

Not everyone respects the rules

implementing Varnish on existing sites

By Thijs Feryn



A photograph of a man with glasses and a striped shirt speaking into a microphone on a stage. A large white speech bubble graphic is overlaid on the image, containing the text "Hi my name is Thijs".

Hi
my name
is Thijs





I'm
an evangelist at





I'm
a board member
at





FOLLOW ME ON
TWITTER
@THIJSFERYN



Combell
Your host on the internet

We all know Varnish

Right?



*We know how to
Install it
Right?*



We know how to
Configure it
Right?



We know how to
Work with ucl
Right?



A close-up photograph of a man's face, looking slightly upwards and to the right with a thoughtful expression. He has dark hair and blue eyes. A white speech bubble originates from his mouth, containing the text "WTF?" in a bold, black, sans-serif font. The background is a dark, textured surface with faint, white, abstract markings resembling circuit boards or network diagrams.

WTF?

Quick reminder!



```
curl http://repo.varnish-cache.org/debian/GPG-key.txt | apt-key add -
```

```
echo "deb http://repo.varnish-cache.org/debian/ squeeze varnish-3.0" >> /etc/apt/sources.list
```

```
apt-get update
```

```
apt-get install varnish
```

Install & configure

```
DAEMON_OPTS="-a :80 \
-T localhost:6082 \
-f /etc/varnish/default.vcl \
-S /etc/varnish/secret \
-s malloc,256m"
```

In “*/etc/default/varnish*”

A photograph of a person from the waist up, wearing a white t-shirt and blue jeans. They are holding their lower back with both hands, suggesting pain or discomfort. The background is a plain, light color.

Backend



Backend

Listen 8080

In “`/etc/apache2/ports.conf`”

Backend

```
backend default {  
    .host = "127.0.0.1";  
    .port = "8080";  
}
```

In “*/etc/varnish/default.vcl*”

```
# sub vcl_recv {
#   if (req.restarts == 0) {
#     if (req.http.x-forwarded-for) {
#       set req.http.X-Forwarded-For =
#           req.http.X-Forwarded-For + ", " + client.ip;
#     } else {
#       set req.http.X-Forwarded-For = client.ip;
#     }
#   }
#   if (req.request != "GET" &&
#       req.request != "HEAD" &&
#       req.request != "PUT" &&
#       req.request != "POST" &&
#       req.request != "TRACE" &&
#       req.request != "OPTIONS" &&
#       req.request != "DELETE") {
#     /* Non-RFC2616 or CONNECT which is weird. */
#     return (pipe);
#   }
#   if (req.request != "GET" && req.request != "HEAD") {
#     /* We only deal with GET and HEAD by default */
#     return (pass);
#   }
#   if (req.http.Authorization || req.http.Cookie) {
#     /* Not cacheable by default */
#     return (pass);
#   }
#   return (lookup);
# }
```

```
# sub vcl_hash {
#     hash_data(req.url);
#     if (req.http.host) {
#         hash_data(req.http.host);
#     } else {
#         hash_data(server.ip);
#     }
#     return (hash);
# }
# sub vcl_fetch {
#     if (beresp.ttl <= 0s ||
#         beresp.http.Set-Cookie ||
#         beresp.http.Vary == "*") {
#         /*
#          * Mark as "Hit-For-Pass" for the next 2 minutes
#          */
#         set beresp.ttl = 120 s;
#         return (hit_for_pass);
#     }
#     return (deliver);
# }
```


The rules

- ✓ Use appropriate **cache-control** headers
- ✓ Use **s-maxage** for expiration
- ✓ Use Surrogate-Capability & Surrogate-Control headers for **ESI** based block caching
- ✓ Avoid using **cookies** for cached pages
- ✓ Use **vary** headers to extend the hash
- ✓ Only cache **GET** or **HEAD**
- ✓ Use **consistent URL's**





No clue

Don't
give
a
sh*t



http://



I work in the
Hosting
Industry

We don't get to choose



the code

we work
with





THINGS I
LEARNED THE
HARD WAY

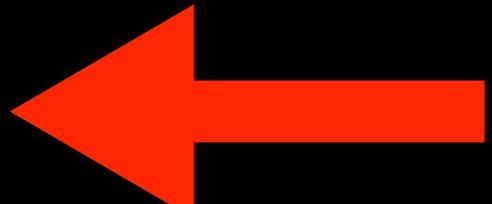
Things I've Learned

- ✓ Varnish default behaviour is **bypassed** with “return”
 - *set-cookie, cookie, max-age, post, ...*
- ✓ beresp.ttl > cache-control: max-age
- ✓ **No cache** headers are ignored (except max-age=0)
- ✓ **Purge** doesn't work with custom **vcl_hash**
- ✓ **Vary** is supported
- ✓ There's a **hit for pass** cache

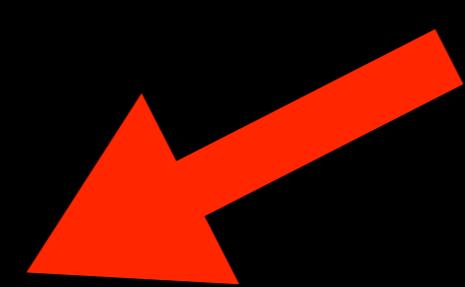
Out of the box



```
# sub vcl_recv {
#   if (req.restarts == 0) {
#     if (req.http.X-Forwarded-For) {
#       set req.http.X-Forwarded-For =
#           req.http.X-Forwarded-For + ", " + client.ip;
#     } else {
#       set req.http.X-Forwarded-For = client.ip;
#     }
#   }
#   if (req.request != "GET" &&
#       req.request != "HEAD" &&
#       req.request != "PUT" &&
#       req.request != "POST" &&
#       req.request != "TRACE" &&
#       req.request != "OPTIONS" &&
#       req.request != "DELETE") {
#     /* Non-RFC2616 or CONNECT which is weird. */
#     return (pipe);
#   }
#   if (req.request != "GET" && req.request != "HEAD") {
#     /* We only deal with GET and HEAD by default */
#     return (pass);
#   }
#   if (req.http.Authorization || req.http.Cookie) {
#     /* Not cacheable by default */
#     return (pass);
#   }
#   return (lookup);
# }
```



```
# sub vcl_hash {
#     hash_data(req.url);
#     if (req.http.host) {
#         hash_data(req.http.host);
#     } else {
#         hash_data(server.ip);
#     }
#     return (hash);
# }
# sub vcl_fetch {
#     if (beresp.ttl <= 0s ||
#         beresp.http.Set-Cookie ||
#         beresp.http.Vary == "*") {
#         /*
#          * Mark as "Hit-For-Pass" for the next 2 minutes
#          */
#         set beresp.ttl = 120 s;
#         return (hit_for_pass);
#     }
#     return (deliver);
# }
```





HTTP request

```
GET /drupal/?q=node/1 HTTP/1.1
Host: 12.12.12.6
User-Agent: Mozilla/5.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*
;q=0.8
Accept-Language: nl,en;q=0.7,fr-be;q=0.3
Accept-Encoding: gzip, deflate
Connection: keep-alive
Referer: http://12.12.12.6/drupal/
Cookie: has_js=1
```

HTTP response

HTTP/1.1 200 OK
Date: Fri, 21 Sep 2012 11:04:52 GMT
Server: Apache/2.2.14 (Ubuntu)
X-Powered-By: PHP/5.3.2-1ubuntu4.18
Expires: Sun, 19 Nov 1978 05:00:00 GMT
Last-Modified: Fri, 21 Sep 2012 11:04:52 +0000
Cache-Control: no-cache, must-revalidate, post-check=0, pre-check=0
Etag: "1348225492"
Content-Language: en
Link: </drupal/?q=node/1>; rel="shortlink",</drupal/?q=node/1>;
rel="canonical"
X-Generator: Drupal 7 (<http://drupal.org>)
Vary: Accept-Encoding
Content-Encoding: gzip
Content-Length: 2710
Keep-Alive: timeout=15, max=95
Connection: Keep-Alive
Content-Type: text/html; charset=utf-8



No
real issues
when you
don't login





HTTP response

HTTP/1.1 200 OK

Date: Fri, 21 Sep 2012 11:22:38 GMT

Server: Apache/2.2.14 (Ubuntu)

X-Powered-By: PHP/5.3.2-1ubuntu4.18

Set-Cookie: frontend=7p8g4e0njnvdg28luppgi5qp64; expires=Fri, 21-Sep-2012
12:22:38 GMT; path=/magento; domain=12.12.12.6; HttpOnly

Expires: Thu, 19 Nov 1981 08:52:00 GMT

Cache-Control: no-store, no-cache, must-revalidate, post-check=0, pre-
check=0

Pragma: no-cache

Vary: Accept-Encoding

Content-Encoding: gzip

Content-Length: 2816

Keep-Alive: timeout=15, max=100

Connection: Keep-Alive

Content-Type: text/html; charset=UTF-8



Custom code

State
is our
enemy



Cookies



Cookies



*HTTP cookie
request header
via browser*

*HTTP set-cookie
response header
via webserver*

We use cookies for

✓ Sessions

✓ Google Analytics

✓ Language

preferences



What do we do?



Nothing!



Remove cookies



Remove client cookies

```
sub vcl_recv {  
    unset req.http.cookie;  
}
```



Remove server cookies

```
sub vcl_fetch {  
    unset beresp.http.set-cookie;  
}
```



Remove some cookies

```
sub vcl_recv {  
    if (req.http.Cookie) {  
        set req.http.Cookie =  
            regsuball(req.http.Cookie, "^(^|; \s*)(_[a-z]+|  
has_js)=[^;]*", ""); *,"\\1");  
        if (req.http.Cookie == "") {  
            remove req.http.Cookie;  
        }  
    }  
}
```

A black and white photograph showing a close-up of a person's arm. A tattoo is visible on the upper arm, with the words "DON'T FORGET" written in a stylized font. A hand is shown gripping the person's elbow from behind. The background is plain and light-colored.

*Ignore
cookies*

Ignore cookies

```
sub vcl_recv {  
    if (req.request = "GET" || req.request =  
"HEAD") {  
        return (lookup);  
    }  
}
```



Ignores default behaviour

Add cookie to hash



Add a specific cookie to hash

```
sub vcl_recv {  
    if (!req.http.Cookie ~ "lang") {  
        return(pass);  
    }  
}  
  
sub vcl_hash {  
    if(req.http.Cookie ~ "lang"){  
        hash_data(regsuball(req.http.Cookie, "^.+;? ?  
lang=([a-zA-Z0-9]+)( |;| ;).*$", "\1"));  
    }  
}
```

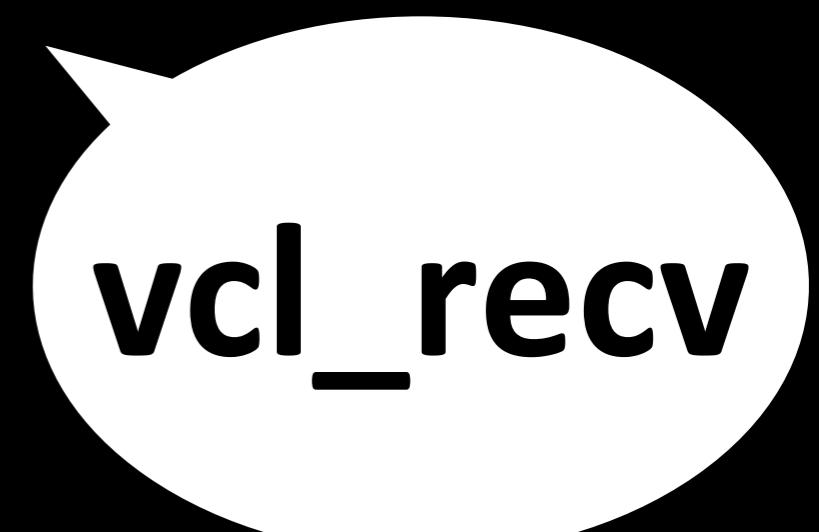


To the
drawing
board

Because there's always something custom

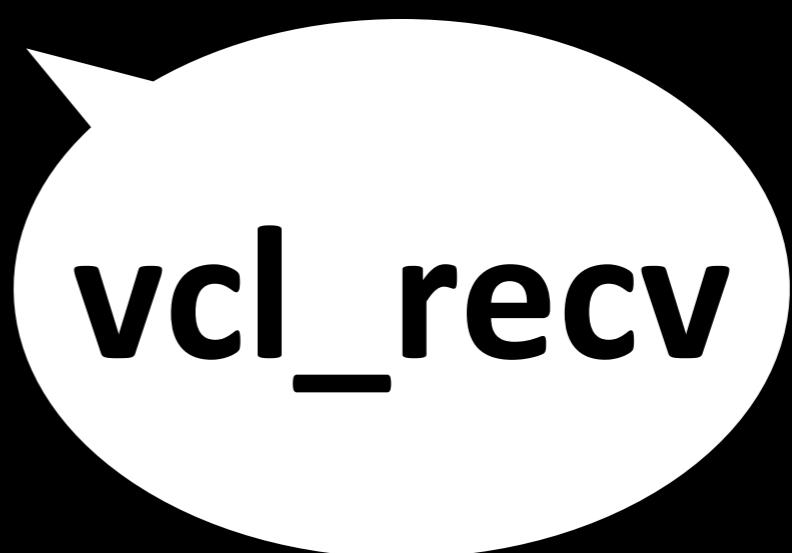
```
#CACHE STATIC FILES
if (req.url ~ "\.(gif|jpg|jpeg|swf|flv|mp3|mp4|pdf|ico|png|gz|
tgz|bz2)(\?.*|)$") {
    unset req.http.cookie;
    set req.url = regsub(req.url, "\?.*?", "");
    return (lookup);
}
#DON'T CACHE THESE
if(req.url ~ "^/(nl|fr)/(product|contest)([0-9]*)-mail"){
    return(pass);
}
if(req.url ~ "^/sales"){
    return(pass);
}
if(req.url ~ "^/redeem-voucher"){
    return(pass);
}
if (req.url ~ "^/wp-(login|admin|signup)" || req.url ~
"preview=true" || req.url ~ "^/xmlrpc.php" || req.url ~ "^/admin-
ajax.php") {
    return(pass);
}

#DON'T CACHE AUTH
if (req.http.Authorization) {
    return (pass);
}
```

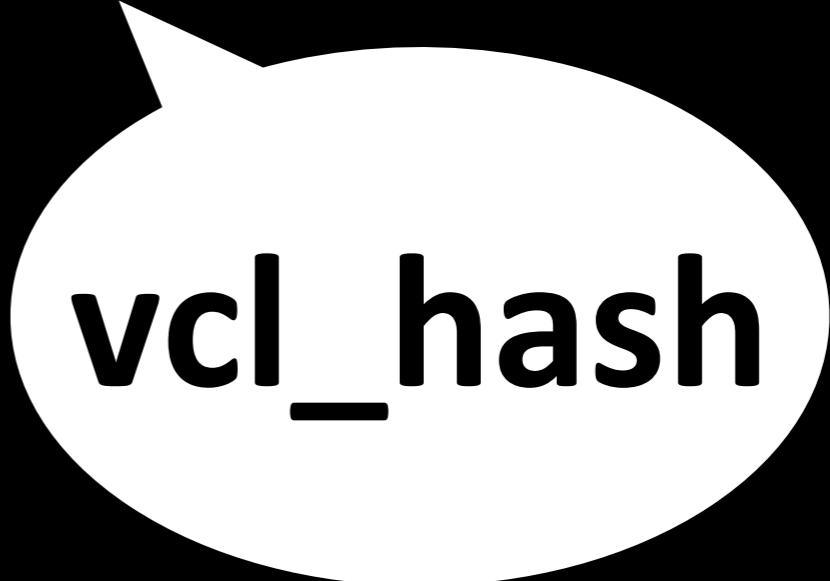


```
#KEEP LANGUAGE COOKIE
if(req.http.Cookie ~ "lang"){
    set req.http.Cookie = ";" + req.http.Cookie;
    set req.http.Cookie = regsuball(req.http.Cookie, "; +",
";");
    set req.http.Cookie = regsuball(req.http.Cookie, ";(lang)=", "; \1=");
    set req.http.Cookie = regsuball(req.http.Cookie, ";[^ ]"
[^;]*", "");
    set req.http.Cookie = regsuball(req.http.Cookie, "^[" ]"
[; ]+$", "");
}

#GET FROM CACHE
return (lookup);
```



#IF LANG COOKIE IS SET, ADD IT TO THE HASH
if(req.http.Cookie ~ "lang"){
 hash_data(regsuball(req.http.Cookie, "^.
+;? ?(lang=[a-zA-Z0-9]+)(|;| ;).*\\$", "\1"));
}



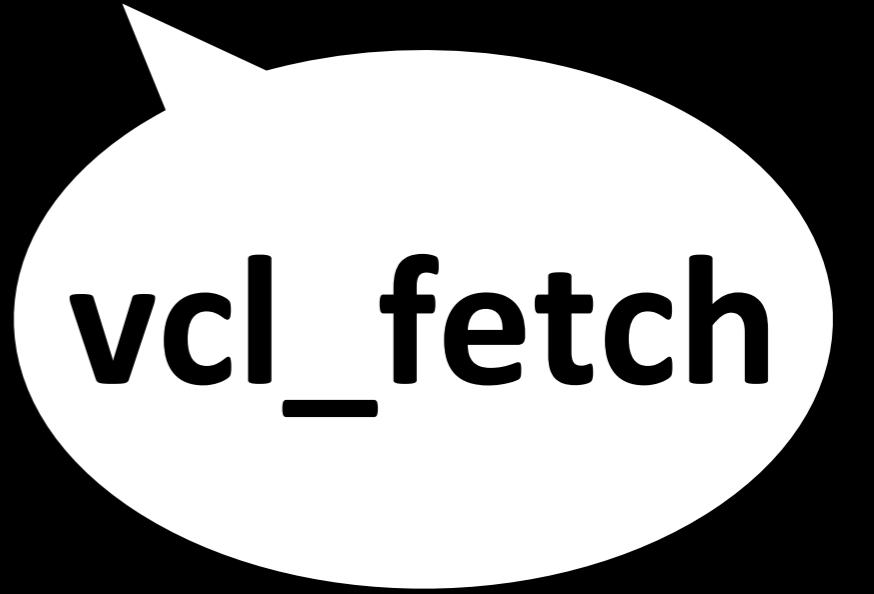
vcl_hash

Or just use
the vary
header



```
#PUT THESE ON THE HIT FOR PASS BLACKLIST
if (req.url ~ "^/wp-(login|admin|signup)" ||
req.url ~ "preview=true" || req.url ~ "^/
xmlrpc.php" || req.url ~ "^/admin-ajax.php") {
    return (hit_for_pass);
}
```

```
#DEFINE TIME TO LIVE
if(req.url ~ "^/(nl|fr)/(sendlist|orders)?"
ordercode=(.+.)){
    set beresp.ttl = 600s;
} else {
    set beresp.ttl = 3600s;
}
```



vcl_fetch



Secure



GET ASYNC OR GET SUNK

hope



Devolution



Devolution

- ✓ Better frameworks
- ✓ CMS framework adoption
- ✓ Smarter developers
- ✓ DevOps evangelization



Already using a
**Page
cache**



They know about

Stale data

They use



```
<esi:include src="/bla"/>
```

Or esi

Surrogate
Capability

Surrogate
Control



Replacing a Copy





A close-up, low-angle shot of an antique typewriter. The machine is dark-colored with a prominent red ribbon visible across the keyboard. The background is a plain, light-colored wall.

THE END

Thanks

