ALMA MATER STUDIORUM – UNIVERSITÀ DI BOLOGNA

Corso di Laurea Triennale in Ingegneria e Scienze Informatiche

DGA Domain Generation Algorithm

Tesi di laurea in:
Programmazione a Oggetti

Relatore
Prof. Mirko Viroli

Candidato
Simone Collorà

Correlatori
Dott. CoSupervisor 1
Dott. CoSupervisor 2

I Sessione di Laurea
Anno Accademico 2024-2025

Abstract

Max 2000 characters, strict.

Optional. Max a few lines.

Contents

List of Figures

List of Listings

Chapter 1

Introduction

Write your intro here.

La sicurezza informatica è un argomento di crescente importanza nel mondo moderno. Con il passare del tempo, i sistemi di protezione sono diventati sempre più sofisticati e potenti ma allo stesso tempo anche gli hackers hanno sviluppato tecniche sempre più avanzate per eludere i sistemi di protezione. Tra queste vi è sicuramente l'uso dei Command and Control (C&C) servers. I C&C sono dei server che manipolano i computer infetti da malwares, chiamati Botnets o Zombi, permettendo all'attaccante di eseguire codice malevolo da remoto. Il malware, però, deve conoscere un indirizzo IP o un dominio per contattare il server. L'attaccante potrebbe inserire in modo bruto l l'indirizzo IP del server nel codice del malware, ma questo metodo è facilmente rilevabile e bloccabile. Gli hackers, quindi, preferiscono utilizzare dei domini generati in modo pseudo casuale per nascondere i loro server chiamati Domain Generation Algorithm (DGA) servers.

Structure of the Thesis

Chapter 2

Background(nome provvisorio)

I suggest referencing stuff as follows: fig. 2.1 or Figure 2.1

I DGA sono algoritmi che generano domini in modo pseudo casuale. Prima viene scelto un seed, di solito la data odierna o le previsioni meteo [3] In questo modo, diventa più difficile per i sistemi di protezione rilevare e bloccare i loro attacchi. Si potrebbe pensare di bloccare direttamente i domini tramite una blacklist ma questo metodo risulta inefficace poiché vengono generati migliaia di domini continuamente. Si pensi che Conficker C, un famoso malware che utilizza DGA, è in grado di generare fino a 50.000 domini pseudo casuali al giorno [1]. Un altro modo per contrastare ciò potrebbe essere quello di fare reverse engineering del DGA per capire quale seed viene utilizzato per generare i domini. Questo però risulta lento e dispendioso e possibilmente inefficace. [2]

2.1 Some cool topic

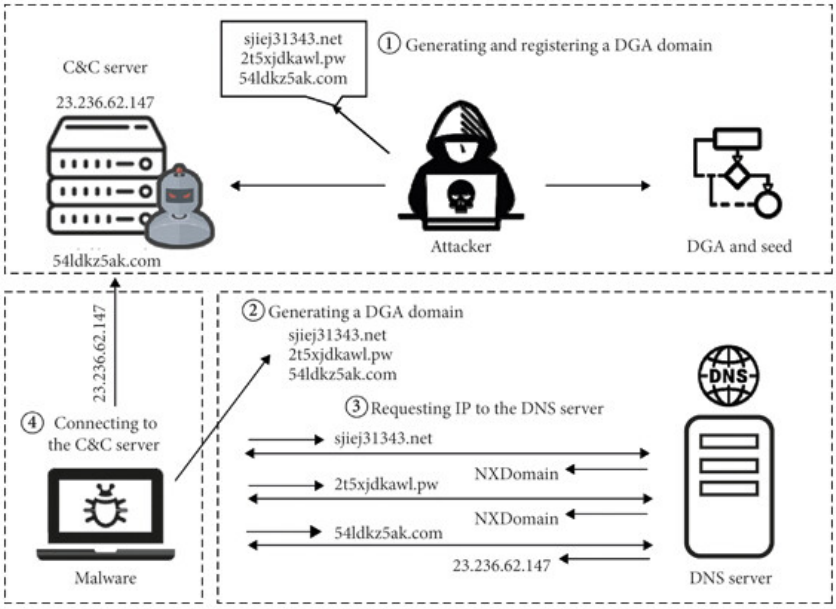

Figure 2.1: esempio del funzionamento di un DGA

Chapter 3

Contribution

You may also put some code snippet (which is NOT float by default), eg: chapter 3.

3.1 Fancy formulas here

```
public class HelloWorld {
  public static void main(String[] args) {
    // Prints "Hello, World" to the terminal window.
    System.out.println("Hello, World");
  }
}
```

Bibliography

[1] Gordon Alley-Young. Conficker worm. In *The Handbook of Homeland Security*, pages 175–175. CRC Press, 2023.

[2] Juhong Namgung, Siwoon Son, and Yang-Sae Moon. Efficient deep learning models for dga domain detection. *Security and Communication Networks*, 2021(1), 2021.

[3] Raaghavi Sivaguru, Chhaya Choudhary, Bin Yu, Vadym Tymchenko, Anderson Nascimento, and Martine De Cock. An evaluation of dga classifiers. In *2018 IEEE International Conference on Big Data (Big Data)*, pages 5058–5067, 2018.

Acknowledgements

Optional. Max 1 page.