BLACKSWAN CYBERSECURITY

THREAT INTELLIGENCE REPORT

LiteLLM Supply Chain Attack (March 24, 2026)

March 26, 2026

LiteLLM is a widely used open-source Python library and proxy (95M+ monthly PyPI downloads) that provides a single OpenAI-compatible interface for 100+ LLM providers (OpenAI, Anthropic, Groq, Azure OpenAI, etc.). It is common in AI agent frameworks, MCP servers, orchestration tools, and production AI gateways.

Key Talking Points

- **This was a classic supply-chain attack on a high-value AI dependency**, not a vulnerability in LiteLLM's code, but a compromise of the official PyPI publishing pipeline. Attackers uploaded malicious versions directly to PyPI, bypassing GitHub releases and normal CI/CD.
- **Part of a broader TeamPCP campaign** targeting cloud-native and AI tooling (previous hits: Trivy scanner, Checkmarx GitHub Actions, Aqua Security repos). The group is methodically moving through trusted open-source projects.
- **Stealth and scale are the real risk**: The malware ran silently on every Python startup (no import required in the worst case), harvested AI API keys, cloud credentials, SSH keys, K8s tokens, and more, then exfiltrated them. It also installed persistence and laterally spread across Kubernetes clusters.
- **Good news**: The window was short (a few hours on March 24), packages were quickly yanked, and official Docker images / LiteLLM Cloud were unaffected. Most environments that pin versions or build from GitHub were safe.
- **Immediate action required**: Treat any environment that installed 1.82.7 or 1.82.8 as potentially compromised and rotate all secrets.

Who Was Affected

- Anyone who ran pip install litellm (or upgraded) without a pinned version on March 24, 2026.
- CI/CD pipelines, dev laptops, Docker builds, or transitive dependencies in AI agent tools.
- **Not affected**: Official LiteLLM Docker images (ghcr.io/berriai/litellm), LiteLLM Cloud, installations from GitHub source, or versions ≤1.82.6.

Supporting Technical Details

Timeline & Root Cause

- **March 24, 2026**:
  - ~10:39–10:52 UTC: Malicious versions 1.82.7 and 1.82.8 published to PyPI (no matching GitHub tag/release).
  - Packages available for ~2–5 hours before community discovery and PyPI quarantine/removal.

BLACKSWAN CYBERSECURITY

THREAT INTELLIGENCE REPORT

LiteLLM Supply Chain Attack (March 24, 2026)

March 26, 2026

- **Root cause**: Attacker used stolen PyPI publishing credentials (likely the PYPI_PUBLISH token exfiltrated in the earlier Trivy supply-chain incident). They bypassed normal CI/CD and uploaded directly.

Affected Versions & Delivery Mechanisms

Version	Malicious Component	Execution Trigger
1.82.7	Injected payload in litellm/proxy/proxy_server.py	On import of proxy_server.py
1.82.8	litellm_init.pth (34 KB) + same payload in proxy_server.py	**Automatic on every Python interpreter startup** (no import needed)

The .pth file in 1.82.8 is especially dangerous because Python executes .pth files in site-packages/ automatically when the interpreter starts — a common technique for stealthy supply-chain malware.

What the Malware Did (Multi-Stage Payload)

1. **Collection**
   - Scans for and exfiltrates: SSH keys/configs (~/.ssh/), .env files, AWS/GCP/Azure credentials, ~/.kube/config, database passwords, .gitconfig, shell history, crypto wallets, and any secret-pattern matches.
   - Dumps environment variables and queries cloud metadata services (AWS IMDS, GCP/Azure container endpoints).
   - Specifically targets LLM API keys (e.g., OPENAI_API_KEY, ANTHROPIC_API_KEY).
2. **Exfiltration**
   - Encrypts data (AES-256-CBC + 4096-bit RSA public key).
   - Bundles into a tar archive and POSTs to attacker-controlled C2: https://models.litellm.cloud/ (not a legitimate LiteLLM domain).
3. **Persistence & Lateral Movement**
   - Installs backdoor at ~/.config/sysmon/sysmon.py + systemd user service (sysmon.service).
   - If a Kubernetes service-account token is present:
     - Reads secrets across **all namespaces**.
     - Deploys privileged alpine:latest pods (named node-setup-*) on every node in kube-system, mounting the host filesystem.
     - Spreads the backdoor cluster-wide.
   - The backdoor also polls checkmarx.zone/raw for follow-on payloads.

Indicators of Compromise (IOCs)

THREAT INTELLIGENCE REPORT

LiteLLM Supply Chain Attack (March 24, 2026)

March 26, 2026

- **Installed package:** litellm==1.82.7 or litellm==1.82.8
- **File**: litellm_init.pth in any site-packages/ directory
- **Persistence**: ~/.config/sysmon/sysmon.py and ~/.config/systemd/user/sysmon.service
- **Kubernetes artifacts**: Pods named node-setup-* in kube-system namespace
- **Network:** Outbound POSTs to models.litellm.cloud
- Cache artifacts: Look in ~/.cache/uv/ or pip caches
- In version 1.82.7: Malicious code injected into litellm/proxy/proxy_server.py (triggers on import).
- **In version 1.82.8: More dangerous and includes a malicious litellm_init.pth file (auto-executes** on **every** Python interpreter startup via Python's site module, even if LiteLLM is never imported).
- **Quick check command**:

```
Bash
pip show litellm
# or
Pip list | grep litellm
```

Filesystem / Persistence IOCs

- **litellm_init.pth** - Present in any Python site-packages/ directory (especially dangerous in v1.82.8).
  - Example locations: /usr/lib/python*/site-packages/litellm_init.pth, ~/.local/lib/python*/site-packages/, or virtualenv site-packages/.
- Persistence backdoor:
  - ~/.config/sysmon/sysmon.py
  - ~/.config/systemd/user/sysmon.service (masquerades as "System Telemetry Service")
- Temporary/exfiltration artifacts (often in /tmp/):
  - /tmp/tpcp.tar.gz (or similar encrypted archive)
  - /tmp/.pg_state
  - /tmp/pglog
  - /tmp/session.key, /tmp/payload.enc, /tmp/session.key.enc

Detection commands (run as root or with sudo for full coverage):

```
Bash
find / -name "litellm_init.pth" 2>/dev/null
find / -path "*/sysmon/sysmon.py" 2>/dev/null
find / -path "*/sysmon.service" 2>/dev/null
ls -la /tmp/tpcp.tar.gz /tmp/.pg_state /tmp/pglog 2>/dev/null
```

3. Network / C2 IOCs

- **Primary exfiltration endpoint**:
  - https://models.litellm.cloud/ (POST requests with X-Filename: tpcp.tar.gz header; **not** affiliated with legitimate LiteLLM)
- **Secondary C2 / payload polling**:
  - https://checkmarx.zone/raw (polled every ~50 minutes by the persistence script)
- **Other related domains sometimes seen in the campaign:** scan.aquasecurtiy.org, trycloudflare.com domains, or ICP-related infrastructure.

Hunt in logs for outbound HTTPS traffic from Python processes to these domains.

4. Kubernetes-Specific IOCs (if the payload reached a cluster)

- Rogue pods in the kube-system namespace:
  - Pod name pattern: node-setup-* (one per node)
  - Image: alpine:latest
  - Privileged pods mounting the host filesystem
- Unusual secret access or kubectl / find + xargs activity from compromised service accounts.

5. Behavioral / Other Indicators

- High-volume credential collection from common paths (~/.ssh/, ~/.aws/, ~/.kube/config, ~/.env files, shell histories, crypto wallets, etc.).
- Sudden outbound traffic from developer machines, CI/CD runners, or AI proxy servers shortly after a pip install/upgrade.
- Process trees showing Python spawning subprocesses for the backdoor.

Recommended Immediate Hunting Steps

1. Scan all Python environments (laptops, servers, containers, CI/CD pipelines) for the affected versions and litellm_init.pth.
2. Block the network IOCs at the firewall/proxy level.
3. If any IOC is found → **Treat the system as compromised**:
   - Isolate the host/container.
   - Rotate **all** secrets: LLM API keys (OpenAI, Anthropic, Azure OpenAI, etc.), cloud credentials (AWS IAM keys, Azure service principals), SSH keys, K8s tokens, database passwords.
   - Rebuild affected systems from clean images.
4. Monitor **CloudTrail** / **Azure Activity Logs** (as we discussed earlier) for post-exfil activity using any newly rotated keys — look for recon (GetCallerIdentity, List*), IAM changes, or high-volume S3/Storage reads.

BLACKSWAN CYBERSECURITY

THREAT INTELLIGENCE REPORT

LiteLLM Supply Chain Attack (March 24, 2026)

March 26, 2026

Official LiteLLM reference: Their security update explicitly lists litellm_init.pth and models.litellm.cloud as the primary IOCs.

Recommended Immediate Actions

1. **Scan all environments**: pip show litellm and find / -name "litellm_init.pth" 2>/dev/null.
2. **Rotate everything** on affected hosts: LLM API keys, cloud credentials, SSH keys, K8s tokens, database passwords.
3. Pin LiteLLM to a known-safe version (>=1.82.9 once released, or lock to 1.82.6).
4. Prefer official Docker images or build from source for production gateways.
5. Monitor CloudTrail / Azure Activity Logs (as discussed previously) for unusual sts:GetCallerIdentity, IAM enumeration, or high-volume S3/Storage reads from newly rotated keys.
6. Review dependency trees in all AI-related projects (many agent frameworks pull LiteLLM transitively).