A PRACTITIONER FRAMEWORK FOR THE AI CYBERSECURITY ECOSYSTEM

The Dual Mandate

Secure AI across build, runtime, defence and autonomy.

And use AI across all four.

Version 2.0 · Includes technology landscape and portfolio worksheet

Ritesh Patel (Rits)
Chief Information Security Officer
TheAISecurityFramework.com

What this document contains

Version 2 of The Dual Mandate expands the framework from a principle into a working instrument. The original four-domain model is unchanged. What is new is the technology landscape that sits underneath the principle, the horizon view that helps a CISO position their programme in time, and the portfolio worksheet that lets a security leader self-assess their coverage and produce a gap analysis.

The structure of this document:

- Foreword. Why this framework exists.
- The framework at a glance. The model in one image.
- The Dual Mandate principle. The failure modes the framework refuses.
- The four domains: Build, Runtime, Defence, Autonomy. Each domain explained, with both Secure and Use lenses and a diagnostic question.
- The technology landscape. New in v2. Capability classes across the four domains, mapped on a three-horizon model.
- Per-domain horizon cards. New in v2. Single-page references for each domain.
- How to use this framework. Diagnostic, programme structure, board narrative.
- The portfolio worksheet. New in v2. A self-assessment instrument that produces a gap analysis.
- Sources and acknowledgements.

Foreword

Every framework in security history has been a response to a moment when the threat outran the model we had for thinking about it. Defence in depth answered the perimeter falling. NIST CSF answered the operational sprawl that left organisations unable to describe their own programme. MITRE ATT&CK answered detection engineering having no shared language for what attackers actually did.

This framework answers a different moment. The arrival of AI capability that is simultaneously the most powerful tool a defender has ever been handed and the most powerful weapon a defender has ever faced. Both statements are true at the same time. The asymmetry is not coming. It is here, and it is documented.

> *Every AI capability you are deploying for productivity is also a capability someone will turn back on you.*

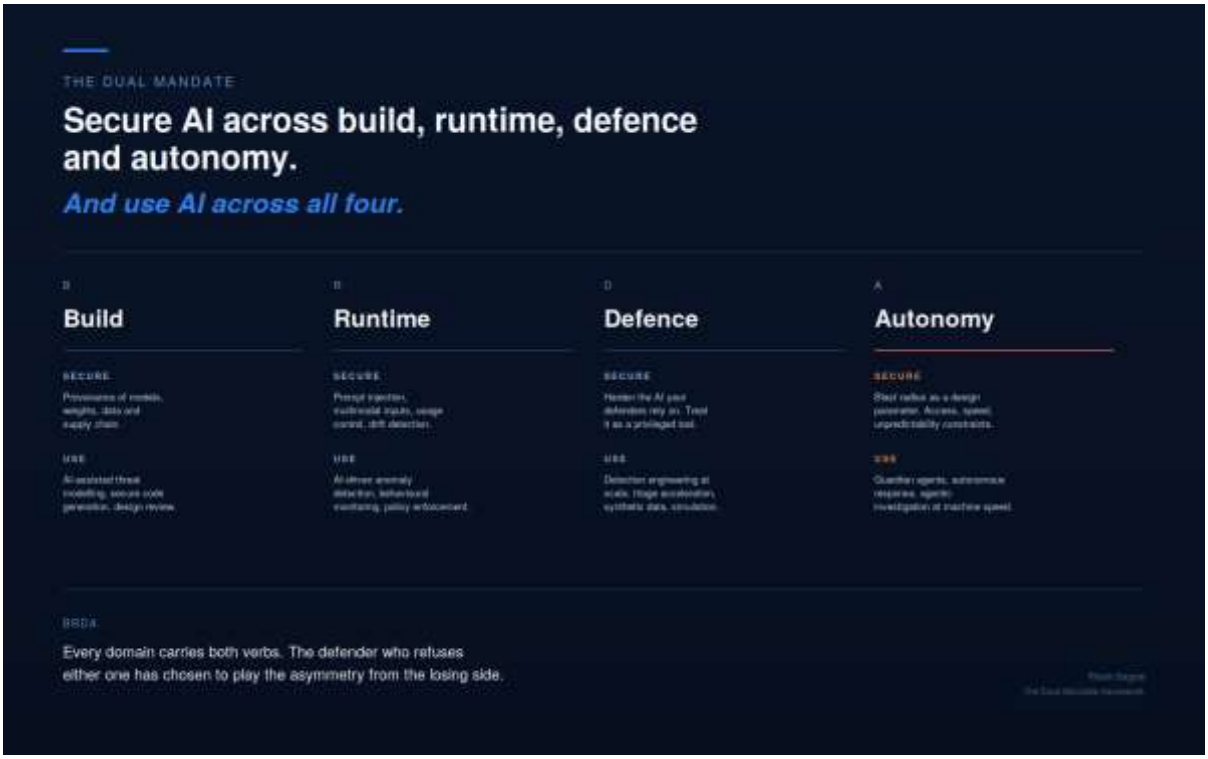
Most AI security programmes I see are doing one thing well. They are governing how AI is built and deployed. That is the part that looks most like the security work we already knew how to do, so it gets finished first, and the finished feeling is mistaken for a finished programme. It is not. It is the easy quarter of the problem.

The Dual Mandate exists because that framing leaves three of the four hardest questions unanswered. What protects the system once it is live and meeting inputs you did not write. What happens when the defending team is sent into a speed fight without the tool the attacker is already using. And what governs an agent that does not answer but acts, at machine speed, before any human can intervene.

The framework is structured around four domains: Build, Runtime, Defence, and Autonomy. BRDA. Across each domain runs a dual lens. Secure the AI in that domain, and use AI within that domain. Both verbs. Always. The defender who refuses either one has chosen to play the asymmetry from the losing side.

The framework at a glance

One page. The whole model. Everything else in this document expands what is shown here.



THE READING ORDER

BRDA is not a maturity progression. It is not the case that you do Build first, then Runtime, then Defence, then Autonomy. The domains run in parallel. Most programmes that get this wrong sequence the domains in maturity order and end up two years in before they have begun the work in Defence and Autonomy. By then the attackers have built a two-year head start.

Read the framework as four parallel tracks. Honest progress means progress in all four, weighted by your highest risk.

The Dual Mandate

The principle is a deliberate refusal of the two failure modes most AI security programmes fall into.

Failure mode one: secure-only thinking

The team treats AI exclusively as something to defend against. The work becomes pure restriction. Policies forbid use, controls limit deployment, the security team becomes the office of saying no. The result is a defending organisation that is institutionally slower than the attacking organisation, because the attacker has no policy committee.

This is the path of caution. It feels safe. It loses the speed race on purpose.

Failure mode two: use-only thinking

The team treats AI exclusively as productivity to capture. Tools are deployed at velocity, copilots and agents are rolled out across business units, and the security work is reduced to a check at the build gate. Nothing meaningful protects the systems in production or governs what they can do.

This is the path of enthusiasm. It feels modern. It exposes the organisation faster than caution would have.

The mandate

Both verbs must travel together, across all four domains. The work is to secure AI as a system, and to use AI as leverage, in the same programme, on the same timeline, with the same seriousness.

> *Secure AI across build, runtime, defence and autonomy. And use AI across all four.*

This is not a balance to strike. It is a discipline to hold. Any domain where one verb is missing is a domain where you have left a structural advantage on the table or accepted a structural risk you did not name.

DOMAIN ONE

Build

Build covers everything that happens before an AI system reaches production. Model selection, weight provenance, training and fine-tuning data, dependencies, the inference path. This is the domain that looks most like traditional software supply chain work, which is why it gets done first, and why most organisations stop here.

SECURE

The attacker does not need to breach your model in production if they can poison what goes into it before it ships. AI inherits every weakness in how it was assembled, and most organisations cannot currently tell you where their model weights came from, what was in the fine-tuning set, or which third-party components sit in the inference path.

- Provenance for models, weights, and training data.
- Supply chain integrity across model dependencies, prompt libraries, and inference tooling.
- Pre-deployment security testing including adversarial robustness, jailbreak resistance, and data leakage.
- Governance of data flowing into training and context, with attention to the data the business does not realise is being used to train someone else's model.

USE

The same build pipeline is the strongest place to apply AI to your own advantage. AI-assisted threat modelling at design time finds risks humans miss. AI code review at pull request catches issues before merge. AI-driven dependency analysis surfaces supply chain concerns at speed.

- AI-assisted threat modelling integrated into design reviews.
- AI-augmented secure code review at pull request and pre-merge.
- Continuous AI scanning of dependencies, configurations, and infrastructure-as-code.
- AI-driven generation of test cases, including security and abuse cases the human author would not consider.

THE DIAGNOSTIC QUESTION

Can you produce, today, a provenance record for the model in your highest-risk AI system? Where the weights came from, what was in the tuning data, who touched it, what is in the inference path. If the honest answer is no, you are not securing the build. You are trusting it. Those are different things, and only one of them survives an incident review.

DOMAIN TWO

Runtime

Runtime covers what protects and governs the AI system once it is live and serving. The test harness is not the world. The world contains adversarial inputs you did not write, multimodal channels you did not anticipate, and usage patterns that drift beyond anything you validated.

SECURE

Indirect prompt injection is the clearest case. The attacker does not target your model. They poison a document, a web page, an email, a calendar invite, and wait for your model to read it on someone's behalf. The instruction arrives inside data the system was designed to trust. Nothing in your build-time testing catches this because the malicious content did not exist when you tested. Multimodal makes it worse. Inputs hidden in images, instructions buried in audio.

- Detection and mitigation of prompt injection across direct and indirect channels.
- Input validation and sanitisation across all modalities the model accepts.
- Usage control and policy enforcement at the inference boundary.
- Drift detection: monitoring for usage patterns that fall outside the validated scope.
- Output monitoring for data leakage, policy violation, and harmful content.

USE

Runtime is also where AI provides the richest defending signal. Behavioural baselines built by models. Anomaly detection that catches what static rules miss. Policy enforcement at speeds no human can match.

- AI-driven anomaly detection on user, system, and network behaviour.
- Behavioural baselining of normal model usage to flag deviation in real time.
- Automated policy enforcement that scales with the volume of AI traffic.
- Real-time correlation across telemetry that would overwhelm a human analyst.

THE DIAGNOSTIC QUESTION

Name the last adversarial input your AI system encountered in production that you did not generate yourself in testing. If you cannot name one, there are two possibilities. Either nobody is probing your system, which is not true. Or they are, and you cannot see it. Decide which of those you actually believe, then decide whether your runtime telemetry would tell you the difference.

DOMAIN THREE

Defence

Defence is the domain where the framework inverts. The first two domains treated AI as an asset to protect. Defence treats AI as leverage for the defending team. This is the domain most organisations have not properly scoped, because it sits uneasily between security tooling and security operations.

SECURE

The AI your defenders use is itself a high-value target. A compromised security copilot is a compromised analyst with privileged access. A poisoned detection model is detections you cannot trust. The AI on the defending side must be treated as privileged infrastructure, not a productivity add-on.

- Treat security AI tooling as privileged infrastructure with appropriate hardening and access control.
- Validate the integrity of detection models, threat intelligence feeds, and AI-augmented decisioning.
- Monitor security AI for prompt injection and manipulation by the very adversaries it is deployed against.
- Apply provenance and supply chain rigour to security AI with the same intensity as to business AI.

USE

This is the domain where use is the whole point. The attacker has already decided AI is leverage. The economics of offence have collapsed. End-to-end automation of reconnaissance, payload generation, social engineering, and exfiltration is real and documented. Anthropic's September 2025 disclosure described a state-aligned group using agentic tooling to execute 80 to 90 percent of a live espionage operation across roughly thirty targets. Unit 42 demonstrated a controlled simulation that cut compromise-to-exfiltration from a median of two days to twenty-five minutes.

The speed asymmetry is not a future risk. It is the current operating condition.

- Detection engineering at scale: AI-generated detection logic across the threat surface.
- Triage acceleration: first-pass review of alerts to give analyst hours back.
- Synthetic data generation for training detections that could not otherwise be built.
- Intelligent simulation: continuous red team activity at a speed human red teams cannot match.
- Investigation assistance: summarisation, correlation, and hypothesis generation across SOC workload.

THE DIAGNOSTIC QUESTION

Is your security team allowed to use AI as aggressively as the people attacking them are? Not in theory. In practice, with the tooling provisioned and the policy signed. If the attackers have automated their kill chain and your analysts are still doing first-pass triage by hand because the AI policy is stuck in legal, you have already answered the question. You just have not said it out loud.

DOMAIN FOUR

Autonomy

Autonomy is the frontier domain. Everything in the previous three assumed a system that answers and waits. Autonomy removes that assumption. The agent acts. It calls tools, chains steps, makes and executes decisions at machine speed, often before any human is in the loop.

This is the domain most organisations have no answer for, and the one the adversary has already published the proof of concept for.

SECURE

The risk model is different. With a chatbot, the worst case is a bad answer. With an agent, the worst case is a bad action, at scale, before anyone notices it was driven. Blast radius stops being an incident review finding and becomes a design parameter. The questions move from 'what could this system say' to 'what could this system do, how fast, and how far before anyone intervenes.'

- Blast radius scoped explicitly per agent: what it can touch, how much, with what authority.
- Credentials and access minimised, scoped, and time-bound. No standing privileged tokens for autonomous systems.
- Speed governors and checkpoints on actions with irreversible consequences.
- Audit trail sufficient to reconstruct an agent's chain of reasoning and tool calls.
- Explicit policy for what agents do and do not act on without human approval.

USE

The same capability that creates the risk creates the response. Defending agents that operate at the speed of the attacking agents. Guardian agents that watch for the patterns no human SOC could catch. Autonomous response within carefully scoped blast radius. The defender who refuses this on principle accepts machine-speed offence against human-speed defence.

- Guardian agents monitoring privileged systems at machine speed.
- Autonomous investigation, scoped to read-only context expansion.
- Autonomous response within tightly defined, reversible action sets.
- Agent-to-agent coordination across security functions where the human is the conductor, not the operator.

THREE CONTROL LEVERS

Every effective control on an autonomous system picks one of three levers.

- Reduce the access the agent has.
- Slow the speed at which it acts.
- Constrain the unpredictability of what it does.

Prohibition picks none of them, which is why bans on agentic AI do not survive contact with a business that wants the productivity. Pick the right lever for each agent. Apply more than one for high-blast-radius actions.

THE DIAGNOSTIC QUESTION

For your most capable agent, what is the worst thing it could do in the ninety seconds before a human could intervene? If you cannot answer that precisely, you have not scoped blast radius. You have assumed it.

NEW IN V2

The technology landscape

A framework without a technology landscape underneath it stays at the level of principle. This section lays out the capability classes that sit beneath each domain, mapped on a three-horizon model. The capabilities are deliberately named as classes, not vendors, so the framework outlasts any specific product's market position.

The horizon model

The horizon view is taken from Sharpe and Hodgson's Three Horizons foresight framework, used widely in strategic planning. Applied to AI security, it gives you three time-and-commitment buckets.

HORIZON 1: FOUNDATIONAL

Capabilities available today and expected in every credible programme. If you do not have these, you have a foundational gap. These are operational expenditure, not strategic investment.

HORIZON 2: EMERGING

Capabilities taking shape, being deployed by leading organisations, where the standards and approaches are maturing but not yet universal. These are the strategic investments that determine whether your programme is keeping pace with the threat or falling behind.

HORIZON 3: FRONTIER

Capabilities at research, pilot, or just-emerging stage. Standards are still forming. You should be scouting these, running small pilots, and maintaining a vendor advisory relationship rather than deploying at scale. The point is not to wait. The point is to know what is coming so it does not surprise you.

The landscape, end to end

All four domains. All three horizons. Both verbs in each cell. Capability classes only, no vendors named.

HOW TO READ THIS VIEW

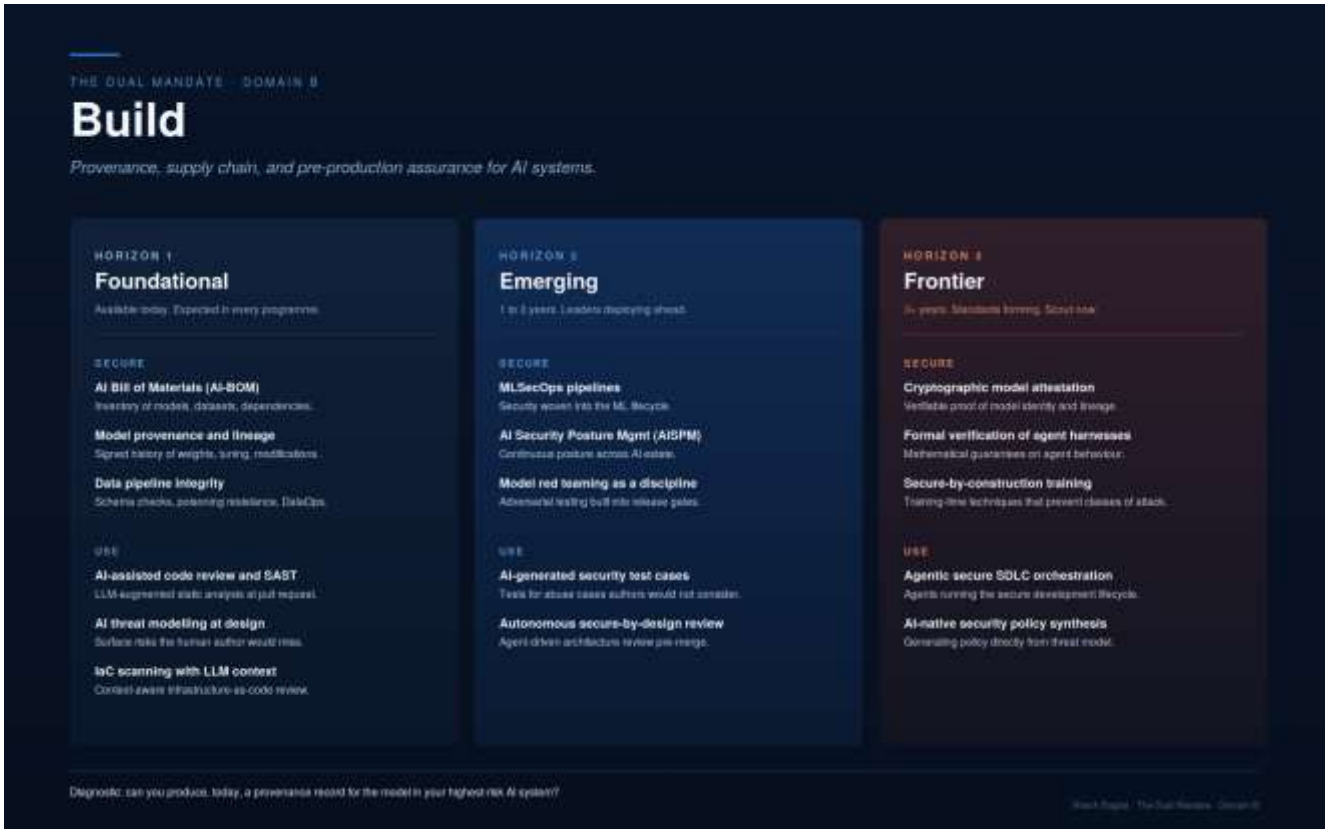
Each row carries both verbs. Each column carries a different commitment of time, capital, and risk appetite. Read it as a portfolio. Most credible programmes should hold capabilities in all three horizons across all four domains, with the balance shifted toward H1 for operational maturity and toward H2/H3 for strategic positioning.

The placement of any specific capability on the horizon line is a judgement call, not a hard fact. Different organisations will see things differently depending on sector and risk appetite. A bank may see ephemeral task-scoped credentials as H1 because it already has that architecture for service accounts. A manufacturer may see SOC copilots as H2 because the SOC is still largely manual. The framework is meant to be discussed and adapted, not adopted as a fixed taxonomy.

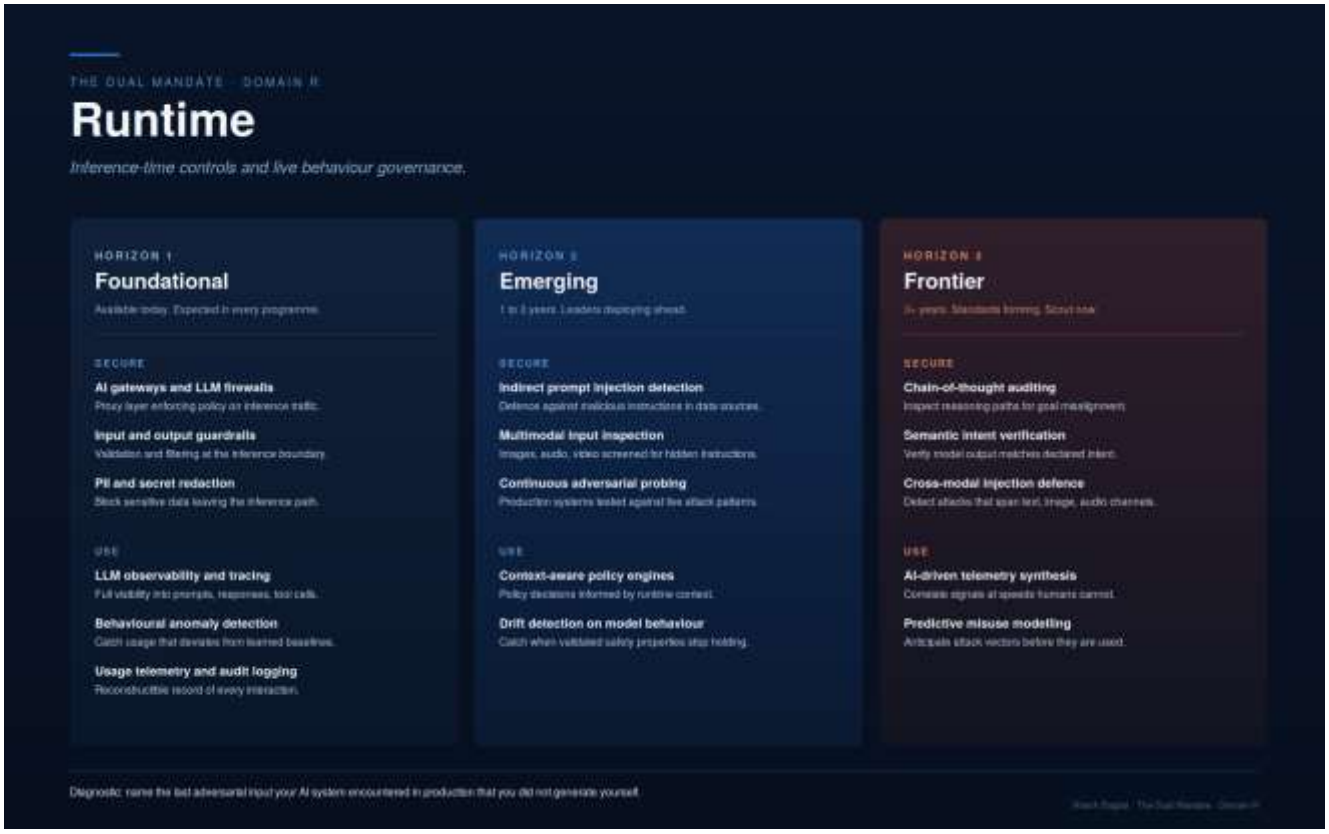
Per-domain horizon cards

Each of the four domains in detail, on a single page. Use these for domain-specific conversations with the team responsible for that area. The diagnostic question for the domain sits at the foot of each card.

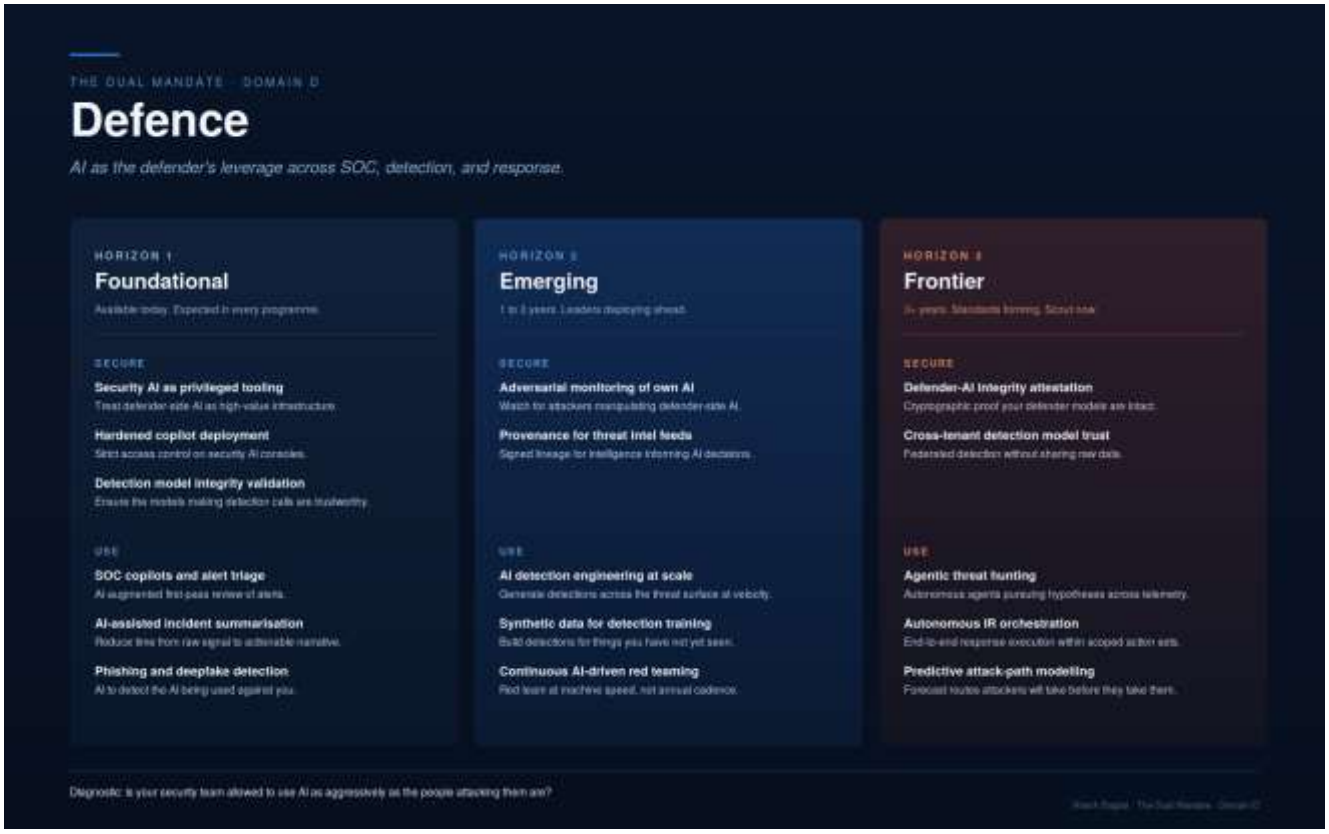
DOMAIN B: BUILD



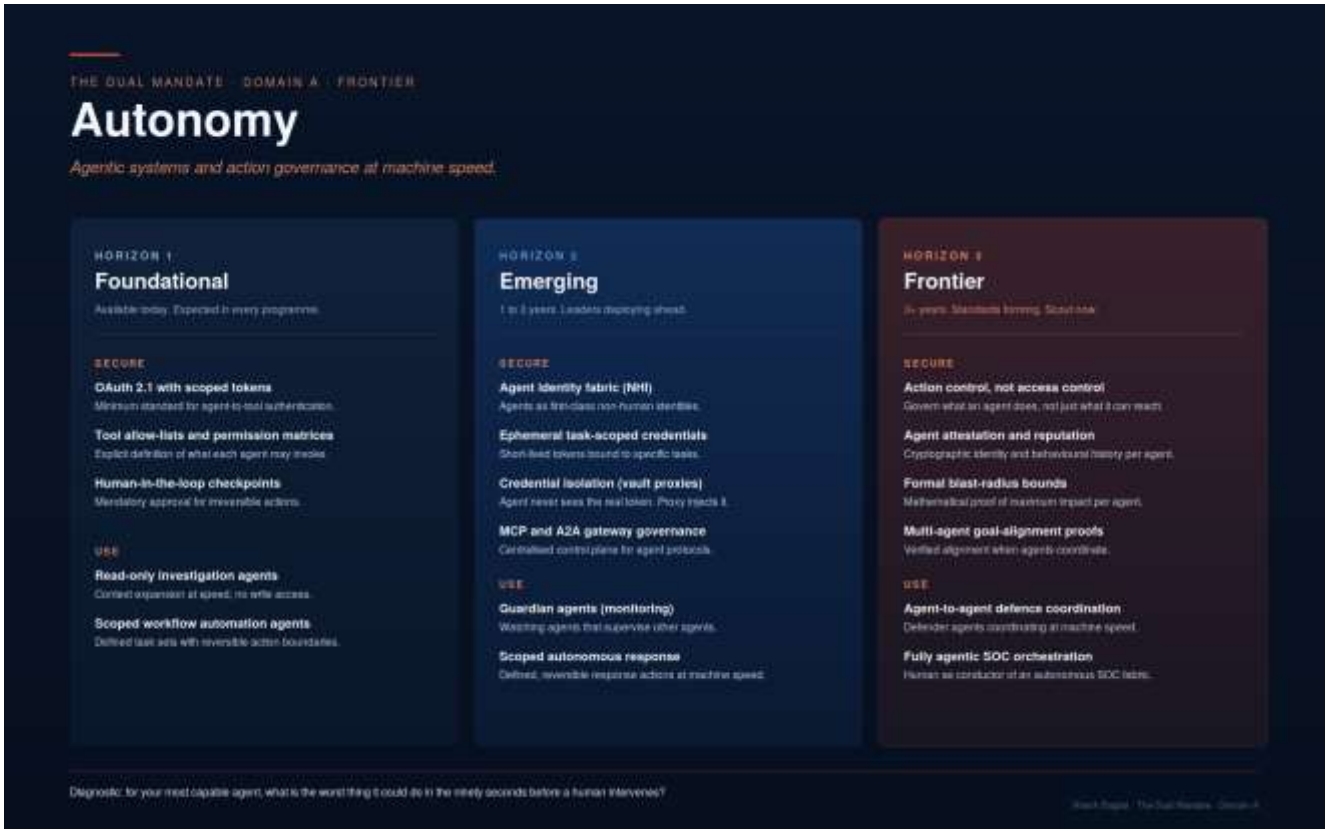
DOMAIN R: RUNTIME



DOMAIN D: DEFENCE



DOMAIN A: AUTONOMY



How to use this framework

Four concrete uses, in increasing order of commitment.

As a diagnostic

Walk a senior team through the four diagnostic questions. Build, Runtime, Defence, Autonomy. Score each honestly. The lowest score is where you start the next quarter's work. Not the question that is easiest to improve. The one you most wanted to skip while reading.

As a programme structure

Reorganise reporting against the four domains. Each domain has an owner, both verbs as named workstreams, and a quarterly readout. The dual mandate is the operating principle, not a footnote. Treat any domain where one verb is missing as a structural gap, not an acceptable trade-off.

As a portfolio audit

Use the technology landscape and the portfolio worksheet (see next section) to assess your coverage across the twenty-four capability cells. The shape of the gaps tells you where the next two years of investment should go. The portfolio view also lets you balance operational spend (H1) with strategic spend (H2) and scouting spend (H3) explicitly, rather than letting the budget default to operational by inertia.

As a board narrative

The board does not need to understand prompt injection. They need to understand that the organisation is doing AI security across four named domains, that each domain has a defence side and a leverage side, and that the leverage side is what keeps you in the speed race. The framework is built to fit on one slide and survive scrutiny.

NEW IN V2

The portfolio worksheet

A self-assessment instrument that produces a gap analysis automatically. The worksheet is published alongside this document as a separate Excel file.

What it does

The worksheet contains all twenty-four capability cells of the framework. You score each one honestly on a zero-to-four scale. Three other tabs then populate automatically: a gap analysis showing your average score by domain and horizon, a priority list of the ten lowest-scoring capabilities, and a one-page dashboard summarising overall maturity, domain scorecard, and verb balance.

The scoring scale

- 0: Not in place. No capability, no planning, no budget.
- 1: Awareness. Recognised, scoping, no deployment.
- 2: Partial. Some areas covered, inconsistent, no defined ownership.
- 3: Established. Deployed, owned, measured, reviewed.
- 4: Mature. Integrated, automated, effective, continuous improvement loop.

How to run the assessment

Quarterly, ninety minutes with the senior security team. Walk through each domain. Score honestly. The dashboard will show you the truth of your programme in numbers. The priority cells will tell you where to start. The verb balance will tell you whether your programme is skewed toward defence (secure-only thinking) or leverage (use-only thinking).

The diagnostic value of this exercise comes from naming the gap. If a cell scores low, that is information. The lowest scores are not failures. They are the next quarter's work.

> *Score what is actually true today. Not what is planned, not what the slide says, not what you would like to be true at the next board meeting.*

Closing

The frameworks that became standards in security did not become standards because they were complete. They became standards because they gave practitioners a shared language for work that was previously done in silos and under different names. Defence in depth, NIST CSF, MITRE ATT&CK. None of them was the last word. All of them were a useful word at a moment when the field needed one.

AI security needs a useful word now. Not another taxonomy of risks. A frame that names both halves of the work in the same breath, and forces every team that uses it to answer both halves.

> *Pick the clause where your honest answer to its question was the weakest. Not the easiest to improve. The one you most wanted to skip while reading. That reluctance is the signal. Start there.*

Build it secure. Run it secure. Wield it for defence. Govern the autonomy. And use AI in every one of those domains as hard as the people attacking you already are.

Both verbs. Every domain. That is the mandate.

Sources and acknowledgements

VERIFIED SOURCES CITED IN THIS DOCUMENT

Anthropic. Disrupting the first reported AI-orchestrated cyber espionage campaign. anthropic.com/news/disrupting-AI-espionage. Detected mid-September 2025, disclosed 13 November 2025.

Palo Alto Networks Unit 42. Unit 42 Develops Agentic AI Attack Framework. Published 14 May 2025. The 25-minute compromise-to-exfiltration figure is drawn from the 2025 Unit 42 Global Incident Response Report referenced within.

HORIZON MODEL

Sharpe, B. and Hodgson, A. The Three Horizons framework. Developed by Bill Sharpe and Tony Hodgson, used widely in strategic foresight and applied here to position AI security capabilities in time.

TECHNOLOGY LANDSCAPE

Capability classes synthesised from public industry research across MLSecOps practice, AI gateway architectures, agentic AI identity and governance work, and security AI deployment patterns. Specific vendor names are intentionally omitted to keep the framework durable across the product churn that always follows a category's emergence.

ON USE AND CIRCULATION

This framework is published openly. You are welcome to adopt it, adapt it, teach with it, present it at your organisation, and build on it. Attribution is appreciated; permission is not required. If you build something interesting on it, I would like to hear about it.

Ritesh Patel (Rits) · TheAISecurityFramework.com

— end —