ONE REDIRECT · TWO PROBLEMS

apply.guilford.edu does not open the application

Typing the bare **apply.guilford.edu** does not reach Slate at all. It redirects off the domain to the marketing admissions page — and **discards every URL parameter** on the way, including the Google Ads click ID. Two separate problems, one redirect.

The application itself is healthy and lives at **apply.guilford.edu/apply/**. Only the bare domain is affected.

REPORTED BY	VERIFIED	LIKELY OWNER	EFFORT
Ellis Agency	28 Jul 2026	Slate admin	One redirect rule

1 The bare domain never reaches Slate

The application system is healthy and lives at `apply.guilford.edu/apply/`. But the **root of the domain** redirects visitors off Slate entirely, to the marketing admissions page — and strips the URL parameters on the way.

What is on the domain

URL	Result	
`/apply/`	**200** — "Application Management"	The application. Working.
`/register/?id=…`	**200**	Enquiry & event forms. Working.
`/portal/campus_visit`	**200**	Visit scheduling. Working.
`/ (bare domain)`	**302** → www.guilford.edu/admissions	Leaves Slate; parameters discarded.

TWO SEPARATE QUESTIONS — PLEASE ANSWER BOTH

1 · Is that destination intended? Anyone who types `apply.guilford.edu` from a flyer, a QR code, or a college fair almost certainly expects to land on the application. Today they land on a marketing page and have to find "Apply" again. If the intent is that this address opens the application, the root should point at `https://apply.guilford.edu/apply/`. We cannot tell from outside whether the current destination was a deliberate choice — **please confirm.**

2 · The redirect discards the query string, whichever destination you choose. That part is a defect regardless and is covered on the next page.

Reproduce it

```
$ curl -sSL -o /dev/null -w '%{url_effective}\n' \
    "https://apply.guilford.edu/?utm_source=test&gclid=ABC123"
https://www.guilford.edu/admissions            ← off Slate, parameters gone

$ curl -sSL -o /dev/null -w '%{url_effective}\n' \
    "https://apply.guilford.edu/apply/?utm_source=test"
https://apply.guilford.edu/apply/?utm_source=test  ← the application, parameters kept
```

The chain from the bare domain is three hops, the first of them insecure: `302 →` `http://www.guilford.edu/admission/`, then `301` to HTTPS, then `301` to `/admissions`.

WHO OWNS THIS — PLEASE READ BEFORE ROUTING THE TICKET

The 302 is issued by **Slate**, not by the web server behind `www.guilford.edu` and not by a CDN — the response sets `_node`, `_hash` and `_hashV`, which are Technolutions session cookies. So the redirect is configured in Slate, and the change belongs with whoever administers it. Tickets like this otherwise bounce between the web team and enrollment for a fortnight.

2 Why this is worth a ticket

Beyond sending applicants to the wrong place, the same redirect destroys two kinds of tracking. The first is the expensive one.

A Google Ads cannot attribute conversions

THE EXPENSIVE ONE

`gclid` is the click identifier Google Ads stamps on every paid click. It is how Ads later matches that click to an application or enquiry. Strip it and Ads never learns the click converted — so the spend reports **zero conversions**, and because Smart Bidding optimises on conversion signal, the algorithm is actively being taught that those clicks are worthless.

B Google Analytics credits the visit to nobody

GA4 reads `utm_source`, `utm_medium` and `utm_campaign` to attribute a session. Without them the visit is credited to whatever the referrer says — usually **Direct**. The visitor still counts; they just count as though nobody sent them, which understates every campaign that uses this address.

C Minor: an insecure hop and two redundant redirects

The first hop targets `http://` rather than `https://`, so there is a brief plaintext request before HSTS upgrades it. Collapsing the chain to one hop also removes two round trips from every visit.

The fix

Once you have settled the destination (page 2), the redirect must **append the incoming query string** and target HTTPS in a single hop. Our recommendation is that the root opens the application — `https://apply.guilford.edu/apply/` — but that is your call, and the query-string requirement holds either way.

```
# nginx — the omitted $is_args$args is the usual culprit
return 301 https://apply.guilford.edu/apply/$is_args$args;

# Apache mod_alias — preserves the query string by default
Redirect 301 / https://apply.guilford.edu/apply/

# Cloudflare Bulk Redirect / Rules
Enable "Preserve query string" on the rule
```

Verifying the fix

```
$ curl -sSL -o /dev/null -w '%{url_effective}  (%{num_redirects} hop)\n' \
    "https://apply.guilford.edu/?utm_source=test&gclid=ABC123"

expected — parameters intact, one hop, HTTPS:
https://apply.guilford.edu/apply/?utm_source=test&gclid=ABC123  (1 hop)
```

Substitute your chosen destination. Parameters present, one hop, HTTPS throughout — and please re-run the second command from page 2 to confirm the working paths are unchanged.

ONE THING WE CANNOT SEE FROM OUTSIDE

We have verified the defect, not its volume. Whether current advertising points at the bare `apply.guilford.edu` depends on the landing URLs held by your enrollment-marketing vendor — a question they can answer in minutes, and the answer turns this from a certainty into a figure. We raised it regardless: a memorable vanity domain is exactly what ends up on printed material and QR codes.