Διαφάνεια 1:

Καλησπέρα σας ονομάζομαι Κωνσταντίνος Γκανετσίδης και θα ήθελα να σας μιλήσω για το hacking με την γλώσσα assembly.
Το Assembly Hacking ή Asm Hacking όπως αλλιώς ονομάζεται πρόκειται για την διαδικασία κατά την οποία κάποιος τροποποιεί τον κώδικα μίας εφαρμογής ή εισάγει ένα νέο κομμάτι εμβόλιμου κώδικα σε αυτήν.

Η εφαρμογή του Assembly Hacking βρίσκει εύφορο έδαφος σε οποιοδήποτε είδος προγράμματος και application.Κυρίως όμως έχει εφαρμοστεί στο τομέα του gaming όπου είναι επίσης γνωστό ως ROM Hacking διότι ο Hacker τροποποιεί το ROM Image του παιχνιδιού.Στον τομέα αυτό κυκλοφορούν μάλιστα και πολλά έτοιμα και αρκετά γνωστά hacks όπως είναι τα
Zelda:Parrallel Worlds
Pokemon Ashgrey verion
Super Mario:Lunar Magic World editor

Διαφανεια 2-3:
Θεωρείται ένα από τα πιο δύσκολα και απαιτητικά ήδη Hacking γιατί προυποθέτει λεπτομερή γνώση της γλώσσας assembly και τη εσωτερικής λειτουργίας του προγράμματος.Οι δυσκολίες όμως δεν τελειώνουν εδώ.Στο ASM Hacking δεν υπάρχει συγκεκριμένο μοτίβο που θα πρέπει να ακολουθήσει κάποιος μιας και ο κώδικας διαφέρει από περίπτωση σε περίπτωση και δεν αρκεί να ξέρεις μόνο μία γλώσσα assembly αλλά αυτήν που χρησιμοποιείται κάθε φορά.Μερικά παραδείγματα βλέπουμε εδώ (MIPS Z80 …).

Οι Developers του πογράμματος προφανώς δεν θέλουν αυτό να τροποποιηθεί, επομένως ένας Assembly hacker πρεπει να ξέρει ακριβώς τι θέλει να πετύχει,ωστε να ξέρει περίπου σε ποιό κομμάτι του κώδικα να ψάξει και να είναι οπλισμένος με αρκετή υπομονή.Τελος εμφανίζεται το πρόβλημα του ROM Expansion όπου όταν κάποιος θέλει να προσθέσει επιπλέον υλικό στον κώδικα ενος παιχνιδιου για παράδειγμα θα πρέπει να αυξήσει το μέγεθος του ROM Image.Το επιπεδο δυσκολίας μιας τέτοιας κινησης ποικοίλει από σχετικά απλό μέχρι σχεδόν αδύνατο ανάλογα με το σύστημα στο οποίο λειτουργεί το παιχνιδι(π.χ. βλ. διαφανεια)

Διαφανεια 4:
Η επιλογή της γλώσσας assembly δε γίνεται τυχαία.
Ο κώδικας του προγράμματος προφανώς δεν διατίθεται ελεύθερα και για αυτό γίνεται η επιλογή της γλώσσας asssembly.Η γλώσσα αυτή είναι χαμηλού επιπέδου και λειτουργεί πολύ κοντα στο hardware.Επομένως είναι δυνατό να αναλυθεί σε αυτήν το πρόγραμμα με μια διαδικασια που λέγεται Reverse Engineering.Κατα την διαδικασια του reverse engineering το επιθυμητό κομμάτι κώδικα γίνεται decompiled,τροποποιείται κατα βούληση.Στην συνέχεια γίνεται compile του τροποποιημένου κώδικα που αντικαθιστά το προτώτυπο.
Αυτό γίνεται με την χρήση εργαλείων όπως είναι οι dissassemblers(μετατροπή compiled program σε assembly και τροποποίηση του με τη χρήση hex editors ή κάποιου assembler),debuggers(οι debuggers τρέχουν το πρόγραμμα βήμα προς

βήμα και επιτρέπουν στον χρήστη να βλέπει σε κάθε βήμα τι υπάρχει στην μνήμη).
Μερικά παραδείγματα τέτοιων προγραμμάτων για windows είναι τα:
ollydbg
Windbg
Tsearch
Cheat engine
Idapro
και φυσικά πολλά άλλα το καθένα με τα δικά του δυνατά και αδύναμα σημεία.

Διαφανεια 5:
Ένα πάρα πολύ απλό και εύκολο παράδειγμα hacking με assembly είναι η τροποποίηση του κώδικα ενός προγράμματος έτσι ώστε να δέχεται οποιοδήποτε registration key.Αυτο γίνεται με την εντόπιση της εντολής του κώδικα που καλεί το μύνημα του error και της προσπέλασης της με κατάλληλη τροποποιήση της εντολής JMP που υπάρχει σε αυτο.Κάτι τέτοιο θα μπορούσε να είναι της μορφής αυτής.

Διαφανεια 6-8:
Εδώ επίσης έχω κάποια screenshot απο ASM hacking με χρηση εμβόλιμου κώδικα για την τροποποίση του συστήματος καταμέτρησης πόντων στο παιχνίδι pinnball των windows.Τα screenshot αυτα είναι από ένα βίντεο στο youtube το link του οποίου υπάρχει στην βιβλιογραφία μου για όποιον ενδιαφέρεται να δει την ολοκληρωμένη διαδικασια μιας και ο περιορισμός χρόνου δεν μου επιτρέπει να το αναπαράγω κατά βήμα.
Ο εντοπισμος του ζητούμενου σημείου του κώδικα γίνεται με την χρηση καποιων breakpoint k watch points παρακολουθηση δηλαδη της μνήμης και ταυτόχρονο τρέξιμο του προγράμματος στο σημείο που θέλουμε να τροποποιήσουμε(εδώ καταμέτρηση πόντών).
Στο συγκεκριμένο βίντεο χρησημοποιείται το πρόγραμμα Cheat Engine το οποιό προσφέρει μαζί με άλλες ευκολίες και έτοιμα templates για την εισαγωγή νέου κώδικα.
Στα αντίστοιχα slides βλέπουμε τον εντοπισμό της θέσης μνήμη στην οποία αποθηκέυεται το σκορ και τον εμβόλιμο κώδικα που χρησιμοποιείται για την τροποποίηση του.

Διαφάνεια 9:
Συμπερασματα:

Παρόλες τις δυσκολίες που υπάρχουν,τα πάντα είναι δυνατα με το ASM Hacking και οι περιορισμοί αφορουν μόνο αυτούς της ικανότητας του Hacker να κατανοήσει και να τροποποιήσει των κώδικα καθώς και φυσικά τα όρια του software/hardware της πλατφόρμας.Τα επιτυχημένα αποτελέσματα που προκύπτουν είναι ένα νεό πρόγραμμα,βασισμένο στο πρωτότυπο και πλήρως προσαρμοσμένο στις προσδοκίες του Hacker.

Μελοντικές επεκτάσεις:

Πρόκειται για μια τεχνική που δεν πιστεύω ότι θα αντικατασταθεί ή θα ξεπεραστεί σύντομα.Η χρησιμότητα της είναι αναμφισβήτητη,η γέννηση της είναι παράλληλη με την γέννηση του program Hacking και δεδομένης της δημοτικότητας της κάθε χρόνος που περνάει εμφανίζονται εργαλέια που την κάνουν ακόμα πιο προσιτή σε άτομα με λιγότερη εμπειρία

Διαφανεια 10:
Βιβλιογραφια:

Άλλα links
(όπως εμφανίζονται στην διαφάνεια).