

MWS Cup 20201

プレミーティング 課題2



2020の問題担当

- 課題2主担当
 - 株式会社サイバーディフェンス研究所 中島 将太
- 問題作成委員
 - セキュアワークス株式会社 中津留 勇
 - 株式会社カスペルスキー 石丸 傑
 - 株式会社日立製作所 石淵 一三
 - 株式会社 エヌ・エフ・ラボラトリーズ 皆川 諒
 - 株式会社 エヌ・エフ・ラボラトリーズ 齋藤 慶太

2021の問題担当

- 課題2主担当
 - 株式会社サイバーディフェンス研究所 中島 将太
- 問題作成委員
 - 株式会社日立製作所 石淵 一三
 - 株式会社 エヌ・エフ・ラボラトリーズ 皆川 諒
 - 株式会社 エヌ・エフ・ラボラトリーズ 齋藤 慶太
 - 学生、若手募集中！



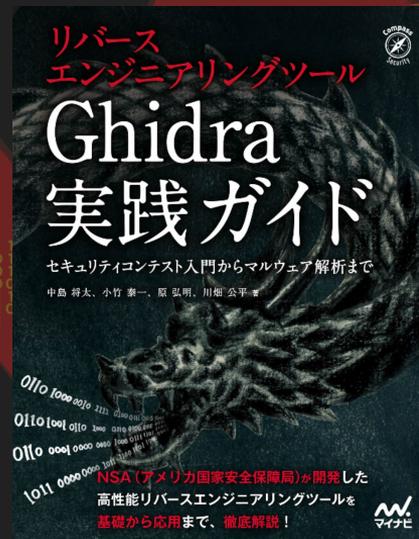
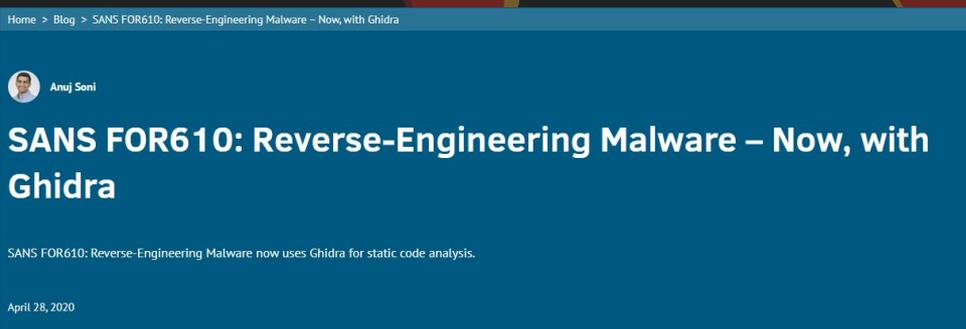
課題2の歴史

- MWS2015: Emdivi RAT
- MWS2016: Daserf RAT
- MWS2017: Oni Ransomware
- MWS2018: PLEAD RAT
- MWS2019: Datper RAT
- MWS2020: Nefilim Ransomware
- MWS2021: ???



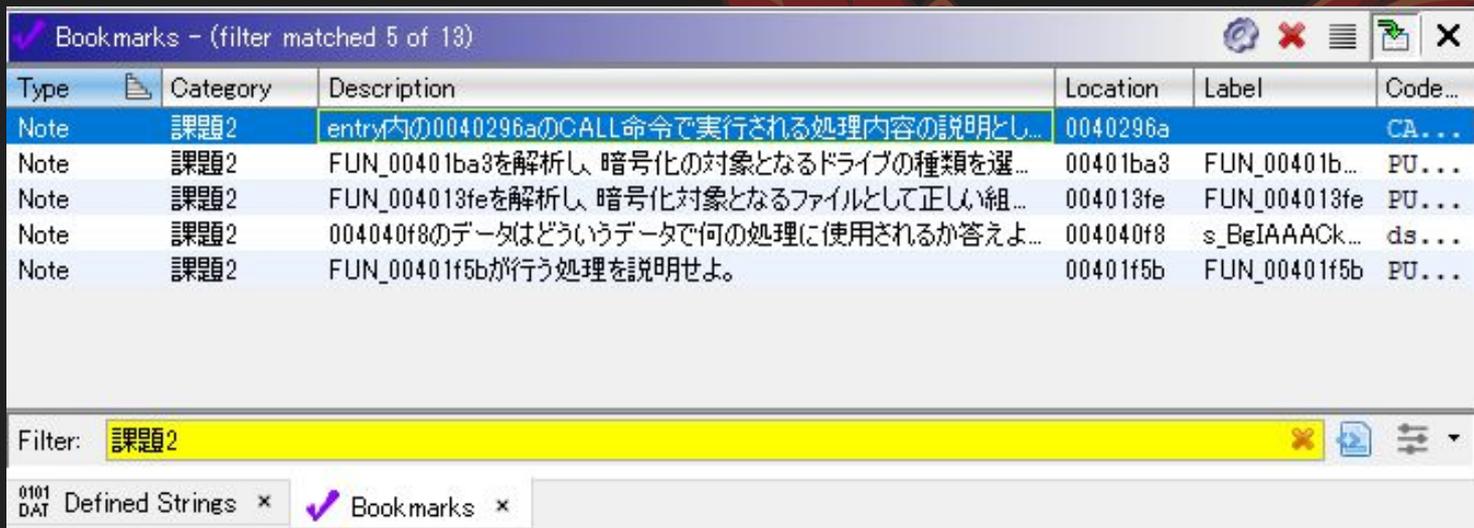
2020の特徴

- IDAからGhidraへ
 - アメリカ国家安全保障局(NSA)が開発したリバースエンジニアリングツール
 - 昨年までidb形式の問題ファイルを配っていたがgzf形式に変更
 - SANSのトレーニングもGhidraに置き換わった
- 日本語の解説書籍も出版されている
 - 筆者...



GhidraのBookmark

- 昨年のアンケートでもブックマークが好評であったためGhidraのBookmark機能を使って問題を登録済み
- ツールバー → Window → Bookmark



便利なプラグイン:Findcrypt

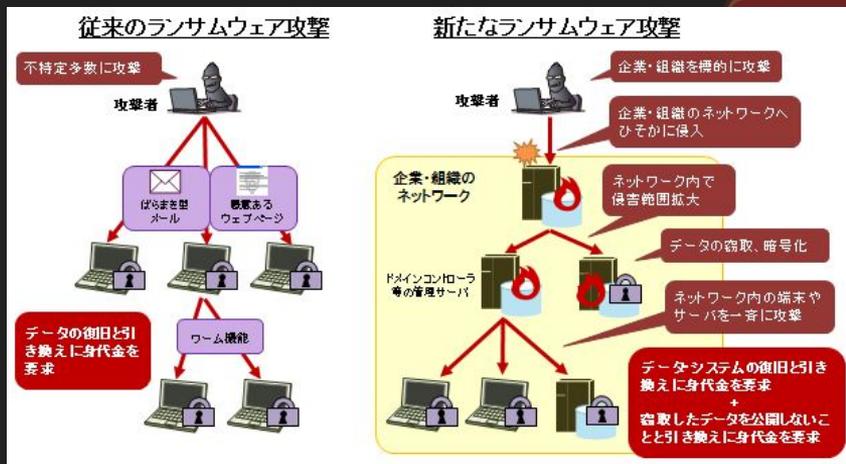
- 暗号処理で使う定数を探す
- FindCryptのGhidra Scriptバージョン
 - <https://github.com/AllsafeCyberSecurity/py-findcrypt-ghidra>

```
• zlib: inflate_lengthStarts, inflate_lengthExtraBits, inflate_distanceStarts, inflate_distanceExtraBits, deflate_lengthCodes
• DES: DES_ip, DES_fp, DES_ei, DES_sbox[1-8], DES_p32i, DES_pc[1-2]
• AES: Rijndael_sbox, Rijndael_inv_sbox, Rijndael_Te[0-4], Rijndael_Td[0-4]
• Blowfish: Blowfish_P_array, Blowfish_S_boxes
• CRC32: CRC32_m_tab_le, CRC32_m_tab_be
• MD5: MD5_T, MD5_initstate
• SHA1: SHA1_H
• SHA224: SHA224_H
• SHA256: SHA256_K, SHA256_H
• SHA512: SHA512_K
• RC5_RC6: RC5_RC6_PQ
• Salsa20_ChaCha: Salsa20_ChaCha_sigma, Salsa20_ChaCha_tau
• Camellia: Camellia_sigma, Camellia_SBOX[1-4]
• Adler-32: Adler32_BASE
• (XX)TEA: (XX)TEA_delta
• xxHash32: xxHash32_PRIME32_[1-5]
• xxHash64: xxHash64_PRIME64_[1-5]
```

```
findcrypt.py> Running...
[*] processing non-sparse consts
[+] found CRC32_m_tab_le for CRC32 at 4b2992d0
[+] found SHA256_K for SHA256 at 4b28d9e0
[*] processing sparse consts
[+] found SHA256_H for SHA256 at 4b2edb20
[+] found MD5_initstate for MD5 at 4b37a610
[*] processing operand consts
findcrypt.py> Finished!
```

問題背景: Human-Operated Ransomware (標的型ランサムウェア)

- 2019年頃からランサムウェアを用いた新たな攻撃が急増
 - a. APTのような手法で横展開し組織の奥深くに侵入する
 - b. 最近では発見した機密情報の窃取も行う
 - c. ランサムウェアを組織全体に配信し、復号と情報公開をネタに脅迫



くわしい攻撃手法が気になった方はこちら

ランサムウェアに標的型攻撃手法を 求めるのは間違っているだろうか

セキュアワークス株式会社

玉田 清貴

山崎 景太

中津留 勇

2020/01/17

Japan Security Analyst Conference 2020

Secureworks®

課題内容

課題2 静的解析

1.ファミリ名

2

2.CALL命令で実行される処理

2

3.暗号化対象ドライブ

2

4.暗号化対象ファイル

2

5.暗号化処理

2

6.暗号化アルゴリズム

2

7.データの役割

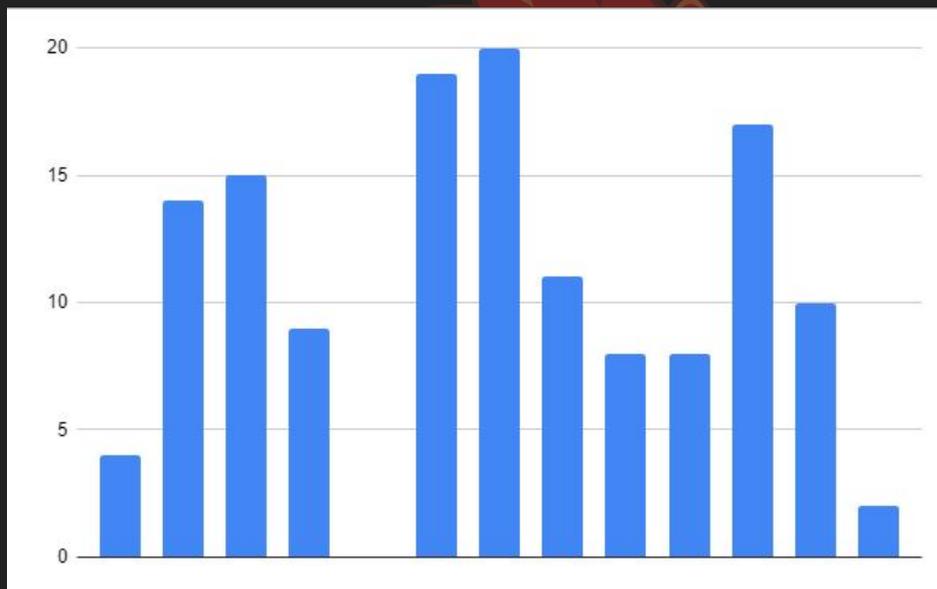
4

8.関数の処理

4

得点分布

- デコンパイラで簡単になったのか平均点はUP
- 静的解析苦手問題は依然とある
- CTFっぽくしたのでブルートフォース的な試行が目立った



勉強方法(過去問)

- MWS2020で問題に使用したマルウェアのハッシュ値
 - b8066b7ec376bc5928d78693d236dbf47414571df05f818a43fb5f52136e8f2e

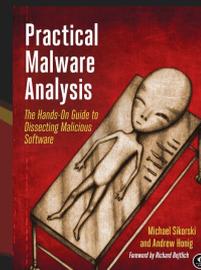
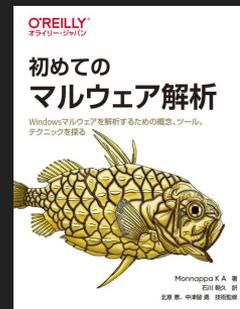
過去の MWS Cup

課題の概要

- 例年、事前課題と当日課題を出題しています。
- 事前課題は、MWS 会期前の約 1 か月間で取り組み、オンラインで解答を提出していただきます。
- 当日課題は、MWS 会期中に約 3 時間で取り組み、その場で解答を提出していただきます。
- 課題に取り組んだ後は、MWS 会期中に、各参加チーム課題に関連するプレゼンテーションを実施してもらいます。
- 採点は、課題の正答数 + 下記指標に基づく発表内容評価で行い、最終的な順位を決定します。
 - 新規性：従来になかったアイデア/データ/ツールであるか？新しい課題設定か？等
 - 実用性：第三者がすぐに活用できるアイデア/データ/ツールであるか？等
 - 有効性：幅広い第三者にとって有益なアイデア/データ/ツールであるか？現存する脅威やデータに効果的か？等
 - プレゼンテーション：起承転結ロジカルな説明か？ハキハキ大きな声で落ち着いて発表してるか？等
- なお、MWS Cup の取り組みを通じて作成したツールやデータ、発表資料等は、MWS コミュニティ活性化を目的に共有をお願いしております。
- 詳しくは MWS Slack の [#mwscup](#) チャンネルをご参照ください。

勉強方法(書籍)

- 初めてのマルウェア解析
 - マルウェア解析全般
 - <https://www.oreilly.co.jp/books/9784873119298/>
- リバースエンジニアリングツールGhidra実践ガイド
 - Ghidraの使い方、マルウェアの静的解析特化
 - <https://book.mynavi.jp/ec/products/detail/id=116258>
- Practical Malware Analysis
 - マルウェア解析全般+サンプル(英語なのと少し古くなってきた)
 - <https://nostarch.com/malware>



勉強方法(Web)

- スライド、動画

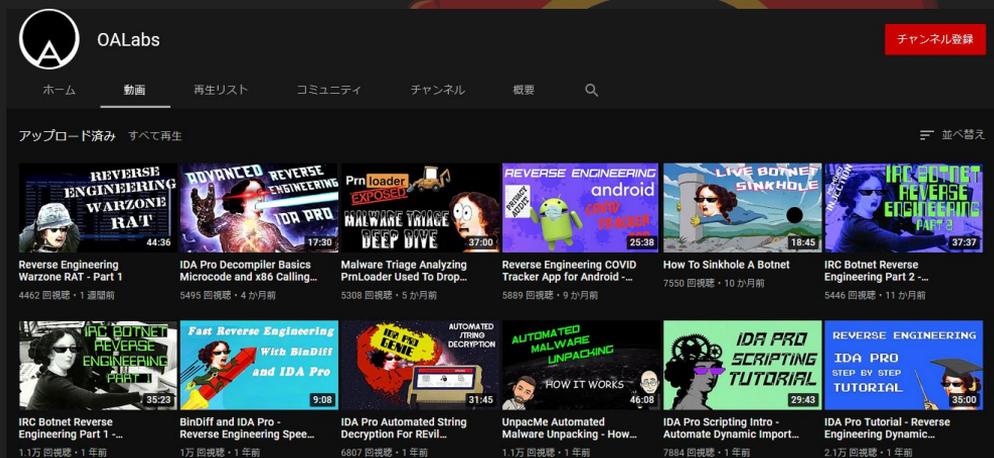
- malware_training_vol1(under construction)

- https://github.com/hasherezade/malware_training_vol1

- OALabs

- <https://www.youtube.com/c/OALabs/videos>

Module 1			Module 2			Module 3		
Slides	Exercises	Topic	Slides	Exercises	Topic	Slides	Exercises	Topic
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	compilation	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Malware missions & tactics (intro)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Evasion and self-defence (intro)
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	PE	<input checked="" type="checkbox"/>	<input type="checkbox"/>	hooking	<input type="checkbox"/> / <input checked="" type="checkbox"/>	<input type="checkbox"/>	Fingerprinting
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Process	<input checked="" type="checkbox"/>	<input type="checkbox"/>	persistence	<input type="checkbox"/>	<input type="checkbox"/>	String obfuscation
<input checked="" type="checkbox"/>	<input type="checkbox"/>	WoW64	<input type="checkbox"/>	<input type="checkbox"/>	UAC bypass	<input type="checkbox"/>	<input type="checkbox"/>	Imports obfuscation
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	shellcode	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Banking trojans	<input type="checkbox"/>	<input type="checkbox"/>	Flow obfuscation
<input type="checkbox"/>	<input type="checkbox"/>	code injection	<input type="checkbox"/>	<input type="checkbox"/>	RATs	<input type="checkbox"/>	<input type="checkbox"/>	Malware antihooking
<input type="checkbox"/> / <input checked="" type="checkbox"/>	<input type="checkbox"/> / <input checked="" type="checkbox"/>	PE loaders	<input type="checkbox"/>	<input type="checkbox"/>	Ransomware	<input type="checkbox"/>	<input type="checkbox"/>	Review of approaches to deobfuscation
			<input type="checkbox"/>	<input type="checkbox"/>	Lateral movements	<input type="checkbox"/>	<input type="checkbox"/>	Kernel-mode malware components



勉強方法 (オンライントレーニング)

- From Zero to Hero (登録あり無料)
 - <https://www.sentinelone.com/lp/zero-to-hero-2021/>
- The Beginner Malware Analysis Course (€39.99)
 - <https://0verfl0w.podia.com/courses/malware-analysis-course>
- Zero2Automated(€149.99)
 - <https://courses.zero2auto.com/>

Zero to Hero Course Syllabus

- Technical overview of injection techniques and persistence mechanisms
- Discovering/recognizing privilege escalation in malware
- Threat actors techniques to gain a foothold on networks
- Deep dive into APTs (advanced persistent threats), eCrime
- Info-stealers and Exploit Kit drive-bys seen in the wild
- Analyzing shellcode usage in malware
- Full analysis of malware techniques – stealth, persistence, algorithms, communication to a C2 server, and advanced capabilities



The Beginner Malware Analysis Course

```
if __name__ == '__main__':
    print("Welcome to the Beginner Malware Analysis Course")
    # Get the user's name
    name = input("Enter your name: ")
    # Greet the user
    print(f"Hello, {name}!")
    # Ask for the user's age
    age = input("Enter your age: ")
    # Convert the age to an integer
    age = int(age)
    # Calculate the user's age in 10 years
    age_in_10_years = age + 10
    # Print the result
    print(f"You will be {age_in_10_years} years old in 10 years.")
    # Ask for the user's favorite color
    color = input("Enter your favorite color: ")
    # Print the result
    print(f"Your favorite color is {color}!")
    # Ask for the user's favorite food
    food = input("Enter your favorite food: ")
    # Print the result
    print(f"Your favorite food is {food}!")
    # Ask for the user's favorite movie
    movie = input("Enter your favorite movie: ")
    # Print the result
    print(f"Your favorite movie is {movie}!")
    # Ask for the user's favorite TV show
    tv_show = input("Enter your favorite TV show: ")
    # Print the result
    print(f"Your favorite TV show is {tv_show}!")
    # Ask for the user's favorite sport
    sport = input("Enter your favorite sport: ")
    # Print the result
    print(f"Your favorite sport is {sport}!")
    # Ask for the user's favorite animal
    animal = input("Enter your favorite animal: ")
    # Print the result
    print(f"Your favorite animal is {animal}!")
    # Ask for the user's favorite planet
    planet = input("Enter your favorite planet: ")
    # Print the result
    print(f"Your favorite planet is {planet}!")
    # Ask for the user's favorite country
    country = input("Enter your favorite country: ")
    # Print the result
    print(f"Your favorite country is {country}!")
    # Ask for the user's favorite city
    city = input("Enter your favorite city: ")
    # Print the result
    print(f"Your favorite city is {city}!")
    # Ask for the user's favorite state
    state = input("Enter your favorite state: ")
    # Print the result
    print(f"Your favorite state is {state}!")
    # Ask for the user's favorite zip code
    zip_code = input("Enter your favorite zip code: ")
    # Print the result
    print(f"Your favorite zip code is {zip_code}!")
    # Ask for the user's favorite street
    street = input("Enter your favorite street: ")
    # Print the result
    print(f"Your favorite street is {street}!")
    # Ask for the user's favorite house number
    house_number = input("Enter your favorite house number: ")
    # Print the result
    print(f"Your favorite house number is {house_number}!")
    # Ask for the user's favorite apartment number
    apartment_number = input("Enter your favorite apartment number: ")
    # Print the result
    print(f"Your favorite apartment number is {apartment_number}!")
    # Ask for the user's favorite room
    room = input("Enter your favorite room: ")
    # Print the result
    print(f"Your favorite room is {room}!")
    # Ask for the user's favorite furniture
    furniture = input("Enter your favorite furniture: ")
    # Print the result
    print(f"Your favorite furniture is {furniture}!")
    # Ask for the user's favorite color
    color = input("Enter your favorite color: ")
    # Print the result
    print(f"Your favorite color is {color}!")
    # Ask for the user's favorite food
    food = input("Enter your favorite food: ")
    # Print the result
    print(f"Your favorite food is {food}!")
    # Ask for the user's favorite movie
    movie = input("Enter your favorite movie: ")
    # Print the result
    print(f"Your favorite movie is {movie}!")
    # Ask for the user's favorite TV show
    tv_show = input("Enter your favorite TV show: ")
    # Print the result
    print(f"Your favorite TV show is {tv_show}!")
    # Ask for the user's favorite sport
    sport = input("Enter your favorite sport: ")
    # Print the result
    print(f"Your favorite sport is {sport}!")
    # Ask for the user's favorite animal
    animal = input("Enter your favorite animal: ")
    # Print the result
    print(f"Your favorite animal is {animal}!")
    # Ask for the user's favorite planet
    planet = input("Enter your favorite planet: ")
    # Print the result
    print(f"Your favorite planet is {planet}!")
    # Ask for the user's favorite country
    country = input("Enter your favorite country: ")
    # Print the result
    print(f"Your favorite country is {country}!")
    # Ask for the user's favorite city
    city = input("Enter your favorite city: ")
    # Print the result
    print(f"Your favorite city is {city}!")
    # Ask for the user's favorite state
    state = input("Enter your favorite state: ")
    # Print the result
    print(f"Your favorite state is {state}!")
    # Ask for the user's favorite zip code
    zip_code = input("Enter your favorite zip code: ")
    # Print the result
    print(f"Your favorite zip code is {zip_code}!")
    # Ask for the user's favorite street
    street = input("Enter your favorite street: ")
    # Print the result
    print(f"Your favorite street is {street}!")
    # Ask for the user's favorite house number
    house_number = input("Enter your favorite house number: ")
    # Print the result
    print(f"Your favorite house number is {house_number}!")
    # Ask for the user's favorite apartment number
    apartment_number = input("Enter your favorite apartment number: ")
    # Print the result
    print(f"Your favorite apartment number is {apartment_number}!")
    # Ask for the user's favorite room
    room = input("Enter your favorite room: ")
    # Print the result
    print(f"Your favorite room is {room}!")
    # Ask for the user's favorite furniture
    furniture = input("Enter your favorite furniture: ")
    # Print the result
    print(f"Your favorite furniture is {furniture}!")
```



Zero 2 Automated

Developed for those looking to enhance their skills further as a Blue-Teamer, Zero2Automated: The Advanced Malware Analysis Course takes a highly practical approach when it comes to learning the advanced principles of Malware Reverse Engineering (with Zero2Hero)

Buy for £149.99