες μπορεί να ακολουθήσει σαν αντίδραση για αυτή τη γνώση (CTI) που πήρε.

Grouping: Δηλώνει ότι τα STIX objects που βρίσκονται στις αναφορές (references) είναι υποχρεωτικό να έχουν περιεχόμενο γύρω από κοινό θέμα.

Identity: Πραγματικά άτομα, επιχειρήσεις, ομάδες (π.χ. IETF), ή και γενικότεροι τομείς ομάδων όπως τομέας οικονομίας.

Indicator: Ίσως το σημαντικότερο STIX Domain Object. Δηλώνει το υπόδειγμα που δείχνει την πιθανότητα ύπαρξης ύποπτης ή κακόβουλης συμπεριφοράς.

Infrastructure: Ενός είδους TTP που περιέχει οποιαδήποτε συστήματα, υπηρεσίες λογισμικού ή γενικά οποιοδήποτε εφόδιο που χρησιμοποιείται (όχι απαραίτητα κακόβουλα).

Intrusion Set: Ένα σύνολο από επιθετικές συμπεριφορές και εφόδια, με κοινά χαρακτηριστικά που πιστεύεται ότι προέρχονται από το ίδιο άτομο ή οργάνωση.

Location: Δείχνει μια γεωγραφική τοποθεσία.

Malware: Ενός είδους TTP που περιγράφει κώδικα με κακόβουλο χαρακτήρα.

Malware Analysis: Τα μεταδεδομένα και αποτελέσματα από μια ανάλυση που έγινε σε ένα Malware.

Note: Προσφέρει επιπλέον πληροφορίες σχετικά με την περιγραφόμενη γνώση σε αυτό το STIX Object.

Observed Data: Προσφέρει πληροφορία σχετικά με οντότητες που συσχετίζονται με την ασφάλεια από κυβερνοεπιθέσεις. Π.χ. αρχεία, συστήματα, δίκτυα κτλ.

Opinion: Μια κριτική για το συγκεκριμένο STIX Object.

Report: Μια συλλογή από Threat Intelligence που επικεντρώνεται σε ένα ή περισσότερα θέματα, όπως ένας κακόβουλης χρήστης, ένα malware, μια τεχνική επίθεσης κ.α.

Threat Actor: Επίσης πολύ σημαντικό STIX Domain Object. Περιέχει ένα άτομο ή μια οργάνωση που πιστεύεται ότι είναι ασκούν κακόβουλες ενέργειες.

Tool: Εργαλείο που μπορεί να έχει χρησιμοποιηθεί ώστε οι δράστες να φέρουν εις πέρας μια κυβερνοεπίθεση.

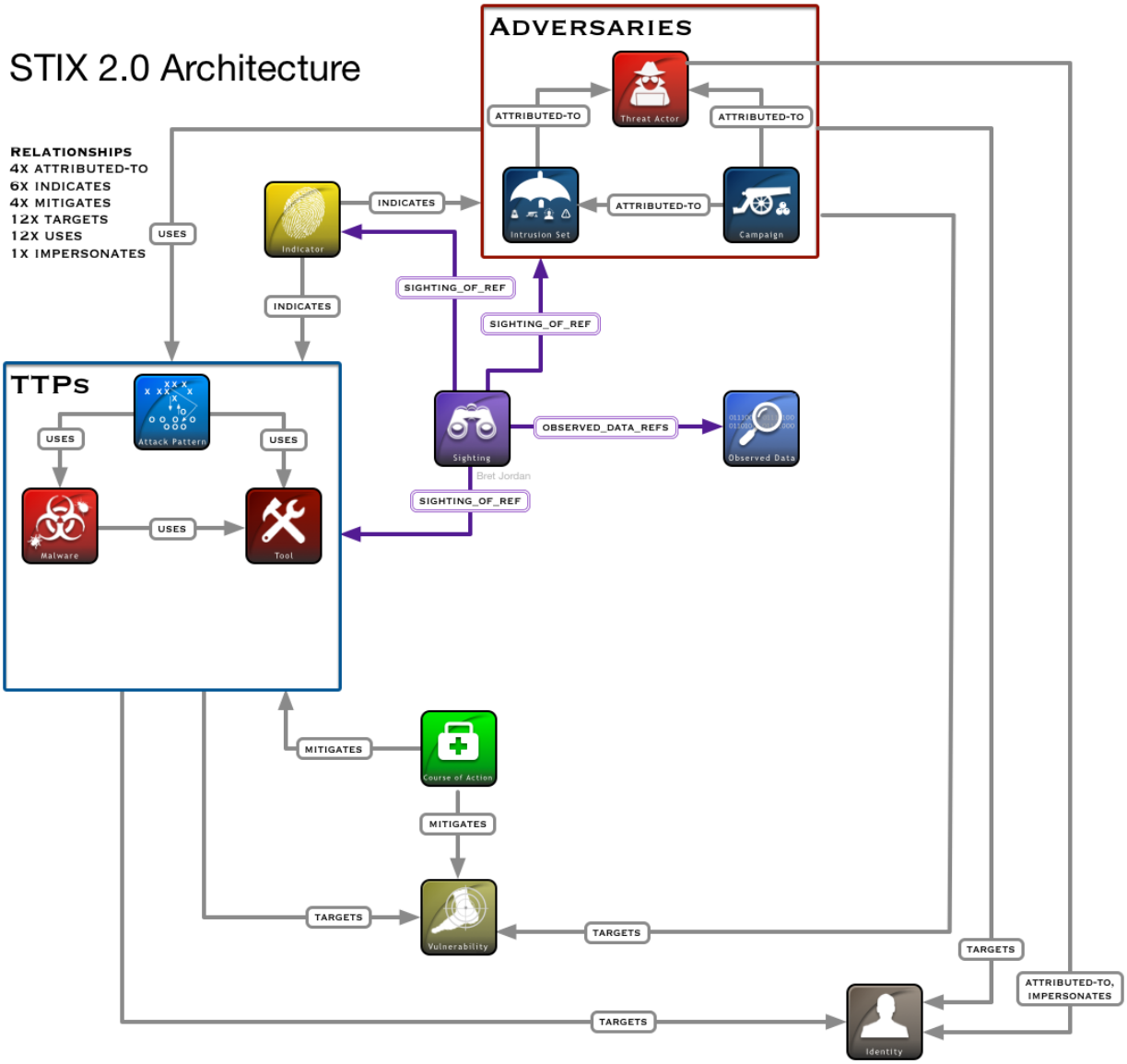
Vulnerability: Μια ‘τρύπα’ ασφάλειας ή μια αδυναμία που έχει ένα λογισμικό που μπορεί να χρησιμοποιηθεί από τον δράστη ώστε να πάρει πρόσβαση σε ένα σύστημα.

Τα STIX Relationship Objects (SROs) που αποτελούν το STIX 2.1 είναι:

Relationship: Χρησιμοποιείται για να ενώσει δύο STIX Domain Objects ή STIX Relationship Objects και περιγράφει με ποια ακριβώς έννοια συνδέονται μεταξύ τους.

Sighting: Δηλώνει μια υπόθεση, ότι αυτή η γνώση (π.χ. ένας δράστης, ένα malware, ένα εργαλείο κ.α.) έχει ξαναπαρουσιαστεί κάπου στο παρελθόν.

Έχοντας περιγράψει τα περιεχόμενα των STIX Objects (SDOs, SROs) μπορούμε να δούμε ένα παράδειγμα της γενικής αρχιτεκτονικής του STIX. [Σχήμα 8]



Σχήμα 7: Αρχιτεκτονική STIX 2.0 [38]

Τα STIX Objects περιγράφονται με τη γλώσσα JSON και ένα απλό παράδειγμα μοιάζει κάπως έτσι:

```
{
      "type": "campaign",
      "id": "campaign--8e2e2d2b-17d4-4cbf-938f-
98ee46b3cd3f",
      "spec_version": "2.1",
      "created": "2016-04-06T20:03:00.000Z",
      "modified": "2016-04-06T20:03:23.000Z",
      "name": "Green Group Attacks Against Finance",
      "description": "Campaign by Green Group against
targets in the financial services sector."
}
```

3.2.2 OpenIOC (Open Indicators Of Compromise)

Τα τελευταία χρόνια, τα Forensics αποτελούν ένα πολύ σημαντικό κομμάτι στον τομέα της ασφάλειας στην επιστήμη της πληροφορίας. Μέσω της διερεύνησης που διεξάγεται όταν ακολουθούνται πρωτόκολλα και μεθοδολογίες forensics μετά από κάποια κυβερνοεπίθεση, είναι εφικτό να γίνει η κατάλληλη καταγραφή και εξακρίβωση του συμβάντος που πήρε μέρος σε κάποιον οργανισμό. Από την αρχή της ανάπτυξης του τομέα των forensics, ο κύριος στόχος των μεθοδολογιών ήταν να βρεθούν στοιχεία που αποδείκνυαν ότι όντως είχε συμβεί κάποια επίθεση από τον κυβερνοχώρο, τι είδους επίθεση ήταν, τι ζημιές προκάλεσε, και μέσα από όλα αυτά να γίνει η εύρεση του καλύτερου εφικτού τρόπου να ορθοποδήσει και πάλι το συντομότερο δυνατό ο οργανισμός.

Η αγορά που κινείται γύρω από την τεχνολογία των forensics κυμαίνεται σε μεγάλες τιμές και πολλές φορές δεν είναι προσεγγίσιμες από μικρομεσαίους οργανισμούς. Δεδομένου ότι οι υπηρεσίες forensics κόστιζαν μεγάλα χρηματικά ποσά, αυτοί οι οργανισμοί που τα πλήρωναν είχαν και τον κύριο λόγο στην κατεύθυνση που θα κινηθεί ο συγκεκριμένος τομέας, καθώς κι αυτοί τον κρατούσαν ζωντανό. Οι κύριοι στόχοι των μεγάλων οργανισμών μετά από κάποια επιτυχημένη κυβερνοεπίθεση που δέχθηκαν, εκτός από το να ορθοποδήσουν, είναι να γίνει και

η συλλογή των απαραίτητων στοιχείων ώστε να καταφέρουν να αποδείξουν δικαστικά ότι όντως υπήρξαν θύματα επίθεσης, αν χρειαστεί.

Η τεχνολογία των forensics τελικά όμως άρχισε σιγά σιγά να βοηθά πολύ και στο κομμάτι της ανίχνευσης για το αν κάποιο σύστημα έχει δεχτεί κάποια επιτυχημένη επίθεση. Αφού ξεκινήσουν οι διαδικασίες forensics σε ένα σύστημα, μπορούν να συλλεχθούν και πληροφορίες που επιδεικνύουν τα χνάρια που άφησε πίσω της η κάθε επίθεση. Μαθαίνοντας τα στοιχεία που αφήνει πίσω μια συγκεκριμένη επίθεση, γίνεται πολύ ευκολότερο να βρεθούν και άλλα συστήματα που έχουν δεχτεί την επίθεση αυτή, ψάχνοντας απλά, για τα ίδια σημάδια που είχε το πρώτο θύμα.

Αυτά τα σημάδια που δείχνουν ότι κατά πάσα πιθανότητα μια κυβερνοεπίθεση έχει προσπαθήσει και έχει πετύχει να δράσει πάνω σε ένα σύστημα, ονομάζονται Indicators of Compromise (IOCs). Μέσω των IOCs, ψάχνοντας για συγκεκριμένα στοιχεία και σημάδια, επιτυγχάνεται δραστική μείωση του χρόνου ανίχνευσης των επιτυχώς παραβιασμένων συστημάτων.

Η σειρά γεγονότων που ακολουθούνταν επί πολλά χρόνια σε περιπτώσεις μεγάλων συμβάντων κυβερνοεπίθεσης σε έναν οργανισμό ήταν η εξής: Ο οργανισμός αντιλαμβανόταν ότι έχει παραβιαστεί. Γινόταν η συλλογή των απαραίτητων στοιχείων για δικαστική κάλυψη. Γραφόντουσαν τεράστιες αναφορές του συμβάντος και τέλος ο οργανισμός έμπαινε σε στάδιο ανάκαμψης. Ο διαμοιρασμός του IOC για αυτή την κυβερνοεπίθεση θα γινόταν όταν αυτές οι αναφορές έπεφταν στα χέρια υπαλλήλων από άλλους οργανισμούς, που αφιερώνοντας τον απαραίτητο χρόνο θα τις διάβαζαν. Όμως από τη στιγμή που η επίθεση πρωτοεμφανίστηκε μέχρι να φτάσει η γνώση για αυτήν σε άλλους οργανισμούς είχε περάσει πολύς χρόνος με αποτέλεσμα ο δράστης να είχε τον χρόνο που χρειαζόταν ώστε να αλλάξει και να προετοιμαστεί ή να παραβιάσει και άλλους οργανισμούς με τον ίδιο τρόπο.

Το κλειδί λοιπόν δεν είναι απλά η ικανότητα να διαμοιράζεται αυτή η γνώση, αλλά να διαμοιράζεται και το γρηγορότερο δυνατό, ώστε να γίνονται με τη σειρά τους οι έλεγχοι των πιθανά παραβιασμένων συστημάτων εξίσου όσο το γρηγορότερο δυνατό. [39]

Η Mandiant, έφερε λύση σε αυτό το πρόβλημα και την ονόμασε OpenIOC. Το OpenIOC είναι ένα open standard format για την καταγραφή, τον ορισμό και τον διαμοιρασμό των IOC σε γλώσσα μηχανής από τον έναν οργανισμό στον άλλο εξαιρετικά γρήγορα.

Το OpenIOC είναι ένα ευέλικτο φορμάτ, με αποτέλεσμα να μπορεί να αλλάζει καθώς συλλέγεται επιπλέον πληροφορία. Αποτελεί έναν πολύ καλό τρόπο να μαζευτεί γνώση και πληροφορία για μια κυβερνοεπίθεση από διάφορες πηγές, ανθρώπινες και μη, και έπειτα να χρησιμοποιηθεί το αποτέλεσμα της παραπάνω διαδικασίας ώστε να γίνουν μαζικές σαρώσεις οργανισμών και να βρεθούν συστήματα που πιθανόν να έχουν παραβιαστεί από την ίδια κυβερνοεπίθεση.

Το OpenIOC χρησιμοποιεί την γλώσσα XML (Extensible Markup Language). Η XML, καθώς είναι μια από τις πιο ευρέως χρησιμοποιημένες γλώσσες για την κωδικοποίηση δεδομένων σε γλώσσα βολική για τις μηχανές, παρέχει διάφορα πλεονεκτήματα. Το κυριότερο από αυτά είναι ότι είναι από τη φύση τους, γλώσσες όπως η XML και η JSON είναι πολύ επεκτάσιμες. Εξαιτίας αυτού, παρόλο που η δομή ενός OpenIOC αντικειμένου είναι μικρή, μέσα μπορούν να μπουν όσα indicator σετ χρειαστεί ώστε να περιγραφεί πλήρως μια πλήρης παραβίαση.

3.2.2.1 Λειτουργικότητα των IOC

Για το OpenIOC, υπάρχουν ήδη πάνω από 500 διαφορετικούς τύπους στοιχείων παραβίασης που μπορούν να συλλεχθούν από έναν οργανισμό, καλύπτοντας ένα πολύ μεγάλο φάσμα επιθέσεων. Οι indicators ποικίλουν από τις απλές τεχνικές ανίχνευσης, χρησιμοποιώντας αθροίσματα ελέγχου MD5, μεγέθη αρχείων, ονόματα, διαδρομές αρχείων (file paths), κλειδιά μητρώου (registry keys) κ.α., μέχρι και τις πιο σύνθετες τεχνικές, που οι δράστες δεν μπορούν να κάνουν τόσο εύκολα ελιγμούς τριγύρω τους, όπως εκτελέσιμες εφαρμογές στον πυρήνα, τον φόρτο του επεξεργαστή κατά τη διάρκεια της παραβίασης κ.α.

Όλα αυτά μπορούν να συνδυαστούν μεταξύ τους σε ένα OpenIOC ώστε να δημιουργηθούν έξυπνες και όσο το δυνατόν πιο επεξηγηματικές περιγραφές των στοιχείων που υποδεικνύουν παραβιάσεις που συνέβησαν ήδη σε ένα σύστημα, αλλά και παραβιάσεις που βρίσκονται εκτελούνται τώρα. [39]

Μερικά παραδείγματα κλασικής περιγραφής indicators σε ένα OpenIOC αντικείμενο είναι τα εξής:

- Αναζήτηση για ένα συγκεκριμένο MD5 άθροισμα ελέγχου σε αρχεία
- Αναζήτηση για μια συγκεκριμένη διεργασία στη μνήμη με γνωστά κακόβουλα χαρακτηριστικά.
- Αναζήτηση για γνωστά κακόβουλα κλειδιά μητρώου

- Συνδυασμοί όλων των παραπάνω

Μια πιο σύνθετη περιγραφή σε ένα OpenIOC αντικείμενο θα ήταν να δημιουργηθεί μια λίστα με όλα τα γνωστά αρχεία που πρέπει να υπάρχουν σε ένα φάκελο συστήματος σε ένα λειτουργικό σύστημα (π.χ. system32) και αντί να πραγματοποιηθεί αναζήτηση για κακόβουλα αρχεία, να γίνει αναζήτηση για οποιοδήποτε αρχείο που είναι εκεί ενώ δεν θα έπρεπε υπό φυσιολογικές συνθήκες.

Πολύ αποδοτικό αλλά ταυτόχρονα και πολυσύνθετο είναι να επικεντρώνονται οι indicators ενός OpenIOC στη μεθοδολογία του πιθανού επιτιθέμενου και όχι στα συγκεκριμένα σημάδια που αφήνει πίσω του. Παραδείγματος χάρη, ένα OpenIOC θα μπορούσε να χρησιμοποιηθεί ώστε να αναζητηθούν σημάδια που δεν επιδεικνύουν απαραίτητα κάποιου είδους παραβίαση αλλά μια απλή έκρυθμη συμπεριφορά σε δικτυακές υπηρεσίες ή στον ρυθμό εισόδου / εξόδου ενός δίσκου. Με αυτόν τον τρόπο, θα “ακολουθήσει” τα βήματα που έκανε κάποια κακόβουλη διεργασία που πιθανόν να βρίσκεται στο σύστημα.

3.2.2.2 Συγγραφή αποτελεσματικών IOCs

Καθώς η ευελιξία των IOC είναι ένα από τα μεγαλύτερα τους πλεονεκτήματα, μπορεί να φτάσει στο σημείο που καθιστά κάποια από αυτά άχρηστα έως και πολύ βλαβερά. Το σημαντικό στην αποτελεσματικότητα ενός μηχανισμού εντοπισμού παραβιάσεων, δεν είναι η ποσότητα των στοιχείων που επιδεικνύουν την εκάστοτε παραβίαση που θα εντοπιστεί αλλά η δυνατότητα ύπαρξης του χαμηλότερου δυνατού ποσοστού false positive συμβάντων.

Ένα IOC που απλά θα περιέχει κάτι όπως “something OR filename = *.exe” είναι σίγουρο ότι θα χτυπήσει σε όλα τα εκτελέσιμα προγράμματα ενός Windows λειτουργικού συστήματος, αυτό όμως δεν σημαίνει ότι όλα τα εκτελέσιμα σε εκείνο το λειτουργικό σύστημα είναι σημάδια παραβίασης.

Τρεις γενικοί κανόνες για τη συγγραφή καλών και αποτελεσματικών indicators μέσα σε OpenIOC είναι οι εξής:

- Τα IOCs να ανιχνεύουν μόνο στοιχεία που συνδέονται με συμπεριφορές παραβίασης.

- Να μην κοστίζουν πολύ κατά την εφαρμογή τους. Δεν είναι συνετό να κοστίζουν πολύ χρόνο ή επεξεργαστική ισχύ στα συστήματα που ερευνούν.

- Να είναι φτιαγμένα με τέτοιο τρόπο, ώστε να δυσκολεύουν πολύ τον δράστη από το να κάνει ελιγμούς και να κρύψει τα σημάδια που αφήνει πίσω του.

3.2.2.3 Κύκλος ζωής των IOC και αξιοποίηση τους

Ο κύκλος ζωής των IOCs σε ένα λειτουργικό περιβάλλον είναι επαναληπτικός. Χάρη στην ευελιξία που έχουν και στην φύση της γλώσσας μηχανής που διατηρούν, η MANDIANT έχει σχεδιάσει τον πιο αποδοτικό τρόπο λειτουργία τους σε ένα εργασιακό περιβάλλον, που οι ίδιοι χρησιμοποιούν και στηρίζουν μέχρι και σήμερα. Τα στάδια του κύκλου ζωής των IOCs είναι τα εξής:

- Εμφάνιση αρχικών στοιχείων
  - ❖ Σε πρώτο στάδιο, στοιχεία παραβίασης θα ανιχνευτούν σε κάποιο σύστημα ή στο δίκτυο ενός οργανισμού. Αυτά μπορεί να προέρχονται από διάφορες πηγές. Παρόλα αυτά, σίγουρα θα πρέπει να είναι αναμφισβήτητα σημάδια παραβίασης.

- Δημιουργία IOCs
  - ❖ Μετά από την ανίχνευση των αρχικών στοιχείων παραβίασης, αυτά τα στοιχεία θα χρησιμοποιηθούν για τη συγγραφή των IOCs που θα τα αντιπροσωπεύουν. Η ποιότητα των IOC που θα γραφτούν εξαρτάται καθαρά από την εμπειρία του μηχανικού που θα τα γράψει και την εμπειρία του μηχανικού που εντόπισε τα αρχικά στοιχεία παραβίασης.

- Αξιοποίηση των IOC που δημιουργήθηκαν
  - ❖ Μετά τη δημιουργία των IOCs, ο κατάλληλος μηχανικός θα τα αξιοποιήσει και θα τα χρησιμοποιήσει για να κάνει ένα πέρασμα του υπόλοιπου οργανισμού με αυτά. Αυτό μπορεί να γίνει με πολλούς τρόπους, αναλόγως με τα λογισμικά ασφάλειας που χρησιμοποιεί ο εκάστοτε οργανισμός. Επίσης μπορεί να

μοιραστεί και αυτά τα IOCs με το διαδίκτυο ή άλλους οργανισμούς με το OpenIOC.

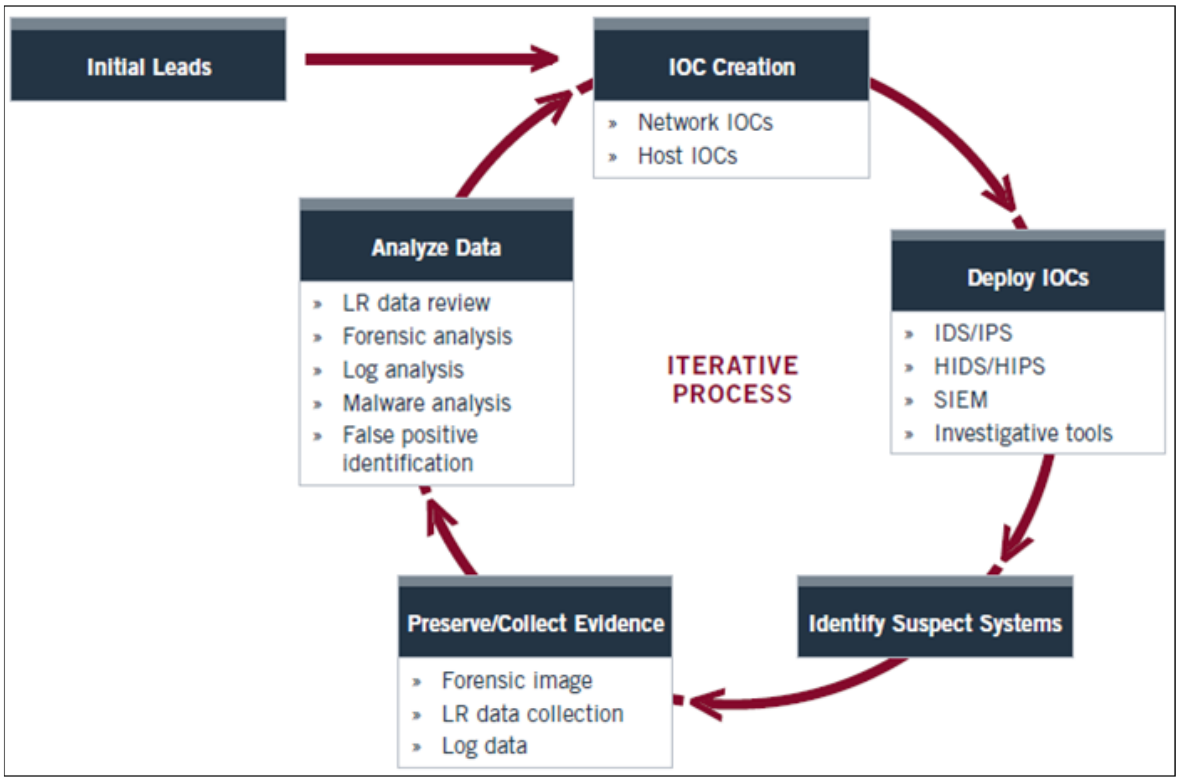
- Εύρεση και άλλων ύποπτων συστημάτων
  - ❖ Με βάση τα IOCs που δημιουργήθηκαν προηγουμένως, ίσως έχουν βρεθεί κι άλλα συστήματα που έχουν τα ίδια στοιχεία παραβίασης στον οργανισμό.

- Συλλογή στοιχείων
  - ❖ Εφόσον κι αν βρεθούν κι άλλα συστήματα στο προηγούμενο βήμα, θα γίνει η συλλογή στοιχείων και από αυτά.

- Ανάλυση νέων στοιχείων που βρέθηκαν.
  - ❖ Αυτά τα νέα στοιχεία μπορεί να υποδείξουν την ύπαρξη κι άλλων παραβιάσεων, ή κι άλλα σημάδια που σχετίζονται με την ίδια παραβίαση, ή να βρεθούν και να αντιμετωπιστούν ενδεχόμενα false positives.

- Αναδιάρθρωση και δημιουργία νέων IOCs
  - ❖ Τώρα μπορούν να δημιουργηθούν καινούργια IOCs ή να βελτιωθούν τα ήδη υπάρχοντα με καινούργια σημάδια και στοιχεία που έχουν προκύψει από τα προηγούμενα βήματα. Σε αυτό το βήμα οδηγούμαστε πίσω στο δεύτερο βήμα, από το οποίο η επανάληψη του κύκλου ζωής των IOCs ξεκινά.

Ο κύκλος αυτός θα συνεχιστεί με τον εξής τρόπο: [Σχήμα 9]



Σχήμα 8: Κύκλος ζωής IOCs

3.2.3 TAXII (Trusted Automated eXchange of Intelligence Information)

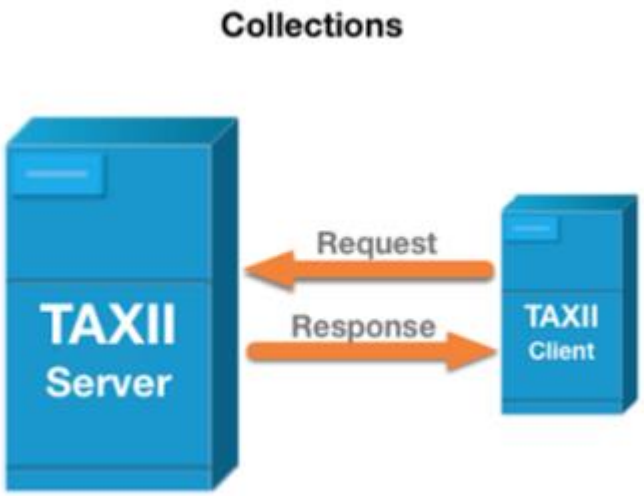
Το TAXII (Trusted Automated eXchange of Intelligence Information, TAXII) είναι μια υπηρεσία που ζει στο επίπεδο εφαρμογής του μοντέλου OSI, και παρέχει μεταφορά και ανταλλαγή Cyber Threat Intelligence μεταξύ συστημάτων και οργανισμών. Το TAXII χρησιμοποιεί το πρωτόκολλο HTTPS ώστε να παρέχει ασφάλεια στις συναλλαγές που πραγματοποιούνται.

Το TAXII είναι σχεδιασμένο ώστε οι TAXII εξυπηρετητές να παρέχουν ένα RESTful API για τους TAXII clients ώστε να γίνονται εύκολα μέσω HTTP/HTTPS μεθόδων οι συναλλαγές μεταξύ τους. Στην τρέχον κατάσταση ανάπτυξης του TAXII

υπάρχουν δύο κύριες υπηρεσίες που το απαρτίζουν. Τα Collections και τα Channels. [40]

Τα TAXII Collections είναι η βασική μορφή αξιοποίησης της ιδεολογίας πίσω από το TAXII. Ακολουθείται το βασικό μοντέλο πελάτη-εξυπηρετητή με HTTP Request μεθόδους από τους TAXII clients προς του TAXII servers. Οι TAXII servers έχουν αποθηκευμένα Cyber Threat Intelligence δεδομένα και όταν οι clients τα ζητήσουν, οι servers τους δίνουν πακέτα που περιέχουν όση πληροφορία έχει οριστεί κατά την ρύθμιση τους ή όση πληροφορία ζητήσουν οι clients σε περίπτωση που το ορίσουν μέσα στο request που έκαναν.

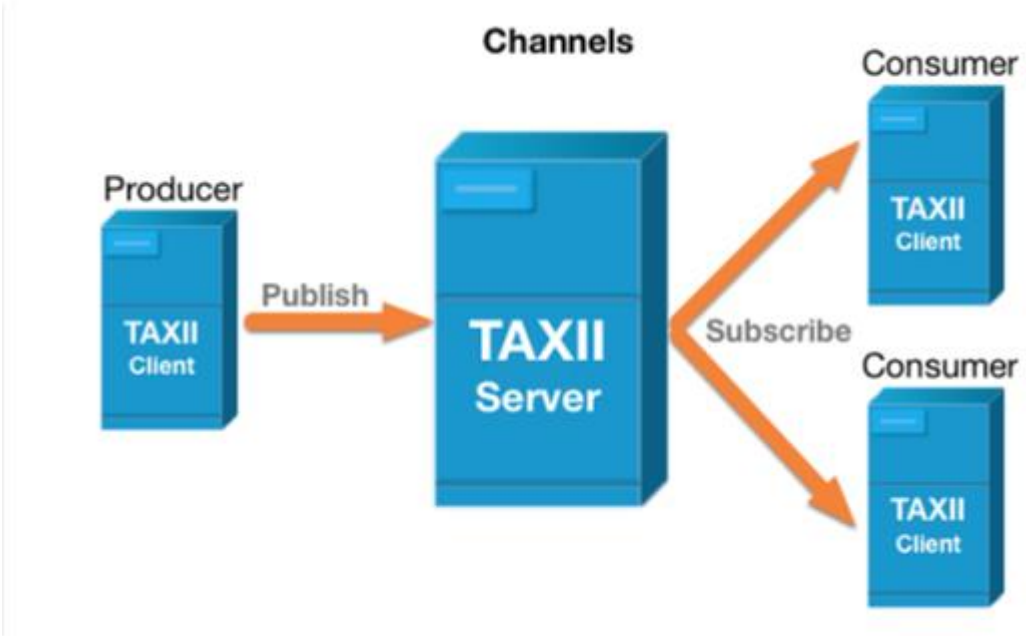
Θεωρητικά TAXII client μπορεί να είναι ένα οποιοδήποτε σύστημα, αλλά συνήθως είναι κάποιο αντιικό λογισμικό (antivirus) ή μηχανισμοί ασφάλειας όπως SIEMs, Firewalls και άλλα. TAXII Servers συνήθως κατέχουν μεγάλοι οργανισμοί που διατηρούν Cyber Threat Intelligence και τη διαμοιράζουν σε όσους την αξιοποιούν.



Σχήμα 9: Αρχιτεκτονική TAXII Collections

Ένα TAXII Channel, ανήκει και διαχειρίζεται από κάποιον TAXII Server. Στην περίπτωση των TAXII Channels η αρχιτεκτονική είναι διαφορετική καθώς πλέον δεν γίνονται αιτήματα προς τους servers για να δώσουν CTI στους clients, αλλά μπαίνουν στη γενική εικόνα οι παραγωγοί CTI (TAXII Producers) και οι συνδρομές (TAXII Subscriptions).

Στην αρχιτεκτονική των TAXII Channels, υπάρχουν ειδικοί παραγωγοί CTI που εκδίδουν και δίνουν το Cyber Threat Intelligence που παράγουν στους TAXII servers. Επιπρόσθετα, υπάρχουν οι καταναλωτές CTI (TAXII Consumers) οι οποίοι έχουν συνδρομή με έναν ή περισσότερους παραγωγούς. Έτσι οι TAXII servers είναι ο μεσάζων στην συμφωνία που έχουν οι καταναλωτές με τους παραγωγούς CTI. Η παραπάνω αρχιτεκτονική ακολουθεί το σχεσιακό μοντέλο πολλά προς πολλά που σημαίνει ότι ο κάθε καταναλωτής μπορεί να έχει συνδρομή με πολλούς παραγωγούς CTI, και το αντίθετο από την μεριά των παραγωγών. Η αρχιτεκτονική των TAXII Channels, έχει ένα μεγάλο πλεονέκτημα έναντι της αρχιτεκτονικής TAXII Collections. Αυτό είναι ότι ενώ η πηγή του Cyber Threat Intelligence για τους καταναλωτές δεν αλλάζει και είναι ακόμη οι servers, η πηγή της γνώσης για τους servers πλέον δεν είναι μόνο οι εαυτοί τους (οι μηχανικοί του οργανισμού που ανήκουν δηλαδή) αλλά έχουν πολλαπλές πηγές γνώσης. Με αυτόν τον τρόπο πολλοί μπορούν να συνεισφέρουν στον διαμοιρασμό Cyber Threat Intelligence προς συμφέρον όλων. Ωστόσο τα TAXII Channels δεν χρησιμοποιούνται καθώς δεν έχουν υλοποιηθεί ακόμη από τον OASIS.



Σχήμα 10: Αρχιτεκτονική TAXII Channels

Το TAXII έχει δημιουργηθεί με στόχο να είναι συμβατό σχεδόν όλα τα πρότυπα επικοινωνίας που χρησιμοποιούν οι οργανισμοί και μπορεί να λειτουργήσει και υλοποιημένο και σε μοντέλα hub-and-spoke και peer-to-peer (P2P). Τέλος, πρέπει να σημειωθεί ότι ενώ το TAXII όντως χρησιμοποιείται για διαμοιρασμό CTI που ακολουθεί το πρότυπο STIX, μπορεί να χρησιμοποιηθεί για οποιαδήποτε μεταφορά δεδομένων ασχέτως της μορφής τους.

3.3 Self-Healing

Το Self-healing πάντα αποτελούσε έναν σύνθετο όρο στην επιστήμη της πληροφορικής. Ένα σύστημα που είναι οπλισμένο με κάποιον Self-healing μηχανισμό, σημαίνει ότι σε περίπτωση που οποιοδήποτε συμβάν που αποτελεί μια βλάβη, με αποτέλεσμα την παύση των υπηρεσιών που προσφέρει το σύστημα, συμβεί, θα αναλάβει ο μηχανισμός Self-healing να προσφέρει την ίαση του. Έτσι θα επαναφέρει το σύστημα σε μια κατάσταση που θα μπορεί να προσφέρει τις λειτουργίες που πρέπει να προσφέρει ακόμα κι αν αυτές δεν αγγίξουν το 100% των προσδοκιών του. [41]

Από τον παραπάνω κατά προσέγγιση ορισμό, προκύπτει ότι το Self-Healing είναι ένας τρόπος να αυξηθεί η ανεξαρτησία των συστημάτων από τους τεχνικούς και τους διαχειριστές τους, αφού θα ήταν αυτόνομα αρκετά ώστε να αντιδρούν στις βλάβες κρατώντας τον εαυτό τους εν λειτουργία. Με βάση αυτό όμως, είναι πολλές οι σκέψεις και οι διχασμοί στην επιστήμη της πληροφορικής γύρω από την εννοιολογία του Self-healing, διότι έρχεται πολύ κοντά με την έννοια και τον σκοπό του γνωστού Fault-tolerance.

Ο A. Avizienis θέτοντας τον ορισμό του fault-tolerance είπε, «Fault-tolerance είναι οι μηχανισμοί και οι τεχνικές που επιτρέπουν σε ένα σύστημα να παρέχει τις υπηρεσίες που είναι υπεύθυνο να παρέχει ακόμη και μετά την ύπαρξη βλαβών» [43]. Οι προαναφερόμενοι διχασμοί βασίζονται πάνω σε αυτό και εκφέρουν την ομοιότητα του Self-healing με τον φόβο ότι το self-healing είναι απλά το fault-tolerance που όλοι ξέρουν τόσα χρόνια [42].

Τα πιο συνηθισμένα παραδείγματα fault-tolerance αρχιτεκτονικών είναι διπλές και τριπλές παροχές ρεύματος σε υποδομές, διπλές και τριπλές γραμμές

σύνδεσης στο διαδίκτυο σε περίπτωση αποτυχίας κάποιας γραμμής, ή η RAID αρχιτεκτονική όταν πρόκειται για ανοχή σφαλμάτων σε σκληρούς δίσκους.

Όλα τα προηγούμενα έχουν ένα κοινό, προσφέρουν την ανοχή σε σφάλματα σε επίπεδο υλικού και αυτές οι λύσεις συνήθως έχουν μεγάλο κόστος.

Το Self-healing εστιάζει στο να πετύχει τον στόχο του χωρίς πλεονάζον πόρους για το σύστημα. Η προσέγγιση ενός μηχανισμού Self-healing είναι διαφορετική, εστιάζοντας στην ανίχνευση του σφάλματος και στην όσο το δυνατό καλύτερη επίλυση του. Μετά από την ίαση ενός Self-healing μηχανισμού δεν είναι σίγουρο ότι το σύστημα θα επανέλθει στο 100% της λειτουργίας του, αλλά φροντίζει όσο καλύτερα μπορεί να κάνει τα απαραίτητα για να συνεχίσει το σύστημα που έχει υποστεί βλάβη να παρέχει τις υπηρεσίες του.

Το Self-healing στηρίζεται σε μηχανισμούς και πρωτόκολλα όπως το Open Command and Control (OpenC2), το Common Vulnerabilities and Exposures (CVE) και το Open Vulnerability and Assessment Language (OVAL).

3.3.1 OpenC2 (Open Command and Control)

Όλο και γρηγορότερα, με την ανάπτυξη του τομέα της ασφάλειας της πληροφορίας, αναπτύσσονται και οι τεχνικές που εκτελούνται οι διαφόρων τύπων κυβερνοεπιθέσεις. Όσο η εκτέλεση των κυβερνοεπιθέσεων γίνεται πιο φθηνή, πιο εύκολη και πιο αυτοματοποιημένη, τόσο περισσότεροι κακόβουλοι χρήστες του διαδικτύου τείνουν να ελκύονται σε τέτοιου είδους δραστηριότητες.

Το παραπάνω έχει ως αποτέλεσμα, ότι η παροχή ασφάλειας σε συσκευές που προγραμματίζονται στατικά, μια φορά πριν την τοποθέτηση τους, και λειτουργούν στην απομόνωση χωρίς κάποια ιδιαίτερη επίβλεψη (π.χ. κεντρικοί μηχανισμοί ασφάλειας, δικτυακές συσκευές, servers χαμηλού φόρτου), δεν είναι πλέον δυνατή και αποτελεσματική. Η μορφή που στήνονται τα περισσότερα πολυμερή συστήματα που δημιουργούν αυτό το πρόβλημα είναι με ξεχωριστούς μηχανισμούς που επικοινωνούν μεταξύ τους μέσω ανοικτών διεπαφών. Έτσι ακολουθώντας διάφορα πρότυπα ανάπτυξης έργων και αξιοποιώντας διάφορα πρωτόκολλα, δημιουργούνται νοητοί τομείς βάσει της λειτουργίας τους, που αποτελούν ένα τελικό σύστημα, ανεξαρτήτως της γεωγραφικής τους θέσης.

Η OpenC2 (Open Command And Control) είναι μια συνοπτική και εύκολα επεκτάσιμη γλώσσα που παρέχει επικοινωνία από μηχανή σε μηχανή με στόχο να κάνει δυνατό τον έλεγχο μέσω εντολών στα συστήματα και στους μηχανισμούς που είναι απαραίτητος. Μέσω της αρχιτεκτονικής της, η OpenC2 δεν οφείλει να γνωρίζει την δομή ενός συστήματος και τον νοητό “χάρτη” που περιγράφει τη θέση και τη λειτουργία των υποσυστημάτων που το απαρτίζουν. Έτσι η OpenC2 παραμένει μια αγνωστική γλώσσα, καθώς δε γνωρίζει όλα τα κομμάτια των συστημάτων που ελέγχει, τις τεχνολογίες που χρησιμοποιούν και τον τρόπο που επικοινωνούν με τα υπόλοιπα λειτουργικά κομμάτια της ίδιας “συνοικίας” [46].

Πρέπει να σημειωθεί ότι η OpenC2 ή μια γλώσσα όπως αυτή, είναι απαραίτητη ώστε να υπάρξει παροχή ασφάλειας από κυβερνοεπιθέσεις σε συστήματα όπως τα προαναφερόμενα, παρόλα αυτά όμως δεν είναι αρκετή [46]. Άλλες λειτουργικότητες πρέπει επίσης να καλυφθούν, ώστε σε συνεργασία με την OpenC2, να εκτελούνται οργανωμένες αλληλουχίες ενεργειών, παίζοντας τον ρόλο της αντίδρασης σε διάφορα συμβάντα που ίσως πάρουν μέρος σε ένα σύστημα υπό εποπτεία. Οπότε συστήματα με δυνατότητες ανάλυσης, παραγωγή αποφάσεων βάσει διαφόρων μεταβλητών αλλά και συστήματα έξυπνων αισθητήρων, φυσικών ή υλοποιημένων με λογισμικό, είναι απαραίτητα για τον προαναφερόμενο στόχο. Η λειτουργία της OpenC2 είναι πλήρως ανεξάρτητη και δεν καλύπτει καμία από τις δυνατότητες των υπόλοιπων απαραίτητων μηχανισμών.

3.3.1.1 Αρχιτεκτονική OpenC2

Η αρχιτεκτονική του OpenC2 βασίζεται σε δύο κύριες λειτουργίες. Την εντολή (OpenC2 Command) και την απόκριση (OpenC2 Response). Οι εντολές δημιουργούνται από τους παραγωγούς (OpenC2 Producers) και στέλνονται στους καταναλωτές (OpenC2 Consumers), οι οποίοι μετά από τις απαραίτητες ενέργειες, στέλνουν πίσω στους παραγωγούς την απόκριση τους. Είναι σημαντικό να σημειωθεί ότι μια εντολή από έναν παραγωγό μπορεί να απευθύνεται σε πολλούς καταναλωτές ταυτόχρονα.

Ένα OpenC2 Command περιγράφει την ενέργεια που πρόκειται να εκτελεστεί στο συγκεκριμένο σύστημα που απευθύνεται. Ορισμένες φορές το OpenC2

Command θα περιέχει και τον μηχανισμό υπεύθυνο για την εκτέλεση των συγκεκριμένων ενεργειών, τον ενεργοποιητή (OpenC2 Command Actuator).

Τα τέσσερα βασικά κομμάτια που συνθέτουν ένα OpenC2 Command είναι τα εξής:

- Action: Είναι υποχρεωτικό να υπάρχει μέσα σε ένα command και δηλώνει την ενέργεια που πρέπει να εκτελεστεί.

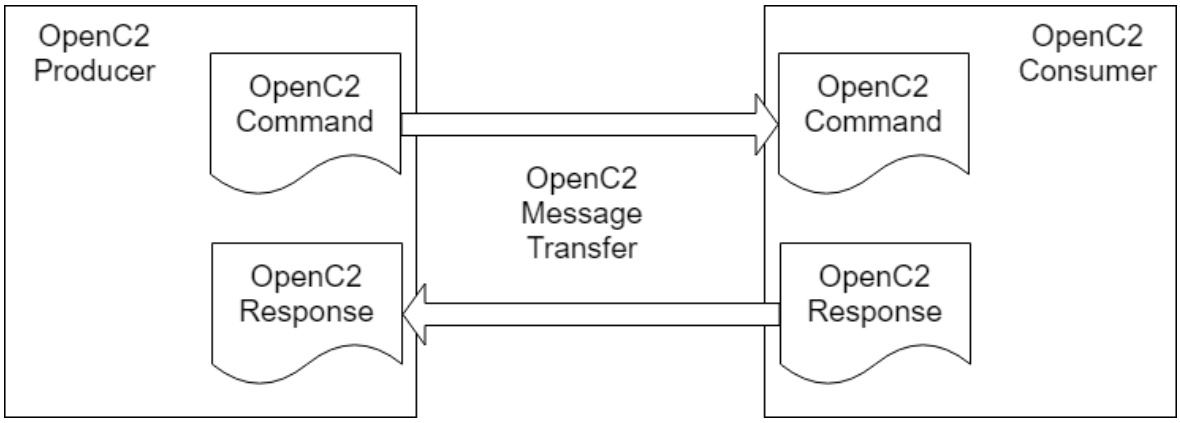
- Target: Είναι επίσης υποχρεωτικό να υπάρχει μέσα σε ένα command και δηλώνει τον δέκτη του command αυτού. Η ενέργεια που περιγράφεται μέσα στο Action θα λάβει μέρος στον στόχο που εκφράζεται στο Target. Μέσα στην ενότητα του target συμπεριλαμβάνονται και επιπλέον πληροφορίες που χαρακτηρίζουν τον δέκτη όσο πιο συγκεκριμένα γίνεται.

- Arguments: Τα Arguments είναι οι παράμετροι που περιγράφουν το πώς πρέπει να εκτελεστεί αυτή η εντολή. Μπορεί να ορίζουν συγκεκριμένη ημερομηνία και χρόνο εκτέλεσης, διάρκεια εκτέλεσης κ.α.. Η χρήση παραμέτρων δεν είναι υποχρεωτική.

- Actuator: Ο Actuator ή ενεργοποιητής εκτελεί την λειτουργία που περιγράφεται στο Action. Το να περιέχεται ο ενεργοποιητής σε κάθε OpenC2 Command, δεν είναι υποχρεωτικό.

Έχοντας μόνο τις πληροφορίες του Action και του Target υποχρεωτικά σε κάθε OpenC2 Command, πολλές λεπτομέρειες της εκτέλεσης της λειτουργίας που περιγράφεται, αφήνονται πλήρως στην κρίση των καταναλωτών.

Τα OpenC2 Responses είναι απλά μηνύματα που στέλνονται από τους καταναλωτές πίσω στον παραγωγό και περιέχουν πληροφορίες όπως αναγνώριση εκτέλεσης εντολής (acknowledgement), κατάσταση συστήματος (status), αποτελέσματα από ένα ερώτημα του παραγωγού (query results) κ.α.. Το μόνο που είναι υποχρεωτικό για τα μηνύματα αποκρίσεων είναι να περιέχουν τον κωδικό κατάστασης (status code) που ενημερώνει τον παραγωγό για την επιτυχημένη ή ανεπιτυχής εκτέλεση της εντολής που αιτήθηκε.



Σχήμα 11: Βασική απεικόνιση λειτουργίας OpenC2 [46]

3.3.1.2 Παράδειγμα OpenC2 Command και Response

Ένα απλό παράδειγμα μιας εντολής OpenC2 είναι το εξής:

```
{
    "action": "contain",
    "target": {
        "device": {
            "device_id": "9BCE8431AC106FAA3861C7E771D20E53"
        }
    }
}
```

Στο παραπάνω περιγράφεται μια εντολή προς μια συσκευή ασφάλειας δικτύων (π.χ. firewall) ώστε το μηχάνημα με ID “9BCE8431AC106FAA3861C7E771D20E53” να απομονωθεί από το δίκτυο που ελέγχει η συσκευή καταναλωτής.

Η απόκριση στο παραπάνω command θα μοιάζει κάπως έτσι αν είναι θετική:

```
{
    "status": 200,
}
```

Σε περίπτωση μη επιτυχημένης εκτέλεσης του OpenC2 Command η απόκριση μπορεί να είναι κάπως έτσι:

```
{
    "status": 103,
    "status_text": {
        "Failure Reason": "Device seems to be offline"
    }
}
```

Σε αυτήν την απόκριση ο καταναλωτής αναφέρει ότι δεν πέτυχε να ολοκληρώσει την δουλειά που του ανέθεσε ο παραγωγός με την αιτιολόγηση ότι “η συσκευή φαίνεται να είναι εκτός δικτύου”.

3.3.2 Common Vulnerabilities and Exposures (CVE)

Το CVE (Common Vulnerabilities and Exposures) είναι ένας τρόπος να αποτυπώνεται η γνώση για γνωστές τρωτότητες που ανακαλύπτονται σε συστήματα υπολογιστών. Το CVE είναι πρακτικά ένα μεγάλο λεξικό ή μια μεγάλη γνωσιακή βάση που αποτελείται από CVE Identifiers. Τα CVE Identifiers βοηθούν στην ευκολότερη μετακίνηση της πληροφορίας από σύστημα σε σύστημα και από οργανισμό σε οργανισμό ώστε να χρησιμοποιείται η γνώση από πολλά εργαλεία ασφάλειας ταυτόχρονα.

3.3.2.1 Ανάγκη ύπαρξης CVE

Κατά την περίοδο της κατακόρυφης ανάπτυξης της τεχνολογίας, γύρω στο 1997, τα περισσότερα εργαλεία επικεντρωμένα στην ασφάλεια της πληροφορίας χρησιμοποιούσαν δικές τους ιδιωτικές βάσεις γνώσης που οι τρωτότητες και αδυναμίες που είχαν βρει βρισκόντουσαν αποθηκευμένες με ένα δικό τους ξεχωριστό για αυτά τα εργαλεία όνομα. Τα προϊόντα ασφάλειας τότε ήταν ακόμη λίγα και λόγω της μη ύπαρξης συνεργασίας μεταξύ τους, συνήθως υπήρχαν οι ίδιες ευπάθειες σε διαφορετικές βάσεις με άλλα ονόματα και χαρακτηριστικά. Αυτό σιγά σιγά όταν το πνεύμα της συνεργασίας κυριάρχησε στον τομέα της ασφάλειας γύρω στο 1999, άρχισε να δημιουργεί πρόβλημα στις προσπάθειες διαφορετικών εργαλείων να συνεργαστούν, αλλά το μεγαλύτερο πρόβλημα ήταν ότι το κάθε

εργαλείο χρησιμοποιούσε διαφορετικούς τρόπους και ορισμούς για να βαθμολογήσει ένα σύστημα με βάση το πόσο ασφαλές είναι. Αυτό ως αποτέλεσμα είχε να είναι αδύνατη η συγκριτική αξιολόγηση των εργαλείων μεταξύ τους, πράγμα το οποίο κρατούσε πίσω τον ανταγωνισμό μεταξύ τους και συνεπάγεται την αγορά.

Το CVE λύνει αυτά τα προβλήματα, προσφέροντας ένα πρότυπο ώστε όλα τα εργαλεία ασφάλειας και όλοι οι οργανισμοί να γράφουν και να αποτυπώνουν τη γνώση σχετικά με τις ευπάθειες που συναντούν με μια κοινή γλώσσα. Έτσι η συνεργασία μεταξύ τους είναι εύκολη και καρποφόρα. Επίσης το CVE προσφέρει επίπεδο αναφοράς ή αλλιώς μια γραμμή βάσης ώστε να μπορούν οι χρήστες να αξιολογήσουν το κάθε εργαλείο που ερευνά και ανιχνεύει ευπάθειες στα συστήματα τους και να διαλέξουν ποιο ταιριάζει περισσότερο σε αυτούς ή στον οργανισμό τους. Εν ολίγοις, τα προϊόντα που είναι συμβατά με το CVE και το χρησιμοποιούν έχουν μεγαλύτερη γνωσιακή βάση από άλλα και είναι πολύ ευκολότερο να συνδεθούν και να συνεργαστούν με άλλα προϊόντα ασφάλειας.

3.3.2.2 Δημιουργία CVE

Κάθε vulnerability που ανακαλύπτεται εκφράζεται ως ένα CVE Identifier. Η διαδικασία είναι η εξής [45]:

- Όποιος χρήστης ή οργανισμός ανακαλύψει ένα καινούργιο vulnerability, συντάσσει μια αναφορά (report) που να το περιγράφει.
- Ο CVE Numbering Authority (CNA) του προσδίδει ένα μοναδικό CVE Identifier.
- Ο CVE Editor το προσθέτει στην λίστα με τα CVE Identifiers στην ιστοσελίδα με τις CVE λίστες. (έως το 2020, https://cve.mitre.org/cve/)
- Από εκεί είναι διαθέσιμο για όλο τον κόσμο.

Τον ρόλο των CVE Numbering Authority και CVE Editor μέχρι και το 2020 τον έχει η MITRE.

3.3.2.3 Αξιοποίηση CVE

Έως και το 2020, τα CVE Identifiers αξιοποιούνται παγκοσμίως από πολλές υπηρεσίες και προϊόντα. Μερικά από τα είδη των προϊόντων που μπορούν να αξιοποιήσουν CVE είναι βάσεις ευπαθειών (vulnerability databases), συστήματα συμβουλών ασφαλείας (security advisory systems), συστήματα ελέγχων ευπαθειών (vulnerability scanners), αντιικά προγράμματα (antiviruses) και από πιο σύνθετους μηχανισμούς όπως συστήματα SIEM και μηχανισμούς Self-healing.

Η Εθνική Βάση Δεδομένων Ευπαθειών των Ηνωμένων Πολιτειών, NVD (National Vulnerability Database), κάνει μια ανάλυση στο κάνε CVE Identifier που διαδίδει και προσθέτει μια CVSS (Common Vulnerability Scoring System) τιμή, η οποία αντιπροσωπεύει τη κρισιμότητα της συγκεκριμένης ευπάθειας που περιγράφεται στο κάθε CVE Identifier.

Επιπρόσθετα, τα CVE Identifiers συνοδεύουν πολλά άρθρα ειδήσεων ασφάλειας αλλά βοηθούν πλέον και στην έγκυρη καταγραφή των εγκλημάτων στον κυβερνοχώρο ώστε να είναι ένα βοήθημα σε δικαστικές υποθέσεις [45].

Κάθε CVE Identifier είναι υποχρεωμένο να περιέχει τα εξής χαρακτηριστικά:

- CVE Identifier Number (π.χ. “CVE-2013-4123”)
- Μια σύντομη περιγραφή της ευπάθειας που αναλύει.
- Την αναφορά της ανακάλυψης της ευπάθειας και όλες τις λεπτομέρειες που την τριγυρίζουν.

Οι Self-healing μηχανισμοί μπορούν να αξιοποιήσουν τα CVE Identifiers ώστε να εμπλουτίζουν τη γνωστική τους βάση σε πραγματικό χρόνο και να είναι πιο έτοιμα να ανταπεξέλθουν σε πιθανά συμβάντα που χρήζουν άμεση αντίδραση.

3.3.2.4 Η κοινότητα πίσω από το CVE

Πίσω από το CVE υπάρχει μια μεγάλη μάζα ανθρώπων σε παγκόσμια κλίμακα που από το 1999 που δημιουργήθηκε μέχρι και σήμερα συνεχίζουν να το αναπτύσσουν ώστε να χρησιμοποιείται από πολλούς οργανισμούς καθημερινά. Οι κυριότερες μορφές μέσα σε αυτήν την μεγάλη κοινότητα που διατηρεί το CVE είναι οι εξής [45]:

- CVE Numbering Authorities (CNAs):

- ❖ Οι CVE Numbering Authorities είναι υπεύθυνοι ώστε να προσδίδεται το συντομότερο δυνατό ένα CVE Identifier σε καινούργιες ευπάθειες ώστε να είναι δυνατή η αναφορά τους παγκοσμίως με μοναδικό τρόπο.

- CVE Editorial Board:
  - ❖ Το CVE Editorial Board αποτελείται από πάρα πολλούς οργανισμούς παγκοσμίως αλλά και από διάφορα εκπαιδευτικά ιδρύματα. Είναι υπεύθυνοι ώστε να ελέγχουν την εγκυρότητα των πληροφοριών που περιέχουν οι νέοι CVE Identifiers και να τους προσθέτουν στη παγκόσμια λίστα.

- CVE Sponsors:
  - ❖ Οι CVE Sponsors είναι αυτοί που στηρίζουν αυτήν την παγκόσμια προσπάθεια. Ο μεγαλύτερος προς το παρόν είναι η US-CERT.

- CVE Compatible Products and Services:
  - ❖ Τα CVE Compatible Products and Services είναι τα προϊόντα και οι υπηρεσίες που έχουν ενσωματώσει την χρήση του CVE μέσα στην δουλειά τους. Χρησιμοποιώντας το και μεταδίδοντας το, βοηθούν σταθερά στην ανάπτυξη του.

3.3.3 Open Vulnerability and Assessment Language (OVAL)

Η Open Vulnerability and Assessment Language (OVAL) είναι μια γλώσσα ή αλλιώς ένα εργαλείο που βοηθά στην έγκαιρη εξάλειψη γνωστών αδυναμιών ασφάλειας που τυχόν υπάρχουν σε συστήματα μέσα σε έναν οργανισμό με έναν πολύ πρωτοπόρο τρόπο.

Η OVAL χρησιμοποιεί την Extensible Markup Language (XML) ώστε να είναι εύκολο να συνδεθεί με άλλα προϊόντα ασφάλειας, καθώς η XML υπάρχει με σκοπό την εύκολη αποτύπωση πληροφοριών με έναν τρόπο έτσι ώστε να είναι απλό να διαβαστούν και να επεξεργαστούν από μια μηχανή.

Ο κορμός της αποτελείται από τα εξής βασικά βήματα [44]:

- Μεγάλοι οργανισμοί όπως NSA και NIST έχουν φτιάξει τις βασικές οδηγίες για τη σωστή ρύθμιση σε συστήματα για να είναι ασφαλή.

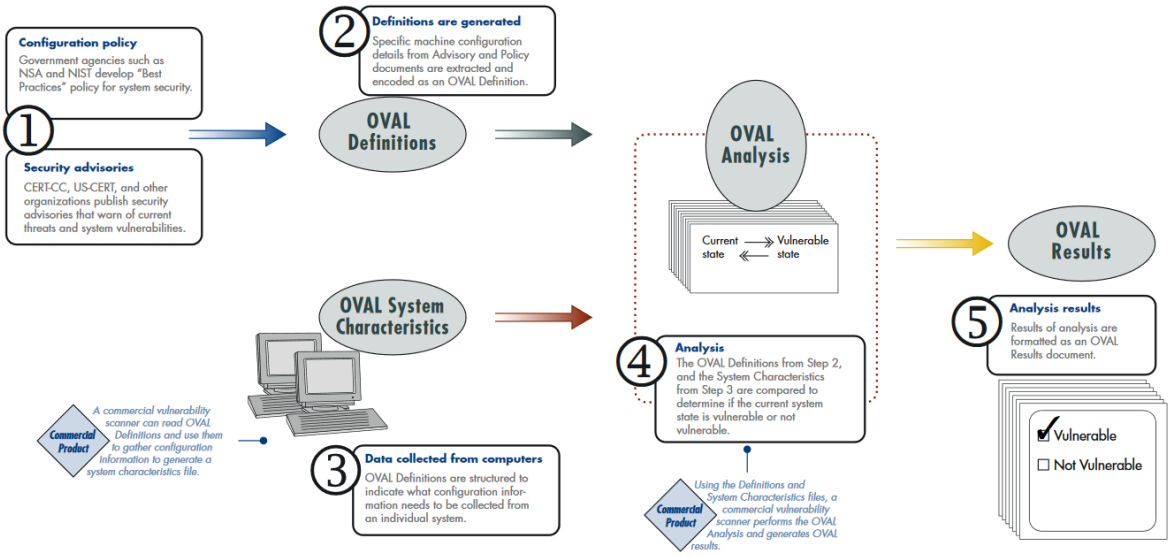
- Μεγάλοι οργανισμοί όπως CERT-CC και US-CERT εκδίδουν συμβουλές ασφάλειας που προειδοποιούν τον κόσμο για νέους κινδύνους στον κυβερνοχώρο και νέες απειλές ή ευπάθειες.

- Από τα παραπάνω παράγονται σε μορφή OVAL κάποιο ορισμοί που υποδεικνύουν πως θα έπρεπε να είναι ένα σύστημα ώστε να μην είναι ευπαθής από μια νέα απειλή.

- Γίνεται καταγραφή της τρέχουσας κατάστασης των συστημάτων που προορίζονται για έλεγχο. Η τρέχον κατάσταση τους αποτυπώνεται με μορφή OVAL.

- Γίνεται έλεγχος σε όποια σύστημα χρειάζεται, συγκρίνοντας την τρέχον κατάσταση του με το πρότυπο παράδειγμα που βγήκε σαν αποτέλεσμα από την μίξη των βασικών οδηγιών και των συμβουλών ασφαλείας από μεγάλους οργανισμούς. Έτσι βρίσκονται οι τρύπες ασφαλείας στα συστήματα που ελέγχονται.

- Βγαίνουν τα αποτελέσματα από τον προηγούμενο έλεγχο σε γλώσσα OVAL. Αυτά περιέχουν τις αδυναμίες του συστήματος που ελέγχθηκε.



Σχήμα 12: Οπτικοποίηση αξιοποίησης OVAL [44]

Η παραπάνω εργασία στην πραγματικότητα είναι τρία απλά XML schemas. Όπου το κάθε ένα αντιστοιχεί απόλυτα σε κάθε ένα από τα τρία βήματα. Πρέπει να αναφερθεί ότι η OVAL δεν είναι ένα εργαλείο που εκτελεί vulnerability assessments αλλά ένα πρότυπο που μπορεί να χρησιμοποιηθεί για να γραφτούν οδηγίες για το πώς ακριβώς να γίνει ένα vulnerability assessment ώστε να βρεθεί με ακρίβεια αν υπάρχει ή όχι ένα vulnerability σε ένα σύστημα.

Η OVAL είναι η πρώτη προσπάθεια που έγινε να αποτυπώνονται σε μια κοινή για όλους γλώσσα λεπτομέρειες και πληροφορίες σχετικά με vulnerabilities και ευπάθειες συστημάτων. Παραμένει ένα open standard από την MITRE ώστε να συνεχίζει να αναπτύσσεται από όλους τους ενδιαφερόμενους και να μην εμπορικοποιηθεί.

Μερικά σενάρια χρήσης της OVAL αναλύονται παρακάτω [44]:

- Vulnerability Assessment
  - ❖ Vulnerability assessment είναι η διαδικασία της έρευνας που γίνεται σε ένα σύστημα για τυχόν τρωτότητες ασφαλείας που έχει. Μέχρι σήμερα οι οργανισμοί που προσφέρουν αυτή την υπηρεσία έχουν αφοσιωμένους

μηχανικούς που με το που ανακαλυφθεί ένα καινούριο vulnerability το μελετούν και το αναλύουν, έπειτα κάνουν διάφορα πειράματα με αυτό και τελικά ανακοινώνουν μια ίαση και έναν τρόπο να καταλάβει κάποιος αν είναι ευπαθής από αυτή την τρωτότητα. Όλα αυτά γίνονται με τρομερή πίεση και το γρηγορότερο δυνατό, καθώς από την στιγμή που μια ευπάθεια ανακοινώνεται παγκοσμίως, μέχρι τη στιγμή που οι μηχανικοί την ασφαλίσουν και οι επηρεαζόμενοι από αυτήν ακολουθήσουν τις οδηγίες των τεχνικών τους και προστατευτούν, υπάρχει ένα παράθυρο χρόνου ώστε και η αντίπαλη μεριά των ατόμων με κακόβουλες προθέσεις να μελετήσει την ευπάθεια, και να γράψει λογισμικό που την αξιοποιεί παραβιάζοντας το σύστημα του θύματος. Όλα τα παραπάνω γίνονται για τον κάθε οργανισμό που προσφέρει τέτοιες υπηρεσίες ξεχωριστά.

Με την χρήση του προτύπου OVAL, οι οργανισμοί θα είχαν μια κοινή παγκόσμια γλώσσα να ώστε να μοιράζονται γνώση μεταξύ τους εύκολα και γρήγορα. Έτσι το πρώτο στάδιο της έρευνας τους θα έπαιρνε πολύ πιο λίγο χρόνο και ίσως μελλοντικά με την κατακόρυφη ανάπτυξη της επιστήμης της Τεχνητής Νοημοσύνης πραγματοποιούταν τελείως η αυτοματοποίηση του πρώτου κομματιού της δουλείας τους, εφόσον το κάθε σύστημα που ερευνά θα μπορούσε να επικοινωνήσει με άλλα στην ίδια γλώσσα και να παραχθούν τα αποτελέσματα σε πολύ λιγότερο χρόνο. Επίσης θα μπορούσαν να μοιράζονται στην κεντρική αποθήκη με ευπάθειες εκφρασμένες με OVAL της MITRE, τις ανακαλύψεις των συνεχών ερευνών τους, ώστε να δημιουργηθεί ένα καλύτερο πλέγμα γνώσης που είναι και αξιοποιήσιμο ενάντια σε ευπάθειες συστημάτων παγκοσμίως.

- Version Management
  - Ο σωστός έλεγχος, καταγραφή και οργάνωση των εκδόσεων των λογισμικών που χρησιμοποιεί ένας οργανισμός ήταν πάντα μια από τις μεγαλύτερες προκλήσεις που είχε ώστε να υπάρξει η απαραίτητη ασφάλεια από ευπάθειες για τα λογισμικά που χρησιμοποιεί. Σε οργανισμούς που αναπτύσσουν εφαρμογές είναι ακόμη σημαντικότερο να γίνεται σωστό version management διότι πρέπει ο κάθε μηχανισμός να δουλεύει όπως προβλέπεται ώστε να μην δημιουργηθεί πρόβλημα στις υπηρεσίες που προσφέρει ή και στην ομαλή συνύπαρξη του με τους υπόλοιπους μηχανισμούς που απαρτίζουν το σύστημα υπηρεσιών που προσφέρει ο οργανισμός. Επιπρόσθετα, στον τομέα της διαχείρισης των

οργανισμών από τους System administrators τους, γίνονται συχνά μαζικές αναβαθμίσεις λογισμικών και έπειτα ο έλεγχος για το αν έγιναν σωστά είναι πολλές φορές χρονοβόρος και δύσκολος.

- ❖ Χρησιμοποιώντας OVAL ένας οργανισμός θα μπορούσε ανά πάσα ώρα και στιγμή κάνοντας απλά μια μαζική σάρωση των συστημάτων του με βάση τις οδηγίες ενός OVAL σχήματος, να έχει την ακριβή εικόνα των εκδόσεων που έχει εγκατεστημένες σε όποια λογισμικά και υπηρεσίες τον ενδιαφέρει. Ο έλεγχος των διαχειριστών μετά από μαζικές αναβαθμίσεις θα ήταν πολύ εύκολος, και ο έλεγχος των μηχανικών που αναπτύσσουν εφαρμογές για τις εκδόσεις των μηχανισμών που χρησιμοποιούν θα ήταν ζήτημα λεπτών. Μάλιστα θα μπορούσε να αυτοματοποιηθεί κιόλας ώστε να γίνονται σταθεροί έλεγχοι προγραμματισμένοι σε ώρες με μικρό φόρτο για τον οργανισμό, ακόμη και καθημερινά. Έτσι θα υπάρχει πάντα πλήρης επίγνωση της κατάστασης των λογισμικών και μηχανισμών στον κάθε οργανισμό.

- Configuration Management
  - ❖ Οι ομάδες ανάπτυξης λογισμικού συχνά αντιμετωπίζουν προβλήματα με τις ρυθμίσεις (configuration) που πρέπει να έχουν εν ενεργεία ώστε να λειτουργούν σωστά όλα οι μηχανισμοί που αποτελούν το σύστημα τους και να συνεργάζονται εν αρμονία. Πολλές φορές αυτό αποτελεί μια δυσκολία και ένα πρόβλημα όταν τα έργα που αναλαμβάνουν οι ομάδες μηχανικών είναι πολύ μεγάλα. Με τον καιρό όπως αλλάζουν τα συστήματα κατά την ανάπτυξη τους, τα έως τότε σωστά ρυθμισμένα κομμάτια του συστήματος σταματούν να λειτουργούν όπως προβλεπόταν. Για αυτόν τον λόγο οι ομάδες ανάπτυξης λογισμικού πάντα κρατούν σημειώσεις και ιστορικά αρχεία που καταγράφουν τις αλλαγές που κάνουν στα κομμάτια που απαρτίζουν το σύστημα τους ώστε αν κάτι πάει στραβά να ξέρουν τι φταίει. Επίσης πολύ συνηθισμένη πλέον στρατηγική είναι να αποθηκεύουν την παρούσα κατάσταση ενός συστήματος όταν αυτό βρίσκεται σε σωστή λειτουργία. Έπειτα, αν κάτι πήγαινε στραβά, οι μηχανικοί θα μπορούσαν να επαναφέρουν το σύστημα στην παλιά του κατάσταση ή να έκαναν εκκίνηση την παλιά κατάσταση του συστήματος σε ένα ξεχωριστό περιβάλλον. Το πρόβλημα με την πρώτη λύση είναι ότι θα έχαναν πολλές ώρες δουλειάς και προόδου από τότε που το σύστημα δούλευε καλά έως και τη στιγμή που αντιλήφθηκαν το πρόβλημα. Το πρόβλημα με τη δεύτερη λύση είναι ότι η

διαφορά μεταξύ του σωστού και του προβληματικού συστήματος γίνεται χειρωνακτικά από τους μηχανικούς ή αλλιώς με το “μάτι”, αυτό παίρνει αρκετό χρόνο και μπορεί να μην επιφέρει πάντα αποτέλεσμα.

Όμως όσο μεγαλώνουν οι διαστάσεις των έργων που αναπτύσσονται, η διατήρηση ιστορικών αντιγράφων ασφαλείας γίνεται όλο και δυσκολότερη. Η χρήση της γλώσσας OVAL μπορεί να αποδειχθεί πολύ χρήσιμη και στο παραπάνω πρόβλημα για τις ομάδες ανάπτυξης λογισμικού. Με την OVAL μπορούν να αποθηκεύονται πολύ εύκολα και γρήγορα καταστάσεις του συστήματος συμπεριλαμβανομένων των ρυθμίσεων του (configurations). Το μεγάλο πλεονέκτημα όμως είναι ότι όταν το αναπτυσσόμενο σύστημα παρουσιάσει κάποιας μορφής βλάβη, θα πρόκειται για ένα ζήτημα μερικών λεπτών ώστε να γίνει η εκκίνηση της σάρωσης του συστήματος συγκρίνοντας το με το OVAL schema που είχε καταγεγραμμένη την σωστή κατάσταση του προηγουμένου. Έτσι θα παραχθεί το OVAL results schema που θα περιέχει ακριβώς τις διαφορές μεταξύ της εκδοχής του συστήματος που δούλευε σωστά και της εκδοχής που δεν δουλεύει. Τότε οι μηχανικοί θα μπορούν να διορθώσουν γρήγορα και στοχευμένα τη βλάβη, εφόσον έχουν εντοπίσει αποτελεσματικά την τοποθεσία της.

ΚΕΦΑΛΑΙΟ 4

Συγκριτική αξιολόγηση εργαλείων SIEM με βάση τις ανάγκες ενός οργανισμού

ΕΙΣΑΓΩΓΗ

Τα Security Information and Event Management (SIEM) είναι σύνθετα πολυμερή συστήματα που ποτέ δεν αναλαμβάνουν τις ίδιες ευθύνες για κάθε οργανισμό. Για ένα τόσο μεγάλο εργαλείο ασφάλειας, πρέπει να γίνεται προσχεδιασμός για τη διαδικασία της διεκπεραίωσης της εγκατάστασης του, αλλά ακόμη πιο σημαντικό είναι να γίνεται έρευνα και αξιολόγηση των διαθέσιμων προϊόντων και βάσει αυτών να επιλέγεται το σωστό για τον εκάστοτε οργανισμό. Μια καλή παρομοίωση είναι τα ψώνια ρούχων. Ο οργανισμός είναι το ανθρώπινο

σώμα, και τα διάφορα προϊόντα ρουχισμού είναι οι διαφορετικές λύσεις SIEM στην αγορά. Ο κάθε καταναλωτής που ψωνίζει έχει τις δικές του διαστάσεις, τις δικές του προτιμήσεις και τις δικές του ανάγκες. Το κάθε ρούχο έχει το δικό του μέγεθος, το δικό του χρώμα και την δική του περίσταση χρήσης. Ο καταναλωτής οφείλει να κάνει την έρευνα και τις δοκιμές των ρούχων που αγοράζει, γιατί σε διαφορετική περίπτωση, αγοράζοντας ένα ρούχο “στα τυφλά” κινδυνεύει να μην του ταιριάξει τελικά. Έτσι και ένας οργανισμός, οφείλει να κάνει την σωστή έρευνα, τη σωστή αξιολόγηση και τελικά την σωστή επιλογή για τον μηχανισμό ασφάλειας που θα αγοράσει. Η διαφορά είναι ότι εφόσον ο ένας οργανισμός είναι πιο περίπλοκος και έχει πιο πολλές ανάγκες και απαιτήσεις από ένα προϊόν που θα τον κρατάει ασφαλή, είναι και δυσκολότερο να γίνει η σωστή ανάλυση των αναγκών και των ίδιων των απαιτήσεων που τελικά έχει από αυτό το προϊόν.

4.1 Διαδικασία προσδιορισμού των αναγκών και απαιτήσεων του οργανισμού

Η καταγραφή των αναγκών και των απαιτήσεων ενός οργανισμού από ένα μεγάλο προϊόν που σκοπεύει να αγοράσει είναι μια μεγάλη διαδικασία που πρέπει να γίνεται σωστά και με προσοχή, γιατί κάθε αγορά ενός εργαλείου για έναν οργανισμό είναι μια επένδυση.

Κάθε οργανισμός έχει δικά του κριτήρια και στάνταρ, αλλά ένας γενικός αλγόριθμος για τον προσδιορισμό των αναγκών ενός οργανισμού είναι ο εξής:

- Δημιουργία ομάδας για τον προσδιορισμό των αναγκών.
  - ❖ Ένας μηχανισμός όπως ένα SIEM, φτάνει και αγγίζει όλα τα κομμάτια ενός οργανισμού. Κάθε τομέας και πυλώνας ενός οργανισμού σίγουρα έχει άλλες απαιτήσεις και προσδοκίες από ένα προϊόν ασφάλειας. Εφόσον ένα τέτοιο προϊόν επηρεάζει όλα τα τμήματα, θα πρέπει το κάθε ένα από αυτά να εκφέρει την άποψη του και να συμβάλει στην τελική απόφαση που θα παρθεί.

    Πολλές φορές οι αποφάσεις των μηχανισμών ασφαλείας βασίζεται πλήρως στην επιθυμία του τμήματος της ασφάλειας μιας και αυτοί είναι οι “ειδικοί” σε αυτό και έπειτα βάσει του χρηματικού ποσού (budget) που διαθέτει η εταιρεία, αγοράζεται

και το προϊόν [48]. Παρόλα αυτά, πρέπει να έχουν λόγο στο παραπάνω και άλλα τμήματα όπως αυτό της ανάλυσης κινδύνου (risk management sector), συμμόρφωσης σε κανονισμούς (audit and compliance sector) και συντήρησης / διαχείρισης συστημάτων (IT sector). Μια πολύ καλή τεχνική είναι να επιλεχθεί ένα άτομο από κάθε τομέα και εκπροσωπώντας το τμήμα του να συμμετέχει στη διαδικασία έρευνας αγοράς.

- Ορισμός των συστημάτων και των υπηρεσιών που χρειάζονται επίβλεψη
  - ❖ Σε επόμενη φάση, αφού έχει σχηματιστεί η σωστή ομάδα, πρέπει να επιλεχθούν τα συστήματα και οι υπηρεσίες που θα εποπτεύονται από τον SIEM μηχανισμό. Οι διάφορες επιλογές και αρχιτεκτονικές είναι πάρα πολλές καθώς ένας οργανισμός μπορεί να έχει διάφορες συσκευές ασφάλειας, διάφορες δικτυακές συσκευές γενικής χρήσης, βάσεις δεδομένων, εφαρμογές λογισμικού, εικονικά μηχανήματα και πολλά άλλα. Σε αυτό το στάδιο πρέπει να ερευνηθεί ο φόρτος εργασίας που έχει το κάθε ένα από τα παραπάνω στοιχεία του οργανισμού και το επίπεδο εποπτείας που χρειάζεται το κάθε ένα.

    Αυτή η έρευνα θα δώσει τεχνικές πληροφορίες που θα χρησιμοποιηθούν ώστε να αποφασιστούν οι απαιτήσεις του οργανισμού από το προϊόν για τις τεχνικές του δυνατότητες. Εδώ πρέπει να δοθεί ιδιαίτερη προσοχή και να ληφθεί υπόψη ότι οι ανάγκες του οργανισμού μελλοντικά μπορεί να αυξηθούν. Η κατά το καλύτερο δυνατό αξιοποίηση του budget της εταιρίας είναι σημαντική ώστε να μην υπάρχουν μελλοντικά προβλήματα σχετικά με το φόρτο που δέχεται ο μηχανισμός SIEM που θα αγοραστεί.

- Ορισμός των ειδικών αναγκών ασφάλειας και συμμόρφωσης του οργανισμού
  - ❖ Η ουσία της αγοράς του μηχανισμού SIEM. Η κατάλληλη ομάδα πρέπει να εξάγει τις ανάγκες ασφάλειας του οργανισμού σε πιο τεχνικό επίπεδο. Το SIEM θα πρέπει να δέσει με τον οργανισμό με τέτοιο τρόπο ώστε να καλύπτονται όλες του οι ανάγκες. Η

ομάδα που ορίζει αυτές τις ανάγκες, δεν είναι πάντα η ικανή ομάδα για να μιλήσει πιο τεχνικά σε επίπεδο μηχανής και συγκεκριμένων υπηρεσιών. Οπότε αφού έχει καλυφθεί το γενικότερο επίπεδο αναγκών, πρέπει να καλυφθεί και το ειδικό επίπεδο αναγκών από τους κατάλληλους ανθρώπους.

Επίσης οι ειδικές μετατροπές στον τρόπο λειτουργίας του οργανισμού ώστε να είναι συμμορφωμένος στο κάποιο ή κάποια νομικά πλαίσια που επιθυμεί, θα πρέπει να οριστούν σε τεχνικό επίπεδο ώστε να καλυφθούν έμπρακτα από τον μηχανισμό SIEM.

- Προσδιορισμός του τρόπου χρησιμοποίησης του προϊόντος
  - ❖ Ο τρόπος που θα αξιοποιήσει ο οργανισμός ένα Security Information and Event Management είναι πολύ σημαντικός στην προ αγοράς έρευνα που παίρνει μέρος. Ο στόχος που έχει θέσει η εταιρεία ως προς το κέρδος της από το SIEM μπορεί να είναι βασισμένος στην ασφάλεια ή στην παραγωγικότητα. Οι SIEM μηχανισμοί που είναι επικεντρωμένοι στην ασφάλεια είναι ρυθμισμένοι έτσι ώστε να παρέχουν τον καλύτερο δυνατό τρόπο να οδηγήσουν την ομάδα ασφάλειας την εταιρίας προς την ίαση του συμβάντος και τη μετρίαση σου. Οι μηχανισμοί SIEM που είναι επικεντρωμένοι στην παραγωγικότητα από την άλλη, είναι ρυθμισμένοι έτσι ώστε να παρέχουν όσο το δυνατό περισσότερα μηνύματα και ειδοποιήσεις ώστε να ενημερώνουν τους υπεύθυνους για την κατάσταση του οργανισμού, εφόσον δεν υπάρχει ομάδα ασφάλειας που κάνει αυτή τη δουλειά σε εικοσιτετράωρη βάση.

- Έλεγχος ετοιμότητας του οργανισμού
  - ❖ Το τελικό βήμα όλης της παραπάνω διαδικασίας είναι να ερευνήσει ο οργανισμός αν ο ίδιος είναι έτοιμος να δεχτεί στην καθημερινότητα του ένα τέτοιο εργαλείο. Αρχίζοντας από την ελλιπή παραγωγή εγγραφών log και άλλων στοιχείων από τα μηχανήματα και τις υπηρεσίες του οργανισμού, μέχρι και την έλλειψη τεχνολογικών εφοδίων και υποδομών για την στήριξη του SIEM. Ξέροντας πόση λεπτομέρεια μπορεί να διαθέσει σε ένα σύστημα ασφάλειας και πόσους φυσικούς πόρους μπορεί

να του δώσει, ένας οργανισμός μπορεί να οργανώσει καλύτερα την ερευνητική του καμπάνια για την αγορά.

4.2 Κριτήρια αξιολόγησης SIEM

Όπως σε κάθε άλλο προϊόν, η αξιολόγηση των Security Information and Event μηχανισμών ως προϊόντα πρέπει να γίνεται με συγκεκριμένα κριτήρια που στόχο έχουν να γίνεται η τελευταία όσο το καλύτερο δυνατό δίκαια και σωστά. Μόνο έτσι θα μπορεί να παραχθεί ένα αντικειμενικό αποτέλεσμα.

Έχουν τεθεί κάποια βασικά κριτήρια που είναι οι λειτουργίες που πρέπει απαραίτητα να καλύπτει ένα προϊόν ώστε να μπορεί να ονομαστεί ένα σύστημα SIEM [49][50][51]. Προϊόντα που δεν πληρούν αυτές τις προϋποθέσεις δε θα συμμετέχουν στην αξιολόγηση που θα πάρει μέρος στην ενότητα 4.3 .

Πίνακας 1 "Βασικές λειτουργίες που πρέπει να έχει κάθε SIEM"

Βασική λειτουργία	Επεξήγηση
Συλλογή δεδομένων	Η δυνατότητα συλλογής δεδομένων σε μορφή logs από πολλές διαφορετικές πηγές όπως firewall logs, network logs, IDS logs, web server logs, και άλλες πηγές αποστολής syslog formatted logs
Παραγωγή αναφορών	Δυνατότητα παραγωγής βασικών αναφορών που θα παράγονται σε προγραμματισμένο χρόνο.
Λειτουργία ειδοποιήσεων	Παραγωγή on-trigger ειδοποιήσεων που θα προωθούνται με διάφορους τρόπους μέσω email, SMS κ.α.
Λειτουργία συσχέτισης δεδομένων	Βασική συσχέτιση ομάδων κανόνων που κάνουν trigger. (π.χ. αν γίνει το Χ συμβάν, να γίνει έλεγχος και για το Ψ)

Λειτουργία βασικών forensics	Δυνατότητα εκτέλεσης προσαρμοσμένων ερωτημάτων στα ιστορικά δεδομένα οποιαδήποτε στιγμή
Ασφάλεια και δυνατότητες αποθήκευσης των δεδομένων	Παροχή ασφάλειας στα ιστορικά δεδομένα που αποθηκεύονται και δυνατότητα για την αποθήκευση τους για μεγάλα χρονικά διαστήματα

Έπειτα, εκτός από τις βασικές προϋποθέσεις του πίνακα 1, κάποια από τα κριτήρια που ορίστηκαν για την αξιολόγηση ενός μηχανισμού SIEM αποτελούνται από πιο προχωρημένες λειτουργίες που προσφέρουν τα προϊόντα στην αγορά. Αυτές οι λειτουργίες περιγράφονται στον πίνακα 2.

Πίνακας 2 "Προχωρημένες λειτουργίες μηχανισμών SIEM"

Προχωρημένη λειτουργία	Επεξήγηση (ζητούμενο)
Εμπλουτισμός πηγών	Δυνατότητα για είσοδο δεδομένων και από συσκευές / υπηρεσίες που παράγουν logs και από αυτές που δεν παράγουν, με άλλους τρόπους. Επίσης δυνατότητα ανάλυσης δικτυακών πακέτων.
Προχωρημένη συσχέτιση δεδομένων μεταξύ τους	Δυνατότητα δημιουργίας πολιτικών συσχέτισης δεδομένων βάσει των αναγκών του οργανισμού. Επίσης ύπαρξη τεχνικών τεχνητής νοημοσύνης για την ανάλυση των δεδομένων που λαμβάνονται.
Big Data Analytics	Δυνατότητα συνεργασίας (integration) με εργαλεία big data analytics ώστε να γίνεται περαιτέρω ανάλυση σε ιστορικά δεδομένα.

Προχωρημένες λειτουργίες παραγωγής αναφορών και ειδοποιήσεων	Ύπαρξη διεπαφών χρήστη που είναι δυνατό να προσαρμοστούν στις προτιμήσεις του. Ύπαρξη διαφόρων επιλογών για τις αναφορές και τις ειδοποιήσεις. Επίσης δυνατότητα συνεργασίας (integration) με εξωτερικά εργαλεία για αναφορές και παρουσίαση δεδομένων.
Λειτουργία προχωρημένων forensics	Προχωρημένες δυνατότητες αιτημάτων για ιστορικά δεδομένα, με λειτουργίες drill-down, pivoting και επανάληψης σεναρίου συμβάντος (session reconstructing).
Προχωρημένη ασφάλεια και δυνατότητες αποθήκευσης και διαχείρισης των δεδομένων	Προχωρημένη ασφάλεια συστήματος και κρυπτογράφηση των ιστορικών του δεδομένων. Επίσης δυνατότητα αποθήκευσης αυτών των δεδομένων σε εξωτερικές αποθηκευτικές υπηρεσίες για καλύτερη διασφάλιση των πληροφοριών σε σενάρια καταστροφής ή βλάβης.
Cyber Threat Intelligence	Ικανότητα του SIEM να αξιοποιήσει πηγές CTI και να κρατιέται ενήμερο βάσει αυτών.
Διαχείριση και ίαση των συμβάντων	Προχωρημένη αντιμετώπιση συμβάντων που χρειάζονται προσοχή. Αυτοματοποιημένες ενέργειες Self-healing για την αντιμετώπιση των τελευταίων, αν όχι του ίδιου του μηχανισμού, από κάποιο εξωτερικό εργαλείο που συνεργάζεται.
Πλήρης παρακολούθηση συμβάντων ασφάλειας	Δυνατότητα αλυσιδωτής παρακολούθησες ενός γεγονότος. Δημιουργία συνδέσμων μεταξύ συμβάντων για την καλύτερη παρακολούθηση τους από μηχανικούς.
Επεκτασιμότητα και διαχείριση δικτυακών πόρων	Ετοιμότητα του μηχανισμού SIEM να επεκταθεί και σε άλλα κομμάτια του δικτύου που βρίσκεται, εποπτεύοντας επιπλέον συσκευές. Επίσης να είναι ικανό να διαχειριστεί τον φόρτο που αυξάνεται στο δίκτυο λόγω των log μηνυμάτων που λαμβάνει από όλες τις συσκευές.

Επίσης πρέπει να σημειωθεί ότι η σύγκριση προϊόντων SIEM βάσει του EPS (Events Per Second) δεν είναι καλή τακτική καθώς αυτό βασίζεται πάρα πολύ στα χαρακτηριστικά του υλικού του φυσικού μηχανήματος που θα εκτελούνται και την κατάσταση τους.

Πολύ σημαντικά είναι και τα κριτήρια των εταιριών που κατασκευάζουν τους μηχανισμούς SIEM της αγοράς. Οι εταιρίες αυτές παρέχουν την υποστήριξη στους πελάτες τους μετά την αγορά και βοηθώντας τους σε διάφορα τεχνικά προβλήματα που μπορεί να προκύψουν αλλά και με τις μελλοντικές αναβαθμίσεις λογισμικού που θα τους προσφέρουν. Τα κριτήρια που τέθηκαν για τις εταιρείες κατασκευάστριες συστημάτων SIEM είναι τα εξής του πίνακα 3 [51]:

Πίνακας 3 "Κριτήρια αξιολόγησης της εταιρίας προέλευσης του προϊόντος"

Γενικά Χαρακτηριστικά πωλήτριας εταιρίας	Επεξήγηση
Βιωσιμότητα	Η βιωσιμότητα της εταιρείας είναι πολύ σημαντική. Αν τα πηγαίνει καλά, πολύ πιθανό να συνεχίσει την πορεία της πολλά χρόνια. Αυτό παρέχει ασφάλεια κάλυψης στους οργανισμούς πελάτες της.
Αφοσίωση	Η αφοσίωση του οργανισμού στο προϊόν αυτό. Αν είναι η μεγάλη του επιτυχία, και ασχολείται συνεχώς με αυτό, αυτόματα γίνεται μια πιο σίγουρη και καλή επένδυση για τον οργανισμό πελάτη.
Κάλυψη υποστήριξης	Η μετά αγοράς υποστήριξη που παρέχει η εταιρία. Το πόσο εύκολο είναι να επικοινωνήσουν οι οργανισμοί πελάτες και οι ώρες που υπάρχει διαθεσιμότητα είναι πολύ σημαντικά χαρακτηριστικά.

Ικανότητες πωλήσεων και marketing	Η δυνατότητα της πωλήτριας εταιρείας στις διαπραγματεύσεις παίζει επίσης μεγάλο ρόλο. Πολλές φορές λόγω της ομάδας πωλήσεων της εταιρίας γίνονται πακέτα τιμών και διαμορφώνονται συνεργασίες που αλλιώς δε θα επαίρναν μέρος. Οπότε μέσω αυτού, ένας πελάτης οργανισμός μπορεί να καταφέρει κάποια καλύτερη τιμή για το SIEM που ενδιαφέρεται.

Εκτός από τα τεχνικά χαρακτηριστικά του προϊόντος ως SIEM, οφείλουν να αξιολογηθούν και τα χαρακτηριστικά που έχει σαν ένα γενικό προϊόν τεχνολογίας, τα οποία βλέπουμε στον πίνακα 4.

Πίνακας 4 "Γενικά χαρακτηριστικά προϊόντος"

Γενικά χαρακτηριστικά προϊόντος	Επεξήγηση
Ευκολία χρήσης	Διεπαφές χρήστη που είναι εύκολο να κατανοηθούν και απλό να χρησιμοποιηθούν.
Κόστος	Η κοστολόγηση του προϊόντος σε σχέση τιμής ποιότητας.
Αρχιτεκτονική	Ποικιλία επιλογών τοποθέτησης και εγκατάστασης. Μη ύπαρξη διαταραχής του ήδη υπάρχοντος τεχνολογικού οικοσυστήματος του οργανισμού.

4.3 Συγκριτική αξιολόγηση SIEM

Τα προϊόντα SIEM που θα αξιολογηθούν είναι τα εξής:

- AlienVault – OSSIM
- HP – ArchSight

- LogRhythm – LogRhythm SIEM
- McAfee – Intel Security
- IBM – QRadar
- AAOS – EventTracker
- NetIQ – Sentinel
- Security Division of EMC – RSA
- Solarwinds – LEM
- Splunk – Splunk Enterprise

Η μεθοδολογία βαθμολόγησης που θα ακολουθήσει όπως και οι συντελεστές βαρύτητας που θα αναλυθούν είναι βασισμένα στο Info-Tech Research Group.

Οι βαθμολογίες για τις προχωρημένες λειτουργίες του πίνακα 2 κυμαίνονται μεταξύ 0 και 4, με το 0 να δηλώνει την μη ύπαρξη της λειτουργίας, το 1 να δηλώνει την μερική ύπαρξη, το 2 να δηλώνει την λειτουργία που απαιτεί επιπλέον πληρωμή, το 3 να δηλώνει την επαρκής ύπαρξη της λειτουργίας και το 4 την πιο καλή παροχή της λειτουργίας που υπάρχει αυτή τη στιγμή στην αγορά.

Οι βαθμολογίες για τα γενικά χαρακτηριστικά των προϊόντων και των εταιρειών πώλησης κυμαίνονται από το 0 έως το 5, με το 0 να είναι πολύ χαμηλό και σταδιακά το επίπεδο να αυξάνεται ως το 5 που είναι πάρα πολύ καλό. Αυτές οι τιμές έχουν οριστεί από την έρευνα αγοράς που έγινε από την Info-Tech Research Group το 2015 [51].

Η βαρύτητες για τις βαθμολογίες ποικίλουν και είναι πάντα ανάλογες με τις ανάγκες του ενδιαφερόμενου για αγορά οργανισμού. Για αυτή την αξιολόγηση θα δώσουμε τις εξής βαρύτητες:

- Αρχιτεκτονική προϊόντος – 25%
- Βιωσιμότητα εταιρείας – 25%
- Κάλυψη υποστήριξης – 17.5%
- Ευκολία χρήσης – 15%
- Κόστος – 10%
- Αφοσίωση εταιρίας – 5%
- Πωλήσεις εταιρείας – 2.5%

Οι βαρύτητα των προχωρημένων τεχνολογικών λειτουργιών είναι ισάξια για κάθε λειτουργία.

Πίνακας 5 "Συγκριτική αξιολόγηση προϊόντων SIEM"

Εταιρία	Βιωσιμότητα	Αφοσίωση	Υποστήριξη	Πωλήσεις	Τελική βαθμολογία
AlienVault	3	3	2	2	**3**
HP	4	2	3	3	**3**
LogRhythm	3	4	3	3	**3**
McAfee	4	3	4	3	**3**
IBM	4	3	4	4	**4**
AAOS	3	4	2	2	**3**
NetIQ	3	3	3	3	**3**
Security Division of EMC	3	3	4	3	**3**
Solarwinds	3	3	3	3	**3**
Splunk	3	3	3	3	**3**

Για λόγους εξοικονόμησης χώρου στον πίνακα 6, οι αντιστοιχίες των τίτλων είναι οι εξής:

1. Εμπλουτισμός πηγών
2. Προχωρημένη συσχέτιση δεδομένων
3. Big Data Analytics
4. Προχωρημένη παραγωγή αναφορών / ειδοποιήσεων
5. Advanced Forensics
6. Ασφάλεια και αποθήκευση δεδομένων
7. Cyber Threat Intelligence
8. Διαχείριση και ίαση συμβάντων

9. Παρακολούθηση συμβάντων ασφάλειας
10. Επεκτασιμότητα και διαχείριση δικτυακών πόρων

Πίνακας 6 "Συγκριτική βαθμολογία των προχωρημένων λειτουργιών προϊόντων SIEM"

Προϊόν	***1.***	***2.***	***3.***	***4.***	***5.***	***6.***	***7.***	***8.***	***9.***	***10.***	***Τελική βαθμολογία***
AlienVault OSSIM	4	2	3	4	4	2	4	2	2	4	***3.1***
HP ArcSight	4	2	0	4	4	4	4	2	2	4	***3.0***
LogRhythm Enterprise	4	4	4	4	4	4	4	4	4	4	***4.0***
McAfee IntelSecurity	4	4	4	4	2	4	4	4	4	4	***3.8***
IBM QRadar	4	2	4	4	4	4	4	2	4	4	***3.6***
AAOS EventTracker	4	2	0	4	4	4	4	2	2	4	***3.0***
NetIQ Sentinel	4	2	2	4	4	4	4	2	4	4	***3.4***
Security Division of EMC RSA	4	2	4	4	2	4	4	0	2	4	***3.0***
Solarwinds LEM	2	2	0	4	2	4	2	4	2	4	***2.6***

Splunk Enterprise	4	4	4	4	3	4	4	4	4	5	***4.0***

ΚΕΦΑΛΑΙΟ 5

Μελέτη περίπτωσης χρήσης (Case Study)

ΕΙΣΑΓΩΓΗ

Για την περίπτωση χρήσης και την παρουσίαση της λειτουργικότητας ενός συστήματος SIEM επέλεξα να μην χρησιμοποιήσω ένα έτοιμο ολοκληρωμένο προϊόν (OSSIM, ArcSight, QRadar κ.α.), αλλά να συνθέσω ένα SIEM με δύο ανεξάρτητα κομμάτια που θα συνεργαστούν μεταξύ τους. Τα συστήματα που επέλεξα είναι το OSSEC και το ELK stack. Και τα δύο είναι λογισμικά ανοιχτού κώδικα. Το OSSEC είναι ένα Host-based Intrusion Detection System (HIDS) και είναι υπεύθυνο για την εποπτεία των agents και την ανάλυση των logs. Το ELK stack αναλαμβάνει να δεχτεί δεδομένα από τον OSSEC σχετικά με τους agents και να τα οπτικοποιήσει. Το ELK stack απαρτίζεται από τρία κομμάτια, το Elastic Search, το Logstash και το Kibana, που θα αναλυθούν περαιτέρω στο 5.2.

Ο SIEM manager στήθηκε σε λειτουργικό σύστημα Linux, Ubuntu 20.04 LTS, είχε την ισχύ ενός τετραπύρηνου επεξεργαστή Intel i7 και είχε μνήμη RAM 16 GB.

Ο SIEM agent στήθηκε σε λειτουργικό σύστημα Linux Ubuntu 20.04 LTS, είχε την ισχύ ενός διπύρηνου επεξεργαστή Intel i3 και μνήμη RAM 4 GB.

Το μηχάνημα που χρησιμοποιήθηκε για να επιτεθεί στο μηχάνημα του SIEM agent επίσης στήθηκε σε λειτουργικό σύστημα Linux Ubuntu 20.04 LTS, είχε την ισχύ ενός τετραπύρηνου επεξεργαστή Intel i7 και μνήμη RAM 8 GB.

5.1 Εγκατάσταση των υποσυστημάτων

5.1.1 OSSEC

Ο OSSEC HIDS είναι από τα πιο γνωστά Host-based Intrusion Detection Systems και είναι η βάση πολλών γνωστών προϊόντων SIEM όπως το OSSIM της AlienVault.

Η εγκατάσταση του είναι πολύ απλή, τα βήματα είναι τα εξής:

- Κατεβάζουμε και εγκαθιστούμε τα απαιτούμενα πακέτα που έχει ορίσει η Atomicorp.
  - libz-dev
  - libssl-dev
  - libpcre2-dev
  - libevent-dev
  - build-essential

- Κατεβάζουμε τη νεότερη έκδοση του OSSEC και αποσυμπιέζουμε το πακέτο.

- Κάνουμε την εγκατάσταση του OSSEC σύμφωνα με τις οδηγίες της Atomicorp [53].

5.1.1.1 Σύνδεση OSSEC agent

Η διαδικασία για τη σύνδεση ενός agent ώστε να εποπτεύεται από το μηχανισμό SIEM είναι η εξής:

Στον OSSEC Server:

- Με τη λειτουργία “manage_agents” του OSSEC, δημιουργούμε έναν νέο agent. Τα στοιχεία που χρειαζόμαστε είναι τα εξής: [Εικόνα 1]
  - Agent name
  - Agent IP address

- Agent ID

- Εξάγουμε το μοναδικό κλειδί για τον agent που δημιουργήσαμε. [Εικόνα 2]

- Κάνουμε επανεκκίνηση τις λειτουργίες του OSSEC server. [Εικόνα 3]

```
****************************************
* OSSEC HIDS v3.6.0 Agent manager.     *
* The following options are available: *
****************************************
   (A)dd an agent (A).
   (E)xtract key for an agent (E).
   (L)ist already added agents (L).
   (R)emove an agent (R).
   (Q)uit.
Choose your action: A,E,L,R or Q: a

- Adding a new agent (use '\q' to return to the main menu).
  Please provide the following:
   * A name for the new agent: my_company_agent1
   * The IP Address of the new agent: 192.168.1.10
   * An ID for the new agent[003]:
Agent information:
   ID:003
   Name:my_company_agent1
   IP Address:192.168.1.10

Confirm adding it?(y/n): y
Agent added with ID 003.


****************************************
```

Εικόνα 1 "Δημιουργία ενός νέου OSSEC Agent

```
Choose your action: A,E,L,R or Q: E

Available agents:
   ID: 001, Name: mycompany_LinuxAgent_1, IP: 192.168.1.10
Provide the ID of the agent to extract the key (or '\q' to quit): 001

Agent key information for '001' is:
MDAxIG15Y29tcGFueV9MaW51eEFnZW50XzEgMTkyLjE2OC4xLjEwIDRhNmI0ZDE5M2QzNTI
1YmYzODU5ZmYwYjk0NTBmYmNiOTljMTdlOTg1MWI3YmE3NDU1Y2M3ZGI2YWYzMmM4YWI=

** Press ENTER to return to the main menu.
```

Εικόνα 2 "Εξαγωγή μοναδικού κλειδιού agent"

```
root@thesis-VM:/var/ossec/bin# /var/ossec/bin/ossec-control restart
Deleting PID file '/var/ossec/var/run/ossec-remoted-8920.pid' not used.
..
Killing ossec-monitord ..
Killing ossec-logcollector ..
ossec-remoted not running ..
Killing ossec-syscheckd ..
Killing ossec-analysisd ..
Killing ossec-maild ..
Killing ossec-execd ..
OSSEC HIDS v3.6.0 Stopped
Starting OSSEC HIDS v3.6.0...
Started ossec-maild...
Started ossec-execd...
Started ossec-analysisd...
Started ossec-logcollector...
Started ossec-remoted...
Started ossec-syscheckd...
Started ossec-monitord...
Completed.
root@thesis-VM:/var/ossec/bin#
```

Εικόνα 3 "Επανεκκίνηση λειτουργιών server"

Στον OSSEC Agent:

- Κατεβάζουμε και εγκαθιστούμε τον OSSEC με τον ίδιο τρόπο που κάναμε και για τον server. Απλά στην επιλογή του τύπου εγκατάστασης επιλέγουμε agent.

- Όταν ζητηθεί η IP address του OSSEC server την τυπώνουμε. [Εικόνα 4]

- Με τη λειτουργιά “manage_agents” προσθέτουμε το μοναδικό κλειδί του agent που πήραμε από τον server. [Εικόνα 5]

- Με τη λειτουργία “ossec-control” κάνουμε εκκίνηση του OSSEC agent. (“`/var/ossec/bin/ossec-control start`”)

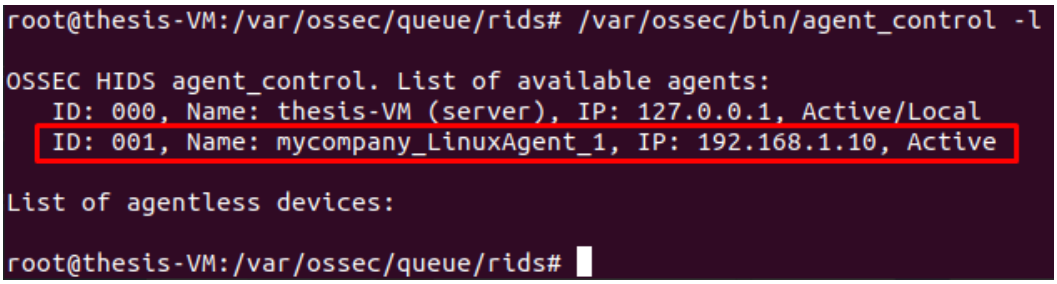
```
3- Configuring the OSSEC HIDS.

  3.1- What's the IP Address or hostname of the OSSEC HIDS server?: 192.168.1.2

   - Adding Server IP 192.168.1.2
```

Εικόνα 4 "Εισαγωγή OSSEC server IP κατά την εγκατάσταση του agent"

```
****************************************
* OSSEC HIDS v3.6.0 Agent manager.     *
* The following options are available: *
****************************************
   (I)mport key from the server (I).
   (Q)uit.
Choose your action: I or Q: i

* Provide the Key generated by the server.
* The best approach is to cut and paste it.
*** OBS: Do not include spaces or new lines.

Paste it here (or '\q' to quit): MDAzIG15X2NvbXBhbnlfYWdlbnQxIDE5Mi4x
NjguMS4xMCA2ZTUzMjdkYzgxMDZkNzNmNzVhM2ViYmViODExN2U3ODM2YjMxYmFjZjkyO
TQyNzVlZDAzOGZiZTA4ODA0NjY4

Agent information:
   ID:003
   Name:my_company_agent1
   IP Address:192.168.1.10

Confirm adding it?(y/n): y
2020/05/29 21:47:29 manage_agents: ERROR: Cannot unlink /queue/rids/s
ender: No such file or directory
Added.
** Press ENTER to return to the main menu.
```

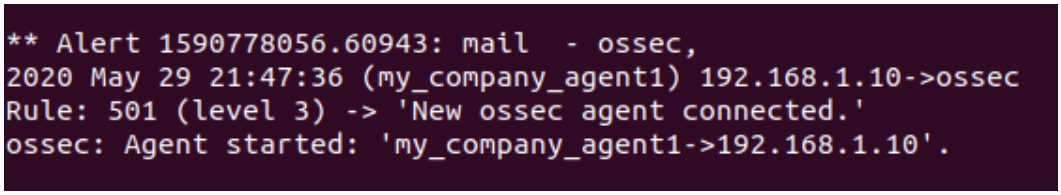
Εικόνα 5 "Προσθήκη του μοναδικού κλειδιού στον agent"

Τέλος, μπορούμε να ελέγξουμε αν η σύνδεση ήταν επιτυχής χρησιμοποιώντας τη λειτουργία “agent-control” στον OSSEC server. [Εικόνα 6]



Εικόνα 6 "Έλεγχος της σύνδεσης OSSEC server - agent"

Στα logs του OSSEC θα δούμε ότι συνδέθηκε ένας agent για πρώτη φορά. [Εικόνα 7] (“`cat /var/ossec/logs/alerts/alerts.log`”)



Εικόνα 7 "New ossec agent connected. log"

5.1.2 ELK

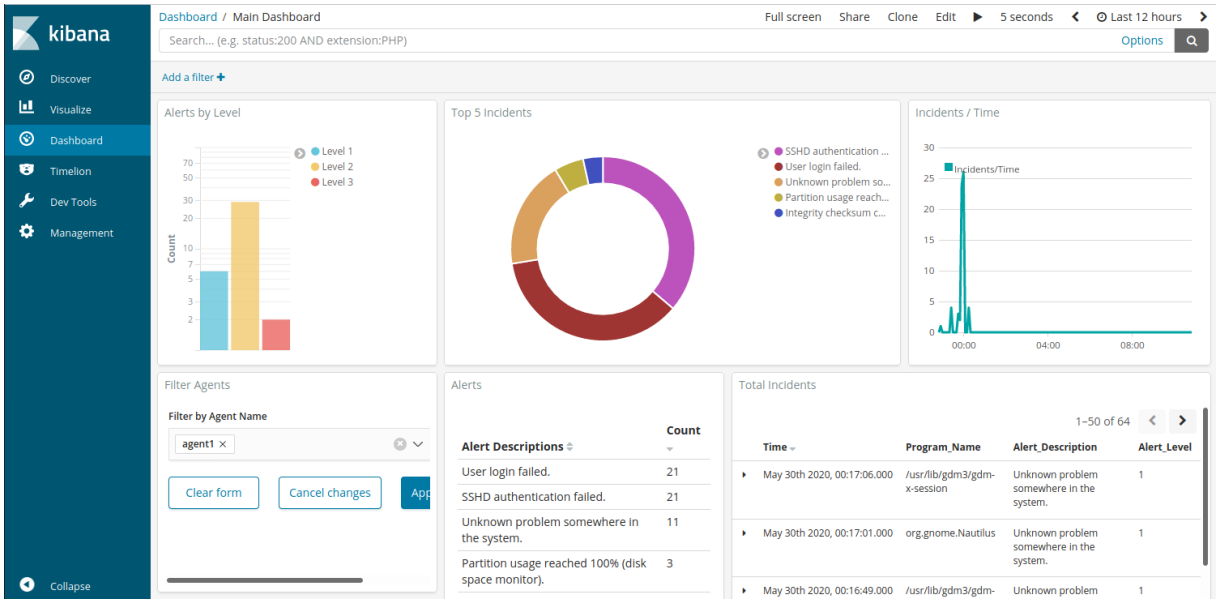
Το ELK stack είναι ένα λογισμικό ανοιχτού κώδικα που είναι υπεύθυνο και αναλαμβάνει να κάνει τρεις λειτουργίες για τον μηχανισμό SIEM που φτιάχνουμε. Την πρώτη την εκτελεί το Logstash το οποίο παίρνει τα logs που παράγει ο OSSEC server και αφού πάρει όσες πληροφορίες χρειάζεται, τα δίνει στο Elastic Search. Αυτό με τη σειρά του χωρίζει τα logs με συγκεκριμένα κριτήρια που ποικίλουν ανά οργανισμό και τοποθέτηση SIEM και τα ταξινομεί σε indexes. Το Kibana παίρνει την πληροφορία από τους indexes που δημιούργησε το Elastic Search κάνοντας του queries και την οπτικοποιεί για τους τελικούς χρήστες.

Την εγκατάσταση του αποτελούν τα εξής βήματα (συνοπτικά):

- Εγκατάσταση Java 8 (openjdk-8-jdk).
- Εγκατάσταση του Elastic search σύμφωνα με τις οδηγίες της Elastic [54].

- Εγκατάσταση του Kibana σύμφωνα με τις οδηγίες της Elastic [54].
- Εγκατάσταση του Logstash σύμφωνα με τις οδηγίες της Elastic [54].
- Δημιουργία του Logstash pipeline το οποίο δίνει στο Logstash την τοποθεσία των logs του OSSEC server.
- Πρόσθεση του χρήστη συστήματος του Logstash στις ομάδες δικαιωμάτων του “root” και “ossec” ώστε να έχει τα κατάλληλα δικαιώματα για να μπορεί να διαβάσει το αρχείο log του OSSEC server.
- Εκκίνηση των υπηρεσιών Logstash, Elastic Search και Kibana.
- Είσοδος στο GUI (graphical user interface) του Kibana και δημιουργία νέου index pattern για να οπτικοποιήσουμε τα δεδομένα από το Elastic Search.
- Τέλος, δημιουργούμε ένα dashboard για τον χρήστη, προσθέτοντας τα επιθυμητά γραφικά στοιχεία (widgets).

Το dashboard που δημιουργήσαμε για τον χρήστη μας στο Kibana μοιάζει κάπως έτσι: [Εικόνα 8]



Εικόνα 8 "Kibana Dashboard"

5.2 Αλληλουχία ενεργειών για τα σενάρια χρήσης

Τα σενάρια χρήσης που θα εξεταστούν είναι ο εντοπισμός επίθεσης DDOS (βλ. 5.2.1) και ο εντοπισμός ενός χρήστη που προσπάθησε να εκτελέσει εντολές με δικαιώματα διαχειριστή (βλ. 5.2.2). Ο ροή των γεγονότων που θα ακολουθήσει από την οπτική γωνία του SIEM manager είναι η εξής:

- Logs που επιδεικνύουν την ύπαρξη αυτών των συμβάντων θα φτάσουν στον SIEM manager.
- Ο κατάλληλος OSSEC αποκωδικοποιητής για το κάθε ένα θα εξάγει τις χρήσιμες πληροφορίες του log βάσει της υπηρεσίας που το παρήγαγε.
- Όλοι οι OSSEC κανόνες που συνδέονται με αυτόν τον αποκωδικοποιητή θα σαρωθούν για να βρεθεί η πιθανή ύπαρξη κάποιας ανωμαλίας.
- Ο κατάλληλος κανόνας που περιγράφει με ακρίβεια το συμβάν θα “χτυπήσει” και ο OSSEC server θα γράψει στο αρχείο log του στοιχεία σχετικά με αυτήν την ανωμαλία.
- Το Logstash θα διαβάσει την εγγραφή log από το αρχείο log του OSSEC και θα το περάσει στο Elastic Search

- Το Elastic Search θα κατατάξει το συμβάν στον σωστό index βάσει των κριτηρίων διαχωρισμού των indexes.
- Το Kibana κάνοντας ερωτήματα στο Elastic Search θα πάρει την πληροφορία σχετικά με το συμβάν και θα την εμφανίσει στον τελικό χρήστη.

Επιπλέον λειτουργίες όπως αποστολή email/SMS και παραγωγή αναφορών δεν θα αναλυθούν.

5.3 Σενάριο εντοπισμού SSH Bruteforce επίθεσης

Για να δημιουργηθούν οι αποκωδικοποιητές και οι κανόνες που θα περιγράφουν ένα συμβάν θα πρέπει πρώτα να γνωρίζουμε το πρότυπο που ακολουθεί η υπηρεσία που τα παράγει στο αρχείο καταγραφής της. Η υπηρεσία για την υπηρεσία SSH είναι η “sshd” και τα log της μοιάζουν κάπως έτσι:

May 30 11:27:04 agent sshd[12761]: Failed password for agent from 192.168.1.11 port 53526 ssh2

Σε αυτή την εγγραφή log η μηχανή ανάλυσης του OSSEC θα καταλάβει ότι πρόκειται για την υπηρεσία SSH από το “*sshd[12761]*“ και θα χρησιμοποιήσει τους αποκωδικοποιητές του SSH για να το αποσαφηνίσει. Ο αποκωδικοποιητής που ταιριάζει με τη δομή του παραπάνω log είναι ο εξής: [Εικόνα 9]

```
<decoder name="ssh-failed">
  <parent>sshd</parent>
  <prematch_pcre2>^Failed \S+ </prematch_pcre2>
  <pcre2 offset="after_prematch">^for (\S+) from (\S+) port \d+</pcre2>
  <order>user, srcip</order>
</decoder>
```

Εικόνα 9 "SSH: Failed Decoder"

- Με το πεδίο “*parent*” δηλώνεται από ποιόν κύριο αποκωδικοποιητή κλήθηκε αυτός.
- Με το πεδίο “*prematch_pcre2*” δηλώνεται το κομμάτι του log που θα πρέπει να υπάρχει ώστε να είναι κατάλληλος αυτός ο αποκωδικοποιητής να αναλύσει αυτό το συμβάν.

- Με το πεδίο “*pcre2 offset=”after_prematch*” δηλώνεται πως θα είναι η συνέχεια του log ακριβώς μετά την κάλυψη που έκανε το πεδίο “*prematch_pcre2*”.

Με τη χρήση του πεδίου “*pcre2*” οι αποκωδικοποιητές κάνουν την κύρια δουλειά τους. Εξάγουν πληροφορίες από το log και τις αποθηκεύουν σε μεταβλητές που μπορούν να χρησιμοποιηθούν από κανόνες αργότερα. Οι μεταβλητές και η αντιστοιχία τους δηλώνεται από το πεδίο “*order*”. Για να επιδείξουμε τα κομμάτια του log που θέλουμε να εξάγουμε σαν πληροφορίες χρησιμοποιούμε pcre2 εκφράσεις καλύπτοντας τα σημεία που θέλουμε (π.χ. “`(\S+)\ `“ είναι ένας οποιοσδήποτε χαρακτήρας εκτός από κενό, όσες φορές χρειαστεί μέχρι να βρεθεί κενό).

Αφού τελειώσει η διαδικασία της αποκωδικοποίησης, σαρώνονται όλοι οι κανόνες που αφορούν την υπηρεσία SSH. Το χαρακτηριστικό πεδίο του κανόνα που θα “χτυπήσει” είναι το “*decoded_as*” που συνδέει και έναν OSSEC κανόνα με έναν OSSEC αποκωδικοποιητή. Ο κανόνας αυτός είναι ο εξής: [Εικόνα 10]

```
<!-- SSHD messages -->
<group name="syslog,sshd,">
  <rule id="5700" level="0" noalert="1">
    <decoded_as>sshd</decoded_as>
    <description>SSHD messages grouped.</description>
  </rule>
```

Εικόνα 10 "SSHD messages grouping rule"

Αμέσως μετά θα σαρωθούν τα “παιδιά” του κανόνα που μαζεύει όλα τα συμβάντα που είναι σχετικά με την υπηρεσία SSH και από τους κανόνες που τον κληρονομούν θα “χτυπήσει” ο εξής: [Εικόνα 11]

```
<rule id="5716" level="2">
  <if_sid>5700</if_sid>
  <pcre2>^Failed|^error: PAM: Authentication</pcre2>
  <description>SSHD authentication failed.</description>
  <group>authentication_failed,</group>
</rule>
```

Εικόνα 11 "SSHD authentication failed rule"

Βάσει των μέχρι τώρα αναφερόμενων κομματιών της αλληλουχίας ενεργειών που εκτελούνται από τον OSSEC manager, θα παραχθεί η εξής εγγραφή log: [Εικόνα 12]

```
** Alert 1590827228.35260: - syslog,sshd,authentication_failed,
2020 May 30 11:27:08 (my_company_agent1) 192.168.1.10->/var/log/auth.log
Rule: 5716 (level 2) -> 'SSHD authentication failed.'
Src IP: 192.168.1.11
User: agent
May 30 11:27:04 agent sshd[12761]: Failed password for agent from 192.168.1.11 port 53526 ssh2
```

Εικόνα 12 "OSSEC log για την αποτυχημένη προσπάθεια SSH login"

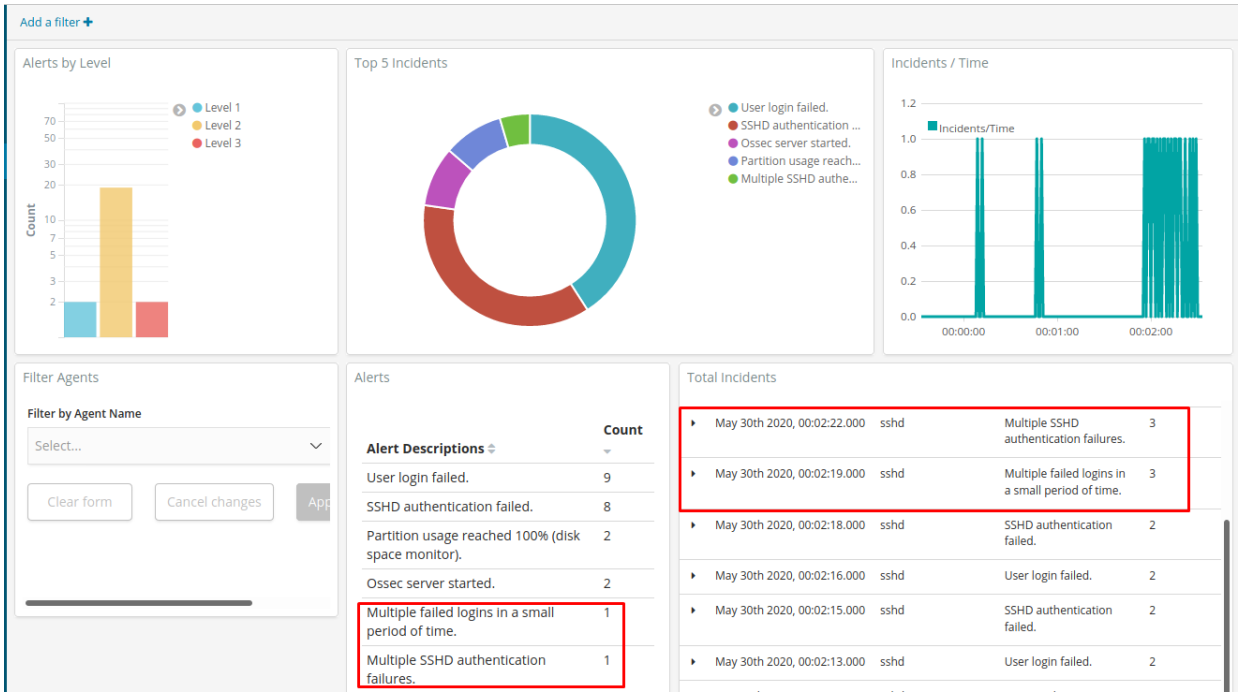
Το συμβάν ασφάλειας μιας DDOS επιθέσεως το χαρακτηρίζει ο κανόνας στην Εικόνα 13.

```
<rule id="5720" level="3" frequency="6" timeframe="30">
  <if_matched_sid>5716</if_matched_sid>
  <same_source_ip />
  <description>Multiple SSHD authentication failures.</description>
  <group>authentication_failures,</group>
</rule>
```

Εικόνα 13 "Multiple SSHD authentication failures rule"

Ο παραπάνω κανόνας δηλώνει ότι πρόκειται για ένα συμβάν με επίπεδο επικινδυνότητας 3 (“*level*”) και ότι αν “χτυπήσει” ο κανόνας με ID 5716 (“*if_matched_sid*”) έξι ή περισσότερες φορές (“*frequency*”) μέσα στο χρονικό παράθυρο των 30 δευτερολέπτων (“*timeframe*”) και μάλιστα όλα τα συμβάντα έχουν την ίδια πηγαία διεύθυνση IP (“*same_source_ip*”), συνέβη ένα συμβάν SSHD Bruteforce.

Η οπτικοποίηση του SSH Bruteforce event στο Kibana dashboard μας είναι η εξής: [Εικόνα 14]



Εικόνα 14 "Kibana - SSH Bruteforce"

Αν επιλέξουμε να δούμε παραπάνω πληροφορίες για το συμβάν, αυτές που έχουμε διαθέσιμες είναι οι εξής: [Εικόνα 15]

May 30th 2020, 00:02:22.000 sshd Multiple SSHD authentication failures. 3

Table JSON View surrounding documents View single document

t	@timestamp	May 30th 2020, 00:02:22.000
t	@version	1
t	Agent_Name	agent1
t	Alert_Description	Multiple SSHD authentication failures.
t	Alert_Level	3
t	Company_Name	my_company
t	Full_Log	May 30 00:02:22 agent sshd[11611]: Failed password for agent from 192.168.1.11 port 53474 ssh2
t	ID	1590786142.7039
t	Program_Name	sshd
#	TimeStamp	1,590,786,142,000
t	_id	tco-YnIBvdYSsxuWNO68
t	_index	my_company-2020.05
#	_score	-
t	_type	doc
t	agent_name	my_company_agent1
?	agentip	⚠ 192.168.1.10
t	decoder	sshd
t	decoder_desc.name	sshd
?	decoder_desc.parent	⚠ sshd
?	decoder_parent	⚠ sshd
t	dstuser	agent
t	host	thesis-VM
t	location	(my_company_agent1) 192.168.1.10->/var/log/auth.log
t	logfile	/var/log/auth.log
t	path	/var/ossec/logs/alerts/alerts.json
?	previous_output	⚠ May 30 00:02:15 agent sshd[11607]: Failed password for agent from 192.168.1.11 port 53470 ssh2
t	rule.comment	Multiple SSHD authentication failures.
#	rule.firedtimes	1
?	rule.frequency	⚠ 6
t	rule.groups	syslog, sshd, authentication_failures
#	rule.level	3
#	rule.sidid	5,720
?	srcip	⚠ 192.168.1.11
t	timestamp	2020 May 30 00:02:22
t	type	json

Εικόνα 15 "Όλες οι πληροφορίες σχετικά με την επίθεση SSH Bruteforce"

5.4 Σενάριο εντοπισμού προσπάθειας χρήστη για εκτέλεση εντολών διαχειριστή

Ένας χρήστης ενός συστήματος μπορεί να προσπαθήσει να εκτελέσει εντολές με δικαιώματα διαχειριστή χρησιμοποιώντας την sudo εντολή. Αν το κάνει, γίνεται έλεγχος από το λειτουργικό σύστημα Linux για το αν βρίσκεται μέσα στο αρχείο που περιέχει τη λίστα με όλους τους χρήστες που έχουν δικαιώματα διαχειριστή, ή αλλιώς τους “sudoers”. Αυτό το σενάριο χρήσης ανιχνεύει τα συμβάντα που κάποιος χρήστης που δεν ανήκει στους “sudoers” προσπαθεί να χρησιμοποιήσει την sudo εντολή.

Ο OSSEC κανόνας που θα μαζέψει όλα τα συμβάντα από τον αποκωδικοποιητή που σαρώνει τα logs σχετικά με το sudo είναι ο εξής: [Εικόνα 16]

```
<!-- Sudo messages -->
<group name="syslog,sudo">
  <rule id="5400" level="0" noalert="1">
    <decoded_as>sudo</decoded_as>
    <description>Initial group for sudo messages</description>
  </rule>
```

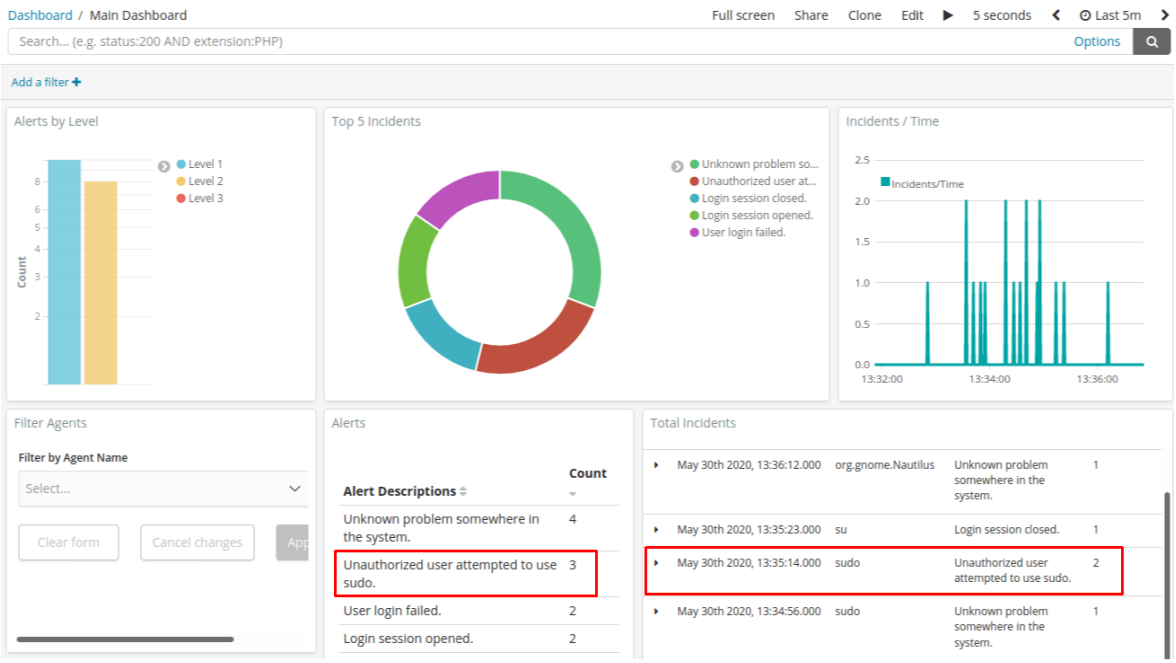
Εικόνα 16 "Sudo messages grouping rule"

Αφού γίνει το trigger του κανόνα με ID 5400 θα σαρωθούν όλοι οι κανόνες που τον κληρονομούν, και αυτός με το ID 5405 που αναφέρεται στο συμβάν του σεναρίου θα “χτυπήσει” επειδή ελέγχει το log για το αλφαριθμητικό “user NOT in sudoers” με τη χρήση του πεδίου “*pcre2*”. [Εικόνα 17]

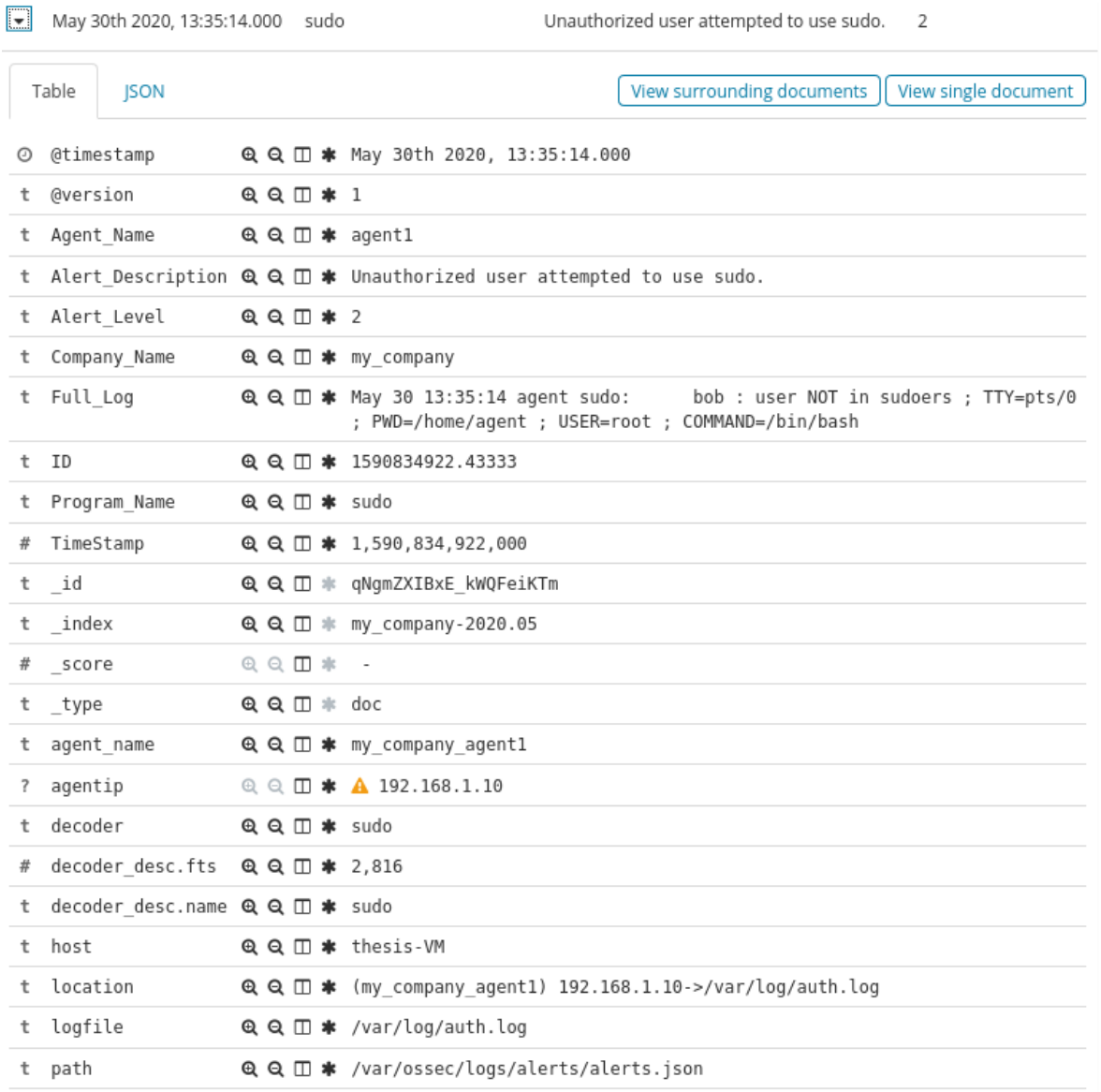
```
<rule id="5405" level="2">
  <if_sid>5400</if_sid>
  <pcre2>user NOT in sudoers</pcre2>
  <description>Unauthorized user attempted to use sudo.</description>
</rule>
```

Εικόνα 17 "Unauthorized user attempted to use sudo rule"

Το Kibana τότε θα οπτικοποιήσει το log που παρήγαγε ο OSSEC server. [Εικόνα 18] και μέσω αυτού μπορούμε να δούμε όλες τις πληροφορίες σχετικά με το συμβάν. [Εικόνα 19]



Εικόνα 18 "Kibana - Unauthorized user sudo attempt"



Εικόνα 19 "Όλες οι πληροφορίες σχετικά με το unauthorized sudo συμβάν"

ΚΕΦΑΛΑΙΟ 6

Συμπεράσματα και μελλοντικές επεκτάσεις

ΕΙΣΑΓΩΓΗ

Σε αυτό το κεφάλαιο αναφέρονται τα συμπεράσματα που προέκυψαν από τη διάρθρωση της παρούσας διπλωματικής και τη σε βάθος μελέτη των μηχανισμών Security Information and Event Management και των προτύπων και τεχνολογιών που υπάρχουν στο ίδιο οικοσύστημα με αυτά, όπως επίσης και από τη συγκριτική αξιολόγηση των προϊόντων SIEM της αγοράς αλλά και από τη διαδικασία αυτής. Επίσης σε αυτό το κεφάλαιο θα γίνει μια σύντομη αναφορά των μελλοντικών επεκτάσεων που μπορούν να υλοποιηθούν γύρω από τα συστήματα SIEM.

Συμπεράσματα

Με την όλο και αυξανόμενη ανάπτυξη του τομέα της ασφάλειας στην επιστήμης της πληροφορίας, οι μηχανισμοί που εκτελούν τα καθήκοντά της αναπτύσσονται εξίσου γρήγορα και όλο ένα και περισσότερο η αγορά και το εμπόριο της τεχνολογίας τείνει προς τις λύσεις που παρέχουν την καλύτερη δυνατή ασφάλεια εκ των ανταγωνιστών τους.

Τα συστήματα SIEM βρίσκονται ακόμα στα αρχικά στάδια εξέλιξης τους και όσο προχωρούν τα χρόνια θα αποκτούν ακόμα σημαντικότερο ρόλο στην αρχιτεκτονική των οργανισμών που οι ρίζες τους απλώνονται και στον κυβερνοχώρο. Τα συστήματα SIEM δε σταματούν μόνο στην κάλυψη των συμβάντων που αποτελούν εγκληματικές πράξεις, η παραγωγικότητα τους φτάνει σε νέα ύψη όταν προσαρμόζονται ώστε να ανταποκρίνονται και σε ζητήματα ενός οργανισμού που δεν είναι απαραίτητα ανωμαλίες ή συμβάντα που χρήζουν προσοχής. Μια ομάδα που διαχειρίζεται τους τεχνολογικούς πόρους μιας εταιρίας, μπορεί να επωφεληθεί πολύ από την κεντρική επίβλεψη και αναφορά που παρέχουν τα συστήματα SIEM ανά πάσα ώρα και στιγμή μέσα στη μέρα.

Τα πρότυπα και οι τεχνολογίες που έχουν αναπτυχθεί ως σήμερα είναι πάρα πολύ αποδοτικές στον σκοπό που εκφέρουν αλλά παρόλα αυτά υπάρχουν μεγάλα

περιθώρια βελτίωσης και ανάπτυξης νέων προτύπων και πρωτοκόλλων τα οποία θα οδηγήσουν τους μηχανισμούς SIEM ένα βήμα παραπέρα κάνοντάς τους το κύριο εργαλείο ασφάλειας του κάθε οργανισμού ανεξάρτητα από το μέγεθος του και τις οικονομικές τους δυνατότητες.

Όσο τα SIEM βρίσκονται στα πρώτα κύρια χρόνια της τεχνολογικής τους ανάπτυξης ο διαμοιρασμός γνώσης μεταξύ των μεγάλων οργανισμών που τα κατασκευάζουν θα είναι περιορισμένος και αυτό είναι ένα εμπόδιο που ακόμα δεν έχει ξεπεραστεί. Στα επόμενα στάδια που οι τεχνικές ανάπτυξης των μηχανισμών SIEM θα είναι πιο εμπεριστατωμένες και ευρέως γνωστές στην κοινότητα της ανάπτυξης λογισμικού, θα υπάρξει μεγάλη πρόοδος στο επίπεδο και στην ποιότητα αυτών των μηχανισμών.

Τελειώνοντας, πρέπει να σημειωθεί, ότι τα συστήματα SIEM δεν πρέπει να χαρακτηρίζονται ως ένα απλό εργαλείο ασφάλειας, αλλά ως ένα εργαλείο που εποπτεύει όλα τα εργαλεία ασφάλειας και μη ενός οργανισμού με αποτέλεσμα να ανυψώνει τα επίπεδα αντίληψης του τελευταίου για τα συμβάντα και τον τρόπο λειτουργίας του. Έτσι εκτός από το επίπεδο της ασφάλειας ενός οργανισμού, αυξάνονται ραγδαία και τα επίπεδα παραγωγικότητας του και οργάνωσης του.

6.1 Μελλοντικές επεκτάσεις – SIGMA Rules Engine

Η διαδικασία έρευνας και μελέτης ενός οργανισμού πριν εγκατασταθεί ένας μηχανισμός SIEM είναι μεγάλη και χρονοβόρα.

Αφού ολοκληρωθεί, οι μηχανικοί ασφάλειας πρέπει να απασχοληθούν για πολλές εργατοώρες ώστε να φτιάξουν αποκωδικοποιητές και κανόνες που θα καλύπτουν πλήρως τις ανάγκες και απαιτήσεις του πελάτη οργανισμού βάσει των ερευνών που έχουν διεξαχθεί. Με κάθε νέο πελάτη οργανισμό και με κάθε νέα εγκατάσταση του SIEM οι μηχανικοί πρέπει να ξοδεύουν τις κατάλληλες εργατοώρες, κάτι που είναι ασύμφορο και για τους δύο εμπλεκόμενους οργανισμούς.

Λύση στο παραπάνω πρόβλημα είναι ένας μηχανισμός που παράγει αυτόματα κανόνες για προϊόντα SIEM βάσει των αναγκών και απαιτήσεων κάθε οργανισμού. Με τη χρήση του SIGMA προτύπου (βλ. 3.1.1), αυτοί οι κανόνες θα είναι αξιοποιήσιμοι από το κάθε προϊόν SIEM που θα μπορεί να τους διαβάσει και να τους αξιοποιήσει.

Η διαδικασία που με τις σημερινές διαδικασίες παίρνει μέρες ή και βδομάδες, θα κρατούσε ώρες. Ο μηχανισμός παραγωγής κανόνων SIGMA θα είχε ως είσοδο τις ανάγκες του οργανισμού και την περιγραφή του τρόπου λειτουργίας του σε καθημερινή βάση. Μέσα σε λίγα λεπτά, αξιοποιώντας γνώση Cyber Threat Intelligence (CTI) (βλ. 3.2), θα έφτιαχνε τους κανόνες που θα προσάρμοζαν το προϊόν SIEM που θα αγόραζε ο οργανισμός και όλες αυτές οι εργατοώρες που σε διαφορετική περίπτωση θα ξόδευαν οι μηχανικοί ασφάλειας, θα μπορούσαν να αξιοποιούνταν σε άλλες εργασίες.

6.2 Μελλοντικές επεκτάσεις – Clustered SIEM αρχιτεκτονική

Όσο το μέγεθος ενός οργανισμού αυξάνεται, αυξάνονται και οι απαιτήσεις που έχει από έναν μηχανισμό όπως τα Security Information and Event Management συστήματα. Σε ακραίες περιπτώσεις όμως, υπάρχουν οργανισμοί που είναι πάρα πολύ μεγάλοι σε τεχνολογικό όγκο που ένα μόνο κεντρικό προϊόν SIEM δεν μπορεί να τις καλύψει όσους πόρους και να του αφιερώσουν. (π.χ. CISCO, AMAZON, GOOGLE)

Η λύση αυτών των εταιριών είναι να έχουν πολλούς από αυτούς τους μηχανισμούς εγκατεστημένους διάσπαρτα ώστε να καλύψουν όσο το καλύτερο δυνατό τους τεχνολογικούς τους πόρους. Αυτό έχει τρομερά αυξημένο κόστος και πολλές φορές υπάρχουν προβλήματα στην ομαλή επικοινωνία των διάσπαρτων αυτών μηχανισμών, με αποτέλεσμα τα κομμάτια των οργανισμών να λειτουργούν πλήρως ανεξάρτητα το ένα από το άλλο σε επίπεδο ασφάλειας.

Λύση στο παραπάνω πρόβλημα θα είναι μια καινούργια αρχιτεκτονική SIEM η οποία θα είχε ένα κεντρικό SIEM σύστημα σαν τον εγκέφαλο της ασφάλειας του οργανισμού και θα υπήρχαν διάσπαρτα σαν αντίγραφα τα κομμάτια του SIEM που απλά συλλέγουν δεδομένα και τα σαρώνουν ώστε να ανιχνευτούν οι απειλές. Έτσι χωρίζοντας τον φόρτο εργασίας σε πολλά επίπεδα, θα γινόταν ένα μαζικό φιλτράρισμα των συμβάντων που θα έφταναν από το ένα επίπεδο στο άλλο, και ο τελικός εγκέφαλος του μηχανισμού SIEM θα είχε να επεξεργαστεί μόνο ένα μικρό υποσύνολο των συμβάντων που θα παίρναν μέρος στο τεχνολογικό οικοσύστημα του οργανισμού. Έτσι, η οπτικοποίηση θα παρέμενε κεντρική ενώ η ανίχνευση θα χωριζόταν σε πολλά συστήματα χωρίς να μειώνεται η ποιότητα και η απόδοση του προϊόντος SIEM.

ΑΝΑΦΟΡΕΣ

[1] Oskars Podzins and Andrejs Romanovs, Dept. of Modeling and Simulation Riga Technical University, Riga, Latvia, “Why SIEM is Irreplaceable in a Secure IT Environment?”, 2019.

[2] Amir Jamil, “The difference between SEM, SIM and SIEM”, March 2010.

[3] A. Williams, “The Future of SIEM – The Market will Begin to Diverge”, Blog, 1st January 2007

[4] Vijay Gurbaxani and Seungjin Whang – “The impact of information systems on organizations and markets” - January 1991.

[5] Maxat Akbanov , Vassilios G.Vassilakis and Michael D.Logothetis – Department of Computer Science, University of York, York, United Kingdom – “Wanna Cry Ransom ware Analysis of Infection”, 2019

[6] SANS Intitute, “A Practical Application of SIM/SEM/SIEM Automating Threat Identification”, 2006

[7] Patent Application Publication, “Method For Simulation Aided Security Event Management”, 2007

[8] Margaret Rouse, Search-Security Tech-Target, “security information management (SIM)”

[9] NIST SP800-92, “Guide to Computer Security Log Management”, 2006

[10] Mariana Hentea, “Intelligent System for Information Security Management: Architecture and Design Issues”, 2017

[11] Kai-Oliver Detken1, Thomas Rix, Carsten Kleiner, Bastian Hellmann, Leonard Renners , “SIEM approach for a higher level of IT security in enterprise networks” , 2017

[12] Alan F. Dutka, “Fundamentals of Data Normalization”, 1989

[13] Sand Dorigo, “Security Information and Event Management”, Master Thesis, 2012

[14] Tony Bradle, Author and Senior Manager of Content Marketing, 2019

[15] Katsaris Dimitrios, “Security Information and Event Management”, Master Thesis, 2014

[16] 2012 Cloud Security Alliance, “SecaaS Implementation Guidance Category 7: Security Information and Event Management”, 2010

[17] Subha, ManageEngine, IT Security Expert, “How to leverage SIEM to meet the GDPR’s requirements”, 2018

[18] General Data Protection Regulation, “GDPR complete regulations list documented”, 2018

[19] H. Debar, France Telecom, D. Curry, Guardian, B. Feinstein, SecureWorks Inc., IETF, RFC 4765, “The Intrusion Detection Message Exchange Format (IDMEF)”, 2007
[20] Daniela Popescul, Universitatea Alexandru Ioan Cuza, “The Confidentiality – Integrity – Accessibility Triad into the Knowledge Security. A Reassessment from the Point of View of the Knowledge Contribution to Innovation”, 2011
[21] K. Kavanagh, M. Nikolett, O. Rochford, “Gartner Magic Quadrant for Security Information and Event Management”, 2014.
[22] David Ferraiolo, D. Richard Kuhn, National Institute of Standards and Technology, “Role-Based Access Controls”, 2009
[23] James L. McClelland, Felix Hill, Maja Rudolph, Jason Baldridge, Hinrich Schütze, Cornell University, “Extending Machine Language Models toward Human-Level Language Understanding”, 2019
[24] Juniper Networks, Log File Sample Content, 2018
[25] Vladlena Benson, Aston Business School, “The state of global cyber security: highlights and key findings”, 2017
[26] Fernando Almeida, Superior Polytechnical Institute of Gaya, “Open Standards And Open Source: Enabling Interoperability”, 2011
[27] John Hubbard, SANS, SecHubb, Tactical Detection and Data Analytics Summit, “Sharing is Caring”, 2018
[28] David Carasso, Splunk’s Chief Mind, “Exploring Splunk”, 2012
[29] J. Arkko, Ericsson, M. Cotton, L. Vegoda, ICANN, IETF, RFC 5737, “IPv4 Address Blocks Reserved for Documentation”, 2010
[30] Florian Roth, NextronSystems, “An Overlooked but Intriguing Sigma Use Case”, 2019
[31] Gartner, Inc., “Threat Intelligence: What is it, and How Can it Protect Youfrom Today’s Advanced Cyber-Attacks?”, 2014
[32] Shackleford Dave, “Who’s Using Cyber threat Intelligence and How?”, 2015.
[33] Dalziel, Henry, “How to Define and Build an Effective Cyber Threat IntelligenceCapability”. 2015.
[34] Troy Mattern, John Felker, Randy Borum, George Bamford, “Operational Levels of Cyber Intelligence”. 2014.
[35] Sara Qamar, Zahid Anwar, Mohammad Ashiqur Rahman, Ehab Al-Shaer, Bei-Tseng Chu. “Data driven analytics for cyber-threat intelligence and information sharing”. 2017.
[36] Wunder John, STIX 2.0 Finish Line, 2017
[37] Barnum Sean, “Standardizing Cyber Threat Intelligence Information with the Structured Threat Information eXpression (STIXGuideTM)”. MITRE, 2014.

[38] OASIS Cyber Threat Intelligence Technical Committee, “STIXGuideTM Version 2.0. Part 2, STIX Objects”, 2017.
[39] Openioc.org, Whitepaper, “Sophisticated Indicators for the Modern Threat Landscape: An introduction to OpenIOC”.
[40] OASIS, “TAXII Introduction and Documentation”, 2020
[41] Debanjan Ghosh, Massachusetts Institude of Technology, Raj Sharman, University at Buffalo New York, Raghav Rao, University of Texas at San Antonio, “Self-healing systems – survey and synthesis”, 2007.
[42] Gabi Dreo Rodosek, Kurt Geihs, Hartmut Schmeck, Burkhard Stiller, “Self-Healing Systems: Foundations and Challenges”.
[43] Avizienis, Laprie, Randell, Landwehr, “Basic concepts and taxonomy of dependable and secure computing”, 2004
[44] MITRE, “Introduction to OVAL”, 2007
[45] MITRE, “Common Vulnerabilities and Exposures – CVE”, 2016
[46] OASIS, “Open Command and Control (OpenC2) Language Specification Version 1.0”, 2019
[47] YARA, Official Guide and Documentation, 2020
[48] Securosis, “Understanding and Selecting SIEM/Log Management”, 2010
[49] Moukafih Nabil, Sabir Soukaina, Abdelmajid Lakbabi, Orhanou Ghizlane, “SIEM Selection Criteria for an efficient contextual security”, 2020
[50] InfoSec, “SIEM Product Comparison Research”
[51] Info-Tech Research Group, “Vendor Landscape: Security Information& Event Management (SIEM)”, 2015
[52] Ertugrul Akbas, “SIEM SureLog Arcsight QRadar LogRhythm AlienVault SolarWinds Performance Comparison”
[53] Atomicorp, “OSSEC documentation”, 2020
[54] Elastic, “Elastic Stack and Product Documentation”, 2020

ΠΑΡΑΡΤΗΜΑΤΑ

OSSEC main configuration [partly 53]

```
<ossec_config>
  <global>
    <email_notification>yes</email_notification>
    <email_to>xxx@xxx.x</email_to>
    <smtp_server>xxxxxxx</smtp_server>
    <email_from>ossecm@thesis-VM</email_from>
    <jsonout_output>yes</jsonout_output>
  </global>

  <rules>
    <include>rules_config.xml</include>
    <include>pam_rules.xml</include>
    <include>sshd_rules.xml</include>
    <include>telnetd_rules.xml</include>
    <include>syslog_rules.xml</include>
    <include>arpwatch_rules.xml</include>
    <include>symantec-av_rules.xml</include>
    <include>symantec-ws_rules.xml</include>
    <include>pix_rules.xml</include>
    <include>named_rules.xml</include>
    <include>smbd_rules.xml</include>
    <include>vsftpd_rules.xml</include>
    <include>pure-ftpd_rules.xml</include>
    <include>proftpd_rules.xml</include>
    <include>ms_ftpd_rules.xml</include>
    <include>ftpd_rules.xml</include>
    <include>hordeimp_rules.xml</include>
    <include>roundcube_rules.xml</include>
    <include>wordpress_rules.xml</include>
    <include>cimserver_rules.xml</include>
    <include>vpopmail_rules.xml</include>
    <include>vmpop3d_rules.xml</include>
    <include>courier_rules.xml</include>
    <include>web_rules.xml</include>
    <include>web_appsec_rules.xml</include>
    <include>apache_rules.xml</include>
    <include>nginx_rules.xml</include>
    <include>php_rules.xml</include>
    <include>mysql_rules.xml</include>
    <include>postgresql_rules.xml</include>
```

```
    <include>ids_rules.xml</include>
    <include>squid_rules.xml</include>
    <include>firewall_rules.xml</include>
    <include>apparmor_rules.xml</include>
    <include>cisco-ios_rules.xml</include>
    <include>netscreenfw_rules.xml</include>
    <include>sonicwall_rules.xml</include>
    <include>postfix_rules.xml</include>
    <include>sendmail_rules.xml</include>
    <include>imapd_rules.xml</include>
    <include>mailscanner_rules.xml</include>
    <include>dovecot_rules.xml</include>
    <include>ms-exchange_rules.xml</include>
    <include>racoon_rules.xml</include>
    <include>vpn_concentrator_rules.xml</include>
    <include>spamd_rules.xml</include>
    <include>msauth_rules.xml</include>
    <include>mcafee_av_rules.xml</include>
    <include>trend-osce_rules.xml</include>
    <include>ms-se_rules.xml</include>
    <!-- <include>policy_rules.xml</include> -->
    <include>zeus_rules.xml</include>
    <include>solaris_bsm_rules.xml</include>
    <include>vmware_rules.xml</include>
    <include>ms_dhcp_rules.xml</include>
    <include>asterisk_rules.xml</include>
    <include>ossec_rules.xml</include>
    <include>attack_rules.xml</include>
    <include>openbsd_rules.xml</include>
    <include>clam_av_rules.xml</include>
    <include>dropbear_rules.xml</include>
    <include>sysmon_rules.xml</include>
    <include>opensmtpd_rules.xml</include>
    <include>exim_rules.xml</include>
    <include>openbsd-dhcpd_rules.xml</include>
    <include>dnsmasq_rules.xml</include>
    <include>nsd_rules.xml</include>
    <include>local_rules.xml</include>
  </rules>

  <syscheck>
    <!-- Frequency that syscheck is executed - default to
every 22 hours -->
    <frequency>79200</frequency>
```

```
    <!-- Directories to check  (perform all possible
verifications) -->
    <directories
check_all="yes">/etc,/usr/bin,/usr/sbin</directories>
    <directories
check_all="yes">/bin,/sbin,/boot</directories>

    <!-- Files/directories to ignore -->
    <ignore>/etc/mtab</ignore>
    <ignore>/etc/mnttab</ignore>
    <ignore>/etc/hosts.deny</ignore>
    <ignore>/etc/mail/statistics</ignore>
    <ignore>/etc/random-seed</ignore>
    <ignore>/etc/adjtime</ignore>
    <ignore>/etc/httpd/logs</ignore>
    <ignore>/etc/utmpx</ignore>
    <ignore>/etc/wtmpx</ignore>
    <ignore>/etc/cups/certs</ignore>
    <ignore>/etc/dumpdates</ignore>
    <ignore>/etc/svc/volatile</ignore>

    <!-- Windows files to ignore -->
    <ignore>C:\WINDOWS/System32/LogFiles</ignore>
    <ignore>C:\WINDOWS/Debug</ignore>
    <ignore>C:\WINDOWS/WindowsUpdate.log</ignore>
    <ignore>C:\WINDOWS/iis6.log</ignore>
    <ignore>C:\WINDOWS/system32/wbem/Logs</ignore>
    <ignore>C:\WINDOWS/system32/wbem/Repository</ignore>
    <ignore>C:\WINDOWS/Prefetch</ignore>
    <ignore>C:\WINDOWS/PCHEALTH/HELPCTR/DataColl</ignore>
    <ignore>C:\WINDOWS/SoftwareDistribution</ignore>
    <ignore>C:\WINDOWS/Temp</ignore>
    <ignore>C:\WINDOWS/system32/config</ignore>
    <ignore>C:\WINDOWS/system32/spool</ignore>
    <ignore>C:\WINDOWS/system32/CatRoot</ignore>
  </syscheck>

  <rootcheck>

<rootkit_files>/var/ossec/etc/shared/rootkit_files.txt</rootk
it_files>

<rootkit_trojans>/var/ossec/etc/shared/rootkit_trojans.txt</r
ootkit_trojans>
```

```
<system_audit>/var/ossec/etc/shared/system_audit_rcl.txt</sys
tem_audit>

<system_audit>/var/ossec/etc/shared/cis_debian_linux_rcl.txt<
/system_audit>

<system_audit>/var/ossec/etc/shared/cis_rhel_linux_rcl.txt</s
ystem_audit>

<system_audit>/var/ossec/etc/shared/cis_rhel5_linux_rcl.txt</
system_audit>
  </rootcheck>

  <global>
    <allow_list>127.0.0.1</allow_list>
    <allow_list>::1</allow_list>
    <allow_list>localhost.localdomain</allow_list>
    <allow_list>127.0.0.53</allow_list>
  </global>

  <remote>
    <connection>syslog</connection>
  </remote>

  <remote>
    <connection>secure</connection>
  </remote>

  <alerts>
    <log_alert_level>1</log_alert_level>
    <email_alert_level>3</email_alert_level>
  </alerts>

  <command>
    <name>host-deny</name>
    <executable>host-deny.sh</executable>
    <expect>srcip</expect>
    <timeout_allowed>yes</timeout_allowed>
  </command>

  <command>
    <name>firewall-drop</name>
    <executable>firewall-drop.sh</executable>
    <expect>srcip</expect>
```

```
    <timeout_allowed>yes</timeout_allowed>
  </command>

  <command>
    <name>disable-account</name>
    <executable>disable-account.sh</executable>
    <expect>user</expect>
    <timeout_allowed>yes</timeout_allowed>
  </command>

  <command>
    <name>restart-ossec</name>
    <executable>restart-ossec.sh</executable>
    <expect></expect>
  </command>


  <command>
    <name>route-null</name>
    <executable>route-null.sh</executable>
    <expect>srcip</expect>
    <timeout_allowed>yes</timeout_allowed>
  </command>

  <!-- Files to monitor (localfiles) -->

  <localfile>
    <log_format>syslog</log_format>
    <location>/var/log/auth.log</location>
  </localfile>

  <localfile>
    <log_format>syslog</log_format>
    <location>/var/log/syslog</location>
  </localfile>

  <localfile>
    <log_format>syslog</log_format>
    <location>/var/log/dpkg.log</location>
  </localfile>

  <localfile>
    <log_format>command</log_format>
    <command>df -P</command>
  </localfile>
```

```
  <localfile>
    <log_format>full_command</log_format>
    <command>netstat -tan |grep LISTEN |egrep -v '(127.0.0.1|
::1)' | sort</command>
  </localfile>

  <localfile>
    <log_format>full_command</log_format>
    <command>last -n 5</command>
  </localfile>
</ossec_config>
```

Sudo decoder [53]

```
<decoder name="sudo">
  <program_name_pcre2>^sudo</program_name_pcre2>
  <pcre2>^[ ]*?(\S+)[ ]:[ ]TTY=\S+[ ];[ ]PWD=(\S+)[ ];[
]USER=(\S+)[ ];[ ]COMMAND=(.+)$|</pcre2>
  <pcre2>^[ ]*?(\S+)[ ]:[ ]TTY=\S+[ ];[ ]PWD=(\S+)[ ];[
]USER=(\S+)[ ];[ ]TSID=\S+[ ];[ ]COMMAND=(.+)$</pcre2>
  <order>dstuser,url,srcuser,status</order>
  <fts>name,dstuser,location</fts>
  <ftscomment>First time user executed the sudo
command</ftscomment>
</decoder>
```

Initial group for sudo messages rule [53]

```
  <rule id="5400" level="0" noalert="1">
    <decoded_as>sudo</decoded_as>
    <description>Initial group for sudo
messages</description>
  </rule>
```

Unauthorized user attempted to use sudo rule [53]

```
  <rule id="5405" level="2">
    <if_sid>5400</if_sid>
    <pcre2>user NOT in sudoers</pcre2>
    <description>Unauthorized user attempted to use
sudo.</description>
  </rule>
```

SSHD messages grouped rule [53]

```
  <rule id="5700" level="0" noalert="1">
    <decoded_as>sshd</decoded_as>
    <description>SSHD messages grouped.</description>
  </rule>
```

SSHD authentication failed rule [53]

```
  <rule id="5716" level="2">
    <if_sid>5700</if_sid>
    <pcre2>^Failed|^error: PAM: Authentication</pcre2>
    <description>SSHD authentication failed.</description>
    <group>authentication_failed,</group>
  </rule>
```

Multiple SSHD authentication failures rule [53]

```
  <rule id="5720" level="3" frequency="6">
    <if_matched_sid>5716</if_matched_sid>
    <same_source_ip />
    <description>Multiple SSHD authentication
failures.</description>
    <group>authentication_failures,</group>
  </rule>
```

Απλό bash script που προσομοιώνει ένα ssh bruteforce attack.

```
#!/bin/bash

while [ true ] ; do
     sshpass –p ‘wrongpassword’ ssh agent@192.168.1.10
     sleep 1
done

exit 0
```