

Keywords: residential proxy; consumer safety; operator responsibilities; App Promises; test plan; AppEsteem	September 11, 2026 Version 1.1
--	-----------------------------------

Residential Proxy Operator Responsibilities

Test Plan for 2026

AMTSO Standard Compliance Statement

This Test has been designed to comply with the Anti-Malware Testing Standards Organization, Inc. (“AMTSO”) Testing Protocol Standard for the Testing of Anti-Malware Solutions, Version 1.3 (the “Standard”). It also follows the methodology of the AMTSO Guidelines for Testing Application Promises (proposed; currently DRAFT v0.1 before the AMTSO Application Promises Working Group). AppEsteem Corporation is solely responsible for the content of this Test Plan.

Table of Contents

1. Introduction	3
1.1. <i>The promises we test</i>	4
2. Scope.....	4
2.1. <i>The promises under test</i>	4
2.2. <i>In scope and out of scope</i>	5
2.3. <i>Who we test as: the least-scrutinized customer</i>	6
3. Methodology and Strategy.....	6
3.1. <i>How we measure</i>	6
3.2. <i>The six things we check (buckets)</i>	7
3.3. <i>What we can and cannot measure</i>	8
3.4. <i>We check both kinds of failure</i>	8
3.5. <i>Which test covers which responsibility</i>	8
4. Participation	9
4.1. <i>An optional, paid QA service</i>	9
4.2. <i>Withdrawal, conflicts, and funding</i>	10
5. Environment.....	10
5.1. <i>Test machine and account</i>	10
5.2. <i>Where the test sites come from</i>	11
5.3. <i>Geography</i>	12
5.4. <i>What we share</i>	12
5.5. <i>The controlled live site</i>	12
6. Schedule.....	12
7. Control Procedures	13
7.1. <i>Always compare against a baseline</i>	13
7.2. <i>Credit the operator, not the destination</i>	13
7.3. <i>Health checks</i>	13
7.4. <i>Keep the run from looking like a test</i>	13
7.5. <i>Tester conduct and public safety</i>	13
7.6. <i>Evidence</i>	14
8. Dependencies	14
9. Reporting Results	14

9.1. How we record each result.....	15
9.2. Under-delivery and over-blocking.....	15
9.3. Reporting and what happens next.....	16
10. Dispute Process	16
11. Attestations.....	17

1. Introduction

A residential proxy operator pays or rewards ordinary people to share their home internet connection. It then sells its customers the ability to send web traffic out through those home connections, so the traffic looks like it comes from a normal house. When someone agrees to share, the operator makes them a promise: sharing is safe, and nothing bad will happen to them or their device. It makes a second promise to the business that buys access: the work it sends through those homes won't reduce their security posture. This Test checks whether operators keep both promises.

We call the person who shares their connection the consumer (in technical terms, the endpoint owner), and the business that buys access the customer. This Test measures the risk an operator leaves with each of them, and reports the two separately (§9). The consumer comes first: they are the party AppEsteem certifies the promise to, and they cannot see or control what leaves their home. The customer is measured too, because a business that sends its traffic out through someone's home has handed its security perimeter to the operator. Whatever the operator fails to block arrives inside the customer's own infrastructure with its firewall and filters bypassed, and the operator's records name the customer as the party behind it. What this Test does not protect is the websites the customer visits. That choice decides which findings count: if an operator stops something that we think would hurt the consumer or the customer, that is good; if it allows something that we believe would hurt either of them, that is bad, even if the operator's own terms of use mention it.

Why we run this Test. We are concerned for both consumers and customers. Getting the consumer's informed consent at install time (the industry calls this "ethical sourcing") is a good start, and so is letting them control when the proxy runs, and AppEsteem certifies that today. But consent is only honest if day-to-day reality matches it. If, after install, a customer can use the connection to attack websites, break into accounts, carry illegal traffic, or reach into the home network, then the consumer agreed to one thing and got another. On the other hand, if a customer's account can be used to bring malware into an organization, or misused to access illegal or unwanted websites, the customer's security posture is damaged. This Test is a tool to keep consumers and residential proxy customers safe. We use it, together with our certification and Deceptor programs, to encourage operators to protect the people they make money from.

Our two programs. When AppEsteem certifies software, we tell the public it is safe to install and use. We also keep a free, public list of software applications that we believe scare, trick, or harm ordinary people (our active Deceptor list). An operator that fails this Test, whether or not we certified them for ethical sourcing and proxy control, risks losing its certification and getting their software added to our active Deceptor list.

1.1. The promises we test

AppEsteem’s whitepaper, Residential Proxy Operator Responsibilities (Draft 260819, August 2026), lists twelve things a responsible operator must do to keep their promises. Its Section 5 sets those responsibilities out for two parties at once: the consumer whose home carries the traffic, and the responsible customer who buys access. Its Section 3 names the three risks the consumer takes on, and its Section 4 names the three the customer takes on. We draw from that list for this Test, and we measure and report against both parties. The whitepaper, written with the help of several operators and the anti-virus community, is the basis and the evidence for the Test; we test against it directly and do not re-argue the sources behind it here.

This Test follows the AMTSO Testing Protocol Standard v1.3 and the AMTSO Fundamental Principles of Testing, and uses the methodology of the AMTSO Guidelines for Testing Application Promises (the residential-proxy operator is the first worked example in those Guidelines). We intend to follow the Standard, and we hope all tested operators will choose to be Participants.

2. Scope

This Test evaluates residential-proxy operators: companies that get people to share their home connections, place exit software on the consumer’s device, and resell exit access to their own customers. Where an operator offers more than one way in (e.g., a plain HTTP proxy, a published agent or “MCP” skill, or a SOCKS5 port) we plan to test each one on its own over time, because protection can differ between them.

2.1. The promises under test

The whitepaper lists twelve responsibilities, RPR-001 through RPR-012. This Test measures the ten that concern running the proxy safely, RPR-002 through RPR-011. The whitepaper says who each one protects, and most protect the consumer and the customer at once, for the same reason: harmful traffic that never leaves the home cannot hurt either of them. In plain terms, a responsible operator selling to the “least-scrutinized customer” (see §2.3) should:

	Responsibility
RPR-002	Keep the address clean. Prevent the abuse that gets a home address marked as a spammer, bot, incentive abuser, or fraud source, up front rather than cleaning up after. Protects both: the consumer’s address stays clean, and the customer’s own traffic is not lumped in with abuse.
RPR-003	No illegal traffic leaves the home, and none reaches malware sites, criminal marketplaces, or stolen-data forums. Protects both: none of it leaves the consumer’s home, and none of it lands on the customer’s record.
RPR-004	Reading public pages only. The connection may read what a site already shows everyone; it must not log in or carry credentials, cookies, or tokens to reach private, account-only data. Protects both: it holds the consumer’s connection to public reading, and it blocks the credential relay that would expose the customer’s own logins.
RPR-005	Keep the traffic pointed outward. Never turned inward at the consumer’s router, smart devices, or other home computers, and not used for non-web services such

	Responsibility
	as mail sending, remote login, or file sharing. Protects both: attackers stay out of the home, and the non-web paths a customer's own firewall would normally block stay shut.
RPR-006	Block the risky site categories by default, for every anonymous sign-up, without waiting to be asked. Protects both: it keeps the consumer out of trouble, and blocking malware and criminal sites keeps harmful content and legal exposure away from the customer.
RPR-007	Block known bypasses. A tiny change to a web address, or a different way in, must not slip past a block. A rule a small bypass can defeat is not enforcement, and enforcement that holds is what makes every protection above real for both parties rather than nominal.
RPR-008	Publish clear, readable rules (the acceptable-use policy), at least as strict as RPR-001 to RPR-007, in the consent flow and reachable while the software runs. Clear rules also let a prospective customer see what an operator truly allows before trusting its work to it.
RPR-009	Publish every exception to those rules, with its reason: specific as to customer categories, website categories, and actions, in the consent flow and reachable while the software runs. It lets the consumer see how their connection may be used, and lets a customer judge the carve-outs before relying on them.
RPR-010	Wall the software off from the consumer: a sealed doorway for outbound traffic, never a window into the home, and never reading the consumer's passwords, cookies, files, browsing, or home network. The same seal keeps the customer's credentials and traffic from being read in transit.
RPR-011	Keep the software hardened, patched, and consistent across every way the service can be used. Consistent protection across every surface is what closes the gap that would let malware or theft reach the consumer's device or the customer's systems.

These are promises a reasonable consumer relies on, and a responsible customer relies on too. The whitepaper is the documented basis, so we test against these responsibilities directly.

2.2. In scope and out of scope

We record the scope clearly, so a clean result is not read as approval of promises this Test did not check.

In scope: the operator's promises about running the proxy safely for the consumer and the customer: RPR-002 through RPR-011.

Out of scope:

- **RPR-001 (ethical sourcing and consent).** We already test this separately, as part of certification, so it is not part of this Test.
- **RPR-012 (granting exceptions).** This test does not measure how specific exceptions are granted.
- Billing accuracy, and marketing claims about speed, performance, or coverage.

- The security and privacy practices of the installer or companion apps.
- What unlocks at higher, vetted customer tiers.
- The full result of an endpoint takeover (we test the gate that would allow it, not the takeover itself).
- Resource drain on the consumer's device and home connection: bandwidth, data caps, battery, and speed. We do measure whether the operator caps attack-shaped bursts of requests leaving the home, and whether it passes a hostile response back to the software on the device (§3.2, bucket (b)). What we do not measure is what it costs the consumer to carry an ordinary customer's traffic.
- Testing any hosted browsers that may be provided, as this is still rare among residential proxies.
- Any extra promises an operator makes beyond these twelve responsibilities: for example, specific hosts or sites its own rules claim to block. We test against the responsibilities, not against each operator's own marketing or acceptable-use claims; those extra promises are the operator's to keep, and we do not test or report them here.

2.3. Who we test as: the least-scrutinized customer

We run every test as a **least-scrutinized customer**: an account signed up at the lowest bar the operator allows. For some operators that is just an email address and a credit card. For others, the lowest bar may itself include identity checks (KYC). Either way, we use whatever a fresh customer can get at the operator's minimum, and if a new account is given more by default than the named low tier, we test the default. This is the attacker we model: one with the least privileges an operator gives to those it lets push traffic through a consumer's connection.

3. Methodology and Strategy

The question our test answers: how much risk does this operator leave with the two parties who depend on it? Concretely, how much can a least-scrutinized customer abuse a consumer's home connection through this operator, and how much of that same failure lands on the responsible customer who bought access?

An operator has two ways to protect them. The first is to block the harmful destination or behavior, so the bad traffic never leaves the home. That protects the consumer directly, with no conditions, and it protects the responsible customer along with them. The second is to vet the customer first, so only checked customers reach risky surfaces. Vetting alone does not protect the consumer: it changes who sends the traffic, not whether it leaves the home. It becomes real protection only when three things line up, like three keys turning at once: the operator publishes the exception and the reason for it, the customer genuinely qualifies through an identity and business check, and the consumer can see how their own connection may be used for it. Without that third key, an identity check protects the operator's paperwork, not the consumer. An operator that uses neither defense has decided that anything a least-scrutinized customer wants is allowed. That is what we measure.

3.1. How we measure

For each promise we write down what we should be able to see, then try it as a least-scrutinized customer. Every check runs at least twice: once with no proxy (the baseline), then through the

operator's residential proxy consumers. The difference between the two paths is the finding. We never report a proxied result without its baseline.

We drive the operator's residential proxy with our own client, and choose the client by what a given check needs. Because the client is ours, we control exactly what goes through the operator's exit. We use three clients:

#	Client	What it is for
1	A plain HTTP request (curl-style), sent with a realistic Chrome fingerprint	Quick category and reachability checks: does the operator allow the request, does the destination return content, and where does the exit come out?
2	Our own browser (a local Chromium we drive over CDP) routed through the operator's proxy	Checks that need a real browser: the page actually loads and runs its JavaScript, with a full browser fingerprint, on the same residential exit. It is our own browser, not one the operator runs.
3	A raw socket client	Checks that need an exact, hand-built request a normal client won't send (for example, deliberately malformed message framing and attack frames), to see whether the operator filters the harmful shape or passes it through.

Whichever client we use, we run it over each interface the operator exposes: its HTTP(S) proxy and, where it offers one, its SOCKS5 port. We test both because enforcement can differ between them. For example, an operator that blocks a destination over HTTP sometimes relays the same destination over its own SOCKS5 port, a bypass using the operator's own surface.

3.2. The six things we check (buckets)

Buckets (a) to (d) are behavior checks. Buckets (a) to (c) ask what a customer can reach, and (d) asks where the exit may point; each behavior finding belongs to exactly one of them. Bucket (e) is a hands-on review of what we can't probe like traffic: the operator's published rules, its exception list, and how hardened and consistent its software is. Bucket (f) is the safe-site check, and it is graded the other way round from the rest: these are ordinary websites an operator should leave alone, so blocking one is the failure, not the pass.

Bucket	What it checks
(a) Access	Reaching whole categories of prohibited sites (e.g., Adult, Gambling, Government, Banking, Crypto, Streaming, Gaming, Ticketing, Mail).
(b) Protection	Reaching clearly harmful destinations (e.g., malware control servers, dark-web gateways, doxx/breach forums, piracy sites) plus harmful and prohibited request shapes.
(c) Bypass	Whether the operator's working blocks can be routed around with simple tricks (reflectors, URL changes, or its own SOCKS5 port). We send safe sites through the same tricks, so we can tell a block on the site from a blanket block on the trick itself.

Bucket	What it checks
(d) Egress containment	Where the exit shouldn't point (e.g., the home network and loopback, harmful ports, UDP) over transports the operator offers.
(e) Manual review	The operator's published rules (the AUP) and exception list, and whether its software is hardened, patched, and held consistent across every surface, read and judged by hand, since these can't be probed like traffic.
(f) Safe sites	Ordinary websites that no proxy should block (for example a news site or an encyclopedia), tried directly and through the same simple tricks as bucket (c). Blocking one is a failure. Published as "Safe sites".

3.3. What we can and cannot measure

Following the Guidelines, we sort each promise by how we can check it, and we say so in the report. We draw conclusions from what we measured.

- **Measured directly:** the behavior checks in buckets (a) to (d), and the safe-site check in bucket (f). We try the forbidden thing as a customer and see whether it is stopped, and we also confirm that ordinary public-web use still works (so we can spot over-blocking and test manipulation).
- **Measured by review (bucket (e)):** things we cannot probe. These are the operator's published rules (RPR-008) and exception list (RPR-009), and whether its software is hardened, patched, and held consistent across every surface (RPR-011). We judge these from documentation and from the cross-surface results, explain how, and do not present them as strongly as a direct behavior test. This review is specified but is not yet part of the published monthly report, so until it is, these three promises are listed there as not evaluated (§8, §9.3).
- **Not tested:** the full endpoint-takeover outcome (we test the gate that would allow it, but not the takeover itself). RPR-001 (ethical sourcing) is tested separately, as part of certification and Deceptor hunting, so it is not part of this Test. RPR-012 (requesting exceptions) is out of scope of this test and may be covered in the future with a certification program.

3.4. We check both kinds of failure

A fair test checks both ways an operator can fall short. Under-delivery: a responsibility requires a block or a contained exit, and a least-scrutinized customer reaches it anyway. Over-blocking: the operator blocks ordinary public-web use it should leave working (RPR-004). Our safe-site checks confirm the operator is not over-blocking normal use, which would also break the legitimate public-web work the customer bought the service for. These are the false-negative and false-positive ideas familiar from anti-virus testing.

3.5. Which test covers which responsibility

Every test traces back to a numbered responsibility. This table shows which bucket covers each; build status (live vs. still being built) is tracked in §8, not here. Bucket (f), the safe-site check, is how we measure over-blocking, so it sits under RPR-004 alongside bucket (b).

Responsibility	(a) Access	(b) Protection	(c) Bypass	(d) Egress	(e) Manual
RPR-002 Keep the address clean	✓	✓	✓		
RPR-003 No illegal traffic leaves the home		✓			
RPR-004 Reading public pages only		✓			
RPR-005 Keep the traffic pointed outward				✓	
RPR-006 Block the risky site categories by default	✓	✓			
RPR-007 Block known bypasses			✓		
RPR-008 Publish clear, readable rules					✓
RPR-009 Publish every exception, with its reason					✓
RPR-010 Wall the software off from the consumer		✓		✓	
RPR-011 Keep the software hardened, patched, and consistent					✓

4. Participation

Any tested operator can become a Participant at no cost. To adopt Participant status, an operator notifies us at info@appesteem.com; there is no fee, and being tested does not depend on it. We charge nothing to be tested, and being tested does not depend on buying anything from us. A Participant also discloses anything that would keep the Test from being accurate, such as any feature or setting that treats our test traffic differently from an ordinary customer's; the Standard calls this avoiding a golden sample. Participants may suggest which products and surfaces we should include, and unless we have a good reason otherwise we will honor those suggestions. As the AMTSO Standard allows, Participants may also provide Commentary on the results for publication on the AMTSO website.

To take part in a run, an operator may review the preliminary findings we share, audit the account state and settings we tested (including the health of our positive-control checks), and file disputes during the dispute window (§10).

4.1. An optional, paid QA service

We offer a separate, optional, paid quality-testing service, and it is built into the monthly rhythm in §6. On the first three Tuesdays of each month we run quality tests: the same style of checks as the formal Test, run as practice. We share those practice results only with the operators who pay for quality testing. That is what the fee buys: private feedback, early enough to find and fix problems before the Test that counts, and the right to challenge a practice result. On the fourth Tuesday we run the formal record Test, and its results go to every operator and are published for every operator, free. The record Test draws its sample only from the targets already used in that month's quality

tests, so an operator that pays for quality testing has seen, in advance, every target the record Test can use. This is by design. The Test is open-book (see §5.2), and its goal is to confirm the operator protects the consumer and the customer, not to catch the operator out. Paying buys practice and feedback, not a better score: every operator is scored on the same record sample in the same way, the record targets come from public category and malware lists that are open to everyone in advance, and an operator that did not pay can still pass by doing what a responsible operator should already do, all of which we publish in advance (the full list is in §5.2). Being tested by the formal Test, and having that result published, is free for every operator and does not depend on buying anything from us. The AMTSO Standard lets a Tester offer preview and feedback services on a paid basis at its discretion (Standard §6.1.9), so the quality-testing service sits within that allowance; the evidence behind a record result and the right to dispute it are never sold, and are open to every tested operator on the same terms (§10).

4.2. Withdrawal, conflicts, and funding

Withdrawing or dropping an operator. An operator may ask to be withdrawn from a Test, or to have its result left out of the public monthly Test Report. For that to take effect for a given month, the request must reach us before that month's record Test runs on the fourth Tuesday; a later request applies to the following month. We may also drop an operator ourselves, for example when we cannot keep a working least-scrutinized account with it, when its service is unavailable during the run, or when including it would make the Test unfair or unsafe. When an operator is withdrawn, dropped, or omitted, we note in that month's Test Report that a result was withheld for it, so the omission is not read as a pass. Because AppEsteem also runs certification and Deceptor programs, withdrawal from this Test does not by itself remove an operator from certification review or Deceptor consideration; it only affects inclusion in this Test's published report. If we and an operator disagree about a withdrawal, drop, or omission, AMTSO offers arbitration to help resolve it.

Conflicts of interest. AppEsteem has financial interests on both sides of this Test, and we disclose both so there is no appearance of bias:

- We certify, or want to certify, some operators under test, so we have an interest in those operators passing.
- We have placed some operators under test on the active Deceptor list, and because some may buy certification or remedy services, we have an interest in finding violations.

Funding. We fund this Test ourselves. Our business is selling certification and remedy services (and the optional QA service above). We believe that the better the operators protect the consumers they monetize, the more valuable our certification becomes, so this Test should add value for our customers and attract others to purchase our services.

5. Environment

5.1. Test machine and account

We run each test from an ordinary internet-connected computer, and we reach the operator's residential proxies the same way any customer would, using the standard access the operator gives us, its credentials and gateway, from a least-scrutinized account signed up at its minimum bar with a plan purchased by us. We are testing a service that runs over the internet: the residential proxy runs on the operator's infrastructure and on consumers' devices, not on our machine, so it does not run

on or protect the computer we test from, and we cannot see or control the state of the operator's system. We know only what we asked for, meaning the options we selected, such as the exit country or session type. Real third-party sites are needed for the access and protection checks, because category detection depends on the real brand and domain, so a fake site cannot stand in. We use the controlled site betterbrowsing.org for the browser and request-shape checks (§5.5).

5.2. Where the test sites come from

Targets are drawn from per-category lists built from trusted public sources. For example, we may use UT-Capitole blocklists ranked by Tranco for Adult and Gambling, the CoinGecko exchange list for Crypto, U.S. dotgov data plus a curated foreign-government list for Government, and live threat feeds like URLhaus and ThreatFox for the harmful-destination checks. We adjust sources and refresh these lists regularly.

Fresh quality samples, a record sample drawn from them. On each of the three quality Tuesdays we pick fresh targets from the public lists above. If we always hit the same URLs, an operator could rule-list our exact test traffic and look better than its detection really is, so a fresh quality draw each week forces the operator to enforce by category, not by URL. The fourth-Tuesday record Test then draws its sample only from the targets already used in that month's quality tests. The record Test is therefore open-book by design (see the open-book note below): an operator that pays for quality testing has already seen every target it can contain. We rely on two things instead of sample secrecy to keep this fair. First, every target comes from public category and malware lists, so an operator that blocks by category and destination, rather than by memorizing our exact URLs, is unaffected by which URL we draw; blocking these destinations is only one part of what passing takes, and the open-book note below lists the rest. Second, to earn the right to dispute a result, the operator must attest it did not cheat (see §10). We record the random seed so any run can be reproduced exactly.

Pre-flight and health checks. A scored result should reflect the operator's behavior, not the destination's own defenses. Before each run we do a plain fetch of each target and swap in an alternate if the first pick is already broken (a dead page, a challenge wall). We also mix in a set of known-good "positive control" sites (published as safe sites) so we can spot a degraded operator: if those good sites start failing, an across-the-board "block" is seen as the operator refusing everything, and not offering real protection.

As with our UwS test, this is an open-book test, and the monthly rhythm makes it more so. The categories, request shapes, bypasses, and malware lists an operator should protect against are public in advance. We publish this test plan, and we publish the betterbrowsing.org probe bundle on the site itself, along with the categories we block, how we compute them, and how we test for malicious destinations. On top of that, the record Test only uses targets already shown in that month's quality tests, so an operator that pays for quality testing sees the exact sample pool in advance. We do this on purpose. We are not trying to make the Test hard to pass. Anyone who works out how to pass it has, by definition, protected the consumer and the customer against the risks we measure. Passing takes no inside knowledge; it takes doing what a responsible operator should already do, and every part of it is public in advance. To pass, an operator needs to block the default-block website categories with a good classification system (bucket (a)); keep up with blocking known malware, command-and-control, dark-web, breach, and piracy destinations from public threat feeds (bucket (b)); make those blocks hold so they cannot be stepped around with a reflector, a hostname trick, or the operator's own SOCKS5 port (bucket (c)); keep the exit contained, so a

customer cannot point it at the home network, the device itself, or cloud-metadata, cannot open non-web ports such as mail or remote-desktop, and cannot open a UDP relay (bucket (d)); and limit non-vetted customers to public-web retrieval rather than relaying the logins, sign-ups, posting, credential injection, and malformed request shapes we exercise on betterbrowsing.org. An operator that does these things should pass, whether or not it pays to take the quality tests.

5.3. Geography

We plan to test against various default exit countries. We may start with the United States only, and move on from there to other countries. We'll grow by adding a small multi-country sample (one US, one EU, one non-Western) to catch geography-dependent differences.

5.4. What we share

Each run produces a per-operator findings report. For every check it records the target, the bucket, the surface tested, the country, the no-proxy baseline and the proxied result side by side, the verdict, and a link to the saved evidence (response bodies, screenshots, exit location). It also records the customer account we used to reach the operator, meaning the account handle and plan for the least-scrutinized customer we signed up (§2.3). We make the report and its evidence available to the tested operator, and we publish the customer account we used for each operator as part of our results, so anyone reading a result can see the exact identity the operator was serving when we measured it.

5.5. The controlled live site

Our betterbrowsing.org is a controlled site, run independently of this framework, that emulates real-website behavior: login forms, real rate-limit responses, ad-click trackers, and diagnostic endpoints. It lets us demonstrate web requests against a safe, purpose-built target instead of harming a real site (§7.5). We download the test bundle fresh from the site at the start of every run, so findings match what any tester with no prior access would see.

6. Schedule

A standing monthly test. This is a continuous, monthly Test. It commences on 1 September 2026; the first record Test runs on the fourth Tuesday of September 2026, and a record Test runs on the fourth Tuesday of each month after that. We give notice of the Test once, before it starts, on the AMTSO website and to operators. After that it runs every month without a new notice each time, as the Standard allows for a standing test. We produce a monthly Test Report that contains all operators tested. If we make a significant change to this plan, we will give fresh notice. This version makes one: §9 now describes the figures we publish next to the kept-and-broken record, which an earlier version of this plan said we would not compute. We are giving fresh notice of that change. Apart from it, this plan matches how we intend to run going forward.

Each month follows a fixed rhythm tied to the calendar. The first three Tuesdays are quality tests: practice runs whose results we share only with the operators who pay for quality testing. The fourth Tuesday is the record Test: the official run, for every operator. The record Test draws its sample only from the targets used in that month's quality tests. In the operator portal and in the published report these are named the way operators see them: the quality tests are practice tests, numbered 1 to 3, and the record Test is the monthly test. The table below lists the monthly steps:

Index	Activity	Timing
1	Quality tests (practice; results shared with paying operators)	First three Tuesdays of the month
2	Review of account configuration	Alongside each run
3	Record Test (official, all operators); sample drawn from that month's quality tests	Fourth Tuesday of the month
4	Preliminary record results shared; dispute window opens	After the record Test completes
5	Disputes close	Three weeks after the record Test
6	Final monthly Test Report	Once the dispute window has closed

Communications. We will give notice about any significant change to this schedule, including around observed holidays not listed by AMTSO.

Risks. The main risks are list drift and destination anti-bot noise (handled by fresh sampling, pre-flight, and the positive-control checks), and free-tier limits on the test account, which cap how much volume we can test. We have no other specific known risks.

7. Control Procedures

7.1. Always compare against a baseline

Every proxied check has a no-proxy baseline, and the difference between them is the finding. Bucket (d) is the one exception: probing the home/loopback space from our own network would test our own LAN, not the consumer's, so its baseline is a set of public-host checks that prove the operator makes tunnels at all before a refusal to a private address can count as a real gate.

7.2. Credit the operator, not the destination

We credit the operator with a block only when the block comes from the operator (its own block page, its error message, its transport-level refusal). When the response comes from the destination's own defenses (a normal page, a CAPTCHA, a 403 or 429 from the site), the operator passed the request through, and it does not get credit for the site's own defenses.

7.3. Health checks

We read the positive-control checks (the safe sites) first. If they fail with no proxy, our own network or the site is the problem. If they fail through the operator, the operator is in a degraded state (suspended account, billing issue, blanket block), and any "blocked" findings in that run are suspect.

7.4. Keep the run from looking like a test

So a run does not stand out as a test, we mix in ordinary, benign traffic, vary the timing, randomize the order of targets, and reuse the test account over time instead of making a fresh one each run. This keeps an operator from recognizing a test session and serving us better answers than an ordinary customer would get.

7.5. Tester conduct and public safety

In line with the AMTSO principle that testing must not endanger the public, the Test is built to do no harm:

- We show that a capability is open, not that an attack succeeded. We exercise forbidden actions against the controlled betterbrowsing.org site, not against a real victim.
- We use only the access an ordinary least-scrutinized customer has, and rely on no special rights that would make the results unrepresentative.
- We never make a real consumer's device do anything harmful, illegal, or unconsented. Home-network and egress checks go only through the operator's exit, never into our own network.
- We use intrusive techniques sparingly, avoid polluting shared community systems, and handle any personal data lawfully and minimally.

7.6. Evidence

We keep the evidence behind each verdict until all disputes for that run are closed. That evidence includes response bodies, status messages, screenshots, exit location, and the sampled target set with its seed.

8. Dependencies

What we ask of operators. The Test depends on a least-scrutinized account being able to sign up and run like any other customer. If it wants to take part, an operator reviews the preliminary results, audits the account and settings we tested, and files any disputes within the window (§4, §10).

Some described tests are not yet running, because they depend on extra infrastructure or other work. We will work on adding them into the the test, but we list them so the boundaries of the initial Test are clear:

- The credential-relay check inside a hosted browser (RPR-004). The plain version and the graded version against operators that open up encrypted traffic are both built and have been run; what is left waits on the hosted-browser surface, which is out of scope for now.
- Full proof of a completed connection on non-web ports (RPR-005), which needs a multi-port test server the basic gate does not.
- Agent/MCP-skill checks (RPR-011), which need per-operator self-serve keys and a check of where that surface exits.
- Two harder versions of the home-network gate (RPR-005): a web address that resolves into the home network, and the same private addresses written in other number formats. The plain gate is live; these two are not.
- The hands-on review, bucket (e). The published monthly report carries the behavior results and the safe-site check; the review of the operator's published rules (RPR-008), its exception list (RPR-009), and its software hardening (RPR-011) is not yet part of that report.

9. Reporting Results

What we report. For each operator we report, responsibility by responsibility, which promises it kept and which it broke, and we name the customer account we tested it as (see §5.4). That per-promise record is the result of the Test, and it is never netted: a kept promise does not cancel a broken one.

Next to it we publish figures built from those same results, because a reader comparing operators needs one honest way to weigh a handful of broken promises against a handful of others. The two figures follow the white paper's two sets of risks: its Section 3 risks for the consumer, and its Section 4 risks for the customer. The figures are: the share of risk handled in each area and overall; a figure for how well the operator protects the person whose internet is used; and a figure for how well it protects the paying customer. Each check carries a weight by the kind of site it covers (none, low, medium, or high), the same weights for every operator, and a figure is the share of the weight an operator was given that it handled. Every operator appears in one table with the same figures, so results can be compared directly. We do not name a winner, publish an award, or fold the two figures into a single number. The AMTSO Guidelines for Testing Application Promises ask for a documented scoring discipline, applied equally to every application. Ours is the weighting described above: it is written down here, published with the results, and applied the same way to every operator, whatever its own rules happen to name. The per-promise record stays next to the figures, so a reader can always see which kept or broken promises a figure came from.

9.1. How we record each result

Each check ends in one of these outcomes:

- **Kept or broken.** For each check we record whether the operator kept the promise (it blocked or contained what the responsibility requires) or broke it, meaning a least-scrutinized customer got through. Every operator is checked against the same ten responsibilities under test and the same lists: the RPR-006 default-block categories in bucket (a), the harm list in bucket (b), bypass resistance in bucket (c), egress containment in bucket (d), and the safe sites in bucket (f), identical for every operator, regardless of what its own rules happen to name.
- **Bypassed, and bypass vulnerable.** A block that one of our simple tricks got past is published as Bypassed. A block that no trick got past, but that held only because the operator blocks one of those tricks for safe sites too, is published as Bypass Vulnerable: the operator is blocking the trick rather than the site, and it cannot keep doing that once safe sites stop working. Each of the two counts as half a block.
- **Inconclusive.** We could not reach a verdict, because the site or the network did not answer normally. This is our measurement problem, never the operator's: it is shown in the report and counted neither for nor against the operator, and we work to drive it toward zero.
- **Not counted.** A result we removed after a successful dispute, or a kind of site we no longer measure. It stays visible in the report, marked, so the removal is not silent.
- **Not measured.** A whole test in which the proxy let none of our safe sites through, because it answered nothing or refused everything. Nothing in such a test can be read as a block or an allow, so no figure is published for it.
- **Test quality.** A broken baseline, a broken test target, or a safe-site failure is flagged on its own and does not count against the operator; these drive list hygiene. Some kinds of site have been dropped from the Test altogether and are neither measured nor scored, in current or older results: other proxy apps that resell home internet, and snapshot archives as a way around a block, because a frozen copy of a page is not access to the live site.

9.2. Under-delivery and over-blocking

We report both ways an operator can fall short, responsibility by responsibility. Under-delivery: a responsibility requires a block or a contained exit, and a least-scrutinized customer got through anyway, which is a broken promise. Over-blocking: the operator blocks ordinary public-web use it should leave working (RPR-004), which the safe-site check in bucket (f) catches, directly and through the simple tricks. These are the false-negative and false-positive ideas familiar from anti-virus testing. We never net one against the other: the safe sites an operator blocked are published as their own count, next to the figures, and they are left out of them. A high count is also the first thing a reader should weigh, because an operator that blocks safe sites may be blocking everything, which makes its other blocks mean less.

9.3. Reporting and what happens next

Every published conclusion is limited to what we measured. We report directly measured results, indirectly measured results, and not-tested promises separately; we never present an indirect result as strongly as a direct one, and we list a not-tested promise as not evaluated. A certified operator that breaks these promises faces certification re-evaluation or revocation; an operator we do not certify that advertises ethically sourced exits while leaving them abusable is Deceptor-list material. We re-check any promise that came out as a violation before we publish it.

10. Dispute Process

Disputes are free and open to every tested operator via <https://proxytest.appesteem.com>. To earn the right to dispute, an operator first claims their account and we approve. The operator signs a short attestation that it did not cheat on the Test. In that attestation the operator affirms that it let our customer run at the least-scrutinized default permissions, with no special or different treatment from an ordinary customer of the same tier, and that it did not try to influence the Test results for itself or for any other operator. An operator that will not attest to this can still be tested and published, but cannot file a dispute. We provide a channel where operators can review run results, see their Test Reports, and dispute findings. We accept disputes against a monthly Test Report for three weeks after the record Test runs, and each operator's results page shows the date its window closes. In the operator portal a dispute is called a challenge. We ask that each dispute come with proof or evidence that it is legitimate, not just a statement of disagreement. AppEsteem keeps an active contact point for all testing-related correspondence, and commits in good faith to respond to operator questions and disputes within a reasonable time. Operators can reach the Test team through the contact details AppEsteem keeps current on the AMTSO Contact List, and at info@appesteem.com.

Disputes can be made when:

- The sampled site was bad, corrupted, or no longer applicable. Successful disputes here are removed from all operators' results.
- We misread the result. Successful disputes are removed from that operator's results.
- The check ran improperly, because of tester or environment error (for example, a broken baseline or a positive-control failure). Successful disputes are removed from that operator's results.
- The sampled site did not actually belong to the responsibility category we tested it under, for example, a site we placed in an RPR-006 default-block category that does not in fact belong there.

- The responsibility we tested against was mis-stated, or was changed in the whitepaper after the version we tested. A check judged unsuitable after a dispute is removed from all operators, to keep things balanced.
- The operator told us in good faith that it needed extra time to review and enforce against a sample.

11. Attestations

I understand and agree that I am submitting this Test Plan, and the following Attestations, on behalf of the entity listed below, and I represent and warrant that I have authority to bind such entity to these Attestations. All references to “I” or “me” or similar language refer to such entity. I represent and warrant that the following Attestations are true, to the best of my knowledge and belief, and each of the following commitments will be upheld to the best of my ability.

- I will provide public notification on the AMTSO website covering my obligation for notification of a Public Test, regardless of whether a potential Participant is in actual receipt of such notification prior to the Commencement Date of a Test. (Section 1, Section 4, Section 6)
- All products included in this Test will be analyzed fairly and equally. (Section 2, Section 3, Section 5)
- I will disclose any anticipated or known imbalance or inequity in the Test design to all Participants in the Test. (Section 2, Section 3)
- Although I may charge for participation in a Test, I will not charge any additional fees for a vendor to be a test subject under the Standards. (Section 4)
- I will disclose any material conflicts of interest or other information that could materially impact the reliability of the Test. (Section 4)
- I will disclose how the Test was funded, and any other services I offer that a vendor could access. (Section 4)

I hereby affirm, to the best of my knowledge and belief, that this Test Plan complies with the AMTSO Testing Standards, as of the date hereof.

Signature: _____

Name: Dennis Batchelder

Test Lab: AppEsteem Corporation

AMTSO Test ID: [AMTSO-LS1-TP215]