ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΡΗΤΗΣ

ΤΜΗΜΑ ΕΠΙΣΤΗΜΗΣ ΥΠΟΛΟΓΙΣΤΩΝ

ΠΑΡΟΥΣΙΑΣΗ / ΕΞΕΤΑΣΗ ΜΕΤΑΠΤΥΧΙΑΚΗΣ ΕΡΓΑΣΙΑΣ

Χρήστου Γεώργιος

Μεταπτυχιακός Φοιτητής

Τμήμα Επιστήμης Υπολογιστών, Πανεπιστήμιο Κρήτης

Επόπτης Μεταπτ. Εργασίας: Καθηγητής Ε. Μαρκάτος

Δευτέρα, 20/2/2017, 15:00

Αίθουσα Τηλεδιάσκεψης Κ206, Τμήμα Επιστήμης Υπολογιστών, Πανεπιστήμιο Κρήτης

"Αρχιτεκτονική Υποστήριξη Ακεραιότητας Ροής Εκτέλεσης"

Η κακόβουλη εκμετάλλευση λογισμικού γίνεται ολοένα και ποιο συχνή, με την εξάπλωση των υπολογιστικών συστημάτων στην καθημερινότητά μας. Τα τελευταία χρόνια, οι επιθέσεις στο λογισμικό γίνονται ποιο εξελιγμένες. Τα αντίμετρα που υπάρχουν στα υπολογιστικά συστήματα, τείνουν να μην προφέρουν αρκετή προστασία. Τα ισχυρά αντίμετρα απαιτούν ενδελεχής ελέγχους που είναι υπολογιστικά ακριβοί.

Ένα ικανό αντίμετρο είναι η Ακεραιότητα Ροής Εκτέλεσης (CFI). Είναι μια πολιτική που αναπτύχθηκε έτσι ώστε να προστατεύει το λογισμικό από επιθέσεις που αλλοιώνουν τη ροή εκτέλεσής του, τη κύρια μέθοδο για επίτευξη τεχνικών επαναχρησιμοποίησης κώδικα, όπως επιστρεφόμενου προγραμματισμού (ROP) και προγραμματισμού με άλματα (JOP). Η ερευνητική κοινότητα πρότεινε αυτή την τεχνική, ως ικανή να αποτρέπει τις επιθέσεις αυτές, με το να ελέγχει ότι κάθε αλλαγή στη ροή εκτέλεσης ενός προγράμματος και προέρχεται από υπολογισμό της νέας διεύθυνσης, καταλήγει σε σωστή διεύθυνση. Η επέκταση συστημάτων με CFI δεν είναι μια ξεκάθαρη διαδικασία, καθώς ο Γράφος ροής ενός προγράμματος δεν μπορεί να καθοριστεί πάντα με ακρίβεια. Ακόμα και σε περιπτώσεις που ο Γράφος ροής είναι πλήρως καθορισμένος, ο ακριβής

έλεγχος ότι οι εντολείς επιστροφής γυρνάνε στη διεύθυνση από όπου έγινε η κλήση, χωρίς τη χρήση μιας προστατευόμενης στοίβας είναι αμφισβητούμενη. Όμως, η ερευνητική κοινότητα αποφεύγει τη χρήση προστατευόμενης στοίβας λόγο της επιβράδυνσης που επιφέρει.

Σε αυτή τη δουλειά, αναγνωρίζουμε τη σημαντικότητα της υλοποίησης μηχανισμών ασφαλείας σε επίπεδο υλικού με σκοπό την ενίσχυση και επιτάχυνσή τους. Δείχνουμε, ότι η υλοποίηση μιας αρχιτεκτονικής με εντολές CFI στο υλικό μαζί με προστατευόμενη μνήμη μέσα στον επεξεργαστή είναι εφικτή και το πρωτότυπο είχε ελάχιστη επιβράδυνση.
Για να υποστηρίξουμε την ιδέα μας, υλοποιήσαμε τις Επεκτάσεις Ελέγχου Ροής Εκτέλεσης (CFIX), τροποποιώντας ένα σύστημα SPARC και αξιολογήσαμε το πρωτότυπο σε πλακέτα FPGA με το να τρέξουμε SPECint προγράμματα μέτρησης επιδόσεων που είχαν εντολές για λεπτομερή έλεγχο ακεραιότητας ροής. Η αξιολόγηση έδειξε ότι τα CFIX μπορούν να προστατεύσουν αποτελεσματικά τις εφαρμογές από επιθέσεις επαναχρησιμοποίησης κώδικα και παράλληλα επιβραδύνουν το σύστημα κατά 1% και αυξάνουν την κατανάλωση ενέργειας κατά 2%, καθιστώντας το σύστημα μας ιδανικό για ενσωματωμένα συστήματα.

Christou Georgios

M.Sc. Thesis

Computer Science Department

University of Crete

Master's Thesis Supervisor: Professor E. Markatos

Monday, 20/2/2017, 15:00
Room K206, Computer Science dept., University of Crete

“Architectural Support for Control FLow Integrity"

ABSTRACT

Abstract Exploitation of software becomes more and more common, as computer systems span across many areas of our lives. Over the recent years, attacks on software become more sophisticated. Deployed countermeasures tend to not provide sufficient protection. Effective countermeasures require thorough checks which are computationally expensive.

One such countermeasure is Control-Flow Integrity (CFI); a policy developed to defend against Control-flow hijacking, the principal method for code-reuse techniques like Return-oriented Programming (ROP) and Jump-oriented Programming (JOP). The community proposed CFI, a technique capable of preventing exploitation by verifying that every (indirect) control-flow transfer points to a legitimate address. Enabling CFI in real world systems is not straightforward, since in many cases the actual Control-flow Graph (CFG) of a program can be only approximated. Even in the case that there is perfect knowledge of the CFG, ensuring that all return instructions will return to their actual call sites, without employing a shadow stack, is questionable. On the other hand, the community has expressed concerns related to significant overheads stemming from deploying a shadow stack.

In this work, we acknowledge the importance of pushing security in the hardware domain, in order to strengthen and accelerate security mechanisms. We project, that implementing a full-featured CFI-enabled Instruction Set Architecture (ISA) in actual hardware with an in-chip secure memory can be efficiently carried out and the prototype experiences negligible overheads. For supporting our case, we implement Control-Flow Integrity Extensions (CFIX) by modifying a SPARC SoC and evaluate the prototype on an FPGA board by running SPECInt benchmarks instrumented with a fine-grained CFI policy. The evaluation shows that CFIX can effectively protect applications from code-reuse attacks, while adding less than 1% runtime overhead and 2% power consumption overhead, making it particularly suitable for embedded systems.