403 Forbidden vs 401 Unauthorized HTTP responses

Asked 9 years, 2 months ago Active 15 days ago Viewed 940k times

2475

For a web page that exists, but for which a user that does not have sufficient privileges, (they are not logged in or do not belong to the proper user group), what is the proper HTTP response to serve? 401? 403? Something else? What I've read on each so far isn't very clear on the difference between the two. What use cases are appropriate for each response?

http-headers http-status-code-403 http-status-codes http-status-code-401 http-response-codes

598

edited Nov 17 '15 at 13:24
MK-rou
329 2 4 19

asked Jul 21 '10 at 7:21
VirtuosiMedia
23.5k 18 89 136

260 401 'Unauthorized' should be 401 'Unauthenticated', problem solved ! – Christophe Roussy May 17 '16 at 12:33

31 I don't remember how many times me and my colleagues have come back to stackoverflow for this question. Maybe HTTP standards should consider modifying the names or descriptions for 401 and 403. – neurite Feb 4 '17 at 1:14

In fact, I am getting a different version of this error. like "os_authType was 'any' and an invalid cookie was sent". So unable to figure out how to solve that. Googled a lot of time , got reasons but didn't get a solution. – Sandeep Anand Feb 12 '18 at 13:59

@Qwerty no, the new RFC7231 obsoletes RFC2616. 403 has a different meaning now. – fishbone Aug 1 '18 at 13:15

@fishbone you also did not note that status code 401 has been removed from that RFC :D – Martin Barker Nov 22 '18 at 12:06

15 Answers

3696

A clear explanation from Daniel Irvine:

> There's a problem with *401 Unauthorized*, the HTTP status code for authentication errors. And that's just it: it's for authentication, not authorization. Receiving a 401 response is the server telling you, "you aren't authenticated–either not authenticated at all or

By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.

This is a response generally returned by your web server, not your web application.

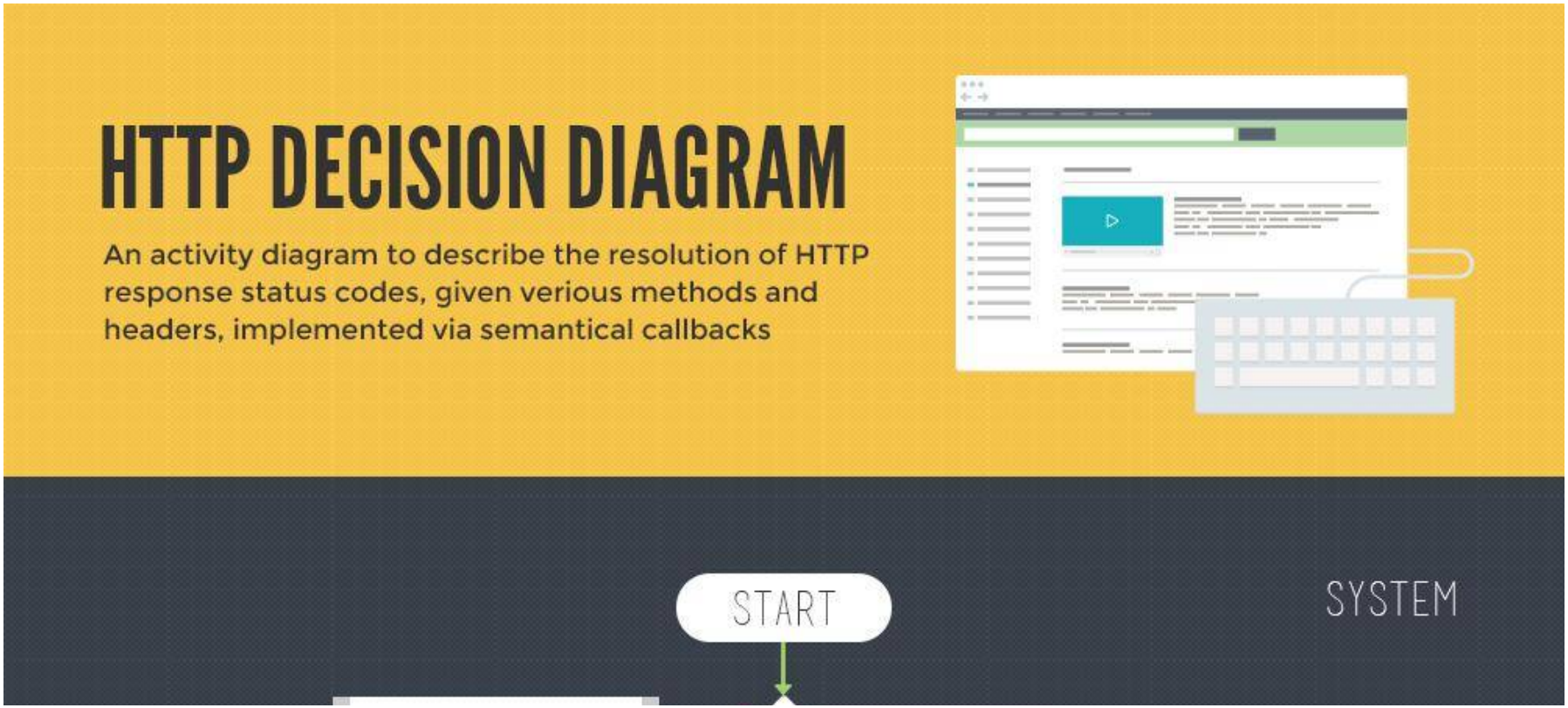
It's also something very temporary; the server is asking you to try again.

So, for authorization I use the *403 Forbidden* response. It's permanent, it's tied to my application logic, and it's a more concrete response than a 401.

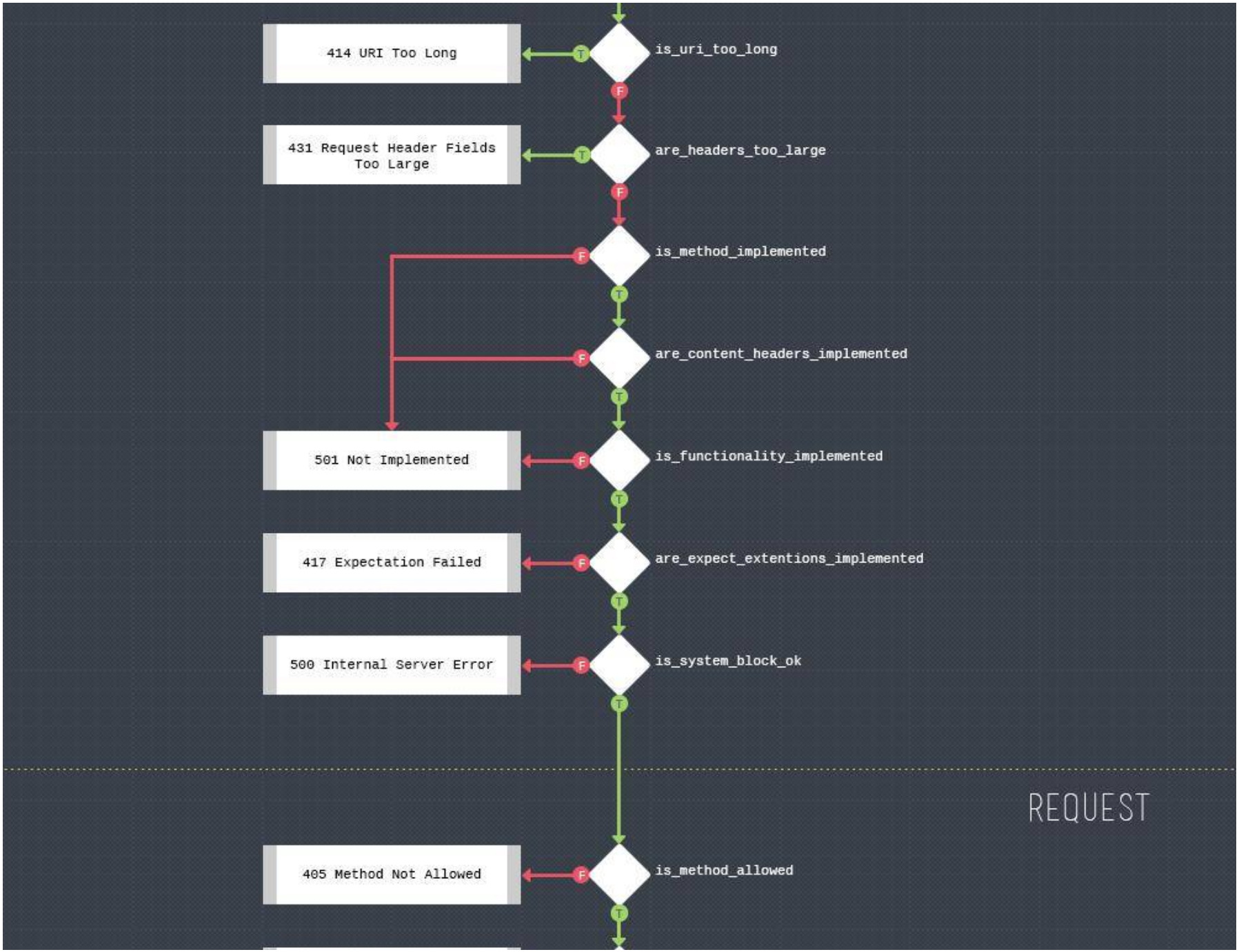
Receiving a 403 response is the server telling you, "I'm sorry. I know who you are–I believe who you say you are–but you just don't have permission to access this resource. Maybe if you ask the system administrator nicely, you'll get permission. But please don't bother me again until your predicament changes."

In summary, a *401 Unauthorized* response should be used for missing or bad authentication, and a *403 Forbidden* response should be used afterwards, when the user is authenticated but isn't authorized to perform the requested operation on the given resource.

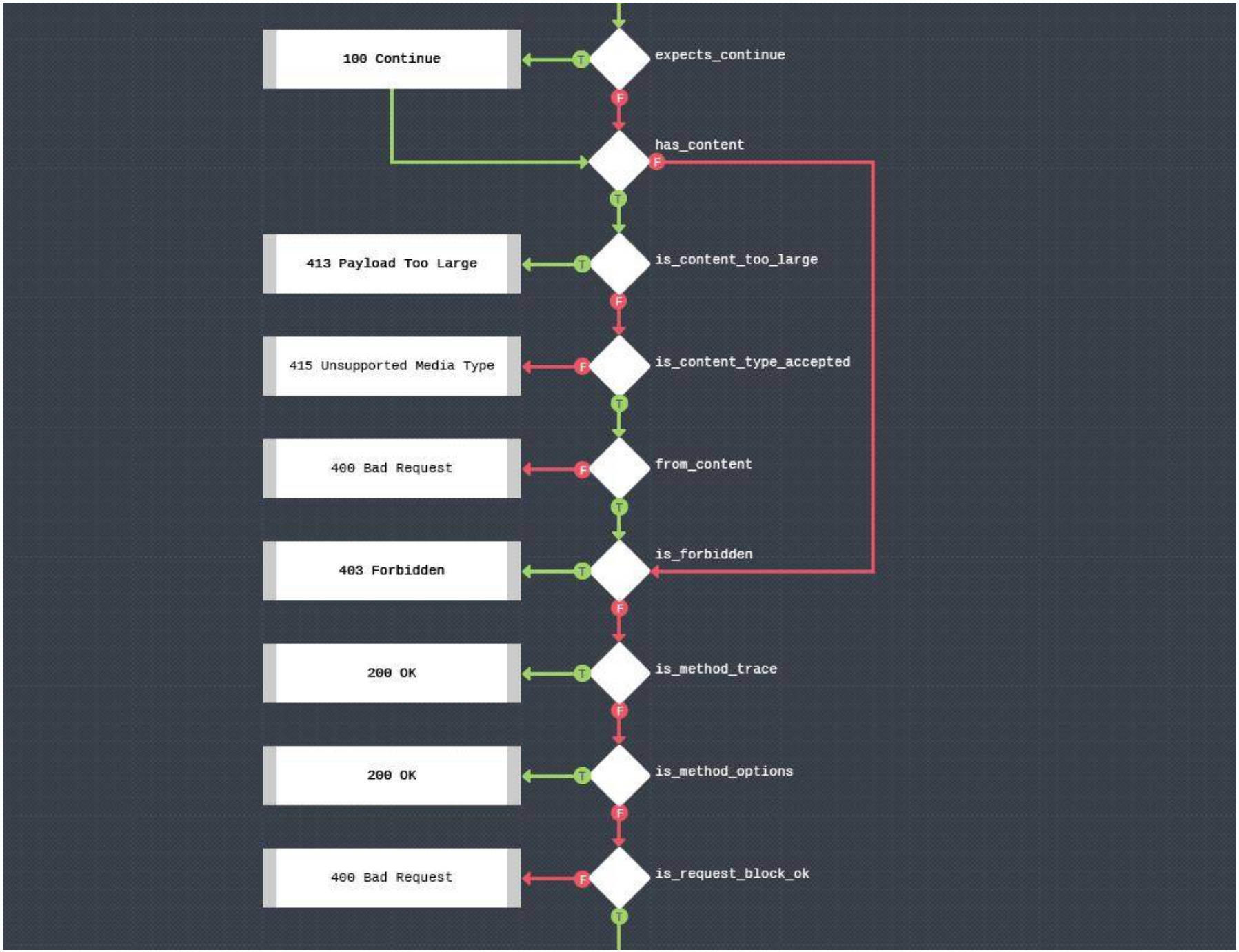
Another nice pictorial format of how http status codes should be used.



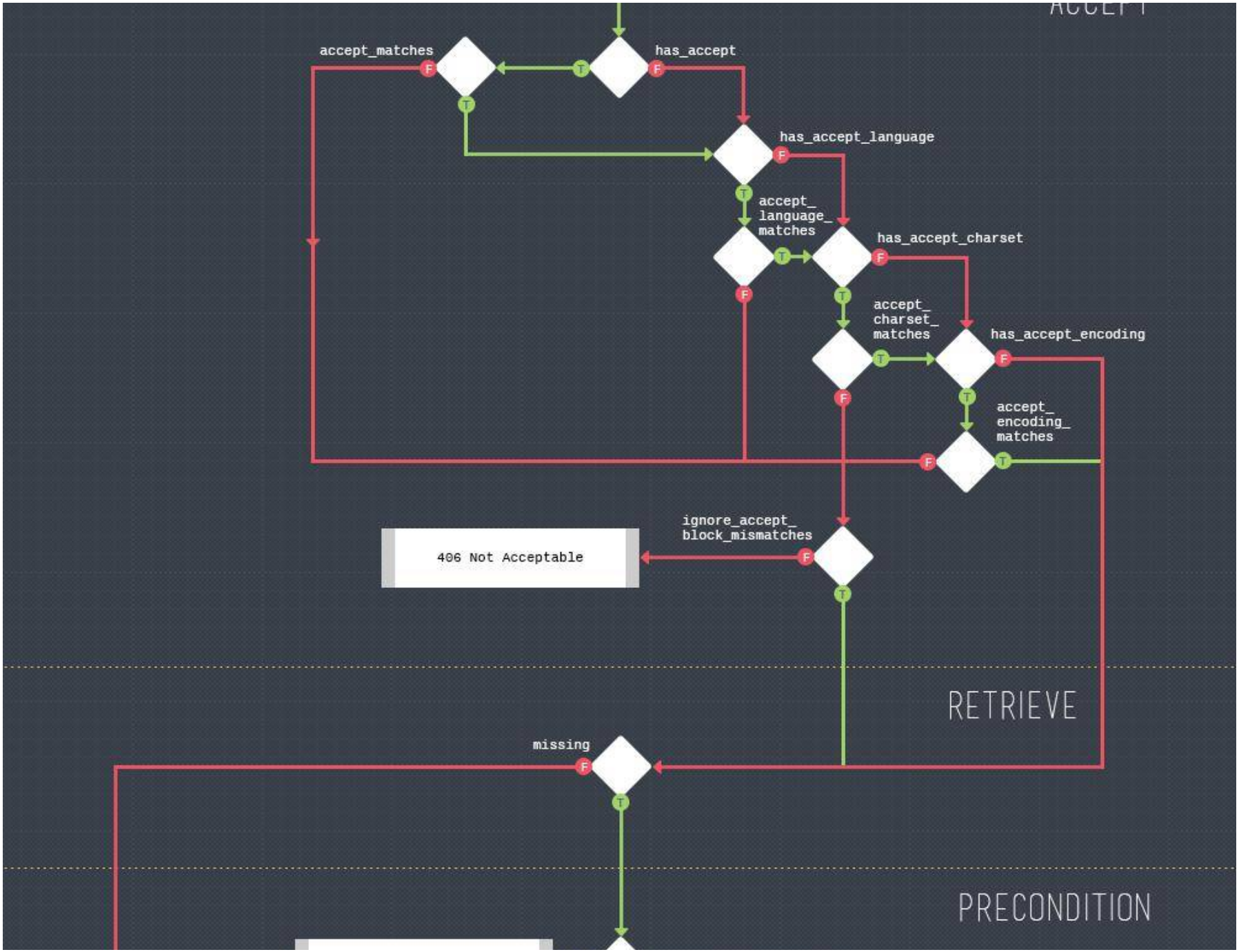
By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.



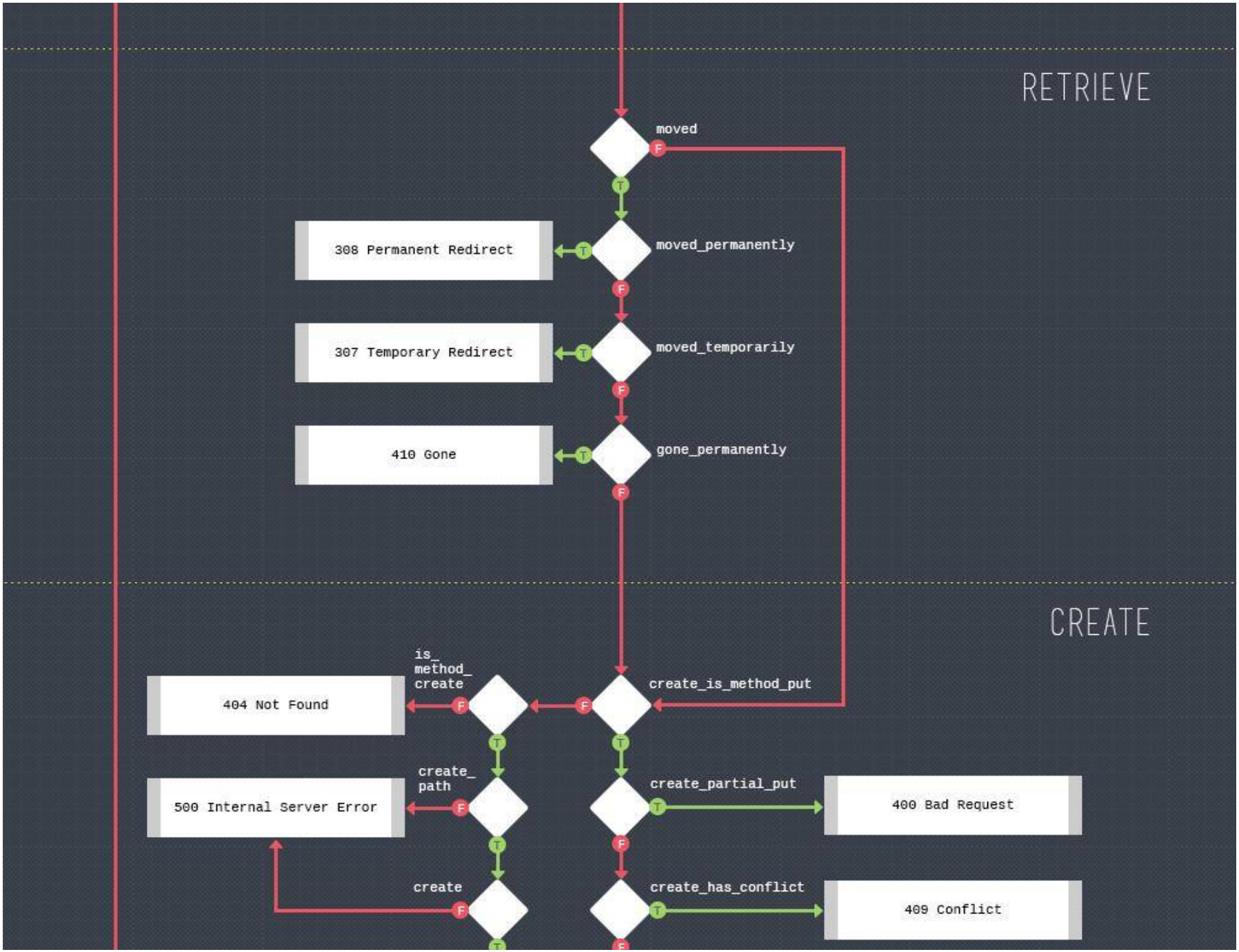
By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.



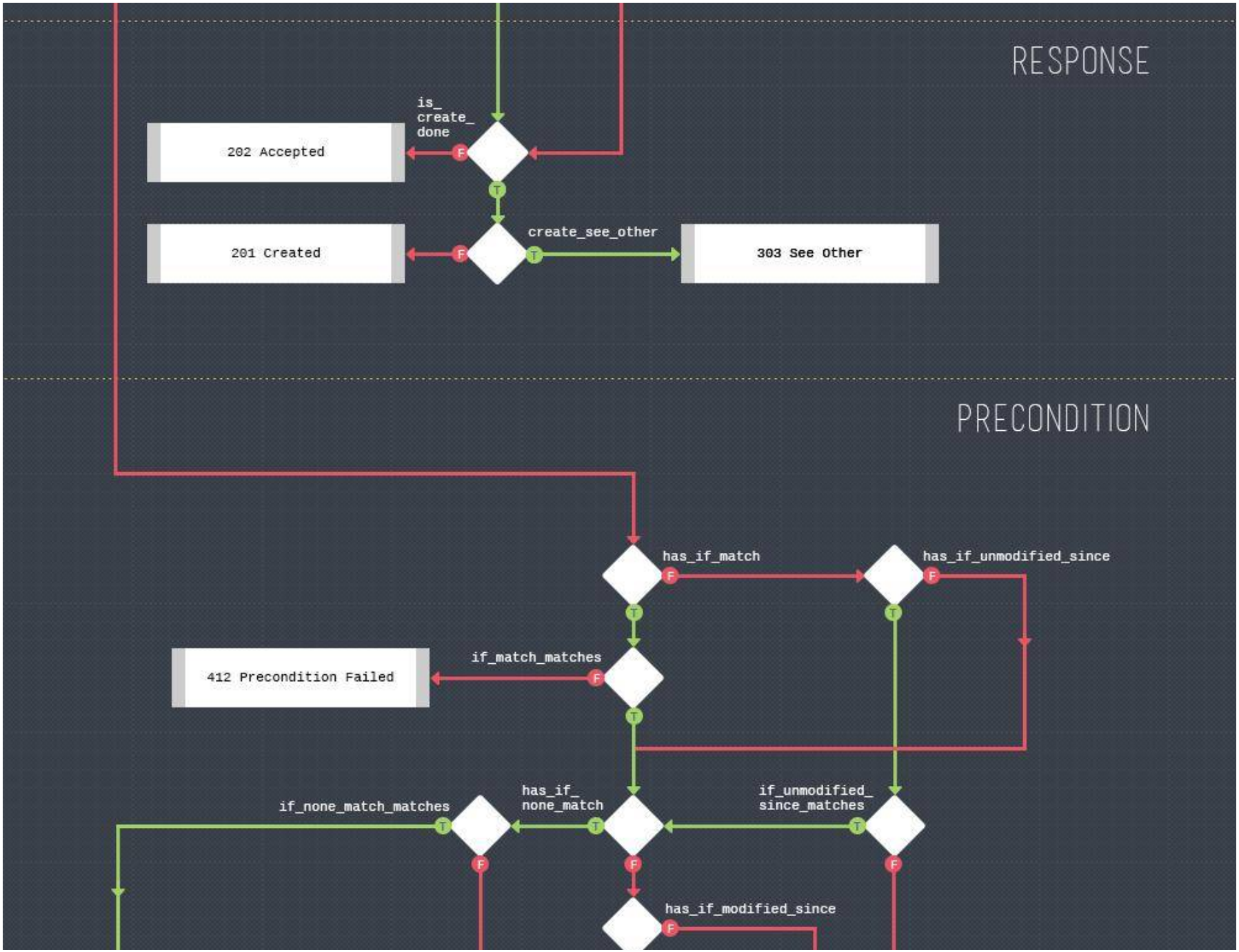
By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.



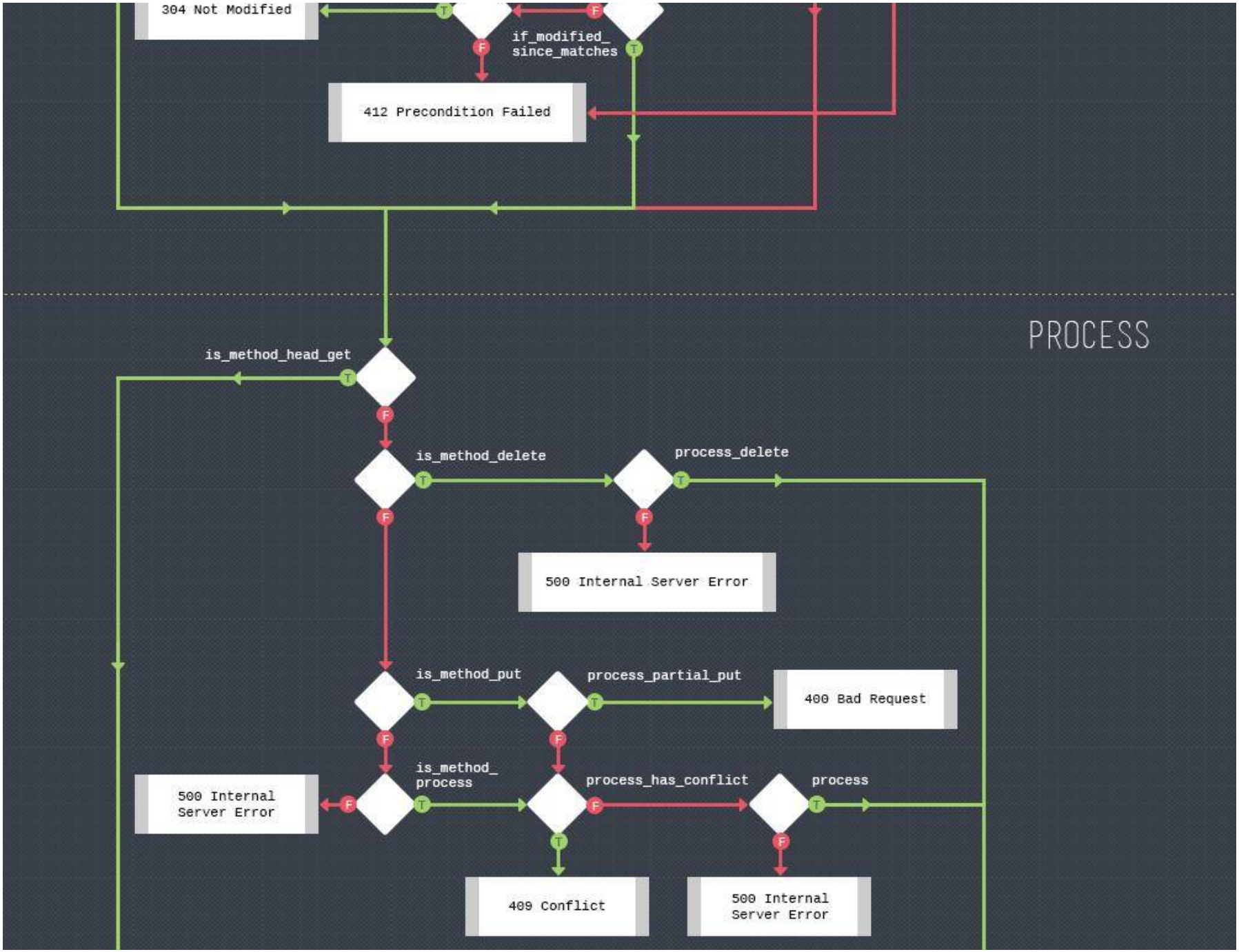
By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.



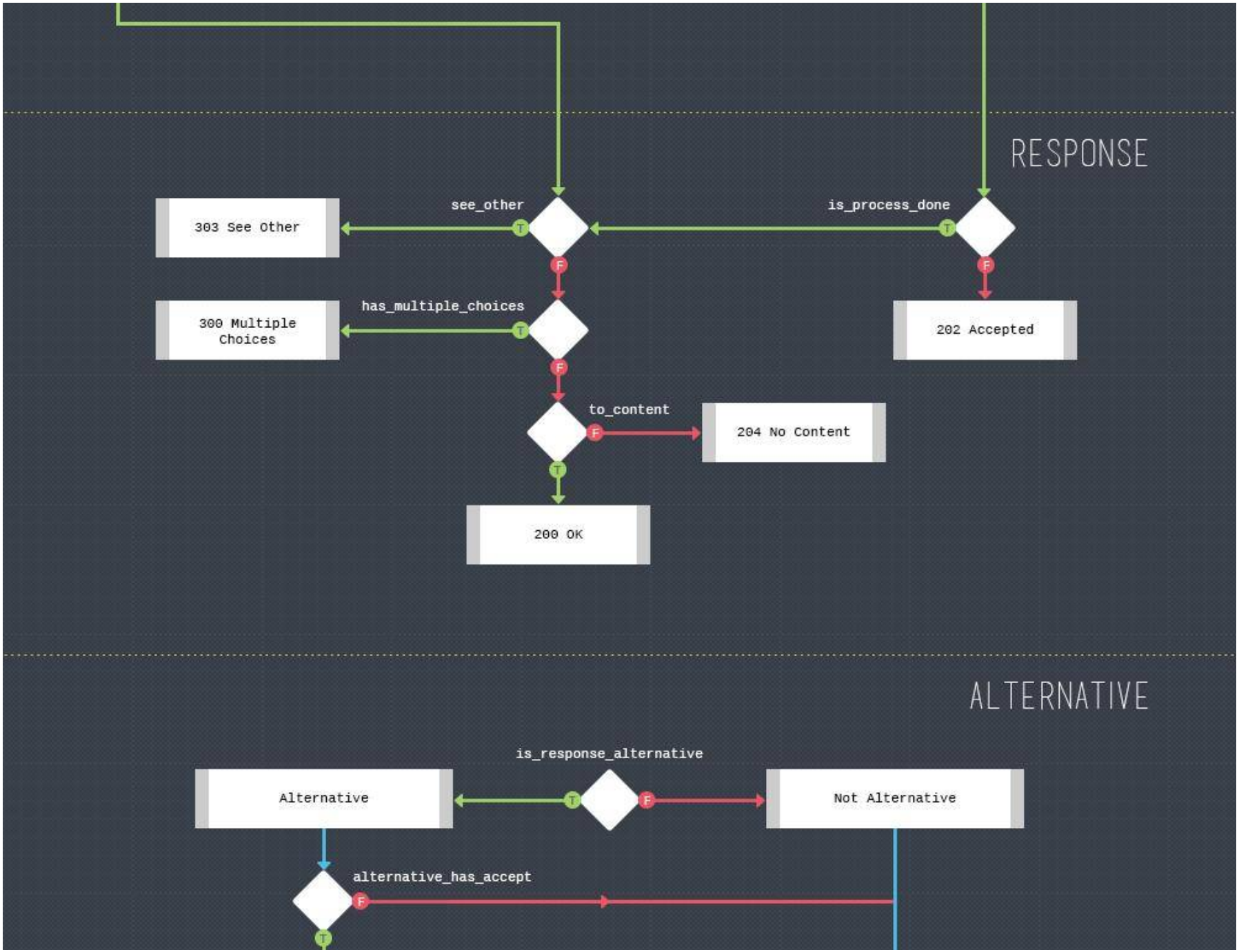
By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.



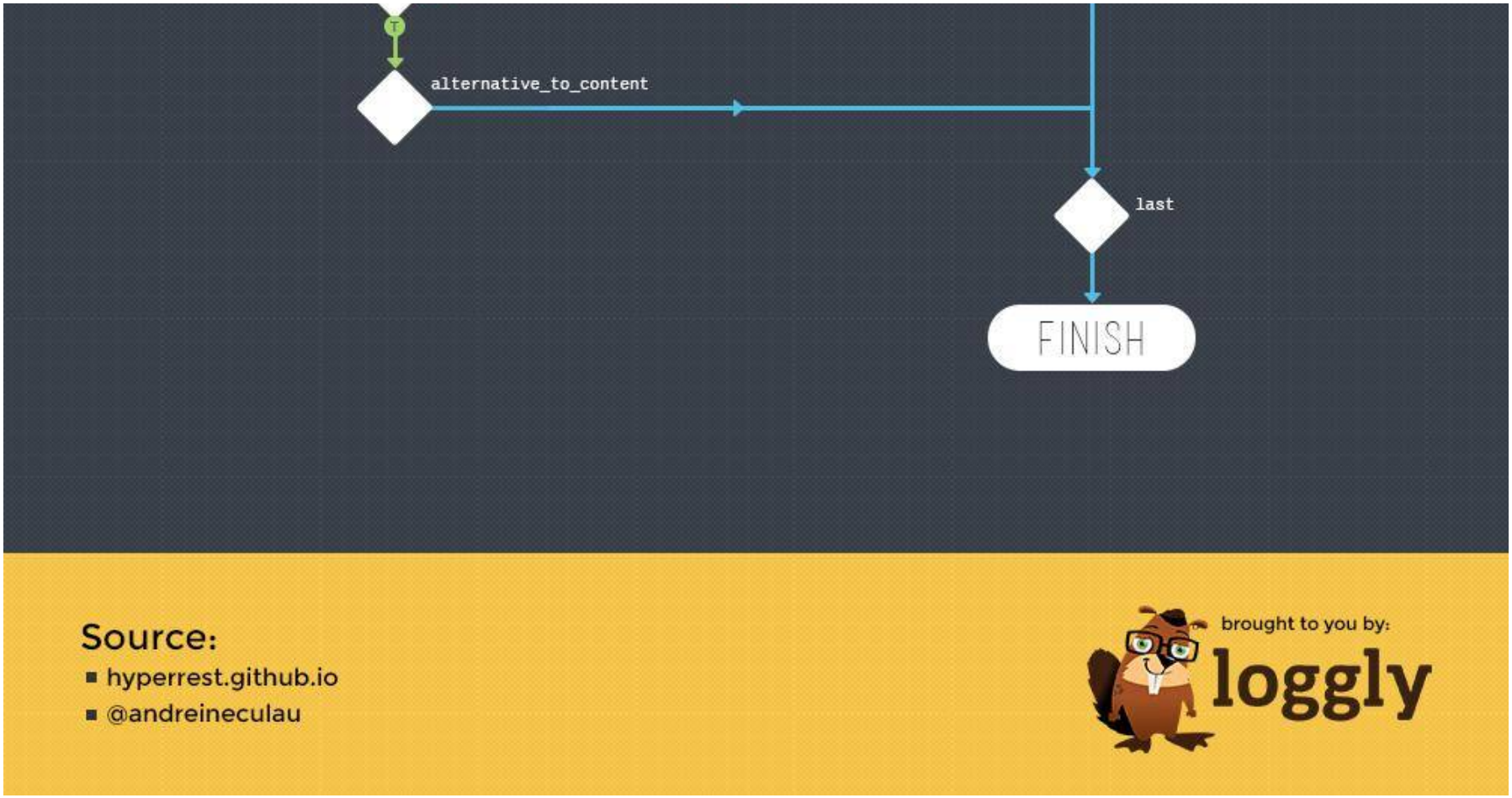
By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.



By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.



By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.



edited Jun 19 at 17:45
Vishrant
5,727 4 31 71

answered Aug 4 '11 at 6:24
JPReddy
42.5k 14 54 91

36 The default IIS 403 message is "This is a generic 403 error and means the authenticated user is not authorized to view the page", which would seem to agree. – Ben Challenor Sep 16 '11 at 13:19

308 @JPReddy Your answer is correct. However, I would expect that 401 to be named "Unauthenticated" and 403 to be named "Unauthorized". It is very confusing that 401, which has to do with Authentication, has the format accompanying text "Unauthorized"....Unless I am not good in English (which is quite a possibility). – p.matsinopoulos Jun 20 '12 at 21:48

By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.

client missing the authorization will get a 401. 403 means "I won't answer to this, whoever you are". RFC states clearly thath "authorization will not help" in the case of 403. – Davide R. Nov 24 '12 at 10:38

80 401 is Authentication error, 403 is Authorization error. Simple as that. – Shahriyar Imanov Mar 25 '13 at 14:09

28 You left out "Well that's my view on it anyway :)" when copying from his blog post and unfortunately his view is wrong. As others have stated 403 means that you can't access the resource regardless of who you are authenticated as. I typically use this status code for resources that are locked down by IP address ranges or files in my webroot that I don't want direct access to (i.e. a script must serve them). – Kyle May 9 '13 at 13:20

374

See RFC2616:

401 Unauthorized:

> If the request already included Authorization credentials, then the 401 response indicates that authorization has been refused for those credentials.

403 Forbidden:

> The server understood the request, but is refusing to fulfill it.

Update

From your use case, it appears that the user is not authenticated. I would return 401.

Edit: RFC2616 is obsolete, see RFC7231 and RFC7235.

edited Aug 11 '17 at 17:36

StampyCode
3,122 1 19 36

answered Jul 21 '10 at 7:28

Oded
426k 78 784 933

20 Thanks, that helped clarify it for me. I'm using both - the 401 for unauthenticated users, the 403 for authenticated users with insufficient permissions. – VirtuosiMedia Jul 21 '10 at 7:51

52 I didn't downvote but I find this answer quite misleading. 403 forbidden is more appropriately used in content that will never be served (like .config files in asp.net). its either that or a 404. imho, it wouldn't be appropriate to return 403 for something that can be accessed but you just didn't have the right

By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.

seem that if you don't want to use HTTP-style authentication, a 401 response code is not appropriate. – Brilliand Mar 20 '12 at 1:42

8 I'll back Billiand here. The statement is "If the request already included Authorization credentials". That means if this is a response from a request which provided the credential (e.g. the response from a RFC2617 Authentication attempt). It is essentially to allow the server to say, "Bad account/password pair, try again". In the posed question, the user is presumably authenticated but not authorized. 401 is never the appropriate response for those circumstances. – ldrut Feb 5 '13 at 17:20

6 Brilliand is right, 401 is only appropriate for HTTP Authentication. – Juampi May 3 '13 at 15:42

285

Something the other answers are missing is that it must be understood that Authentication and Authorization in the context of RFC 2616 refers ONLY to the HTTP Authentication protocol of RFC 2617. Authentication by schemes outside of RFC2617 is not supported in HTTP status codes and are not considered when deciding whether to use 401 or 403.

Brief and Terse

Unauthorized indicates that the client is not RFC2617 authenticated and the server is initiating the authentication process. Forbidden indicates either that the client is RFC2617 authenticated and does not have authorization or that the server does not support RFC2617 for the requested resource.

Meaning if you have your own roll-your-own login process and never use HTTP Authentication, 403 is always the proper response and 401 should never be used.

Detailed and In-Depth

From RFC2616

> 10.4.2 401 Unauthorized
>
> The request requires user authentication. The response MUST include a WWW-Authenticate header field (section 14.47) containing a challenge applicable to the requested resource. The client MAY repeat the request with a suitable Authorization header field (section 14.8).

and

> 10.4.4 403 Forbidden The server understood the request but is refusing to fulfil it. Authorization will not help and the request

By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.

The first thing to keep in mind is that "Authentication" and "Authorization" in the context of this document refer specifically to the HTTP Authentication protocols from RFC 2617. They do not refer to any roll-your-own authentication protocols you may have created using login pages, etc. I will use "login" to refer to authentication and authorization by methods other than RFC2617

So the real difference is not what the problem is or even if there is a solution. The difference is what the server expects the client to do next.

401 indicates that the resource can not be provided, but the server is REQUESTING that the client log in through HTTP Authentication and has sent reply headers to initiate the process. Possibly there are authorizations that will permit access to the resource, possibly there are not, but let's give it a try and see what happens.

403 indicates that the resource can not be provided and there is, for the current user, no way to solve this through RFC2617 and no point in trying. This may be because it is known that no level of authentication is sufficient (for instance because of an IP blacklist), but it may be because the user is already authenticated and does not have authority. The RFC2617 model is one-user, one-credentials so the case where the user may have a second set of credentials that could be authorized may be ignored. It neither suggests nor implies that some sort of login page or other non-RFC2617 authentication protocol may or may not help - that is outside the RFC2616 standards and definition.

Edit: RFC2616 is obsolete, see RFC7231 and RFC7235.

edited Feb 2 at 12:19

nakhodkiin
310 1 3 12

answered Feb 5 '13 at 17:14

ldrut
3,317 1 13 4

7 So what should we do when the user requests a page that requires non-http authentication? Send status code 403? – marcovtwout Mar 25 '14 at 11:00

2 This is the answer that answered my questions on the distinction. – Patrick Apr 2 '14 at 15:48

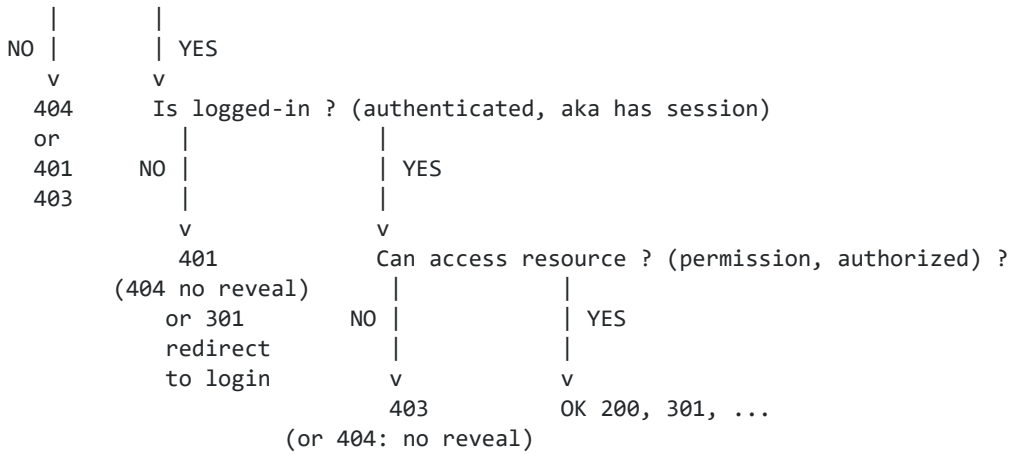
8 This is important: "if you have your own roll-your-own login process and never use HTTP Authentication, 403 is always the proper response and 401 should never be used." – ggg Dec 31 '14 at 6:25

1 @marcovtwout Send a 302 to your login-page, or a 403 containing a body with information how to log in? – Alex Feb 2 '15 at 11:38

4 Doesn't RFC7235 provide for "roll-your-own" or alternate auth challenges? Why can't my app's login flow present its challenge in the form of a `WWW-Authenticate` header? Even if a browser doesn't support it, my React app can... – jchook Oct 11 '16 at 15:53

By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.

116



Checks are usually done in this order:

- 401 if not logged-in or session expired
- 403 if user does not have permission to access resource (file, json, ...)
- 404 if resource does not exist (or not willing to reveal anything)

UNAUTHORIZED: Status code (401) indicating that the request requires **authentication**, usually this means user needs to be logged-in (session). User/agent unknown by the server. Can repeat with other credentials. NOTE: This is confusing as this should have been named 'unauthenticated' instead of 'unauthorized'. This can also happen after login if session expired. Special case: Can be used instead of 404 to avoid revealing presence or non-presence of resource (credits @gingerCodeNinja)

FORBIDDEN: Status code (403) indicating the server understood the request but refused to fulfill it. User/agent known by the server but has **insufficient credentials**. Repeating request will not work, unless credentials changed, which is very unlikely in a short time span. Special case: Can be used instead of 404 to avoid revealing presence or non-presence of resource (credits @gingerCodeNinja)

NOT FOUND: Status code (404) indicating that the requested resource is not available. User/agent known but server will not reveal anything about the resource, does as if it does not exist. Repeating will not work. This is a special use of 404 (github does it for example).

edited Sep 25 at 8:33

answered Feb 23 '15 at 11:00

Christophe Roussy
10.7k 2 59 63

By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.

For example I have logged in and I can access a page but it does not permission enabled for me. Which status code will return? – barteloma Apr 1 '17 at 20:26

@bookmarker Loggin in is called authentication, which is the first step. So if you do not have permission after logging in you will get 403 Forbidden (insufficient credentials means you do not have enough permissions). – Christophe Roussy Apr 3 '17 at 8:07

2 Clear, and simple explanation. Just what I need. – Estevez Feb 8 '18 at 8:44

1 @MattKocaj note that the `no reveal` case can sometimes be detected via subtle timing differences and should not be seen as a security feature, it may just slow down attackers or help a little with privacy. – Christophe Roussy Sep 16 at 8:21

108

According to RFC 2616 (HTTP/1.1) 403 is sent when:

> The server understood the request, but is refusing to fulfill it. Authorization will not help and the request SHOULD NOT be repeated. If the request method was not HEAD and the server wishes to make public why the request has not been fulfilled, it SHOULD describe the reason for the refusal in the entity. If the server does not wish to make this information available to the client, the status code 404 (Not Found) can be used instead

In other words, if the client CAN get access to the resource by authenticating, 401 should be sent.

answered Jul 21 '10 at 7:26

Cumbayah
3,665 1 20 27

4 And if it's not clear if they can access or not? Say that I have 3 user levels - Public, Members, and Premium Members. Assume that the page is for Premium Members only. A public user is basically unauthenticated and *could* be in either Members or Premium Members when they log in. For the Member user level, a 403 would seem appropriate. For Premium Members, the 401. However, what do you serve the Public? – VirtuosiMedia Jul 21 '10 at 7:40

26 imho, this is the most accurate answer. it depends on the application but generally, if an authenticated user doesn't have sufficient rights on a resource, you might want to provide a way to change credentials or send a 401. I think 403 is best suited for content that is never served. In asp.net this would mean web.config files *.resx files etc. because no matter which user logs in, these files will NEVER be served so there is no point in trying again. – Mel Dec 22 '11 at 5:01

6 +1, but an uncertain +1. The logical conclusion is that a 403 should never be returned as either 401 or 404 would be a strictly better response. – CurtainDog Jun 21 '13 at 7:09

11 @Mel I think a file that should not be accessed by the client should be a 404. It's a file that is internal to the system; the outside should not even know it

By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.

1 While this seems to me like it's probably an accurate interpretation of the old RFC 2616, note that RFC 7231 defines the semantics of a 403 differently, and in fact explicitly states that *"The client MAY repeat the request with new or different credentials."* So while this answer was accurate in 2010, it's completely wrong today, because the meaning of the status code has been rewritten beneath our feet. (Annoyingly, the Changes from RFC 2616 appendix doesn't acknowledge the change!) – Mark Amery Apr 30 '17 at 17:00

44

Assuming HTTP authentication (*WWW-Authenticate* and *Authorization* headers) **is in use**, if authenticating as another user would grant access to the requested resource, then 401 Unauthorized should be returned.

403 Forbidden is used when access to the resource is forbidden to everyone or restricted to a given network or allowed only over SSL, whatever as long as it is no related to HTTP authentication.

If HTTP authentication is not in use and the service a cookie-based authentication scheme as is the norm nowadays, then a 403 or a 404 should be returned.

Regarding 401, this is from RFC 7235 (Hypertext Transfer Protocol (HTTP/1.1): Authentication):

> 3.1. 401 Unauthorized
>
> The 401 (Unauthorized) status code indicates that the request has not been applied because it lacks valid authentication credentials for the target resource. The origin server MUST send a WWW-Authenticate header field (Section 4.4) containing at least one challenge applicable to the target resource. **If the request included authentication credentials, then the 401 response indicates that authorization has been refused for those credentials**. The client MAY repeat the request with a new or replaced Authorization header field (Section 4.1). If the 401 response contains the same challenge as the prior response, and the user agent has already attempted authentication at least once, then the user agent SHOULD present the enclosed representation to the user, since it usually contains relevant diagnostic information.

The semantics of 403 (and 404) have changed over time. This is from 1999 (RFC 2616):

> 10.4.4 403 Forbidden
>
> The server understood the request, but is refusing to fulfill it.
> **Authorization will not help** and the request SHOULD NOT be repeated.
> If the request method was not HEAD and the server wishes to make
> public why the request has not been fulfilled, it SHOULD describe the reason for the refusal in the entity. If the server does not wish to make this information available to the client, the status code 404

By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.

In 2014 RFC 7231 (Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content) changed the meaning of 403:

> 6.5.3. 403 Forbidden
>
> The 403 (Forbidden) status code indicates that the server understood the request but refuses to authorize it. A server that wishes to make public why the request has been forbidden can describe that reason in the response payload (if any).
>
> **If authentication credentials were provided in the request, the server considers them insufficient to grant access. The client SHOULD NOT automatically repeat the request with the same credentials. The client MAY repeat the request with new or different credentials. However, a request might be forbidden for reasons unrelated to the credentials.**
>
> An origin server that wishes to "hide" the current existence of a forbidden target resource MAY instead respond with a status code of 404 (Not Found).

Thus, a 403 (or a 404) might now mean about anything. Providing new credentials might help... or it might not.

I believe the reason why this has changed is RFC 2616 assumed HTTP authentication would be used when in practice today's Web apps build custom authentication schemes using for example forms and cookies.

edited Mar 12 at 11:05

answered Feb 27 '13 at 9:44

Erwan Legrand
3,232 22 24

2 This is interesting. Based on RFC 7231 and RFC 7235, I don't see an obvious distinction between 401 and 403 – Brian Feb 27 '15 at 15:20

2 403 means "I know you but you can't see this resource." There's no reason for confusion. – Michael Blackburn Aug 22 '16 at 16:10

"If the request included authentication credentials, then the 401 response indicates that authorization has been refused for those credentials. The client MAY repeat the request with a new or replaced Authorization header field (Section 4.1)." However, then "4.2. The 'Authorization' header field allows a user agent to authenticate itself with an origin server". Looks like in RFC7235 they use the term "authorization" like it was "authentication". In that case, it might seem that an authenticated but not authorized user should not get a 401, but rather 403 – arcuri82 Mar 23 '18 at 8:49

By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.

26

system admin page, or perhaps more commonly, is a record in a system that the user doesn't have access to. Ideally you wouldn't want a malicious user to even know that there's a page / record there, let alone that they don't have access. When I'm building something like this, I'll try to record unauthenticate / unauthorized requests in an internal log, but return a 404.

OWASP has some more information about how an attacker could use this type of information as part of an attack.

answered Dec 25 '14 at 9:09

Patrick White

515 5 12

3 The use of a 404 has been mentioned in previous answers. You're on point re: information leakage and this should be an important consideration for anyone rolling their own authentication/authorization scheme. +1 for mentioning OWASP. – Dave Watts Mar 10 '15 at 11:53

21

This question was asked some time ago, but people's thinking moves on.

Section 6.5.3 in this draft (authored by Fielding and Reschke) gives status code 403 a slightly different meaning to the one documented in RFC 2616.

It reflects what happens in authentication & authorization schemes employed by a number of popular web-servers and frameworks.

I've emphasized the bit I think is most salient.

> **6.5.3. 403 Forbidden**
>
> The 403 (Forbidden) status code indicates that the server understood the request but refuses to authorize it. A server that wishes to make public why the request has been forbidden can describe that reason in the response payload (if any).
>
> If authentication credentials were provided in the request, the server considers them insufficient to grant access. ***The client SHOULD NOT repeat the request with the same credentials. The client MAY repeat the request with new or different credentials.*** However, a request might be forbidden for reasons unrelated to the credentials.
>
> An origin server that wishes to "hide" the current existence of a forbidden target resource MAY instead respond with a status code of 404 (Not Found).

Whatever convention you use, the important thing is to provide uniformity across your site / API.

By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.

Dave Watts
770 6 8

2 The draft was approved and is now RFC 7231. – Vebjorn Ljosa Apr 20 '17 at 12:36

11

TL;DR

- 401: A refusal that has to do with authentication
- 403: A refusal that has NOTHING to do with authentication

Practical Examples

If **apache** *requires authentication* (via `.htaccess`), and you hit `Cancel`, it will respond with a `401 Authorization Required`

If **nginx** finds a file, but has no *access rights* (user/group) to read/access it, it will respond with `403 Forbidden`

RFC (2616 Section 10)

401 Unauthorized (10.4.2)

Meaning 1: **Need to authenticate**

> The request requires user authentication. ...

Meaning 2: **Authentication insufficient**

> ... If the request already included Authorization credentials, then the 401 response indicates that authorization has been refused for those credentials. ...

403 Forbidden (10.4.4)

Meaning: **Unrelated to authentication**

By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.

More details:

- The server understood the request, but is refusing to fulfill it.
- It SHOULD describe the reason for the refusal in the entity
- The status code 404 (Not Found) can be used instead

(If the server wants to keep this information from client)

edited Jul 18 '18 at 7:03

answered Feb 25 '15 at 9:03

Levit

13.1k 7 41 42

9

they are not logged in or do not belong to the proper user group

You have stated two different cases; each case should have a different response:

1. If they are not logged in at all you should return **401 Unauthorized**
2. If they are logged in but don't belong to the proper user group, you should return **403 Forbidden**

Note on the RFC based on comments received to this answer:

If the user is not logged in they are un-authenticated, the HTTP equivalent of which is 401 and is misleadingly called Unauthorized in the RFC. As section 10.4.2 states for **401 Unauthorized**:

> "The request requires user *authentication*."

If you're unauthenticated, 401 is the correct response. However if you're unauthorized, in the semantically correct sense, 403 is the correct response.

By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.

9,590 6 57 80

5 This is not correct. Refer to RFC and to @Cumbayah's answer. – Davide R. Nov 24 '12 at 10:40

7 @DavideR. the RFC uses *authentication* and *authorization* interchangeably. I believe it makes more sense when read with the *authentication* meaning. – Zaid Masud Nov 25 '12 at 1:59

This answer is reversed. Unauthorized is not the same as Un-authenticated. @DavideR is right. Authentication and Authorization are NOT interchangeable – BozoJoe Oct 17 '13 at 20:24

1 2616 should be burned. Several newer RFCs are much clearer that there is a need to differentiate between "I don't know you" and "I know you but you can't access this." There is *no* legitimate reason to acknowledge the existence of a resource that will never be fulfilled (or not fulfilled via http), which is what the 403-truthers are suggesting. – Michael Blackburn Aug 22 '16 at 16:06

4

401: I don't know who you are. This an authentication error. **403**: I know who you are. But you don't have permission to access this resource. This is an authorization error.

answered Aug 6 at 12:37

Akshay Misal

79 4

3

This is simpler in my head than anywhere here, so:

401: You need HTTP basic auth to see this.

403: You can't see this, and HTTP basic auth won't help.

If the user just needs to log in using you site's standard HTML login form, 401 would not be appropriate because it is specific to HTTP basic auth.

I don't recommend using 403 to deny access to things like `/includes`, because as far as the web is concerned, those resources don't exist at all and should therefore 404.

This leaves 403 as "you need to be logged in".

By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.

edited Oct 14 '17 at 18:46

answered Sep 23 '17 at 12:33

Vladimir Kornea
2,256 2 29 35

3

I think it is important to consider that, to a browser, 401 initiates an authentication dialog for the user to enter new credentials, while 403 does not. Browsers think that, if a 401 is returned, then the user should re-authenticate. So 401 stands for invalid authentication while 403 stands for a lack of permission.

Here are some cases under that logic where an error would be returned from authentication or authorization, with important phrases bolded.

- A resource requires authentication but **no credentials** were **specified**.

401: The client should specify credentials.

- The specified credentials are in an **invalid format**.

400: That's neither 401 nor 403, as syntax errors should always return 400.

- The specified credentials reference a **user** which **does not exist**.

401: The client should specify valid credentials.

- The specified **credentials** are **invalid** but specify a valid user (or don't specify a user if a specified user is not required).

401: Again, the client should specify valid credentials.

- The specified **credentials** have **expired**.

401: This is practically the same as having invalid credentials in general, so the client should specify valid credentials.

- The specified credentials are completely valid but do not **suffice** the particular **resource**, though it is possible that credentials with more permission could.

403: Specifying valid credentials would not grant access to the resource, as the current credentials are already valid but only do not have permission.

By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.

- The specified credentials are completely valid but the particular **client** is **blocked** from using them.

403: If the client is blocked, specifying new credentials will not do anything.

edited Dec 14 '18 at 21:53

answered Jun 2 '18 at 23:34

Grant Gryczan

268 3 17

0

Given the latest RFC's on the matter (7231 and 7235) the use-case seems quite clear (italics added):

- 401 is for unauthenticated ("lacks valid authentication"); i.e. 'I don't know who you are, or I don't trust you are who you say you are.'

> **401 Unauthorized**
>
> The 401 (Unauthorized) status code indicates that the request has not been applied because it *lacks valid authentication* credentials for the target resource. The server generating a 401 response MUST send a WWW-Authenticate header field (Section 4.1) containing at least one challenge applicable to the target resource.
>
> If the request included authentication credentials, then the 401 response indicates that authorization has been refused for those credentials. The user agent MAY repeat the request with a new or replaced Authorization header field (Section 4.2). If the 401 response contains the same challenge as the prior response, and the user agent has already attempted authentication at least once, then the user agent SHOULD present the enclosed representation to the user, since it usually contains relevant diagnostic information.

- 403 is for unauthorized ("refuses to authorize"); i.e. 'I know who you are, but you don't have permission to access this resource.'

> **403 Forbidden**
>
> The 403 (Forbidden) status code indicates that the server understood the request but *refuses to authorize* it. A server that wishes to make public why the request has been forbidden can describe that reason in the response payload (if any).
>
> If authentication credentials were provided in the request, the server considers them insufficient to grant access. The client SHOULD NOT automatically repeat the request with the same credentials. The client MAY repeat the request with new or different credentials. However, a request might be forbidden for reasons unrelated to the credentials.

By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.

edited Jun 5 '18 at 17:21

answered Jun 5 '18 at 15:26

cjbarth

2,047 3 27 47

2 -1; these passages have already been quoted in other answers here, and yours adds nothing new. I'd argue that it's patently *not* clear what the distinction is; you summarise the two codes as "lacks valid authentication" and "refuses to authorise" but I cannot conceive of any situation in which one of those short descriptions would apply where the other could not be interpreted to apply as well. – Mark Amery Jun 5 '18 at 15:59

There are many answers here that cover many RFC's and are edited and updated muddying the waters. I included a link to explain what `authenticated` is and what `authorized` is and left off all outdated RFC's so that the application is clear. – cjbarth Jun 5 '18 at 17:17

Your edit clarifies your interpretation of the two codes, which seems to match many other people's interpretation. However, I personally believe that interpretation makes little sense. The use of the phrase *"**If** authentication credentials were provided"* in the 403 description implies that a 403 can be appropriate even if no credentials were provided - i.e. the "unauthenticated" case. Meanwhile, to me the most natural interpretation of the phrase *"for the target resource"* being included in the 401 description is that a 401 can be used for a user who is authenticated but not authorized. – Mark Amery Jun 6 '18 at 11:36

-4

In the case of 401 vs 403, this has been answered many times. This is essentially a 'HTTP request environment' debate, not an 'application' debate.

There seems to be a question on the roll-your-own-login issue (application).

In this case, simply not being logged in is not sufficient to send a 401 or a 403, unless you use HTTP Auth vs a login page (not tied to setting HTTP Auth). It sounds like you may be looking for a "201 Created", with a roll-your-own-login screen present (instead of the requested resource) for the application-level access to a file. This says:

"I heard you, it's here, but try this instead (you are not allowed to see it)"

answered Dec 12 '14 at 19:01

Shawn

3

What exactly is being created? – Grant Gryczan Jun 9 '18 at 1:25

By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.

Thank you for your interest in this question. Because it has attracted low-quality or spam answers that had to be removed, posting an answer now requires 10 reputation on this site (the association bonus does not count).

Would you like to answer one of these unanswered questions instead?

By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.