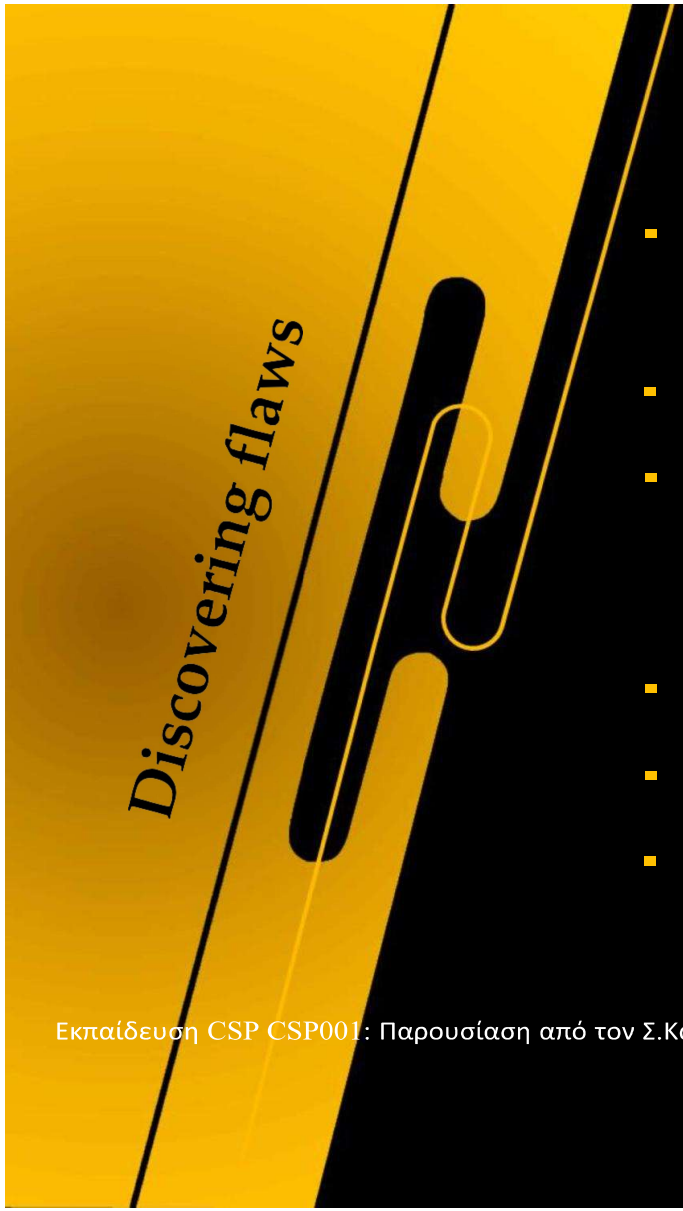
Ασφαλής ανάπτυξης λογισμικού υγειονομικής περίθαλψης

CSP009

ΠΑΡΟΥΣΙΑΣΗ ΑΠΟ: ΣΤΥΛΙΑΝΟ ΚΑΡΑΓΙΑΝΝΗ (PDMFC, ΠΟΡΤΟΓΑΛΙΑ)



Στατική και δυναμική δοκιμή ασφάλειας εφαρμογών (SAST και DAST)

Ανακαλύπτοντας αδυναμιών

- Τα εργαλεία στατικού ελέγχου ασφάλειας εφαρμογών (SAST) αναλύουν τον πηγαίο κώδικα προγραμματισμού για ευπάθειες πριν από τη διαδικασία εκτέλεσης εφαρμογών σε περιβάλλον παραγωγής. Παραδείγματα περιλαμβάνουν τα CodeQL, SonarQube, Checkmarx (λογισμικό στατικής ανάλυσης κώδικα για ασφάλεια) και Fortify Static Code Analyzer.
- Η SAST εντοπίζει ευπάθειες, όπως μη επικυρωμένη είσοδος δεδομένων σε έναν εξυπηρετητή PACS, που δυνητικά μπορεί να οδηγήσει σε έγχυση SQL (π.χ. SELECT * FROM Patients WHERE Name = 'input').
- Η ανάλυση SAST του κώδικα προγραμματισμού ενός εξυπηρετητή PACS μπορεί να αποκαλύψει ευπάθειες όπως το CVE-2019-5649, μια ευπάθεια έγχυσης SQL στη λειτουργία αναζήτησης ασθενών. Η αποκατάσταση περιλαμβάνει την εξυγίανση των εισόδων δεδομένων και τη χρήση παραμετροποιημένων ερωτημάτων για την αποτροπή επιθέσεων έγχυσης SQL.

- Δυναμικό Εφαρμογή Ασφάλεια Δοκιμές (DAST) αξιολογεί την εκτέλεση εφαρμογών για Ευπάθειες με προσομοίωση επιθέσεων.
- Παραδείγματα εργαλείων περιλαμβάνουν τα OWASP ZAP, Burp (εργαλείο penetration testing για web εφαρμογές) και Acunetix
- Οι σαρώσεις DAST μπορεί να αποκαλύψουν ευπάθειες όπως η ανεπαρκής επικύρωση εισόδου δεδομένων σε φόρμα σύνδεσης HIS, που δυνητικά οδηγεί σε XSS (π.χ. <script>alert('XSS')</script>).

Το SAST στην πράξη Μερός 1

Σάρωση κώδικα προγραμματισμού GitHub

Το GitHub Code Scanning είναι ενσωματωμένο εργαλείο SAST που παρέχεται από το GitHub, προσφέροντας ακατάπαυστη λειτουργία με τη ροή ανάπτυξης λογισμικού.

Διαδικασία

- Ενεργοποιήστε τη σάρωση κώδικα προγραμματισμού GitHub στις ρυθμίσεις του αποθετηρίου.
- Ρυθμίστε τη ροή εργασιών σάρωσης ώστε να ενεργοποιεί αυτόματα σαρώσεις σε μεταβιβάσεις κώδικα προγραμματισμού ή στο pull requests(αίτημα για συγχώνευση κώδικα).
- Προσαρμόστε τις ρυθμίσεις ανάλυσης και καθορίστε σύνολα κανόνων ασφαλείας προσαρμοσμένα στις απαιτήσεις των εφαρμογών υγειονομικής περίθαλψης.
- Επανεξέταση των αποτελεσμάτων της σάρωσης στο περιβάλλον εργασίας του GitHub, συμπεριλαμβανομένων των εντοπισμένων ευπαθειών, των επιπέδων σοβαρότητας και των θέσεων του επηρεαζόμενου κώδικα προγραμματισμού.
- Συνεργασία με τα μέλη της ομάδας για την κατευθυνόμενη ταξινόμηση των ζητημάτων στο πλαίσιο του pull requests ή των issues.

GitHub Code Scanning

Το SAST στην πράξη Μέρος 2

Σάρωση κώδικα προγραμματισμού GitHub

Οφέλη

- Η «συνεργασία» με τα αποθετήρια GitHub απλοποιεί τη διαδικασία ανάπτυξης, επιτρέποντας στους προγραμματιστές να διευθετήσουν θέματα ασφάλειας παράλληλα με τις αλλαγές στον κώδικα προγραμματισμού.
- Παρέχει αξιοποιήσιμες πληροφορίες για ευπάθειες, επιτρέποντας στους προγραμματιστές να δίνουν προτεραιότητα και να αποκαθιστούν αποδοτικά τα προβλήματα.
- Υποστηρίζει ένα ευρύ φάσμα γλωσσών προγραμματισμού και πλαισίων λογισμικού που χρησιμοποιούνται συνήθως στην ανάπτυξη λογισμικού υγειονομικής περίθαλψης.

Διαδικασία

- Μετά από μια αποστολή κώδικα (push) ή ένα αίτημα έλξης (pull request), το GitHub Code Scanning ενεργοποιεί αυτόματα έναν έλεγχο του κώδικα. Ο έλεγχος εντοπίζει ευπάθειες όπως SQL injection, XSS ή μη ασφαλείς μεθόδους ταυτοποίησης μέσα στον πηγαίο κώδικα.

- Οι εντοπισμένες ευπάθειες αναφέρονται μέσα από το περιβάλλον του GitHub, επιτρέποντας στους προγραμματιστές να τις εξετάσουν και να τις αντιμετωπίσουν άμεσα.
- Οι προγραμματιστές συνεργάζονται για την αποκατάσταση των ευπαθειών παράγοντας αλλαγές στον κώδικα προγραμματισμού και υποβολή αιτημάτων για αναθεώρηση.
- Μόλις μετριαστούν οι ευπάθειες, οι αλλαγές στον κώδικα προγραμματισμού συγχωνεύονται πίσω στον κύριο κλάδο(main branch), εξασφαλίζοντας ασφαλή διαδικασία μεταφοράς κώδικα από ανάπτυξη σε χρήση παραγωγής.

DAST στην πράξη Μέρος

Ανακαλύπτοντας αδυναμίες

OWASP ZAP (Zed Attack Proxy)

- Το OWASP ZAP είναι ένας σαρωτής ασφαλείας εφαρμογών ιστού ανοικτού κώδικα που χρησιμοποιείται για να τον εντοπισμό Ευπαθειών σε διαδικτυακές εφαρμογές.

Διαδικασία

- Ρυθμίστε το OWASP ZAP ώστε να πράξει ως διακομιστής μεσολάβησης μεταξύ του φυλλομετρητή (browser - προγράμμα περιήγησης) και της εφαρμογής-στόχος.
- Ρυθμίστε αυτοματοποιημένες σαρώσεις ή εξερευνήστε χειροκίνητα την εφαρμογή για τον εντοπισμό ευπαθειών.
- Αναλύστε τα αποτελέσματα της σάρωσης και δημιουργήστε αναφορές που περιγράφουν λεπτομερώς τις εντοπισμένες ευπάθειες.

Οφέλη

- Παρέχει φιλικό προς το χρήστη περιβάλλον για αυτοματοποιημένες και χειροκίνητες δοκιμές.
- Προσφέρει προηγμένες δυνατότητες για δοκιμές ταυτοποίησης και διαχείριση συνεδριών.

DAST in Practice Pt.2

DAST στην πράξη Μέρος 2

DAST στην πράξη Μέρος 2

Burp Suite (εργαλείο penetration testing για web εφαρμογές)

Το Burp Suite είναι ευρέως χρησιμοποιούμενο εμπορικό εργαλείο DAST για εφαρμογές ιστού. δοκιμές ασφαλείας.

Διαδικασία

- Ρυθμίστε το Burp Suite για την υποκλοπή και την ανάλυση αιτημάτων HTTP και απαντήσεις.
- Αποδώστε αυτοματοποιημένες σαρώσεις ή εξερευνήστε χειροκίνητα την εφαρμογή για να εντοπίσετε ευπάθειες.
- Αξιοποιήστε τα ενσωματωμένα εργαλεία για την επιβεβαίωση και την εκμετάλλευση ευπαθειών.

Οφέλη

- Προσφέρει εκτενή κάλυψη ευπαθειών ιστού, συμπεριλαμβανομένων των OWASP Top 10.
- Παρέχει προηγμένα χαρακτηριστικά για δοκιμές διείσδυσης και εκμετάλλευση ευπαθειών.

CWE and Code Scanning

Απαρίθμηση κοινών αδυναμιών (CWE)

CWE και σαρώσεις κώδικα προγραμματισμού

Η απαρίθμηση κοινών αδυναμιών (Common Weakness Enumeration - CWE) είναι ένας κατάλογος αδυναμιών λογισμικού και υλικού που αναπτύχθηκε από την κοινότητα. Η CWE παρέχει μια κοινή γλώσσα για την περιγραφή των ευπαθειών και των αδυναμιών ασφαλείας σε συστήματα λογισμικού.

- Κάθε καταχώρηση CWE περιγράφει έναν συγκεκριμένο τύπο αδυναμίας, όπως υπερχείλιση buffer, Έγχυση SQL ή cross-site scripting (XSS). Οι καταχωρίσεις CWE περιλαμβάνουν λεπτομερείς περιγραφές, παραδείγματα, δυνητικές συνέπειες και στρατηγικές μετριασμού για την αντιμετώπιση της αδυναμίας. Τα εργαλεία σάρωσης κώδικα προγραμματισμού, όπως το GitHub Code Scanning, χρησιμοποιούν συχνά το CWE ως αναφορά για τον εντοπισμό και την ταξινόμηση των ευπαθειών στον πηγαίο κώδικα.
- Με την αντιστοίχιση των εντοπισμένων ευπαθειών σε καταχωρίσεις CWE, τα εργαλεία σάρωσης κώδικα βοηθούν τους προγραμματιστές να κατανοήσουν τη φύση και τη σοβαρότητα των ζητημάτων ασφάλειας στο αποθετήριο κώδικα τους. Η αντιστοίχηση με το CWE δίνει την δυνατότητα στους προγραμματιστές να δίνουν προτεραιότητα στις προσπάθειες αποκατάστασης με βάση τη σοβαρότητα και τον αντίκτυπο των εντοπισμένων αδυναμιών.
- Οι αναφορές σάρωσης κώδικα προγραμματισμού περιλαμβάνουν συνήθως αναγνωριστικά CWE μαζί με τις ευπάθειες που εντοπίστηκαν, παρέχοντας στους προγραμματιστές χρήσιμες πληροφορίες για τη βελτίωση της ασφάλειας του κώδικα.

Code Scanning Pt.3

Επιλογές GitHub Μέρος 1

Ασφάλεια

Το GitHub μπορεί να προσδιορίσει το ποσοστό των διαφορετικών γλωσσών προγραμματισμού που χρησιμοποιούνται σε ένα αποθετήριο. Το GitHub αναλύει τον κώδικα προγραμματισμού σε ένα αποθετήριο και παρέχει πληροφορίες σχετικά με το ποσοστό κάθε γλώσσας προγραμματισμού που χρησιμοποιείται.

- Πολιτική ασφαλείας: Καθορίστε τον τρόπο με τον οποίο οι χρήστες θα πρέπει να αναφέρουν ευπάθειες ασφαλείας για αυτό το αποθετήριο. Ενεργοποιώντας πολιτική ασφαλείας, οι ιδιοκτήτες αποθετηρίων μπορούν να καθορίσουν κατευθυντήριες γραμμές για την αναφορά ευπαθειών ασφαλείας, ενισχύοντας τη συνεργασία και εξασφαλίζοντας έγκαιρες αντιδράσεις σε δυνητικές απειλές.
- Συμβουλές ασφαλείας: Προβολή ή αποκάλυψη συμβουλών ασφαλείας για αυτό το αποθετήριο. Το GitHub επιτρέπει την αποκάλυψη συμβουλών ασφαλείας που σχετίζονται με το αποθετήριο, διασφαλίζοντας τη διαφάνεια και επιτρέποντας στους χρήστες να ενημερώνονται για πιθανούς κινδύνους ασφαλείας.
- Ιδιωτική αναφορά ευπαθειών: Επιτρέπει στους χρήστες να αναφέρουν ιδιωτικά δυνητικές ευπάθειες ασφάλειας. Η ενεργοποίηση της ιδιωτικής αναφοράς ευπαθειών επιτρέπει στους χρήστες να αναφέρουν εμπιστευτικά ζητήματα ασφαλείας, προωθώντας ένα ασφαλές περιβάλλον συνεργασίας και άμεσης επίλυσης των ευπαθειών.

Code Scanning Pt.3

Επιλογές GitHub Μέρος 2

Ασφάλεια

- Ειδοποιήσεις Dependabot: Λάβετε ειδοποίηση όταν κάποια από τις εξαρτήσεις(dependencies) σας έχει ευπάθεια. Οι ειδοποιήσεις του Dependabot ενημερώνουν τους ιδιοκτήτες του αποθετηρίου για ευπάθειες σε εξαρτήσεις, βοηθώντας τους να παραμένουν προληπτικοί στη διαχείριση και ενημέρωση των εξαρτήσεων ώστε να μετριάζονται πιθανοί κίνδυνοι ασφαλείας.
- Ειδοποιήσεις σάρωσης κώδικα προγραμματισμού: Αυτόματος εντοπισμός κοινών ευπαθειών και σφαλμάτων κώδικα προγραμματισμού. Η δυνατότητα σάρωσης κώδικα του GitHub ανιχνεύει αυτόματα κοινές ευπάθειες και λάθη κώδικα στην βάση κώδικα του αποθετηρίου, ενδυναμώνοντας τους προγραμματιστές να εντοπίζουν και να διευθετούν δυνητικές απειλές ασφάλειας νωρίς στη διαδικασία ανάπτυξης λογισμικού.
- Μυστικές ειδοποιήσεις σάρωσης: Ειδοποιηθείτε όταν ένα μυστικό μεταφερθεί σε αυτό το αποθετήριο. Οι ειδοποιήσεις σάρωσης μυστικών ειδοποιούν τους ιδιοκτήτες αποθετηρίου όταν ευαίσθητες πληροφορίες, όπως διακριτικά πρόσβασης ή διαπιστευτήρια, προωθούνται κατά λάθος στο αποθετήριο, βοηθώντας στην αποτροπή ανεξουσιοδότητης πρόσβασης και παραβίασης δεδομένων.

Code Scanning Pt.1

Σάρωση κώδικα προγραμματισμού Μέρος 1

Σάρωση κώδικα προγραμματισμού GitHub Μέρος 1

Οι ροές εργασίας σάρωσης κώδικα στο GitHub παρέχουν αυτοματισμούς για την ολοκλήρωση της στατικής ανάλυσης κώδικα στον κύκλο ζωής της ανάπτυξης λογισμικού. Ακολουθεί μια περιγραφή σε κουκκίδες:

Διαμορφώσεις:

- Καθορισμός ροών εργασίας σαρώσεων κώδικα προγραμματισμού χρησιμοποιώντας αρχεία διαμόρφωσης YAML(π.χ., .github/workflows/code-scanning.yml).
- Καθορίστε συνθήκες ενεργοποίησης για την εκκίνηση ελέγχων κώδικα, όπως κατά την αποστολή (push), τη δημιουργία pull request ή σε προγραμματισμένα χρονικά διαστήματα.

Επιλογή εργαλείων:

- Επιλέξτε το(τα) επιθυμητό(-ά) εργαλείο(-α) σάρωσης κώδικα προγραμματισμού που θα χρησιμοποιηθεί(-ουν) κατά την ροή εργασίας
- Η σάρωση κώδικα προγραμματισμού GitHub «συνεργάζεται» εγγενώς με τα αποθετήρια GitHub, απλοποιώντας την εγκατάσταση και τη διαμόρφωσή της.

Βήματα ροής εργασίας:

- Καθορισμός βημάτων ροής εργασιών για την εκτέλεση σαρώσεων κώδικα προγραμματισμού με τη χρήση επιλεγμένων εργαλείων.
- Διαρθρώνετε παραμέτρους όπως συχνότητα σάρωσης, βάθος σάρωσης και πολιτικές σάρωσης.

Code Scanning Pt.2

Σάρωση κώδικα προγραμματισμού Μέρος 2

Σάρωση κώδικα προγραμματισμού GitHub Μέρος 2

Εκτέλεση σάρωσης:

- Αυτοματοποιημένοι έλεγχοι κώδικα ενεργοποιούνται βάσει προκαθορισμένων συνθηκών ροής εργασίας.
- Οι έλεγχοι αναλύουν τον πηγαίο κώδικα για ευπάθειες ασφαλείας, σφάλματα προγραμματισμού και άλλα προβλήματα.

Ανάλυση αποτελεσμάτων:

- Μετά την ολοκλήρωση της , δημιουργούνται αποτελέσματα σάρωσης τα οποία είναι διαθέσιμα μεσώ του περιβάλλον εργασίας του GitHub.
- Οι προγραμματιστές μπορούν να έχουν πρόσβαση σε λεπτομερείς αναφορές που επισημαίνουν τις εντοπισμένες ευπάθειες, τα επίπεδα σοβαρότητας και τις πληγείσες θέσεις στον κώδικα.

Ειδοποιήσεις και αναθεώρηση:

- Προαιρετικά, διαμορφώστε ειδοποιήσεις για να ενημερώνουν τους προγραμματιστές και τα σχετικά ενδιαφερόμενα μέρη για τα αποτελέσματα των ελέγχων.
- Ανασκόπηση των ευρημάτων σάρωσης μέσα σε pull requests, issues ή σε αφιερωμένους πίνακες ελέγχου (dashboards), ανάλογα με τη διαμόρφωση του workflow.

Code Scanning Pt.3

Σάρωση κώδικα προγραμματισμού Μέρος 3

Ψήφισμα

Αποκατάσταση και Μετριασμός:

- Συνεργαστείτε με τα μέλη της ομάδας για την άμεση αντιμετώπιση των εντοπισμένων ευπαθειών.

- Οι προγραμματιστές μπορούν να ιεραρχούν και να επιδιορθώνουν τα ζητήματα απευθείας μέσω της διεπαφής του GitHub, αξιοποιώντας pull requests ή συστήματα παρακολούθησης θεμάτων (issue tracking).

Συνεχής Βελτίωση:

- Επανασχεδιάστε περιοδικά τα workflows σάρωσης κώδικα βάσει ανατροφοδότησης και των εξελισσόμενων απαιτήσεων του έργου.
- Προσαρμόστε τις πολιτικές σάρωσης, τις ρυθμίσεις των εργαλείων και τα σημεία ενσωμάτωσης για βέλτιστη ασφάλεια κώδικα.

Ενσωμάτωση με CI/CD:

- Ενσωματώστε απρόσκοπτα τα workflows σάρωσης κώδικα με τα υπάρχοντα pipelines CI/CD για ολοκληρωμένη επικύρωση κώδικα και αυτοματοποίηση ανάπτυξης.
- Διασφαλίστε ότι τα αποτελέσματα της σάρωσης ενημερώνουν τη διαδικασία έκδοσης λογισμικού, αποτρέποντας την εισαγωγή ευπαθειών σε περιβάλλοντα παραγωγής.

CodeQL and Workflows

Ροές εργασίας του GitHub στη σάρωση ασφαλείας

Ορισμός

Οι ροές εργασίας του GitHub για την ασφάλεια αναφέρονται σε διαδικασίες αυτοματισμού ή ακολουθίες ενεργειών που ενεργοποιούνται από γεγονότα σε ένα αποθετήριο GitHub, με στόχο την ενίσχυση της ασφάλειας μέσω του εντοπισμού ευπαθειών, σφαλμάτων κώδικα προγραμματισμού ή διαρροών ευαίσθητων πληροφοριών στην κωδικοποιημένη βάση.

- Οι ροές εργασίας του GitHub βελτιστοποιούν τις πρακτικές ασφαλείας, αυτοματοποιώντας τους ελέγχους ασφαλείας, διασφαλίζοντας την ποιότητα του κώδικα προγραμματισμού και αποτρέποντας παραβιάσεις ασφαλείας πριν από τη διαδικασία μεταφοράς κώδικα από ανάπτυξη σε χρήση παραγωγής.
- Οι ροές εργασίας του GitHub μπορούν να διαμορφωθούν ώστε να εκτελούν διάφορα εργαλεία σάρωσης ασφαλείας, όπως το CodeQL, για να αναλύουν τον κώδικα προγραμματισμού για δυνητικές ευπάθειες, σφάλματα ασφαλείας ή ζητήματα συμμόρφωσης.
- Το CodeQL είναι ένα ισχυρό εργαλείο στατικής ανάλυσης που παρέχεται από το GitHub για τον εντοπισμό ευπαθειών ασφάλειας και σφαλμάτων κώδικα προγραμματισμού σε αποθέματα κώδικα. Επιτρέπει στους προγραμματιστές να εγγραφούν προσαρμοσμένα ερωτήματα για την ανάλυση μοτίβων κώδικα προγραμματισμού και τον εντοπισμό δυνητικών κινδύνων ασφάλειας.

Code Scanning

Σάρωση κώδικα προγραμματισμού με χρήση του GitHub

Πληροφορίες

- Συνεχής σάρωση: Η σάρωση κώδικα προγραμματισμού του GitHub παρέχει συνεχή ανάλυση του κώδικα για ευπάθειες, διασφαλίζοντας την προληπτική ανίχνευση και τον μετριασμό των κινδύνων ασφάλειας.
- Η σάρωση κώδικα προγραμματισμού κατηγοριοποιεί τις ευπάθειες σε διαφορετικά επίπεδα σοβαρότητας. Προσδιορισμός του σχετικού αρχείου πηγής και της γραμμής κώδικα προγραμματισμού όπου εντοπίστηκαν δυνητικές ευπάθειες.
- Χρονοπρογραμματισμένες σαρώσεις: επιτρέποντας τακτικές και έγκαιρες επιθεωρήσεις ασφαλείας του αποθετηρίου κώδικα.
- Λεπτομέρειες:Εκτεταμένη εξήγηση του θέματος ασφάλειας,βοήθεια σε κατανόηση και επίλυση.
- Αντιστοίχιση CWE: Οι ευπάθειες αντιστοιχίζονται με τα πρότυπα Common Weakness Enumeration (CWE), παρέχοντας μια τυποποιημένη ταξινόμηση των αδυναμιών στο λογισμικό.

Σας ευχαριστώ

Παρουσιαστής: Στυλιανός Καραγιάννης (PDMFC, Πορτογαλία)

Παρακαλούμε στείλτε όλες τις ερωτήσεις στη διεύθυνση:
stylianos.karagiannis@pdmfc.com