School of

Engineering and Computer Science

Te Kura Mātai Pūkaha, Pūrorohiko

CYBR 171 T1 2023

Ngā whakapūtanga o Te Haumaru rorohiko
Cybersecurity Fundamentals

Anti-Malware

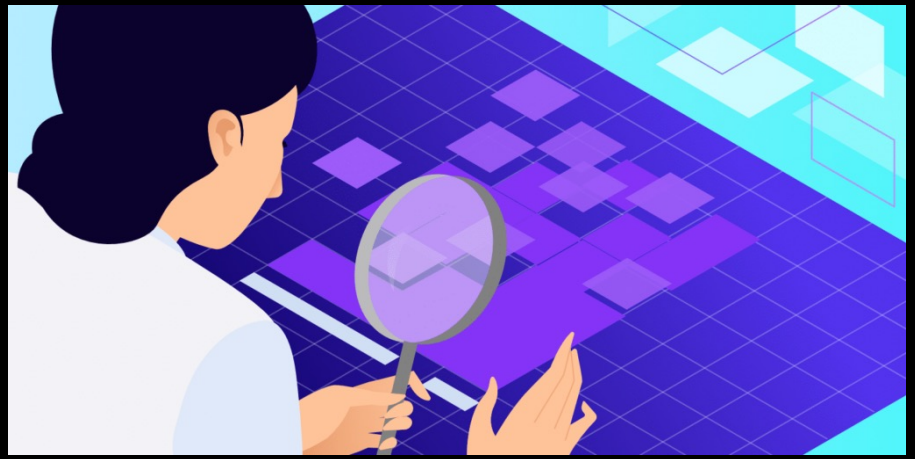
Learning goals

- Anti-malware or anti-virus
- Methods for detecting malware
  - Integrity checkers
  - Signature-based detection
  - Heuristic and Behavioural Detection
- When can the use of an anti-malware system be harmful?

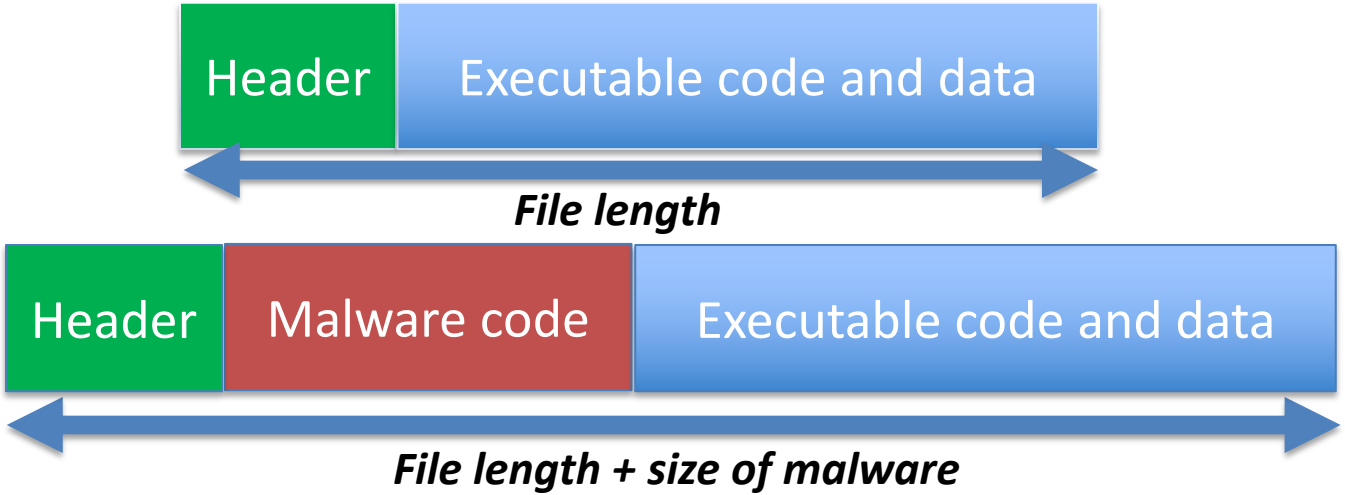
Anti-malware versus antivirus

- Anti-malware = marketing term mostly.
- Anti virus+
  - Detect and locate the malware.
  - Identify specific malware.
  - **Automatic removal**.
- This lecture considers how malware can be **detected** and **potential problems** with anti-malware or anti-virus systems

PART I:
Method 1—Integrity Checkers

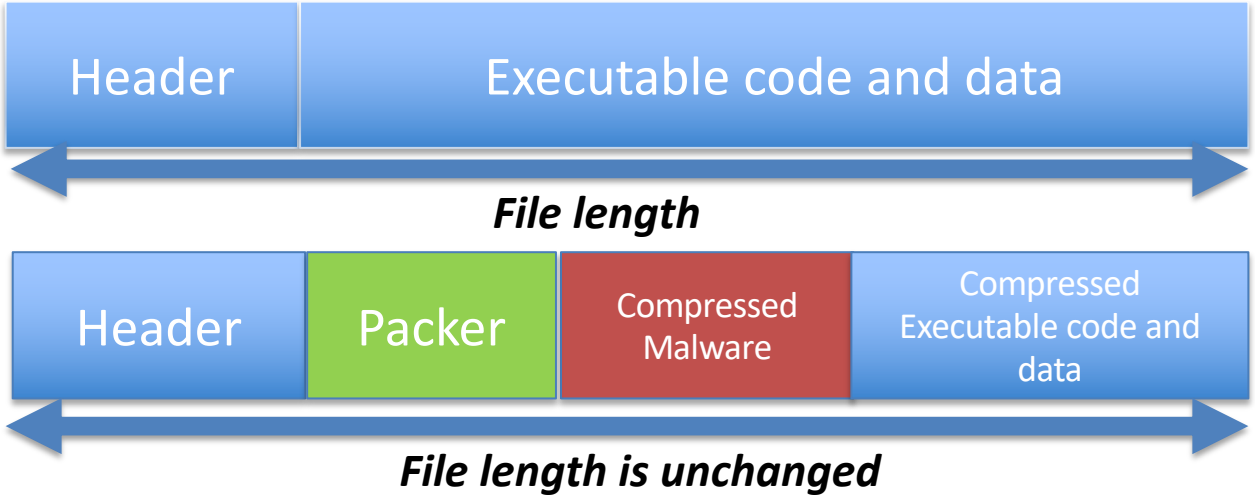


Integrity checkers



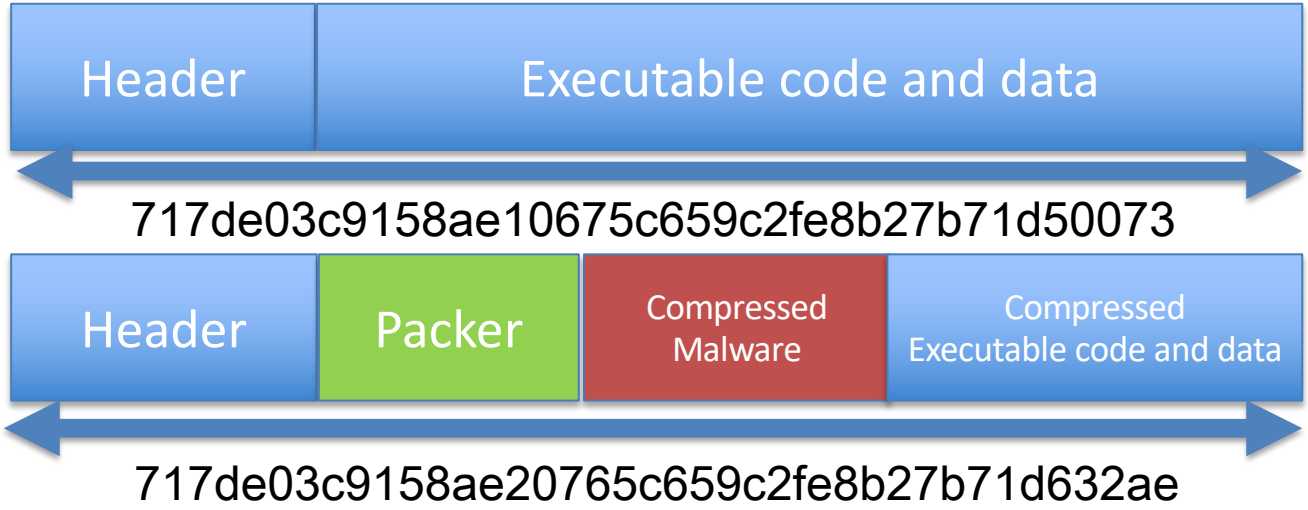
- Viruses make the size of a file grow.
- The computer keeps a list of the lengths.
- **Periodically checks** against the list.
- Any **unexpected change** indicates a problem.

Problem: *Packers* defeat length checks



- Packers **compress** the malware and program.
- Overall size matches the expected ones!
- Many different packers exist for malware.

Problem: Compression virus



- The computer keeps a list of the **lengths** or **hashes**.
- Periodically checks against the list.
- Any unexpected change indicates a problem.

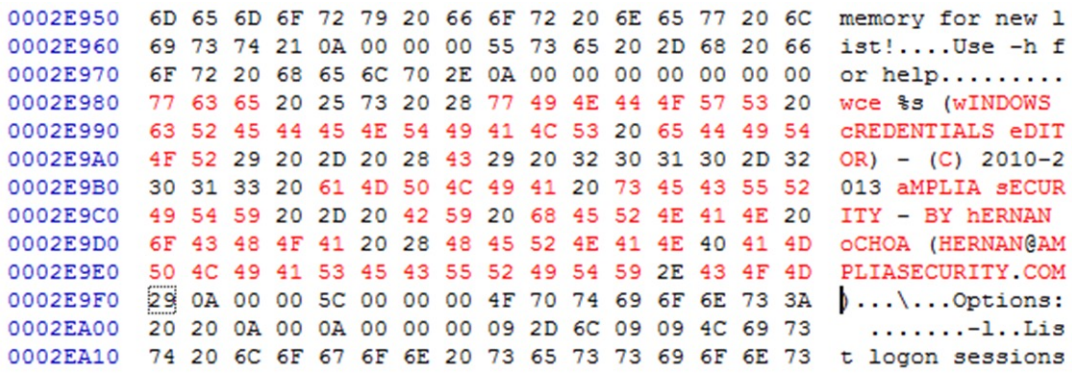
Question

- **Hash-based** integrity checkers fail
- Consider using an **integrity checker** for your personal computer
  - What files might be **wrongly flagged** as malware?
    - Also known as a **false positive**
  - How could you **avoid** this?
  - What are the **drawbacks** of your solution?

PART II:

Method 2—Signature-based detection

Signature-based detection

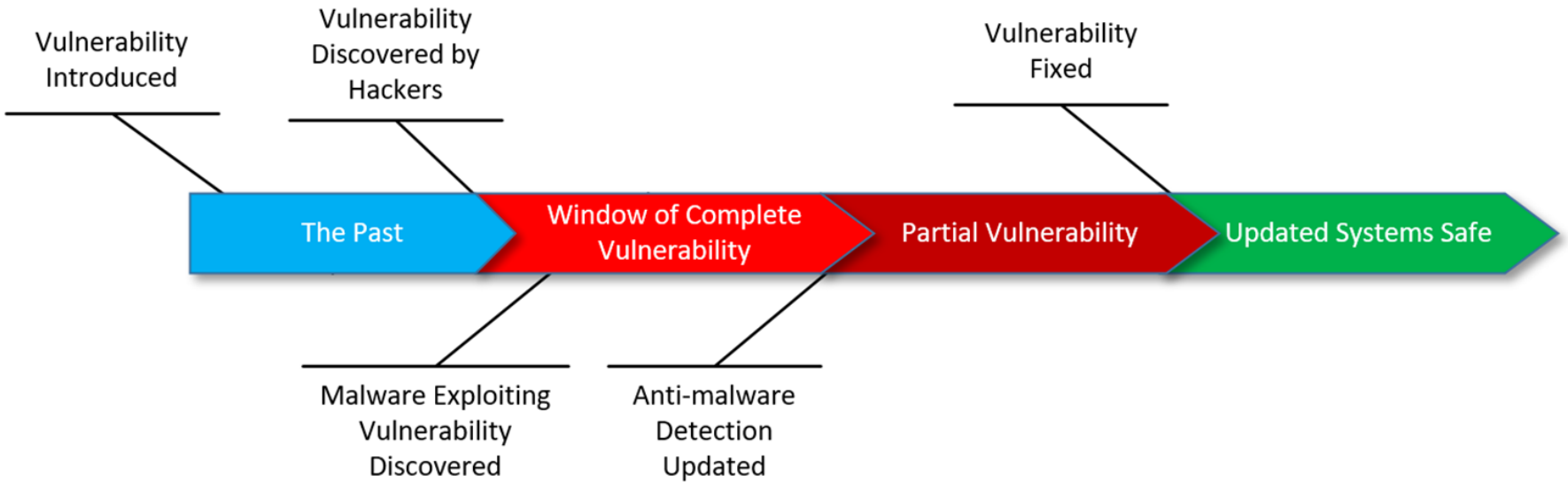


```
0002E950  6D 65 6D 6F 72 79 20 66 6F 72 20 6E 65 77 20 6C  memory for new l
0002E960  69 73 74 21 0A 00 00 00 55 73 65 20 2D 68 20 66  ist!....Use -h f
0002E970  6F 72 20 68 65 6C 70 2E 0A 00 00 00 00 00 00 00  or help.........
0002E980  77 63 65 20 25 73 20 28 77 49 4E 44 4F 57 53 20  wce %s (wINDOWS
0002E990  63 52 45 44 45 4E 54 49 41 4C 53 20 65 44 49 54  cREDENTIALS eDIT
0002E9A0  4F 52 29 20 2D 20 28 43 29 20 32 30 31 30 2D 32  OR) - (C) 2010-2
0002E9B0  30 31 33 20 61 4D 50 4C 49 41 20 73 45 43 55 52  013 aMPLIA sECUR
0002E9C0  49 54 59 20 2D 20 42 59 20 68 45 52 4E 41 4E 20  ITY - BY hERNAN
0002E9D0  6F 43 48 4F 41 20 28 48 45 52 4E 41 4E 40 41 4D  oCHOA (HERNAN@AM
0002E9E0  50 4C 49 41 53 45 43 55 52 49 54 59 2E 43 4F 4D  PLIASECURITY.COM
0002E9F0  29 0A 00 00 5C 00 00 00 4F 70 74 69 6F 6E 73 3A  )...\...Options:
0002EA00  20 20 0A 00 0A 00 00 00 09 2D 6C 09 09 4C 69 73    .......-l..Lis
0002EA10  74 20 6C 6F 67 6F 6E 20 73 65 73 73 69 6F 6E 73  t logon sessions
```

- Database of malware **signatures** (sometimes called DAT files).
- Search for bit **pattern** or a **hash**.
- Requires regular updates.
- Limited to detection of **known** malware.

Problem: *Zero-day exploits*

- A zero-day exploit is an attack that exploits a **previously unknown** security vulnerability.

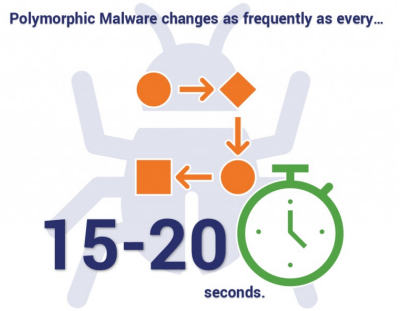


Problem: *Mutating* malware

- Malware developers mutate existing malware into unrecognisable forms
- Signature-based anti-virus
  - Malware developers take an **existing** virus
  - **Mutate** the binary code
  - No longer matches the signature
- Key techniques:
  - **Encrypted** viruses
  - **Polymorphic** viruses
  - **Metamorphic** viruses

Polymorphic malware

- Polymorphism "**change the appearance of**"
- Packer encrypts malware at the time of distribution or copying
- Packer decrypts at run time into memory
- Key embedded with the virus
- Packer, also known as crypter



Key changes each time that malware distributed or replicates

Defences against polymorphic viruses

- **Real-time defences**
  - Run in a special **emulator**
  - Monitor memory
  - Look for deciphered code and apply signatures
- **Post-mortem analysis**
  - Find the key
  - Decrypt the code before the analysis

Key changes each time that malware replicates

Metamorphic malware

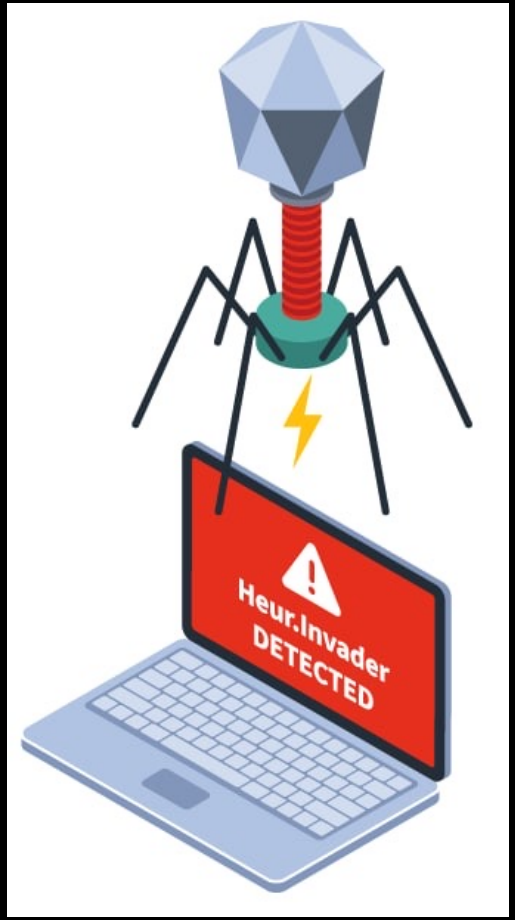
- Metamorphic malware "**automatically recodes itself each time it propagates itself or is distributed**"
- **Obfuscation** is another name for this technique
- Legitimate programs sometimes use the same technique
- Simple techniques include **adding instructions**
  - Random lengths of do nothing (called **NOP**)
  - **Add** and **subtract** a constant value from a variable
  - Add **useless if-then** statements and loops

Metamorphic malware (cont.)

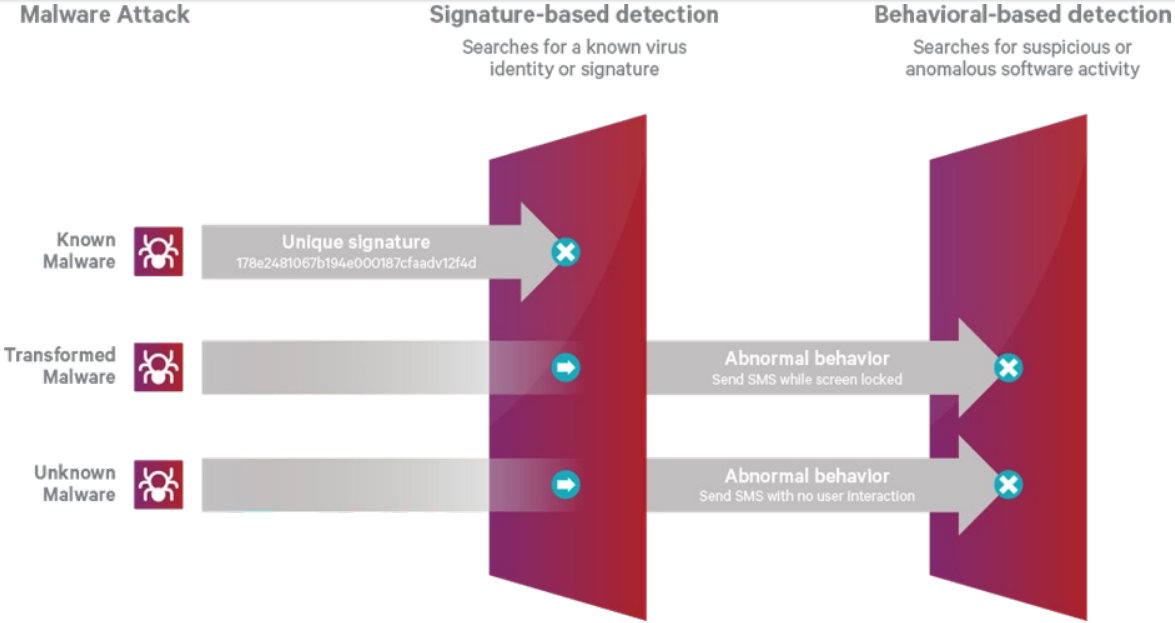
- Mutation engine applied to malware
- Simple techniques include adding instructions
  - Random lengths of do nothing (called NOP)
  - Add and subtract a constant value from a variable
  - Add useless if-then statements and loops

Malware code is transformed by a mutation engine

PART III:
Method 3—Heuristic and Behavioural Detection



Behavioural detection



- Focus on **behaviour** rather than **signatures**.
- A behaviour-based signature will identify **mutants** of the original malware
- System services and resources are used in the same manner

Problem: *Behavioural detection*

- Skilled analyst time required to build the behavioural signatures
  - Must **run** malware
  - Observe behaviour
  - Define abnormal behaviour
- Behaviour detection on a live system is dangerous?
  - What if we miss an action?
  - Can't undo bad things.

Heuristic Detection

- Machine learning helps computers learn without direct instructions
  - Collect examples of **malware** and **goodware**
  - Convert to data
  - Algorithm identifies patterns
- Can be applied to:
  - Malware **code** itself
  - System calls and resource usage
  - Network communication

Z. Bazrafshan, H. Hashemi, S. M. H. Fard and A. Hamzeh, "A survey on heuristic malware detection techniques," *The 5th Conference on Information and Knowledge Technology*, 2013, pp. 113-120, doi: 10.1109/IKT.2013.6620049.

Problem: Heuristic detection

- Accurate detection requires lots of training samples that are correctly identified
- Inaccurate detection
  - **False negative** "looks like goodware but isn't"
    - Could be costly
  - **False positive** "looks like malware but it isn't"
    - Ransomware encrypts lots of files at once.
    - System software might do the same thing.

Solution: Combined Techniques

- Most modern anti-malware use **multiple techniques**
- Signature-based to remove obvious malware
- Heuristics-based behaviour analysis for unknown malware
- **Whitelisting** of safe applications to avoid **false positives**

PART IV:

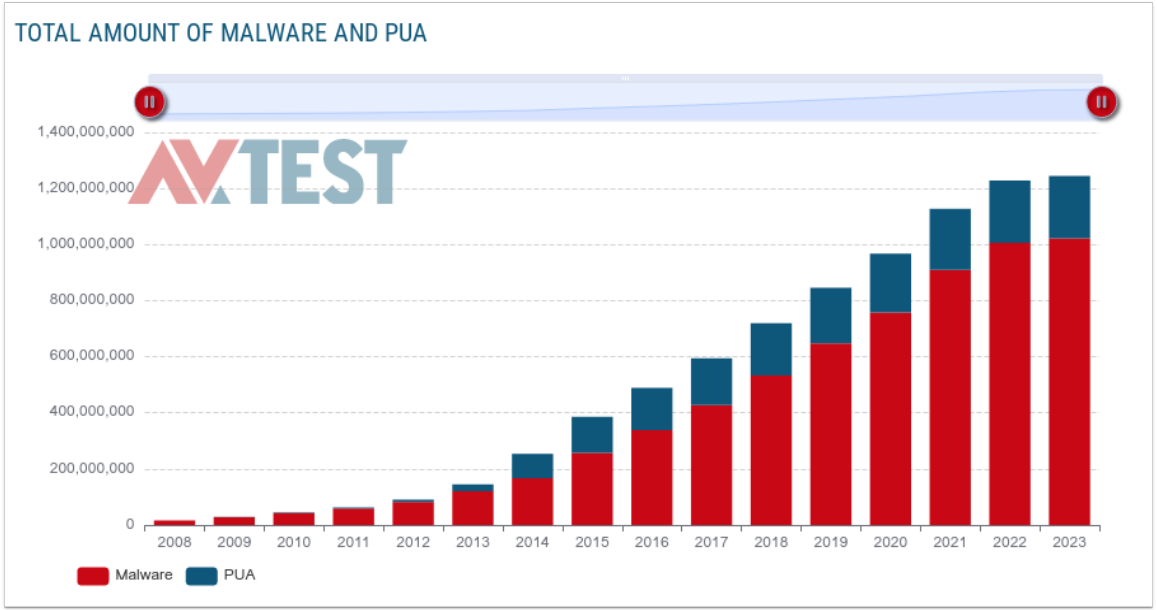
When can the use of anti-malware system be *harmful*?

Anti-malware systems increase your attack surface

- "*How to compromise the enterprise endpoint*" Project Zero team.
- Multiple security flaws in Symantec (previously other antivirus).
- Emulator doing the unpacking itself was vulnerable to flaws allowing machine to be exploited.

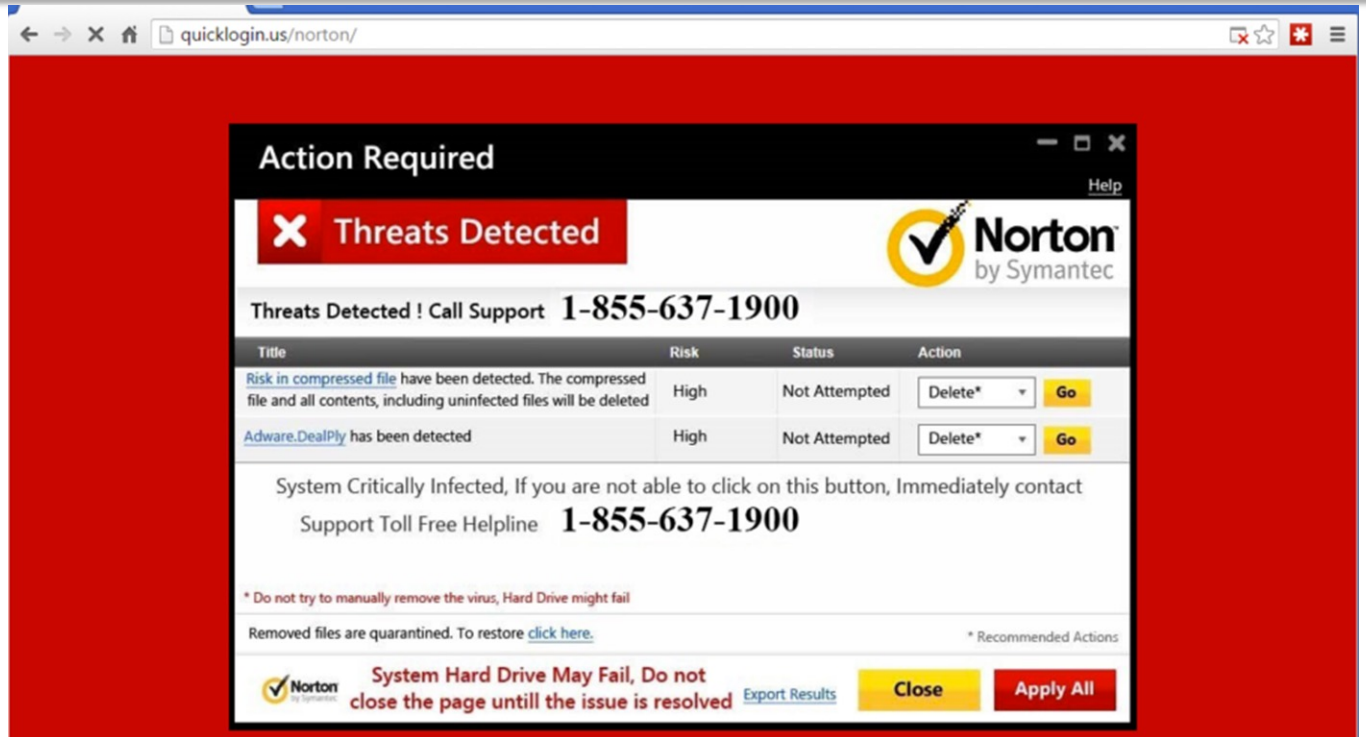
Can we keep up with malware growth?

- Challenges:
  - Rapid growth of Malware (AVTest 2023).
  - Ease of mutation.
  - Speed of distribution of updates.



Source: https://www.av-test.org/en/statistics/malware/

Fake Anti-malware and Tech support scams



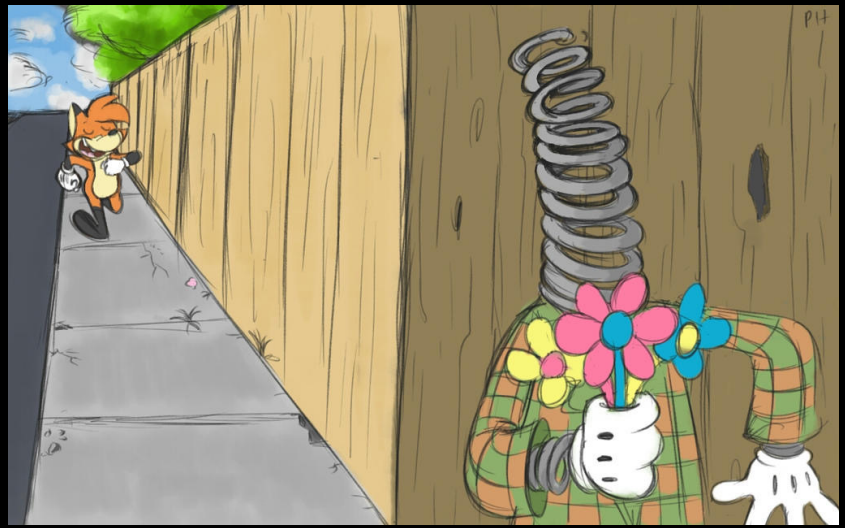
Fear is used to get people to install malware or remote control software.

Prevention as an alternative

- **Prevention** = don't get infected or at least limit the damage. Organisations can approach this in four ways: *policy*, *awareness*, *vulnerability mitigation* and *threat mitigation*.
- **Policy**: manage the implementation of countermeasures.
- **Awareness**: influence individual behaviour to practice "safe computing" or "cyber hygiene".
- **Vulnerability mitigation**: patching/updates, access controls on access to files .
- **Threat mitigation**: least privilege assigned to users

https://www.linkedin.com/pulse/should-prevention-core-principle-artificial-larry-bridgesmith-j-d-/

PART V:
What’s next



What's up next

- We will have our third guest lecture on Thursday
  - **Nigel Bates** (*Architecture and Security Manager*)

- We will look at Networking and communication
  - Internet,
  - wireless,
  - security challenges and
  - role of TCP/IP