

Practical Malware Analysis & Triage

Malware Analysis Report

Remote Access Trojan Malware

Aug. 2022 | eyyys3c | v1.0

# Table of Contents

[Table of Contents 2](#_Toc85207993)

[Executive Summary 3](#_Toc85207994)

[Malware Composition 5](#_Toc85207996)

[msdcorelib.exe 5](#_Toc85207997)

[Basic Static Analysis 6](#_Toc85207999)

[Basic Dynamic Analysis 7](#_Toc85208000)

[Advanced Static Analysis 8](#_Toc85208001)

[Advanced Dynamic Analysis 9](#_Toc85208002)

[Indicators of Compromise 10](#_Toc85208003)

[Network Indicators 10](#_Toc85208004)

[Host-based Indicators 11](#_Toc85208005)

[Rules & Signatures 13](#_Toc85208006)

[Appendices 14](#_Toc85208007)

[A. Yara Rules 14](#_Toc85208008)

[B. Callback URLs 14](#_Toc85208009)

[C. Decompiled Code Snippets 15](#_Toc85208010)

# Executive Summary

|  |  |
| --- | --- |
| SHA256 hash |  |

Remote Access Trojans (RATs) are malware designed to allow an attacker to remotely control an infected computer. In this sample, the malware will start to query a specific call back URL to download the initial payload *msdcorelib.exe* which will also open a listening socket on the infected machine.

URL: http://serv1.ec2-102-95-13-2-ubuntu.local/

Open listening socket on port 5555

msdcorelib.exe

# Malware Composition

RAT consists of the following components:

|  |  |
| --- | --- |
| File Name | SHA256 Hash |
| msdcorelib.exe |  |

msdcorelib.exe

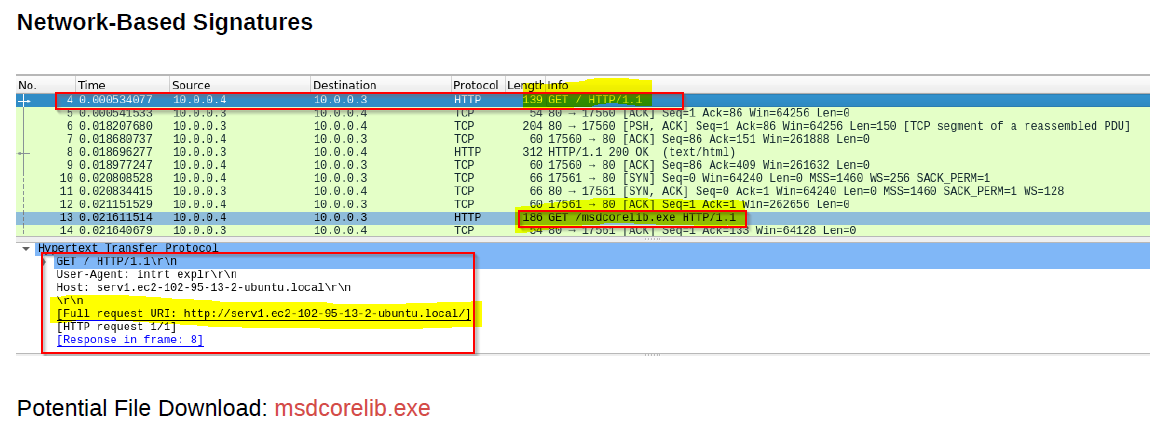
The executable that runs after a successfully connecting to the callback URL

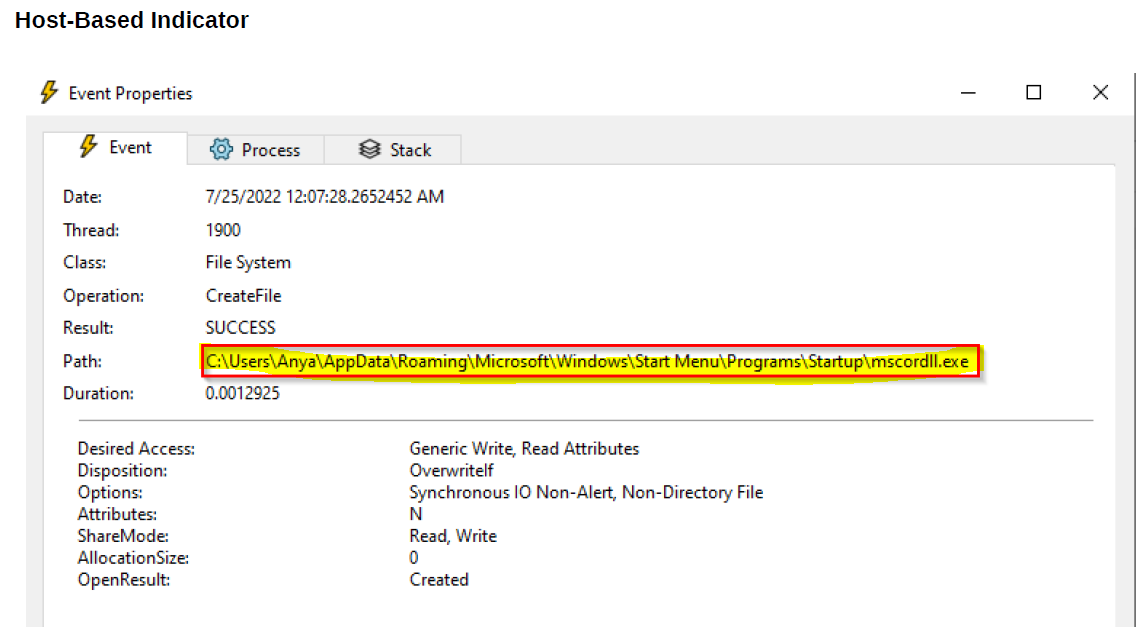
# Basic Static Analysis

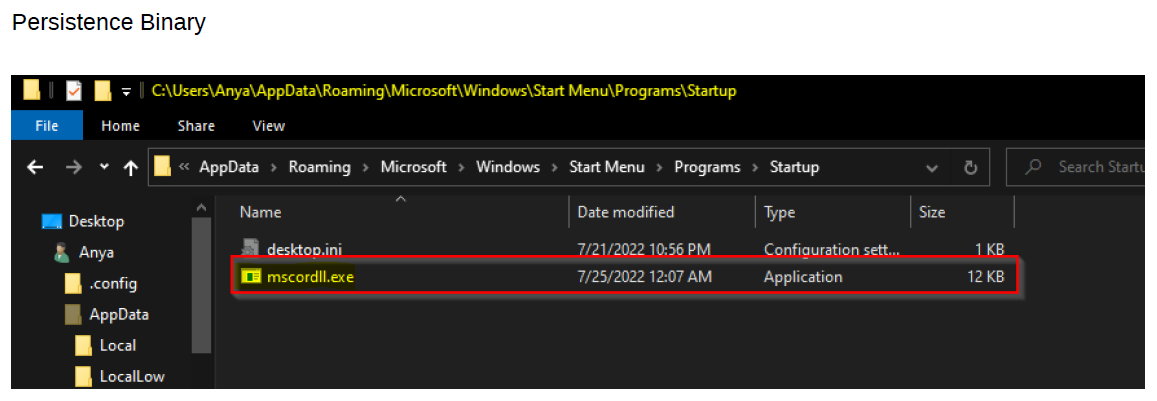
# Floss/Strings

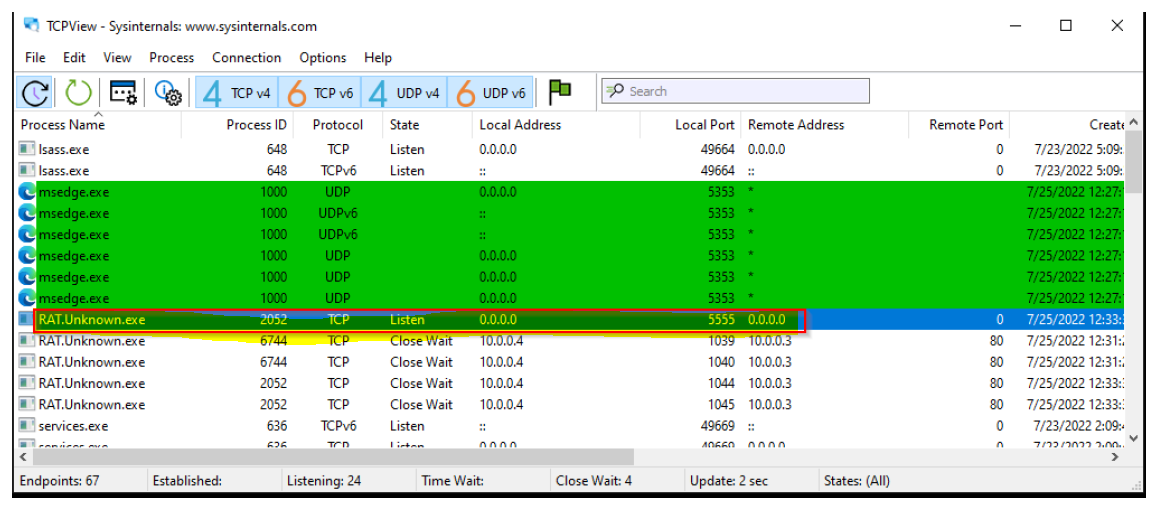
# 

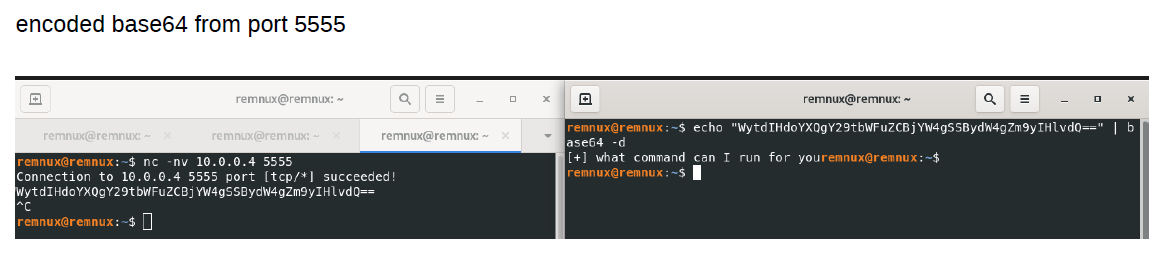
# Basic Dynamic Analysis

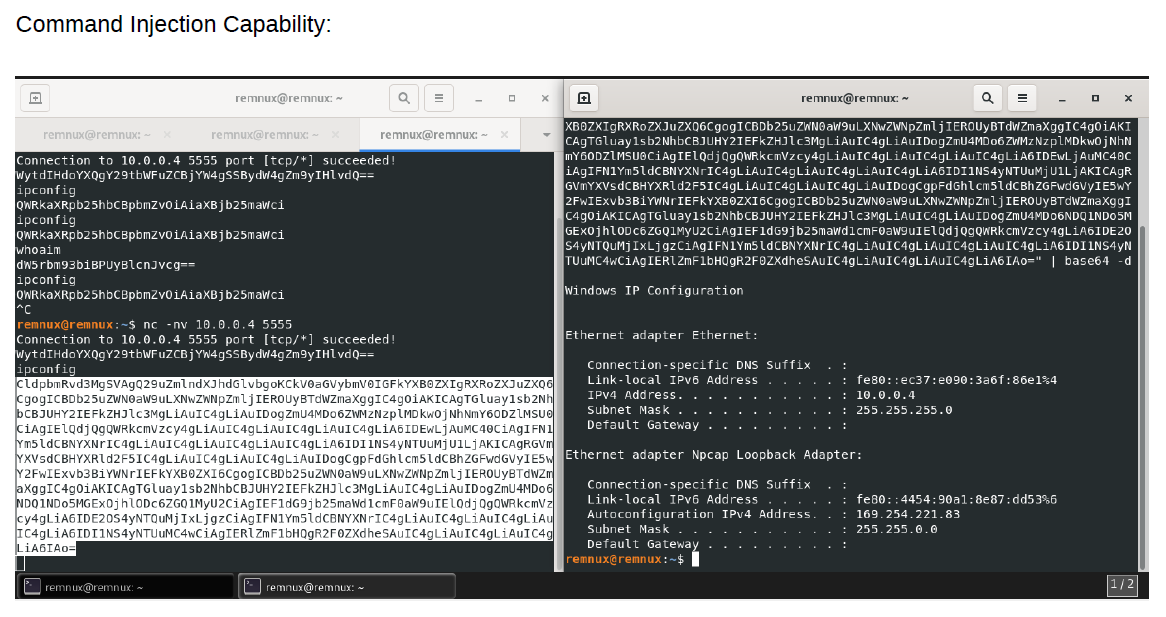












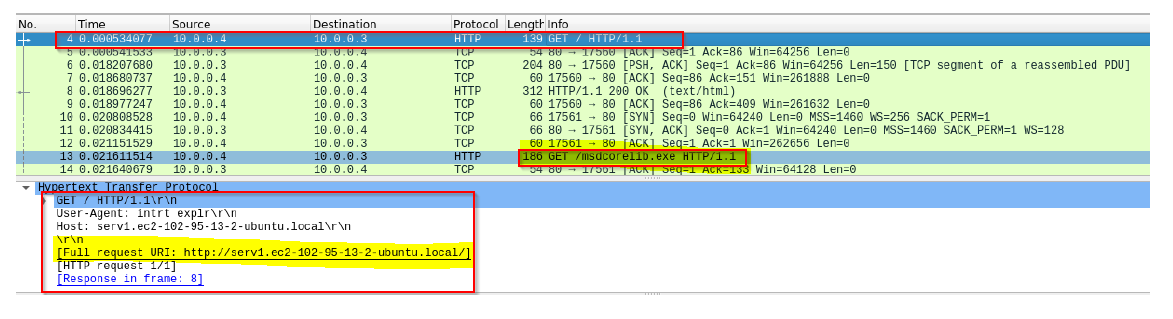
# Advanced Static Analysis

# Advanced Dynamic Analysis

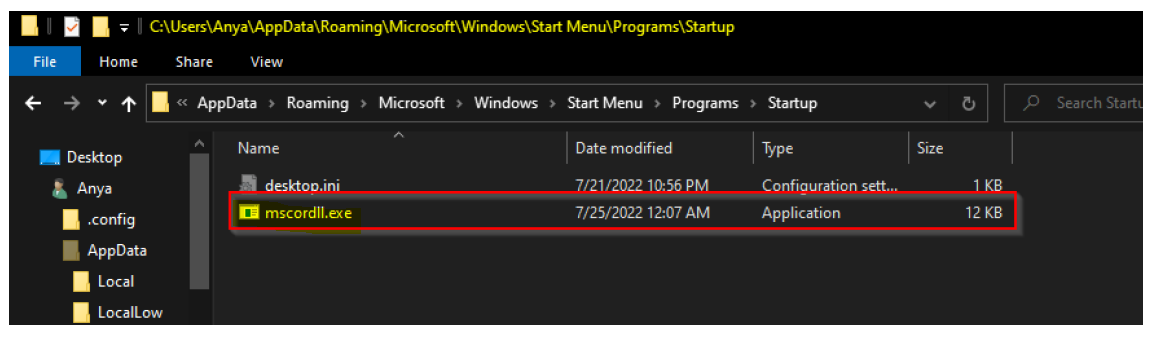
# Indicators of Compromise

The full list of IOCs can be found in the Appendices.

## Network Indicators



## Host-based Indicators



# Rules & Signatures

A full set of YARA rules is included in Appendix A.

# Appendices

## Yara Rules

## Callback URLs

|  |  |
| --- | --- |
| Domain | Port |
| hxxp://serv1.ec2-102-95-13-2-ubuntu.local/ | 80 |

## Decompiled Code Snippets

Text

Description automatically generated

*Fig 5: Process Injection Routine in Cutter*