



GHOST
PROTOCOL

DR

THE FIELD LIBRARY

DEFENCE

FROM THE DEFENDER'S CHAIR

Detection and Response

The defender's mindset, logging and telemetry, detection engineering, alerting and triage, and incident response.



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

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

THE GHOST PROTOCOL FIELD LIBRARY · DEFENCE

Detection and Response

The defender's mindset, logging and telemetry, detection engineering, alerting and triage, and incident response.

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 YOU ARE HERE

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 BUILT

The Bestiary BUILT

The Glossary and the Ledger BUILT

Contents

- Opening 5
- 1** The Defender's Mindset 6
- 2** Logging and Telemetry 11
- 3** Detection Engineering 17
- 4** Alerting and Triage 22
- 5** Incident Response 26
- Worked answers 32

The other chair



Blink was still glowing from the last track, where every door had opened. The attacker's chair suited the eager cursor well.

BLINK "We got in everywhere. We are good at this."

SOL "You are. Now sit in the other chair. Everything you learned to do, someone has to catch, and that someone is often blind for a simple reason: they never collected the one log that would have shown them. The defender does not need to outthink the attacker. They need to see. This track is how you make a room that sees, and what to do in the first hour when it does."

Worm's eye, the one that misses nothing, opened a little wider at the word see.

WYRM "watch. then you see."

This book is the defender's craft: collect the right traces, write the detection that fires on behaviour and not on yesterday's hash, cut the noise so a real alert is not buried, and respond when the real one lands.

CHAPTER ONE

The Defender's Mindset

Until now the book put you on the attacking side of the glass. This volume turns it around. The defender works at scale, after the fact, from traces, and has to decide and act on incomplete evidence without freezing. This chapter is the ground the rest of the volume stands on: how a defender thinks, what a SOC actually does, and why chasing indicators loses to watching behaviour.

The offensive volumes trained one habit: find the single path in. A defender cannot think that way. You do not get to pick one target and ignore the rest; you hold the whole estate, and the attacker only has to be right once. That sounds like a losing hand, and read a certain way it is. This chapter is about the other way to read it, the one advantage you do hold, and the shift in thinking that turns a wall of logs into a place you can actually work.

The asymmetry, and the ground you own

State the disadvantage plainly, because pretending it is smaller than it is gets people hurt. The attacker chooses the time, the target, and the method, needs one path to succeed, and can try quietly for months. You have to cover every host, every account, and every service, every day, and a miss anywhere can be the miss that matters. No amount of effort makes that symmetric.

Here is what you have back. You own the ground. Every packet crosses a network you control, every process runs on an endpoint you can instrument, every logon hits a directory you can read. The attacker is a guest on your machines, and a guest leaves marks. The whole craft of detection is turning that home advantage into evidence: putting sensors where the attacker has to pass, keeping the record where they cannot reach it, and knowing your own estate well enough that the abnormal stands out from the normal. A defender who knows their network cold beats an attacker who is learning it as they go. A defender who has never looked at their own logs is defending nothing.

The SOC is a Function, not a room

Security operations, the **security operations centre** or SOC, is often drawn as a room full of screens. Ignore the room. What matters is the function, a small number of jobs that have to happen whether one person does all of them at 2am or a team splits them across shifts. Four jobs carry most of it.

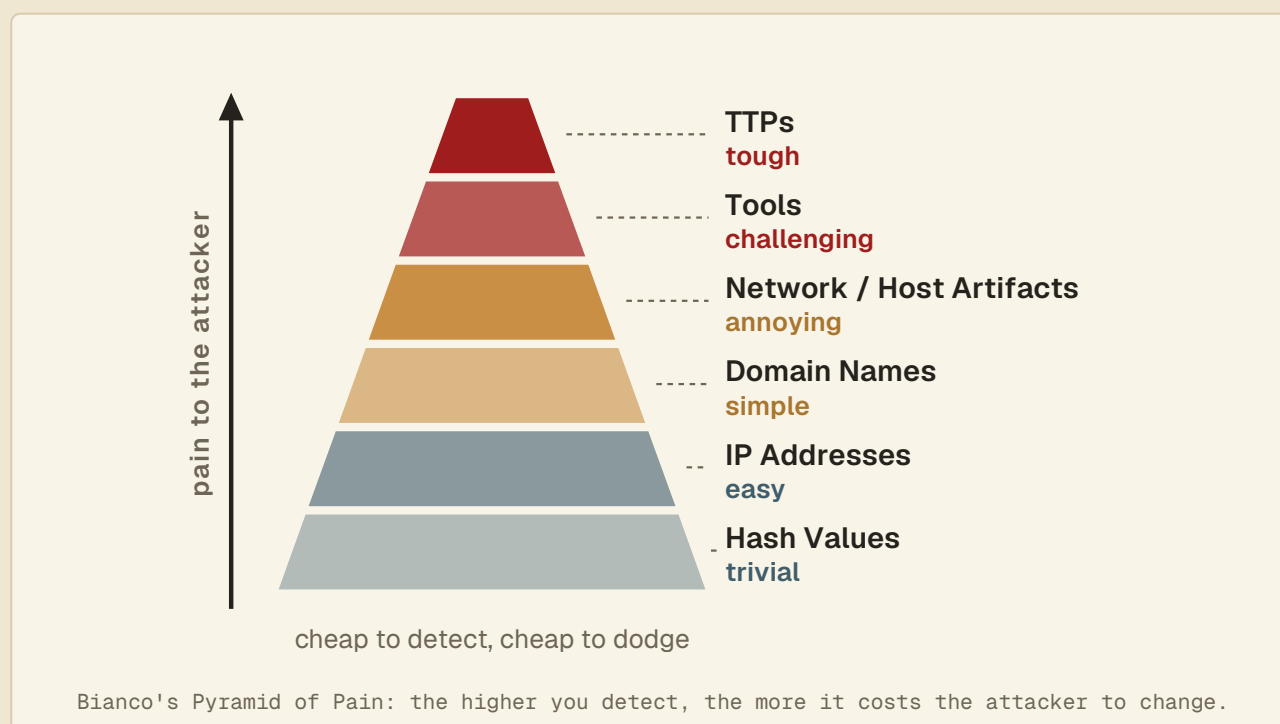
Triage is the front door: an alert fired, is it real, how bad, does it go further. **Investigation** takes a real alert and reconstructs what happened, across hosts and time, into a story you can act on. **Threat hunting** is the opposite direction: no alert fired, but you go looking anyway, on the theory that the quiet attacker is the one your alerts missed. **Detection engineering** is the feedback loop: every incident and every hunt that found something becomes a new detection, written down, so the next time it fires on its own. Incident response, the subject of Chapter Five, is what the function does when triage turns into something real.

Working alone, which is the usual case for a Ghost Protocol intern on a client's small estate, you wear all four hats in one night. The value of naming them is that you can tell which one you are wearing, and

stop yourself from mixing them. Hunting with a triage mindset makes you close the tab the moment nothing screams. Triaging with a hunter's mindset makes you chase every curiosity while the real alert ages. The discipline is knowing the job you are in.

Indicators, and the Pyramid of Pain

The simplest way to catch a known attacker is to write down the artifacts they used and watch for them again: the file hash of their malware, the IP address of their server, the domain they called home. These are **indicators of compromise**, IOCs, and they are the first tool everyone reaches for. They are also the weakest, and in 2013 David Bianco drew the picture that explains why. He called it the Pyramid of Pain, and it is the single most useful idea for deciding where to spend your effort.



Read the words on the right. If you detect on a file hash, the attacker changes one byte and the hash is new; the cost to them is close to zero, which is why it says trivial. An IP address is a new line on a hosting bill. A domain is a five-minute registration. Host and network artifacts, the registry key the malware writes or the odd user-agent string it sends, take a little tooling to change. Their actual tools cost real effort to re-build or re-buy. And at the top, their tactics, techniques and procedures (TTPs), the way they operate, the fact that they dump credentials from memory and then move over SMB and then encrypt, is the expensive thing to change, because it is their method, their muscle memory, the training behind their hands. Detect them there and you do not just block today's attempt. You make the whole campaign cost them, which is the only kind of win that holds.

Why behaviour beats chasing indicators

Put the pyramid to work and a rule falls out. The lower the level, the cheaper the detection is to build and the faster it goes stale; the higher the level, the harder it is to write and the longer it keeps paying. A feed of bad IPs is easy to subscribe to and will be half wrong by next week, because the IPs rotate. A detection for "a Microsoft Office process spawned PowerShell, which then reached out to the internet"

does not care what IP the server is on today. It describes the behaviour, and the behaviour is what the attacker is least able to drop.

This does not mean indicators are useless. A known-bad hash or a sinkholed domain is cheap to match and catches the lazy and the recycled, and some of your adversaries are lazy and recycled. The point is where you aim your scarce, expensive effort. You buy the cheap IOC coverage because it is cheap, and you spend your engineering time climbing the pyramid, writing detections for techniques, because that is where a detection earns its keep over months instead of hours. A SOC that only chases IOCs is always one step behind the last report. A SOC that detects behaviour is waiting where the attacker has to walk.



again?

Blink found a feed of four thousand malicious IPs and wanted to block all of them and file the night as handled. Wyrm did not look up. The feed was three weeks old; most of those addresses had been handed back to the pool and now pointed at ordinary sites, and the one address that mattered tonight was not on it. Block the cheap stuff because it is cheap. Do not mistake a blocklist for a defence.

Signatures, anomalies, and the base rate

There are two broad ways to decide that something is worth an alert, and you will use both. A **signature** says "this exact pattern is bad": the command line contains this string, the connection went to this domain. It is precise and explainable, and it only catches what you already described. An **anomaly** model says "this is unlike the normal for this host or this user": a workstation that never speaks to a domain controller suddenly does, at 3am, a lot. It can catch the thing you never wrote down, and it pays for that with noise.

The hard number behind the noise is the base rate. On a real estate, the overwhelming majority of unusual-looking events are benign: a sysadmin working late, a new tool rolled out, a backup job that moved. If a detection fires on one in a thousand events and the bad thing is one in a million, then even a very accurate detection buries each true alert under a heap of false ones, and the analyst drowns. This is the central problem of the whole defensive job, and Chapter Four is about surviving it. Hold the idea now, because it reframes what a good detection is. A good detection is not the one that catches the most. It is the one whose alerts an analyst can actually afford to read.

Evidence, not vibes

The offensive side of this Library has one unbending rule: a claim the evidence does not support is not a finding, it is a feeling, and feelings do not go in the report. The defensive side inherits that rule without a single change. A detection firing is a hypothesis, not a verdict. "The endpoint agent flagged this" is where the work starts, not where it ends. You confirm from the primary record: the raw log line, the process tree, the actual bytes on disk, the connection that actually happened. An alert is a lead. The looking is yours.

This matters more under defensive pressure than offensive, because the cost of being wrong points both ways. Call a real incident benign and you let an attacker keep working. Call a benign event an incident and you can take down a production service, wake a client in the small hours, and burn the

trust you cannot buy back, all over a sysadmin's late-night script. The defender who survives is not the one who never doubts. It is the one who is wrong cheaply, early, and out loud, and who checks before acting.

**SOL SAYS**

You will be wrong often in this work, and that is not the problem. The problem is the analyst who falls in love with the first theory and then reads every new log line as proof of it. Hold your first explanation loosely. Write it down so you can see it, then go looking for the evidence that would kill it, not only the evidence that would crown it. The best defenders I have watched are not the ones who are right first. They are the ones who notice they are wrong before it costs anything.

■ EXERCISE

On a machine that is yours, spend an hour just watching it be normal. On Windows open Event Viewer and read the Security log while you log off and back on; find your own logon event and read every field in it. On Linux tail `/var/log/auth.log` (or `journalctl -f`) while you `sudo` something. You are not hunting anything. You are building the one thing every detection depends on and most people skip: a felt sense of what your own system looks like when nothing is wrong. You cannot spot the abnormal until the normal is boring to you.

■ VERIFY IT

Do not take "IOCs go stale fast" on this book's word. Take any public threat-feed of malicious IPs or domains that carries dates, pull a sample from a month or two ago, and check how many still resolve to anything hostile today. Most will not. Now do the same thought for a behaviour: ask yourself how an attacker would stop using valid accounts, or stop running code, and notice there is no quick answer. You have just measured the pyramid for yourself, which is worth more than the drawing.

■ THE LINE

Detection is surveillance, and surveillance is of people, not only machines. The logon records you read, the web history your proxy keeps, the files your endpoint agent indexes: those belong to real employees doing real jobs, and the right to collect and read them comes from an authorisation and a policy, not from your curiosity. You monitor the estate you are engaged to defend, under the scope that engagement sets, and no further. The law and the oath are Volume 0 in full, and they do not relax because you switched to the defending side. A defender who reads what they were not authorised to read is not a defender.

■ ON YOUR FEET

It is 2am. One endpoint, a laptop belonging to someone in finance, raises a single alert: the antivirus quarantined a file it called a credential stealer. Nothing else is firing anywhere. Blink wants to pull that laptop off the network right now. Before you touch anything, what is the first thing you check, and what is the one question that decides whether this is a closed single-machine event or the first visible edge of something larger?



WHAT WYRM KEEPS

The attacker needs one path; the defender holds the whole ground. You cannot win on symmetry, so you win on home advantage: sensors where they must pass, and detections aimed at behaviour, not artifacts.

- The SOC is four jobs, not a room: triage, investigation, hunting, detection engineering. Alone you wear all four in a night, so know which one you are in.
- The Pyramid of Pain (David Bianco, 2013): hashes and IPs are trivial for the attacker to change, techniques are expensive. Spend cheap effort on IOCs, spend engineering effort climbing toward TTPs.
- Signatures catch what you described; anomaly models catch what you did not, and pay in noise. The base rate means most odd events are benign, so the measure of a good detection is whether its alerts can be read.
- An alert is a hypothesis, not a verdict. Confirm from the primary record before acting, because a wrong call takes down a service or lets an attacker stay. Be wrong cheaply and early.