0-DAY NO CVE HIGH PUBLIC

# CakePHP 5.4.2, Http Client regex suffix cookie domain

CookieCollection matches domains as a regex suffix, not RFC 6265 domain-match. api.partner.com leaks to evilapi.partner.com. Set-Cookie Domain= is not checked against the response host. Labbed 5.4.2. No CVE yet.

| | |
|---|---|
| NAME<br>CakePHP 5.4.2, Http Client regex suffix cookie domain | TYPE<br>0-day analysis |
| CVE<br>n/a | CVE RISK<br>high |
| DISCLOSURE STATUS<br>public | VENDOR<br>CakePHP |
| AFFECTED<br>CakePHP 5.4.2 (887b641) Cake\\Http\\Client default cookie jar. Stock cakephp/app does not call a second host by itself. other.com does not match. Not a browser cookie bug. | PUBLISHED<br>12 Oct 2026 |
| UPDATED<br>12 Oct 2026 | TAGS<br>0-day, cakephp, cookie, cwe-565, cwe-201, http-client |

TL;DR

Lab files and proof-of-concept **here**.

## 01 bank.com matches notbank.com

I am @abraxas_null. The proof of concept is on GitHub: **abraxas/cakephp-cookie-suffix** (loopback client). The lab stack is `lab/` : Dockerfile, docker-compose.yml, run.sh, poc.py, oracle.php. Authorized lab only. It talks to loopback.

This is **CakePHP 5.4.2** ( `887b641de9f7aba06071f98f17edcbc573cbcf63` ). No CVE yet. CWE-565 / CWE-201. 8.2 High ( `CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:L/A:N` ). Remote cookie leak / plant. App uses `Cake\Http\Client` with the default jar across hosts.

RFC 6265 domain-match is exact equality or a **dot-separated** suffix. Cake uses `preg_match('/' . preg_quote($domain) . '$/', $host)`. A cookie for `api.partner.com` is sent to `evilapi.partner.com`. A cookie for `bank.com` is sent to `notbank.com`. `other.com` does not match.

`Cookie::createFromHeaderString()` also copies `Set-Cookie` `Domain=` with no check against the response host. `addFromResponse` from `attacker.test` with `Domain=victim.internal` stores that cookie and later attaches it to `victim.internal`.

This is not a browser cookie bug. Stock `cakephp/app` does not call a second host by itself. An app that uses the default jar across OAuth, webhooks, a user-supplied URL, or a second API is the exposed case.

## 02 Why this leftover, on the same pin

I was already on 5.4.2 for the SmtpTransport pair-only strip. Http Client cookie matching is a regex suffix in the same tree. Ten hunts. Cookie jar is hunt ten.

I only wanted Critical or High. Local is in-scope. Medium stays on the floor unless it chains. Framework leftover: needs an app that keeps the jar across hosts.

## 03 Pinning 5.4.2

Same tag as the SMTP pack. `887b641`. Git clone tag 5.4.2 in PHP 8.3. The oracle runs `CookieCollection` from the pin. No extra HTTP origin is required.

Intake is security@cakephp.org.

## 04 The skip list is half the hunt

Same skip list as the SMTP pack. GHSA-vjqc, GHSA-fxf7, GHSA-2qh5 (incomplete, different leftover), GHSA-wpvj, GHSA-qh8m, GHSA-6g8q, 5.4.1 RateLimit ipHeader, 5.4.2 `filename=`. CSRF, View LFI, XXE, mass assignment, helper XSS, ungated `HTTP_CLIENT_IP` already gated. I did not re-file any of that.

Browser cookie jars are not this bug. I wrote that down before I opened `CookieCollection`.

## 05 Ten surfaces, then the suffix

CSRF / FormProtection / `_method` override: patched. View `element()` path: patched. FunctionsBuilder SQLi: patched. Paginator XSS: patched. RateLimit default: patched. Multipart `filename=`: patched.

Http Client cookie matching:

PHP Copy

```php
$pattern = '/' . preg_quote($domain, '/') . '$/';
if (!preg_match($pattern, $host)) {
```

```
    continue;
}
```

`addFromResponse` passes the request host only as a default. A `Domain=` attribute from `Set-Cookie` overwrites it and is stored.

An app that uses `Cake\Http\Client` with the default jar across more than one host can leak a partner session cookie to a suffix-confused sibling, or plant a cookie onto an internal host after fetching an attacker URL.

## 06 What I almost filed instead

Treating this as a browser cookie bug. Browsers do RFC 6265. Cake does a regex suffix.

Claiming `other.com` also matches. It does not. That is the negative control.

Inflating a Cookie header oracle to account takeover without an app that uses the jar across hosts. Lab stops at the Cookie header. Session fixation on the next Client call is inferred.

Same-host send as a finding. Cookie domain `api.partner.com` to host `api.partner.com` still sends `sid=SECRET`. Expected, not a bug.

Public-suffix rejection as the whole fix. Needed, but the first bug is the missing dot.

## 07 The lab

Git clone tag **5.4.2** in PHP 8.3. Loopback oracle HTTP at `127.0.0.1:19420`.

01 Add `sid=SECRET` domain `api.partner.com`. `addToRequest` `https://evilapi.partner.com/` → `Cookie: sid=SECRET`.

02 Control `https://other.com/` → no Cookie header.

03 Same host `https://api.partner.com/` still sends `sid=SECRET`.

04 Extra suffix `bank.com` → `https://notbank.com/` sends `sid=SECRET`.

05 `addFromResponse` from `https://attacker.test/` with `Set-Cookie: sid=x; Domain=victim.internal; Path=/`. Next `https://victim.internal/` sends `sid=x`.

Replay:

PLAIN TEXT Copy

```
git clone https://github.com/abraxas/cakephp-cookie-suffix
cd cakephp-cookie-suffix/lab
./run.sh
```

Bind it to loopback. The loopback client is `lab/run.sh` with a banner on it.

PLAIN TEXT Copy

```
SUCCESS cakephp-cookie-suffix suffix-leak=yes control-other=yes plant-domain=yes
```

## 08 Wrong turns, in the order I made them

**Browser RFC as Cake's behavior.** It is not. The regex is the leftover.

**other.com as a hit.** Control is empty. Suffix, not substring in the middle.

**Skeleton app.** Framework leftover. Needs a second host.

**A reverse shell.** Theatre. The oracle is a Cookie header.

## 09 What an attacker can do

Needs an application that keeps `Cake\Http\Client` cookies across more than one host (OAuth, webhooks, a user-supplied URL, a second API).

| PRIVILEGE | METHOD / SINK |
| --- | --- |
| **Leak a stored cookie to a suffix-confused sibling** | `CookieCollection::addToR` for `https://evilapi.partner.` after a cookie with domain `api.partner.com` |
| **Same leak for a registrable suffix** | Cookie domain `bank.com`, r host `notbank.com` |
| **Negative control** | Request host `other.com` |
| **Plant a cookie for an internal host** | `addFromResponse` of `Set-C` `sid=x; Domain=victim.inte` from `https://attacker.tes` |
| **Session fixation / stolen partner session on the next Client call** | Planted or leaked `sid` used API session |
| **Same-host send still works** | Cookie domain `api.partner` to host `api.partner.com` |

Closed on this leftover: `other.com` does not match. Browser cookie jars are not this bug. No shell. Not unauthenticated RCE on a skeleton app.

## 10 The fix

RFC 6265 domain-match (`host === domain || str_ends_with($host, '.' . $domain)`). Host-only cookies when `Domain` is omitted. Reject `Domain` that does not domain-match the **request** host. Reject public suffixes.

Until then, do not share a `Cake\Http\Client` cookie jar across hosts you do not already trust with each other's cookies. Per-host clients are a workaround, not a fix.

I am not going to print a Set-Cookie you can paste at someone else's webhook fetch. The missing dot is the useful part. If you own the app, run the proof of concept against loopback.

## 11 References


- Proof of concept: abraxas/cakephp-cookie-suffix · cakephp-cookie-suffix-Abraxas-Labs.py
- Lab: `lab/` · Dockerfile · docker-compose.yml · run.sh · poc.py · oracle.php
- @abraxas_null · github.com/abraxas · abraxaslabs.tech · abraxas.null@proton.me
- CWE-565 · CWE-201
- Tree: cakephp/cakephp 5.4.2 (`887b641de9f7aba06071f98f17edcbc573cbcf63`) · `CookieCollection.php` · `Cookie.php`

- RFC 6265 domain-match
- Same pin: cakephp-smtp-crlf · cakephp-trusted-proxies

**Abraxas Labs**

Independent security research by abraxas.

Analysis of 0-days, n-days, malware, and the

leftovers that still matter.

**GitHub** **@abraxas_null** **abraxas.null@proton.me** **abraxaslabs.tech**



Independent research. Private analysis by arrangement. No ads. abraxas.null@proton.me · For defensive awareness and authorized testing only.

© 2026 Abraxas Labs