Malware Detection

Prof. Ravi Sandhu
Executive Director and Endowed Chair

Lecture 12

ravi.utsa@gmail.com
www.profsandhu.com
Virus detection is undecidable

Anti-virus (more generally anti-malware) is a great business model
  - Need regular updates
  - Infinite supply of new malware

Malware can be stealthy
Malware can be really stealthy
Malware Detection Techniques

Nwokedi Idika and Aditya Mathur, A Survey of Malware Detection Techniques, Purdue University, Feb 2007.
Malware Detection Techniques

Misuse Detection

- Signature-based
  - static
  - dynamic
  - hybrid

- Anomaly-based
  - static
  - dynamic
  - hybrid

Behavior-Based Detection

- Specification-based
  - static
  - dynamic
  - hybrid

Nwokedi Idika and Aditya Mathur, A Survey of Malware Detection Techniques, Purdue University, Feb 2007.
Signature Limitations

\[ U = \text{set of all malicious behavior} \]
\[ S = \text{set of all known signatures} \]

S needs regular updates

Nwokedi Idika and Aditya Mathur, A Survey of Malware Detection Techniques, Purdue University, Feb 2007.
Anomaly Based

Training Phase

Detection Phase

Infer patterns

Infer specifications
Anomaly Based Limitations

\[ A = \text{set of all behaviors} \]
\[ V = \text{set of all valid behaviors} \]
\[ V_{\text{approx}} = \text{approximation to } V \]

Blue area is false positives
If white area extends outside blue area we have false negatives

Nwokedi Idika and Aditya Mathur, A Survey of Malware Detection Techniques, Purdue University, Feb 2007.
Defeat signature-based detection

- Encrypted malware
- Polymorphic malware
- Metamorphic malware

- Rootkit can misrepresent the existence or content of executable files

Encrypted Malware

1. Encrypted Main Body
2. Key
3. Decryptor
4. Cleartext Main Body

- Execute malware
- Propagate malware
Encrypted Malware

- Encrypted Main Body
- Key
- Decryptor
- execute malware
- marker
- reveals signature
- Cleartext Main Body
- Propagate malware
Polymorphic Malware

- Encrypted Main Body
- Key
- Decryptor
- Cleartext Main Body

execute malware

propagate malware

- Encrypted Main Body
- Key’
- Obfuscated Decryptor
Polymorphic Malware

- Encrypted Main Body
- Key
- Decryptor

execute malware

Cleartext Main Body

- Encrypted Main Body
- Key’
- Obfuscated Decryptor

propagate malware

no signature
Polymorphic Malware

Execute in a sandbox and detect the signature after decryption

encrypted Main Body → Key, Decryptor → cleartext Main Body

No signature

propagate malware
Polymorphic Malware

- Encrypted Main Body
  - Key
  - Decryptor
  - execute malware
  - propagate malware

- Encrypted Main Body
  - Key’
  - Obfuscated Decryptor

Cleartext Main Body

Mutation Engines automate this construction

Execute in a sandbox and detect the signature after decryption

Mutations:
- Execute in a sandbox and detect the signature after decryption
- Automate this construction
Metamorphic Malware

Original Main Body

propagate malware

Obfuscated Main Body

propagate malware

Obfuscated Main Body

execute malware

Obfuscated Main Body

execute malware

Obfuscated Main Body

execute malware

Obfuscated Main Body

no signature
Obfuscation Techniques

- Dead-Code Insertion
- Register Reassignment
- Subroutine Reordering
- Instruction substitution
- Code transposition
- Code Integration
Really Stealthy Malware

- Not visible in source code
- Reappears in binary code due to malware infected compiler
- In theory could reappear in binary code due to other components in binary execution workflow
  - Loader
  - Linker
  - OS
  - BIOS

Malicious Compiler Inserts a Backdoor

OS Login module

Malicious Compiler Binary

Infected Login Binary
Malicious Compiler Inserts a Backdoor

Assumption:
Malicious behavior cannot be detected in binary, but may be detectable in compiler source.
Malicious Self-Compiler in Binary and Source

- Malicious Compiler source for language L
- Compiler binary for language L
- Malicious Compiler binary for language L
Malicious Self-Compiler in Binary but not in Source

1. Compiler source for language L
2. Malicious Compiler binary for language L
3. Malicious Compiler binary for language L
Malicious Self-Compiler in Binary but not in Source

Compiler source for language L

Malicious Compiler binary for language L

Malicious Compiler binary for language L

partial countermeasure