RESEARCH PROJECT

PRIVACY & ANONYMITY

TRANSFORMATION OF AN ASUS X55C

HARDWARE ANALYSIS | OPERATING SYSTEMS | NETWORK PRIVACY | BEHVORK DISCIPLINE | TOR EXPLORATION

A PRACTICAL RESEARCH ON PRIVACY, ANONYMITY AND THE LIMITS OF FUNCTIONALITY

DEVICE	DATE	STATUS
ASUS X55C-S0202H	27 AUGUST 2026	COMPLETE

PRIVACY & ANONYMITY

TRANSFORMATION OF AN ASUS X55C

ABSTRACT

This research investigates the progressive transformation of a conventional laptop into a system designed to minimize digital exposure, persistence, and attribution. The experimental subject is an ASUS X55C, initially documented in its original state and subsequently evaluated through a sequence of increasingly privacy-oriented configurations.

The study examines privacy and anonymity as a layered property rather than as a single technical feature. The investigation progresses from baseline identification and reconnaissance through hardware analysis, operating-system selection, network configuration, behavioral considerations, and the use of privacy-oriented technologies such as Tor. Each stage evaluates not only the reduction of identifiable information, but also its effect on system functionality, persistence, accessibility, and practical usability.

The project distinguishes between measures that were experimentally applied to the system and measures that remain theoretical due to hardware limitations, operational risk, or impracticality. This distinction allows the final assessment to address a central question: how far can the privacy and anonymity of a conventional computer be increased before the measures required begin to fundamentally compromise its usefulness as a general-purpose system?

The final analysis therefore treats privacy as a trade-off rather than an absolute state. Increasing resistance to identification and tracking generally introduces additional constraints, operational complexity, and loss of functionality. The resulting system is evaluated not according to whether it can achieve absolute anonymity, but according to the practical boundary between increased privacy and continued usability.

KEYWORDS

Privacy · Anonymity · Digital Footprint · Operational Security · Linux · Tor · Tails · Network Privacy · Persistence · Threat Modeling

RESEARCH QUESTION

How far can the privacy and anonymity of a conventional laptop be increased before the resulting restrictions make the system impractical for normal use?

RESEARCH OBJECTIVE

The primary objective of this research is to identify and analyze the practical boundary between **privacy, anonymity, and usability** by progressively reducing the mechanisms through which a conventional computer can expose, retain, or associate information with its user.

Contents

RESEARCH OVERVIEW

Project Scope

This research examines the progressive transformation of a conventional laptop into a system designed to minimize digital exposure and increase resistance to identification, tracking, and attribution.

The ASUS X55C serves as the experimental subject throughout the investigation. Its original configuration is first documented to establish a reliable baseline. From that point, the research progressively examines different layers of the system, ranging from physical hardware and operating-system characteristics to network communication, user behavior, and privacy-oriented technologies.

The investigation is structured as a sequence of phases. Each phase addresses a specific layer of the system and evaluates both the privacy benefits introduced and the practical consequences resulting from those changes.

Research Logic

The progression follows a simple principle:

Increase Privacy → Reduce Exposure → Introduce Restrictions → Evaluate Usability

The purpose is not to achieve an absolute state of anonymity, which cannot be guaranteed by technical configuration alone. Instead, the investigation seeks to identify the point at which additional privacy measures produce diminishing practical value while increasingly restricting the functionality of the system.

The final result is therefore evaluated as a **privacy–usability trade-off**, rather than as a binary distinction between anonymous and non-anonymous systems.

Research Progression

00 — BASELINE

Establishing the original physical, hardware, software, and network state of the system.

↓

01 — RECONNAISSANCE

Identifying information exposed by the machine, including system characteristics, interfaces, identifiers, services, and persistent elements.

↓

02 — HARDWARE ANALYSIS

Examining hardware components capable of contributing to identification, surveillance, or information exposure, together with theoretical privacy-oriented modifications.

↓

03 — OPERATING SYSTEM PROGRESSION

Evaluating the transition between operating environments with progressively stronger privacy and anonymity characteristics.

↓

04 — NETWORK

Analyzing network interfaces, connectivity, addressing, routing, and the mechanisms through which network activity can expose information about the system.

↓

05 — BEHAVIORAL FOOTPRINT

Examining the role of user behavior, identity separation, metadata, account usage, and operational discipline in maintaining privacy.

↓

06 — TOR EXPLORATION

Evaluating Tor as an additional privacy layer, with emphasis on its characteristics, limitations, latency, accessibility, and practical usability.

↓

07 — PRACTICAL STATE

Determining the resulting state of the system after the experimentally applicable privacy measures and evaluating the remaining balance between privacy and usability.

↓

08 — THEORETICAL EXTREME

Exploring the theoretical limit of the system under a privacy-first design, including measures that are impractical, unsafe, or incompatible with normal computer use.

METHODOLOGY

Experimental Design

The investigation was conducted as a sequential technical assessment of a single physical system. The ASUS X55C was first documented in its existing condition, establishing the reference state against which subsequent observations could be compared.

Changes were then introduced progressively rather than simultaneously. This approach allows the effect of each layer to be considered independently and reduces the risk of attributing an observed change to a modification that was not responsible for it.

The investigation combines direct observation, system enumeration, configuration analysis, practical testing, and theoretical assessment. Depending on the phase, evidence was collected from the physical device, operating system, network environment, applications, and observed system behavior.

Baseline and Comparison

The initial configuration functions as the control reference for the investigation. Hardware characteristics, software configuration, network interfaces, persistent identifiers, and other relevant properties were documented before privacy-oriented modifications were considered.

Subsequent configurations are evaluated relative to this initial state. This comparison makes it possible to determine not only what was changed, but also what information or functionality was affected by the change.

The baseline is therefore treated as a fixed reference point throughout the research rather than as an isolated preliminary stage.

Evidence Collection

Evidence was collected through several complementary sources. System information and configuration data provide direct technical evidence, while photographs and screenshots document the physical and graphical state of the system.

Command outputs and configuration files are retained as supporting evidence where they are relevant to a conclusion. The main body of the research focuses on interpreted results rather than reproducing complete raw outputs. Detailed technical evidence is reserved for the appendices where appropriate.

This separation is intended to maintain the readability of the main research while preserving sufficient material for verification.

Evaluation Criteria

Each modification or configuration is considered according to its observable effect on the system. The principal criteria are:

Exposure — whether identifiable or potentially sensitive information remains accessible through the system.

Persistence — whether information, identifiers, configuration data, or traces remain between sessions.

Attribution — whether activity can be associated with the device, its network connection, or an identifiable user.

Functionality — whether the system continues to provide its expected technical capabilities.

Operational Complexity — whether maintaining the configuration introduces additional procedures, dependencies, or opportunities for user error.

These criteria are considered together rather than independently. A configuration that reduces one form of exposure may simultaneously introduce limitations elsewhere in the system.

Controlled Progression

The order of the investigation follows a layered approach. Physical characteristics are considered before software and network configurations, while behavioral factors are evaluated after the technical environment has been established.

This ordering reflects the dependency between the layers. Software controls cannot completely compensate for information exposed by physical hardware, while network privacy mechanisms cannot compensate for operational behavior that directly reveals an identity.

The resulting assessment therefore considers both individual controls and the interaction between multiple controls.

Practical Testing and Observation

Where technically and safely possible, configurations were tested directly on the system. Their behavior, limitations, and effect on normal operation were recorded during the investigation.

Not every theoretically possible modification was implemented. Measures involving substantial physical alteration, elevated risk of hardware damage, irreversible changes, or configurations that would prevent meaningful continued testing were treated separately.

This ensures that the research does not present theoretical possibilities as experimentally verified results.

Theoretical Assessment

The final theoretical analysis extends the investigation beyond the configurations that were practical to implement.

Theoretical measures are evaluated according to their expected effect on the system and their relationship to the objectives of the research. They are not assigned the same evidential status as directly tested configurations.

This distinction is maintained throughout the document:

Observed — directly identified or measured during the investigation.

Tested — deliberately applied and evaluated on the system.

Theoretical — technically plausible but not experimentally implemented.

This classification establishes a clear boundary between empirical evidence and analytical projection.

Limitations

The results of this investigation should be interpreted within the limitations of the experimental environment. The study concerns a single physical device and does not represent every hardware configuration, operating system, network environment, or threat model.

Furthermore, privacy and anonymity are properties that extend beyond the endpoint computer. External networks, remote services, application behavior, metadata, infrastructure, and human decisions can all influence the final level of exposure.

Consequently, the purpose of the methodology is not to produce a universal measurement of anonymity, but to establish a reproducible framework for examining how progressively stronger privacy measures alter a conventional computer and its practical operation.

TEST SUBJECT

ASUS X55C

The experimental subject of this research is an **ASUS X55C**, a conventional consumer laptop originally designed for general-purpose computing. The system was selected as the physical platform for the investigation because its relatively conventional architecture provides a suitable basis for examining how privacy-oriented modifications affect an ordinary computer.

Before any experimental changes were introduced, the machine was documented in its original condition. This initial documentation established the physical, hardware, software, and storage characteristics used as the reference point throughout the investigation.

Baseline Hardware

Component	Baseline Specification
Model	ASUS X55C
Processor	Intel Celeron 1000M
Memory	4 GB DDR3 RAM
Storage	Western Digital WD3200BPVT — 320 GB HDD
Graphics	Integrated Intel graphics
Display	15.6-inch display
Network	Ethernet and 2.4 GHz Wi-Fi
Optical Drive	DVD optical drive
Battery	Functional only with external power
Operating Environment	Documented during Phase 00

Physical Condition

The device was examined before the beginning of the experimental process. Its physical condition, removable components, storage medium, connectivity interfaces, and visible hardware characteristics were recorded as part of the baseline.

Several physical limitations were identified during the initial inspection. These characteristics are relevant to the research because they affect both the practical usability of the machine and the feasibility of certain theoretical privacy measures.

The physical state of the device was therefore treated as part of the experimental environment rather than as an incidental characteristic.

Role in the Investigation

The ASUS X55C remains the reference system throughout the research. Rather than evaluating privacy technologies independently on different machines, the investigation follows the transformation of the same physical platform across progressively different configurations.

This provides a consistent hardware foundation for comparing the effects of changes to the operating system, network environment, software configuration, and operational practices.

The machine can therefore be considered the **constant experimental subject**, while the configurations applied to it represent the changing variables of the investigation.

Experimental Subject: ASUS X55C

Initial State: Conventional consumer configuration

Reference: Phase 00 — Baseline

Purpose: Evaluation of privacy, anonymity, and usability trade-offs

Baseline Reference

The complete technical state of the machine at the beginning of the investigation is documented in **Phase 00 — Baseline**. The information presented here provides only the essential profile required to identify the experimental subject.

Detailed hardware observations, system information, storage characteristics, network interfaces, and supporting evidence are retained in the corresponding phase and technical appendices.

PHASE 00

BASELINE CHARACTERIZATION

Establishing the original state of the experimental system

00.1 — Objective

The first stage of this research establishes the baseline condition of the experimental machine before any privacy, anonymity, or security-oriented modifications are introduced.

The objective is to document the physical, hardware, firmware, software, network, and functional state of the ASUS X55C as accurately as possible. This information provides the reference point against which subsequent modifications and observations can be evaluated.

The baseline was established through four complementary methods: physical inspection, automated system enumeration, BIOS/UEFI inspection, and functional verification.

A distinction was maintained between information directly observed during the investigation, information reported by the operating system, and properties that remained unverified.

00.2 — Initial Physical Inspection

Before powering on the machine, a physical inspection of the ASUS X55C was performed.

The chassis showed significant signs of wear, including scratches, accumulated dirt, and structural damage around the left hinge. The integrated keyboard was also heavily degraded, with several missing keys, non-functional keys, and keys producing incorrect characters.

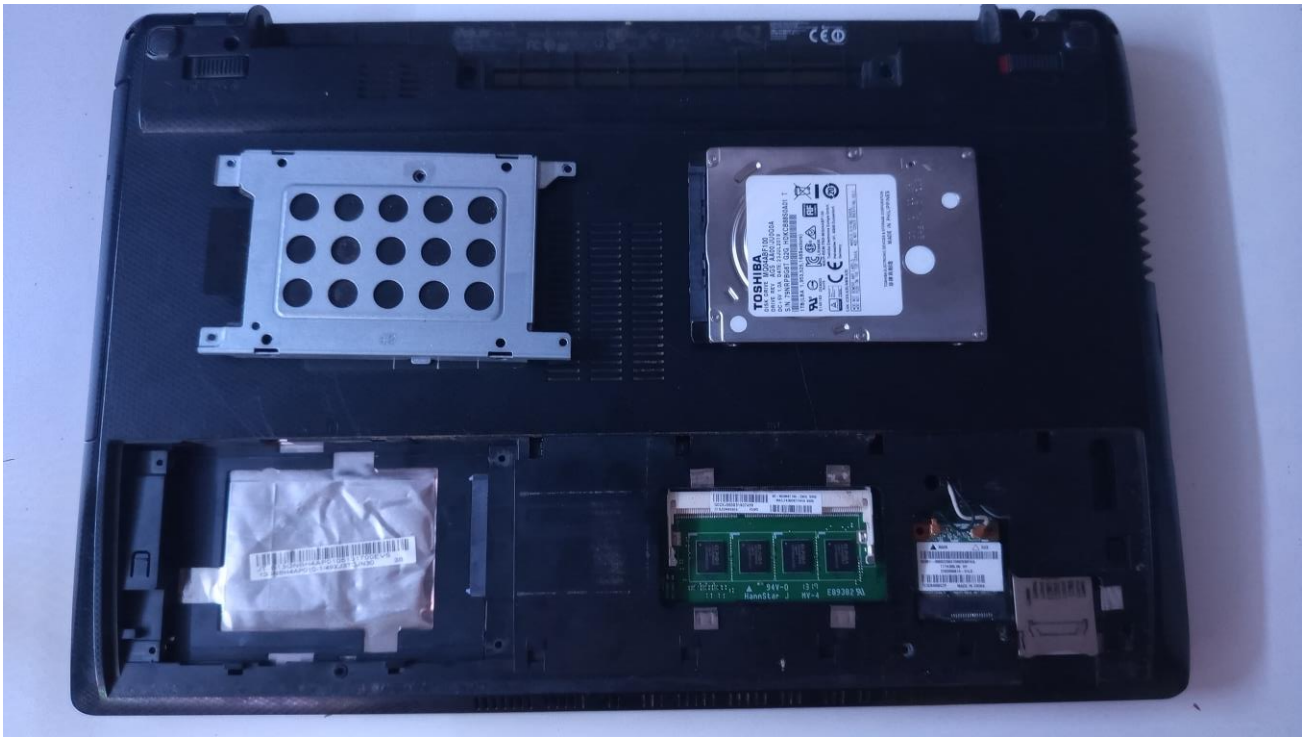
The display and touchpad appeared physically intact. The battery was present, while the removable maintenance areas provided access to the storage and memory components.

Photographs were taken of the accessible internal components, particularly the HDD and RAM areas, to preserve a visual record of the machine before any intervention.

Complete disassembly was intentionally avoided. Removing the keyboard presented a significant risk of damaging already degraded hardware, so the internal inspection was restricted to components that could be accessed safely.

This limitation is relevant to the interpretation of the baseline: the absence of an observation must not be interpreted as evidence that a particular component is absent.

No hardware component was replaced or deliberately disabled during this stage.



00.3 — Initial Boot

Following the physical inspection, the laptop was powered on in its original experimental state.

The first boot was used to confirm that the machine remained operational and to establish the state of the installed operating system before any modifications were introduced.

The system successfully booted into Arch Linux using the Xfce desktop environment. Despite its physical degradation, the machine remained sufficiently functional for continued experimentation.

At this point, the laptop was still considered to be in its original baseline state.

00.4 — Automated System Enumeration

After confirming that the system was operational, an automated hardware and software inventory was performed using:

inxi -Fxxx

The purpose of this enumeration was to obtain a broad technical overview of the machine without manually inspecting every subsystem individually.

The processor was identified as an **Intel Celeron 1000M** with two physical cores, and the system contained **4 GiB of RAM**.

Storage consisted of a **Toshiba MQ04ABF100 1 TB mechanical hard drive operating at 5400 RPM**. The system used a GPT partition table containing a VFAT /boot partition and an ext4 root filesystem, with swap provided through zram.

The graphics subsystem consisted of integrated Intel HD Graphics using the i915 driver. The internal display was identified as an LG Display panel with a resolution of **1366×768** and a reported diagonal size of approximately **15.5 inches**.

The network hardware included a **Ralink RT5390 802.11n wireless adapter** and a **Qualcomm Atheros AR8161 Gigabit Ethernet controller**. Both interfaces were subsequently confirmed to be operational.

Additional hardware detected during enumeration included a **Chicony USB 2.0 UVC webcam**, Intel High Definition Audio hardware, a **TSSTcorp SN-208D optical drive**, and the integrated touchpad.

The software environment at the time of collection consisted of:

Component	Version / State
Operating System	Arch Linux
Kernel	7.1.6-arch1-1
Desktop Environment	Xfce 4.20.2
Window Manager	xfwm4 4.20.0
Display Server	X.Org 1.21.1.24
Display Manager	LightDM 1.33.1
Init System	systemd 261
Audio System	PipeWire 1.6.8
Shell	Bash 5.3.15
Browser	Firefox 154.0

At the time of enumeration, the system reported **534 installed packages** and approximately **150 running processes**.

The inventory also exposed several hardware and system identifiers, including network interface identifiers and other values capable of uniquely identifying hardware. Complete identifier values were retained in the private research material and are not reproduced in this report.

The inxi output represents a point-in-time snapshot. Dynamic values such as CPU frequency, temperature, process count, memory usage, interface state, and battery state therefore describe the condition of the system at the moment of collection rather than permanent properties of the machine.

00.5 — BIOS/UEFI Inspection

The system firmware was subsequently inspected to establish its original configuration.

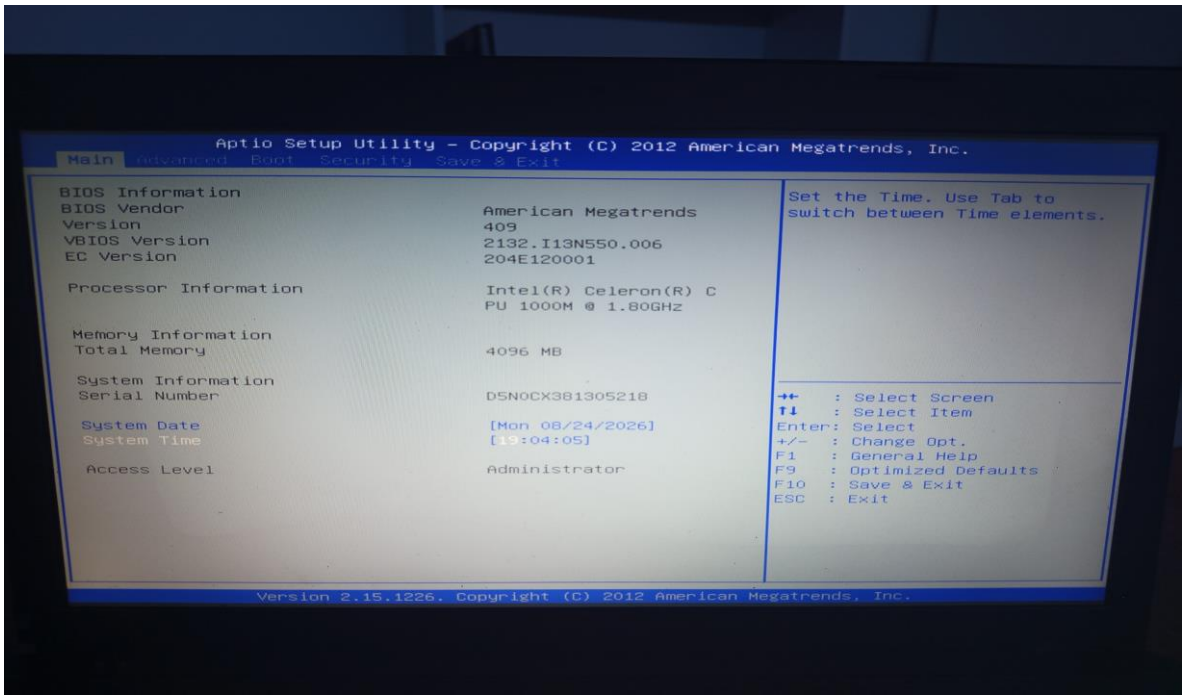
The firmware was identified as **American Megatrends X55C.409**, dated **09 January 2013**.

The machine was operating in **UEFI mode**. Secure Boot was disabled, and no administrator, user, or HDD password was configured at the time of inspection.

The boot configuration contained entries associated with the Linux Boot Manager, a fallback Linux Boot Manager, the optical drive, and network/PXE boot functionality.

Additional firmware settings relevant to the later security and anonymity analysis included legacy USB support, the internal pointing device, and Intel Virtualization Technology.

No BIOS/UEFI setting was modified during this phase. The firmware configuration therefore forms part of the original baseline against which future changes can be compared.



00.6 — Functional Verification

Following the hardware and firmware inventory, the principal components of the system were tested directly.

The internal display and touchpad were fully functional. Audio output was operational, while the integrated microphone was functional but produced very poor-quality recordings.

The Chicony webcam was confirmed to be functional.

Both Wi-Fi and Ethernet connectivity were successfully tested. The laptop was also connected to a **Xiaomi Redmi 13** through USB tethering. The smartphone exposed an RNDIS network interface to Linux and provided Internet connectivity through its own Wi-Fi connection.

The USB ports were tested and found to be functional.

The optical drive was tested and confirmed to be capable of reading CDs.

The battery was found to be heavily degraded. The system reported a current capacity of approximately **11 Wh**, compared with a design capacity of **48.6 Wh**, corresponding to approximately **22.7%** of its original reported capacity. In practical terms, the machine requires external power for normal operation.

The integrated keyboard was considered unsuitable as the primary input device due to its physical degradation. An external **Razer X10 V3** keyboard was therefore designated for use during the remainder of the experiment.

00.7 — Baseline System Profile

The baseline state established during this phase can be summarized as follows:

Component	Baseline State
CPU	Intel Celeron 1000M, 2 cores
RAM	4 GiB
Storage	Toshiba MQ04ABF100, 1 TB, 5400 RPM HDD
GPU	Integrated Intel HD Graphics
Display	LG Display, 1366×768
Wi-Fi	Ralink RT5390 802.11n
Ethernet	Qualcomm Atheros AR8161 Gigabit Ethernet
Webcam	Chicony USB 2.0 UVC
Audio	Intel HD Audio / PipeWire
Optical Drive	TSSTcorp SN-208D

Operating System	Arch Linux
Desktop Environment	Xfce 4.20.2
Kernel	7.1.6-arch1-1
Browser	Firefox 154.0
Firmware	American Megatrends X55C.409
Secure Boot	Disabled

From a physical perspective, the most significant limitations identified during the baseline were the damaged left hinge, degraded keyboard, and heavily degraded battery.

Despite these limitations, the machine retained sufficient functionality to serve as a practical experimental platform.

00.8 — Information Not Established

The baseline was intended to establish a reliable starting point rather than exhaustively enumerate every property of the system.

Several characteristics therefore remained undetermined at the conclusion of this phase. These include the complete motherboard-level hardware topology, exact RAM module configuration, HDD health according to SMART data, complete USB and PCI topology, exact microphone and touchpad controllers, operational IPv6 state, active DNS and routing configuration, complete systemd service inventory, listening network services, browser fingerprinting characteristics, and the exact encryption state of the storage.

These properties were intentionally left for subsequent investigation rather than being inferred from incomplete evidence.

In particular, the presence of an ext4 filesystem does not establish whether the storage is encrypted. This requires separate verification.

00.9 — Baseline Preservation

The state documented during this phase was designated:

BASELINE-00

From this point onward, any modification to the hardware, firmware, operating system, network configuration, or browser environment is considered an experimental intervention.

Whenever possible, later measurements will use the same methods and commands established during the baseline. This allows subsequent configurations to be compared against the original state using consistent evidence rather than subjective impressions.

The experimental sequence is therefore defined as:

Baseline → Observation → Modification → Measurement → Comparison

The original baseline data remains preserved and is not altered to reflect subsequent changes.

00.10 — Phase Conclusion

Phase 00 established the initial physical, hardware, firmware, software, and functional state of the ASUS X55C before the beginning of the privacy and anonymity transformations.

The machine is an old and physically degraded consumer laptop, but it remains sufficiently functional for controlled experimentation. The baseline identified several hardware and software surfaces relevant to the research, including wireless and wired networking, USB connectivity, webcam and microphone functionality, storage, firmware, operating system, and browser components.

Equally important, the phase established a distinction between **known**, **observed**, and **undetermined** properties of the system. This prevents assumptions from being treated as baseline facts and provides a clearer foundation for subsequent investigation.

The next phase therefore moves from characterization to **reconnaissance**. Rather than immediately modifying the system, Phase 01 will examine what information the machine exposes while operating normally, with particular attention to local identifiers, network identity, services, processes, persistent information, and other potential sources of correlation or identification.

PHASE 01

RECONNAISSANCE

Identifying what the system exposes before modification

01.1 — Objective

The purpose of this phase was to determine what information the ASUS X55C exposes in its original, pre-modification state.

The reconnaissance focused on system identity, hardware identifiers, user information, network configuration, running services, processes, persistence mechanisms, storage, and other locally accessible information that could contribute to identification or correlation.

The guiding question for this phase was:

What does the machine currently reveal?

No privacy, anonymity, networking, operating-system, or hardware modifications were intentionally performed during this phase.

01.2 — Methodology

Reconnaissance was performed locally through system enumeration tools and direct inspection of relevant Linux interfaces.

The primary sources used during acquisition included:

inxi · ip · nmcli · ss · ps · systemctl · lspci · lsusb · lsblk · udevadm

Additional information was obtained from:

/etc/os-release · /etc/machine-id · /proc/sys/kernel/random/boot_id · /sys/class/dmi/id/

Systemd configuration, XDG configuration, and Firefox installation and profile information were also inspected.

Collection was performed in both normal-user and privileged contexts where elevated access was required.

Raw command output was preserved separately as experimental evidence. The present section contains the analytical results rather than the complete acquisition output.

01.3 — System and User Identity

The system was identified as an **ASUS X55C** running **Arch Linux x86_64**, with kernel 7.1.6-arch1-1 and Xfce 4.20.2.

The active local account was:

Property	Value
Username	demigod7777lab
UID	1000
GID	1000
Privilege group	wheel
Elevation	sudo

The system hostname was:

aetherlab

These values constitute local identity surfaces. They may appear in system logs, processes, configuration files, environment variables, and other locally accessible resources.

Although these identifiers primarily describe the local system rather than its external network identity, they represent information that would need to be considered when attempting to minimize local attribution.

01.4 — Persistent Identifiers

The reconnaissance identified several persistent identifiers associated with the system and its hardware.

Relevant sources included:

- /etc/machine-id
- DMI product and system identifiers
- motherboard identifiers
- storage serial information
- hardware-related UUIDs

The complete identifier values were retained in the raw research evidence and are not reproduced unnecessarily in the main document.

A separate boot-specific identifier was also observed through:

/proc/sys/kernel/random/boot_id

This distinction is important. The machine ID represents a persistent system-level identifier, while the boot ID is associated with a particular boot instance.

The presence of multiple independent identifiers demonstrates that the identity of the machine is distributed across several layers. Removing or changing a single identifier would therefore not constitute complete identity removal.

01.5 — Hardware Exposure

The operating system exposed identifying information associated with multiple physical components.

Relevant hardware identified during reconnaissance included:

Component	Identified Hardware
CPU	Intel Celeron 1000M
Storage	Toshiba MQ04ABF100 HDD
Wi-Fi	Ralink RT5390
Ethernet	Qualcomm Atheros AR8161
Webcam	Chicony USB webcam
Graphics	Intel Ivy Bridge integrated graphics
Audio	Intel 7 Series/C216 audio controller

PCI and USB identifiers were also observable.

Some hardware-level information, particularly information obtained through DMI interfaces, required elevated privileges to access.

This demonstrates that the system's observable identity is not exclusively determined by the operating system. Information about the underlying physical platform remains exposed to the operating environment.

01.6 — Network Exposure

The system exposed multiple network interfaces:

Interface	Role
wlp2s0	Internal Wi-Fi
enp3s0	Ethernet
enp0s20u2	USB/RNDIS network interface
lo	Local loopback

The internal Wi-Fi and Ethernet interfaces were present but were not the active Internet path during acquisition.

The active connection used the following topology:

ASUS X55C → USB/RNDIS → Redmi 13 → Wi-Fi → Internet

The RNDIS interface exposed its own MAC address and private IPv4 configuration.

The system also exposed routing information, DNS configuration, IPv6 information, and NetworkManager state.

This demonstrates that network identity is composed of several interacting layers rather than being represented by a single IP address.

01.7 — Services, Processes and Persistence

The running system contained the expected components of a graphical Linux desktop environment, including system services, user processes, IPC mechanisms, and application processes.

Among the active system services identified were:

- NetworkManager
- LightDM
- systemd-timesyncd
- systemd-logind
- udisks2
- upower
- polkit
- wpa_supplicant
- user-session services

Running processes and sockets were also enumerated.

Persistence mechanisms were investigated through systemd timers, XDG autostart locations, cron locations, and user-level configuration.

No conventional cron directories were found within the inspected configuration. Systemd timers and XDG autostart mechanisms were present.

An important distinction was maintained during the analysis: the existence of a socket was not automatically interpreted as network exposure. Local UNIX sockets were distinguished from sockets providing network communication.

01.8 — Software and Session Fingerprint

The software environment exposed several characteristics that can contribute to system fingerprinting.

Relevant characteristics included:

- Arch Linux
- Linux kernel version
- Xfce version
- X.Org
- systemd
- PipeWire
- Bash
- installed package set
- Firefox 154.0
- locale
- timezone
- X11 session information

Additional session-specific information was accessible through environment variables and runtime directories.

None of these characteristics necessarily provides a unique identity in isolation. However, their combination contributes to a broader observable configuration that can distinguish one system from another.

Browser-specific fingerprinting was deliberately not treated as complete during this phase. A more detailed browser-level analysis is reserved for **Phase 05 — Behavioral Footprint**.

01.9 — Principal Findings

The reconnaissance identified several independent information surfaces:

Surface	Observed	Relevance
Host / user identity	Yes	Medium
Machine ID	Yes	High
Boot ID	Yes	Medium
DMI / hardware identifiers	Yes	High
Storage identifiers	Yes	High
MAC addresses	Yes	High
IP / network configuration	Yes	High
Software fingerprint	Yes	Medium
Services / processes	Yes	Medium
Session information	Yes	Medium

The principal finding of this phase is that **the machine does not possess a single identity**.

Instead, its observable identity is distributed across system identifiers, physical hardware, network interfaces, software configuration, and session state.

Consequently, modifying one identifier cannot be assumed to anonymize the machine as a whole.

01.10 — Phase Conclusion

Phase 01 successfully characterized the principal information exposed by the ASUS X55C before experimental modification.

The reconnaissance identified persistent system identifiers, hardware identifiers, network-interface identifiers, dynamic network information, software fingerprinting surfaces, active services, processes, persistence mechanisms, and session information.

All acquisition was performed without intentionally modifying the system.

The raw outputs have been preserved separately from this analytical record, allowing the findings presented here to remain supported by the original evidence.

The reconnaissance therefore establishes the information-exposure baseline required for the next stage of the investigation.

PHASE 02

HARDWARE ANALYSIS

Assessing physical components as sources of exposure, identification, and attack surface

02.1 — Objective

The purpose of this phase was to analyze the physical hardware of the ASUS X55C from the perspective of privacy, anonymity, identification, and attack-surface reduction.

Phase 00 established the physical and technical baseline, while Phase 01 demonstrated that the system exposes multiple independent identity surfaces, including DMI information, storage identifiers, MAC addresses, PCI/USB identifiers, network information, software characteristics, and session information.

The objective of this phase was therefore to determine which physical components contribute to these surfaces, which components could theoretically be removed or isolated, what functionality would be lost by doing so, and which equivalent software-level interventions may provide similar reductions in exposure.

No physical modification was performed during this phase.

02.2 — Preservation of the Experimental Hardware

The ASUS X55C already exhibits significant physical degradation, including damage around the left hinge, a degraded integrated keyboard, and a heavily degraded battery.

Because the machine itself constitutes the experimental platform, physically dismantling components could introduce damage unrelated to the research and potentially terminate the experiment.

For this reason, physical interventions were deliberately excluded from the practical procedure.

The hardware modifications discussed in this phase therefore represent theoretical interventions: actions that could reasonably be performed if preservation of the machine were not a constraint.

This distinction is important. A technically desirable intervention does not necessarily justify physically executing it.

02.3 — Hardware Relevant to Anonymity

The principal hardware identified during the baseline was evaluated according to its potential contribution to identification, fingerprinting, persistence, or information exposure.

Component	Identified State	Anonymity Relevance
CPU	Intel Celeron 1000M	Low–Medium
RAM	4 GiB	Low
HDD	Toshiba MQ04ABF100, 1 TB	High
Wi-Fi	Ralink RT5390	High
Ethernet	Qualcomm Atheros AR8161	High
Motherboard	ASUS X55C platform	High
Webcam	Chicony USB 2.0 UVC	Medium
Audio	Intel HD Audio	Low–Medium
Microphone	Functional	Medium
GPU	Integrated Intel HD Graphics	Low–Medium
Display	LG Display, 1366×768	Low–Medium
Optical Drive	TSSTcorp SN-208D	Low
Touchpad	Functional	Low
Keyboard	Physically degraded	Very Low
Battery	Heavily degraded	Very Low
USB interfaces	Functional	Medium

These classifications indicate potential relevance to the research and do not imply that every component contains a unique identifier.

02.4 — Persistent Hardware Identity

The most significant hardware-related exposure surfaces are components capable of providing persistent identifiers.

The motherboard and firmware can expose DMI/SMBIOS information. The storage device can expose hardware and storage identifiers, while network adapters expose interface-specific identifiers such as MAC addresses.

Phase 01 confirmed the presence of DMI and hardware identifiers, storage identifiers, and hardware-related UUIDs.

This demonstrates that the identity of the machine is not determined solely by the operating system.

Replacing or modifying one component therefore does not automatically eliminate the identity of the entire platform. Multiple independent hardware surfaces can continue contributing to the observable identity of the system.

02.5 — Storage

The system uses a **Toshiba MQ04ABF100 1 TB mechanical HDD operating at 5400 RPM**.

Storage represents one of the most significant persistent surfaces because it contains both hardware-level information and the persistent state of the operating system.

Potentially relevant information stored on the device includes:

- filesystem identifiers;
- system configuration;
- system and application logs;
- browser profiles;
- application data;
- cached information;
- credentials or authentication material;
- locally stored identifiers;
- machine-specific configuration.

Phase 01 also classified storage identifiers as a high-relevance identity surface.

Simply removing the HDD would eliminate access to the current operating-system installation and therefore is not considered a practical privacy intervention.

The theoretically appropriate intervention would instead be **replacement with another storage device**, followed by a controlled comparison of the resulting system state and identifiers.

No replacement was performed during this phase.

02.6 — Network Hardware

The two principal physical network adapters are the **Ralink RT5390 Wi-Fi adapter** and the **Qualcomm Atheros AR8161 Gigabit Ethernet controller**.

Both represent relevant identity surfaces because network interfaces possess their own hardware identity and can expose MAC addresses.

Phase 01 confirmed multiple network interfaces and identified MAC addresses and network configuration among the most relevant observed exposure surfaces.

The internal Wi-Fi adapter is theoretically one of the strongest candidates for physical removal because its functionality can potentially be replaced by another network path.

However, no physical removal was performed.

The machine has already demonstrated the ability to use USB tethering through a **Redmi 13**, where the phone exposed an RNDIS network interface and provided Internet connectivity.

A theoretical configuration could therefore remove the internal Wi-Fi adapter while retaining network functionality through Ethernet or USB tethering.

02.7 — Sensors and Peripheral Hardware

The machine contains a **Chicony USB 2.0 UVC webcam** and an integrated microphone. Both were confirmed functional during the baseline.

These components are less significant as persistent identity sources than the storage and network hardware, but they represent unnecessary sensors for a system whose objective is privacy and anonymity research.

The webcam can also be enumerated as USB hardware, meaning its presence contributes to the observable hardware configuration.

Theoretically, the webcam could therefore be physically disconnected or removed.

The microphone could similarly be physically isolated if its implementation permitted this without damaging the machine.

Neither intervention was executed.

02.8 — Other Hardware Surfaces

The ASUS X55C also contains an optical drive, USB interfaces, integrated graphics, display hardware, touchpad, keyboard, and battery.

The optical drive was identified as a **TSSTcorp SN-208D** and was confirmed operational. Its relevance to anonymity is comparatively low because it does not constitute a primary network or persistent identity surface.

Its theoretical removal could reduce the amount of hardware exposed to enumeration, but the expected privacy benefit is limited.

The CPU, RAM, GPU, display, motherboard, and USB interfaces should remain in place because they are either essential to normal operation or sufficiently useful for continuation of the research.

Removing hardware solely because it can be fingerprinted would be counterproductive if the resulting machine could no longer perform the experiment.

02.9 — Theoretical Physical Interventions

Based on the observed hardware, the following interventions were considered:

Intervention	Expected Benefit	Functional Impact	Decision
Remove Wi-Fi adapter	Reduce wireless hardware identity	Internal Wi-Fi lost	Theoretically justified
Remove webcam	Remove unnecessary sensor and USB device	Webcam lost	Theoretically justified
Isolate microphone	Remove or limit unnecessary sensor	Microphone lost	Theoretically justified
Remove optical drive	Reduce unnecessary hardware	Optical media unavailable	Optional
Replace HDD	Replace persistent storage state/device	Requires OS migration	Appropriate future intervention
Remove Ethernet	Remove network interface identity	Wired networking lost	Not justified
Remove RAM	Minimal privacy benefit	System unusable	Reject
Remove CPU	No practical benefit	System unusable	Reject
Remove motherboard	Potentially changes platform identity	System destroyed	Reject

Remove GPU/display	No practical benefit	System unusable	Reject

The objective is therefore not maximum hardware reduction.

It is **maximum reduction of unnecessary exposure while preserving experimental functionality**.

02.10 — Physical Removal vs. Logical Disablement

Physical removal and logical disablement are not equivalent interventions.

A physically removed device is no longer present on the relevant hardware bus and cannot normally be enumerated by the operating system.

A logically disabled device, by contrast, remains physically present and may continue to be visible to lower layers of the system, even if its driver is prevented from loading or applications are prevented from accessing it.

For example, a webcam could remain visible through USB enumeration while its UVC driver is prevented from providing a usable video device.

The relevant exposure hierarchy can therefore be represented as:

Physical Hardware → Bus Enumeration → Kernel Detection → Driver Initialization → Device Interface → Application Access

Intervention at one layer does not imply that information disappears from every other layer.

This distinction is particularly important for the anonymity objective. Preventing an application from accessing a device is different from preventing Linux from identifying the device, and neither is necessarily equivalent to physically removing it.

02.11 — Practical Non-Destructive Strategy

Because physical modifications were excluded, the practical strategy is to investigate equivalent logical controls during subsequent phases.

Potential areas of investigation include:

- disabling unnecessary network interfaces;

- preventing specific drivers from initializing;
- controlling access to camera and audio devices;
- disabling unnecessary services;
- controlling device permissions;
- reducing unnecessary network exposure;
- examining which hardware identifiers remain visible after logical disablement.

The purpose of these interventions is not to claim that the hardware has disappeared.

Instead, the objective is to determine **which layer of exposure has been eliminated and which layers remain observable**.

This allows physical and logical interventions to be compared without introducing unnecessary risk to the experimental machine.

02.12 — Expected Effect on the Hardware Attack Surface

If the theoretically preferred physical interventions were implemented, the machine would retain its fundamental computing platform while eliminating several unnecessary peripheral surfaces.

The most significant theoretical reductions would result from:

Wi-Fi removal → reduction of wireless-interface identity and wireless functionality.

Webcam removal → elimination of the physical camera and its USB enumeration.

Microphone isolation → elimination or reduction of an unnecessary audio sensor.

Storage replacement → replacement of persistent storage state and potentially storage-level identifiers.

These changes would not eliminate the identity of the machine.

The motherboard, firmware, CPU, GPU, RAM, display, and other components would continue contributing to the hardware fingerprint, while any remaining network interface would continue providing its own identity.

Hardware reduction should therefore be understood as **surface reduction**, not complete anonymization.

02.13 — Limitations

The analysis in this phase is constrained by the decision not to physically dismantle the ASUS X55C.

Consequently, the exact physical construction and accessibility of certain components were not experimentally verified.

In particular, this phase does not establish the precise physical connection method of the webcam and microphone, the complete RAM module configuration, the complete motherboard topology, or whether additional hardware such as Bluetooth is physically present.

These properties must not be inferred from their absence in the available enumeration.

Furthermore, logical disablement was not treated as equivalent to physical removal because the former has not yet been experimentally measured for each device.

The conclusions of this phase are therefore based on the confirmed baseline hardware and the theoretical consequences of potential interventions.

02.14 — Phase Conclusion

Phase 02 established that the ASUS X55C contains several hardware-level surfaces relevant to privacy and anonymity, with the most significant being the storage device, network adapters, motherboard/DMI information, and peripheral sensors.

The theoretically strongest physical interventions would be removal of the internal Wi-Fi adapter and webcam, isolation of the microphone, and eventual replacement of the HDD. The optical drive could also be removed, although its anonymity relevance is comparatively low.

CPU, RAM, motherboard, GPU, display, Ethernet, and USB interfaces should remain because their removal would either destroy essential functionality or provide insufficient privacy benefit relative to the cost.

However, because the ASUS X55C is already physically degraded and constitutes the experimental platform itself, no physical modifications were performed.

The central finding of this phase is that **hardware-level anonymity is not binary**. A device can be physically present, detectable by the kernel, initialized by a driver, accessible to applications, or completely removed. These represent distinct exposure states and must be evaluated separately.

PHASE 03

OPERATING SYSTEM PROGRESSION

Evaluating how different operating-system models affect privacy, anonymity, persistence, and usability

03.1 — Objective

The purpose of this phase was to evaluate how progressively privacy-oriented operating systems affect the anonymity, persistence, security, usability, and operational characteristics of the ASUS X55C.

Unlike Phase 02, which focused on individual hardware components, this phase evaluates the **operating system as a complete privacy layer**.

The experiment followed a progressive model:

Windows → Arch Linux → Kicksecure → Tails

The objective was not simply to determine which operating system was "most private". Instead, the purpose was to observe how the characteristics of the same physical machine changed as increasingly restrictive privacy models were introduced, and to identify the point at which additional privacy mechanisms began to compromise normal usability.

The central relationship examined during this phase was:

Privacy / Anonymity ↑ → Persistence and Convenience ↓

The phase also provided an opportunity to verify an important limitation established in the previous stages: changing the operating system does not change the underlying physical identity of the ASUS X55C.

03.2 — Experimental Progression

The operating systems were evaluated sequentially rather than simultaneously. Each stage represented a progressively different approach to privacy and system operation.

Stage	Operating System	Primary Model	Persistence	Privacy Orientation
03.1	Windows	Conventional desktop OS	High	Low

03.2	Arch Linux	User-controlled Linux	High	Configurable
03.3	Kicksecure	Security/privacy-oriented Linux	High	High
03.4	Tails	Amnesic privacy system	Very Low by default	Very High

The progression therefore represents more than a comparison between operating systems.

It represents a transition between fundamentally different models of how a computer can be operated:

Conventional persistent system → User-controlled system → Privacy-oriented persistent system → Amnesic privacy system

03.3 — Windows

Windows represented the conventional operating-system stage of the experiment.

The computer operated as a normal persistent workstation. The operating system, applications, configuration, user data, browser information, and other system state were stored on the internal storage and remained available between sessions.

This model provided the highest degree of convenience.

Applications could be installed normally, files could be stored locally, system configurations could persist, and the machine could be used as a conventional personal computer without requiring a specialized privacy workflow.

However, this persistence also represented an important privacy characteristic.

A conventional persistent operating system continuously maintains local state. Logs, application data, browser information, cached content, configuration files, and other artifacts can remain available after use.

The Windows stage therefore established the conventional end of the operating-system progression: a model that prioritizes **functionality, persistence, and convenience** rather than minimizing retained state.

Observed Characteristics

The Windows environment demonstrated:

- Persistent operating-system installation.
- Persistent user environment.

- Persistent application configuration.
- Persistent local files.
- Conventional browser storage.
- Conventional network configuration.
- Normal hardware enumeration.
- Normal system identification.
- High usability and application compatibility.

The important finding was not that Windows inherently prevents privacy-oriented operation, but that its conventional operating model assumes persistence and general-purpose usability as primary objectives.

03.4 — Arch Linux

The next stage replaced Windows with Arch Linux.

This represented the first major change in the experimental model.

Arch provided substantially greater control over the operating environment. Services, packages, drivers, networking components, applications, and other system elements could be selectively configured according to the requirements of the investigation.

This produced an important experimental result:

Greater control over the operating system does not automatically produce anonymity.

The ASUS X55C remained the same physical machine.

The motherboard, firmware, CPU, storage device, network adapters, and other hardware characteristics identified during Phases 00–02 remained present. Changing the operating system therefore did not eliminate the underlying hardware identity of the machine.

Arch consequently provided a substantially greater ability to **control the system's exposure**, but did not inherently provide an amnesic or anonymous environment.

Observed Characteristics

The Arch stage demonstrated:

- Persistent installation on internal storage.
- Full local user environment.
- Persistent configuration.

- Persistent application and browser data.
- Extensive administrative control.
- Manual control over installed services and software.
- Full access to the underlying hardware.
- Conventional network operation.
- Conventional filesystem persistence.
- High usability.

The most important difference from the Windows stage was therefore not that the machine became anonymous.

The difference was that the operator gained significantly greater control over **what the computer was running, which components were present, and how the system could be configured**.

Arch became the first stage in which privacy hardening could be performed systematically rather than relying primarily on the default behavior of a conventional desktop operating system.

03.5 — Kicksecure

The experiment then progressed from Arch Linux to Kicksecure.

This represented a shift from a general-purpose, highly configurable Linux environment toward an operating system specifically designed around security and privacy considerations.

The difference was primarily architectural.

With Arch, privacy depended heavily on the configuration created by the operator.

With Kicksecure, a larger portion of the environment was already designed around security and privacy principles. This reduced the amount of work required to construct a privacy-oriented environment manually while introducing a more specialized operational model.

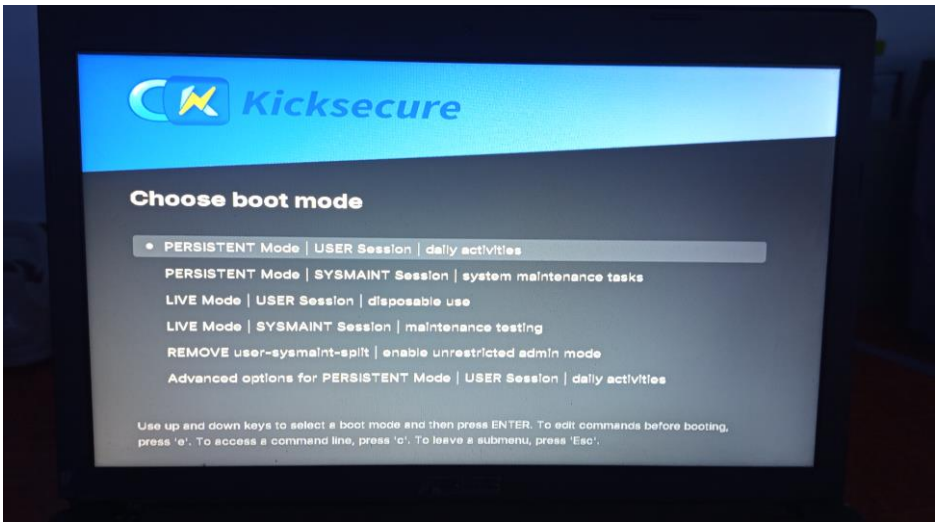
However, Kicksecure remained a **persistent operating system**.

Local files, configurations, applications, and system state could continue to persist between sessions. It therefore did not represent an amnesic environment.

This distinction is central to the experiment:

A security- and privacy-oriented operating system is not automatically an amnesic operating system.

Kicksecure consequently occupied an intermediate position between a highly configurable conventional Linux installation and the fundamentally different operating model represented by Tails.



Observed Characteristics

The Kicksecure stage demonstrated:

- Persistent operating-system installation.
- Persistent filesystem.
- Persistent configuration.
- Persistent user environment.
- Stronger privacy and security orientation.
- Greater reliance on mechanisms provided by the operating environment.
- Reduced requirement to manually construct every privacy mechanism.
- Continued compatibility with normal desktop operation.
- Continued dependence on the underlying physical hardware.

Boot Configuration Issue

An operational issue was encountered during the transition to Kicksecure.

The installed Kicksecure environment did not automatically appear as a selectable entry in the existing Arch systemd-boot environment. Attempts were made to identify its EFI/GRUB configuration and integrate the installation with the existing boot infrastructure.

This demonstrated an additional practical consequence of introducing increasingly specialized operating environments:

Greater system specialization can also increase operational complexity.

Rather than extensively modifying the existing Arch boot infrastructure, the investigation proceeded to the Tails environment using external media.

This limitation was recorded as part of the experimental result. It affected the operational workflow but did not invalidate the assessment of Kicksecure as an operating-system model.

03.6 — Tails

The final operating-system stage used Tails.

This represented the largest conceptual and architectural change of the phase.

Unlike Windows, Arch Linux, and the installed Kicksecure environment, Tails was operated from external media rather than as the primary persistent operating system installed on the internal drive.

The purpose of this architecture is fundamentally different from that of a conventional persistent workstation.

Instead of treating the computer as a permanently configured environment, Tails is designed around temporary sessions and significantly reduced default persistence.

The relationship between the user and the machine therefore changed substantially.

A conventional persistent workflow can be represented as:

Boot → Use → Save → Shut down → Continue later

The Tails model is closer to:

Boot → Establish temporary environment → Use → Shut down → Discard session

This represented the most significant privacy-oriented change observed during Phase 03.

Observed Characteristics

The Tails stage demonstrated:

- Operation from external media.

- Reduced dependence on the internal operating-system installation.
- Temporary session-oriented operation.
- Significantly reduced default persistence.
- Reduced reliance on internal storage for normal session state.
- A substantially more restrictive privacy model.
- Greater separation between individual sessions.
- Reduced convenience compared with a conventional persistent installation.

The cost of this model was also immediately apparent.

The system could no longer be treated in exactly the same way as a conventional desktop installation. Persistent applications, configurations, files, and workflows were no longer the default operating model.

This represented the first stage in which the experiment produced a significant **privacy-versus-convenience trade-off**.

03.7 — Operating-System Models

The four operating systems represented four distinct approaches to operating a computer.

Operating System	Primary Characteristic
Windows	Conventional persistent operation
Arch Linux	Extensive operator control
Kicksecure	Security/privacy-oriented persistent operation
Tails	Amnesic-oriented privacy operation

The progression can therefore be represented conceptually as:

Conventional System

↓

User-Controlled System

↓

Privacy/Security-Oriented System

↓

Amnesic Privacy System

The experiment consequently did not demonstrate a simple transition from "non-private" to "private".

Instead, each operating environment changed a different aspect of the system's exposure and persistence model.

03.8 — Persistence

Persistence was one of the clearest variables affected by the operating-system progression.

Windows maintained a conventional persistent environment.

Arch maintained the same general persistent model while providing substantially greater administrative control.

Kicksecure continued to use a persistent installation while introducing a stronger security and privacy orientation.

Tails represented the major change because the normal operating model became session-based rather than permanently persistent.

Operating System	Persistent Environment	Session Persistence
Windows	Yes	High
Arch	Yes	High
Kicksecure	Yes	High
Tails	No by default	Low

This does not mean that Tails is incapable of persistence under every configuration. Rather, persistence is no longer the fundamental assumption of the system.

That distinction is central to the experiment.

03.9 — Hardware Identity

Changing the operating system did not eliminate the underlying hardware identity identified during the previous phases.

The same ASUS X55C hardware remained present throughout the progression.

Consequently, the following classes of information remained fundamentally associated with the physical platform:

- CPU characteristics;
- motherboard and platform characteristics;
- DMI/SMBIOS information;
- storage hardware;
- network hardware;
- USB hardware;
- display characteristics;
- other hardware-specific properties.

This demonstrates an important limitation of operating-system-level anonymity:

Changing the operating system does not equal changing the physical identity of the machine.

An operating system can influence what information is exposed by software and how the machine communicates, but it cannot retroactively remove the existence or characteristics of the underlying physical hardware.

This is consistent with the conclusions established during Phase 02.

03.10 — Usability Trade-Off

The most important result of Phase 03 was not simply the increase in privacy orientation.

It was the progressive loss of convenience associated with increasingly restrictive privacy models.

Windows provided the least restrictive environment and the highest degree of conventional usability.

Arch retained most general-purpose functionality while providing significantly greater control over the operating environment.

Kicksecure introduced a more specialized security and privacy model while retaining a persistent desktop environment.

Tails produced the strongest change because persistence and convenience were no longer treated as primary design objectives.

Stage	Privacy Orientation	Persistence	Usability
Windows	Low	High	Very High
Arch	Configurable	High	High

Kicksecure	High	High	High–Medium
Tails	Very High	Very Low by default	Medium–Low

These usability classifications are contextual rather than absolute. Tails can remain highly usable for privacy-oriented workflows, but it is less convenient as a replacement for a conventional permanently configured desktop.

The purpose of this comparison is therefore not to determine which operating system is "best".

It is to determine **what is sacrificed as the privacy model becomes progressively more restrictive**.

03.11 — Experimental Findings

Several conclusions were established during the progression.

First, the operating system is a major component of the machine's privacy model. The same physical ASUS X55C behaved substantially differently depending on the operating environment.

Second, greater configurability does not automatically provide anonymity. Arch demonstrated that a highly controlled Linux environment can remain a conventional persistent system.

Third, security-oriented systems such as Kicksecure provide a more specialized privacy model without necessarily abandoning persistence.

Fourth, Tails introduced the largest architectural change because the operating system itself is designed around temporary sessions and reduced persistence.

Fifth, changing the operating system did not eliminate the physical identity of the ASUS X55C. Hardware-level identifiers and characteristics remained properties of the machine rather than of the operating system.

Finally, the experiment demonstrated that privacy is not a binary property.

The machine did not transition from:

"anonymous" → "not anonymous"

Instead, different layers of exposure were progressively altered:

Persistent State → Software Control → Privacy-Oriented Architecture → Amnesic Operation

03.12 — Limitations

This phase evaluated the operating-system layer and therefore does not establish complete anonymity of the ASUS X55C.

The underlying hardware remained unchanged, meaning that the hardware-level identifiers and fingerprinting surfaces identified during Phases 01 and 02 remained relevant.

The experiment also did not attempt to quantify every possible network-side or browser-side fingerprint at this stage.

The Tails environment was evaluated primarily as an operating-system model. Browser-specific anonymity mechanisms were not treated as part of this phase.

The Kicksecure stage was also affected by the existing boot configuration. Although the system was installed on the machine's storage, it was not integrated into the existing Arch systemd-boot menu during the experiment.

This limitation affected the operational workflow but did not invalidate the comparison of the operating-system models.

03.13 — Conclusion

Phase 03 demonstrated that changing the operating system can substantially alter the privacy characteristics of the same physical computer without eliminating its underlying hardware identity.

The progression from Windows to Arch Linux increased operator control while retaining the conventional persistent-desktop model.

The transition from Arch to Kicksecure introduced a more explicit security and privacy architecture while continuing to rely on persistent storage.

The transition to Tails represented the largest architectural change, replacing the conventional persistent-workstation model with a temporary, amnesic-oriented environment.

The principal finding of this phase is therefore:

The strongest reduction in persistence was achieved not simply by configuring a conventional Linux installation more aggressively, but by changing the operating model itself to an amnesic environment.

This reduction, however, introduced a corresponding decrease in convenience and persistence.

The operating-system progression can therefore be summarized as:

Windows → Conventional persistent operation

Arch Linux → Controlled persistent operation

Kicksecure → Security/privacy-oriented persistent operation

Tails → Amnesic privacy-oriented operation

The results demonstrate that operating-system selection is a major component of the overall privacy architecture, but it cannot independently eliminate the physical identity of the device or guarantee anonymity.

PHASE 04

NETWORK

Evaluating network identity, routing, and privacy mechanisms

04.1 — Objective

The purpose of this phase was to evaluate how network configuration affects the privacy and anonymity of the ASUS X55C.

Following the progression from Windows to Arch Linux, Kicksecure, and finally Tails, the investigation moved from the operating-system layer to the network layer.

The analysis focused on network interfaces, MAC addressing, IP addressing, DNS behavior, routing, Tor, and Tor Bridges.

The central relationship examined during this phase was:

Network Exposure ↓ → Anonymity ↑

At the same time, the effect of additional privacy mechanisms on connectivity, complexity, and usability was evaluated.

04.2 — Network Interface

The ASUS X55C's network hardware was successfully detected by Tails and remained operational throughout the experiment.

The physical network adapters identified during the previous phases were not physically modified.

This established an important distinction between **hardware identity** and **network identity**.

The underlying network hardware remained unchanged, while the operating environment was able to control how that hardware interacted with the surrounding network.

The experiment therefore demonstrated that network exposure can be modified without physically replacing or removing the network adapters.

04.3 — MAC Address Anonymization

Tails' **MAC Address Anonymization** was enabled during the experiment.

After activation, the network interface presented a MAC address different from the original hardware MAC address identified during the previous phases.

This demonstrated that the physical MAC address can be concealed from the local network without physically modifying the network adapter.

The original hardware identifier remained present within the device, but it was no longer the identifier presented during normal network communication.

This represented the first significant reduction in hardware-level identification at the network layer.

04.4 — IP Address

The network configuration was examined after connectivity had been established.

The ASUS X55C successfully obtained a local network address and maintained Internet connectivity.

However, two distinct levels of addressing had to be considered.

The **local IP address** identifies the machine within its immediate network environment, while the **public IP address** represents the apparent origin of Internet traffic from the perspective of external destinations.

This distinction became particularly relevant after the Tails networking architecture was enabled, since the public Internet no longer directly observed the original public address associated with the connection.

04.5 — Tor

Tails was successfully connected to the Tor network and Internet connectivity was established through the Tails/Tor architecture.

This produced the most significant change in the network layer during the phase.

The resulting communication model can be represented as:

ASUS X55C → Local Network → Tor → Internet

Instead of communicating directly with Internet destinations, the ASUS X55C's traffic was routed through the Tor network.

External destinations therefore no longer directly observed the original public IP address associated with the network connection.

The network identity visible to Internet destinations was consequently separated from the original connection, representing a substantial increase in anonymity compared with a conventional direct Internet connection.

04.6 — DNS and Network Isolation

DNS and general network behavior were also examined while operating within Tails.

The resulting networking environment was substantially more restrictive than the conventional Arch environment used previously.

Applications were not simply permitted to establish arbitrary direct Internet connections outside the Tails networking architecture.

This reduced the possibility of an individual application bypassing the intended anonymous network path.

The privacy benefit therefore did not depend solely on the availability of Tor. The operating system itself was designed around the assumption that Internet traffic should operate through its privacy-oriented networking architecture.

This represented an important architectural difference from a conventional Linux installation, where applications can generally establish network connections according to the local system configuration.

04.7 — Tor Bridge

A **Tor Bridge** was also evaluated as an additional network privacy mechanism.

The purpose of a bridge differs from simply concealing the public IP address.

A conventional Tor connection can still allow the local network or Internet service provider to determine that a connection to the Tor network is being established.

A Bridge introduces an additional intermediary, making the connection to the Tor network less directly identifiable from the local network.

The progression can therefore be represented as:

Direct Internet → Tor → Tor + Bridge

The bridge configuration increased the complexity of the connection while providing an additional layer of network-level privacy.

04.8 — Network Exposure

The principal changes observed during the phase can be summarized as follows:

Network Component	Conventional Environment	Tails
Physical MAC	Exposed to local network	Anonymized
Local IP	Present	Present
Public IP	Directly associated with connection	Hidden behind Tor
DNS	Conventional system networking	Tails/Tor networking model
Internet Routing	Direct	Through Tor
Direct Application Connections	Normally possible	Restricted
Tor Usage	Not inherent	Integrated
Tor Bridge	Not applicable	Available

The principal finding is that Tails did not eliminate the existence of the ASUS X55C's network hardware.

Instead, it substantially reduced the amount of network-level information directly exposed by that hardware.

04.9 — Privacy vs. Usability

The network modifications introduced a clear operational trade-off.

The conventional networking model provided simpler and more direct connectivity.

Tails introduced additional network infrastructure between the ASUS X55C and Internet destinations. This increased privacy but also introduced additional complexity and potential performance limitations.

Tor connections can be slower and less predictable than direct connections, while the use of Bridges introduces an additional dependency into the connection process.

The system therefore remained functional, but the network was no longer optimized purely for speed, simplicity, and direct connectivity.

The observed relationship was consistent with the trade-off identified during Phase 03:

Privacy ↑ → Convenience and Network Performance ↓

The reduction in convenience was therefore not an unintended side effect of the experiment. It formed part of the practical cost associated with the stronger privacy model.

04.10 — Experimental Findings

Several findings were established during this phase.

First, the physical network hardware remained identifiable at the hardware level despite changing the operating system.

Second, Tails was able to modify the network identity presented to the local network through MAC Address Anonymization.

Third, the use of Tor separated the public Internet identity of the ASUS X55C from the original network connection.

Fourth, Tails provided a considerably more restrictive networking environment than the conventional Arch installation.

Fifth, Tor Bridges provided an additional layer of protection against direct identification of Tor usage by the local network.

Finally, the experiment demonstrated that network anonymity is composed of multiple independent layers rather than being provided by a single mechanism.

04.11 — Limitations

The network configuration does not eliminate every possible method of identifying the ASUS X55C.

The underlying physical network hardware remains unchanged, meaning that its original identifiers continue to exist within the device.

Furthermore, Tor does not make the existence of a network connection invisible. Without additional mechanisms such as Bridges, the local network or ISP may still be able to determine that the machine is connecting to the Tor network.

Network anonymity also depends on correct operating-system and application behavior.

This phase therefore evaluates the **network layer specifically** and does not attempt to establish complete anonymity across all possible hardware, browser, application, and behavioral fingerprinting mechanisms.

04.12 — Conclusion

Phase 04 demonstrated a substantial change in the network behavior of the ASUS X55C after moving to Tails.

The physical network identity of the machine remained unchanged, but the identity exposed to the local network and Internet could be significantly reduced through MAC Address Anonymization and Tor.

The resulting progression was:

Physical Network Identity → MAC Anonymization → Tor Routing → Tor Bridge

Each layer addressed a different form of network exposure.

The principal conclusion of this phase is:

Tails cannot change the physical identity of the ASUS X55C, but it can substantially change how that identity is exposed through the network.

The network layer therefore reached a considerably stronger privacy state than the conventional networking environments examined earlier in the investigation.

PHASE 05

OPERATIONAL BEHAVIOR

Evaluating how operator behavior affects privacy, anonymity, and digital footprint

05.1 — Objective

The purpose of this phase was to evaluate how the behavior of the operator affects the anonymity and privacy of the ASUS X55C.

The previous phases focused primarily on the machine itself. Hardware, operating system, and network configuration can reduce the amount of information technically exposed by the computer, but these mechanisms cannot prevent the operator from voluntarily creating identifying links.

A user can therefore compromise an otherwise privacy-oriented environment through their own actions.

The experiment consequently moved from the technical characteristics of the machine to the **operational behavior of the person using it**.

The central relationship examined during this phase was:

Operational Discipline ↑ → Personal Footprint ↓

while also evaluating the opposite effect:

Operational Restrictions ↑ → Convenience and Freedom of Use ↓

The objective was not to eliminate every possible human behavior that could create a footprint. Instead, the objective was to progressively adopt privacy-oriented behaviors and determine which restrictions remained practical and which began to interfere with normal computer usage.

05.2 — Operational Separation

The first principle evaluated was the separation between different identities and activities.

Anonymity becomes significantly weaker when activities that are supposed to remain separate are directly associated with an identifiable identity.

The experiment therefore treated different activities as separate contexts rather than assuming that technical anonymity alone would prevent correlation.

The general principle established was:

An anonymous environment should not be treated as an extension of an identifiable personal environment.

This means that using the same accounts, identifiers, or personal information across both environments can create a direct association regardless of the privacy characteristics of the underlying system.

The practical consequence was that personal and anonymous activities were treated as separate operational contexts.

05.3 — Account Usage

Account usage was evaluated as one of the most direct methods through which the operator could reveal their identity.

Logging into a personal account immediately provides an external service with information that can associate the activity with an existing identity.

This can include:

- personal email accounts;
- social-media accounts;
- personal cloud services;
- accounts containing identifying information;
- previously established usernames;
- other accounts directly associated with the operator.

The experiment therefore established a basic operational rule:

Do not use an identifiable personal account when the objective of the session is anonymity.

This demonstrates an important limitation of technical anonymity.

A highly privacy-oriented operating system and network configuration cannot make an already authenticated personal account anonymous.

05.4 — Information Disclosure

The second major behavioral factor was the information voluntarily provided to online services.

The operator can create an identifying footprint without revealing an IP address or hardware identifier simply by providing enough personal information.

Examples include:

- real name;
- personal email;
- telephone number;
- physical location;
- school or workplace;
- personal social-media profiles;
- photographs;
- personally identifying descriptions;
- information about relationships or activities.

The experiment therefore adopted a principle of **data minimization**.

Only information necessary for the intended activity should be provided.

This reduced the amount of information available for directly associating the activity with the operator.

05.5 — Identifier Reuse

Another important source of correlation was the reuse of identifiers.

A username, email address, profile name, or other identifier that is repeatedly used across different services can create a link between otherwise separate activities.

The experiment therefore considered identifiers as potential correlation points.

The principle became:

An identifier that exists across multiple contexts can become a bridge between those contexts.

This means that technical separation alone is insufficient if the same recognizable identity is repeatedly reproduced by the operator.

The footprint is therefore not only generated by the machine.

It can also be generated by the choices made while using the machine.

05.6 — Behavioral Fingerprinting

Human behavior itself can provide information.

Even when explicit personal information is not provided, repeated patterns of behavior may make activities distinguishable.

These patterns can include:

- repeated activity times;
- writing style;
- language habits;
- recurring interests;
- predictable navigation patterns;
- repeated usernames;
- frequently visited services;
- characteristic interaction patterns.

This creates a distinction between **technical fingerprinting** and **behavioral fingerprinting**.

Technical fingerprinting attempts to distinguish the device or software environment.

Behavioral fingerprinting attempts to distinguish the operator based on recurring patterns.

The experiment therefore treated predictable behavior as another potential source of correlation.

05.7 — Metadata Awareness

The operator's footprint can also be created through information attached to files rather than through direct communication.

Files may contain metadata describing their origin or creation environment.

Examples include:

- document metadata;
- image metadata;
- timestamps;
- filenames;
- embedded location information;
- software information;
- device-specific metadata.

The operational approach was therefore to consider the information contained within a file separately from the visible contents of the file.

This established another important principle:

A file can reveal information about its creator even when that information is not visible in the file's primary content.

This is particularly relevant when transferring photographs, documents, or other files through an anonymous environment.

05.8 — Session Discipline

The experiment also considered the way sessions were used.

A privacy-oriented environment becomes less useful if multiple contexts are continuously mixed together.

The operator therefore treated sessions as distinct activities rather than assuming that the privacy characteristics of the environment automatically separated everything performed inside it.

The general model became:

Define the purpose of the session → perform only the relevant activity → avoid introducing unrelated identifying information.

This reduces unnecessary connections between otherwise unrelated activities.

The approach also fits the amnesic model introduced in Phase 03, where sessions are not intended to function as a permanently accumulated personal environment.

05.9 — Progressive Restrictions

The experiment did not immediately adopt every possible behavioral restriction.

Instead, restrictions were progressively introduced to determine their practical impact.

The progression can be represented as:

Normal Behavior

↓

Data Minimization

↓

Identity Separation

↓

Identifier Separation

↓

Metadata Awareness

↓

Strict Operational Separation

At each level, the operator retained more privacy-oriented discipline but sacrificed some degree of convenience.

The experiment therefore demonstrated that operational anonymity is not simply a technical configuration.

It requires continuous behavior from the operator.

05.10 — Usability Impact

The most significant finding of this phase was the effect of operational discipline on convenience.

Simple measures such as avoiding unnecessary disclosure require little effort.

More restrictive measures become progressively more difficult because normal Internet usage is built around persistent identities and convenience.

For example, avoiding personal accounts means that services requiring authentication become less convenient.

Separating identities means that previously convenient account reuse is no longer available.

Minimizing personal information makes registration and communication more restrictive.

Avoiding predictable behavior requires continuous awareness rather than a one-time configuration.

The relationship observed was therefore:

Operational Discipline	Privacy	Convenience
Normal behavior	Low	Very High
Basic data minimization	Increased	High
Identity separation	Increased	Medium–High
Identifier separation	High	Medium
Strict operational separation	Very High	Low

The important observation was that the reduction in convenience did not come from the computer itself.

It came from the **rules imposed on the operator**.

05.11 — Operational Footprint

After applying the behavioral restrictions, the concept of the system's footprint became broader.

The footprint was no longer limited to:

Hardware + Operating System + Network

It also included:

Operator Behavior

The effective anonymity of the system could therefore be represented conceptually as:

Hardware Exposure + Software Exposure + Network Exposure + Behavioral Exposure

Reducing only the first three components does not guarantee anonymity if the fourth remains uncontrolled.

This phase consequently demonstrated that operational security is an essential part of the anonymity model.

05.12 — Experimental Findings

Several conclusions were established during this phase.

First, the operator can compromise an otherwise privacy-oriented environment through identifiable accounts or voluntarily provided information.

Second, identity separation reduces the possibility of directly associating different activities.

Third, identifier reuse can create correlation between otherwise separate contexts.

Fourth, behavioral patterns can themselves become a source of identification.

Fifth, files can contain metadata capable of revealing information about their origin.

Finally, increasing operational discipline progressively reduces convenience.

The experiment therefore demonstrated that:

Anonymity is not maintained by technology alone. It also depends on how the technology is used.

05.13 — Practical Limit

The experiment established that some behavioral measures can be incorporated into normal usage without significant difficulty.

However, as the restrictions become stricter, normal activities increasingly become inconvenient.

The operator must remember more rules, maintain stronger separation between contexts, avoid previously convenient services, and deliberately consider whether individual actions could create a correlation.

At a certain point, the privacy model begins to interfere with the purpose of using a general-purpose computer.

This represents the behavioral component of the project's central trade-off:

Privacy ↑ → Behavioral Freedom ↓

The exact boundary is subjective because usability depends on the intended activity. Nevertheless, the experiment demonstrated that sufficiently strict operational anonymity requires a level of discipline that is fundamentally different from normal computer usage.

05.14 — Conclusion

Phase 05 demonstrated that the anonymity of the ASUS X55C is not determined exclusively by its hardware, operating system, or network.

The operator itself represents an additional source of identifying information.

The progression therefore became:

Technical Controls → Network Controls → Operational Controls

The most important finding was that a privacy-oriented computer can still become identifiable when the operator reconnects anonymous activity with an existing identity through accounts, identifiers, personal information, metadata, or recurring behavioral patterns.

The experiment also demonstrated the practical limit of behavioral anonymity.

Basic privacy-conscious behavior is relatively easy to maintain. However, progressively stricter operational separation requires increasingly deliberate behavior and sacrifices convenience.

Therefore:

The more aggressively anonymity is pursued, the more the operator must sacrifice the convenience and freedom normally associated with using a personal computer.

PHASE 06

TOR EXPLORATION

Evaluating the practical experience of browsing through the Tor network

06.1 — Objective

The purpose of this phase was to explore the Web through the Tor network after the privacy-oriented operating-system and network configurations established during the previous phases.

Unlike Phase 04, which focused primarily on network infrastructure, this phase focused on the **practical experience of using the Internet through Tor**.

The experiment examined conventional Web navigation, .onion services, search engines, accessibility, performance, and the differences between the conventional Web and the Tor environment.

The objective was not to search for as much content as possible.

Instead, the objective was:

How is the Internet actually experienced when accessed through an anonymity-oriented network?

06.2 — Tor Browser

The exploration was performed using **Tor Browser** within the Tails environment.

This provided a practical continuation of the previous phases.

The ASUS X55C was no longer simply using Tor as a networking mechanism. The Web browser itself was now operating as part of an environment designed around anonymous communication.

The resulting model was:

ASUS X55C → Tails → Tor → Internet

This made Tor an integral part of the browsing experience rather than simply an optional network configuration.

06.3 — Conventional Web Navigation

Initial navigation was performed through conventional Internet services.

The general browsing experience remained functional, but the connection was noticeably different from a normal direct Internet connection.

Some websites behaved normally, while others introduced additional verification mechanisms or restrictions.

The observed limitations included:

- increased loading times;
- occasional connection delays;
- CAPTCHAs;
- websites refusing connections from Tor exit nodes;
- services behaving differently depending on the exit node;
- occasional availability problems.

This demonstrated that anonymity can affect usability even when the underlying system remains functional.

The Web is generally optimized around conventional client connections, whereas Tor introduces an additional anonymity layer that some services treat differently.

06.4 — .onion Services

The experiment then moved from conventional websites to services specifically available through the Tor network.

These services use the .onion domain rather than conventional Internet domains.

The main difference was that these services are designed to be accessed within the Tor ecosystem and do not operate like conventional websites.

This demonstrated that Tor is not simply a mechanism for accessing the normal Internet anonymously.

It also provides an alternative addressing environment in which services can exist without exposing a conventional public server address.

The .onion environment therefore represents another important part of the Tor ecosystem.

06.5 — Search Engines

Different search engines were examined to understand how content discovery works within the Tor environment.

The experiment demonstrated that **not every Tor search engine serves the same purpose**.

Some focus primarily on indexing .onion services, while others attempt to provide broader or deeper indexes.

Ahmia

Ahmia was examined as an example of a search engine focused on discovering Tor hidden services.

Its purpose is relatively straightforward: provide a searchable interface for .onion services that have been indexed.

This makes it considerably more accessible for initial exploration of the Tor ecosystem.

Tor66

Tor66 was examined as an example of a broader .onion search index.

Its results demonstrate that the Tor ecosystem contains considerably more services than those commonly encountered through conventional search engines.

This also illustrated an important limitation:

The existence of an indexed .onion service does not imply that the service is trustworthy, useful, or safe.

Search engines provide discovery. They do not automatically provide validation.

06.6 — Problematic and Illegal Content

The exploration also established that some Tor search indexes can expose links to services containing material that would be illegal, harmful, or otherwise inappropriate.

This represents one of the fundamental differences between the conventional Web and parts of the Tor ecosystem.

The objective of this research was **not** to access, download, or consume illegal material.

Instead, the experiment treated these indexes as an observation of the ecosystem itself.

The relevant distinction was therefore:

A search engine can expose the existence of services without those services being legitimate or safe.

This also demonstrated why unrestricted exploration of anonymous networks is inappropriate for a controlled research experiment.

The research remained focused on understanding the architecture, navigation mechanisms, and usability characteristics of Tor.

06.7 — Availability and Reliability

The availability of Tor services varied considerably.

Unlike conventional websites, .onion services may be temporary, poorly maintained, overloaded, or completely unavailable.

As a result, a search result did not necessarily correspond to a functioning service.

The experiment encountered the practical distinction between:

Indexed → Accessible → Functional

These are not equivalent states.

A service can appear in an index while being unavailable when accessed.

This made navigation within the Tor ecosystem considerably less predictable than conventional Web browsing.

06.8 — Performance

The use of Tor also introduced a noticeable performance cost.

Connections could take longer to establish and pages could load more slowly than through a conventional direct connection.

This is an expected consequence of routing traffic through multiple relays rather than establishing a direct connection to the destination.

The performance difference became particularly noticeable when interacting with services that were already slow or poorly maintained.

The experiment therefore confirmed the trade-off already observed in Phase 04:

Greater network anonymity can introduce a performance cost.

06.9 — CAPTCHAs and Blocking

One of the most visible usability problems was the increased occurrence of CAPTCHAs and connection restrictions.

Some websites appeared to treat Tor exit-node traffic as higher-risk traffic.

This resulted in additional verification or, in some cases, complete refusal of access.

Consequently, the same website could behave differently depending on whether it was accessed through a conventional connection or through Tor.

This represented another practical consequence of anonymity:

Anonymity can reduce the trust that some services place in a connection.

The user therefore sometimes has to sacrifice convenience in exchange for reduced network attribution.

06.10 — Conventional Web vs. Tor Environment

The experiment demonstrated a clear difference between the two environments.

Characteristic	Conventional Web	Tor Environment
Connection speed	Generally higher	Generally lower
Website compatibility	High	Variable
CAPTCHAs	Less frequent	More frequent
Connection blocking	Less common	More common

.onion services	Not available	Available
Service reliability	Generally higher	Variable
Network attribution	More direct	Reduced
Privacy orientation	Low–Medium	High

The important observation was that Tor should not be understood simply as a slower version of the normal Internet.

It provides a **different networking and service environment** with different properties and limitations.

06.11 — Experimental Findings

Several conclusions were established during the exploration.

First, Tor provides access to both conventional Internet services and services specifically designed for the Tor ecosystem.

Second, .onion services provide an alternative addressing model that does not function like conventional Internet domains.

Third, search engines within Tor vary considerably in scope, indexing quality, and purpose.

Fourth, the presence of search indexes can expose services containing unsafe, illegal, or otherwise problematic material.

Fifth, Tor introduces practical usability costs, including slower connections, CAPTCHAs, blocked websites, and unreliable services.

Finally, anonymity does not automatically imply trustworthiness.

A service being accessible through Tor does not make it legitimate, safe, or reliable.

06.12 — Practical Limit

The phase demonstrated that Tor is usable for normal browsing and provides access to an additional ecosystem of .onion services.

However, the increased privacy comes with noticeable limitations.

The user must accept:

- slower browsing in many circumstances;
- additional verification;
- occasional blocked websites;
- unreliable .onion services;
- more difficult content discovery;
- greater caution when evaluating unknown services.

Therefore, Tor remains practical for privacy-oriented browsing, but it is less convenient than using the conventional Web directly.

The experiment reinforced the relationship established throughout the research:

Privacy ↑ → Convenience ↓

06.13 — Conclusion

Phase 06 demonstrated the practical consequences of operating within the Tor ecosystem.

The experiment moved beyond configuring Tor as a network mechanism and examined what it was actually like to use the resulting environment.

The main findings were that Tor provides:

Reduced network attribution

Access to .onion services

Alternative methods of content discovery

Greater privacy at the network level

while introducing:

Reduced performance

Increased CAPTCHAs

Website blocking

Variable service availability

Greater operational caution

The exploration also demonstrated that the Tor ecosystem is not inherently synonymous with illegal activity.

It contains legitimate services, privacy-oriented infrastructure, ordinary information, and communities, but also indexes and services that can expose harmful or illegal material.

The experiment therefore focused on understanding the ecosystem rather than attempting to explore its most extreme content.

The final observation of this phase was:

Tor significantly changes the way the Internet is experienced. It reduces direct network attribution, but the resulting anonymity comes with measurable costs in speed, compatibility, reliability, and convenience.

PHASE 07

FINAL CONCLUSION — PRACTICAL STATE

Evaluating the final practical state of the ASUS X55C after Phases 00–06

07.1 — Objective

Phase 07 concludes the practical portion of the experiment.

The objective is to determine the theoretical state of the ASUS X55C **after applying the measures actually explored throughout Phases 00–06**, without introducing additional modifications.

The experiment progressed from the original machine through increasingly privacy-oriented configurations:

Baseline → Reconnaissance → Hardware Analysis → Operating System → Network → Operational Behavior → Tor

The purpose of this phase is therefore not to perform another intervention, but to determine **what capabilities were retained, what capabilities were reduced, and what was sacrificed in exchange for greater privacy and anonymity**.

The central question is:

How usable is the ASUS X55C compared with its original state after the practical privacy measures explored throughout the research?

07.2 — Final Practical Configuration

At the end of the experimental progression, the ASUS X55C can theoretically be considered to be operating in the following state:

Operating System

Tails

Network

Tails Networking + MAC Address Anonymization + Tor

with Tor Bridges available as an additional measure.

Browser

Tor Browser

Operational Model

Reduced information disclosure + identity separation + identifier separation + metadata awareness + increased operational discipline

Hardware

Unmodified physical hardware

This last point is important.

The hardware interventions considered in Phase 02 remained theoretical. No physical components were removed because doing so could have damaged the ASUS X55C.

Therefore, the final practical state represents the maximum privacy-oriented configuration achieved **without physically modifying the machine**.

07.3 — Persistence

One of the largest changes occurred in the persistence model.

The original computer was intended to operate as a conventional persistent workstation. Files, applications, configuration, and user data could remain available between sessions.

The final environment instead uses Tails as an amnesic-oriented operating system.

This substantially reduces the amount of information that normally remains on the machine after a session.

The resulting progression was therefore:

Persistent workstation → Controlled persistent Linux → Privacy-oriented persistent system → Amnesic-oriented environment

This represents one of the largest reductions in local persistence achieved during the experiment.

However, this also means that the machine is less suitable for workflows that depend on permanently installed software, persistent configuration, or long-term local data.

07.4 — Network Identity

The network identity of the ASUS X55C was also significantly altered.

The physical network hardware remained unchanged, but its exposure was reduced through the network mechanisms used during the experiment.

The final practical model became approximately:

ASUS X55C → Tails → Tor → Internet

MAC Address Anonymization further reduced the exposure of the physical network interface to the local network.

Consequently, the machine retained its underlying hardware identity, but the network identity presented during normal operation was substantially different from the original configuration.

The important distinction is:

The machine's physical identity was not removed; its network exposure was reduced.

07.5 — Browser and Web Usage

The transition to Tor Browser further restricted the conventional Web experience.

The machine remained capable of accessing ordinary websites, but the browsing environment was no longer equivalent to a conventional desktop browser.

The final Web environment could involve:

- increased latency;
- CAPTCHAs;
- blocked services;
- websites that behave differently when accessed through Tor;
- variable .onion availability;
- reduced convenience;
- greater operational discipline.

At the same time, the machine gained access to .onion services and an environment specifically designed around anonymous communication.

Therefore, the Web remained usable, but the model of usage had changed substantially.

07.6 — Operational Behavior

Technical protections alone were no longer considered sufficient.

The final practical configuration assumes that the operator continues to follow the behavioral principles established in Phase 05.

This means avoiding unnecessary association between anonymous activity and an identifiable personal identity.

The operator must therefore maintain awareness of:

- account usage;
- identifier reuse;
- personal information;
- metadata;
- session separation;
- recurring behavioral patterns.

This introduces an unusual characteristic into the final state.

The privacy of the system is no longer determined exclusively by its configuration.

It also depends on **continuous human discipline**.

A technically privacy-oriented system can still become identifiable through careless operation.

07.7 — Capabilities Lost

The increase in privacy resulted in a measurable reduction in several capabilities that were available in the original environment.

Capability	Initial State	Final Practical State
Persistent environment	High	Very Low by default
Local data persistence	High	Strongly reduced
Software installation	High	More restricted
Configuration persistence	High	Strongly reduced
Conventional browsing	Normal	Restricted

Website compatibility	High	Variable
Network attribution	More direct	Strongly reduced
Connection simplicity	High	Reduced
Internet performance	Normal	Reduced in some situations
Identity separation	Low requirement	High requirement
Behavioral freedom	High	Reduced
Operational complexity	Low	High
Hardware identity	Present	Still present

The most important observation is that **the computer itself did not become incapable of performing these tasks because of its hardware**.

Many limitations were introduced deliberately by the privacy model.

07.8 — Capabilities Retained

Despite these restrictions, the ASUS X55C remained a functional computer.

It could still:

- boot an operating system;
- establish network connectivity;
- browse the Web;
- access conventional Internet services;
- access .onion services;
- run applications available within the Tails environment;
- communicate through Tor;
- perform general privacy-oriented computing tasks.

Therefore, the experiment did not reach a state where the computer was completely unusable.

Instead, it reached a state where **normal personal-computer usage became significantly more restrictive**.

This distinction is important.

The machine remained technically functional, but its intended role changed.

07.9 — Privacy vs. Usability

The progression can be summarized as follows:

Property	Original State	Final Practical State
Privacy orientation	Low	Very High
Persistence	High	Very Low by default
Network exposure	High	Strongly reduced
Convenience	Very High	Reduced
Web compatibility	Very High	Variable
Operational freedom	Very High	Reduced
Required user discipline	Low	High
Configuration complexity	Low	High
Local traces	High potential	Strongly reduced
Hardware exposure	Present	Still present

The experiment therefore did not produce a simple transformation from:

"unsafe" → "safe"

Instead, it produced a transformation from:

"Convenient persistent computer"

toward:

"Privacy-oriented temporary computing environment."

07.10 — The Practical Limit

At this point, the experiment reached an important boundary.

The ASUS X55C was still useful for privacy-oriented activities, but progressively fewer aspects of normal computer usage could be performed without compromising the operational model.

The system became less suitable for activities involving:

- persistent personal data;
- permanent application environments;
- conventional account usage;
- unrestricted Web browsing;
- maximum network performance;

- unrestricted software selection;
- casual mixing of different identities.

The machine therefore remained functional, but **its general-purpose nature had been significantly reduced**.

This establishes the practical limit reached by the experiment:

The ASUS X55C can remain usable while operating under a strong privacy model, but maintaining that model requires sacrificing persistence, convenience, compatibility, performance, and behavioral freedom.

07.11 — What Was Actually Achieved

The progression demonstrated that privacy can be increased through several independent layers.

The original machine was progressively transformed through:

Operating System

Network Configuration

Browser

Operational Behavior

The hardware remained unchanged.

This is significant because it demonstrates that a substantial increase in practical privacy does not necessarily require physically modifying the machine.

However, it also demonstrates the limitation of software-based anonymity:

The underlying hardware identity remains part of the computer regardless of which operating system is running.

Therefore, the final practical state should not be interpreted as complete or absolute anonymity.

It represents the strongest privacy-oriented state achieved **within the physical and operational constraints of the experiment**.

07.12 — Final Practical Assessment

The ASUS X55C began the experiment as a conventional personal computer optimized primarily for usability and persistence.

By the end of Phase 07, it had become a considerably more restrictive environment centered around temporary operation, reduced network attribution, and disciplined user behavior.

The transformation can be represented as:

Conventional PC

↓

User-Controlled Linux

↓

Privacy-Oriented Linux

↓

Amnesic Operating Environment

↓

Privacy-Oriented Network

↓

Tor Browser

↓

Strict Operational Behavior

The result is not an **"anonymous computer"** in an absolute sense.

It is a computer in which **many more layers of potential identification have been deliberately reduced**.

The cost was a substantial reduction in convenience and freedom.

07.13 — Conclusion

Phase 07 demonstrated the practical consequences of the entire experiment.

The ASUS X55C retained its physical identity and fundamental hardware capabilities, but its software environment, network exposure, persistence, and operational behavior were progressively transformed.

The greatest losses were not basic computing capabilities.

They were **convenience, persistence, compatibility, performance, and freedom of operation**.

The greatest gains were the reduction of persistent state, reduction of direct network attribution, stronger separation between identities, and greater control over the information voluntarily exposed by the operator.

The final practical conclusion is therefore:

The ASUS X55C remains functional, but the more strongly anonymity is enforced, the further its operation moves away from that of a normal personal computer.

This establishes the practical boundary of the experiment and provides the basis for the final theoretical analysis in **Phase 08 — Theoretical Extreme**.

PHASE 08

THEORETICAL EXTREME

Exploring the theoretical maximum of privacy and anonymity

08.1 — Objective

Phase 08 represents the final and entirely theoretical stage of the research.

Unlike the previous phases, no additional physical or software modifications are performed on the ASUS X55C. Instead, the objective is to extrapolate the results obtained throughout Phases 00–07 and determine what the machine would theoretically become if every practical privacy measure were pushed as far as reasonably possible.

The central question is:

If every unnecessary source of identification were removed or disabled, the most privacy-oriented software and network configuration were used, and the strictest operational rules were followed, how usable would the computer remain?

This phase therefore explores the theoretical boundary between:

Maximum Anonymity

and

Minimum Functionality

08.2 — Starting Point

The theoretical scenario begins with the practical state reached in Phase 07.

The machine is assumed to be operating with:

- **Tails**
- **Tor networking**
- **MAC Address Anonymization**
- **Tor Browser**
- **reduced information disclosure**
- **identity separation**
- **identifier separation**

- **metadata awareness**
- **strict operational discipline**

The difference is that Phase 08 removes the practical constraints that prevented more aggressive interventions.

The hardware restrictions considered in Phase 02 are now assumed to be implemented **theoretically**.

08.3 — Hardware Reduction

The first step would be to examine every physical component according to one question:

Does this component provide functionality that is necessary for the intended anonymous operation?

Components that are unnecessary could theoretically be removed or permanently disabled.

Potential targets would include the webcam, microphone, Bluetooth hardware, Wi-Fi hardware if an alternative network interface were used, optical drive, unnecessary peripheral interfaces, unnecessary storage devices, and other hardware not required for the intended workload.

The objective would not be to remove hardware simply because it exists.

The objective would be to reduce the number of physical components capable of providing additional information or functionality that is not required.

This would theoretically reduce the machine's attack and identification surface.

However, every removal introduces a corresponding loss of functionality.

For example:

Remove webcam → Camera functionality lost

Remove microphone → Audio-input functionality lost

Remove Wi-Fi adapter → Wireless connectivity lost

Hardware reduction therefore immediately introduces the central trade-off of this phase.

08.4 — Storage

Storage would represent another important theoretical decision.

A conventional internal storage device provides persistence. Persistence is useful, but it also creates a location where information can remain after the computer has been shut down.

In the theoretical extreme, persistent storage would therefore be minimized or eliminated wherever possible.

The idealized model would be:

Boot from external media → Operate temporarily → Shut down → Leave minimal persistent information on the machine

This would move the computer even further toward the amnesic model already introduced by Tails.

The consequence would be a substantial reduction in convenience.

The computer could no longer be treated as a conventional workstation containing a permanent personal environment.

08.5 — Operating System

The operating-system configuration would remain based on the most privacy-oriented practical model reached during the experiment:

Tails

The theoretical configuration would avoid returning to a conventional persistent operating system.

The operating system would therefore remain:

- amnesic-oriented;
- externally booted;
- privacy-focused;
- configured around Tor;
- minimally persistent.

The objective would be to ensure that the operating system itself does not unnecessarily reintroduce persistence that previous stages deliberately removed.

08.6 — Network

The theoretical network configuration would apply every appropriate privacy mechanism already investigated.

The conceptual model would become:

Network Interface → MAC Anonymization → Tails → Tor → Optional Bridge → Internet

The objective would be to minimize the amount of network information that can directly associate the machine with its physical network connection.

However, even in this theoretical configuration, the machine would still require some form of network connectivity to communicate externally.

This creates an unavoidable constraint:

A computer cannot simultaneously communicate with an external network and completely eliminate every property associated with that communication.

Network anonymity therefore remains a matter of reducing attribution rather than making communication physically nonexistent.

08.7 — Browser

The theoretical browsing environment would use:

Tor Browser

No return to conventional browsers would be necessary.

The browser would remain inside the privacy-oriented operating environment and communicate through the Tor networking architecture.

This would maximize consistency between the operating system, network, and browser layers.

However, the limitations observed in Phase 06 would remain.

Websites could still:

- block Tor connections;
- request CAPTCHAs;

- behave differently;
- become unavailable;
- operate slowly.

Increasing anonymity does not remove these fundamental compatibility limitations.

08.8 — Operational Behavior

The theoretical extreme would also require the operator to follow the strictest behavioral rules established throughout Phase 05.

The operator would theoretically have to avoid:

- personal accounts;
- reuse of identifiable usernames;
- unnecessary personal information;
- unnecessary metadata;
- mixing anonymous and identifiable contexts;
- predictable identity associations;
- unnecessary disclosure of location or personal circumstances.

Every action would therefore have to be considered in terms of whether it could create a correlation with an existing identity.

At this point, operational security would cease to be something performed occasionally.

It would become a permanent requirement.

This creates a particularly important limitation:

The computer can be configured once, but operational anonymity has to be maintained continuously.

08.9 — Theoretical Hardware State

If every theoretically unnecessary component were removed or disabled, the ASUS X55C would eventually approach a minimal computing platform.

The machine would retain only the components required for:

1. powering the system;
2. executing the operating system;
3. interacting with the user;
4. establishing the required network connection;
5. performing the intended computational tasks.

Everything else would be considered a potential reduction target.

However, this produces a fundamental paradox.

The more components are removed, the smaller the hardware exposure becomes.

But simultaneously:

The fewer components remain, the fewer things the computer can actually do.

Eventually, the distinction between **reducing the attack surface** and **destroying functionality** becomes extremely small.

08.10 — Theoretical Software State

The same principle applies to software.

A theoretically extreme installation would contain only the software required for the intended anonymous workflow.

Unnecessary services, applications, background processes, persistent configurations, network functionality, and integrations would be removed or disabled whenever possible.

This would minimize the software attack surface and reduce unnecessary information generation.

However, a completely minimal system would also lose many capabilities normally expected from a general-purpose computer.

The result would therefore no longer resemble a conventional desktop environment.

It would instead become a specialized machine designed around a very small set of tasks.

08.11 — Theoretical Operational State

At the extreme, the operator would effectively be using the computer under a strict operational protocol.

A simplified workflow would become:

Prepare isolated environment

↓

Boot temporary system

↓

Establish privacy-oriented network

↓

Use Tor Browser

↓

Perform only the intended activity

↓

Avoid identifiable information

↓

Avoid mixing identities

↓

Terminate session

↓

Discard temporary state

This workflow would minimize unnecessary exposure.

However, it would also make ordinary spontaneous computer usage significantly more difficult.

The user would have to think about privacy before performing actions that would normally require no consideration.

08.12 — Functionality Lost

At this theoretical extreme, a significant proportion of normal computer functionality would be sacrificed.

Capability	Normal Computer	Theoretical Extreme
Persistent files	Full	Minimal or absent
Persistent applications	Full	Minimal
Webcam	Available	Removed/disabled
Microphone	Available	Removed/disabled
Bluetooth	Available	Removed/disabled
Conventional browsing	Full	Restricted
Direct networking	Available	Restricted
Personal accounts	Normal	Avoided
Identity mixing	Easy	Prohibited
Software choice	Broad	Highly restricted
Network performance	Normal	Reduced
Convenience	Very High	Very Low
Operational freedom	Very High	Very Low
Privacy orientation	Low	Extremely High

The machine would technically remain operational, but its purpose would have changed.

It would no longer function primarily as a general-purpose personal computer.

It would function as a **specialized anonymous computing terminal**.

08.13 — The Paradox of Maximum Anonymity

The theoretical extreme reveals the central paradox of the entire research.

Every layer provides some additional reduction in exposure:

Remove hardware → Fewer physical surfaces

Remove persistence → Fewer local traces

Use Tails → Less persistent state

Use Tor → Reduced network attribution

Use Tor Browser → Privacy-oriented Web environment

Separate identities → Reduced correlation

Minimize information → Reduced voluntary disclosure

But every additional restriction also removes something.

Eventually the system reaches a point where the question is no longer:

“Can we make the computer more anonymous?”

but:

“Is there still enough computer left to make the system useful?”

This is the fundamental boundary investigated by the research.

08.14 — Theoretical Final State

If every measure discussed throughout the research were applied simultaneously, the ASUS X55C would theoretically exist in a state approximately represented by:

Minimal Hardware

↓

Minimal Software

↓

Amnesic Operating System

↓

MAC Anonymization

↓

Tor / Bridge

↓

Tor Browser

↓

No Persistent Identity

↓

Strict Operational Separation

↓

Minimal Information Disclosure

The resulting machine would have a very small operational footprint compared with its original state.

However, it would also have a very small **functional footprint**.

It would still be capable of performing its intended anonymous tasks, but many characteristics of a normal computer would have disappeared.

08.15 — Theoretical Usability

The final theoretical state would therefore not be described as completely unusable.

That would be technically inaccurate.

The machine could still perform a narrow range of operations.

The more accurate description would be:

Highly usable for a narrow privacy-oriented purpose, but increasingly impractical as a general-purpose computer.

This distinction is fundamental.

A system designed only to perform a small number of anonymous tasks can remain functional even after extensive restrictions.

A system expected to replace a normal personal computer cannot maintain the same level of functionality while simultaneously minimizing every possible source of identification and persistence.

08.16 — Final Theoretical Assessment

The complete research therefore produces two different endpoints.

Practical Endpoint — Phase 07

The ASUS X55C remains physically intact and operational while using:

Tails + privacy-oriented networking + Tor Browser + strict operational behavior

It remains usable, but with significant reductions in persistence, convenience, and compatibility.

Theoretical Endpoint — Phase 08

The ASUS X55C is assumed to have:

Minimal hardware + minimal software + minimal persistence + maximum network privacy + maximum operational discipline

At this point, the computer remains technically functional but becomes increasingly specialized and impractical for ordinary use.

The theoretical progression can therefore be represented as:

General-Purpose Computer → Privacy-Oriented Computer → Anonymous Workstation → Specialized Anonymous Terminal

08.17 — Final Conclusion

Phase 08 demonstrates the theoretical limit of the research.

It is possible to continue reducing potential sources of identification, persistence, and exposure, but this process does not occur without cost.

Every additional privacy measure removes or restricts some capability:

Less hardware → Less functionality

Less software → Fewer capabilities

Less persistence → Less convenience

More network anonymity → More latency and restrictions

More operational discipline → Less behavioral freedom

The theoretical extreme therefore does not produce a magical **"100% anonymous computer."**

Instead, it produces a machine with an increasingly narrow purpose.

The final conclusion of the entire research is consequently:

Absolute anonymity and unrestricted functionality are fundamentally incompatible objectives.

The ASUS X55C could theoretically be transformed into an extremely privacy-oriented computing platform, but achieving that state would require sacrificing enough hardware, persistence, software flexibility, network freedom, and operational convenience that the machine would cease to function as a normal personal computer.

The paradox can therefore be summarized as:

The closer the machine gets to eliminating every possible source of identification, the less it remains a conventional computer.

This represents the final theoretical boundary identified by the research.

PHASE 08 — THEORETICAL EXTREME

RESEARCH — COMPLETE

CONCLUSIONS

This research examined the progressive transformation of a conventional laptop (ASUS X55C) into a privacy-oriented and anonymity-focused computing platform. The investigation proceeded through eight phases, progressively introducing controls across hardware, software, network, and behavioral domains.

Principal Findings

1. Privacy is Not Binary: The research demonstrated that privacy and anonymity cannot be treated as simple binary states. Instead, they represent a spectrum of capabilities and restrictions that can be independently modified across multiple technical and behavioral layers. The same physical computer behaved substantially differently depending on configuration, yet none of the configurations achieved absolute anonymity.

2. Hardware Identity Persists: Changing the operating system, network configuration, or browser environment does not eliminate the underlying physical identity of the device. The motherboard, firmware, storage hardware, and network adapters remain identifiable regardless of software-level modifications. This establishes a fundamental limit to software-only anonymity.

3. Layered Exposure Reduction: Practical privacy improvement requires simultaneous work across multiple independent layers: hardware, operating system, network, browser, and operator behavior. Reducing exposure at one layer without addressing others produces incomplete privacy protection. Each layer represents an independent identity surface.

4. Privacy-Usability Trade-off: Progressive increases in privacy orientation consistently introduced measurable costs to system usability. These costs were not accidental side effects but fundamental consequences of the privacy model. The machine became progressively less suitable for general-purpose computing as the privacy orientation increased.

5. Operational Discipline is Essential: Technical controls cannot maintain anonymity if the operator compromises the system through careless behavior. Voluntarily disclosed information, reused identifiers, mixing of identities, and behavioral patterns can eliminate technical privacy gains. Anonymity therefore requires continuous human discipline.

6. The Practical Limit: A configuration exists at which additional privacy measures begin to eliminate more functionality than they protect. The ASUS X55C could remain usable with the practical configuration presented in Phase 07, but approaching theoretical maximum privacy would require sacrificing enough capability that the machine would cease to function as a general-purpose computer.

Theoretical vs. Practical Privacy

The research established a clear distinction between theoretical maximum privacy and practical usable privacy. The theoretical extreme discussed in Phase 08 represents the boundary where maximizing every privacy measure would produce a machine that is technically functional but increasingly specialized and impractical.

The practical configuration presented in Phase 07 represents a more realistic boundary. The ASUS X55C with Tails, MAC address anonymization, Tor networking, Tor Browser, and strict operational discipline remains functional for privacy-oriented activities while maintaining enough capability to serve as a general-purpose temporary computing environment.

The important conclusion is that the pursuit of absolute privacy represents a diminishing-return problem. Initial privacy improvements can be achieved efficiently through relatively simple technical and operational changes. Progressively stronger privacy measures introduce increasingly greater functional costs while reducing the actual improvement in privacy achieved.

Implications for Privacy Research

This research demonstrates several important implications for privacy and anonymity research more broadly:

• Hardware remains a critical identity surface that software-based interventions cannot completely eliminate.

• Anonymity cannot be engineered purely through technical means; it requires compatible human behavior.

• Privacy-oriented systems are not simple improvements to conventional systems; they represent fundamentally different operational models.

• Trade-offs between privacy and usability are unavoidable, and the boundary between acceptable and unacceptable trade-offs is subjective.

• A single device cannot simultaneously maximize both general-purpose functionality and privacy orientation.

LIMITATIONS

This research should be interpreted within the constraints of its design and scope. Several important

limitations affect the generalizability and interpretation of the results.

Scope and Generalization

This investigation examined a single physical device (ASUS X55C) and therefore does not represent every possible hardware configuration, network environment, or computing scenario. The ASUS X55C is a conventional consumer laptop from 2013, and its characteristics may not be representative of modern hardware.

The privacy model examined throughout this research does not account for threats operating outside the endpoint computer, including network-level observation, application-level leaks, cloud service behavior, or threats introduced through compromised external services. A complete privacy model would need to address these additional layers.

Furthermore, the research did not attempt to quantify the actual identifiability or anonymity of the resulting configuration. Instead, it examined the progressively reduced exposure of various identity surfaces. A true assessment of anonymity would require adversarial testing against specific threat models, which was beyond the scope of this investigation.

Hardware Preservation Constraint

The decision to preserve the ASUS X55C without physical modification affected the completeness of the practical phase. Several hardware interventions discussed theoretically in Phase 02 remain unverified experimentally. The removal of the internal Wi-Fi adapter, webcam, or microphone; the isolation or modification of storage; and other physical modifications remain theoretical rather than empirical results.

This constraint was necessary to preserve the experimental platform itself, but it means that the practical conclusion represents a software-based privacy improvement rather than a complete hardware and software redesign.

Behavioral Measurement

The behavioral analysis in Phase 05 and throughout the research relies on theoretical assessment of behavioral risk rather than systematic measurement. The actual identifiability of an operator following privacy-conscious behavior rules was not measured against practical adversaries or fingerprinting systems.

Behavioral fingerprinting is an area of active research, and the actual risk posed by specific behaviors or information-disclosure patterns may differ from the assessment provided in this research.

Tor Network Dynamics

Phase 06 examined the Tor network during a specific time period and with a particular exit-node configuration. The Tor network is dynamic, and the availability, performance, and filtering of Tor connections can change. Subsequent exploration may yield different results depending on network conditions, exit-node selection, and service-provider policies toward Tor traffic.

Threat Model Assumptions

This research did not explicitly define a threat model against which the privacy measures were evaluated. The improvements achieved are relative to conventional desktop computing but not necessarily sufficient against all potential adversaries or threat scenarios.

A state-level adversary with access to network infrastructure, application behavior, or hardware-level mechanisms may maintain the ability to identify the device or correlate activities despite the privacy measures discussed in this research.

FUTURE RESEARCH DIRECTIONS

This research identifies several areas in which future work could extend or verify the findings presented.

Experimental Hardware Modifications

The hardware interventions discussed theoretically in Phase 02 remain untested. Conducting these experiments on a separate but equivalent device would provide empirical data about the actual effect of hardware removal on system identity, functionality, and observable characteristics.

In particular, controlled experiments examining the removal of Wi-Fi adapters, webcams, and storage devices would clarify whether software-based disablement is equivalent to physical removal or whether additional identity surfaces remain accessible.

Adversarial Fingerprinting

Subjecting the privacy-oriented configurations to systematic fingerprinting attacks would quantify the actual anonymity of the resulting system. This could include browser fingerprinting, behavioral fingerprinting, application-level leaks, and network-level analysis.

Such testing would establish whether the reduced exposure actually translates to reduced identifiability

under adversarial conditions or whether alternative vectors remain capable of re-identifying the system.

Longitudinal Behavior Analysis

This research examined behavioral privacy principles but did not measure the consistency with which an operator could maintain these principles over extended periods. A longitudinal study examining the actual effectiveness of behavioral privacy discipline would provide insights into the practical sustainability of operational security.

Alternative Operating Systems

This research examined Windows, Arch Linux, Kicksecure, and Tails. Other privacy-oriented operating systems such as Whonix, Qubes OS, or others might provide different trade-offs between privacy and usability. Systematic comparison across additional platforms would provide a more complete understanding of operating-system-level privacy options.

Network-Level Privacy

This research focused on the Tor network as the primary network privacy mechanism. Additional research examining alternative anonymity networks, such as I2P, or emerging technologies might identify superior or complementary privacy mechanisms.

Furthermore, research on the effectiveness of Tor Bridges, pluggable transports, and other censorship-resistance mechanisms would clarify their practical value against different adversaries.

Multi-Device Systems

This research examined privacy from the perspective of a single device. In practice, privacy-conscious users typically interact with multiple devices, cloud services, and external infrastructure. Research examining privacy at the system level rather than the device level would provide more practical insights for users in modern computing environments.

REFERENCES

- [1] Tor Project. "Tor: Overview." https://www.torproject.org/ (accessed 2026)
- [2] The Tails Team. "Tails - Amnesic Incognito Live System." https://tails.boum.org/ (accessed 2026)
- [3] Whonix. "Whonix: Anonymous Operating System." https://www.whonix.org/ (accessed 2026)

- [4] Kicksecure. "Kicksecure: Security-Hardened Operating System." https://www.kicksecure.com/ (accessed 2026)
- [5] Dingledine, R., Mathewson, N., & Syverson, P. (2003). "Tor: The Second-Generation Onion Router." USENIX Security Symposium.
- [6] Lapsley, P. (2015). "An Introduction to Tor." Electronic Frontier Foundation.
- [7] Privacyguides.org. "Privacy Guides: Privacy & Security Resources." https://www.privacyguides.org/ (accessed 2026)
- [8] Goldschlag, D. M., Reed, M. G., & Syverson, P. F. (1999). "Hiding routing information." Information Hiding Workshop.
- [9] Newman, L. H. (2017). "How to stay anonymous online." Wired Magazine.
- [10] Pauly, T. (2018). "Browser fingerprinting: A primer." Electronic Frontier Foundation.
- [11] Qubes OS. "Qubes OS - A reasonably secure OS." https://www.qubes-os.org/ (accessed 2026)
- [12] Szydlowski, M., et al. (2012). "Blanket Execution: Dynamic similarity-based binding of android services." IEEE Security and Privacy.
- [13] Trackography Project. "Privacy and DNS." https://trackography.org/ (accessed 2026)
- [14] Bowers, A., et al. (2019). "Identifying and measuring persistent identity exposure on the Web." IEEE Security & Privacy.

APPENDIX: GLOSSARY OF TERMS

Anonymity: The state of being unidentifiable. In this research, it refers to the reduction of information available for associating activities with specific identities.

Tor (The Onion Router): A network designed to provide anonymity by routing traffic through multiple encrypted relays, making the origin and destination of traffic difficult to identify.

.onion Services: Services accessible exclusively through the Tor network, using .onion addresses rather than conventional Internet domain names.

MAC Address (Media Access Control): A hardware identifier associated with network interfaces. MAC address anonymization changes the identifier presented to local networks.

Persistent Storage: Data that remains on a computer after shutdown. A privacy concern because it can retain activity traces.

Amnesic Operating System: An operating system designed to minimize persistent local state, typically by booting from external media and discarding session data at shutdown.

Kicksecure: A security-hardened Linux operating system designed around privacy and security principles.

Tails (Amnesic Incognito Live System): A privacy-oriented operating system designed around amnesic operation and Tor networking, booted from external media.

Behavioral Fingerprinting: The process of identifying individuals based on recurring behavioral patterns rather than technical identifiers.

OPSEC (Operational Security): Practices designed to prevent sensitive information disclosure through operational behavior.

Privacy-Usability Trade-off: The general principle that increasing privacy often reduces convenience and functionality.

DMI/SMBIOS: Desktop Management Interface / System Management BIOS - firmware-level information containing hardware and system identifiers.

Exit Node: In Tor, the final relay through which traffic exits the Tor network to reach its destination.

Tor Bridge: An unlisted Tor relay designed to make Tor usage less easily observable to local networks or Internet service providers.