



CyberDragon Browser

V1.6.4

Quick Manual

Foreword

Why Another Browser?

When you mention the word "browser" most people will think about Mozilla Firefox, Google Chrome or Microsoft Internet Explorer. Some might think about Safari or Opera. Some may even think about Lynx, Netscape Navigator or some other old browser.

So there are plenty of browsers out there already, why yet another browser?
Simple: None of those offered privacy and security features that I wanted *out-of-the-box*.

Sure, I could always install extensions and 3rd party software for Firefox and Chrome to do these things. I could tweak about:config settings of Firefox for endless hours, tweaking cookie settings, disabling geo location, disabling saving ssl content to disk, etc.....

I could have (and have done) all this. Zillions of times. Everytime I needed new computer. Everytime Windows crashed and I had to install it all over again.

Every damn time I had to go over and over the same ritual of downloading Firefox, downloading NoScript, downloading HTTPS Everywhere, downloading Adblock Edge, getting all the filters for Adblock Edge, tweaking about:config settings, yadda-yadda-yadda

Also, in addition, I absolutely hate ads!

And I wanted to be in charge of my browser and know *exactly* what it was doing.
These three things: Simplicity, Privacy & Control were the things that I needed.

So one day I said to myself: "%&#! all this! I will make myself an browser that has by default, without installing any extra software, all the privacy and security features that I need! And one that will block ads & trackers too and run from USB stick if needed!"

And so the CyberDragon Browser 1.0 was born ...

Index

Minimum System Requirements.....	4
General.....	4
Tracker Blocker.....	8
Stop Tracking me Dammit!.....	8
Cookie Control.....	16
Crunshing Cookies.....	16
Proxy.....	24
Hiding your tracks.....	24
Encryption.....	39
Keeping your data safe.....	39
Appendix A. Key shortcuts.....	44
Appendix B. How to use CyberDragon with Tor?.....	45
Appendix C. Linux.....	57
Appendix D. Donations.....	59

Minimum System Requirements

These are the absolute minimum system requirements:

Intel Pentium III Processor (won't work with CPU that does not have MMX and SSE)

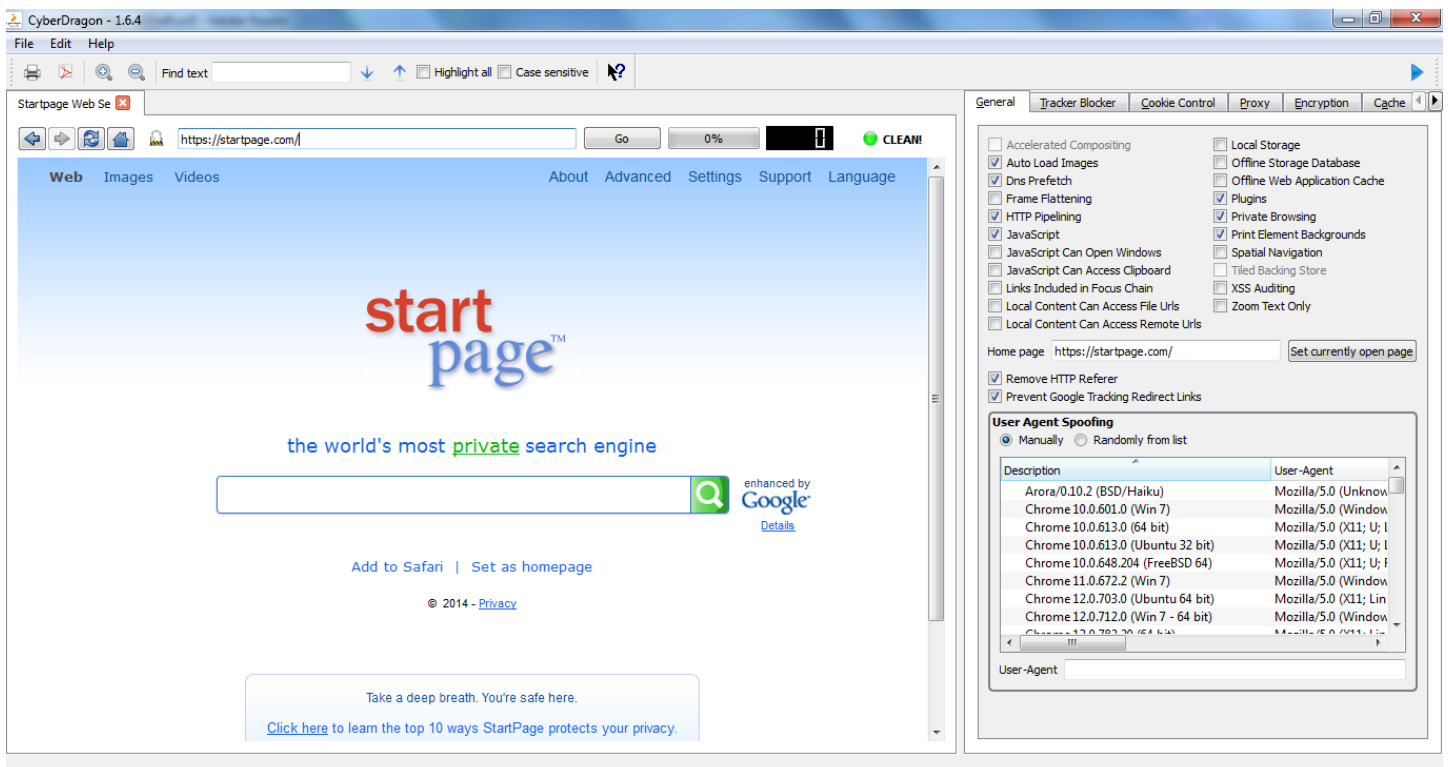
40+ MB free RAM (the more tabs you have open the more it will need)

140 MB free hard drive space

Windows Professional SP3 (Windows 7 recommended)

General

Here's the normal view that you see when starting CyberDragon browser:



And here's a very quick view of the basic functionality, more details follow.



"What's This?" Button

You click this button and then hover mouse over some other GUI widget. If the cursor will change into a question mark then it means that particular widget provides some additional information that you will get after clicking it. Many of the controls in General tab provide additional info when clicked with "What's This?" enabled.

Hide/Show Button

By clicking this button you can hide and show the CyberDragon settings.

Splitter

When you move mouse cursor over splitter area the cursor will change to two headed arrow. Keeping left mouse button pressed down and dragging to left or right you can resize settings area.

Prevent Google Tracking Redirect Links

☒ Prevent Google Tracking Redirect Links

I got idea for this from here (seems to be offline now) <http://blog.ssokolow.com/archives/2011/11/08/bypassing-google-tracking-redirects-with-https-everywhere/>

That blog explained very clearly how Google Search tracks all the links you click through it. Enabling this feature and after the Google page has completely loaded, CyberDragon will dynamically rewrite those tracking links to their normal, non-tracking form. Enabling HTTPS Enforcing option from Encryption tab will also enable this option.

Note: You must let the Google Search page to load completely for this to work, so don't rush. Only after it has successfully and completely loaded (100%) will CyberDragon rewrite those links. This was done because of code simplicity and performance reasons.

HTTP Referer hiding

☒ Remove HTTP Referer

If checked then CyberDragon will not send any HTTP referer field to web servers. HTTP Referer is one way of tracking you although not as popular as cookies and other ways.

For more information about HTTP Referer visit:

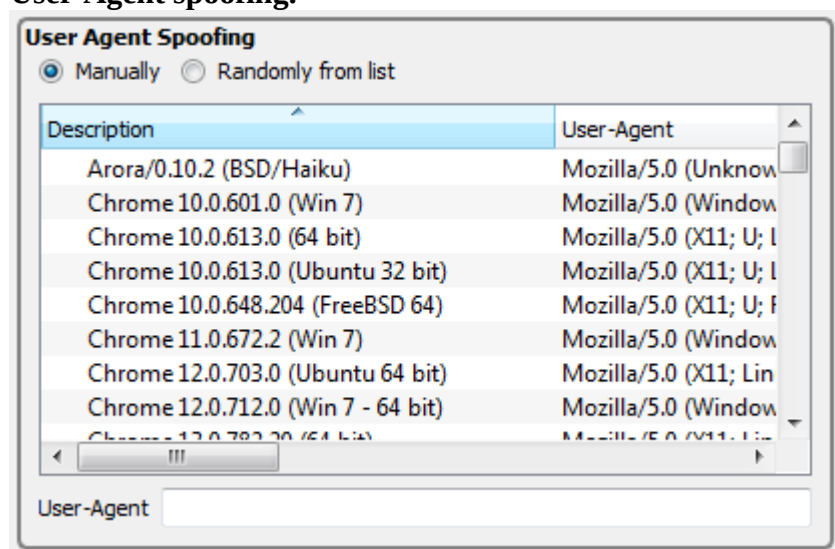
https://en.wikipedia.org/wiki/HTTP_referer#Referer_hiding

For testing that HTTP Referer hiding works visit:

<http://darklaunch.com/tools/test-referer>

You need to have JavaScript enabled for above test page.

User-Agent spoofing.



Here you can control the **User-Agent string** that is sent by **CyberDragon** to web servers. There are two cases where this might be useful:

Case 1. You want to prevent trackers from identifying your browser. To do this either choose **Manually** and write anything you wish to **User-Agent** field or select **Randomly from list**.

If you chose **Manually** and leaved **User-Agent** field empty then the default User-

Agent string is sent.

The default User-Agent string is the form of:

Mozilla/5.0 (%Platform%%Security%%Subplatform%) AppleWebKit/
%WebKitVersion% (KHTML, like Gecko) CyberDragon Safari/%WebKitVersion%

In this string the following values are replaced at run-time:

%Platform% expands to the windowing system followed by "; " if it is not Windows (e.g. "X11; ").

%Security% expands to "N; " if SSL is disabled.

%Subplatform% expands to the operating system version (e.g. "Windows NT 6.1" or "Intel Mac OS X 10.5")

%WebKitVersion% is the version of [WebKit](#) the application was compiled against.

You can also choose some of the predefined User-Agent strings from the list and if you wanted to, edit them to your wishes.

Note: If you really want to send "empty" User-Agent string then you can do so by inserting one space to **User-Agent** field. However, some pages (<https://startpage.com> for example) do not like the idea of empty User-Agent strings and will start complaining.

Case 2. There are some web pages that are only accessible by specific browser. In that case select one of the predefined User-Agent strings and edit the User-Agent field if needed to.

Note: There is currently no controls to add/change or delete predefined User-Agent strings from the list but if you really want to do it you can open file called *useragents.txt* from the CyberDragon folder and change it to your wishes.

For more information about User-Agent string see:
https://en.wikipedia.org/wiki/User_agent#User_agent_spoofing

For testing your User-Agent string
<http://whatsmyuseragent.com>

Tracker Blocker.

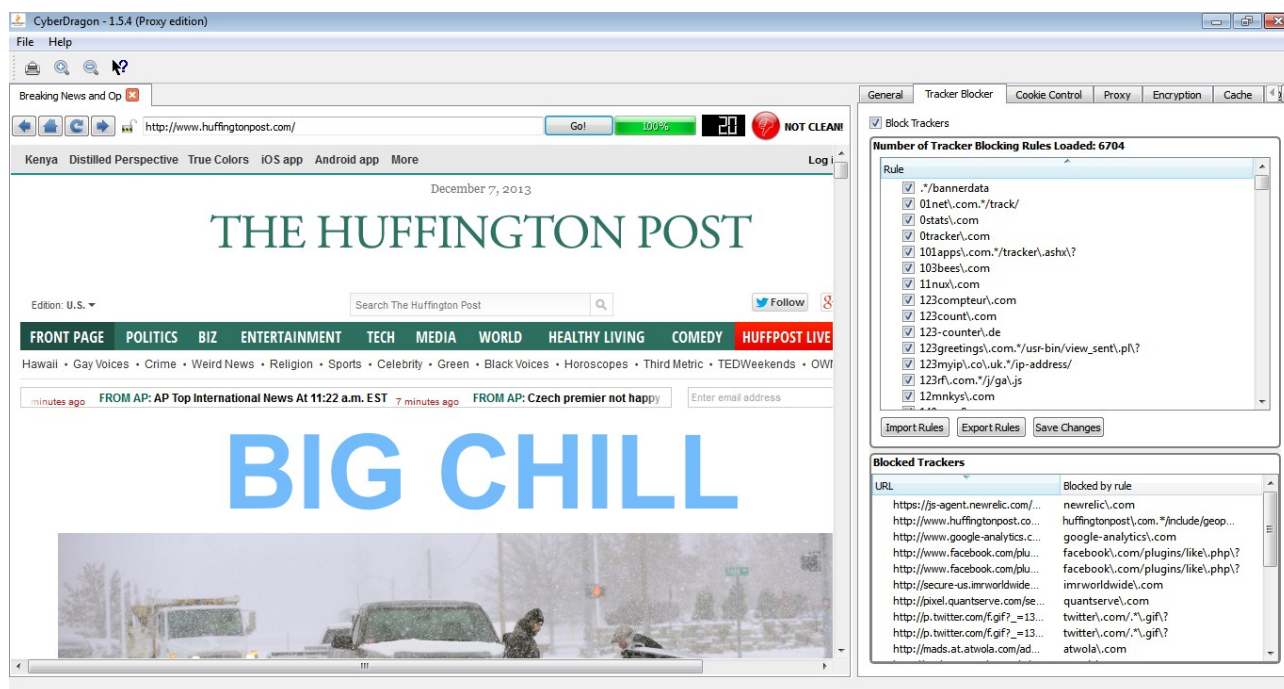
Stop Tracking me Dammit!

Everytime you visit some site you will be silently and invisibly tracked by organisations, institutions, government agencies and most importantly companies. The reasons they track you varies from control to making just plain puck with your surfing habits. They invisibly and forcibly load your computer with unwanted and unneeded banners, scripts, widgets, advertising and other stuff, slowing your Internet connection, slowing your computer, invading your privacy, exposing you to security risks and most importantly making billions of dollars with your data.

Well, not anymore!

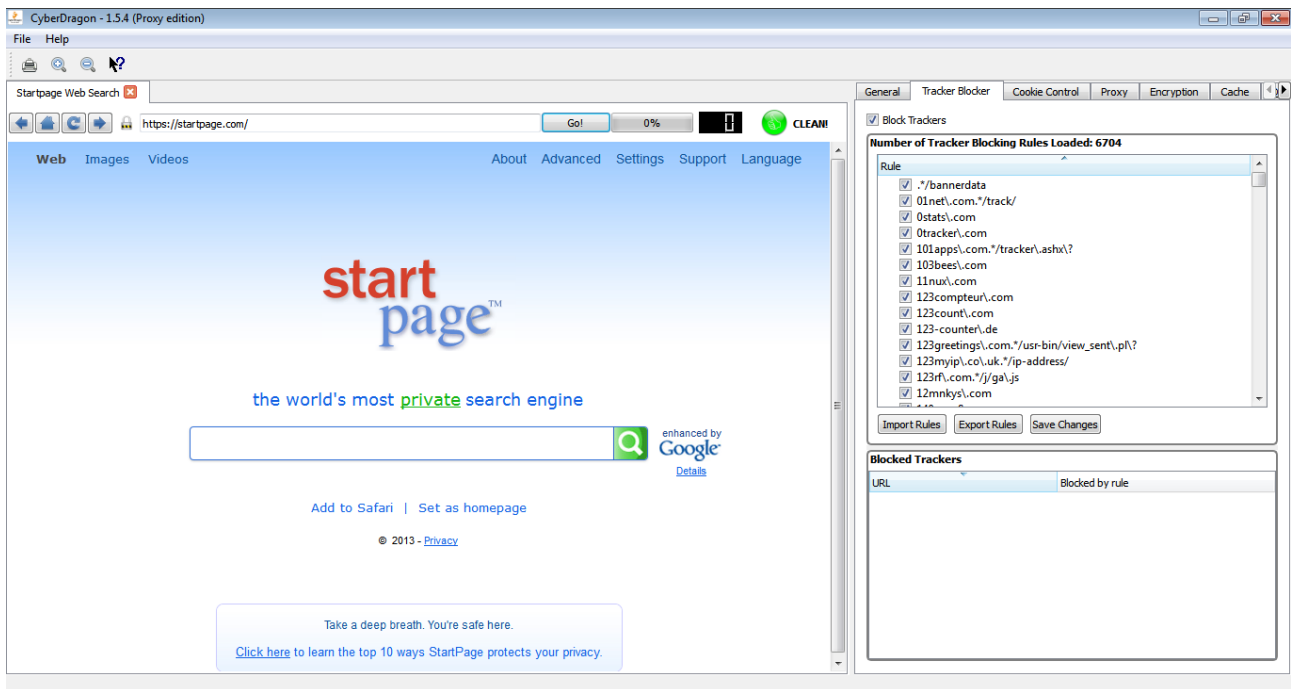
CyberDragon has built-in tracker blocker that let's you see who the trackers are and block them. It will let you to track the trackers!

Heres where the magic happens.

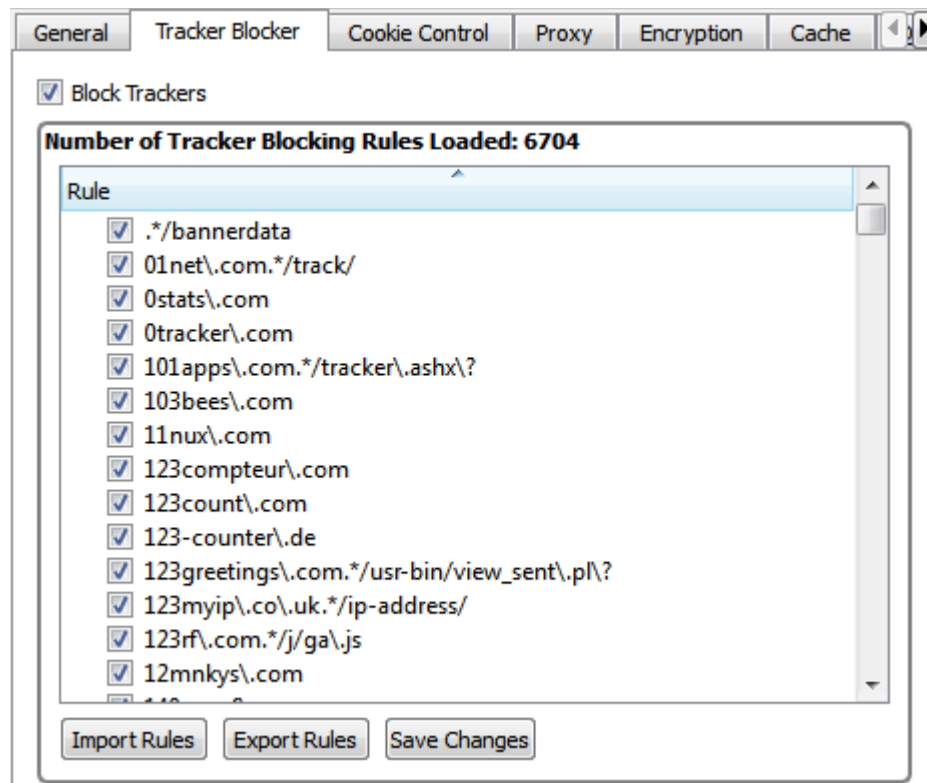


At the top right corner of the browser view you see how many trackers were found on this specific *page*. This number does not include any blocked cookies (more of that later), just the trackers that could be found. In addition to this numerical information it will also loudly tell you that this page was filthy by red orb with thumb down and message NOT CLEAN!

For sites that does not include any trackers (yes, there are those. For example: <https://startpage.com>) it will show zero trackers and green orb and message CLEAN!



Then you have the master tracker blocker view that currently has over 6000 tracker blocker rules. And just below that you have the Blocked trackers view that will show you the bad guys URL and the rule that matched it. This is especially important to know because without knowing the correct tracker blocker rule you might not be able to disable it for temporarily.



Now why would you want to disable a tracker blocker rule? Even for temporarily? Well, you see, sometimes some sites use trackers as part of their functionality.

For example, YouTube comment feature uses tracker(s).

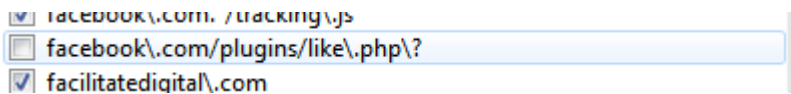
I don't currently remember the specific rule(s) but it might be the rules `s\\.youtube\\.com` and `s2\\.youtube\\.com`. In addition you might need to enable some cookies on Cookie Control tab and also check for blocked mixed content from Encryption tab. Lot's of trouble just for enabling YouTube video commenting

Or it might be that there is an actual legitame site that has ended up because of my mistake into that master tracker blocker rule list (Hey! Im a human being. And human beings make errors) and you need to disable it, or maybe even remove it.

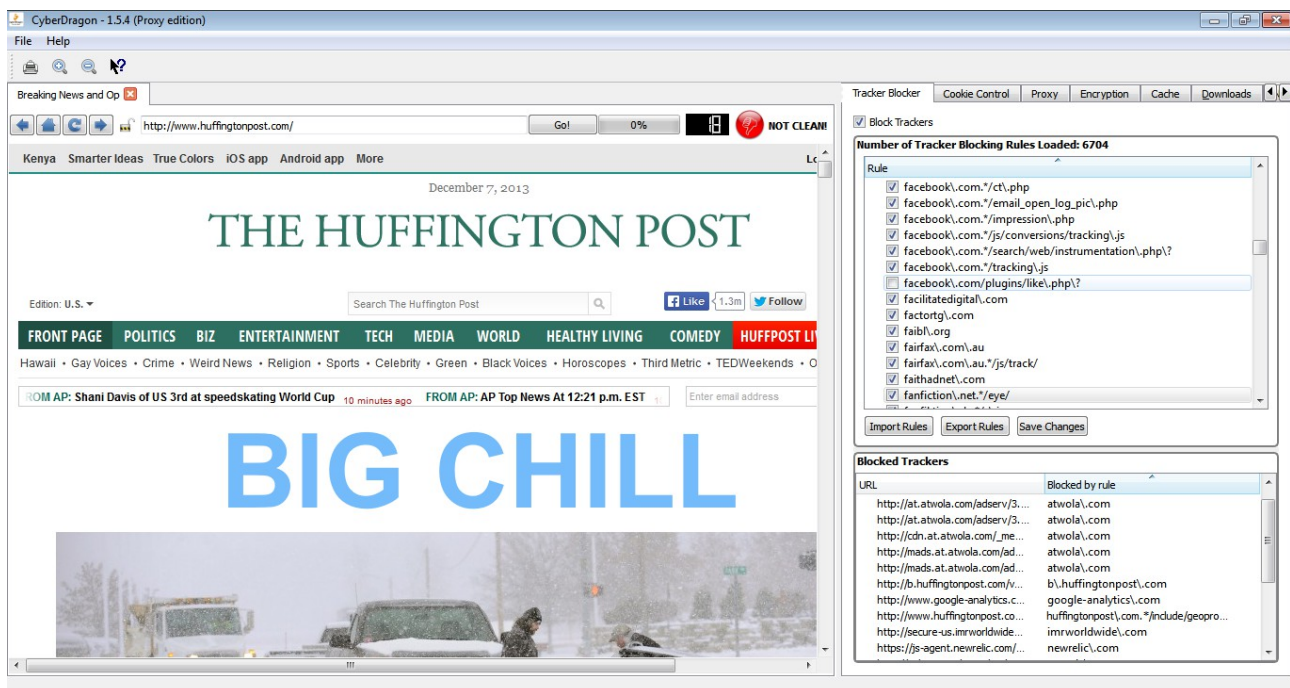
Let's go throught all the options you can do with this master blocker list.

First: Disabling/Enabling Tracker Blocker Rule.

Like I told before you can temporarily disable/enable tracker blocker rule from the list by just clicking the checkbox in front of the rule. It might be that you want to just test if some rule is giving you trouble or not. Much easier than removing, testing, adding routine that you would have to do otherwise....

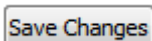


Let's temporarily disable facebook\.com/plugins/like\.php\? rule.



And then reload page. As you can see the tracker count is now 18 and you will also see facebook's like widget appearing this time. There is also no facebook\.com/plugins/like\.php\? appearing on Blocked Trackers view this time. Now that you have confirmed that tracker blocker works please enable facebook\.com/plugins/like\.php\? rule again.

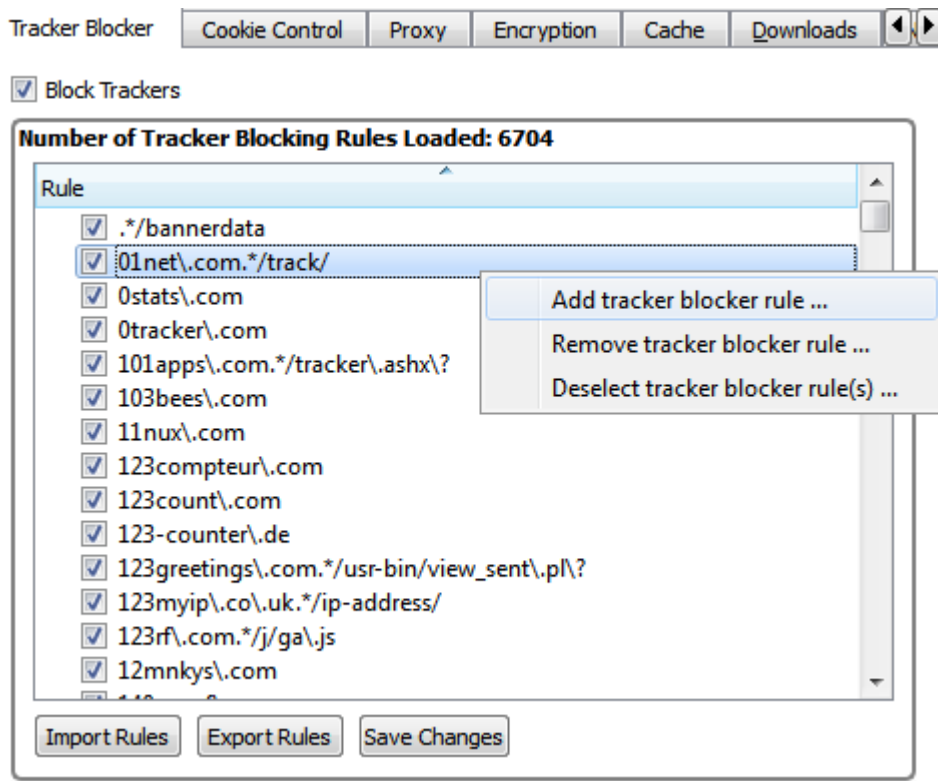
Saving Changes



None of the operations, except disabling/enabling tracker rule and exporting rules to file, are saved without pressing this button. Because we are talking about master tracker blocker rule list here (the very thing that makes blocking those trackers possible) I have decided that you must confirm all the changes you make to it (importing, adding, changing and removing) permanent by pressing this button. Think it like as a last chance before there is no turning back in case you make a very serious mistake to the list (actually, there is hope even in that case: you can close the CyberDragon and manually edit file called **filters.txt** but you know how fun that is). After you have pressed this button the CyberDragon will tell you what operation(s) you have made to master tracker blocker rule list.

Adding completely new Tracker Blocker Rule.

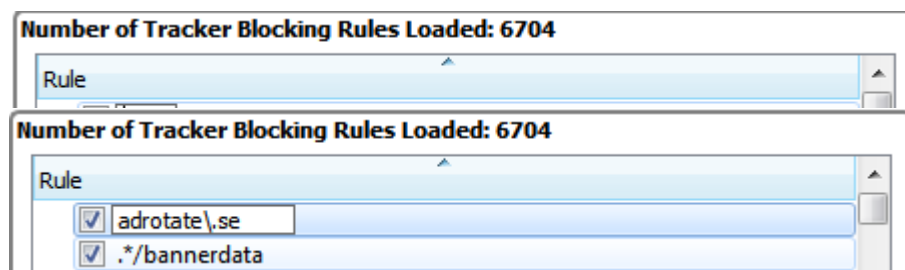
You can add completely new tracker blocker rule by right clicking with mouse over master tracker blocker list and selecting "Add Tracker Blocker Rule".



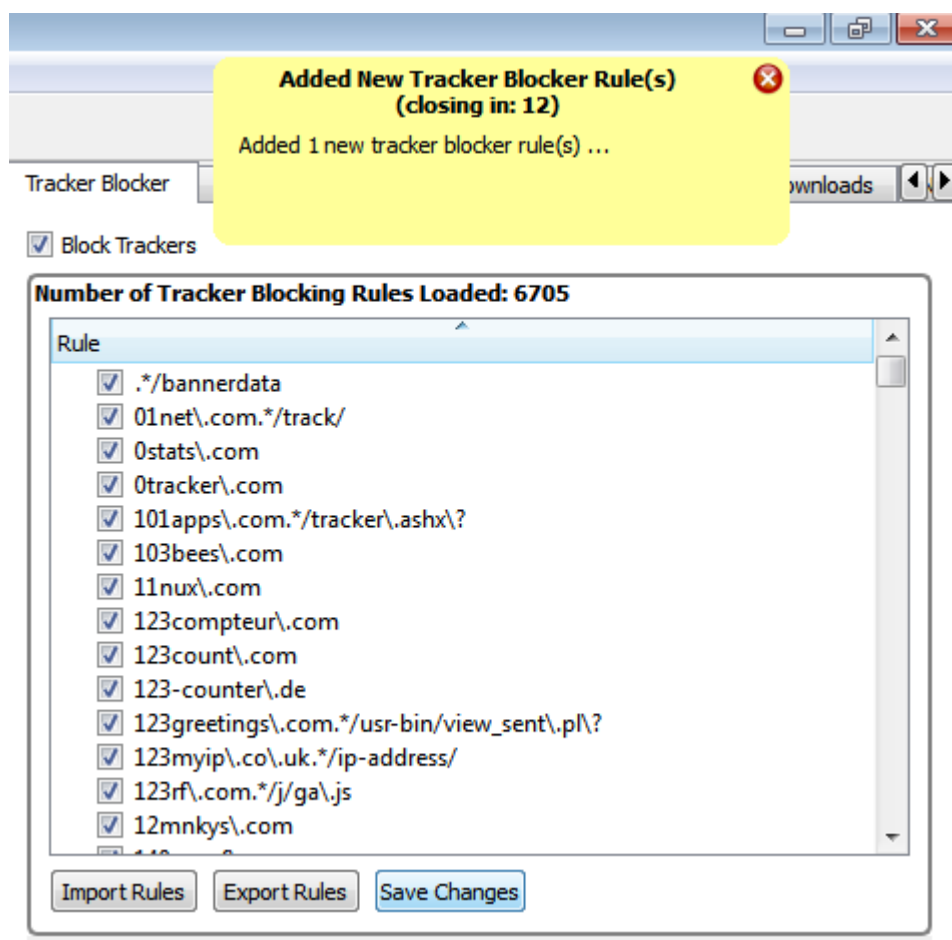
It will add an new empty field at the top of that 6000 list where you can type the name of the bad guys domain/subdomain or specific URL that points to some path or file. This list is automatically sorted so after you have pressed Enter it will put this new rule into it's proper place, unless it is an duplicate rule.

Duplicates are silently discarded to keep this list as compact as possible. Empty fields, however, are currently not discarded. So be carefull to not give any empty fields (they will end at the top of the list) and remove them!

Let's add a new rule that is missing from that list: adrotate.se



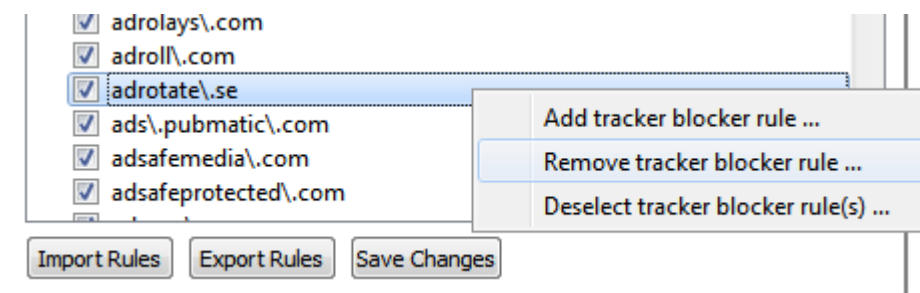
Press Enter and finally click "Save Changes"



As you can see CyberDragon will inform you that it has added new tracker blocker rule. If you had not pushed "Save Changes" button then all the changes you would have made so far with the master list would have disappeared at the exit of CyberDragon. Now remember, CyberDragon keeps this list sorted. So if you wonder where your brand new rule went just scroll down the list.

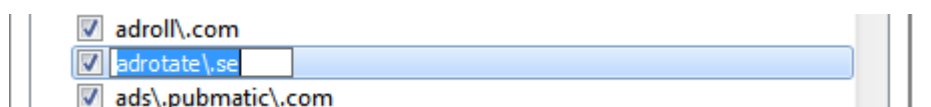
Removing Tracker Blocker Rule.

Pretty simple. Just click the rule (or rules, you can press Shift + left mouse button or Ctrl + left mouse button to select multiple rules) that you want to remove and select "Remove tracker blocker rule" from context menu that pops up with right mouse click over master tracker blocker view.



Changing Tracker Blocker Rule.

What? You found a mistake in master list? Okay. In that case just double-click on the rule that needs correction and press Enter-key.



Importing Tracker Blocker Rule.

Import Rules

This is a much much much easier way of adding new tracker blocker rules (especially if you have lot's of them) than the method mentioned previously. Basically you make a normal text file which contains one rule at each line, click this button, select your file and press Ok. Your new rules will be merged to master rule list (duplicates and empty lines are skipped though...). When you are happy press "Save Changes" button. This is a great way to add and share rules with other CyberDragon users and keep the list up-to-date.

Exporting rules.

Export Rules

And lastly, there is a way you can export rule(s) to an external text file (so that you can share them with your fellow CyberDragon users later). If you have not selected any specific rule(s) from the list then pressing this button will export the whole list (as does selecting one rule from the list, pressing Ctrl + A and then pushing Export button). If you want to export some specific rule(s) then press Shift + left mouse button (or Ctrl + left mouse button if you don't want continuous selection) and then press Export button.

Now you know all the functionality of the Tracker Blocker tab. However, one word about the format of those rules. Why they look so funny ? Why it reads s\youtube\com for example? Instead of just s.youtube.com ?

The reason is that they are regular expressions.

Regular expressions is too deep subject to handle here but I will give few quick examples.

Let's just say that they offer a way to make a very compact list of tracking rules if need to. For those interested please check the following links:

<http://perldoc.perl.org/perlretut.html>

<http://qt-project.org/doc/qt-5.0/qtcore/qregularexpression.html>

Characters dot (.), asterix (*), question mark (?) and plus (+) have a special meaning in regular expressions. If you meant literal of those characters then you have to prefix them with '\' character.

- . = means any character.
So example.com would be wrong. It would match exampleecom, example2com, exampleucom etc... Right domain match rule in this case would be example\com
- * = means zero (0) or more occurrences of the character before it.
So .* means "zero or more occurrences of any character"
For example: example.*\com would match example.com, example12343.com, examplebaddomain.com etc....
example1*\com would match: example.com, example1.com, example11.com, example111.com etc...
- {m} = Match exactly *m* characters before it. Rule example\.{2} would match exmple.it.
example.fr, example.jp, etc..
Rule exaple\.{3} would match: example.com, example.gov example.edu etc...
- {m,n} = Match at least *m* but at most *n* characters before it.
Rule example\.{2,3} would match: example.de, example.gov, example.fr, example.edu etc...
- + = means at least one (1) or more occurrences of the character before it.

Rule `example+\.` would match `example.com`, `examplee.com`, `exampleeeee.com` etc..

`[]` = Set of characters.

`|` = Alternative (OR operation)

And now the examples:

Example 1: There is an bad guy throwing targeted ads analyzing and profiling at fictitious domain `evilads.com`. We have rule `evilads\.` in our list and it is blocking their crap nicely. Now, they suddendly registered three new domains: `evilads.fr`, `evilads.jp` and `evilads.de`.

Instead of adding three more rules for these rules we change the old rule to this:

`evilads\[com|fr|jp|de]+\.`

That `|` character means OR operation. So what this rule basically says is: "Match all domain names that include `evilads.` and that end with `com` OR `fr` OR `jp` OR `de`. All with just one line of rule! Without regular expressions we would have to write four rules (for `com`, `fr`, `jp` and `de`)!

Example 2: Bad guys at previous example went crazy and registered all the remaining tld domains for their name. Last time I checked, there were over 100+ tlds ... So instead of writing 100+ blocking rules we change our old rule to this: `evilads\.{2,3}`

Yes, just one rule that will block all their domains. That is basically: "Match all domain names that include `evilads.` and that has at least 2 character (like `de`, `fr`, `jp`, `gh` etc..) but at most 3 character (like `com`, `edu`, `mil` etc..) ending.

One rule. Over 100 matches.

By now Im sure you have realized the power of regular expressions.

Final note before we go to Cookie Control: All Blocker Trackers views and the number of trackers blocked, are *tab specific*. That means, each tab handles it's own tracking blocking, cookie blocking (next chapter) and mixed content blocking (last chapter).

Each tab is like mini browser contained on it's own private window (tab) with private network manager, private cookie control, private encryption control, 6 concurrent connections per tab (current Qtwebkit limitation) but shared disk cache (if enabled).

Cookie Control.

Crunshing Cookies

Network cookies are small text files that are stored on your computer hard drive by your browser each time you visit certain websites. The data they store about you in those cookie files is specific to server you have just connected and you have no control of it. Further, those sites could allow third parties to store even more cookies on your hard drive.

It's important to note that not *all* websites use cookies. There are cookie free sites and not all cookies are bad (although most of them are).

There are basically two uses and two types of cookies:

1. **Persistent cookies** (non-session cookies, tracking cookies)

These are cookies that are used to track you and have very rarely any other use. They are used by companies to track your surfing habits so that they can profile you and send you ads and make money. There are very very few cases where permanent cookies have valid usage.

Cookie usage: Tracking, selling ads and very rarely anything else.

2. **Session cookies**

These cookies have a lifetime only while you are logged into a certain service, like online bank, webmail or any other service that needs authentication. These type of cookies have valid usage. For example: without session cookies you would have to type your username and password each time over and over again when you browsed throught your webmail. Session cookies will be destroyed after you have logged of the service.

Cookie usage: Making the many online services possible to use.

Traditionally, you would have very little control of what cookies to accept.

You could tell browser to either block all cookies (which is not really practical if you want to use Gmail, Yahoo, Facebook or even your online bank), allow all cookies (which would not be very smart, unless of course, you enjoy getting Viagra and diet ads into your mail) or something middle between.

We are interested of this middle ground.

With some browsers, you could further restrict cookies by blocking 3rd party cookies and maybe even allowing only session cookies. CyberDragon Browser goes even further with this protection. By default it will allow only cookies that are:

1. Secure.

Cookies that have **Secure** attribute set will be only sent through encrypted HTTPS connection. This will make hijacking your cookies with packet sniffers much much harder.

Sites that use authentication (like online banks) usually use Secure attribute with their cookies.

2. Safe.

Cookies that have **HttpOnly** attribute set will not be possible to access and manipulate with JavaScript scripting language. This restriction will mitigate (but not completely eliminate) XSS- (cross-site-scripting) attacks. Sadly few sites are setting their cookies with HttpOnly attribute.

Note that the name is a little bit of misnomer: Cookies that have HttpOnly attribute set *can* be sent through HTTPS connections too, not just HTTP. Secure attribute and HttpOnly attribute do not exclude each other. HttpOnly attribute just means that don't allow any scripts to access cookies and Secure attribute means that only allow sending of cookies through HTTPS. Yeah, it's a bad name but I did not invent it.

3. Session cookies.

Obviously, because CyberDragon Browser is all about keeping your surfing habits out of advertisers and other groups, we don't want to allow non-session cookies that are permanently stored on your hard drive and then used to track you as long as you use that same computer.

4. Are not 3rd party cookies.

CyberDragon Browser will by default only allow cookies that come from the site you are visiting. Any other third party cookies are stopped.

These are good defaults and in a perfect world these would be all that is needed to surf without worrying about advertisers and others while still using your favorite online service without a glitch.

Unfortunately, we don't live in a perfect world and not everyone sets their cookies correctly.

That's why we have to sometimes make an exception to this global cookie policy for specific cookies. Next: custom cookie rules!

Custom cookie rules will allow you to make an exception to global cookie policy you saw previously. This way you can make an online service that is ... ahem... broken less broken.

What you are basically telling CyberDragon is that: "Hey, if you see this cookie do this and just skip the global cookie settings, okay?". And CyberDragon will comply, either accepting or blocking cookie, depending what you set it to do.

These custom cookie rules are matched based on three criterias:

- First match the domain (or subdomain) of the cookie against the custom cookie rule.
- If the first test passed then match the path of the cookie against the custom cookie rule.
- Thirdly, if the second test passed too then match the cookie name-value pair against custom cookie rule.
- Lastly, if the third and final test was passed then proceed with the action user had set on this specific custom cookie rule, either by allowing it or blocking it.

And that's basically it. All the three fields: domain, path and name-value pair are regular expressions (take a look at previous chapter if you have already forgotted what they are) and with them you can make very powerfull cookie rules.

Important Note: The order of the rules is important!

For example: If you want to allow all cookies from yahoo.com *except* ads.yahoo.com then you *must* put the rule that blocks ads.yahoo.com first, before rule that allows yahoo.com.

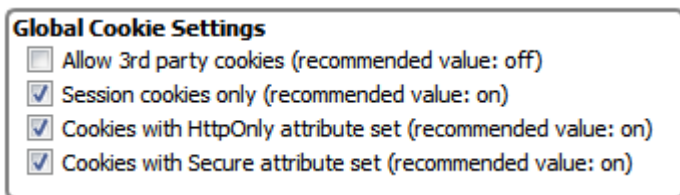
The reason is that the custom cookie rule checking will stop checking immediately after it has met first rule that matches. Also note that there is no need to set any cookie attributes like Secure or HttpOnly on custom cookie rules. We are only interested of the domain (or subdomain), path and name-value pair of cookie and act accordingly.

Next: Cookie Control tab explained in detail.

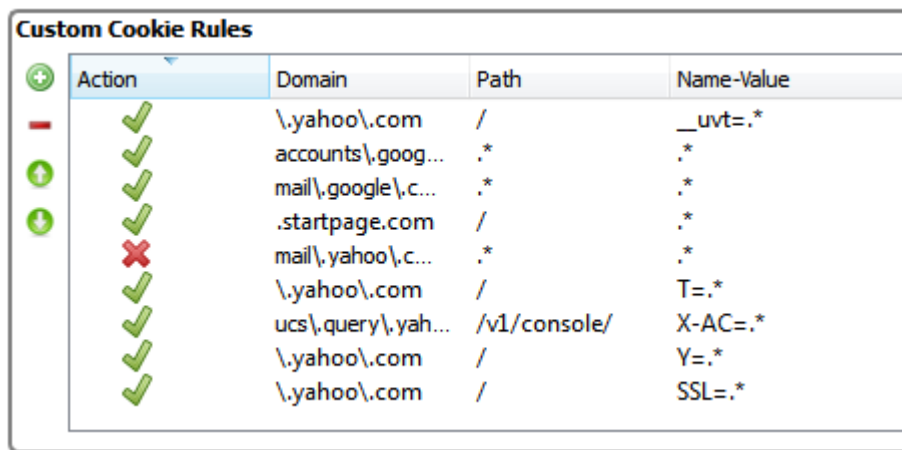
This is where all the magic happens.



First you have global cookie settings. As you can see the settings are pretty self-explanatory.



After that you have custom cookie rules view where you can define your own cookie rules to either allow or block specific cookie(s).



The buttons at the left side of custom cookie views are, from top to bottom: add custom cookie rule, remove custom cookie rule, move custom cookie rule up, move custom cookie rule down. These options are also available through context menu that you get when you right click on custom cookie rule view.

Allowing/Blocking cookie

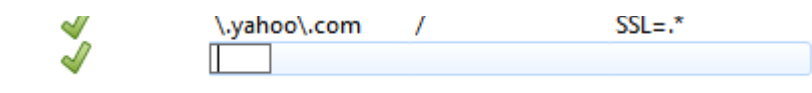
Pretty simple. Just click the icon on the left to either allow or block.



Add custom cookie rule



When you select add custom cookie rule (by either pushing the button or selecting it from context menu) a new empty field will appear at the bottom of the custom cookie view. Double-clicking on the field will allow you to edit it.



There you must add the three required values: domain/subdomain, path (usually just '/' or maybe even '.*' if you want to be very lax) and name-value pair. It is recommended that you fill this last part as *name=.** at first, unless there is an specific need for exact match.

For example: If the name-value pair looks like somekind of user ID, like *SID=235353534*, then it should be writed like this *SID=.** because it is very unlikely that the value will be same again the next time. Next time you try to log the same server it could send you cookie with name-value pair as *SID=676868767* and it would not match the rule and. So always *name=.**

So look carefully what the server sends you from the Cookie List view and construct your rules accordingly. After you are happy with the rule click it's Allow/Block icon and you are ready! However, if you have several rules for the same domain/subdomain where some rules block and some rules allow cookies then remember to check the order of the rules (like I told previously).

This was The Hard Manual Way™, of adding completely new cookie rule. There is an better, easier way but we don't get there just yet. Keep reading ...

Remove custom cookie rule



This is really simple! Just click the rule (or rules, you can select several rules with pressing Shift or Ctrl down with one hand and clicking with left mouse button on other. Also, Ctrl + A will select all rules if that's what you want) and press remove custom cookie rules button (or select it from context menu that will pop up when you right mouse click custom cookie view)

Move custom cookie rule up



Also simplicity in itself. Select rule (can only move one rule at the time currently) and push Move custom cookie rule up or select it from menu.


Move custom cookie rule down



Same as above but obviously to another direction ...

Cookie List

And lastly there is an live view of cookies that the server/site you are connected with tried to ram through your throat.

 ☒ Clear Cookie List on page load

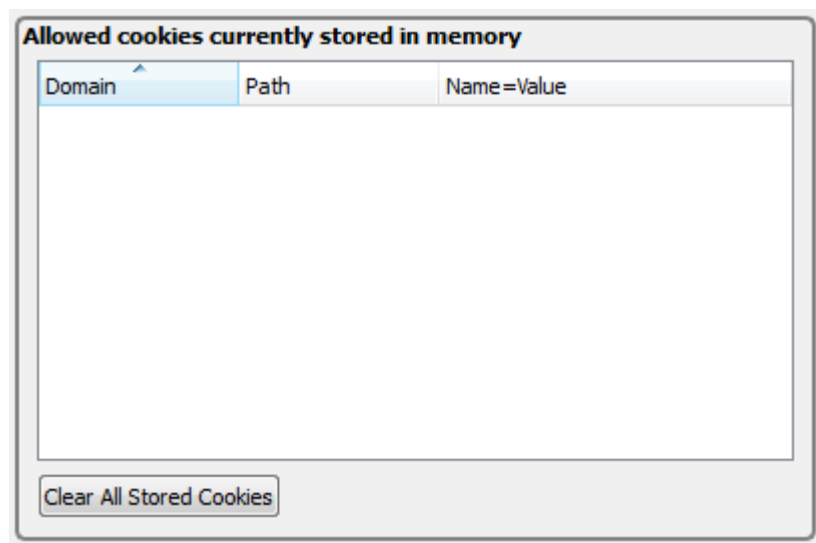
Time	Action	Domain	Path	Name
la 7. joulu 2...	✖ Blocked	.huffingtonpost...	/	snn_r
la 7. joulu 2...	✖ Blocked	.huffingtonpost...	/	chec
la 7. joulu 2...	✖ Blocked	.huffingtonpost...	/	sailth
la 7. joulu 2...	✖ Blocked	.google.com	/	NID=
la 7. joulu 2...	✖ Blocked	.twitter.com	/	guest
la 7. joulu 2...	✖ Blocked	.twitter.com	/	guest
la 7. joulu 2...	✖ Blocked	.google.com	/	NID=
la 7. joulu 2...	✖ Blocked	.twitter.com	/	guest
la 7. ioulu 2...	✖ Blocked	.aoogle.com	/	NID=

What you see is the very live cookies that tried to sneak into your computer while you visited some site. For each cookie the following info is told: Time when it tried to invade your privacy, action CyberDragon took depending of either global cookie settings or custom cookie rules (custom cookie rules are always checked first, before global settings!), the domain/subdomain that the cookie belongs to, cookie path, cookie name-value pair, cookies expiration time (if it has time then it's permanent cookie, if it's empty then it's session cookie), if the cookie has Secure attribute set and if the cookie has HttpOnly attribute set.

This view is automatically sorted based on time of cookie arrival but you can sort it anyway you like (just click the header for your sorting criteria). Also there are three controls: Move cookie rules up, Clear cookies on each page load and Clear cookie list.

Clearing cookies already stored in memory

Althought CyberDragon will keep allowed cookies only in memory and they will all be cleared at exit, you might still sometimes want to clear cookies immediately. If you scroll down the Cookie tab a little bit you will see the following:



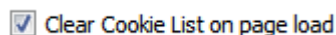
This will show all the currently allowed and stored cookies (if any) for *currently open tab*. To clear them just click "Clear All Stored Cookies".

Move cookie rules up



This is the easy way how to add new custom cookie rules that I told you before. Just select the cookie (or cookies with Shift + left mouse or Ctrl + left mouse) that you want to add to custom cookie rule view and press Move up button (or select it from context menu with right click on Cookie List view). Your new cookie rules appear on custom cookie view where you can allow/block, edit, remove and move up/down them just like I previously told you. This way you don't have to do tiresome typing of all those rules and can only concentrate of allowing/blocking and possibly, editing/fine tuning the rules.

Clear Cookie List on page load



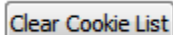
What this little checkbox basically does is that it tells CyberDragon to clear the Cookie List view everytime the URL changes (that is, each time there is an new page load). Without this option your Cookie List view would gather tons of cookies over time while you are surfing. So it is recommended that you keep this enabled always *except* in one case, logging the very first time to some online service that you have not visited before with CyberDragon.

The reason is simple: Most online services redirect user from login page to some other page where the actual authentication process is done. There that page quickly checks what cookies your browser send to it and then further send you to either to your proper place or rudely slams error page telling you that you have not enabled cookies. So if you have *Clear cookies on each page load* checked you have about few milliseconds time to see the cookies that were blocked by the actual authentication page !!! (remember, this option clears cookie list view with *each* page load, not just user typing www-address and pressing Go! button or clicking some hyperlink ...)

So for this reason, for very first time, you should *disable Clear cookies on each page load* option so that you can actually see what cookies were tried to send to you, in time order, and blocked. That's the only way you can see them and add them to custom cookie rules view (like previously told).

After you have added the right cookierules and set them to allow and confirmed that you can login successfully you can then check *Clear cookies on each page load* option back again.

Clear cookie list



Over the time, *if* you have *Clear cookies on each page load* unchecked, your cookie list view will gather tons and tons of blocked (and maybe few allowed, depending of your settings) cookies. To clear this list you can push this button.

Also remember, this will only clear cookie *list* view for the *currently open tab*, not for any other open tabs. *It will also not clear any cookies from memory that have already been previously allowed and set.*

There is also not yet any "Clear all cookies" button but CyberDragon can manage just fine without it because no cookies, session or non-session, are ever stored permanently when CyberDragon exits. So it will always be cookie clean at next startup. This is a feature not a bug.

This ends the cookies chapter. Next stop: Proxies.

Proxy.

Hiding your tracks

Most of the data in the Internet is connected with IP-protocol. There are other protocols (like ICMP-protocol which the common ping program uses) that work at the same level as IP-protocol. And there are protocols (like ARP-protocol) that work below IP-protocol level.

But basically it's always IP-protocol.

Be it HTTP, DNS, SMTP or whatever, always it is IPv4 (or sometimes IPv6) protocol at the bottom of the stack that tells computer: A) where the data came from and B) where it is going.

The actual delivery work of the data is left to protocols above IP-protocol. Most often used is TCP-protocol (which the HTTP-protocol uses) and UDP-protocol (which the DNS-protocol uses).

But we are not interested of any of those other protocols. We are interested only of IP-protocol because that contains the information that is most important to us. Namely, the source address. Because communication in Internet is bi-directional, all data contains the source IP-address so that the receiver of that data can *track you* and send reply back.

Of course, that source IP-address can be forged. But then you would not get answer back from the server you contacted. Of course, if you have a bad intention of doing DoS-attack then that inability to receive answer does not matter anyway ...

So, what do you do if you want to receive answer from server you are contacting but do not want to reveal it your true IP-address ?

Simple. You use proxy.

What is Proxy?

Proxy is any computer in a Internet that acts as an middleman between you and target server. There are freely available, public proxies and there are proxies that are only available to paying customers.

Also many schools, organizations, corporations and even countries use proxies. To restrict and monitor your Internet use and for censorship.

What we are interested of are freely available, public proxies that don't need authentication. The idea is that you select one of those proxies to route your traffic to target server and the target server will only see the proxy IP-address, not your true IP-address.

These kind of proxies are called anonymous proxies and they do not reveal their client true IP-address to target server or even the fact that the server is talking with proxy instead of client machine.

Next I will show all the proxy controls but before that ...



Important Note! When using proxy always keep these three things in mind:

1. **Check if the proxy really is anonymous!** Always check the anonymity level of the proxy *before* you are going to use it! Just because you have found proxies from some free proxy list site does not mean that they are anonymous! Even if they say on their pages that they are anonymous you should still do the checking by hand by setting the proxy/proxies and then visiting some of the many IP-address showing sites that you can found by searching from the Net with keywords: "what is my ip".

Since version 1.6.4 CyberDragon has new, faster proxy checker that gives much more accurate results. It will now also let you see why proxy did not work and also classify the proxy's anonymous level either to "No" (meaning it will reveal your true IP address), "Anonymous" (it will hide your true IP address but web server will know that you are using proxy) or "High-anonymous" (will hide your true IP and the server you are connecting to does not know you are using proxy)

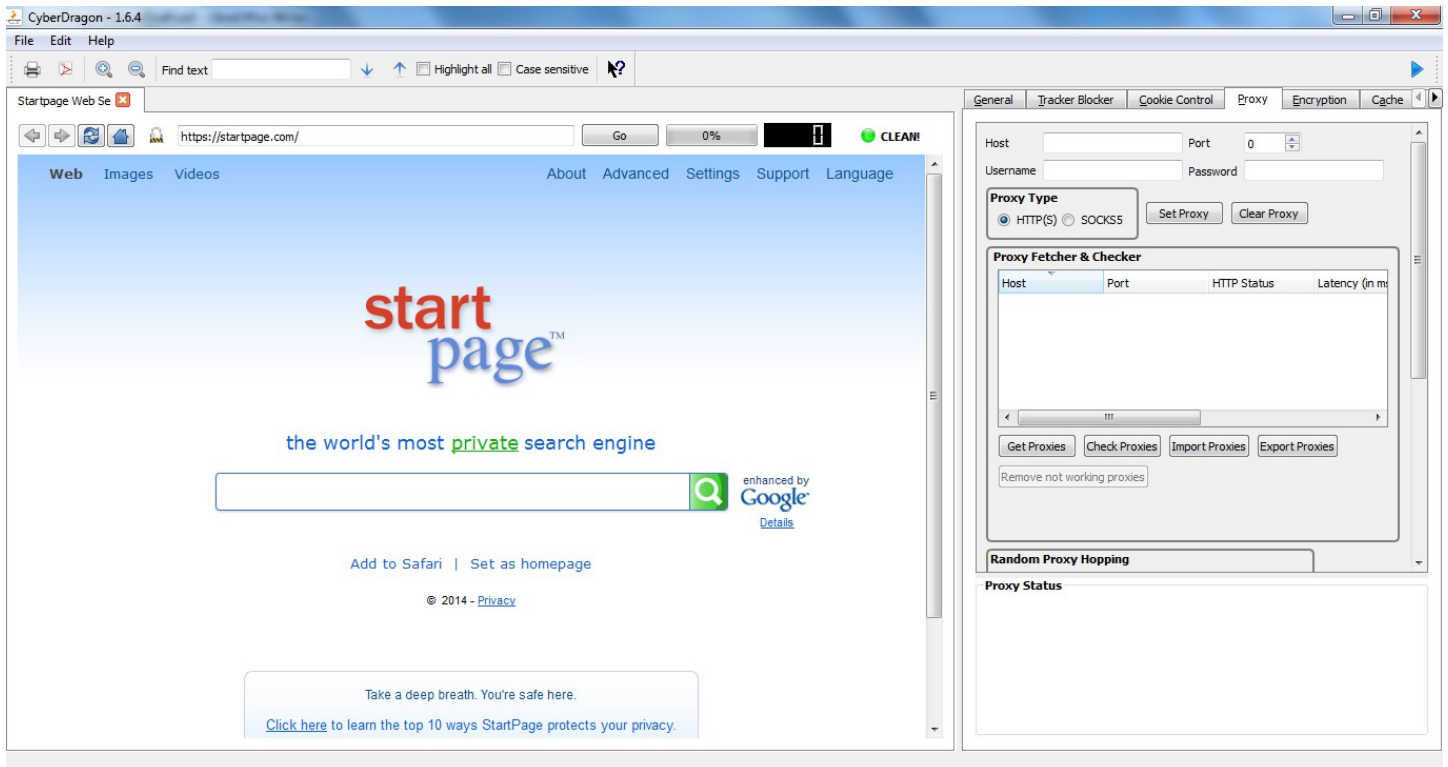
2. **Always use proxy that supports encryption!** Don't ever use proxy that does not support HTTPS (HTTP protocol over the top of SSL/TLS)! If the proxy you are using does not support HTTPS then you can't access any https:// site. Worse, anybody can see (even the proxy owner!) what you do while surfing. So always use proxy with SSL¹ support!

¹ Strictly technically speaking, HTTPS, SSL and TLS are all different things. HTTPS is just HTTP-protocol layered on top of SSL- or TLS-protocol. SSL is older encryption protocol that is luckily being replaced by newer, TLS encryption protocol. But *within the context of this manual* they all mean the same thing: encryption. So try not to get confused.

3. And lastly, never, ever, use proxy with online bank, webmail or any service that requires authentication. Not even if you are using proxy with HTTPS support. Just use direct connection and forget proxy when using those services. There is always possibility that the proxy owner could steal your username & password. This is not a theoretical threat. It can be done, and has been done.

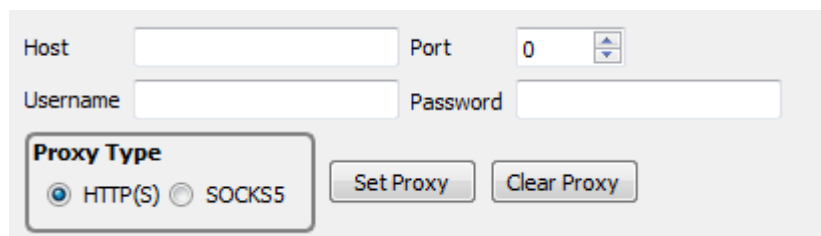
So now that is clear let's start:

Go to Proxy tab by pressing Alt + P. It should look something like this.



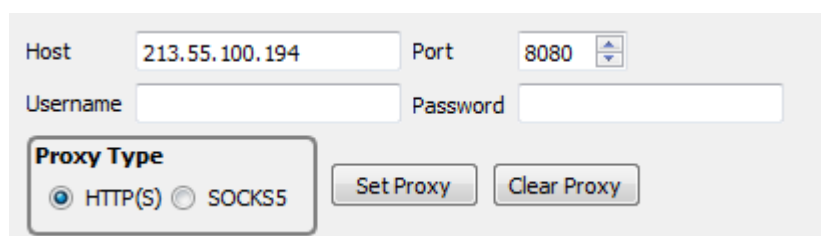
Manually setting proxy

Although CyberDragon has lot's of cool features like proxy fetcher, proxy checker and random proxy hopping, the very first thing you should be familiar with is how to set proxy manually by hand.



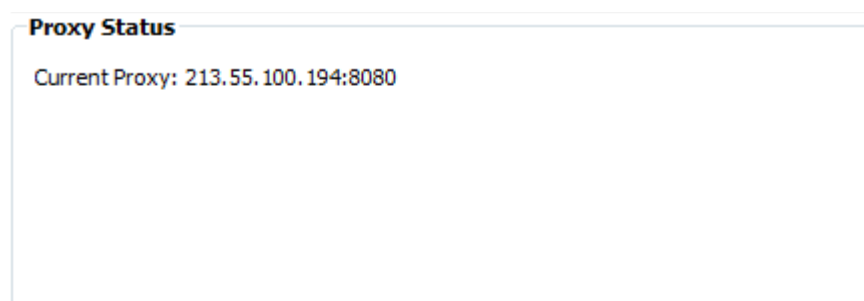
What you see here from left to right and top to bottom is: field for proxy address, proxy port number, field for username (if any), field for password(if any), selection of proxy type, set proxy button and clear proxy button.

Let's set manually some proxy that is listed at <http://hidemyass.com/proxy-list/>

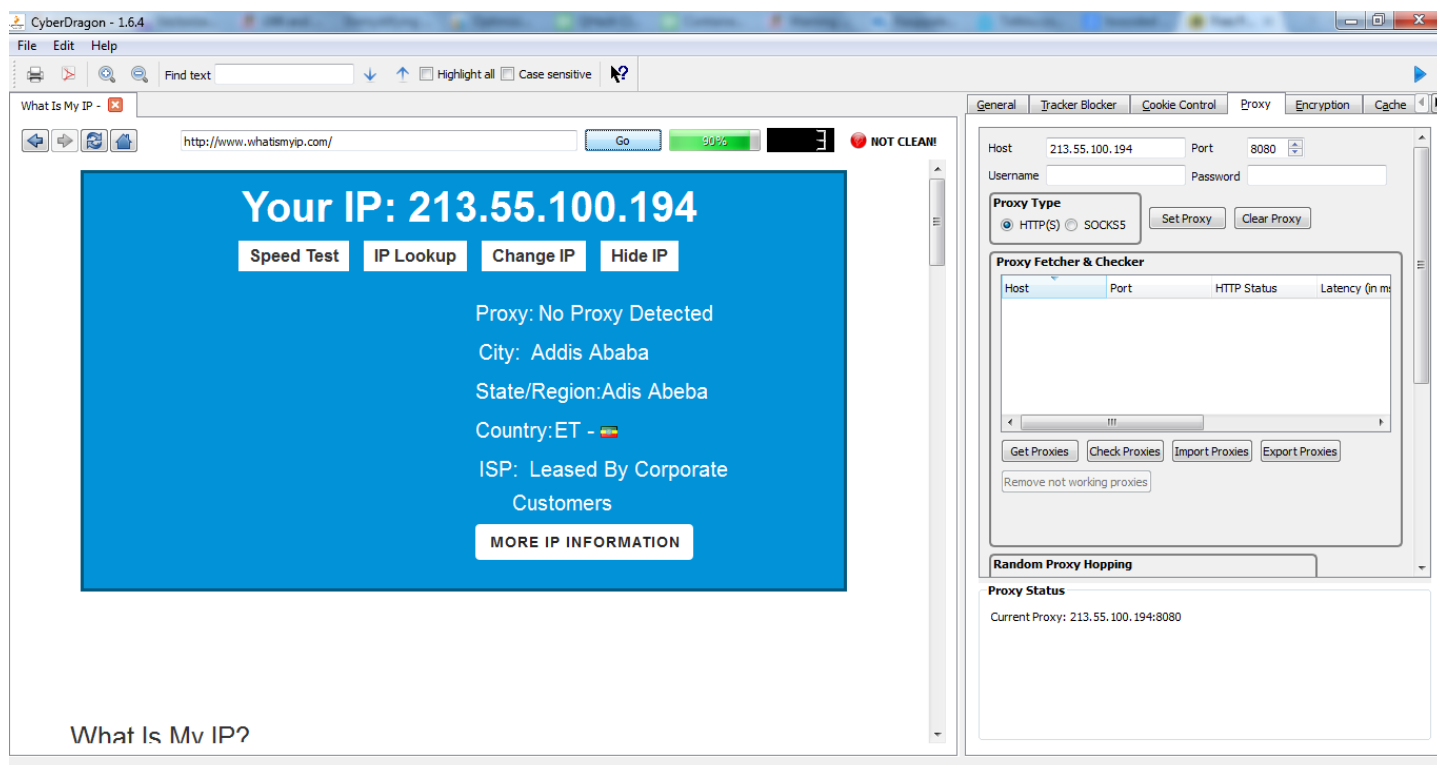


Most of the time you are going to use HTTP(S) proxies and very rarely SOCKS5 proxies (see Appendix B at the end of this manual for how to set Tor SOCKS5 proxy) so you can leave the proxy type as it is. Press Set Proxy button.

Now you see that the Status at the bottom right corner box changed to 213.55.100.194:8080. That means that all the following connections will go through this proxy.



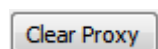
Let's check if this proxy really is anonymous like that Hide My Ass page told us. Go to address <http://www.whatismyip.com>



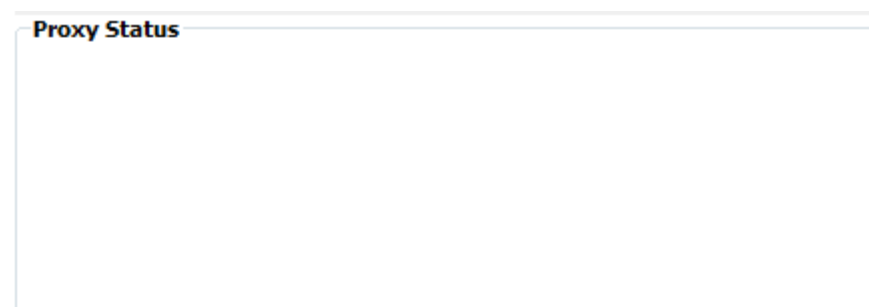
Okay, we got lucky this time. This proxy really *is* anonymous. The page shows proxy IP-address and not our true IP-address. Even better, this test page could not even detect that we are using proxy but thinks that we are directly connected to it. So this current proxy is not just Anonymous proxy, its High-anonymous!

So now you know how to set & test individual proxies. That is all fine and well but it is also very tedious way of getting good, working proxies. Luckily CyberDragon offers a better way and Im going to show you that next.

But before that, clear the proxy you don't need anymore by pressing Clear Proxy button.

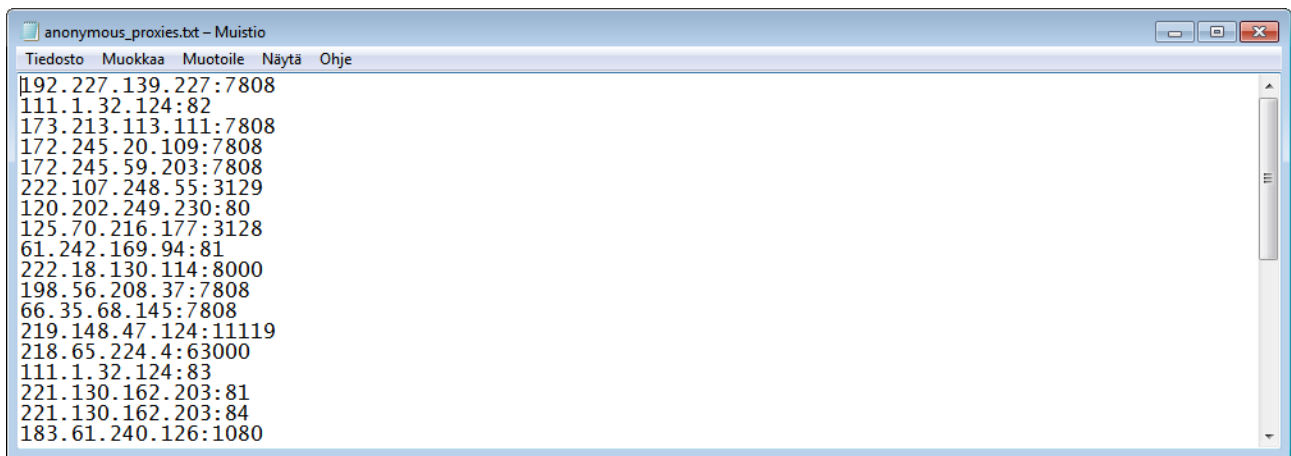


You see that the proxy IP-address & port are cleared and the bottom right box shows no proxy anymore.



Proxy Importing

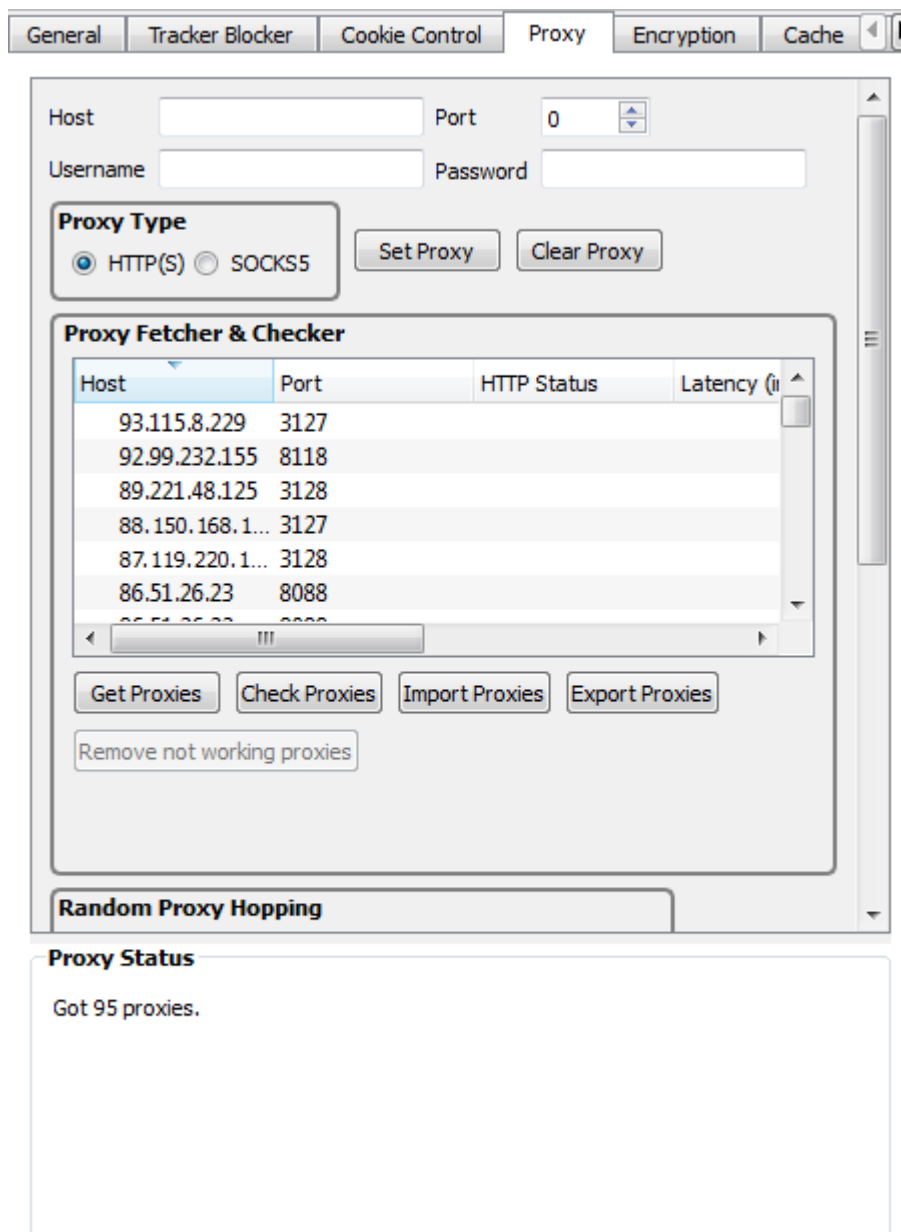
Instead of copying each proxy IP-address and port number from web pages there is an marginally better way. You can tell CyberDragon to load a proxy list from a text file. The format of this text file is trivial: Each line contains proxy ip address and port number in ip:port format like below.



After you have your proxy list file ready you can import it to CyberDragon by pressing Import Proxies button.

Import Proxies

After you have pushed this button the CyberDragon will load your proxies for you so that you can easily select them from the list. It will also tell you how many proxies were important at the Status box.



Let's select one proxy from that list.

The screenshot shows a web-based interface for managing proxies. At the top, there are input fields for 'Host' (89.221.48.125) and 'Port' (3128), along with 'Username' and 'Password' fields. Below these is a 'Proxy Type' section with radio buttons for 'HTTP(S)' (selected) and 'SOCKS5'. There are 'Set Proxy' and 'Clear Proxy' buttons. The main section is titled 'Proxy Fetcher & Checker' and contains a table of proxies. The table has columns for 'Host', 'Port', 'HTTP Status', and 'Latency (ms)'. The proxy 89.221.48.125:3128 is selected. Below the table are buttons for 'Get Proxies', 'Check Proxies', 'Import Proxies', 'Export Proxies', and 'Remove not working proxies'. At the bottom, there is a 'Random Proxy Hopping' section and a 'Proxy Status' section showing the current proxy as 89.221.48.125:3128.

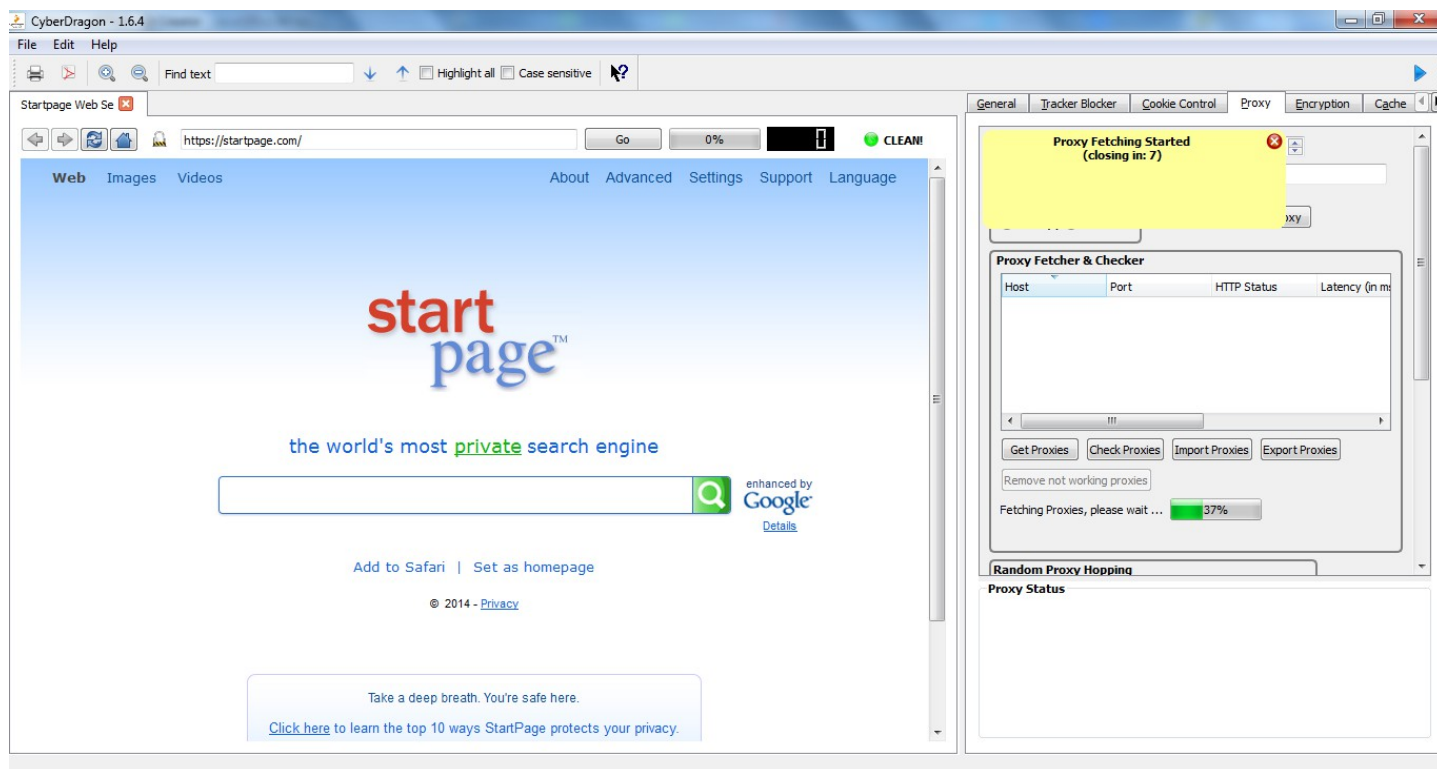
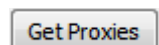
Host	Port	HTTP Status	Latency (ms)
93.115.8.229	3127		
92.99.232.155	8118		
89.221.48.125	3128		
88.150.168.1...	3127		
87.119.220.1...	3128		
86.51.26.23	8088		
86.51.26.23	8088		

As you can see, when you clicked that particular proxy (89.221.48.125) it was immediately set and there was no need to push Set Proxy button. Set Proxy button is only needed when you want to set your proxy *manually*, by hand and it is not listed in the proxy list. You can select any proxy from that list by just clicking it and when you are finished you just click Clear Proxy button. This couldn't be any easier.

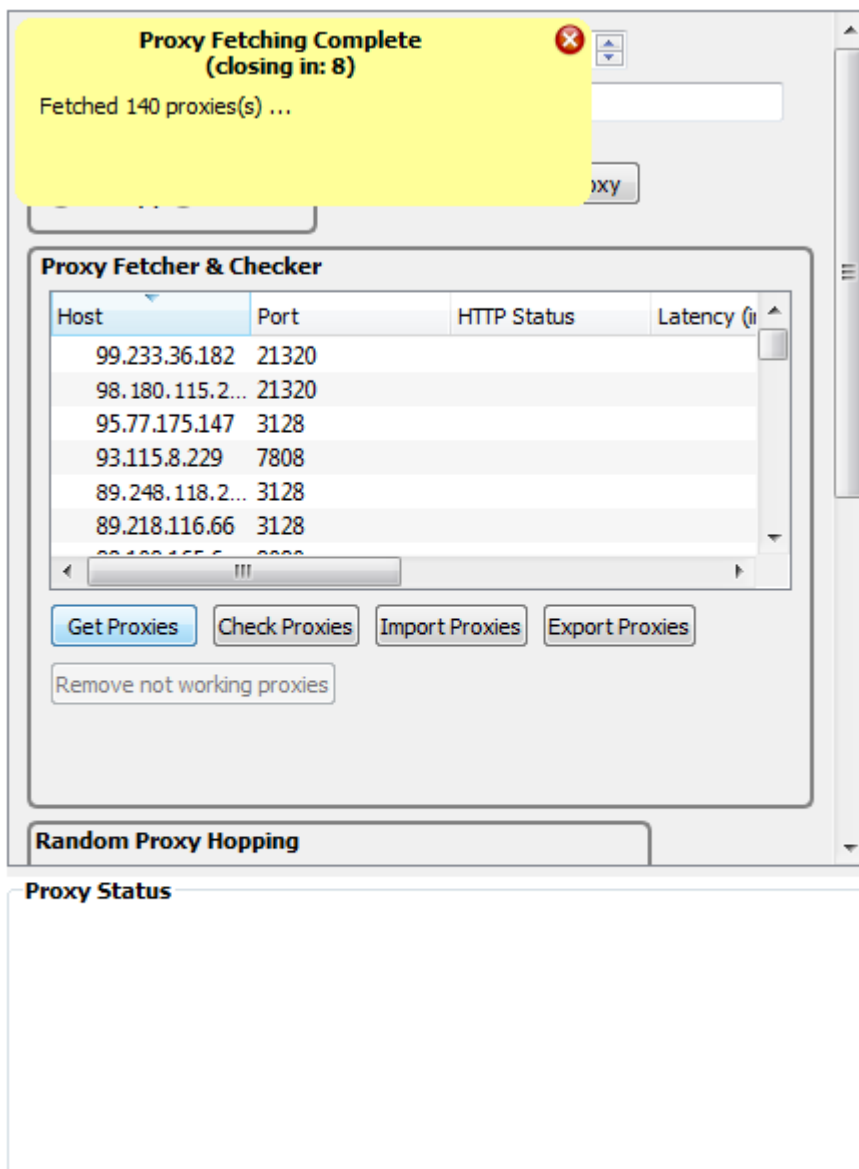
Proxy Fetching

But what if you don't have premade proxylist at hand and still want some? Well, you could use the short proxy list file called `anonymous_proxies.txt` that is included with CyberDragon. Or, you fetch "few" proxies from net. Let's fetch few proxies from net now!

Push Get Proxies button.



As you can see (from the Notification at the top right corner and Status box at the bottom) CyberDragon started proxy fetching from Net. It also provides a nice progressbar for you.



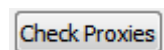
140 proxies, not bad.

So when you need proxy list you can A) import proxies from file B) fetch proxies from net or C) import *and* fetch (or fetch *and* import) proxies :-)

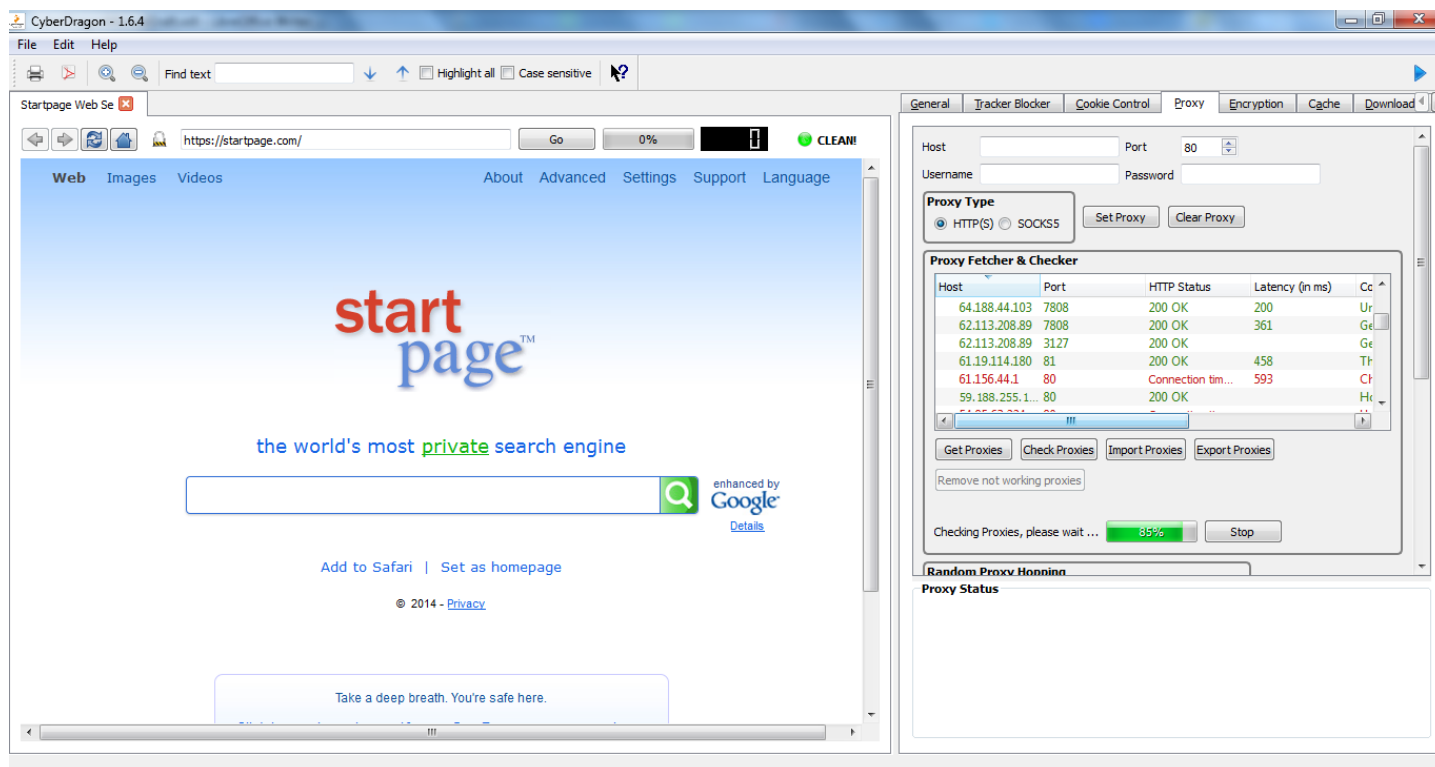
Anyway you like it. And no matter how you got your list CyberDragon will keep it nice and compact by removing any duplicate proxies :-)

Proxy Checking

So now you know how to get your proxy list. But we don't know anything about these proxies or even if they are online and working at all! So let's check them next. Press Check Proxies button.



After you have pushed Check Proxies button the proxy list will temporarily clear itself and then start listing proxies and their current status.



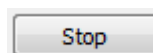
You can see the current status of proxy checking from the progressbar. And as you can see, there is now some new information about proxies too, like if there were any connection problems (HTTP status), latency, country the proxy belongs to, SSL support and if the proxy is Anonymous.

The lower the latency number the more faster that proxy will handle your page request. If the latency field is empty then that means that that particular proxy was blocking our ping request.

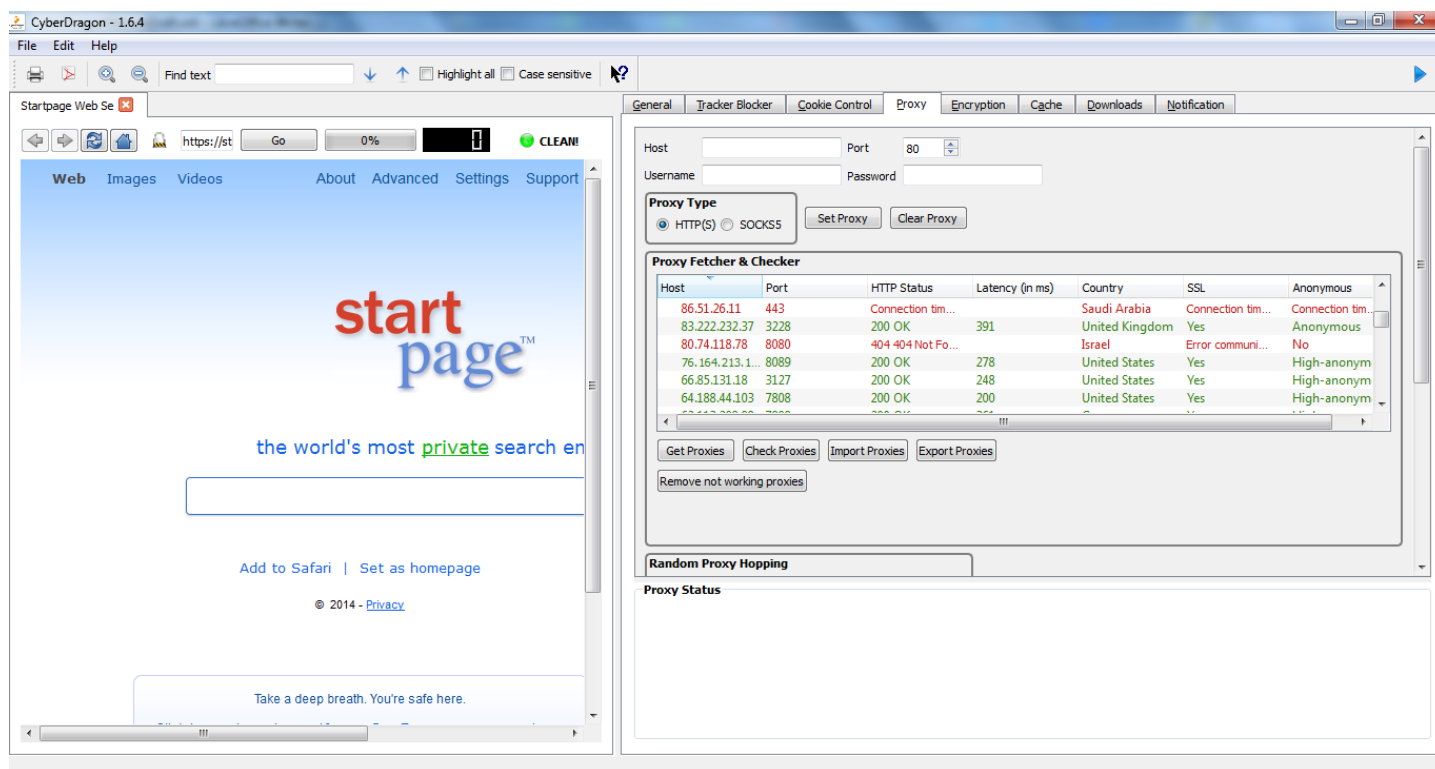
If the SSL field says "Yes" then it means that this proxy supports HTTPS. If it says "No" then it does not support SSL.

If the Anonymous field says "No" then that proxy is not anonymous and will reveal your true IP address. If it says "Anonymous" then it hides your true IP address *but* the server you are connecting will still know that you are using proxy and might block your connection. If it says "High-anonymous" then that proxy hides your IP address and the target server has no way of knowing if you are using proxy or not.

If the checking takes too long you can stop proxy checking anytime by pressing Stop button.



This is what it will show after it has finished checking (view resized to make all the fields shown).



If the proxies had any issues (like being offline, denying connection, blocking connection, not supporting SSL, or not being anonymous) then it will be listed as red. Otherwise it will be listed as green and everything should be ok.

Sometimes you might get strange results, like HTTP Status showing "200 OK" but either SSL or Anonymous field showing some error.

This will usually mean that the proxy in question is blocking one of the test pages that proxy checker uses to determine proxy working. This is very common in China and middle-east.

CyberDragon will use http://en.wikipedia.org/wiki/Main_Page for testing HTTP Status, https://en.wikipedia.org/wiki/Main_Page for testing SSL status and <http://www.goldenpirates.org/proxy/azenv.php> to determine Anonymous status.

I will add in 1.6.5 an possibility to override these three target test pages for your own liking.

Remember: For maximum privacy, always, if possible, select High-anonymous proxies from the list!

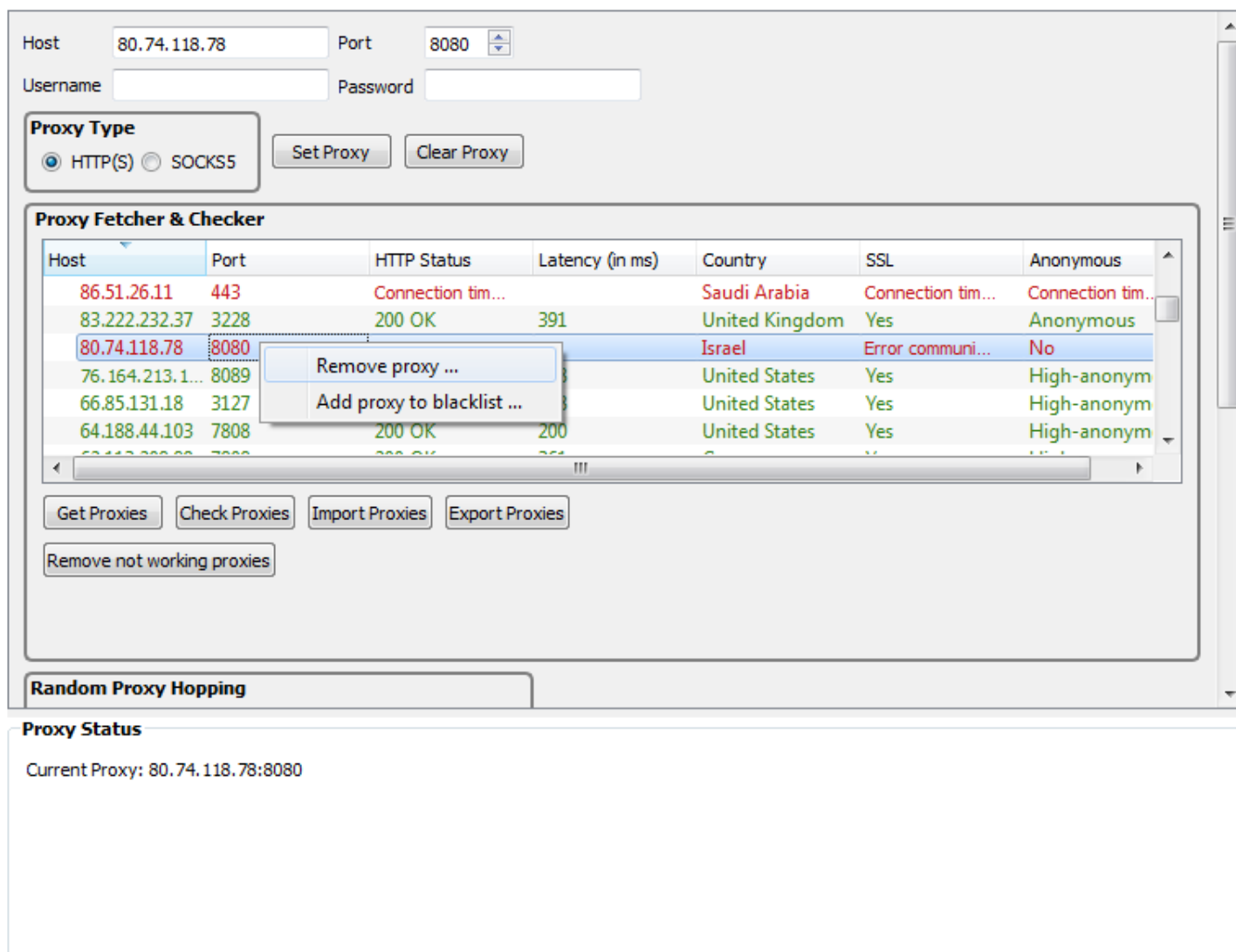
Anonymous proxies are fine too but then the other side will know that you are using proxy even though they might not know your true IP address.

One last note about Proxy fetcher & checker: Proxy checking & fetching work currently only with HTTP(S) proxies. SOCKS5 proxies work right now only when set manually.

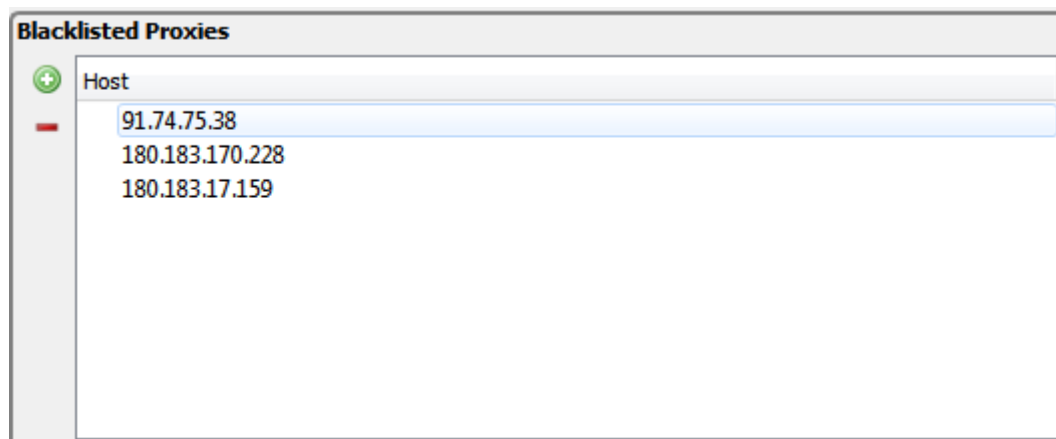
Removing proxies & blacklisting proxies

After you have checked your proxy list you can decide which proxies to keep and which proxies to remove or even blacklist.


To remove proxy from the list, right click over it and select Remove proxy from the context menu.




To blacklist a proxy, right click over it and select Add proxy to blacklist.



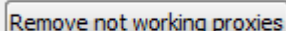
All the blacklisted proxies will have their IP-address shown here and they will never appear to your proxy list when you import or fetch new proxies.

You can also manually add proxies to blacklist by pressing Add proxy to blacklist button 

Note that you only need to add blacklisted proxy IP-address. No need to add it's port number.

If you want to remove blacklisted proxy/proxies then select them from list and press Remove proxy from blacklist button 

Removing not working proxies



This is a new addition in 1.6.4. Because it can get very tedious to remove and/or blacklist proxies one-by-one from that list I have included this button as an quick shortcut that will immediately remove all the bad (red) proxies from the list and leave only good ones (green) left.

So now you know how to get your proxy list, how to check it and how to remove bad proxies and make sure that only best proxies are on that list (proxies that have low latency, support SSL, are anonymous and preferable are located in some free country).

The last thing to show you is how to export those good proxies to external proxy list text file so that you don't have to go over the whole checking process again next time you start CyberDragon but can just import your super-duper-proxy list.

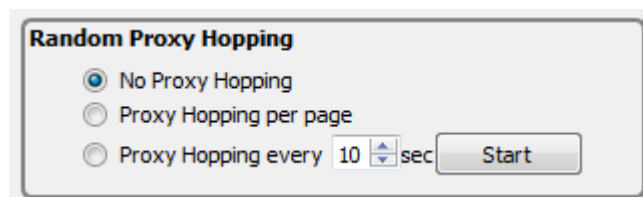
If you want to export all current proxies then press Export Proxies button. 

If you want to export just some proxies then select them and again press Export Proxies button.

One final thing before ending proxy chapter: Random proxy hopping.

What is Random Proxy Hopping?

Random proxy hopping is the idea that your current proxy is not fixed all the time. That is, your connections go through random proxy either by each time you visit some page (Proxy hopping with each page) or after timeout has been reached (Proxy hopping once every ...)



Random Proxy Hopping

☒ No Proxy Hopping

☐ Proxy Hopping per page

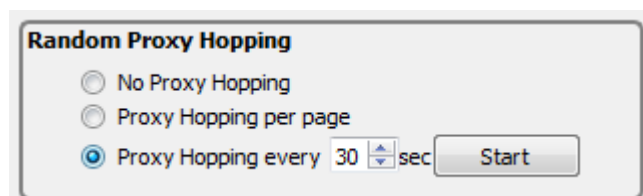
☐ Proxy Hopping every 10 sec

By default, the Random Proxy Hopping is disabled and you can't use it before you have imported or fetched some proxies first because obviously, it needs some proxies to work with.

You can always disable random proxy hopping by selecting "No proxy hopping" option.

Let's check the other two options.

If you select "Proxy hopping once every" option then you need to specify the timeout value that will be waited between proxy switches and then press Start button. It will wait for the timeout value that you specified (30 seconds in here) and after that timeout has passed it will randomly select (hop) some proxy from your list after each 30 seconds.



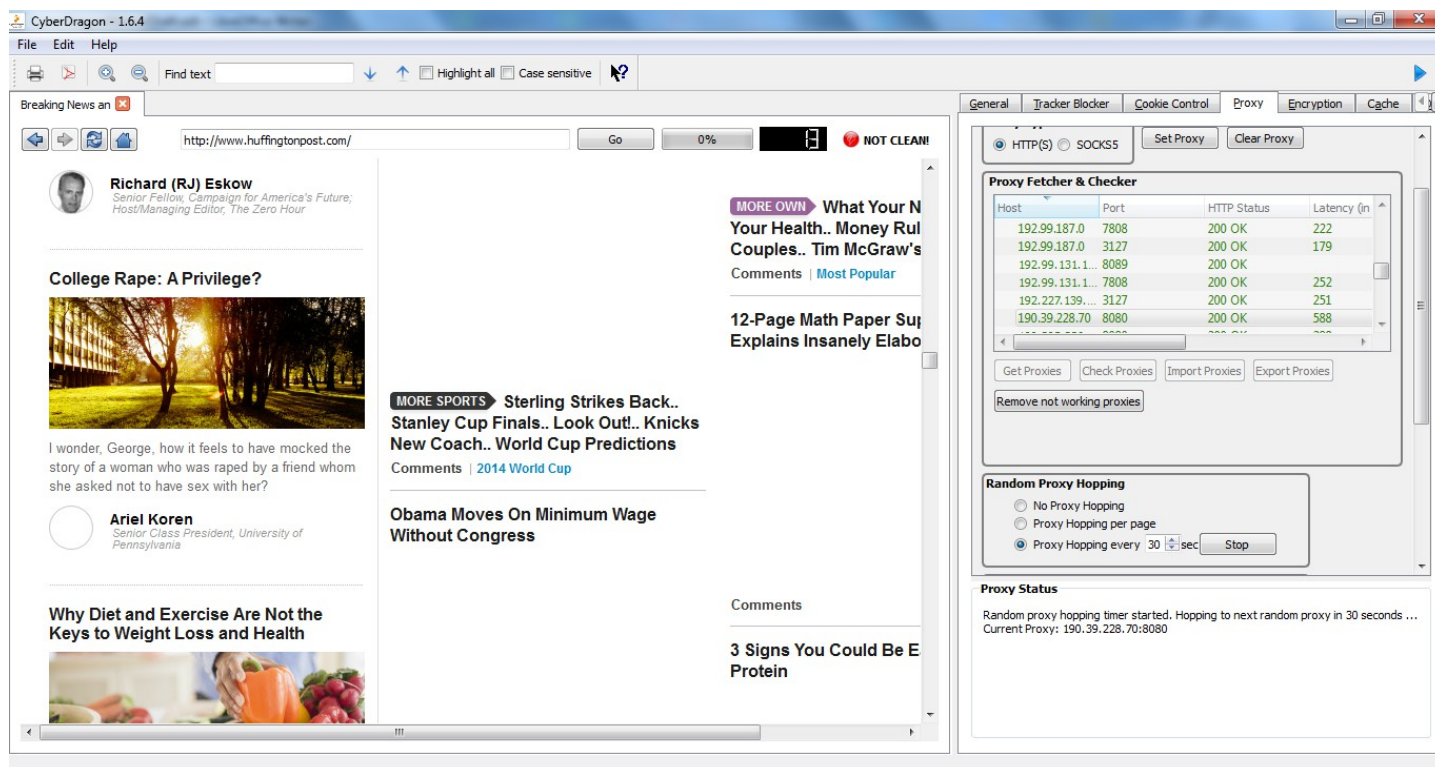
Random Proxy Hopping

☐ No Proxy Hopping

☐ Proxy Hopping per page

☒ Proxy Hopping every 30 sec

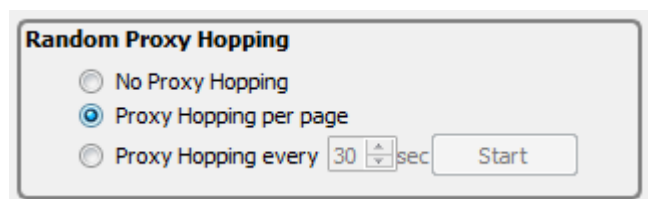
Random proxy hopping with timeout at working.



As you can see it will show that random proxy hopping is active, the timeout that will be waited before next hop and the currently selected proxy. You can now continue surfing normally and each time 30 seconds has passed your proxy will change. After you have finished using it just press Stop button or select "No proxy hopping" option.

While the timed random proxy hopping is cool there is always the possibility that some of your proxies on your list has gone offline between proxy switches and for your bad luck it will select it. In that case there is no other option but to Stop/Start it again.

If you want little bit more control over your proxy hopping in problem cases then you should try Proxy hopping with each page.



When this option is selected your new proxy will be randomly selected from the list only when you click a link or refresh current page. So incase that you happen to get a dead proxy (or proxy that is slow or otherwise works poorly), all you have to do is click some link or press reload button (or F5 key).

As a final note, random proxy hopping works best with a good quality list of proxies (low latency).

Encryption.

Keeping your data safe



Encryption is important. Without it all your data is like postcard, everyone can see it. Like all other browsers CyberDragon supports the standard HTTPS encryption. However, at this time, there is not much control of the browser settings of this HTTPS encryption. Currently the only option is to block mixed content and enabled/disable, reorder some ciphers that are used during HTTPS connections (but I will add some more encryption specific features later).

What is mixed content?

Normally, when you visit encrypted site (like your online bank or let's say <https://startpage.com>) you will see https:// in front of the page address and also that small padlock icon will close (and if you put mouse over it an tooltip will appear after few secs. and say Encrypted).

This will tell you that you have just entered HTTPS encrypted site and you are safe and nobody can see what you are doing.

That's not the whole story though. You see, even though the *site itself is HTTPS encrypted*, there might be some third party content that is delivered to that page via unencrypted HTTP protocol (via http:// links). The content could be images, style sheets, scripts, whatever...

This poses an privacy risk (they can see from what IP-address you loaded they stuff and also send cookies. Or at least try sending cookies ... :-)) and possible security risk (if the target of http:// link is malicious JavaScript for example) for the user visiting such site.

For these reasons CyberDragon will by default block any http:// content that tries to sneak into an https:// protected site.

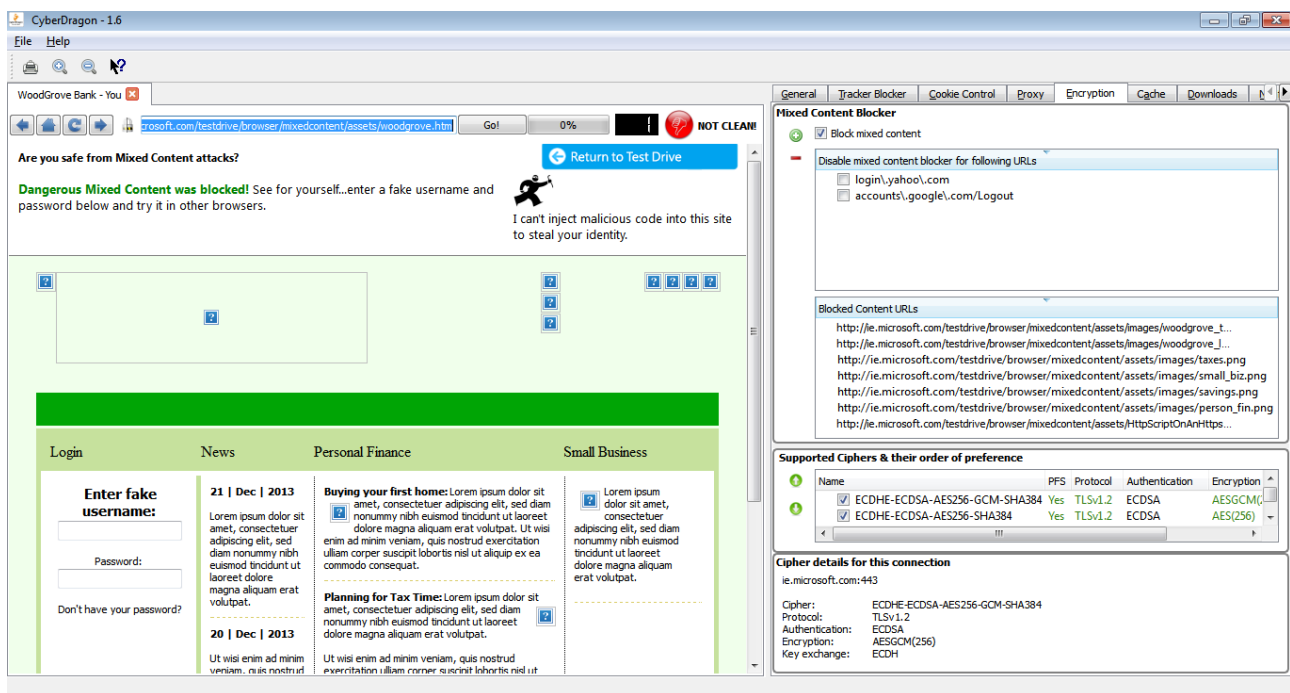
However, sometimes, for a proper working of the site, exceptions must be made.

Some online banks, or e-commerce sites might use unencrypted links for some content that are actually needed. These are often results of accidents or just badly coded webpages where links were absolute links like for example `http://somebank.com/someimportantcontent.html`, instead of relative links like `/someimportantcontent.html`

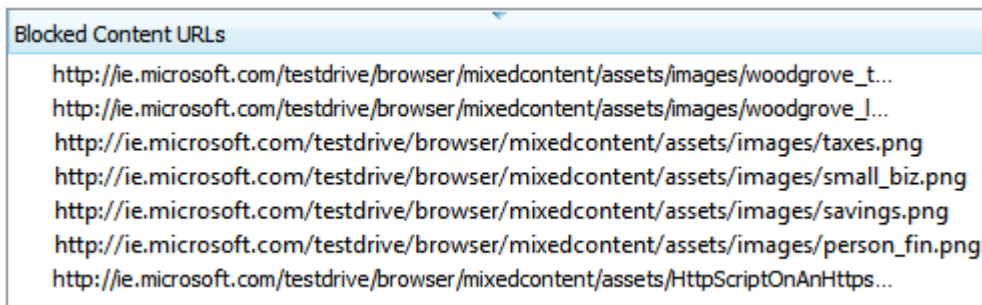
In those cases a rule must be made that will tell CyberDragon: "Hey! Don't use mixed content blocking for this particular domain, subdomain or page.

Let's go to some mixed content test page and see how mixed content blocker works.

<https://ie.microsoft.com/testdrive/browser/mixedcontent/assets/woodgrove.htm>

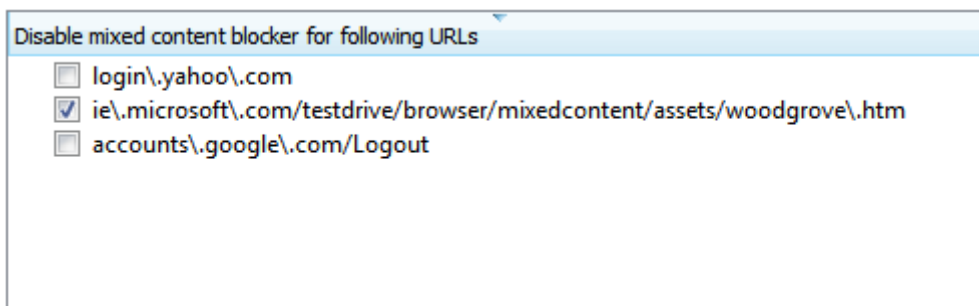


This is the Microsoft Mixed Content Block test page and what you see here is that CyberDragon mixed content blocker stopped the test page from trying to do bad things.

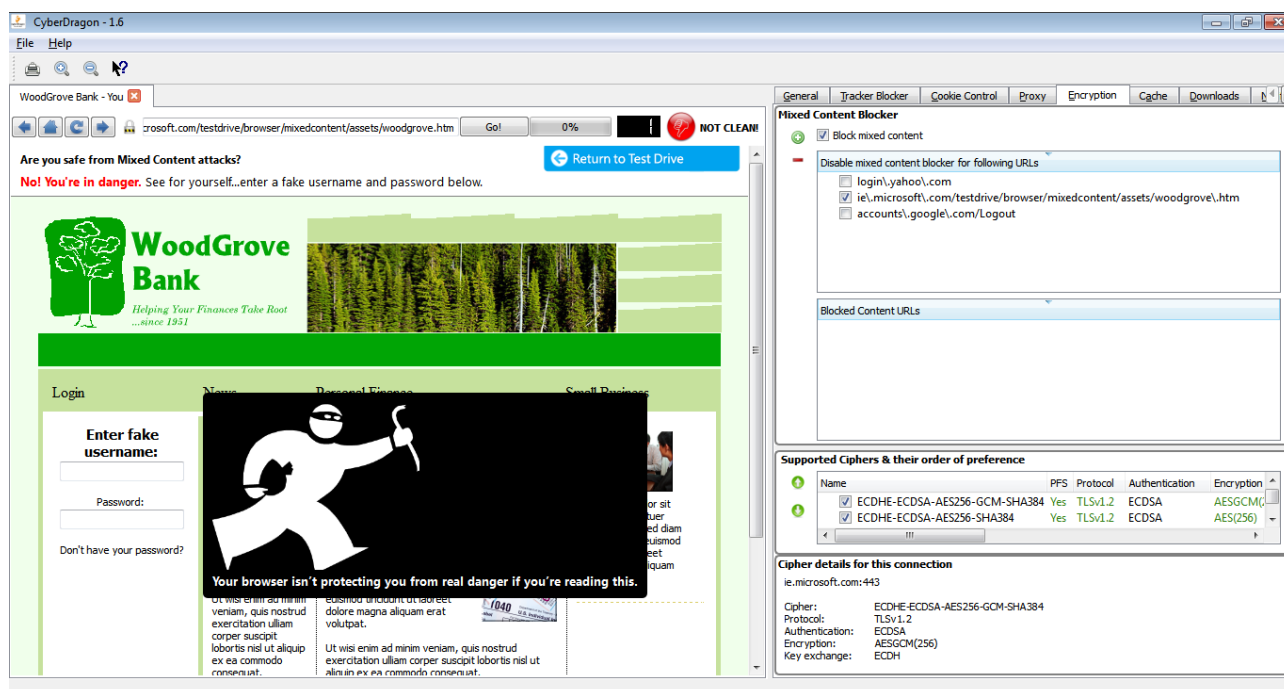


Most of those http:// blocked content were passive content, images. However there is a some blocked active content too, a JavaScript file.

Now let's temporarily make an exception and allow ie.microsoft.com/testdrive/browser/mixedcontent/assets/woodgrove.htm to load mixed content.



And reload page by pressing F5 and let's see what happens.



Yikes! Luckily this was just a test page :-)

However, this shows you that mixed content blocking is not a joke and it should be always enabled and only in very very special cases (like with your online bank) you might need to disallow it for the proper working of their broken pages.

Some banks and other sites are especially bad with mixed content stuff but hopefully they will get their pages fixed.

Adding exception for mixed content blocking



For adding an exception to mixed content blocking just click "Add mixed content URL" button. By default it will add the currently open tab address but you can edit it anyway you like.

Remember: This address must be the problematic https:// site or page that needs to allow unencrypted content. Not the blocked content address itself! Also, you can leave the https:// out of the rule name.

You can also temporarily disable/enabled the rule whenever you wish by clicking the checkbox in front of the

rule.

Removing mixed content URLs

For removing the rule, select it from the list and press "Remove mixed content URL".



As with Tracker Blocker, and Cookie Control, the Encryption Blocked Mixed Content URLs view is specific to that particular open tab.

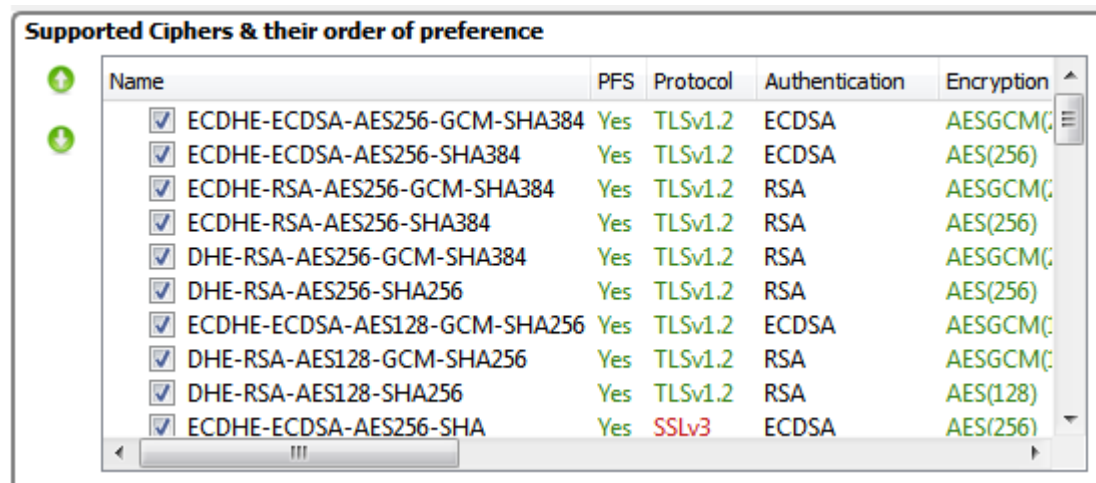
Enabling/Disabling, reordering ciphers used during HTTPS

The screenshot shows the CyberDragon 1.6 browser interface. The main window displays the Google homepage. The settings panel on the right is open to the 'Encryption' tab. The 'Mixed Content Blocker' section shows a list of URLs to block mixed content, including 'login.yahoo.com', 'ie.microsoft.com/testdrive/browser/mixedcontent/assets/woodgrove.htm', and 'accounts.google.com/Logout'. The 'Supported Ciphers & their order of preference' section shows a table of supported ciphers and their order of preference.

Name	PFS	Protocol	Authentication	Encryption
<input checked="" type="checkbox"/> ECDHE-ECDSA-AES256-GCM-SHA384	Yes	TLSv1.2	ECDSA	AESGCM
<input checked="" type="checkbox"/> ECDHE-ECDSA-AES256-SHA384	Yes	TLSv1.2	ECDSA	AES(256)
<input checked="" type="checkbox"/> ECDHE-RSA-AES256-GCM-SHA384	Yes	TLSv1.2	RSA	AESGCM
<input checked="" type="checkbox"/> ECDHE-RSA-AES256-SHA384	Yes	TLSv1.2	RSA	AES(256)
<input checked="" type="checkbox"/> DHE-RSA-AES256-GCM-SHA384	Yes	TLSv1.2	RSA	AESGCM
<input checked="" type="checkbox"/> DHE-RSA-AES256-SHA256	Yes	TLSv1.2	RSA	AES(256)
<input checked="" type="checkbox"/> ECDHE-ECDSA-AES128-GCM-SHA256	Yes	TLSv1.2	ECDSA	AESGCM
<input checked="" type="checkbox"/> DHE-RSA-AES128-GCM-SHA256	Yes	TLSv1.2	RSA	AESGCM
<input checked="" type="checkbox"/> DHE-RSA-AES128-SHA256	Yes	TLSv1.2	RSA	AES(128)
<input checked="" type="checkbox"/> ECDHE-ECDSA-AES256-SHA	Yes	SSLv3	ECDSA	AES(256)

Cipher details for this connection
encrypted.google.com:443
Cipher: ECDHE-ECDSA-AES256-GCM-SHA384
Protocol: TLSv1.2
Authentication: ECDSA
Encryption: AESGCM(256)
Key exchange: ECDH

Take a look of the following.



Name	PFS	Protocol	Authentication	Encryption
<input checked="" type="checkbox"/> ECDHE-ECDSA-AES256-GCM-SHA384	Yes	TLSv1.2	ECDSA	AESGCM(256)
<input checked="" type="checkbox"/> ECDHE-ECDSA-AES256-SHA384	Yes	TLSv1.2	ECDSA	AES(256)
<input checked="" type="checkbox"/> ECDHE-RSA-AES256-GCM-SHA384	Yes	TLSv1.2	RSA	AESGCM(256)
<input checked="" type="checkbox"/> ECDHE-RSA-AES256-SHA384	Yes	TLSv1.2	RSA	AES(256)
<input checked="" type="checkbox"/> DHE-RSA-AES256-GCM-SHA384	Yes	TLSv1.2	RSA	AESGCM(256)
<input checked="" type="checkbox"/> DHE-RSA-AES256-SHA256	Yes	TLSv1.2	RSA	AES(256)
<input checked="" type="checkbox"/> ECDHE-ECDSA-AES128-GCM-SHA256	Yes	TLSv1.2	ECDSA	AESGCM(128)
<input checked="" type="checkbox"/> DHE-RSA-AES128-GCM-SHA256	Yes	TLSv1.2	RSA	AESGCM(128)
<input checked="" type="checkbox"/> DHE-RSA-AES128-SHA256	Yes	TLSv1.2	RSA	AES(128)
<input checked="" type="checkbox"/> ECDHE-ECDSA-AES256-SHA	Yes	SSLv3	ECDSA	AES(256)

What you see here is the currently supported cryptographic ciphers that CyberDragon is using. What this specifically is showing is showing all the enabled/disabled ciphers and also, this is important, the *preferred order* of those ciphers that are used when initiating HTTPS connection.

So in plain English: The most top *enabled* cipher that is supported by both CyberDragon and the server that it tries to connect is used first. If that fails then the next cipher in the list is used and so on ...

There in addition of disabling/enabling these cryptographic ciphers, you can also reorder their position in the list by selecting them and then pressing either move up or move down button.

Also, you will see that there are lot's green stuff in there. Green is good. Orange is not so good. And red is bad. So if you want to be safe from NSA try to make so that the top most ciphers have:

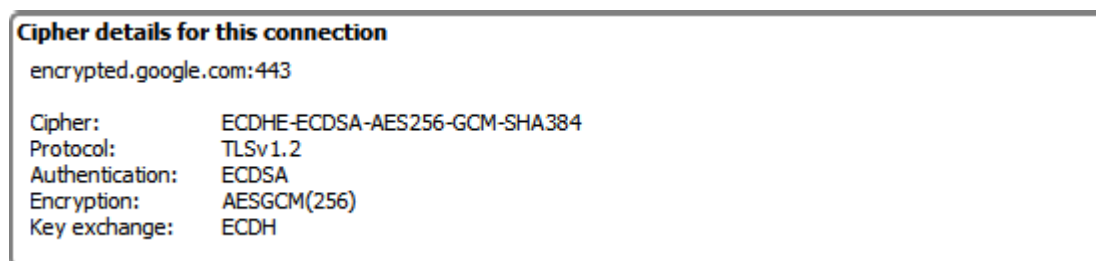
PFS (Perfect Forward Security) field set to Yes. Protocol set to TLSv1.2. And Encryption set to AES (any AES).

If you are really paranoid you could disable all other ciphers that don't support those criterias. Just keep in mind that then some very old (and insecure) servers might stop working for you, so maybe it's not a big loss anyway ...

After NSA scandal only a total dork has not upgraded their servers by now...

One note about disabling ciphers: These cipher settings are *application wide and they affect all the network connections that CyberDragon does*. So if you disable some ciphers and start experiencing, for example, SSL checking failures with Proxy checker, then you should enable the ciphers back again.

Last and final thing. Showing the cipher that was used during HTTPS connection:



encrypted.google.com:443	
Cipher:	ECDHE-ECDSA-AES256-GCM-SHA384
Protocol:	TLSv1.2
Authentication:	ECDSA
Encryption:	AESGCM(256)
Key exchange:	ECDH

Pretty trivial and does not need explanation.

Appendix A. Key shortcuts

Ctrl + F	Go to Find text field (type something to start searching)
Ctrl + P	Print current tab
Ctrl + T	Open new tab
Ctrl + W	Close current tab
Ctrl + +	Zoom in current page
Ctrl + -	Zoom out current page
F3	Find next
Shift + F3	Find previous
F5	Reload current tab
F6	Switch between web page and URL bar
F12	Hide/Show settings
Backspace	Go forward in page history
Shift + Backspace	Go backward in page history
Alt + Home	Go to homepage
Alt + G	Go to General tab
Alt + T	Go to Tracker Blocker tab
Alt + C	Go to Cookie Control tab
Alt + E	Go to Encryption tab
Alt + A	Go to Cache tab
Alt + D	Go to Downloads tab
Alt + N	Go to Notification tab

Appendix B. How to use CyberDragon with Tor?

Hello.

I will show here step-by-step how to setup CyberDragon browser to use anonymous Tor network.

Tor project has it's own browser called Tor Browser Bundle. Tor Browser Bundle is basically a Firefox fork that has few settings tweaked and most of the plugins stuff disabled and some third party extensions added (like HTTPS Everywhere from EFF). Also in that bundle is included the Tor program itself and its graphical frontend, Vidalia.

Now, Tor project recommends to use it's own browser and nothing else.

Well, I have been using Tor (and Vidalia) on my Linux machines long before they decided to start bundling Tor + Vidalia + Firefox fork together. So I honestly think I know what I am doing ...

And because I am also the author of CyberDragon browser and I know *my code* then there *should be no* privacy leaks when using it with Tor.

However, I have not done any checking with packet sniffer or anything like that to *really confirm* it and even thought I have checked my code many many times there still might be a bug or two in the rendering engine itself (QtWebkit) that CyberDragon uses and which I am not aware of.
So there certainly is a possibility for leaks.

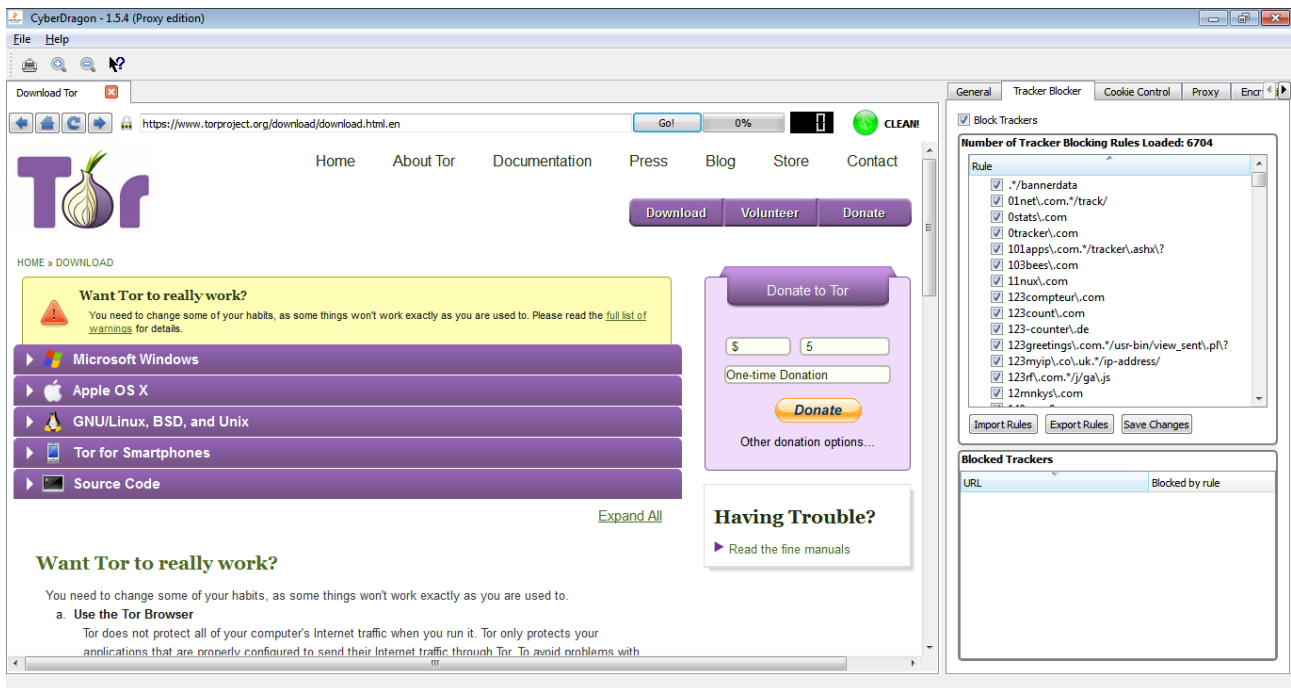
So if you find that CyberDragon leaks DNS stuff or your IP address while using Tor then immediately inform me by contacting <http://www.binarytouch.com/contact.php> and howto check and confirm that so that I can try to fix it.

Note also that CyberDragon as yet does not have functionality like HTTPS Everywhere. But it will have at the time of 1.6.x when I will start Encrypted edition. Just after finishing 1.5.x Proxy Edition serie.

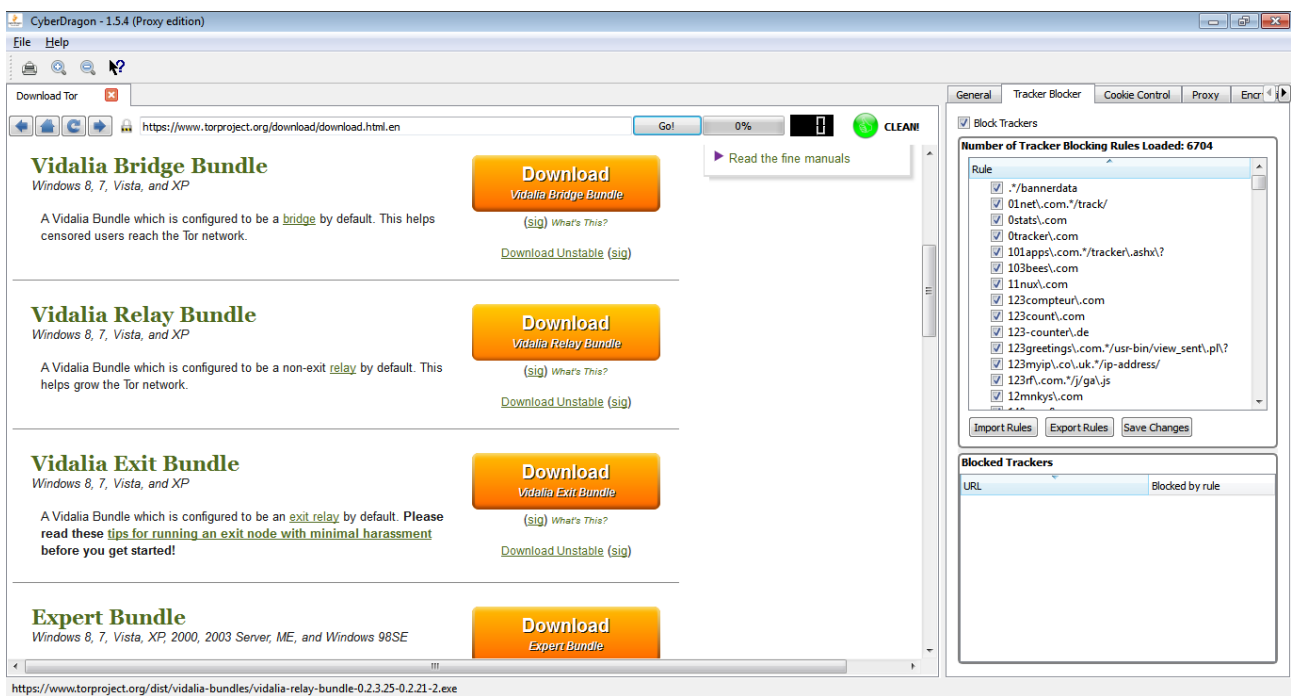
So while you use it with Tor always make sure you have https:// in front of your address and that little small padlock is closed and it says Encrypted when you hover mouse over it.

Now, let's proceed...

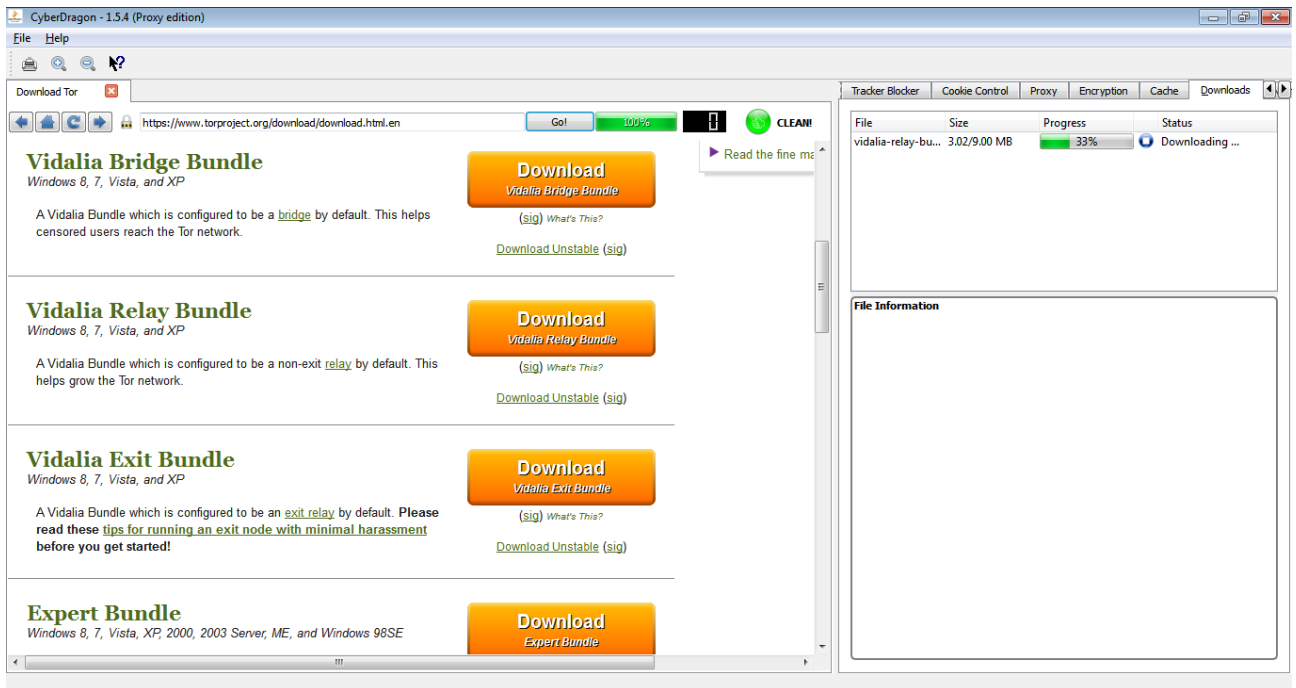
First start CyberDragon Browser and go to address: <https://www.torproject.org/download/download.html.en>



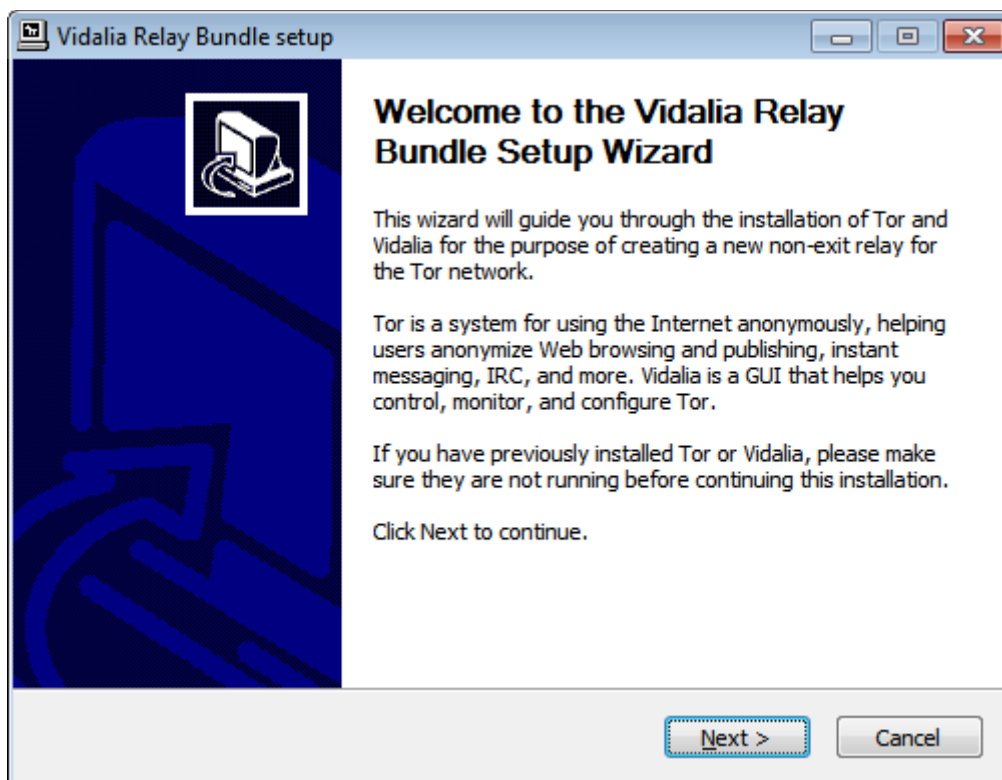
Next click Microsoft Windows link.
Scroll down until you see Vidalia Relay Bundle.



Start download and go to download tab by pressing Alt + D

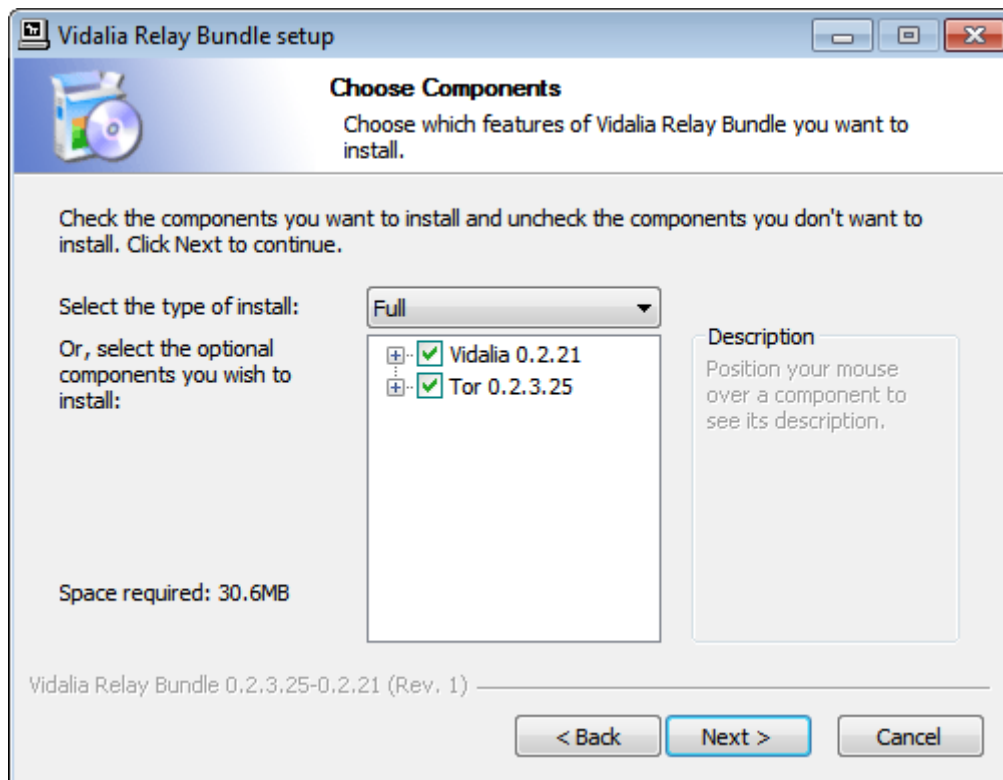


After it has finished double click the file name from the list and answer yes if UAC screen pops up. This is what it should show now.

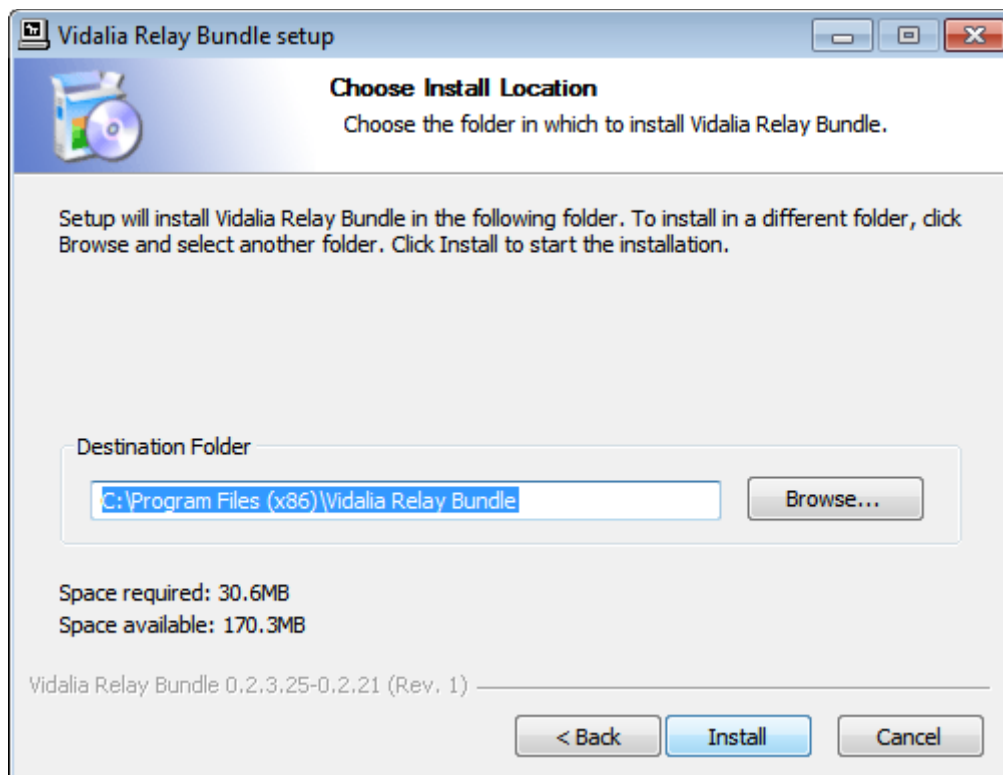


Click Next.

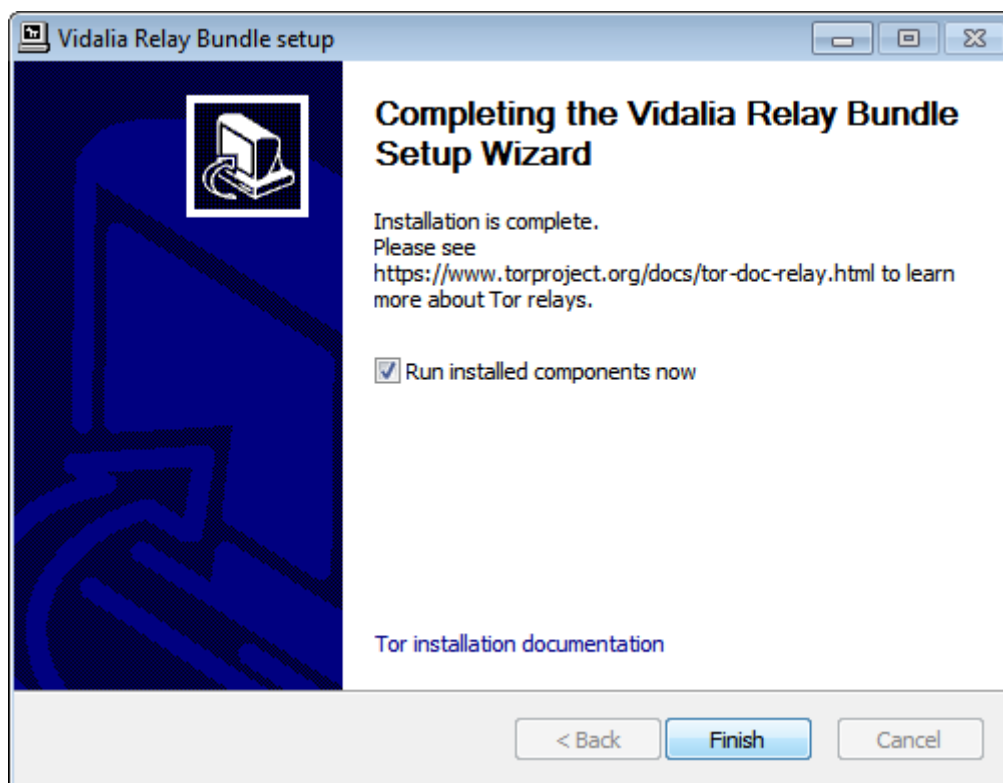
Defaults are ok, click Next.



And next ...

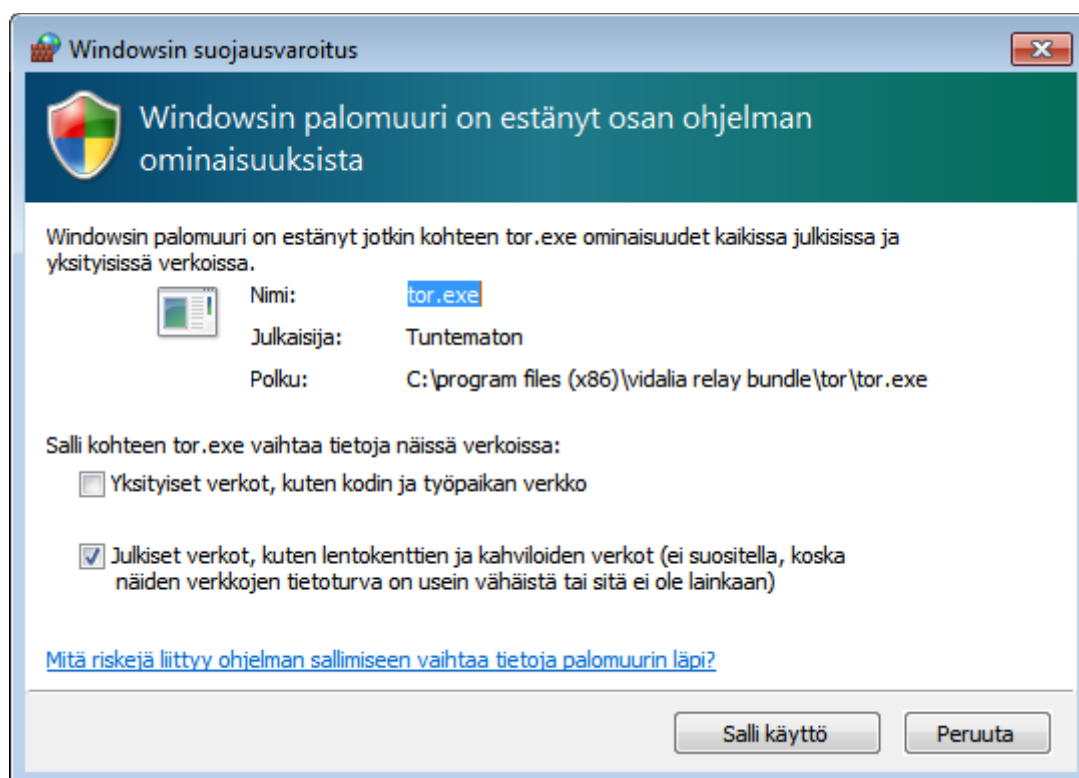


And Install...



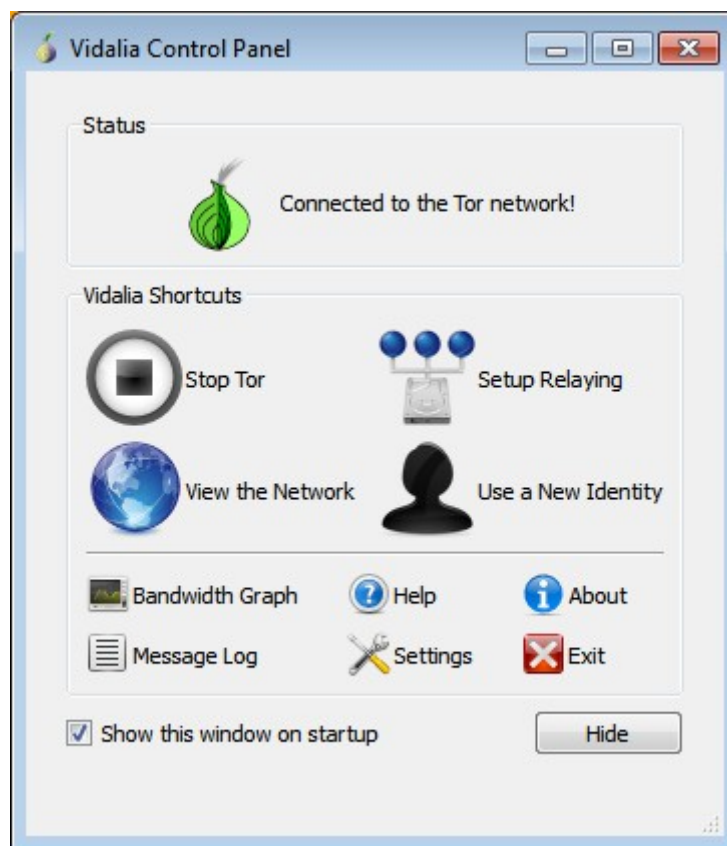
Complete the thing by running Finish.

After that Windows Firewall might start barking. Keep it happy and answer Accept.



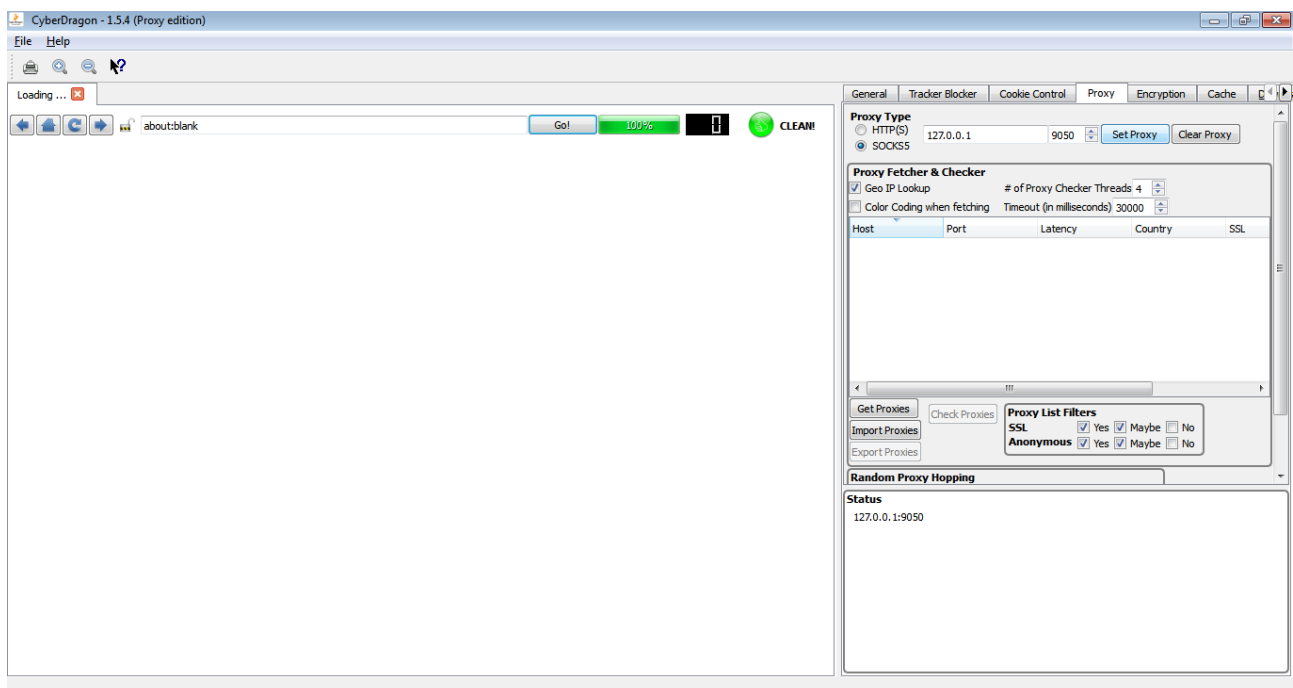
You have now Tor running.

Now go back to CyberDragon again and to it's Proxy tab.

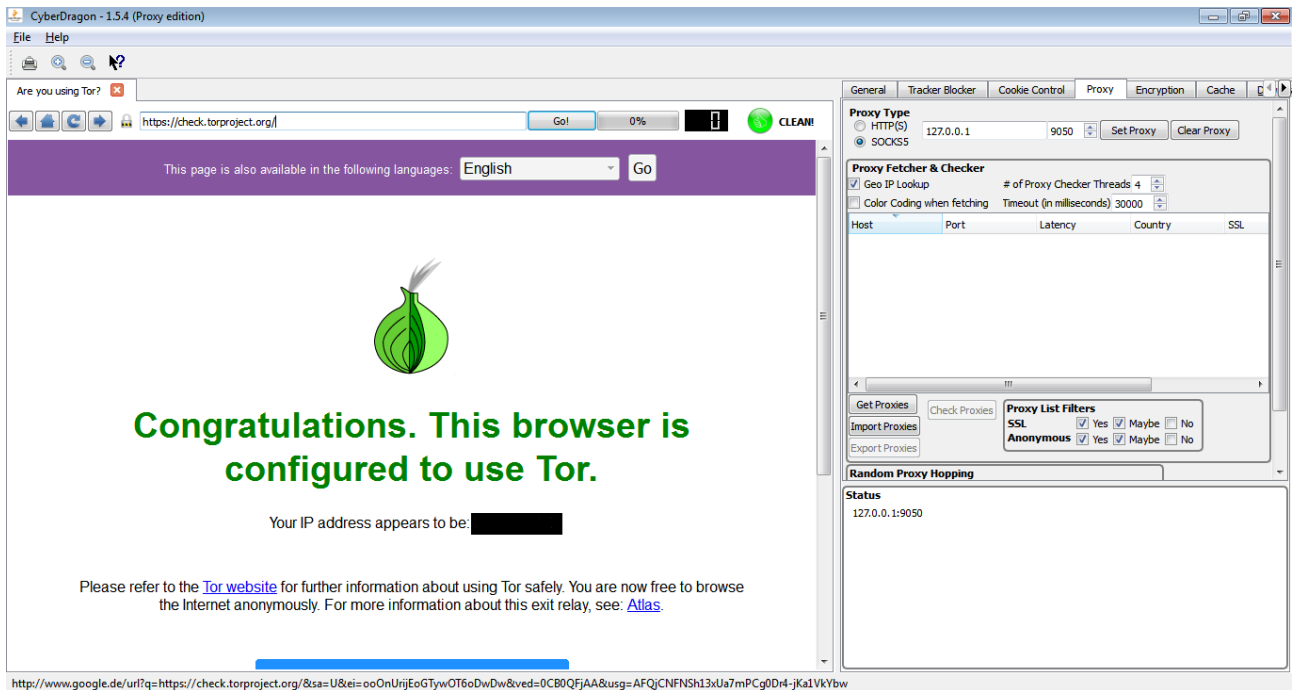


Select Proxy Type SOCKS5 and 127.0.0.1 (this means your own machine) as proxy IP address and port 9050 and click Set Proxy. The only time you have to click Set Proxy button is when you *manually* enter proxy ip:port. Otherwise you just select it from proxy list.

Status box at right bottom should now show 127.0.0.1:9050



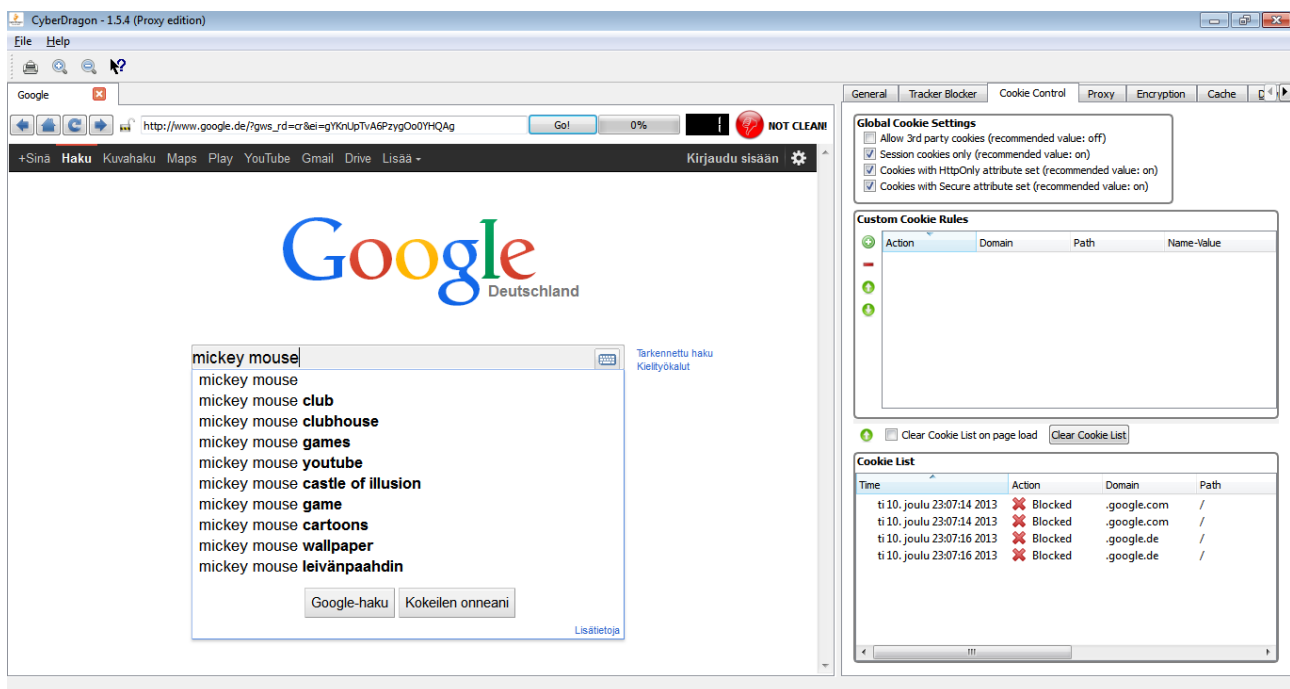
Let's cormirm that we really are using Tor and that it is working with our browser.
Go to the following address: <https://check.torproject.org/>



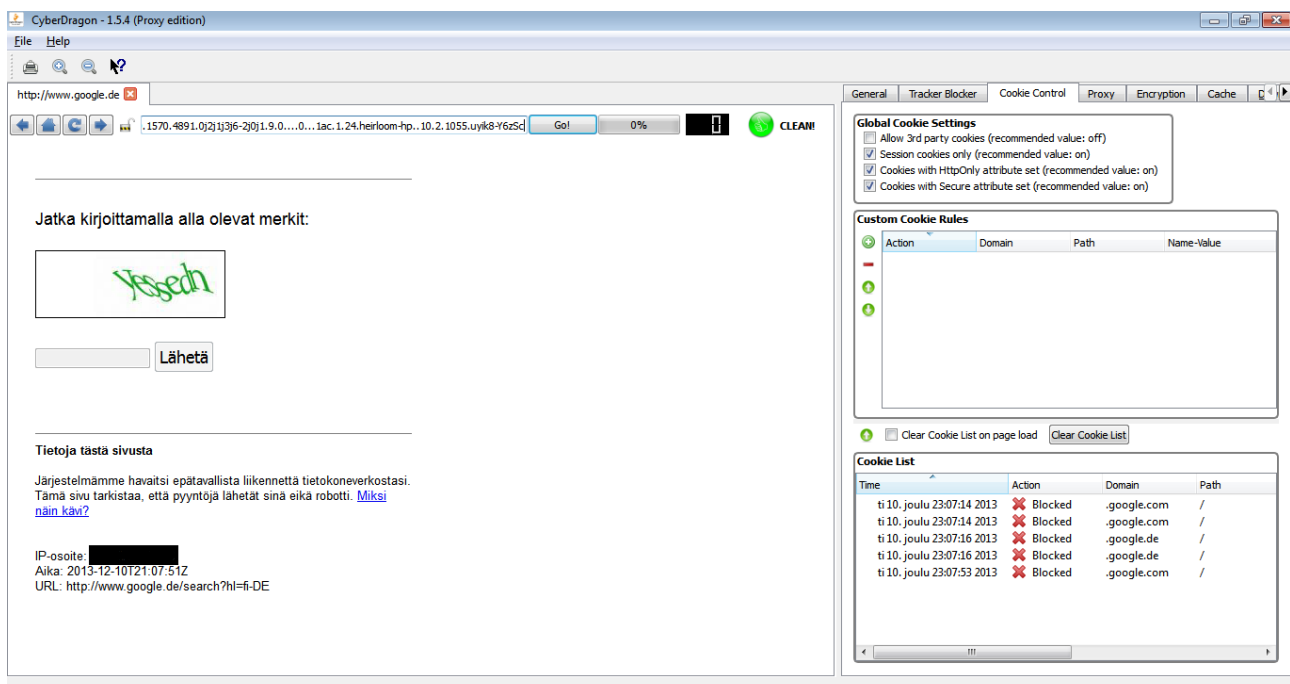
Yeah, it's working. I have not censored the IP-address because it would show my true IP-address (it doesn't) but because it would otherwise reveal the unselfish person/organization/company exit-node IP-address which are so important for Tor users. So no need to reveal it here for those in power.

Next let's try google and solve one of the problems that you will get with it when using Tor.

Let's search something...



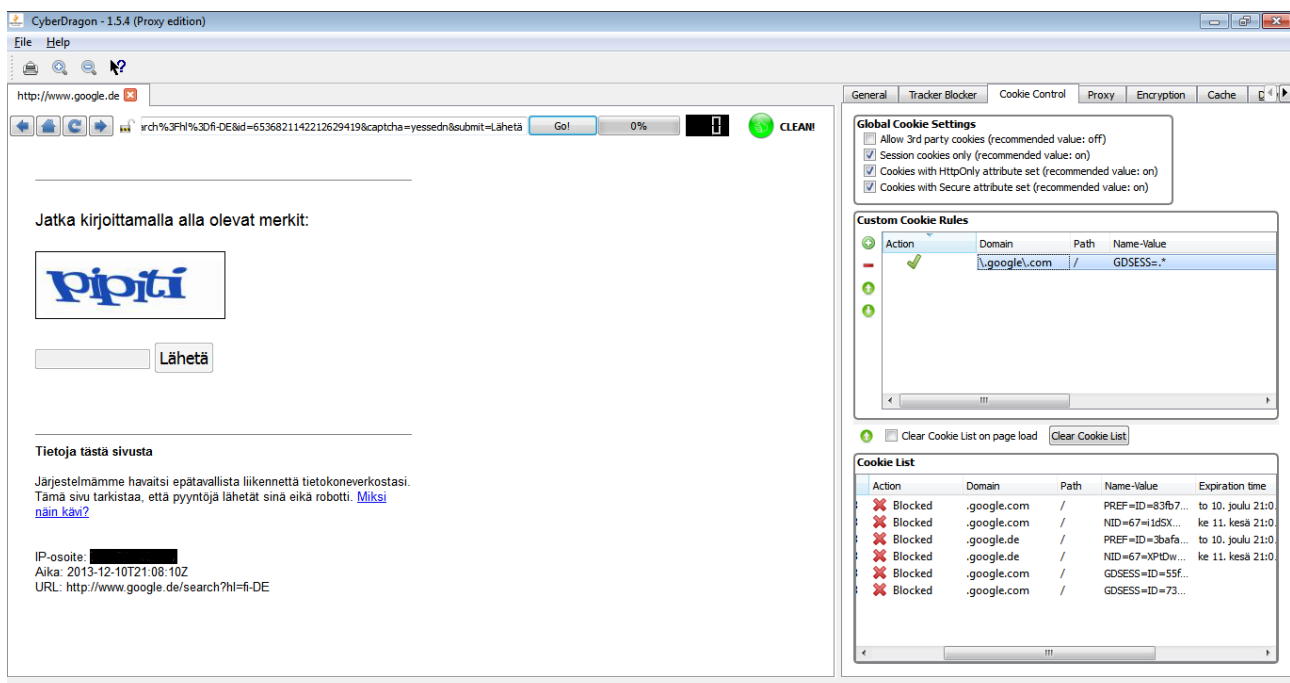
WTF???



Okay, what you see is that Google has somekind of protection that tries to prevent massive amounts of network traffic from accessing it from few IP-addresses (and that black box is again tor exit-node IP address, not mine). Usually those are results of somekind of malware attack (like worms), botnets, proxy users or Tor users. So even tough Google is not Internet operator company it still wants to play network cop and even tough the intention is good this really is more of the thing that ISP should be doing and not search engine company that sells it's user searches and data for advertisers...

Anyway, let's try that Captcha and see what happens

Okay, so it didn't let us pass. Tracker number shows zero and Mixed blocked content warning didn't fire up. So it obviously needs some cookie(s) enabled to let us pass.



Okay, after setting the Cookie List to sort all the cookies by Time of they arrival and checking the last arriving cookies we can see few session cookies there (Expiration time shows empty meaning that it is session cookie) at the bottom of the list.

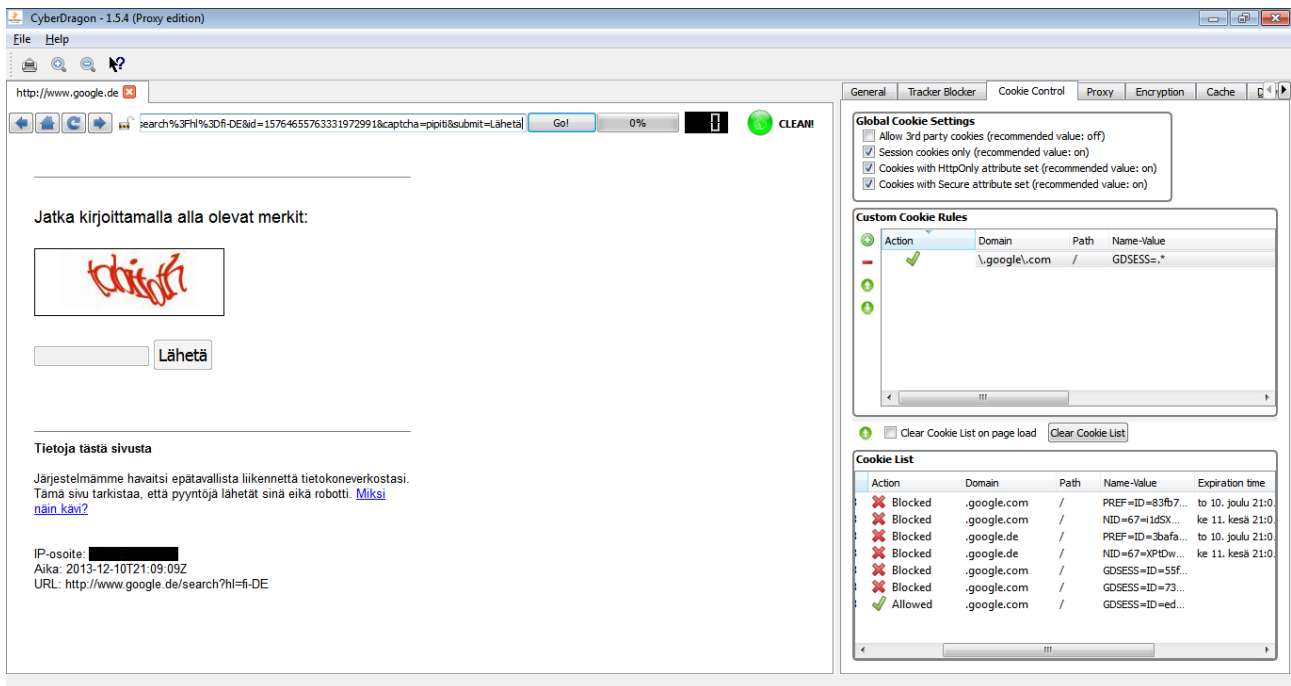
Session cookies are always good candidates to enable. So let's click the last cookie on that list, press that little green up arrow on top of Cookie List view to move it to Custom Cookie rules view, enable it (there's a bug that won't let you edit the cookie without enabling it first, I fix it in next version...) and edit its domain and name-value pair (all this is already done in the above picture).

As you can see from above picture I set domain as `.google\com`. I could have left it as just `.google.com` but because this is a regular expression it would have not been *exactly* correct. Later CyberDragon version will do that automatically for you, I promise...

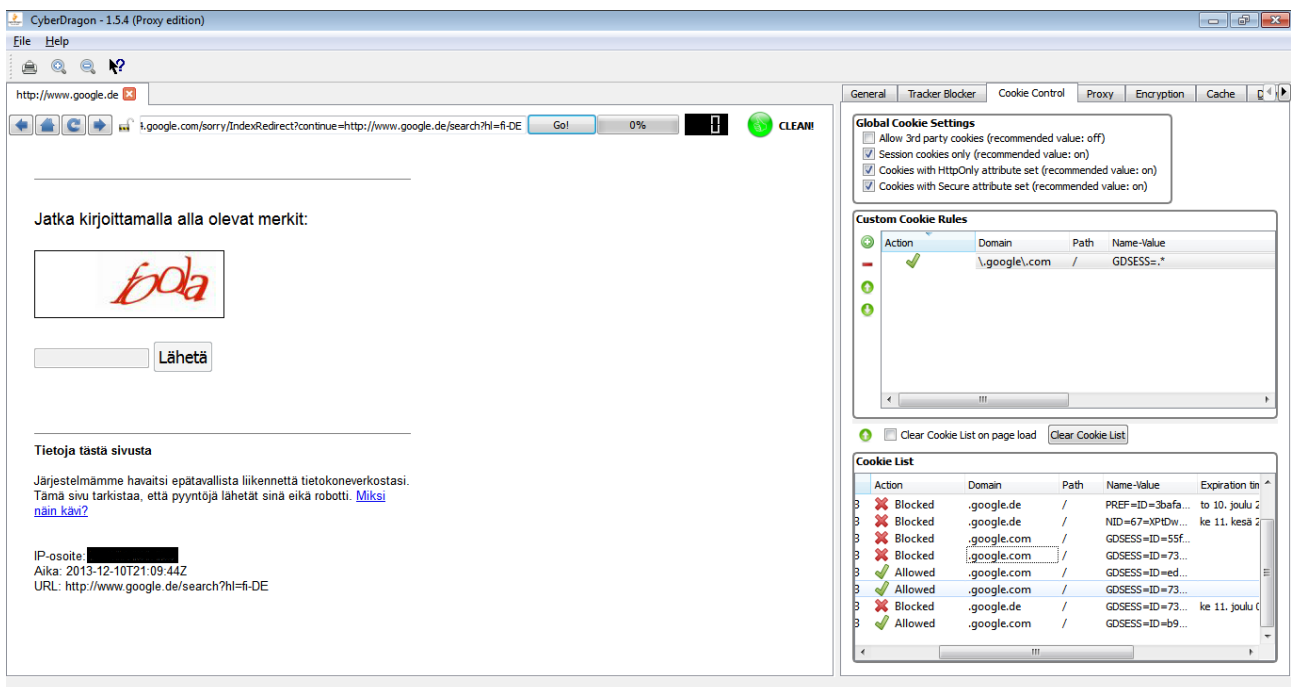
As for name-value pair I let the name value as it is (GDSESS) and modified the value part to be `.*` meaning any value. So the final name-value pair is: `GDSESS=.*`

Now let's try again that Captcha ...

Hmmm... Still nothing although it shows clearly at Cookie List that GDSESS cookie from domain `.google.com` is allowed. Maybe I typed that Captcha wrong? New try



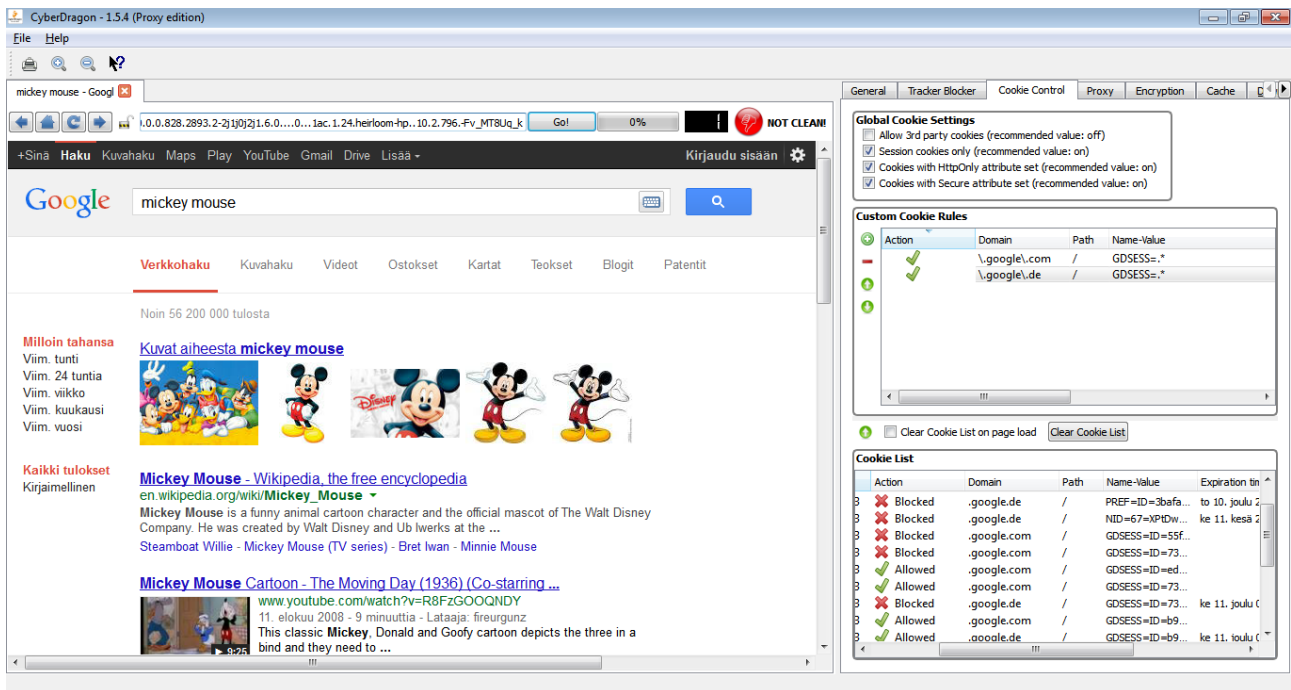
Damn! Still nothing! But wait a minnit There's something new appearing in that Cookie List



Aha! Enabling GDSESS variable for .google.com was not enough. Because my traffic goes throught Tor and exits at somewhere at Germany currently (Tor keeps changing your exit node) I need to enable that for .google.de domain too ...

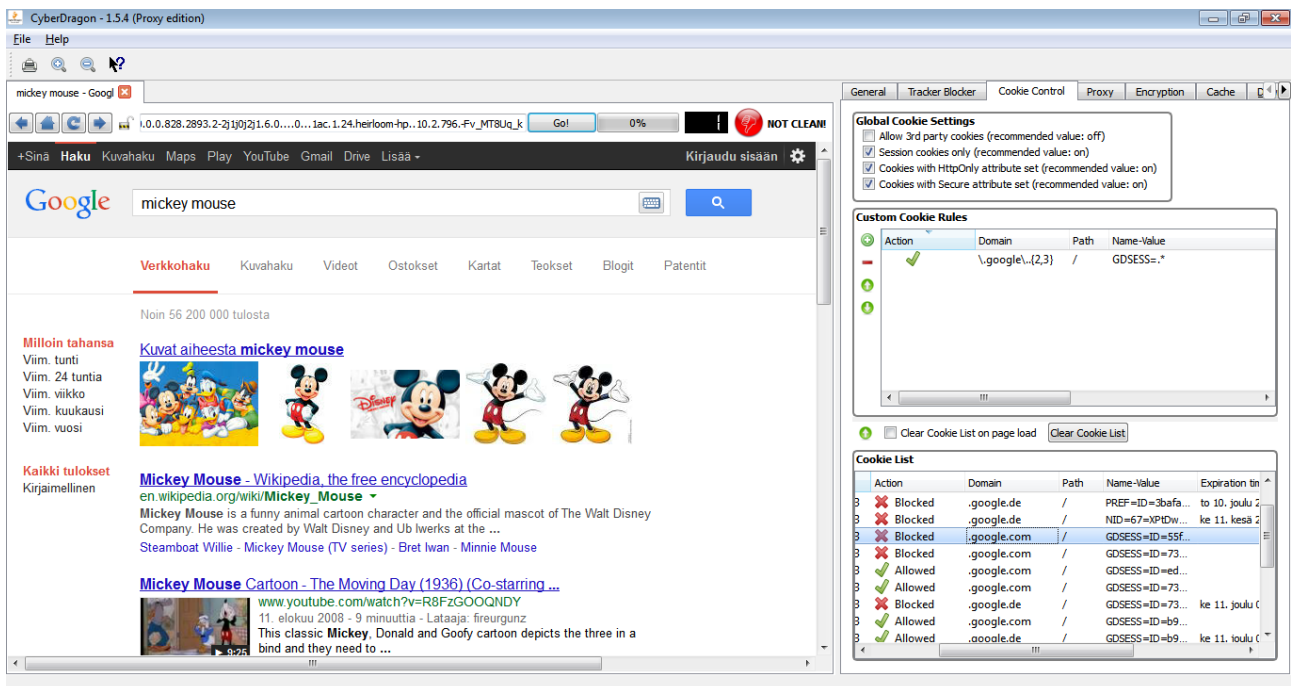
Let's add another custom cookie rule with the same GDSESS valuable as previously but set domain this time to \.goolge\de and let see what happens....

Tadaaa!



However, now we have another problem. If Google needs that GDSESS cookie for all the other tld domains too, not just .com, then we either have to add all them (which would be tiresome and stupid) or make a better rule.

Let's remove that .google\..de rule and make just one single, better rule to rule them all.



As you can see there is now just one rule and domain part is now in a form of: .google\.{2,3} Notice the extra dot between \. and {2,3} ? It means any character. And {2,3} means: at least two but no more than three.

So in essence this regular expression rule means domain name that starts with .google. and ends with any two (like **de**) or three (like **com**) characters.

So, there! You have now your custom cookie rules set for google when using Tor network. And they will be saved automatically when you exit CyberDragon.

Unfortunately, as Tor sometimes changes your exit-node connection, you will then have to fill that Captcha form again thanks to Google.... :-/

But at least it will now let you pass it! :-)

Keep in mind that all the troubles with Google had nothing to do with CyberDragon. This will happen with *any browser* that tries to access Google through Tor.

Just use CyberDragon browser with Tor like you would use Tor Browser and you should be safe.

See more from here:

<https://www.torproject.org/download/download.html.en#warning>

Surf Safe!

Appendix C. Linux

Getting & Starting CyberDragon under Linux

Here's how you can use CyberDragon browser with Linux.

First: Get it from <http://sourceforge.net/projects/cyberdragonbrowser/files/Linux/>

Second: Extract it to some place. Most modern Linux distributions come with a nice GUI for that task but if you are unlucky and don't have any then you can do the following command from Terminal:

```
tar -xSvf CyberDragonLinux_1.6.1.tar.bz2
```

Third: Go to the CyberDragon folder and find a file called CyberDragon and double-click it *or* from Terminal:

```
cd CyberDragon_1.6.1  
./CyberDragon
```

Please note that CyberDragon needs to have some **X Window System desktop environment** (like Gnome, KDE, LXDE or Xfce) or **X Window Manager** (like Fluxbox or Openbox) running.

How to use proxy pingin with non-root user on Fedora 20 ?

In Linux only superuser (a.k.a. root) is allowed to open raw socket for doing ICMP echo (a.k.a. ping). Normally, for non-root users, it's not possible to get latency (ping) of proxies without some extra work.

For non-root users there are two options:

1. Set the setuid bit on of the CyberDragon.bin binary.
(not really recommended, because it gives too much rights)
2. Install libcap and then use setcap command to give CyberDragon.bin binary the privileges to use raw sockets even for non-root users.
(use this if possible)

Option 1. setuid

```
chmod u+s CyberDragon.bin
```

Option 2. libcap

```
yum install libcap  
chmod u-s CyberDragon.bin  
setcap cap_net_raw+ep CyberDragon.bin
```

How to make Flash work with Fedora 20 & CyberDragon?

1. Go to <http://www.if-not-true-then-false.com/2010/install-adobe-flash-player-10-on-fedora-centos-red-hat-rhel/> and follow the instructions to install Flash.
2. After installing make sure that you have directory /usr/lib/mozilla/plugins and that under that directory is a symlink pointing to /usr/lib/flash-plugin/libflashplayer.so

If not then create the directory and symlink with the following commands:

```
mkdir -p /usr/lib/mozilla/plugins && cd /usr/lib/mozilla/plugins  
ln -s /usr/lib/flash-plugin/libflashplayer.so .
```

Note the dot (.) at the end of second command.

3. Go to <https://www.youtube.com> and start watching

Appendix D. Donations

If you find CyberDragon browser useful then please consider supporting it by donating. There are two ways how you can donate.

1. Go to <http://www.binarytouch.com/>. On the right side there is a my bitcoin address that you can use if you have bitcoins and want to donate some small sum. There is also QR code for easier access with smartphone.

If you don't have bitcoins then you can use PayPal to donate. You don't need PayPal account yourself but you do need a credit card.

2. Another option for supporting this project is by going to <http://www.amazon.com/> and buying my physicist father book(s).

He has published the following books (which I helped to edit):

Increase Endurance, Strength, Hormones and Sex with Adaptogenic Herbs and Foods
<http://www.amazon.com/Increase-Endurance-Strength-Hormones-Adaptogenic/dp/1491281251/>

Aphrodite: Stimulate Sex Life, Libido, Erection and Orgasm Naturally with Aphrodisiacs
<http://www.amazon.com/Aphrodite-Stimulate-Erection-Naturally-Aphrodisiacs/dp/1489538054/>

Obesity: Decrease Overweight Quickly and Naturally
<http://www.amazon.com/Obesity-Decrease-Overweight-Quickly-Naturally/dp/1482560046/>

Decrease Hypertension and Cholesterol Naturally
<http://www.amazon.com/Decrease-Hypertension-Cholesterol-Naturally/dp/1478335025/>

Thank You!