ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ – ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Πρόγραμμα Μεταπτυχιακών Σπουδών

«Κυβερνοασφάλεια και Επιστήμη Δεδομένων»

Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	**Ανάλυση Πολιτικών Κανόνων Τειχών Προστασίας Με Χρήση Τεχνολογίας Natural Language Processing** **Analysis Of Firewall Policy Rules Using Natural Language Processing Technology**
Ονοματεπώνυμο Φοιτητή	**Παναγιώτης Τσακιρίδης**
Πατρώνυμο	**Δημήτριος**
Αριθμός Μητρώου	**ΜΠΚΕΔ21053**
Επιβλέπων	**Παναγιώτης Κοτζανικολάου, Αν. Καθηγητής**

Ημερομηνία Παράδοσης **Μάρτιος 2024**

Τριμελής Εξεταστική Επιτροπή

Χρήστος Δουληγέρης Καθηγητής	Μιχαήλ Ψαράκης Αν. Καθηγητής	Παναγιώτης Κοτζανικολάου Αν. Καθηγητής

Περίληψη

Η παρούσα διπλωματική εργασία επικεντρώνεται στη σύγκριση της ιδανικής πολιτικής ασφαλείας, η οποία παρέχεται από μια εταιρία σε φυσική γλώσσα, με την υπάρχουσα πολιτική ασφαλείας που ανακτάται μέσω της τεχνολογίας GRPC και αναλύεται για εξαγωγή περισσότερων πληροφοριών. Η ιδανική πολιτική ασφαλείας λειτουργεί ως βέλτιστο πρότυπο ασφάλειας και μετατρέπεται σε κανόνες iptables για το περιβάλλον Linux, χρησιμοποιώντας προηγμένες τεχνικές Επεξεργασίας Φυσικής Γλώσσας (NLP) που είναι κλάδος της Τεχνητής Νοημοσύνης (AI).

Η επόμενη φάση περιλαμβάνει την ενδελεχή αξιολόγηση επτά διαφορετικών αλγορίθμων NLP για τη δημιουργία embeddings. Στόχος αυτής της αξιολόγησης είναι η επιλογή του αλγορίθμου που παράγει τα πιο αποδοτικά και ακριβή αποτελέσματα σύγκρισης κανόνων iptables για την ανάλυση περεταίρω των πολιτικών ασφαλείας. Μετά την επιλογή του καταλληλότερου αλγορίθμου, η εργασία προχωρά στη μετατροπή των κανόνων ασφαλείας από τις δύο πολιτικές - την υπάρχουσα και την ιδανική - σε αντίστοιχες αναπαραστάσεις embeddings.

Στη συνέχεια, εφαρμόζεται μια μετρική στα embeddings για τον υπολογισμό του ποσοστού ομοιότητας των επιμέρους κανόνων. Αυτή η λεπτομερής διαδικασία παρέχει μια ακριβή εικόνα των ομοιοτήτων και των διαφορών στους κανόνες ασφαλείας, επιτρέποντας τη σύνθεση ενός συνολικού ποσοστού ομοιότητας των δύο πολιτικών. Τα αποτελέσματα αυτής της συγκριτικής ανάλυσης παρουσιάζονται οπτικοποιημένα, διευκολύνοντας την κατανόηση και επιτρέποντας μια διαφανή και αντικειμενική αξιολόγηση των πολιτικών ασφαλείας.

Μέσω αυτής της καινοτόμου μεθοδολογίας, η διπλωματική εργασία παρέχει ένα ισχυρό εργαλείο για τη βελτίωση των εταιρικών πολιτικών ασφαλείας, ενισχύοντας την ασφάλεια και την αξιοπιστία των δικτυακών υποδομών.

Λέξεις-κλειδιά : Πολιτική Ασφαλείας, GRPC Τεχνολογία, Επεξεργασία Φυσικής Γλώσσας (NLP), Τεχνητή Νοημοσύνη (AI), Iptables, Ανάλυση Δεδομένων, Αλγόριθμοι NLP, Embeddings, Μετρικές Ομοιότητας, Οπτικοποίηση Δεδομένων, Βελτίωση Εταιρικής Ασφάλειας, Δικτυακές Υποδομές

Abstract

This thesis focuses on comparing the ideal security policy, which is provided by an organization-company in natural language, with the existing security policy retrieved using GRPC technology and analyzed to extract more information. The ideal security policy acts as an optimal security model and is converted into iptables rules for the Linux environment, using advanced Natural Language Processing (NLP) techniques, a branch of Artificial Intelligence (AI).

The next phase involves a thorough evaluation of seven different NLP algorithms for generating embeddings. The goal of this evaluation is to select the algorithm that produces the most efficient and accurate iptables rule comparison results for further analysis of security policies. After selecting the most suitable algorithm, the work proceeds by converting the security rules of the two policies - existing and ideal - into corresponding embeddings representations.

Next, a metric is applied to the embeddings to calculate the similarity percentage of each rule. This detailed process provides an accurate picture of the similarities and differences in the security rules, allowing for the composition of an overall similarity percentage of the two policies. The results of this comparative analysis are presented visually, which facilitates understanding and allows for a transparent and objective evaluation of the security policies. Through this innovative methodology, the thesis provides a powerful tool for improving corporate security policies, enhancing the security and reliability of network infrastructures.

Keywords: Security Policy, GRPC Technology, Natural Language Processing (NLP), Artificial Intelligence (AI), iptables, Data Analysis, NLP Algorithms, Embeddings, Similarity Metrics, Data Visualization, Corporate Security Improvement, Network Infrastructures

Περιεχόμενα

1 Εισαγωγή

1.1 Περιγραφή

Η κυβερνοασφάλεια αποτελεί μία από τις πλέον κρίσιμες και δυναμικές περιοχές στον σύγχρονο τεχνολογικό κόσμο. Με την αυξημένη εξάρτηση των επιχειρήσεων και οργανισμών από τις ψηφιακές υποδομές, η ανάγκη για ισχυρές και αξιόπιστες στρατηγικές ασφάλειας είναι πιο επιτακτική από ποτέ.

Στον τομέα της κυβερνοασφάλειας, τα firewalls αποτελούν έναν από τους πρωταρχικούς μηχανισμούς προστασίας των δικτυακών υποδομών. Ως κρίσιμα στοιχεία της δικτυακής ασφάλειας, τα firewalls αναλαμβάνουν την επιτήρηση και τον έλεγχο της εισερχόμενης και εξερχόμενης κίνησης, αποτρέποντας την πρόσβαση ανεπιθύμητων ή επιβλαβών δεδομένων. Ωστόσο, η αυξανόμενη πολυπλοκότητα των κυβερνοαπειλών, όπως οι επιθέσεις ransomware και phishing, έχει επιφέρει νέες προκλήσεις στην αποδοτική διαχείριση και ρύθμιση των firewalls.

Πιο συγκεκριμένα, η κυβερνοασφάλεια, στον σύγχρονο ψηφιακό κόσμο, αντιμετωπίζει συνεχώς αυξανόμενες προκλήσεις. Με ένα αξιοσημείωτο ποσοστό 48% των οργανισμών να αναφέρει αύξηση των κυβερνοεπιθέσεων το 2023 σε σύγκριση με το 2022, το πεδίο της κυβερνοασφάλειας απαιτεί διαρκώς εξελιγμένες λύσεις και στρατηγικές. Η σημασία του ανθρώπινου παράγοντα είναι καθοριστική, καθώς το 59% των εταιριών στον κλάδο της κυβερνοασφάλειας υπογραμμίζουν την υποστελέχωση των ομάδων τους.[1]

Αυτή η αυξημένη πολυπλοκότητα των κυβερνοαπειλών επιβάλλει την ανάγκη για στρατηγικές ασφαλείας που να είναι πιο ισχυρές και αξιόπιστες από ποτέ. Είναι πλέον επιτακτικό για τις εταιρίες και τους οργανισμούς να αντιμετωπίσουν αυτές τις προκλήσεις και να αναπτύξουν προηγμένες λύσεις για την προστασία των ψηφιακών τους περιουσιών. Είναι αυτή η ανάγκη για ασφάλεια που μας οδηγεί στην εξέταση του ρόλου των firewalls και των iptables στην προστασία των δικτυακών υποδομών.

Σε αυτό το πλαίσιο, τα iptables στα Linux συστήματα αποτελούν ένα βασικό εργαλείο για τη διαχείριση της δικτυακής ασφάλειας, καθώς παρέχουν έναν ευέλικτο και δυναμικό τρόπο για τον έλεγχο της εισερχόμενης και εξερχόμενης κίνησης δικτύου. Παρά τις δυνατότητές τους, τα iptables φέρουν τις δικές τους προκλήσεις. Συντακτικά λάθη κατά την καταχώρηση των κανόνων μπορούν να οδηγήσουν σε σοβαρές αδυναμίες ή ακόμα και σε παράλυση των δικτυακών λειτουργιών. Επιπλέον, η σειρά των κανόνων εντός των αλυσίδων είναι ουσιώδης, καθώς οι κανόνες εφαρμόζονται βάσει της σειράς που έχουν οριστεί όπου και ενδέχεται να δημιουργήσουν συγκρούσεις ή ανεπιθύμητες εξαιρέσεις. Η αποσφαλμάτωση και ο εντοπισμός σφαλμάτων στα iptables απαιτεί λεπτομερή εξέταση και γνώση των εντολών, κάτι που είναι χρονοβόρο και απαιτεί ειδίκευση.

Στη σύγχρονη εποχή, όπου η κυβερνοασφάλεια αποτελεί έναν τομέα υψηλής προτεραιότητας, η διαχείριση και ενημέρωση των κανόνων ασφαλείας στα iptables, ιδιαίτερα σε μεγάλες και πολύπλοκες υποδομές, αποτελεί μια σημαντική πρόκληση. Η αναγκαιότητα για έναν αυτοματοποιημένο, αξιόπιστο, εύχρηστο και σύγχρονο τρόπο διαχείρισης αυτών των κανόνων είναι πιο επιτακτική από ποτέ, με στόχο τη διασφάλιση της συνεχούς προστασίας και της αποδοτικότητας των δικτυακών λειτουργιών.

Σε αυτό το πλαίσιο, η ενσωμάτωση της Τεχνητής Νοημοσύνης (AI) στη διαχείριση των iptables ανοίγει νέους δρόμους για την ενίσχυση και την αξιολόγηση των κανόνων ασφαλείας. Μέσω της AI, είναι δυνατή η ανάπτυξη πιο προηγμένων, δυναμικών και ευφυών μοντέλων πολιτικών προστασίας, που δεν βασίζονται μόνο στην παθητική ανάλυση της κίνησης δικτύου, αλλά και στην ενεργητική αξιολόγηση, επεξεργασία και βελτιστοποίηση των κανόνων ασφαλείας.

Η Επεξεργασία Φυσικής Γλώσσας (NLP), ως ένας κλάδος της AI, προσφέρει επιπλέον δυνατότητες στην ανάλυση και διαχείριση των iptables. Με την εφαρμογή τεχνικών NLP, είναι δυνατή η ερμηνεία και η αυτοματοποίηση της διαμόρφωσης κανόνων ασφαλείας από περιγραφές φυσικής γλώσσας, μειώνοντας τον κίνδυνο ανθρώπινων λαθών. Επιπρόσθετα, η χρήση των embeddings που παράγονται από τους αλγορίθμους της NLP μπορεί να

διευκολύνει την ανίχνευση του βαθμού ομοιότητας ή διαφοράς μεταξύ των τρεχόντων κανόνων και ενός ιδανικού συνόλου κανόνων ασφαλείας. Αυτό επιτρέπει την ακριβή αξιολόγηση της απόδοσης και της αποτελεσματικότητας των τρεχόντων κανόνων σε σχέση με τις καλύτερες πρακτικές ή τις ενδεδειγμένες προδιαγραφές ασφαλείας. Σε αυτό το σημείο αξίζει να σημειώσουμε πως με τον όρο embeddings αναφερόμαστε σε αναπαραστάσεις λέξεων ή κειμένου που έχουν δημιουργηθεί από αλγορίθμους Τεχνητής Νοημοσύνης (AI) και είναι σχεδιασμένες να περιλαμβάνουν τη σημασιολογία και τον συντακτικό ρόλο των λέξεων. Αυτές οι αναπαραστάσεις μπορούν να είναι πολύ χρήσιμες για διάφορες εφαρμογές NLP, όπως η αναζήτηση, η σημασιολογική ανάλυση, η ταξινόμηση κειμένου, η μετάφραση, η συσταδοποίηση, όπως φυσικά και στην περίπτωση που παρουσιάζεται σε αυτήν την εργασία.

1.2 Σκοπός

Ο κύριος σκοπός της παρούσας διπλωματικής εργασίας εστιάζει στην ανάπτυξη και υλοποίηση μιας εξελιγμένης μεθοδολογίας, όσον αφορά την αυτοματοποιημένη αξιολόγηση της πολιτικής ασφαλείας ενός εταιρικού περιβάλλοντος. Αυτή η μεθοδολογία στηρίζεται στη χρήση προηγμένων τεχνολογιών Τεχνητής Νοημοσύνης (AI) και, πιο συγκεκριμένα, στις τεχνικές της Επεξεργασίας Φυσικής Γλώσσας (NLP).

Στην καρδιά αυτής της μεθοδολογίας βρίσκεται η χρήση της τεχνολογίας GRPC για την απόκτηση των υφιστάμενων κανόνων ασφαλείας των iptables από την εταιρεία. Αυτή η διαδικασία επιτρέπει την αποδοτική και άμεση συλλογή δεδομένων, καθιστώντας το έδαφος για μια πλήρη αξιολόγηση. Παράλληλα, η εταιρεία παρέχει την ιδανική πολιτική ασφαλείας σε φυσική γλώσσα, η οποία χρησιμοποιείται ως βάση σύγκρισης. Η ιδανική αυτή πολιτική ασφαλείας αναπτύσσεται και μετατρέπεται σε κανόνες iptables μέσω της εφαρμογής τεχνικών NLP, επιτρέποντας την ακριβή σύγκριση μεταξύ των υπαρχόντων και των ιδανικών κανόνων ασφαλείας.

Καθοριστικής σημασίας στη διπλωματική εργασία αποτελεί η αξιολόγηση επτά διαφορετικών αλγορίθμων NLP για την παραγωγή embeddings πάνω σε κανόνες iptables. Αν και όλοι οι αλγόριθμοι εστιάζουν στην ανίχνευση ομοιοτήτων μεταξύ των κανόνων iptables, τα τελικά αποτελέσματα που παρέχουν διαφέρουν σημαντικά σε ποιότητα και ακρίβεια. Η επιλογή του καταλληλότερου αλγορίθμου θα βασιστεί στην ικανότητά του να παρέχει τα πιο ακριβή και εύστοχα αποτελέσματα σύγκρισης.

Στην συνέχεια, τα embeddings που παράγονται από τον επιλεγμένο αλγόριθμο θα υποστούν σύγκριση με μια συγκεκριμένη μετρική, προκειμένου να αξιολογηθεί η τρέχουσα πολιτική ασφαλείας της εταιρείας σε σχέση με την ιδανική. Αυτή η ανάλυση θα καταλήγει στην παραγωγή ενός ποσοστού ομοιότητας, προσφέροντας μια σαφή και ποσοτικά καθορισμένη εικόνα των αποκλίσεων μεταξύ των δύο πολιτικών. Επιπρόσθετα, η οπτικοποίηση των αποτελεσμάτων θα προσδώσει μεγαλύτερη διαύγεια και κατανόηση στη συγκριτική ανάλυση, παρέχοντας μια ολοκληρωμένη προσέγγιση στην αξιολόγηση της πολιτικής ασφαλείας.

Συνολικά, η εργασία αυτή στοχεύει στη δημιουργία ενός πρακτικού εργαλείου για την ενίσχυση της κυβερνοασφάλειας σε επίπεδο πολιτικών ασφάλειας οργανισμών και εταιριών. Μέσω της αξιολόγησης και σύγκρισης των πολιτικών ασφαλείας με χρήση AI και NLP, η εργασία αναμένεται να προσφέρει σημαντικές βελτιώσεις στην ακρίβεια και αποδοτικότητα της αξιολόγησης της ασφάλειας στις εταιρίες-οργανισμούς.

1.3 Δομή

Η παρούσα διπλωματική εργασία διαρθρώνεται σε καθοριστικές ενότητες, κάθε μία από τις οποίες αποτελεί ένα βήμα προς την επίτευξη του στόχου μας - την ενίσχυση της κυβερνοασφάλειας μέσω της Τεχνητής Νοημοσύνης και της επεξεργασίας φυσικής γλώσσας. Η κάθε ενότητα αναπτύσσεται μεθοδικά, προσφέροντας βαθιά κατανόηση σε διάφορες πτυχές του έργου μας και καταλήγει στην παροχή συμπερασμάτων για μελλοντική έρευνα.

Η δομή της εργασίας παρατίθεται εν συντομία ως εξής:

- Στο Κεφάλαιο 2 εξετάζονται προηγούμενες σχετικές μελέτες και ερευνητικές εργασίες.
- Στο Κεφάλαιο 3 αναλύονται η λήψη της υπάρχουσας πολιτικής ασφαλείας μέσω gRPC, καθώς και η μετατροπή της ιδανικής πολιτικής σε κανόνες iptables μέσω NLP.
- Στο Κεφάλαιο 4 αξιολογούνται οι αλγόριθμοι NLP παραγωγής embenddings.
- Στο Κεφάλαιο 5 εξετάζεται η μεθοδολογία σύγκρισης των πολιτικών ασφαλείας και η οπτικοποίηση των αποτελεσμάτων.
- Στο Κεφάλαιο 6 παρουσιάζεται ο τρόπος υλοποίησης της σύγκρισης των πολιτικών ασφαλείας.
- Στο Κεφάλαιο 7 αναλύεται η εφαρμογή της μεθοδολογίας σε μια πραγματική εταιρική περίπτωση.
- Στο Κεφάλαιο 8 παρουσιάζονται τα κύρια συμπεράσματα και οι προτάσεις για μελλοντική έρευνα.
- Στο Κεφάλαιο 9 περιλαμβάνονται όλες οι αναφορές και πηγές που χρησιμοποιήθηκαν.

2 Σχετικές Έρευνες

Η χρήση Τεχνητής Νοημοσύνης σε κανόνες firewall είναι διαδεδομένη τα τελευταία χρόνια λόγω της αυξημένης ανάγκης για πιο εξελιγμένη ασφάλεια δικτύου αλλά και για την εφαρμογή πιο σύγχρονων και έγκυρων μεθόδων ανάλυσης πολιτικών ασφαλείας. Η ενσωμάτωση της AI στα συστήματα firewall επιτρέπει την αυτοματοποίηση και την ταχύτερη απόκριση σε κυβερνοαπειλές, μειώνοντας τον κίνδυνο ανθρώπινων λαθών. Αρκετές μελέτες έχουν επικεντρωθεί σε αλγόριθμους Τεχνητής Νοημοσύνης που μπορούν να αναλύσουν μεγάλες ποσότητες δεδομένων κυκλοφορίας, να αναγνωρίζουν ανωμαλίες και να προσαρμόζουν δυναμικά τους κανόνες ασφαλείας.
Πιο συγκεκριμένα :

Οι Erdem Ucar et al. [2] μελετάνε την ανίχνευση ανωμαλιών στο firewall σύμφωνα με επιδόσεις αλγορίθμων ταξινόμησης μηχανικής μάθησης. Ο αλγόριθμος k-Nearest Neighbors (kNN) έδειξε υψηλή απόδοση στην ανάλυση τειχών προστασίας, ενώ η χρήση βάσεων δεδομένων φαίνεται να επιφέρει πλεονεκτήματα. Η μελέτη επίσης υπογραμμίζει ότι μέθοδοι υπολογισμού υψηλής απόδοσης μπορούν να μειώσουν τον χρόνο ανάλυσης.

Η έρευνα των Cornelius Diekmann et al. [3] πραγματοποιεί στατική ανάλυση των κανόνων Iptables με χρήση του εργαλείου Isabelle/HOL. Πιο συγκεκριμένα, αναλύουν τη σημασιολογία πίσω από τη συμπεριφορά των κανόνων Iptables και παρουσιάζουν διαδικασίες απλοποίησης, μετατρέποντας την γλώσσα των Iptables σε απλούστερο μοντέλο. Η αξιολόγηση της εργασίας σε πραγματικά σετ κανόνων δείχνει ότι το πλαίσιο παρέχει ενδιαφέροντα αποτελέσματα και μπορεί να βοηθήσει ή να υπερτερήσει άλλων πλαισίων στατικής ανάλυσης.

Η πρόταση των Sergio Rivera et al. [4] για την ενδιάμεση γλώσσα POLANCO αποτελεί μία σημαντική προσθήκη στον τομέα της διαχείρισης δικτυακών πολιτικών. Η POLANCO είναι σχεδιασμένη για να μετατρέπει πολιτικές από φυσική γλώσσα σε μορφή που μπορεί να μεταφραστεί αυτόματα σε κανόνες και ενέργειες δικτυακής διαμόρφωσης λογισμικού (Software-Defined Networking, SDN). Η κύρια δυνατότητα του POLANCO είναι η μετάφραση πολιτικών σε κανόνες SDN, με στόχο την αυτόματη επιβολή των πολιτικών. Αυτό είναι ιδιαίτερα σημαντικό σε περιβάλλοντα όπως τα πανεπιστήμια, όπου υπάρχουν ποικίλες και συχνά αλλαγές στις δικτυακές πολιτικές. Επιπλέον, το POLANCO μπορεί να προσαρμόσει τους κανόνες SDN με βάση τις αλλαγές στην κατάσταση του δικτύου, προσφέροντας μεγαλύτερη ευελιξία και ανταπόκριση στις ανάγκες του δικτύου.

Η πρόταση του Pinyi Shi [5], παρουσιάζει έναν μηχανισμό ανάλυσης πολιτικής δικτύου (Network Policy Analyzer-NPA) ο οποίος εξετάζει τη ποιότητα των πολιτικών δικτύου και εξάγει μια αναφορά που μπορεί να δοθεί στους ενδιαφερόμενους για να βελτιώσουν τις πολιτικές τους. Επιπρόσθετα, παρουσιάζεται ένα chatbox (Network Policy Conversation Engine-NPCE) για τους χειριστές δικτύων για να κάνουν ερωτήσεις σε φυσικές γλώσσες που

ελέγχουν εάν υπάρχει παραβίαση πολιτικής στο δίκτυο. Ακόμα, εξερευνά την κατανόηση των συνδέσεων και των επιχειρηματικών σχέσεων των Διαδικτυακών-Αυτόνομων Συστημάτων (ASes) και τέλος παρουσιάζει πως διαχειρίζεται με επιτυχία πολιτικές δικτύου σε ένα υβριδικό δίκτυο SDN.

Η προσέγγιση του Pinyi Shi [6], επιτρέπει στους χρήστες να ελέγχουν παραβάσεις της πολιτικής ασφάλειας δικτύου μέσω ενός συστήματος (Network Policy Conversation Engine - NPCE). Πιο συγκεκριμένα, δημιουργεί Μηχανισμό Συνομιλίας Πολιτικής Δικτύου όπου μεταφράζει queries φυσικής γλώσσας σε queries μιας database όπου περιλαμβάνει την κυκλοφορία δικτύου. Επιπρόσθετα οι χρήστες μπορούν να κάνουν ερωτήσεις σε φυσική γλώσσα χωρίς να γνωρίζουν λεπτομέρειες των queries. Ακόμα, ένα επίπεδο αντιστοίχισης εξασφαλίζει τη μετάφραση των queries των χρηστών σε διάφορα queries βάσης δεδομένων και τέλος πολλές χρήσιμες πληροφορίες σε ερωτήσεις φυσικής γλώσσας μπορούν να εξαχθούν για να δημιουργήσουν νέα queries.

Οι Masoud Narouei et al. [7] προτείνουν ένα νέο πλαίσιο για την εξαγωγή των πολιτικών ελέγχου πρόσβασης (ACP-access control policies) από έγγραφα φυσικής γλώσσας χωρίς περιορισμούς χρησιμοποιώντας σημασιολογική επισήμανση ρόλων (SRL-semantic role labeling).

Οι Ugur Unal et al. [8] προτείνουν το AnomalyAdapters (AAs) που είναι ένα επεκτάσιμο μοντέλο ανίχνευσης πολλαπλών ανωμαλιών σε εργασίες. Χρησιμοποιεί προεκπαιδευμένο μοντέλο transformers για την κωδικοποίηση μιας ακολουθίας καταγραφής και χρησιμοποιεί προσαρμογείς για να μάθει μια δομή καταγραφής και τους τύπους ανωμαλιών.

Η προσέγγιση των Philip Huff et al. [9] εστιάζει σε μια νέα προσέγγιση για την αξιολόγηση του κινδύνου λογισμικού ανάλογα με το δίκτυο και τις ρυθμίσεις του τείχους προστασίας. Χρησιμοποιώντας μηχανική μάθηση και επεξεργασία φυσικής γλώσσας, αυτόματα συνδυάζει τα χαρακτηριστικά των ευπαθειών με τα δίκτυα, τους στόχους και τις απειλές για να εκτιμήσει τη δυνατότητα εκμετάλλευσης των ευπαθειών. Τα αποτελέσματα δείχνουν ότι η μέθοδος μπορεί να αναγνωρίσει με ακρίβεια τις υπηρεσίες δικτύου και να αναλύσει την προσβασιμότητα του δικτύου, ενώ επισημαίνει ότι λίγες μόνο ευπάθειες αποτελούν πραγματικό κίνδυνο, αλλά εάν παραμείνουν χωρίς να επιλυθούν, μπορεί να επεκταθούν ανατρέποντας τα αντίμετρα του τείχους προστασίας.

Στο πλαίσιο της έρευνάς μας, εξερευνάται η μετατροπή κανόνων δικτύου από φυσική γλώσσα σε iptables μέσω της επεξεργασίας φυσικής γλώσσας(NLP), σε συνδυασμό με τη σύγκριση αυτών των κανόνων με τους εν χρήση κανόνες σε Linux συστήματα. Η σύγκριση βασίζεται στη μετατροπή των κανόνων σε embeddings με τη χρήση αλγορίθμων επεξεργασίας φυσικής γλώσσας (NLP) και την αξιολόγηση του ποσοστού ομοιότητας μέσω σχετικών μετρικών.

Κατ' αρχάς, οι προσεγγίσεις των Erdem Ucar et al. και Philip Huff et al. εστιάζουν σε ανίχνευση ανωμαλιών και αξιολόγηση κινδύνου λογισμικού αντίστοιχα, ενώ η δική μας εργασία αποκλίνει προς τη μετατροπή και τη σύγκριση κανόνων δικτύου, παρέχοντας ένα πιο άμεσο εργαλείο για τη διαχείριση της ασφάλειας δικτύου.

Στον τομέα της μετατροπής και ανάλυσης δικτυακών πολιτικών, η έρευνα των Cornelius Diekmann et al. και Sergio Rivera et al. αφορά τη στατική ανάλυση των κανόνων iptables και τη μετάφραση δικτυακών πολιτικών σε SDN αντίστοιχα. Η δική μας προσέγγιση επικεντρώνεται στη δυναμική μετατροπή κανόνων σε Iptables και στην αξιολόγηση της ομοιότητας με τους υπάρχοντες κανόνες, προσφέροντας μία άμεση και πρακτική λύση στη διαχείριση των δικτυακών πολιτικών.

Όσον αφορά την ανάλυση και βελτίωση της ποιότητας των πολιτικών δικτύου, η προσέγγιση του Pinyi Shi με το Network Policy Analyzer-NPA συμπληρώνεται από την εργασία μας, καθώς προσφέρεται μια αυτοματοποιημένη λύση για τη μετατροπή και αξιολόγηση των κανόνων δικτύου.

Τέλος, η δική μας έρευνα είναι συγγενική με τις προσεγγίσεις των Masoud Narouei et al. και Ugur Unal et al σε σχέση με την εξαγωγή πολιτικών από φυσική γλώσσα και την ανίχνευση ανωμαλιών αντίστοιχα, αλλά προσθέτει την καινοτόμο δυνατότητα της άμεσης

σύγκρισης και αξιολόγησης της ομοιότητας των κανόνων iptables, εμπλουτίζοντας έτσι τις τεχνικές διαχείρισης των δικτυακών πολιτικών.

3 Μεθοδολογία Επεξεργασίας Κανόνων Iptables

Σε αυτό το κεφάλαιο περιγράφεται η μεθοδολογία την οποία και ακολουθήσαμε για να φέρουμε τα δεδομένα των δύο πολιτικών στην ίδια μορφή έτσι ώστε να φτάσουμε στο τελικό αποτέλεσμα που είναι η σύγκριση της υπάρχουσας και της ιδανικής πολιτικής ασφαλείας.

Βήμα 1 : Αρχικά, πραγματοποιούμε τη λήψη της υπάρχουσας πολιτικής, επιτρέποντας τη συλλογή σημαντικών πληροφοριών για περαιτέρω ανάλυση και κατανόηση της.

Βήμα 2 : Ακολούθως, αφού έχει δοθεί η ιδανική πολιτική από την εταιρία την μετατρέπουμε σε κανόνες Iptables προετοιμάζοντας το έδαφος για μια ολοκληρωμένη ανάλυση και σύγκριση με την υπάρχουσα πολιτική.

3.1 Εισαγωγή Στα Iptables

Τα Iptables αποτελούν ένα ισχυρό εργαλείο για τη διαχείριση της κυκλοφορίας δεδομένων σε ένα δίκτυο, κυρίως στα συστήματα που βασίζονται σε Linux. Σκοπός τους είναι η παροχή ενός συστήματος φίλτρων για την ελέγχουσα ροή δεδομένων μέσω του δικτύου. Αυτό επιτυγχάνεται μέσω της εφαρμογής κανόνων, οι οποίοι διαμορφώνουν τη συμπεριφορά της εισερχόμενης και εξερχόμενης κυκλοφορίας στο δίκτυο.

3.1.1 Κύριες Λειτουργίες των Iptables

Τα Iptables προσφέρουν μια σειρά από λειτουργίες για τη διαχείριση δικτυακής κυκλοφορίας και ασφάλειας. Αυτές οι λειτουργίες περιλαμβάνουν:

- **Φιλτράρισμα Πακέτων:** Αυτή η λειτουργία επιτρέπει τον έλεγχο της διέλευσης πακέτων δεδομένων μέσω του δικτύου, αποφασίζοντας ποια πακέτα θα επιτραπούν και ποια θα απορριφθούν.
- **Ελέγχους Ασφαλείας:** Οι κανόνες των Iptables βοηθούν στην προστασία του δικτύου από ανεπιθύμητες ή επιβλαβείς συνδέσεις, αυξάνοντας έτσι την γενική ασφάλεια του δικτύου.
- **Δρομολόγηση Δεδομένων:** Μέσω των Iptables, ο διαχειριστής μπορεί να καθοδηγήσει την κυκλοφορία δεδομένων σε συγκεκριμένες διαδρομές εντός του δικτύου, βελτιστοποιώντας την απόδοση και την αξιοπιστία.
- **Μετάφραση Διευθύνσεων Δικτύου (NAT - Network Address Translation):** Τα Iptables μπορούν να χρησιμοποιηθούν για τη μετατροπή των διευθύνσεων των πηγαίων ή προορισμών IP των πακέτων. Αυτό είναι χρήσιμο για πολλά σενάρια, όπως η κοινή χρήση μιας δημόσιας IP διεύθυνσης από πολλά ιδιωτικά δίκτυα.
- **Λογοδοσία και Παρακολούθηση:** Τα Iptables επιτρέπουν την καταγραφή δεδομένων για την κυκλοφορία δικτύου, παρέχοντας στατιστικά και πληροφορίες για την ανάλυση και τον έλεγχο της κυκλοφορίας.
- **Περιορισμός Ρυθμού (Rate Limiting):** Με τη χρήση των Iptables, μπορεί να γίνει ο περιορισμός του ρυθμού μετάδοσης για συγκεκριμένες ροές δεδομένων, χρήσιμο για την αποφυγή καταχρήσεων των δικτυακών πόρων ή για τη διαχείριση της επιβάρυνσης δικτύου.
- **Ελέγχους Κατάστασης Συνδέσεων (Connection State Tracking):** Τα Iptables μπορούν να παρακολουθούν την κατάσταση των δικτυακών συνδέσεων (π.χ., νέες,

υπάρχουσες, ή διακοπές συνδέσεων) και να εφαρμόζουν κανόνες βάσει αυτής της κατάστασης.

Αυτές οι λειτουργίες αποτελούν τον πυρήνα των δυνατοτήτων των Iptables και είναι ζωτικής σημασίας για την ασφαλή και αποτελεσματική διαχείριση του δικτύου[10].

3.1.2 Σημαντικές Έννοιες

Στα Iptables, κάποιες έννοιες είναι ζωτικής σημασίας για την κατανόηση και την αποτελεσματική εφαρμογή τους. Αυτές περιλαμβάνουν:

- **Κανόνες (Rules):** Οι κανόνες είναι οδηγίες που καθορίζουν τον τρόπο διαχείρισης εισερχόμενων ή εξερχόμενων πακέτων. Αυτοί μπορούν να συμπεριλάβουν κριτήρια όπως οι διευθύνσεις IP, οι θύρες πρωτοκόλλων, και τύποι πακέτων.
- **Αλυσίδες (Chains):** Οι αλυσίδες είναι ομάδες κανόνων που εφαρμόζονται σε διάφορα σημεία επεξεργασίας των πακέτων. Οι κύριες αλυσίδες είναι η INPUT (για εισερχόμενα πακέτα), OUTPUT (για εξερχόμενα πακέτα), και FORWARD (για πακέτα που διοχετεύονται μέσω του δικτύου).
- **Πολιτικές (Policies):** Οι πολιτικές καθορίζουν την προεπιλεγμένη συμπεριφορά μιας αλυσίδας, αν κανένας από τους κανόνες δεν εφαρμόζεται σε ένα συγκεκριμένο πακέτο.
- **Μονάδες Επέκτασης (Extensions):** Τα Iptables υποστηρίζουν διάφορες επεκτάσεις που επιτρέπουν πιο πολύπλοκες λειτουργίες, όπως η εξέταση πακέτων βάσει τύπων πρωτοκόλλων ή η δημιουργία πιο εξελιγμένων κανόνων φιλτραρίσματος.
- **Ταμπλό Κανόνων (Tables):** Τα Iptables χρησιμοποιούν διάφορα ταμπλό (όπως το 'filter', 'nat', και 'mangle') για την οργάνωση των κανόνων ανάλογα με την λειτουργία τους.
- **Κριτήρια Επιλογής (Match Criteria):** Είναι οι συνθήκες που πρέπει να πληροί ένα πακέτο για να εφαρμοστεί σε αυτό ένας συγκεκριμένος κανόνας. Περιλαμβάνουν παραμέτρους όπως οι διευθύνσεις IP, οι θύρες, τα πρωτόκολλα, και άλλα χαρακτηριστικά των πακέτων.
- **Δράσεις (Actions/Targets):** Ένα πακέτο μπορεί να υποβληθεί σε διάφορες δράσεις. Οι πιο συνηθισμένες δράσεις περιλαμβάνουν το ACCEPT (αποδοχή του πακέτου), DROP (απόρριψη του πακέτου), ή REJECT (απόρριψη του πακέτου με απάντηση λάθους).
- **Καταγραφή (Logging):** Αυτή η λειτουργία επιτρέπει την καταγραφή πληροφοριών για συγκεκριμένα πακέτα που πληρούν συγκεκριμένα κριτήρια. Είναι χρήσιμη για την ανάλυση δικτυακής κυκλοφορίας και για σκοπούς ασφαλείας.
- **Κατάσταση Πακέτου (Packet State):** Αναφέρεται στην κατάσταση μιας σύνδεσης, όπως NEW (για νέες συνδέσεις), ESTABLISHED (για ήδη υφιστάμενες συνδέσεις), και RELATED (για πακέτα που σχετίζονται με ήδη υφιστάμενες συνδέσεις).
- **Περιορισμοί Πακέτων (Packet Mangling):** Αυτή η διαδικασία επιτρέπει την τροποποίηση ορισμένων πεδίων μέσα στα δικτυακά πακέτα. Χρησιμοποιείται για ειδικές διαδικασίες διαχείρισης της κυκλοφορίας[11].

Η κατανόηση αυτών των εννοιών είναι απαραίτητη για την αποτελεσματική χρήση των Iptables, καθώς επιτρέπει στους διαχειριστές δικτύων να δημιουργήσουν πιο εξελιγμένους και στοχευμένους κανόνες για την επίτευξη της επιθυμητής ασφάλειας και διαχείρισης του δικτύου τους[12].

3.1.3 Πρακτική Εφαρμογή

Η πρακτική εφαρμογή των Iptables στο πεδίο της δικτυακής κίνησης περιλαμβάνει την εφαρμογή σεναρίων και τεχνικών που αντικατοπτρίζουν πραγματικές ανάγκες ασφαλείας και διαχείρισης. Παρακάτω αναφέρονται κάποιες τυπικές εφαρμογές:

- **Ασφάλεια Διακομιστή Web:** Δημιουργία κανόνων που περιορίζουν την πρόσβαση στον διακομιστή μόνο από συγκεκριμένες διευθύνσεις IP ή δικτυακές ζώνες, αποτρέποντας παράνομες προσπάθειες πρόσβασης.
- **Κανόνες για VPN Διασύνδεση:** Ρύθμιση των Iptables για να επιτρέπουν κυκλοφορία μόνο μέσω συγκεκριμένων VPN θυρών, ενισχύοντας την ασφάλεια των εξ αποστάσεως συνδέσεων.
- **Περιορισμός DoS Επιθέσεων:** Εφαρμογή κανόνων που ανιχνεύουν και περιορίζουν τον υπερβολικό αριθμό αιτημάτων από μία διεύθυνση IP, αποτρέποντας επιθέσεις τύπου Denial-of-Service.
- **Διαμόρφωση Port Forwarding:** Ρύθμιση των Iptables για να ανακατευθύνουν την κυκλοφορία από μία θύρα σε άλλη, χρήσιμο για τη δημιουργία ασφαλέστερων περιβαλλόντων δικτύωσης.
- **Εφαρμογή Πολιτικών Πρόσβασης:** Χρήση των Iptables για τον ορισμό συγκεκριμένων πολιτικών πρόσβασης για εφαρμογές ή υπηρεσίες, καθορίζοντας ποιες δικτυακές αιτήσεις είναι επιτρεπτές.
- **Παραμετροποίηση Βάσει Σεναρίων:** Χρήση αυτοματοποιημένων σεναρίων για την εφαρμογή σύνθετων διαμορφώσεων στα Iptables, εξασφαλίζοντας ευελιξία και επαναληψιμότητα στις διαχειριστικές διαδικασίες[13].

Κάθε μία από αυτές τις εφαρμογές απαιτεί συγκεκριμένη τεχνική εμπειρία και κατανόηση τόσο των δικτυακών αρχών όσο και των λειτουργικών δυνατοτήτων των Iptables.

3.2 Λήψη Υπάρχουσας Πολιτικής Ασφαλείας Μέσω Τεχνολογίας GRPC

Η πρώτη φάση της εργασίας μας περιλαμβάνει τη συλλογή δεδομένων, από ένα σύστημα Linux που χρησιμοποιεί κανόνες Iptables, σε ένα win 10 μηχάνημα όπου και θα τους επεξεργαστεί. Πιο συγκεκριμένα συλλέγουμε την υπάρχουσα πολιτική ασφαλείας δηλαδή όλους τους κανόνες Iptables που εφαρμόζει το linux μηχάνημα σε μορφή txt αρχείου.

Η συλλογή δεδομένων αποτελεί κρίσιμη διαδικασία για την ανάλυση ασφαλείας ενός συστήματος. Στο πλαίσιο αυτό, η τεχνολογία gRPC (Google Remote Procedure Call) αποτελεί ένα αποτελεσματικό μέσο για την ανταλλαγή δεδομένων μεταξύ διαφορετικών συστημάτων.
Το gRPC είναι ένα ανοιχτού κώδικα πρωτόκολλο αλληλεπίδρασης απομακρυσμένων διακομιστών και πελατών που αναπτύχθηκε από την Google. Το όνομα "gRPC" αντιπροσωπεύει το "g" που προέρχεται από το "Google", ενώ το "RPC" σημαίνει "Remote Procedure Call" (Απομακρυσμένη Διαδικασία Κλήσης). Αυτό το πρωτόκολλο επιτρέπει σε διαφορετικά συστήματα να επικοινωνούν μεταξύ τους και να ανταλλάσσουν δεδομένα. Η βασική ιδέα πίσω από το gRPC είναι η δυνατότητα κλήσης συναρτήσεων ή μεθόδων σε ένα απομακρυσμένο σύστημα χωρίς την ανάγκη για λεπτομερείς υποδομές και πολυπλοκότητα. Ακόμα, βασίζεται στο πρότυπο HTTP/2 για τη μεταφορά δεδομένων, το οποίο προσφέρει βελτιωμένη απόδοση και ασφάλεια σε σχέση με το προηγούμενο πρότυπο HTTP/1.1. Επιπλέον, υποστηρίζει διάφορες γλώσσες προγραμματισμού, όπως το C++, Java, Python, Go, και πολλές άλλες.

Επιπρόσθετα, η ασφάλεια των δεδομένων κατά τη μετάδοση είναι πρωταρχικής σημασίας. Η αποστολή αρχείων μέσω δικτύων, ειδικά σε ανοιχτά ή δημόσια δίκτυα, εγείρει σημαντικά ζητήματα ασφάλειας, όπως την παρεμβολή τρίτων, την παραβίαση δεδομένων και την απώλεια εμπιστευτικότητας. Για την αντιμετώπιση των παραπάνω προκλήσεων,

εφαρμόστηκε η τεχνολογία TLS στη διαδικασία αποστολής αρχείων. Η TLS είναι μία ευρέως αναγνωρισμένη τεχνολογία που παρέχει κρυπτογράφηση στην επικοινωνία δεδομένων στο επίπεδο της μεταφοράς, διασφαλίζοντας έτσι ότι τα δεδομένα παραμένουν ακέραια και εμπιστευτικά κατά τη μετάδοση. Τέλος, υποστηρίζει διάφορα είδη αυθεντικοποίησης, κρυπτογράφησης και άλλες προηγμένες λειτουργίες ασφαλείας που το καθιστούν ιδανικό για εφαρμογές όπου η ασφάλεια και η απόδοση είναι κρίσιμες[14].

3.3 Αναζήτηση Θυρών Και Τεχνολογιών

Η ασφάλεια των δικτύων αποτελεί προτεραιότητα για κάθε οργανισμό. Στο πλαίσιο αυτό, χρειάζεται ανάλυση της υπάρχουσας πολιτικής ασφαλείας έτσι ώστε να εντοπιστούν ποιες θύρες και τεχνολογίες χρησιμοποιούνται έτσι ώστε να έχουμε μια πρότερη γνώση της υπάρχουσας πολιτικής.

Αφού έχουμε λάβει το αρχείο με την υπάρχουσα πολιτική ασφαλείας πλέον είμαστε σε θέση να το επεξεργαστούμε.

Στην αρχή, το αρχείο αναγνωρίζεται και διαβάζεται με τη βοήθεια γλώσσας προγραμματισμού Python[15]. Ο κώδικας αναζητά τις εγγραφές κανόνων στο αρχείο και εντοπίζει τις θύρες που αυτοί αναφέρονται. Ακολούθως, κάθε θύρα που εντοπίζεται αντιστοιχίζεται με συγκεκριμένες τεχνολογίες ή υπηρεσίες.

Η διαδικασία αυτή δεν είναι μόνον ένα τεχνικό βήμα, αλλά και ένα σημαντικό εργαλείο για τη βελτίωση της ασφάλειας του δικτύου. Αναγνωρίζοντας τις θύρες και κατανοώντας τις συσχετισμένες τεχνολογίες, οι διαχειριστές δικτύου είναι σε θέση να προσδιορίσουν πιθανές ευπάθειες και να λάβουν κατάλληλα μέτρα για την προστασία του δικτύου από απειλές και επιθέσεις.

Στην ουσία, η αναζήτηση θυρών και τεχνολογιών στο αρχείο αποτελεί σημαντικό κομμάτι της διαδικασίας διαχείρισης ασφάλειας δικτύου, που συντελεί στην προστασία των ευαίσθητων δεδομένων και στην διασφάλιση της σταθερότητας του συστήματος.

3.4 Μετατροπή Ιδανικής Πολιτικής Σε Κανόνες Iptables Με Χρήση Τεχνολογίας NLP

Σε αυτό το κεφάλαιο θα αξιοποιήσουμε την τεχνολογία NLP για να μετατρέψουμε κανόνες από φυσική γλώσσα σε κανόνες Iptables.

3.4.1 Εισαγωγή Στο NLP

Η Επεξεργασία Φυσικής Γλώσσας (Natural Language Processing - NLP) αποτελεί έναν από τους πιο συναρπαστικούς και εξελισσόμενους τομείς της τεχνητής νοημοσύνης. Το NLP επικεντρώνεται στη δημιουργία συστημάτων που επιτρέπουν στις μηχανές να κατανοούν, να ερμηνεύουν, να αναπαράγουν και να αλληλοεπιδρούν με την ανθρώπινη γλώσσα σε φυσική μορφή. Αυτή η ικανότητα επιτρέπει στους υπολογιστές να αλληλοεπιδρούν με τους χρήστες με φυσικό τρόπο, να αναλύουν τεράστιες ποσότητες γλωσσικών δεδομένων και να εκτελούν πολύπλοκες γλωσσικές εργασίες, όπως η μετάφραση, η περίληψη και η αυτόματη απάντηση ερωτημάτων.

Η εξέλιξη του NLP ξεκίνησε από αρχικές προσπάθειες στη δεκαετία του 1950 με τα πρώτα πειράματα στη μηχανική μετάφραση. Με την πάροδο του χρόνου, η τεχνολογία είχε προχωρήσει από απλές λεξικογραφικές μεθόδους σε πιο περίπλοκες τεχνικές βασισμένες στην στατιστική και την μηχανική μάθηση. Σημαντικά ορόσημα στην εξέλιξη του NLP περιλαμβάνουν την εμφάνιση των αλγορίθμων μηχανικής μάθησης, όπως τα νευρωνικά δίκτυα, καθώς και την ανάπτυξη προηγμένων τεχνικών όπως η επεξεργασία βαθιάς μάθησης και οι μετασχηματισμοί του διανυσματικού χώρου.

Στον σύγχρονο κόσμο, το NLP είναι απαραίτητο για την ανάπτυξη ευφυών εφαρμογών, όπως οι ψηφιακοί βοηθοί, η αυτόματη μετάφραση και η ανίχνευση

συναισθημάτων. Η εξέλιξη αυτής της τεχνολογίας έχει μεταμορφώσει τον τρόπο που αλληλοεπιδρούμε με τις μηχανές και έχει διευρύνει τα όρια του τι είναι δυνατόν στον τομέα της τεχνητής νοημοσύνης[16].

Η επεξεργασία προτάσεων αποτελεί έναν ουσιώδη παράγοντα στην εφαρμογή των τεχνικών Επεξεργασίας Φυσικής Γλώσσας (NLP)[17]. Αυτή η διαδικασία ενσωματώνει μια σειρά από περίπλοκες και πολυεπίπεδες μεθόδους, κάθε μία εξυπηρετώντας έναν ξεχωριστό αλλά συμπληρωματικό σκοπό, όπου μπορούμε να πούμε πως έχουμε τις εξής τεχνικές :

- **Τμηματοποίηση Κειμένου (Tokenization)**: Το κείμενο διαχωρίζεται σε μικρότερες μονάδες, όπως λέξεις ή φράσεις. Αυτή η διαδικασία βοηθά στην ανάλυση και κατανόηση της δομής του κειμένου.
- **Μορφολογική Ανάλυση (Morphological Analysis)**: Αναλύεται η μορφολογία των λέξεων, όπως οι κλίσεις και οι ενεστωτικές μορφές, για να κατανοηθεί η γραμματική τους λειτουργία στην πρόταση.
- **Συντακτική Ανάλυση (Syntactic Analysis)**: Αναλύεται η συντακτική δομή των προτάσεων. Χρησιμοποιούνται διάφοροι αλγόριθμοι για να καθοριστεί πώς τα διαφορετικά τμήματα της πρότασης συνδέονται μεταξύ τους.
- **Σημασιολογική Ανάλυση (Semantic Analysis)**: Εδώ γίνεται η ερμηνεία της σημασίας των λέξεων και των φράσεων μέσα στο πλαίσιο της πρότασης. Αυτό συμβάλλει στην κατανόηση του τι πραγματικά εννοεί ο ομιλητής ή ο συγγραφέας.
- **Διαχείριση Πραγματολογίας (Pragmatic Analysis)**: Στο τελευταίο αυτό στάδιο, η σημασία της πρότασης εξετάζεται μέσα στο συνολικό πλαίσιο χρήσης, λαμβάνοντας υπόψη το πολιτισμικό και κοινωνικό πλαίσιο, καθώς και την πρόθεση του ομιλητή ή του συγγραφέα.
- **Αναγνώριση Οντοτήτων (Named Entity Recognition - NER)**: Η NER αφορά την ανίχνευση και ταξινόμηση ονομαστικών οντοτήτων (όπως ονόματα ατόμων, οργανισμών, τοποθεσιών) μέσα σε ένα κείμενο.
- **Ανάλυση Συναισθήματος (Sentiment Analysis)**: Αυτή η τεχνική αναλύει το κείμενο για να καθορίσει το συναίσθημα που εκφράζεται (θετικό, αρνητικό, ουδέτερο).
- **Ανάλυση Συνάφειας και Συνοχής (Coherence and Cohesion Analysis)**: Εξετάζει πώς συνδέονται οι προτάσεις και οι παράγραφοι μεταξύ τους για να σχηματίζουν ένα συνεκτικό και συνεπές κείμενο.
- **Ανάλυση Εξαρτήσεων (Dependency Analysis)**: Αφορά την ανάλυση των σχέσεων μεταξύ των λέξεων σε μια πρόταση, για να κατανοηθεί πώς οι λέξεις αλληλοεπηρεάζονται και συνδέονται μεταξύ τους.
- **Σύνθεση Κειμένου (Text Generation)**: Περιλαμβάνει τη δημιουργία νέου κειμένου, όπως η αυτόματη παραγωγή περιλήψεων ή της σύνθεσης φυσικού κειμένου από δεδομένα.
- **Εξαγωγή Πληροφοριών (Information Extraction)**: Η διαδικασία της αναγνώρισης και της εξαγωγής συγκεκριμένων πληροφοριών από κείμενα, όπως ημερομηνίες, τοποθεσίες, και σχέσεις μεταξύ οντοτήτων.
- **Αυτόματη Απάντηση Ερωτημάτων (Question Answering)**: Αφορά τη δημιουργία συστημάτων που μπορούν να απαντήσουν ερωτήσεις βασισμένες σε κείμενα ή άλλες πηγές πληροφοριών.

Συνοψίζοντας, η επεξεργασία προτάσεων στην τεχνολογία επεξεργασίας φυσικής γλώσσας (NLP) περιλαμβάνει πολλές τεχνικές, από τη διάσπαση του κειμένου σε μικρότερα τμήματα μέχρι τη δημιουργία νέου κειμένου. Αυτές οι τεχνικές συνεργάζονται για την ανάλυση, κατανόηση και επεξεργασία του κειμένου, προσφέροντας σημαντική λειτουργικότητα στις εφαρμογές NLP.

3.4.2 Μετατροπή Ιδανικής Πολιτικής Σε Κανόνες Iptables

Σε αυτήν την ενότητα, θα εξετάσουμε πιο λεπτομερώς τον τρόπο με τον οποίο η τεχνολογία της Επεξεργασίας Φυσικής Γλώσσας (NLP) μπορεί να χρησιμοποιηθεί για να μετατραπεί η Ιδανική Πολιτική ασφαλείας. που μας έχει δοθεί σε φυσική γλώσσα. σε κανόνες Iptables.

Η ιδέα είναι, αφού μας έχει δοθεί από την εταιρία η ιδανική πολιτική ασφαλείας σε txt αρχείο και σε φυσική γλώσσα, να αξιοποιήσουμε τις δυνατότητες της τεχνητής νοημοσύνης και της NLP για να αναγνωρίσουμε τα στοιχεία ασφάλειας και να τα μετατρέψουμε σε κανόνες που κατανοούν τα Iptables. Αυτή η προσέγγιση μπορεί να βοηθήσει στην αυτοματοποίηση της διαδικασίας ρύθμισης της δικτυακής ασφάλειας, εξοικονομώντας χρόνο και μειώνοντας τον κίνδυνο ανθρώπινων λαθών.

Για να μπορέσουμε όμως να φτάσουμε στην υλοποίηση της μετατροπής υπάρχουν και οι ανάλογες προϋποθέσεις συντακτικής διαμόρφωσης από την εταιρία οι οποίες είναι οι εξής :

- Η εταιρία θα πρέπει να μας δίνει την ιδανική πολιτική σε μορφή κατανοητή και στην γλώσσα των αγγλικών. Ο κώδικας δεν αναγνωρίζει άλλες γλώσσες πέρα των αγγλικών.
- Δεν πρέπει να υπάρχουν λάθη στο συντακτικό αλλά και λάθη σε λέξεις. Ο κώδικας αυτήν την στιγμή δεν έχει την δυνατότητα να αναγνωρίσει λάθος λέξη ή κάποιου είδους autocorrect και να δίνει τις σωστές λέξεις.
- Οι προτάσεις δεν πρέπει να έχουν παραλείψεις ή να εννοούνται πράγματα. Όλα θα πρέπει να δηλώνονται.
- Οι εταιρία θα πρέπει να δίνει πολιτική ασφαλείας σε μορφή προτάσεων και όχι κειμένου. Η κάθε πρόταση θα συμβολίζει και μια εντολή ασφαλείας.
- Η κάθε εντολή θα είναι της προστακτικής μορφής όπου θα εξηγεί το τι θέλει να συμβεί σε κινήσεις πακέτων σε συγκεκριμένες θύρες ή όχι.
- Οι λέξεις μπορεί να είναι είτε στα κεφαλαία είτε στα πεζά.

Ένα απλό παράδειγμα μιας πρότασης είναι η παρακάτω εντολή.

«Allow incoming traffic on TCP port 443 (HTTPS) from any source IP address.»

Σε αυτήν την πρόταση βλέπουμε ότι διευκρινίζεται το τι ακριβώς θέλουμε να συμβεί στην συγκεκριμένη θύρα και με λεπτομέρειες.

Χρησιμοποιώντας τώρα την τεχνολογία NLP, αυτό που κάνουμε ουσιαστικά είναι να δημιουργήσουμε tokens σε κάθε ξεχωριστή πρόταση και στην συνέχεια να συγκρίνονται με λέξεις κλειδιά που έχουμε δώσει με τελικό αποτέλεσμα την δημιουργία εντολών σε Iptables.

Τέλος, δεν υπάρχει καμία ειδική απαίτηση από την εταιρία όσον αφορά τη χρήση συγκεκριμένων λέξεων στις προτάσεις. Εκμεταλλευόμαστε την τεχνολογία των όμοιων λέξεων που έχουμε ενσωματώσει στις λέξεις κλειδιά, προκειμένου να διασφαλίσουμε ένα ευρύ φάσμα λεξιλογίου χωρίς περιορισμούς.

3.5 Μετατροπή Κανόνων Iptables Σε Κατάλληλη Μορφή Για Σύγκριση Και Επαναφορά

Σε αυτήν την ενότητα εξετάζουμε τη μεθοδολογία μετατροπής των κανόνων Iptables σε μορφή συμβατή με τις εντολές iptables-restore και iptables-save σε Linux συστήματα[18]. Η κύρια πρόκληση αυτής της μετατροπής είναι η δυνατότητα σύγκρισης μεταξύ διαφορετικών πολιτικών ασφαλείας σε Linux μηχανήματα ενώ η δευτερεύουσα είναι η επαναφορά αυτών των κανόνων σε αυτά τα μηχανήματα. Δεδομένου ότι οι κανόνες σε φυσική γλώσσα εκφράστηκαν σε κανόνες Iptables, η μετατροπή τους σε μια κοινή και συμβατή μορφή, όπως μας δίνονται μέσω της εντολής iptables-save, είναι κρίσιμη για την αποτελεσματική σύγκριση και ανάλυση των διαφορών μεταξύ των πολιτικών ασφαλείας.

Η μετατροπή των κανόνων σε αυτήν τη συγκεκριμένη μορφή έχει δύο σημαντικούς λόγους.

- **Ευκολία Σύγκρισης:**
    - Εξασφαλίζεται ότι οι κανόνες έχουν ίδια μορφή με άλλους κανόνες από Linux συστήματα που έχουν αποθηκευτεί με την εντολή iptables-save.
    - Η ομοιότητα της μορφής διευκολύνει την αντιστοίχιση των κανόνων και βοηθά στην κατανόηση των διαφορών μεταξύ τους.

- **Ευκολία Επαναφοράς:**
    - Επιτρέπεται η δημιουργία αρχείων που μπορούν να επαναφερθούν εύκολα σε ένα Linux σύστημα με τη χρήση της εντολής iptables-restore. Αυτό απλοποιεί τη διαδικασία αντιγραφής ή επαναφοράς κανόνων ασφαλείας από ένα σύστημα σε ένα άλλο, εξασφαλίζοντας ότι οι κανόνες θα επαναφερθούν ακριβώς όπως έχουν οριστεί, χωρίς περαιτέρω προσαρμογές.

Η μορφή αυτή πρέπει να πληροί ορισμένες προϋποθέσεις για να είναι αποδεκτή από τα Linux συστήματα.

Κάθε κανόνας πρέπει να είναι σε μία γραμμή, να περιέχει τις απαραίτητες παραμέτρους και να είναι τοποθετημένος μεταξύ των αντίστοιχων αλυσίδων (chains). Οι αλυσίδες και οι προκαθορισμένοι κανόνες πρέπει επίσης να συμπεριληφθούν στο αρχείο με τη σωστή σύνταξη. Εφόσον εξασφαλίσουμε πως οι κανόνες ακολουθούν αυτές τις οδηγίες, θα είμαστε σε θέση να συγκρίνουμε την συγκεκριμένη πολιτική με οποιαδήποτε άλλη πολιτική που έχουμε πάρει από άλλο linux μηχάνημα και να έχουμε την δυνατότητα να επαναφέρουμε τις ρυθμίσεις τους χρησιμοποιώντας την εντολή iptables-restore.

Πιο αναλυτικά, το αρχείο με τους κανόνες πρέπει να έχει την εξής μορφή:

- **Αλυσίδες (Chains):** Οι αλυσίδες πρέπει να ορίζονται πρώτες στο αρχείο. Κάθε αλυσίδα ξεκινά με την εντολή * ακολουθούμενη από το όνομα της αλυσίδας όπως για παράδειγμα *filter, *mangle, *raw, *nat
- **Αρχική πολιτική:** Ακολουθεί προσαρτημένος πίνακας με προκαθορισμένη πολιτική αποδοχής ή απόρριψης πακέτων. (π.χ., :INPUT ACCEPT [0:0]).
- **Επιτρεπτικοί/Απορριπτικοί Κανόνες:** Ακολουθούν κανόνες όπου ο καθένας πρέπει να είναι σε μία γραμμή και να περιέχει όλες τις απαραίτητες πληροφορίες για τη σύνδεση, το πρωτόκολλο, τις πηγές και τους προορισμούς.
- **Εντολή COMMIT:** Η λέξη "COMMIT" πρέπει να είναι σε μία ξεχωριστή γραμμή και δηλώνει το τέλος των κανόνων για τη συγκεκριμένη αλυσίδα.

Άλλα επιπρόσθετα πλεονεκτήματα που θα έχουμε μέσω της μετατροπής των κανόνων είναι :

- **Κατανοητή Δομή:** Οι νέοι κανόνες είναι οργανωμένοι σε διάφορους πίνακες (filter, nat, mangle, raw) και αλυσίδες (INPUT, FORWARD, OUTPUT κ.λπ.). Αυτή η δομή καθιστά κατανοητό και οργανωμένο προγραμματισμό των κανόνων.
- **Έλεγχος Σφαλμάτων:** Ο κώδικας περιλαμβάνει αντιμετώπιση σφαλμάτων και εξαιρέσεις για πιθανά λάθη στη μορφή των κανόνων Iptables που διαβάζονται. Αυτό βοηθά στον εντοπισμό και στην αντιμετώπιση προβλημάτων με τη μετατροπή.

Συνολικά, η μετατροπή των κανόνων σε κοινή μορφή διευκολύνει την επακριβή ανάλυση και σύγκριση των κανόνων ασφαλείας μεταξύ διαφορετικών πηγών, επιτρέποντας την αποτελεσματική διαχείριση και σύγκριση των πολιτικών ασφαλείας.

4 Μεθοδολογία Χρήσης Τεχνολογίας NLP Για Δημιουργία Embeddings

Η μεθοδολογία που εφαρμόζουμε για την ανάλυση πολιτικών ασφαλείας σε Linux συστήματα ενσωματώνει προηγμένες τεχνικές της Επεξεργασίας Φυσικής Γλώσσας (NLP) για μια ενδελεχή κατανόηση των εντολών Iptables και των αντίστοιχων πολιτικών ασφαλείας. Πριν τη σύγκριση των πολιτικών ασφαλείας μεταξύ της ιδανικής και της εν χρήση πολιτικής θα πρέπει πρώτα να κάνουμε μια ενδελεχή ανάλυση και επεξεργασία αυτών έτσι ώστε να φέρουμε τους κανόνες σε κατάλληλη μορφή για σύγκριση μέσω τεχνολογίας NLP.

4.1 Εισαγωγή Στις Αρχιτεκτονικές Των NLP Και Τα Embeddings

Η εξέλιξη του NLP έχει οδηγήσει στην ανάπτυξη ποικίλων αρχιτεκτονικών που διαφέρουν ως προς την πολυπλοκότητα και την εφαρμογή τους. Αυτές οι αρχιτεκτονικές ενσωματώνουν τόσο τις πρώιμες προσεγγίσεις όσο και τις πιο σύγχρονες τεχνικές. Παρακάτω βλέπουμε την εξέλιξη των αρχιτεκτονικών :

Αρχικές Αρχιτεκτονικές : Στις αρχές της χρησιμοποίησης του NLP, οι αρχιτεκτονικές βασίζονταν κυρίως σε κανονιστικές μεθόδους και απλές στατιστικές προσεγγίσεις. Αυτές περιλάμβαναν μοντέλα όπως τα απλά νευρωνικά δίκτυα, τα οποία χρησιμοποιούνταν για κατανόηση της γλώσσας και την επεξεργασία βασικών ερωτημάτων όπως τα :

- **Feedforward Neural Networks (FNNs)**: Πρόκειται για μια από τις πιο βασικές μορφές νευρωνικών δικτύων. Σε αυτά, η πληροφορία κινείται μόνο προς τα εμπρός, από την είσοδο προς την έξοδο, χωρίς κάποια αναδρομική ή επαναλαμβανόμενη διαδικασία[19].
- **Βασικά Στατιστικά Μοντέλα**: Συχνά χρησιμοποιούνταν στατιστικές μέθοδοι για την επεξεργασία γλωσσικών δεδομένων. Αυτές περιλαμβάνουν μοντέλα όπως n-gram, τα οποία βασίζονται στη συχνότητα εμφάνισης γλωσσικών μονάδων (όπως λέξεις ή φράσεις) σε ένα δεδομένο σύνολο κειμένου[20].
- **Απλά Νευρωνικά Δίκτυα**: Πριν την εμφάνιση των πιο πολύπλοκων νευρωνικών δικτύων, τα απλά νευρωνικά δίκτυα όπως τα perceptrons χρησιμοποιούνταν για βασικές εργασίες ταξινόμησης και ανάλυσης γλωσσικών δεδομένων.

Εξέλιξη σε πιο προηγμένες Αρχιτεκτονικές : Με την πρόοδο της τεχνολογίας και την εμφάνιση της μηχανικής μάθησης, οι αρχιτεκτονικές του NLP έγιναν πιο πολύπλοκες και αποδοτικές. Ειδικά με την εισαγωγή των νευρωνικών δικτύων, ανοίχτηκαν νέοι δρόμοι για την ανάπτυξη μοντέλων τα οποία έχουν την ικανότητα να διαχειρίζονται πολύπλοκες ακολουθίες δεδομένων και να κατανοούν το πλαίσιο μεγάλων τμημάτων κειμένου όπως τα :

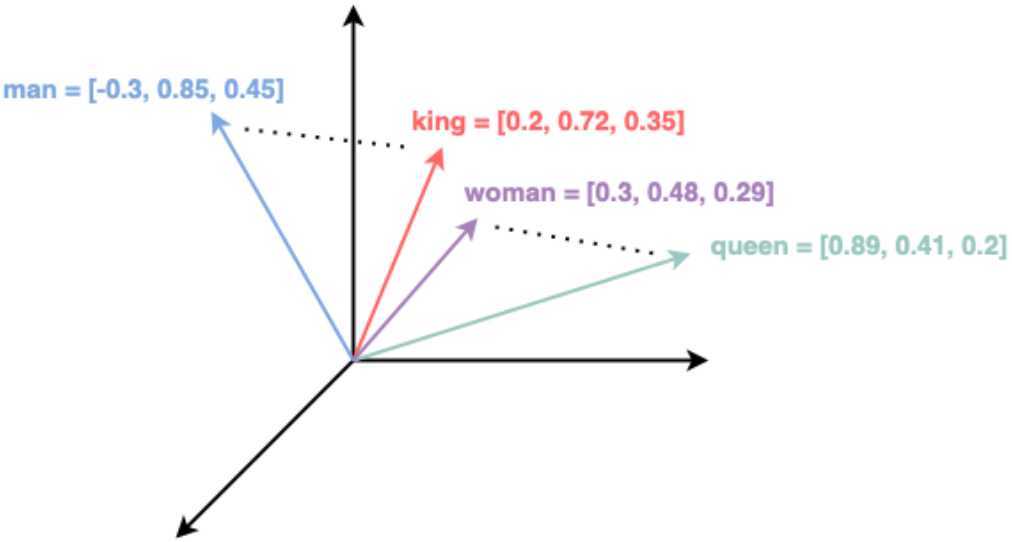
- **RNN (Recurrent Neural Network)**: Τα RNN είναι κατάλληλα για εργασίες NLP λόγω της ικανότητάς τους να διαχειρίζονται ακολουθίες δεδομένων, όπως το κείμενο, λόγω της δυνατότητάς τους να "θυμούνται" προηγούμενες εισόδους μέσω των εσωτερικών καταστάσεων [21].
- **LSTM (Long Short-Term Memory)**: Μια εξελιγμένη μορφή των RNN, τα LSTM είναι σχεδιασμένα για να αποφεύγουν το πρόβλημα της εξαφάνισης της διαφοράς (vanishing gradient) που συνήθως συναντάται στα απλά RNN [22].

- **GRU (Gated Recurrent Unit)**: Παρόμοια με τα LSTM, τα GRU είναι μια πιο απλοποιημένη εκδοχή τους, προσφέροντας παρόμοια αποδοτικότητα με λιγότερους υπολογισμούς [23].
- **CNN (Convolutional Neural Network)**: Αν και συνηθέστερα χρησιμοποιούνται στην επεξεργασία εικόνων, τα CNN έχουν βρει εφαρμογή και στο NLP για την επεξεργασία και ανάλυση τοπικών χαρακτηριστικών του κειμένου [24].
- **Capsule Networks**: Ενώ είναι ακόμη σε αρχικό στάδιο ανάπτυξης για NLP, οι κάψουλες (capsules) προσπαθούν να αντιμετωπίσουν κάποιες από τις αδυναμίες των CNN και RNN, προσφέροντας βελτιωμένη αναγνώριση μοτίβων και σχέσεων σε δεδομένα [25].
- **Autoencoders**: Αυτά τα μοντέλα χρησιμοποιούνται για την αναπαράσταση και τη μείωση των διαστάσεων των δεδομένων. Στο NLP, μπορούν να εκπαιδευτούν για να συμπυκνώσουν το κείμενο σε πιο συμπαγείς μορφές, διατηρώντας παράλληλα τις σημαντικές πληροφορίες [26].
- **Attention Mechanisms**: Αν και αυτή η τεχνολογία έχει γίνει πιο γνωστή μέσω των Transformer Networks, οι μηχανισμοί προσοχής ξεκίνησαν να ενσωματώνονται σε πιο προηγμένες αρχιτεκτονικές πριν από την εμφάνιση των Transformers. Επιτρέπουν στα μοντέλα να 'εστιάζουν' σε συγκεκριμένα τμήματα της εισόδου για βελτιωμένη επεξεργασία [27].
- **Sequence to Sequence Models (Seq2Seq)**: Αυτά τα μοντέλα, συχνά βασισμένα σε RNN ή LSTM, έχουν εφαρμογές στη μηχανική μετάφραση, όπου η είσοδος (μια ακολουθία σε μία γλώσσα) μετατρέπεται σε μια αντίστοιχη έξοδο (μια ακολουθία σε άλλη γλώσσα) [28].

Εποχή των προηγμένων αρχιτεκτονικών: Η πιο πρόσφατη και επαναστατική εξέλιξη στις αρχιτεκτονικές του NLP είναι η εισαγωγή των Transformer μοντέλων.

- **Transformers:** Αυτά τα μοντέλα, έχουν αλλάξει θεμελιωδώς τον τρόπο με τον οποίο τα συστήματα NLP επεξεργάζονται τη γλώσσα. Χρησιμοποιούν μεγάλες ποσότητες δεδομένων και βαθιά νευρωνικά δίκτυα για να κατανοήσουν σε βάθος τις σχέσεις και τις νοηματικές δομές μέσα στη φυσική γλώσσα. Αυτή η αρχιτεκτονική έχει καθιερωθεί ως μια από τις πιο επιτυχημένες στο NLP, χάρη στους μηχανισμούς αυτο-προσοχής (self-attention) και διασταυρωμένης προσοχής (cross-attention) που επιτρέπουν την παράλληλη επεξεργασία των δεδομένων και την πιο βαθιά κατανόηση των ακολουθιών [29].

Η πρόοδος στις αρχιτεκτονικές του NLP έχει συνδεθεί άρρηκτα με την εξέλιξη και την εφαρμογή των embeddings[30], τα οποία αποτελούν τον πυρήνα της σύγχρονης επεξεργασίας φυσικής γλώσσας. Τα embeddings, ως διανυσματικές αναπαραστάσεις των λέξεων ή των φράσεων όπως βλέπουμε και σε ένα απλό παράδειγμα στην εικόνα 1, επιτρέπουν στα μοντέλα NLP να ερμηνεύουν και να επεξεργάζονται την ανθρώπινη γλώσσα με τρόπο που αναγνωρίζει τη σημασία και τις σχέσεις των λέξεων.



Εικόνα 1 Αναπαράσταση λέξεων σε διανύσματα (embeddings)

Από τις πρώιμες αρχιτεκτονικές, όπως τα Feedforward και Recurrent Neural Networks, τα embeddings χρησιμοποιούνταν ως μέσα για να μετατρέψουν το κείμενο σε αριθμητικές τιμές που τα μοντέλα μπορούσαν να επεξεργαστούν. Αυτό ήταν βασικό για την εκτέλεση διαφόρων εργασιών, όπως η κατηγοριοποίηση κειμένου και η ανάλυση συναισθημάτων.

Με την εμφάνιση πιο προηγμένων αρχιτεκτονικών, όπως τα LSTM, GRU, και στη συνέχεια τα Transformer μοντέλα, τα embeddings εξελίχθηκαν σε πιο πολύπλοκες και δυναμικές μορφές. Τα σύγχρονα μοντέλα δημιούργησαν context-aware embeddings, που μπορούν να κατανοήσουν τη σημασία της λέξης μέσα στο συγκεκριμένο πλαίσιο της. Αυτό άνοιξε νέες διαστάσεις στην επεξεργασία γλωσσικών δεδομένων, επιτρέποντας πιο λεπτομερείς και περίπλοκες ερμηνείες και αναλύσεις.

Συνολικά, η αλληλεπίδραση μεταξύ των αρχιτεκτονικών NLP και των embeddings έχει υπάρξει ένας δυναμικός τομέας εξέλιξης. Με την κάθε νέα αρχιτεκτονική, τα embeddings γίνονται πιο πολύπλοκα και ικανά, δίνοντας στα συστήματα NLP μεγαλύτερη ικανότητα κατανόησης και επεξεργασίας της φυσικής γλώσσας, προσφέροντας στους χρήστες ακόμα πιο ακριβείς και πλούσιες γλωσσικές αναλύσεις.

Στην επεξεργασία της φυσικής γλώσσας και την τεχνητή νοημοσύνη, χρησιμοποιούνται διάφορα είδη embeddings για να αναπαραστήσουν λέξεις, προτάσεις, ή ακόμα και ολόκληρα κείμενα σε μορφή διανυσμάτων. Παρακάτω παραθέτουμε μερικά από τα πιο συνηθισμένα είδη:

- **Word Embeddings**: Αντιπροσωπεύουν λέξεις ή φράσεις από το λεξιλόγιο σε μορφή διανυσμάτων. Παραδείγματα περιλαμβάνουν τα Word2Vec, GloVe και FastText [31].
- **Sentence/Paragraph Embeddings**: Αντιπροσωπεύουν ολόκληρες προτάσεις ή παραγράφους. Τεχνικές όπως η Doc2Vec ή οι πρόσφατες προσεγγίσεις με transformers παράγουν τέτοιου είδους embeddings[32].
- **Character Embeddings**: Αυτά τα embeddings αντιπροσωπεύουν μεμονωμένους χαρακτήρες, χρησιμοποιώντας πληροφορίες σε επίπεδο χαρακτήρα για να κατανοήσουν καλύτερα την γλώσσα, ιδιαίτερα χρήσιμο σε γλώσσες με περίπλοκη σύνταξη ή ορθογραφία[33].
- **Positional Embeddings**: Χρησιμοποιούνται στα μοντέλα transformer για να δώσουν πληροφορίες σχετικά με τη θέση των λέξεων ή χαρακτήρων σε μια πρόταση[34].

- **Graph Embeddings**: Αυτά τα embeddings αντιπροσωπεύουν τους κόμβους, τις ακμές ή ολόκληρα γραφήματα σε διανυσματική μορφή, χρήσιμα για ανάλυση δικτύων και γραφημάτων[35].

Επιπρόσθετα τα embeddings που παράγονται από τεχνικές NLP αποδεικνύονται ιδιαίτερα χρήσιμα σε συγκρίσεις με αλγορίθμους ομοιότητας. Καθώς τα embeddings αντιπροσωπεύουν τα σημασιολογικά χαρακτηριστικά των λέξεων ή φράσεων, είναι δυνατόν να μετρήσουμε την ομοιότητα μεταξύ διαφορετικών κειμένων ή μεταξύ λέξεων.

Οι αλγόριθμοι ομοιότητας μπορούν να χρησιμοποιηθούν για τον υπολογισμό της απόστασης μεταξύ των embeddings-διανυσμάτων, προσφέροντας έτσι έναν τρόπο να μετρηθεί η σημασιολογική ομοιότητα μεταξύ διαφορετικών κειμένων. Αυτό είναι εξαιρετικά χρήσιμο σε πολλές εφαρμογές, όπως η ανάλυση συναισθημάτων, η αυτόματη κατηγοριοποίηση κειμένου, η αναζήτηση πληροφοριών και η σύγκριση κειμένων. Η ικανότητα αυτή να παρέχουν σημαντικές συγκρίσεις μεταξύ δομών γλώσσας αποτελεί σημαντικό εργαλείο στον τομέα της επεξεργασίας φυσικής γλώσσας.

Ο λόγος που δημιουργούμε embeddings στην επεξεργασία φυσικής γλώσσας και πιο συγκεκριμένα σε Iptables είναι να αναπαραστήσουμε τους κανόνες των Iptables σε έναν μαθηματικό χώρο όπου μπορούμε να μετρήσουμε την ομοιότητα μεταξύ τους. Η δημιουργία αυτών των αναπαραστάσεων επιτρέπει την αποτελεσματική σύγκριση και ανάλυση τους, προσφέροντας έναν πολύτιμο τρόπο να αντιληφθούμε τη σημασιολογική ομοιότητα μεταξύ διαφορετικών κανόνων και εν συνεχεία πολιτικών ασφαλείας. Αυτό διευκολύνει την σύγκριση των κανόνων, καθιστώντας τα embeddings ένα κρίσιμο εργαλείο στον χώρο των iptables και της τεχνητής νοημοσύνης.

Συνεπώς, η δημιουργία και αξιοποίηση των embeddings κατά την επεξεργασία της φυσικής γλώσσας και πιο συγκεκριμένα των κανόνων Iptables είναι κρίσιμη για την αποτελεσματική υλοποίηση συστημάτων NLP και την σύγκριση και εύρεση ομοιότητας μεταξύ των κανόνων.

4.2 Επιλογή Προεκπαιδευμένων Μοντέλων Με Βάση Την Αρχιτεκτονική Transformers

Στον σύγχρονο κόσμο της επεξεργασίας της φυσικής γλώσσας (NLP), η αρχιτεκτονική Transformers έχει αναδειχθεί ως μία από τις πλέον επιδραστικές και ισχυρές προσεγγίσεις. Αυτή η αρχιτεκτονική, που επικεντρώνεται στην εκμετάλλευση του μηχανισμού προσοχής για την αναγνώριση σχέσεων μεταξύ λέξεων σε μεγάλες ακολουθίες κειμένου, έχει οδηγήσει στη δημιουργία πολυάριθμων καινοτόμων μοντέλων που έχουν επαναστατήσει στην επεξεργασία και την κατανόηση φυσικού κειμένου. Βάσει αυτής της προηγμένης αρχιτεκτονικής, έχουμε επιλέξει να χρησιμοποιήσουμε μια σειρά από προεκπαιδευμένα μοντέλα για την επίτευξη υψηλών επιδόσεων πάνω στην εφαρμογή της σύγκρισης κανόνων πολιτικών ασφαλείας. Συγκεκριμένα, η επιλογή μας περιλαμβάνει τα εξής μοντέλα :

- Universal Sentence Encoder (USE)
- BERT (Bidirectional Encoder Representations from Transformers)
- RoBERTa (Robustly optimized BERT approach)
- MPNet (Masked and Permuted Pre-training for Language Understanding)
- MiniLM
- GPT-2 (Generative Pre-trained Transformer 2)
- T5 (Text-to-Text Transfer Transformer)

Η χρήση αυτών των προηγμένων μοντέλων επεξεργασίας φυσικής γλώσσας (NLP) πάνω στην σύγκρισή που πρόκειται να κάνουμε, είναι κατάλληλη για πολλούς σημαντικούς λόγους που είναι οι εξής:

- **Πολυπλοκότητα και Πλούτος της Γλώσσας:** Η φυσική γλώσσα είναι πολύπλοκη και πολυδιάστατη, με τις λέξεις να έχουν πολλαπλές σημασίες ανάλογα με το πλαίσιο τους. Αυτά τα μοντέλα, χάρη στις μεθόδους Deep Learning και την αρχιτεκτονική των Transformers που χρησιμοποιούν, είναι ικανά να κατανοήσουν και να επεξεργαστούν το πλαίσιο και τη σημασιολογία του κειμένου σε βάθος.
- **Ευελιξία και Εφαρμοστικότητα:** Η δυνατότητα των μοντέλων να προσαρμόζονται σε διάφορα καθήκοντα NLP με ελάχιστες ή χωρίς καθόλου τροποποιήσεις τους καθιστά ιδιαίτερα πολύτιμα. Από την ανάλυση συναισθήματος και την αναγνώριση οντοτήτων, μέχρι την αυτόματη παραγωγή κειμένου και την απάντηση ερωτήσεων, η ευελιξία τους επιτρέπει να εξυπηρετούν ένα ευρύ φάσμα εφαρμογών.
- **Προηγμένη Κατανόηση Κειμένου:** Μοντέλα όπως το USE και το BERT έχουν δείξει την ικανότητά τους να κατανοούν τη σημασιολογική ομοιότητα μεταξύ προτάσεων ή παραγράφων, προσφέροντας αυξημένη ακρίβεια στην επεξεργασία και στην ανάλυση κειμένου.
- **Αποδοτική Παραγωγή Κειμένου:** Προηγμένα μοντέλα Τεχνητής Νοημοσύνης όπως το GPT-2 και το T5 έχουν μεταμορφώσει τον τομέα της παραγωγής κειμένου, προσφέροντας την ικανότητα να παράγουν κείμενο που είναι συνεκτικό, πειστικό και εκπληκτικά παρόμοιο με αυτό ενός ανθρώπου. Αυτή η εξέλιξη έχει αυξήσει σημαντικά τις προοπτικές στην αυτόματη δημιουργία περιεχομένου, ανοίγοντας νέους δρόμους για την εφαρμογή της τεχνολογίας στην παραγωγή ποιοτικού και πειστικού κειμένου.
- **Μεταφορά Γνώσης:** Η λεπτή ρύθμιση (fine-tuning) των προεκπαιδευμένων μοντέλων για συγκεκριμένες εφαρμογές επιτρέπει την μεταφορά γνώσης από την πλατιά προεκπαίδευση σε ειδικά καθήκοντα, βελτιώνοντας την απόδοση και μειώνοντας τον χρόνο και τους πόρους που απαιτούνται για την εκπαίδευση από την αρχή.
- **Κατανόηση Περίπλοκων Δομών:** Μοντέλα όπως το MPNet και το MiniLM δείχνουν προηγμένη κατανόηση συντακτικών και σημασιολογικών δομών στη γλώσσα, προσφέροντας βελτιωμένη επεξεργασία και ανάλυση κειμένου.

Συνολικά, η χρήση αυτών των μοντέλων προσφέρει μια αποδοτική, ευέλικτη, και ισχυρή προσέγγιση στην επίλυση διαφόρων προκλήσεων στον τομέα του NLP, από την αυτόματη κατανόηση και επεξεργασία της φυσικής γλώσσας μέχρι την παραγωγή πλούσιου και συνεκτικού κειμένου.

4.3 Ανάλυση Τεχνικών Πτυχών Των Μοντέλων

Σε αυτήν την ενότητα θα εξετάσουμε ενδελεχώς τις τεχνικές πτυχές των μοντέλων NLP, με έμφαση στην αρχιτεκτονική, την υλοποίηση και τις λειτουργικές δυνατότητες τους. Αυτή η ανάλυση έχει σκοπό να παρέχει μια σαφή και συνολική κατανόηση των βασικών στοιχείων που συνθέτουν τα μοντέλα, καθώς και των τεχνολογικών προκλήσεων και των λύσεων που αντιμετωπίζουν κατά την ανάπτυξή τους.

4.3.1 Universal Sentence Encoder (USE)

Το Universal Sentence Encoder (USE) είναι ένα μοντέλο για την επεξεργασία φυσικής γλώσσας, αναπτυγμένο από τη Google το οποίο αποσκοπεί στη δημιουργία υψηλής ποιότητας sentence embeddings. Η κύρια διαφορά του USE από άλλα μοντέλα, όπως το BERT ή το GPT, είναι ότι είναι ειδικά σχεδιασμένο για να ενσωματώνει ολόκληρες προτάσεις ή ακόμη και παραγράφους σε ένα μόνο διάνυσμα, κρατώντας τη σημασιολογική πληροφορία[36].

Ας εξετάσουμε μερικές από τις βασικές τεχνικές λεπτομέρειες του USE:

- **Διαστάσεις Διανυσμάτων:** Ένας από τους κύριους παράγοντες που καθορίζουν την απόδοση του USE είναι η διάσταση του διανύσματος που παράγει για κάθε πρόταση. Τυπικά, το USE παράγει διανύσματα 512 διαστάσεων για κάθε πρόταση, με περίπου 256 εκατομμύρια παραμέτρους. παρέχοντας ένα πλούσιο σύνολο πληροφοριών σε συμπαγή μορφή.
- **Αρχιτεκτονική Μοντέλου:** Το USE βασίζεται σε δομές βαθιάς μάθησης όπως τα δίκτυα Transformer ή τα δίκτυα CNN (Convolutional Neural Networks). Αυτές οι αρχιτεκτονικές είναι ικανές να κατανοούν τη σημασία και το πλαίσιο των λέξεων σε μια πρόταση.
- **Εκπαίδευση Μοντέλου:** Το USE εκπαιδεύεται σε μεγάλα σύνολα δεδομένων, περιλαμβάνοντας κείμενα από πολλές πηγές και γλώσσες. Η εκπαίδευση γίνεται με τη χρήση τεχνικών όπως το supervised learning και το transfer learning.
- **Αντίκτυπος στην Απόδοση:** Τα διανύσματα που παράγονται από το USE είναι χρήσιμα σε ποικίλες εφαρμογές NLP όπως η σημασιακή αναζήτηση, η κατηγοριοποίηση κειμένων και η αναγνώριση συναισθήματος.
- **Ενσωμάτωση σε Εφαρμογές:** Το USE είναι διαθέσιμο μέσω TensorFlow Hub, επιτρέποντας την εύκολη ενσωμάτωση σε εφαρμογές, με λιγότερη ανάγκη για εξειδικευμένη γνώση στην επεξεργασία γλώσσας.

4.3.2 BERT (Bidirectional Encoder Representations from Transformers)

Το BERT (Bidirectional Encoder Representations from Transformers) είναι ένα επαναστατικό μοντέλο στον τομέα της επεξεργασίας φυσικής γλώσσας (NLP), αναπτυγμένο από τη Google[37].

Το BERT, ως μοντέλο, ενσωματώνει και συνδυάζει τρία βασικά είδη embeddings στην αρχιτεκτονική του για να επιτύχει την ανάλυση και την κατανόηση φυσικής γλώσσας:

- **Word (Token) Embeddings**: Αυτά είναι τα βασικά embeddings κάθε λέξης (ή token) που χρησιμοποιούνται στο μοντέλο. Κάθε λέξη στην πρόταση μετατρέπεται σε ένα διάνυσμα που αντιπροσωπεύει τη σημασία της.
- **Positional Embeddings**: Λόγω της ακολουθιακής φύσης της γλώσσας, το BERT χρησιμοποιεί επίσης positional embeddings για να κωδικοποιήσει τη θέση κάθε λέξης στην πρόταση. Αυτό είναι σημαντικό γιατί η σημασία μιας λέξης μπορεί να αλλάξει ανάλογα με το πού βρίσκεται στην πρόταση.
- **Segment Embeddings**: Αυτά τα embeddings χρησιμοποιούνται από το BERT για να διακρίνει διαφορετικά τμήματα κειμένου, ιδιαίτερα σε καταστάσεις όπου το μοντέλο επεξεργάζεται περισσότερες από μία προτάσεις (όπως στην κατανόηση της σχέσης μεταξύ δύο προτάσεων).

Ας εξετάσουμε μερικές από τις βασικές τεχνικές λεπτομέρειες του BERT:

- **Αρχιτεκτονική Μοντέλου:** Το BERT βασίζεται στην αρχιτεκτονική του Transformer, μια δομή που χρησιμοποιεί μηχανισμούς προσοχής (attention mechanisms) για να κατανοήσει τις σχέσεις μεταξύ των λέξεων σε ένα κείμενο.
- **Διαδικασία Εκπαίδευσης:** Το BERT εκπαιδεύεται με δύο βασικές μεθόδους: την πρόβλεψη λέξεων με Masked (Masked Language Model - MLM) και την πρόβλεψη επόμενης πρότασης (Next Sentence Prediction - NSP). Αυτές οι τεχνικές επιτρέπουν στο BERT να αποκτήσει μια βαθιά κατανόηση του γλωσσικού πλαισίου.
- **Διαστάσεις και Παραλλαγές:** Το BERT έρχεται σε διάφορες εκδόσεις, όπως το BERT-Base (με 12 επίπεδα, 12 αντίγραφα attention, 110 εκατομμύρια παραμέτρους

και 768 διαστάσεις) και το BERT-Large (με 24 επίπεδα, 16 αντίγραφα attention, 340 εκατομμύρια παραμέτρους και 1024 διαστάσεις).
- **Γλωσσική Υποστήριξη:** Παρόλο που αρχικά εκπαιδεύτηκε κυρίως σε αγγλικά δεδομένα, έχουν αναπτυχθεί παραλλαγές του BERT για διάφορες άλλες γλώσσες.
- **Εφαρμογές στην NLP:** Το BERT έχει επιδείξει εξαιρετικές επιδόσεις σε μια ποικιλία εφαρμογών NLP, όπως η κατανόηση κειμένου, η αναγνώριση ονομάτων οντοτήτων, η απάντηση ερωτήσεων, και η μετάφραση.

4.3.3 RoBERTa (Robustly optimized BERT approach)

Το RoBERTa (Robustly optimized BERT approach) είναι μια βελτιωμένη εκδοχή του BERT, αναπτυγμένη από την Facebook AI. Στοχεύει στη βελτίωση της απόδοσης του BERT μέσω διάφορων τεχνικών βελτιστοποιήσεων[38].

Το RoBERTa αποτελεί μια βελτιστοποιημένη έκδοση του αρχικού BERT και χρησιμοποιεί επίσης τους εξής τύπους embeddings:

- **Word (Token) Embeddings**: Όπως και το BERT, το RoBERTa χρησιμοποιεί token embeddings για να αντιπροσωπεύσει τις λέξεις σε μορφή διανυσμάτων. Αυτά τα embeddings κωδικοποιούν τη σημασία των λέξεων.
- **Positional Embeddings**: Το RoBERTa, όπως και το BERT, ενσωματώνει πληροφορίες σχετικά με τη θέση των λέξεων στην πρόταση, ώστε να μπορεί να κατανοεί τη σημασία με βάση τη σειρά εμφάνισης των λέξεων.

Ωστόσο, αντίθετα από το BERT, το RoBERTa δεν χρησιμοποιεί segment embeddings, καθώς έχει βελτιστοποιηθεί μέσω διαφορετικών τεχνικών προεκπαίδευσης, όπως η δυναμική αύξηση του μεγέθους των παρτίδων και η πιο μακροχρόνια εκπαίδευση.

Ας δούμε μερικές από τις κύριες τεχνικές λεπτομέρειες του RoBERTa:

- **Αρχιτεκτονική Μοντέλου:** Ακολουθεί την ίδια βασική αρχιτεκτονική με το BERT, δηλαδή τη δομή των Transformers, παρέχοντας βαθιά διανυσματικές αναπαραστάσεις για τις λέξεις.
- **Διαδικασία Εκπαίδευσης:** Το RoBERTa αλλάζει τη διαδικασία εκπαίδευσης του BERT, απομακρύνοντας την πρόβλεψη της επόμενης πρότασης (Next Sentence Prediction - NSP) και επικεντρώνοντας περισσότερο στην εκπαίδευση με masked μοντέλο γλώσσας (Masked Language Model - MLM).
- **Δεδομένα Εκπαίδευσης:** Χρησιμοποιεί μεγαλύτερα σύνολα δεδομένων και περισσότερα δεδομένα για την εκπαίδευση σε σύγκριση με το BERT, προσφέροντας έτσι πιο πλούσιες γλωσσικές πληροφορίες.
- **Διαστάσεις Διανυσμάτων:** Το RoBERTa, όπως και το BERT, παράγει διανυσματικές αναπαραστάσεις των λέξεων. Για την έκδοση RoBERTa-Base, τα διανύσματα έχουν διαστάσεις 768, ενώ για την έκδοση RoBERTa-Large, οι διαστάσεις αυξάνονται σε 1024.
- **Αριθμός Επιπέδων και Κεφαλών Attention:** Το RoBERTa-Base χρησιμοποιεί 12 επίπεδα (layers) με 12 κεφαλές attention (heads), ενώ το RoBERTa-Large χρησιμοποιεί 24 επίπεδα με 16 κεφαλές attention.
- **Συνολικός Αριθμός Παραμέτρων:** Η έκδοση RoBERTa-Base έχει περίπου 125 εκατομμύρια παραμέτρους, ενώ η έκδοση RoBERTa-Large έχει περίπου 355 εκατομμύρια παραμέτρους.
- **Απόδοση σε Εφαρμογές NLP:** Το RoBERTa έχει επιδείξει βελτιωμένη απόδοση σε πολλές εργασίες NLP σε σύγκριση με το BERT, όπως στην κατανόηση κειμένου, στην αναγνώριση ονομάτων οντοτήτων, και στην απάντηση ερωτήσεων.

4.3.4 MPNet (Masked and Permuted Pre-training for Language Understanding)

Το MPNet (Masked and Permuted Pre-training for Language Understanding) είναι ένα μοντέλο στον τομέα της επεξεργασίας φυσικής γλώσσας (NLP), αναπτυγμένο από τη Microsoft. Σχεδιάστηκε για να αντιμετωπίσει κάποιες προκλήσεις που σχετίζονται με την προεκπαίδευση (pre-training) των μοντέλων γλώσσας όπως το BERT[39].

Σχετικά με τα είδη embeddings που παράγει το MPNet:

- **Word (Token) Embeddings**: Το MPNet, όπως και τα περισσότερα μοντέλα βασισμένα σε transformers, χρησιμοποιεί word embeddings για να κωδικοποιήσει τις λέξεις σε διανύσματα. Αυτό βοηθά το μοντέλο να κατανοεί τη σημασία της κάθε λέξης μέσα στο πλαίσιο της πρότασης.
- **Positional Embeddings**: Το MPNet ενσωματώνει επίσης πληροφορίες σχετικά με τη θέση των λέξεων στην πρόταση, χρησιμοποιώντας positional embeddings. Αυτό επιτρέπει στο μοντέλο να διατηρεί τη σημασιολογική δομή της πρότασης.

Ας εξετάσουμε τις βασικές τεχνικές λεπτομέρειες του MPNet:

- **Αρχιτεκτονική Μοντέλου:** Το MPNet βασίζεται στην αρχιτεκτονική Transformer, όπως και το BERT, αλλά με μια καινοτομία στην εκπαίδευση του μοντέλου. Χρησιμοποιεί μια συνδυασμένη τεχνική mask και μεταθέσεων (masking and permutation) για να προετοιμάσει το μοντέλο να κατανοήσει καλύτερα το πλαίσιο των λέξεων.
- **Διαδικασία Εκπαίδευσης:** Στο MPNet, ορισμένες λέξεις στο κείμενο είναι κρυμμένες (masked) ή μετατοπισμένες (permuted). Αυτό βοηθά το μοντέλο να μάθει πιο αποτελεσματικά τη σημασία και το πλαίσιο των λέξεων σε μια πρόταση.
- **Διαστάσεις και Παραλλαγές:** Το MPNet διατίθεται σε διάφορες εκδόσεις με διαφορετικό αριθμό επιπέδων και διαστάσεων, ανάλογα με τις ανάγκες της συγκεκριμένης εφαρμογής.
- **Εφαρμογές στην NLP:** Το MPNet έχει δείξει εξαιρετικές επιδόσεις σε ποικίλες εργασίες NLP, όπως η κατανόηση κειμένου, η αναγνώριση ονομάτων οντοτήτων, και η απάντηση ερωτήσεων.
- **Βελτιστοποίηση Απόδοσης:** Μέσω της μοναδικής του προσέγγισης στο pre-training, το MPNet προσπαθεί να βελτιστοποιήσει την κατανόηση της σειράς των λέξεων και της σημασίας τους στο πλαίσιο μιας πρότασης ή ενός κειμένου.

4.3.5 MiniLM

Το MiniLM είναι ένα μικρότερο και πιο αποδοτικό μοντέλο γλωσσικής κωδικοποίησης που αναπτύχθηκε από την Microsoft με στόχο την παροχή των πλεονεκτημάτων των μεγαλύτερων μοντέλων Transformer, όπως το BERT και το RoBERTa, αλλά με λιγότερους πόρους και γρηγορότερη απόδοση[40].

Όσον αφορά στα embeddings που παράγει το MiniLM, αυτά περιλαμβάνουν κυρίως:

- **Word (Token) Embeddings**: Αυτά τα embeddings αντιπροσωπεύουν τις επιμέρους λέξεις ή tokens σε μορφή διανυσμάτων. Κάθε λέξη στην είσοδο μετατρέπεται σε ένα διάνυσμα που αποτυπώνει τη σημασία της λέξης μέσα στην πρόταση.
- **Positional Embeddings**: Επειδή τα transformers δεν έχουν ενσωματωμένη αίσθηση της σειράς των λέξεων, το MiniLM χρησιμοποιεί επίσης positional embeddings για να παρέχει πληροφορίες για τη θέση κάθε λέξης στην πρόταση. Αυτό βοηθά το μοντέλο να καταλάβει πώς η σημασία μιας λέξης μπορεί να αλλάζει ανάλογα με τη θέση της.

Ας εξετάσουμε μερικές από τις τεχνικές λεπτομέρειες του MiniLM:

- **Αρχιτεκτονική Μοντέλου:** Το MiniLM βασίζεται στην αρχιτεκτονική Transformer, αλλά έχει σχεδιαστεί για να είναι πολύ μικρότερο και πιο αποδοτικό σε σύγκριση με τα πλήρη μεγέθη των μοντέλων όπως το BERT. Χρησιμοποιεί μια τεχνική που ονομάζεται "knowledge distillation" για να συμπυκνώσει τη γνώση ενός μεγαλύτερου μοντέλου σε ένα μικρότερο.
- **Διαστάσεις Διανυσμάτων και Παραμέτρων:** Το MiniLM έρχεται σε διάφορες εκδόσεις, με την πιο δημοφιλή να είναι η εκδοχή που παράγει διανύσματα 384 ή 768 διαστάσεων. Ο αριθμός των παραμέτρων είναι σημαντικά μικρότερος σε σύγκριση με τα παραδοσιακά μεγάλα μοντέλα, κάτι που καθιστά το MiniLM πιο κατάλληλο για περιβάλλοντα με περιορισμένους υπολογιστικούς πόρους.
- **Εφαρμογές στην NLP:** Αν και το MiniLM είναι μικρότερο σε μέγεθος, διατηρεί εκπληκτική απόδοση σε πολλές εργασίες NLP. Είναι ιδανικό για εφαρμογές όπως η κατηγοριοποίηση κειμένου, η εξαγωγή ονομάτων οντοτήτων, και η απάντηση ερωτήσεων.
- **Βελτιστοποίηση για Απόδοση και Χρήση Πόρων:** Το MiniLM είναι ιδανικό για σενάρια όπου οι υπολογιστικοί πόροι είναι περιορισμένοι, όπως σε κινητές συσκευές ή σε edge computing συστήματα.

4.3.6 GPT-2 (Generative Pre-trained Transformer 2)

Το GPT-2 (Generative Pre-trained Transformer 2) είναι ένα μοντέλο που αναπτύχθηκε από την OpenAI και έχει κεντρικό ρόλο στην επεξεργασία φυσικής γλώσσας (NLP). Έχει ως κύριο στόχο την παραγωγή κειμένου και την κατανόηση γλωσσικών δομών.

Το GPT-2 χρησιμοποιεί κυρίως τους εξής δύο τύπους embeddings:

- **Word (Token) Embeddings**: Το GPT-2 μετατρέπει κάθε λέξη ή token σε ένα διάνυσμα. Αυτά τα embeddings κωδικοποιούν τη σημασία κάθε λέξης ή token στη γλώσσα.
- **Positional Embeddings**: Επειδή το GPT-2 επεξεργάζεται το κείμενο ακολουθιακά και δεν έχει ενσωματωμένη αίσθηση της σειράς των λέξεων, χρησιμοποιεί επίσης positional embeddings για να παρέχει πληροφορίες σχετικά με τη θέση της κάθε λέξης στην πρόταση. Αυτό βοηθά το μοντέλο να κατανοεί πώς η σημασία της λέξης μπορεί να εξαρτάται από τη θέση της μέσα στην πρόταση.

Το GPT-2 είναι ένα μοντέλο που έχει εκπαιδευτεί κυρίως για γεννητικές εργασίες, όπως η δημιουργία κειμένου, και είναι λιγότερο εστιασμένο στην παραγωγή εξειδικευμένων embeddings όπως τα sentence embeddings που παράγονται από μοντέλα όπως το BERT ή το Sentence-BERT. Ωστόσο, τα word και positional embeddings που παράγονται από το GPT-2 είναι πολύτιμα για την κατανόηση της σημασιολογικής και συντακτικής δομής του κειμένου.

Ας δούμε τις βασικές τεχνικές λεπτομέρειες του GPT-2:

- **Αρχιτεκτονική Μοντέλου:** Το GPT-2 χρησιμοποιεί την αρχιτεκτονική Transformer, η οποία είναι ιδιαίτερα αποτελεσματική στην κατανόηση και παραγωγή φυσικής γλώσσας. Βασίζεται σε μηχανισμούς προσοχής (attention mechanisms) για να κατανοήσει τις σχέσεις και το πλαίσιο των λέξεων σε ένα κείμενο.
- **Διαστάσεις και Παραλλαγές:** Το GPT-2 διατίθεται σε διάφορες εκδόσεις με διαφορετικό αριθμό παραμέτρων. Η μικρότερη έκδοση έχει 124 εκατομμύρια παραμέτρους, ενώ η μεγαλύτερη έκδοση, GPT-2 1.5B, έχει περίπου 1.5 δισεκατομμύρια παραμέτρους.

- **Εφαρμογές στην NLP:** Το GPT-2 έχει την ικανότητα να παράγει συνεκτικό και συχνά πειστικό κείμενο. Χρησιμοποιείται για εφαρμογές όπως η παραγωγή κειμένου, η απάντηση σε ερωτήσεις, μετάφραση και δημιουργία περιεχομένου.
- **Εκπαίδευση και Προπαρασκευή:** Το GPT-2 εκπαιδεύεται σε ένα ευρύ φάσμα δεδομένων για να αποκτήσει μια γενική κατανόηση της φυσικής γλώσσας. Η μέθοδος προπαρασκευής του βασίζεται στην αυτόματη παραγωγή κειμένου χωρίς να απαιτείται ξεχωριστή εκπαίδευση για συγκεκριμένες εργασίες.

4.3.7 T5 (Text-to-Text Transfer Transformer)

Το T5, ή "Text-to-Text Transfer Transformer", είναι ένα μοντέλο στον τομέα της επεξεργασίας φυσικής γλώσσας (NLP) που αναπτύχθηκε από την Google. Το T5 αποσκοπεί να απλοποιήσει την NLP μετατρέποντας όλες τις εργασίες γλώσσας σε ένα ενιαίο πρόβλημα "κειμένου προς κείμενο" (text-to-text)[41].
Όσον αφορά στα embeddings που παράγει το T5:

- **Word (Token) Embeddings**: Το T5 μετατρέπει κάθε λέξη ή token σε ένα διάνυσμα. Αυτά τα embeddings αποτελούν τη βάση για την επεξεργασία του κειμένου, αντιπροσωπεύοντας τη σημασία κάθε λέξης μέσα στην πρόταση.
- **Positional Embeddings**: Όπως και άλλα μοντέλα βασισμένα σε transformers, το T5 χρησιμοποιεί επίσης positional embeddings για να κωδικοποιήσει τη θέση κάθε λέξης στην πρόταση. Αυτό βοηθά το μοντέλο να κατανοεί τη σημασία των λέξεων σε σχέση με τις άλλες και τη σειρά τους.

Ας δούμε τις βασικές τεχνικές λεπτομέρειες του T5:

- **Αρχιτεκτονική Μοντέλου:** Το T5 βασίζεται στην αρχιτεκτονική Transformer, παρόμοια με το BERT. Ωστόσο, αντί να χρησιμοποιεί διαφορετικές αρχιτεκτονικές για τις εργασίες ενσωμάτωσης και παραγωγής κειμένου, το T5 εφαρμόζει την ίδια αρχιτεκτονική Transformer σε όλες τις εργασίες.
- **Διαστάσεις και Παραμέτροι:** Το T5 διατίθεται σε διάφορες εκδόσεις με διαφορετικό αριθμό παραμέτρων, από το T5-Small με 60 εκατομμύρια παραμέτρους μέχρι το T5-XXL με περίπου 11 δισεκατομμύρια παραμέτρους.
- **Εφαρμογές στην NLP:** Με τη μετατροπή όλων των εργασιών σε μορφή "κειμένου προς κείμενο", το T5 μπορεί να εκπαιδευτεί σε διάφορες εργασίες όπως η ανάλυση συναισθήματος, η απάντηση σε ερωτήσεις και η περίληψη κειμένου.
- **Εκπαίδευση και Προπαρασκευή:** Το T5 εκπαιδεύεται σε ένα εκτεταμένο σύνολο δεδομένων, χρησιμοποιώντας τη μέθοδο της "denoising" για να μάθει να ανακτά το αρχικό κείμενο από μια διαταραγμένη εκδοχή του, ενισχύοντας έτσι την ικανότητα κατανόησης και παραγωγής κειμένου.

4.4 Αξιολόγηση Μοντέλων Με Βάση Αποτελεσμάτων

Σε αυτήν την ενότητα, θα αξιολογήσουμε τα μοντέλα βάσει των αποτελεσμάτων που επιτυγχάνουν σε επιλεγμένες εργασίες, χρησιμοποιώντας μετρήσεις όπως η ακρίβεια. Πιο συγκεκριμένα, προσπαθούμε να έχουμε όσο το δυνατόν χαμηλότερα ποσοστά ομοιότητας καθώς με βάση την δομή και την λειτουργία των Iptables διαφορετικές λέξεις στον κανόνα έχουν τελείως διαφορετική συμπεριφορά σε μια πολιτικής ασφαλείας.

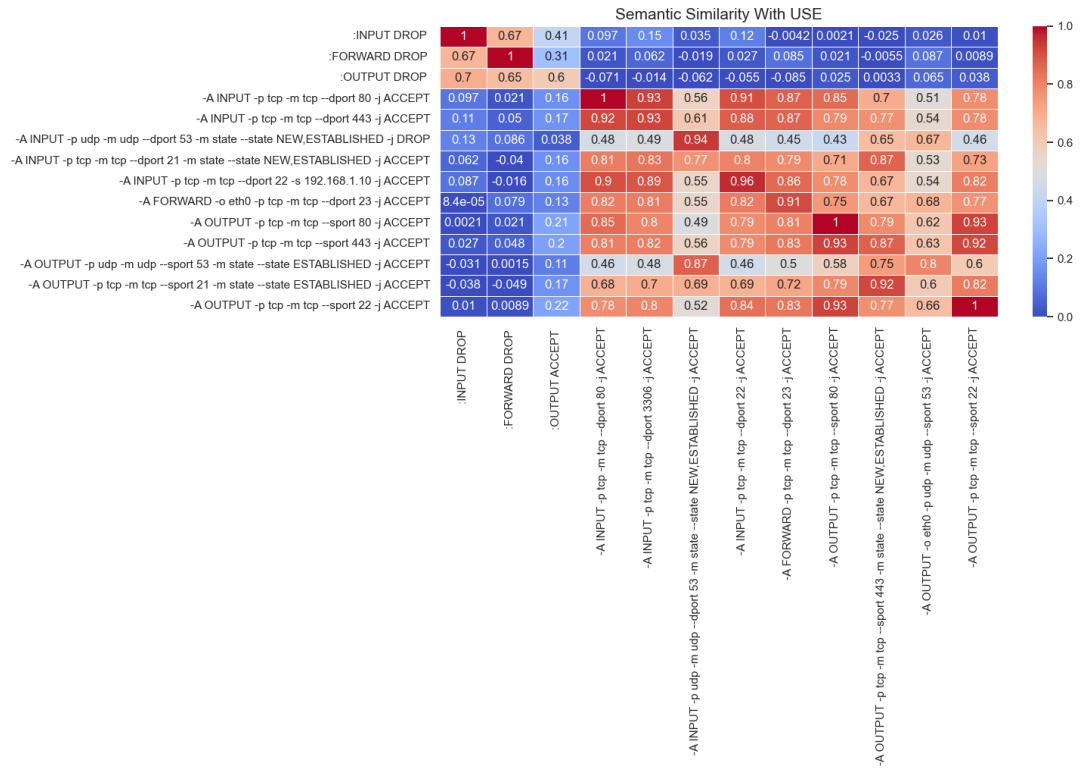
Η παρουσίαση των αποτελεσμάτων μας θα λάβει τη μορφή μιας διαδραστικής θερμοκρασιακής καταγραφής (heatmap), όπου θα συγκρίνουμε δύο διαφορετικές πολιτικές ασφαλείας βασισμένες σε κανόνες Iptables. Στη σύγκριση αυτή, θα αξιολογήσουμε τις πολιτικές ασφαλείας με βάση τα embeddings που έχουν παραχθεί από κάθε αλγόριθμο. Ως

κεντρικός μετρικός αλγόριθμος για την αξιολόγηση της ομοιότητας των embeddings χρησιμοποιούμε την συνημιτονοειδή ομοιότητα (cosine similarity).

Αυτή η μέθοδος παρέχει μια οπτικά προσβάσιμη και εύκολα κατανοητή απεικόνιση του πόσο κοντά ή μακριά βρίσκονται οι διάφορες πολιτικές ασφαλείας μεταξύ τους, όπου οι παραλλαγές στο χρώμα και την ένταση στο heatmap αντικατοπτρίζουν την ομοιότητα ή τη διαφορά τους. Η χρήση της κοσινοειδούς ομοιότητας είναι ιδιαίτερα χρήσιμη σε αυτή την περίπτωση, καθώς αυτή η μετρική είναι αποτελεσματική στο να καταδείξει την ομοιότητα μεταξύ διανυσμάτων σε χώρους υψηλής διάστασης, όπως είναι τα embeddings που προκύπτουν από τους αλγορίθμους NLP.
Παρακάτω εμφανίζουμε ένα ένα τα αποτελέσματα των μοντέλων.

4.4.1 Αποτελέσματα Μοντέλου USE

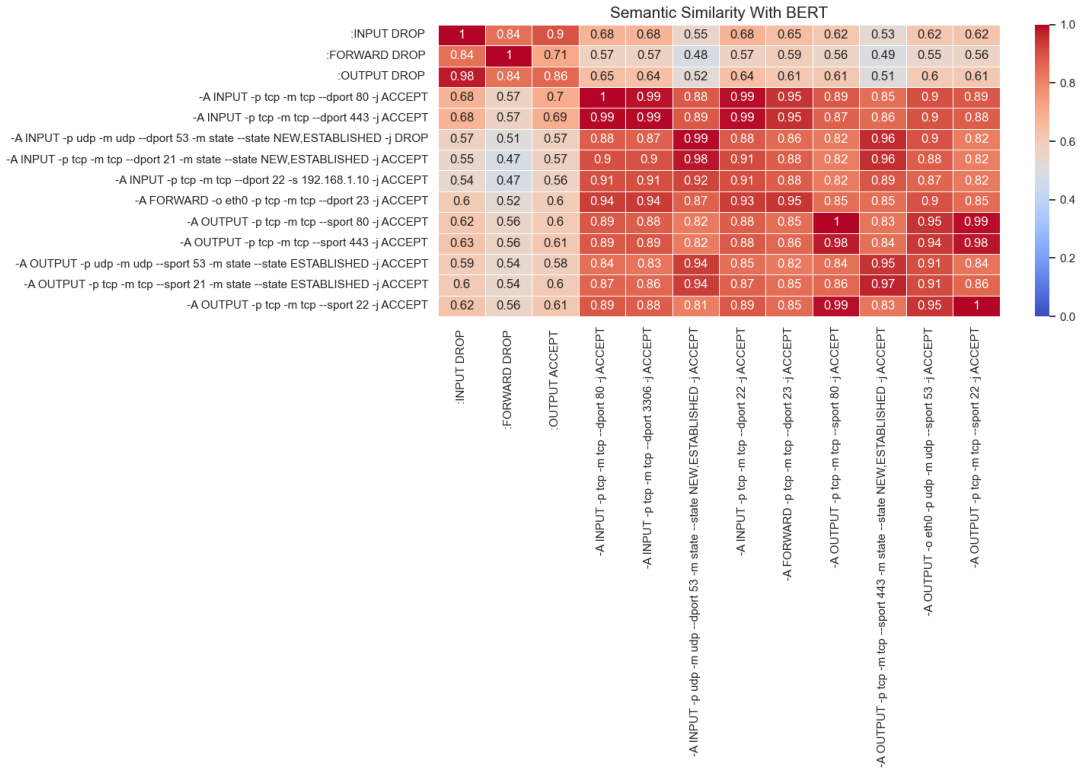
Για την αξιολόγηση του μοντέλου θα χρησιμοποιούμε την έκδοση με βάση το Transformer όπου περιέχει 512 διαστάσεις. Στην εικόνα 2 παρουσιάζονται τα αποτελέσματα σύγκρισης των embeddings από κανόνες Iptables δύο διαφορετικών πολιτικών ασφαλείας, όπου παρατηρούμε εξαιρετικά ικανοποιητικά αποτελέσματα καθώς βλέπουμε χαμηλά ποσοστά ομοιότητας.



Εικόνα 2 Σύγκριση με USE

4.4.2 Αποτελέσματα Μοντέλου BERT

Για την αξιολόγηση του μοντέλου θα χρησιμοποιούμε την έκδοση bert-large-uncased όπου περιέχει 1024 διαστάσεις με περίπου 340 εκατομμύρια παραμέτρους. Στην εικόνα 3 παρουσιάζονται τα αποτελέσματα σύγκρισης των embeddings κανόνων Iptables δύο διαφορετικών πολιτικών ασφαλείας, όπου παρατηρούμε καλά αποτελέσματα. Τα ποσοστά ομοιότητας είναι σχετικά μεγαλύτερα με αυτά του μοντέλου USE.

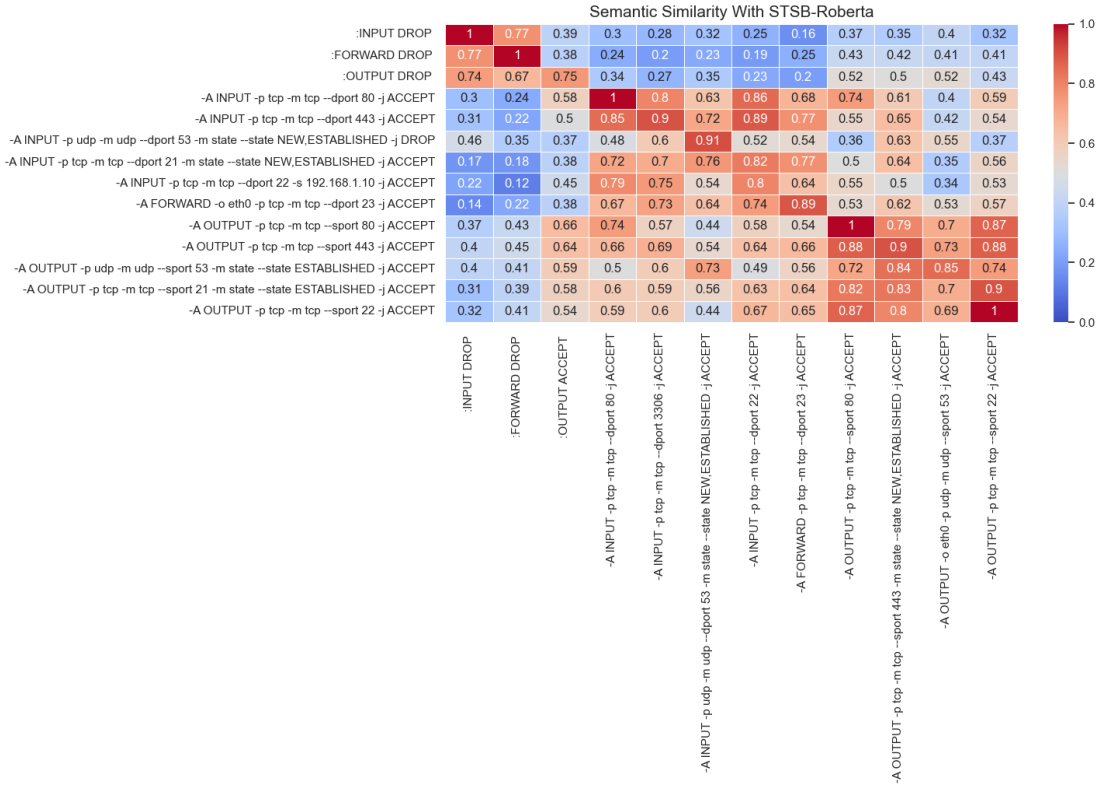


Εικόνα 3 Σύγκριση με BERT

4.4.3 Αποτελέσματα Μοντέλου STSB-Roberta

Για την αξιολόγηση του μοντέλου θα χρησιμοποιούμε μια βελτιωμένη έκδοση της Roberta την STSB-Roberta-large (Semantic Textual Similarity Benchmark - STS-B) όπου περιέχει 1024 διαστάσεις με περίπου 355 εκατομμύρια παραμέτρους.

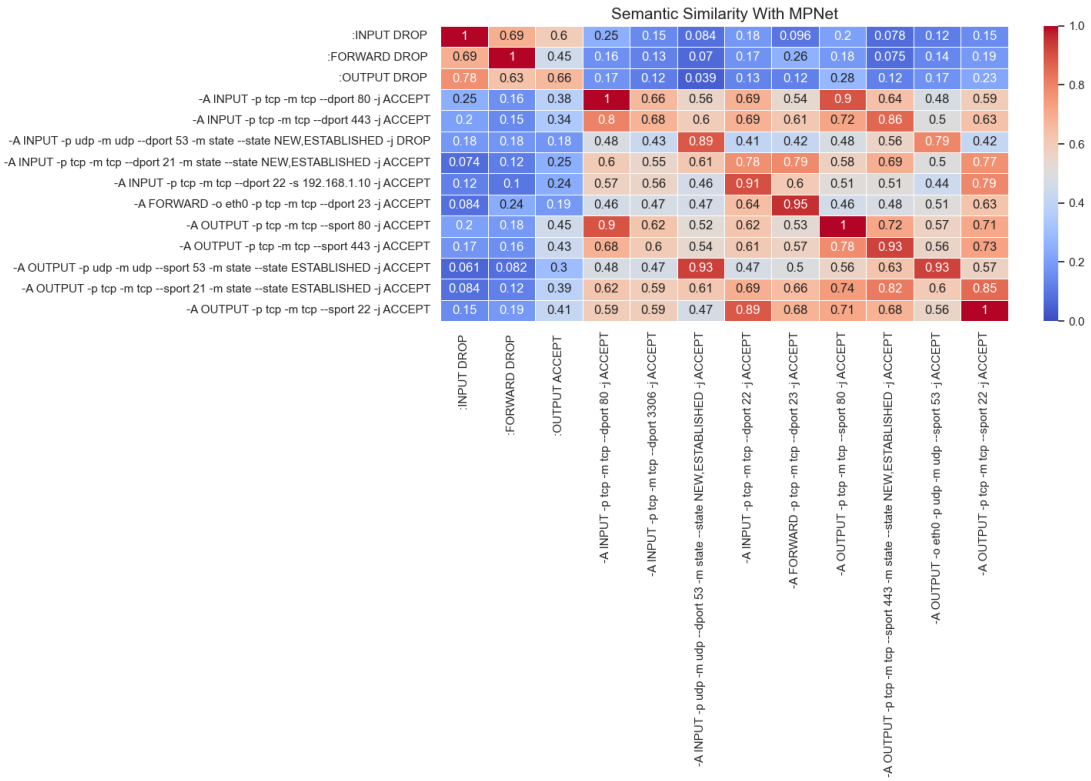
Το "STSB-Roberta-large" έχει ειδικά προσαρμοστεί και εκπαιδευτεί για την αξιολόγηση της σημασιολογικής ομοιότητας μεταξύ προτάσεων, ενσωματώνοντας αυτά τα είδη embeddings για την αποτελεσματική και ακριβή ανάλυση των κειμένων κάτι που σημαίνει ότι δίνει έμφαση στην παραγωγή ποιοτικών sentence embeddings. Στην εικόνα 4 παρουσιάζονται τα αποτελέσματα μιας σύγκρισης των embeddings κανόνων Iptables δύο διαφορετικών πολιτικών ασφαλείας, όπου παρατηρούμε ικανοποιητικά αποτελέσματα. Δίνει καλύτερα αποτελέσματα σε σχέση με το μοντέλο Bert και αρκετά καλά αποτελέσματα όπως και το μοντέλο USE.



Εικόνα 4 Σύγκριση με STSB-Roberta

4.4.4 Αποτελέσματα Μοντέλου MPNet

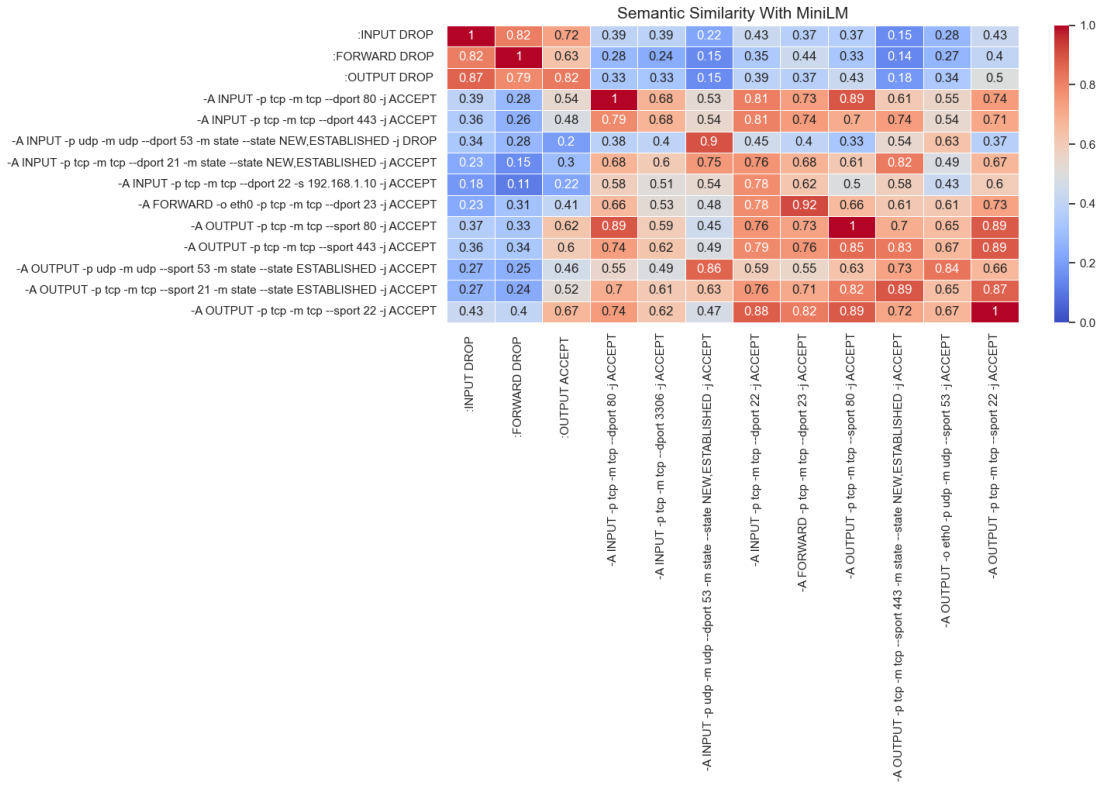
Για την αξιολόγηση του μοντέλου θα χρησιμοποιούμε την έκδοση all-mpnet-base-v2, μια παραλλαγή του MPNet όπου περιέχει 768 διαστάσεις. Χρησιμοποιείται κυρίως για εργασίες όπως η κατανόηση κειμένου, η σημασιολογική ανάλυση, και άλλες εφαρμογές στην επεξεργασία φυσικής γλώσσας. Ενώ τα βασικά embeddings που παράγει το μοντέλο είναι σε επίπεδο λέξεων, η συνδυασμένη ανάλυση των word και positional embeddings επιτρέπει την εκτίμηση και την ανάλυση ολόκληρων προτάσεων ή και μεγαλύτερων τμημάτων κειμένου. Στην εικόνα 5 παρουσιάζονται τα αποτελέσματα μιας σύγκρισης των embeddings κανόνων Iptables δύο διαφορετικών πολιτικών ασφαλείας, όπου παρατηρούμε εξαιρετικά ικανοποιητικά αποτελέσματα. Δίνει πάρα πολύ καλά αποτελέσματα μαζί με τα μοντέλα USE και Roberta.



Εικόνα 5 Σύγκριση με MPNet

4.4.5 Αποτελέσματα Μοντέλου MiniLM

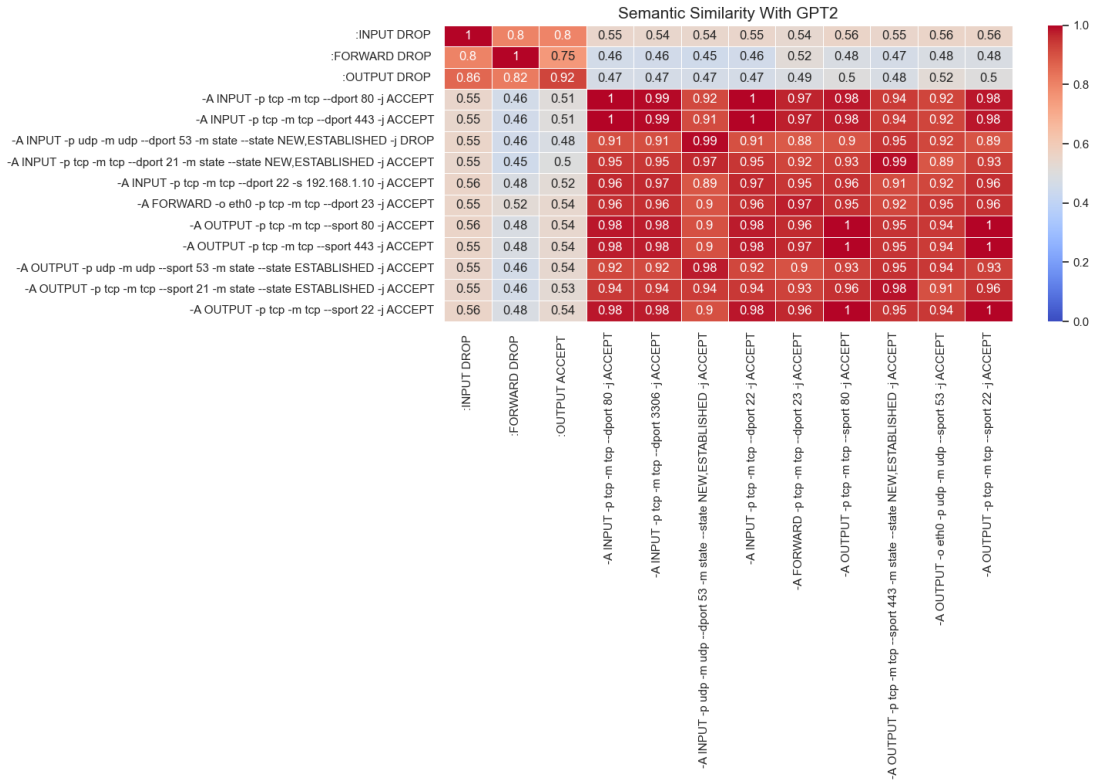
Για την αξιολόγηση του μοντέλου θα χρησιμοποιούμε την έκδοση all-MiniLM-L12-v2 όπου περιέχει 384 διαστάσεις. Είναι μια ειδική εκδοχή του Sentence-BERT που χρησιμοποιεί την αρχιτεκτονική MiniLM, συνδυάζοντας τα πλεονεκτήματα του s-bert στην παραγωγή sentence embeddings με την αποδοτικότητα και την ελαφρότητα του MiniLM. Στην εικόνα 6 παρουσιάζονται τα αποτελέσματα μιας σύγκρισης των embeddings κανόνων Iptables δύο διαφορετικών πολιτικών ασφαλείας, όπου παρατηρούμε ικανοποιητικά αποτελέσματα όπως και τα προηγούμενα αποτελέσματα.



Εικόνα 6 Σύγκριση με MiniLM

4.4.6 Αποτελέσματα Μοντέλου GPT-2

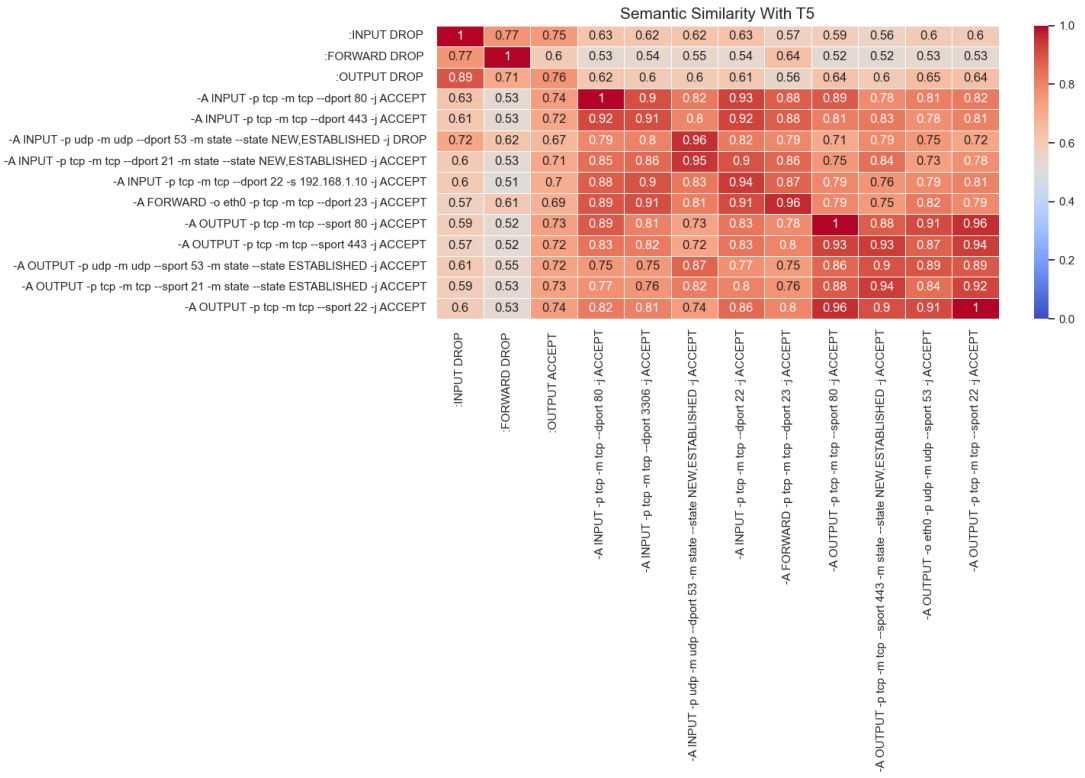
Για την αξιολόγηση του μοντέλου θα χρησιμοποιούμε την έκδοση gpt2-large όπου περιέχει 1280 διαστάσεις με 774 εκατομμύρια παραμέτρους. Στην εικόνα 7 παρουσιάζονται τα αποτελέσματα μιας σύγκρισης των embeddings κανόνων Iptables δύο διαφορετικών πολιτικών ασφαλείας, όπου δεν παρατηρούμε καλά αποτελέσματα. Δίνει αρκετά υψηλές τιμές συγκρίσεων σε αντίθεση με τα άλλα μοντέλα.



Εικόνα 7 Σύγκριση με GPT2

4.4.7 Αποτελέσματα Μοντέλου T5

Για την αξιολόγηση του μοντέλου θα χρησιμοποιούμε την έκδοση gtr-t5-xl όπου περιέχει 768 διαστάσεις. Το μοντέλο GTR-T5-XL (Generative Transformer Retrieval T5 Extra Large) είναι μια προηγμένη εκδοχή του T5 που αναπτύχθηκε για τη βελτιστοποίηση στην ανάκτηση πληροφορίας. Αυτό το μοντέλο, όπως και το πρωτότυπο T5, δεν παράγει απευθείας sentence embeddings με τον ίδιο τρόπο που το κάνουν μοντέλα όπως το BERT ή το Sentence-BERT. Ωστόσο, λόγω της φύσης των εργασιών ανάκτησης πληροφορίας, το GTR-T5-XL μπορεί να είναι πιο κατάλληλο για τη δημιουργία ή τη χρήση sentence embeddings σε σύγκριση με το αρχικό T5. Στην εικόνα 8 παρουσιάζονται τα αποτελέσματα μιας σύγκρισης embeddings κανόνων Iptables δύο διαφορετικών πολιτικών ασφαλείας, όπου παρατηρούμε καλά αποτελέσματα αλλά μεγαλύτερα ποσοστά τιμών σε σχέση με τα υπόλοιπα μοντέλα.



Εικόνα 8 Σύγκριση με T5

4.5 Επιλογή Μοντέλου Δημιουργίας Embeddings Μέσω Σύγκρισης

Για να κρίνουμε τις επιδόσεις ενός μοντέλου, πρέπει πρώτα να ορίσουμε ένα κατώφλι (threshold). Αυτό σημαίνει ότι από την στιγμή όπου μία μόνο λέξη ή γράμμα στους κανόνες των Iptables μπορεί να προκαλέσει τελείως διαφορετική συμπεριφορά στην πολιτική ασφαλείας, θεωρούμε ότι όσο χαμηλότερη είναι η τιμή στη σύγκριση μεταξύ των κανόνων, τόσο καλύτερο είναι το αποτέλεσμα.

Ακόμα, βάση του ορισμένου κατωφλίου και των ευρημάτων από την αξιολόγηση διάφορων μοντέλων, διαπιστώσαμε ότι τα sentence embeddings, ιδιαίτερα τα μοντέλα USE και MPNet, διακρίθηκαν για τις επιδόσεις τους. Αν και αξιολογήθηκαν πολλά προχωρημένα και δημοφιλή μοντέλα με ευρείς δυνατότητες και παραμέτρους, τα sentence embeddings αποδείχτηκαν τα πιο αποτελεσματικά, ειδικά στη σημασιολογική ανάλυση και επεξεργασία κειμένου.

Για μια πιο λεπτομερή σύγκριση των μοντέλων, δημιουργήσαμε ένα διάγραμμα, όπως φαίνεται στην εικόνα 9, που παρουσιάζει τις τιμές ομοιότητας για 14 τυχαία ζευγάρια κανόνων Iptables. Από τα αρχικά 7 μοντέλα, επιλέξαμε να συγκρίνουμε περαιτέρω τα 5 πιο αποδοτικά, αφήνοντας έξω τα GPT2 και T5, λόγω των υψηλών τιμών που παρουσίασαν. Έτσι, επικεντρωθήκαμε στη σύγκριση των μοντέλων USE, BERT, STSB-Roberta, MPNet και MiniLM.

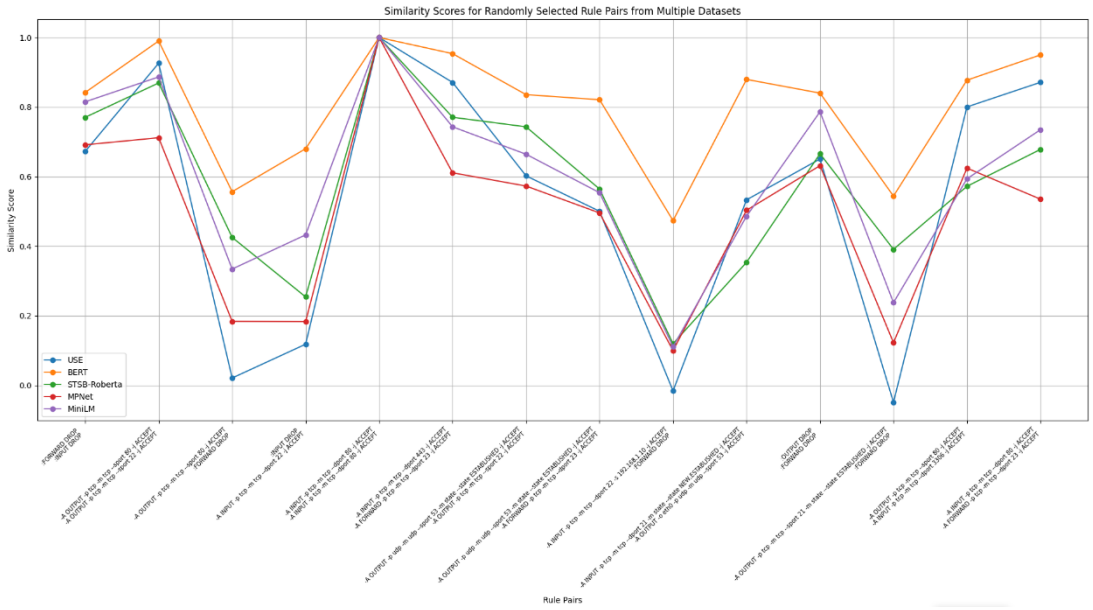
Τα ευρήματα από την ανάλυση του διαγράμματος περιλαμβάνουν τα εξής στοιχεία:

- Το μοντέλο BERT καταγράφει συνεπώς υψηλές τιμές με σχετικά στενό εύρος διακύμανσης.

- Τα άλλα μοντέλα εμφανίζουν παρόμοιες μεταξύ τους επιδόσεις σε αρκετά ζευγάρια συγκρίσεων.
- Το μοντέλο MPNet, μαζί με το USE, εμφανίζει εξαιρετικά αποτελέσματα.
- Συχνότερα όμως το μοντέλο USE παρουσιάζει τις χαμηλότερες τιμές ανάμεσα στα μοντέλα.

Επιλέγουμε το μοντέλο USE ως το πιο κατάλληλο για τις ανάγκες μας λόγω των ακόλουθων πλεονεκτημάτων:

- **Ειδίκευση στη Σημασιολογική Ανάλυση**: Το USE έχει σχεδιαστεί ειδικά για σημασιολογική ανάλυση, παρέχοντας πιο ακριβείς και σχετικές αποτιμήσεις.
- **Συνεπείς και Χαμηλές Τιμές:** Παρουσιάζει συνεπώς χαμηλές τιμές στις συγκρίσεις, δείχνοντας υψηλή ακρίβεια και συνέπεια.
- **Στοχευμένη Επιλογή Τεχνολογίας**: Η επιλογή του USE επισημαίνει τη σημασία της επιλογής εξειδικευμένων τεχνολογιών, που ανταποκρίνονται στις συγκεκριμένες ανάγκες και στόχους μας όπως η αρχιτεκτονική transformers και η εφαρμογή των sentence embeddings.



Εικόνα 8 Σύγκριση Μοντέλων

5 Μεθοδολογία Σύγκρισης Πολιτικών Ασφαλείας

Σε αυτό το κεφάλαιο θα αναλύσουμε την μεθοδολογία μας πάνω στην σύγκριση των πολιτικών ασφαλείας. Πιο συγκεκριμένα θα αναλύσουμε πρώτα με ποιον τρόπο επεξεργαζόμαστε, συγκρίνουμε και οπτικοποιούμε τα αποτελέσματα για τις πολιτικές ασφαλείας μας.

5.1 Επεξεργασία Πολιτικών Ασφαλείας

Πριν προχωρήσουμε στην σύγκριση των πολιτικών θα πρέπει πρώτα να δημιουργήσουμε τα embeddings μας δηλαδή να επεξεργαστούμε τις πολιτικές μας. Για να φτάσουμε όμως σε αυτό το σημείο θα πρέπει να θέσουμε κάποιες προϋποθέσεις όπως και διαχωρισμού των εντολών.

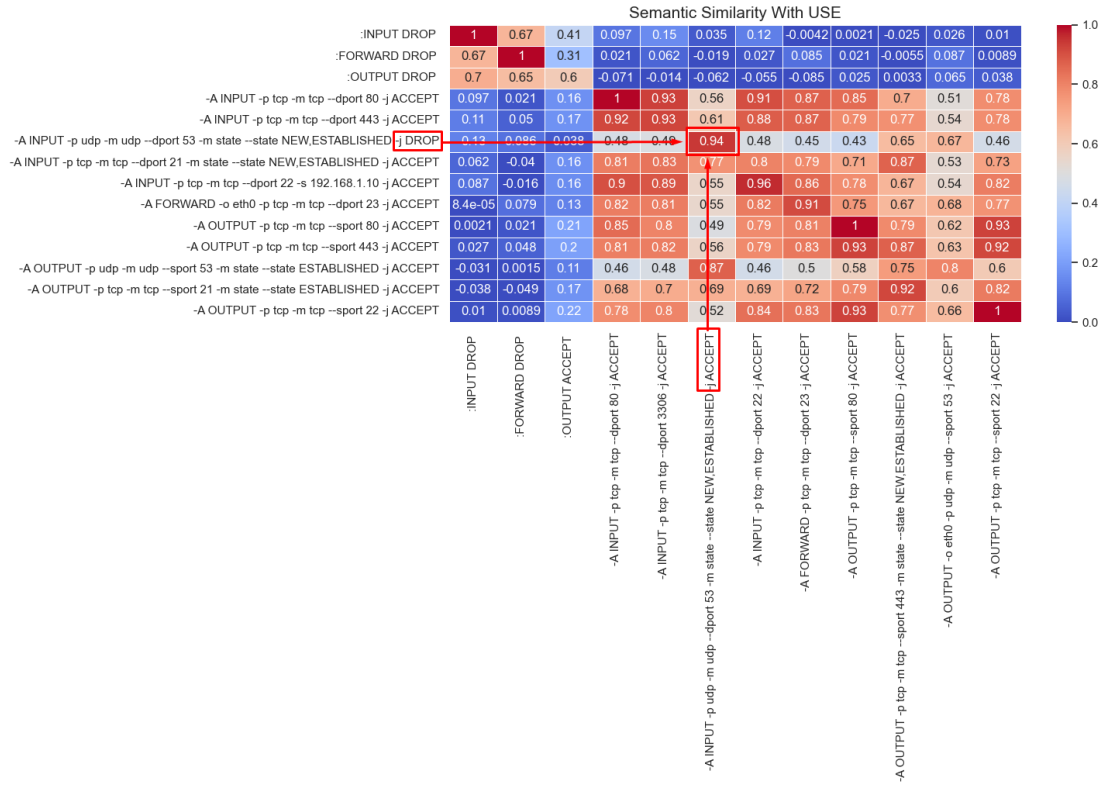
5.1.1 Θέσπιση Προϋποθέσεων Για Επεξεργασία Πολιτικών Ασφαλείας

Σε αυτό το βήμα θέτουμε προϋποθέσεις για να περάσουμε σε επεξεργασία των εντολών των πολιτικών ασφαλείας.

Όπως παρατηρούμε στην εικόνα 10 διαπιστώνουμε ότι υπάρχουν κανόνες που στις πολιτικές ασφαλείας τους έχουν τελείως διαφορετική λειτουργία, όμως στην σύγκριση μεταξύ τους μας εμφανίζεται ότι έχουν τεράστια ομοιότητα. Για παράδειγμα βλέπουμε ότι έχουμε δύο εντολές τις :

- -A INPUT -p udp -m udp --dport 53 -m state --state NEW,ESTABLISHED -j DROP
- -A INPUT -p udp -m udp --dport 53 -m state --state NEW,ESTABLISHED -j ACCEPT

Στις εντολές αυτές το μόνο στοιχείο που αλλάζει είναι ο στόχος του κανόνα -j (-jump). Παρόλο που με βάση τον στόχο έχουμε τελείως διαφορετική συμπεριφορά των κανόνων, στο heatmap και στην σύγκριση που κάνουμε διαπιστώνουμε ότι μας δίνει ομοιότητα των δύο κανόνων της τάξης του 94%. Με βάση αυτό το αποτέλεσμα αλλά και με βάση άλλα υψηλά αποτελέσματα ομοιότητας αν και είχαν διαφορετικές θύρες και διαφορετικές αλυσίδες θεσπίζουμε προϋποθέσεις έτσι ώστε να αποφασίσουμε στο ποιες εντολές θα υποστούν περεταίρω επεξεργασία για να βρεθεί το ποσοστό ομοιότητας τους και ποιες όχι.



Εικόνα 9 Heatmap σύγκρισης όλων των εντολών μεταξύ τους

Πιο συγκεκριμένα για να πούμε ότι δύο εντολές, ανάμεσα στις πολιτικές ασφαλείας, έχουν κάποια ομοιότητα μεταξύ τους τότε θα πρέπει:

- Να ανήκουν στην ίδια αλυσίδα (chain). Εάν έχουμε ακριβώς τις ίδιες εντολές αλλά σε διαφορετική αλυσίδα η ομοιότητα μεταξύ των εντολών θα είναι πάρα πολύ υψηλή αλλά το περιεχόμενο θα είναι τελείως διαφορετικό.
- Να απευθύνονται στην ίδια θύρα (port). Ακόμα και ίδιες εντολές να έχουμε με διαφορετικές θύρες το περιεχόμενο της εντολής πλέον είναι τελείως διαφορετικό αλλά το ποσοστό ομοιότητας είναι μεγάλο.
- Να έχουμε ίδιο στόχο του κανόνα (target), δηλαδή εάν θα πρέπει να έχουμε drop ή accept των πακέτων σε κάποιον συγκεκριμένο κανόνα. Με μια μόνο διαφορετική λέξη ανάμεσα στις εντολές μπορεί να έχουμε μικρή απόκλιση στην ομοιότητα αλλά μεγάλη στο περιεχόμενο των εντολών.

Αυτές οι εντολές που δεν πληρούν αυτές τις προϋποθέσεις, διακρίνονται για το γεγονός ότι δεν ικανοποιούν καμία συναφή ιδιότητα ανάμεσα στις δύο πολιτικές. Με άλλα λόγια, δεν υπάρχει κοινό στοιχείο που να συνδέει αυτές τις εντολές, καθιστώντας τις ανεξάρτητες.

5.1.2 Διαχωρισμός Εντολών Και Δημιουργία Embeddings

Βάσει των προϋποθέσεών μας, μπορούμε να οργανώσουμε τους κανόνες μας σε τρεις κύριες κατηγορίες ως εξής:

- **Πανομοιότυποι Κανόνες**: Αυτοί οι κανόνες είναι ακριβώς ίδιοι και στις δύο πολιτικές. Δεν υπάρχει καμία διαφορά ανάμεσα στους κανόνες αυτούς.
- **Ανόμοιοι Κανόνες:** Αυτοί οι κανόνες δεν έχουν καμία ομοιότητα με οποιονδήποτε άλλον κανόνα ανάμεσα στις δύο πολιτικές. Είναι μοναδικοί και διαφορετικοί σε κάθε πολιτική.
- **Εν μέρη Όμοιοι Κανόνες:** Αυτοί οι κανόνες έχουν ομοιότητες ανάμεσα τους, αλλά ταυτόχρονα διαφέρουν σε άλλα τμήματα των κανόνων τους.

Αφού έχουμε διαχωρίσει τους κανόνες στις παραπάνω κατηγορίες, το επόμενο βήμα είναι να ασχοληθούμε με τους κανόνες που είναι "εν μέρη όμοιοι". Σύμφωνα με τις προϋποθέσεις μας, επικεντρωνόμαστε στους κανόνες που έχουν την ίδια αλυσίδα (chain), την ίδια θύρα (port) και τον ίδιο στόχο (target) ανάμεσα στις δύο πολιτικές. Ωστόσο, αυτοί οι κανόνες διαφέρουν σε άλλα τμήματα τους.

Για να προχωρήσουμε περαιτέρω, χρησιμοποιούμε το μοντέλο USE (Universal Sentence Encoder) για να μετατρέψουμε αυτούς τους κανόνες σε embeddings. Αυτή η διαδικασία επιτρέπει να αναγνωρίσουμε περαιτέρω παρόμοιες πτυχές μεταξύ των κανόνων και να λάβουμε αποφάσεις ή να εξάγουμε συμπεράσματα βάσει αυτών των αναπαραστάσεων.

5.2 Σύγκριση Πολιτικών Ασφαλείας

Η σύγκριση πολιτικών ασφαλείας και η οπτικοποίηση αποτελεσμάτων αποτελούν κρίσιμα στοιχεία για την ανάλυση και τη βελτίωση των συστημάτων ασφαλείας. Μέσω της σύγκρισης διάφορων πολιτικών, είναι δυνατόν να αναδειχθούν οι αδυναμίες που μπορεί να υπάρχουν, επιτρέποντας την εφαρμογή βέλτιστων πρακτικών. Ταυτόχρονα, η οπτικοποίηση των αποτελεσμάτων παρέχει έναν ευκρινή και προσιτό τρόπο παρουσίασης των δεδομένων, επιτρέποντας στους ενδιαφερόμενους να κατανοήσουν πιο αποτελεσματικά την απόδοση και την αποτελεσματικότητα των πολιτικών ασφαλείας. Συνεπώς, η συνδυασμένη χρήση αυτών των δύο προσεγγίσεων αποτελεί καίριο εργαλείο για την προώθηση της ασφάλειας και της προστασίας των συστημάτων.

5.2.1 Επιλογή Μετρικής Ομοιότητας

Στην ενότητα μας εξετάζουμε τη διαδικασία επιλογής μετρικών ομοιοτήτων για την σύγκριση πολιτικών ασφαλείας. Πιο συγκεκριμένα χρειάζεται να επιλέξουμε μια μετρική ομοιότητας, που καθορίζει τον τρόπο με τον οποίο θα μετρηθεί η συνάφεια μεταξύ των αναπαραστάσεων. Η απόφαση της προσέγγισης αυτής εξαρτάται συχνά από τη φύση των δεδομένων και τον επιδιωκόμενο στόχο ανάλυσης.

Η επιλογή της μετρικής ομοιότητας για τη σύγκριση των embeddings είναι κρίσιμη για την αξιολόγηση της συναφούς πληροφορίας που περιέχουν. Αν και οι επιλογές είναι ποικίλες, υπάρχουν κάποιοι γενικοί παράγοντες που πρέπει να ληφθούν υπόψη:

- **Είδος των Δεδομένων:**
  - Εάν τα ενσωματωμένα προέρχονται από εικόνες, μια μετρική όπως η cosine similarity μπορεί να είναι κατάλληλη λόγω της ικανότητάς της να αντιμετωπίζει αλλοιώσεις στην κλίμακα.
  - Για ενσωματωμένα που αντιπροσωπεύουν κείμενο, μετρικές όπως η Jaccard similarity ή η cosine similarity μπορεί να είναι κατάλληλες.
- **Υπολογιστική Πολυπλοκότητα:**
  - Επειδή ορισμένες μετρικές μπορεί να απαιτούν περισσότερους υπολογισμούς, η υπολογιστική πολυπλοκότητα πρέπει επίσης να ληφθεί υπόψη, ιδιαίτερα σε μεγάλα σύνολα δεδομένων.

Η ανάγκη για υπολογισμό ομοιότητας μεταξύ διανυσμάτων αποτελεί βασικό στάδιο στην σύγκριση κειμένων όπου είναι και το στοιχείο μας. Σε αυτό το πλαίσιο, η επιλογή του κατάλληλου αλγορίθμου ομοιότητας και του κατάλληλου εύρους αξιολόγησης είναι ζωτικής σημασίας για την ακρίβεια και την αποδοτικότητα των αναλύσεων. Πιο συγκεκριμένα η κύρια αναζήτηση μετρικού αλγορίθμου επικεντρώνεται στην εμφάνιση αποτελεσμάτων ανάμεσα σε συγκριμένο εύρος τιμών έτσι ώστε να βγάλουμε στο τέλος ένα ποσοστό ομοιότητας όπου κυμαίνεται ανάμεσα στο 0 και στο 100.
Σε αυτό το πλαίσιο, παρουσιάζονται δύο προσεγγίσεις.

- Η πρώτη χρησιμοποιεί αλγόριθμους με ενσωματωμένα διανύσματα (embeddings), προσφέροντας έναν αριθμό μεταξύ του εύρους [-1,1]. Αυτός ο αριθμός αντικατοπτρίζει το βαθμό ομοιότητας μεταξύ των διανυσμάτων, όπου το -1 αντιπροσωπεύει την πλήρη αντίθεση και το 1 την πλήρη ομοιότητα. Η κλιμάκωση των τιμών στο εύρος [0, 1] διευκολύνει τη σύγκριση και την αξιολόγηση και είναι εφικτή. Σε αυτήν την περίπτωση οι μετρικοί αλγόριθμοι μπορεί να είναι η cosine similarity ή η pearson correlation.

- Η δεύτερη προσέγγιση επικεντρώνεται σε δυαδικά ενσωματωμένα διανύσματα, χρησιμοποιώντας αλγορίθμους που παράγουν αξιολογήσεις στο εύρος [0, 1] με δυαδικούς αριθμούς. Εδώ, η επιλογή μετρικής, όπως η dice similarity ή η jaccard score, αντανακλά τη φύση των δυαδικών ενσωματωμένων διανυσμάτων και προσφέρει συγκεκριμένο εύρος αξιολόγησης. Η μετατροπή ενός δεκαδικού αριθμού σε έναν δυαδικό είναι εξίσου εφικτή.

- Εντούτοις, αποφασίσαμε να αποκλείσουμε αλγορίθμους ομοιότητας όπως την Απόσταση Levenshtein, την Απόσταση Hamming, την Ευκλείδεια Απόσταση και την Απόσταση Manhattan. Αυτό οφείλεται στο γεγονός ότι, παρά τη διαδεδομένη χρήση τους, δεν προσφέρουν ένα συγκεκριμένο εύρος τιμών στο οποίο μπορούν να κλιμακωθούν. Στη συγκεκριμένη περίπτωση, αναζητούμε μια αξιολόγηση στο εύρος [0, 1] ή σε ένα άλλο συγκεκριμένο εύρος που μπορεί να προσαρμοστεί με ευελιξία.

Με βάση τις προσεγγίσεις αναλύσαμε τους εξής μετρικούς αλγορίθμους :

- **Ομοιότητα Συνημίτονου (Cosine Similarity):**

  Ο αλγόριθμος Cosine Similarity χρησιμοποιείται για τον υπολογισμό της ομοιότητας μεταξύ δύο αναπαραστάσεων. Είναι βασισμένος στη γωνία μεταξύ των διανυσμάτων και παρέχει μια ποσοτική εκτίμηση της ομοιότητας[42].

  Ειδικότερα, η Ομοιότητα Συνημίτονου (Cosine Similarity) είναι μια μέθοδος που χρησιμοποιείται για την μέτρηση της ομοιότητας μεταξύ δύο διανυσμάτων. Συνήθως χρησιμοποιείται σε περιβάλλοντα όπου τα δεδομένα εκφράζονται ως διανύσματα, όπως στην ανάλυση κειμένου και την εξόρυξη πληροφοριών. Μετρά τη γωνία μεταξύ δύο διανυσμάτων σε έναν πολυδιάστατο χώρο και προσδιορίζει πόσο κοντά είναι αυτά τα δύο διανύσματα μεταξύ τους. Η μέθοδος χρησιμοποιείται συχνά σε προβλήματα συσταδοποίησης, ανάκτησης πληροφοριών και αναγνώρισης προτύπων. Γενικά είναι κατάλληλη για σύγκριση συνεχών συνόλων διανυσμάτων.

  Η υπολογιστική φόρμουλα για την Ομοιότητα Συνημίτονου μεταξύ δύο διανυσμάτων X και Y είναι η εξής:

$$\cos(\theta) = \frac{X \cdot Y}{\|X\|\|Y\|} \frac{\sum_{i=1}^{n} X_i Y_i}{\sqrt{\sum_{i=1}^{n} X_i^2} \sqrt{\sum_{i=1}^{n} Y_i^2}}$$

Όπου:

- ‖X‖ και ‖Y‖ είναι τα συνήθη διανύσματα.

Η τιμή της Ομοιότητας Συνημίτονου βρίσκεται μεταξύ -1 και 1. Όταν τα δύο διανύσματα είναι ίδια, η ομοιότητα είναι 1. Αντίθετα, όταν τα δύο διανύσματα είναι αντίθετα, η ομοιότητα είναι -1. Όταν δεν υπάρχει καμία ομοιότητα μεταξύ των δύο διανυσμάτων, η τιμή είναι 0.

Η Ομοιότητα Συνημίτονου χρησιμοποιείται σε πολλές εφαρμογές, όπως η σύγκριση εγγράφων κειμένου, η προτεινόμενη αναζήτηση και η συσταδοποίηση κειμένου.

- **Pearson συντελεστής συσχέτισης (Pearson correlation coefficient):**

Ο Συντελεστής Συσχέτισης Pearson (Pearson correlation coefficient) είναι ένα μέτρο στατιστικής που μετρά τον βαθμό συσχέτισης μεταξύ δύο μεταβλητών. Συχνά χρησιμοποιείται για να αξιολογήσει τον βαθμό που δύο μεταβλητές μετακινούνται μαζί, δηλαδή όταν η μία αυξάνεται η άλλη τείνει να αυξάνεται ή να μειώνεται.
Ο συντελεστής συσχέτισης Pearson, συμβολίζεται συνήθως με το γράμμα "r", $e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots$, $-\infty < x < \infty$ (όταν η μία μεταβλητή αυξάνεται, η άλλη τείνει να αυξάνεται), οι τιμές που πλησιάζουν το -1 υποδεικνύουν αρνητική συσχέτιση (όταν η μία μεταβλητή αυξάνεται, η άλλη τείνει να μειώνεται), ενώ ο συντελεστής 0 υποδεικνύει απουσία συσχέτισης, αντίθετα όταν η τιμή συσχέτισης πλησιάζει το 1 υποδεικνύουν θετική συσχέτιση. Γενικά είναι κατάλληλη για σύγκριση συνεχών συνόλων διανυσμάτων.
Ο τύπος του συντελεστή συσχέτισης Pearson για δύο μεταβλητές X και Y είναι:

$$r = \frac{\sum (X_i - X')(Y_i - Y')}{\sqrt{\sum (X_i - X')^2 (Y_i - Y')^2}}$$

Όπου:

- Xi και Yi είναι οι τιμές των παρατηρήσεων
- X’ και Y’ είναι οι μέσες τιμές των μεταβλητών X και Y, αντίστοιχα.

Ο Pearson συντελεστής συσχέτισης μετράει μόνο γραμμικές σχέσεις και είναι ευαίσθητος σε ακραίες τιμές (outliers)[43].

- **Ομοιότητα συσχέτισης Dice (Dice similarity coefficient):**

Η Dice Similarity είναι μια μετρική που χρησιμοποιείται για τη σύγκριση της ομοιότητας μεταξύ δύο συνόλων. Συγκεκριμένα, χρησιμοποιείται για να μετρήσει πόσο είναι παρόμοια δύο σύνολα από στοιχεία, όπως λέξεις, χαρακτήρες ή δυαδικά ψηφία αλλά γενικά είναι κατάλληλη για σύγκριση διακριτών συνόλων. Η μετρική Dice Similarity υπολογίζεται με τον τύπο:

Dice Similarity=2×|A∩B||A|+|B|Dice Similarity=|A|+|B|2×|A∩B|

$$DSC = \frac{2\left|X \cap Y\right|}{\left|X\right| + \left|Y\right|}$$

Όπου :

- X και Y είναι τα δύο σύνολα που συγκρίνονται
- |X∩Y| αναπαριστά τον αριθμό των στοιχείων που υπάρχουν ταυτόχρονα και στα δύο σύνολα.

Η τιμή της Dice Similarity κυμαίνεται από 0 έως 1, με 0 να υποδηλώνει καμία ομοιότητα και 1 να υποδηλώνει πλήρη ομοιότητα. Χρησιμοποιείται συχνά σε προβλήματα σύγκρισης κειμένων, συνόλων δεδομένων ή δυαδικών αναπαραστάσεων για να αξιολογήσει το βαθμό ομοιότητας μεταξύ τους.

- **Ομοιότητα Jacard (Jaccard Similarity):**

Η Ομοιότητα Jaccard είναι ένα μέτρο που χρησιμοποιείται για την αξιολόγηση της ομοιότητας μεταξύ δύο συνόλων. Συχνά χρησιμοποιείται σε προβλήματα που αφορούν την αναγνώριση προτύπων, την αναζήτηση πλησιέστερων γειτόνων και σε δυαδικά δεδομένα, δηλαδή σε σύνολα όπου τα στοιχεία είναι παρόντα ή απόντα (π.χ. 0 ή 1, αλήθεια ή ψέμα, κ.λπ.) αλλά γενικά είναι κατάλληλη για σύγκριση διακριτών συνόλων.

Η τιμή της Ομοιότητας Jaccard υπολογίζεται με την παρακάτω φόρμουλα:

$$JS = \frac{\left|X \cap Y\right|}{\left|X \cup Y\right|}$$

Όπου :

- X και Y είναι τα δύο σύνολα που εξετάζουμε.
- |X∩Y| είναι το πλήθος των κοινών στοιχείων ανάμεσα στα σύνολα X και Y.
- |X∪Y| είναι το πλήθος των μοναδικών στοιχείων που υπάρχουν σε κάποιο από τα δύο σύνολα.

Η τιμή της Ομοιότητας Jaccard κυμαίνεται μεταξύ 0 και 1. Όσο πιο κοντά είναι στο 1, τόσο περισσότερα κοινά στοιχεία έχουν τα σύνολα, προσεγγίζοντας την πλήρη ομοιότητα. Αν η τιμή είναι 0, δεν υπάρχουν κοινά στοιχεία ανάμεσα στα σύνολα.

Η Ομοιότητα Jaccard είναι ιδιαίτερα χρήσιμη όταν τα σύνολα που εξετάζονται είναι μεγάλα και όταν η παρουσία ή απουσία συγκεκριμένων στοιχείων είναι σημαντική, ανεξάρτητα από τη σειρά των στοιχείων στο σύνολο.

Συμπέρασμα : Με βάση τις δύο προσεγγίσεις και τις αναλύσεις των μετρικών που παρουσιάστηκαν για τον υπολογισμό της ομοιότητας μεταξύ διανυσμάτων, φαίνεται ότι η πρώτη προσέγγιση που χρησιμοποιεί αλγόριθμους με ενσωματωμένα διανύσματα και

μετρικές όπως η cosine similarity και η pearson correlation έχει τα πλεονεκτήματα της ευελιξίας και της εύκολης κλιμάκωσης των τιμών στο εύρος [0, 1]. Επιπρόσθετα, δεδομένου ότι τα embeddings βρίσκονται σε δεκαδικούς αριθμούς, για να χρησιμοποιηθούν από μετρικές που λειτουργούν με δυαδικούς αριθμούς, απαιτείται η μετατροπή τους σε δυαδική αναπαράσταση, η οποία μπορεί να επιβαρύνει τον επεξεργαστή με αυξημένη υπολογιστική ισχύ. Ακόμα ένας λόγος είναι ότι, καθώς όλοι οι αλγόριθμοι παραγωγής embeddings δημιουργούν συνεχή διανύσματα, οι μετρικές όπως η cosine similarity και η pearson correlation είναι ιδανικές καθώς χρησιμοποιούνται για τον υπολογισμό της ομοιότητας μεταξύ συνεχών διανυσμάτων. Αυτή η μέθοδος είναι ιδιαίτερα χρήσιμη σε περιπτώσεις όπου τα δεδομένα αναπαρίστανται ως διανύσματα σε έναν πολυδιάστατο χώρο, όπως συμβαίνει συχνά στην επεξεργασία φυσικής γλώσσας και συγκεκριμένα στις δύο μετρικές.

Ακόμα, η cosine similarity έχει αποδειχθεί ότι είναι ένα αποτελεσματικό μέτρο ομοιότητας για τον υπολογισμό της συσχέτισης μεταξύ διανυσμάτων, ειδικά σε περιπτώσεις όπου έχουμε να κάνουμε με αναπαραστάσεις κειμένου. Η επιλογή της συνήθως οδηγεί σε πιο ενδεδειγμένα και ερμηνεύσιμα αποτελέσματα σε σχέση με άλλες μετρικές. Ερευνητικά δεδομένα ενισχύουν το συμπέρασμα αυτό και πιο συγκεκριμένα ότι είναι η πλέον κατάλληλη μετρική για τέτοιου είδους αναλύσεις καθώς διαπιστώθηκε ότι είχε την καλύτερη τιμή καταλληλόλητας μεταξύ των συγκεκριμένων μετρικών ομοιότητας που αναλύσαμε.

Συνεπώς, με βάση τα παραπάνω, φτάνουμε στο συμπέρασμα ότι η cosine similarity είναι η καλύτερη επιλογή για τον υπολογισμό της ομοιότητας μεταξύ διανυσμάτων σε περιπτώσεις ανάλυσης κειμένων, λόγω της ευελιξίας και της αποδοτικότητάς της, καθώς και της ικανότητάς της να παρέχει ερμηνεύσιμα αποτελέσματα.

5.2.2 Τρόπος Υπολογισμού Συνολικού Ποσοστού Ομοιότητας Δύο Πολιτικών Ασφαλείας

Ο τρόπος υπολογισμού του συνολικού ποσοστού ομοιότητας μεταξύ δύο πολιτικών ασφαλείας αντιπροσωπεύει ένα κρίσιμο στοιχείο στην ανάπτυξη και αξιολόγηση συστημάτων ασφαλείας. Αυτή η διαδικασία στοχεύει στον προσδιορισμό του βαθμού ομοιότητας μεταξύ διαφορετικών πολιτικών ασφαλείας, παρέχοντας έναν ποσοτικό δείκτη για το επίπεδο συμφωνίας μεταξύ τους.

Η διαδικασία υπολογισμού βασίζεται σε τρεις κατηγορίες κανόνων: ακριβείς αντιστοιχίσεις, κανόνες βασισμένους σε ομοιότητα με άλλους κανόνες και κανόνες χωρίς ομοιότητα. Κάθε κανόνας συνεισφέρει στο συνολικό ποσοστό ομοιότητας με ένα συγκεκριμένο βάρος.

Συγκεκριμένα, υπολογίζουμε τον βαθμό ομοιότητας μεταξύ των πολιτικών ασφαλείας, λαμβάνοντας υπόψιν τα βάρη των κανόνων που έχουμε θέσει σε αυτούς. Ας εξετάσουμε αναλυτικά τα βήματα του υπολογισμού του συνολικού ποσοστού ομοιότητας μεταξύ των πολιτικών:

- **Υπολογισμός συνολικού αριθμού κανόνων**: Ο συνολικός αριθμός των κανόνων λαμβάνεται ως το άθροισμα όλων των κανόνων και των δύο πολιτικών ασφαλείας.

- **Υπολογισμός Βάρους ομοιότητας για τους πανομοιότυπους κανόνες**: Για τους πανομοιότυπους κανόνες δηλαδή για τους κανόνες που βρίσκονται και στις δύο πολιτικές, θέτουμε βάρος του βαθμού ομοιότητας ως 1, διότι στην πλήρη ομοιότητα φτάνουμε στη μέγιστη τιμή του εύρους μας [0, 1].

- **Υπολογισμός βάρους ομοιότητας για τους κανόνες που δεν έχουν καμία αντιστοιχία με άλλους κανόνες**: Για τους κανόνες που δεν έχουν καμία αντιστοιχία με άλλους ανάμεσα στις δύο πολιτικές, το βάρος ορίζεται ως 0, διότι δεν υπάρχει ομοιότητα και φτάνουμε στην χαμηλότερη τιμή του εύρους μας [0,1].

- **Υπολογισμός βάρους για όσους κανόνες είναι όμοιοι:** Για τους κανόνες που έχουν ομοιότητα με άλλους κανόνες ανάμεσα στις δύο πολιτικές το βάρος καθορίζεται ανάλογα με τον υπολογισμό που μας έχει δώσει ο αλγόριθμος μετρικής ομοιότητας όπου βρίσκεται στο εύρος (0, 1).

- **Υπολογισμός συνολικού βάρους:** Προσθέτουμε όλα τα βάρη όλων των κανόνων για να προκύψει το συνολικό βάρος.

- **Υπολογισμός του συνολικού ποσοστού ομοιότητας**: Το συνολικό βάρος διαιρείται με τον συνολικό αριθμό των κανόνων και πολλαπλασιάζεται με το 100 για να παράξει το ποσοστό ομοιότητας.

5.3 Οπτικοποίηση Αποτελεσμάτων

Η οπτικοποίηση αποτελεσμάτων αποτελεί ένα σημαντικό στάδιο στην ανάλυση και κατανόηση των αποτελεσμάτων ενός συστήματος ή μιας διαδικασίας. Στην περίπτωση της σύγκρισης πολιτικών ασφαλείας και πιο συγκεκριμένα κανόνων Iptables, η οπτικοποίηση μπορεί να βοηθήσει τον χρήστη να αντλήσει εύκολα και γρήγορα συμπεράσματα από τα αποτελέσματα.

Κατά τη διάρκεια του τελευταίου σταδίου της ανάλυσης, προχωράμε στην οπτικοποίηση των αποτελεσμάτων. Στο πλαίσιο αυτό, εφαρμόζουμε δύο κύριες λειτουργίες οπτικοποίησης που συμβάλλουν στην καλύτερη αναπαράσταση των αποτελεσμάτων, οι οποίες περιλαμβάνονται παρακάτω:

Εμφάνιση αποτελεσμάτων στην κονσόλα: Χρησιμοποιούμε την εντολή print στη γλώσσα προγραμματισμού Python για να εμφανίσουμε κείμενο, αριθμούς ή άλλα δεδομένα στην οθόνη μας. Αυτή η μέθοδος καταλληλόλητας για απλές αναγνωστικές εκφράσεις αποτελεσμάτων είναι χρήσιμη κατά την ανάπτυξη και τον έλεγχο του κώδικα. Πιο αναλυτικά στην περίπτωση αυτή εμφανίζονται ως αποτελέσματα του κώδικα τα παρακάτω :

- **Εμφάνιση ακριβών αντιστοιχίσεων:** Εμφανίζει τους κανόνες που είναι ίδιοι και στις δύο πολιτικές βοηθώντας τον χρήστη να επιβεβαιώσει την ακριβή ταύτιση.
- **Εμφάνιση κανόνων με ομοιότητα :** Εμφανίζονται οι εντολές που έχουν ομοιότητα μεταξύ τους.
- **Εμφάνιση ποσοστού ομοιότητας :** Μαζί με τις εντολές που είναι όμοιες εμφανίζει και το ποσοστό ομοιότητας τους.
- **Παρουσίαση Διαφορών:** Εμφανίζει τις διαφορές μεταξύ των κανόνων που έχουν ομοιότητα, εστιάζοντας στις επιπρόσθετες λέξεις που μπορεί να υπάρχουν.
- **Παρουσίαση Κανόνων χωρίς αντιστοιχίσεις:** Εμφανίζει τους κανόνες που δεν έχουν καμία αντιστοίχιση ανάμεσα στις δύο πολιτικές δηλαδή είναι μοναδικοί.
- **Εμφάνιση συνολικού βάρους, συνολικού αριθμού κανόνων και ποσοστού ομοιότητας:** Εμφανίζει το συνολικό βάρος, τον συνολικό αριθμό των κανόνων και τέλος με την διαίρεση αυτών των δύο και τον πολλαπλασιασμό με το 100 εμφανίζεται το συνολικό ποσοστό ομοιότητας των δύο πολιτικών ασφαλείας.

Εμφάνιση αποτελεσμάτων μέσω ενός πλαισίου εργασίας (framework) για τη δημιουργία διαδραστικών εφαρμογών web : Το αποτέλεσμα είναι μια σελίδα που περιέχει τρία κύρια χαρακτηριστικά οπτικοποίησης που είναι τα εξής :

- **Pie Chart (Γράφημα Πίτας):** Παρουσιάζει το συνολικό ποσοστό ομοιότητας μεταξύ δύο πολιτικών ασφαλείας όπως και το ποσοστό που δεν είναι όμοιες. Επομένως έχουμε δύο μόνο κομμάτια στο pie chart. Τα ποσοστά εμφανίζονται μέσα στο γράφημα. Τα χρώματα των κομματιών του pie chart αντιπροσωπεύουν το ποσοστό ομοιότητας και το ποσοστό αντίθεσή τους.

- **Bar Chart (Γράφημα Κάθετων Μπάρων):** Παρουσιάζει το ποσοστό ομοιότητας μεταξύ των κανόνων που βρέθηκαν πως είναι όμοιοι ανάμεσα στις δύο πολιτικές. Κάθε μπάρα αντιπροσωπεύει έναν κανόνα, και το χρώμα αντιπροσωπεύει σε ποια πολιτική ασφαλείας ανήκει ο κάθε κανόνας. Μέσα στις μπάρες εμφανίζεται το ποσοστό ομοιότητας μεταξύ των κανόνων.
- **Πίνακες με Κανόνες:** Παρέχονται δύο πίνακες ξεχωριστά που περιλαμβάνουν τους κανόνες που δεν βρέθηκε κάποια ομοιότητα και τους κανόνες που βρέθηκαν να είναι πανομοιότυποι ανάμεσα στις δύο πολιτικές. Στον πίνακα με τους κανόνες που δεν είναι όμοιοι εμφανίζονται δύο στήλες δηλαδή εμφανίζονται ξεχωριστά οι κανόνες ανά πολιτική ασφαλείας.

Η οπτικοποίηση αποτελεί καθοριστικό εργαλείο στη διαδικασία ανάλυσης κανόνων iptables. Η σημασία της είναι πολύπλευρη και επηρεάζει θετικά την ερμηνεία των αποτελεσμάτων. Πιο συγκεκριμένα μας παρέχεται:

- **Ευκολία Κατανόησης:** Η οπτική αναπαράσταση παρέχει συνοπτική και ευανάγνωστη προβολή των πολύπλοκων δεδομένων. Ο χρήστης μπορεί να αποκτήσει πολύ πιο γρήγορη κατανόηση των διαφορών μεταξύ δύο πολιτικών ασφαλείας.
- **Αντιστοίχιση Πληροφοριών:** Η οπτική αναπαράσταση διευκολύνει την ταυτοποίηση πληροφοριών και την ανίχνευση κενών ασφαλείας. Ο χρήστης μπορεί να επικεντρωθεί σε συγκεκριμένα σημεία, βοηθώντας στην εύρεση πιθανών προβλημάτων ή ασυνήθιστων περιπτώσεων.
- **Ενίσχυση Αποτελεσματικότητας:** Η οπτική αναπαράσταση ενισχύει την αποτελεσματικότητα της ανάλυσης. Ο χρήστης μπορεί να εξερευνήσει τις ομοιότητες και τις διαφορές μεταξύ πολιτικών ασφαλείας, βοηθώντας στη λήψη ουσιαστικών αποφάσεων.

Συνολικά, η οπτικοποίηση αποτελεί ισχυρό σύμμαχο στην κατανόηση και αξιολόγηση της σύγκρισης κανόνων Iptables, προσφέροντας ένα ενιαίο πλαίσιο για την ερμηνεία των αποτελεσμάτων.

6 Υλοποίηση Σύγκρισης Πολιτικών Ασφαλείας

Στο παρόν κεφάλαιο εξηγούμε όλη την διαδικασία υλοποίησης της μεταφοράς των εν χρήση κανόνων, της μετατροπής της ιδανικής πολιτικής σε κανόνες προς επεξεργασία και τέλος την σύγκριση αυτών με βάση την μεθοδολογία μας.

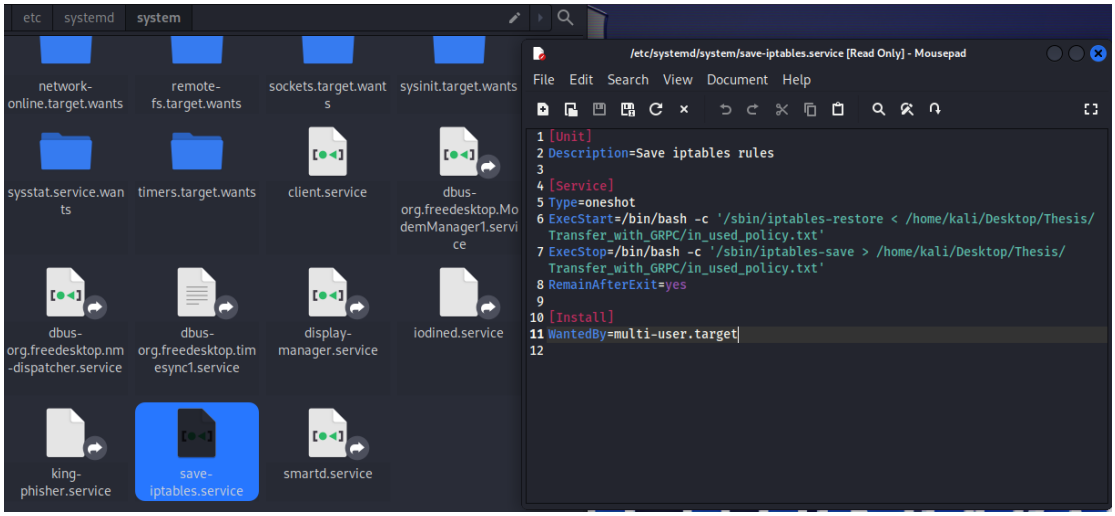
6.1 Αυτόματη Μεταφορά Αρχείου Κανόνων Iptables Από Linux Μηχάνημα σε Windows

Για την πιο σύντομη και λειτουργική επεξεργασία των κανόνων Iptables ή γενικότερα της πολιτικής ασφαλείας ενός Linux μηχανήματος προσπαθούμε να συλλέξουμε τους κανόνες με έναν πιο αυτοματοποιημένο τρόπο. Ο αυτοματοποιημένος τρόπος υλοποιείται μέσω τεχνολογίας gRPC (Google Remote Procedure Call) αλλά και μέσω της δημιουργίας δύο services στο Linux μηχάνημά μας.

6.1.1 Δημιουργία Service Για Αυτόματη Αποθήκευση Και Επαναφορά Πολιτικής Ασφάλειας

Όταν προβαίνουμε σε τροποποιήσεις στην πολιτική ασφάλειας ενός Linux μηχανήματος, είναι σημαντικό να διασφαλίσουμε ότι οι αλλαγές που έχουμε κάνει αποθηκεύονται προκειμένου να μην χαθούν σε περίπτωση επανεκκίνησης ή τερματισμού του συστήματος. Για να αποφύγουμε την επανάληψη της διαδικασίας αποθήκευσης και αποκατάστασης και να παρακάμψουμε πιθανά λάθη όπως την παράληψη της διαδικασίας αποθήκευσης, μπορούμε να δημιουργήσουμε μία υπηρεσία (service), στο Linux μηχάνημά μας, που θα εκτελεί αυτές τις ενέργειες αυτόματα.

Στην Εικόνα 11 φαίνεται η υπηρεσία μας, την οποία έχουμε αποθηκεύσει στον κατάλογο "system". Κατά την διαδικασία τερματισμού του μηχανήματος, εκτελείται η εντολή ExecStop που πραγματοποιεί την αποθήκευση (save) της πολιτικής σε ένα αρχείο μέσα σε έναν συγκεκριμένο φάκελο. Κατά την εκκίνηση του μηχανήματος, η εντολή στο ExecStart αποκαθιστά (restore) την αποθηκευμένη πολιτική από το συγκεκριμένο φάκελο. Με αυτό τον τρόπο, εξασφαλίζουμε ότι οι αλλαγές στην πολιτική ασφάλειας αποθηκεύονται και επαναφέρονται αυτόματα κάθε φορά που το μηχάνημα επανεκκινείται ή τερματίζεται, εξοικονομώντας χρόνο και αποφεύγοντας πιθανά λάθη.



Εικόνα 10 Service Αποθήκευσης Και Επαναφοράς Πολιτικής

6.1.2 Δημιουργία Εκτελέσιμων Αρχείων Για Ασφαλή Μεταφορά Αρχείου Μέσω GRPC

Για να μπορέσουμε να λάβουμε στον κεντρικό μας υπολογιστή το αρχείο με την πολιτική ασφαλείας από το Linux μηχάνημα δημιουργούμε εκτελέσιμα αρχεία με κώδικα python όπου χρησιμοποιούμε τεχνολογία gRPC.

Το σύστημα αποτελείται από έναν πελάτη(client) και έναν διακομιστή(server), οι οποίοι επικοινωνούν μεταξύ τους. Ο διακομιστής είναι το windows μηχάνημα ενώ ο πελάτης είναι το Linux μηχάνημα. Σε κάθε μηχάνημα υπάρχει και το αντίστοιχο εκτελέσιμο αρχείο για να πραγματοποιήσουμε την σύνδεση και την μεταφορά του αρχείου.

Η λειτουργία του διακοσμητή(server) μας είναι :

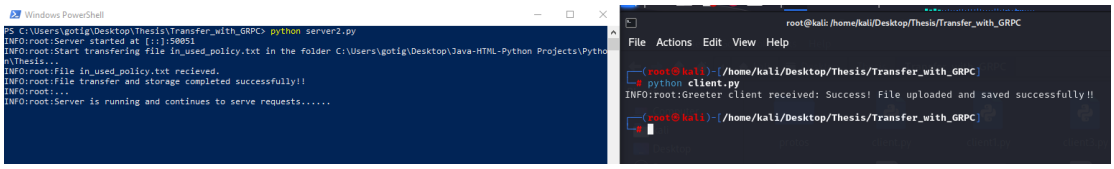
- **Επικοινωνία με τον Πελάτη:** Ο διακομιστής ανοίγει μια συγκεκριμένη θύρα (50051) για να λάβει αιτήματα από τον πελάτη και να απαντήσει σε αυτά.
- **Αποθήκευση Δεδομένων:** Λαμβάνει κομμάτια δεδομένων από τον πελάτη και τα αποθηκεύει σε ένα τοπικό αρχείο.

Η λειτουργία του πελάτη(client) μας είναι :

- **Δημιουργία Αιτήματος:** Ο πελάτης χρησιμοποιεί τη βιβλιοθήκη gRPC για να δημιουργήσει αιτήματα προς τον διακομιστή δηλαδή σε συγκεκριμένη IP του διακοσμητή και στην συγκεκριμένη θύρα(50051) που είναι πλέον ανοικτή.
- **Αποστολή Δεδομένων:** Ο πελάτης στέλνει κομμάτια δεδομένων στον διακομιστή για αποθήκευση.

Όλες οι λειτουργίες πραγματοποιούνται με δύο βασικές βιβλιοθήκες της python τις gRPC και Proto. Οι βιβλιοθήκες χρησιμοποιούνται για την υλοποίηση της επικοινωνίας RPC μεταξύ του διακομιστή και του πελάτη, με την βιβλιοθήκη Proto να προσφέρει μια γρήγορη και αποτελεσματική μέθοδο για την κωδικοποίηση και αποκωδικοποίηση των δεδομένων και το gRPC να διευκολύνει την ανταλλαγή αιτημάτων και αποκρίσεων μεταξύ τους μέσω ενός καναλιού(channel) που δημιουργεί.

Στην εικόνα 12 βλέπουμε την επικοινωνία και μεταφορά του αρχείου in_use_policy.txt από το linux μηχάνημα στον κεντρικό μας υπολογιστή.

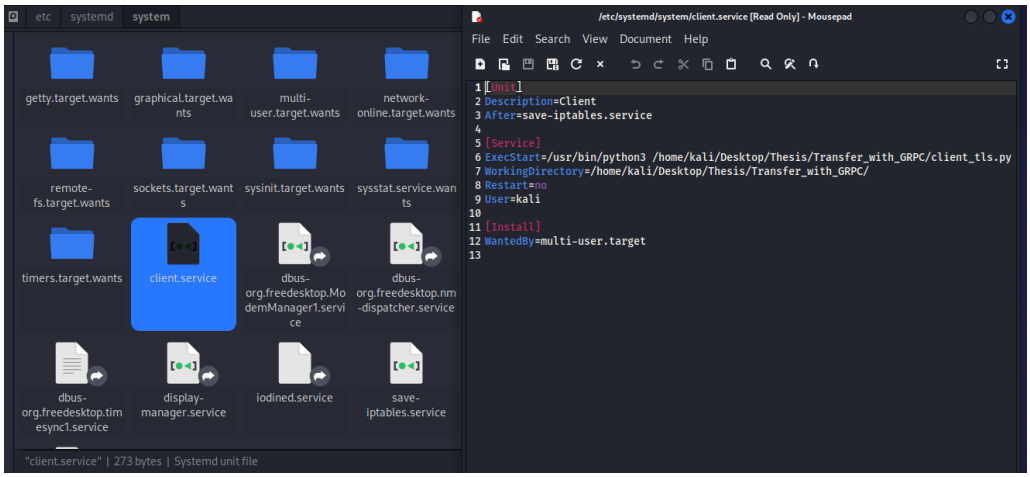


Εικόνα 11 Επικοινωνία και μεταφορά αρχείου

6.1.3 Δημιουργία Service Για Αυτόματη Μεταφορά Αρχείου Μετά Από Επανεκκίνηση Linux Μηχανήματος

Για να μπορέσουμε να πραγματοποιήσουμε όλη την διαδικασία αυτοματοποιημένα και πιο συγκεκριμένα την μεταφορά του αρχείου χωρίς να εκτελούμε εμείς το python script στον client θα δημιουργήσουμε ένα αρχείο service στο Linux μηχάνημα έτσι ώστε μετά από κάθε επανεκκίνηση ή έναρξη του μηχανήματος να εκτελείται το script και να στέλνεται αυτόματα η πολιτική. Επιπρόσθετα το service θα εκτελείται μετά το service που φτιάξαμε για αυτόματη αποθήκευση και μεταφορά δηλαδή μετά το save-iptables.service. Αυτό το βλέπουμε με την εντολή after = save-iptables.service.

Όπως βλέπουμε και στην εικόνα 13 στο exec start δηλαδή μετά από κάθε έναρξη του μηχανήματος έχουμε την εκτέλεση του python script του πελάτη (client) ενώ το service είναι αποθηκευμένο στο system.



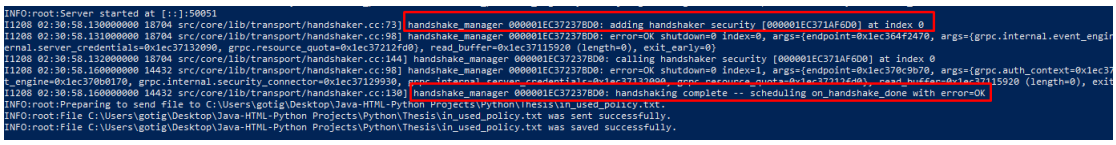
Εικόνα 12 Service Μεταφοράς Αρχείου

6.1.4 Προσθήκη Κρυπτογράφησης Και Τεχνολογίας TLS

Ένας σημαντικός τομέας που αξιοποιείται είναι η ασφάλεια κατά τη διαδικασία αποστολής αρχείων. Συγκεκριμένα, η ενσωμάτωση κρυπτογράφησης και της τεχνολογίας Transport Layer Security (TLS) αποτέλεσε κεντρικό στοιχείο για την εξασφάλιση της εμπιστευτικότητας και της ακεραιότητας των δεδομένων που μεταδίδονται[44].
Η διαδικασία ενσωμάτωσης της TLS περιλαμβάνει τα ακόλουθα βήματα:

- **Δημιουργία Κλειδιών και Πιστοποιητικών:** Πραγματοποιήθηκε η δημιουργία ενός ζεύγους κλειδιών (δημόσιο και ιδιωτικό) και ενός αυτο-υπογεγραμμένου πιστοποιητικού μέσω του ανοικτού κώδικα λογισμικού παραγωγής πιστοποιητικών openssl. Δημιουργείται κλειδί με βάση τον αλγόριθμο κρυπτογράφησης RSA με μέγεθος 4096 bits.
- **Ενσωμάτωση στον Server:** Ο Server ρυθμίστηκε να χρησιμοποιεί το ιδιωτικό κλειδί και το πιστοποιητικό για την εκκίνηση του TLS handshake.
- **Ρυθμίσεις στον Client:** Ο client ρυθμίστηκε να εμπιστεύεται το πιστοποιητικό του Server, διευκολύνοντας την επαλήθευση της ταυτότητας του server και τη δημιουργία ενός ασφαλούς καναλιού.

Στην εικόνα 14 με βάση το debugging στο κώδικα του GRPC βλέπουμε ότι το handshake προστέθηκε στην διαδικασία επικοινωνίας όπου η διαδικασία μεταφοράς ολοκληρώνεται επιτυχώς.



Εικόνα 13 Εμφάνιση handshake

6.2 Εντοπισμός Θυρών Και Υπηρεσιών Πολιτικής Ασφαλείας Linux Μηχανήματος.

Αφού παραλάβουμε την πολιτική ασφαλείας του Linux μηχανήματος ένα από τα πρώτα βήματα που κάνουμε είναι να αναλύσουμε σε ποιες θύρες επικεντρώνεται περισσότερο η πολιτική[45]. Η προσέγγιση που ακολουθήσαμε για να πραγματοποιήσουμε τον εντοπισμό των θυρών ήταν με τεχνολογία NLP και πιο συγκεκριμένα με tokens όπου στην συνέχεια με την βοήθεια άλλων βιβλιοθηκών εντοπίζουμε και τις ονομασίες των υπηρεσιών που υποστηρίζει η πολιτική.

6.2.1 Εντοπισμός Θυρών Με Τεχνολογία NLP Και Βιβλιοθήκη Spacy

Μια από τις βιβλιοθήκες ανάλυσης φυσικής γλώσσας είναι η Spacy όπου μας βοηθάει για το διαχωρισμό του κειμένου σε tokens παρέχοντας μια βάση για την αναγνώριση και εξαγωγή πληροφοριών από τους κανόνες. Αυτή η διαδικασία είναι βασική για την κατανόηση της δομής και του περιεχομένου των κανόνων. Στην συνέχεια με τη χρήση λέξεων-κλειδιών μπορούμε να αναγνωρίσουμε τις θύρες στους κανόνες ασφαλείας σε ένα κείμενο. Κάθε κανόνας συνήθως περιέχει σχετικές πληροφορίες, όπως το πρωτόκολλο και τη θύρα. Συγκρίνοντας τα tokens μέσω spacy και των λέξεων κλειδιών, όπως για παράδειγμα --dport, --sport ή -p όπου είναι μέρος των iptables κανόνων, εντοπίζουμε ακριβώς τις θύρες όπου αναφέρεται η πολιτική Ασφαλείας.

6.2.2 Εξαγωγή Πληροφοριών Θύρας Με Βιβλιοθήκη Socket

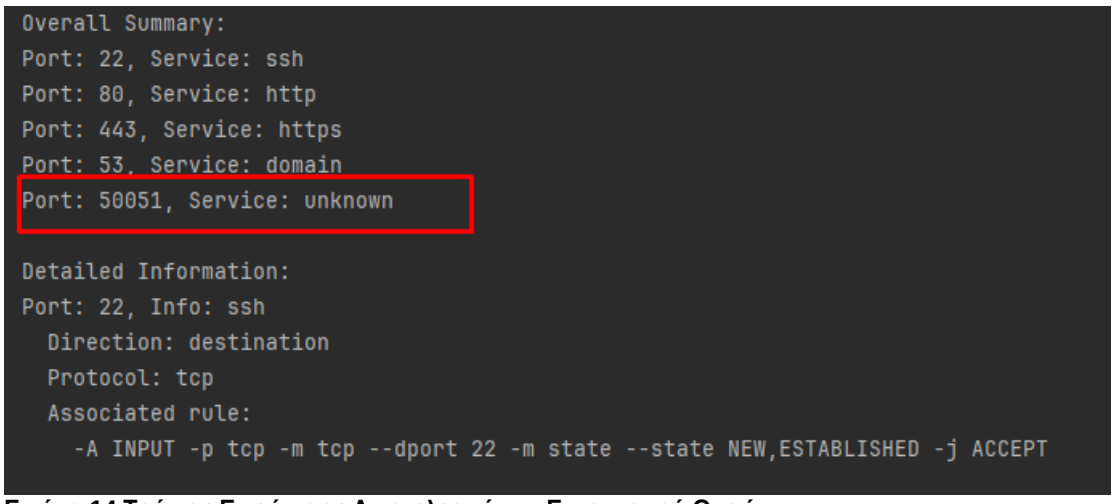
Η βιβλιοθήκη socket στην Python είναι ένα ισχυρό εργαλείο για τη δικτυακή επικοινωνία και παίζει έναν κεντρικό ρόλο στην επεξεργασία των κανόνων ασφαλείας σε ένα δίκτυο. Ένα από τα βασικά χαρακτηριστικά της είναι η δυνατότητα χρήσης της συνάρτησης getservbyport(port, protocol). Αυτή η συνάρτηση επιτρέπει τον εντοπισμό της αντιστοίχισης μιας θύρας (port) με την αντίστοιχη υπηρεσία που εκτελείται σε αυτήν. Για παράδειγμα, μπορούμε να χρησιμοποιήσουμε αυτή τη συνάρτηση για να εντοπίσουμε ότι η θύρα 80 συνήθως χρησιμοποιείται για HTTP επικοινωνία ή η θύρα 443 για HTTPS[46].

Η συνδυασμένη χρήση της βιβλιοθήκης socket με τις τεχνικές επεξεργασίας φυσικής γλώσσας (NLP) μας επιτρέπει να εμβαθύνουμε στην κατανόηση των κανόνων πρόσβασης, εντοπίζοντας και αναγνωρίζοντας αυτόματα τις συγκεκριμένες υπηρεσίες που εκτελούνται σε κάθε θύρα. Αυτό βοηθάει στη βελτιστοποίηση της ασφάλειας και της απόδοσης του δικτύου, προσφέροντας παράλληλα μια πιο διεξοδική κατανόηση της κυκλοφορούσας κίνησης και των αντίστοιχων κανόνων πρόσβασης. Επιπρόσθετα μας εντοπίζει θύρες στις οποίες ακόμα μπορεί να μην υπάρχουν υπηρεσίες σε αυτές με απώτερο σκοπό την εύρεση μιας ευπάθειας του συστήματος.

6.2.3 Τρόπος Εμφάνισης Αποτελεσμάτων

Όπως βλέπουμε και στην εικόνα 15, τα αποτελέσματα που εμφανίζονται παρέχουν πληροφορίες για τις συγκεκριμένες θύρες στο δίκτυο. Στην ενότητα "Overall Summary," παρουσιάζονται οι θύρες και οι αντίστοιχες υπηρεσίες που εκτελούνται σε αυτές. Στην ενότητα "Detailed Information," περιγράφονται περισσότερες λεπτομέρειες για κάθε θύρα, συμπεριλαμβανομένων του πρωτοκόλλου, της κατεύθυνσης (όπως η εισερχόμενη ή εξερχόμενη), και του συγκεκριμένου κανόνα πρόσβασης που εφαρμόζεται για την συγκεκριμένη θύρα. Επιπρόσθετα σε θύρες όπου δεν υποστηρίζεται κάποια υπηρεσία μας εμφανίζει την λέξη unknown.

Αυτή η αναλυτική προβολή των θυρών και των σχετικών υπηρεσιών παρέχει εμβάθυνση στην κατανόηση των τύπων κίνησης που επιτρέπονται, των πρωτοκόλλων αλλά και των υπηρεσιών που εκτελούνται σε κάθε συγκεκριμένη θύρα διευκολύνοντας την ανάλυση της υπάρχουσας πολιτικής ασφαλείας.



Εικόνα 14 Τρόπος Εμφάνισης Αποτελεσμάτων Εντοπισμού Θυρών

6.3 Υλοποίηση Μετατροπής Φυσικής Γλώσσας σε Κανόνες Iptables Με Τεχνολογία NLP

Αναπτύσσοντας ένα σύστημα μετατροπής φυσικής γλώσσας σε κανόνες iptables, χρησιμοποιούμε προηγμένες τεχνικές NLP (Natural Language Processing) για να αναλύσουμε και να επεξεργαστούμε την ιδανική πολιτική ασφαλείας που μας παρέχεται από την εταιρία σε φυσική γλώσσα. Αυτός ο μετασχηματισμός είναι κρίσιμος για την δημιουργία κανόνων Iptables που είναι κατανοητοί, εύκολα επεξεργάσιμοι και διαχειρίσιμοι.

Ο κώδικας που αναπτύσσουμε ενσωματώνει λειτουργίες από την βιβλιοθήκη SpaCy, μια κορυφαία εργαλειοθήκη για την επεξεργασία φυσικής γλώσσας. Η SpaCy χρησιμοποιείται για να επεξεργαστεί το κείμενο εισόδου, διαχωρίζοντάς το σε tokens, που αποτελούν τις βασικές μονάδες του κειμένου (λέξεις, σημεία στίξης κλπ.), επιτρέποντας περαιτέρω επεξεργασία.

Η WordNet, από την άλλη πλευρά, αποτελεί ένα εκτεταμένο λεξικό συνωνύμων και αντώνυμων της αγγλικής γλώσσας και είναι ιδιαίτερα χρήσιμη στην επεξεργασία φυσικής γλώσσας. Οργανώνει τις λέξεις σε σύνολα συνδεδεμένων λέξεων (synsets) με βάση την σημασιολογίας τους, επιτρέποντας την εξαγωγή συνωνύμων. Αυτό είναι ιδιαίτερα χρήσιμο για την αναγνώριση και ερμηνεία της ποικιλίας των εκφράσεων και των εννοιών που περιλαμβάνονται στην πολιτική ασφαλείας που παρέχεται από την εταιρία.

Εν κατακλείδι, συνδυάζοντας τις δυνατότητες των βιβλιοθηκών SpaCy και WordNet, αυτό το σύστημα NLP παρέχει ένα ισχυρό εργαλείο για την αυτόματη μετατροπή φυσικής γλώσσας σε κανόνες Iptables.
Πιο συγκεκριμένα ο κώδικας ακολουθεί τα εξής βήματα :

Βήμα 1 :Εισάγει τις απαραίτητες βιβλιοθήκες space και wordnet από nltk όπου όπως και η spacy έτσι και η nltk είναι βιβλιοθήκες αποκλειστικά για επεξεργασία φυσικής γλώσσας.

Βήμα 2 :Δέχεται φυσική γλώσσα ως είσοδο δηλαδή τους κανόνες που έχει θέσει η εταιρία ως ιδανική πολιτική ασφαλείας σε μορφή txt αρχείου.

Βήμα 3 :Δημιουργεί διαχωρισμό των λέξεων(tokenization): Η SpaCy διαχωρίζει το κείμενο σε λεκτικές μονάδες, γνωστές ως tokens. Αυτό το βήμα είναι σημαντικό για τον προσδιορισμό των λέξεων στο κείμενο.

Βήμα 4 :Για όλα τα tokens που έχει εξάγει ο κώδικας, με βάση την βιβλιοθήκη wordnet από nltk, δημιουργούμε όλα τα συνώνυμα που μπορεί να υπάρχουν.

Βήμα 5 :Δίνουμε λέξεις-κλειδιά που προσδιορίζουν όλες τις ενέργειες και τις προτεραιότητες στους κανόνες iptables. Οι λέξεις κλειδιά που ορίζουμε είναι οι παρακάτω :

- **"default" και "policy":** Χρησιμοποιούνται για να καθορίσουν την προεπιλεγμένη συμπεριφορά (αποδοχή ή απόρριψη) των πακέτων.
- **"allow", "drop":** Καθορίζουν την κατάσταση των πακέτων σε κάθε εντολή (αποδοχή ή απόρριψη).
- **"incoming", "outgoing", "inbound", "send", "forward":** Καθορίζουν την κατεύθυνση των πακέτων (εισερχόμενα, εξερχόμενα, εσωτερικά).
- **"interface":** Χρησιμοποιείται για τον καθορισμό της διεπαφής δικτύου από όπου έρχονται ή όπου πηγαίνουν τα πακέτα.
- **"port":** Ορίζει τη θύρα που πρέπει να χρησιμοποιηθεί για τη σύνδεση.

- **"tcp", "udp":** Καθορίζουν το πρωτόκολλο της σύνδεσης (TCP ή UDP).
- **"state, “new”, “established”, “related":** Ορίζει την κατάσταση της σύνδεσης.

Βήμα 6 :Εκτελούμε σύγκριση των λέξεων κλειδιών με όλα τα συνώνυμα των tokens.

- Εντοπίζουμε επιπρόσθετα εάν υπάρχει κάποια IP μέσα στην πρόταση που μας δίνεται με βάση ένα προκαθορισμένο μοτίβο αναζήτησης της διεύθυνσης. Το μοτίβο έχει την μορφή να υπάρχουν τέσσερις αριθμοί από ένα έως τρία ψηφία και ανάμεσά τους τελείες. Με βάση πάλι λέξεις κλειδιά όπως το **“to”** και το **“from”** εντοπίζουμε εάν έχουμε αποστολή ή λήψη πακέτων από την συγκεκριμένη διεύθυνση.
- Μετά την σύγκριση, ανάλογα με τις λέξεις-κλειδιά που αναγνωρίζονται, καθορίζονται οι ενέργειες που πρέπει να εφαρμοστούν, όπως η κατεύθυνση του κανόνα (INPUT, OUTPUT, FORWARD), η διεπαφή δικτύου (interface), οι θύρες, το πρωτόκολλο (TCP ή UDP) και η κατάσταση της σύνδεσης (state).
- Με βάση τις πληροφορίες που αναγνωρίζονται από τα tokens και τις συγκεκριμένες λέξεις-κλειδιά, δημιουργείται ο κανόνας Iptables με την αντίστοιχη σύνταξη, συμπεριλαμβάνοντας τις παραμέτρους που θέσαμε παραπάνω.

Για παράδειγμα, εάν δώσουμε ένα txt αρχείο με κανόνες σε φυσική γλώσσα θα μας δώσει το αποτέλεσμα της εικόνας 16.

```
Total number of generated iptables rules: 4. Here is the list of rules:

The default policy of incoming traffic is set to drop.
Allow incoming traffic on TCP port 80 (HTTP) from any source IP address.
Allow outgoing traffic on TCP port 443 (HTTPS) to any destination IP address.
Allow new and established incoming traffic on UDP port 53 (DNS) from any source IP address.

iptables -P INPUT DROP
iptables -A INPUT -p tcp -m tcp --dport 80 -j ACCEPT
iptables -A OUTPUT -p tcp -m tcp --sport 443 -j ACCEPT
iptables -A INPUT -p udp -m udp --dport 53 -m state --state NEW,ESTABLISHED -j ACCEPT

Iptables rules have been written to the file 'test/test1.txt'.
```

Εικόνα 15 Εμφάνιση Κανόνων iptables

6.4 Μετατροπή Κανόνων Σε Μορφή Iptables-Save Με Χρήση Εργαλείου iptables-Converter

Το επόμενο βήμα που θα ακολουθήσουμε είναι να μετατρέψουμε τους κανόνες αυτούς σε κατάλληλη μορφή για να είναι συμβατοί με τις εντολές iptables-restore και iptables-save των Linux συστημάτων.
Η διαδικασία αυτή απαιτεί προσοχή στη σύνταξη και τη δομή των κανόνων, καθώς και στην ακριβή τοποθέτησή τους μεταξύ των αντίστοιχων αλυσίδων.

6.4.1 Εργαλείο Iptables-Converter

Κύριο εργαλείο στη διαδικασία αυτή είναι το iptables-converter, ένα εργαλείο ανοικτού κώδικα δημιουργημένο από τον Johannes Hubertz [*https://github.com/sl0/conv#iptables-converter*]. Η κύρια λειτουργία του είναι η μετατροπή εντολών Iptables σε μορφή iptables-

save, επιταχύνοντας την επεξεργασία τους. Αυτό το εργαλείο είναι χρήσιμο για την αποδοτικότερη διαχείριση των κανόνων ασφαλείας δικτύου σε συστήματα που χρησιμοποιούν Iptables. Επίσης, διευκολύνει τη διαδικασία αποθήκευσης και επαναφόρτωσης των κανόνων, καθιστώντας την πιο απλή και λιγότερο επιρρεπή σε λάθη. Ακόμα είναι ιδιαίτερα χρήσιμο για διαχειριστές συστημάτων και ειδικούς ασφαλείας δικτύων που χρειάζονται να διαχειριστούν σύνθετες πολιτικές φιλτραρίσματος δικτύου.

Το iptables-converter περιλαμβάνει ένα ευρύ φάσμα χαρακτηριστικών όπως :

- Μετατροπή εντολών Iptables σε μορφή iptables-save.
- Υποστήριξη για τη μετατροπή ip6tables σε μορφή ip6tables-save.
- Δυνατότητα καθορισμού αρχείων πηγής και προορισμού.
- Λειτουργία sloppy για αυτόματους ορισμούς αλυσίδων χρηστών.
- Συμβατότητα με Python 2.7, 3.5, και 3.6.
- Διαθέσιμο ως Python module για εύκολη εισαγωγή.

Εκτός από τα προηγουμένως αναφερθέντα χαρακτηριστικά, το iptables-converter περιλαμβάνει επίσης τις παρακάτω δυνατότητες :

- Υποστήριξη για τα φίλτρα, mangle, nat, και raw tables
- Διεξαγωγή αυτόματων δοκιμών (tests) μέσω pytest
- Δημιουργία τεκμηρίωσης με χρήση του Sphinx
- Δημιουργία πακέτων Debian και RPM
- Τεστ αυτοματοποίησης στο travis-ci.org

6.4.2 Χρήση Και Κώδικας Εργαλείου Iptables-Converter

Ο ανοικτός κώδικας του εργαλείου χρησιμοποιεί δύο βασικές βιβλιοθήκες της Python: την collections και την re (regular expressions)[47]. Ας δούμε πώς βοηθούν αυτές οι βιβλιοθήκες στην υλοποίηση του κώδικα:

Χρήση βιβλιοθήκης re (Regular Expressions): Η βιβλιοθήκη re στην Python είναι ένα ισχυρό εργαλείο για την επεξεργασία κειμένου με regular expressions (regex), που επιτρέπει τον προγραμματιστή να αναγνωρίζει, να αναλύει και να επεξεργάζεται συγκεκριμένα μοτίβα κειμένου. Στο πλαίσιο της ανάλυσης και της επεξεργασίας των εντολών Iptables από ένα αρχείο κειμένου, η χρήση της βιβλιοθήκης re είναι ιδιαίτερα χρήσιμη.

Στον κώδικα, η βιβλιοθήκη re χρησιμοποιείται για να επιτύχει τα εξής:

- **Έλεγχος για Ειδικούς Χαρακτήρες**:
    - Χρησιμοποιείται η re.search για να ελέγξει αν μια γραμμή περιέχει ειδικούς χαρακτήρες όπως '$', '',",". Αυτό είναι σημαντικό για να εντοπιστεί αν η γραμμή περιέχει κάποιες συγκεκριμένες δομές που δεν είναι αποδεκτές ή απαιτούν διαφορετική επεξεργασία.
- **Αναγνώριση Εντολών iptables** :
    - Χρησιμοποιούνται τακτικές εκφράσεις για να διαπιστωθεί αν μια γραμμή αρχίζει με μια εντολή iptables, όπως '/sbin/iptables' ή απλά 'iptables'. Αυτό είναι κρίσιμο για την επεξεργασία της γραμμής ως κανόνα iptables.

Χρήση Βιβλιοθήκης collections: Η βιβλιοθήκη collections περιλαμβάνει υψηλού επιπέδου δομές δεδομένων που βοηθούν στην αποθήκευση και επεξεργασία δεδομένων.

Στον κώδικα, η βιβλιοθήκη collections χρησιμοποιείται για να επιτύχει τα εξής:

- **Δημιουργία Προσαρμοσμένων Λεξικών**:
  - Οι κλάσεις Chains και Tables κληρονομούν από την UserDict, κάτι που τους επιτρέπει να συμπεριφέρονται όπως λεξικά, με πρόσθετες προσαρμοσμένες λειτουργίες και ιδιότητες.
- **Επέκταση Λειτουργικότητας**:
  - Με την κληρονομικότητα της UserDict, οι κλάσεις Chains και Tables μπορούν να επεκτείνουν ή να τροποποιήσουν τις προεπιλεγμένες λειτουργίες ενός λεξικού για να προσαρμοστούν στις ανάγκες της εφαρμογής.
- **Προσθήκη Ειδικών Μεθόδων**:
  - Οι κλάσεις Chains και Tables περιλαμβάνουν ειδικές μεθόδους όπως put_into_fgr, reset, table_printout, put_into_tables και read_file, οι οποίες διαχειρίζονται ειδικά τους κανόνες iptables.

Ο κώδικας χρησιμοποιεί δύο βασικές κλάσεις: την Chains και την Tables. Ας εξετάσουμε πώς αυτές οι κλάσεις βοηθούν στην υλοποίηση του λογισμικού:

- **Κλάση Chains**:
  - **Αναπαράσταση Αλυσίδων**: Η κλάση Chains αναπαριστά μια αλυσίδα στο Iptables. Κάθε αλυσίδα έχει ένα όνομα και μια λίστα από κανόνες ασφαλείας. Αυτή η αναπαράσταση επιτρέπει την ευέλικτη αποθήκευση των κανόνων και την αναζήτηση αλυσίδων βάσει των ονομάτων τους.
  - **Διαχείριση Κανόνων**: Οι κανόνες ασφαλείας αποθηκεύονται ως στοιχεία στη λίστα της αλυσίδας. Αυτό επιτρέπει την εύκολη προσθήκη, αφαίρεση και τροποποίηση των κανόνων χωρίς την ανάγκη για πολύπλοκες διαδικασίες αναζήτησης.
  - **Αρχικοποίηση και Επαναφορά**: Η κλάση Chains παρέχει μεθόδους για την αρχικοποίηση των αλυσίδων σε μια καθορισμένη κατάσταση και την επαναφορά τους σε αυτή την αρχική κατάσταση όταν απαιτείται.
- **Κλάση Tables**:
  - **Οργάνωση Αλυσίδων σε Κατηγορίες**: Η κλάση Tables οργανώνει τις αλυσίδες ανά τύπο πίνακα (π.χ., filter, nat, mangle). Αυτή η οργάνωση διευκολύνει τη διαχείριση και την αναζήτηση αλυσίδων σε κάθε κατηγορία.
  - **Εξαγωγή Δεδομένων**: Η κλάση Tables παρέχει μεθόδους για την εξαγωγή των κανόνων από τις αλυσίδες και την αποθήκευσή τους σε ένα αρχείο κειμένου.
  - **Ευκολία Προσπέλασης και Αναζήτησης**: Η οργάνωση των αλυσίδων σε κατηγορίες διευκολύνει την προσπέλαση και την αναζήτηση των αλυσίδων ανάλογα με τις ανάγκες του χρήστη.

Οι παραπάνω κλάσεις βοηθούν στη διαχείριση και απεικόνιση των κανόνων ασφαλείας των iptables με μια δομή που είναι οργανωμένη, ευανάγνωστη και επεκτάσιμη. Πιο συγκεκριμένα η υλοποίηση ακολουθεί τα εξής βήματα :

Βήμα 1 :Ο κώδικας εκτελεί τις απαραίτητες ενέργειες για τη δημιουργία των προκαθορισμένων αλυσίδων και την τοποθέτηση των κανόνων ασφαλείας σε αυτές. Οι τέσσερις αλυσίδες που προστίθενται είναι με την σειρά raw, nat, mangle και filter όπου το τέλος της κάθε αλυσίδας δηλώνεται με την λέξη COMMIT.

Βήμα 2 :Διαβάζει ένα αρχείο κειμένου που περιέχει εντολές iptables. Κάθε γραμμή του αρχείου αναλύεται, και αναγνωρίζονται οι διάφορες εντολές, όπως προσθήκη, διαγραφή και αλλαγή κανόνων ασφαλείας. Ο κώδικας εξετάζει εάν υπάρχουν εντολές εισαγωγής,

διαγραφής, και αλλαγής πολιτικών, καθώς και τις ειδικές εντολές όπως η δημιουργία και η διαγραφή αλυσίδων.

Βήμα 3 :Αντιμετωπίζει πιθανά σφάλματα, όπως μη έγκυρες εντολές ή προσπάθεια αντιστοίχισης μη υποστηριζόμενων αλυσίδων. Αγνοεί τις γραμμές που περιέχουν σχόλια και αγνοεί τις γραμμές που περιέχουν εντολές διακοπής.

Βήμα 4 :Εντοπισμός παραμέτρων όπως -t (table). Όταν ο κώδικας συναντά για παράδειγμα την παράμετρο -t mangle, καταλαβαίνει ότι η εντολή αφορά την αλυσίδα "mangle". Στη συνέχεια, διαχειρίζεται την εντολή ανάλογα, επιλέγοντας το σωστό αντικείμενο Chains που αντιστοιχεί στην αλυσίδα "mangle". Το ίδιο συμβαίνει και για όλες τις αλυσίδες.

Βήμα 5 :Μετά την ανάγνωση του αρχείου, την τροποποίηση των εντολών και την τοποθέτησή τους στις κατάλληλες αλυσίδες, ο κώδικας παράγει ένα αρχείο εξόδου που περιλαμβάνει τις ανανεωμένες ρυθμίσεις ασφαλείας των δικτύων, ταξινομημένες σύμφωνα με τις προδιαγραφές του χρήστη.

Για παράδειγμα, εάν δώσουμε το παραγόμενο txt αρχείο με κανόνες σε Iptables θα μας δώσει το αποτέλεσμα της εικόνας 17.

```
The rules from test1.txt are:

iptables -P INPUT DROP
iptables -A INPUT -p tcp -m tcp --dport 80 -j ACCEPT
iptables -A OUTPUT -p tcp -m tcp --sport 443 -j ACCEPT
iptables -A INPUT -p udp -m udp --dport 53 -m state --state NEW,ESTABLISHED -j ACCEPT


The rules from test2.txt are:

*raw
:PREROUTING ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
COMMIT
*nat
:PREROUTING ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
:POSTROUTING ACCEPT [0:0]
COMMIT
*mangle
:PREROUTING ACCEPT [0:0]
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
:POSTROUTING ACCEPT [0:0]
COMMIT
*filter
:INPUT DROP [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
-A INPUT -p tcp -m tcp --dport 80 -j ACCEPT
-A INPUT -p udp -m udp --dport 53 -m state --state NEW,ESTABLISHED -j ACCEPT
-A OUTPUT -p tcp -m tcp --sport 443 -j ACCEPT
COMMIT
```

Εικόνα 16 Πολιτική ασφαλείας σε κατάλληλη μορφή

6.5 Υλοποίηση Σύγκριση Ιδανικής και Εν Χρήση Πολιτικής Ασφαλείας

Η τεχνητή νοημοσύνη (ΤΝ) και η επεξεργασία φυσικής γλώσσας (NLP) είναι πράγματι κρίσιμα εργαλεία στο σύγχρονο τεχνολογικό τοπίο, προσφέροντας πρωτοποριακές λύσεις σε πολλούς τομείς. Η εφαρμογή τους στην ανάλυση και την σύγκριση πολιτικών ασφαλείας είναι εξαιρετικά ενδιαφέρουσα.

Η χρήση αλγορίθμων ΤΝ και τεχνικών NLP για την ανίχνευση και ανάλυση των διαφορών μεταξύ ιδανικής και εν χρήση πολιτικής ασφαλείας μπορεί να αποκαλύψει σημαντικές πτυχές που ίσως διαφεύγουν από μια πιο συμβατική ανάλυση. Η ικανότητα της ΤΝ να αναλύει μεγάλες ποσότητες δεδομένων και να εντοπίζει μοτίβα και σχέσεις είναι ιδιαίτερα χρήσιμη σε αυτό το πλαίσιο.

Με την μετατροπή πλέον των κανόνων της ιδανικής πολιτικής σε πανομοιότυπη μορφή με την εν χρήση πολιτικής πλέον έχουμε την δυνατότητα να επεξεργαστούμε τις δύο πολιτικές με τον ίδιο τρόπο. Αυτό επιτρέπει την αποτελεσματική μετατροπή των κανόνων σε κατάλληλη μορφή για σύγκριση, την εφαρμογή αλγορίθμων ομοιότητας και την εμφάνιση των αποτελεσμάτων σύγκρισης και ποσοστού ομοιότητας των πολιτικών, εντοπίζοντας συγκεκριμένες διαφορές και ελλείψεις. Η διαδικασία αυτή αποτελεί ένα σύνθετο εγχείρημα όπου και θα αναλύσουμε στο παρόν κεφάλαιο.

6.5.1 Συλλογή δεδομένων

Η φάση της συλλογής δεδομένων περιλαμβάνει την επεξεργασία των αρχείων των πολιτικών ασφαλείας και την ανάλυση των κανόνων από αυτά.

Πιο συγκεκριμένα ακολουθούμε τα παρακάτω βήματα :

- **Ανάγνωση και Ανάλυση Αρχείων Iptables**: Ο κώδικας ανοίγει και διαβάζει το περιεχόμενο δύο διαφορετικών αρχείων Iptables. Για κάθε γραμμή στα αρχεία, αναλύει τα δεδομένα και αφαιρεί συγκεκριμένα στοιχεία (όπως αγκύλες).
- **Ανάλυση Δεδομένων**: Κάθε γραμμή εξετάζεται για να κατανοηθεί αν πρόκειται για την αρχή μιας αλυσίδας, έναν κανόνα ή το τέλος μιας αλυσίδας.
- **Διαχείριση Αλυσίδων iptables**: Κάθε αρχείο Iptables μπορεί να περιέχει πολλαπλές αλυσίδες (chains) κανόνων. Ο κώδικας αναγνωρίζει αυτές τις αλυσίδες και οργανώνει τους κανόνες ανά αλυσίδα.
- **Επιστροφή Δομημένων Δεδομένων**: Ο κώδικας επιστρέφει τα δεδομένα των κανόνων Iptables σε μορφή λεξικού για κάθε αρχείο, με τα κλειδιά του λεξικού να αντιπροσωπεύουν τις διάφορες αλυσίδες και τις τιμές να είναι λίστες των κανόνων Iptables. Πιο συγκεκριμένα, χρησιμοποιώντας την βιβλιοθήκη defaultdict, ο κώδικας δημιουργεί μια δομημένη αναπαράσταση των κανόνων, όπου κάθε αλυσίδα (π.χ., *filter) είναι ένα κλειδί στο λεξικό, και οι σχετικοί κανόνες αποτελούν τις τιμές σε μορφή λίστας.

Ένα παράδειγμα συλλογής και επεξεργασίας δεδομένων από ένα αρχείο πολιτικής ασφαλείας φαίνεται παρακάτω.

Δεδομένα αρχείου πριν την επεξεργασία :

```
*filter
-A INPUT -m conntrack --ctstate NEW -j ACCEPT
-A INPUT -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
COMMIT
```

Συλλογή Δεδομένων μετά την επεξεργασία :

```
{
   '*filter': [
      '-A INPUT -m conntrack --ctstate NEW -j ACCEPT',
      '-A INPUT -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT'
   ]
}
```

6.5.2 Εντοπισμός Πανομοιότυπων Εντολών

Στο πλαίσιο της ανάλυσης κανόνων ασφαλείας Iptables, εφαρμόζουμε αλγορίθμους με στόχο τον εντοπισμό πανομοιότυπων αντιστοιχιών ανάμεσα σε διαφορετικά σύνολα κανόνων. Αυτή η διαδικασία εστιάζει στην αναγνώριση ομοιοτήτων στις εντολές των κανόνων, παρέχοντας βαθύτερη κατανόηση των πολιτικών.

Πιο συγκεκριμένα για τον εντοπισμό των πανομοιότυπων κανόνων ακολουθούμε τα παρακάτω βήματα :

- **Μέθοδος Σύγκρισης Κανόνων :** Η διαδικασία ξεκινά με τη συνάρτηση compare_rules_exact, η οποία ελέγχει κάθε αλυσίδα κανόνων στο πρώτο σύνολο. Εάν μια αλυσίδα βρεθεί να υπάρχει και στο δεύτερο σύνολο, τότε πραγματοποιείται μια λεπτομερής σύγκριση των εντολών εντός της αλυσίδας. Κάθε εντολή εξετάζεται για την πανομοιότυπη ύπαρξή της στις αντίστοιχες αλυσίδες.
- **Καταγραφή Αποτελεσμάτων :** Τα αποτελέσματα της ανάλυσης διαχωρίζονται σε δύο κατηγορίες:
  - exact_matches: Λίστα που περιλαμβάνει τις εντολές με ακριβή αντιστοίχιση
  - non_matches: Λίστα με τις εντολές που δεν έχουν ακριβή αντιστοίχιση

Η επιτυχής αναγνώριση ακριβών αντιστοιχιών είναι κρίσιμη για την ακριβή εκτίμηση της συνοχής και της ασφάλειας της υπάρχουσας πολιτικής. Τα ευρήματα αυτής της διαδικασίας αξιοποιούνται σε μελλοντικές φάσεις για καταγραφή αριθμού πανομοιότυπων κανόνων και περεταίρω επεξεργασία των λοιπών κανόνων. Επιπρόσθετα, η υλοποίηση της διαδικασίας είναι τόσο απλή όσο και αποτελεσματική, χρησιμοποιώντας βασικούς βρόχους επανάληψης για τη σύγκριση των αλυσίδων και των εντολών. Αν και δεν απαιτείται η χρήση περίπλοκων ή εξειδικευμένων βιβλιοθηκών για αυτήν τη φάση, η ακριβής αντιστοίχιση των εντολών αποτελεί ένα θεμελιώδες βήμα στην προετοιμασία για μια συνολική και ακριβή ανάλυση της πολιτικής ασφαλείας.

6.5.3 Δημιουργία Δομών Δεδομένων

Αφού πλέον έχουμε διαχωρίσει τις εντολές που είναι πανομοιότυπες πλέον κρατάμε τις υπόλοιπες εντολές ξεχωριστά από τα δύο αρχεία όπου και θα τις επεξεργαστούμε περαιτέρω. Στο επόμενο στάδιο θα προετοιμάσουμε τις δομές δεδομένων των υπό επεξεργασία εντολών έτσι ώστε με βάση αυτές τις δομές να μπορέσουμε να περάσουμε στο επόμενο στάδιο που θα χρησιμοποιηθούν για την δημιουργία των embeddings, κρατώντας παράλληλα πληροφορίες κανόνων που ενδέχεται να μην έχουν αντιστοίχιση.
Για την δημιουργία των δομών δεδομένων μας ακολουθούμε τα εξής βήματα :

Βήμα 1 - Επανάληψη σε Κανόνες: Ο κώδικας αρχίζει με την επανάληψη μέσω μιας συλλογής από κανόνες από το αρχείο της ιδανικής πολιτικής ασφαλείας. Κάθε κανόνας εξετάζεται για την εξαγωγή διαφόρων ιδιοτήτων:

- port: Προσδιορίζεται η θύρα προορισμού (dport) ή πηγής (sport).
- direction: Καθορίζεται η κατεύθυνση του κανόνα.
- protocol: Αναγνωρίζεται το πρωτόκολλο του κανόνα.

- action: Ορίζεται η ενέργεια του κανόνα.
- source_ip, dest_ip: Εξάγονται οι διευθύνσεις IP πηγής και προορισμού.
- icmp_type: Αν το πρωτόκολλο είναι ICMP, προσδιορίζεται ο τύπος του ICMP.

Βήμα 2 - Έλεγχος Συνθηκών: Αν υπάρχουν οι ιδιότητες port, direction και action, τότε δημιουργείται ένα "κλειδί κανόνα" από αυτές τις τιμές.

Βήμα 3 - Σύγκριση με Άλλους Κανόνες: Ακολουθεί μια δεύτερη επανάληψη σε ένα διαφορετικό σύνολο κανόνων. Δηλαδή κανόνες από την εν χρήση πολιτική ασφαλείας. Για κάθε κανόνα, εκτελείται μια παρόμοια διαδικασία εξαγωγής ιδιοτήτων και δημιουργίας ενός δεύτερου "κλειδιού κανόνα".

Βήμα 4 - Εύρεση Ομοιότητας: Ελέγχεται αν το κλειδί κανόνα από τον πρώτο κανόνα υπάρχει στο σύνολο των κλειδιών από τον δεύτερο κανόνα. Αν ναι, τότε καθορίζεται ότι βρέθηκε ένας παρόμοιος κανόνας.

Βήμα 5 - Επεξεργασία Παρόμοιων Κανόνων: Αν βρεθεί παρόμοιος κανόνας, συγκεντρώνονται όλα τα στοιχεία από τους σχετικούς κανόνες και επεξεργάζονται περεταίρω.

6.5.4 Ανάλυση Δεδομένων Μέσω Embeddings

Σε αυτό το στάδιο, αφού έχουμε εντοπίσει όμοιους κανόνες μέσω των κλειδιών, εφαρμόζουμε το μοντέλο Universal Sentence Encoder (USE) για τη μετατροπή των εντολών σε αναπαραστάσεις, προετοιμάζοντας τα δεδομένα για την επόμενη φάση, η οποία αφορά τον υπολογισμό της ομοιότητας μεταξύ των κανόνων.

Πιο συγκεκριμένα, χρησιμοποιούμε την βιβλιοθήκη TensorFlow Hub όπου μας παρέχει προ-εκπαιδευμένα μοντέλα που μπορούν να χρησιμοποιηθούν για διάφορες εφαρμογές, όπως η αναπαράσταση κειμένου. Τα μοντέλα αυτά έχουν συνήθως εκπαιδευτεί σε μεγάλα σύνολα δεδομένων κειμένου για να μάθουν να αναπαριστούν αποτελεσματικά τη σημασία των λέξεων και των προτάσεων[48].

Τα embeddings που παράγονται από το USE είναι διανυσματικές αναπαραστάσεις χαρακτηριστικών με υψηλή διάσταση. Η αρχιτεκτονική του USE, συγκεκριμένα, χρησιμοποιεί ένα νευρωνικό δίκτυο που μπορεί να αντιμετωπίσει προτάσεις ή κείμενο σε ένα ευρύ φάσμα εφαρμογών.

Στο στάδιο της "Δημιουργίας Αναπαραστάσεων με το Universal Sentence Encoder (USE)", ακολουθούνται τα εξής βήματα :

- **Εισαγωγή του Universal Sentence Encoder (USE):** Το USE φορτώνεται χρησιμοποιώντας το κατάλληλο URL του μοντέλου ("https://tfhub.dev/google/universal-sentence-encoder/4"). Αυτό εγγυάται ότι το μοντέλο είναι διαθέσιμο για χρήση. Το URL αναφέρεται στο TensorFlow Hub (tfhub.dev), μια πλατφόρμα που παρέχει προ-εκπαιδευμένα μοντέλα μηχανικής μάθησης που μπορούν να χρησιμοποιηθούν εύκολα και αποτελεσματικά σε διάφορα έργα. Το συγκεκριμένο URL αναφέρεται στην τέταρτη έκδοση του Universal Sentence Encoder (USE), ένα προηγμένο μοντέλο που έχει αναπτυχθεί από την Google Research. Η αναφορά στο "4" στο τέλος του URL υποδηλώνει την έκδοση του μοντέλου. Κάθε νέα έκδοση μπορεί να περιλαμβάνει βελτιώσεις, επιδιορθώσεις ή επεκτάσεις σε σχέση με τις προηγούμενες.

- **Χρήση του USE για Δημιουργία Αναπαραστάσεων:** Για κάθε πρόταση ή κείμενο που πρέπει να αναπαρασταθεί, το USE εφαρμόζεται και παράγει ένα embedding, που ουσιαστικά είναι μια αναπαράσταση με χαρακτηριστικά της σημασιολογίας του

κειμένου. Πιο συγκεκριμένα δημιουργούμε έναν μονοδιάστατος πίνακας με 512 στοιχεία και κάθε στοιχείο είναι ένας αριθμός τύπου float32.

- **Αποθήκευση των Αναπαραστάσεων:** Τα αποτελέσματα των αναπαραστάσεων αποθηκεύονται για χρήση. Αυτά τα embeddings χρησιμοποιούνται στη συνέχεια για τη σύγκριση, ανάλυση και αξιολόγηση των κανόνων.

Συνοψίζοντας, η διαδικασία δημιουργίας αναπαραστάσεων με το Universal Sentence Encoder συμπεριλαμβάνει τη φόρτωση του μοντέλου, την προεπεξεργασία των δεδομένων και τη χρήση του μοντέλου για τη δημιουργία σημασιολογικών αναπαραστάσεων. Αυτές οι αναπαραστάσεις στη συνέχεια χρησιμοποιούνται για σύγκριση των κανόνων.

6.5.5 Εφαρμογή Αλγορίθμου Ομοιότητας Cosine Similarity

Η εφαρμογή αλγορίθμου ομοιότητας της cosine similarity προορίζεται για τον υπολογισμό της ομοιότητας μεταξύ κανόνων που έχουν τροποποιηθεί σε embeddings στο σύστημα μας. Στην εφαρμογή αυτή, ο στόχος είναι να αντιστοιχίσουμε κανόνες μεταξύ των διαφορετικών πολιτικών, προκειμένου να εντοπίσουμε σε τι ποσοστό είναι όμοιοι.

Ας δούμε τα βήματα που ακολουθούνται:

Βήμα 1 :Φόρτωση Βιβλιοθήκης torch.nn.fuctional:

Πρόκειται για μέρος του πακέτου PyTorch, που είναι μια βιβλιοθήκη για βαθιά μηχανική μάθηση. Η υποβιβλιοθήκη torch.nn.functional περιλαμβάνει συναρτήσεις χαμηλού επιπέδου για την εκτέλεση διαφόρων λειτουργιών, συμπεριλαμβανομένης της υπολογιστικής ομοιότητας (όπως η cosine_similarity). Χρησιμοποιείται εδώ για την υλοποίηση της συνάρτησης ομοιότητας Cosine Similarity[49].

Βήμα 2 :Σύγκριση των Αναπαραστάσεων για τον Εντοπισμό Ομοιοτήτων**:**

- **Χρήση βιβλιοθήκης torch.from_numpy :** Εδώ παίρνουμε την πρώτη αναπαράσταση από τον πίνακα embeddings, τη μετατρέπουμε σε έναν tensor του PyTorch. Ένα tensor είναι βασικά ένας πολυδιάστατος πίνακας. Η χρήση τους είναι συνηθισμένη σε εφαρμογές μηχανικής μάθησης και βαθιάς μάθησης, καθώς διευκολύνουν τους υπολογισμούς και την αποδοτική χρήση των δεδομένων. Προσθέτουμε μια επιπλέον διάσταση στον tensor, το οποίο κάνει τα διανύσματα να γίνονται δισδιάστατα tensors με μία γραμμή και πολλές στήλες, αντί για απλά διανύσματα. Αυτό συχνά είναι απαραίτητο για την σωστή λειτουργία των συναρτήσεων σε βιβλιοθήκες όπως η PyTorch.

- **Χρήση βιβλιοθήκης cosine_similarity:** Η συνάρτηση cosine_similarity υπολογίζει την ομοιότητας μεταξύ δύο tensors με βάση την μετρική cosine. Στη συγκεκριμένη περίπτωση, υπολογίζεται η ομοιότητα μεταξύ της πρώτης και της δεύτερης αναπαράστασης που προσαρτήθηκαν στην επιπλέον διάσταση.

Βήμα 3 :Καταγραφή των Αποτελεσμάτων:

Η καταγραφή των αποτελεσμάτων στον κώδικα πραγματοποιείται με τη χρήση δύο κυρίων δομών δεδομένων, την λίστα similarities και το λεξικό rule_based_similarity.

Ας αναλύσουμε τη διαδικασία βήμα προς βήμα:

- **Δημιουργία Λίστας Ομοιοτήτων:** Η λίστα similarities δημιουργείται για κάθε κανόνα και περιλαμβάνει τις τιμές ομοιότητας μεταξύ δύο συγκεκριμένων κανόνων. Δηλαδή έχουν γίνει συγκρίσεις μεταξύ των embenddings και η τιμή της σύγκρισης αποθηκεύεται εκεί.

- **Δημιουργία Λεξικού Κανόνων Βασισμένου στην Ομοιότητα:** Το λεξικό rule_based_similarity αντιστοιχεί σε κάθε κανόνα της λίστας των όμοιων κανόνων. Δηλαδή περιέχει όλες τις πληροφορίες των όμοιων κανόνων.

- **Έλεγχος Ανύπαρκτης Ομοιότητας:** Καθώς όλοι οι κανόνες περνάνε από επεξεργασία πραγματοποιείται έλεγχος για να διαπιστωθεί αν βρέθηκε ομοιότητα για τον συγκεκριμένο κανόνα. Δηλαδή εάν έχει βρεθεί οποιαδήποτε ομοιότητα με οποιονδήποτε κανόνα από την άλλη πολιτική ασφαλείας. Αν δεν βρέθηκε, τότε ο κανόνας προστίθεται στη λίστα no_match_rules δηλαδή ότι ο κανόνας δεν έχει κάποιον άλλο όμοιο κανόνα.

6.5.6 Οπτικοποίηση Αποτελεσμάτων

Σε αυτήν την ενότητα παρουσιάζεται η υλοποίηση της οπτικοποίησης των αποτελεσμάτων στο πλαίσιο της σύγκρισης πολιτικών ασφαλείας. Η οπτικοποίηση περιλαμβάνει ποικίλα γραφήματα και πίνακες για την καλύτερη κατανόηση των αποτελεσμάτων.

Παρουσίαση αποτελεσμάτων στην Κονσόλα

Η εκτύπωση αποτελεσμάτων διαδραματίζει κρίσιμο ρόλο στη διαδικασία ανάλυσης και παρουσίασης των διαφορετικών στοιχείων μεταξύ δύο πολιτικών ασφαλείας. Ο τρόπος που εκτυπώνονται τα αποτελέσματα παρέχει στον χρήστη μια σαφή και κατανοητή εικόνα των διαφορών και των ομοιοτήτων μεταξύ των πολιτικών.

Αφού έχουμε καταγράψει σε ξεχωριστές λίστες όλα τα αποτελέσματα που θέλουμε, πλέον με μια προσπέλαση της κάθε λίστας μπορούμε να εκτυπώσουμε τα αποτελέσματα.
Η εκτύπωση περιλαμβάνει τέσσερα κυρίως τμήματα:

- **Πανομοιότυποι Κανόνες (Exact Matches):** Εδώ παρουσιάζονται οι κανόνες που ταιριάζουν ακριβώς μεταξύ των δύο πολιτικών. Αυτοί οι κανόνες είναι πανομοιότυποι και δεν υπάρχει καμία διαφορά μεταξύ τους. Βλέπουμε τους κανόνες στην εικόνα 18 όπως εμφανίζονται.

```
------ Exact Matches ------
":INPUT DROP "
"-A INPUT -p tcp -m tcp --dport 21 -m state --state NEW,ESTABLISHED -j ACCEPT"
"-A INPUT -p udp -m udp --dport 53 -m state --state NEW,ESTABLISHED -j ACCEPT"
```

Εικόνα 17 Εμφάνιση Πανομοιότυπων Κανόνων

- **Ομοιότητες (Similarities):** Παρουσιάζονται πληροφορίες για τους κανόνες που έχουν ομοιότητες μεταξύ τους. Με βάση τις λίστες παρουσιάζονται οι δύο κανόνες από τις ξεχωριστές πολιτικές μαζί με το ποσοστό ομοιότητας που έχει βρεθεί. Όπως βλέπουμε και στην εικόνα 19 πρώτα εμφανίζεται ο κανόνας της Ιδανικής πολιτικής και μετά ο κανόνας της εν χρήση πολιτικής. Επιπρόσθετα εμφανίζονται οι διαφορές

αυτών των δύο κανόνων δηλαδή ποιες λέξεις υπάρχουν μόνο στον κανόνα της ιδανικής πολιτικής και ποιες λέξεις μόνο στον κανόνα της εν χρήση πολιτικής.

```
------ Similarities ------

Command in Ideal Policy: "-A INPUT -p tcp -m tcp --dport 80 -j ACCEPT"
Command in Used Policy : "-A INPUT -p udp -m udp --dport 80 -j ACCEPT"
Similarity      : 83.82%
Differences     :
  Unique words from command in Ideal Policy:
    - tcp
  Unique words from command in Used Policy:
    - udp
```

Εικόνα 18 Εμφάνιση Ομοιοτήτων Κανόνων

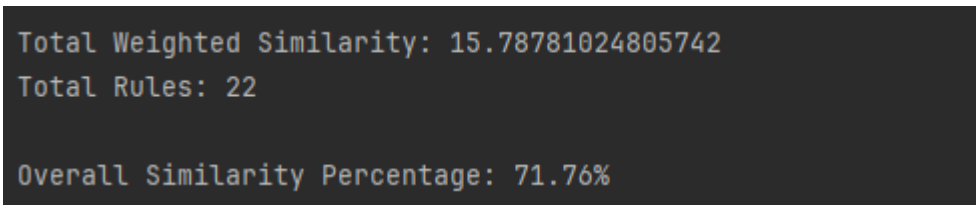
- **Κανόνες που δεν βρέθηκε καμία ομοιότητα μεταξύ των δύο πολιτικών (No Matches):** Εμφανίζονται οι κανόνες που δεν έχουν καμία ομοιότητα με άλλους κανόνες και παρουσιάζονται ξεχωριστά για τις δύο πολιτικές για να έχουμε καλύτερη ανάλυση αποτελεσμάτων. Όπως βλέπουμε και στην εικόνα 20 έχουμε δύο λίστες από κανόνες με πρώτη λίστα όσους κανόνες βρίσκονται στην ιδανική πολιτική και δεύτερη λίστα όσους κανόνες βρίσκονται στην εν χρήση πολιτική.

```
------ No Matches Found from Ideal Policy ------
":OUTPUT DROP "
"-A INPUT -p tcp -m tcp --dport 443 -j ACCEPT"
"-A INPUT -p tcp -m tcp --dport 21 -m state --state NEW,ESTABLISHED -j ACCEPT"
"-A OUTPUT -p tcp -m tcp --sport 21 -m state --state ESTABLISHED -j ACCEPT"

------ No Matches Found from the In-used Policy ------
":OUTPUT ACCEPT "
"-A INPUT -p tcp -m tcp --dport 3306 -j ACCEPT"
```

Εικόνα 19 Κανόνες που δεν βρέθηκαν ομοιότητες με άλλους κανόνες

- **Συνολική Ομοιότητα και Στατιστικά:** Με βάση τους υπολογισμούς που έχουν γίνει για το συνολικό βάρος και τους συνολικούς κανόνες που έχουμε, εμφανίζουμε στο τέλος όλα τα δεδομένα μας για να έχουμε και μια πλήρη απεικόνιση των αποτελεσμάτων. Όπως βλέπουμε και στην εικόνα 21 εμφανίζεται το συνολικό ποσοστό ομοιότητας μεταξύ των δύο πολιτικών ασφαλείας, καθώς και άλλα στατιστικά στοιχεία όπως το συνολικό βάρος της ομοιότητας και ο συνολικός αριθμός των κανόνων και των δύο πολιτικών.



Εικόνα 20 Εμφάνιση τελικών στατιστικών στοιχείων συνολικής ομοιότητας

Παρουσίαση αποτελεσμάτων με την Εφαρμογή Dash

Η εφαρμογή Dash από την Plotly είναι ένα πλαίσιο ανάπτυξης ιστοσελίδων στην Python, ιδανικό για τη δημιουργία διαδραστικών πίνακων ελέγχου (dashboards) και εφαρμογών αναλυτικής οπτικοποίησης.
Πιο συγκεκριμένα, η υλοποίηση που δίνεται αφορά τις παρακάτω βιβλιοθήκες :

- **Dash :** Η Dash είναι μια Python βιβλιοθήκη για τη δημιουργία διαδραστικών εφαρμογών Web προκειμένου να παρουσιάσει και να αναλύσει αποτελέσματα σύγκρισης μεταξύ δύο πολιτικών ασφαλείας. Η εφαρμογή εκτελείται τοπικά και παρέχει ένα περιβάλλον οπτικοποίησης που περιλαμβάνει γραφήματα και πίνακες για την καλύτερη κατανόηση των διαφορών μεταξύ των πολιτικών. Στη συγκεκριμένη υλοποίηση, η Dash χρησιμοποιείται για τη δημιουργία ενός web server που φιλοξενεί την οπτική αναπαράσταση των αποτελεσμάτων σύγκρισης πολιτικών ασφαλείας. Η εφαρμογή παρέχει γραφήματα πίτας, γραφικές αναπαραστάσεις ποσοστιαίας ομοιότητας, και πίνακες με ακριβείς αντιστοιχίες κανόνων. Όταν ο χρήστης εκτελεί τον κώδικα, η Dash δημιουργεί έναν ενσωματωμένο web server που ακούει σε μια συγκεκριμένη θύρα (στην περίπτωση αυτή η θύρα είναι η 8050). Στη συνέχεια, η εφαρμογή ανοίγεται σε ένα προεπιλεγμένο Web browser (στην περίπτωσή μας επιλέξαμε το firefox). Ακόμα, ο χρήστης μπορεί να αποκτήσει πρόσβαση στην εφαρμογή πλοηγούμενος στη διεύθυνση http://127.0.0.1:8050/ από τον τοπικό του υπολογιστή. Και τέλος, ο server συνδέεται με την εφαρμογή και παρέχει τα αποτελέσματα της σύγκρισης πολιτικών ασφαλείας μέσω της Dash, επιτρέποντας στον χρήστη να αλληλοεπιδρά με τα γραφήματα και τους πίνακες για την καλύτερη κατανόηση των δεδομένων[50].

- **Plotly (plotly.graph_objects):** Η Plotly είναι μια βιβλιοθήκη γραφικών για τη δημιουργία διαδραστικών γραφημάτων στην Python. Χρησιμοποιείται ευρέως σε πολλά πεδία, συμπεριλαμβανομένης της επιστήμης δεδομένων, της ανάλυσης δεδομένων και της αναπαράστασης δεδομένων. Στην περίπτωση της Dash, η Plotly χρησιμοποιείται για τη δημιουργία των γραφημάτων και των πινάκων που εμφανίζονται στην εφαρμογή. Ένα πλεονέκτημα της Plotly είναι η εύκολη δυνατότητα ενσωμάτωσης διαδραστικών στοιχείων στα γραφήματα, επιτρέποντας στους χρήστες να αλληλοεπιδρούν με τα δεδομένα τους. Συνδυαζόμενη με τη Dash, η Plotly επιτρέπει τη δημιουργία εξαιρετικά εντυπωσιακών και προσαρμόσιμων διαδραστικών εφαρμογών αναπαράστασης δεδομένων. Η χρήση της Plotly μαζί με τη Dash στον κώδικα μας, επιτρέπει τη δημιουργία ενός γραφήματος πίτας, ενός διαγράμματος μπάρας, και δύο πινάκων. Επίσης, η Plotly μας δίνει τη δυνατότητα να προσαρμόσουμε λεπτομερώς την εμφάνιση των γραφημάτων και των πινάκων, όπως φαίνεται από τις διάφορες παραμετροποιήσεις (π.χ., χρώματα, μέγεθος γραμματοσειράς, θέση ετικέτας).

- **Pandas (pandas):** Η Pandas είναι μια ισχυρή βιβλιοθήκη για τη διαχείριση και ανάλυση δεδομένων. Συνεργάζεται καλά με την Plotly (που χρησιμοποιείται στο Dash) για τη δημιουργία γραφημάτων και διαγραμμάτων. Τα DataFrame της Pandas

μπορούν να περαστούν απευθείας στα αντικείμενα γραφημάτων της Plotly, καθιστώντας ευκολότερη τη δημιουργία διαδραστικών γραφημάτων στο Dash[51].

- **Webbrowser (webbrowser):** Η βιβλιοθήκη webbrowser στην Python χρησιμοποιείται για το αυτόματο άνοιγμα του προεπιλεγμένου προγράμματος περιήγησης του υπολογιστή μας. Παρέχει μια απλή διεπαφή για τον έλεγχο του προγράμματος περιήγησης για το άνοιγμα συγκεκριμένων URL. Χρησιμοποιώντας τη webbrowser, μπορούμε να ενσωματώσουμε την εμφάνιση της εφαρμογής Dash σε ένα παράθυρο περιήγησης, προκειμένου να παρουσιάσουμε τα αποτελέσματα της ανάλυσης ασφάλειας πολιτικών.

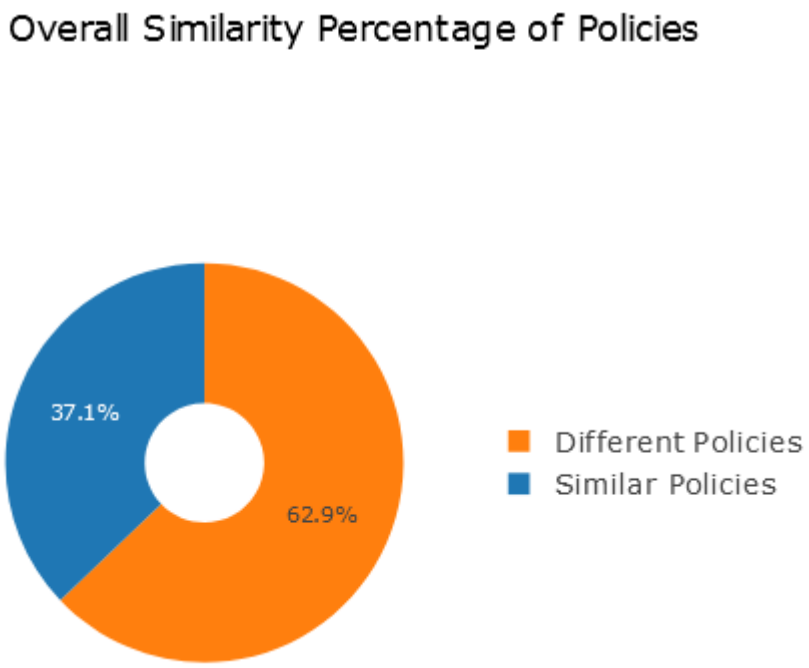
Σε αυτό το πλαίσιο, εκμεταλλευόμαστε διάφορα εργαλεία, όπως διαγράμματα πίτας, διαγράμματα ραβδών, πίνακες ομοιότητας και πίνακες διαφορών, προκειμένου να παρουσιάσουμε και να αναλύσουμε τα δεδομένα μας. Αυτά τα εργαλεία διευκολύνουν την κατανόηση των ομοιοτήτων και διαφορών μεταξύ των πολιτικών ασφαλείας, επιτρέποντας στους χρήστες να παρακολουθούν αποδοτικά τα αποτελέσματα της ανάλυσης.

Ας εξετάσουμε πιο αναλυτικά τα διάφορα είδη διαγραμμάτων και πινάκων που χρησιμοποιούμε:

Διάγραμμα Πίτας (Pie Chart)

Παρουσιάζουμε ένα διάγραμμα πίτας που αντιπροσωπεύει το συνολικό ποσοστό ομοιότητας και διαφοράς μεταξύ των δύο πολιτικών ασφαλείας. Το γράφημα αυτό παρέχει μια εποπτική αναφορά για το πόσο συνολικά συμφωνούν ή διαφέρουν οι πολιτικές. Όπως βλέπουμε και στην εικόνα 22 εμφανίζονται τα εξής δεδομένα :

- Το διάγραμμα πίτας αναπαριστά το συνολικό ποσοστό ομοιότητας μεταξύ των δύο πολιτικών ασφαλείας.
- Χρησιμοποιεί τις δύο κατηγορίες "Similar Policies" και "Different Policies" για να δείξει την αναλογία μεταξύ τους.
- Πάνω σε κάθε μέρος της πίτας εμφανίζεται και το ανάλογο ποσοστό ομοιότητας.
- Οι χρήστες μπορούν να καταλάβουν εύκολα το ποσοστό ομοιότητας χωρίς να αναλύσουν λεπτομερώς τα δεδομένα.

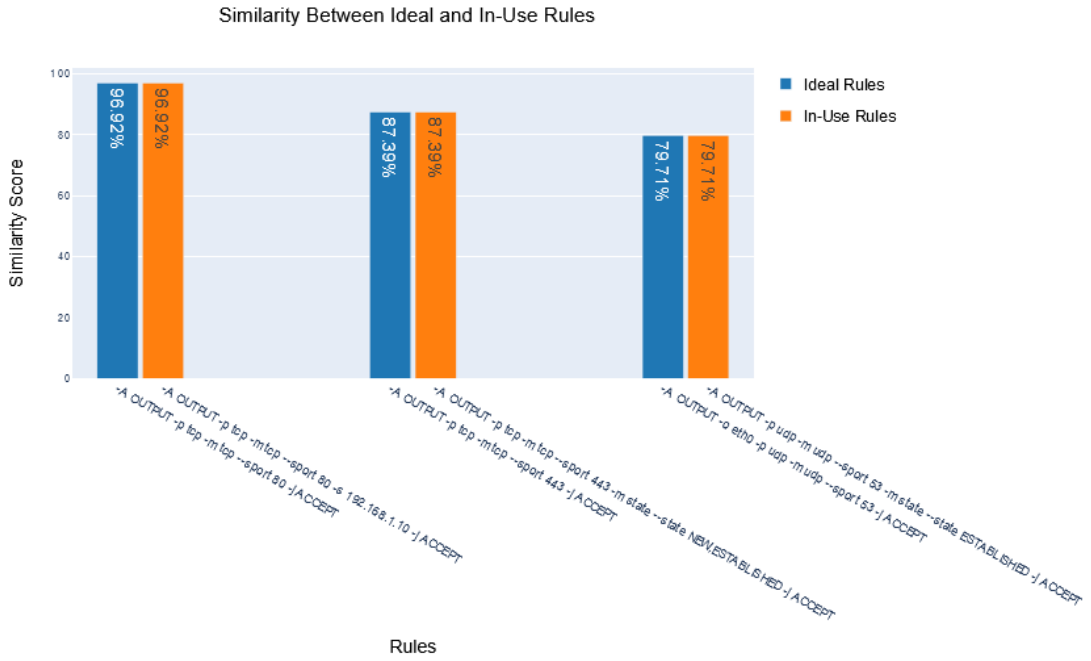


Εικόνα 21 Διάγραμμα Πίτας

Διαγράμματα Ράβδων (Bar Chart)

Παρουσιάζουμε ένα διάγραμμα ράβδων που αναλύει το ποσοστό ομοιότητας μεταξύ των κανόνων ανάμεσα στις πολιτικές. Κάθε ράβδος αντιπροσωπεύει έναν κανόνα, ενώ οι ράβδοι σχετίζονται μεταξύ τους για να παρέχουν μια πλήρη εικόνα των ομοιοτήτων ανάμεσα σε δύο κανόνες των πολιτικών. Όπως βλέπουμε και στην εικόνα 23 εμφανίζονται τα εξής δεδομένα :

- Το διάγραμμα ράβδων παρουσιάζει την ομοιότητα μεταξύ των κανόνων ανάμεσα στη ιδανική και την εν χρήση πολιτική.
- Ο κάθε κανόνας εμφανίζεται ως μια μπάρα, όπου το μήκος της αντιπροσωπεύει το ποσοστό ομοιότητας του.
- Οι μπάρες είναι ανα δύο καθώς έχουμε σύγκριση συγκεκριμένων κανόνων ανάμεσα στις δύο πολιτικές.
- Χρησιμοποιούν διαφορετικά χρώματα όπου οι κανόνες από την ιδανική πολιτική έχουν το μπλε χρώμα ενώ οι κανόνες από την εν χρήση πολιτική έχουν το πορτοκαλί χρώμα.
- Πάνω στις μπάρες αναγράφεται το ακριβές ποσοστό ομοιότητας.
- Κάτω από τις μπάρες αναγράφεται ο κανόνας που έχει το συγκεκριμένο ποσοστό.
- Η αντιπροσώπευση με μπάρες επιτρέπει στους χρήστες να συγκρίνουν εύκολα τα δεδομένα και να εντοπίσουν τις διαφορές.



Εικόνα 22 Διάγραμμα μπάρας

Πίνακας Πανομοιότυπων Κανόνων

Παρουσιάζουμε πίνακα ομοιότητας που περιέχει τις αντίστοιχες πολιτικές που θεωρούνται πανομοιότυπες. Όπως βλέπουμε και στην εικόνα 24 ο πίνακας περιέχει τους κανόνες που αντιστοιχούν και στις δύο πολικές ασφαλείας.

Similar Rules

Rules
:INPUT DROP
:FORWARD DROP
-A INPUT -p tcp -m tcp --dport 80 -j ACCEPT
-A INPUT -p udp -m udp --dport 53 -m state --state NEW,ESTABLISHED -j ACCEPT
-A OUTPUT -p tcp -m tcp --sport 80 -j ACCEPT
-A OUTPUT -p tcp -m tcp --sport 22 -j ACCEPT

Εικόνα 23 Πίνακας με όμοιους κανόνες

Πίνακας Κανόνων Χωρίς Ομοιότητα

Παρουσιάζουμε πίνακα κανόνων χωρίς ομοιότητα που επισημαίνει τους κανόνες που δεν έχουν καμία αντιστοιχία με άλλους κανόνες. Όπως βλέπουμε και στην εικόνα 25 εμφανίζονται τα εξής δεδομένα :

- Έχουμε έναν πίνακα με δύο στήλες
- Η κάθε στήλη περιέχει κανόνες από την κάθε πολιτική ξεχωριστά
- Οι κανόνες των δύο πολιτικών δεν έχουν καμία ομοιότητα μεταξύ τους.

Not Similar Rules

Ideal Rules	In-Use Rules
:FORWARD DROP	:FORWARD ACCEPT
:OUTPUT DROP	:OUTPUT ACCEPT
-A INPUT -p tcp -m tcp --dport 80 -j ACCEPT	-A INPUT -p tcp -m tcp --dport 3306 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 443 -j ACCEPT -A INPUT -p udp -m udp --dport 53 -m state --state NEW,ESTABLISHED -j ACCEPT	
-A INPUT -p tcp -m tcp --dport 22 -s 192.168.1.10 -j ACCEPT	
-A FORWARD -o eth0 -p tcp -m tcp --dport 23 -j ACCEPT	
-A OUTPUT -p udp -m udp --sport 53 -m state --state ESTABLISHED -j ACCEPT	
-A OUTPUT -p tcp -m tcp --sport 21 -m state --state ESTABLISHED -j ACCEPT -A OUTPUT -p tcp -m tcp --sport 22 -j ACCEPT	

Εικόνα 24 Πίνακας με διαφορετικούς κανόνες

7 Εφαρμογή Σε Έναν Οργανισμό

Σε αυτό το κεφάλαιο θα εφαρμόσουμε όλη την διαδικασία σε έναν οργανισμό. Η διαδικασία περιλαμβάνει την εφαρμογή της σύγκρισης ανάμεσα σε 5 εν χρήση και μιας ιδανικής πολιτικής ασφαλείας έτσι ώστε να αποκτήσουμε καλύτερη εικόνα για την ασφάλεια ενός οργανισμού.

7.1 Θεωρητική Κατασκευή Ιδανικής Πολιτικής Ασφαλείας Για Έναν Οργανισμό

Η εγκαθίδρυση μιας ιδανικής πολιτικής ασφαλείας σε έναν οργανισμό αποτελεί θεμελιώδη προσέγγιση για τη διασφάλιση της ολοκληρωμένης προστασίας των πληροφοριών και των ανθρώπινων πόρων. Στο πλαίσιο αυτό, η παρούσα υποενότητα εξετάζει τη θεωρητική κατασκευή μιας πολιτικής ασφαλείας που ανταποκρίνεται στις συγκεκριμένες ανάγκες και προκλήσεις του οργανισμού σε ένα θεωρητικό επίπεδο.
Πιο συγκεκριμένα θέτονται υπόψιν τα παρακάτω βασικά στοιχεία :

- **Σκοπός και Στόχοι της Πολιτικής Ασφαλείας:** Ο πρώτος τομέας επικεντρώνεται στον καθορισμό των στόχων και του σκοπού της πολιτικής ασφαλείας. Μέσα από την ανάλυση των βασικών αναγκών του οργανισμού, προσδιορίζονται οι προτεραιότητες που θα καθοδηγήσουν τη διαμόρφωση της πολιτικής.

- **Θεωρητικά Θεμέλια Πολιτικής Ασφαλείας:** Σε αυτή τη φάση, εξετάζονται διάφορες θεωρητικές προσεγγίσεις, όπως η προληπτική ασφάλεια, η ανίχνευση κινδύνων και η αντίδραση σε περιστατικά. Επιπλέον, εφαρμόζονται ιδέες από κλασικές και

σύγχρονες θεωρίες ασφαλείας για να διαμορφωθεί ένα στιβαρό θεωρητικό υπόβαθρο.

- **Διαδικασία Δημιουργίας Πολιτικής:** Αναλύονται τα βήματα που πρέπει να ακολουθηθούν για τη δημιουργία μιας ιδανικής πολιτικής ασφαλείας. Προτείνονται μέθοδοι ενσωμάτωσης των μελών του οργανισμού σε αυτή τη διαδικασία, προάγοντας τον συνεργατικό χαρακτήρα της.

- **Αξιολόγηση και Βελτιώσεις:** Επικεντρώνονται στα κριτήρια αξιολόγησης της αποτελεσματικότητας της πολιτικής ασφαλείας. Παράλληλα, δίνονται προτάσεις για βελτιώσεις, προσαρμογές και ευέλικτες διαδικασίες προκειμένου να ανταποκριθεί η πολιτική στην εξέλιξη του περιβάλλοντος.

Μετά την εκτενή μας ανάλυση των θεμελιωδών αρχών και των βασικών βημάτων στη δημιουργία μιας γενικής πολιτικής ασφαλείας, και την αναγνώριση της σημασίας της συνεχούς προσαρμογής και βελτίωσης, είναι πλέον καιρός να εφαρμόσουμε αυτές τις αρχές σε ένα πιο συγκεκριμένο πλαίσιο. Στο επόμενο βήμα μας, θα διερευνήσουμε τη θεωρητική κατασκευή μιας ιδανικής πολιτικής ασφαλείας, εστιάζοντας στην προστασία και διαχείριση δικτυακών συστημάτων. Η ενότητα αυτή θα παρουσιάσει συγκεκριμένες εντολές και πρακτικές που εφαρμόζονται στην εισερχόμενη και εξερχόμενη κίνηση, καθώς και στην προώθηση της κίνησης μεταξύ δικτυακών συσκευών σε φυσική γλώσσα, δίνοντας έτσι μια ολοκληρωμένη εικόνα των πρακτικών εφαρμογών των θεωρητικών αρχών που εξετάστηκαν προηγουμένως. Η παρακάτω σειρά εντολών αποτελεί τη θεωρητική κατασκευή μιας πολιτικής ασφαλείας για ένα δίκτυο.

Ακολουθεί η αναλυτική παρουσίαση των εντολών ασφαλείας:

- **Προεπιλεγμένες Κινήσεις Απόρριψης πακέτων** : Καθορίζεται προεπιλεγμένη πολιτική ασφαλείας για απόρριψη εισερχόμενης, εξερχόμενης και προώθησης κίνησης.

    - The default policy of incoming traffic is set to drop.
    - The default policy of outgoing traffic is set to drop.
    - The default policy of forward traffic is set to drop.

Κίνηση HTTP/HTTPS:

- **Κίνηση HTTP/HTTPS :** Επιτρέπεται η απρόσκοπτη ροή κίνησης σε TCP θύρα 80 (HTTP) και 443 (HTTPS) από οποιαδήποτε πηγή.

    - Allow incoming traffic on TCP port 80 (HTTP) from any source IP address.
    - Allow outgoing traffic on TCP port 80 (HTTP) to any destination IP address.
    - Allow incoming traffic on TCP port 443 (HTTPS) from any source IP address.
    - Allow outgoing traffic on TCP port 443 (HTTPS) to any destination IP address.

- **Κίνηση DNS:** Επιτρέπεται η εισερχόμενη κίνηση σε UDP θύρα 53 (DNS) για νέα ή υπάρχουσα σύνδεση και εξερχόμενης κίνησης σε UDP θύρα 53 (DNS) για υφιστάμενη σύνδεση.

    - Allow new and established incoming traffic on UDP port 53 (DNS) from any source IP address.
    - Let established outgoing traffic on UDP port 53 (DNS) to any destination IP address.

- **Εισερχόμενη Κίνηση FTP :** Επιτρέπεται η εισερχόμενη κίνηση σε TCP πόρτα 21 (FTP) από οποιαδήποτε πηγή IP, τόσο για νέες συνδέσεις όσο και για υφιστάμενη.
    - Permit new and established incoming traffic on TCP port 21 (FTP) from any source IP address.
- **Εξερχόμενη Κίνηση FTP :** Επιτρέπεται η εξερχόμενη κίνηση στην TCP θύρα 21 (FTP) για υφιστάμενες συνδέσεις προς οποιαδήποτε πηγή IP.
    - Permit established outgoing traffic on TCP port 21 (FTP) to any destination IP address.
- **Εισερχόμενη Κίνηση SSH :** Επιτρέπεται η εισερχόμενη κίνηση στην TCP θύρα 22 (SSH) από την διεύθυνση IP 192.168.1.10.
    - Allow incoming traffic on TCP port 22 (SSH) from 192.168.1.10 IP address.
- **Εξερχόμενη Κίνηση SSH :** Επιτρέπεται η εξερχόμενη κίνηση στην TCP θύρα 22 (SSH) προς οποιαδήποτε πηγή IP.
    - Allow outgoing traffic on TCP port 22 (SSH) to any destination IP address.
- **Προώθηση Εξερχόμενης Κίνησης Telnet :** Επιτρέπεται η προώθηση εξερχόμενης κίνησης στη διεπαφή eth0 στην TCP θύρα 23 (Telnet) προς οποιαδήποτε πηγή IP.
    - Allow forward outgoing traffic on interface eth0 on TCP port 23 (Telnet) to any destination IP address.

7.2 Μετατροπή Ιδανικής Πολιτικής Σε Κανόνες Iptables

Η Ιδανική Πολιτική που προτάθηκε προηγουμένως μεταφράζεται σε συγκεκριμένους κανόνες iptables, οι οποίοι καθορίζουν τις απαραίτητες ρυθμίσεις ασφαλείας για εισερχόμενη, εξερχόμενη και προωθούμενη κίνηση. Η μετατροπή έγινε με βάση την υλοποίηση του κώδικα μετατροπής μέσω nlp τεχνολογίας. Ακολουθούν οι κανόνες Iptables:

```
# Ορισμός προεπιλεγμένων πολιτικών ασφαλείας
iptables -P INPUT DROP
iptables -P OUTPUT DROP
iptables -P FORWARD DROP

# Κίνηση HTTP/HTTPS
iptables -A INPUT -p tcp -m tcp --dport 80 -j ACCEPT
iptables -A OUTPUT -p tcp -m tcp --sport 80 -j ACCEPT
iptables -A INPUT -p tcp -m tcp --dport 443 -j ACCEPT
iptables -A OUTPUT -p tcp -m tcp --sport 443 -j ACCEPT

# Κίνηση DNS
iptables -A INPUT -p udp -m udp --dport 53 -m state --state NEW,ESTABLISHED -j ACCEPT
iptables -A OUTPUT -p udp -m udp --sport 53 -m state --state ESTABLISHED -j ACCEPT

# Κίνηση FTP
iptables -A INPUT -p tcp -m tcp --dport 21 -m state --state NEW,ESTABLISHED -j ACCEPT
```

```
iptables -A OUTPUT -p tcp -m tcp --sport 21 -m state --state ESTABLISHED -j ACCEPT

# Κίνηση SSH
iptables -A INPUT -p tcp -m tcp --dport 22 -s 192.168.1.10 -j ACCEPT
iptables -A OUTPUT -p tcp -m tcp --sport 22 -j ACCEPT

# Προώθηση εξερχόμενης κίνησης Telnet
iptables -A FORWARD -o eth0 -p tcp -m tcp --dport 23 -j ACCEPT
```

Οι παραπάνω εντολές δημιουργούν ένα συγκεκριμένο σύνολο κανόνων iptables που είναι συμβατοί με την Ιδανική Πολιτική Ασφαλείας. Η προσεκτική εφαρμογή και παρακολούθηση αυτών των κανόνων στο σύστημα θα διασφαλίσει την ασφάλεια και τη σωστή λειτουργία του δικτύου.

7.3 Μετατροπή Ιδανικής Πολιτικής Σε Κανόνες Iptables για Linux Συστήματα

Η ιδανική πολιτική ασφαλείας που ορίσαμε θα μετατραπεί σε συγκεκριμένους κανόνες Iptables για Linux μηχάνημα. Η υλοποίηση πραγματοποιείται με βάση το εργαλείο ανοικτού κώδικα iptables-converter[55], ενώ η σύνταξη των κανόνων είναι πλέον συμβατή με τον τρόπο που λειτουργεί το λειτουργικό Linux. Ακολουθούν οι κανόνες iptables:

```
*raw
:PREROUTING ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
COMMIT

*nat
:PREROUTING ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
:POSTROUTING ACCEPT [0:0]
COMMIT

*mangle
:PREROUTING ACCEPT [0:0]
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
:POSTROUTING ACCEPT [0:0]
COMMIT

*filter
:INPUT DROP [0:0]
:FORWARD DROP [0:0]
:OUTPUT DROP [0:0]

# Κίνηση HTTP/HTTPS
-A INPUT -p tcp -m tcp --dport 80 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 443 -j ACCEPT
-A OUTPUT -p tcp -m tcp --sport 80 -j ACCEPT
-A OUTPUT -p tcp -m tcp --sport 443 -j ACCEPT

# Κίνηση DNS
-A INPUT -p udp -m udp --dport 53 -m state --state NEW,ESTABLISHED -j ACCEPT
```

```
-A OUTPUT -p udp -m udp --sport 53 -m state --state ESTABLISHED -j ACCEPT

# Κίνηση FTP
-A INPUT -p tcp -m tcp --dport 21 -m state --state NEW,ESTABLISHED -j ACCEPT
-A OUTPUT -p tcp -m tcp --sport 21 -m state --state ESTABLISHED -j ACCEPT

# Κίνηση SSH από την διεύθυνση 192.168.1.10
-A INPUT -p tcp -m tcp --dport 22 -s 192.168.1.10 -j ACCEPT
-A OUTPUT -p tcp -m tcp --sport 22 -j ACCEPT

# Προώθηση εξερχόμενης κίνησης Telnet
-A FORWARD -o eth0 -p tcp -m tcp --dport 23 -j ACCEPT

COMMIT
```

Αυτοί οι κανόνες iptables εφαρμόζουν μια ισχυρή πολιτική ασφαλείας, επιτρέποντας μόνο την αναγκαία κίνηση για την ικανοποίηση των απαιτήσεων της Ιδανικής Πολιτικής. Επιπλέον, εξασφαλίζουν τη συμβατότητα με το περιβάλλον λειτουργίας των λειτουργιών των Linux.

7.4 Απόκτηση Εν Χρήση Πολιτικής Ασφαλείας

Βασιζόμενοι στην υλοποίηση που προσδιορίσαμε, έχουμε αναπτύξει πέντε διακριτές πολιτικές ασφαλείας, οι οποίες εφαρμόζονται σε διάφορα επίπεδα ασφάλειας. Αυτές οι πολιτικές ακολουθούν την ίδια διαδικασία απόκτησης και εφαρμογής.

Η τεχνολογία που χρησιμοποιούμε για την επικοινωνία μεταξύ των συστημάτων είναι η gRPC. Σε αυτό το πλαίσιο, ορίζουμε τον Οργανισμό ως τον "πελάτη" (client) και τον δικό μας υπολογιστή ως τον "διακομιστή" (server). Ο διακομιστής είναι προγραμματισμένος να ακούει για εισερχόμενες συνδέσεις μόνο στη θύρα 50051. Αυτό σημαίνει ότι δημιουργεί ένα ασφαλές κανάλι επικοινωνίας, το οποίο προστατεύεται μέσω της χρήσης του πρωτοκόλλου TLS (Transport Layer Security). Το πρωτόκολλο TLS εξασφαλίζει ότι όλα τα δεδομένα που μεταδίδονται μέσω αυτού του καναλιού είναι κρυπτογραφημένα και παραμένουν ασφαλή.
Όταν ο Οργανισμός (ως client) θέλει να δημιουργήσει μια σύνδεση με τον server, χρησιμοποιεί το δημόσιο πιστοποιητικό του server για να εγκαθιδρύσει μια ασφαλή σύνδεση μέσω του καναλιού TLS. Αυτό διασφαλίζει ότι η επικοινωνία μεταξύ του client και του server είναι κρυπτογραφημένη και προστατευμένη από παρεμβάσεις τρίτων. Οι πολιτικές ασφαλείας που λαμβάνονται μέσω αυτής της σύνδεσης είναι επίσης κρυπτογραφημένες, προσφέροντας ένα επιπλέον επίπεδο ασφάλειας. Μετά τη μετάδοση, αυτές οι πολιτικές αποκρυπτογραφούνται για την εφαρμογή τους.

Αυτή η διαδικασία εξασφαλίζει ότι η επικοινωνία και η ανταλλαγή δεδομένων μεταξύ του Οργανισμού και του διακομιστή είναι ασφαλής και προστατευμένη από ανεπιθύμητες παρεμβολές.

7.5 Σύγκριση Ιδανικής Και Εν Χρήση Πολιτικής Ασφαλείας

Σε αυτό το κεφάλαιο θα πραγματοποιήσουμε 5 συγκρίσεις Πολιτικών ασφαλείας με κλιμακούμενο επίπεδο ασφαλείας.

7.5.1 Εφαρμογή Εν χρήση Πολιτικών Ασφαλείας

Στο πλαίσιο της παρούσας υποενότητας, επικεντρωνόμαστε στην επίδειξη της εφαρμογής των πολιτικών ασφαλείας που έχουν καθοριστεί για το περιβάλλον μας. Οι παρακάτω

ενότητες παρέχουν μια επισκόπηση των συγκεκριμένων κανόνων που υλοποιούνται μέσω των Iptables, προσφέροντας συγκεκριμένες προδιαγραφές ασφάλειας για το δίκτυο μας. Μέσω αυτών των πολιτικών, επιδιώκουμε να επιτύχουμε την αξιοπιστία και την προστασία των δικτυακών μας πόρων σε μια κλιμακούμενη προσέγγιση έτσι ώστε να δημιουργήσουμε ένα σχετικό παράδειγμα σύγκρισης των συγκεκριμένων 5 εν χρήση πολιτικών ασφαλείας της εταιρίας με την ιδανική πολιτική ασφαλείας.

Κάθε πολιτική ασφαλείας περιγράφει ένα κλιμακούμενο επίπεδο ασφαλείας:

- **Επίπεδο 1:** Βασικό επίπεδο ασφαλείας.
    - Σε αυτό το επίπεδο, όλα τα πακέτα είναι επιτρεπόμενα σε όλες τις κατευθύνσεις (INPUT, FORWARD, OUTPUT).
    - Προσανατολίζεται σε βασικές λειτουργίες, επιτρέποντας εισερχόμενα πακέτα UDP στην θύρα 80 και πακέτα TCP εξερχόμενα από την θύρα 80.

      ```
      # Generated by iptables-save v1.8.9 (nf_tables) on Thu Nov 9 18:11:35 2023
      *filter
      :INPUT ACCEPT [0:0]
      :FORWARD ACCEPT [0:0]
      :OUTPUT ACCEPT [0:0]
      -A INPUT -p udp -m udp --dport 80 -j ACCEPT
      -A OUTPUT -p tcp -m tcp --sport 80 -j ACCEPT
      COMMIT
      # Completed on Thu Nov 9 18:11:35 2023
      ```

- **Επίπεδο 2:** Αυξημένο επίπεδο ασφαλείας.
    - Απορρίπτονται όλα τα πακέτα στην εισερχόμενη κίνηση ενώ επιτρέπονται στην εξερχόμενη και στην προώθηση.
    - Επιτρέπεται εισερχόμενη TCP κίνηση στη θύρα 21 για νέες ή ήδη υφιστάμενες συνδέσεις.
    - Επιτρέπεται η εισερχόμενη κίνηση στην θύρα TCP 21
    - Επιτρέπονται όλες οι κινήσεις στην θύρα TCP 3306.
    - Επιτρέπονται εξερχόμενα πακέτα UDP στην θύρα 80 και πακέτα TCP στην θύρα 443 για νέες ή υφιστάμενες συνδέσεις.

      ```
      # Generated by iptables-save v1.8.9 (nf_tables) on Thu Nov 9 18:55:35 2023
      *filter
      :INPUT DROP [0:0]
      :FORWARD ACCEPT [0:0]
      :OUTPUT ACCEPT [0:0]
      -A INPUT -p tcp -m tcp --dport 21 -m state --state NEW,ESTABLISHED -j ACCEPT
      -A INPUT -p tcp -m tcp --dport 3306 -j ACCEPT
      -A OUTPUT -p udp -m udp --sport 80 -m state --state NEW,ESTABLISHED -j ACCEPT
      -A OUTPUT -p tcp -m tcp --sport 443 -m state --state NEW,ESTABLISHED -j ACCEPT
      COMMIT
      # Completed on Thu Nov 9 18:55:35 2023
      ```

- **Επίπεδο 3:** Προηγμένο επίπεδο ασφαλείας.

- Απορρίπτονται όλα τα πακέτα στην εισερχόμενη κίνηση ενώ επιτρέπονται στην εξερχόμενη και στην προώθηση.
- Επιτρέπεται εισερχόμενη κίνηση με πακέτα TCP σε θύρες 21 και 3306, καθώς και εισερχόμενα πακέτα UDP στην πόρτα 53 για νέες ή υφιστάμενες συνδέσεις.
- Επιτρέπεται εξερχόμενη κίνηση από την θύρα 80 με πακέτα TCP από τον υπολογιστή με διεύθυνση IP 192.168.1.10.
- Επιτρέπονται εξερχόμενα πακέτα TCP στην θύρα 443 για νέες ή υφιστάμενες συνδέσεις.
- Επιτρέπονται εξερχόμενα πακέτα UDP στην θύρα 53 που προέρχονται από την διεπαφή eth0.

```
# Generated by iptables-save v1.8.9 (nf_tables) on Thu Nov  9 19:11:38 2023
*filter
:INPUT DROP [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
-A INPUT -p tcp -m tcp --dport 21 -m state --state NEW,ESTABLISHED -j ACCEPT
-A INPUT -p tcp -m tcp --dport 3306 -j ACCEPT
-A INPUT -p udp -m udp --dport 53 -m state --state NEW,ESTABLISHED -j ACCEPT
-A OUTPUT -p tcp -m tcp --sport 80 -s 192.168.1.10 -j ACCEPT
-A OUTPUT -p tcp -m tcp --sport 443 -m state --state NEW,ESTABLISHED -j ACCEPT
-A OUTPUT -o eth0 -p udp -m udp --sport 53 -j ACCEPT
COMMIT
# Completed on Thu Nov  9 19:11:38 2023
```

- **Επίπεδο 4:** Υψηλό επίπεδο ασφαλείας.

- Απορρίπτονται όλα τα πακέτα στην εισερχόμενη κίνηση και στην προώθηση ενώ επιτρέπονται στην εξερχόμενη.
- Επιτρέπεται εισερχόμενη κίνηση σε θύρες 80, 3306 και 22 με TCP πακέτα.
- Επιτρέπεται εισερχόμενη κίνηση στην θύρα 53 με UDP πακέτα για νέες ή υφιστάμενες συνδέσεις.
- Επιτρέπεται προώθηση κίνησης στην θύρα 23 με TCP πακέτα.
- Επιτρέπονται εξερχόμενα πακέτα TCP στις θύρες 80 και 22.
- Επιτρέπονται εξερχόμενα πακέτα UDP στην θύρα 443 για νέες ή υφιστάμενες συνδέσεις.
- Επιτρέπονται εξερχόμενα πακέτα UDP στην θύρα 53 που προέρχονται από την διεπαφή eth0.

```
# Generated by iptables-save v1.8.9 (nf_tables) on Thu Nov 9 19:28:42 2023
*filter
:INPUT DROP [0:0]
:FORWARD DROP [0:0]
:OUTPUT ACCEPT [0:0]
-A INPUT -p tcp -m tcp --dport 80 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 3306 -j ACCEPT
-A INPUT -p udp -m udp --dport 53 -m state --state NEW,ESTABLISHED -j ACCEPT
-A INPUT -p tcp -m tcp --dport 22 -j ACCEPT
-A FORWARD -p tcp -m tcp --dport 23 -j ACCEPT
-A OUTPUT -p tcp -m tcp --sport 80 -j ACCEPT
```

```
-A OUTPUT -p tcp -m tcp --sport 443 -m state --state NEW,ESTABLISHED -j
ACCEPT
-A OUTPUT -o eth0 -p udp -m udp --sport 53 -j ACCEPT
-A OUTPUT -p tcp -m tcp --sport 22 -j ACCEPT
COMMIT
# Completed on Thu Nov 9 19:28:42 2023
```

- **Επίπεδο 5:** Πολύ υψηλό επίπεδο ασφαλείας.

    - Απορρίπτονται όλα τα πακέτα στην εισερχόμενη, εξερχόμενη και στην προώθηση.
    - Επιτρέπεται εισερχόμενη κίνηση σε θύρες 80, 3306 με TCP πακέτα.
    - Επιτρέπεται εισερχόμενη κίνηση σε θύρες 53, 21 για νέες ή υφιστάμενες συνδέσεις με TCP πακέτα.
    - Επιτρέπεται εισερχόμενη κίνηση από την θύρα 22 από τον υπολογιστή με διεύθυνση IP 192.168.1.10 με TCP πακέτα.
    - Επιτρέπεται προώθηση κίνησης στην θύρα 23 με TCP πακέτα.
    - Επιτρέπονται εξερχόμενα πακέτα TCP στις θύρες 80 και 22.
    - Επιτρέπονται εξερχόμενα πακέτα TCP στην θύρα 443 για υφιστάμενες συνδέσεις.
    - Επιτρέπονται εξερχόμενα πακέτα UDP στην θύρα 53, καθώς και εξερχόμενα πακέτα TCP στην θύρα 21 για υφιστάμενες συνδέσεις.

    ```
    # Generated by iptables-save v1.8.9 (nf_tables) on Thu Nov 9 19:55:22 2023
    *filter
    :INPUT DROP [0:0]
    :FORWARD DROP [0:0]
    :OUTPUT DROP [0:0]
    -A INPUT -p tcp -m tcp --dport 80 -j ACCEPT
    -A INPUT -p tcp -m tcp --dport 3306 -j ACCEPT
    -A INPUT -p udp -m udp --dport 53 -m state --state NEW,ESTABLISHED -j
    ACCEPT
    -A INPUT -p tcp -m tcp --dport 21 -m state --state NEW,ESTABLISHED -j
    ACCEPT
    -A INPUT -p tcp -m tcp --dport 22 -s 192.168.1.10 -j ACCEPT
    -A FORWARD -p tcp -m tcp --dport 23 -j ACCEPT
    -A OUTPUT -p tcp -m tcp --sport 80 -j ACCEPT
    -A OUTPUT -p tcp -m tcp --sport 443 -m state --state NEW,ESTABLISHED -j
    ACCEPT
    -A OUTPUT -p udp -m udp --sport 53 -m state --state ESTABLISHED -j
    ACCEPT
    -A OUTPUT -p tcp -m tcp --sport 21 -m state --state ESTABLISHED -j
    ACCEPT
    -A OUTPUT -p tcp -m tcp --sport 22 -j ACCEPT
    COMMIT
    # Completed on Thu Nov 9 19:55:22 2023
    ```

7.5.2 Εμφάνιση Συγκρίσεων Στην Εφαρμογή Dash

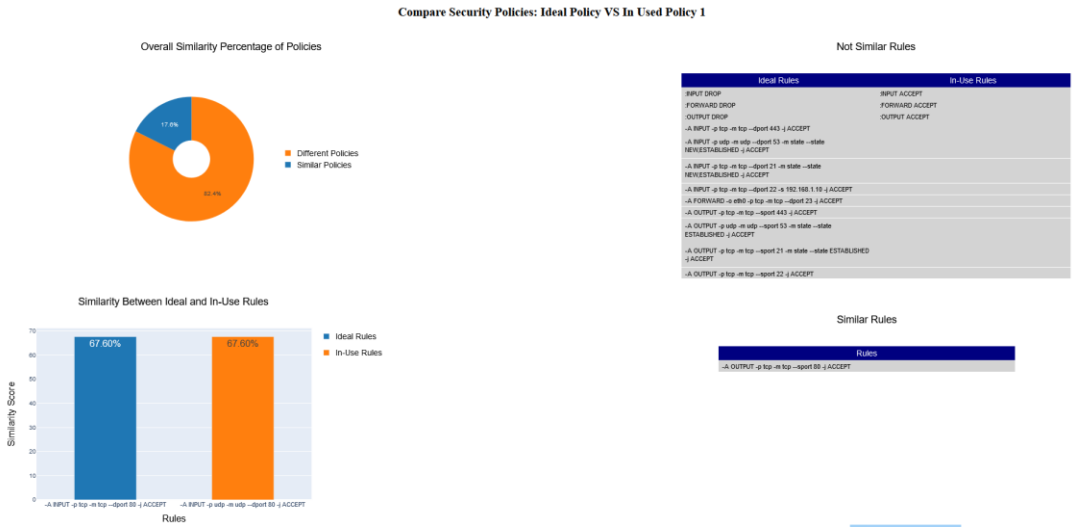
Σε αυτήν την υποενότητα επικεντρωνόμαστε στην εμφάνιση συγκρίσεων χρησιμοποιώντας την εφαρμογή Dash. Μέσω αυτής της παρουσίασης, εμφανίζουμε συνοπτικά και εμπεριστατωμένα όλες τις συγκρίσεις και τα αποτελέσματα μεταξύ της ιδανικής πολιτικής και των 5 εν χρήση πολιτικών ασφαλείας.

- **Σύγκριση Ιδανικής Έναντι Εν Χρήση Πολιτικής Ασφαλείας Επιπέδου 1**

  Με βάση την εικόνα 26, έχουμε τα εξής αποτελέσματα για την σύγκριση :

  - Το συνολικό ποσοστό ομοιότητας των δύο πολιτικών φτάνει στο ποσοστό **17,6%**
  - Υπάρχει μόνο ένα ζευγάρι πανομοιότυπων εντολών.
  - Υπάρχει μόνο ένα ζευγάρι όμοιων εντολών με ποσοστό 67.60%

  Συμπέρασμα : Βασιζόμενοι στα παραπάνω ευρήματα από τη σύγκριση της ιδανικής έναντι της εν χρήση Πολιτικής Ασφαλείας Επιπέδου 1, μπορούμε να συνάγουμε τα εξής :
  Το συνολικό ποσοστό ομοιότητας που φτάνει στο 17,6% υποδηλώνει ότι οι δύο πολιτικές διαφέρουν σε μεγάλο βαθμό. Η ύπαρξη ενός μόνο ζευγαριού πανομοιότυπων εντολών υποδεικνύει περιορισμένη αντιστοιχία μεταξύ του ιδανικού μοντέλου και της πραγματικής πολιτικής που εφαρμόζεται.
  Επιπλέον, το γεγονός ότι υπάρχει μόνο ένα ζευγάρι όμοιων εντολών με ποσοστό 67,60% υποδεικνύει ότι, αν και υπάρχει κάποια αντιστοιχία, υπάρχουν σημαντικές διαφορές και προσαρμογές στην πραγματική εφαρμογή της πολιτικής ασφαλείας σε σχέση με το ιδανικό σενάριο.
  Συνολικά, τα παραπάνω συμπεράσματα υποδηλώνουν την ανάγκη για περαιτέρω ανάλυση και πιθανές βελτιώσεις στην πολιτική ασφαλείας Επιπέδου 1 προκειμένου να προσεγγίσει περισσότερο το ιδανικό μοντέλο.



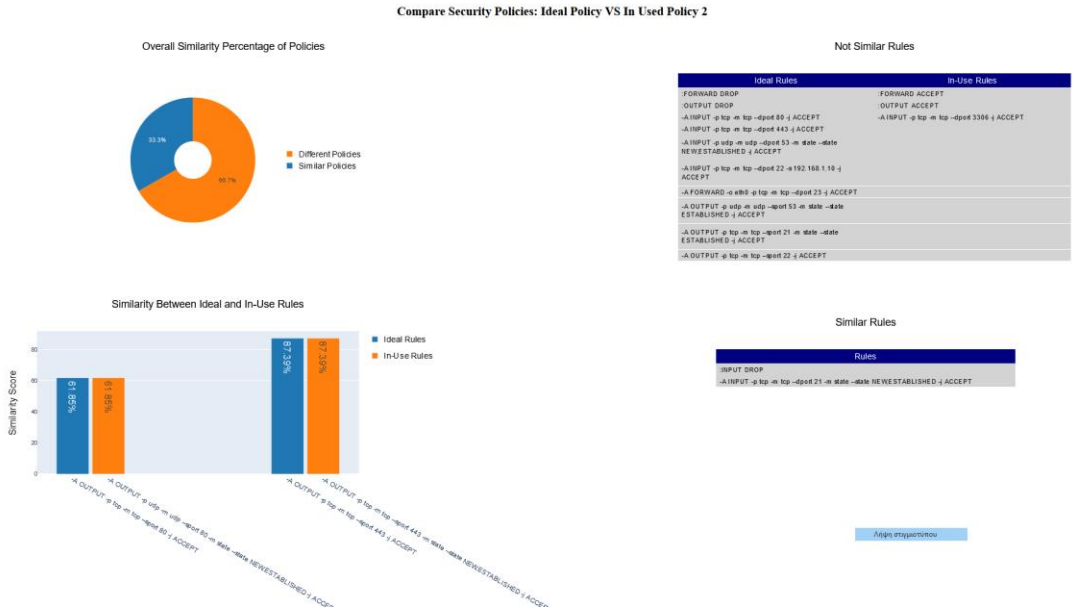
Εικόνα 25 Σύγκριση ιδανικής έναντι εν χρήση πολιτικής ασφαλείας επιπέδου 1

- **Σύγκριση Ιδανικής Έναντι Εν Χρήση Πολιτικής Ασφαλείας Επιπέδου 2**

  Με βάση την εικόνα 27, έχουμε τα εξής αποτελέσματα για την σύγκριση :

  - Το συνολικό ποσοστό ομοιότητας των δύο πολιτικών φτάνει στο ποσοστό **33,3%**
  - Υπάρχουν δύο ζευγάρια πανομοιότυπων εντολών.
  - Υπάρχουν δύο ζευγάρια όμοιων εντολών με ποσοστό 61.85% και 87,39%

Συμπέρασμα : Βασιζόμενοι στα παραπάνω ευρήματα από τη σύγκριση της ιδανικής έναντι της εν χρήση Πολιτικής Ασφαλείας Επιπέδου 2, μπορούμε να συνάγουμε τα εξής :
Το συνολικό ποσοστό ομοιότητας του 33,3% υποδεικνύει κάποια αντιστοιχία μεταξύ των δύο πολιτικών, αλλά παραμένει σε σχετικά χαμηλά επίπεδα.
Επιπλέον, η ύπαρξη δύο ζευγαριών πανομοιότυπων εντολών υποδεικνύει ότι υπάρχουν τμήματα στις δύο πολιτικές που λειτουργούν με παρόμοιο τρόπο, αλλά αυτά είναι περιορισμένα.
Τέλος, τα δύο ζευγάρια όμοιων εντολών με ποσοστά 61,85% και 87,39% υποδεικνύουν σημαντική αντιστοιχία σε συγκεκριμένες εντολές. Αυτό ενδείκνυται ότι ορισμένες πτυχές της εν χρήση Πολιτικής Ασφαλείας Επιπέδου 2 είναι πιο κοντά στην ιδανική κατάσταση.
Συνολικά, τα αποτελέσματα υποδεικνύουν ότι η πολιτική ασφαλείας χρειάζεται πιθανώς περαιτέρω προσαρμογές και βελτιώσεις για να είναι πιο συμβατή με το ιδανικό σενάριο.



Εικόνα 26 Σύγκριση ιδανικής έναντι εν χρήση πολιτικής ασφαλείας επιπέδου 2

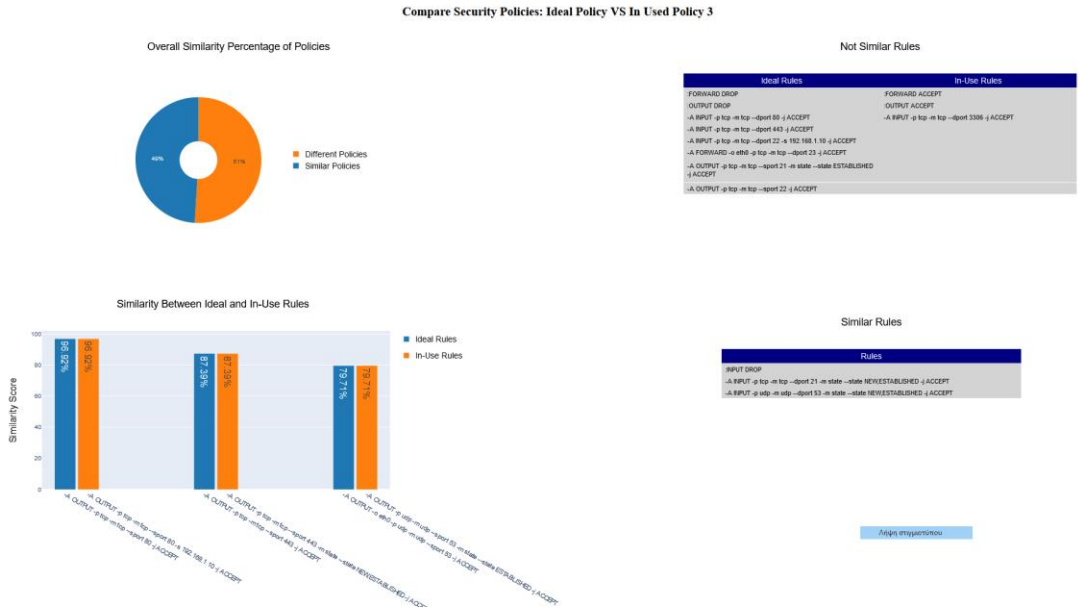
- **Σύγκριση Ιδανικής Έναντι Εν Χρήση Πολιτικής Ασφαλείας Επιπέδου 3**

  Με βάση την εικόνα 28, έχουμε τα εξής αποτελέσματα για την σύγκριση :

  - Το συνολικό ποσοστό ομοιότητας των δύο πολιτικών φτάνει στο ποσοστό **49%**
  - Υπάρχουν τρία ζευγάρια πανομοιότυπων εντολών.
  - Υπάρχουν τρία ζευγάρια όμοιων εντολών με ποσοστό 96,92%, 87,39% και 79,71%

  Συμπεράσματα Βασιζόμενοι στα παραπάνω ευρήματα από τη σύγκριση της ιδανικής έναντι της εν χρήση Πολιτικής Ασφαλείας Επιπέδου 3, μπορούμε να συνάγουμε τα εξής :

Το συνολικό ποσοστό ομοιότητας του 49% υποδεικνύει ότι υπάρχει μια σημαντική αντιστοιχία μεταξύ των δύο πολιτικών, αλλά εξακολουθούν να υπάρχουν σημαντικές διαφορές.
Επιπρόσθετα, η ύπαρξη τριών ζευγαριών πανομοιότυπων εντολών υποδεικνύει ότι υπάρχουν περιοχές όπου οι δύο πολιτικές λειτουργούν παρόμοια, ενώ παράλληλα υπάρχουν και διαφορετικές πτυχές.
Ακόμα, τα τρία ζευγάρια όμοιων εντολών με ποσοστά 96,92%, 87,39% και 79,71% υποδεικνύουν ότι σε συγκεκριμένες περιοχές, η πολιτική ασφαλείας είναι ιδιαίτερα παρόμοια με το ιδανικό σενάριο.
Συνολικά, τα αποτελέσματα υποδεικνύουν μια σταδιακή βελτίωση στη συμβατότητα μεταξύ της εν χρήση Πολιτικής Ασφαλείας Επιπέδου 3 και της Ιδανικής, αλλά παραμένουν περιθωριακές διαφορές που μπορεί να απαιτούν περαιτέρω προσαρμογές για τη βελτίωση της απόδοσης.



Εικόνα 27 Σύγκριση ιδανικής έναντι εν χρήση πολιτικής ασφαλείας επιπέδου 3

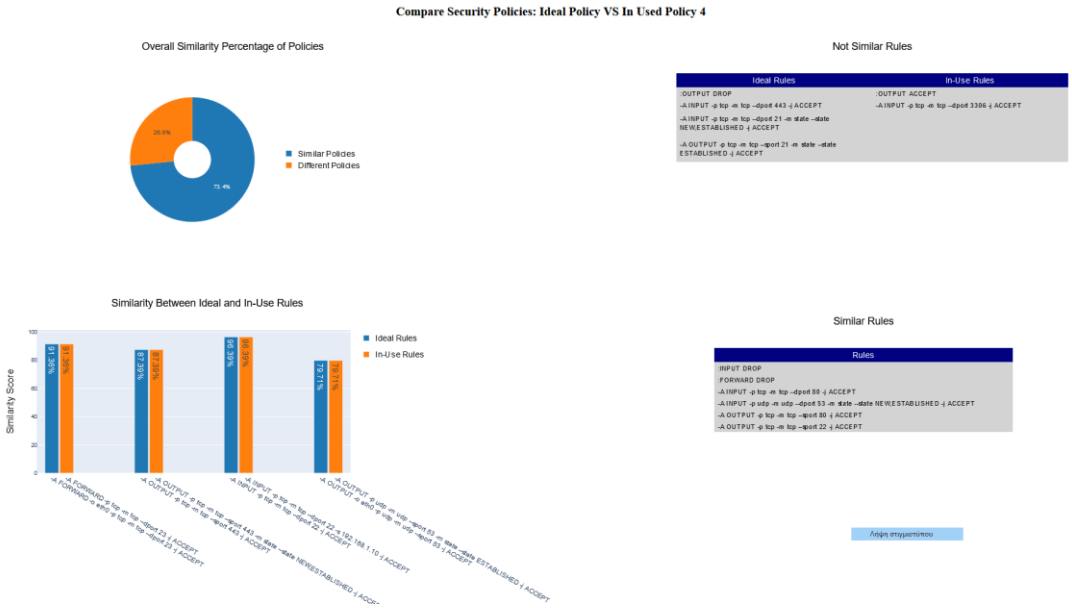
- **Σύγκριση Ιδανικής Έναντι Εν Χρήση Πολιτικής Ασφαλείας Επιπέδου 4**

  Με βάση την εικόνα 29, έχουμε τα εξής αποτελέσματα για την σύγκριση :

  - Το συνολικό ποσοστό ομοιότητας των δύο πολιτικών φτάνει στο ποσοστό **73,4%**
  - Υπάρχουν έξι ζευγάρια πανομοιότυπων εντολών.
  - Υπάρχουν τέσσερα ζευγάρια όμοιων εντολών με ποσοστό 91,36%, 87,39%, 96,39% και 79,71%

  Συμπεράσματα : Βασιζόμενοι στα παραπάνω ευρήματα από τη σύγκριση της ιδανικής έναντι της εν χρήση Πολιτικής Ασφαλείας Επιπέδου 4, μπορούμε να συνάγουμε τα εξής :
  Το υψηλό συνολικό ποσοστό ομοιότητας του 73,4% υποδεικνύει ότι η πολιτική ασφαλείας Επιπέδου 4 έχει καταφέρει να προσεγγίσει σημαντικά το ιδανικό σενάριο.
  Επιπρόσθετα, η ύπαρξη έξι ζευγαριών πανομοιότυπων εντολών υποδεικνύει ότι οι δύο πολιτικές λειτουργούν παρόμοια σε πολλές πτυχές.

Ακόμα, τα τέσσερα ζευγάρια όμοιων εντολών με ποσοστά 91,36%, 87,39%, 96,39% και 79,71% επιβεβαιώνουν την υψηλή συμφωνία σε συγκεκριμένες εντολές και διαδικασίες μεταξύ των δύο πολιτικών.
Συνολικά, τα αποτελέσματα υποδεικνύουν σημαντική πρόοδο στην προσαρμογή της πολιτικής ασφαλείας Επιπέδου 4 προς το ιδανικό σενάριο, με μικρές πιθανές περαιτέρω βελτιώσεις που μπορεί να ενισχύσουν ακόμα περισσότερο τη συμβατότητα.



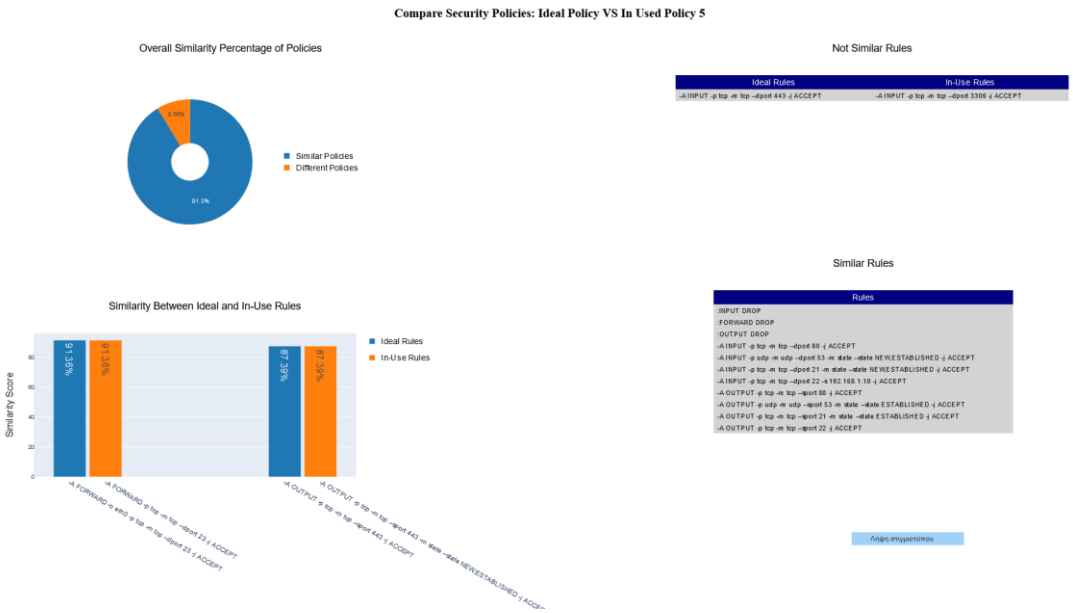
Εικόνα 28 Σύγκριση Ιδανικής έναντι εν χρήση πολιτικής ασφαλείας επιπέδου 4

- **Σύγκριση ιδανικής έναντι Εν χρήση Πολιτικής Ασφαλείας Επιπέδου 5**

  Με βάση την εικόνα 30, έχουμε τα εξής αποτελέσματα για την σύγκριση :

  - Το συνολικό ποσοστό ομοιότητας των δύο πολιτικών φτάνει στο ποσοστό **91,3%**
  - Υπάρχουν έντεκα ζευγάρια πανομοιότυπων εντολών.
  - Υπάρχουν δύο ζευγάρια όμοιων εντολών με ποσοστό 91,36% και 87,39%

  Συμπεράσματα : Βασιζόμενοι στα παραπάνω ευρήματα από τη σύγκριση της ιδανικής έναντι της εν χρήση Πολιτικής Ασφαλείας Επιπέδου 5, μπορούμε να συνάγουμε τα εξής :
  Το υψηλό συνολικό ποσοστό ομοιότητας του 91,3% υποδεικνύει ότι η πολιτική ασφαλείας Επιπέδου 5 έχει επιτύχει σημαντική συμβατότητα με το ιδανικό σενάριο.
  Επιπρόσθετα, η ύπαρξη έντεκα ζευγαριών πανομοιότυπων εντολών υποδεικνύει ότι οι δύο πολιτικές συγκλίνουν σε πολλές πτυχές, ενισχύοντας την συνολική ομοιότητα.
  Ακόμα, τα δύο ζευγάρια όμοιων εντολών με ποσοστά 91,36% και 87,39% υποδεικνύουν ότι αν και η συνολική συμφωνία είναι υψηλή, υπάρχουν ακόμα ορισμένες εντολές που μπορούν να βελτιωθούν.
  Συνολικά, τα αποτελέσματα υποδεικνύουν σημαντική εξέλιξη και πρόοδο στην προσαρμογή της πολιτικής ασφαλείας Επιπέδου 5 προς το ιδανικό σενάριο, καθιστώντας τη συνολικά πολύ συμβατή με τις αρχιτεκτονικές ασφαλείας.



Εικόνα 29 Σύγκριση ιδανικής έναντι εν χρήση πολιτικής ασφαλείας επιπέδου 5

8 Συμπεράσματα Και Μελλοντικές Προτάσεις

8.1 Συμπεράσματα

Στο πλαίσιο της παρούσας έρευνας, αναλύθηκε και προτάθηκε μια καινοτόμος μέθοδος για τη σύγκριση και αξιολόγηση Πολιτικών Ασφαλείας. Η μεθοδολογία αυτή ανοίγει νέους δρόμους στην ανάπτυξη και βελτίωση των συστημάτων ασφαλείας δικτύων, ενισχύοντας την απόδοση και την ασφάλειά τους.

Αρχικά, μέσω της τεχνολογίας gRPC, καταφέραμε να λάβουμε αυτοματοποιημένα την εν ενεργεία πολιτική ασφαλείας μιας επιχείρησης. Παράλληλα, από την ιδανική πολιτική ασφαλείας, διατυπωμένη σε φυσική γλώσσα, δημιουργήσαμε κανόνες Iptables χρησιμοποιώντας τεχνικές επεξεργασίας φυσικής γλώσσας (NLP).

Για τη σύγκριση των δύο πολιτικών ασφαλείας, μετατρέψαμε τους κανόνες σε embeddings και δοκιμάσαμε επτά διαφορετικούς αλγορίθμους NLP για την παραγωγή τους. Βάσει των αποτελεσμάτων μας, ο Universal Sentence Encoder (USE) αποδείχθηκε ως ο πλέον αποτελεσματικός αλγόριθμος για τη μετατροπή των κανόνων σε embeddings. Η μετρική cosine χρησιμοποιήθηκε για τη σύγκριση των embeddings, παρέχοντας έναν ακριβή τρόπο για την εκτίμηση του ποσοστού ομοιότητας.

Τέλος, παρέχοντας πληροφορίες για την αναγνώριση όμοιων και διαφορετικών κανόνων, καθώς και ποσοστά ομοιότητας μεταξύ των κανόνων, οπτικοποιήσαμε τα αποτελέσματα για μια πιο αποτελεσματική και εύχρηστη σύγκριση. Η μεθοδολογία αυτή αναδεικνύει την πρακτική αξία της συγκεκριμένης προσέγγισης στην ενίσχυση της κυβερνοασφάλειας.

8.2 Μελλοντικές Προτάσεις

Παρά τις επιτυχίες, αναδείχθηκαν πολλές προκλήσεις για μελλοντικές βελτιώσεις, ιδιαίτερα στην ακριβή μετατροπή των κανόνων σε φυσική γλώσσα. Αυτές οι βελτιώσεις ανοίγουν το

δρόμο για μελλοντική έρευνα, η οποία μπορεί να επικεντρωθεί στη βελτίωση της μεθοδολογίας, την εξέλιξη των αλγορίθμων NLP, και την περαιτέρω ανάπτυξη αυτοματοποιημένων εργαλείων αξιολόγησης όσων αφορά την μετατροπή πολιτικών ασφαλείας από φυσική γλώσσα σε κανόνες iptables.

Πιο συγκεκριμένα η ενσωμάτωση προηγμένων μεθόδων συντακτικής ανάλυσης (syntactic analysis) θα μπορούσε να αυξήσει την ακρίβεια και την αποδοτικότητα της διαδικασίας μετατροπής λαμβάνοντας υπόψιν την γραμματική δομή της κάθε πρότασης. Επιπλέον, η βελτιωμένη αναγνώριση οντοτήτων (entity recognition) θα επιτρέψει μια πιο λεπτομερή και στοχευμένη εξαγωγή σημαντικών στοιχείων από τις πολιτικές ασφαλείας όπως αναγνώριση πρωτοκόλλων και IP διευθύνσεων. Αυτό ισχύει ιδιαίτερα σε περιπτώσεις όπου η πολιτική ασφαλείας περιγράφεται μέσω πιο σύνθετων δομών φυσικής γλώσσας όπως για παράδειγμα ενός ενιαίου κειμένου με πολλές διαφορετικές τεχνικές ορολογίες[31]. Παράλληλα, η ενσωμάτωση μιας λειτουργίας autocorrect θα μπορούσε να είναι επωφελής. Το autocorrect θα βοηθούσε στην αυτόματη διόρθωση ορθογραφικών και γλωσσικών λαθών, βελτιώνοντας έτσι τη συνολική ποιότητα και επαγγελματική εμφάνιση των πολιτικών ασφαλείας που δίδονται από την εταιρία σε φυσική γλώσσα. Αυτή η λειτουργία θα ήταν ιδιαίτερα χρήσιμη σε αυτά τα κείμενα με εξειδικευμένη ορολογία, όπου τα λάθη μπορεί να έχουν σημαντικές συνέπειες για την ακρίβεια και την ερμηνεία των πληροφοριών.

Εκτός από τις προτεινόμενες βελτιώσεις στη συντακτική ανάλυση καθώς και στην αναγνώριση οντοτήτων, μια σημαντική πτυχή που πρέπει να ληφθεί υπόψη είναι η ταχεία εξέλιξη των αλγορίθμων NLP που παράγουν embeddings. Λόγω αυτής της δυναμικής εξέλιξης, είναι πιθανό ότι σε λίγα χρόνια θα έχουμε στη διάθεσή μας αλγορίθμους που θα υπερβαίνουν σημαντικά τις σημερινές δυνατότητες στην ακρίβεια και την αποδοτικότητα με αποτέλεσμα να επιτρέψουν ακόμα πιο ακριβείς και αποδοτικές συγκρίσεις των πολιτικών ασφαλείας[52]. Η επικείμενη εξέλιξη στους αλγορίθμους NLP που παράγουν embeddings αναμένεται να φέρει σημαντικές βελτιώσεις στις διαδικασίες μετατροπής και ανάλυσης των πολιτικών ασφαλείας. Αυτή η πρόοδος θα επιτρέψει την ακριβέστερη και αποδοτικότερη ερμηνεία των δεδομένων, οδηγώντας σε αυξημένη ακρίβεια στα ποσοστά σύγκρισης. Καθώς οι τεχνολογίες εξελίσσονται, ανοίγονται νέες δυνατότητες για την ανάπτυξη και βελτίωση εργαλείων ασφαλείας, καθώς και για την πιο αποδοτική αυτοματοποίηση των συστημάτων ασφαλείας[53].

Με την περαιτέρω ανάπτυξη και ενσωμάτωση αυτών των βελτιώσεων, η έρευνα στον τομέα της κυβερνοασφάλειας θα μπορούσε να φτάσει σε νέα επίπεδα αποδοτικότητας και ακρίβειας, ανοίγοντας τον δρόμο για πιο σύγχρονα και αποδοτικά εργαλεία αυτοματοποίησης της σύγκρισης και βελτίωσης των πολιτικών ασφαλείας σε έναν περιβάλλον όπου αλλάζει συνεχώς.

9 Βιβλιογραφία

[1] J. Lau, ‘State of Cybersecurity 2023: Navigating Current and Emerging Threats’. 2023. [Έκδοση σε ψηφιακή μορφή]. Διαθέσιμο στο: https://www.isaca.org/resources/news-and-trends/isaca-now-blog/2023/state-of-cybersecurity-2023-navigating-current-and-emerging-threats

[2] E. Ucar και E. Ozhan, ‘The Analysis of Firewall Policy Through Machine Learning and Data Mining’, *Wirel. Pers. Commun.*, τ. 96, τχ. 2, σσ. 2891–2909, Σεπτεμβρίου 2017, doi: 10.1007/s11277-017-4330-0.

[3] C. Diekmann, L. Hupel, J. Michaelis, M. Haslbeck, και G. Carle, ‘Verified iptables Firewall Analysis and Verification’, *J. Autom. Reason.*, τ. 61, τχ. 1–4, σσ. 141–189, 2017, doi: 10.1007/s10817-017-9445-1.

[4] P. S. Rivera, Z. Fei, και J. Griffioen, ‘POLANCO: Enforcing Natural Language Network Policies’, στο *Proceedings of the 2020 29th International Conference on Computer Communications and Networks (ICCCN)*, Honolulu, HI, USA, 2020. doi: 10.1109/ICCCN49398.2020.9209748.

[5] P. Shi, ‘Improving Network Policy Enforcement Using Natural Language Processing and Programmable Networks’. 2022. doi: 10.13023/etd.2022.365.
[6] P. Shi, Y. Song, Z. Fei, και J. Griffioen, ‘Checking Network Security Policy Violations via Natural Language Questions’, στο *Proceedings of the 2021 International Conference on Computer Communications and Networks (ICCCN)*, 2021.
[7] M. Narouei, H. Takabi, και R. Nielsen, ‘Automatic Extraction of Access Control Policies from Natural Language Documents’, *IEEE Trans. Dependable Secure Comput.*, τ. 17, τχ. 3, σσ. 506–517, 2020.
[8] U. Ünal και H. Dağ, ‘AnomalyAdapters: Parameter-Efficient Multi-Anomaly Task Detection’, *IEEE Access*, 2022, doi: 10.1109/ACCESS.2022.3141161.
[9] P. Huff και Q. Li, ‘Towards Automated Assessment of Vulnerability Exposures in Security Operations’, στο *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, 2021.
[10] L. Ceragioli, L. Galletta, και M. Tempest, ‘From Firewall to Functions and Back’, στο *Italian Conference on Cybersecurity*, 2019.
[11] D. Koutras, C. Grigoriadis, M. Papadopoullos, P. Kotzanikolaou, και C. Douligeris, ‘Automating environmental vulnerability analysis for network services’, στο *2022 IEEE Symposium on Computers and Communications (ISCC)*, Rhodes, Greece: IEEE, Ιουνίου 2022, σσ. 1–7. doi: 10.1109/ISCC55528.2022.9912946.
[12] N. Limanova και E. Tretyakov, ‘Iptables for Security of Linux-based Information Networks’, *Bull. Sci. Pract.*, 2022, doi: 10.33619/2414-2948/84/44.
[13] C. Ikerionwu και V. Nwachukwu, ‘An Enhanced Model for Mitigating DDos Attacks on Linux Servers using IPTables and Bash scripts’, *Int. J. Comput. Appl. Technol.*, 2021.
[14] A. Girretti, ‘Understanding the gRPC Specification’, στο *SpringerLink*, 2022. [Έκδοση σε ψηφιακή μορφή]. Διαθέσιμο στο: https://link.springer.com/chapter/10.1007/978-1-4842-8008-9_3
[15] V. Manca και V. Bonnici, *Introduction to Python*, τ. 48. 2023. doi: 10.1007/978-3-031-44501-9_7.
[16] T. Amaratunga, ‘NLP Through the Ages, Understanding Large Language Models’, 2023, σσ. 9–54. doi: 10.1007/979-8-8688-0017-7_2.
[17] P. M. Nadkarni, L. Ohbo-Machado, και W. W. Chapman, ‘Natural language processing: an introduction’, *J. Am. Med. Inform. Assoc.*, τ. 18, τχ. 5, σσ. 544–551, 2011, doi: 10.1136/amiajnl-2011-000464.
[18] A. Vazquez, ‘Firewalls’, στο *SpringerLink*, 2016. [Έκδοση σε ψηφιακή μορφή]. Διαθέσιμο στο: https://link.springer.com/chapter/10.1007/978-1-4842-2379-6_10
[19] O. G. Yalcin, *Feedforward Neural Networks*. 2021.
[20] R. Lee, ‘N-Gram Language Model’, στο *SpringerLink*, 2023. [Έκδοση σε ψηφιακή μορφή]. Διαθέσιμο στο: https://link.springer.com/chapter/10.1007/978-981-99-1999-4_2
[21] J. Xiao και Z. Zou, ‘Research Progress of RNN Language Model’, στο *IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA)*, 2020, σσ. 1285–1288. doi: 10.1109/ICAICA50127.2020.9182390.
[22] G. B. Houdt, C. Mosquera, και G. Napoles, ‘A Review on the Long Short-Term Memory Model’, *Artif. Intell. Rev.*, τ. 53, τχ. 1, σσ. 5929–5955, 2020, doi: 10.1007/s10462-020-09838-1.
[23] K. Rahul, R. K. Banyal, P. Goswami, και V. Kumar, ‘Machine learning algorithms for big data analytics’, *Adv. Intell. Syst. Comput.*, τ. 1227, σσ. 359–367, 2021, doi: 10.1007/978-981-15-6876-3_27.
[24] S. Albawi, T. A. Mohammed, και S. Al-Zawi, ‘Understanding of a convolutional neural networks’, στο *International Conference on Engineering and Technology (ICET)*, 2017, σσ. 1–6. doi: 10.1109/ICENGTECHNOL.2017.8308186.
[25] R. Shi και L. Niu, ‘A brief survey on Capsule Network’, στο *IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, 2020, σσ. 682–686. doi: 10.1109/WIIAT50758.2020.00103.
[26] S. Chandar *κ.ά.*, ‘An Autoencoder Approach to Learning Bilingual Word Representations’, στο *Advances in Neural Information Processing Systems*, 2014, σσ. 1853–1861.

[27] D. Hu, 'An Introduction Survey on Attention Mechanisms in NLP Problems', 2020.
[28] Y. Keneshloo, T. Shi, N. Ramakrishnan, και C. K. Reddy, 'Deep Reinforcement Learning for Sequence-to-sequence Models', *IEEE Trans. Neural Netw. Learn. Syst.*, τ. 31, τχ. 7, σσ. 2469–2489, 2020, doi: 10.1109/TNNLS.2019.2929141.
[29] T. Agarwal, J. Jangid, και G. Kumar, 'Transformers and Natural Language Processing: A Recent Development', *Tuijin JishuJournal Propuls. Technol.*, τ. 44, τχ. 1, σσ. 140–143, 2023, doi: 10.52783/tjjpt.v44.i1.2225.
[30] J. K. Tripathy *κ.ά.*, 'Comprehensive analysis of embeddings and pre-training in NLP', 2021, doi: 10.1016/j.cosrev.2021.100433.
[31] R. P. Lebret, 'Word Embeddings for Natural Language Processing', PhD Thesis, Lausanne, EPFL, 2016. doi: 10.5075/epfl-thesis-7148.
[32] Y. Cha και Y. Lee, 'Advanced sentence-embeddings method considering token importance based on explainable artificial intelligence and text summarization model', *Neurocomputing*, τ. 564, σ. 126987, 2023, doi: 10.1016/j.neucom.2023.126987.
[33] Y. Kim, Y. Jernite, D. Sontag, και A. Rush, 'Character-Aware Neural Language Models'. 2016.
[34] K. Chen, R. Wang, M. Utiyama, και E. Sumita, 'Recurrent Positional Embedding for Neural Machine Translation', στο *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019, σσ. 1361–1367. doi: 10.18653/v1/D19-1139.
[35] I. Makarov, D. Kiselev, N. Nikitinsky, και L. Subelj, 'Survey on graph embeddings and their applications to machine learning problems on graphs', *PeerJ Comput. Sci.*, τ. 7, 2021, doi: 10.7717/peerj-cs.357.
[36] D. Cera και others, 'Universal Sentence Encoder', στο *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, Brussels, Belgium: Association for Computational Linguistics, 2018, σσ. 169–174. doi: 10.18653/v1/D18-2029.
[37] Y. Liu και others, 'RoBERTa: A Robustly Optimized BERT Pretraining Approach'. 2019.
[38] J. Devlin, M.-W. Chang, K. Lee, και K. Toutanova, 'BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding', στο *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Minneapolis, Minnesota: Association for Computational Linguistics, 2019, σσ. 4171–4186. doi: 10.18653/v1/N19-1423.
[39] K. Song και others, 'MPNet: Masked and Permuted Pre-training for Language Understanding'. 2020. [Έκδοση σε ψηφιακή μορφή]. Διαθέσιμο στο: https://proceedings.neurips.cc/paper_files/paper/2020/file/c3a690be93aa602ee2dc0ccab5b7b67e-Paper.pdf
[40] W. Wang και others, 'MiniLM: Deep Self-Attention Distillation for Task-Agnostic Compression of Pre-Trained Transformers'. 2020. [Έκδοση σε ψηφιακή μορφή]. Διαθέσιμο στο: https://proceedings.neurips.cc/paper_files/paper/2020/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf
[41] A. Mastropaolo και others, 'Studying the Usage of Text-To-Text Transfer Transformer to Support Code-Related Tasks', στο *2021 IEEE/ACM 43rd International Conference on Software Engineering (ICSE)*, 2021. doi: 10.1109/ICSE43902.2021.00041.
[42] E. Garcia, 'Cosine Similarity Tutorial'. 2015.
[43] P. Schober, C. Boer, και L. A. Schwarte, 'Correlation Coefficients: Appropriate Use and Interpretation', *Anesth Analg*, τ. 126, τχ. 5, σσ. 1763–1768, 2018, doi: 10.1213/ANE.0000000000002864.
[44] G. Arfaoui και others, 'The Privacy of the TLS 1.3 Protocol', 2019.
[45] D. Koutras, P. Dimitrellos, P. Kotzanikolaou, και C. Douligeris, 'Automated WiFi Incident Detection Attack Tool on 802.11 Networks', στο *2023 IEEE Symposium on Computers and Communications (ISCC)*, Gammarth, Tunisia: IEEE, Ιουλίου 2023, σσ. 464–469. doi: 10.1109/ISCC58397.2023.10218077.

[46] A. Singh, *Socket Programming with Python*. 2019.
[47] C. Chapman και K. T. Stolee, 'Exploring regular expression usage and context in Python', στο *Proceedings of the 25th International Symposium on Software Testing and Analysis*, 2016, σσ. 282–293. doi: 10.1145/2931037.2931073.
[48] B. Pang, E. Nijkamp, και Y. N. Wu, 'Deep Learning With TensorFlow: A Review', *J. Educ. Behav. Stat.*, τ. 45, τχ. 2, σσ. 227–248, 2020, doi: 10.3102/1076998619872761.
[49] D. Rao και B. McMahan, *Natural Language Processing with PyTorch: Build Intelligent Language Applications Using Deep Learning*. O'Reilly Media Inc., 2019.
[50] E. Dabbas, *Interactive Dashboards and Data Apps with Plotly and Dash: Harness the Power of a Fully Fledged Frontend Web Framework in Python*. Packt Publishing Ltd., 2021.
[51] J. Bernard, 'Python Data Analysis with pandas', στο *DOI Book Series*, 2016, σσ. 37–48. doi: 10.1007/978-1-4842-0241-8_5.
[52] Mrs. N. S M, 'The Future of Natural Language Processing: A Survey of Recent Advances and Emerging Trends', *J. Scholast. Eng. Sci. Manag.*, τ. 2, τχ. 6, σσ. 26–35, 2023, doi: 10.5281/zenodo.8243058.
[53] C. Medoh και A. Telukdarie, 'The Future of Cybersecurity: A System Dynamics Approach', *Procedia Comput. Sci.*, τ. 200, σσ. 318–326, 2022, doi: 10.1016/j.procs.2022.01.230.