Item 249
git_comments:

git_commits:

1. **summary:** Update to mochiweb 2.19.0
   message: Update to mochiweb 2.19.0 It has a fix to revert user socket buffer size to 8192 and also allow setting this buffer values directly (not necessarily via {recbuf, ...}). Fixes #1810 Warning: 2.19.0 blacklists a series of OTP releases: 21.2, 21.2.1, 21.2.2 This is done via a runtime check of the ssl application version. The blacklist seems valid as there is a bug which prevents data from being delivered on TSL sockets. That could affect either CouchDB server side (chttpd) or replication client side (ibrowse).

github_issues:

1. **title:** Requests return 400 Bad Request when URL length exceeds 1460 characters
   body: When using `chttpd.server_option = [{recbuf, undefined}]`, requests with URL length exceeding 1459 characters fail with `400 Bad Request` with no error in the logs. (The relevant URL length excludes protocol and userinfo). Example of failing request: ``` curl -v http://127.0.0.1:5984/_users/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ``` URL length is 1467 (1460 excluding protocol and userinfo) ## Expected Behavior Should return: ``` Host: 127.0.0.1:5984 User-Agent: curl/7.58.0 Accept: */* HTTP/1.1 404 Object Not Found Cache-Control: must-revalidate Content-Length: 41 Content-Type: application/json Date: Thu, 13 Dec 2018 13:18:50 GMT Server: CouchDB/2.3.0 (Erlang OTP/20) X-Couch-Request-ID: 0b4ad7148f X-CouchDB-Body-Time: 0 {"error":"not_found","reason":"missing"} ``` ## Current Behavior Actually returns: ``` HTTP/1.1 Host: 127.0.0.1:5984 User-Agent: curl/7.58.0 Accept: */* HTTP/1.1 400 Bad Request Connection: close Content-Length: 0 Date: Thu, 13 Dec 2018 13:15:09 GMT Server: MochiWeb/1.0 (Any of you quaids got a smint?) ``` Example of successful request: ``` curl -v http://127.0.0.1:5984/_users/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ``` URL length is 1466 (1459 excluding protocol and userinfo). ## Possible Cause It appears that, by not setting a `recbuf` value, mochiweb_socket_server sockets end up with a `buffer` size of 1460 (the user-level buffer - http://erlang.org/doc/man/inet.html), which causes this limitation (`recbuf` and `sndbuf` are indeed defaulting to much larger values). This default value of 1460 seems to be coming from [Erl inet_drv.c] (https://github.com/erlang/otp/blob/56f93ad10f89e6b8d3372e45127ec9fdc3fca35b/erts/emulator/drivers/common/inet_drv.c#L923) and initialized [here] (https://github.com/erlang/otp/blob/56f93ad10f89e6b8d3372e45127ec9fdc3fca35b/erts/emulator/drivers/common/inet_drv.c#L9057). ## Steps to Reproduce Install CouchDB 2.3 (or build from `master`) and run with default configuration (specifically `chttpd.server_option = [{recbuf, undefined}]`). Execute a request with a URL length (excluding protocol and authority) exceeding 1460 characters (method or endpoint appear to be irrelevant). ## Context We are using PouchDB to interact with our CouchDB database and ran into this error with view queries. PouchDB does issue a POST request instead of a GET, but only when the query string exceeds 2000 characters (see [here] (https://github.com/pouchdb/pouchdb/blob/f87c57a3b3b791dbb62bd881378755ca0a195d13/packages/node_modules/pouchdb-abstract-mapreduce/src/index.js#L262)). ## Your Environment * Version used: 2.3.0, master * Using: Chrome, cURL, wget * Operating system: Ubuntu 18, Ubuntu 16 (desktop) Even if it is decided that this is desired behavior, I believe this change should at least be documented.
2. **title:** Requests return 400 Bad Request when URL length exceeds 1460 characters
   body: When using `chttpd.server_option = [{recbuf, undefined}]`, requests with URL length exceeding 1459 characters fail with `400 Bad Request` with no error in the logs. (The relevant URL length excludes protocol and userinfo). Example of failing request: ``` curl -v http://127.0.0.1:5984/_users/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ``` URL length is 1467 (1460 excluding protocol and userinfo) ## Expected Behavior Should return: ``` Host: 127.0.0.1:5984 User-Agent: curl/7.58.0 Accept: */* HTTP/1.1 404 Object Not Found Cache-Control: must-revalidate Content-Length: 41 Content-Type: application/json Date: Thu, 13 Dec 2018 13:18:50 GMT Server: CouchDB/2.3.0 (Erlang OTP/20) X-Couch-Request-ID: 0b4ad7148f X-CouchDB-Body-Time: 0 {"error":"not_found","reason":"missing"} ``` ## Current Behavior Actually returns: ``` HTTP/1.1 Host: 127.0.0.1:5984 User-Agent: curl/7.58.0 Accept: */* HTTP/1.1 400 Bad Request Connection: close Content-Length: 0 Date: Thu, 13 Dec 2018 13:15:09 GMT Server: MochiWeb/1.0 (Any of you quaids got a smint?) ``` Example of successful request: ``` curl -v http://127.0.0.1:5984/_users/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ``` URL length is 1466 (1459 excluding protocol and userinfo). ## Possible Cause It appears that, by not setting a `recbuf` value, mochiweb_socket_server sockets end up with a `buffer` size of 1460 (the user-level buffer - http://erlang.org/doc/man/inet.html), which causes this limitation (`recbuf` and `sndbuf` are indeed defaulting to much larger values). This default value of 1460 seems to be coming from [Erl inet_drv.c] (https://github.com/erlang/otp/blob/56f93ad10f89e6b8d3372e45127ec9fdc3fca35b/erts/emulator/drivers/common/inet_drv.c#L923) and initialized [here] (https://github.com/erlang/otp/blob/56f93ad10f89e6b8d3372e45127ec9fdc3fca35b/erts/emulator/drivers/common/inet_drv.c#L9057). ## Steps to Reproduce Install CouchDB 2.3 (or build from `master`) and run with default configuration (specifically `chttpd.server_option = [{recbuf, undefined}]`). Execute a request with a URL length (excluding protocol and authority) exceeding 1460 characters (method or endpoint appear to be irrelevant). ## Context We are using PouchDB to interact with our CouchDB database and ran into this error with view queries. PouchDB does issue a POST request instead of a GET, but only when the query string exceeds 2000 characters (see [here] (https://github.com/pouchdb/pouchdb/blob/f87c57a3b3b791dbb62bd881378755ca0a195d13/packages/node_modules/pouchdb-abstract-mapreduce/src/index.js#L262)). ## Your Environment * Version used: 2.3.0, master * Using: Chrome, cURL, wget * Operating system: Ubuntu 18, Ubuntu 16 (desktop) Even if it is decided that this is desired behavior, I believe this change should at least be documented.
   label: code-design
3. **title:** Requests return 400 Bad Request when URL length exceeds 1460 characters
   body: When using `chttpd.server_option = [{recbuf, undefined}]`, requests with URL length exceeding 1459 characters fail with `400 Bad Request` with no error in the logs. (The relevant URL length excludes protocol and userinfo). Example of failing request: ``` curl -v http://127.0.0.1:5984/_users/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ``` URL length is 1467 (1460 excluding protocol and userinfo) ## Expected Behavior Should return: ``` Host: 127.0.0.1:5984 User-Agent: curl/7.58.0 Accept: */* HTTP/1.1 404 Object Not Found Cache-Control: must-revalidate Content-Length: 41 Content-Type: application/json Date: Thu, 13 Dec 2018 13:18:50 GMT Server: CouchDB/2.3.0 (Erlang OTP/20) X-Couch-Request-ID: 0b4ad7148f X-CouchDB-Body-Time: 0 {"error":"not_found","reason":"missing"} ``` ## Current Behavior Actually returns: ``` HTTP/1.1 Host: 127.0.0.1:5984 User-Agent: curl/7.58.0 Accept: */* HTTP/1.1 400 Bad Request Connection: close Content-Length: 0 Date: Thu, 13 Dec 2018 13:15:09 GMT Server: MochiWeb/1.0 (Any of you quaids got a smint?) ``` Example of successful request: ``` curl -v http://127.0.0.1:5984/_users/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ``` URL length is 1466 (1459 excluding protocol and userinfo). ## Possible Cause It appears that, by not setting a `recbuf` value, mochiweb_socket_server sockets end up with a `buffer` size of 1460 (the user-level buffer - http://erlang.org/doc/man/inet.html), which causes this limitation (`recbuf` and `sndbuf` are indeed defaulting to much larger values). This default value of 1460 seems to be coming from [Erl inet_drv.c] (https://github.com/erlang/otp/blob/56f93ad10f89e6b8d3372e45127ec9fdc3fca35b/erts/emulator/drivers/common/inet_drv.c#L923) and initialized [here] (https://github.com/erlang/otp/blob/56f93ad10f89e6b8d3372e45127ec9fdc3fca35b/erts/emulator/drivers/common/inet_drv.c#L9057). ## Steps to Reproduce Install CouchDB 2.3 (or build from `master`) and run with default configuration (specifically `chttpd.server_option = [{recbuf, undefined}]`). Execute a request with a URL length (excluding protocol and authority) exceeding 1460 characters (method or endpoint appear to be irrelevant). ## Context We are using PouchDB to interact with our CouchDB database and ran into this error with view queries. PouchDB does issue a POST request instead of a GET, but only when the query string exceeds 2000 characters (see [here] (https://github.com/pouchdb/pouchdb/blob/f87c57a3b3b791dbb62bd881378755ca0a195d13/packages/node_modules/pouchdb-abstract-mapreduce/src/index.js#L262)). ## Your Environment * Version used: 2.3.0, master * Using: Chrome, cURL, wget * Operating system: Ubuntu 18, Ubuntu 16 (desktop) Even if it is decided that this is desired behavior, I believe this change should at least be documented.
4. **title:** Requests return 400 Bad Request when URL length exceeds 1460 characters
   body: When using `chttpd.server_option = [{recbuf, undefined}]`, requests with URL length exceeding 1459 characters fail with `400 Bad Request` with no error in the logs. (The relevant URL length excludes protocol and userinfo). Example of failing request: ``` curl -v http://127.0.0.1:5984/_users/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ``` URL length is 1467 (1460 excluding protocol and userinfo) ## Expected Behavior Should return: ``` Host: 127.0.0.1:5984 User-Agent: curl/7.58.0 Accept: */* HTTP/1.1 404 Object Not Found Cache-Control: must-revalidate Content-Length: 41 Content-Type: application/json Date: Thu, 13 Dec 2018 13:18:50 GMT Server: CouchDB/2.3.0 (Erlang OTP/20) X-Couch-Request-ID: 0b4ad7148f X-CouchDB-Body-Time: 0 {"error":"not_found","reason":"missing"} ``` ## Current Behavior Actually returns: ``` HTTP/1.1 Host: 127.0.0.1:5984 User-Agent: curl/7.58.0 Accept: */* HTTP/1.1 400 Bad Request Connection: close Content-Length:

0 Date: Thu, 13 Dec 2018 13:15:09 GMT Server: MochiWeb/1.0 (Any of you quaids got a smint?) ``` Example of successful request: ``` curl -v http://127.0.0.1:5984/_users/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ``` URL length is 1466 (1459 excluding protocol and userinfo). ## Possible Cause It appears that, by not setting a `recbuf` value, mochiweb_socket_server sockets end up with a `buffer` size of 1460 (the user-level buffer - http://erlang.org/doc/man/inet.html), which causes this limitation (`recbuf` and `sndbuf` are indeed defaulting to much larger values). This default value of 1460 seems to be coming from [Erl inet_drv.c] (https://github.com/erlang/otp/blob/56f93ad10f89e6b8d3372e45127ec9fdc3fca35b/erts/emulator/drivers/common/inet_drv.c#L923) and initialized [here] (https://github.com/erlang/otp/blob/56f93ad10f89e6b8d3372e45127ec9fdc3fca35b/erts/emulator/drivers/common/inet_drv.c#L9057). ## Steps to Reproduce Install CouchDB 2.3 (or build from `master`) and run with default configuration (specifically `chttpd.server_option = [{recbuf, undefined}]`). Execute a request with a URL length (excluding protocol and authority) exceeding 1460 characters (method or endpoint appear to be irrelevant). ## Context We are using PouchDB to interact with our CouchDB database and ran into this error with view queries. PouchDB does issue a POST request instead of a GET, but only when the query string exceeds 2000 characters (see [here] (https://github.com/pouchdb/pouchdb/blob/f87c57a3b3b791dbb62bd881378755ca0a195d13/packages/node_modules/pouchdb-abstract-mapreduce/src/index.js#L262)). ## Your Environment * Version used: 2.3.0, master * Using: Chrome, cURL, wget * Operating system: Ubuntu 18, Ubuntu 16 (desktop) Even if it is decided that this is desired behavior, I believe this change should at least be documented.

5. **title:** Requests return 400 Bad Request when URL length exceeds 1460 characters
   body: When using `chttpd.server_option = [{recbuf, undefined}]`, requests with URL length exceeding 1459 characters fail with `400 Bad Request` with no error in the logs. (The relevant URL length excludes protocol and userinfo). Example of failing request: ``` curl -v http://127.0.0.1:5984/_users/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ``` URL length is 1467 (1460 excluding protocol and userinfo) ## Expected Behavior Should return: ``` Host: 127.0.0.1:5984 User-Agent: curl/7.58.0 Accept: */* HTTP/1.1 404 Object Not Found Cache-Control: must-revalidate Content-Length: 41 Content-Type: application/json Date: Thu, 13 Dec 2018 13:18:50 GMT Server: CouchDB/2.3.0 (Erlang OTP/20) X-Couch-Request-ID: 0b4ad7148f X-CouchDB-Body-Time: 0 {"error":"not_found","reason":"missing"} ``` ## Current Behavior Actually returns: ``` HTTP/1.1 Host: 127.0.0.1:5984 User-Agent: curl/7.58.0 Accept: */* HTTP/1.1 400 Bad Request Connection: close Content-Length: 0 Date: Thu, 13 Dec 2018 13:15:09 GMT Server: MochiWeb/1.0 (Any of you quaids got a smint?) ``` Example of successful request: ``` curl -v http://127.0.0.1:5984/_users/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ``` URL length is 1466 (1459 excluding protocol and userinfo). ## Possible Cause It appears that, by not setting a `recbuf` value, mochiweb_socket_server sockets end up with a `buffer` size of 1460 (the user-level buffer - http://erlang.org/doc/man/inet.html), which causes this limitation (`recbuf` and `sndbuf` are indeed defaulting to much larger values). This default value of 1460 seems to be coming from [Erl inet_drv.c] (https://github.com/erlang/otp/blob/56f93ad10f89e6b8d3372e45127ec9fdc3fca35b/erts/emulator/drivers/common/inet_drv.c#L923) and initialized [here] (https://github.com/erlang/otp/blob/56f93ad10f89e6b8d3372e45127ec9fdc3fca35b/erts/emulator/drivers/common/inet_drv.c#L9057). ## Steps to Reproduce Install CouchDB 2.3 (or build from `master`) and run with default configuration (specifically `chttpd.server_option = [{recbuf, undefined}]`). Execute a request with a URL length (excluding protocol and authority) exceeding 1460 characters (method or endpoint appear to be irrelevant). ## Context We are using PouchDB to interact with our CouchDB database and ran into this error with view queries. PouchDB does issue a POST request instead of a GET, but only when the query string exceeds 2000 characters (see [here] (https://github.com/pouchdb/pouchdb/blob/f87c57a3b3b791dbb62bd881378755ca0a195d13/packages/node_modules/pouchdb-abstract-mapreduce/src/index.js#L262)). ## Your Environment * Version used: 2.3.0, master * Using: Chrome, cURL, wget * Operating system: Ubuntu 18, Ubuntu 16 (desktop) Even if it is decided that this is desired behavior, I believe this change should at least be documented.
6. **title:** Requests return 400 Bad Request when URL length exceeds 1460 characters
   body: When using `chttpd.server_option = [{recbuf, undefined}]`, requests with URL length exceeding 1459 characters fail with `400 Bad Request` with no error in the logs. (The relevant URL length excludes protocol and userinfo). Example of failing request: ``` curl -v http://127.0.0.1:5984/_users/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ``` URL length is 1467 (1460 excluding protocol and userinfo) ## Expected Behavior Should return: ``` Host: 127.0.0.1:5984 User-Agent: curl/7.58.0 Accept: */* HTTP/1.1 404 Object Not Found Cache-Control: must-revalidate Content-Length: 41 Content-Type: application/json Date: Thu, 13 Dec 2018 13:18:50 GMT Server: CouchDB/2.3.0 (Erlang OTP/20) X-Couch-Request-ID: 0b4ad7148f X-CouchDB-Body-Time: 0 {"error":"not_found","reason":"missing"} ``` ## Current Behavior Actually returns: ``` HTTP/1.1 Host: 127.0.0.1:5984 User-Agent: curl/7.58.0 Accept: */* HTTP/1.1 400 Bad Request Connection: close Content-Length: 0 Date: Thu, 13 Dec 2018 13:15:09 GMT Server: MochiWeb/1.0 (Any of you quaids got a smint?) ``` Example of successful request: ``` curl -v http://127.0.0.1:5984/_users/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ``` URL length is 1466 (1459 excluding protocol and userinfo). ## Possible Cause It appears that, by not setting a `recbuf` value, mochiweb_socket_server sockets end up with a `buffer` size of 1460 (the user-level buffer - http://erlang.org/doc/man/inet.html), which causes this limitation (`recbuf` and `sndbuf` are indeed defaulting to much larger values). This default value of 1460 seems to be coming from [Erl inet_drv.c] (https://github.com/erlang/otp/blob/56f93ad10f89e6b8d3372e45127ec9fdc3fca35b/erts/emulator/drivers/common/inet_drv.c#L923) and initialized [here] (https://github.com/erlang/otp/blob/56f93ad10f89e6b8d3372e45127ec9fdc3fca35b/erts/emulator/drivers/common/inet_drv.c#L9057). ## Steps to Reproduce Install CouchDB 2.3 (or build from `master`) and run with default configuration (specifically `chttpd.server_option = [{recbuf, undefined}]`). Execute a request with a URL length (excluding protocol and authority) exceeding 1460 characters (method or endpoint appear to be irrelevant). ## Context We are using PouchDB to interact with our CouchDB database and ran into this error with view queries. PouchDB does issue a POST request instead of a GET, but only when the query string exceeds 2000 characters (see [here] (https://github.com/pouchdb/pouchdb/blob/f87c57a3b3b791dbb62bd881378755ca0a195d13/packages/node_modules/pouchdb-abstract-mapreduce/src/index.js#L262)). ## Your Environment * Version used: 2.3.0, master * Using: Chrome, cURL, wget * Operating system: Ubuntu 18, Ubuntu 16 (desktop) Even if it is decided that this is desired behavior, I believe this change should at least be documented.
   label: code-design
7. **title:** Requests return 400 Bad Request when URL length exceeds 1460 characters
   body: When using `chttpd.server_option = [{recbuf, undefined}]`, requests with URL length exceeding 1459 characters fail with `400 Bad Request` with no error in the logs. (The relevant URL length excludes protocol and userinfo). Example of failing request: ``` curl -v http://127.0.0.1:5984/_users/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ``` URL length is 1467 (1460 excluding protocol and userinfo) ## Expected Behavior Should return: ``` Host: 127.0.0.1:5984 User-Agent: curl/7.58.0 Accept: */* HTTP/1.1 404 Object Not Found Cache-Control: must-revalidate Content-Length: 41 Content-Type: application/json Date: Thu, 13 Dec 2018 13:18:50 GMT Server: CouchDB/2.3.0 (Erlang OTP/20) X-Couch-Request-ID: 0b4ad7148f X-CouchDB-Body-Time: 0 {"error":"not_found","reason":"missing"} ``` ## Current Behavior Actually returns: ``` HTTP/1.1 Host: 127.0.0.1:5984 User-Agent: curl/7.58.0 Accept: */* HTTP/1.1 400 Bad Request Connection: close Content-Length: 0 Date: Thu, 13 Dec 2018 13:15:09 GMT Server: MochiWeb/1.0 (Any of you quaids got a smint?) ``` Example of successful request: ``` curl -v http://127.0.0.1:5984/_users/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ``` URL length is 1466 (1459 excluding protocol and userinfo). ## Possible Cause It appears that, by not setting a `recbuf` value, mochiweb_socket_server sockets end up with a `buffer` size of 1460 (the user-level buffer - http://erlang.org/doc/man/inet.html), which causes this limitation (`recbuf` and `sndbuf` are indeed defaulting to much larger values). This default value of 1460 seems to be coming from [Erl inet_drv.c] (https://github.com/erlang/otp/blob/56f93ad10f89e6b8d3372e45127ec9fdc3fca35b/erts/emulator/drivers/common/inet_drv.c#L923) and initialized [here] (https://github.com/erlang/otp/blob/56f93ad10f89e6b8d3372e45127ec9fdc3fca35b/erts/emulator/drivers/common/inet_drv.c#L9057). ## Steps to Reproduce Install CouchDB 2.3 (or build from `master`) and run with default configuration (specifically `chttpd.server_option = [{recbuf, undefined}]`). Execute a request with a URL length (excluding protocol and authority) exceeding 1460 characters (method or endpoint appear to be irrelevant). ## Context We are using PouchDB to interact with our CouchDB database and ran into this error with view queries. PouchDB does issue a POST request instead of a GET, but only when the query string exceeds 2000 characters (see [here] (https://github.com/pouchdb/pouchdb/blob/f87c57a3b3b791dbb62bd881378755ca0a195d13/packages/node_modules/pouchdb-abstract-mapreduce/src/index.js#L262)). ## Your Environment * Version used: 2.3.0, master * Using: Chrome, cURL, wget * Operating system: Ubuntu 18, Ubuntu 16 (desktop) Even if it is decided that this is desired behavior, I believe this change should at least be documented.
   label: code-design
8. **title:** Requests return 400 Bad Request when URL length exceeds 1460 characters
   body: When using `chttpd.server_option = [{recbuf, undefined}]`, requests with URL length exceeding 1459 characters fail with `400 Bad Request` with no error in the logs. (The relevant URL length excludes protocol and userinfo). Example of failing request: ``` curl -v http://127.0.0.1:5984/_users/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ``` URL length is 1467 (1460 excluding protocol and userinfo) ## Expected Behavior Should return: ``` Host: 127.0.0.1:5984 User-Agent: curl/7.58.0 Accept: */* HTTP/1.1 404 Object Not Found Cache-Control: must-revalidate Content-Length: 41 Content-Type: application/json Date: Thu, 13 Dec 2018 13:18:50 GMT

Server: CouchDB/2.3.0 (Erlang OTP/20) X-Couch-Request-ID: 0b4ad7148f X-CouchDB-Body-Time: 0 {"error":"not_found","reason":"missing"} ``` ## Current Behavior Actually returns: ``` HTTP/1.1 Host: 127.0.0.1:5984 User-Agent: curl/7.58.0 Accept: */* HTTP/1.1 400 Bad Request Connection: close Content-Length: 0 Date: Thu, 13 Dec 2018 13:15:09 GMT Server: MochiWeb/1.0 (Any of you quaids got a smint?) ``` Example of successful request: ``` curl -v http://127.0.0.1:5984/_users/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ``` URL length is 1466 (1459 excluding protocol and userinfo). ## Possible Cause It appears that, by not setting a `recbuf` value, mochiweb_socket_server sockets end up with a `buffer` size of 1460 (the user-level buffer - http://erlang.org/doc/man/inet.html), which causes this limitation (`recbuf` and `sndbuf` are indeed defaulting to much larger values). This default value of 1460 seems to be coming from [Erl inet_drv.c] (https://github.com/erlang/otp/blob/56f93ad10f89e6b8d3372e45127ec9fdc3fca35b/erts/emulator/drivers/common/inet_drv.c#L923) and initialized [here] (https://github.com/erlang/otp/blob/56f93ad10f89e6b8d3372e45127ec9fdc3fca35b/erts/emulator/drivers/common/inet_drv.c#L9057). ## Steps to Reproduce Install CouchDB 2.3 (or build from `master`) and run with default configuration (specifically `chttpd.server_option = [{recbuf, undefined}]`). Execute a request with a URL length (excluding protocol and authority) exceeding 1460 characters (method or endpoint appear to be irrelevant). ## Context We are using PouchDB to interact with our CouchDB database and ran into this error with view queries. PouchDB does issue a POST request instead of a GET, but only when the query string exceeds 2000 characters (see [here] (https://github.com/pouchdb/pouchdb/blob/f87c57a3b3b791dbb62bd881378755ca0a195d13/packages/node_modules/pouchdb-abstract-mapreduce/src/index.js#L262)). ## Your Environment * Version used: 2.3.0, master * Using: Chrome, cURL, wget * Operating system: Ubuntu 18, Ubuntu 16 (desktop) Even if it is decided that this is desired behavior, I believe this change should at least be documented.

9. **title:** Requests return 400 Bad Request when URL length exceeds 1460 characters
   body: When using `chttpd.server_option = [{recbuf, undefined}]`, requests with URL length exceeding 1459 characters fail with `400 Bad Request` with no error in the logs. (The relevant URL length excludes protocol and userinfo). Example of failing request: ``` curl -v http://127.0.0.1:5984/_users/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ``` URL length is 1467 (1460 excluding protocol and userinfo) ## Expected Behavior Should return: ``` Host: 127.0.0.1:5984 User-Agent: curl/7.58.0 Accept: */* HTTP/1.1 404 Object Not Found Cache-Control: must-revalidate Content-Length: 41 Content-Type: application/json Date: Thu, 13 Dec 2018 13:18:50 GMT Server: CouchDB/2.3.0 (Erlang OTP/20) X-Couch-Request-ID: 0b4ad7148f X-CouchDB-Body-Time: 0 {"error":"not_found","reason":"missing"} ``` ## Current Behavior Actually returns: ``` HTTP/1.1 Host: 127.0.0.1:5984 User-Agent: curl/7.58.0 Accept: */* HTTP/1.1 400 Bad Request Connection: close Content-Length: 0 Date: Thu, 13 Dec 2018 13:15:09 GMT Server: MochiWeb/1.0 (Any of you quaids got a smint?) ``` Example of successful request: ``` curl -v http://127.0.0.1:5984/_users/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ``` URL length is 1466 (1459 excluding protocol and userinfo). ## Possible Cause It appears that, by not setting a `recbuf` value, mochiweb_socket_server sockets end up with a `buffer` size of 1460 (the user-level buffer - http://erlang.org/doc/man/inet.html), which causes this limitation (`recbuf` and `sndbuf` are indeed defaulting to much larger values). This default value of 1460 seems to be coming from [Erl inet_drv.c] (https://github.com/erlang/otp/blob/56f93ad10f89e6b8d3372e45127ec9fdc3fca35b/erts/emulator/drivers/common/inet_drv.c#L923) and initialized [here] (https://github.com/erlang/otp/blob/56f93ad10f89e6b8d3372e45127ec9fdc3fca35b/erts/emulator/drivers/common/inet_drv.c#L9057). ## Steps to Reproduce Install CouchDB 2.3 (or build from `master`) and run with default configuration (specifically `chttpd.server_option = [{recbuf, undefined}]`). Execute a request with a URL length (excluding protocol and authority) exceeding 1460 characters (method or endpoint appear to be irrelevant). ## Context We are using PouchDB to interact with our CouchDB database and ran into this error with view queries. PouchDB does issue a POST request instead of a GET, but only when the query string exceeds 2000 characters (see [here] (https://github.com/pouchdb/pouchdb/blob/f87c57a3b3b791dbb62bd881378755ca0a195d13/packages/node_modules/pouchdb-abstract-mapreduce/src/index.js#L262)). ## Your Environment * Version used: 2.3.0, master * Using: Chrome, cURL, wget * Operating system: Ubuntu 18, Ubuntu 16 (desktop) Even if it is decided that this is desired behavior, I believe this change should at least be documented.
10. **title:** Requests return 400 Bad Request when URL length exceeds 1460 characters
    body: When using `chttpd.server_option = [{recbuf, undefined}]`, requests with URL length exceeding 1459 characters fail with `400 Bad Request` with no error in the logs. (The relevant URL length excludes protocol and userinfo). Example of failing request: ``` curl -v http://127.0.0.1:5984/_users/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ``` URL length is 1467 (1460 excluding protocol and userinfo) ## Expected Behavior Should return: ``` Host: 127.0.0.1:5984 User-Agent: curl/7.58.0 Accept: */* HTTP/1.1 404 Object Not Found Cache-Control: must-revalidate Content-Length: 41 Content-Type: application/json Date: Thu, 13 Dec 2018 13:18:50 GMT Server: CouchDB/2.3.0 (Erlang OTP/20) X-Couch-Request-ID: 0b4ad7148f X-CouchDB-Body-Time: 0 {"error":"not_found","reason":"missing"} ``` ## Current Behavior Actually returns: ``` HTTP/1.1 Host: 127.0.0.1:5984 User-Agent: curl/7.58.0 Accept: */* HTTP/1.1 400 Bad Request Connection: close Content-Length: 0 Date: Thu, 13 Dec 2018 13:15:09 GMT Server: MochiWeb/1.0 (Any of you quaids got a smint?) ``` Example of successful request: ``` curl -v http://127.0.0.1:5984/_users/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ``` URL length is 1466 (1459 excluding protocol and userinfo). ## Possible Cause It appears that, by not setting a `recbuf` value, mochiweb_socket_server sockets end up with a `buffer` size of 1460 (the user-level buffer - http://erlang.org/doc/man/inet.html), which causes this limitation (`recbuf` and `sndbuf` are indeed defaulting to much larger values). This default value of 1460 seems to be coming from [Erl inet_drv.c] (https://github.com/erlang/otp/blob/56f93ad10f89e6b8d3372e45127ec9fdc3fca35b/erts/emulator/drivers/common/inet_drv.c#L923) and initialized [here] (https://github.com/erlang/otp/blob/56f93ad10f89e6b8d3372e45127ec9fdc3fca35b/erts/emulator/drivers/common/inet_drv.c#L9057). ## Steps to Reproduce Install CouchDB 2.3 (or build from `master`) and run with default configuration (specifically `chttpd.server_option = [{recbuf, undefined}]`). Execute a request with a URL length (excluding protocol and authority) exceeding 1460 characters (method or endpoint appear to be irrelevant). ## Context We are using PouchDB to interact with our CouchDB database and ran into this error with view queries. PouchDB does issue a POST request instead of a GET, but only when the query string exceeds 2000 characters (see [here] (https://github.com/pouchdb/pouchdb/blob/f87c57a3b3b791dbb62bd881378755ca0a195d13/packages/node_modules/pouchdb-abstract-mapreduce/src/index.js#L262)). ## Your Environment * Version used: 2.3.0, master * Using: Chrome, cURL, wget * Operating system: Ubuntu 18, Ubuntu 16 (desktop) Even if it is decided that this is desired behavior, I believe this change should at least be documented.
11. **title:** Requests return 400 Bad Request when URL length exceeds 1460 characters
    body: When using `chttpd.server_option = [{recbuf, undefined}]`, requests with URL length exceeding 1459 characters fail with `400 Bad Request` with no error in the logs. (The relevant URL length excludes protocol and userinfo). Example of failing request: ``` curl -v http://127.0.0.1:5984/_users/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ``` URL length is 1467 (1460 excluding protocol and userinfo) ## Expected Behavior Should return: ``` Host: 127.0.0.1:5984 User-Agent: curl/7.58.0 Accept: */* HTTP/1.1 404 Object Not Found Cache-Control: must-revalidate Content-Length: 41 Content-Type: application/json Date: Thu, 13 Dec 2018 13:18:50 GMT Server: CouchDB/2.3.0 (Erlang OTP/20) X-Couch-Request-ID: 0b4ad7148f X-CouchDB-Body-Time: 0 {"error":"not_found","reason":"missing"} ``` ## Current Behavior Actually returns: ``` HTTP/1.1 Host: 127.0.0.1:5984 User-Agent: curl/7.58.0 Accept: */* HTTP/1.1 400 Bad Request Connection: close Content-Length: 0 Date: Thu, 13 Dec 2018 13:15:09 GMT Server: MochiWeb/1.0 (Any of you quaids got a smint?) ``` Example of successful request: ``` curl -v http://127.0.0.1:5984/_users/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ``` URL length is 1466 (1459 excluding protocol and userinfo). ## Possible Cause It appears that, by not setting a `recbuf` value, mochiweb_socket_server sockets end up with a `buffer` size of 1460 (the user-level buffer - http://erlang.org/doc/man/inet.html), which causes this limitation (`recbuf` and `sndbuf` are indeed defaulting to much larger values). This default value of 1460 seems to be coming from [Erl inet_drv.c] (https://github.com/erlang/otp/blob/56f93ad10f89e6b8d3372e45127ec9fdc3fca35b/erts/emulator/drivers/common/inet_drv.c#L923) and initialized [here] (https://github.com/erlang/otp/blob/56f93ad10f89e6b8d3372e45127ec9fdc3fca35b/erts/emulator/drivers/common/inet_drv.c#L9057). ## Steps to Reproduce Install CouchDB 2.3 (or build from `master`) and run with default configuration (specifically `chttpd.server_option = [{recbuf, undefined}]`). Execute a request with a URL length (excluding protocol and authority) exceeding 1460 characters (method or endpoint appear to be irrelevant). ## Context We are using PouchDB to interact with our CouchDB database and ran into this error with view queries. PouchDB does issue a POST request instead of a GET, but only when the query string exceeds 2000 characters (see [here] (https://github.com/pouchdb/pouchdb/blob/f87c57a3b3b791dbb62bd881378755ca0a195d13/packages/node_modules/pouchdb-abstract-mapreduce/src/index.js#L262)). ## Your Environment * Version used: 2.3.0, master * Using: Chrome, cURL, wget * Operating system: Ubuntu 18, Ubuntu 16 (desktop) Even if it is decided that this is desired behavior, I believe this change should at least be documented.
12. **title:** Requests return 400 Bad Request when URL length exceeds 1460 characters
    body: When using `chttpd.server_option = [{recbuf, undefined}]`, requests with URL length exceeding 1459 characters fail with `400 Bad Request` with no error in the logs. (The relevant URL length excludes protocol and userinfo). Example of failing request: ``` curl -v http://127.0.0.1:5984/_users/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ``` URL length is 1467 (1460 excluding protocol and userinfo) ## Expected Behavior Should return: ``` Host: 127.0.0.1:5984 User-Agent: curl/7.58.0 Accept: */* HTTP/1.1 404 Object Not Found Cache-Control: must-revalidate Content-Length: 41 Content-Type: application/json Date: Thu, 13 Dec 2018 13:18:50 GMT

Server: CouchDB/2.3.0 (Erlang OTP/20) X-Couch-Request-ID: 0b4ad7148f X-CouchDB-Body-Time: 0 {"error":"not_found","reason":"missing"} ``` ## Current Behavior Actually returns: ``` HTTP/1.1 Host: 127.0.0.1:5984 User-Agent: curl/7.58.0 Accept: */* HTTP/1.1 400 Bad Request Connection: close Content-Length: 0 Date: Thu, 13 Dec 2018 13:15:09 GMT Server: MochiWeb/1.0 (Any of you quaids got a smint?) ``` Example of successful request: ``` curl -v http://127.0.0.1:5984/_users/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ``` URL length is 1466 (1459 excluding protocol and userinfo). ## Possible Cause It appears that, by not setting a `recbuf` value, mochiweb_socket_server sockets end up with a `buffer` size of 1460 (the user-level buffer - http://erlang.org/doc/man/inet.html), which causes this limitation (`recbuf` and `sndbuf` are indeed defaulting to much larger values). This default value of 1460 seems to be coming from [Erl inet_drv.c] (https://github.com/erlang/otp/blob/56f93ad10f89e6b8d3372e45127ec9fdc3fca35b/erts/emulator/drivers/common/inet_drv.c#L923) and initialized [here] (https://github.com/erlang/otp/blob/56f93ad10f89e6b8d3372e45127ec9fdc3fca35b/erts/emulator/drivers/common/inet_drv.c#L9057). ## Steps to Reproduce Install CouchDB 2.3 (or build from `master`) and run with default configuration (specifically `chttpd.server_option = [{recbuf, undefined}]`). Execute a request with a URL length (excluding protocol and authority) exceeding 1460 characters (method or endpoint appear to be irrelevant). ## Context We are using PouchDB to interact with our CouchDB database and ran into this error with view queries. PouchDB does issue a POST request instead of a GET, but only when the query string exceeds 2000 characters (see [here] (https://github.com/pouchdb/pouchdb/blob/f87c57a3b3b791dbb62bd881378755ca0a195d13/packages/node_modules/pouchdb-abstract-mapreduce/src/index.js#L262)). ## Your Environment * Version used: 2.3.0, master * Using: Chrome, cURL, wget * Operating system: Ubuntu 18, Ubuntu 16 (desktop) Even if it is decided that this is desired behavior, I believe this change should at least be documented.

13. **title:** Requests return 400 Bad Request when URL length exceeds 1460 characters
    body: When using `chttpd.server_option = [{recbuf, undefined}]`, requests with URL length exceeding 1459 characters fail with `400 Bad Request` with no error in the logs. (The relevant URL length excludes protocol and userinfo). Example of failing request: ``` curl -v http://127.0.0.1:5984/_users/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ``` URL length is 1467 (1460 excluding protocol and userinfo) ## Expected Behavior Should return: ``` Host: 127.0.0.1:5984 User-Agent: curl/7.58.0 Accept: */* HTTP/1.1 404 Object Not Found Cache-Control: must-revalidate Content-Length: 41 Content-Type: application/json Date: Thu, 13 Dec 2018 13:18:50 GMT Server: CouchDB/2.3.0 (Erlang OTP/20) X-Couch-Request-ID: 0b4ad7148f X-CouchDB-Body-Time: 0 {"error":"not_found","reason":"missing"} ``` ## Current Behavior Actually returns: ``` HTTP/1.1 Host: 127.0.0.1:5984 User-Agent: curl/7.58.0 Accept: */* HTTP/1.1 400 Bad Request Connection: close Content-Length: 0 Date: Thu, 13 Dec 2018 13:15:09 GMT Server: MochiWeb/1.0 (Any of you quaids got a smint?) ``` Example of successful request: ``` curl -v http://127.0.0.1:5984/_users/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ``` URL length is 1466 (1459 excluding protocol and userinfo). ## Possible Cause It appears that, by not setting a `recbuf` value, mochiweb_socket_server sockets end up with a `buffer` size of 1460 (the user-level buffer - http://erlang.org/doc/man/inet.html), which causes this limitation (`recbuf` and `sndbuf` are indeed defaulting to much larger values). This default value of 1460 seems to be coming from [Erl inet_drv.c] (https://github.com/erlang/otp/blob/56f93ad10f89e6b8d3372e45127ec9fdc3fca35b/erts/emulator/drivers/common/inet_drv.c#L923) and initialized [here] (https://github.com/erlang/otp/blob/56f93ad10f89e6b8d3372e45127ec9fdc3fca35b/erts/emulator/drivers/common/inet_drv.c#L9057). ## Steps to Reproduce Install CouchDB 2.3 (or build from `master`) and run with default configuration (specifically `chttpd.server_option = [{recbuf, undefined}]`). Execute a request with a URL length (excluding protocol and authority) exceeding 1460 characters (method or endpoint appear to be irrelevant). ## Context We are using PouchDB to interact with our CouchDB database and ran into this error with view queries. PouchDB does issue a POST request instead of a GET, but only when the query string exceeds 2000 characters (see [here] (https://github.com/pouchdb/pouchdb/blob/f87c57a3b3b791dbb62bd881378755ca0a195d13/packages/node_modules/pouchdb-abstract-mapreduce/src/index.js#L262)). ## Your Environment * Version used: 2.3.0, master * Using: Chrome, cURL, wget * Operating system: Ubuntu 18, Ubuntu 16 (desktop) Even if it is decided that this is desired behavior, I believe this change should at least be documented.
    label: code-design
14. **title:** Requests return 400 Bad Request when URL length exceeds 1460 characters
    body: When using `chttpd.server_option = [{recbuf, undefined}]`, requests with URL length exceeding 1459 characters fail with `400 Bad Request` with no error in the logs. (The relevant URL length excludes protocol and userinfo). Example of failing request: ``` curl -v http://127.0.0.1:5984/_users/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ``` URL length is 1467 (1460 excluding protocol and userinfo) ## Expected Behavior Should return: ``` Host: 127.0.0.1:5984 User-Agent: curl/7.58.0 Accept: */* HTTP/1.1 404 Object Not Found Cache-Control: must-revalidate Content-Length: 41 Content-Type: application/json Date: Thu, 13 Dec 2018 13:18:50 GMT Server: CouchDB/2.3.0 (Erlang OTP/20) X-Couch-Request-ID: 0b4ad7148f X-CouchDB-Body-Time: 0 {"error":"not_found","reason":"missing"} ``` ## Current Behavior Actually returns: ``` HTTP/1.1 Host: 127.0.0.1:5984 User-Agent: curl/7.58.0 Accept: */* HTTP/1.1 400 Bad Request Connection: close Content-Length: 0 Date: Thu, 13 Dec 2018 13:15:09 GMT Server: MochiWeb/1.0 (Any of you quaids got a smint?) ``` Example of successful request: ``` curl -v http://127.0.0.1:5984/_users/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ``` URL length is 1466 (1459 excluding protocol and userinfo). ## Possible Cause It appears that, by not setting a `recbuf` value, mochiweb_socket_server sockets end up with a `buffer` size of 1460 (the user-level buffer - http://erlang.org/doc/man/inet.html), which causes this limitation (`recbuf` and `sndbuf` are indeed defaulting to much larger values). This default value of 1460 seems to be coming from [Erl inet_drv.c] (https://github.com/erlang/otp/blob/56f93ad10f89e6b8d3372e45127ec9fdc3fca35b/erts/emulator/drivers/common/inet_drv.c#L923) and initialized [here] (https://github.com/erlang/otp/blob/56f93ad10f89e6b8d3372e45127ec9fdc3fca35b/erts/emulator/drivers/common/inet_drv.c#L9057). ## Steps to Reproduce Install CouchDB 2.3 (or build from `master`) and run with default configuration (specifically `chttpd.server_option = [{recbuf, undefined}]`). Execute a request with a URL length (excluding protocol and authority) exceeding 1460 characters (method or endpoint appear to be irrelevant). ## Context We are using PouchDB to interact with our CouchDB database and ran into this error with view queries. PouchDB does issue a POST request instead of a GET, but only when the query string exceeds 2000 characters (see [here] (https://github.com/pouchdb/pouchdb/blob/f87c57a3b3b791dbb62bd881378755ca0a195d13/packages/node_modules/pouchdb-abstract-mapreduce/src/index.js#L262)). ## Your Environment * Version used: 2.3.0, master * Using: Chrome, cURL, wget * Operating system: Ubuntu 18, Ubuntu 16 (desktop) Even if it is decided that this is desired behavior, I believe this change should at least be documented.
15. **title:** Requests return 400 Bad Request when URL length exceeds 1460 characters
    body: When using `chttpd.server_option = [{recbuf, undefined}]`, requests with URL length exceeding 1459 characters fail with `400 Bad Request` with no error in the logs. (The relevant URL length excludes protocol and userinfo). Example of failing request: ``` curl -v http://127.0.0.1:5984/_users/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ``` URL length is 1467 (1460 excluding protocol and userinfo) ## Expected Behavior Should return: ``` Host: 127.0.0.1:5984 User-Agent: curl/7.58.0 Accept: */* HTTP/1.1 404 Object Not Found Cache-Control: must-revalidate Content-Length: 41 Content-Type: application/json Date: Thu, 13 Dec 2018 13:18:50 GMT Server: CouchDB/2.3.0 (Erlang OTP/20) X-Couch-Request-ID: 0b4ad7148f X-CouchDB-Body-Time: 0 {"error":"not_found","reason":"missing"} ``` ## Current Behavior Actually returns: ``` HTTP/1.1 Host: 127.0.0.1:5984 User-Agent: curl/7.58.0 Accept: */* HTTP/1.1 400 Bad Request Connection: close Content-Length: 0 Date: Thu, 13 Dec 2018 13:15:09 GMT Server: MochiWeb/1.0 (Any of you quaids got a smint?) ``` Example of successful request: ``` curl -v http://127.0.0.1:5984/_users/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ``` URL length is 1466 (1459 excluding protocol and userinfo). ## Possible Cause It appears that, by not setting a `recbuf` value, mochiweb_socket_server sockets end up with a `buffer` size of 1460 (the user-level buffer - http://erlang.org/doc/man/inet.html), which causes this limitation (`recbuf` and `sndbuf` are indeed defaulting to much larger values). This default value of 1460 seems to be coming from [Erl inet_drv.c] (https://github.com/erlang/otp/blob/56f93ad10f89e6b8d3372e45127ec9fdc3fca35b/erts/emulator/drivers/common/inet_drv.c#L923) and initialized [here] (https://github.com/erlang/otp/blob/56f93ad10f89e6b8d3372e45127ec9fdc3fca35b/erts/emulator/drivers/common/inet_drv.c#L9057). ## Steps to Reproduce Install CouchDB 2.3 (or build from `master`) and run with default configuration (specifically `chttpd.server_option = [{recbuf, undefined}]`). Execute a request with a URL length (excluding protocol and authority) exceeding 1460 characters (method or endpoint appear to be irrelevant). ## Context We are using PouchDB to interact with our CouchDB database and ran into this error with view queries. PouchDB does issue a POST request instead of a GET, but only when the query string exceeds 2000 characters (see [here] (https://github.com/pouchdb/pouchdb/blob/f87c57a3b3b791dbb62bd881378755ca0a195d13/packages/node_modules/pouchdb-abstract-mapreduce/src/index.js#L262)). ## Your Environment * Version used: 2.3.0, master * Using: Chrome, cURL, wget * Operating system: Ubuntu 18, Ubuntu 16 (desktop) Even if it is decided that this is desired behavior, I believe this change should at least be documented.
16. **title:** Requests return 400 Bad Request when URL length exceeds 1460 characters
    body: When using `chttpd.server_option = [{recbuf, undefined}]`, requests with URL length exceeding 1459 characters fail with `400 Bad Request` with no error in the logs. (The relevant URL length excludes protocol and userinfo). Example of failing request: ``` curl -v http://127.0.0.1:5984/_users/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ``` URL length is 1467 (1460 excluding protocol and userinfo) ## Expected Behavior Should return: ``` Host: 127.0.0.1:5984 User-Agent: curl/7.58.0 Accept: */*

HTTP/1.1 404 Object Not Found Cache-Control: must-revalidate Content-Length: 41 Content-Type: application/json Date: Thu, 13 Dec 2018 13:18:50 GMT Server: CouchDB/2.3.0 (Erlang OTP/20) X-Couch-Request-ID: 0b4ad7148f X-CouchDB-Body-Time: 0 {"error":"not_found","reason":"missing"} ``` ## Current Behavior Actually returns: ``` HTTP/1.1 Host: 127.0.0.1:5984 User-Agent: curl/7.58.0 Accept: */* HTTP/1.1 400 Bad Request Connection: close Content-Length: 0 Date: Thu, 13 Dec 2018 13:15:09 GMT Server: MochiWeb/1.0 (Any of you quaids got a smint?) ``` Example of successful request: ``` curl -v http://127.0.0.1:5984/_users/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ``` URL length is 1466 (1459 excluding protocol and userinfo). ## Possible Cause It appears that, by not setting a `recbuf` value, mochiweb_socket_server sockets end up with a `buffer` size of 1460 (the user-level buffer - http://erlang.org/doc/man/inet.html), which causes this limitation (`recbuf` and `sndbuf` are indeed defaulting to much larger values). This default value of 1460 seems to be coming from [Erl inet_drv.c] (https://github.com/erlang/otp/blob/56f93ad10f89e6b8d3372e45127ec9fdc3fca35b/erts/emulator/drivers/common/inet_drv.c#L923) and initialized [here] (https://github.com/erlang/otp/blob/56f93ad10f89e6b8d3372e45127ec9fdc3fca35b/erts/emulator/drivers/common/inet_drv.c#L9057). ## Steps to Reproduce Install CouchDB 2.3 (or build from `master`) and run with default configuration (specifically `chttpd.server_option = [{recbuf, undefined}]`). Execute a request with a URL length (excluding protocol and authority) exceeding 1460 characters (method or endpoint appear to be irrelevant). ## Context We are using PouchDB to interact with our CouchDB database and ran into this error with view queries. PouchDB does issue a POST request instead of a GET, but only when the query string exceeds 2000 characters (see [here] (https://github.com/pouchdb/pouchdb/blob/f87c57a3b3b791dbb62bd881378755ca0a195d13/packages/node_modules/pouchdb-abstract-mapreduce/src/index.js#L262)). ## Your Environment * Version used: 2.3.0, master * Using: Chrome, cURL, wget * Operating system: Ubuntu 18, Ubuntu 16 (desktop) Even if it is decided that this is desired behavior, I believe this change should at least be documented.
label: code-design

17. **title:** Requests return 400 Bad Request when URL length exceeds 1460 characters
body: When using `chttpd.server_option = [{recbuf, undefined}]`, requests with URL length exceeding 1459 characters fail with `400 Bad Request` with no error in the logs. (The relevant URL length excludes protocol and userinfo). Example of failing request: ``` curl -v http://127.0.0.1:5984/_users/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ``` URL length is 1467 (1460 excluding protocol and userinfo) ## Expected Behavior Should return: ``` Host: 127.0.0.1:5984 User-Agent: curl/7.58.0 Accept: */* HTTP/1.1 404 Object Not Found Cache-Control: must-revalidate Content-Length: 41 Content-Type: application/json Date: Thu, 13 Dec 2018 13:18:50 GMT Server: CouchDB/2.3.0 (Erlang OTP/20) X-Couch-Request-ID: 0b4ad7148f X-CouchDB-Body-Time: 0 {"error":"not_found","reason":"missing"} ``` ## Current Behavior Actually returns: ``` HTTP/1.1 Host: 127.0.0.1:5984 User-Agent: curl/7.58.0 Accept: */* HTTP/1.1 400 Bad Request Connection: close Content-Length: 0 Date: Thu, 13 Dec 2018 13:15:09 GMT Server: MochiWeb/1.0 (Any of you quaids got a smint?) ``` Example of successful request: ``` curl -v http://127.0.0.1:5984/_users/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ``` URL length is 1466 (1459 excluding protocol and userinfo). ## Possible Cause It appears that, by not setting a `recbuf` value, mochiweb_socket_server sockets end up with a `buffer` size of 1460 (the user-level buffer - http://erlang.org/doc/man/inet.html), which causes this limitation (`recbuf` and `sndbuf` are indeed defaulting to much larger values). This default value of 1460 seems to be coming from [Erl inet_drv.c] (https://github.com/erlang/otp/blob/56f93ad10f89e6b8d3372e45127ec9fdc3fca35b/erts/emulator/drivers/common/inet_drv.c#L923) and initialized [here] (https://github.com/erlang/otp/blob/56f93ad10f89e6b8d3372e45127ec9fdc3fca35b/erts/emulator/drivers/common/inet_drv.c#L9057). ## Steps to Reproduce Install CouchDB 2.3 (or build from `master`) and run with default configuration (specifically `chttpd.server_option = [{recbuf, undefined}]`). Execute a request with a URL length (excluding protocol and authority) exceeding 1460 characters (method or endpoint appear to be irrelevant). ## Context We are using PouchDB to interact with our CouchDB database and ran into this error with view queries. PouchDB does issue a POST request instead of a GET, but only when the query string exceeds 2000 characters (see [here] (https://github.com/pouchdb/pouchdb/blob/f87c57a3b3b791dbb62bd881378755ca0a195d13/packages/node_modules/pouchdb-abstract-mapreduce/src/index.js#L262)). ## Your Environment * Version used: 2.3.0, master * Using: Chrome, cURL, wget * Operating system: Ubuntu 18, Ubuntu 16 (desktop) Even if it is decided that this is desired behavior, I believe this change should at least be documented.
18. **title:** Requests return 400 Bad Request when URL length exceeds 1460 characters
body: When using `chttpd.server_option = [{recbuf, undefined}]`, requests with URL length exceeding 1459 characters fail with `400 Bad Request` with no error in the logs. (The relevant URL length excludes protocol and userinfo). Example of failing request: ``` curl -v http://127.0.0.1:5984/_users/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ``` URL length is 1467 (1460 excluding protocol and userinfo) ## Expected Behavior Should return: ``` Host: 127.0.0.1:5984 User-Agent: curl/7.58.0 Accept: */* HTTP/1.1 404 Object Not Found Cache-Control: must-revalidate Content-Length: 41 Content-Type: application/json Date: Thu, 13 Dec 2018 13:18:50 GMT Server: CouchDB/2.3.0 (Erlang OTP/20) X-Couch-Request-ID: 0b4ad7148f X-CouchDB-Body-Time: 0 {"error":"not_found","reason":"missing"} ``` ## Current Behavior Actually returns: ``` HTTP/1.1 Host: 127.0.0.1:5984 User-Agent: curl/7.58.0 Accept: */* HTTP/1.1 400 Bad Request Connection: close Content-Length: 0 Date: Thu, 13 Dec 2018 13:15:09 GMT Server: MochiWeb/1.0 (Any of you quaids got a smint?) ``` Example of successful request: ``` curl -v http://127.0.0.1:5984/_users/aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ``` URL length is 1466 (1459 excluding protocol and userinfo). ## Possible Cause It appears that, by not setting a `recbuf` value, mochiweb_socket_server sockets end up with a `buffer` size of 1460 (the user-level buffer - http://erlang.org/doc/man/inet.html), which causes this limitation (`recbuf` and `sndbuf` are indeed defaulting to much larger values). This default value of 1460 seems to be coming from [Erl inet_drv.c] (https://github.com/erlang/otp/blob/56f93ad10f89e6b8d3372e45127ec9fdc3fca35b/erts/emulator/drivers/common/inet_drv.c#L923) and initialized [here] (https://github.com/erlang/otp/blob/56f93ad10f89e6b8d3372e45127ec9fdc3fca35b/erts/emulator/drivers/common/inet_drv.c#L9057). ## Steps to Reproduce Install CouchDB 2.3 (or build from `master`) and run with default configuration (specifically `chttpd.server_option = [{recbuf, undefined}]`). Execute a request with a URL length (excluding protocol and authority) exceeding 1460 characters (method or endpoint appear to be irrelevant). ## Context We are using PouchDB to interact with our CouchDB database and ran into this error with view queries. PouchDB does issue a POST request instead of a GET, but only when the query string exceeds 2000 characters (see [here] (https://github.com/pouchdb/pouchdb/blob/f87c57a3b3b791dbb62bd881378755ca0a195d13/packages/node_modules/pouchdb-abstract-mapreduce/src/index.js#L262)). ## Your Environment * Version used: 2.3.0, master * Using: Chrome, cURL, wget * Operating system: Ubuntu 18, Ubuntu 16 (desktop) Even if it is decided that this is desired behavior, I believe this change should at least be documented.

github_issues_comments:

1. Hi @dianabarsan , thanks for the report. This is definitely not intended - we'll take a closer look. See https://github.com/apache/couchdb/issues/1409 for the history here; we can't directly just revert the change because the other use case will be impacted, so this may take a bit of research to resolve sufficiently.
2. **body:** http://erlang.org/pipermail/erlang-questions/2011-June/059571.html https://github.com/ninenines/cowboy/issues/3 https://github.com/ninenines/cowboy/commit/29e71cf4daec684c13047952a95ec0dc9540aad5 We're weighing our options now. Currently hoping improving mochiweb to pass `[{buffer, 8192}]` usefully thru one or another setting in `*.ini` or similar would be enough. The cowboy solution is going to be more reliable in the long run, but is a fairly large retrofit to mochiweb.
label: code-design
3. Hi @dianabarsan, Thank you for the bug report and excellent analysis. You're absolutely correct, `chttpd.server_option = [{recbuf, undefined}]).` ends up not setting recbuf value, which means Erlang userland buffer gets a default value of 1460 (when recbuf is set so is the userland buffer). This buffer size limit combined with a long-standing bug in Erlang's `{packet, http}` parser ends crashing the socket receive code with an `emsgsize` error. It's not possible to set the buffer size in mochiweb currently, so I will make a PR request to fix that. Notice: ``` 30> f(), SockInfo = fun(S) -> inet:getopts(S, [recbuf, buffer]) end, {ok, LS} = gen_tcp:listen(0, [{recbuf, 8192}]), LRes = SockInfo(LS). {ok,[{recbuf,8192},{buffer,8192}]} ``` ``` 31> f(), SockInfo = fun(S) -> inet:getopts(S, [recbuf, buffer]) end, {ok, LS} = gen_tcp:listen(0, []), LRes = SockInfo(LS). {ok,[{recbuf,131072},{buffer,1460}]} ```
4. pull request: https://github.com/mochi/mochiweb/pull/208
5. Hi @wohali , @nickva Thank you very much for the quick responses and for the additional context! The Erlang http packet parser bug was the missing piece in my understanding of why this is happening. We will be updating our configs to set a `recbuf` value for 2.3.0 installs.
6. **body:** @dianabarsan Be advised that this may adversely affect your attachment performance (if you use attachments), see #1409. We are considering a 2.3.1 fix for this bug since it's rather a surprising regression.
label: code-design
7. **body:** @nickva I'm not sure that the fix in https://github.com/mochi/mochiweb/pull/208 is going to satisfy the situation in #1409. Basically, the patch as written I *think* only sets the buffer to `8192` if `recbuf` is undefined. This means if you need >8k of headers you have *no choice* but to peg `recbuf` higher, which causes the issue seen in #1409. I don't think it's fair to tradeoff a fix for this with poor attachment performance again. Ideally we shouldn't be using a fixed 8k value in mochiweb but instead calling `getopts` to figure out the OS's window size (at least initially) and use that to set `buffer`'s size. I know you had runtime reservations about the performance of inserting a `getopts` call with every socket open but we should characterize this. An alternative without going back to

mochiweb would be to pass in `{buffer, BIGNUM}` . I don't know if your patch to mochiweb allows passing `buffer`9 in now as part of `server_options`, does it? If so this could be a documented workaround; based on testing on Linux the default was like 50k on my system, and 50k of headers should be sufficient. So in this case, we'd tell people to leave recbuf as undefined and just pass in `{buffer, 100000}` or so. As long as `buffer >= recbuf` you're fine, and we can provide the `remsh` command to determine your actual `recbuf` value as set by the OS. But the only real fix for this (arbitrary header length + undefined `recbuf`) is to stop using `{packet, http}` and allowing multiple `recv`s on the socket to get all of the header content if necessary (with appropriate limits on the # of headers and max length for each) which is a MASSIVE change to mochiweb and no one is volunteering to write it. So I'm hoping the tradeoff above is sufficient. /cc @kocolosk @davisp
label: code-design

8. @nickva bump, we need to solve this, see #1843. Any comments on my proposal?
9. @wohali oh, sorry, dropped this one, it's already solved, just need to merge latest mochiweb master to our copy and tag it
10. @nickva did you read my comment above? https://github.com/apache/couchdb/issues/1810#issuecomment-448060589 Maybe we just call it good enough with 8192 for now, but I'm not sure it's sufficient. Do we want to cut a 2.3.1 with this?
11. @wohali I wasn't sure I understood some parts 1409 so didn't know how and whether it would affect that case. > Basically, the patch as written I think only sets the buffer to 8192 if recbuf is undefined. This means if you need >8k of headers you have no choice but to peg recbuf higher, which causes the issue seen in #1409. The patch sets the buffer to 8k if recbuf if not defined, because that was the setting before the recbuf undefined became an option. I am not sure if 1k userland buffer size is what affected case 1409 the most. But this patch also allows setting a custom buffer size in general via mochiweb's socket options. > An alternative without going back to mochiweb would be to pass in {buffer, BIGNUM} . Don't think it would work currently on a listening mochiweb socket (need this patch for that). It could be set later in chttpd but by that point the headers would have failed to parsed. > and 50k of headers should be sufficient. I think it's not as much about 50k worth of headers but 50k being the longest line in the header (just the `GET /....long...path` line). That used to work well with an 8k buffer but doesn't with an 1k buffer anymore.
12. @nickva can you hop on IRC briefly to finish hashing this out? I want to be sure I understand where you're talking about the OS buffer (recbuf) and the mochiweb buffer (buffer) above, so that we don't force people to have to manually specify recbuf's size just to bump buffer's size.
13. **body:** OK, for those who are curious - @nickva 's patch to mochiweb does the right thing, it's just that above our discussion of `recbuf` vs. `buffer` as mochiweb settings isn't 100% clear. With the patch to mochiweb, and a stock CouchDB (no changes in `*.ini`), we set mochiweb's `recbuf` to undefined. Mochiweb then lets the kernel manage its buffer size itself (`recbuf` is undefined), and sets the Erlang kernel buffer to `8192`. Because the Erlang `http` packet parser requires the entire header line (URL path, cookie, anything is fair game) to be in the buffer returned from the kernel, if the request exceeds `8192` bytes, erlang/mochiweb explode, and CouchDB will return a 400. (The parser can't handle header lines split across buffers.) The workaround for this is to set your `server_options = [{buffer, 16384}]` or other suitably large size, leaving `recbuf` undefined. This way the kernel still manages its buffer size automatically, and only the Erlang userland buffer is made bigger to handle that. This doesn't catch 100% of the use cases - `buffer` can only be made as large as the kernel `recbuf`, and if the kernel buffer isn't big enough for all the headers, you still fail. In that case - and in my testing, you're talking about ~50k on Linux - you have to bump `recbuf` (which, in turn, auto-bumps `buffer` to the same size). This should be very, very rare, though, on modern OSes. Unfortunately there's no easy way to make mochiweb smarter here. If `buffer < recbuf`, and the line length is > `buffer`, mochiweb gets an `emsgsize` error back and returns a 400 before Couch even gets its hands on anything. If `buffer = recbuf` and the Request-uri is still too long, Erlang fails with an `http_error` message. I think our best bet is to document this. What's left to do: 1. Write a test that bumping `[{buffer, 16384}]` for a 16k URL path is successful 1. Document the above in a useful way in the CouchDB Documentation troubleshooting section 1. Release 2.3.1, as this is very surprising for users.
    label: code-design
14. Is there a due date for the release of 2.3.1 as this bug is biting us currently.
15. We are currently in the release process - need to finish acceptance testing.
16. **body:** @wohali Awesome! For now we implemented a workaround. Thanks for the prompt reply and keep up the good work!
    label: code-design

github_pulls:

github_pulls_comments:

github_pulls_reviews:

jira_issues:

jira_issues_comments: