Read or listen to this article online

Scan the QR code to open this article on XO.PL

Article source:

https://xo.pl/it-made-fun-inet-proxy

INTERNET

Proxy: the middleman who goes shopping for you

A brokerage office called “We'll Handle It For You” opened in the Quiet Network Estate. The sign outside said: “Don't go yourself. Send us!” The manager promised to collect purchases, deliver parcels and negotiate with sellers. Greg asked whether the office could also explain to the IT department why his printer refused to print. The manager turned pale. 🤣

Let somebody else go for me

— I need a catalogue from the shop on the other side of town — said Greg.

— Leave the order with me — offered Mr Proxy.

— Can't I go myself?

— Of course. But if you're using this office, I contact the shop and bring back the reply.

— Does the shop know I'm the buyer?

— It mainly sees my connection. What else it learns depends on how the order is handled.

That's the idea of a **forward proxy**: an intermediary on the client's side. A browser or another program sends its request to the proxy, which forwards it to the destination server according to its configuration and brings back the response. It may also enforce access policies or log requests.

— So you're an extra stop?

— Yes, but sometimes a useful one.

— And other times?

— Other times I sign every form three times.

Corporate purchasing department

The next day, the boss of Johnson Company stormed into the office.

— From now on, employees order supplies only through purchasing!

— Even paper clips? — asked Greg.

— Especially paper clips. Someone ordered gold ones last week.

Companies may use forward proxies to enforce access policies, authenticate users or keep connection logs. Some installations can block particular destinations or categories of websites. The exact behavior depends on the configuration: the word “proxy” alone does not mean every request gets filtered.

— What if an application doesn't know it should use the intermediary?

— It may connect directly, if the network permits it.

— So settings and network rules matter?

— Naturally.

— I know one employee who's going to try bypassing purchasing.

— Which one?

— The one who ordered golden paper clips.

Now the intermediary stands in front of the company

Across the street was a business called “Three Kitchens, One Sign”. Customers walked in with orders but had no idea which of three cooks to approach.

— I WANT SOUP! — yelled the first.

— I NEED AN INVOICE! — shouted the second.

— AND I WANT THE MANAGER!

A second Mr Proxy took a seat at reception.

— Relax. Everyone comes to one address. I'll decide which kitchen handles each order.

That's a **reverse proxy**. It sits in front of servers, accepts clients' requests and routes them to the appropriate backend services. A visitor connects to the public entrance without having to know every internal server.

— So the first Proxy works for the buyer, and this one for the seller?

— Exactly.

— And they're both named Proxy?

— Yes.

— How can I tell them apart?

— Look at which side of the counter they're standing on.

One address, three kitchens and a broken oven

Reception received four soup orders, five pizza orders and one note reading “Please don't break anything”.

— Soup to kitchen one, pizza to kitchen two, invoices to department three.

— What if oven two breaks down? — Greg asked.

— If there's another healthy station that can do the same work, I can send the order there.

A reverse proxy can route traffic using hostnames, URL paths and other rules. It may also **balance load** between servers and monitor their health. It cannot magically create capacity if every kitchen is closed.

Real-world examples include Nginx, HAProxy and appropriately configured Apache. One product can serve more than one role; you don't necessarily need a separate booth for every feature.

— So if every oven fails, I get no pizza?

— Not even a napkin.

— Then perhaps I shouldn't ask for the bill.

HTTPS: the sealed envelope

Greg carried a sealed envelope to the counter.

— Please deliver this to the shop. And don't read it!

— Fine — said Mr Proxy. — But we need to agree what I'm allowed to do.

With ordinary **HTTPS connections** through a forward proxy, the `CONNECT` method is often used to establish a tunnel to the target server. The intermediary forwards encrypted traffic but, without additional TLS inspection, cannot read the full request and response contents.

— So you might know where I'm going, but not necessarily what I ordered?

— In many configurations, yes. That still doesn't guarantee anonymity.

A reverse proxy, on the other hand, may **terminate TLS**, decrypt the request and forward it to a backend server — sometimes using another encrypted connection. In that case, it's an authorized receiving desk, not just a courier carrying a sealed envelope.

— Isn't that opening somebody's mail?

— If it's the company's own reception, handling the order is part of the job. What matters is who has access and under what rules.

— So the padlock doesn't answer every question?

— No padlock ever did. It's just metal.

Proxy isn't a VPN, even in a nice suit

A truck driver with VPN painted on the side walked into the office.

— I hear you also transport other people's things.

— Yes — replied Proxy. — But not in quite the same way.

A proxy intermediates particular connections or protocols. A VPN creates a network tunnel and, depending on routes and configuration, can carry traffic from many applications through it. The technologies have different purposes. Neither guarantees anonymity or universal security on its own.

— So a proxy is an employee handling specific errands, while a VPN is a truck that can take the whole route?

— A useful simplification.

— Could I put the employee inside the truck?

— The approaches can be combined.

— Then I need a transport manager.

— No, Greg. We really don't.

Why is there a newspaper under your counter?

Greg noticed a familiar stack of newspapers.

— Oh no. Yesterday's edition again?!

— Not every stored copy is outdated — Proxy protested. — I may keep some responses for other customers, if the rules permit it.

Some proxies can cache responses to avoid fetching the same content repeatedly. That is **caching at the intermediary**, not the **cache inside Greg's browser**. Storage rules still matter, especially for private or encrypted content.

— Is that the same cupboard as at the hotel?

— No. At the hotel the newspaper was beside your table. Here a permitted shared copy might serve several customers.

— So there are two cupboards?

— Two different places.

— And the storage fee?

— That's charged separately.

The intermediary's final report

That evening, the two Proxy offices held a joint meeting.

— I take requests from clients and deal with servers on their behalf — said forward proxy.

— I stand in front of the servers and direct requests to the correct backend — replied reverse proxy.

— And we can both help enforce policies?

— If properly configured.

— Can you make coffee?

— We're not the coffee department.

A proxy is an intermediary in network communication. A forward proxy acts on the client's side, while a reverse proxy provides the entrance to services on the server side. Depending on the setup, a proxy may assist with access control, request routing, load balancing or caching.

Greg approached the counter.

— So who's getting me some paper clips?

Both intermediaries pointed at each other.

— Here we go again — sighed the manager.

Management of “We'll Handle It For You” reports: all orders were forwarded in accordance with procedure. Forward proxy referred the customer to reverse proxy, reverse proxy sent the case to the backend, and the backend requested a ticket number. Greg finally received his paper clips. Gold ones. The invoice included a line item: “Intermediation fee for intermediary intermediation”. 🛒📨🤣❤️