Operating System Hardening Based on Privacy, Security and Performance: Customization of Microsoft Windows

University of Turku
Department of Computing
Master of Science (Tech) Thesis
Cyber Security
June 2024
Timo Ilonen

Supervisors:
Seppo Virtanen
Jouni Isoaho

UNIVERSITY OF TURKU
Department of Computing

Timo Ilonen: Operating System Hardening Based on Privacy, Security and Performance: Customization of Microsoft Windows

Master of Science (Tech) Thesis, 74 p.
Cyber Security
June 2024

Microsoft Windows is the current market leader in the field of desktop operating systems, a position it has held for decades. This mass adoption is driven by a version-based approach, where each new iteration is promoted as an upgrade to users, and old versions are eventually retired, receiving no further support. Yet can each new version truly be considered superior to its predecessor, or is the operating system instead regressing in key areas?

Current versions of Windows, Windows 11 in particular, boast ever greater security features, such as Trusted Platform Module (TPM) 2.0 enforcement for hardware-based security, yet concerns grow over its respect for users' privacy thanks to increasing telemetry measures. An operating system that is simultaneously more secure, yet cannot itself be trusted, begins sounding like a contradiction of terms.

In this thesis, a comprehensive overview of Windows security from both a current and historical perspective is performed. Its current state of data collection is also covered. With this knowledge in mind, the perspective needed to tackle the research challenge - a practical evaluation of Windows in terms of privacy, security, performance and usability - will have been gained.

Standardized testing across several versions of the Windows operating system was conducted, from Windows XP to Windows 11, as well as altered versions of Windows 11 running under a virtual environment. Ubuntu, a Linux distribution, was also included for comparison. The purpose of this testing was to accurately gauge the current state of Windows in these key areas, to predict the direction it is likely to be heading, as well as to give recommendations for different user groups regarding their choice of operating system.

It was found that Windows has indeed become more resilient to malware over time, though this development already plateaued with Windows 10. Meanwhile, data collection has grown massively with Windows 10 and Windows 11, while performance has slightly decreased. However, third-party modifications provided meaningful improvements in these areas. The operating system's level of security saw no measurable change, though a system running fewer services may have a smaller attack surface. Linux was found to excel in system performance in particular.

Keywords: Microsoft Windows, operating system security, malware, telemetry, system performance

Contents

List of Figures

List of Tables

List of acronyms

AD Active Directory

AES Advanced Encryption Standard

AI Artificial Intelligence

API Application Programming Interface

ARP Address Rsolution Protocol

ASLR Address Space Layout Randomization

DAP Data Execution Prevention

DES Data Encryption Standard

DICE Device Identifier Composition Engine

DL Deep Learning

DLL Dynamic-link Library

DMA Direct Memory Access

DNN Deep Neural Network

DNS Domain Name System

DoS Denial of Service

DPAPI Data Protection Application Programming Interface

DPC Deferred Procedure Call

DRM Digital Rights Management

DRTM Dynamic Root of Trust for Measurement

EFS Encrypting File System

FAT File Allocation Table

HTML Hypertext Markup Language

HVCI Hypervisor Code Integrity

IDS Intrusion Detection System

IOMMU Input/Output Memory Management Unit

IoT Internet of Things

LAN Local Area Network

MBEC Mode Based Execution Control

MDM Modern Device Management

MDNS Multicast Domain Name System

ML Machine Learning

MS-DOS Microsoft Disk Operating System

NLP Natural Language Processing

NTFS New Technology File System

NTML2 New Technology LAN Manager

NT New Technology

OEM Original Equipment Manufacturer

OS Operating System

PIN Personal Identification Number

RAM Random Access Memory

RIoT Robust Internet of Things

SHACK Secure Hardware Cryptography Key

SHA Secure Hashing Algorithm

SMM System Management Mode

SRTM Static Root of Trust for Measurement

SV2 Sun Valley 2

TPM Trusted Platform Module

UAC User Account Control

UEFI Unified Extensible Firmware Interface

UWP Universal Windows Platform

VBS Virtualization Based Security

WDAC Windows Defender Application Control

WPE Windows Preinstallation Environment

1 Introduction

With a market share of 72%[1] in the desktop space at the time of writing, Microsoft Windows continues to be a clear market leader in the desktop operating system field. Windows 11, released in October of 2021, has gained a moderate portion of this share, but still lags far behind Windows 10, with a 28% portion of the overall Windows share, compared to the 67% of Windows 10 [2]. Considering Windows 10 only has about 18 months of support remaining, it is reasonable to expect a decline in Windows' future market share.

Windows differs from its competitor, Linux, in several ways. Firstly, Windows is presented as a unified experience, where a certain version, such as Windows 10, is the same for everyone, at least without further modification. It utilizes a generalist strategy, vying for market share by appealing to most, and then using that scale to establish itself as a dominant force in the market. Newer versions are then presented as successors that users are expected to migrate to, at the very least once official support for an older version ends. What then, when a successor turns out to be inferior to its predecessor? In the Linux ecosystem, such issues are largely averted by specialization - a common element across distributions is found in the kernel, but substantial changes are made to tailor the operating system for different demographics. In the present situation, the current market leader, Windows 10, is soon to be discontinued, yet its successor has failed to accumulate the needed market share while enforcing strict hardware requirements that prevent a substantial portion

of the market from making the switch. With these factors in mind, it is valuable to pursue research regarding the history, current state and near future of Windows in multiple facets, to assess its past, current and future state, as well as that of its competitor, Linux, to determine the best options for different user groups.

In this thesis, Microsoft Windows will be assessed in the key fields of security, privacy, performance and usability. Standardized tests in these fields willbe conducted across over 20 years of the operating system's history to establish an overview of its current state and where it is likely to be heading. Its competitor, Linux, will be used to contrast these findings and provide additional context. A general assessment of each version of the operating system will be given, to serve as guidance for different user demographics to make informed decisions regarding their operating system of choice, and what may be lost or gained by switching to an alternative. Deep dive analysis of individual features is avoided, as this would harm the scope of the thesis, and is best left to future study.

The rest of the thesis will proceed as follows. In Chapter 2, the current state of Windows security and privacy will be covered. A comprehensive overview of the security features present in contemporary versions of Windows will be presented, as well as the types of user data collected. The focus will then be on the history of these security features, establishing a timeline. By now, a general understanding of the operating system and its features will have been acquired, and relevant research questions will be established, to ponder upon the significance of the study.

In Chapter 3, a literature review will be conducted. This focuses on security, particularly malware detection. This will aid the reader in getting in touch with the bleeding edge of current Windows security research.

In Chapter 4, the test environment will be established. The chosen operating systems, virtual environment, tools as well as malware payloads will be presented.

In Chapter 5, the testing will commence. Each chosen operating system will be

subjected to the chosen tests as well as a subjective user experience review.

Finally, in Chapter 6, conclusions will be drawn. The results of each operating system will be summarized and compared, and the future implications of the findings considered. The limitations of the study will be gauged and possible future research avenues will be suggested.

2 Assessment of Microsoft Windows

2.1 Overview of present Windows security, privacy and usability

While Windows 11 offers a reasonable range of new security features, these come at the cost of hardware accessibility - only processors with a Trusted Platform Module (TPM) 2.0 chip included are officially supported by the operating system. In practice, this means Intel processors of 8th generation and newer, released in 2017, and AMD Ryzen processors of 2nd generation or newer, released in 2018 [3]. Installation on older processors is possible via editing the registry or via custom installation media, but such systems fall outside official support and do not have access to the added hardware-based security features.

Trusted Platform Module technology is used to enable hardware-based security features. It is designed to be resistant to tampering and impossible to run malicious code on. It can generate and manage cryptographic keys, be used for device authentication and secure the boot process [4]. On newer systems, Microsoft Pluton may be used alongside or to replace the TPM module. Pluton, according to Microsoft, is a "chip-to-cloud security technology [5]." It utilizes Zero Trust principles (verify explicitly, use least privilege access, and always assume breach [6]) along with a secure subsystem and Microsoft authored software. It may be best understood as a fuller implementation of the TPM 2.0 specification, with an architecture spanning

software, firmware and hardware.

Beyond these hardware-based security features, Windows 11 provides the following [7]:

- **System Guard** A collection of features that protect against attacks on the boot process and kernel in particular, with Secure Boot, Secure Launch and Memory Integrity protecting the system during boot, startup and against memory-based attacks respectively.
  - **Static Root of Trust for Measurement (SRTM)** Prevents the installation of rootkits and bookits, which would boot before Windows to attain a high level of privilege.
  - **Dynamic Root of Trust for Measurement (DRTM)** A more flexible version of the former, allowing untrusted code to boot the system but then forcing it into a trusted state. Has the benefit of not relying on unwieldy UEFI blocklists/allowlists.
  - **System Management Mode (SMM)** A CPU mode that listens for specific system operations and executes them above privilege levels available to the OS. To prevent malicious code execution, its memory access and allocation are restricted.

- **Kernel DMA Protection** Protects against Direct Memory Access drive-by attacks, which are done by inserting malicious peripheral devices, such as those utilizing USB, Thunderbolt and CFexpress. These devices can access memory and perform operations without the processor. The Input/Output Memory Management Unit (IOMMU) only allows DMA remapping compatible devices to access memory allocated to them.

- **Secure Boot and Trusted Boot** Protect the system during the boot process.

Secure Boot handles secure communication between the Unified Extensible Firmware Interface (UEFI) and the kernel. Signatures are enforced at every level. Trusted Boot continues the process and verifies the signature of the Windows kernel, which then does the same to critical system components and refuses to load any that fail the check.

- **Cryptography and Certificate Management** Windows supports many cryptographic functions such as random number generations, symmetric/asymmetric encryption, hashing, signing and verification as well as key agreement and derivation. Certificates are automatically updated and validated using certificate trust lists.

- **Windows Security** A centralization application for security management and monitoring. Sections included are virus & threat protection; account protection; firewall & network protection; app & browser control; device security; device performance & health and family options.

  - **Microsoft Defender Antivirus**[8] A built-in anti-virus suite, It supports anomaly detection, quarantining, and automatic file scanning, among other features. It uses predictive technologies to help in detection of new malware, even in offline environments.

  - **Windows Defender Firewall** Managing inbound and outbound connections, it protects users from unauthorized traffic. Traffic may be controlled based on application, source and destination IP addresses, default gateway/DHCP/DBS/subnet, protocop, interface and IP address type.

  - **Microsoft Defender SmartScreen** Protects against phishing and malware in sites and applications. It compares both sites and downloaded files against dynamic lists of known malicious sources.Drive-by attacks,

malicious advertisements and scams are also protected against. If a site does not have an established reputation, a warning is given to the user.

- **Windows application security** The suite of features used to secure users from malicious actors when using applications.
  - **Smart App Control** Prevents the execution of malicious applications. It only allows applications that are predicted to be safe to be run based on constantly updated information.
  - **Windows Defender Application Control (WDAC)** Enforces a trust-based model for applications in enterprise environments.
  - **User Account Control (UAC)** Helps prevent against malware by forcing applications to run with non-administrator privileges unless otherwise authorized. Helps prevent automatic installation and unintended changes to the system.
  - **Application isolation features** Enable applications to run in secure environments, such as virtualized containers for browsing websites. Windows Sandbox may run older 32-bit applications in a virtual environment to prevent any impact on the larger operating system.
  - **Application containers** In general, resources under Windows run under a medium integrity level. Universal Windows Platform (UWP) applications, however, are restricted to a low integrity level container environment, which restricts their access to system resources. Malware running in such an environment has less leverage to escape.
- **Windows identity protection**
  - **Windows Hello** Allows the replacement of traditional username/password login via biometric identification. Possible metrics include facial

recognition, fingerprint, or PIN. In an enterprise environment, these may be combined with a security key or certificate for more robust authentication. May be combined with presence sensing for automatic device locking and unlocking. For security critical environments, enhanced sign-in may be used via specialized hardware for additional protection against biometric data manipulation.

  - **Passkeys** Allow for passwordless authentication on sites and applications using an external device utilizing methods such as fingerprints and PIN.
  - **Credential Guard** Employs hardware-based VBS to guard credentials. They are stored in an isolated environment. Even should malware gain access to administrator privileges, it cannot access these secrets. Remote Guard may be employed in a remote environment - in such a scenario, a compromised host cannot leak credentials.

- **Cloud security** Modern versions of Windows feature extensive cloud integration for applications and devices, with a focus on enterprise environments.
  - **Mirosoft Entra single-sign on** Cloud-based identity management for enterprise environments. Entra simplifies Windows deployment on work-provided devices, access to applications and resources, cloud management of devices and device sign-in with work and school accounts.
  - **Remote wipe** Enables lost devices to have their data wiped remotely. Options are available from simply removing associated accounts to a complete data wipe and device reset.
  - **Modern device management through MDM** An enterprise management tool for the enforcement of company security policies on devices. The majority of the features listed so far may be enforced.

- **Universal Print** Enables a Zero Trust security model for printers, allowing them to be separated from the rest of the infrastructure.
- **Windows Autopatch** Automates the update process for Microsoft software for an enterprise. Included products are Windows itself, Office, Edge and Teams.

Some Windows security components have shown potential for improvement in the hands of researchers. Naik et al. found the Windows firewall to be incapable of detecting DoS DoS (Denial of Service) attacks. By integrating fuzzy reasoning components, this functionality was successfully added, making the integrated firewall better in this regard than many commercial solutions [9]. However, others show worrying vulnerabilities. Kim et al. found that on systems lacking hardware support, Windows Hello credentials were improperly stored and could be extracted. Normally, Windows Hello credentials are stored in the TPM on supported systems [10].

Looking at these features, two developments become apparent in the Windows security field. Firstly, practically no critical component relies solely on software based security anymore. Extensive hardware-driven security features may be found in areas from the core of the boot sequence to how malware is contained via virtualization. Secondly, extensive cloud-based or otherwise online-enabled security functionality may be found in several aspects of the operating system. It may best be understood as part of a larger ecosystem of online and hardware-driven security, well past any notions of merely being a self-contained software environment. Such development comes with a number of real concerns regarding user privacy. An increasingly online operating system must exchange more information with its creators. How can users feel secure in what kind of information is gathered about them and their systems, and who gains access to this information? Should these increasingly involved security features be viewed as benign, or as methods for a large corporation to exercise

greater control over consumers and their privacy?

Let us begin by taking a closer look at the full list of implications of the hardware requirements for Windows 11, and what these requirements may evolve into. The TPM 2.0 module's and Pluton security chip's features were previously covered in this thesis. In his July 2022 blog post "The dangers of Microsoft Pluton (updated)", Gabriel Sieben covers the true specifications and requirements of these technologies, their implications and what may be possible determine on a system adhering to the full set of features [11].

Microsoft was originally criticized regarding not only the strictness of the system requirements, but also their vague justifications. Uncovering the full list of features as well as their functionality required parsing together different sources by the author, but as per Sieben's blog, we have the following list of components [11]:

- A full TPM 2.0 implementation, developed by Trusted Computing Group (TCG)
- SHACK (Secure Hardware Cryptography Key) implementation
- DICE (Device Identifier Composition Engine) implementation, also designed by TCG
- Robust Internet of Things (RIoT) specification compliance

Furthermore, Pluton comes with other security features that were previously only required for OEM (Original Equipment Manufacturer) systems that adhered to the Secure-core PC specification. These specifications are:

- Dynamic Root of Trust for Measurement (DRTM)
- System Management Mode with Device Guard
- Memory Access Protection

- Hypervisor Code Integrity (HVCI)

Furthermore, systems adhering to these specifications now have to disable the 3rd party UEFI certificate by default, meaning Linux operating systems cannot boot by default. On Linux systems, the Pluton chip operates as a standard TPM 2.0 implementation.

Regarding the specific hardware requirements of Windows 11, the functionality at the root appears to be an optional feature of VBS, called memory integrity, also known as Hypervisor-protected code integrity. Using HVCI on systems lacking support for mode-based execution control (MBEC) leads to greatly reduced performance. MBEC thus appears to be the closest thing to an exact identifiable requirement.

Describing these requirements helps little without the appropriate background knowledge of why they were chosen. Aside from Windows-based computers, Pluton was also developed for the Xbox range of home video game consoles and Azure Sphere, an application platform for IoT (Internet of Things) devices. It will be helpful to take a look at Microsoft's "Seven Properties of Highly Secure Devices" [12]:

- Hardware-based root of trust, to ensure cryptographic keys cannot be forged. Side-channel attacks are protected against physically. Hardware has two strengths over software when it comes to protecting against attackers: hardware built for a single purpose cannot be repurposed as an attack vector by a malicious actor. Furthermore, hardware is suited for protecting against hardware-based attacks.

- Small trusted computing base, where private keys are stored in a hardware vault, and software is in self-protecting layers. A small trusted computing base has the benefit of a small surface area for attackers to make use of, and a lower chance for bugs or features that may serve as vulnerabilities.

- Defense in depth, ensuring every threat is protected against with multiple mitigations, and a successful attack on one layer does not compromise the others. In a single-layer system, just one vulnerability may expose the entire system, or more.

- Compartmentalization, where software components are protected by hardware barriers to aid in the above. An example may be found in the use of virtual machines.

- Certificate-based authentication, where a signed certificate proves the device's identity, rather than a password. Certificates carry the benefit of not being able to be stolen, forged or otherwise tampered with.

- Renewable security: a device with such security can automatically update to a more secure state, even after a breach. Lower layers must rebuild and renew higher level security. Remote attestation (changes to the system being automatically reported) and rollback prevention ensure that a device cannot be reverted to a vulnerable state.

- Failure reporting ensures that software failures are reported to a cloud-based system. With enough reports, even rare occurrences can be accounted for. This allows the building of a system's so-called immune system.

With Pluton, concerns arise regarding the extent of how far the technology may be pushed to enable functionality beyond what may be considered security-driven, and instead infringing upon the privacy and rights of users. In Sieben's blog, an example is given of a potential anti-cheat solution, where the technology may be used to exert great control over the user for the supposed benefit of preventing cheating in video games. Using this technology stack, it would be possible to cryptographically prove that the device has been securely booted; the kernel module is loaded; said

module is certain to not have been modified in any way; Windows is up to date; and through HVCI/MBCE code integrity, code injection is nigh impossible. Such a stack may also be used to strengthen DRM (Digital Rights Management) measures by not only preventing code injection, but by enforcing such a scheme for access to software. Further investigation is warranted regarding the implications of Zero Trust Computing. Under this scheme, a device has secret keys that cannot be accessed by the CPU or extracted from the system. Such keys could then be used to create documents and other content that cannot be accessed by anything other than a specific device. One may imagine a scenario, in which an organization enforces Pluton and Azure enrolment on all devices. Any content created would be cryptographically certified and non-certified content could be disallowed. Thus, users could be allowed to only read approved documents on authorized systems. The system is resistant to tampering and the extensive stack of Pluton-powered security features cannot be disabled.

It is important to note that these scenarios have thus far been entirely hypothetical; Pluton and TPM are presently advertised as being entirely security-focused - however, keeping potential future applications of this technology in mind is important, lest we find ourselves in a boiling frog situation, where increased security is used as an excuse to push for increased control over devices and their users. Whether these scenarios extrapolated from potential capabilities of currently promoted technology will ever come to be realized can ultimately only be answered by the passage of time, so let attention instead be turned to more contemporary concerns. Telemetry in Windows has been a growing concern ever since the introduction of Windows 10, which was extensively promoted as a free upgrade to existing Windows users, a trend that would continue with Windows 11. For a commercial company to promote its market-leading product free of cost could only be regarded with healthy suspi-

cion; if not monetary, the cost could be expected to be paid in other ways. Telemetry refers to the collection and processing of user data. A privacy-concerned user would naturally then be concerned about why, when and how such data is gathered, what it contains, whether it is truly anonymous, how it is transmitted, and to whom. Improperly handled data is not only a severe privacy concern, but also presents an additional attack vector for malicious actors. To establish a baseline, the categories of data gathered under a "Full" level of telemetry in Windows 10 will be covered, as this forms the basis for modern Windows telemetry. The categories are [13]:

- Common data, including OS version, device type, user ID, diognostic level and device ID
- Inking, typing and speech utterance data, including type of pen, ink strokes, speech recognition results
- Content consumption data, such as video metadata, music metadata, information about books accessed on the Windows Store, and image metadata and method of viewing
- Device, connectivity and configuration data, such as device properties, device capabilities, networking information and peripheral information
- Product and service performance data, such error codes and messages, system log files, and dumps resulting from crashes and hangs
- Software data, including information on installed applications, drivers, and updates, and when and how these were installed
- Licensing and purchase data, such as product type, name and price, as well as existing subscriptions

While the types of telemetry collected is known, this still leaves a number of real concerns for users:

- Who has access to this information, and for how long?
- Which parties is the information sold to? Is it adequately anonymized, and can it be guaranteed it does not leak?
- Why is this information collected? Is it used to improve the product, or simply to generate revenue?
- Will additional types of information be collected in the future? Can Microsoft be trusted to accurately communicate on any changes to telemetry?

A practical example of this change in design from a relatively self-contained operating system to an always-online telemetry-reliant ecosystem may be seen when comparing out-of-the-box versions of newer and older versions of Windows. The Youtube channel "The PC Security Channel", in their video "Has Windows become Spyware?", compared fresh installs of both Windows 11 and Windows XP 64-bit [14], [15]. The difference was stark: using the network analyzer Wireshark, outgoing DNS packets were analyzed on devices after first boot. Windows 11 was found to automatically connect to a number of sites for trend research and privacy management companies. These sites were typically bereft of meaningful information and were even blocked by default by a third-party ad-tracking blocker. MSN.com and Bing were also both listed, despite no searches for either being made yet on the system. Meanwhile, the Windows XP system had barely any DNS traffic at all - the only notable URLs accessed were related to Windows Update. While it is true that the features offered by these different versions of the OS are vastly different and thus direct comparisons may not be the most accurate, it still does not inspire confidence that millions of new devices connect to untrustworthy third-party sites as their first course of action.

2.2 Changes over time

The focus of this thesis pertains to modern versions of Windows in common use today. **Windows 10** and **Windows 11** are the current versions receiving active support. Popular earlier versions such as **Windows XP** and **Windows 7** will be subjected to the same tests as their successors. The purpose of this inclusion is not to argue for the use of outdated operating systems, but to accurately gauge the extent of their deprecation and to satisfy an academic curiosity. The aim is to attain an accurate picture of the extent of security vulnerabilities faced by users who continue to use these operating systems. It will then be possible to draw meaningful conclusions and give accurate advise regarding the prospect of upgrading or replacing these systems. This will serve to contrast the official stance of the developers, whose recommendations for users to upgrade should be critiqued from two different angles. Firstly, one should hopefully be able to rely on their judgment regarding the criticality and necessity of upgrading the Windows operating system, be it for the purposes of replacing an older version that is past its life cycle, or to continue updating a current operating system with new features. The developers have access to the greatest extent of information regarding the state of the operating system in multiple facets and should thus be the most qualified to make assessments regarding its security, should the operating system in question be one that recently entered its end of life, or one still receiving active support for years to come. However, this expertise should not be trusted blindly, as it must be considered that a commercial company like Microsoft has other motives to push users to upgrade. Acquiring and maintaining market share is arguably one of the main goals of most companies and promoting new security features is an angle that may be utilized more depending on market trends. Whether these new features are robust and offer meaningful improvements compared to those offered by older, more established operating systems is something that should be considered. One must keep in mind that developing

security features requires much more testing and additional expertise compared to regular features, particularly if these features become widespread or a core part of the system, as potential vulnerabilities may have catastrophic implications. Outside the field of operating systems, one may look at the Heartbleed bug found in OpenSSL [16], which enabled access to sensitive data in affected systems. Around a sixth of the Internet's secure web servers were believed to be affected at the time.

While the focus of the thesis will be on modern as well as previously popular, older versions of Windows, it is helpful to present a historical overview of its legacy, particularly the development of major security features. This will help us establish a baseline frame of reference and grant perspective once each operating system is taken a look at in more detail [17]. See figure 2.1 for a quick overview of these features.

The early days of Windows security were dire from a modern perspective, and this was the case for a surprisingly long time. From MS-DOS, released in 1981, until Windows 98, released in 1998, the operating system lacked in many security features considered rudimentary today. Login features were basic and multiple accounts weren't supported. Password were not stored in the operating system, and logging capabilities were limited. The file system used at this time, File Allocation Table (FAT) had no security measures or proper access control.

A major step forward on the security front would be seen with the introduction of Windows NT. With NT, FAT was replaced with the New Technology File System (NTFS), NTFS extended file and folder name lengths, introduced proper access control, logging and encryption.

Windows 2000 would permit for asymmetric encryption of private keys via the Data Protection Application Programming Interface (DPAPI). Furthermore, it introduced the AD (Active Directory) service for network resource management.

With Windows XP, multi-user functionality would be greatly expanded. The

credentials for multiple users were now locally stored and a password reset wizard was introduced. AutoPlay managed removable media, and the Windows Security Center would constantly monitor the system. The Encrypting File System (EFS), introduced alongside NTFS, would receive further improvements [18]. It would permit users to encrypt and decrypt files and folders easily, while also supporting data recovery and access sharing.

Windows Vista, while not a popular version of the OS, would introduce several mainstay features. User Account Control (UAC) prevented changes on the OS without the approval of the administrator. Windows Defender would be shipped as the first bundled antivirus solution included with a Windows release. Bitlocker, a full volume encryption solution was also added.

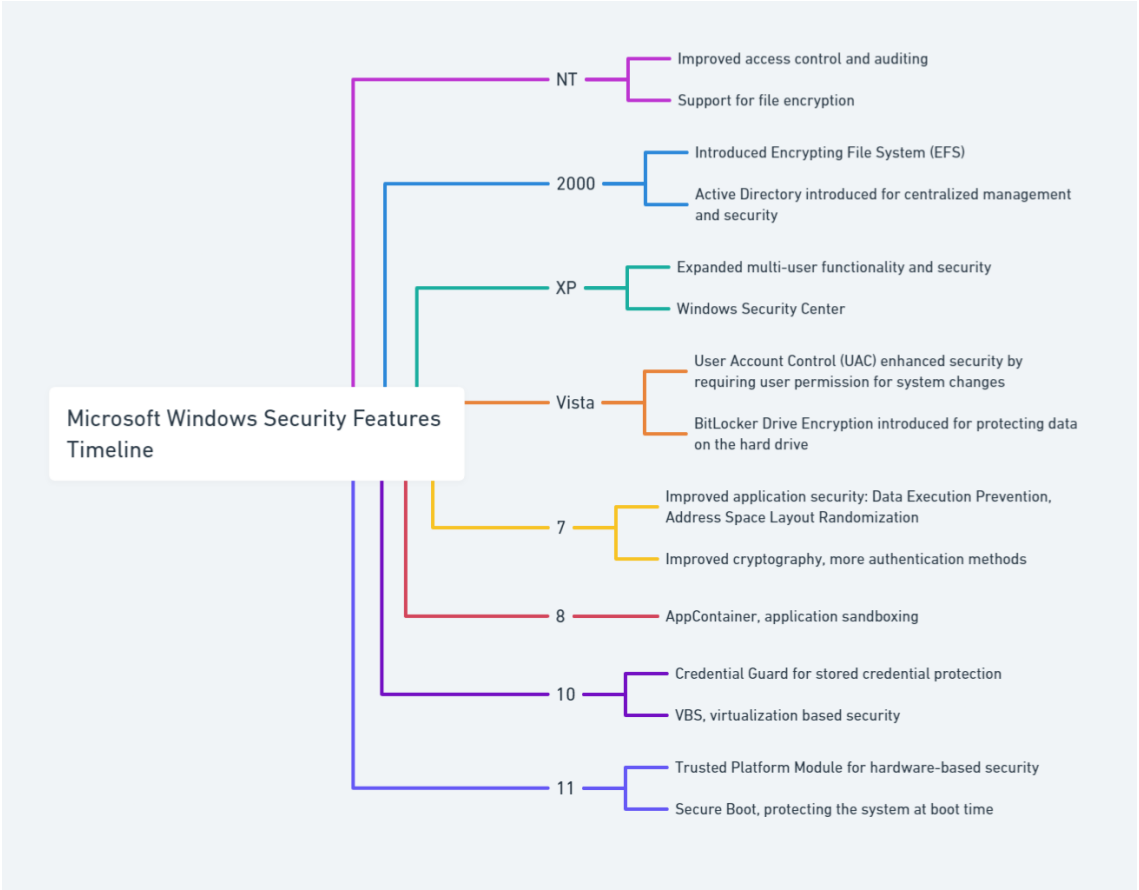

Figure 2.1: Important Windows security additions by version. [1]

Windows 7 shifted the focus towards application security and memory integrity. Data Execution Prevention (DAP) prevented code injection, while Address Space Layout Randomization (ASLR) aimed to prevent memory-based attacks via memory randomization. Cryptography would see improvements on several fronts: new algorithms such as AES and Triple DES were supported. Elliptic curve cryptography was now also supported, and the LAN manager now used NTML2 hashes [19]. Windows 7 would also see support for new authentication methods, such as biometric and smart card access. These features could be combined with traditional passwords as a form of two-factor authentication. Windows 8's major contribution towards OS security came in the form of AppContainer. A sandboxing measure, it would restrict the resources available to a program. Finally, Windows 10, the older of the two Windows operating systems still within their life cycle, introduced Credential Guard, which protects stored credentials against theft. It utilizes virtualization-based security (VBS) to isolate such sensitive data. Should a malicious actor gain access to administrator rights on the system, they still can't access credentials protected by VBS [20].

2.3 Research questions and objectives

The research part of this thesis will focus on aspects of security and privacy of different versions of Microsoft Windows, and how features related to these affect system usability and performance. The aim is to establish a meaningful timeline, showcasing how these aspects have changes over time, for better or for worse. Modified versions of Windows, as well as some alternative Linux-based distributions will be used to provide additional data and context. While conducting the research, these

[1]This graph has been generated with the assistance of AI using Whimsical at www.whimsical.com. The goal of using AI here was to aid in designing the structure of the graph and for basic text input before editing. Prompt used: "Timeline for briefly showcasing important security features for each release of Microsoft Windows. Nodes: NT, 2000, XP, Vista, 7, 8, 10, 11. Three rows"

questions will be kept in mind:

- How has the level of security across different versions of Windows evolved over time? Is it possible to detect meaningful breakpoints in feature levels, where a newer version can be deemed objectively more secure? Can Windows be considered adequately protected against malware on its own and if so, when did it become secure enough?
- How has Windows' respect for user privacy changed over time? How invasive has data collection become?
- Do altered versions of Windows provide benefits on these fronts? Can third-party modifications and toolboxes provide users with meaningful improvements, or are they more likely to be placebo, or unduly compromise system security?
- How viable of an alternative is a modern, popular distribution of Linux such as Ubuntu? Does it offer meaningful benefits or drawbacks in terms of security and privacy?
- Are system performance and the user experience negatively affected by security and data collection measures? Can some features be argued to come at excessive performance cost?

These answers may then be attempted to be provided:

- Is Windows on a positive trajectory, or are improvements in security and features foreseen to be overshadowed by concerns regarding privacy?
- Which versions can still be recommended for use today, and for whom? What kind of advise can be given to different user groups, such as casual users, power users and gamers?

2.4 Significance of the study

Individual tests and reviews of the topics covered in this thesis are not too difficult to find. Benchmarking the performance of individual Windows security components, or monitoring network traffic for suspicious packets are by no means new points of interest for cybersecurity enthusiasts. However, such studies come with multiple caveats that this thesis aims to address, namely:

Historical context: most tests of the nature conducted in this thesis apply to a specific version of the operating system. By conducting these tests on multiple different operating systems, some historical, greater context is attained that helps in mapping the actual developments of relevant features.

Up-to-date testing: security feature testing is a practice that is very sensitive to the passage of time. Operating systems and their features that were previously known to be secure may lose this status over time - old tests may become obsolete at best, or misleading at worst, if the findings therein are no longer correct. While some aspects of this thesis will certainly become obsolete over time, others will be relevant for much longer. In this field, up-to-date data carries significant value.

Standardized testing: by performing the same tests on as wide a variety of configurations as feasible, further historical context is gained and it may be easier to estimate future development. It will not be necessary to parse together results from multiple different sources while trying to account for differences in testing methodology.

Valuable conclusions: having run standardized tests on different configurations, practical data will hopefully be attained that can then be used to provide meaningful insight, something that limited testing cannot provide. Examples include being able to give general advise on these matters to different user groups, and attempting to predict the future trajectory of the Windows ecosystem.

3 Literature review

3.1 Key concepts

Developments in the malware research field indicate that both malware research and malware development are becoming increasingly complex. Traditional heuristics, such as signature detection are no longer sufficient - models are trained to predict and identify samples as they appear. At the same time, malware authors create increasingly sophisticated code that is able to alter itself. Currently, this cat-and-mouse game is going in the defenders' favor.

3.2 Past studies on the research problem

Maniriho et al. (2024) present an extensive literature review of Windows malware detection and surrounding concerns in their paper, "A systematic literature review on Windows malware detection: Techniques, research issues, and future directions". Covering a range of 219 research papers, the review focused on malware detection in executable files. Particular focus was given to trends surrounding machine learning (ML) and deep learning (DL) as well towards identifying experimental biases [21].

The research questions pertain to the types of malware detection techniques; the used datasets; used ML and DL algorithms; performance evaluation metrics; experimental factors and biases; and research challenges.

Challenges that were found regarding the data sets used in reviewed papers in-

cluded incomplete details and lacking timestamps. Some data sets lacked sufficient benign and malware samples, and weren't kept updated to allow them to stay relevant.

On the machine learning front, ML was found to have been steadily gaining popularity, with Support Vector Machine and Random Forest as the most widely used algorithms. Deep learning (DL) algorithms were found to be even more prevalent, with convolutional neural networks and recurrent neural networks being the most common.

Common challenges faced by the majority of papers included a vast difference between malware and benign samples used. Rates of false positives and false negatives were high - it was noted that such engineering could be intentional on the part of attackers, as validating these these results can be tedious. Overfitting and underfitting are an ever present challenge in ML and DL, where the model may perform well on the training data, but poorly on the test data. Furthermore, overfitted models are subject to integrating noise. ML algorithms were noted for requiring vast amounts of data, as well as human processing. Adversial attacks present a real challenge for ML and DL based detection - adversial training has therefore been used for increased robustness of the models.

Moussaileb et al. meanwhile focused on ransomware in Windows environments in their 2021 paper, "A Survey on Windows-based Ransomware Taxonomy and Detection Mechanisms: Case Closed?". A general trend seen in recent years indicates that while ransomware attacks are in decline, their focus has also shifted from individuals to enterprises. Government, retail, manufacturing, education and consulting were listed as the industries most affected [22]. It was noted that from the attacker's perspective, unbreakable cryptography is paramount, as it is the only way to guarantee the desired outcome of extorting money from victims. The majority

of ransomware employ a multi-phase attack of delivery (searching for vulnerabilities and penetrating the system), deployment (loading libraries), destruction (querying target file system, receiving of encryption keys, encryption) and dealing (presenting the victim the steps required to regain access to files). Ransomware use several methods to avoid detection. Polymorphic ransomware can alter itself to better avoid detection, using methods such as code obfuscation, sandbox evasion and polymorphic blending. Typical detection methods look for changes in the beginning of files - advanced ransomware can then avoid editing bits likely to cause detection. VirusTotal reported an average detection rate of about 83% for 11 types of ransomware - by employing several different obfuscation techniques, the researches were able to lower that rate to 32% for day one samples. Methods used for detection include monitoring DNS traffic for gibberish requests, applying machine learning to network traffic monitoring, and employing system honeypots that detect suspicious activity in file traversal. In conclusion, the researchers found the defense against ransomware to be in strong standing, as a multitude of solutions for accurate detection exists in current literature, and successful ransomware deployment relies on breaches on multiple levels.

An assessment of the capabilities of neural networks in the field of security is in order. In their 2017 conference paper, "Towards Evaluating the Robustness of Neural Networks", Carlini et al. discuss the challenges of employing neural networks in this area. They posit that some methods proposed to be effective in hardening neural networks are in fact not, and that several different attacks can be used to prove this [23].

Securing neural networks has been a challenging task. Adversial attacks can alter the data used to an imperceptible degree, yet result in a different classification. One common defense method in use has been defensive distillation. Distillation is a

process whereby knowledge from a deep neural network (DNN) is transferred to a different neural network. This is useful for resource constrained devices in particular [24]. Papernot et al. in their paper "Distillation as a Defense to Adversarial Perturbations Against Deep Neural Networks" found distillation to reduce the effectiveness of adversial sample crafting from 95% down to 0.5% in a best case scenario, and thus proposed models utilizing it to be more suitable for security sensitive settings [24].

Carlini et al. however found this recommendation to be untenable, as they were able to construct new attacks that distillation did not help against. Furthermore, they showed that adversial examples from an unsecured model are transferable to the distilled, secured model. Thus, any defense must demonstrate its ability to break this transferability.

The effect that adversial examples have cannot be understated: not only do they circumvent the application of the model, but render it wholly unsuitable for use. A model used in self-driving cars that can be fooled into classifying a car as a non-car is certain to have catastrophic consequences. Malware that manages to have itself classified as benign has completely circumvented the system - a video containing non-speech audio that a model detects as speech may be used for a variety of drive-by attacks.

Concluding, the researchers suggest that anyone looking to demonstrate the robustness of their model employ powerful attacks, such as those developed for the study, and prove that transferability fails.

Thus far, the assumption has been that of malware as executable files. This is not always the case, as fileless malware is an emerging threat, as covered by Kara in his 2023 paper "Fileless malware threats: Recent advances, analysis approach through memory forensics and research challenges" [25]. Fileless malware does not

depend on a host file to execute. This allows it to easily bypass intrusion detection systems (IDS). Such malware often operates by embedding itself into some part of the system, such as the Windows registry, or through process injection. Fileless malware differs from traditional file-based malware in its mode of delivery, execution and persistence.

The end target of fileless malware is typically RAM. Several analysis methods exist, as described by Bozkir et al. in their 2021 paper, "Catch them alive: A malware detection approach through memory forensics, manifold learning and computer vision". Analysis can be divided into three categories: static (API calls, Opcodes, control flowcharts, N-grams), dynamic (function calls, function parameters, Windows registry, network activities) and memory-based (memory dumps, DLLs, network activities). Memory-based analysis in particular is promoted as an efficient method [26]. Memory is where malware is in its most vulnerable state, as it is not encrypted. One method that can be employed is to compare suspected infected memory with known clean memory.

Kara further cites the ability of memory analysis to detect malware that leaves no traces in storage media. Static analysis, meanwhile, is effective against code obfuscation and polymorphic and metamorphic malware. The primary challenge of dynamic analysis is avoiding detection.

The nature of fileless malware makes testing in a virtual environment the only practical method. However, smart malware can detect this, and shut itself down. Machine learning is stated a strong candidate for a future method to aid in detecting fileless malware [25].

Further challenges found in existing literature include the fact that automatic analysis cannot find signatures for fileless malware; that its behavior is unclear; it has several qualities that are unrelated to each other; and that current analysis techniques are inadequate.

Areez et al. in their 2021 paper "Windows PE Malware Detection Using Ensemble Learning" discuss the different methods machine learning and deep learning methods may be designed to aid in malware detection. They claim that traditional methods of malware detection are no longer sufficient. Signature analysis is time-consuming and relies on existing strings, making it poorly suited for responding to new threats, while heuristic analysis is also time-consuming and prone to errors [27].

Ensemble methods refer to the practice of using multiple machine learning algorithms for improved accuracy while reducing chances of overfitting. The paper suggests an ensemble learning framework utilizing different kinds of neural networks in the initial phase, and a machine learning method in the later stage.

The authors refer to the work of Cakir et al., who employed DL methods for malware classification. They used natural language processing (NLP) methods for this task - Word2vec, a method which generates relationships between words. This is used with disassembled code from an executable. With this approach high levels of detection accuracy were achieved [28].

Areez et al. likewise focused on Windows executables, both malware and benign files. Their method of combining DNNs with ML was able to outperform other methods. An advantage was cited in their end-to-end learning process without the need for manual feature engineering, though a downside was that both types of samples had to be identified and labeled by humans. Further study towards frameworks that could do the same while unsupervised was proposed [27].

Finally, a look at the future of AI and ML in a recent analysis by Microsoft contributors. In "Securing the Future of Artificial Intelligence and Machine Learning at Microsoft", the authors examine the various challenged faced today and in the near future in the field of AI-driven security [29]. Their findings indicate the ML

models are not capable of adequately discerning between malicious and benign, but anomalous input. Furthermore, many data sets are publicly sourced and lacking in curation. This allows actually malicious data to over time be classified as trustworthy. Furthermore, the output of AI/ML is too easily blindly trusted, while the process itself is largely treated as a black box, leading to issues of accountability. AI is subject to unique attack vectors, such as image, video and gesture data. Ways AI could deal with malicious input includes monitoring input that strays too far from the norm, such as at odd times, or too quickly, and when multiple users input similar suspect queries. Furthermore, AI must be designed as trustworthy, as it is divulged with vast amounts of information, some in excess of what it should likely be given, and it will then process and reveal such information to users and other AI. To address these and other issues related to the use of AI and ML, the researchers proposed a number of practices that should be adopted: penetration testing should be extended to AI and ML, just as in other cybersecurity fields; appropriate training could be employed that helps clarify design principles and who is responsible for implementing them; ML and related data could be further hardened; establishing a centralised auditing and forensics library; improved vernacular detection and categorization; dedicated fuzzing tools for developers to test their AI with.

4 Operating system hardening: Design of environment

4.1 Operating systems

The first choice of a legacy Windows operating system is Windows XP. Originally released in October of 2001, XP reached end of support for client versions in April of 2014. Windows XP came in both 32-bit and 64-bit variants and received three service packs through its lifecycle. The version of choice for this experiment is **Windows XP 64-bit SP2**. The reasoning for this choice is that the goal is to test the operating system as it would be used today, rather than during the peak of its popularity. Thus, 64-bit support is of far greater importance. Windows XP did receive a third Service Pack, SP3, but only with 32-bit support. Thus, this update is skipped.

Windows 7 was a critically and commercially successful version version of the operating system, released in October of 2009, and reaching the end of extended support in January of 2020. Here, the choice of version is clear - **Windows 7 64-bit SP1**.

With these two legacy operating systems, some practical issues immediately present themselves. Windows Update, used to deliver security updates, no longer functions on either Windows XP or Windows 7, merely returning an error message.

Thus, out of the box, these operating systems lack critical security updates, only including whatever was packaged with the installation media.

The goal with this experiment is to evaluate operating systems in their optimal out-of-the-box state. This means a lack of third-party modifications unless otherwise desired, as these would introduce unwanted variability and unreliability to the results. However, evaluating these legacy operating systems on their security features, while leaving them lacking in critical security updates, would likewise give us an inaccurate picture of their performance. Fortunately, a solution exists that allows us to remedy this situation while ensuring the intended end result. Legacy Update is a third-party service operated by volunteers that restores Windows Update functionality on legacy operating systems [30]. It only installs official Microsoft-provided updates, making it suitable for this experiment. See figure 4.1.

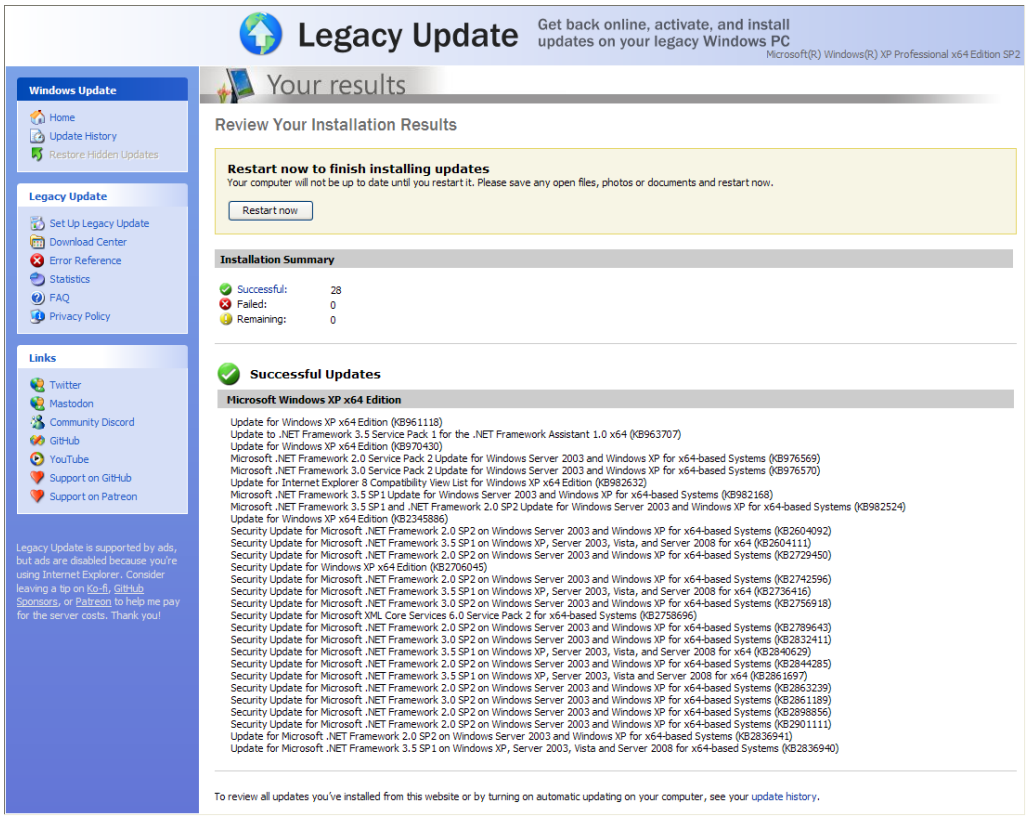

Figure 4.1: Legacy Update in use under Windows XP

It should be noted that even after installing all available updates, one major concern remains under Windows XP: no antivirus program. The operating system warns the user with nags, strongly urging the user to install one. This presents us with a challenge - an unprotected legacy system is certain to be extremely vulnerable to attacks, yet adding a third-party antivirus program would introduce variables. Thus, tests will be conducted in two scenarios for Windows XP - without antivirus, and with antivirus. This will hopefully give us insight into whether this additional layer of security will meaningfully aid in making this legacy operating system usable today. The antivirus of choice is Avast.

By default, Windows 7 ships with no antivirus, and presents the user with the same nags as XP. However, Microsoft Security Essentials is available via Windows Update. It is Microsoft's first foray into building a first-party antivirus solution, and while it has been superseded by other solutions on newer versions, it still receives up-to-date virus and spyware definitions under Windows 7, making it suitable for the experiment.

Moving on to contemporary versions of Windows, installing and updating them was more straightforward. These versions still receive updates via Windows Update and are able to install them more efficiently, requiring fewer restarts, and rechecks for remaining updates. The choice of version leaves little option, with **Windows 10 Version 22H2** the only up-to-date version of the operating system available to consumers.

Windows 11 continues this trend, offering no real options beyond **Windows 11 Version 23H2**. With Windows 11, this version will serve as the baseline, as the environment will be altered with via third-party methods. "The Ultimate Windows Utility" by Chris Titus will serve to contrast a base installation. Its features will be covered in the Tools section.

An even further modified installation will be achieved via the use of a custom

third-party ISO. **Windows 11 Ghost Spectre** promises to 'debloat' the system, remove telemetry and improve performance. Several versions are available depending on the extent of desired alterations - covering these is beyond the scope of this section. The **Superlite + Defender** version is chosen, as this goes further in the changes it makes, while still enabling Windows Defender. Notable defaults include the use of the administrator account as the default user account, and disabling UAC (User Access Control). It also bypasses the requirement for logging into a Microsoft account. This is the author's personal choice of operating system for the time being.

Finally, a Linux distribution will be chosen, to evaluate the prospect of replacing Windows entirely. There are very many distributions available for different use cases and with varying levels of support. For this experiment a long-standing distribution that is suitable for many different users is needed, thus **Ubuntu 22.04 LTS** is chosen.

See figure 4.2 for an overview of the Virtualbox environment.

To recap, the operating systems used for this experiment are as follows:

- Windows XP 64-bit SP2 w/o antivirus
- Windows XP 64-bit SP2 with antivirus
- Windows 7 64-bit SP1
- Windows 10 22H2
- Windows 11 23H2
- Windows 11 23H2 with utility toolbox (The Ultimate Windows Utility)
- Windows 11 23H2 custom (Ghost Spectre Superlite + Defender)
- Ubuntu 22.04 LTS

4.2 Virtual environment

The virtual environment for this experiment is Virtualbox 7.0 by Oracle. The relevant specifications of the host machine are as follows:

- Windows 11 23H2 Ghost Spectre Superlite
- AMD Ryzen 7 5800X3D 8C/16T
- 64GB 3200MHz CL16 DDR4
- Mushkin Vortex 2000GB PCIe 4.0 NVME SSD system drive

Each VM is given eight cores with a 50% execution cap to work with. This ensures the responsiveness of the host system. Legacy OS are given 8192MB of memory, while the rest are given 16384MB.

For the malware testing portion, the VM environment will be hardened:

- Removal of guest additions
- Disconnection from the internet
- Disabling shared clipboard and drag-and-drop
- Reversing any possible infection with snapshots

4.3 Tools

- Phoronix Test Suite is a multi-platform suite for system benchmarking. A large number of tests are available.
- Latencymon is a PC latency monitoring tool, capable of measuring several kinds of latencies, as well as the number of pagefaults thrown by processes.

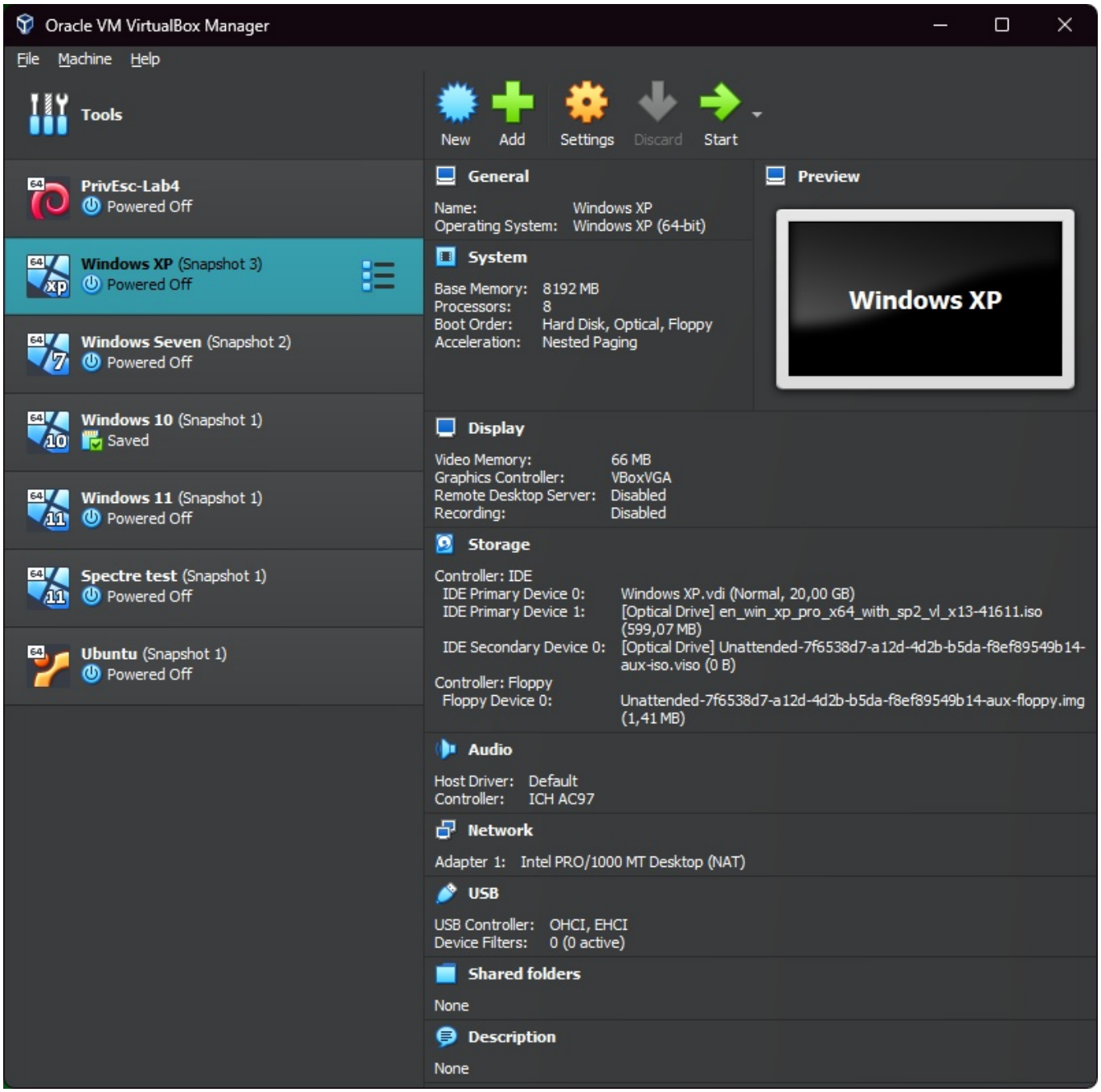

Figure 4.2: Virtualbox overview

- Wireshark is a network protocol analyzer, suitable for packet capturing
- Firefox 43.0, for use on Windows XP as the default browser unless otherwise noted
- Firefox 115.0, for use on Windows 7 as the default browser unless otherwise noted
- Firefox 124.0, for use on newer versions of Windows, as well as on Linux
- Internet Explorer 8, used on Windows XP and 7 where necessitated by testing
- Malwarebytes, an anti-malware solution used for comparison testing on Windows XP
- The Ultimate Windows Utility, a toolbox for disabling Windows features and data collection. The exact set of options chosen will be covered in the test section [31].

4.4 Payloads

There exists a vast range of potential malware sources online. Two sets of tests are conducted, one consisting of benign files, while the other is a selection of genuine malware.

WICAR.org provides a collection on benign samples to be tested in a web browser [32]. These are listed and described on their website, and this information is represented in table 4.1.

For the genuine malware portion, one of the numerous available repositories available on Github is chosen. The one provided by Enderman offers several categories: 'enderware' (custom homemade prank applications), fake scanners, jokes, 'modern',

ransomware, rogues and trojans. A few from each of these categories are chosen, save for 'modern'. See table 4.2.

A proper comparison between Windows and Linux malware is not possible, as Windows executables do not run on Linux. However, Linux-specific malware may still be used to run a limited test for the sake of comparison. Here, a small selection from MalwareBazaar will be used [33]. Ransomware is by far the most common type of malware on Linux, likely due to its use in enterprise environments on on servers. These files lack proper names, being instead referred to with SHA-256 hashes. See table 4.3.

4.5 Overview of test environment

The order of tests for each operating system is a as follows:

- Run Wireshark on a freshly booted system with no active programs, monitoring DNS traffic until no new, unique traffic occurs. List down and categorize each entry.
- Attempt to run each malware sample. Mark whether the infection attempt is successful or a failure. When unsure, note it and presume success. Mark down any noteworthy behavior.
- Run Blender-BMW27-CPU Only under Phoronix, which automatically runs a total of three times and provides a score in seconds for each test, the average, as well as the deviation. While the tests are running, measure average and maximum system latency using Latencymon.
- Under Linux, substitute these programs for the closest equivalents, if necessary.

Table 4.1: Benign test files

Name	Description
EICAR test virus	The official EICAR.COM anti-virus test file. This is a 16bit DOS COM file and cannot run on recent OSes, but should be detected.
MS14-064 XP and below	All Windows NT/95/98/2000/XP IE3+ Internet Explorer Windows OLE Automation Array (pre-XP) CVE-2014-6332
MS14-064 2003 to Windows 10	All Windows 2003/Vista/2008/7/8/10 IE6+ Internet Explorer Windows OLE Automation Array (post XP) CVE-2014-6332
Java JRE 1.7 Applet	win32 (Java 7 JRE/JDK) Chrome Firefox IE Java 7 Applet Remote Code Execution (Browser Independent) CVE-2012-4681
MS03-020	win32 NT/XP/2003 IE6 MS03-020 Internet Explorer's handling of the OBJECT type attribute CVE-2003-0344
MS05-054	] win32 XP IE6 MS05-054 Microsoft Internet Explorer JavaScript OnLoad Handler CVE-2005-1790
MS09-002	win32 XP/Vista IE7 Internet Explorer 7 CFunctionPointer Uninitialized Memory Corruption CVE-2009-0075
MS09-072	win32 IE6 Internet Explorer Style getElementsByTagName Memory Corruption CVE-2009-3672
MS10-090	win32 IE6 Internet Explorer CSS SetUserClip Memory Corruption CVE-2010-3962
Firefox 5.0 - 15.0.1 exposedProps	Windows Firefox 5.0 to 15.0.1 exposedProps CVE-2012-3993
Embedded VLC AMV	Windows VLC v1.1.4 to 1.1.8 Browser Independent AMV invalid pointer CVE-2010-3275
Adobe Flash Hacking Team leak	Hacking Team July 2015 data leak Adobe Flash 18.0.0.194 Use After Free CVE-2015-5119
JavaScript Crypto Miner	JavaScript based Cryptocurrency Miner Consumes 70% of CPU and some RAM (Proceeds will be used to fund WICAR)

Table 4.2: Malware test files

Name	Description
Deskbottom Useroverflow	Enderware
Fake login prompt Fake Microsoft Support	Fake scanners
Screenscrew Trololo	Jokes
BadRabbit Cerber 5	Ransomware
Antivirus 2010 Windows Accelerator Pro	Rogue
DesktopPuzzle You Are An Idiot	Trojan

Table 4.3: Linux malware test files

SHA 256	Description
06abc46d5dbd012b170c97d142c6b679183159197e9d3f6a76ba5e5abf999725	Ransomware
f5de75a6db591fe6bb6b656aa1dcfc8f7fe0686869c34192bfa4ec092554a4ac	Ransomware
63cceba7384b2a44242598bdb8c9a2f87e455a0a88eaf9aa401fd43fb990b062	Botnet
713b699c04f21000fca981e698e1046d4595f423bd5741d712fd7e0bc358c771	Ransomware
556e5cb5e4e77678110961c8d9260a726a363e00bf8d278e5302cb4bfccc3eed	Ransomware
8b57e96e90cd95fc2ba421204b482005fe41c28f506730b6148bcef8316a3201	Ransomware

Listed here has been the initial test configuration. Some alterations were deemed necessary during testing. These changes and their reasoning are discussed in the implementation section as they become relevant.

5 Operating system hardening: Implementation

5.1 Windows XP

5.1.1 Privacy analysis

Windows XP was found to perform very little in the way of DNS requests on its own, only connecting to legacyupdate.net. No original Windows Update traffic was logged. The only additional traffic generated by an idling system was generic MDNS (Multicast Domain Name System) and ARP (Address Rsolution Protocol) queries.

5.1.2 Malware testing

The process of installing an antivirus on Windows XP could be described as a hurdle. Very few antivirus vendors still offer support, and some restrict it to SP3. The initial choice of Malwarebytes could not be installed, giving an error message during the process, likely due to the lack of up-to-date system-level certificates. Avast Antivirus was chosen as a substitute, and it did provide functional protection. The UI produced a less than ideal user experience, occasionally giving warnings about an invalid license, and producing upgrade nag windows that could not be closed. Avast successfully blocked several exe-based malware in succession. However, others avoided detection, notably those utilizing multiple files, such as (HTML-based fake

Table 5.1: Windows XP malware test

Name	**Type**	**No AV**	**w/AV**	**Notes**
EICAR test virus	Benign	✓	x	Attempts to run
Java JRE 1.7 Applet	Benign	∼	x	
Embedded VLC AMV	Benign	∼	x	
JavaScript Crypto M.	Benign	x	x	"Cryptominer attempt"
Deskbottom	Enderware	✓	✓	Partial success
Useroverflow	Enderware	✓	✓	100 users created. Lag
Fake login prompt	Fake scanner	✓	x	
Fake Msoft Support	Fake scanner	✓	✓	
Screenscrew	Jokes	✓	x	
Trololo	Jokes	✓	x	
BadRabbit	Ransomware	∼	x	
Cerber 5	Ransomware	✓	x	
Antivirus 2010	Rogue	✓	x	
Windows Accel. Pro	Rogue	✓	x	
DesktopPuzzle	Trojan	✓	x	
You Are An Idiot	Trojan	✓	✓	

login sites. The benign test files proved to be largely uninteresting. The selection was already initially cut down greatly, removing exploits that relied on Internet Explorer 6, an extremely outdated browser, as well as Adobe Flash. The remaining tests produced little in the way of interesting results, and it is doubtful this trend will change with later operating systems. For this reason, these tests will not be conducted in other environments, but they will be left here for the sake of posterity. See table 5.1.

An unintentional, normally positive side effect related to the networking hardware used was discovered during the setup process, as shown in figure 5.1. The author's Asus brand router came shipped with security features provided by their partner, Trend Micro. This secured the system against the "threats" on the EICAR test page. These features were disabled so as not to interfere with testing, but they provide an interesting potential field of study, as well as a practical means of additional security for legacy operating systems.

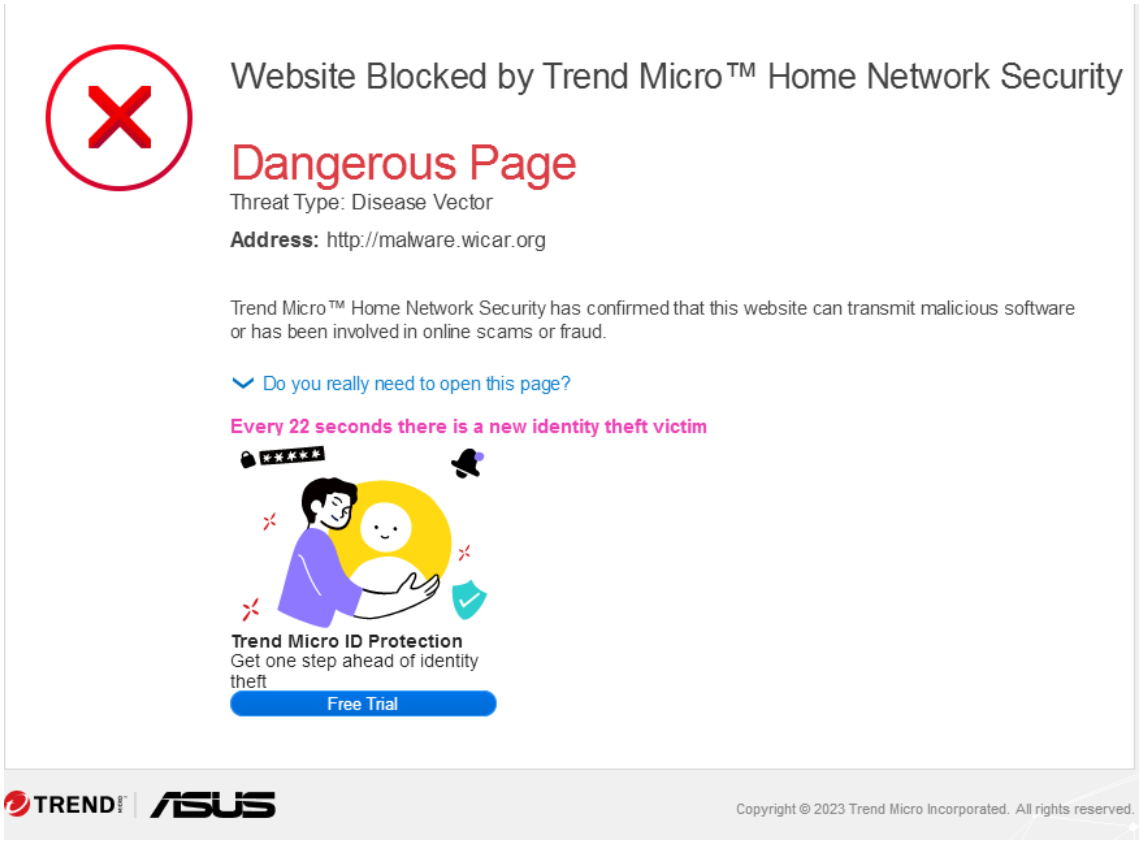

Figure 5.1: Router-based web security

5.1.3 User experience

The subjective Windows XP user experience in 2024 is best described as arduous. Very little can be trusted to work out of the box - manual setup is required to acquire a functional web browser, such as Mozilla Firefox. Internet Explorer is best treated as a dead program, as one would have to manually import new certificates to be able to browse nearly any site. Acquiring functioning versions of common programs requires research as a rule, and the degree to which said programs are outdated can vary greatly. Much credit should be given to Legacy Update for enabling this operating system to still receive legacy updates. No performance benchmarking programs that could provide data fit for comparison with newer operating systems

Table 5.2: Windows 7 DNS traffic

Target DNS	**Description**
time.windows.com	Time service
au.download.windowsupdate.com	Windows Update
teredo.ipvv6.microsoft.com	IPV4-to-IPV6 tunneling
crl.verisign.com	Name server provider
xxx.digicert.com	SSl authority
ipv6.msftncsi.com	Windows network awareness
xxx.trafficmanager.net	Traffic load balancer
activation.sls.microsoft.com	Windows activation service
go.microsoft.com	Generic Microsoft URL
wdcp.microsoft.com	Protection service endpoint

were able to be procured. Such data would also only serve academic curiosity at best, as any user looking to maximize computing performance is also likely to want the newest features to make use of that capability, such as DirectX 12 for gaming, which XP cannot provide. One major use case for a legacy operating system is legacy software. Just as new software may not run on a legacy system, so may legacy software not be operable on a new operating system. A prime example may be found in video games, which serve as a strong motivator for retro hobbyists to build dedicated period-correct machines, in terms of both hardware and software. Such builds are beyond the scope of this thesis, but do provide an interesting potential avenue of future study regarding the applications of legacy operating systems.

5.2 Windows 7

5.2.1 Privacy analysis

No suspicious DNS activity happens under Windows 7. All detected URLs are tied to important system functionality, such as the time service, Windows Update, networking services, certificate services, Windows activation, or system protection. See table 5.2.

5.2.2 Malware testing

Table 5.3: Windows 7 malware test

Name	**Type**	**Infection**	Notes
Deskbottom	Enderware	✓	Partial success
Useroverflow	Enderware	x	
Fake login prompt	Fake scanners	✓	
Fake Microsoft Support	Fake scanners	✓	
Screenscrew	Jokes	✓	
Trololo	Jokes	✓	Threat detected
BadRabbit	Ransomware	x	
Cerber 5	Ransomware	x	
Antivirus 2010	Rogue	x	
Windows Accelerator Pro	Rogue	x	
DesktopPuzzle	Trojan	x	
You Are An Idiot	Trojan	x	

As shown in table 5.3, Security Essentials did an adequate job at defending against the most serious of threats. Ransomware in particular was effectively blocked. A curious behavior was noted with 'Trololo', which disables the Explorer and taskbar and flashes a moving window with a mocking message. It was detected by UAC, and after allowing it to run, it deployed correctly, though Security Essentials warned about suspicious files at the same time, and asked for confirmation to clean the system. This removed the mocking window, but did not restore full system functionality otherwise. 'Useroverflow' was also stopped, which was not necessarily expected. It does nothing malicious on its own, but allows the user to select and arbitrary number of user accounts to create, which will then likely cripple the system. 'Deskbottom' continues its trend as the odd one out, displaying limited functionality via empty pop up windows, that does not appear to be detected by the antivirus in any way. Loose, malicious HTML files still avoid detection. These do little on their own as they still rely on user input for a successful scam to take place.

5.2.3 User experience

Windows 7, despite being a legacy OS, offers a considerably more user-friendly experience compared to XP today. Finding browsers and programs that still support it is not too difficult, though in some cases older versions of those programs will be necessary to procure. Gaming presents several concerns for users looking to extend the lifespan of their operating system. Steam, the largest digital PC video game platform, no longer supports Windows 7, meaning games that themselves still may support the OS, rely on a delivery platform that does not. Furthermore, DirectX 12 also lacks support for the full feature set, meaning games relying on it are unable to be run under Windows 7, even if the hardware used supports said functionality.

One likely contributing factor towards Windows 7's enduring success as the operating system of choice for a dedicated minority of users may be found in its user experience, particularly the user interface (UI). Windows 7 presents an intuitive and cohesive set of menus that are easy to navigate, and is a marked improvement over XP in this regard, requiring far less adjustment when returning to it after several years. Newer versions such as Windows 10 and 11 are much more iterative in this regard, and in fact still display leftover Windows 7 graphics in some of their less used elements, creating a less cohesive experience.

5.3 Windows 10

5.3.1 Privacy analysis

Table 5.4 shows how the number of entries increases greatly under Windows 10. Here, the extent of always-online functionality is made clear - several applications rely on regular internet access. Some addresses provide cause for concern, as the reason for visiting them is not clear. Bing and MSN are both pinged despite no internet browser having been opened nor searches made. Both OneNote and OneDrive

Table 5.4: Windows 10 DNS traffic

Target DNS	**Description**
time.windows.com	Time service
au.download.windowsupdate.com	Windows Update
licensing.mp.microsoft.com	Online activation, app licensing
xxx.digicert.com	SSl authority
xxx.msftncsi.com	Windows network awareness
go.microsoft.com	Generic Microsoft URL
www.bing.com	Bing search engine
weather.microsoft.com	Weather service
wns.windows.com	Push notification service
xxx.msn.com	MSN
cdn.onenote.net	OneNote
v20.events.data.microsoft.com	Diagnostic endpoint
edge-consumer-static.azureedge.com	Azure cloud edge
iris.microsoft.com	Spotlight content
arc.msn.com	Spotlight content
settings-win.data.microsoft.com	Update endpoint for apps
x.delivery.mp.microsoft.com	Updates and Store downloads
nav.smartscreen.microsoft.com	SmartScreen reporting/notifications
storecatalogrevocation.storequality.microsoft.com	Revokes malicious Store apps
maps.microsoft.com	Map downloads
oneclient.sfx.ms	OneDrive
virtualearth.net	Language and geographic services

are also present, even though neither are logged in. In fact, no Microsoft account has been added for the environment.

5.3.2 Malware testing

Windows 10 shows slight improvements over Windows 7 on the security front, as shown in table 5.5. It is capable of detecting multi-file phishing sites as scams, though the single HTML file 'fake login prompt' still evades detection. 'Trololo' produces a rather humorous result, first prompting the user with UAC, and then notifying that the app requires .NET 3.5 to run. Installing it allows the prank virus to execute. 'Deskbottom' was clearly blocked this time around, rather than spawning an empty window.

Table 5.5: Windows 10 malware test

Name	**Type**	**Infection**	Notes
Deskbottom	Enderware	x	Blocked
Useroverflow	Enderware	x	
Fake login prompt	Fake scanners	✓	
Fake Microsoft Support	Fake scanners	x	
Screenscrew	Jokes	✓	
Trololo	Jokes	✓	Requires .NET 3.5
BadRabbit	Ransomware	x	
Cerber 5	Ransomware	x	
Antivirus 2010	Rogue	x	
Windows Accelerator Pro	Rogue	x	
DesktopPuzzle	Trojan	x	
You Are An Idiot	Trojan	x	

Table 5.6: Windows 10 performance results

Blender test	**Seconds**
Test 1	895.71
Test 2	867.49
Test 3	870.19
Average	877.80
Deviation	1.77%
Latency	**Microseconds**
Highest measured interrupt to process latency	104970.50
Average measured interrupt to process latency	258.41
Highest measured interrupt to DPC latency	93608.60
Average measured interrupt to DPC latency	208.87

5.3.3 Performance benchmarks

With performance testing, the aim is to provide a rough baseline of system performance and latency, which can then be used to compare operating systems in these areas. While such metrics are unlikely to be of much interest to casual users, gamers and power users may very well be persuaded to change OS by the promise of additional performance. Measuring the responsiveness of a system can be an incredibly complex task - by measuring system latency under a stress load, values will be gained for the worst case scenario, as well as an average latency number. This

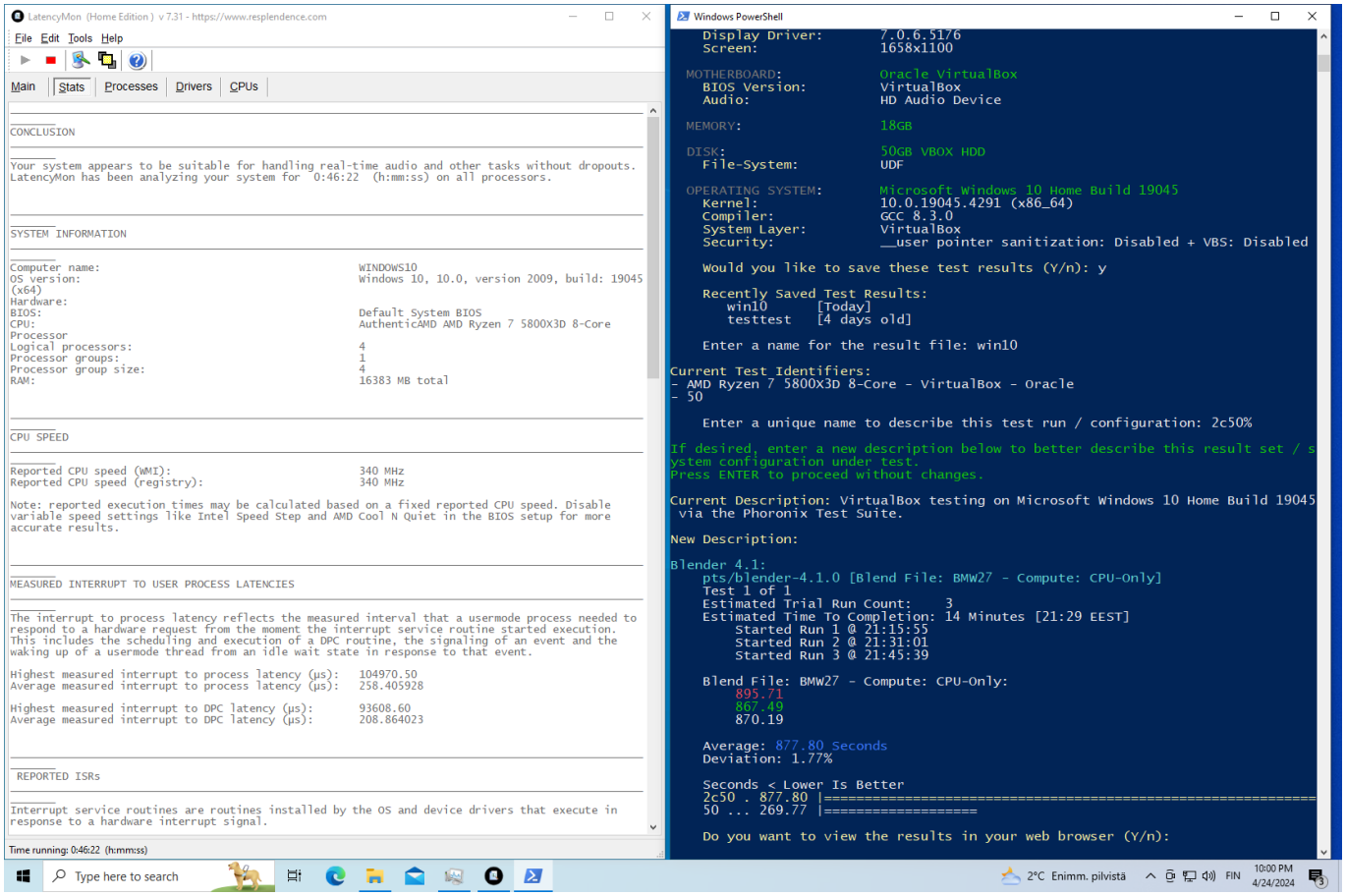

Figure 5.2: Performance testing under Windows 10

is by no means perfect as far as accurate measurements go, but will provide with a rough baseline for the sake of comparison. See figure 5.2 for an example of the testing process.

Building a stable testing environment proved itself a surprisingly challenging task, mainly due to the difficulty of constraining the virtualization software's resource usage. The Vm was initially granted access to half of the host system's cores under Virtualbox (8 out of 16), under the impression the remaining 8 were simply Virtualbox's way of representing hyperthreading - in other words, 8/16 cores assigned would permit the VM to utilize the equivalent of 4 cores and 8 threads' worth of processing power. This turned out to not be the case, as with this configuration the VM could utilize all of the host's processing power, leading to excessive

latency. An execution cap option is provided, but with it set to 50%. the VM still nearly overwhelmed the host, with CPU utilization in the 80-90% range. Thus, it became clear that further restricting the available processing power was necessary, even this meant tests would take much longer. The VM was given access to 4 cores at a 50% execution cap.

The test suite of choice was Phoronix Test Suite v10.8.4, due to its wide range of available tests, and compatibility between Windows and Linux. The test chosen was Blender-BMW27-CPU only. This test was automatically run three times and the result measured in seconds, lower being better. Latencymon was simultaneously started to measure system latency, and the result was taken once the stress test had concluded. The performance results for Windows 10 can be seen in table 5.6.

5.4 Windows 11

5.4.1 Privacy analysis

With Windows 11, total network traffic becomes rather encumbering to parse through. Over an hour of idling, nearly 200 000 total packets were sent, 0.2% of which were DNS traffic. A further increase in addresses tied to updates for the system and its applications is seen. What is not apparent from this list is how many times some of these domains and their subdomains were queried. MSN and Bing were both pinged a large number of times, before the system saw any real use. As an aside, a large number of queries were also sent to Mozilla, and Firefox is installed on the system, but had not been launched. It is unclear what the purpose of these queries was. See table 5.7.

Table 5.7: Windows 11 DNS traffic

Target DNS	**Description**
time.windows.com	Time service
au.download.windowsupdate.com	Windows Update
crl.verisign.com	Name server provider
xxx.digicert.com	SSl authority
ipv6.msftncsi.com	Windows network awareness
xxx.trafficmanager.net	Traffic load balancer
activation.sls.microsoft.com	Windows activation service
go.microsoft.com	Generic Microsoft URL
wdcp.microsoft.com	Protection service endpoint
www.bing.com	Bing search engine
weather.microsoft.com	Weather service
wns.windows.com	Push notification service
xxx.msn.com	MSN
cdn.onenote.net	OneNote
v20.events.data.microsoft.com	Diagnostic endpoint
edge-consumer-static.azureedge.com	Azure cloud edge
iris.microsoft.com	Spotlight content
arc.msn.com	Spotlight content
settings-win.data.microsoft.com	Update endpoint for apps
x.delivery.mp.microsoft.com	Updates and Store downloads
nav.smartscreen.microsoft.com	SmartScreen reporting/notifications
storecatalogrevocation.storequality.microsoft.com	Revokes malicious Store apps
maps.microsoft.com	Map downloads
oneclient.sfx.ms	OneDrive
virtualearth.net	Language and geographic services
www.msftconnecttest.com	Connection indicator
g.live.com	OneDrive
ecs.office.com	Microsoft Office
self.events.data.microsoft.com	Connected User Experiences
*.prod.do.dsp.mp.microsoft.com	Windows Update/app downloads
storeedgefd.dsx.mp.microsoft.com	Images for applications
displaycatalog.mp.microsoft.com	Store communication
prod-azurecdn-akamai-iris.azureedge.net	Cortana and Live tiles
tsfe.trafficshaping.dsp.mp.microsoft.com	Content regulation
*.delivery.mp.microsoft.com	Updates and Store
ocsp.ussertrust.com	RSA certificates
xxx.comodoca.com	SSL certificates
x1.c.lencr.org	Encryption service

Table 5.8: Windows 11 malware test

Name	**Type**	**Infection**	Notes
Deskbottom	Enderware	✓	Popup, Explorer lockup
Useroverflow	Enderware	∼	Runs, fails to add users
Fake login prompt	Fake scanners	✓	
Fake Microsoft Support	Fake scanners	x	
Screenscrew	Jokes	✓	Lessened effect
Trololo	Jokes	✓	Severe effect
BadRabbit	Ransomware	x	
Cerber 5	Ransomware	x	
Antivirus 2010	Rogue	x	
Windows Accelerator Pro	Rogue	x	
DesktopPuzzle	Trojan	✓	
You Are An Idiot	Trojan	∼	No visible effect

Table 5.9: Windows 11 performance results

Blender test	**Seconds**
Test 1	890.85
Test 2	888.12
Test 3	894.31
Average	891.09
Deviation	0.35%
Latency	**Microseconds**
Highest measured interrupt to process latency	108630.60
Average measured interrupt to process latency	120.39
Highest measured interrupt to DPC latency	87343.90
Average measured interrupt to DPC latency	90.19

5.4.2 Malware testing

Windows 11 displays some regressions over 10, as seen in table 5.8. Some malware that was previously completely blocked now runs though is seemingly impotent (Useroverflow) or runs successfully (DesktopPuzzle). Others display worse symptoms, such as a completely black desktop with 'Trololo'.

5.4.3 Performance benchmarks & user experience

Windows 11 appears to perform slightly worse overall in this test, taking 1.5% longer on average than Windows 10, though with considerably less variation. It should be noted the average latency here very quickly dropped from figures seen in the 400-600 range during the stress test. It is unclear why the drop was so large, but as the test was concluded the exact same way as under Windows 10, this can only be attributed to the tool's behavior. See table 5.9.

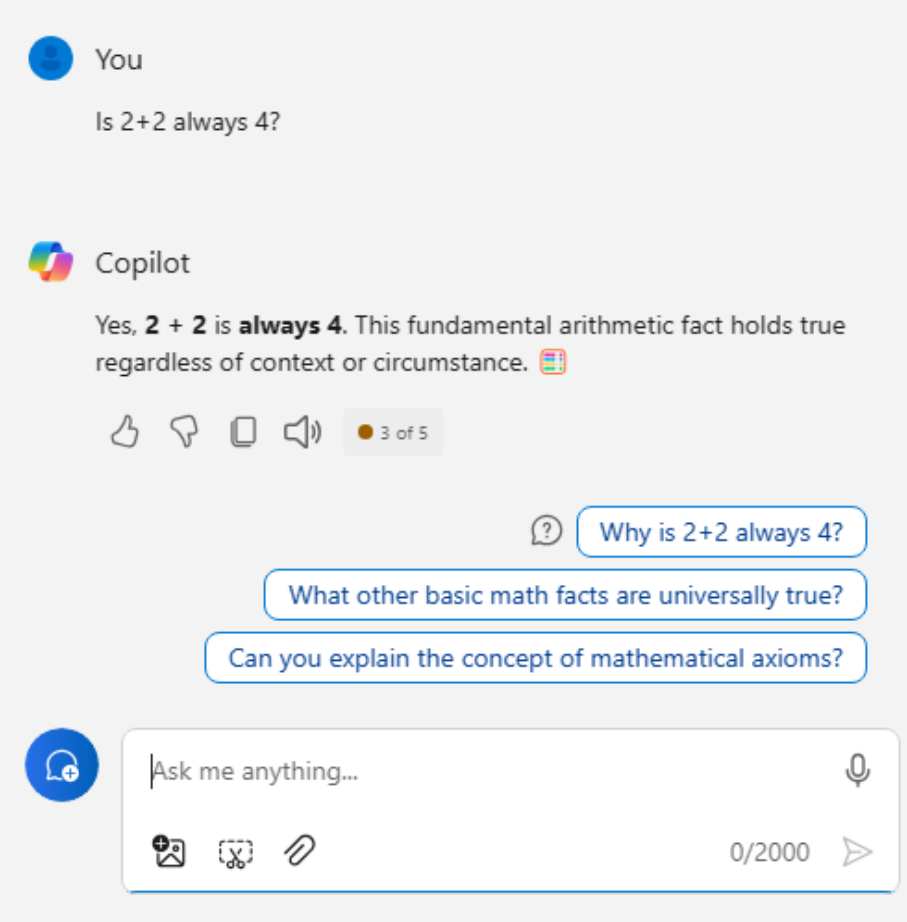

Figure 5.3: Copilot in use

5.4.4 User experience

Windows 11 may be summarized as being a slightly worse version of Windows 10. In the benchmarks, performance was slightly worse, as was malware detection. Data collection and an 'always online' ecosystem are a tough sell for most users - among gamers and power users, methods for disabling these are a common topic. Gamers may find use for some new or enhanced features such as Auto HDR for automatic HDR in whitelisted SDR games, DirectStorage for faster loading times, and Game Bar for recording functionality.

The near future of the OS also looks uncertain as far as upcoming and recent features are concerned. In a recent, currently optional update, Start menu ads for Store applications were enabled for users [34]. Copilot, the AI assistant that is enabled by default on updated systems and is showcased in figure 5.3, has been described as 'limited and frustrating' [35].

5.5 Windows 11 + The Ultimate Windows Utility

The next attempt is to improve the Windows 11 experience by utlizing a third-party toolbox. The Ultimate Windows Utility, so-titled by its author Chris Titus, is a multipurpose tool for 'debloating' and tweaking a Windows install. It is open source and hosted on Github, where it is contributed to by a team of volunteers.

The tool is installed via a script on an elevated PowerShell:

```
{shell}
                iwr -useb https://christitus.com/win | iex
```

The utility comes with extensive functionality, much of which falls outside the scope of this experiment, such as automatically installing applications, or the cre-

ation of a custom Windows ISO. Interest pertains to the tweaks section, as well as the included O&O ShutUp10++ utility.

The utility may be used to perform more invasive measures as well, such as the removal of core system components, which may result in unintended side effects. Such measured will be skipped and instead opt for recommended tweaks. In the Tweaks section, the recommended selection for Desktops is chosen, as shown in figure 5.4. Taskbar widgets will also be disabled.

In the O&O ShutUp10++ utility, a few presets are presented; 'recommended', 'recommended and somewhat recommended', and 'all settings'. Each individual privacy tweak is listed, as well as which of the three aforementioned categories it is included in. This list is the first time we've come across a cohesive and easy to follow listing of the various data collection measures included in Windows, and it is presented in a free, third-party tool, rather than by Microsoft themselves. Here, the 'recommended' option is chosen. This disables various suggestions, limits application access to user information, disables the synchronization of Windows settings, AI functionality, and lock screen features, among others.

Both of these tools merit deeper analysis regarding the impact of each of the settings presented, but for the purposes of this experiment, recommended settings are chosen in both cases. This emulates how a typical user is likely to use these tools, and the authors' judgment is trusted in separating safe settings from those best left to power users.

5.5.1 Privacy analysis

A marked reduction in pinged URLs may be seen, though a few new entries were also detected, as shown in table 5.10. The total number of both DNS packets, and packets overall, was a bit more than half compared to the base Windows 11 install.

Based on these results, the anti-telemetry measures taken seem to be working.

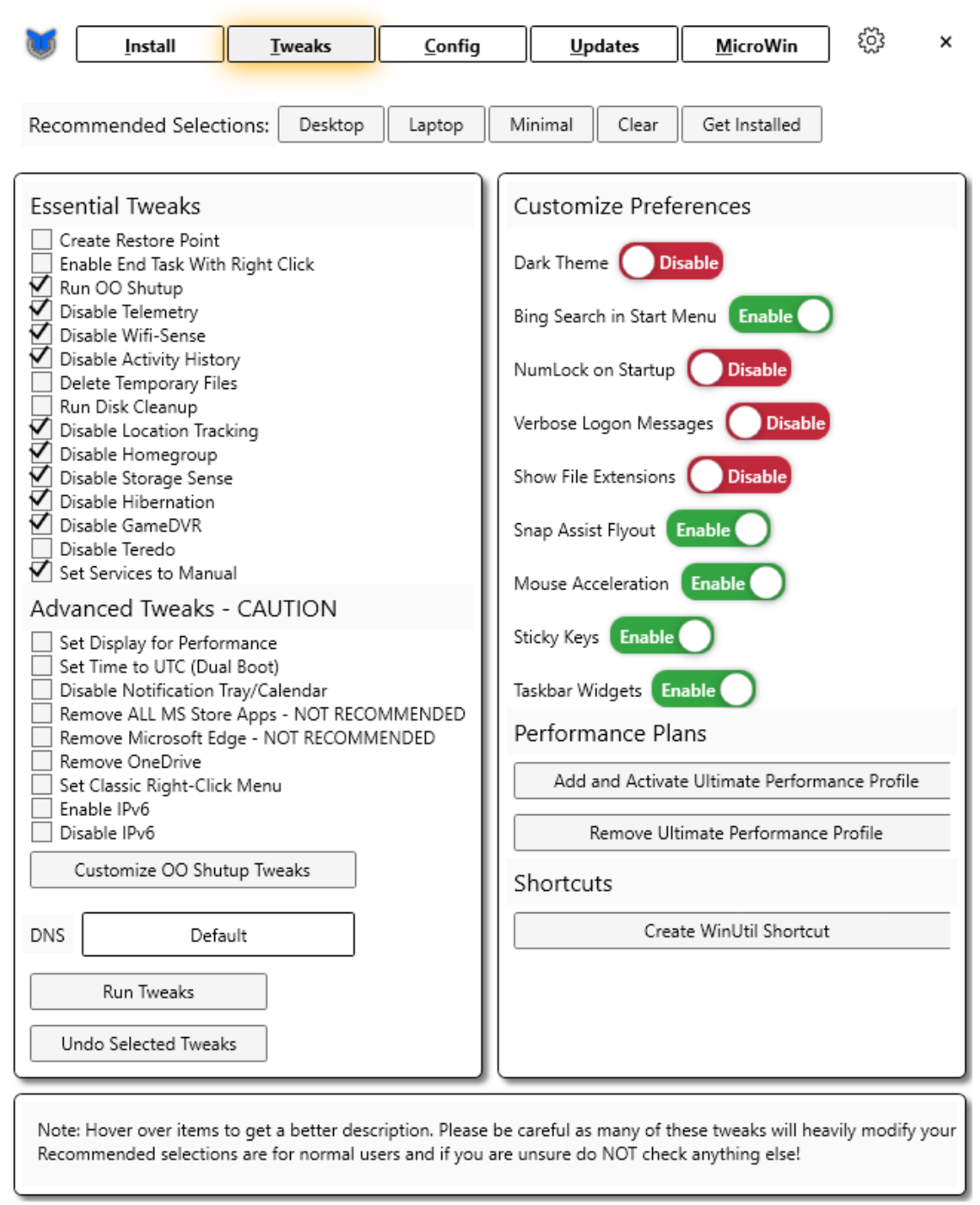

Figure 5.4: The Ultimate Windows Utility

Table 5.10: Windows 11 + Ultimate Windows utility DNS traffic

Target DNS	**Description**
time.windows.com	Time service
ipv6.msftncsi.com	Windows network awareness
xxx.trafficmanager.net	Traffic load balancer
activation.sls.microsoft.com	Windows activation service
wdcp.microsoft.com	Protection service endpoint
www.bing.com	Bing search engine
wns.windows.com	Push notification service
xxx.msn.com	MSN
cdn.onenote.net	OneNote
v20.events.data.microsoft.com	Diagnostic endpoint
edge-consumer-static.azureedge.com	Azure cloud edge
settings-win.data.microsoft.com	Update endpoint for apps
x.delivery.mp.microsoft.com	Updates and Store downloads
oneclient.sfx.ms	OneDrive
virtualearth.net	Language and geographic services
www.msftconnecttest.com	Connection indicator
g.live.com	OneDrive
ecs.office.com	Microsoft Office
self.events.data.microsoft.com	Connected User Experiences
*.prod.do.dsp.mp.microsoft.com	Windows Update/app downloads
storeedgefd.dsx.mp.microsoft.com	Images for applications
displaycatalog.mp.microsoft.com	Store communication
*.delivery.mp.microsoft.com	Updates and Store
tsfe.trafficshaping.dsp.mp.microsoft.com	Content regulation
xxx.spotify.com	Spotify
officeclient.microsoft.com	Microsoft Office
*.pipe.aria.microsoft.com	Skype configuration
login.live.com	Device authentication
x1.c.lencr.org	Encryption service

However, the system certainly cannot be claimed to be free of data collection yet. Further study may be conducted to determine how far it is necessary to go to shut down telemetry completely under Windows 11, if such a feat is even possible, and how the usability of the system in such a scenario would be affected.

5.5.2 Malware testing

Table 5.11: Windows 11 + Ultimate Windows utility malware test

Name	**Type**	**Infection**	Notes
Deskbottom	Enderware	✓	Popup, Explorer lockup
Useroverflow	Enderware	∼	Runs, fails to add users
Fake login prompt	Fake scanners	✓	
Fake Microsoft Support	Fake scanners	x	
Screenscrew	Jokes	✓	Lessened effect
Trololo	Jokes	✓	Severe effect
BadRabbit	Ransomware	x	
Cerber 5	Ransomware	x	
Antivirus 2010	Rogue	x	
Windows Accelerator Pro	Rogue	x	
DesktopPuzzle	Trojan	✓	
You Are An Idiot	Trojan	∼	No visible effect

The modified system's performance against malware was the exact same as the base install's. This was to be expected, as the utility makes no claims regarding improved cybersecurity. See table 5.11.

5.5.3 Performance benchmarks & user experience

A marked improvement of approximately 6.5% reduction in time taken is seen in the test suite, under the same Balanced power profile, as shown in table 5.12. The latency numbers have seemingly not improved - quite the opposite. It should be noted that during the actual test, seen averages were much more modest than under the base Windows 11 test, slowly rising to the range of 120-150 microseconds, compared to 400-600 microseconds. This presents a challenge in the testing methdology,

Table 5.12: Windows 11 + Ultimate Windows utility performance results

Blender test	**Seconds**
Test 1	846.39
Test 2	839.27
Test 3	812.25
Average	832.64
Deviation	2.16%
Latency	**Microseconds**
Highest measured interrupt to process latency	141722.80
Average measured interrupt to process latency	153.80
Highest measured interrupt to DPC latency	89093.90
Average measured interrupt to DPC latency	126.31

as visual inspection is used, while the actual behavior of these measurements during the test may vary more than the end results suggest.

5.5.4 User experience

The core user experience is changed little by the changes made by the utility toolkit. It largely changes things 'under the hood' - the end user is more likely to notice a slight decrease in resource usage and online interactivity in the UI. The utility's practical worth may best be measured as providing piece of mind to users, as an easy-to-use method for nudging the OS towards a direction that is more agreeable to most than what is provided by default.

5.6 Windows 11 Ghost Spectre

Ghost Spectre is a custom version of the Windows 11 OS, modified by a third-party developer of the same name. It is also available for Windows 10 and supports several installation options, depending on the extent of 'bloat' removal desired. The option chosen for this test is 'Superlite + Defender', seen in figure 5.5. This is also the option in use on the author's host system.

The author's claims regarding the features present in Ghost Spectre are verbatim as follows, cleaned up for grammar [36]:

- Improved DirectX12
- Compact integrated + LZX (algorithm)
- Bloatware free
- Optimized Pagefile/Services/Scheduled/Search Indexer
- Privacy optimizations & Performance mode
- Ghost Toolbox (Add or remove Microsoft Store, and more)
- Supports any language and keyboard setting
- Supports UWP Games and UWP Apps
- Update capable, can update to the latest Windows 11 build
- Window Update can be paused until 2077
- Custom icons and themes
- Custom WPE bootable
- Normal SV2 bootable
- Built-in TPM and option for removal of TPM

The followwing components and applications have been disabled or removed:

- Removes Windows and System Apps
- Removes Windows Security, Defender and Smartscreen
- Disables Remote Desktop, tablet keyboard, NFC, clipboard, focus Assist (Superlite only)

- Disables Print spooler
- Removes OneDrive
- Disables Action Center and notifications (Superlite only)
- Disables telemetry (Superlite only)
- Removes error reporting (Superlite only)
- Disables UAC (never notify)
- Removes WinSxS backup

  Ghost Spectre is potentially very extensive in the changes it makes, capable of removing core system components if the user does not need them. A toolbox is included to aid in reinstalling these if needed, as well as to serve as a generally helpful method for installing typical programs and their requirements.

5.6.1 Privacy analysis

Ghost Spectre sees a further reduction in DNS and overall network traffic, as shown in table 5.13. Over an hour of observation, approximately 100 DNS and 2200 total packets were sent. A sizeable portion of the DNS traffic was was once again generated by Firefox, despite the browser not being launched. Several notable omissions can be seen; MSN is completely absent, and Bing was only seen once, whereas these two previously dominated much of the traffic. OneNote and OneDrive are completely absent on the system and are thus also not present in network traffic. Oddly, some Store traffic can be seen, despite the fact that it is not installed on the system. Location services are disabled by default, so no related traffic is present.

Table 5.13: Windows 11 Ghost Spectre DNS traffic

Target DNS	**Description**
time.windows.com	Time service
ipv6.msftncsi.com	Windows network awareness
xxx.trafficmanager.net	Traffic load balancer
wdcp.microsoft.com	Protection service endpoint
www.bing.com	Bing search engine
wns.windows.com	Push notification service
v20.events.data.microsoft.com	Diagnostic endpoint
edge-consumer-static.azureedge.com	Azure cloud edge
settings-win.data.microsoft.com	Update endpoint for apps
www.msftconnecttest.com	Connection indicator
ecs.office.com	Microsoft Office
self.events.data.microsoft.com	Connected User Experiences
*.prod.do.dsp.mp.microsoft.com	Windows Update/app downloads
storeedgefd.dsx.mp.microsoft.com	Images for applications
displaycatalog.mp.microsoft.com	Store communication
*.delivery.mp.microsoft.com	Updates and Store
tsfe.trafficshaping.dsp.mp.microsoft.com	Content regulation
officeclient.microsoft.com	Microsoft Office
login.live.com	Device authentication
xx.x.lencr.org	Encryption service
storecatalogrevocation.storequality.microsoft.com	Revokes malicious Store apps
iris.microsoft.com	Spotlight content

Table 5.14: Windows 11 Ghost Spectre malware test

Name	**Type**	**Infection**	Notes
Deskbottom	Enderware	✓	Popup, Explorer lockup
Useroverflow	Enderware	✓	Success
Fake login prompt	Fake scanners	∼	"IE11 no longer supported"
Fake Microsoft Support	Fake scanners	x	
Screenscrew	Jokes	✓	Lessened effect
Trololo	Jokes	✓	Severe effect
BadRabbit	Ransomware	x	
Cerber 5	Ransomware	x	
Antivirus 2010	Rogue	x	
Windows Accelerator Pro	Rogue	x	
DesktopPuzzle	Trojan	✓	
You Are An Idiot	Trojan	∼	No visible effect

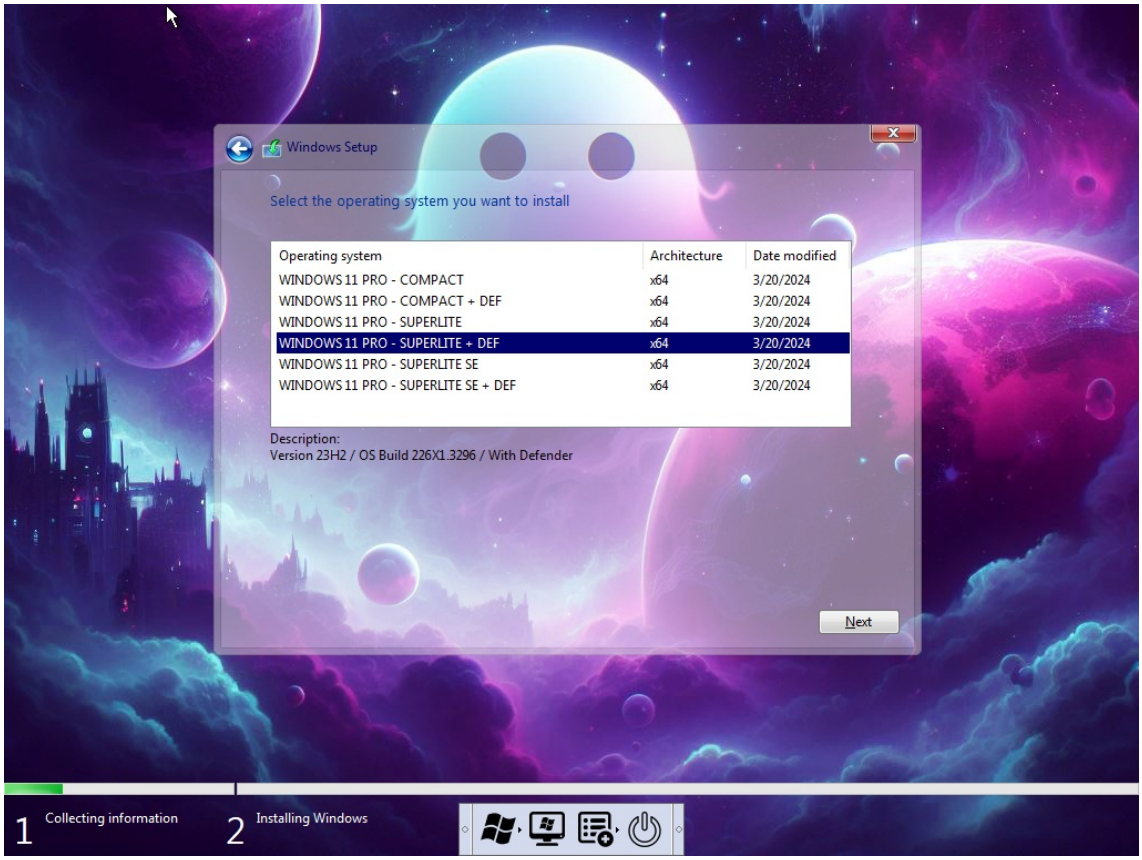

Figure 5.5: Windows 11 Ghost Spectre installation

5.6.2 Malware testing

Ghost Spectre displays some interesting results due to its default configuration. Firstly, the default account is the Administrator account - that is, not simply an account in the Administrators group. This seems to permit UserOverflow to execute correctly, capable of creating an arbitrary number of user accounts. 'Fake login prompt' fails to run initially, but this is merely due to the fact that Edge is not present on the system, which seems to cause HTML files to default to Internet Explorer 11, which is also not present. It opens normally in Firefox. Beyond these small differences, the results are unchanged, which is to be expected. See table 5.14.

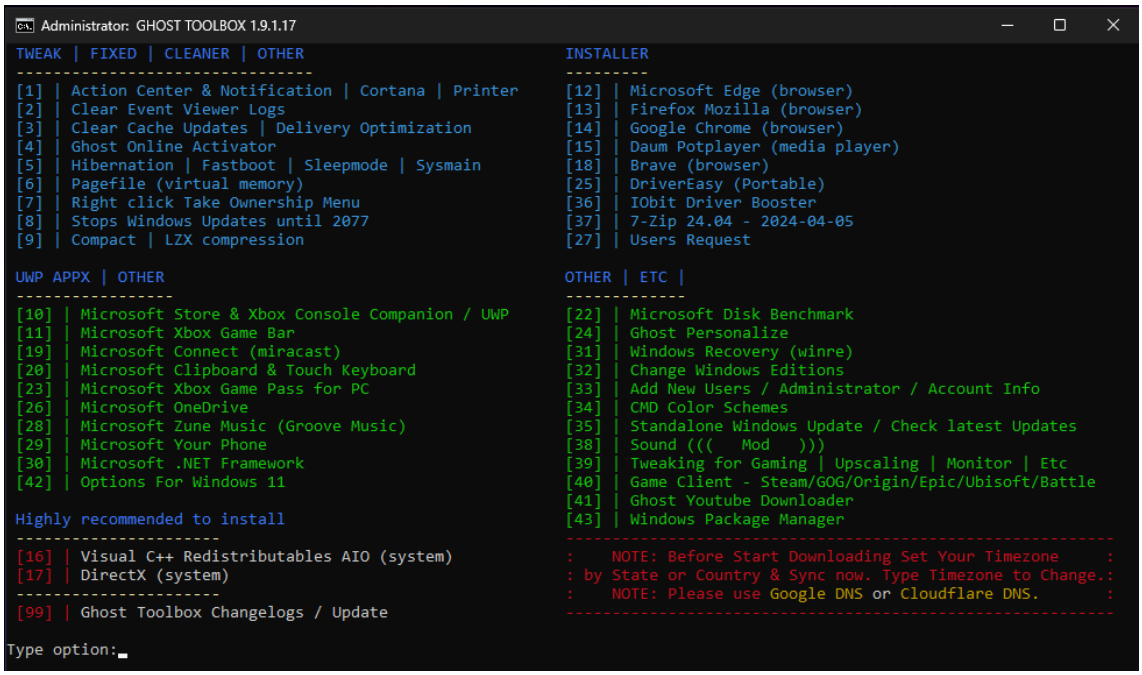

Figure 5.6: Windows 11 Ghost Spectre toolbox

Table 5.15: Windows 11 Ghost Spectre performance results

Blender test	**Seconds**
Test 1	835.21
Test 2	810.67
Test 3	813.63
Average	819.84
Deviation	1.63%
Latency	**Microseconds**
Highest measured interrupt to process latency	105255.40
Average measured interrupt to process latency	393.26
Highest measured interrupt to DPC latency	83918.90
Average measured interrupt to DPC latency	316.46

5.6.3 Performance benchmarks & user experience

The performance benchmark shows a further, modest improvement of 1.5% less time taken compared to Windows 11 + Ultimate Windows utility, and 8% compared to a base Windows 11 install. The latency numbers show mixed results. See table 5.15.

5.6.4 User experience

Ghost Spectre makes a host of quality-of-life improvements for the end user. A dark theme is incorporated, and taskbar and Start Menu behavior is restored. The included toolbox has lots of extra functionality, as shown in figure 5.6. Windows Update is indefinitely paused, and no online login is enforced. The use of the Administrator account by default is a questionable choice, as it does not seem to offer much meaningful benefit, but comes with some concerns of incompatibility and security. Some programs refuse to run on this account, forcing the user to create a second, lower privilege account anyway.

A major concern of course is privacy. Ghost Spectre is an unofficial, third party modification of Windows, made by a sole developer working under a pseudonym, and with no open source disclosure of the changes made. Thus, trust, if it can be considered as such, is largely switched from a global corporation to a single developer, whose intentions may not be benign. This renders the custom OS wholly unsuitable for any sort of enterprise environment, as well as casual users. Only power users who understand and are willing to take these risks should consider adopting it.

Table 5.16: Ubuntu 22.04 LTS DNS traffic

Target DNS	**Description**
connectivity-check.ubuntu.com	Tests connection
security.ubuntu.com	Security updates
archive.ubuntu.com	Archive for updates
esm.ubuntu.com	Sotware security maintenance
xxx.snapcraft.io	Ubuntu app store
ns1.canonical.com	Name server

5.7 Linux Ubuntu 22.04 LTS

5.7.1 Privacy analysis

The traffic list on a fresh Ubuntu system is remarkably brief and easy to follow. Nothing pointing towards data collection can be seen - the addresses all have to do with security, the application store, or connectivity testing. See table 5.16.

5.7.2 Malware testing

With Linux malware testing, a number of difficulties that made standardized testing impractical were encountered. Firstly, the initial selection of hand picked samples failed to be able to run reliably and produced no immediately obvious effects. These files had to be run from the terminal, before which they had to be set as executable. Even then, results were inconclusive, and obvious signs of working ransomware were not seen. An example may be seen in Figure 5.7.

Thus, a difference in approach became necessary. A different malware repository, Linux Malware Samples on Github, was chosen [37]. Unfortunately, these samples lack proper classification, making them a large collection of samples with no information regarding what they do. The decision to simply execute a large number and observe the results was made. Across approximately 20 samples, none produced immediately obvious clues as to their behavior. It was only after leaving the VM running for some time that the system complained about a full disk,

showing 0 bytes free of the 20GB allocated, and the system monitor showed several of the executables running in the background, with some consuming a considerable portion of CPU resources. See figures 5.7, 5.8. One text file was produced, detailing the functionality of a crypto miner. Thus, it can be concluded that at least some CPU mining is taking place on the VM. Nothing hinting towards ransomware can be detected. Botnets, which harness infected systems to participate in Distributed Denial of Service (DDoS) attacks are also one of the more common types of malware on Linux, but as the VM is isolated, the likelihood of such an infection cannot accurately be gauged.

A few things become clear. One, Linux systems can indeed be infected with malware. However, the behavior of said malware differs considerably from that seen on Windows, as the malware above all else aims to be insidious and not make its presence known. No obvious scam programs and the like were seen. The question is, what channels Linux malware uses for distribution, and who are the typical targets. Ransomware appears very common, but could not be activated in testing, if present. It seems reasonable to assume that ransomware under Linux primarily targets enterprise environments, and such ransomware would be sophisticated enough not to encrypt the system unless if was certain it had reached a valid target.

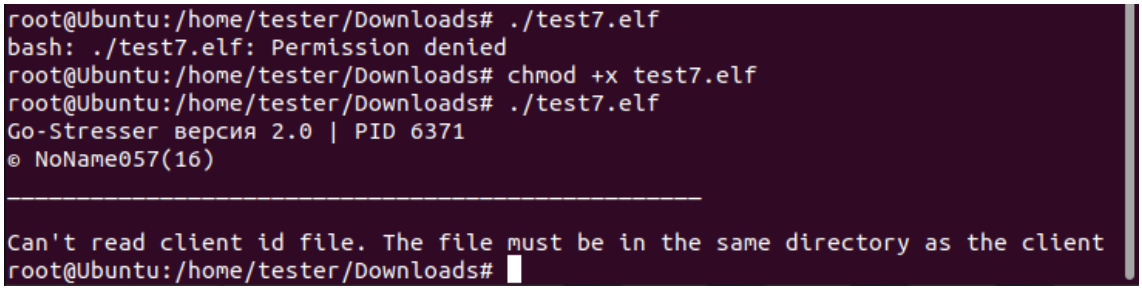

Figure 5.7: Attempt at Linux malware testing

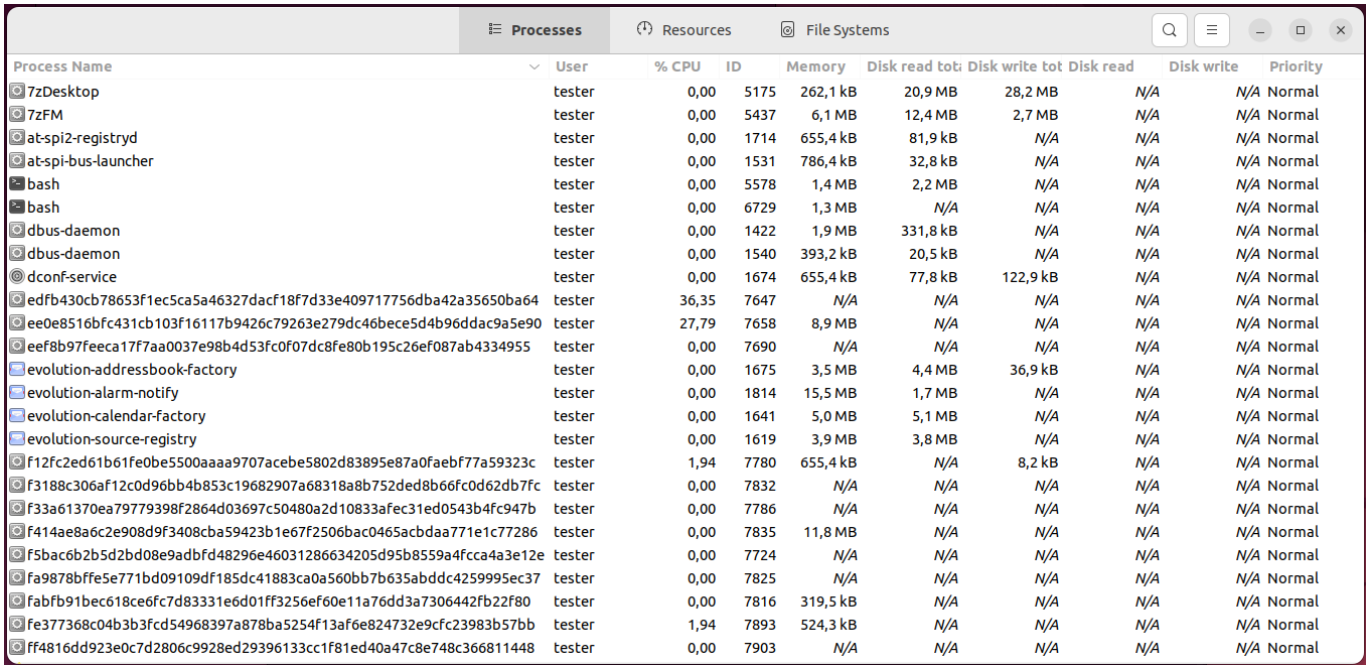

Figure 5.8: Malicious software as seen in the system monitor

Table 5.17: Ubuntu performance results

Blender test	**Seconds**
Test 1	654.87
Test 2	648.93
Test 3	655.75
Average	653.18
Deviation	0.57%

5.7.3 Performance benchmarks

As seen in table 5.17, Ubuntu shows a remarkable improvement of 26.7% less time taken than base Windows 11. Such an improvement warrants double-checking to ensure latency testing did not unduly interfere with the test results under Windows, as latency testing at the same time was not possible under Ubuntu. A rerun of the test under base Windows 11 but without latency monitoring resulted in an average of 885.08 seconds, which is within margin of error and gives no reason to assume latency monitoring during the stress test unduly affects the result.

Ubuntu latency testing proved a greater challenge than initially expected - Latencytop, the initial choice, turned out not to be compatible with this build of Ubuntu, and would have required rebuilding the kernel. Even then, the results would not have been directly comparable, as the programs do not measure the same types of latencies. It is not expected for these measurements to play a major part in convincing users to make the switch from Windows from Linux in any case.

5.7.4 User experience

Considerations regarding Linux's suitability for different user groups as an alternative to Windows require taking into account multiple aspects. For casual users, the general user experience and a familiar 'feel' of the system may be most important - multiple distributions do specifically aim for this approach, such as Ubuntu and Linux Mint. For many casual users, an upgrade path from Windows 10 to such an OS may be possible, though it is important to note that use of the terminal is nearly impossible to avoid.

Other merits of the Linux ecosystem may also amount to little if it is missing in crucial software that the user requires. For example, fully functional local Microsoft Office software may be a strict requirement that Linux cannot provide.

6 Conclusion

6.1 Summary of findings

In this thesis, five different operating systems across eight different configurations have been compared in the areas of privacy, security, performance and user experience. Based on these results, findings will now be summarized and evaluations given.

Windows XP is best treated as a dead operating system for all but specific uses cases. It is wholly unsuitable as a general purpose, modern OS due to a critical lack of security updates and up-to-date software. Installing antivirus software did greatly improve the level of protection against malware, but finding such software that still supported Windows XP was a challenge in itself, and even then, such measures are best treated as borrowed time. The operating system is certainly not without its merits - it had the best respect for the user's privacy of the OS's tested, and will certainly continue to see use in specialized systems due to its compatibility with software of its time.

Windows 7, on the other hand, is much more usable for most users, still having access much up-to-date or recently unsupported software. Its protection against Malware is improved thanks to built-in security software that still receives updates. A certain degree of dedication and technical knowledge is required to safely use the operating system, as it no longer receives security updates, and an increasing

number of programs will requires the use of alternatives or workarounds as official support declines.

Windows 10 is a major step forward on the security front, providing the most secure user experience yet and is entirely usable without third-party anti-virus software. However, it takes very concerning measures with its data collection, and relies on being always online. The increase in DNS requests on an idle system compared to previous versions was staggering.

Windows 11 relies on a small set of new features to distinguish itself positively from Windows 10. This is unlikely to sell it to hesitant customers when weighed against the negatives found in an even increased level of data collection, though at least in testing no suspicious third-party DNS activity was detected. The level of malware protection remains the same, and performance has decreased slightly. Requiring TPM 2.0 will likely remain a major roadblock for the mass adoption of this operating system, and it is unlikely things will change with its successor.

On a more positive note, Windows 11 can indeed be meaningfully improved by third-party modifications. The Ultimate Windows Utility achieved what it advertised, reducing data collection traffic, as well as providing a slight performance boost. Furthermore, it should be noted that it improves system security by disabling many services from running automatically, thus reducing the attack vector. Configuration with the tool was quick and simple, making it easy to recommend to a wide variety of users.

The results achieved by Windows 11 Ghost Spectre were even more impressive, netting further gains in traffic reduction and performance. These come with the major caveat of how trustworthy potential users consider the customized system, but for a subset of power users and gamers, these improvements may be sufficient to warrant an upgrade to Windows 11.

Linux testing remains too limited to be conclusive. Differing from Windows

in a multitude of ways, several factors must be taken into consideration to make meaningful recommendations to different users. However, a definite improvement could be seen in productivity performance.

6.2 Recommendations based on findings

With Microsoft Windows, two trends are obvious: the operating system has become both increasingly secure, but also less respectful of user privacy over time. Microsoft clearly takes system security seriously, providing an extensive suite of software-based security features, as well as currently focusing on security beyond typical measures, with strict requirements for hardware-driven solutions. Yet, the most security-minded people are also likely to be cognizant of their privacy, and Windows is currently in poor standing on that front. A hardened system, that is both secure and privacy respecting, is not easily available, and whether the degree to which third-party modifications aid in this is sufficient, or whether those modifications are permissible in a given environment, is a different matter.

It is now time for the series of recommendations for different user groups, beginning with casual users. They are less likely to be concerned with cutting-edge features, comprehensive security suites and productivity performance, and may instead favor an easy-to-use, familiar environment, with familiar programs. As such users are unlikely to adopt new hardware until their current hardware breaks or becomes old, many will likely cling on to Windows 10 well past it end of support, likely longer than the dedicated minority still using Windows 7 today. Linux provides many options for such users, but this is likely to require supervision by a more technologically inclined acquaintance, to prevent undue frustration. Still, it is likely a substantial portion of these users will over time upgrade to Windows 11 regardless, if only due to it being bundled with a new computer.

Gamers are primarily concerned with performance, cutting-edge features, and

compatibility. Windows 11 is arguably the strongest contender for these criteria, as it provides additional features over Windows 10. Linux has made massive strides on this front in recent years, becoming a relevant option for gamers looking for an alternative to the Windows ecosystem. Proton, a software compatibility layer, allows native Windows games, barring those relying on intrusive anti-cheat measures, to be played on Linux with a generally high degree of success. Dedicated 'distros', such as Nobara Linux, are specifically built with gaming in mind. Such options should above all else appeal to gamers who also fall under the category of power users. It is proposed that Linux will carve out a meaningful portion of the PC gaming OS market in the years to come.

Power users looking for maximum performance, control over the system and an extensive software suite are likely to already be familiar with Linux. Here, the question largely revolves around what software the user needs to have access to - for some users, a single piece of software may necessitate a dedicated Windows installation. Power users are above all else likely to know precisely what they need and want from an operating system, and are willing to learn new skills and tools to accommodate this.

Regarding the usefulness of third-party modifications, The Ultimate Windows Utility is considered to be an easy recommendation for a wide range of users, barring enterprise environments. Its ease of use and the tangible improvements it provides should make Windows a more comfortable choice of operating system for most. Ghost Spectre is best only considered by experienced users who understand the risks involved, and place value in the experience it provides. The author will continue to use it as his daily OS, while aiming to become better acquainted with Linux as an alternative and potential future replacement.

6.3 Limitations of the study

The testing conducted in this thesis has consisted of standardized benchmarks, aimed to be replicated across different systems to attain a general overview of each system's performance in these areas. The strength in this approach is that it allows us to produce comparable results from a moderately sized sample in a reasonable time frame. The downside is that each of these operating systems, and even their individual components, could be subjected to far more thorough analysis that the approach chosen simply cannot accommodate. As an example, each new security addition in a given version of Windows could be dissected and subjected to specialized testing to evaluate its robustness.

For privacy testing, the method chosen was measuring the number of DNS requests as well as categorizing connection endpoints for the purposes of analysis. While this gives a general understanding of the operating system's level of data collection, it is far from conclusive, as encrypted traffic is not analyzed, for example. By no means should 'less DNS traffic equals better privacy' be the key takeaway when analyzing something as complex as an operating system.

Greater difficulty was encountered in producing comparable results than initially expected. It quickly became apparent producing performance benchmarks for legacy systems was not viable, and it was concluded such results would have been little more than a curiosity regardless. In hindsight, it is not surprising that Linux would give additional difficulty in producing comparable results, given its fundamental differences with Windows. Latency testing was concluded infeasible, while malware testing required bending expectations. The initial choices of samples could not be reliably executed, and while otherwise well documented, lacked information regarding the expected results and expected environment. The second batch of samples did execute, but lacked documentation and their exact behavior was difficult to ascertain. It became clear that if malware managed to reach a Linux system, and

was executed, it could indeed infect the system - though this is hardly conclusive evidence that Linux is as susceptible as Windows is on this front. For example, the means of malware delivery may considerably limit successful malware deployment under Linux. Or, it could be that only a subset of Linux systems are typically targeted. Linux is present on a wide variety of different hardware configurations beyond home computers.

The use of a virtual environment with Virtualbox presents additional challenges that may interfere with producing accurate test results. It is implicitly expected for the virtual environment to behave the same as a hardware environment, and for performance or malware behavior to not be unduly affected. In reality, sophisticated malware may well refuse to run in a virtual environment, or the operating system may behave differently in subtle ways. It was previously noted that it was difficult to restrain the VM's resource usage to a level where it would not overwhelm the host system during the stress test. A 50% execution cap in the Virtualbox settings was chosen. Later testing suggested that this may have resulted in a large increase in the VM OS's latency, and that an unrestricted execution cap may have been the more optimal choice.

6.4 Suggestions of future research

Several aspects of the systems tested in this thesis are suitable candidates for future, more in-depth study:

- Individual Windows security components could be stress tested to determine their exact contributions as well as limits. As an example, one may develop software that attempts to bypass Microsoft Defender SmartScreen.
- Windows hardware-based security could be the target of further analysis, as it was not covered in the testing done in this thesis.

- Linux warrants more extensive testing on multiple fronts to accurately gauge its capability as an alternative to Windows. For example, different distros could be compared to evaluate their usefulness for different user groups.

- Linux security as a whole is a far more complex subject than presented here. Different scenarios, such as Linux enterprise security could serve as a case of study.

- Beyond the field of security, Linux may be evaluated as a gaming or productivity platform. Examples include performance benchmarking and compatibility analysis for different games, or performance and feature analysis for productivity.

References

[1] Statcounter. "Desktop Operating System Market Share Worldwide Feb 2023 - Feb 2024". (2024), [Online]. Available: https://gs.statcounter.com/os-market-share/desktop/worldwide (visited on 03/11/2024).

[2] S. Endicott. "Windows 10 still has more than double the market share of Windows 11, and that doesn't look like it will change any time soon". (2024), [Online]. Available: https://www.windowscentral.com/software-apps/windows-11/windows-10-still-has-more-than-double-the-market-share-of-windows-11-and-that-doesnt-look-like-it-will-change-any-time-soon (visited on 03/15/2024).

[3] Microsoft contributors. "Windows Processor Requirements". (2023), [Online]. Available: https://learn.microsoft.com/en-us/windows-hardware/design/minimum/windows-processor-requirements (visited on 05/16/2024).

[4] Microsoft contributors. "Trusted Platform Module Technology Overview". (2024), [Online]. Available: https://learn.microsoft.com/en-us/windows/security/hardware-security/tpm/trusted-platform-module-overview (visited on 03/17/2024).

[5] P. Matarazzo and V. Pamnani. "Microsoft Pluton security processor". (2024), [Online]. Available: https://learn.microsoft.com/en-us/windows/security/hardware-security/pluton/microsoft-pluton-security-processor (visited on 03/16/2024).

[6] Microsoft contributors. "What is Zero Trust?" (2024), [Online]. Available: https://learn.microsoft.com/en-us/security/zero-trust/zero-trust-overview (visited on 03/16/2024).

[7] Microsoft contributors. "Windows operating system security". (2023), [Online]. Available: https://learn.microsoft.com/en-us/windows/security/operating-system-security/ (visited on 03/17/2024).

[8] Microsoft contributors. "Microsoft Defender Antivirus in Windows Overview". (2024), [Online]. Available: https://learn.microsoft.com/en-us/microsoft-365/security/defender-endpoint/microsoft-defender-antivirus-windows?view=o365-worldwide (visited on 03/18/2024).

[9] N. Naik, P. Jenkins, R. Cooke, D. Ball, A. Foster, and Y. Jin, "Augmented windows fuzzy firewall for preventing denial of service attack", in *2017 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, 2017, pp. 1–6. DOI: 10.1109/FUZZ-IEEE.2017.8015701.

[10] K. Ejin and C. Hyoung-Kee, "Security Analysis and Bypass User Authentication Bound to Device of Windows Hello in the Wild", *Security and Communication Networks*, vol. 2021, 2021. DOI: https://doi.org/10.1155/2021/6245306.

[11] Gabriel Sieben. "The dangers of Microsoft Pluton (updated)". (2022), [Online]. Available: https://gabrielsieben.tech/2022/07/25/the-power-of-microsoft-pluton-2/ (visited on 03/21/2022).

[12] G. L. Galen Hunt and E. B. Nightingale. "The Seven Properties of Highly Secure Devices". (2017), [Online]. Available: https://www.microsoft.com/en-us/research/wp-content/uploads/2017/03/SevenPropertiesofHighlySecureDevices.pdf (visited on 03/27/2024).

[13] J. Frew. “Privacy and Windows 10: Your Guide to Windows Telemetry”. (2017), [Online]. Available: https://www.makeuseof.com/tag/privacy-windows-10-guide/ (visited on 03/30/2024).

[14] The PC Security Channel. “Has Windows become Spyware?” (2023), [Online]. Available: https://www.youtube.com/watch?v=IT4vDfA_4NI (visited on 04/02/2024).

[15] J. Norem. “Windows 11 Collects an Awful Lot of Telemetry About Your PC”. (2023), [Online]. Available: https://www.extremetech.com/computing/342941-windows-11-collects-an-awful-lot-of-telemetry-about-your-pc (visited on 04/02/2024).

[16] CISA. “OpenSSL ’Heartbleed’ vulnerability (CVE-2014-0160)”. (2016), [Online]. Available: https://www.cisa.gov/news-events/alerts/2014/04/08/openssl-heartbleed-vulnerability-cve-2014-0160 (visited on 03/13/2024).

[17] R. Williams. “Windows OS security brief history”. (2019), [Online]. Available: https://resources.infosecinstitute.com/topics/operating-system-security/windows-os-security-brief-history/ (visited on 03/14/2024).

[18] R. Awati. “Encrypting File System (EFS)”. (2021), [Online]. Available: https://www.techtarget.com/searchsecurity/definition/Encrypting-File-System (visited on 03/14/2024).

[19] R. Mazerik. “Windows 7 Security Features”. (2014), [Online]. Available: https://resources.infosecinstitute.com/topics/operating-system-security/windows-7-security-features/ (visited on 03/15/2024).

[20] P. Matarazzo. “Credential Guard overview”. (2023), [Online]. Available: https://learn.microsoft.com/en-us/windows/security/identity-protection/credential-guard/ (visited on 03/15/2024).

[21] P. Maniriho, A. N. Mahmood, and M. J. M. Chowdhury, "A systematic literature review on Windows malware detection: Techniques, research issues, and future directions", *Journal of Systems and Software*, vol. 209, p. 111 921, 2024. DOI: https://doi.org/10.1016/j.jss.2023.111921.

[22] R. Moussaileb, N. Cuppens, J.-L. Lanet, and H. L. Bouder, "A Survey on Windows-based Ransomware Taxonomy and Detection Mechanisms", *ACM Computing Surveys*, vol. 54, no. 6, pp. 1–36, 2021. DOI: 10.1145/3453153.

[23] N. Carlini and D. Wagner, "Towards Evaluating the Robustness of Neural Networks", in *2017 IEEE Symposium on Security and Privacy (SP)*, 2017, pp. 39–57. DOI: 10.1109/SP.2017.49.

[24] N. Papernot, P. McDaniel, X. Wu, S. Jha, and A. Swami, "Distillation as a Defense to Adversarial Perturbations Against Deep Neural Networks", in *2016 IEEE Symposium on Security and Privacy (SP)*, 2016, pp. 582–597. DOI: 10.1109/SP.2016.41.

[25] I. Kara, "Fileless malware threats: Recent advances, analysis approach through memory forensics and research challenges", *Expert Systems with Applications*, vol. 214, pp. 119–133, 2023. DOI: https://doi.org/10.1016/j.eswa.2022.119133.

[26] A. S. Bozkir, E. Tahillioglu, M. Aydos, and I. Kara, "Catch them alive: A malware detection approach through memory forensics, manifold learning and computer vision", *Computers & Security*, vol. 103, pp. 102–166, 2021. DOI: https://doi.org/10.1016/j.cose.2020.102166.

[27] N. A. Azeez, O. E. Odufuwa, S. Misra, J. Oluranti, and R. Damaševičius, "Windows PE Malware Detection Using Ensemble Learning", *Informatics*, vol. 8, no. 1, 2021. DOI: 10.3390/informatics8010010.

[28] B. Cakir and E. Dogdu, "Malware classification using deep learning methods", ser. ACMSE '18, Association for Computing Machinery, 2018. DOI: 10.1145/3190645.3190692.

[29] T. Lanfear, J. M. A. Lobo, and A. Buckgit. "Securing the Future of Artificial Intelligence and Machine Learning at Microsoft". (2024), [Online]. Available: https://learn.microsoft.com/en-us/security/engineering/securing-artificial-intelligence-machine-learning (visited on 04/07/2024).

[30] A. Demasi. "Legacy Update". (2024), [Online]. Available: https://legacyupdate.net/ (visited on 04/12/2024).

[31] C. Titus, *The Ultimate Windows Utility*, 2024. [Online]. Available: https://christitus.com/windows-tool/ (visited on 04/16/2024).

[32] WICAR. "Test malware!" (2024), [Online]. Available: https://www.wicar.org/test-malware.html (visited on 04/16/2024).

[33] MalwareBazaar, *Linux malware samples*, 2024. [Online]. Available: https://bazaar.abuse.ch/browse/tag/linux/ (visited on 04/19/2024).

[34] T. Warren. "Windows 11 Start menu ads are now rolling out to everyone". (2024), [Online]. Available: https://www.theverge.com/2024/4/24/24138949/microsoft-windows-11-start-menu-ads-recommendations-setting-disable (visited on 04/27/2024).

[35] J. Carrasqueira. "Copilot isn't great, but other AI features in Windows 11 are really cool". (2023), [Online]. Available: https://www.xda-developers.com/copilot-bad-windows-11-ai-really-cool/ (visited on 04/27/2024).

[36] Ghost Spectre. "W11 HOME 23H2". (2024), [Online]. Available: https://ghostclouds.xyz/wp/w11-home-23h2-22631/ (visited on 05/04/2024).

[37] Virus Samples Team, *Linux Malware Samples*, 2021. [Online]. Available: `https://github.com/MalwareSamples/Linux-Malware-Samples` (visited on 05/09/2024).