

## **Guiding Research Questions:**

- What historical malware taxonomies exist, and how does this map to more current examples?
- What are the best techniques for setting up a malware sandbox environment in which it is safe to run malware? And what kind of data collection can you do in such environments?
- Which worm propagation techniques remain viable in modern systems, and which fail due to architectural changes?

## **Sources (Title, Author, Link):**

**What historical malware taxonomies exist, and how does this map to more current examples?**

### **A Taxonomy of Computer Worms**

Weaver and coauthors (year varies by version)

Link: <https://people.cs.vt.edu/~kafura/cs6204/Readings/Context-Problems/WormTaxonomy.pdf>

Description: Classifies computer worms by propagation strategy trigger conditions and payload type to give a structured overview of major worm families.

Applicability: Useful. Gives a foundational framework for categorizing malware behaviors which can help structure datasets or guide comparisons when analyzing evolutionary trends.

### **An Empirical Study of Malware Evolution**

Gupta et al. (publication venue varies)

Link: <https://wiscr.cs.wisc.edu/papers/malware-comsnets.pdf>

Description: Presents an analysis of a large collection of malware samples over time tracking changes in size complexity and functional features.

Applicability: High. Directly relevant for projects concerned with tracking how malware traits change over time and for building quantitative measures of evolution.

### **Prudent Practices for Designing Malware Experiments: Status Quo and Outlook**

Christian Rossow and coauthors 2012

Link: <https://christian-rossow.de/publications/guidelines-ieee2012.pdf>

Description: Surveys current methodologies used in malware research highlights ethical and safety concerns and proposes best practices for experimental design.

Applicability: Very high. Essential reading if your work involves executing or manipulating malware in controlled environments to ensure safe reproducible experiments.

### **A Comprehensive Analysis of WannaCry: Technical Analysis, Reverse Engineering, and Motivation**

Fahad Alraddadi (course or project report)

Link:

[https://people-ece.vse.gmu.edu/coursewebpages/ECE/ECE646/F20/project/F18\\_presentations/Session\\_III/Session\\_III\\_Report\\_3.pdf](https://people-ece.vse.gmu.edu/coursewebpages/ECE/ECE646/F20/project/F18_presentations/Session_III/Session_III_Report_3.pdf)

Description: Breaks down the WannaCry ransomware campaign with technical dissection of its components reverse engineered routines and discussion of attacker motivation.

Applicability: Moderate. Offers a detailed case study that can inform understanding of real world ransomware behavior but may be limited in generalizability and formal rigor compared with peer reviewed work.

**What are the best techniques for setting up a malware sandbox environment in which it is safe to run malware? And what kind of data collection can you do in such environments?**

### **Practical Malware Analysis: The Hands-On Guide to Dissecting Malicious Software**

*Michael Sikorski, Andrew Honig, No Starch Press, 2012*

Link: <https://nostarch.com/malware>

Description: A foundational book that walks through safe malware lab setup, including isolated networks, VM snapshots, and host monitoring techniques.

Applicability: Very high. This is the de facto reference for building a safe sandbox and understanding what data (filesystem, registry, memory, network traffic) can realistically be collected.

### **The V-Network Testbed for Malware Analysis**

*Ahmad, Woodhead, Gan, 2019*

Link:

[https://gala.gre.ac.uk/id/eprint/15871/7/15871%20WOODHEAD\\_V-Network\\_Testbed\\_2016.pdf](https://gala.gre.ac.uk/id/eprint/15871/7/15871%20WOODHEAD_V-Network_Testbed_2016.pdf)

Description: Describes a virtualized network testbed designed specifically for controlled malware execution and observation across multiple hosts.

Applicability: High. Closely aligned with your goal of running self-propagating malware while logging infections and propagation behavior in a closed environment.

### **SEED Labs: Malware and Worm Attacks (Lab Environment Documentation)**

*SEED Security Labs, Syracuse University*

Link: [https://seedsecuritylabs.org/Labs\\_20.04/](https://seedsecuritylabs.org/Labs_20.04/)

Description: Educational labs that demonstrate safe, isolated malware and worm experiments using VMs and controlled networks.

Applicability: Very high. Strong precedent for running real malware behavior safely in an academic context, including explicit containment and logging strategies.

### **NIST SP 800-125B: Secure Virtual Network Configuration for Virtual Machine Technologies**

NIST (Chandramouli et al.), 2016

Link: <https://nvlpubs.nist.gov/nistpubs/specialpublications/nist.sp.800-125b.pdf>

Brief description: Official guidance on virtual network isolation, segmentation, traffic control, and monitoring for VM environments.

Applicability: Very high for the “safe to run malware” part. You can cite NIST to justify host-only networks, segmentation, firewalling, and controlled egress policies, then implement those as guardrails.

### **Which worm propagation techniques remain viable in modern systems, and which fail due to architectural changes?**

#### **A Taxonomy of Computer Worms**

*Nicholas Weaver, Vern Paxson, Stuart Staniford, Robert Cunningham, 2003*

Link: <https://people.cs.vt.edu/~kafura/cs6204/Readings/Context-Problems/WormTaxonomy.pdf>

Description: Introduces a systematic taxonomy of worm propagation techniques, including target discovery, propagation vectors, activation, and payloads.

Applicability: Extremely high. This paper gives you the conceptual framework to categorize Morris-era techniques and directly ask which categories still exist or have modern analogs.

#### **The Internet Worm Program: An Analysis**

*Eugene H. Spafford, 1988*

Link: <https://spaf.cerias.purdue.edu/tech-reps/823.pdf>

Description: A detailed technical analysis of the Morris worm, including the specific vulnerabilities and trust assumptions it exploited.

Applicability: Essential baseline. This is your ground truth for identifying the original techniques before testing how they behave under modern system constraints.

#### **How to Own the Internet in Your Spare Time**

*Stuart Staniford, Vern Paxson, Nicholas Weaver, 2002*

Link: [https://www.usenix.org/legacy/event/sec02/full\\_papers/staniford/staniford.pdf](https://www.usenix.org/legacy/event/sec02/full_papers/staniford/staniford.pdf)

Description: Explores high-speed worm propagation strategies and models their effectiveness under different network assumptions.

Applicability: High. Useful for reasoning about which propagation ideas scale today and how changes like NAT, firewalls, and patching alter feasibility.

#### **Code Red: A Case Study on the Spread and Victims of an Internet Worm**

*David Moore, Colleen Shannon, Jeffery Brown, 2002*

Link: [https://www.caida.org/catalog/papers/2002\\_codered/codered.pdf](https://www.caida.org/catalog/papers/2002_codered/codered.pdf)

Description: Empirical measurement of a real-world worm outbreak, focusing on infection dynamics and population estimation.

Applicability: High. Serves as a methodological reference for how to measure propagation success and failure in your own controlled experiments.

### **WannaCry Ransomware Attack: Analysis and Mitigation**

*Microsoft Security Response Center, 2017*

Link:

<https://www.microsoft.com/en-us/security/blog/2017/05/12/wannacrypt-ransomware-worm-targets-out-of-date-systems/>

Description: Technical analysis of WannaCry's worm component and why it succeeded against unpatched modern systems.

Applicability: Very high. Demonstrates that worm-style propagation is still viable today under certain architectural and operational conditions.

### **Zero Trust Architecture (NIST SP 800-207)**

*National Institute of Standards and Technology, 2020*

Link: <https://csrc.nist.gov/publications/detail/sp/800-207/final>

Description: Defines modern architectural principles intended to eliminate implicit trust and limit lateral movement.

Applicability: High. Provides a concrete explanation for why many classic worm techniques fail today and a guide for which architectural changes to introduce when "modernizing" your experimental network.