MIRAGE: Model-based Intent Reasoning for Adaptive Generated Environments

The MayaJaal Project

***Abstract*—MayaJaal keeps a hospital network working through an intrusion. A gateway sits in front of the services staff sign in to; when a signed-in session starts to act like an intruder, it is never refused: its non-clinical traffic is silently moved into a synthetic "shadow" hospital generated from a per-session HMAC seed, while real records stay behind the policy. MIRAGE (Model-based Intent Reasoning for Adaptive Generated Environments) is the adaptive layer: a Bayesian filter over the session's goal and kill-chain stage, a lane adjuster that can only make treatment stricter, and a planner that adds lures to the shadow's portal pages. MIRAGE-S weighs a lure's information gain against a model of the attacker's suspicion and looks two lures ahead. In an in-model simulation over four attack goals, a cautious simulated attacker abandons $68.9 \pm 1.0\%$ of sessions under the earlier planner and $8.6 \pm 0.6\%$ under MIRAGE-S, and the true goal is identified in $64.9 \pm 0.9\%$ of sessions instead of $45.5 \pm 1.0\%$. Against a bold attacker that never leaves, MIRAGE-S is slower than the earlier planner ($+0.84 \pm 0.11$ requests), and looking ahead on information alone gains nothing. On the running system, one scripted attack took 1,199 real patient records without deception and 0 with MayaJaal, and two attacker sessions shared 0% of their decoy hospital (a static decoy: 100%). We report no human red-team data.**

I. INTRODUCTION

The 2024 ransomware attack on Synnovis, pathology provider to several London NHS trusts, forced planned care to be postponed [1]. An intruder with one valid login can do real harm, yet a hospital cannot cut access when unsure, because clinicians need records. Deception fits: a decoy wastes the intruder's time, reveals intent and raises a high-signal alarm [2]. Static honeypots can be fingerprinted [3], [4]; per-user, plausible bait is harder to tell from the real thing [5], [6]. Two lines of work shape this version. Shinde et al. [9] plan decoys with interactive POMDPs that model the attacker's mental state and how deception changes it; we keep that idea but reduce the nested model to one scalar, the attacker's suspicion. Beltrán López et al. [10] redirect connections to an on-demand clone so that the switch is hard to detect; we apply the same stealth concern to lures: changing a page the session has already seen is costed as a tell. Contributions: (i) sticky diversion that never fakes clinical reads; (ii) a (goal×stage) filter that can only raise a lane; (iii) HTTP lures scored on take and ignore; (iv) a suspicion-aware lookahead planner; (v) an evaluation that reports where it loses, plus end-to-end runs and encryption at rest on the running system.

II. THREAT MODEL

The adversary holds valid credentials for a clinical account (a phished nurse) and reaches the hospital only through the network services it exposes; the prototype exposes two, the clinical record interface (FHIR) and the IT portal. Its goal is *recon*, *exfil*, *tamper* (lab results) or *disrupt*. The defender wins if none of the 12 canary secrets seeded into the real tier leaves through the gateway. The gateway, policy engine and shadow key are assumed uncompromised; timing channels are out of scope.

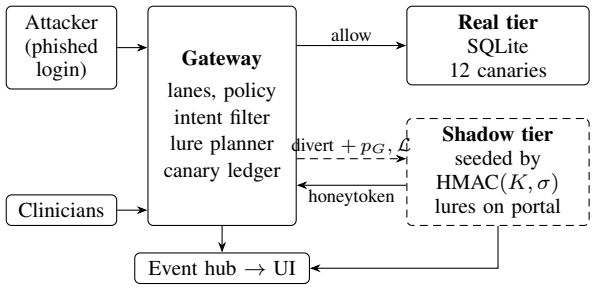


Fig. 1. Architecture. The gateway is the only public port; suspect clinical reads are refused, never faked. The shadow has no path to the real tier and gets only p_G and the lures $\mathcal{L}$ in a header.

III. METHOD

A. Gateway lanes and sticky diversion

Each session has a lane $\ell \in \{$*normal* $<$ *suspicious* $<$ *attacker*$\}$ and a *diverted* bit. Fixed signals raise the lane: admin or backup access by a non-admin, bulk export from an interactive role, cross-hospital enumeration, lab writes by roles that do not write labs; a honeytoken touch sets *attacker*. The lane never decreases. The action is a lookup $a = \text{ACT}[\text{bucket}][\ell]$: sensitive buckets are diverted from *suspicious* upward, while clinical reads get a 401 step-up or a 403, so a misclassified clinician never acts on a fake chart. Diversion is *sticky*: afterwards every non-clinical request is diverted, so the session sees one consistent world.

B. Intent filter

The hidden state is (g, s), with goal $g \in \{\text{b(enign)}, \text{recon}, \text{exfil}, \text{tamper}, \text{disrupt}\}$ and stage $s \in (\text{recon}, \text{access}, \text{collect}, \text{act})$. The prior is $b_0(g, s) = \pi(g)\rho(s)$, with $\pi = (.80, .08, .05, .04, .03)$ and $\rho = (.70, .20, .08, .02)$.

Predict. Let $m_g = \sum_s b(g, s)$ and $q_g = b(g, \cdot)/m_g$. Stages only move forward: $q'_g = q_g A$, where A advances a stage with probability $\alpha = 0.15$ and *act* is absorbing. Goals are sticky. An attack goal keeps a fraction $1 - \delta$ of its mass ($\delta = 0.04$), and a benign session turns hostile with probability $\tau = 0.005$. The

mass that switches is spread over attack goals in proportion to $\pi_A(g) \propto \pi(g)$ and enters at the stage prior:

$$\bar{b}(g,s) = m_g(1-\delta_g)\, q'_g(s) + \iota_g\, \rho(s), \qquad (1)$$
$$\iota_g = \big(\delta(1-m_{\mathrm{b}}) + \tau m_{\mathrm{b}}\big)\pi_A(g),$$

where $\delta_{\mathrm{b}} = \tau$ and $\iota_{\mathrm{b}} = 0$. No mass ever flows back to benign, so only evidence can clear a session.

Likelihood. Each request maps to one of 22 categories c (ward list, cross-hospital list, record read, note read, bulk export, lab write, backup, admin page, …) and two flags f: *denied* (401/403) and *honeytoken*. A hand-written table gives goal weights $W_{c,g}$ and stage multipliers $M_{c,s}$; benign sessions have no stage:

$$P(c \mid g,s) \propto W_{c,g}\, M_{c,s}^{[g\neq \mathrm{b}]} \quad \text{(normalised over } c\text{)}. \qquad (2)$$

Flags have tables $P(f \mid g)$. Only a flag that is present counts as evidence.

Update. Requests are correlated, so evidence is tempered by $\gamma = 0.7$ and the k-th consecutive repeat of a category is further scaled by $1/\sqrt{k}$:

$$b_t(g,s) \propto \bar{b}(g,s)\, P(c_t \mid g,s)^{\gamma/\sqrt{k_t}} \prod_{f\in F_t} P(f \mid g)^{\gamma} \cdot \big[E_{\kappa g}^{\,o}(1-E_{\kappa g})^{1-o}\big]. \qquad (3)$$

The bracketed term applies when a lure κ resolves with outcome $o \in \{0,1\}$ (taken or ignored, §III-D), with $E_{\kappa g} = P(\text{engage} \mid g,\kappa)$; a taken lure replaces the generic honeytoken flag. The goal posterior is $p_G(g) = \sum_s b_t(g,s)$.

C. Monotone lane adjustment

The filter affects routing only through $\ell \leftarrow \max(\ell, \lambda(p_{\mathrm{b}}))$. Here $p_{\mathrm{b}} = p_G(\mathrm{b})$ is taken after the *previous* request, and λ returns *attacker* if $p_{\mathrm{b}} < 0.10$, *suspicious* if $p_{\mathrm{b}} < 0.35$, and no change otherwise. For every bucket, access to the real tier does not increase as ℓ rises. A wrong or manipulated posterior can therefore make the gateway stricter, but it can never grant access that the rules refuse.

D. Lures over HTTP

The attacker has no shell, so a lure is a link. Five kinds have hand-set take rates $E_{\kappa g}$ (exfil takes a backup dump with probability 0.85). Archives and a research file appear on the shadow's *backups* page, credentials and a VPN profile on its *documents* page; the real tier answers 403 on the same routes. Since only the gateway sees every request, the planner runs there: each diverted request carries the posterior and the set $\mathcal{L}$ of lures that should exist in `X-MJ-Posterior`, and the shadow plants what is missing before serving. A lure is *shown* when its host page is fetched after planting, *taken* when its link is fetched and the shadow reports a new honeytoken touch, and *ignored* after three further requests. Both outcomes enter (3).

E. Planning: information against suspicion

The earlier planner (IG-1) chooses the lure with the largest expected entropy reduction [7],

$$I(\kappa; b) = H(G) - \sum_{o\in\{0,1\}} P(o \mid \kappa)\, H(G \mid o,\kappa), \qquad (4)$$

and stays believable through fixed pacing: one lure per 4 requests, 3 per session. MIRAGE-S replaces the pacing rule with a model of the attacker. Let λ be the attacker's log-odds that the host is a decoy, starting at $\mathrm{logit}\, 0.05$. Each lure shown adds

$$w = 0.4 + 1.6\,[\text{late}] + 0.35\, n + 0.8\,[\text{burst}], \qquad (5)$$

where *late* means its host page had already been viewed when it was planted, n counts lures already shown, and *burst* means fewer than 3 requests since the last lure. A *cautious* attacker then leaves with probability $q^c = \sigma(1.5\lambda)$; a *bold* one never does. The defender keeps $\phi = P(\text{cautious})$, initially 0.5, and lowers it by Bayes' rule each time a lure is shown and the session continues. In lure state $s = (\lambda, \phi, \text{pages viewed}, \text{lures planted})$, planting κ is worth

$$Q_d(\kappa \mid b,s) = (1-q)\big[I(\kappa;b) + \gamma\, \mathbb{E}_o V_{d-1}(b_o, s')\big] - q\,\big[H(b) + \beta\big], \quad V_d = \max\big(0, \max_{\kappa'} Q_d(\kappa' \mid \cdot)\big), \qquad (6)$$

with $q = \phi\, q^c$, $V_0 = 0$, $\gamma = 0.9$, $\beta = 0.5$ bits for the engagement itself and depth $d = 2$. If the attacker leaves, the remaining $H(b)$ bits are never resolved. In s' the next lure comes 3 requests later, by which time each unviewed page has been opened with probability $1-(1-r)^3$ ($r = 0.15$ documents, 0.05 backups) and a lure there would be late. A lure is planted when $\max_\kappa Q_d \geq 0.03$ bits and, inside the burst window, waiting is not worth more. All weights are judgement calls. With $\phi = 0$, (6) is information-only lookahead.

F. Session-seeded shadow generation

Every shadow entity e (patient, lab panel, dump, file, honeytoken, lure) draws from a PRNG seeded with $\mathrm{HMAC\text{-}SHA256}(K, \sigma \| e)$ [8] for session σ and process key K, so the world is lazy but deterministic, differs per session, and is built only from public templates. Tests confirm the shadow tier never imports the real tier or reads its data. A ledger scans every response to a non-clinician for the canaries (plain, hex, base64).

G. Encryption at rest

Notes, SSNs, addresses, phones, service secrets and the canary table are stored as AES-256-GCM ciphertext bound to table, column and row. The data key is stored only wrapped by a key that is Shamir-split 3-of-6 across the hospitals and rebuilt in memory at start; any two shares fail. Names, birth dates and conditions stay plaintext; in the prototype all shares sit on one machine.

TABLE I

MATCHED ENGAGEMENT, FOUR ATTACK GOALS, 2,000 SEEDS PER CELL, MEAN ± 95% HALF-WIDTH. "MIXED" POOLS BOLD AND CAUTIOUS SESSIONS EQUALLY; Δ IS PAIRED BY SEED AGAINST IG-1. SIMULATION OF OUR OWN MODEL, NOT RED-TEAM DATA.

	Bold attacker		Cautious attacker				Mixed	
Lure strategy	Steps ↓	Identified %	Steps ↓	Identified %	Stayed (req.)	Left %	Δ steps	Δ ident. (pts)
none	21.31 ± 0.16	62.5 ± 0.9	21.31 ± 0.16	62.5 ± 0.9	30.0 ± 0.0	0.0 ± 0.0	$+2.20 \pm 0.13$	$+3.2 \pm 0.7$
random, paced	18.76 ± 0.18	70.1 ± 0.9	22.69 ± 0.20	39.4 ± 1.0	16.2 ± 0.2	68.8 ± 1.0	$+1.63 \pm 0.11$	-4.6 ± 0.5
IG-1, paced (earlier planner)	17.31 ± 0.18	73.1 ± 0.9	20.89 ± 0.22	45.5 ± 1.0	16.2 ± 0.2	68.9 ± 1.0	—	—
IG-1, every request	15.76 ± 0.20	76.2 ± 0.8	25.38 ± 0.20	18.9 ± 0.8	5.2 ± 0.1	100.0 ± 0.0	$+1.47 \pm 0.12$	-11.8 ± 0.6
lookahead, information only	15.75 ± 0.20	76.2 ± 0.8	25.53 ± 0.20	18.4 ± 0.8	5.2 ± 0.1	100.0 ± 0.1	$+1.54 \pm 0.12$	-12.0 ± 0.6
suspicion-aware, one step	17.95 ± 0.19	70.2 ± 0.9	18.97 ± 0.20	63.6 ± 1.0	27.2 ± 0.2	11.2 ± 0.7	-0.64 ± 0.09	$+7.6 \pm 0.6$
MIRAGE-S (suspicion + lookahead)	18.15 ± 0.19	69.6 ± 0.9	18.86 ± 0.20	64.9 ± 0.9	27.9 ± 0.1	8.6 ± 0.6	-0.60 ± 0.10	$+8.0 \pm 0.6$
MIRAGE-S, robust likelihood	19.67 ± 0.17	66.7 ± 0.9	19.99 ± 0.18	64.3 ± 0.9	29.0 ± 0.1	4.2 ± 0.4	$+0.73 \pm 0.11$	$+6.2 \pm 0.6$

IV. EVALUATION

Simulation. `mayajaal.mirage.evaluate_planners` is an *in-model simulation against a simulated attacker policy*. Stages advance with probability 0.2 per request; categories mix $P(c \mid g, s)$ with a goal-agnostic P_{gen} half and half. Diversion happens once $p_{\text{b}} < 0.35$. A lure is seen on the next request and taken with probability $E_{\kappa g}$ (*matched*) or $0.5E_{\kappa g} + 0.2$ (*misspecified*). The attacker is *bold*, *cautious* (leaves per (5)) or *shifted* (cautious with suspicion weights the defender does not know). Strategies share request, take and leave draws per seed. We report requests until $p_G(g^\star) > 0.8$ (censored at 30), share identified, requests stayed and share that left: 2,000 seeds per cell, 95% intervals over four goals.

Results (Table I). *Bold attacker.* Nothing is lost by planting, so unpaced greedy is fastest (-1.55 ± 0.12 requests vs. IG-1). MIRAGE-S plants 1.93 ± 0.01 lures per session where IG-1 plants 3 and is *slower* than IG-1 by $+0.84 \pm 0.11$ requests: the price of hedging against an attacker who might leave. Information-only lookahead matches unpaced greedy (15.75 ± 0.20 vs. 15.76 ± 0.20), so looking ahead buys nothing by itself. *Cautious attacker.* IG-1 drives $68.9 \pm 1.0\%$ of sessions away and identifies fewer than no lures ($45.5 \pm 1.0\%$ vs. $62.5 \pm 0.9\%$). MIRAGE-S loses $8.6 \pm 0.6\%$, identifies $64.9 \pm 0.9\%$ and needs -2.04 ± 0.16 requests vs. IG-1; without lookahead it loses $11.2 \pm 0.7\%$. *Wrong models.* Against the shifted attacker MIRAGE-S loses $26.8 \pm 1.0\%$ (IG-1: $94.9 \pm 0.5\%$) and identifies $54.5 \pm 1.0\%$, below no lures ($62.5 \pm 0.9\%$). With misspecified engagement it identifies $60.9 \pm 0.9\%$ of cautious sessions (IG-1: $39.3 \pm 1.0\%$). A robust variant loses fewest sessions ($4.2 \pm 0.4\%$) but is slowest against bold ones ($+2.36 \pm 0.13$), so it is not the default. *Benign.* $1.7 \pm 0.6\%$ diverted under every strategy; clinical reads are never faked.

End-to-end runs. One fixed request sequence (ward list, admin, backups, bulk export, notes, documents, an archive) runs through the live gateway as the phished nurse under each configuration; numbers come from tier tags, the ledger and the scorer (Table II). Two fresh sessions share 100% of decoy patients under the static decoy and 0% under MayaJaal, and 2 of 4 trackers were taken. A raw-byte scan of the encrypted database found 0 canaries and 0 SSN-shaped strings.

TABLE II

ONE SCRIPTED ATTACK THROUGH THE RUNNING SYSTEM, PER CONFIGURATION.

	No deception	Static decoy	MayaJaal
Real records exported	1,199	0	0
Canaries leaked (of 12)	0	0	0
Attacker reported success	yes	yes	yes
Decoy shared by 2 sessions	—	100%	0%

V. LIMITATIONS AND ETHICS

The evaluation is circular by design. Requests, engagement and the attacker's suspicion are drawn from tables we wrote; none was fitted to data. The cautious attacker is our construction, and MIRAGE-S beats the earlier planner only to the extent that real intruders react to late, dense or bursty lures roughly as (5) says. We read only the abstracts of [9], [10] and borrow ideas at that level; MIRAGE-S is not an I-POMDP solver. The end-to-end runs are one fixed script per configuration; it never touched canary-bearing fields. **False positives.** A flagged clinician is stepped up or denied, never shown false clinical data; diversions of non-clinical work need human review. **Ethics.** Nothing here shows the system would have stopped a real incident.

REFERENCES

[1] NHS England, statements on the Synnovis cyber incident affecting south-east London NHS trusts, Jun. 2024.

[2] F. Cohen, "A note on the role of deception in information protection," *Computers & Security*, vol. 17, no. 6, pp. 483–506, 1998.

[3] T. Holz and F. Raynal, "Detecting honeypots and other suspicious environments," in *Proc. IEEE SMC Information Assurance Workshop*, 2005.

[4] A. Vetterl and R. Clayton, "Bitter harvest: Systematically fingerprinting low- and medium-interaction honeypots at Internet scale," in *Proc. USENIX WOOT*, 2018.

[5] B. M. Bowen, S. Hershkop, A. D. Keromytis, and S. J. Stolfo, "Baiting inside attackers using decoy documents," in *Proc. SecureComm*, 2009.

[6] A. Juels and R. L. Rivest, "Honeywords: Making password-cracking detectable," in *Proc. ACM CCS*, 2013.

[7] D. V. Lindley, "On a measure of the information provided by an experiment," *Ann. Math. Statist.*, vol. 27, no. 4, pp. 986–1005, 1956.

[8] H. Krawczyk, M. Bellare, and R. Canetti, "HMAC: Keyed-hashing for message authentication," RFC 2104, 1997.

[9] A. Shinde, P. Doshi, and O. Setayeshfar, "Active deception using factored interactive POMDPs to recognize cyber attacker's intent," arXiv:2007.09512, 2020.

[10] P. Beltrán López, P. Nespoli, and M. Gil Pérez, "Cyber deception reactive: TCP stealth redirection to on-demand honeypots," arXiv:2402.09191, 2024.