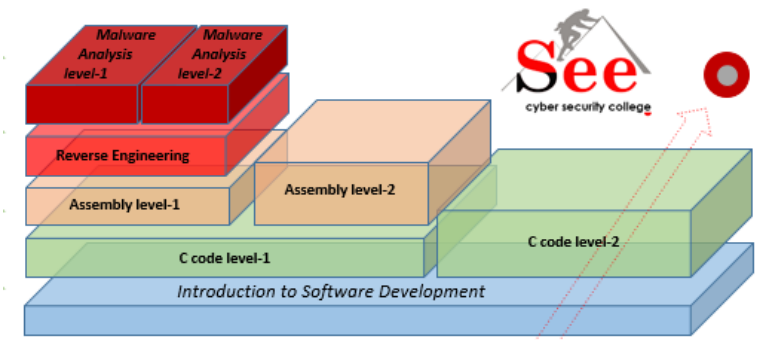
ASSEMBLY

FOR INTEL IA-32, X86-64

LANGUAGE



See
cyber security college

לדעת ASSEMBLY.

הבנת עקרונות ה- low level הנמצאים במעבדי אינטל משמשת כיסוד מובהק להבנת הנדסה לאחור (reverse engineering) של תוכנות, תכנון קומפיילרים, הבנת מערכות הפעלה וניצול פגיעויות במערכות אלו.

בתכנית זו המבוססת בעיקר על התנסות, נלמד את העקרונות הבסיסיים ונתאר את החומרה שקוד אסמבלי משתמש בו. נתאר את הפקודות הנפוצות הנראים כקוד אסמבלי, נלמד אותן לעומקן ונבין מה מתרחש "מאחורי הקלעים".

וְלֹא פָּחוֹת חָשׁוּב: נִלְמָד כְּפִי שָׁרַק שֵׁא סֶקְיוּרִיטִי יוֹדַעַת לָלַמֵּד, עִם הַלֵּב.

ASSEMBLY for intel ia-32, x86-64 LANGUAGE

אודות המכללה

מכללת See Security הנה מכללה **בינלאומית** התמחותית **למקצועות הסייבר**, אחת מ 7 -מכללות מסוגה בעולם ועוסקת בלעדית בתחום זה בכל זמנה, תוך שימוש במתודולוגית הדרכה שנבנתה עבור גורמים ממלכתיים.

המכללה מייצאת את תכניות הלימודים לכל רחבי העולם באמצעות המותג *See Security International* ובאמצעות גופי סייבר ישראליים ידועי-שם העוסקים ביצוא בטחוני. מנהל הקבוצה שבה משולבת המכללה, מר אבי ויסמן, הינו ממובילי ענף הסייבר, יועץ לממשלת ישראל בנושא "אסדרת מקצועות הגנת הסייבר בישראל", פרשן בערוצי השידור בארץ ובחו"ל, מקימו של הפורום הלאומי לאבטחת מידע IFIS (לצד האלוף במיל' וראש המל"ל לשעבר, יעקב עמידרור), מנכ"ל משותף בחברה להשמת כוח אדם בענף הסייבר *SeeHR*, בחברה לייעוץ הגנת סייבר *See Secure Consulting*, בחברה לפתרונות Managed SEIM/SOC בשם *See Events* ובמכללה הבינלאומית לסייבר *See Security College International*.

אודות אסדרת מקצועות הסייבר בישראל: הרשות הלאומית להגנת סייבר

הרשות אשר פועלת במסגרת משרד ראש הממשלה כיחידה עצמאית, החליטה להפעיל אסדרה (רגולציה) מחייבת בנושא הגדרתם של המקצועות השונים בעולם הסייבר, ומפעילה המלצות ברורות בנוגע לתכני הידע לכל מקצוע. חלק ממקצועות הסייבר דורשים הבנת Assembly.

אודות התכנית ללימודי Assembly

בהקשר של הגנת סייבר, נדרש ידע ב-Assembly כבסיס ללימודי Reverse Engineering, אשר בתורו – מהווה בסיס ללימודי Malware Analysis.

מעבדי אינטל משמשים ככוח העיקרי במחשוב האישי כבר מעל 30 שנה. הבנת עקרונות ה- low level הנמצאים במעבדי אינטל משמשת לא רק כיסודות להבנת ארכיטקטורת מעבדי אינטל, אלא גם לחומרה אחרת, וכן משמשת כיסוד מובהק להבנת הנדסה לאחור (reverse engineering) של תוכנות, תכנון קומפיילרים, הבנת מערכות הפעלה וניצול פגיעויות במערכות אלו. בתכנית זו המבוססת בעיקר על התנסות, נלמד את העקרונות הבסיסיים ונתאר את החומרה שקוד אסמבלי משתמש בו. נתאר את הפקודות הנפוצות הנראים כקוד אסמבלי, נלמד אותן לעומקן ונבין מה מתרחש "מאחורי הקלעים".

קהל יעד

לבעלי עניין להתפתח בתחום ההנדסה לאחור או לפתח כישורי ניתוח נוזקות.

מטרת התכנית

- בסיום התכנית תהיה לבוגרים הבנה מעמיקה בקריאת קוד אסמבלי והבנתו.
- תשתית לתכניות מתקדמות יותר בתחום, ובראשן – לימודי Reverse Engineering ו- Malware Analysis.

תנאי קבלה

נסיון קודם בפיתוח בשפת תכנות מודרנית כלשהי, או סיום תכנית הלימודים לשפת C.

עלות

סך 9,000 שח + 400 שח דמי רישום (כולל מע"מ).

מתכונת הלימודים

משך התוכנית כ- 40 שעות במתכונת של 10 מפגשי ערב. הלימודים מתקיימים בקמפוס See Security ברמת-גן (צמוד לתחנת רכבת מרכז). המסלול נפתח פעמיים בשנה.

חובות אקדמיות

- קיימת חובת נוכחות ב-80% מהמפגשים.
- בנושאים הטכניים - תרגול (Hands-on) בכיתה (מעבדה).

זכאות לתעודה והסמכות בינלאומיות

לעומדים בדרישות, תוענק תעודת סיום מטעם See-Security.

See security technologies ltd
InfoSec & Cyber Security College

הערות

- פתיחת התכנית מותנית בכמות של 10 נרשמים.
- דמי ההרשמה אינם מוחזרים, אלא במקרה של אי פתיחת התכנית על ידי המכללה.
- המכללה מביאה לידיעת הנרשמים והסטודנטים כי ייתכנו שינויים במערך התכנית, במועדי הלימוד והבחינות או בכל נושא אחר. הודעה על שינוי תימסר למשתתפים.

למידע נוסף / פגישת יעוץ

מידע מינהלי: אלוירה אליסייב, 03-6122831, 052-8787889,
elvira@see-security.com

יעוץ אקדמי: אבי ויסמן, 03-5799555, 054-5222305,
avi@see-security.com

תכנית לימודים:

- Logical and Arithmetic bit shift instructions and the cases where each would be used (shl (logical shift left), shr (logical shift right), sal (arithmetic shift left), sar(arithmetic shift right))
- Signed and unsigned multiplication and division
- Special one instruction loops and how C functions like memset or memcpy can be implemented in one instruction plus setup (rep stos (repeat store to string), rep mov (repeat mov)
- Misc instructions like leave and nop (no operation)
- Running examples in the Visual Studio debugger on Windows and the Gnu Debugger (GDB) on Linux
- Stepping through a small program and watching the changes to the stack at each instruction (push, pop, call, ret (return), mov)
- Stepping through a slightly more complicated program (adds lea (load effective address), add, sub)
- Understanding the correspondence between C and assembly control transfer mechanisms (e.g. goto in C == jmp in ams)
- Understanding conditional control flow and how loops are translated from C to asm(conditional jumps, jge(jump greater than or equal), jle(jump less than or equal), ja(jump above), cmp (compare), test, etc)
- Boolean logic (and, or, xor, not)

See
see security tecnologies ltd
InfoSec & Cyber Warfare College

<u>הצהרת תלמיד בלימודי Assembly</u>

הריני מאשר בזאת כי קיבלתי דף מידע זה, הבנתי את תכנו והסכמתי לתנאים המפורטים בו.

שם הנרשם: _____ תאריך: _____ חתימה _____