

**Kwaadaardig internetverkeer onderscheppen**

**Met Python**

Kenzo Staelens

Jonathan van Caloen

Jonas Van den Berghe

Michaël Vasseur

Bachelor Elektronica-ICT Mentor: Tom Cordemans

2022-2023



**Kwaadaardig internetverkeer onderscheppen**

**Met Python**

Kenzo Staelens

Jonathan van Caloen

Jonas Van den Berghe

Michaël Vasseur

Bachelor Elektronica-ICT Mentor: Tom Cordemans

2022-2023

# Kwaadaardig internetverkeer onderscheppen met Python

K. Staelens, J. van Caloen, J. Van den Berghe, M. Vasseur

Dit project onderzoekt een manier om kwaadaardig internetverkeer te onderscheppen met software die vooral geschreven is in de programmeertaal Python. Als gateway, de plaats waar het verkeer toekomt en vertrekt, wordt gebruik gemaakt van een webserver. Deze fungeert eveneens als een firewall met onze software. De focus bij het bekijken van de binnenkomende pakketten ligt voornamelijk op de tweede en derde netwerklaag.

In het eerste hoofdstuk van het rapport wordt informatie gegeven met betrekking tot malware. Het begrijpen van hoe malware werkt is de sleutel tot het blokkeren van de pakketten. Hier kan ook informatie gevonden worden over de verschillende types, het gebruik en de verschillende soorten manieren waarop een aanval kan plaatsvinden. Hierna worden de bestaande oplossingen besproken. Een algemeen begrip van hoe deze werken helpt bij het bouwen van een nieuwe en functionele oplossing. Vervolgens worden mogelijke oplossingen besproken en vergeleken. In dit hoofdstuk worden ook de criteria voor het efficiënt onderscheppen en blokkeren van malware opgelijst. Daaropvolgend wordt een korte omschrijving van de mogelijke oplossingen en een gedetailleerde vergelijking gegeven. In het laatste hoofdstuk wordt besproken hoe de gekozen oplossing werkt. Hier kunnen ook alle technische aspecten teruggevonden worden. Verder is hier ook een evaluatie te vinden. Ten slotte worden ook de voor- en nadelen van een mogelijk vervolg van het project besproken, namelijk de combinatie met artificiële intelligentie.

Trefwoorden: Firewall, infrastructuur, packet analyser, webserver, malware, Python

# inhoudsopgave

Kwaadaardig internetverkeer onderscheppen met Python 4

inhoudsopgave 5

Codefragmentenlijst 6

figurenlijst 7

Tabellenlijst 8

afkortingenlijst 9

Inleiding 10

1. malware 11

1.1 types 11

1.2 tekenen van infectie 11

1.3 gebruik 11

1.4 Type aanvallen 11

2. firewalls en antivirussoftware 11

1.1 Firewalls 11

1.2 Antivirussoftware 11

3. mogelijke oplossingen 11

3.1 criteria voor malware detectie 11

3.2 mogelijke oplossingen 11

3.2.1 python 11

3.2.2 c# 11

3.3 vergelijking 11

4. uiteindelijke oplossing 12

4.1 laag drie: Netwerk 12

4.2 laag vier: Transport 12

4.3 user interface 12

5. evaluatie 12

Conclusie 13

handleiding 14

Literatuurlijst 15

Bijlagen 16

bijlage 1: kopieën van datasheets 17

Attachment 2: vergaderverslagen 18

Attachment 3: logboek rapporteren 19

# Codefragmentenlijst

# figurenlijst

# Tabellenlijst

# afkortingenlijst

DoS Denial of Service

# Inleiding

In dit rapport wordt een mogelijke oplossing omschreven om kwaadaardig internetverkeer, beter bekend als malware, uit de pakketten te filteren die bij het netwerk toekomen. De software om deze taak tot een goed einde te brengen is vooral geschreven in Python. Ze loopt op een centrale server en fungeert ook als firewall.

Al het kwaadaardige verkeer dat gedetecteerd kan worden op netwerklaag drie door te kijken naar het IP-adres van de afzender wordt geblokkeerd. Dit is handig omdat er zo op basis van geolocatie bepaalde regio’s uitgesloten kunnen worden. Datzelfde gedeelte van de software laat ook de detective van sommige Denial of Service (DoS) aanvallen toe. Enkel pakketten bekijken op laag drie in echter niet voldoende. De analyse op laag vier zorgt ervoor dat ook de inhoud van het ontvangen pakket kan geanalyseerd worden. De inhoud van de pakketten zijn in het merendeel van de gevallen beschermd door HTTPS-encryptie. Er is dus een sleutel nodig om de vertaling van de pakketten mogelijk te maken. Dit is mogelijk doordat de server opgesteld is als een Man-in-The-Middle apparaat. Zo heeft de software toegang tot alle sleutels om de nodige vertalingen te doen.

De software om al deze zaken te doen bestaat al. Firewalls en antivirusprogramma’s kunnen deze taken zonder problemen volbrengen. Deze kennis wordt dan ook gebruikt doorheen dit project. Het doel van dit project is dan ook niet van iets volledig nieuw te ontwikkelen maar om zoveel mogelijk zaken bij te leren en deze te combineren toe een functioneel programma.

De belangrijkste onderzoeksmethode is literatuurstudie van onlinebronnen. Op deze manier kan veel kennis verworven over de bestaande oplossingen voor het onderscheppen van malware. Op de tweede plaats zijn vooral de eigen testen belangrijk. De interface is geëvalueerd door middel van gebruikerstesten.

In het eerste hoofdstuk wordt een introductie tot malware gegeven. Het is zeer belangrijk om te begrijpen hoe verschillende types werken en hoe de inhoud eruitziet alvorens deze te kunnen detecteren. Hierna volgt een korte omschrijving over hoe firewalls en antivirussoftware malware detecteren, onderscheppen en blokkeren. Een lijst van mogelijke oplossingen is de volgende stap in het rapport. De focus hier ligt op de programmeertaal, de netwerklagen en de nodige infrastructuur voor de applicatie. Op het einde van dit hoofdstuk worden de oplossingen met elkaar vergeleken. Het laatste hoofdstuk staat helemaal in het teken van de uiteindelijk gekozen oplossing. Hier worden technische aspecten besproken en wordt ook de werking geëvalueerd. Ten slotte volgt nog een voorstel tot een mogelijk vervolg. Hier worden de mogelijkheden van de combinatie met artificiële intelligentie besproken.

# 1. malware

## 1.1 types

## 1.2 tekenen van infectie

## 1.3 gebruik

## 1.4 Type aanvallen

# 2. firewalls en antivirussoftware

## 1.1 Firewalls

## 1.2 Antivirussoftware

# 3. mogelijke oplossingen

## 3.1 criteria voor malware detectie

## 3.2 mogelijke oplossingen

### 3.2.1 python

### 3.2.2 c#

## 3.3 vergelijking

# 4. uiteindelijke oplossing

## 4.1 laag drie: Netwerk

## 4.2 laag vier: Transport

## 4.3 user interface

# 5. evaluatie

# Conclusie

# handleiding

# Literatuurlijst

# Bijlagen

# bijlage 1: kopieën van datasheets

# Attachment 2: vergaderverslagen

# Attachment 3: logboek rapporteren

|  |  |  |
| --- | --- | --- |
| Jonas Van den Berghe | Schrijven eerste file | Schrijven inleiding, abstract, toevoegen titels, maken inhoudstafel. |
|  |  |  |
|  |  |  |
|  |  |  |