Checking http// connectivitycheck. android. com/ generate_204

Android is made by Google, but at least as of 2018 it can still be used without Google services. Reasons for this include privacy and independence. Of course, it would be better to switch to another platform entirely, but privacy focused free phone software or alternative operating system for Android hardware is not yet there. One of the features of Calling Android home, which is difficult to disable or remove, is to check your Internet connection after joining a Wi-Fi network. Each Android device asks after connecting to a wireless network. This google page is empty and returns status code 204. Most captive portals use a status code of 200 to service a page or 203 to redirect somewhere, so this works as a simple check whether the device is behind the portal. Now I wanted my Fairphone 2 from connecting to this server. Setting a verified URL to some nonsense will turn off the check, but then Android claims it doesn't have an internet connection all the time, showing a terrible exclamation point icon. But which server should I use instead? As I work multiple web servers for my own and friends websites, the way to go was to create a link that generates 204 results myself. To create the correct answer I made a simple 204.php as follows. <?php http_response_code(204);= exit();= alternatively,= if= php= is= not= available,= one= can= also= add= a= simple= redirect= to= the= websever= configuration,= in= my= case= apache:=?><VirtualHost *:80=> ... RedirectMatch 204 (.*)/204$ ... </VirtualHost> Then, in the root terminal on the phone I entered the following two commands, from pat_512 to the Fairphone Forum (see link below). In fact, it can also be used to track the device, recording on the server when it was in which Wi-Fi, or rather when it had, what public IP address. Setting a URL differently across devices can be used to distinguish between their queries. Related links someone might say it only happens to me because I'm in Cuba, but I would like to believe that even in developed countries with u.S. Internet access, sometimes you may experience this problem. The problem is that since some minor Android 5 updates (and the world has only recently got version 7), Android doesn't like connecting to a Wi-Fi network without internet access. Even if you check Don't ask again above, Android won't automatically connect to this low-end connection. Sometimes I want Wi-Fi without internet It's not a rant about internet access in Cuba. I am part of Delta Proyecto, a science and comedy show that is directed by professors and students of the Faculty of Mathematics and Computer Science every Friday night at a movie theater in Havana. In this show, we store local Wi-Fi for the audience to participate and for the staff to communicate with each other. And since it's just internal, it wouldn't be connected to the internet, even if we could. More examples come to mind I could have a bunch of robot robots android devices talk to each other, again via Wi-Fi without necessarily having to have internet access. And you don't want Android to give up connecting to a completely real network. Solution Is a trick and it can break at any time because I'm aiming for a specific implementation of connectivity verification on Android. And you should always evolve for interfaces, not implementations. However, I don't have an interface to develop here, so that's what it is. Android is just trying URLs like this to decide if the connection has internet access or not: And it expects them to return HTTP status 204 (No content). I'm not sure if there's more, but I've seen these two with Android 5 & 6. The solution is to make him believe that he has internet access to tell your DNS to point these addresses to the host on your network and then have that host reply 204 on the path / generate_204. All in DD-WRT there are many ways to achieve this, but since my Wi-Fi router has a DD-WRT, I decided to do it right there. So I didn't need to rely on two devices. First, I told DnsMasq to specify that the domain is the IP address of the router as it is: # Change 10.0.0.1 to the IP of your HTTP server host record = connectivitycheck.android.com,10.0.0.1 host record = connectivitycheck.gstatic.com,10.0.0.1 And secondly, I enabled lighttpd and added the following line to lighttpd.conf: url.rewrite-once = (^/generate_204$ = > generate_204.php) And that PHP script that should be hosted somewhere lighttpd can find it, has only this line: <?php header (HTTP/1.0 204 No content); ?> After these two steps, and turning on Wi-Fi and then on, Android went from Sad Wi-Fi to Happy WiFi: ⇨ From Sad Wi-Fi to Happy WiFi Gotta say it's amazing for DD-WRT to enable PHP. I'm not a fan of PHP, I really hate it and agree that it's a fractal of bad design and other security concerns, but it's a high-level language and it was much easier for this problem than writing a server or CGI program in C. Importantly the HTTP server that you specify dns should listen to port 80. This is usually the case, but DD-WRT already uses port 80 for its own web server and port 81 for lighttpd. What I did was have a DD-WRT web server listen only on port 443 – which even with an invalid SSL certificate, I recommend-. And then I created a lighttpd listen on port 80, to answer for generate_204 and redirect blank requests to https:// ..., like this: $HTTP[host] =~ (.*) { url.redirect = (^/$ => https://%1/) } Chip in If you know an easier way to achieve it with Lighttpd, or want to share a way to do it in another installation, fire away in the comments below. If I can collect a few ways, I'll add them as a list to the post. Hi everyone,Iam having a strange problem as I can not connect to my Wi-Fi using my Pixel XL-Android 9I is точка доступу (Linksys-LAPC1750PRO & connected to pfSense, Wi-Fi works perfectly on all other devices and connects to the internet fine, except my Pixel XL- (Android 9) can not get IP and gets MSG as Click to log in . I don't have a captive portal installed on pfSense.It only happens on my Pixel XL and Samsung S8, however the iPads connect perfectly to the AP and other Wi-Fi devices.. Also, my Pixel XL connects to my Wi-Fi modem perfectly fine.... I checked other forums and links I see this problem but no solution ... Please help / advise ... When a device tries to connect to an access point or router, it must log on to the portal captivity page with the message Log in to the Wi-Fi network. How does the access point / router determine what the device needs logging in and what process is behind it? Hi, https login: where did the certificate come from? Self-signed? With Letsencrypt (== acme package) ? => Signature certificates should not be used on the captive portal, except when you want to guide each user every time you all live - and you probably don't want to. Use real certificates or move away from https login mode. You mention mobile phones, but can you be more specific? iOS (iPad - iphone, etc.) ? Anphroid? (and yes, what age / generation, etc.) ? All the latest devices, even MIcorosft OS as 10 (and 8 and 7) and Server, etc. will throw away http:// ... (like request to see if a known response is returned. You can see for yourself : it returns the simple word Success. This URL is used by Apple devices. If the answer is different from Success, then the device (OS, in fact) takes over the portal, and runs abrowser for you, so that the end user (== you) can interact, which can be a portal. I'm edgy that your device (==OS) throws to detect the presence of a portal, but something didn't work out well. I have some good - and bad news: it's probably not a pfSense ^^ problem (try the iPhone, it will run immediately ^^). I have the following code in the application that will check for an Internet connection. This works, that is, will say that there is no internet if not (as opposed to use, ConnectivityManager, which will tell you only if you are connected to the router / 3g, but does not really check the internet connection), but only until someone connects to a Wi-Fi location that redirects the user to the login page. Then the app will return the code 200 and try to continue, but eventually crashes (and crashes a bit). My app will be used mainly in a place that has such a Wi-Fi location. How do I check if a request has been redirected? class online expands AsyncTask<String, string,= string=> { відповіли = false; @Override захищена порожнеча onPreExecute() { super.onPreExecute(); pDialog2 = новий ProgressDialog(Main.this); pDialog2.setMessage(Перевірка інтернету, будь ласка, зачекайте...); </String,> </String,> pDialog2.set Grounding(false); pDialog2.show(); } protected doInBackground(String... args) { try { URL = new URL (; HttpURL Connection urlc = (HttpURLConnection) url.openConnection(); urlc.setConnectTimeout(6000); Timeout in seconds urlc.setReadTimeout(6000); urlc.connect(); if (urlc.getResponseCode() == 200) { answered = true; } more { } } catch (IOException e) { } try { int waited = 0; while (!responded & (waited < 5000)) { mHandler.postDelayed(new Runnable() { public void() { } }, 100); waited += 100; } finally { if (!responded) { h.sendEmptyMessage(0); } more { h.sendEmptyMessage(1); } } return null; } } } protected void onPostExecute(file_url) { pDialog2.dismiss(); } }