

Система захисту інформації .Net додатків

Виконав:

Студент групи КА-51м

Федь Володимир Вадимович

Науковий керівник:

Кандидат технічних наук, доцент

Дідковська Марина Віталіївна





Актуальність дослідження

- програмне забезпечення піддається впливу великої кількості дестабілізуючих факторів;
- несанкціоноване копіювання програм;
- незаконне розповсюдження та використання;
- значні матеріальні збитки фірм-виробників програмного забезпечення.

Мета, об'єкт та предмет дослідження



- **Об'єкт дослідження** – система захисту програмного забезпечення
- **Предмет дослідження** – методи захисту програмного забезпечення
- **Мета** – дослідження методів захисту програмного забезпечення. Розробка системи захисту інформації .Net додатків.



Постановка задачі

- аналіз проблеми захисту ПЗ;
- аналіз існуючих методів захисту ПЗ;
- розробка системи захисту ПЗ;
- аналіз та перевірка якості роботи розробленої системи.



Існуючі методи захисту ПЗ

Основні методи захисту ПЗ

Апаратні

- ключі з пам'яттю
- ключі з таймером
- ключі з відомим алгоритмом
- ключі з програмованим алгоритмом

Програмні

- алгоритми заплутування
- алгоритми шифрування даних
- метод перевірки цілісності програмного коду
- емуляція процесорів
- виконання на сервері
- парольний захист



Використані методи захисту ПЗ

- Обфускація;
- захист текстових констант;
- захист числових констант;
- парольний захист;
- перевірка цілісності коду;
- шифрування файлів.

Приклад роботи розробленої системи захисту ПЗ



Protector

File

C:\Users\Vova\Documents\Obfuscator\bin\Options.xml

Load Options

Output

```
<?xml version="1.0" encoding="utf-8" ?>
<Obfuscator>
  <OutputFolder>out</OutputFolder>
  <Module file="TestProject.exe">
    <SkipNamespace name="TestProjec" skipTypes="false" />
  </Module>
  <Module file="TestProject2.exe">
    <SkipField type="Namespace.TypeName" name="Pub.*" />
    <SkipNamespace name="Namespace" skipTypes="true" />
    <SkipField type="Namespace.TypeName" name="Pub.*" />
    <Skip Type name="Namespace.TypeName2" skipMethods="true"
skipProperties="true" />
    <Skip Type name="Namespace.TypeName2" skipMethods="true"
skipFields="true" skipProperties="true" />
  </Module>
</Obfuscator>
```

Protect



Критерії якості роботи обфускації

- **Maintainability Index** – комплексний показник якості коду. Чим більше значення цієї метрики, тим легше підтримувати код.

$$MI = \max(0, (171 - 5.2 * \ln HV - 0.23 * CC - \frac{16.2}{1.71} * \ln LoC))$$

- **Cyclomatic Complexity** – характеризує кількість лінійно незалежних шляхів в алгоритмі роботи програми на основі її вихідних текстів.

$$CC = E - N + 2P$$

Результати роботи отриманої системи захисту



Порівняння метрик коду до та після захисту простої програми

Назва метрики	До захисту	Розроблена система захисту	Obfuscator	NetOrbiter
Maintainability Index	84	74	84	76
Cyclomatic Complexity	2	30	2	26

Результати роботи отриманої системи захисту



Порівняння метрик коду до та після захисту більш складної програми

Назва метрики	До захисту	Розроблена система захисту	Obfuscator	NetOrbiter
Maintainability Index	62	58	62	60
Cyclomatic Complexity	25	153	30	100

Порівняльний аналіз отриманої системи захисту ПЗ



Назва методу	Розроблена система захисту	Obfuscator	NetOrbiter
Обфускація	+	+	+
Захист текстових констант	+	+	+
Захист цифрових констант	+	-	-
Парольний захист	+	-	+
Перевірка цілісності коду	+	-	+
Шифрування файлів	+	-	-
Апаратний ключ	-	+	+



Висновки

- Проаналізовано існуючі основні принципи захисту ПЗ;
- Проаналізовано існуючі методи захисту ПЗ. Запропоновано модифіковані методи, що у комплексному використанні мають найбільше покриття уразливих місць програми;
- Розроблено систему захисту інформації .Net додатків на основі обраних методів захисту ПЗ, яка збільшує складність читання декомпільованого коду в середньому у 8 разів.

Практична цінність та перспективи подальших досліджень



Отримана система може бути використана розробниками для захисту програмного забезпечення на платформі .Net

Перспективи подальших досліджень:

- провести дослідження серед експертів з реверсної інженерії з метою виявлення уразливих місць розробленої системи;
- додати більше методів захисту ПЗ до системи;
- реалізувати можливість вибору у системі набору методів, якими буде захищене ПЗ.





Апробація отриманих результатів

- Федь В. В. Порівняльний аналіз криптографічних методів / В.В. Федь // Системний аналіз та інформаційні технології: матеріали 19-ї Міжнародної науково-технічної конференції SAIT 2017. – К.: НК "ІПСА" НТУУ "КПІ ім. Ігоря Сікорського", 2017. – С. 328;

Дякую за увагу