



GHOST
PROTOCOL

CF

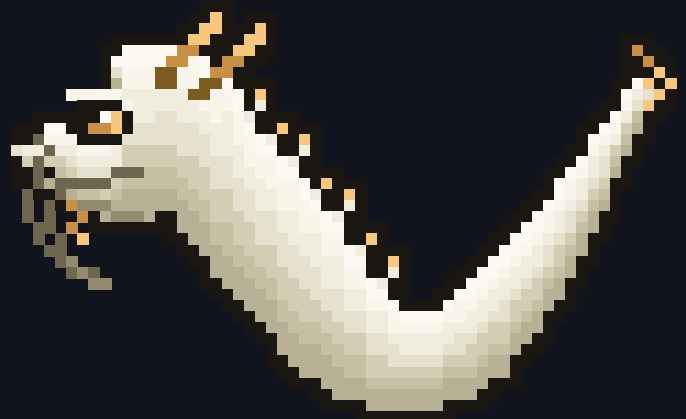
THE FIELD LIBRARY

COMPANION

COMPOSITE, AND TRUE TO LIFE

Case Files

Five composite, fictional cases, each teaching one lesson of the library through a story.



Ghost Protocol (Pvt) Ltd · Colombo · ghosts.lk

Security internship training · The Field Library · Edition 1 · 2026

THE GHOST PROTOCOL FIELD LIBRARY · COMPANION

Case Files

Five composite, fictional cases, each teaching
one lesson of the library through a story.

Ghost Protocol (Pvt) Ltd · Colombo

The Field Library, Edition 1 · 2026

Copyright 2026 Ghost Protocol (Private) Limited. All rights reserved.

Sold as a PDF for the buyer's own reading. Please do not share or resell the file.

The library

The Field Library is a whole shelf, not one book. The Core Path is the spine everyone reads. The rest is depth and breadth, taken on as the work calls for it. The shelf is complete now, and you are reading the one marked here.

A · THE CORE PATH

O The Oath BUILT

I Groundwork BUILT

II The Web BUILT

III The Wider Field BUILT

IV The Work BUILT

V The Craft BUILT

B · DEEP TRACKS

AI and Agents BUILT

Reading Code BUILT

Internal and Active Directory BUILT

The Web, Deeper BUILT

Cloud and Containers BUILT

Mobile and APIs BUILT

Reverse Engineering BUILT

Hardware and the Edges BUILT

C · THE OPERATIONS

Red and Purple Team BUILT

OSINT and Recon BUILT

Report Writing Mastery BUILT

The Consulting Craft BUILT

Bug Bounty BUILT

D · THE OTHER SIDE

Detection and Response BUILT

Threat Hunting and Forensics BUILT

Hardening and Architecture BUILT

E · THE GROUND RULES

The Law, Deeper BUILT

Standards and Compliance BUILT

Ethics and Disclosure BUILT

F · COMPANIONS

The Reference Pack BUILT

The Lab Workbook BUILT

Case Files YOU ARE HERE

The Bestiary BUILT

The Glossary and the Ledger BUILT

Contents

- Opening 5
- 1** The Free Scan 6
- 2** The IDOR 10
- 3** The Agent That Acted 14
- 4** The Key in the Metadata 17
- 5** The Temptation 20
- Worked answers 24

Stories, not records



Blink found the folder marked Case Files and got excited, expecting the real thing: named clients, real engagements, the actual war stories.

BLINK "Finally. The real ones. Who was it? What did we find?"

SOL "None of these are real in that way, and that is the point. Every case here is composite and invented: the lesson is true, the client is not, the names are nobody's. We do not turn a real engagement into a story with a client's name on it, ever, because a client's trust does not end when the test does. So read these for the lesson. The lesson is real. The rest is a teaching device."

Wyrms' eye made sure of the one thing that mattered: no real name anywhere in the folder.

WYRM "no names. real lessons."

So the book is five cases, each a story that carries one lesson from the shelf, from the free scan that became an engagement to the temptation a tester was right to decline.

CHAPTER ONE

The Free Scan

A free external look surfaced a real problem on a prospect's public surface, and the right handling turned it into a clean, paid engagement. The lesson is how a free scan is run, and the written yes it must never start without.

This volume teaches by story. Each case is one engagement told end to end, with the lesson pulled out at the close and tied back to the volume that teaches it in full. You are the new ghost on the watch, and the first case is the quietest one: a prospect you have not met, a free scan, and a finding that most people would have fumbled into a crime.

■ READ THIS FIRST

Every case in this book is fictional and composite. The companies, people, domains, and numbers are invented, and any real engagement that taught a lesson has been dissolved into the pattern so that no client, person, or piece of real work is identifiable. The placeholder domains (`client.example` and the like) are reserved for documentation and resolve to nothing. The lessons are real. The stories are not.

The prospect

Halverstone Freight is a regional freight firm, invented for this case, that runs a customer portal for booking and tracking. Ghost Protocol reached out and offered the thing the house leads with: a free external scan, a short and non-intrusive look at the public surface, given without a price attached, that tells a company one true thing it did not already know. Volume IV, The Work, is where the whole approach lives, and its first rule governs this case: lead with the free scan, keep the price quiet, and let the finding do the talking.

Before any of that could run, one thing had to arrive, and it is the part interns skip. Someone with authority over Halverstone's systems said yes to the free scan, in writing. That written yes is not the full Test Authorisation a paid engagement needs. It is the smaller permission that lets even a free, passive scan begin at all. Until it landed in your inbox, the only thing you were allowed to do was the thing any member of the public may do: open their website in a browser and read it. That is not a test. Everything past it waited for the yes.

What a Free scan is, and is not

A free scan is not a pentest with the invoice removed. With the written yes in hand, it is still strictly the view a stranger on the internet already has: public DNS records, the certificate transparency logs that list every hostname a certificate was issued for, the pages a browser loads without a login, the files a web server offers up on its own. You touch nothing that needs a credential, you send nothing a normal visitor would not, and you stop at the edge of the public surface. The yes authorised a look, not an intrusion. Under Sri Lanka's Computer Crime Act, taught in Volume 0, access without authorisation is a criminal offence, and the word "free" changes none of that.

THE PERMISSION LADDER

MORE PERMISSION ↑

Active testing	needs a signed Test Authorisation
Run the free scan	needs a written yes from someone with authority
Visit the public site	any visitor may; this is not a test

What you may do at each stage, and the written permission each one needs.

What the scan found

The certificate transparency logs listed a hostname the main site never linked to: `ops-staging.halverstone.example`. A certificate had been issued for it, so it existed and was public. You put it in a browser. It loaded a login page for an internal operations tool, the kind of dashboard a dispatcher uses to see every booking, every route, every customer address. It was sitting on the open internet with nothing in front of it but a username box. The web server was also misconfigured to list its own directories, so a visitor could see the folder tree, and one folder was named in a way that suggested database exports.

That is a real finding, reached entirely from a stranger's seat. An internal tool that should live behind a VPN or an allowlist was reachable by anyone who read a public log. You had not logged in, probed, or downloaded a thing. The value was already in your hands: this is exposed, here is how a stranger finds it, here is why it matters.

Blink, who had been reading over your shoulder, was already typing.



not ours.

Blink's fingers were on `admin` / `admin`. One green eye opened. The written yes authorised a passive look at the public surface, nothing more. Trying a credential against that login, even a guess that fails, is an attempt at access that the free-scan yes never granted, and the law does not grade the attempt by whether it worked. The login page being reachable is the finding. Typing into it is the offence. And the folder that looked like database exports stays unopened for the same reason: you can report that it appears to be exposed, from the directory listing a visitor is shown, without ever pulling a file that is not yours.

The clean note, and the engagement

What went to Halverstone was short and plain. One page: here is a hostname of yours that is public, here is the internal login and the directory listing anyone can see, here is the risk in a sentence a

manager understands, and here is the single fact that we looked only at what any visitor sees and stopped there. No price. No fear. An offer to walk them through it.

They called back the same week, because the finding was true and the note had not tried to sell them anything. From that call came the thing that makes everything afterwards legal: a signed Test Authorisation that named the hosts in scope, the window, and the contact for stop-and-report. The free-scan yes had let you look. This fuller paper is what let you test. Only then, with it in hand, did the staging tool become something you were allowed to log in to and probe.

■ THE LINE

Ryan put the rule in one place, and it governs every first contact: "No scan of any client or prospect, not even the free scan, until someone with authority over the target has said yes in writing. Visiting a prospect's public website the way any visitor would is not a test; anything more waits for that written yes."

Read it twice. The free scan is not exempt because it is free, and not because it is passive. The only thing exempt is what any member of the public may do: open a website and read it. The yes must come from someone with authority over the target, not a friendly contact who lacks it, and the fuller active test that follows needs its own written Test Authorisation. Volume 0 is the whole of the law here, and it does not bend for a prospect you are keen to win.



SOL SAYS

The free scan is not bait and it is not a favour. It is the cleanest introduction in this trade, because it gives before it asks and it proves the house can be trusted with the dangerous thing first. Notice what won Halverstone: not a scary number, not a discount, one true finding handled with restraint. A prospect can tell the difference between a firm that found a real problem and a firm that manufactured a worry to sell a fix. Be the first kind. The quiet note that keeps the price out of the first sentence is the one that gets the call back.

■ ON YOUR FEET

You are running an authorised free scan on a different prospect, the written yes already in hand, when it turns up a public `.git` folder on their marketing site, downloadable by anyone, that would let a visitor reconstruct the site's source. You have not downloaded it. Your colleague says, "pull it and grep it for secrets, then we will really have something to show them." What do you put in the note to the prospect, what do you refuse to do before the full Test Authorisation is signed, and why is the refusal the thing that makes the finding worth more, not less?



WHAT WYRM KEEPS

A free scan runs only after someone with authority over the target has said yes in writing. It then sees only what a stranger already sees, and stops there. The finding is the lead. The written yes is what makes the look lawful.

- No scan of any client or prospect, not even the free scan, until a written yes arrives from someone with authority over the target. Visiting the public website as any visitor would is not a test; anything more waits for that yes. Volume 0 is the law.
- With the yes in hand, a free scan is public surface only: DNS, certificate transparency, pages a browser loads, files a server offers. No logging in, no credential guesses, no downloading what a visitor is not handed.
- Lead with the free scan, keep the price quiet, let one true finding do the talking. The whole approach is Volume IV, The Work, and the sales ethic: never manufacture fear.
- A reachable internal login is the finding; typing into it is unauthorised access. The prospect's "yes" to go further becomes a written Test Authorisation, naming hosts, window, and stop-and-report contact, before one active request is sent.

CHAPTER TWO

The IDOR

A clean, textbook access-control bug, and two ways to write it up: the one that reads like a breach you caused, and the one a developer can act on and a court would not frown at. Same finding, very different report.

The bug in this case is the easy part. You will find it in an afternoon once you have read Volume II. The hard part, and the part that separates a tester from a professional, comes after: how you prove it without touching data you should not, and how you write it so the finding is the mechanism and not a pile of someone else's private life. This case is Volume II meeting Report Writing Mastery, and the second one is where interns go wrong.

The engagement

Quillmere, invented here, runs an appointment-booking platform for a group of clinics, reached at `clinicbook.example`. The authorisation covered the booking web application in full. You held two test accounts the client had created for exactly this purpose, account A and account B, both yours, both marked as test patients with fake details.

Logged in as account A, you booked an appointment and watched the request that loaded its own detail page.

```
GET /api/appointments/48217 HTTP/1.1
Cookie: session=...account A's session...
```

The appointment id was a plain sequential number. So you logged in separately as account B, in a different browser profile with B's own session, and asked for A's appointment by its id.

```
GET /api/appointments/48217 HTTP/1.1
Cookie: session=...account B's session...
```

That pairing is the finding. Account B, a different login with a different session cookie, received account A's record because the server checked only that B was logged in, never that the object belonged to B. It is insecure direct object reference, the horizontal access-control flaw Volume II calls the number-one class, and you proved it against two objects that were both yours. No third party's data was touched to establish it.

■ VERIFY IT

The proof of IDOR is cross-account, never self-account. Account A reading A's own appointment proves nothing: of course it can. The finding is B's session returning A's object, with two distinct logins shown side by side, different cookies, and a response that is genuinely A's record. One session and one object is a screenshot of the application working as designed. Two sessions and one object is a vulnerability. If you cannot show two sessions, you have not yet shown the bug.