CEN429 GÃ¼venli Programlama Hafta-13

Tigress ve Ã‡eÅŸitlilik Teknikleri

Yazar: Dr. Ã–Ğr. Ãœyesi UÄŞur CORUH

İçindekiler

Şekil Listesi

Tablo Listesi

1 CEN429 GÃ¼venli Programlama

1.1 Hafta-13

1.1.0.1 Tigress ve Ã‡eÅŸitlilik Teknikleri Ö°ndir

- PDF[1]
- DOC[2]
- SLIDE[3]
- PPTX[4]

1.1.1 Outline

- Tigress ve Ã‡eÅŸitlilik Teknikleri
- Obfuscation YÃ¶ntemleri
- SaldÄ±rÄ±lara KarÅŞÄ± Savunma

1.1.2 Hafta-13: Tigress ve Ã‡eÅŸitlilik Teknikleri

Bu hafta, kodun analiz edilmesini zorlaÅŸtÄ±ran ve programÄ± saldÄ±rÄ±lara karÅŞÄ± daha direnÃ§li hale getiren Ã§eÅŸitlilik (diversification) tekniklerini ve Tigress gibi obfuscation araÃ§larÄ±nÄ± inceleyeceÄŞiz. Bu teknikler, programÄ±n Ã§alÄ±ÅŞtÄ±ÄŞÄ± her seferinde farklÄ±laÅŞmasÄ±nÄ± saÄŞlar, bÃ¶ylece saldÄ±rganlarÄ±n aynÄ± yÃ¶ntemlerle programÄ± analiz etmelerini zorlaÅŞtÄ±rÄ±r.

1.1.2.1 1. Tigress Ã‡eÅŞitlilik (Diversity) Teorik AÃ§Ä±klama: Tigress, bir programÄ± farklÄ± ÅŞekillerde dÃ¶nÃ¼ÅŞtÃ¼rerek, saldÄ±rÄ±lara karÅŞÄ± direnÃ§li hale getiren gÃ¼Ã§lÃ¼ bir obfuscation aracÄ±dÄ±r. Bir programÄ±n her Ã§Ä±ktÄ±sÄ± benzersiz bir yorumlayÄ±cÄ± (interpreter) oluÅŞturur. Bu, programÄ±n davranÄ±ÅŞÄ±nÄ± rastgeleleÅŞtirir ve analiz edilmesini zorlaÅŞtÄ±rÄ±r.

[1] pandoc_cen429-week-13.tr_doc.pdf
[2] pandoc_cen429-week-13.tr_word.docx
[3] cen429-week-13.tr_slide.pdf
[4] cen429-week-13.tr_slide.pptx

- **Tigressâ€™te KullanÄ±lan YÃ¶ntemler:**
  - **Instruction Dispatch TÃ¼rleri:**
    - Switch, direkt, endirekt, Ã§aÄŸrÄ± (call), if-else, lineer, binary, interpolasyon.
  - **Operand TÃ¼rleri:**
    - YÄ±ÄŸÄ±n (stack), registerlar.
  - **RastgeleleÅŸtirilen OperatÃ¶rler:**
    - FarklÄ± operandlar ve operator kombinasyonlarÄ± kullanarak kodun karmaÅŸÄ±klaÅŸtÄ±rÄ±lmasÄ±.
  - **Ã‡eÅŞitli DÃ¶nÃ¼ÅŞÃ¼mler:**
    - **Code Flattening:** ProgramÄ±n akÄ±ÅŞ kontrolÃ¼nÃ¼n dÃ¼zleÅŞtirilmesi.
    - **Merge/Split Fonksiyonlar:** BirleÅŞtirilen ya da bÃ¶lÃ¼nen fonksiyonlar.
    - **Opaque Predicates:** Kodda gizli ve deÄŞiÅŞtirilemeyen koÅŞul ifadeleri ekleme.

Uygulama Ã–rneÄŞi:

```
tigress --Transform=Virtualize --Functions=fib --VirtualizeDispatch=switch --out=v1.c test1.c
gcc -o v1 v1.c
```

2. Kodda Ã‡eÅŞitlilik SaÄŞlama Teorik AÃ§Ä±klama: Ã‡eÅŞitlilik, kodun analizini zorlaÅŞtÄ±rmak amacÄ±yla farklÄ± yÃ¶ntemlerle rastgeleleÅŞtirilmesini iÃ§erir. Bu yÃ¶ntemler, bir saldÄ±rganÄ±n programÄ± tersine mÃ¼hendislikle Ã§Ã¶zmesini zorlaÅŞtÄ±rÄ±r. Tigress ile bir program her Ã§alÄ±ÅŞtÄ±rÄ±ldÄ±ÄŞÄ±nda benzersiz bir sanal makine oluÅŞturulabilir.

3. SaldÄ±rÄ±lar ve KarÅŞÄ± SaldÄ±rÄ±lar Teorik AÃ§Ä±klama: Bir saldÄ±rgan, programÄ±n sanal talimat setini Ã§Ã¶zerek kodun nasÄ±l Ã§alÄ±ÅŞtÄ±ÄŞÄ±nÄ± anlamaya Ã§alÄ±ÅŞabilir. Bunun iÃ§in Ã§eÅŞitli saldÄ±rÄ± yÃ¶ntemleri geliÅŞtirilmiÅŞtir, ancak Tigress bu saldÄ±rÄ±lara karÅŞÄ± bazÄ± karÅŞÄ± saldÄ±rÄ± teknikleri sunar.

4. Algoritmik YÃ¶ntemler ve Ã‡eÅŞitlilik SaÄŞlama Teorik AÃ§Ä±klama: Ã‡eÅŞitlilik saÄŞlama algoritmalarÄ±, programÄ±n Ã§alÄ±ÅŞmasÄ±nÄ± karmaÅŞÄ±klaÅŞtÄ±rmak iÃ§in Ã§eÅŞitli seviyelerde uygulanabilir. Bu yÃ¶ntemler, bir saldÄ±rganÄ±n programÄ± Ã§Ã¶zme olasÄ±lÄ±ÄŞÄ±nÄ± azaltmak iÃ§in kullanÄ±lÄ±r.

SonuÃ§ Bu hafta, Ã§eÅŞitlilik saÄŞlama ve kendini deÄŞiÅŞtiren kod gibi ileri dÃ¼zey kod obfuscation tekniklerini Ã¶ÄŞrendik. Bu teknikler, programlarÄ±n saldÄ±rÄ±lara karÅŞÄ± daha direnÃ§li hale getirilmesini saÄŞlar ve saldÄ±rganlarÄ±n kodu Ã§Ã¶zmesini zorlaÅŞtÄ±rÄ±r. Tigress gibi araÃ§lar, kodu rastgeleleÅŞtirerek her seferinde farklÄ± bir yapÄ± oluÅŞturur, bu da kodun analizi ve tersine mÃ¼hendislik yapÄ±lmasÄ±nÄ± daha zor hale getirir.

$13. Hafta - Sonu$