Security and Network Engineering

University of Amsterdam

Residential Proxies in Telecom Networks

Thies Nieborg

August 24, 2026

Supervisors: Anand Groenewegen, Marios Avgeris

Second reader: Paola Grosso

University of Amsterdam

Abstract

Residential Proxies (RESIPs) route internet traffic through IP addresses assigned to home broadband connections. While RESIPs are utilized for legitimate web operations, they are increasingly exploited by threat actors to execute large-scale web scraping, credential stuffing, and Distributed Denial of Service (DDoS) attacks. For consumers, this means their household devices are enrolled either voluntarily via bandwidth-sharing applications or involuntarily through silent malware infections, exposing their home networks to security risks. For Internet Service Providers (ISPs), these nodes operate without generating a significant amount of abuse reports, leaving operators blind to the compromise. Globally, it degrades threat intelligence, as malicious traffic blends seamlessly with ordinary consumer behavior. This thesis evaluates detection and mitigation strategies within the operational network of a Dutch ISP, KPN. By analyzing network-level behavioral signatures, such as download and upload traffic symmetry, diurnal patterns, and local DNS bypasses, this research demonstrates that active proxy nodes can be reliably identified without relying on invasive Deep Packet Inspection (DPI). However, under European net neutrality guidelines, the GDPR, and the Dutch Telecommunications Act, proactive network-wide interception and automated IP-based blocking are legally unfeasible due to privacy and proportionality constraints. This study concludes that a technically viable strategy must combine passive, retrospective flow-level analysis and localized Customer-Premises Equipment (CPE) telemetry for detection, with a mitigation approach that pairs automated Domain Name System (DNS) blocklists and targeted subscriber notifications. This hybrid framework enables ISPs to protect network integrity, fulfill their duty of care to safeguard customer security, and empower customers to remediate device compromises while respecting regulatory limits and user privacy.

Contents

CHAPTER 1

Introduction

Over the past decade, the cyber threat landscape has been reshaped by increasingly professionalized and financially motivated actors. High-impact ransomware campaigns and large-scale software supply chain attacks have demonstrated that adversaries can compromise thousands of organizations at once while remaining difficult to attribute and disrupt. In response, defenders have invested heavily in hardening perimeters and improving detection, which in turn has pushed attackers to adopt more covert infrastructures and sophisticated methods of hiding their operational footprints. Traditionally, threat actors relied on data center proxies or anonymity networks like Tor to mask their activities. However, defenders can filter data center or Tor IP address ranges to prevent web scraping. A more effective strategy for threat actors is to blend malicious traffic into ordinary consumer networks. The use of Residential Proxies (RESIPs) has increased significantly, as can be seen by the recent disruption of a major RESIP network by the Google Threat Intelligence Group [1] and operational data from Santos [2]. A RESIP, also called a covert network, is a type of proxy server that routes internet traffic through real IP addresses assigned to private households, making web requests appear as if they originate from ordinary residential locations rather than data centers.

These proxies are implemented in various ways. One of the most common methods is to incentivize users to share their bandwidth in exchange for credits [3]. This is usually done in the form of an Software Development Kit (SDK) added to a free mobile game, or free VPN app [4]. Another common method is by compromising Internet of Things (IoT) devices and using them as proxies [5]. Because these IP addresses are operationally indistinguishable from legitimate residential traffic, RESIPs provide adversaries with a highly effective mechanism for obscuring malicious activities. They enable large-scale web scraping, circumvention of geo-restrictions, covert Search Engine Optimization (SEO) reconnaissance, and resilient Distributed Denial of Service (DDoS) operations, all while blending seamlessly into ordinary consumer traffic. At the same time, such proxies are also employed for more legitimate purposes, including accessing region-locked streaming content, detecting end-to-end Domain Name System (DNS) manipulation, and validating DNSSEC's public key infrastructure [6, 7].

For a major Internet Service Provider (ISP), this is a unique challenge. The Netherlands has a high-density fiber-optic infrastructure and strong privacy regulations, which make it an attractive target for proxy providers. The impact of individual compromised devices is relatively low for the ISP. Small amounts of malicious traffic do not affect overall network performance or raise immediate concerns. However, with the increasing number of IoT devices in people's homes, each with potentially weak security, the risk of widespread abuse grows [8].

Traditional detection methods, such as active scanning by sending requests to every IP address in the public IPv4 address space, are often ineffective because infected devices typically reside behind NATs and firewalls, making them unreachable from the outside [9]. At the same time, legal and privacy constraints limit the extent to which an ISP can inspect customer traffic. As a result, it remains unclear how network operators can reliably identify and characterize RESIPs at scale in an operational broadband network.

1.1 Problem statement

ISPs receive more and more abuse reports showing that attacks originate from their IP address space [10]. From the ISP's perspective, these incidents are difficult to handle: traffic looks like customer activity, individual flows are low-volume, and legal as well as privacy constraints prevent invasive inspection or intervention [4]. Besides this, these RESIPs come and go. In research by Martins and Donnini [10], 60% of RESIP IP addresses are observed only once. At the same time, external parties expect the ISP to act on recurring abuse tied to these address ranges.

This creates an operational problem, with no widely established solution. There is no clear, scalable, privacy-preserving method to identify which customer connections are associated with malicious RESIP activity. This thesis therefore focuses on how a large ISP can use predominantly passive network data and external threat intelligence to detect and mitigate RESIPs within its own infrastructure. It develops a generalizable method and evaluates it within the operational environment of KPN, a Dutch ISP.

1.1.1 Research question

The primary objective of this research is: **how can an ISP effectively identify and mitigate RESIP nodes within its own infrastructure?**

To answer this question, the following sub-questions are explored:

1. **What network-observable behavioral signatures distinguish RESIP traffic from normal subscriber traffic in NetFlow and DNS log data?**
   A project-controlled residential connection running several RESIPs applications is analyzed. Features such as upload/download ratio, TLS Server Name Indication (SNI) distributions, and DNS patterns are examined to isolate distinguishing flow and packet features. The methodology for identifying these signatures is detailed in Section 4.4, implemented via the project-controlled test setup in Section 5.3, and the results are evaluated in Section 6.4.

2. **To what extent can cross-referencing passive network observations with external threat intelligence improve detection confidence?**
   Active client-side measurements of RESIP providers are performed to extract exit IP addresses. These are cross-referenced with commercial threat feeds and internal abuse logs to assess reporting overlap and detection reliability. The methodology is outlined in Section 4.1, implemented in Sections 5.2 and 5.4.3, and the resulting overlap and intelligence feed confidence are evaluated in Sections 6.2 and 6.4.3.

3. **What ISP-level mitigation actions are legally permissible?**
   A legal and policy analysis of net neutrality guidelines, and the Dutch Telecommunications Act (*Telecommunicatiewet*) [11] is combined with consultations of KPN's Legal and abuse handling departments to map technical interventions to regulatory limits. The legal framework is structured in Section 4.5, implementation details are provided in Section 5.5, and the permissible mitigation options are assessed in Section 6.6.

1.1.2 Contributions

This thesis makes several key contributions to the state of the art. First, a methodology is devised to capture and analyze RESIP traffic on a real customer broadband connection under research control, validating network-level signatures under realistic conditions. Second, a passive measurement framework is implemented to extract network-observable signatures, such as symmetrical upload/download ratios, TLS SNI distributions, and port activity, without relying on invasive Deep Packet Inspection (DPI). Third, the scale and overlap of proxy activity is measured by cross-referencing active exit node discovery with commercial threat intelligence and internal KPN abuse tickets, assessing the real-world visibility of these nodes. Finally, legal, regulatory, and abuse handling professionals within KPN are consulted to map technical network-layer interventions to Dutch and European regulatory boundaries, establishing a spectrum of feasible mitigation options.

1.2 Outline of the thesis

The remainder of this thesis is structured as follows. Chapter 2 reviews previous research on RESIP architectures and existing flow-based detection techniques, focusing on the absence of ISP-level mitigations. Chapter 3 provides the theoretical background on data sources such as NetFlow, DNS, and TLS, together with the regulatory constraints and operational tools required for the research. The methodology is presented in Chapter 4, which describes the pipeline for analyzing residential connections and correlating flow monitoring data with known Command and Control (C2) identifiers. Chapter 5 details the concrete setup, including the specific tools and scripts used. The core findings are presented in Chapter 6, offering an analysis of aggregated behavioral statistics and evaluating potential mitigation strategies within Dutch and European regulatory boundaries. Chapter 7 summarizes the research and proposes future work directions, and Chapter 8 discusses the ethical considerations underpinning this research.

CHAPTER 2

Related Work

The evolution of RESIPs has been researched extensively as threat actors move away from blocked data center IP addresses. This transition represents a shift toward using legitimate consumer infrastructure to obfuscate malicious activity. This obfuscation gives anonymity and agility to the threat actor. A foundational mapping of these activities was conducted by Mi et al. [12], who provided an analysis on RESIP networks and their reliance on back-connect gateways. In this infrastructure, a user connects to a central provider server, which then dynamically tunnels client traffic through different residential exit nodes. This provides agility for the user of the proxy by being able to quickly switch IP addresses. Their research identified over six million unique IP addresses associated with such services, highlighting the scale of these networks. Building on this dataset, Hanzawa and Kikuchi [13] analyzed the characteristics of RESIP hosts specifically within Japan. They observed that, in contrast to the global trend of routers and wireless access points being the primary devices, the majority of active nodes in Japan were mobile laptop PCs, with port scanning identified as the most frequent abuse activity.

The categorization of IP addresses as residential, business, or data center typically occurs using commercial IP address geolocation services, such as MaxMind. These services aggregate data from various sources, including Regional Internet Registries, routing tables, user-contributed data, and sometimes direct ISP contributions, to associate IP address blocks with geographic locations and usage types. The reliability of this data is not perfect, misclassifications can occur due to outdated information, dynamic address assignment, or intentional obfuscation by users or organizations. Komosny, Voznak, and Rehman [14] found that the accuracy of these services on a regional level was around 50%, but the errors are more frequent in smaller areas or when distinguishing between residential and non-residential allocations.

Although dedicated RESIP networks only became a major research topic around 2019, the utilization of residential IP address space for obfuscation has a longer history. Alrwais et al. [15] observed threat actors using residential infrastructure to bypass blocking in the context of bulletproof hosting services, demonstrating the long-standing value of these networks for evasion. Recent developments include the adaptation of botnets to RESIP infrastructure, as seen in the Aisuru botnet's shift from DDoS attacks to leveraging RESIPs for web scraping and other automated tasks [16]. Similarly, ISPs such as Comcast have noted a measurable impact from the abuse of residential IP address space [17]. National security organs, such as the Dutch National Cyber Security Centre (NCSC), have also warned of the growing impact of RESIPs on digital safety, noting their use in bypassing geographic blocklists and security monitoring [4].

Industry reports, such as the one from Muwandi [5], estimate that a significant proportion, around 10%, of all web traffic is now generated by bots. An increasing proportion of these bots utilize RESIPs, as traditional data center proxies become easier for websites to detect and block. These data center proxies are easier to block because their IP ranges belong to known hosting providers and are consolidated on shared blocklists, which can be blocked without the risk of collateral damage to legitimate user traffic [7]. Another direction these networks are taking is a focus on mobile proxies, where at least 4 proxy providers were found offering money for embedding a mobile proxy SDK inside Android applications [18]. In this way, users of the app can,

sometimes knowingly, allow their devices to act as proxies. These mobile proxies are particularly challenging to mitigate because cellular networks rely heavily on Carrier-Grade Network Address Translation (CGNAT). Under CGNAT, thousands of legitimate mobile subscribers share the same public IP address. Consequently, blocking an IP address associated with a mobile proxy risks causing significant collateral damage by blocking traffic for many innocent users sharing that same gateway.

Beyond voluntary integration, RESIP networks have also expanded through the large-scale compromise of consumer hardware, as can be seen with Badbox. In research presented by the Satori Threat Intelligence and Research Team [19], the Badbox malware started with a supply chain attack where hundreds of thousands off-the-shelf Android-based boxes and automotive entertainment systems were compromised before even reaching the user. These compromised devices were enrolled into a global botnet that functioned as a massive RESIP network and was taken down in July 2025. Most of these malware campaigns are not alone, as seen by Santos [2], where at least 15% of all distinct egress IP addresses in their research were also simultaneously flagged for active malware infections.

Detecting RESIP-based activity presents unique challenges. Traditional approaches, such as round-trip time (RTT) analysis as shown by Chiapponi, Dacier, and Thonnard [20], have proven unreliable, while more advanced correlation methods show promise [21]. Some simpler heuristics have been developed by Tosun et al. [22], who provided a Python script that can be run over a folder of Packet Captures (PCAPs). The adoption of JA4 TLS fingerprinting, by Khan et al. [6], and other behavioral analytics, by Huang et al. [23], have emerged as techniques for distinguishing proxy traffic from legitimate user activity. JA4 analysis is particularly useful for RESIP because, even when requests originate from seemingly consumer IP addresses, consistent TLS handshake fingerprints can reveal the shared tooling or automation frameworks behind otherwise distributed proxy traffic. Another approach, as shown by Mi et al. [12] and Kikuchi et al. [24], is to purchase access to these proxy services and register new, dedicated domain names to serve as controlled destination endpoints. By routing requests through the proxies to these custom domains, researchers can log the incoming connections on their servers and thereby identify the public IP addresses of the active exit nodes. This way, a list of active RESIP IP addresses can be constructed, enabling defenders to block proxy-relayed traffic at target web servers, or allowing ISPs to alert compromised subscribers and initiate device remediation. However, static IP address-based detection remains challenging due to the rapid rotation and shared nature of RESIP infrastructure, which leads to high IP address churn rates [10].

All of these detection methods rely on the capturing of network traffic. For example, Khan et al. [6] in collaboration with Verkleij [25], conducted this traffic collection on a separate /24 subnet, which was categorized as residential. They hosted their own RESIP on a Virtual Machine (VM) and signed up for several brokers. Up to 368GB of user data was collected over a span of 8 months. In a separate approach, Huang et al. [23] used servers located in university institutions, or Virtual Private Server (VPS) hosted by cloud computing services. At the ISP level, the technical and legal procedures for this type of data capture have been described by Branch [26], though implementation details and regulatory limitations vary widely across jurisdictions. The specific legal constraints and policies governing traffic monitoring, privacy, and data management within the Dutch regulatory framework and KPN's internal guidelines are detailed in Chapter 3.

To aid with analysis on captured traffic, a baseline of normal traffic is also required. Previous authors have worked together with a Spanish ISP to provide a dataset called UGR'16 [27]. UGR'16 is a long-term labeled network traffic dataset containing both benign flows and diverse attack scenarios, and is widely used as a benchmark in intrusion and anomaly detection research. This can be used in combination with the data from Khan et al. [6] and Mi et al. [12] to test classification methods on separate datasets. The data in UGR'16 is NetFlow data. To supplement this, the dataset from Zápotocký, Fiala, and Fesl [28] is used. They provide a user DNS fingerprint dataset, from which common domains and the number of queries per timeframe can be extracted. A baseline of traffic asymmetry can be found in the research from Brake and Bruer [29]. They argue that there is far more data for a user to consume than produce. This asymmetry can be used later in this study to indicate that RESIP traffic is anomalous.

When it comes to blocking RESIPs, sinkholing is the most common mitigation technique. It relies on disrupting the communication between the infected device/user and the C2 server [1]. By

redirecting malicious DNS queries, such as those used to discover the C2 servers, to a controlled server, a defender can effectively disrupt the RESIP node. Organizations like Shadowserver host such sinkholes, identify compromised IP addresses, and share actionable intelligence with ISPs and National Computer Security Incident Response Teams (CSIRTs) [30]. This disrupts the operations of the node, and allows ISPs to notify and help the affected end-users.

While existing academic literature explores the evolution, impact, and detection of RESIPs, it largely overlooks mitigation strategies at the ISP level, where developing practical detection and mitigation approaches without compromising user privacy remains an open challenge. Furthermore, previous research has typically relied on virtual machines within server environments or dedicated subnets to host proxy nodes, which fails to capture the realistic routing behavior and user-traffic dynamics of typical consumer connections. This thesis directly addresses these research gaps, as summarized in Table 2.1, by evaluating detection and mitigation from the operational perspective of a major ISP, KPN. First, rather than utilizing synthetic environments, a project-controlled test setup on an active consumer broadband line to capture network-observable signatures under realistic subscriber conditions is established. Second, a passive, privacy-preserving detection methodology using NetFlow, TLS SNI distributions, and DNS logs is designed, thereby avoiding invasive deep packet inspection. Finally, the gap between technical detection and operational feasibility is addressed by conducting a legal and policy analysis that maps network-layer interventions, such as reactive DNS blocking and subscriber notification, against the Dutch Telecommunications Act (*Telecommunicatiewet*) and European net neutrality guidelines to evaluate the regulatory boundaries and feasibility of different ISP-level mitigation options.

Table 2.1: Overview of existing work on RESIP networks and the remaining gap on legally feasible ISP-level mitigation

Research Theme & Studies	Abuse / impact	Detection & mitigation	Legal constraints	Residential connection
Scale & Impact of RESIPs Hanzawa and Kikuchi [13], Muwandi [5], Mi et al. [18], Satori Threat Intelligence and Research Team [19], Krebs [16], Comcast Threat Research Labs (CTRL) [17], Santos [2], and National Cyber Security Centre (NCSC) [4]	Yes	No	No	No
Detection & Threat Intelligence Chiapponi, Dacier, and Thonnard [20], Ali et al. [21], Khan et al. [6], Verkleij [25], Huang et al. [23], Google Threat Intelligence Group [1], The Shadowserver Foundation [30], and Brake and Bruer [29]	No	Yes	No	No*
Combined Analysis & Mitigation Mi et al. [12], Alrwais et al. [15], Tosun et al. [22], Kikuchi et al. [24], Martins and Donnini [10], and Khan et al. [7]	Yes	Partial / Yes	No	No
Legal & Policy Frameworks Branch [26], Kingdom of the Netherlands [11], and KPN B.V. [31, 32]	No	No	Yes	No
This work	**Yes**	**Yes**	**Yes**	**Yes**

* Utilized VM deployments or controlled subnets rather than active consumer lines.

CHAPTER 3

Background

Before describing the methodology, this chapter introduces the technical building blocks, legal constraints, and operational context required for the research.

3.1 Data

In network security monitoring, different data sources provide different levels of visibility into communication. NetFlow describes network traffic through flow metadata, including the source IP address, source port, destination IP address, destination port, and transport protocol[1]. In practice, the most modern and flexible form is IP Flow Information Export (IPFIX)[2]. NetFlow/IPFIX records are usually exported from routers, switches, or firewalls to a flow collector for analysis. When working with raw packet captures, tools such as Tranalyzer2 can be used to convert PCAP files into NetFlow-compatible flow records offline[3]. Tranalyzer2 processes the PCAP input and exports flow records with a configurable set of features, making it a good fit for generating structured flow data from captured network traffic. An example of a single NetFlow/IPFIX-style record is:

```
src_ip=192.0.2.10 dst_ip=198.51.100.5 src_port=52344 dst_port=443 \
protocol=TCP packets=12 bytes=843 \
start=2026-05-21T10:15:32Z end=2026-05-21T10:15:36Z
```

Another data source that can be used for this analysis is DNS records. DNS shows the relationship between a queried domain and the source host that made the request, making it useful for understanding which systems are attempting to reach which internet services. At the same time, DNS visibility is incomplete. Increasing use of encrypted resolvers, including DNS over HTTPS (DoH) and DNS over TLS (DoT), reduces the amount of directly observable domain lookups on the network[4]. For this reason, DNS should be interpreted together with other metadata sources rather than as a complete record of destination intent. In addition to forward DNS queries, this study also makes use of Reverse DNS (rDNS). rDNS maps an IP address back to a hostname using PTR records in the `in-addr.arpa` (IPv4) or `ip6.arpa` (IPv6) namespaces. For example, an address such as `192.0.2.1` may resolve to a name like `192-0-2-1.fixed.kpn.net.` Many ISPs encode information about access type or customer segment into these hostnames. By resolving rDNS for observed IP addresses and matching the resulting names against known patterns, an enrichment signal is obtained that helps distinguish, for instance, residential from business lines. As with forward DNS, rDNS data can be incomplete or outdated and is therefore used only as an auxiliary classification feature. A typical resolver log entry for a DNS query might look as follows:

[1] RFC3954: https://datatracker.ietf.org/doc/html/rfc3954
[2] RFC7011: https://datatracker.ietf.org/doc/html/rfc7011
[3] Tranalyzer2: https://tranalyzer.com/
[4] RFC8484 (DoH): `https://datatracker.ietf.org/doc/html/rfc8484`; RFC7858 (DoT): `https://datatracker.ietf.org/doc/html/rfc7858`

```
2026-05-21T10:15:33Z client=192.0.2.10 \
query=example.com type=A transport=UDP src_port=51324 \
resolver_ip=203.0.113.53 answer=93.184.216.34
```

A third data source is JA4 TLS, which contrasts with the others based on deeper packet inspection of connection setup rather than only flow metadata. JA4 is a fingerprinting method derived from observable properties of the TLS handshake [6]. In particular, it summarizes features such as the TLS version, the ordered list of cipher suites, advertised extensions, and other handshake characteristics into a compact identifier. The goal is not to identify a single device uniquely, but to group together connections that are generated by the same TLS implementation or software stack. This makes JA4 valuable for identifying applications, clustering similar encrypted traffic. In the context of RESIPs, JA4 TLS can help determine whether multiple connections are produced by the same proxy client, by the same embedded browser stack, or by recurring backhaul infrastructure, even when the application payload remains encrypted. An example of a JA4 TLS fingerprint and associated metadata is:

```
timestamp=2026-05-21T10:15:33Z \
src_ip=192.0.2.10 dst_ip=198.51.100.5 dst_port=443 \
sni=api.example.com \
ja4=t13d1516h2_5d7a3f1b_8da3d7f4_29a3b4c1
```

Related to this is the SNI field in the TLS handshake. SNI allows a client to indicate the hostname it wants to reach before the encrypted session is established, so that servers hosting multiple domains can present the correct certificate[5]. For measurement, this makes SNI a useful complement to DNS, since it provides a domain-level signal even when DNS lookups are hidden. Its visibility is, however, limited by the deployment of Encrypted Client Hello (ECH)[6]. When ECH is used, the real SNI is encrypted, and passive observers may only see an outer cover name, reducing the usefulness of SNI as a monitoring source.

3.2 Limitations

Network-level defenses against C2 traffic, such as inspecting user traffic or blocking suspicious IP addresses, are constrained by both privacy norms and open internet (net neutrality) rules. It is a complex landscape of Dutch and European regulations. The most important regulation for this is the principle of Net Neutrality, codified in the EU Open Internet Regulation (Regulation (EU) 2015/2120) [33]. This framework mandates that ISPs treat all traffic equally, prohibiting the blocking, throttling, or degradation of specific content, applications, or services, except under strictly defined exceptions such as maintaining network integrity or complying with legal orders.

The Body of European Regulators for Electronic Communications (BEREC) plays a central role in the European oversight of this principle. BEREC provides guidelines to ensure the consistent application of these rules across member states, and monitors how ISPs manage traffic to provide a benchmark for what constitutes "reasonable traffic management". While BEREC coordinates at the EU level, enforcement is delegated to National Regulatory Authoritys (NRAs).

In the Netherlands, this oversight is carried out by the Authority for Consumers and Markets (*Autoriteit Consument & Markt*) (ACM). The ACM is tasked with supervising compliance with both the EU Open Internet Regulation and the Dutch Telecommunications Act, the *Telecommunicatiewet* [11]. Under Dutch law, the ACM ensures that ISPs do not violate the open internet by arbitrarily blocking IP addresses or DNS names. For this research, these regulations are critical. Because RESIPs often reside within legitimate consumer network blocks, any defensive filtering or blocking strategy by an ISP may inadvertently violate net neutrality by discriminating against traffic based on its source or destination. This could potentially trigger regulatory intervention by the ACM. This is further exemplified by the architectural split in the infrastructure of these providers. As analyzed by Khan et al. [6], while the egress points, which act as exit nodes, are located on legitimate residential connections, the entry gateways and backhaul control servers that coordinate this traffic are hosted on cloud and VPS providers, such as Vultr or Digital

[5]RFC6066 (TLS Extensions, including SNI): `https://datatracker.ietf.org/doc/html/rfc6066`
[6]Specification for TLS Encrypted ClientHello: `https://datatracker.ietf.org/doc/rfc9849/`

Ocean. These gateway IP addresses are recycled rapidly, meaning that static ISP-level blocking at the gateway boundary is often reactive and targets nodes that are already decommissioned.

Currently, as a security measure, some ISPs employ an opt-in system for blocking suspicious IP addresses and domains. Under this model, users can manually configure their systems to use specific DNS servers that automatically filter hosts categorized as malicious by the provider. As an extension of this approach, ISPs can display a warning page advising against proceeding, while still allowing the user to bypass the block. Belgium, however, employs an opt-out mechanism through the Belgian Anti-Phishing Shield[7]. Under this system, DNS-level filtering is enabled by default for all subscribers, though users retain the option to disable it.

3.3 Terms of Service (ToS) of residential proxy providers

Besides technical traffic and legal analysis, it is also important to understand how RESIP providers present and regulate their own services. One relevant source for this is their ToS and related Acceptable Use Policys (AUPs). Examining these documents gives context for later technical observations since it shows whether suspicious or abusive traffic appears to contradict stated provider policy.

For this, the ToS of leading providers is examined, specifically focusing on those frequently associated with high-volume traffic: Bright Data and Oxylabs [21, 24]. These providers prohibit malicious use through their AUPs. For instance, Bright Data explicitly mandates that clients shall not: "cause any damage or service disruption to any third party computers or service; [or] distribute cracking, warez, virus, adware, worms, trojan horses, [or] malware."

Similarly, Oxylabs emphasizes an "Ethical Proxy Acquisition Framework" claiming to provide services only to clients who are investigated. However, a significant gap exists between these written policies and technical enforcement. While providers claim to employ Know Your Customer (KYC) protocols, initial observations suggest these are often bypassed via automated sign-ups or delegated to third-party resellers.

Furthermore, the suspicious nature of the ecosystem is often hidden by a split between the backhaul infrastructure and the front-end provider [23]. The entities maintaining the SDKs embedded in consumer applications (the "suppliers") are frequently separate from the brands selling the proxy access (the "providers") [12]. This compartmentalization allows providers to market a clean, corporate image while the actual sourcing of IPs remains ethically ambiguous. For example, users are often shown reassuring, but vague prompts like, "We make sure nothing illegal is happening", which hides the fact that their residential connection may be used for credential stuffing or DDoS attacks that the provider's ToS technically forbids but practically enables[8].

3.4 KPN specific background

This research is performed in collaboration with KPN, which offers several network-level security features to its customers. To contextualize the experimental setup, this section details the operational characteristics of this ISP. Although this study is conducted within KPN's network environment, the underlying methodology and findings remain generalizable to other ISPs.

KPN customers can opt in to a security feature named *KPN Veilig Browsen*[9]. This feature operates as a DNS-based blocklist: when enabled, DNS requests for domains identified as malicious, fraudulent, or otherwise harmful by integrated threat intelligence feeds are blocked before a connection is established. This provides subscribers with an additional layer of network-level protection without requiring software installation on individual client devices.

To ensure the generalizability of the measurements, the experimental setup explicitly opted out of *KPN Veilig Browsen*. This avoided introducing an ISP-specific security control that

[7] Belgian Anti-Phishing Shield: https://baps.safeonweb.be/nl/baps

[8] Screenshot showing reassuring prompts in Bright Data's Earnapp: `https://help.earnapp.com/hc/article_attachments/31351253084945`

[9] *KPN Veilig Browsen*: `https://www.kpn.com/superveilig/malwarefilter`

may not be present, or might be configured differently, at other providers. Consequently, DNS resolution reflected a standard recursive resolver baseline without additional filtering.

To enable proper research, two residential connections were acquired at a residential site. These lines were each subscribed to a basic 100 Mbit plan. From the network perspective, these lines behaved like regular KPN customer connections. This was important, because it allowed the research to be carried out in an environment that closely resembled a normal customer setup, while still remaining accessible for controlled testing. In this way, the setup helped reduce the risk that observed results were caused by differences between an institutional environment and a standard residential connection.

3.5 Tools used

The following tools were used throughout this work to collect, inspect, transform, and analyze network security data.

Wireshark[10] is a widely-used open-source network protocol analyzer that supports deep packet inspection across hundreds of protocols. Beyond interactive packet analysis, its command-line companion `tcpdump` was used to convert raw packet captures (PCAP files). Specifically, **Dumpcap**[11], Wireshark's lightweight packet-capture utility, was used for continuous, low-overhead capturing of raw network traffic at the access link.

NSD[12] is an open-source authoritative DNS server. In this work, NSD was used to provide authoritative DNS service for the experimental domain infrastructure, ensuring controlled resolution behavior during data collection.

NGINX[13] is a high-performance HTTP server and reverse proxy. In this work, NGINX was deployed on the experimental webserver infrastructure to handle inbound HTTP and HTTPS requests from the proxy networks and log client source IP addresses at the application layer.

Tranalyzer2[14] is a modular, high-performance flow generator and packet processing framework designed for both offline analysis of packet captures and online traffic monitoring. Its core functionality is extended through a plugin architecture. The following plugins were employed in this work:

- `basicFlow`: extracts fundamental per-flow identifiers and timestamps, including the 5-tuple, flow direction, start and end times, and aggregate packet and byte counts
- `dnsDecode`: parses and decodes DNS messages embedded in flow traffic, extracting query names, record types, response codes, Time to Live (TTL) values, and resolved addresses
- `findexer`: assigns persistent flow indices to facilitate cross-referencing between flow records and the underlying raw packets
- `httpSniffer`: reconstructs HTTP transactions and extracts application-layer fields including request methods, URIs, host headers, status codes, and content types
- `protoStats`: aggregates transport- and application-protocol statistics across flows, providing a protocol distribution summary
- `sslDecode`: extracts TLS/SSL handshake metadata, including negotiated cipher suites, certificate subject information, and Server Name Indication (SNI) values
- `txtSink`: serializes all processed flow records to plain-text output files for ingestion by downstream analysis.

Python 3.13[15] served as the primary analysis environment throughout this work. **Jupyter**[16] notebooks were used for interactive experimentation, prototyping, and structured presentation

[10] Wireshark: `https://www.wireshark.org`
[11] Dumpcap: `https://www.wireshark.org/docs/man-pages/dumpcap.html`
[12] NSD: `https://www.nlnetlabs.nl/projects/nsd/about/`
[13] NGINX: `https://nginx.org`
[14] Tranalyzer2: `https://tranalyzer.com`
[15] Python: `https://www.python.org`
[16] Jupyter: `https://jupyter.org`

of intermediate results. **NumPy**[17] and **pandas**[18] provided efficient numerical computation and tabular data manipulation capabilities, respectively. **Dnspython**[19] was used for programmatic DNS queries and structured parsing of DNS responses, while **requests**[20] facilitated interaction with web-based APIs and external data enrichment services. **Matplotlib**[21] was used for generating static data visualizations, including traffic volume graphs and distribution plots, while **Folium**[22] was employed to construct a geographic map representing the distribution of RESIP exit nodes. The specific versions of all Python libraries and dependencies used in this research are defined in the project's Git repository[23] via the `uv.lock` file.

[17] NumPy: `https://numpy.org`
[18] Pandas: `https://pandas.pydata.org`
[19] Dnspython: `https://www.dnspython.org`
[20] Requests: `https://requests.readthedocs.io`
[21] Matplotlib: `https://matplotlib.org`
[22] Folium: `https://python-visualization.github.io/folium/`
[23] `https://github.com/UvAThies/MScThesis`

CHAPTER 4

Methodology

In this chapter, the overall methodological pipeline used in this study is described. Figure 4.1 summarizes the main stages and their interactions. The technical pipeline starts with analyzing existing datasets to estimate the size of RESIP activity (Step 1), gathering current ground truth (Step 2), conducting a project-controlled residential connection experiment (Step 3), and deriving behavioral signatures in operational NetFlow and DNS data (Step 4). These technical steps exist in a bidirectional relationship with the overarching legal and policy framework (Step 5). The legal framework receives technical findings, for example signatures, context and in turn it provides regulatory and ethical feedback. Finally, these technical capabilities and legal boundaries output into a shared result representing permissible interventions for the ISP (Step 6). The remainder of this chapter follows the structure of the flowchart, discussing each step in order.

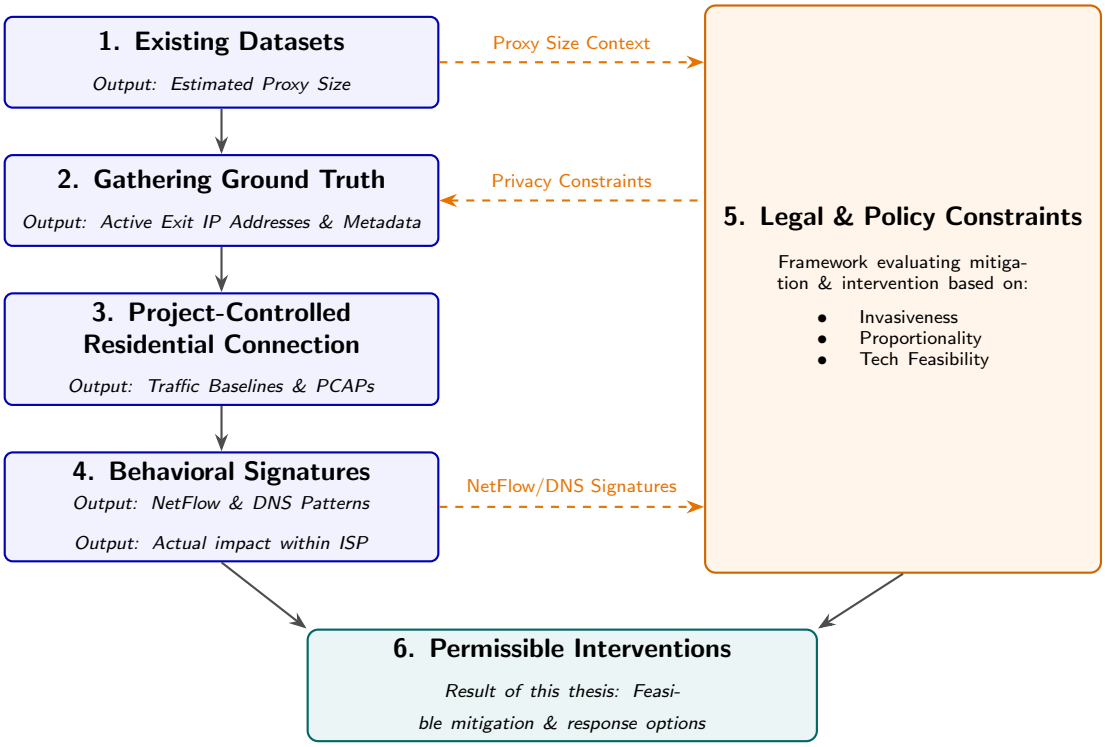

Figure 4.1: High-level flowchart of the methodological pipeline, showing the four technical stages on the left, the overarching legal and policy evaluation on the right with feedback loops constraining the technical phases, and the final set of permissible interventions at the bottom.

4.1 Analysis of existing datasets

Previous research has demonstrated that RESIPs exist at significant scale and has released sets of impacted IP addresses. For example, the work of Mi et al. [12] identifies approximately six million unique IP addresses as RESIP nodes. To approximate the size and structure of RESIP networks inside an ISP's address space, public datasets, such as the one from Mi et al. [12], are re-used and enriched with external intelligence. Common commercial intelligence platforms, such as Spur [34] and IPinfo.io [10], tag IP addresses that are associated with proxy services or suspicious behavior. Conceptually, researchers can collect these addresses and query external intelligence providers for IP addresses annotated as RESIPs. This can then be filtered and aggregated to produce an estimate of the number of customer IP addresses that are currently, or have previously been, part of a RESIP network. This provides a first look into the prevalence of RESIPs inside the network, which can later be compared to active measurements or observations.

4.2 Gathering ground truth

To continue, a set of IP addresses currently associated with RESIP activity can be gathered by utilizing the RESIP services. Instead of repeating a full market reconnaissance, the study relies directly on the selection criteria from Kikuchi et al. [24]. In addition, providers are selected that offer residential exit nodes in the Netherlands, allowing for filtering by ISP and Autonomous System Number (ASN). Configurable rotation behavior via an API or control panel is also required to obtain a new IP address for each request.

For each selected provider, automated measurements are performed to find affected customers. Requests are sent through the provider's RESIPs to a measurement service that returns the visible public IP address. The responses are aggregated over time to count unique exit IP addresses and to determine their approximate geolocation.

The measurement infrastructure is implemented as a controlled system under research administration. For this purpose, a dedicated domain is registered and its authoritative DNS configured to point to a server operated for this research. On this server, a minimal web service is hosted to record incoming requests and return the source IP address as seen by the application layer. This setup allows consistent observation of which public IP address is used as the RESIP exit node, while retaining full control over logging, timing, and request handling. In addition to the HTTP observations, the DNS and application logs generated by this infrastructure provide supporting measurement data for later correlation and analysis. This system is reused in Section 4.3.1. To provide a secondary measurement system, an external endpoint like `ifconfig.me` is used.

To gather more information about these IP addresses, a rDNS lookup is performed for each address. This can show more information about usage and locations of the IP addresses. Exact geolocation by directly matching IP addresses to individual customers is avoided for ethical and privacy reasons. Instead, a coarse location is retrieved using a geo-IP database from MaxMind [35]. The resulting dataset forms a provider-specific ground truth of residential exit IP addresses and associated metadata, such as residential or business customers.

4.3 Project-controlled residential connection experiment

To move beyond external datasets and give insight into the behavior of RESIP software in a realistic setting, a project-controlled residential connection is designed. The goal of this experiment is to observe end-to-end traffic patterns when a residential connection is turned into a proxy node and used by external clients. In addition, this setup allows testing of any mitigation techniques arising during this research, without affecting real customers.

4.3.1 Architecture

Figure 4.2 summarizes the conceptual architecture of the experiment. The goal of this architecture is to control as many of the parameters as possible. Any signatures identified can be tested

at several levels, from blocking at a customer router to blocking at the receiving end. The setup consists of three main parts:

1. **Measurement environment** (under research control), containing:
   - an **authoritative DNS server** for a dedicated domain,
   - a **web server** that responds to HTTP(S) requests and reports the observed source IP address, and
   - a **transmitter** that triggers traffic via the RESIPs towards the authoritative DNS server and web server.
2. **Residential connection**, containing:
   - a consumer **modem** terminating a residential line, and
   - an **interceptor** placed between the modem and an endpoint device that runs RESIP applications.
3. **Unknown hops**, representing the path through the RESIP provider and any intermediate infrastructure between the measurement environment and the residential site.

An important requirement is that the residential connection remain as close as possible to a typical residential setup. Therefore, the setup relies on an unmodified consumer modem with default firewall rules, ensuring that no additional devices are connected to the same line. This protects both the integrity of the experiment and the safety of other systems. To obtain full visibility of the incoming and outgoing traffic without NAT interfering, an interceptor is placed at the residential site to capture all traffic traversing the access link. Although a typical residential connection does not feature such an interceptor, its presence is not an assumption or an experimental modification that alters the network's characteristics. Instead, it serves as a transparent monitoring addition that captures traffic at the link layer. Because it does not modify IP addressing, routing, NAT translation, or the behavior of the modem and the endpoint device, the network behaves identically to a standard residential setup, while providing full measurement visibility. This access-side capture is complemented by the measurement environment, which operates in a controlled setting where all incoming DNS and web traffic can be monitored.

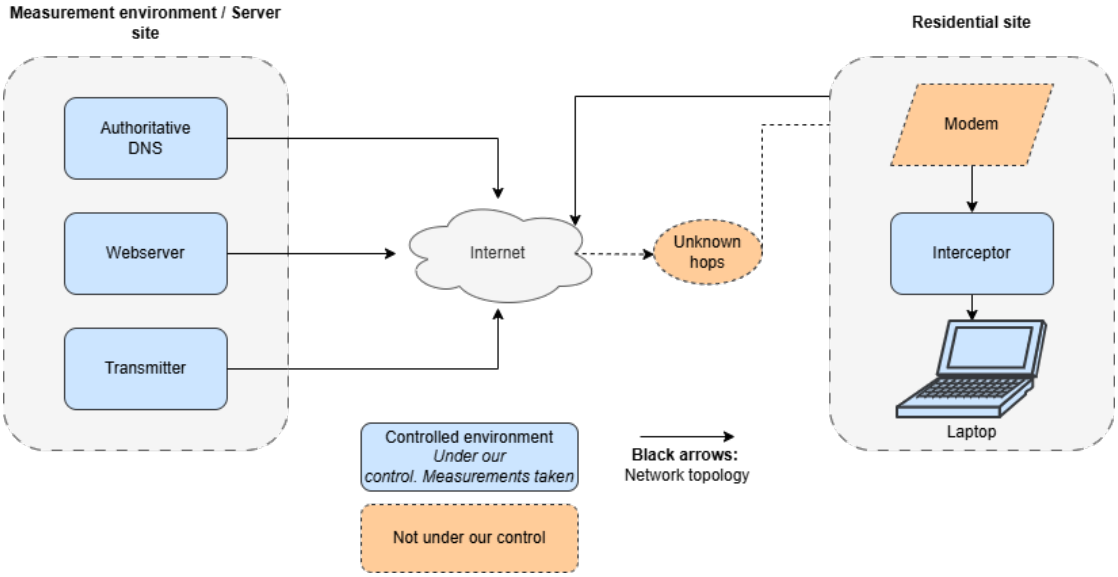

Figure 4.2: High-level overview of the controlled measurement setup. Green nodes are under research control and can be modified accordingly. Orange nodes are unknowns and uncontrollable.

The experiment consists of three steps:

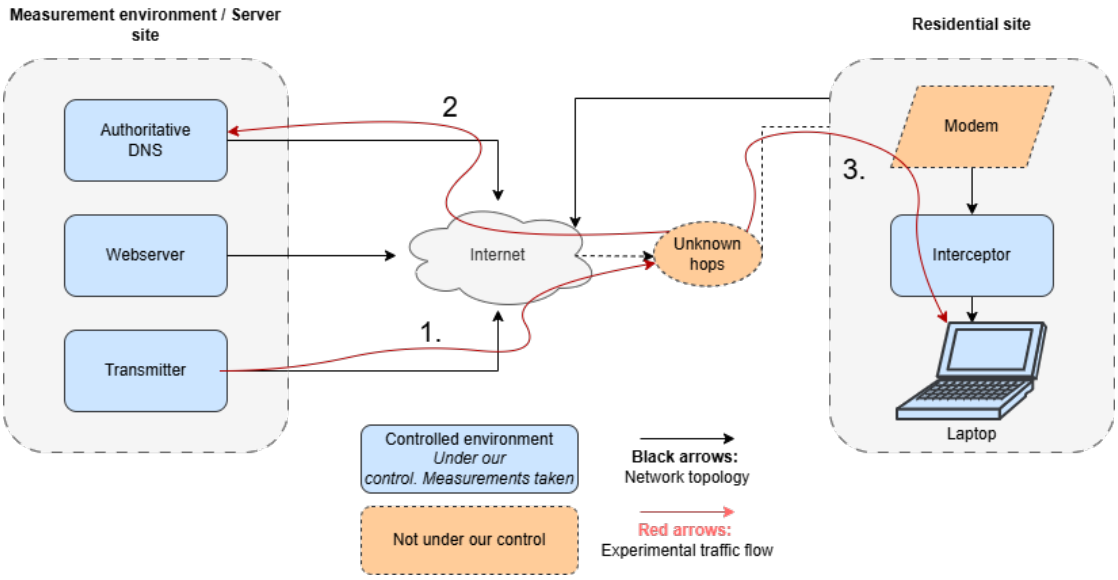

Figure 4.3: Overview of the request flow through the RESIP network, showing the three observable measurement points.

1. **Baseline capture without proxy software.** The endpoint is connected to the residential line with no RESIP applications installed or activated. Traffic is captured at the interceptor to establish a baseline profile of background traffic such as operating system updates.

2. **Install and activate proxy applications.** RESIP applications are selected using the criteria from Kikuchi et al. [24] and installed on the endpoint. The applications are configured to make the residential line available as an exit node through one or more proxy providers.

3. **Proxy usage.** From the transmitter, controlled requests are sent through the proxy providers towards the authoritative DNS server and web server, as illustrated by Step 1 in Figure 4.3. At the same time, the interceptor continues to capture all traffic on the residential line. The expectation is that some of these requests will travel through the residential line, giving a full view of the request. This gives two complementary vantage points: server-side observations at the measurement environment, including DNS queries and HTTP(S) requests originating from the residential exit node and intermediate hops (Step 2 in Figure 4.3), and access-side observations at the interceptor, capturing packets to and from the proxy backhaul and any other destinations contacted by the proxy applications (Step 3 in Figure 4.3).

For the packet-level analysis of the project-controlled residential connection, Tranalyzer2 is used on the captures collected at the interceptor. To start, the traffic is filtered to isolate only the traffic of interest. This means filtering for traffic originating from and destined to the single endpoint machine that runs the RESIP applications. The small amount of normal background traffic, such as periodic operating-system or application traffic, is deliberately kept because its volume is limited, and it provides a realistic baseline against which proxy-related behavior can be distinguished.

Within Tranalyzer2, the focus is on identifying the largest senders and receivers, as these flows are expected to reveal backhaul infrastructure. In addition, DNS traffic observed during each step of the experiment is examined, allowing a comparison of the baseline situation with the period in which proxy software is active and externally triggered. Because DNS activity is increasingly being hidden by encrypted resolvers such as DoH, it is not sufficient to solely rely on visible DNS packets. To obtain a broader view of name resolution behavior, the SNI field in TLS

handshakes is inspected. The SNI value provides an indication of the hostname a client intends to contact. By combining direct DNS measurements with SNI-based analysis, it is possible to better estimate which domains and services are contacted by the endpoint.

Specifically, the DNS infrastructure analysis evaluates two key dimensions:

- **DNS Resolution Paths:** By recording the source IP address of queries arriving at the self-hosted authoritative DNS server, the path of the request is traced. If the query originates from the ISP's recursive resolvers, the proxy client resolves domains locally using the subscriber line's default network settings. If the query originates from third-party public resolvers or proxy gateway datacenters, the DNS resolution is performed remotely.
- **Encrypted DNS Bypass:** The traffic captures at the interceptor are analyzed for connections directed to port 853 (DoT) or port 443 (DoH) targeting known public resolver lists to determine if local DNS servers are actively bypassed by encrypted application-layer queries.

Other indicators, such as DNS query timing correlation and packet-level header fingerprinting, are explicitly excluded from this analysis. Although technically feasible, these features are highly volatile and easily altered by proxy client updates, representing a fragile cat-and-mouse game between network defenders and proxy providers.

4.4 Behavioral signatures in NetFlow and DNS

With a definitive list of IP addresses acting as RESIP exits, as well as backhaul domains and control infrastructure, the focus shifts to identifying behavioral signatures that can reveal RESIP activity in network-level data. The goal is to determine which observable patterns in NetFlow and DNS logs are reliably associated with residential proxy traffic, and whether these patterns are detectable in a realistic ISP environment.

Several paths can be employed for this:

1. flow-level metrics for residential connections communicating with known RESIP backhaul ranges, such as traffic volume, directional symmetry, and connection duration,
2. DNS query patterns for domains associated with RESIP providers, including control and relay hostnames,
3. discrepancies between standard DNS resolution and TLS SNI hostnames, which can reveal centralized name resolution practices typical of proxy networks.

Where prior work has proposed DPI, this study deliberately limits itself to flow, SNI and DNS data [12, 6, 21]. This is both for legal and ethical reasons and because large-scale payload inspection is not feasible or desirable in the ISP context.

4.4.1 External intelligence

To give more context to the previous findings, these can be correlated with two additional sources.

1. **External intelligence.** Using external platforms, as in Section 4.1, to add annotations and tags indicating proxy usage or anonymity services.
2. **Internal abuse data.** Within the ISP, the customer abuse handling system records incidents and complaint tickets per customer IP address. The IP addresses identified as RESIP nodes can be linked to the number of abuse cases associated with them.

These external intelligence sources can be automated into a pipeline for continuous monitoring. For example, Spur.us publishes a commercial threat intelligence feed listing active IP addresses associated with RESIP services, including details such as the provider name and the timestamp of last activity. Similarly, customer abuse data can be programmatically retrieved via an API exposing aggregated incident reports and complaint tickets handled by the ISP's security

teams. By integrating these datasets into an automated pipeline, newly identified RESIP exit nodes can be cross-referenced against external classification feeds and internal security logs in real time.

This allows an estimation of whether RESIP participation correlates with higher rates of abuse reports and whether other parties already know about the discovered RESIPs.

4.5 Legal and policy constraints on mitigation

Finally, the technical findings must be evaluated in relation to the legal and policy frameworks. The goal is to identify which mitigation options are compatible with the law, ISP's obligations, and customer expectations.

The broader regulatory context of net neutrality, BEREC guidance, ACM supervision, and the Dutch Telecommunications Act (*Telecommunicatiewet*) [11] has already been outlined in Section 3.2. Building on that background, this part of the thesis combines a document analysis with input from professionals. First, relevant Dutch and European sources are examined in more detail, including the EU Open Internet Regulation and its BEREC guidelines, ACM policy documents and decisions, and the Dutch Telecommunications Act (*Telecommunicatiewet*) [11]. The goal of this is to identify constraints and degrees of freedom for network-layer interventions, such as monitoring, blocking, or customer notification. This analysis is complemented by consultations with legal, regulatory, and operational experts. These discussions provide practical and legal context for interpreting the regulations and assessing the feasibility of possible response options.

By combining these sources, the research develops a mapping between mitigation techniques from the technical chapters and the legal, regulatory, and organizational boundaries within which ISPs must operate.

CHAPTER 5

Implementation

To answer the research questions using the previously defined methodology in Chapter 4, a concrete measurement environment and a dedicated residential connection was established on KPN's production network. The setup was designed to produce results representative of a typical ISP environment and to capture the impact of RESIP traffic on a residential access line. Although the implementation is specific to KPN, the design and principles can also be applied at other ISPs. Most code has been made publicly available at `https://github.com/UvAThies/MScThesis`.

5.1 Analysis of existing datasets

As a historical baseline, the study used the dataset published by Mi et al. [12], who identified more than six million IP addresses active as RESIPs in 2017–2018. The original dataset consists of individual IP addresses annotated as RESIP nodes. To estimate which of these historical RESIP nodes fell within KPN's current customer address space, the Mi et al. [12] list was cross-referenced with an IP-to-ASN mapping. Specifically, a WHOIS-based IP-to-ASN dataset from db-ip.com[1] was used, filtering all entries whose origin ASN is AS1136. AS1136 is the primary customer and business ASN for KPN. This filtering step gave the subset of historically observed RESIPs that either were, or had been, routed under KPN's main customer network. The analysis for this can be found inside the associated GitHub repo, in the file *AnalyzeRPAASdata2017.ipynb*.

5.1.1 Enrichment with commercial threat intelligence

To complement the historical data with more recent observations, a commercial intelligence platform, Spur, was queried [34]. Spur annotates IP addresses and ASNs with tags indicating their association with proxy services and other suspicious or non-residential usage patterns. Other commercial IP intelligence providers, such as IPinfo.io[2], offer similar datasets but only under enterprise contracts. Access to such commercial feeds was out of scope for this research, and the analysis was therefore limited to publicly available datasets and the Spur platform.

Spur was queried for all IP addresses currently associated with AS1136, restricted to tags indicating RESIP usage. The query was executed on 21 April 2026. On that date, Spur reported 38,418 unique IP addresses within AS1136 tagged as RESIPs, of which 17,230 were attributed to the Bright Data proxy service. These figures characterized the size of the Spur-derived dataset that was used in later stages of the analysis.

The resulting IP address sets from Mi et al. [12], db-ip.com, and Spur were subsequently available for comparison with passive NetFlow and DNS observations, and for assessing the overlap between external intelligence and internal network measurements.

[1] Dataset fetched from `https://db-ip.com/db/download/ip-to-asn-lite` in April 2026.
[2] See `https://ipinfo.io/`.

5.2 Gathering ground truth

5.2.1 Provider selection

In line with the criteria in Section 4.2, several RESIP providers were evaluated and two were selected that met the requirements: Bright Data and Decodo. The aim of this study is not to provide an exhaustive overview of the global RESIP ecosystem, but rather a preliminary indication of the amount of RESIPs active on KPN's network. Selecting two representative providers keeps the scope manageable while still capturing realistic RESIP usage targeting KPN. For both providers, accounts were created and subscriptions to their RESIP offerings established. Both of these providers allowed for explicitly requesting an IP address from KPN's customer ASN 1136. This capability narrowed down the search space and provided a targeted view of the impact within KPN's network, rather than across the global RESIP population. The same approach is easily generalizable to other ISPs.

5.2.2 Measurement endpoints

To obtain a provider-specific ground-truth set of residential exit IP addresses, two types of HTTP endpoints were used:

1. **Self-hosted endpoint.** A dedicated domain was registered and its authoritative DNS configured to point to a server under research administration. On this server, a minimal web service was run that logged incoming requests and returned the source IP address as seen at the application layer. More details will follow in Section 5.3.2.
2. **External endpoint (`ifconfig.me`).** In addition, requests were sent to `ifconfig.me`, a public service that returns the caller's public IP address.

For both endpoints, requests were sent via the RESIP providers. This allowed a comparison of how the providers treated traffic to a self-hosted, research-controlled service versus a well-known external service.

5.2.3 Active probing procedure

For each provider and each measurement endpoint, automated probing was performed using a simple Python script. The script:

1. Established an HTTP connection to the provider's RESIP endpoint, such as `brd.superproxy.io` for Bright Data, authenticating with a username and password.
2. Issued an HTTP request via the proxy to either the self-hosted service or to `ifconfig.me`.
3. Parsed the response and extracted the observed exit IP address.

To control the load, the script used four concurrent workers. Each worker enforced a minimum delay of 1 second between consecutive requests, with a small random jitter added to avoid strictly periodic traffic patterns. Requests were issued in batches of 500 per provider. During each run, the set of unique exit IP addresses observed so far was maintained. The probing continued until the fraction of repeated (duplicate) IP addresses reached approximately 90% of all observed responses, indicating that additional requests yielded diminishing numbers of new exit IP addresses. At that point, the run for that provider/endpoint combination was stopped, and the collected unique IP addresses were stored. Rather than attempting to map all exit IP addresses, the objective was to capture a representative sample sufficient to map the scope of affected customers. The automatic script can be found inside *GatherProxyIPs.ipynb*

For Decodo, usable datasets were obtained from both the self-hosted endpoint and `ifconfig.me`. For Bright Data, the self-hosted endpoint yielded only a few unique exit IP addresses per run, whereas the `ifconfig.me` endpoint yielded over 2,000 unique exit IP addresses under the same configuration. For this reason, the `ifconfig.me`-based measurements were used as the primary ground truth for Bright Data, and the self-hosted measurements (supplemented by `ifconfig.me`) for Decodo.

5.2.4 Reverse DNS enrichment

To better understand which segments of KPN's customer base were affected, the collected exit IP addresses were enriched with rDNS information. For each unique IP address observed in the active measurements, a PTR lookup was performed in the corresponding `in-addr.arpa` zone to obtain the rDNS hostname.

KPN's rDNS naming conventions encode information about customer segment. By matching hostnames against known patterns, a coarse classification of each exit IP address into high-level categories such as residential, business, or other service types was obtained. This classification was later used to quantify the distribution of RESIP activity across different parts of KPN's customer base.

5.2.5 Geolocation and privacy considerations

To obtain a coarse geographical distribution of the exit nodes, each unique exit IP address was resolved using a commercial GeoIP database from MaxMind [35]. The resulting dataset provided approximate location information, such as city or region, for each observed exit IP address. The code used for this can be found in *GeoIpAnalysis.ipynb*.

5.3 Project-controlled residential connection experiment

As discussed in Chapter 2, previous research has not performed experiments on an actual customer line. This study extends this line of research by running a subset of previous experiments on a controlled KPN residential connection.

5.3.1 Baseline

To compensate for the lack of raw packet captures from ordinary subscribers, the controlled experiment was compared against an external DNS fingerprint dataset derived from real users [28]. Although this dataset does not provide raw domain lists, it includes aggregate per-day behavioral features like daily query count, daily unique domain count, detected query types, and hourly activity measures. Besides this, information for bandwidth asymmetry was gathered from the research of Brake and Bruer [29]. This could reveal anomalous bandwidth ratios and other suspicious signs in network traffic.

5.3.2 Actual setup

The controlled experiment was implemented using a dedicated residential connection and a small set of virtual machines under research administration. The setup was intentionally kept close to a normal residential environment while still allowing full packet capture at the access link and full observability at the receiving infrastructure.

The residential side, the right side of Figure 4.2, consisted of:

1. a **physical laptop** running **Windows 11 Pro**,
2. a **KPN modem** connected to a residential DSL line,
3. and a **capture VM**, also called an interceptor, placed between the modem and the endpoint to continuously record traffic going across the access link.

The laptop was configured as a realistic endpoint device:

- hibernation, sleep mode, and automatic shutdown disabled,
- RESIP applications installed and activated. EarnApp was used in the first implementation round, followed by PawnsApp in a second round. Testing multiple applications was chosen to capture the architectural diversity of routing mechanisms within the RESIP ecosystem and ensure that behavioral findings are not unique to one provider.

No other devices were connected to this residential network during the experiment. This ensured that the captured traffic could be attributed to the single endpoint under test, apart from unavoidable background traffic from the operating system and the modem. Windows 11 Pro was used, to be able to activate Remote Desktop Protocol (RDP). RDP served as a remote management solution, as the laptop was placed in a rack that was not easily accessible. As a safety precaution, the battery was removed and the laptop only ran on AC power. Each of the two implementation rounds ran for two weeks.

The measurement environment, the left side of Figure 4.2, consisted of three virtual machines:

1. **Authoritative DNS** server for the measurement domain
   - authoritative for the experiment domain running NSD as a nameserver,
   - provisioned with two public IP addresses,
   - continuously captured packets,
   - firewall restricted primarily to port 53.
2. **Webserver**
   - exposed HTTP and HTTPS on ports 80 and 443,
   - ran a minimal NGINX web service that returns the requester's visible IP address,
   - used a Let's Encrypt certificate for HTTPS on a wildcard domain,
   - continuously captured packets,
   - kept a broader inbound firewall policy to observe unexpected traffic.
3. **Transmitter**
   - executed the request generation scripts,
   - sent requests through RESIP providers,
   - did not capture packets locally,

A wildcard domain forced fresh DNS resolution for each request. Every request through the proxy could head through a new original subdomain, meaning DNS requests had to reach the authoritative server. All systems involved in packet capture were provisioned with at least 300GB of storage to allow for multiple days of rolling packet traces. Packet capture was implemented with `dumpcap` as a systemd service. Using `dumpcap` rather than interactive capture tools minimized overhead and provided robust unattended operation. The capture service used rotating files with a fixed interval, allowing long-running measurements without producing a single unmanageable trace file.

```
# /etc/systemd/system/dumpcap.service
[Unit]
Description=Continuous packet capture with dumpcap
After=network-online.target

[Service]
Type=simple
ExecStart=/usr/bin/dumpcap -i eno4 \
  -w /data/capture/capture.pcapng \
  -b interval:600 \
  -b files:4320
Restart=always
RestartSec=5

[Install]
WantedBy=multi-user.target
```

This configuration rotated captures every 600 seconds and retained 4320 files in the ring buffer. On systems where only a subset of traffic was relevant, an additional capture filter was applied to reduce disk usage, especially on the DNS and web receiving infrastructure.

5.3.3 Trace selection and filtering

For analysis, a file listing all packet captures belonging to the experiment was first created.

```
/data/capture/capture_00001.pcapng
/data/capture/capture_00002.pcapng
/data/capture/capture_00003.pcapng
...
```

This file was stored as:

```
all_filter_file.txt
```

A Berkeley Packet Filter (BPF) filter was then applied to keep the residential test traffic while excluding clearly irrelevant management and link-local traffic. The test setup contained two residential lines on VLAN 97 and VLAN 98. The example below shows the filter for the second residential line.

```
vlan 98 and not (
    (port 3389 and host 192.168.98.4) or
    icmp or stp or arp or llc or icmp6 or
    src net fe80::/10 or
    src net 239.255.255.0/24 or
    src net ff02::/16 or
    src net 224.0.0.0/4 or
    dst net 224.0.0.0/4 or
    (udp and (port 67 or port 68 or port 1900))
)
```

This filter removed:

- RDP management traffic,
- ICMP,
- ARP,
- STP and LLC,
- IPv6 link-local traffic,
- multicast traffic,
- DHCP,
- SSDP.

The goal was not to eliminate all background activity, but to remove infrastructure and management protocols that would otherwise dominate the captures while not contributing to the proxy analysis.

5.3.4 Flow analysis

For packet-to-flow conversion Tranalyzer2 was used. The selected trace set and BPF filter were provided directly to Tranalyzer2.

```
tranalyzer2 -w /data/results/all/filtered \
    -R ~/results/prod/all_filter_file.txt \
    -F ~/results/prod/bpf_filter
```

This produced the standard Tranalyzer2 output files:

```
filtered_flows.txt
filtered_headers.txt
filtered_protocols.txt
```

These outputs were then analyzed using Python scripts to derive the main traffic characteristics of the RESIP endpoint, including:

- largest senders and receivers,
- long-lived connections,
- upload/download ratios,
- protocol distribution,
- TLS metadata such as SNI,
- visible DNS activity.

5.4 Behavioral signatures in NetFlow and DNS

The starting point for the behavioral analysis was the set of backhaul IP addresses and domains discovered in the project-controlled residential connection experiment (Section 5.3). Using Tranalyzer2 flows and TLS metadata from the residential captures, IP addresses that acted as upstream infrastructure for Bright Data and Decodo were identified. The role of specific endpoints was further confirmed by inspecting TLS certificates presented during the connections, including certificates for wildcard hostnames under the provider's domain, such as `*.luminati-net.com`.

The resulting backhaul lists consisted of:

- KPN subscriber IP addresses that functioned as RESIP exits (from the ground-truth measurements), and
- external IP addresses and domains that appeared as backhaul or control infrastructure for Earnapp and Pawns.app.

These lists form the basis for the behavioral analysis. From the candidate signatures outlined in the methodology, three were selected for investigation: flow-level volume and symmetry metrics, DNS and SNI resolution patterns, and correlation with external threat intelligence. These were chosen because they are directly observable from the captures without payload inspection, they target properties that differ between normal residential usage and proxy operation, and they remain robust against minor client-side changes. They are not an exhaustive set of possible signatures. Other indicators, such as connection duration distributions or port usage patterns, may prove useful in future work.

5.4.1 Flow data sources

A primary objective of the methodology was to evaluate behavioral signatures in KPN's production environment. Accessing live NetFlow dashboards and production flow collectors is subject to strict regulatory clearance and internal access controls. Due to these restrictions, administrative clearance for direct access to KPN's production NetFlow dashboards could not be finalized within the project's timeframe. This constraint was addressed by implementing a dedicated flow pipeline.

Instead of querying live production network dashboards, all flow-level analysis was performed by capturing raw packet traffic on the project-controlled residential connection, as detailed in Section 5.3. By processing the captured PCAPs using Tranalyzer2, NetFlow-compatible flow records were generated. This mirrors production NetFlow data, and consequently, the behavioral signatures developed and verified in this study are fundamentally equivalent to live network logs. Specifically, the flow-level analysis in this thesis was based on: flows derived from the project-controlled residential connection using Tranalyzer2, as described in Section 5.3.

From the residential connection captures, features compatible with what a production NetFlow/IPFIX system would expose were extracted, including:

- total number of flows per endpoint,

- volume of uploaded and downloaded bytes, and their ratio,
- share of TLS traffic versus non-encrypted protocols,
- presence and frequency of long-lived connections.

5.4.2 DNS and SNI patterns

Due to ethical and privacy constraints, no customer-level DNS logs from KPN's production resolvers were used in this study. Instead, DNS-related behavioral indicators were derived from the packet captures of the project-controlled residential connection.

From these PCAPs, the following were extracted:

- the total number of DNS queries originating from the residential endpoint,
- the number of unique fully qualified domain names (FQDNs) queried,
- the set and count of domains related to RESIP providers, such as control domains and backhaul hostnames,
- the number of TLS connections with a visible SNI hostname and the number of unique SNI values.

Where DNS queries were encrypted or not visible, the SNI field from TLS handshakes was used as a proxy for destination. By comparing the DNS and SNI patterns between the baseline and proxy-active periods, changes in name resolution behavior associated with RESIP activity could be identified.

To track **DNS Resolution Paths**, the authoritative DNS server running NSD was configured to also capture PCAPs. These PCAPs were then analyzed using Tranalyzer2, to extract source IP addresses of DNS queries. These source IPs were then analyzed using WHOIS and rDNS PTR lookups. This allowed distinguishing between queries forwarded by KPN's recursive resolvers and queries coming from other sources.

To analyze **Encrypted DNS Bypass**, flow logs and packet captures at the interceptor were parsed for TCP/UDP traffic destined for port 53 (standard DNS), port 853 (DoT), and port 443 (DoH). For DoH detection, traffic to port 443 was filtered against a curated list of public DoH IP addresses, and the corresponding TLS SNI fields, such as `cloudflare-dns.com` and `dns.google`, were extracted to identify active encrypted queries.

5.4.3 External intelligence and internal abuse correlation

To put the observed RESIP nodes into a broader context, these were correlated with internal abuse information. For each subscriber IP address in the ground truth and backhaul sets, the number of abuse tickets recorded for that address in the available observation period was queried. Only aggregate counts per IP address were used. No incident content or customer-identifying information was accessed. This gave, for each IP address:

- whether it had any associated abuse history, and
- how many abuse incidents had been recorded.

These counts were later used to explore whether participation in RESIP networks correlated with higher rates of abuse reports and to what extent discovered RESIP addresses overlapped with IP addresses already known to be problematic. In addition to the bulk dataset correlation, the project-controlled residential connection's public IP address was queried against the Spur.us API. This tracking allowed monitoring whether a newly activated proxy node is detected and tagged by commercial threat intelligence feeds, providing a direct measurement of the classification accuracy of such platforms in practice.

5.5 Legal and policy constraints on mitigation

The legal and policy analysis started with assembling a collection of documents that captured all constraints, internal and external, on RESIP mitigation. This included the EU Open Internet Regulation and its BEREC implementation guidelines [33, 36], ACM guidance and policy documents on traffic management and open internet supervision in the Netherlands [37], relevant provisions of the Dutch Telecommunications Act (*Telecommunicatiewet*) [11], ENISA recommendations on network and information security for electronic communications [38], and KPN's own security and privacy policies, in particular the KPN Security Policy (KSP) [32].

These texts were read with a narrow focus to look for provisions that explicitly permitted, restricted, or conditioned network-level actions such as monitoring, rate limiting, blocking, or contacting customers. For each of these samples, it was noted which types of data processing were allowed, under what circumstances interventions were considered proportionate, and when prior authorization (for example, a legal order) was required.

The document analysis was then complemented with input from the Abuse and Legal teams within KPN. These conversations provided operational, legal, and regulatory context for interpreting the findings and assessing possible response options.

Based on the combination of textual analysis and these meetings, each mitigation measure was put on a spectrum from "low-impact and clearly permissible" to "high-impact and legally sensitive". The result was not a formal legal opinion, but an overview of which technical options were likely to be acceptable in practice, and under which conditions they could be deployed.

CHAPTER 6

Results

This chapter presents the results obtained from the implementation and measurement setup. It begins with an analysis of existing external datasets to establish historical and current evidence of RESIP activity within KPN's network. These observations were then compared with the ground-truth data collected through active measurements, providing a foundation for the analyses in the remainder of the chapter.

6.1 Analysis of existing datasets

6.1.1 Historical residential proxy observations

Using the dataset from Mi et al. [12], **12,037** unique IP addresses were identified that were both listed as RESIPs in 2017–2018 and currently mapped to KPN's customer ASN 1136. These addresses gave historical evidence of KPN subscriber lines that had at some point been used as RESIP nodes.

6.1.2 Current residential proxy observations from Spur

To obtain a more recent view, the Spur platform was queried for all IP addresses within ASN 1136 that are tagged as RESIPs. On 21 April 2026, Spur reported **38,418** unique IP addresses in AS1136 annotated as proxy endpoints. Of these, **17,230** addresses (approximately 45%) were attributed specifically to the Bright Data RESIP service.

Compared to the historical list derived from Mi et al. [12], the Spur snapshot thus contained roughly three times as many KPN IP addresses currently observed as RESIP nodes. These two datasets together provided a lower bound on the size of RESIPs within KPN's network and served as a reference point for the other measurements presented in the following sections.

6.2 Gathering ground truth

To evaluate proxy visibility, a ground-truth dataset of exit IP addresses was established. These addresses were harvested by requesting proxy tunnels specifically routed through KPN's customer address space, AS1136.

6.2.1 Dataset sizes

Using the probing procedure described in Section 5.2, four datasets were obtained, one per provider and endpoint combination. Across all datasets combined, 4 093 IP address occurrences were observed, corresponding to 3 571 unique IP addresses in total. This difference indicated that some addresses appeared in more than one dataset. Their sizes were:

- **Bright Data → self-hosted infrastructure:** 18 unique exit IP addresses.

- **Bright Data → external "ifconfig" infrastructure:** 2 752 unique exit IP addresses.
- **Decodo → self-hosted infrastructure:** 744 unique exit IP addresses.
- **Decodo → external "ifconfig" infrastructure:** 579 unique exit IP addresses.

6.2.2 Reverse DNS distributions

Figure 6.1 illustrates the rDNS distributions for each dataset. Across all measurements, the vast majority of exit IP addresses resolved to hostnames under the `fixed.kpn.net.` domain, which corresponds to KPN's standard residential broadband connections. Only a minor fraction of hostnames fell into other categories.

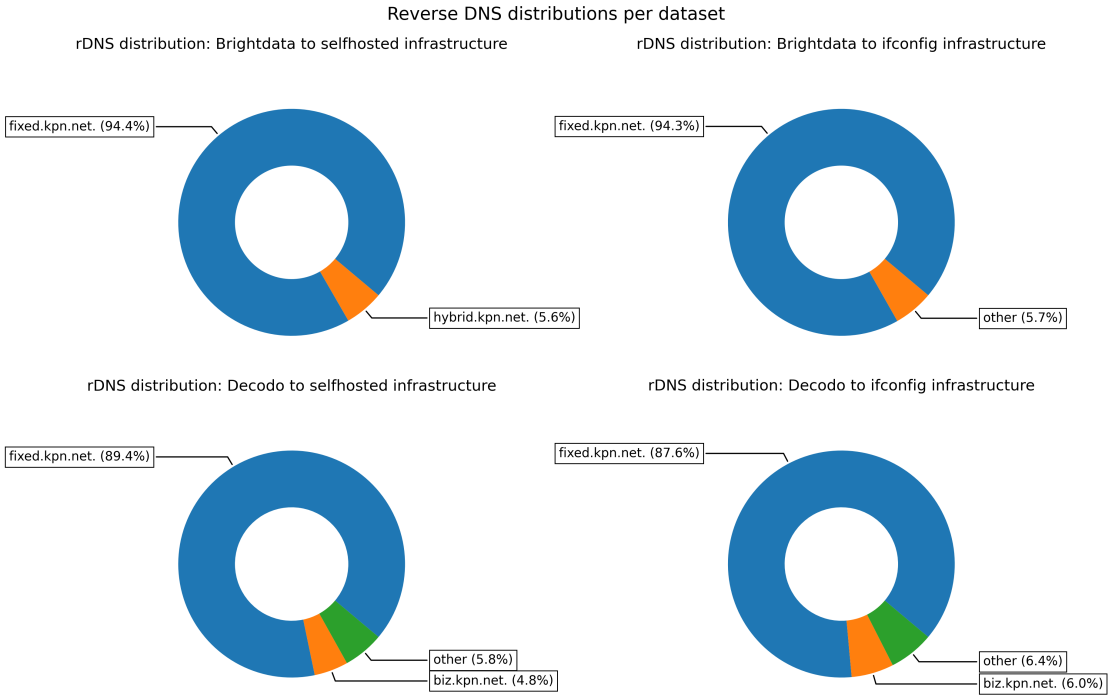

Figure 6.1: rDNS distributions per dataset for Bright Data and Decodo, split by measurement endpoint.

6.2.3 Geographic distribution

Figure 6.2 illustrates the estimated geographic distribution of the identified RESIP nodes across the Netherlands based on geolocation data. The nodes are widely distributed, closely mirroring the general population density and KPN's residential customer footprint. Major urban centers show a higher concentration of active exit nodes, whereas more rural areas exhibit a proportionally lower density. This widespread distribution indicates that RESIP activity is not localized to specific regions or local infrastructure, but rather organically dispersed among standard residential broadband subscribers.

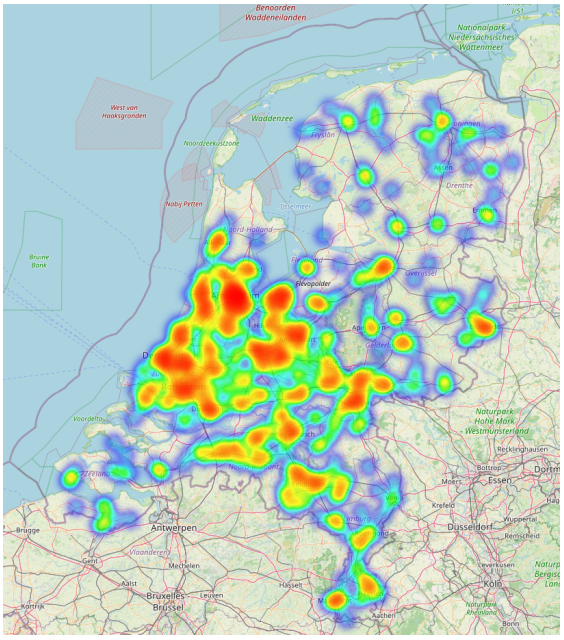

Figure 6.2: Geographic distribution of identified RESIP exit nodes across the Netherlands, based on MaxMind geolocation data.

6.2.4 Overlap between datasets

The overlap between the four datasets was limited. Out of the 4 093 total occurrences, the deduplication across datasets resulted in the aforementioned 3 571 unique IP addresses:

- 0 IP addresses occurred in both the *Bright Data → self-hosted* and *Bright Data → ifconfig* datasets.
- 24 IP addresses occurred in both the *Bright Data → ifconfig* and *Decodo → ifconfig* datasets.
- 22 IP addresses occurred in both the *Bright Data → ifconfig* and *Decodo → self-hosted* datasets.
- 494 IP addresses occurred in both the *Decodo → ifconfig* and *Decodo → self-hosted* datasets.

This limited overlap indicates that these two specific providers, Bright Data and Decodo, largely utilize distinct subsets of the residential IP pool, showing minimal overlap with each other. While it cannot be definitively claimed that the broader proxy ecosystem operates without overlap, these observations highlight a significant separation between the node pools of individual providers. For the purpose of establishing a definitive ground truth for the analyses in the remainder of this chapter, the overlapping occurrences between the active measurements were deduplicated.

As highlighted in the Implementation chapter, the Bright Data proxy service exhibited significantly different behavior depending on the target endpoint. While active measurements targeting the self-hosted endpoint yielded only 18 unique exit IP addresses, routing traffic to the external ifconfig endpoint exposed over 2,700 unique IP addresses under identical configuration parameters. This discrepancy suggests that Bright Data employs target-dependent routing logic, selectively utilizing different subsets of its proxy pool based on the destination.

6.2.5 Packet headers and TTL distributions

At the network packet level, there were no anomalies observed in the IP TTL values or the transport-layer flags during proxy activity. Because the RESIP software acts as an application-

layer relay, either Socket Secure (SOCKS) or HTTP, it terminates the client's incoming connection and initiates an entirely new TCP session from the exit node to the target destination. Consequently, the outbound IP packets from the exit node carry standard, OS-default TTL values matching the host operating system and the hops in between. For example, during the Bright Data tests, the TTL was observed to usually be between 50 and 60 (66.76% of the packets). During the Decodo tests, it was mostly 57 and 58 (69.84%). This is expected behavior, as there are multiple hops between KPN's network and the test host.

Similarly, at the application layer, the proxy client's HTTP headers, such as the User-Agent, Accept-Encoding, and connection management fields, are forwarded transparently by the proxy network as-is. Standard HTTP headers and request structures showed no modification or insertion of proxy-related telemetry. From the perspective of the destination webserver, the requests are indistinguishable from those originating directly from a standard residential client, highlighting why packet-header and protocol-level anomalies are insufficient for network-level identification.

6.3 Project-controlled residential connection experiment

6.3.1 Baseline and expected behavior

Under normal operating conditions, residential lines have asymmetric traffic patterns. Average residential download volumes are typically 14 times larger than upload volumes as shown by Brake and Bruer [29]. Normal baseline DNS activity generates approximately 14,171 queries per day, resolving roughly 590 unique domains daily as shown by Zápotocký, Fiala, and Fesl [28] in 2024.

6.3.2 Experimental setup

The controlled experiment was conducted using a dedicated, project-controlled residential connection in the KPN network, which hosted no residents or customer-generated traffic. The proxy applications, EarnApp and Pawns.app, were installed and activated sequentially on a dedicated test laptop connected to this network to generate realistic traffic patterns. Testing multiple applications was a deliberate choice to capture the diversity within the RESIP ecosystem. As demonstrated in the subsequent traffic analyses, different providers employ fundamentally distinct routing mechanisms: some proxy nodes act as direct, decentralized exit points connecting straight to target services, such as EarnApp, while others function strictly as encrypted tunnels routing all traffic back to a centralized provider relay, such as Pawns.app. By analyzing more than one application, the research ensures that the identified behavioral signatures account for this diversity and are not unique to a single provider's technical implementation.

6.3.3 Data collection and volumes

Table 6.1 summarizes the collected data volumes across the baseline, EarnApp, gap, and PawnsApp periods. The baseline period yielded 188 MB of traffic. The active EarnApp period produced 177 GB of traffic, while the PawnsApp period generated 36 GB of traffic. The gap period was used to reset the application state and provide a clear boundary.

This significant difference in overall traffic volume between the two proxy applications is expected, as EarnApp and Pawns.app have different architectures and uses. EarnApp also renders hidden advertisements in the background, increasing its traffic volume. This can be seen in Table 6.2, where Google's advertisement caches show up as Top Talkers.

Table 6.2 and Table 6.3 display the top 10 remote endpoints by L7 traffic volume for the EarnApp and Pawns.app periods, respectively.

The two tables reveal structural differences in how the two proxy networks route traffic:

- **EarnApp traffic profile:** The volume is distributed across target platforms and proxy control servers. Outbound target requests are dominated by local KPN Google cache

Source	Period	Data volume
Baseline	9–11 May 2026	188.20 MB
EarnApp	11–25 May 2026	177.15 GB
Gap	25 May–1 June 2026	94.10 MB
PawnsApp	1–17 June 2026	36.44 GB

Table 6.1: Summary of captured data volumes across baseline, active proxy, and gap periods.

Address	Description	L7 volume (GB)
195.121.114.12	KPN Google cache	25.06
64.225.66.88	Bright Data infrastructure (CN=*.luminatinet.com)	21.87
195.121.114.15	KPN Google cache	19.16
195.121.114.13	KPN Google cache	18.39
95.179.156.61	Bright Data infrastructure (rDNS: 95-179-156-61.lum-int.io. & CN=*.luminatinet.com)	15.60
195.121.114.14	KPN Google cache	13.24
195.121.114.16	KPN Google cache	12.30
79.170.100.7	Bol.com infrastructure (CN=assets.s-bol.com)	6.32
2a02:a47f:e003:12::d	Google infrastructure (CN=*.googlevideo.com)	4.06
74.125.100.138	Google infrastructure (CN=*.c.docs.google.com)	3.33

Table 6.2: Top 10 remote endpoints by application-layer traffic volume during the EarnApp activation period.

servers (totaling over 87 GB) and Google infrastructure, alongside Dutch e-commerce services like Bol.com (6.32 GB). Backhaul infrastructure endpoints owned by Bright Data (`64.225.66.88` and `95.179.156.61`) accounted for 21.87 GB and 15.60 GB respectively, respectively. This indicates the node maintained a heavy communication channel with Bright Data's central network while simultaneously functioning as an automated ad-verification and rendering client.

- **Pawns.app traffic profile:** In contrast, Pawns.app traffic is heavily centralized, with a single endpoint (`193.228.193.166`, pointing to `relay.pawns.app`) representing 19 GB. The remaining top talkers are CDNs and search engines carrying negligible traffic volumes (< 0.5 GB). This suggests that Pawns.app strictly tunnels its client traffic back through a centralized relay server rather than initiating direct, high-volume target requests from the node itself.

6.4 Behavioral signatures in NetFlow and DNS

Overall, the active proxy measurements vastly deviated from the expected baseline behavior. The perfectly symmetric data flows, as seen in Figure 6.3, observed during activation and the resolution of over 5,400 unique domains daily, as seen in Figure 6.6, drastically exceed the expected residential baseline of an asymmetric, download-heavy connection resolving roughly 590 domains per day.

Based on these project-controlled connection findings, several behavioral signatures can be defined to identify RESIP endpoints in NetFlow and DNS logs. In order to align with the implementation structure, these signatures are analyzed below across flow-level characteristics, DNS/SNI resolution patterns, and external threat intelligence correlation.

Address	Description	L7 volume (GB)
193.228.193.166	Pawns.app infrastructure (CN=relay.pawns.app)	19.17
151.101.205.237	Fastly CDN infrastructure	0.46
151.101.206.172	Fastly CDN infrastructure	0.43
151.101.37.237	Fastly CDN infrastructure	0.40
172.217.26.227	Google infrastructure	0.36
216.239.34.223	Google infrastructure	0.22
142.250.207.35	Google infrastructure	0.19
142.251.220.142	Google infrastructure	0.18
151.101.206.186	Fastly CDN infrastructure	0.17
178.156.140.194	Pawns.app infrastructure (CN=bootstrap.pawns.app)	0.16

Table 6.3: Top 10 remote endpoints by application-layer traffic volume during the Pawns.app activation period.

6.4.1 Flow-level behavioral signatures

Upload/download ratio

Figure 6.3 displays the daily traffic volume and the download/upload ratio over time. During active proxy periods, the daily traffic volume peaked at approximately 17.5 GB for EarnApp and 8.3 GB for PawnsApp. The download/upload ratio stayed consistently at 1.0, representing perfectly symmetric traffic. This symmetry is highly anomalous compared to typical residential traffic, where a ratio of 14 would be expected[29]. This 1.0 ratio was also observed during the baseline and gap periods. However, this is an artifact of the extremely low background traffic volume (282 MB total over eight days), where small periodic keep-alive flows and OS updates happen to be symmetric, rather than representing active, high-volume residential usage patterns.

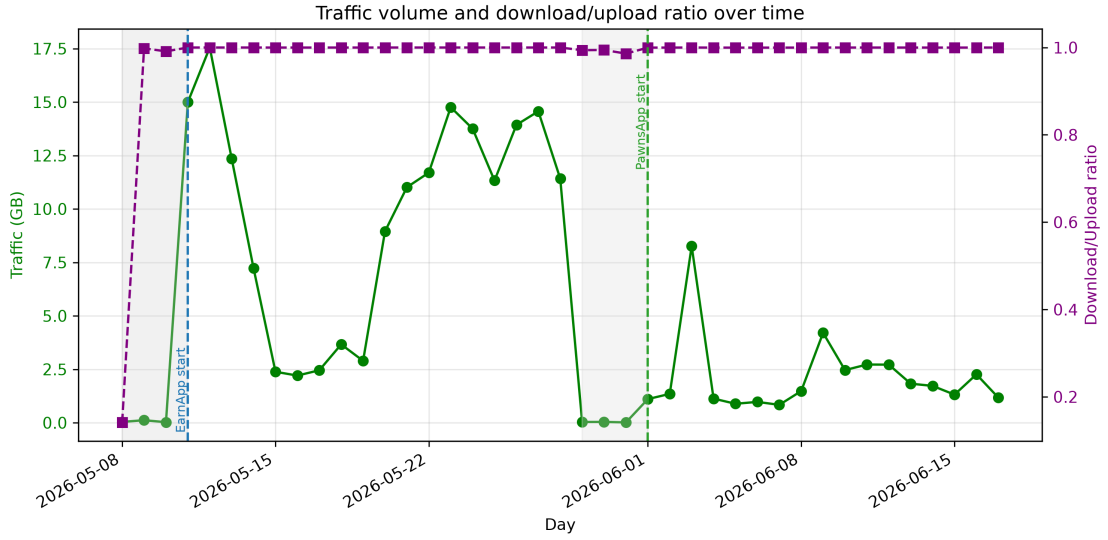

Figure 6.3: Daily traffic volume (green) and download/upload ratio (purple) over the course of the experiment.

Diurnal traffic patterns

Figures 6.4 and 6.5 show the hourly traffic distribution over the course of the day. The EarnApp traffic peaked at hour 0 UTC with 29.5 GB total, followed by minor peaks at hours 1, 2, 3, and 8. The PawnsApp traffic exhibited a single dominant peak at hour 3 UTC, reaching 8.1 GB,

with other hours remaining mostly below 2 GB. The concentration of traffic at fixed hours of the night, combined with a highly uneven hourly distribution in which PawnsApp's peak hour carries four times the volume of any other hour, is inconsistent with human behaviour. Human residential traffic typically distributes gradually across local waking hours. The repeatability of these peaks across the full measurement period suggests that the traffic is not organically driven by the residential subscriber, regardless of the specific underlying cause, be it automated scheduling or activity from proxy clients across different time zones.

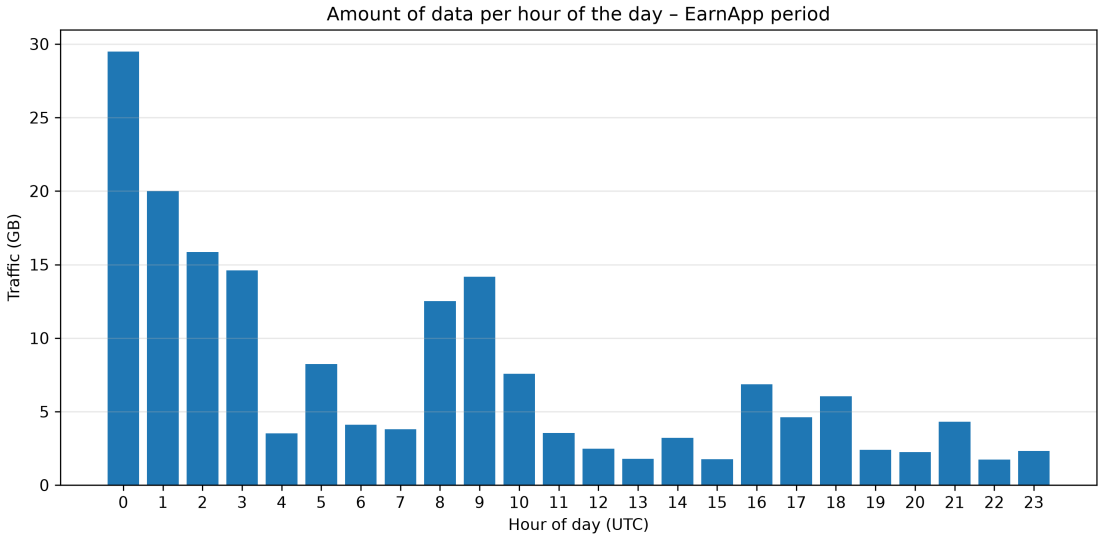

Figure 6.4: Hourly traffic volume over the day during the EarnApp activation period.

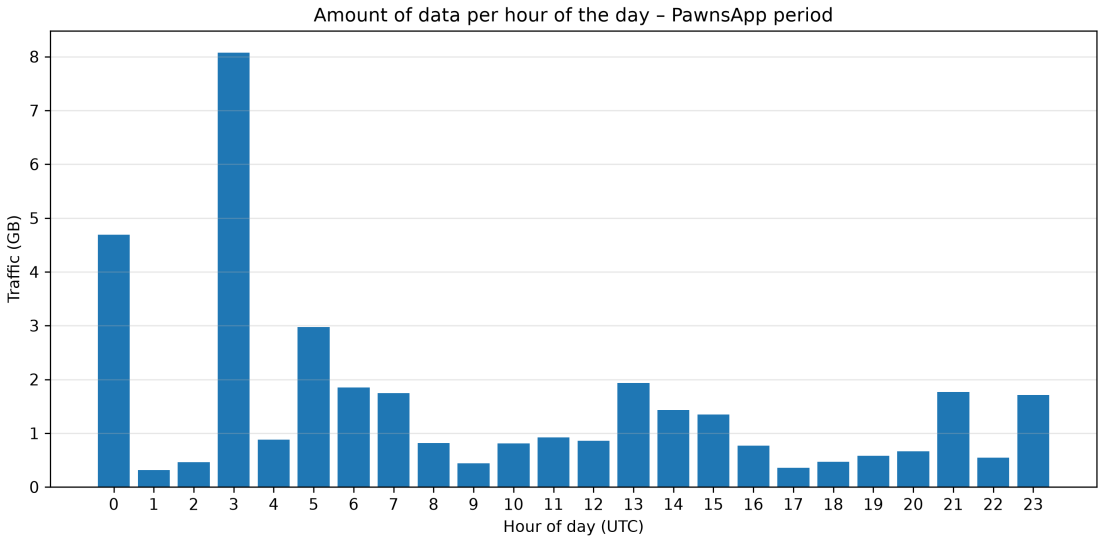

Figure 6.5: Hourly traffic volume over the day during the PawnsApp activation period.

6.4.2 DNS and SNI patterns

DNS vs TLS name resolution

Figure 6.6 compares the standard DNS query counts, unique DNS names, and unique combined DNS and SSL names over time. While standard DNS query volume rose significantly during active proxy periods, the number of unique domain names resolved via port 53 remained flat and low, staying below 100 per day. This high volume of total queries was driven almost entirely by the proxy applications repeatedly resolving their own C2 infrastructure, such as `client.earnapp.com` and `api.pawns.app`, to maintain connectivity and request new tasks. These requests were performed against the local router, and didn't use external DNS infrastructure. In contrast, the unique combined DNS and SSL names peaked at over 5,400 per day during the EarnApp activation. This discrepancy indicates that the destination domains are only visible at the TLS layer and are not resolved via local DNS. This effect was less pronounced during the Pawns.app activation due to its lower traffic volume and different routing architecture.

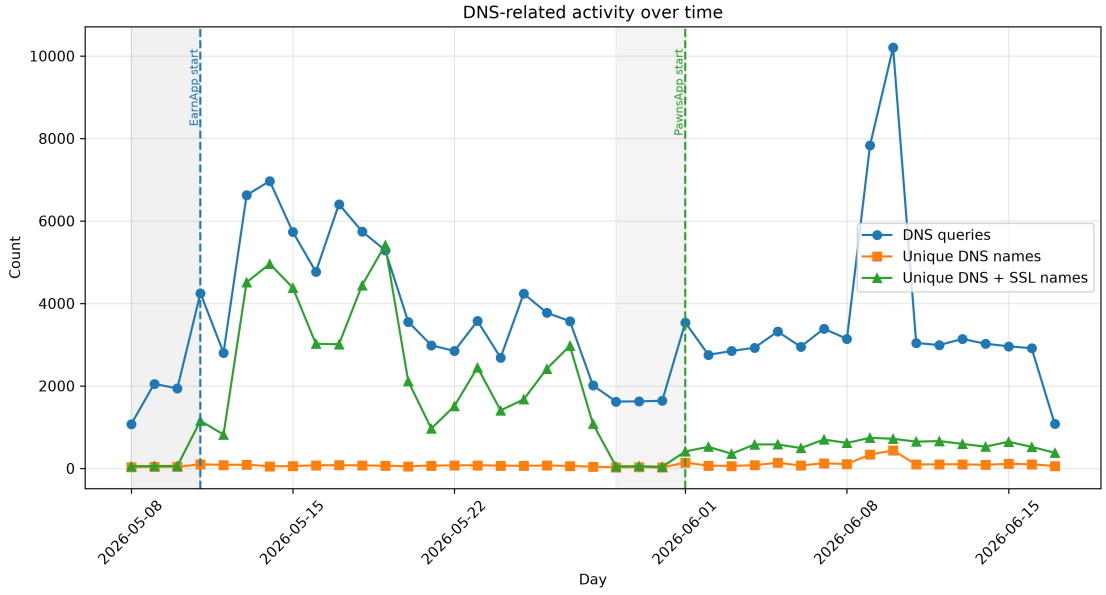

Figure 6.6: Daily DNS query counts, unique DNS domain names, and unique combined DNS and TLS SNI names over the course of the experiment.

Figure 6.7 shows the daily number of TLS flows on port 443 over time, confirming a large increase in HTTPS connections matching the proxy activation periods.

To analyze this DNS behavior in more detail, the measurement results were grouped into two key areas:

Centralized DNS resolution and IP routing

The absence of unique DNS queries corresponding to the observed SNI names, combined with a lack of identifiable DoH or DoT traffic, points to a centralized resolution architecture. Rather than relying on the residential node to resolve target domains, the proxy provider performs DNS resolution centrally. The proxy control server then dispatches tasks to the residential client containing the already-resolved destination IP address alongside the required SNI string. Consequently, the local proxy client bypasses local name resolution entirely, establishing a direct TCP connection to the provided IP address and populating the TLS Client Hello with the instructed SNI. This architecture prevents local ISP DNS filters from blocking scraping targets and conceals the proxy's specific activities from traditional DNS monitoring.

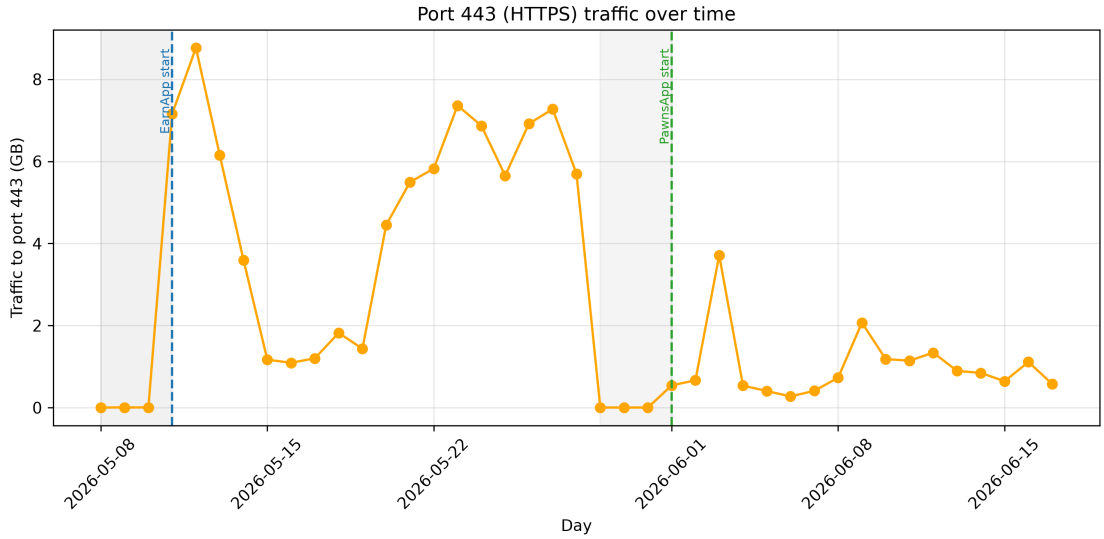

Figure 6.7: Daily count of TLS flows on port 443, showing the increase during proxy activation periods.

DNS resolution paths

To determine how DNS resolution is routed through the proxy networks, the source IP addresses of the queries arriving at the self-hosted authoritative DNS server were analyzed. Table 6.4 and Table 6.5 show the top requestors (DNS recursive resolvers) that queried the server during the Bright Data (EarnApp) and Experiment 2 (Decodo) measurement runs.

Rank	Source IP	Requests	ASN	Organization
1	92.53.92.36	1721	49505	JSC Selectel
2	92.53.92.34	787	49505	JSC Selectel
3	92.53.92.35	779	49505	JSC Selectel
4	165.232.116.76	246	14061	DigitalOcean, LLC
5	104.23.173.72	144	13335	Cloudflare, Inc.
6	141.101.75.34	118	13335	Cloudflare, Inc.
7	172.71.93.209	118	13335	Cloudflare, Inc.
8	162.159.112.71	116	13335	Cloudflare, Inc.
9	69.10.36.54	115	19318	Interserver, Inc
10	104.23.167.72	113	13335	Cloudflare, Inc.

Table 6.4: Top DNS requestors during Bright Data measurements.

Several key observations can be gathered from the requestor distributions. As expected, the actual public IP address of the residential exit nodes was never directly observed in the authoritative DNS logs. This is typical for standard DNS infrastructure, where clients resolve names by querying recursive resolvers, which then query the authoritative nameserver on their behalf, effectively masking the origin subscriber. However, no recursive resolvers owned by KPN (AS1136) appeared in the top requestor lists, which suggests that the default local ISP DNS resolvers are completely bypassed. Instead, the proxy applications rely heavily on alternative public resolvers. In both experiments, Cloudflare (AS13335) represents a significant source of DNS queries, indicating that applications or their target traffic are configured to utilize public resolvers like `1.1.1.1`. During Experiment 2 (Table 6.5), SURF B.V. (AS1103) is also visible, representing queries originating from the university network where the transmitter scripts ran to resolve subdomains before initiating the proxy tunnels. Most anomalously, the top requestors in the Bright Data run (Table 6.4) were IP addresses belonging to JSC Selectel (AS49505), a

Rank	Source IP	Requests	ASN	Organization
1	104.23.171.71	121	13335	Cloudflare, Inc.
2	104.23.167.71	115	13335	Cloudflare, Inc.
3	104.23.173.72	107	13335	Cloudflare, Inc.
4	172.71.93.209	106	13335	Cloudflare, Inc.
5	104.23.169.34	105	13335	Cloudflare, Inc.
6	162.159.112.71	104	13335	Cloudflare, Inc.
7	141.101.75.33	99	13335	Cloudflare, Inc.
8	145.100.96.11	89	1103	SURF B.V.
9	172.71.97.143	81	13335	Cloudflare, Inc.
10	172.71.97.142	76	13335	Cloudflare, Inc.

Table 6.5: Top DNS requestors during Decodo measurements.

Russian hosting and cloud provider, accounting for the vast majority of queries. The presence of a Russian infrastructure provider is unexpected and may be a routing side effect of Bright Data's global proxy overlay, where DNS resolution requests are backhauled through specific gateways located in Russia, or it could be unrelated to the research. DNS is by design public, so it is not unexpected to receive unrelated queries, but the amount is remarkable.

Destination top-level domains

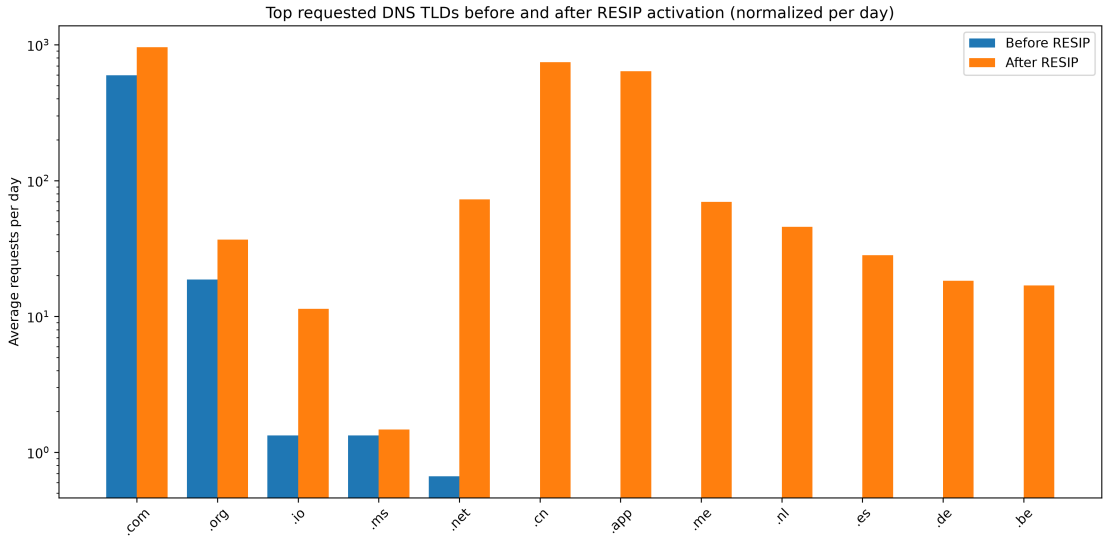

Figure 6.8: Top requested DNS TLDs before and after RESIP activation, normalized per day.

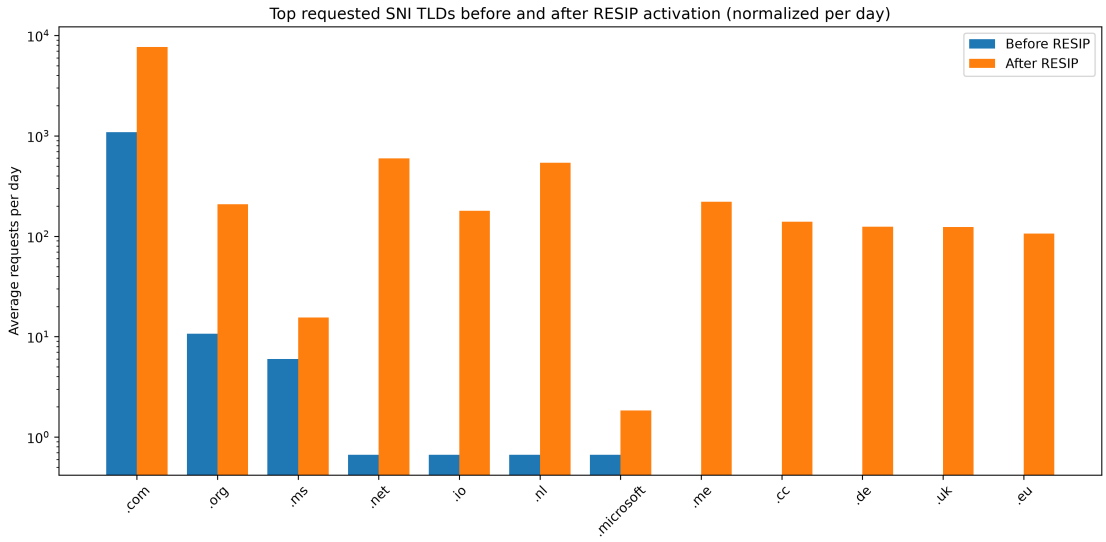

Figure 6.9: Top requested SNI TLDs before and after RESIP activation, normalized per day.

6.4.3 External Threat Intelligence

Figure 6.10 shows the distribution of abuse tickets per affected RESIP IP address across the four datasets.

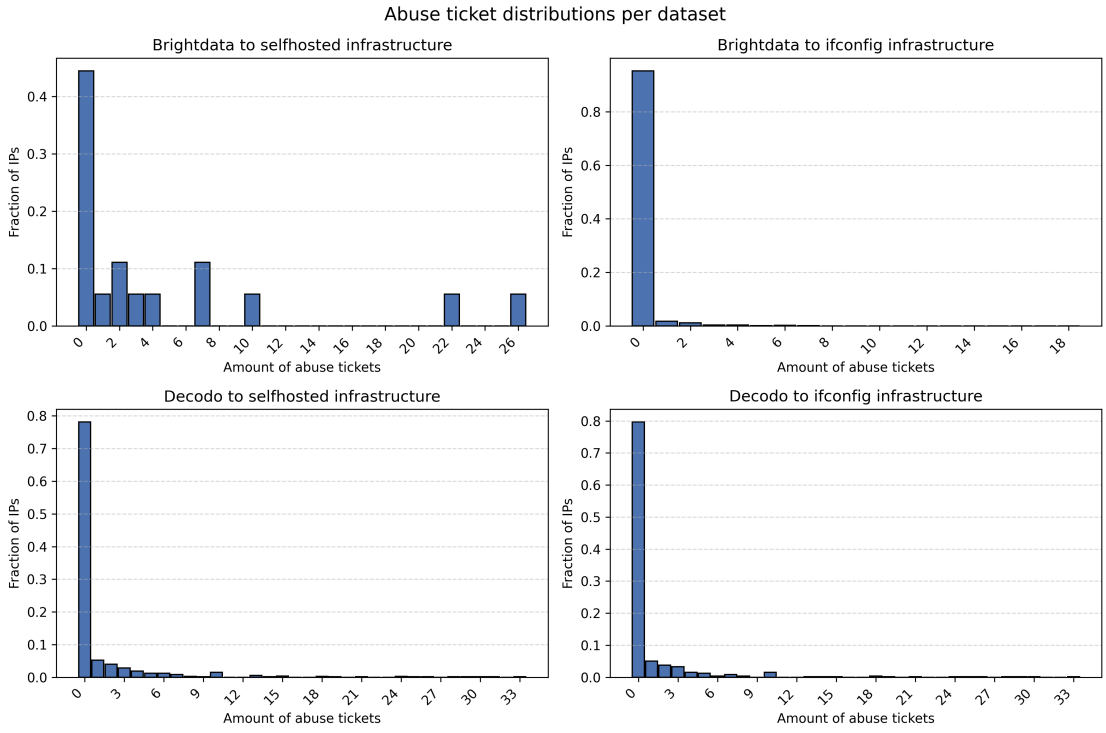

Figure 6.10: Distribution of abuse ticket counts per RESIP exit IP address across the four measurement datasets.

For the largest dataset (Bright Data via the external ifconfig endpoint), over 95% of the exit IP addresses had zero associated abuse tickets. This means fewer than 5% generated any

abuse reports during the measurement period. Similarly, for the Decodo self-hosted and ifconfig datasets, approximately 80% of the exit IP addresses had zero associated abuse tickets. This shows that the vast majority of active RESIP exit nodes do not trigger any abuse reports. In contrast, the Bright Data self-hosted dataset exhibited a relatively high number of abuse tickets per IP address. Because this dataset contained only 18 unique exit IP addresses, it is highly probable that these specific nodes are shared by a large number of proxy users, which increases the likelihood of abusive traffic being routed through them.

This indicates that abuse databases do not capture the vast majority of active RESIP exit nodes. Relying solely on external threat intelligence platforms or abuse reports is insufficient for detecting these nodes.

To further evaluate the visibility of proxy nodes in commercial intelligence platforms (as established in Section 5.4.3), the public IP address of the project-controlled residential connection was monitored via the Spur.us API throughout the active experiment. Despite functioning as an active exit node, Spur.us returned no active proxy tagging for the vast majority of the experimental period. The IP address only appeared briefly in the intelligence feed tagged as IPRoyal, which is expected since Pawns.app provides bandwidth for IPRoyal. Furthermore, the active device count reported by Spur.us for this IP only showed activity on the 28th and 29th of May, which coincided with the gap in the experiment when no proxy applications were active. Well after the experimental period concluded, Spur.us tagged the IP address as a Bright Data proxy as well, which may be due to delays in their intelligence ingestion pipelines.

To supplement the external threat intelligence analysis and evaluate detection from a localized network operator perspective, an internal Cyber Threat Intelligence (CTI) platform operated by KPN was utilized. This platform was queried for the public IP address of the project-controlled residential connection, as well as the backhaul infrastructure IP addresses identified during the active measurements (Experiments 1 and 2). Consistent with the findings from Spur.us, the internal CTI platform did not tag the project-controlled residential IP address as a VPN, proxy, or RESIP node. However, it did flag several identified Bright Data backhaul IP addresses, categorizing them as "unusual" and indicating they were previously seen in Advanced Persistent Threat (APT) reports, which associate these infrastructure nodes with targeted cyber campaigns. Furthermore, these Bright Data backhaul IP addresses were flagged as having been previously recorded in external abuse databases. Similarly, the central relay infrastructure for Pawns.app was flagged as having been previously reported on a threat list. This reveals that while RESIP exit nodes remain largely uncatalogued by external intelligence providers, some of the centralized backhaul infrastructure utilized by proxy networks is indeed identified and categorized as anomalous or malicious.

6.5 Technical viability of network-level mitigation

Building upon the evaluation framework established in the Implementation chapter, this section assesses the technical viability of various network-level mitigation strategies based on the identified behavioral signatures and infrastructure characteristics.

6.5.1 Ineffectiveness of reactive IP blocking

Given the sheer volume of unique IP addresses identified in external datasets (Section 6.1), reactive IP blocking based on such intelligence is highly ineffective and risky. Because residential IP addresses are dynamically allocated and could be reassigned among subscribers, applying network-level blocks based on these lists would inevitably lead to severe false positives, incorrectly cutting off legitimate customers from internet access. Furthermore, as shown in Section 6.4.3, attempting to block proxy traffic based on reactive IP blacklists sourced from threat intelligence completely misses the vast majority of proxy nodes operating silently within the network, since over 95% generate no abuse reports.

The distinct separation of residential exit nodes between different proxy providers (Section 6.2) further complicates IP-based mitigation. While different providers may share centralized backhaul infrastructure or utilize overlapping cloud services for their control planes, they exploit

completely different subsets of residential IP addresses for their exit nodes. Because these subscriber lines do not overlap, blocking the dynamic IP addresses associated with one service has zero impact on the broader proxy ecosystem. As a result, network operators cannot rely on identifying a few common residential nodes to disrupt multiple proxy services simultaneously.

Ultimately, completely blocking these IP addresses is unwanted, as customers are often unaware that they are hosting these proxies. Issuing security warnings to the subscriber is far safer and more proportionate than abruptly blocking their dynamic IP address. However, even automated warnings based on external IP feeds must be handled carefully, as customers could feel that their traffic is being monitored if the context is not properly communicated.

6.5.2 NetFlow-based detection versus automated blocking

Flow-level behavioral signatures, such as perfectly symmetric upload/download ratios and unnatural diurnal traffic peaks, provide strong indicators for detecting proxy activity within an ISP network. However, utilizing proactive NetFlow monitoring should focus on detection rather than enforcing automated network blocks. The computational overhead required to continuously analyze and correlate residential traffic flows at an ISP scale makes real-time intervention technically challenging. Furthermore, acting on these volumetric anomalies with automated blocking risks significant false positives, potentially affecting legitimate customers who happen to exhibit unusual but benign traffic patterns.

Instead, NetFlow analysis proves highly valuable for retrospective detection and intelligence gathering. While actively blocking connections to proxy control servers or C2 nodes is largely ineffective, as these IP addresses often cycle rapidly or are hosted on shared cloud infrastructure, historical NetFlow data provides a reliable mechanism for post-incident discovery. For example, when a new proxy C2 server is identified, historical NetFlow logs can be queried to pinpoint which residential connections previously communicated with that infrastructure. This retroactive discovery allows network operators to accurately detect compromised nodes and issue targeted security warnings to customers, utilizing NetFlow as a powerful diagnostic tool rather than a strict enforcement mechanism.

6.5.3 DNS-based mitigation and CPE enforcement

Because specific DNS and SNI resolution patterns emerge during proxy activation, relying on reactive DNS blocking is technically quite effective. While proxy clients frequently bypass the default local ISP DNS infrastructure to resolve final target destinations (Section 6.4.2), the initial connection to the centralized proxy backhaul servers is still resolved using local DNS. Therefore, blocking these backhaul domains at the network level can successfully disrupt the proxy control channel. However, because the actual destination domains are hidden within encrypted TLS layers, standard network-level DNS sinkholing or blocking mechanisms cannot observe or intercept the target traffic itself.

A potential solution to gain more visibility is shifting DNS monitoring and policy enforcement directly to the Customer-Premises Equipment (CPE). Because the CPE acts as the primary residential gateway, firmware-level controls can enforce strict DNS rules at the edge of the network. Reports from customer forums suggest that the Dutch ISP Ziggo implements a form of this edge-level enforcement, with users claiming that Ziggo-provided modems drop DNS responses that resolve to private IP addresses to provide DNS rebind protection [1].

Applying a similar CPE-based approach could restore visibility for proxy detection. By actively monitoring local DNS queries at the router level, or by blocking outbound connections to known public DoH/DoT resolvers to force a fallback to standard unencrypted DNS, the CPE can successfully identify the unique resolution patterns of proxy clients. This edge-based detection could then be used to automatically trigger security warnings for the affected customer, providing a localized detection mechanism. However, intentionally blocking DoH or DoT connections introduces severe privacy and security regressions for the subscriber. Depriving users of encrypted DNS exposes their entire browsing history to passive eavesdropping and undermines

[1] Forum post with Ziggo user complaining about DNS rebind protection: `https://community.vodafoneziggo.nl/t5/Internet/Ziggo-blocks-certain-\gls{dns}-queries/td-p/1296732`

modern internet privacy standards, making DoH blocking an unacceptable and disproportionate mitigation strategy.

6.5.4 SSL/TLS certificate fingerprinting

Another mitigation strategy involves identifying and blocking proxy connections based on SSL/TLS certificate fingerprints. Since proxy applications generate a large volume of HTTPS traffic (as established in Section 6.4.2), inspecting the unencrypted TLS handshakes to identify known proxy control server certificates could potentially be used for network-level detection. This can already be seen in Table 6.2, where certificates of Bright Data show up with the same Common Name (CN). Furthermore, the centralized infrastructure components of these proxy networks are often exposed, leading to the discovery of proxy endpoints through external scanning platforms like Shodan for specific SSL/TLS certificate names to reveal active C2 servers.

While performing deep packet inspection to continuously extract and match TLS metadata at an ISP scale requires substantial computational resources, shifting this detection mechanism to the CPE makes it technically viable. By inspecting TLS handshakes directly at the residential gateway, the CPE can identify known proxy control server certificates without the massive overhead required for network-wide monitoring. However, this approach remains highly reactive and suffers from significant limitations. As observed in the traffic profiles, proxy backhaul servers use dedicated infrastructure with certificates issued by standard authorities like Sectigo (for Bright Data) or Let's Encrypt (for Pawns.app). Because adversaries can cycle these domain names and certificates rapidly using automated tools, any blacklist of proxy SSL/TLS certificate fingerprints would become obsolete almost immediately. Thus, even with CPE-based enforcement, adversaries can easily evade detection by rotating their endpoints and certificates, making SSL fingerprinting an unscalable and fragile method for long-term mitigation.

6.6 Legal and policy constraints on mitigation

It must be emphasized that the analysis presented in this section does not constitute formal legal advice. It is a combination of internal discussions with stakeholders and a general interpretation of existing frameworks, intended solely to evaluate the real-world feasibility of the proposed technical interventions, rather than serve as definitive legal advice.

The evaluation of the proposed mitigation strategies against existing legal and policy frameworks highlighted constraints. To assess the viability of these interventions, it is necessary to separate the legality of *detecting* RESIP activity from the legality of the subsequent *response*. Both phases are strictly governed by the EU Open Internet Regulation [33], BEREC guidelines [36], the Dutch Telecommunications Act (*Telecommunicatiewet*) [11], and the GDPR.

6.6.1 Legality of detection mechanisms

Detecting RESIP activity requires visibility into subscriber traffic, which inherently conflicts with privacy regulations. A discussion with internal legal and privacy stakeholders confirmed that proactive detection mechanisms are legally sensitive.

- **Proactive network monitoring:** Continuously analyzing NetFlow records or DNS queries to identify subscriber proxy endpoints is legally problematic. Even when only analyzing communication patterns rather than content, this metadata qualifies as *verkeersgegevens* (traffic data) under Dutch telecommunications law. The permitted processing grounds for such data are narrow, and broad, indiscriminate interception is considered disproportionate for the specific threat of RESIPs.
- **CPE-based telemetry:** Shifting detection to the CPE avoids inspecting in-flight core network traffic, but still poses privacy risks. Localized DNS monitoring on the router theoretically captures proxy resolution patterns, but intercepting or logging user queries directly at the gateway risks violating subscriber privacy expectations. Instead, proactive

firmware controls that restrict unauthorized Universal Plug and Play (UPnP) configurations or detect malicious client installations provide a more privacy-preserving detection path.

- **Retrospective discovery:** Using historical NetFlow data to retroactively identify which residential connections communicated with newly discovered proxy C2 servers is considered more proportionate than continuous proactive monitoring, as the search is targeted and driven by specific threat intelligence.

Internal assessments concluded that the proactive use of traffic metadata lacks a definitive legal basis, placing these practices in a regulatory grey area. Consequently, processing this metadata can only be justified under exceptional circumstances, such as when proxy abuse poses a direct threat to network integrity and security.

6.6.2 Legality of response actions

Once a proxy node is detected, the ISP must determine the appropriate intervention. As established by the EU Open Internet Regulation [33], ISPs are generally prohibited from blocking, slowing down, or discriminating between specific applications or services. Any traffic management must be reasonable, transparent, non-discriminatory, and proportionate.

- **Automated blocking:** Exceptions for network security allow ISPs to implement measures to preserve network integrity [37]. However, reactive IP or DNS blocking of known proxy domains faces severe legal proportionality hurdles. Because these infrastructure elements cycle rapidly, blocking them risks false positives that incorrectly cut off legitimate customers, violating net neutrality rules. Consequently, automated blocking is viewed by legal stakeholders as a disproportionate response. Furthermore, if RESIPs are blocked, customers may not know that they are affected, and thus can't take action to stop the proxy abuse.

- **Malware blocklist integration:** Alternatively, ISPs can append identified proxy infrastructure to consumer malware protection blocklists, such as KPN Veilig. While this effectively neutralizes the malicious traffic at the DNS level, it limits the ISP's operational insight. Although failed DNS queries remain visible in the logs, deeper visibility into the client's subsequent network behavior and intent is lost. Furthermore, because the block happens silently, the customer remains entirely unaware that their device is compromised. The underlying malware or proxy client remains fully functional on the host device, free to reach out to alternative, unknown domains that have not yet been blacklisted. Another significant concern is that not all proxy traffic is unwanted. Some users may have intentionally installed these sharing applications to generate passive income. In such cases, the block would disrupt what the subscriber considers legitimate, consented traffic, potentially violating user autonomy and generating customer complaints. To mitigate these downsides, this technical restriction could be deployed in close cooperation with targeted customer notifications, transforming a silent block into an actionable security alert for the user.

- **Targeted customer notification:** Sending security warnings to the affected subscriber is legally the safest and most proportionate intervention. Since customers are often unaware that their devices are acting as proxy nodes, notifying them fulfills the ISP's duty of care while respecting net neutrality. CPE controls can also be utilized to actively warn the user about insecure configurations on their local network without silently intercepting or blocking their traffic.

Based on these findings, broad network-level monitoring or generic automated blocking of RESIPs is largely disproportionate and legally unfeasible. Instead, interventions should prioritize privacy-preserving, localized detection at the CPE combined with targeted customer notification. To operationalize this, ISPs can pair automated DNS blocklists with localized user alerts: when a proxy domain is blocked, a targeted customer notification is triggered to prevent the mitigation

from remaining silent. Concurrently, CPE telemetry can identify the specific vulnerabilities, such as open ports, weak credentials, or outdated firmware, that enabled the compromise. Combining immediate DNS blocking with explicit notifications about these CPE insecurities gives users the power to remediate the root cause, preventing the underlying malware from simply migrating to unknown domains.

CHAPTER 7

Conclusion

7.1 Discussion

This thesis set out to investigate the presence of RESIPs within a large telecommunications network and to evaluate potential network-level mitigation strategies. The historical dataset from 2017–2018 compared with recent Spur data from 2026 demonstrates that the problem is growing, with the number of observed proxy IP addresses in KPN's network roughly tripling. With over 38,000 addresses currently tagged within AS1136 alone, and 60% of RESIP nodes observed only once before rotating away, the problem is both large and structurally resistant to reactive countermeasures.

This growth presents a critical challenge, in line with observations from Google Threat Intelligence Group [1], Martins and Donnini [10], and the Dutch National Cyber Security Centre (NCSC) [4], regarding the escalating use of RESIP networks by threat actors. As adversaries increasingly leverage these infrastructures for resilient DDoS operations, large-scale web scraping, and covert reconnaissance, the volume of abuse reports tied to residential IP spaces continues to rise [10]. Since RESIPs seamlessly blend malicious traffic with legitimate consumer activity, traditional intervention strategies, such as static IP-based blocking, active scanning, and IP range blocklists, remain largely ineffective. This highlights the urgent need for the scalable, privacy-preserving detection methods explored in this research to mitigate the risk of widespread abuse, particularly concerning vulnerable IoT devices within consumer households [8].

To answer the first research question regarding behavioral signatures, the results demonstrate that RESIP nodes drastically deviate from typical residential baselines. Rather than the expected asymmetric download-heavy traffic, active proxy nodes exhibit perfectly symmetric traffic flows with an upload/download ratio of 1.0, and distinct diurnal peaks that indicate automated scheduling rather than human usage. Furthermore, proxy traffic frequently bypasses local ISP DNS infrastructure, instead relying on public resolvers or encrypted channels (DoH/-DoT). This results in anomalies in the DNS traffic where standard DNS queries remain low, but unique TLS connections spike into the thousands. This combination of signatures: traffic symmetry, diurnal scheduling, and DNS/TLS divergence, constitutes a reliable behavioral fingerprint detectable from standard NetFlow and TLS metadata. However, the architectural diversity observed between different applications demonstrates that the RESIP ecosystem is far from uniform. For example, EarnApp operates as a decentralized exit node with distributed target traffic, whereas Pawns.app routes all traffic through a single centralized relay. Furthermore, the false positive rate of these signatures against real-world residential traffic, particularly peer-to-peer applications, gaming, and video conferencing, has not yet been measured.

Regarding the second research question, the findings show that cross-referencing passive network observations with external threat intelligence offers very little confidence improvement for detection. Over 95% of Bright Data and approximately 80% of Decodo active proxy exit nodes evaluated in this study generated zero abuse tickets, indicating that traditional abuse databases are largely blind to the silent operation of RESIPs. Furthermore, due to commercial licensing and query rate constraints associated with the Spur enterprise API, individual queries could

not be executed for the entire set of identified exit IP addresses. Consequently, the real-time classification performance of commercial IP threat intelligence was evaluated by monitoring the public IP address of the single project-controlled residential connection over the course of the experiment, rather than through a bulk correlation of the entire ground-truth dataset. Ultimately, Spur failed to flag the active residential node during the experiment, only tagging it long after the tests concluded. Similarly, KPN's internal CTI platform failed to detect the exit node itself but successfully flagged its centralized backhaul and relay infrastructure, indicating that while exit nodes remain largely hidden, centralized control planes are often known to defenders.

To answer the third research question, both detection and response activities are heavily constrained by the GDPR, the Dutch Telecommunications Act (*Telecommunicatiewet*), and net neutrality regulations. Proactive, network-wide monitoring of traffic data (*verkeersgegevens*) lacks a clear legal basis and is disproportionate. Instead, targeted retrospective log analysis and localized CPE telemetry offer privacy-preserving detection paths that avoid inspecting in-transit core network traffic. On the response side, automated blocking is prohibited under net neutrality rules due to the risk of collateral damage on dynamic IP addresses. Consequently, the most legally permissible and effective intervention combines automated DNS blocklists with targeted customer notifications. By pairing immediate network-level blocking with actionable alerts regarding specific CPE insecurities, the ISP fulfills its duty of care and empowers the user to remediate the root cause, preventing the underlying malware from silently migrating to unknown domains.

Finally, to answer the primary research question, an ISP can effectively identify and mitigate RESIP nodes by adopting a hybrid, privacy-preserving framework. Rather than relying on continuous, network-wide inspection or brittle threat intelligence feeds, detection should combine retrospective flow analysis with localized CPE-level telemetry. Mitigation must avoid aggressive, silent automated blocking, which violates net neutrality, leaves customers unaware, and fails to stop malware communicating with unknown domains. Instead, the framework should pair immediate DNS blocklists with targeted customer notifications. This balanced approach protects the operational integrity of the ISP network, empowers users to remediate the underlying compromise, and respects regulatory limits and customer privacy.

7.2 Limitations

Several limitations and threats to validity must be acknowledged. In terms of construct validity, the measurements rely heavily on SNI patterns and specific DNS behaviors observed from only two proxy providers (Bright Data and Decodo) running two applications. This may not accurately capture the entire RESIP ecosystem's behavior. Regarding conclusion validity, the evaluation of commercial threat intelligence was constrained by API access limitations, meaning the real-time classification was monitored for a single IP address rather than bulk-correlating the full ground-truth dataset. This limits the ability to determine exact device lifespans, as some devices like digital photo frames may simply connect infrequently rather than representing dead nodes. Concerning internal validity, production NetFlow access could not be obtained within the project timeline. Flow analysis was performed on PCAP-derived records using Tranalyzer2, which mirrors production NetFlow data in structure but has not been validated against live ISP collectors at scale. Additionally, the false positive rate of the identified signatures against other high-volume symmetric residential traffic remains unmeasured. Finally, concerning external validity, the experimental measurements were strictly confined to KPN's residential network (AS1136). While the geographic distribution showed an organic spread across the Netherlands, these specific network signatures and the accompanying legal framework are highly localized and may not directly generalize to other ISPs globally. Furthermore, although this thesis discusses regulatory constraints under GDPR, the Dutch Telecommunications Act, and net neutrality, all legal analyses and interpretations are provided strictly for academic research purposes and do not constitute formal legal advice.

7.3 Future work

While this thesis establishes a foundational methodology for passive RESIP detection, several paths remain for future research to expand upon these findings.

First, further research should focus on expanding the geographic and architectural visibility of RESIPs beyond a single ISP (KPN, AS1136) and a limited set of proxy providers. Investigating traffic traversing major Internet exchanges (IXPs), such as NL-IX or AMS-IX, could provide a more comprehensive, ISP-agnostic view of affected users across multiple providers by passively analyzing SNI fields and flow metadata. Additionally, conducting comparative studies in various countries with differing bandwidth distributions and consumer patterns would help evaluate the generalizability of the behavioral signatures identified in this study. Expanding the measurement pipeline to include more simultaneous testlines and a broader variety of proxy applications (such as Honeygain, PacketStream, or IPRoyal) would also reveal differences in client-side software behaviors, backhaul routing topologies, and control infrastructure designs.

Second, as privacy standards evolve, finding a way to detect proxy traffic without relying on plaintext TLS SNI data is critical. The impending widespread adoption of ECH [1] will obfuscate SNI fields, rendering current TLS-based visibility mechanisms obsolete. Future studies must develop SNI-independent detection heuristics, focusing on flow-level statistical characteristics, such as packet length distributions, inter-arrival times, burstiness, and flow duration. Another direction is to focus on encrypted TLS fingerprinting techniques such as JA4 fingerprints or ALPN negotiations that remain visible to network observers.

Third, given the legal and privacy challenges of central, network-wide traffic inspection within an ISP, future work should design, implement, and benchmark edge-level detection agents for consumer CPE platforms, such as OpenWrt. This requires researching how to execute lightweight flow correlation, DNS resolution path tracking, and DoH/DoT bypass detection under the CPU and memory constraints of standard consumer routers. Furthermore, investigating user-friendly and privacy-preserving mechanisms for displaying CPE-triggered warnings to subscribers is crucial.

Fourth, as defenders implement behavioral detection, proxy providers are expected to adapt. Proactive research is needed to study potential provider evasion tactics, such as introducing artificial traffic asymmetry, such as padding download traffic or introducing random delays to mimic residential browsing, randomizing scheduling to flatten diurnal peaks, or utilizing OS-native DNS resolution calls instead of direct IP routing. Analyzing the limitations of behavioral classifiers against these countermeasures will help design more resilient detection methods.

Fifth, future research should investigate the ToS and Acceptable Use Policies of other residential ISPs and proxy providers in practice. This would involve a comparative analysis of how different network operators contractually address or restrict proxy sharing software on customer lines, how these terms are enforced in practice, and a broader comparative study of how various proxy providers police their own networks to prevent abuse.

Finally, while this study processed simulated NetFlow records using packet captures on a project-controlled connection, future research should evaluate the proposed signatures (specifically symmetric upload/download ratios and diurnal peaks) on real-world, large-scale NetFlow production collectors. This would allow researchers to assess the true false-positive rates when faced with other high-volume symmetric customer activities, such as peer-to-peer sharing, online gaming, and remote video conferencing, and to determine the feasibility of running such classifiers at line rate in an ISP core network.

[1] See: `https://www.security.com/expert-perspectives/navigating-encrypted-client-hello-ech-insights-rsac-2025`

CHAPTER 8

Ethics

When conducting research on RESIPs within an ISP, it is essential to balance technological advancements with legal and ethical responsibilities. As stated in the Dutch Telecommunications Act (*Telecommunicatiewet*) [11], interception of user traffic is strictly prohibited except under specific, legally defined circumstances. Therefore, although an ISP may have the technical means to identify such RESIPs, it is necessary to first consider whether it has the legal right to do so and under which conditions such identification would be permissible.

When starting this research, all identification and mitigation were first carried out in a controlled environment on separate residential connections that hosted proxy provider services operated under research control. In this setup, metadata for traffic originating from the research-controlled RESIPs was captured to study identification and mitigation techniques without affecting real subscribers. Data from users of these proxies came from bandwidth providers and was not traceable to individuals, similar to Khan et al. [6]. Separately, access to commercial proxy providers was purchased to obtain lists of RESIPs. For these externally sourced addresses, the only direct interaction was a single HTTP request to a publicly available IP intelligence service (such as ipinfo.io) from each address, limited to high-level information (e.g., ASN). No further probing, scanning, or data collection was performed in this phase on IP addresses or systems outside the controlled environment.

At the same time, discussions were initiated with the legal department within KPN. This was to protect consumers and ensure that any potential detection and mitigation technique was assessed for compatibility with applicable law and internal KPN policies before it was considered for use on production networks. Only after it had been established that specific options were legally permissible and proportionate were carefully scoped real-world applications of these techniques considered.

To further protect subscriber privacy, all analysis was conducted on pseudonymized identifiers, and only the minimum data necessary to answer the research questions was retained. Raw telemetry was stored in a secure, access-controlled environment and deleted after analysis. Only metadata already collected for operational purposes (NetFlow, DNS logs) was used. Any findings were reported at an aggregate level, ensuring that no individual subscriber could be identified. In the unlikely event that any data was collected falling outside of this scope, it was reported to the relevant authorities, securely deleted, and excluded from analysis.

Equally important is the principle of proportionality in mitigation. Subscribers whose devices have been unknowingly compromised and enrolled as proxy nodes are victims, not offenders. The research aims to benefit these victims by enabling the detection and mitigation of such abuse. Abrupt disconnection would be a disproportionate response and, as the United Nations affirms, risks violating the right to access the internet, which is considered a fundamental human right [39]. While the inclusion of their (pseudonymized) data in this study was a limited intrusion into privacy, this was strictly minimized and secured. It was also justified by the benefit of detecting these compromises earlier and providing affected customers with solutions. Today, subscribers whose connections show signs of abuse are already notified, and in severe cases may be temporarily placed in a quarantine environment with restricted connectivity until the issue

is resolved. The techniques developed in this thesis were intended to improve the detection that feeds into this established process, not to introduce more intrusive methods.

Bibliography

[1] Google Threat Intelligence Group. *No Place Like Home Network: Disrupting the World's Largest Residential Proxy Network*. Jan. 2026. URL: https://cloud.google.com/blog/topics/threat-intelligence/disrupting-largest-residential-proxy-network (visited on 7-4-2026).

[2] Valter Santos. "The Symbiosis of Residential Proxy Services and Malware Ecosystems". In: (May 2026). URL: https://www.bitsight.com/blog/residential-proxy-services-malware-ecosystems (visited on 27-5-2026).

[3] Pawns.app. *Home Terms Of Use of use Exclusive Bonus Available Mobile App Only April 07-14 Sign Up Terms*. Tech. rep. Mar. 2026. URL: https://pawns.app/terms-of-use/ (visited on 9-4-2026).

[4] National Cyber Security Centre (NCSC). "Residential proxies en hun grote impact op de digitale veiligheid in Nederland". In: (May 2026). URL: https://www.ncsc.nl/expertblogs/residential-proxies-en-hun-grote-impact-op-de-digitale-veiligheid-in-nederland (visited on 27-5-2026).

[5] Tafara Muwandi. *2025 Advanced Persistent Bots Report*. Tech. rep. F5 Labs, 2025. URL: https://www.f5.com/labs/articles/threat-intelligence/2021-credential-stuffing-report (visited on 2-4-2026).

[6] Etienne Khan et al. "A First Look at User-Installed Residential Proxies From a Network Operator's Perspective". In: *2024 20th International Conference on Network and Service Management (CNSM)*. IEEE, Oct. 2024, pp. 1–9. ISBN: 978-3-903176-66-9. DOI: 10.23919/CNSM62983.2024.10814519. URL: https://ieeexplore.ieee.org/document/10814519/.

[7] Etienne Khan et al. "Stranger VPNs: Investigating the Geo-Unblocking Capabilities of Commercial VPN Providers". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Vol. 13882 LNCS. Springer Science and Business Media Deutschland GmbH, 2023, pp. 46–68. ISBN: 9783031284854. DOI: 10.1007/978-3-031-28486-1_3. URL: https://link.springer.com/10.1007/978-3-031-28486-1_3.

[8] European Union Agency For Network And Information Security. *Baseline security recommendations for IoT in the context of critical information infrastructures*. ENISA, Nov. 2017. ISBN: 9789292042363. URL: https://www.enisa.europa.eu/sites/default/files/publications/WP2017%20O-1-1-2%201%20Baseline%20Security%20Recommendations%20for%20IoT%20in%20the%20context%20of%20CII_FINAL.pdf (visited on 16-6-2026).

[9] Gertjan Halkes and Johan Pouwelse. "UDP NAT and Firewall Puncturing in the Wild". In: 2011, pp. 1–12. DOI: 10.1007/978-3-642-20798-3_1. URL: http://link.springer.com/10.1007/978-3-642-20798-3_1.

[10] Tiago Martins and Fernanda Donnini. "The Residential Proxy Problem Shared Infrastructure and Rapid Rotation". In: (Jan. 2026). URL: https://ipinfo.io/blog/residential-proxy-shared-infrastructure-churn (visited on 27-5-2026).

[11] Kingdom of the Netherlands. *Telecommunicatiewet [Telecommunications Act]*. Sept. 2025. URL: https://wetten.overheid.nl/jci1.3:c:BWBR0009950&z=2025-09-01&g=2025-09-01 (visited on 2-4-2026).

[12] Xianghang Mi et al. "Resident Evil: Understanding Residential IP Proxy as a Dark Service". In: *2019 IEEE Symposium on Security and Privacy (SP)*. IEEE, May 2019, pp. 1185–1201. ISBN: 978-1-5386-6660-9. DOI: 10.1109/SP.2019.00011. URL: https://ieeexplore.ieee.org/document/8835239/.

[13] Akihiro Hanzawa and Hiroaki Kikuchi. "Analysis on malicious residential hosts activities exploited by residential IP proxy services". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Vol. 12583 LNCS. Springer Science and Business Media Deutschland GmbH, 2020, pp. 349–361. ISBN: 9783030652982. DOI: 10.1007/978-3-030-65299-9_26.

[14] Dan Komosny, Miroslav Voznak, and Saeed Ur Rehman. "Location Accuracy of Commercial IP Address Geolocation Databases". In: *Information Technology and Control* 46 (3 Sept. 2017), pp. 333–344. ISSN: 2335-884X. DOI: 10.5755/j01.itc.46.3.14451. URL: https://itc.ktu.lt/index.php/ITC/article/view/14451.

[15] Sumayah Alrwais et al. "Under the Shadow of Sunshine: Understanding and Detecting Bulletproof Hosting on Legitimate Service Provider Networks". In: *2017 IEEE Symposium on Security and Privacy (SP)*. IEEE, May 2017, pp. 805–823. ISBN: 978-1-5090-5533-3. DOI: 10.1109/SP.2017.32. URL: http://ieeexplore.ieee.org/document/7958611/.

[16] Brian Krebs. *Aisuru Botnet Shifts from DDoS to Residential Proxies*. Oct. 2025. URL: https://krebsonsecurity.com/2025/10/aisuru-botnet-shifts-from-ddos-to-residential-proxies/ (visited on 2-4-2026).

[17] Comcast Threat Research Labs (CTRL). *ResProxies – What Hasn't Been Herd Yet*. Sept. 2025. URL: https://corporate.comcast.com/stories/resproxies-what-hasnt-been-herd-yet (visited on 2-4-2026).

[18] Xianghang Mi et al. "Your Phone is My Proxy: Detecting and Understanding Mobile Proxy Networks". In: *28th Annual Network and Distributed System Security Symposium, NDSS 2021*. The Internet Society, 2021. ISBN: 1891562665. DOI: 10.14722/ndss.2021.24008.

[19] Satori Threat Intelligence and Research Team. *Satori Threat Intelligence Disruption: BADBOX 2.0 Targets Consumer Devices with Multiple Fraud Schemes*. Mar. 2025. URL: https://www.humansecurity.com/learn/blog/satori-threat-intelligence-disruption-badbox-2-0/ (visited on 7-4-2026).

[20] Elisa Chiapponi, Marc Dacier, and Olivier Thonnard. "Poster: The Impact of the Client Environment on Residential IP Proxies Detection". In: *Proceedings of the 2023 ACM on Internet Measurement Conference*. ACM, Oct. 2023, pp. 712–713. ISBN: 9798400703829. DOI: 10.1145/3618257.3624993.

[21] Temoor Ali et al. "Beyond RTT: An Adversarially Robust Two-Tiered Approach for Residential Proxy Detection". In: (Feb. 2026). DOI: https://dx.doi.org/10.14722/ndss.2026.24208. URL: https://dx.doi.org/10.14722/ndss.2026.242086.

[22] Altug Tosun et al. "RESIP Host Detection: Identification of Malicious Residential IP Proxy Flows". In: *2021 IEEE International Conference on Consumer Electronics (ICCE)*. Vol. 2021-January. IEEE, Jan. 2021, pp. 1–6. ISBN: 978-1-7281-9766-1. DOI: 10.1109/ICCE50685.2021.9427688. URL: https://ieeexplore.ieee.org/document/9427688/.

[23] Ronghong Huang et al. "Shining Light into the Tunnel: Understanding and Classifying Network Traffic of Residential Proxies". In: *Proceedings 2021 Network and Distributed System Security Symposium* (Apr. 2024). DOI: DOI:10.48550/arxiv.2404.10610. URL: http://arxiv.org/abs/2404.10610.

[24] Hiroaki Kikuchi et al. "Honey-Proxy: Revealing Malicious Activities via Residential Proxies". In: vol. 5. 2025, pp. 1–12. DOI: 10.1007/978-3-031-87775-9_1. URL: https://link.springer.com/10.1007/978-3-031-87775-9_1.

[25] M.F. Verkleij. "A Toolkit for Performing Network Measurements on Bandwidth Brokers". PhD thesis. University of Twente, Aug. 2025. URL: https://purl.utwente.nl/essays/108490.

[26] Philip Branch. *Lawful Interception of the Internet*. Tech. rep. 2003, pp. 38–51. DOI: https://doi.org/10.25916/sut.26277076. URL: http://www.swin.edu.au/ajets.

[27] Gabriel Maciá-Fernández et al. "UGR'16: A new dataset for the evaluation of cyclostationarity-based network IDSs". In: *Computers & Security* 73 (Mar. 2018), pp. 411–424. ISSN: 01674048. DOI: 10.1016/j.cose.2017.11.004. URL: https://linkinghub.elsevier.com/retrieve/pii/S0167404817302353.

[28] Josef Zápotocký, Jan Fiala, and Jan Fesl. "A user DNS fingerprint dataset". In: *Data in Brief* 54 (June 2024), p. 110389. ISSN: 23523409. DOI: 10.1016/j.dib.2024.110389. URL: https://linkinghub.elsevier.com/retrieve/pii/S2352340924003585.

[29] Doug Brake and Alexandra Bruer. *Broadband Myth Series: Do We Need Symmetrical Upload and Download Speeds?* Tech. rep. Information Technology & Innovation Foundation, May 2021. URL: https://itif.org/publications/2021/05/12/broadband-myth-series-do-we-need-symmetrical-upload-and-download-speeds/ (visited on 21-5-2026).

[30] The Shadowserver Foundation. *The Shadowserver Foundation*. 2026. URL: https://www.shadowserver.org (visited on 9-4-2026).

[31] KPN B.V. *Algemene Voorwaarden voor vaste en mobiele telecommunicatiediensten*. Tech. rep. July 2025. URL: https://www.kpn.com/algemene-voorwaarden/prive/internet (visited on 2-4-2026).

[32] KPN B.V. *KPN Security Policy*. Feb. 2026. URL: https://policy-public.kpnnet.org/ksp (visited on 7-4-2026).

[33] European Parliament and Council of the European Union. *Regulation ({EU}) 2015/2120 of the {European Parliament} and of the {Council} of 25 {November} 2015 laying down measures concerning open internet access*. May 2024. URL: https://eur-lex.europa.eu/eli/reg/2015/2120/2024-05-15/eng (visited on 14-5-2026).

[34] Spur Intelligence. *Spur IP Enrichment & Intelligence - Detect Residential Proxies*. Spur. 2026. URL: https://spur.us/ (visited on 6-5-2026).

[35] MaxMind, Inc. *MaxMind GeoIP2*. https://www.maxmind.com/en/geoip2-services-and-databases. 2026. (Visited on 6-5-2026).

[36] Body of European Regulators for Electronic Communications. *BEREC Guidelines on the Implementation of the Open Internet Regulation*. Tech. rep. June 2022. URL: https://www.berec.europa.eu/sites/default/files/files/document_register_store/2022/6/BoR_%2822%29_81_Update_to_the_BEREC_Guidelines_on_the_Implementation_of_the_Open_Internet_Regulation.pdf (visited on 27-5-2026).

[37] Autoriteit Consumer & Markt. *Traffic Management - Voorlichtend document*. Tech. rep. Jan. 2020. URL: http://eur-lex.europa.eu/legal-content/NL/TXT/PDF/?uri=CELEX:32015R2120&rid=1. (visited on 27-5-2026).

[38] Eleni Vytogiannni and Marnix Dekker. "Guideline on assessing security measures in the context of Article 3(3) of the Open Internet regulation". In: (Dec. 2018). DOI: 10.2824/94531. URL: https://www.enisa.europa.eu/sites/default/files/publications/2018-12-12%20Guideline%20on%20assessing%20security%20measures.pdf.

[39] Human Rights Council. *The promotion, protection and enjoyment of human rights on the Internet*. July 2021. URL: https://digitallibrary.un.org/record/3937534?v=pdf#files (visited on 7-4-2026).