ProxyLook

FREE EBOOK · 2026 EDITION

The Multi-Account Handbook

Antidetect browsers, cloud phones and proxies, working together.



6
chapters on keeping accounts separate

3
layers: browser, phone and proxy

0
sign-ups, gates or affiliate links

proxylook.com/ebooks

FREE EBOOK · 2026 EDITION

The Multi-Account Handbook

Antidetect browsers, cloud phones and proxies, working together.

Published by ProxyLook · proxylook.com · First published November 2026

ProxyLook is an independent directory of proxy networks and antidetect browsers. This book is free. Share it with anyone, unchanged and with credit to ProxyLook.

Written and edited by the ProxyLook editorial team. Tool details come from the ProxyLook antidetect directory as it stood in November 2026. Many platforms limit or forbid multiple accounts. The rules differ by platform, and the operator carries the risk.

Prices, pools and features change often. This book links to live figures on proxylook.com instead of printing numbers that go stale. Vendor-stated figures are labelled as such. The book contains no affiliate links.

Nothing here is legal advice. Laws and platform rules differ by country and by service. Check the rules that apply before acting.

© 2026 ProxyLook. Product names belong to their owners.

START HERE

How to use this book

An agency runs forty client accounts from one office. A seller manages three storefronts. A QA team checks ads from six countries. Each of them needs the same thing: accounts that stay cleanly separate.

This book explains how platforms connect accounts to each other, and how browser profiles, cloud phones and proxies keep them apart. It is a companion to The Proxy Playbook, which covers proxies in depth.

The chapters build on each other. The first explains the signals. The next three cover the tools: antidetect browsers, cloud phones and the proxies behind them. The last two cover choosing a tool and running accounts day to day.

43	6	7	8	6
PAGES	CHAPTERS	DIAGRAMS	TABLES	CHECKLISTS

A word on limits. No tool makes an account safe from enforcement, and this book does not pretend otherwise. Separation removes the accidental links between accounts. What happens on the accounts is still up to the operator, and so is knowing each platform’s rules.

CONTENTS

What is inside

PART I · THE PROBLEM

01 **Why Accounts Get Linked** 5

How platforms connect accounts / The six signals in detail / What a proxy covers and what it does not / Legitimate reasons to keep accounts separate / The rules come first

PART II · THE TOOLS

02 **Antidetect Browsers Explained** 11

What a browser profile is / What a fingerprint contains / How an antidetect browser separates profiles / Chromium and Firefox engines / Local profiles, cloud profiles and team sharing / What these tools cannot do

03 **Cloud Phones and Mobile Identities** 16

Why a phone is not a small browser / Cloud phone, emulator or real handset / What a cloud phone identity includes / How the traffic flows / Metered running time / Cloud phone or browser profile

04 **Pairing Proxies with Profiles** 21

One identity, one IP / Static ISP, residential or mobile / Sticky sessions and the cost of rotation / Location, timezone and language must agree / Checking for leaks / Setting up one profile end to end

PART III · THE PRACTICE

05 **Choosing a Tool** 27

Start from the work / Eight criteria / Ten tools in the ProxyLook directory / Browser profiles, cloud phones or both / Running a fair trial

06 **Running Accounts Day to Day** 32

A naming and record-keeping system / Starting new accounts gradually / Team workflow, roles and handover / Security / Automation with restraint / When an account is restricted

REFERENCE

A **Appendix** 38

New-profile checklist / What must agree / Weekly hygiene checklist / Glossary / ProxyLook tools and further reading

CHAPTER 01

Why Accounts Get Linked

The signals platforms use to connect accounts, which of them a proxy covers, and the legitimate reasons to keep client and brand accounts apart.

- Six signals that link accounts
- Legitimate reasons for separation
- What a proxy covers and misses
- Platform rules and operator risk

A social media manager signs in to three client accounts from one office laptop before lunch. To the platform, those three brands now share a device, a browser and an IP address. If one of them is ever restricted, the other two are standing right beside it.

How platforms connect accounts

Platforms link accounts for sound reasons. Linking is how they catch spam rings, payment fraud and people who return after a ban. The same systems also sweep up honest operators who work carelessly.

No platform publishes its full rulebook, and the details change without notice. What can be described is what a website or app is technically able to see. Those observations fall into six groups, stacked from the network up to the content itself.

1 · Network	The IP address and the ASN that owns it.
2 · Device and browser	The fingerprint: graphics, fonts, screen and settings.
3 · Stored state	Cookies, local storage and remembered logins.
4 · Behavior and timing	When sessions happen and what they do.
5 · Payment and contact	Cards, phone numbers, emails and addresses.
6 · Content	Images, captions, links and listings.

Six layers a platform can compare between two accounts. One match is a hint. Several matches are a link.

A single shared signal rarely settles anything. Hundreds of strangers share a coffee shop's IP address, and a great many people own the same phone model. Confidence builds when several layers match at once.

The six signals in detail

IP address and ASN

Every request arrives from an IP address. That address belongs to a network with an Autonomous System Number, or ASN. The ASN names the owner: a home broadband provider, a mobile carrier or a hosting company. Two accounts that always appear from the same address look related. A Texas shop's account that connects from a hosting network abroad looks odd even with no second account involved.

Device and browser fingerprint

A browser answers many small questions for any page that asks. It reports which fonts are installed and how the graphics card draws a test image. It reports the screen size and the timezone of the clock. Together the answers form a fingerprint that stays much the same from visit to visit. Chapter 2 takes it apart piece by piece. Mobile apps can read even more, which is the subject of chapter 3.

Cookies and local storage

Sites leave small files and stored values in the browser. When an operator signs out of one account and into another, those values are still there. This is the most direct link of all, and the easiest to create by accident.

Behavior and timing

Five accounts that post in the same minute every morning behave like one operator. So do accounts that follow the same people in the same order, or always log in one after another. Scheduling tools and copy-and-paste routines create these patterns without anyone meaning to.

Payment and contact details

A card number, a billing address, a recovery phone number or an email is a hard identifier. Two ad accounts paid by one card are connected, whatever the browser says. For an agency the fix is simple and honest: the client's own card and contact details belong on the client's account.

Content

The same product photo, caption or tracking link posted from several accounts ties them together. Image files can also carry metadata such as the camera model and the editing software. Regional accounts that share one asset library will look related. That is fine when they are openly the same brand.

Signal	What it reveals	What keeps it separate
IP address and ASN	Location and type of network	One dedicated proxy IP per identity
Browser fingerprint	The device and browser in use	One browser profile per identity
Cookies and storage	Earlier logins in the same browser	Isolated profile storage, never shared
Behavior and timing	One operator working many accounts	Separate schedules and a human pace
Payment and contact	The person or company behind the account	Each client's own genuine details
Content	Shared assets, links and wording	Assets prepared per account

Each signal, what a platform learns from it, and the practice that keeps it apart.

What a proxy covers and what it does not

A proxy sits between the browser and the platform and swaps the visible IP address. That takes care of layer one. It does nothing for the five layers above it. Readers who want a refresher on proxy types will find it in The Proxy Playbook.

A proxy covers

- The IP address the platform sees
- The ASN and network type behind it
- The country and city that address maps to

A proxy does not cover

- The browser or device fingerprint
- Cookies and stored logins
- Timing, payment details and content
- Browser features that leak the real IP

> **A proxy changes the return address on the envelope, not the handwriting inside.**

Consider an operator who runs three accounts through three proxies in one ordinary browser. One layer out of six has changed. The cookies sit side by side, and the fingerprint is identical on all three.

The reverse mistake is as common. Three carefully separated browser profiles that all exit through the office connection share layer one. The What Is My IP tool shows the address, location and ASN a site sees from any given session.

The remaining layers need other tools and habits. An antidetect browser handles fingerprint and storage, as chapter 2 explains. A cloud phone does the same job for app-first platforms. Behavior, payment details and content are down to how the team works. Tools cannot supply those.

Legitimate reasons to keep accounts separate

Most multi-account work is ordinary business. The common cases look like this.

- **Agencies with client permission.** A team of four manages forty client accounts. Each client expects its account to stay apart from the other thirty-nine.
- **Brands with regional accounts.** A retailer runs accounts for the United States, Germany and Japan, each with local staff and local content.
- **Sellers with several storefronts.** Some marketplaces allow more than one store, usually with conditions. A seller with two distinct product lines may qualify.
- **QA and ad verification.** Testers need clean sessions in specific countries to see what a local visitor sees. The ad verification page covers the proxy side.
- **Containment.** If one client's login is phished, separation stops the damage at one profile.

Many platforms also provide an official route for this work, such as business manager tools, agency partner access or team seats. Where an official route exists, it is the first choice. Separate profiles and proxies keep sessions tidy. They do not replace permission.

KEY IDEA **Separation means consistency**

The aim is not disguise. Each account should look like what it is: one business, on one device, in one place, behaving steadily over time. Every later chapter applies that single idea to a different layer.

The rules come first

Many platforms limit how many accounts one person or business may hold. Some forbid more than one outright. Others allow several but ban using them to dodge limits, inflate engagement or return after a ban. The rules differ by platform and by account type, and they change.

Running multiple accounts can breach a platform's terms of service. The operator carries that risk, not the tool vendor and not the proxy provider. The usual penalties are restriction, suspension, withheld funds or the loss of every linked account. Reading the current terms for each platform is part of the job. For agencies it also belongs in the client contract.

WATCH OUT **No setup prevents bans**

Antidetect browsers, cloud phones and proxies reduce accidental links between accounts. They do not make any account immune to enforcement, and no honest vendor promises that. An account that breaks a platform's rules can be restricted however carefully it was set up.

This handbook stays on the permitted side of that line. It does not cover getting around a ban or creating accounts in bulk. It leaves out anything that depends on false identity or payment details. The tools it describes are listed in the ProxyLook antidetect browser directory. The same warning applies to every one of them.

KEY TAKEAWAYS

- Platforms compare six layers: network, fingerprint, stored state, behavior, payment and contact details, and content.
- One shared signal is a hint, and several shared signals together are a link.
- A proxy changes the IP address, ASN and apparent location, and nothing else.
- Fingerprint and storage need separate profiles, while behavior, payment and content depend on working habits.
- Legitimate separation rests on permission: client consent, official access routes and each platform's current terms.
- Multi-accounting can breach terms of service, the operator carries the risk, and no setup prevents bans.

CHAPTER 02

Antidetect Browsers Explained

What a browser profile and a fingerprint are, how an antidetect browser keeps each one separate and consistent, and where these tools stop helping.

- What one browser profile holds
- Engines, local and cloud profiles
- The eight parts of a fingerprint
- Team sharing and hard limits

A fresh Chrome profile on an office laptop starts with empty cookies, but the machine behind it has not changed. The same graphics card, fonts and screen are on show to every site. An antidetect browser exists to close that gap, one profile at a time.

What a browser profile is

A profile is a self-contained browser identity. Opening the profile named `acme-ig-us-01` brings up a window with its own cookies, fingerprint and proxy. When it closes and the next profile opens, nothing carries over.

Ordinary browsers offer profiles too, and private windows. Both separate storage. Neither changes the fingerprint, so every profile on the machine still describes the same device. An antidetect browser separates both.

1 · Fingerprint settings	The device this profile presents to sites.
2 · Cookies and storage	Logins, preferences and site data, kept apart.
3 · Proxy assignment	The one IP address this profile exits through.
4 · Extensions and bookmarks	Add-ons and saved pages for this account only.
5 · Notes and tags	Client, owner and status, for the team.

What one antidetect browser profile holds. Everything in the stack travels together and stays out of other profiles.

The stack matters because the parts depend on each other. A fingerprint that says Windows in Chicago needs a proxy in or near Chicago. Chapter 4 covers that pairing.

What a fingerprint contains

A fingerprint is not one value. It is a set of ordinary answers a browser gives to any page that asks. Each answer is dull when taken alone. Combined, they narrow a visitor down to a small crowd, sometimes to one device.

Component	What the page reads	Why it varies
Canvas	A hidden image the browser is asked to draw	Graphics hardware and drivers render slightly differently
WebGL	Graphics card vendor, model and a 3D test	Depends on the GPU and its driver
Fonts	Which typefaces are installed	Operating system and installed software differ
Audio	How the browser processes a test sound	Small differences in the audio stack
Screen	Resolution, color depth and pixel ratio	Monitor and display settings
Timezone	The system clock's zone, such as America/Chicago	Where the device is set up
Language	Preferred languages, such as en-US	System and browser settings
User agent	Browser name, version and operating system	Which browser build is installed

The eight fingerprint components this chapter refers to, and where each one comes from.

The user agent is the easiest of the eight to inspect. Pasting one into the user-agent parser shows the browser, version and operating system it claims. The other seven are harder to see, and that is why they are more useful to a platform.

Three of them come from the machine's hardware: canvas, WebGL and audio. They are the hardest to separate without a dedicated tool. Changing a timezone takes a click. Changing how a graphics card draws an image does not.

How an antidetect browser separates profiles

When a profile is created, the tool picks a set of fingerprint values and saves them. Every later launch reuses the same set. Two rules decide whether the result is any good.

Separate. The values differ from every other profile in the workspace. Client A and client B no longer present the same device.

Consistent. The values agree with each other and stay the same over time. A profile that claims Chrome on Windows should list Windows fonts and a graphics card that Windows machines actually use. A profile with a Windows user agent and Mac fonts is stranger than an honest laptop.

> **KEY IDEA** **Consistent beats random**
>
> A returning customer does not arrive on a different device every morning. A profile that reshuffles its fingerprint on each launch looks less like a person, not more. The values are best set once at creation and then left alone.

Vendors reach this result in different ways, and the quality of the outcome varies. A feature list does not reveal it. Chapter 5 suggests how to judge fingerprint quality during a trial.

Chromium and Firefox engines

Every antidetect browser is built on a real browser engine. Most use Chromium, the open-source base of Chrome. That makes sense, because Chrome-family browsers are what most desktop visitors use.

Some vendors give their Chromium build a name. Multilogin calls its core Mimic, and GoLogin calls its core Orbita. Others ship a Firefox option alongside. AdsPower has SunBrowser on Chromium and FlowerBrowser on Firefox. Kameleo has Chroma on Chromium and Junglefox on Firefox.

Two practical points follow from the engine.

- **The claim must match the engine.** Chromium and Firefox behave differently in ways a page can test. A Chromium core should present as Chrome, and a Firefox core as Firefox.
- **The version must be current.** Real browsers update themselves every few weeks. A profile stuck on a core from last year stands out, so vendor update speed matters.

Firefox profiles are a reasonable choice for accounts whose real audience skews that way, or for QA on Firefox. For most client work, a current Chromium core is the plain default.

Local profiles, cloud profiles and team sharing

Profiles have to live somewhere. The two options trade convenience against control.

Local profiles

Profile data stays on the operator's computer. Nothing leaves the machine, but the profile is tied to it. A lost laptop means a lost session.

Cloud profiles

Profile data syncs to the vendor's servers. Any approved teammate can open the same profile from another computer. The team is trusting the vendor with session data.

Cloud sync is what makes team sharing work. The login session lives inside the profile, in its cookies. A teammate who opens a shared profile is already signed in. No password is pasted into a chat window, and nobody needs to know the client's password to do a day's posting.

Roles then limit who can open, edit, export or delete each profile. When a contractor leaves, an admin removes one seat instead of changing forty passwords. Sensitive accounts should still have their passwords rotated on a schedule.

One habit keeps shared profiles healthy: one person in a profile at a time. Two copies of the same session running in two places is the pattern of a hijacked account. Most tools lock a profile while it is open. Teams should confirm that during a trial.

What these tools cannot do

An antidetect browser covers two of the six layers from chapter 1: fingerprint and stored state. The limits are worth stating plainly.

- It cannot change behavior. Ten profiles posting the same caption at 9:00 are still one operator.
- It cannot separate payment details, phone numbers or emails that are shared between accounts.
- It cannot fix a poor proxy. A clean fingerprint behind a shared or flagged IP is still exposed at layer one.

- It cannot stand in for a phone. App-first platforms read device signals a browser does not have.
- It cannot stop a browser feature from leaking the real IP unless that feature is configured. The WebRTC leak test checks the most common case.
- It cannot make an account immune to enforcement or guarantee that a fingerprint passes every check.

It also cannot make a forbidden account allowed. Running multiple accounts can breach a platform's terms of service, and the operator carries that risk whatever software is in use. Tool attributes change often, so the live details sit in the ProxyLook antidetect browser directory.

KEY TAKEAWAYS

- A profile bundles a fingerprint, isolated storage and one proxy into a single identity.
- A fingerprint combines canvas, WebGL, fonts, audio, screen, timezone, language and user agent.
- Good profiles are separate from each other and consistent within themselves over time.
- The engine should match the browser the profile claims to be, on a current version.
- Cloud profiles let a team share sessions without sharing passwords, one operator at a time.
- These tools cover fingerprint and storage only, and they do not change what a platform permits.

CHAPTER 03

Cloud Phones and Mobile Identities

Why app-first platforms need a phone and not a browser, how cloud phones compare with emulators and handsets, and when each tool is the right one.

- Why apps see more than browsers
- Metered running time as a model
- Cloud phone, emulator or handset
- Choosing phone or browser profile

A retail brand's regional TikTok account is created on a phone, posts from a phone and goes live from a phone. A desktop browser profile can sign in, but the desktop site does not offer everything the app does. Accounts that live in an app are better served by something that is actually a phone.

Why a phone is not a small browser

A web page runs inside a sandbox. It can ask the browser a fixed set of questions and nothing more. An app is installed on the device and sits much closer to it.

Depending on the permissions the operating system grants, an app can read things a page cannot:

- The device brand, model and Android version.
- Device and advertising identifiers that persist between sessions.
- Carrier and SIM details, and whether the connection is mobile data or Wi-Fi.
- Sensor readings, battery state and GPS location.
- Whether the device and the app pass integrity checks, such as Google's Play Integrity.

A browser profile can present a mobile user agent and a phone-sized screen. That produces the mobile website, not the app. For platforms that put their main features in the app, the difference is practical before it is technical. Live streaming, some seller tools and some messaging features may exist only there.

History matters too. An account that has only ever appeared from a phone has a pattern. A sudden move to a desktop browser is a change in that pattern, and changes invite verification prompts.

Cloud phone, emulator or real handset

There are three ways to give an account a phone. A small agency often starts with a drawer of handsets and stops when the drawer is full.

Option	What it is	Works well for	Main drawback
Real handset	A physical phone on a desk	A few high-value accounts	Does not scale; handover means posting a parcel
Emulator	Android simulated on a desktop computer	App testing and development	Apps can often tell it is not a phone
Cloud phone	An Android device hosted by a vendor	Many app accounts shared by a team	Ongoing fees and trust in the vendor

Three ways to run an app-based account, and what each costs in practice.

An emulator is built for developers. It typically runs on desktop hardware, lacks real sensors and carries build details that mark it as an emulator. That is fine for checking a layout. It is a weak home for a long-lived client account.

A cloud phone is an Android environment that runs on a vendor's servers and streams its screen to the operator. Taps and typing travel back the other way. Vendors differ in the hardware underneath and in how complete the device looks to an app. Those are questions for the vendor's documentation and for a trial.

Three tools in the ProxyLook directory illustrate the range. DuoPlus offers Android cloud phones only, opened from a browser tab. GeeLark offers Android cloud phones plus browser profiles. Multilogin offers browser profiles plus Android cloud phones.

What a cloud phone identity includes

A cloud phone is the mobile version of a browser profile. It bundles a device description, stored app data and a network exit into one identity. The typical contents are:

- **Device description.** Brand, model, Android version and screen size.
- **Identifiers.** The device and advertising IDs that apps read.
- **Carrier and SIM details.** The network operator and country the device reports.

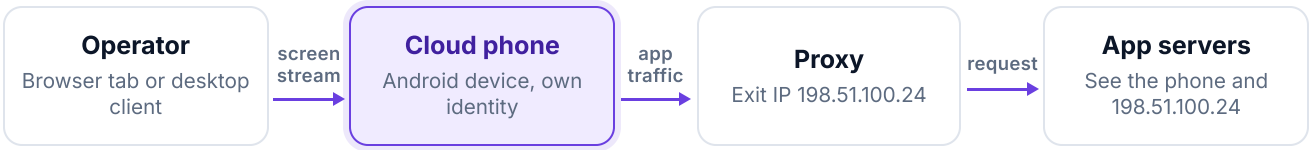
- **Location settings.** GPS position, timezone and system language.
- **Apps and their data.** Installed apps, logins and caches.
- **Network.** The proxy the phone sends its traffic through.

Which of these can be set, and which are fixed, depends on the vendor. The rule from chapter 2 carries over unchanged. The values must agree with each other, and they must stay the same for the life of the account. A phone reporting a French carrier, a Texas GPS position and a Tokyo timezone contradicts itself three times.

> **KEY IDEA** **One account, one phone**
>
> A cloud phone is best treated like a handset issued to a staff member. The account stays on that phone. Moving it to a new device every week recreates the pattern of an account being passed around.

How the traffic flows

Two separate connections are in play. Mixing them up causes most setup mistakes.



The operator watches and controls the cloud phone. The app's traffic leaves from the phone through its proxy, not from the operator's computer.

The first connection is the screen stream between the operator and the vendor. The platform never sees it. An operator can work from the office, from home or from a train, and the app's view does not change.

The second connection is the app's own traffic. It leaves the cloud phone and should pass through a proxy assigned to that phone. Without one, the traffic exits from the vendor's own network. A phone that claims a mobile carrier but connects from a hosting network breaks the consistency rule at layer one.

A mobile proxy is the natural match, because phones normally connect through carriers. A residential IP also fits, since phones spend much of the day on home Wi-Fi. Chapter 4 covers the choice. Either way, the phone's GPS position, timezone and language should follow the proxy's location.

Metered running time

Browser profiles are usually sold by the number of profiles. Cloud phones often add a second meter: time. The vendor charges for the minutes a phone is switched on. GeeLark meters running time. DuoPlus charges a per-device fee plus metered running time.

The model rewards short, deliberate sessions. Consider two accounts. One needs twenty minutes a day to post and answer comments. The other hosts a three-hour live stream twice a week. Under a time meter, the second uses roughly three times the running time of the first, on identical phones.

Three habits keep the bill predictable.

- Shut each phone down when the session ends. A phone left running overnight is billed overnight.
- Estimate before buying: phones, multiplied by minutes per day, multiplied by working days.
- Review usage monthly and retire phones for accounts that are no longer managed.

Rates change often and are not printed here. Current figures are on each tool's page in the antidetect browser directory.

Cloud phone or browser profile

The choice follows the platform and the task, not the tool's popularity. A browser profile is usually cheaper to run and quicker to work in. A cloud phone is the better fit only where the app is the real product.

If...	Then...
The platform works fully in a desktop browser	A browser profile is enough
The account was created in the app and lives there	Use a cloud phone
A needed feature exists only in the app	Use a cloud phone
QA needs to check a mobile web page	A browser profile with a mobile setup will do
QA needs to see in-app ads in another country	Use a cloud phone with a proxy there
A client has both web and app accounts	Use both, under one record per account

A quick test for choosing between a browser profile and a cloud phone.

A cloud phone changes the device layer and nothing else. Behavior, payment details and content still link accounts, exactly as in chapter 1. App-first platforms also tend to have strict account rules. Running multiple accounts can breach their terms of service, and the operator carries that risk. A cloud phone does not make an account immune to restriction.

KEY TAKEAWAYS

- Apps can read device details that web pages cannot, so app-first accounts need a phone identity.
- Emulators suit testing, handsets suit a few accounts, and cloud phones suit teams with many.
- A cloud phone bundles device details, identifiers, location settings, app data and a proxy.
- App traffic leaves from the cloud phone through its proxy, never from the operator's own connection.
- Metered running time rewards short sessions and switching phones off afterward.
- A browser profile is enough unless the account or the feature lives in the app.

CHAPTER 04

Pairing Proxies with Profiles

How to give each profile one fitting IP address, keep location settings in agreement with it, check for leaks, and set up a profile from start to finish.

- One identity, one IP address
- Matching timezone and language
- Static ISP, residential or mobile
- Leak checks and setup steps

A client's account has signed in from the same Dallas address every weekday for a year. One morning it appears from Frankfurt, and ten minutes later from Madrid. The fingerprint has not changed at all, and the platform still asks for a verification code.

One identity, one IP

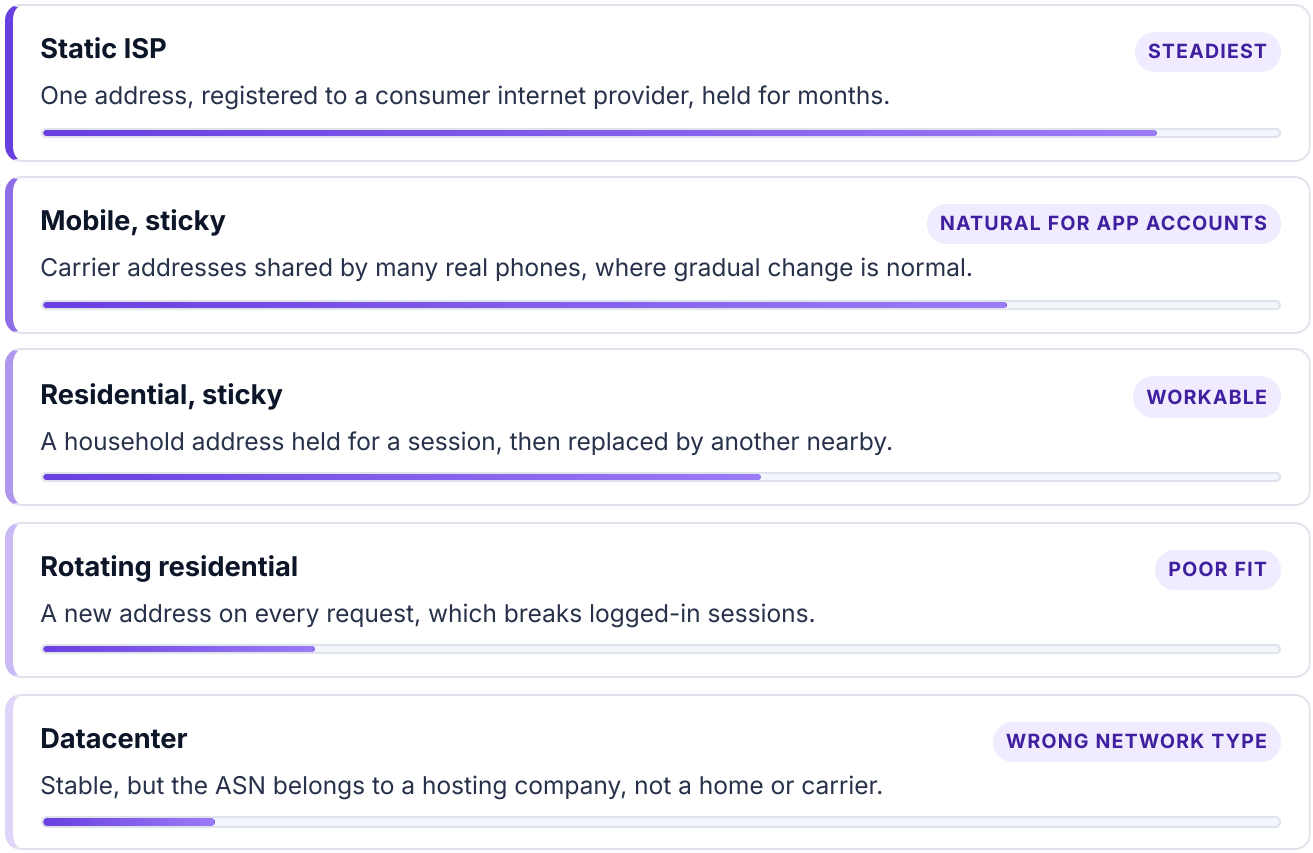
The rule is short. Each profile or cloud phone gets its own proxy, and keeps it. Two accounts never share an address, and one account does not hop between addresses.

Both halves matter. Sharing an IP between two profiles rebuilds the layer-one link that the profiles were meant to remove. Swapping a profile's IP every few days makes a steady account look like a traveler with no fixed home.

The simplest way to keep the rule is to label the proxy with the profile's name. The proxy for `acme-ig-us-01` is called `acme-ig-us-01-isp` and is used nowhere else. Dedicated proxies, which are assigned to a single customer, make this easy to enforce.

Static ISP, residential or mobile

The Proxy Playbook explains how each proxy type works. For long-lived, logged-in accounts the question is narrower: which type behaves most like one real customer's connection over months?



Relative fit for a long-lived, logged-in account. The order is a rule of thumb, not a measurement.

Static ISP addresses are the usual choice for desktop accounts. They are hosted in data centers but registered to consumer internet providers. The account sees the same address every day, as a home or office customer would. They are sold per IP, and the ISP proxies page lists providers.

Mobile addresses suit cloud phones and app-first accounts. Carriers place many subscribers behind each public address, and a phone's address shifts as it moves. An account on a sticky mobile session in one country and carrier matches how phones really behave.

Residential addresses come from household connections that can drop offline at any time. A sticky session holds one for a while, then hands over another. That is acceptable when the replacement stays in the same city and provider. It is less tidy than a static address.

Datacenter addresses are not useless. They are a sound, cheap choice for QA on public pages where nothing is logged in. Live pricing for every type is on the ProxyLook pricing page.

Sticky sessions and the cost of rotation

Rotation was built for scraping. Each request leaves from a new address so that no single address draws attention. For a logged-in account it does the opposite.

One session cookie arriving from five cities in a minute is a familiar sight to a security system. It is what a stolen session looks like. The likely results are a forced logout, a verification challenge or a temporary lock. None of them is a ban, but each one costs time and adds a mark to the account's history.

A sticky session asks the provider to hold one address for a set period. Providers usually set this through the proxy username, in a form like `USERNAME-country-us-session-a1b2c3`. The exact syntax comes from each provider's documentation.

> **TIP** **Set the session longer than the shift**
>
> If an operator works in a profile for two hours, a ten-minute sticky session will rotate a dozen times. The duration should outlast the work session, and the profile should be closed before it expires. For accounts opened daily, a static address avoids the question.

Location, timezone and language must agree

A proxy that exits in Dallas sets expectations. The profile's clock should read `America/Chicago`. Its language should be `en-US`. On a cloud phone, the GPS position should sit in or near Dallas. A Dallas address with a Berlin clock and a German language list contradicts itself.

Many antidetect browsers can fill in timezone, language and location from the proxy's address when the profile is created. The option is worth using, and the result is worth checking once by eye.

QA and ad verification teams apply the same rule in code. The script below opens a page as a visitor in Dallas would see it. The proxy, locale and timezone are set together so the three agree.

PYTHON · PLAYWRIGHT

```python
from playwright.sync_api import sync_playwright

PROXY = {
    "server": "http://gate.example-proxy.com:7000",
    "username": "USERNAME-country-us-session-a1b2c3",
    "password": "PASSWORD",
}

with sync_playwright() as p:
    browser = p.chromium.launch(proxy=PROXY)
    context = browser.new_context(
        locale="en-US",
        timezone_id="America/Chicago",
    )
    page = context.new_page()
    page.goto("https://example.com", timeout=30000)
    page.screenshot(path="dallas-view.png")
    browser.close()
```

The username format is illustrative. A plain script like this suits checks on public pages. It is not a substitute for a full browser profile on a client's logged-in account.

Checking for leaks

A leak is any path by which the real connection shows through the proxy. Two are common, and both can be checked before an account is ever opened.

WebRTC. Browsers use WebRTC for voice and video calls. To set up a call, it can reveal the device's real IP address to a page, bypassing the proxy. Antidetect browsers have a WebRTC setting per profile. The WebRTC leak test, opened inside the profile, should show the proxy's address or nothing. It should never show the office address.

DNS. Before loading a site, a browser looks up its name. If that lookup goes to the office internet provider while the traffic exits in Dallas, the two do not match. The fix is configuration. With an HTTP proxy, the proxy resolves names itself. With SOCKS5, the option that sends DNS through the proxy should be switched on.

The What Is My IP tool confirms the basics from inside the profile: public address, location, provider and ASN. If any of those show the office, the proxy is not applied.

Setting up one profile end to end

The table gives a starting point for the proxy. The steps then take one account from an empty record to a working profile.

Account type	Proxy type	Reason
Client social account, worked on desktop	Static ISP in the client's country	Same address every working day
App-first account on a cloud phone	Mobile, sticky session	Matches a phone on a carrier
Marketplace storefront	Static ISP, dedicated	Accounts that hold funds need the steadiest address
Client ad account	Static ISP, dedicated	A steady address suits accounts that spend money
Regional brand account	Static ISP or sticky residential in that region	Address agrees with the audience
QA and ad verification, not logged in	Residential, by test country	Coverage matters more than stability

- A starting match between account type and proxy type. Platform rules and budgets may change the choice.

With the proxy type chosen, the setup follows the same order every time.

1. **Write the record first.** Note the client, platform, profile name and intended location before creating anything.
2. **Assign one proxy.** Label it with the profile name and confirm no other profile uses it.
3. **Create the profile.** Pick an operating system and a current engine version, then save the fingerprint.
4. **Enter the proxy.** Set timezone, language and location to follow the proxy's address.
5. **Run the checks.** Inside the profile, open What Is My IP and the WebRTC leak test.
6. **Sign in once.** Complete any verification with the client, who holds the codes.
7. **Close and log.** Record the date, the exit address and the person responsible.

A matched proxy removes one avoidable link between accounts. It does not change what a platform allows. Multi-accounting can breach terms of service, and the operator carries that risk.

KEY TAKEAWAYS

- Each profile or cloud phone gets one proxy that no other identity uses.
- Static ISP addresses suit desktop accounts, and sticky mobile addresses suit app-first accounts.
- Rotation in the middle of a logged-in session resembles a stolen session and triggers checks.
- Timezone, language and location should follow the proxy's address.
- WebRTC and DNS settings need checking inside the profile before the first sign-in.
- A repeatable setup order, starting with the written record, prevents most pairing mistakes.

CHAPTER 05

Choosing a Tool

Eight criteria for comparing antidetect browsers and cloud phones, a scorecard to fill in, and a plain introduction to ten tools in the ProxyLook directory.

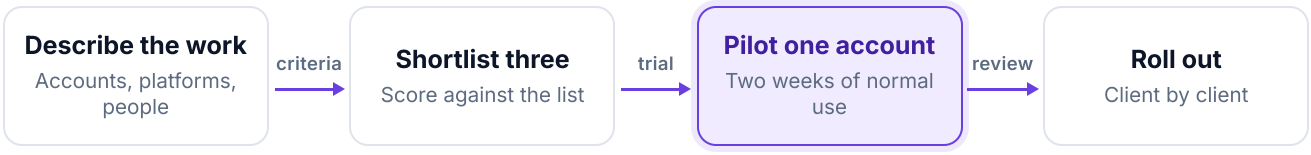
- Eight criteria that matter
- Ten tools and their listed attributes
- A scorecard to fill in
- Browser profiles, cloud phones or both

A two-person agency wants a low entry cost and a tool that can be learned in an afternoon. A forty-seat ad team wants roles, access logs and an API. Both say they need an antidetect browser, and the same product is rarely right for both.

Start from the work

Feature lists are long and look alike. A short description of the actual work cuts through them. Five lines are enough: how many accounts, which platforms, web or app, how many people, and what should be automated.

An example: "Forty accounts across Instagram and TikTok, half of them app-first. Four staff and one contractor. Scheduled posting only, no scripted engagement." That description already rules out a browser-only tool and puts team features near the top.



- A selection process in four stages. The pilot on one low-stakes account is where most of the learning happens.

The pilot matters most because it tests the things a product page cannot show. Those include fingerprint consistency over several days, handover between two people, and how support responds.

Eight criteria

Engine

Chapter 2 covered Chromium and Firefox cores. The questions here are which engines a tool ships and how quickly the vendor follows new browser releases. A team that only needs Chrome-like profiles can ignore Firefox support entirely.

Fingerprint quality

This is the hardest criterion to judge and the one that matters most. No feature list settles it. During a trial, the same profile should be opened on three different days and its reported values compared. The values should be identical each time and should agree with each other.

Automation support

QA and ad verification teams usually script their checks with Selenium, Puppeteer or Playwright. A tool that supports those libraries lets existing scripts drive its profiles. RPA is the no-code alternative: recorded steps replayed inside a profile. Teams that click everything by hand can score this criterion low.

Team features

Seats, roles, profile sharing, transfer and activity logs. A solo operator needs none of them. An agency with a contractor needs all of them from the first day.

Cloud phones

Chapter 3 set out when an account needs a phone. If any managed account is app-first, either the tool includes cloud phones or a second tool has to.

Free plan

A free plan allows a real trial without a deadline. It is useful for learning and for very small operations. Limits on profiles and seats usually arrive quickly, so the paid tiers still need reading.

Security

The tool holds live sessions for every client. Two-factor authentication on the tool's own login is the minimum. Encryption of stored profile data and a clear statement of who can access it come next. Vendors that publish this plainly are easier to trust.

Pricing model

Models differ more than prices do. Some tools charge by profile count, some by seat, and cloud phones may charge by running time. The useful comparison is the full monthly cost of the described work, not the headline number.

Criterion	What to check	Weighs most for
Engine	Chromium, Firefox or both; update speed	Teams with Firefox audiences or QA
Fingerprint quality	Same values on day one, three and ten	Everyone
Automation	Selenium, Puppeteer, Playwright, RPA, API	QA and ad verification teams
Team features	Roles, sharing, transfer, activity logs	Agencies and contractors
Cloud phones	Included, separate or absent	App-first accounts
Free plan	Profile and seat limits	Solo operators and trials
Security	2FA, encryption, access policy	Anyone holding client sessions
Pricing model	Per profile, per seat or per minute	Growing teams

- A scorecard to copy. Score each shortlisted tool from one to five per row, then weight the rows that fit the work.

Ten tools in the ProxyLook directory

The cards below introduce ten tools, in alphabetical order within each group. No ranking is implied. Each card states only what the ProxyLook directory records as of November 2026, and each name links to its page.

AdsPower
Two engines: SunBrowser on Chromium and FlowerBrowser on Firefox. Includes built-in RPA and a window synchronizer.

Dolphin Anty
Chromium based. A free plan is available.

GoLogin
Runs a Chromium core called Orbita. A free plan is available.

ixBrowser
Chromium based, with a free plan.

Kameleo
Two cores: Chroma on Chromium and Junglefox on Firefox. Supports Selenium, Puppeteer and Playwright.

Octo Browser
Chromium based, with API access.

Those six are desktop browser tools. The next four are different in kind: three involve Android cloud phones, and one is an open-source project for scripted work.

Camoufox

A free, open-source Firefox fork built for Playwright automation.

DuoPlus

Android cloud phones only, opened from a browser tab. Charges a per-device fee plus metered running time. From the team behind AdsPower.

GeeLark

Android cloud phones plus browser profiles. Running time is metered.

Multilogin

Browser profiles plus Android cloud phones. The Chromium core is called Mimic. Supports Selenium, Puppeteer and Playwright, and has a free plan.

KEY IDEA **Not listed does not mean missing**

The cards repeat only the attributes held in the directory. A feature that is not mentioned may still exist. Plans, limits and prices also change often. The live pages in the antidetect browser directory are the place to confirm details.

Browser profiles, cloud phones or both

The split from chapter 3 decides the shape of the purchase before any brand does.

- **Web accounts only.** Any of the browser tools can be shortlisted. Cloud phone support can be scored at zero.
- **App-first accounts only.** A phone-only product such as DuoPlus is a candidate, as are the tools that offer both.
- **A mix of both.** One tool that covers both, such as GeeLark or Multilogin, keeps records and billing in one place. Two specialist tools can also work, at the cost of two sets of seats and logs.

A mixed setup needs one rule either way. Each account has one home, a profile or a phone, and the register says which. An account opened from both on alternate days presents two devices.

Running a fair trial

A trial is only useful if it copies real work. The steps below fit into two weeks.

1. **Pick a low-stakes account.** Use an in-house brand account, or a client account with the client's written agreement.
2. **Build the profile properly.** Follow the setup order from chapter 4, with a dedicated proxy.
3. **Check consistency.** Record what the profile reports on three separate days and compare the notes.
4. **Test a handover.** Have a second person open the profile from another computer and confirm the lock.
5. **Read the exit terms.** Find out how profiles are exported if the team later moves to another tool.
6. **Contact support once.** Ask a real question and note how long a useful answer takes.

No tool on any shortlist makes an account immune to restriction. Multi-accounting can breach a platform's terms of service, and the operator carries that risk whichever product is chosen. A good tool removes accidental links and keeps the team organized. ProxyLook's scoring approach for the directory is published on its methodology page.

KEY TAKEAWAYS

- A five-line description of the work filters tools faster than any feature comparison.
- Fingerprint quality matters most and can only be judged in a trial over several days.
- Automation, team features and cloud phones matter a great deal to some teams and not at all to others.
- Pricing models differ, so the fair comparison is the monthly cost of the described work.
- Web-only work needs a browser tool, app-first work needs cloud phones, and mixed work needs both.
- The ten tools are described by their directory attributes only, and live pages hold the current details.

CHAPTER 06

Running Accounts Day to Day

The routines that keep client accounts orderly: naming, records, gradual starts, team roles, security, restrained automation and a calm response to restrictions.

- Naming and the account register
- Roles, handover and security
- Gradual starts for new accounts
- Responding to a restriction

At six on a Friday evening a client asks who signed in to its account last Tuesday, and from where. A team with a register answers in two minutes. A team without one starts scrolling through chat history.

A naming and record-keeping system

Good operations start with a boring name. A pattern such as client, platform, region and number is enough: `acme-ig-us-01`. The same name goes on the browser profile, the proxy label and the password manager entry. Anyone on the team can then trace one account across all three.

The register is a single shared sheet with one row per account. It answers three questions at a glance: which profile, which proxy, which client. A fuller record looks like this.

YAML · ACCOUNT REGISTER

```yaml
- profile: acme-ig-us-01
  client: Acme Outdoor
  platform: Instagram
  permission: services agreement, signed 2026-03-02
  home: browser profile
  proxy_label: acme-ig-us-01-isp
  proxy_type: static ISP
  proxy_exit: 203.0.113.45, Dallas, United States
  timezone: America/Chicago
  language: en-US
  owner: Priya
  backup: Tom
  two_factor: authenticator app, client holds backup codes
  created: 2026-03-04
  last_review: 2026-11-06
  incidents: none
```

Two fields deserve a comment. The `permission` line points to the document in which the client authorized the work. The record holds no password. Passwords live in the password manager and nowhere else.

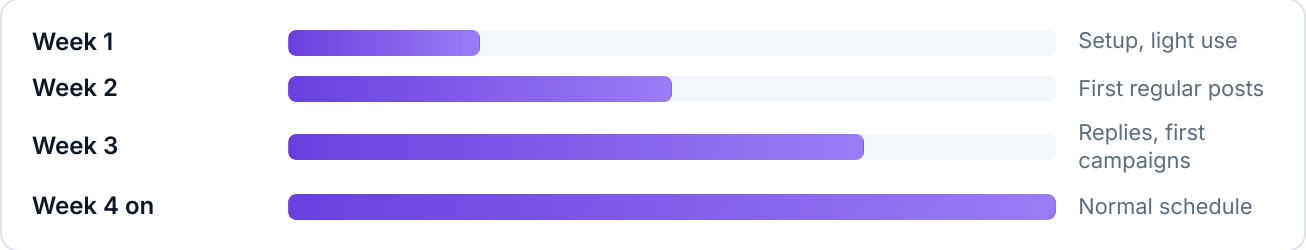
A register only works if it is edited at the moment of change. A proxy swapped on Wednesday and recorded on Friday leaves two days in which the sheet is wrong. The rule that holds up in practice is short: no change to a profile, proxy or owner without a matching edit to its row.

Starting new accounts gradually

A retailer opens a new account for its German market. On day one, the sensible work is setup: genuine business details, a verified email and phone number, a profile photo and a short description. Launching ads and fifty posts in the first hour is not how a real regional team behaves.

Platforms tend to watch new accounts more closely than established ones. Ordinary, gradual activity is the honest answer to that, because a new account run by a real team starts slowly anyway.

Genuine details matter more than pace. The email, phone number and business information should belong to the client or brand that owns the account. An account built on borrowed or invented details is fragile from the first day, whatever the setup around it.



An illustrative ramp for a new account. The pace is a working habit, not a platform rule or a measured threshold.

The ramp applies to single accounts opened for a real purpose. Creating accounts in bulk, or opening spares in case one is lost, is a different activity. Most platforms forbid it, and this handbook does not cover it.

Team workflow, roles and handover

Most account trouble inside agencies is caused by people, not fingerprints. Two operators open the same profile. A contractor keeps access for months after a project ends. Roles in the antidetect tool prevent both.

Role	Can do	Should not do
Admin	Create profiles, assign proxies, manage seats	Work in client profiles day to day
Account manager	Open and edit profiles for assigned clients	Export or delete profiles
Operator or contractor	Open assigned profiles only	Change proxies or fingerprint settings
Reviewer	Read the register and activity logs	Open any profile

A four-role model for a small agency. Role names and options vary between tools.

Handover needs a routine as well. The outgoing operator closes the profile, writes one line in the register and names the next person. The incoming operator reads that line before opening anything. One person works in a profile at a time, as chapter 2 explained.

When someone leaves, the admin removes the seat the same day. The same goes for clients. When a contract ends, the agency hands back access, deletes the profile and releases the proxy.

Security

An agency's antidetect workspace is a set of signed-in sessions for every client it serves. It deserves the care given to a bank login.

- [ ] Turn on two-factor authentication for every platform account, with an authenticator app where offered.
- [ ] Leave backup codes with the client, who owns the account.
- [ ] Turn on two-factor authentication for the antidetect tool and the proxy dashboard as well.
- [ ] Keep passwords in a password manager, in one vault per client.
- [ ] Never send a password or a code through chat or email.
- [ ] Review the tool's access log weekly for unexpected openings.
- [ ] Give each person the lowest role that lets the work get done.

The access log is the most neglected item on that list. Five minutes a week is enough to spot a profile opened at an odd hour. The same check catches one opened by someone who has moved to another client.

Automation with restraint

Automation is safest where it replaces chores. Scheduling approved posts through a platform's own tools is a chore. So is pulling weekly reports, or checking that a landing page loads in six countries. The ad verification page covers proxies for that last case.

Automated engagement is another matter. Scripted follows, likes, comments, messages and reviews are restricted or banned on most platforms. They also produce exactly the timing patterns described in chapter 1. This handbook treats them as out of bounds.

Window synchronizers and RPA deserve the same care. Repeating one click across ten client profiles in the same second creates ten identical timestamps. Used for QA across regions, that is harmless. Used for posting, it ties the ten accounts together.

> **TIP** **A simple test for any automation**
>
> Two questions are enough. Does the platform's current policy allow it? Would the client be comfortable reading a description of it? If either answer is no, the task stays manual or is dropped.

When an account is restricted

Restrictions happen to careful teams too. What matters is a calm, documented response through the platform's own channels.

1. **Stop activity.** Pause scheduled posts and any automation on the affected account.
2. **Read the notice.** Save a screenshot and note the stated reason, the date and any deadline.
3. **Tell the client.** Report the same day, with the notice attached.
4. **Appeal through the platform.** Use the official appeal or review form, with accurate information. The account owner should submit it.
5. **Leave the setup alone.** Do not open a replacement account or switch proxies to get around the restriction.
6. **Record the outcome.** Add the incident to the register and review what led to it.

Step five is where teams make things worse. A replacement account opened during a restriction can turn a temporary limit into a permanent one. It can also draw in the client's other accounts. If the appeal fails, the decision stands. The next conversation is with the client, not with a new profile.

> **WATCH OUT** **The compliance reminder**
>
> Many platforms limit or forbid multiple accounts, and the rules differ from one platform to the next. Running them can breach terms of service, and the operator carries that risk. Nothing in this handbook prevents bans. It only removes accidental links between accounts that are allowed to exist.

Policies change, so a yearly reading of each platform's terms belongs on the team calendar. The ProxyLook pages for social media and e-commerce proxies cover the network side. Tool details are in the antidetect browser directory.

KEY TAKEAWAYS

- One naming pattern across profile, proxy and password manager makes every account traceable.
- The register records profile, proxy, client, permission and owner, and never a password.
- New accounts start with genuine details and gradual, ordinary activity.
- Roles, single-operator handover and same-day removal of leavers prevent most internal mistakes.
- Automation belongs on chores such as scheduling, reporting and QA, not on engagement.
- A restricted account is handled through the platform's own appeal process, and the operator carries the risk throughout.

APPENDIX

Appendix

Two checklists, a stack of settings that must agree, a glossary of sixteen terms, and the ProxyLook tools and books that support the handbook.

- New-profile checklist
- Glossary of sixteen terms
- Weekly hygiene checklist
- ProxyLook tools and further reading

The six chapters reduce to a few lists that belong beside the account register. This appendix collects them: one checklist for a new profile, one for the weekly review, a glossary and the tools worth bookmarking.

New-profile checklist

Run this list once for every new browser profile or cloud phone, before the first sign-in.

- [ ] Confirm the client's written permission and the platform's current account rules.
- [ ] Add a row to the register with the profile name, client and owner.
- [ ] Assign one dedicated proxy and label it with the profile name.
- [ ] Check the proxy's country, provider and network type with the proxy checker.
- [ ] Create the profile on a current engine version and save the fingerprint.
- [ ] Set timezone, language and location to follow the proxy's address.
- [ ] Open What Is My IP inside the profile and confirm the proxy's address appears.
- [ ] Run the WebRTC leak test inside the profile.
- [ ] Sign in once, with two-factor authentication on and backup codes held by the client.
- [ ] Store the password in the password manager, then close the profile and log the date.

What must agree

Most avoidable verification prompts trace back to two settings that disagree. The stack below lists what should tell the same story for a single account.

1 · Proxy address	Country, city and network type.
2 · Timezone	The zone that city actually uses.
3 · Language	A language list that fits the country.
4 · Device claim	Operating system, engine, fonts and graphics that belong together.
5 · Location settings	GPS and carrier details on a cloud phone.
6 · Account details	Business address, phone number and payment country.

- Six things that should agree for one account. A mismatch at any level is worth fixing before the first sign-in.

> **KEY IDEA** **Set once, then leave alone**
>
> Agreement on day one is half the job. The other half is keeping the same values for the life of the account. A profile that is edited every month is a profile that drifts.

Weekly hygiene checklist

Fifteen minutes on a fixed day covers the list for a small agency.

- [ ] Compare the register with the tool: every profile has a row, and every row has a profile.
- [ ] Confirm each proxy is still assigned to one profile only.
- [ ] Spot-check two profiles for exit address and WebRTC behavior.
- [ ] Read the access log for unexpected openings or unfamiliar devices.
- [ ] Remove seats for anyone who left or changed clients.
- [ ] Shut down cloud phones left running and review metered time.
- [ ] Check for proxy expiry dates in the coming two weeks.
- [ ] Note any verification prompts or restrictions in the register.

- [ ] Delete profiles and release proxies for clients whose contracts ended.

Glossary

The terms below are used throughout the handbook. Definitions are kept to one line.

Term	Meaning
Antidetect browser	A browser that gives each profile its own fingerprint and storage
ASN	The number identifying the network that owns an IP address
Browser profile	One isolated identity: fingerprint, cookies, storage and proxy
Canvas fingerprint	A value derived from how a device draws a hidden image
Cloud phone	An Android device hosted by a vendor and controlled remotely
Cookie	A small file a site stores to remember a session
Dedicated proxy	An IP address assigned to a single customer
DNS leak	Name lookups that bypass the proxy and reveal the real network
Emulator	Software that simulates a phone on a desktop computer
Fingerprint	The combined device and browser traits a site can read
Mobile proxy	An IP address belonging to a mobile carrier
RPA	Recorded, no-code steps replayed inside a profile
Static ISP proxy	A fixed IP in a data center, registered to a consumer provider
Sticky session	A proxy setting that holds one IP for a set time
Two-factor authentication	A second proof at sign-in, such as an app code
WebRTC leak	A browser call feature exposing the real IP behind a proxy

- Sixteen terms used in this handbook, in alphabetical order.

ProxyLook tools and further reading

The three free tools linked in the first checklist cover most routine checks. The proxy checker reports country, ASN, provider and network type for an address. What Is My IP shows what a site sees from the current session. The WebRTC leak test looks for the real address behind a proxy.

For tools and deeper reading:

- The antidetect browser directory holds live details for the ten tools in chapter 5 and the rest of the list.
- The Proxy Playbook is the flagship book. It covers proxy types, protocols and buying decisions in depth.
- Scraping Without Blocks is the companion for data collection teams.

One reminder closes the book. Many platforms limit or forbid multiple accounts, and running them can breach terms of service. The operator carries that risk. These lists keep permitted accounts tidy. They do not prevent bans.

KEY TAKEAWAYS

- Every new profile gets the same ten checks before its first sign-in.
- Proxy address, timezone, language, device claim, location settings and account details should agree.
- A fifteen-minute weekly review keeps the register, seats and proxies accurate.
- The live directory and the two companion books hold the detail this handbook leaves out.

KEEP GOING

Where to go next

Tools change faster than a book can. The live directory has current plans, features and reviews.

Everything below is free and needs no account.

Antidetect directory

Every browser and cloud phone tool ProxyLook lists, with live plans and features.

`proxylook.com/antidetect-browsers`

Provider directory

Proxy networks to pair with profiles, filtered by type and location.

`proxylook.com/all-providers`

WebRTC leak test

Check that a profile is not exposing the real IP.

`proxylook.com/tools/webrtc-leak-test`

What is my IP

Confirm the exit IP, its location and its network.

`proxylook.com/tools/what-is-my-ip`

The Proxy Playbook

The flagship guide to choosing, setting up and running proxies.

`proxylook.com/ebooks/the-proxy-playbook`

Scraping Without Blocks

A developer’s guide to proxies, fingerprints and reliable data collection.

`proxylook.com/ebooks/scraping-without-blocks`

ProxyLook

Independent research on proxy networks and antidetect browsers. Free to read, free to share.

proxylook.com