



21212 Academy

Configurando vários domínios no nginx sem repetição

Intro

Dando continuidade ao último artigo sobre landing pages para desenvolvedores, onde explico um jeito mais "direto ao ponto" pra galera técnica o que é uma Landing Page, para que ela serve e algumas sugestões de como implementá-las. Agora neste artigo vamos ver uma configuração do servidor nginx que vai facilitar bastante a configuração das suas landing pages com pouca repetição.

Aqui na 21212 Academy nós procuramos realizar aquilo que pregamos. Como por exemplo validar hipóteses com landing pages, assunto do artigo anterior.

Recentemente criamos 7 landing pages com os respectivos domínios:

- www.modelo-financeiro-startup.com.br
- www.como-atrair-mais-clientes.com.br
- www.como-validar-minha-ideia.com.br
- www.conselhos-para-startups.com.br
- www.fluxo-de-caixa-startup.com.br
- www.o-que-e-o-lean-canvas.com.br
- www.como-criar-um-mvp.com.br

Elas foram criadas usando o framework Middleman e seguem uma estrutura bastante parecida. Minha primeira reação foi procurar uma maneira de declarar estas landing pages todas com um mínimo de duplicação na configuração do nginx.



Config do nginx

Normalmente uma configuração de uma landing page estática no nginx é bastante simples:

```
server {  
    listen 80; # que porta estamos ouvindo  
    server_name www.abc.com.br; # que hostname(s) estamos  
    interessados  
    root /srv/abc.com.br; # aonde estão os arquivos no HD  
    index index.html index.htm; # urls que terminam em "/" abrem  
    estes arquivos  
    # para esta regexp (case-insensitive), nao logue nada e  
    cacheie pra sempre  
    location ~*  
    ^.+\. (ogg|ogv|svg|svgz|eot|otf|woff|mp4|ttf|rss|atom|jpg|jpeg|gif|  
    png|ico|zip|tgz|gz|rar|bz2|doc|xls|exe|ppt|tar|mid|midi|wav|bm  
    p|rtf|css|js)$ {  
        access_log off; log_not_found off; expires max;  
    }  
}
```

Copiar e colar este bloco de server {} estava fora de cogitação. Eu queria um bloco de server {} somente. Para isso eu precisaria usar variáveis. E pra usar variáveis eu precisaria de condicionais dentro do arquivo de configuração. Mas quando o próprio desenvolvedor te desestimula a usar a parada, melhor evitar né :)



A diretiva map

Resolvi estudar então a diretiva map, que funciona como um switch-case (ou case-when) sem a penalidade de desempenho. Recapitulando, uma estrutura de switch-case em uma linguagem derivada da sintaxe de C, segue o seguinte padrão:

```
char* str;
switch (variavel) {
    case 1:
        str = "Um";
        break;
    case 2:
        str = "Dois";
        break;
    default:
        str = "Erro!";
        break;
}
```

Este exemplo usando a diretiva map seria mais ou menos assim:

```
map $variavel $str {
    default Erro;

    1 Um;
    2 Dois;
}
```

Basicamente ele mapeia (duh) os valores da primeira variável (coluna da esquerda), atribuindo o segundo valor à variável da direita.

Sabendo disso, usei dois maps (um para normalizar o hostname e outro para definir o caminho dos arquivos):

```
map $http_host $canonical_url {
    hostnames;

        .como-atrair-mais-clientes.com.br
www.como-atrair-mais-clientes.com.br;
        .modelo-financeiro-startup.com.br
www.modelo-financeiro-startup.com.br;
        .como-validar-minha-ideia.com.br
www.como-validar-minha-ideia.com.br;
        .conselhos-para-startups.com.br
```



```
www.conselhos-para-startups.com.br;
                                .fluxo-de-caixa-startup.com.br
www.fluxo-de-caixa-startup.com.br;
                                .o-que-e-o-lean-canvas.com.br
www.o-que-e-o-lean-canvas.com.br;
    .como-criar-um-mvp.com.br www.como-criar-um-mvp.com.br;
}
```

O comando `hostnames` faz com que a coluna da esquerda sirva como coringa, para dar match tanto com "example.com" ou "*.example.com" se voce utilizar a string ".example.com". Com este exemplo acima nós mapeamos qualquer combinação para a versão canônica (www.example.com) e atribuímos este valor a variável `$canonical_url`.

O segundo bloco mapeia os caminhos:

```
map $http_host $canonical_path {
    default "";

                                www.como-atrair-mais-clientes.com.br
como-conseguir-os-100-primeiros-clientes;
    www.como-validar-minha-ideia.com.br como-validar-sua-ideia;
                                www.conselhos-para-startups.com.br
conselhos-juridicos-para-startups;
                                www.fluxo-de-caixa-startup.com.br
como-criar-um-fluxo-de-caixa;
                                www.o-que-e-o-lean-canvas.com.br
lean-canvas-e-business-model-canvas;
                                www.como-criar-um-mvp.com.br
como-criar-uma-startup-comecando-pelo-mvp;
}
```

Agora que temos como garantir que a url será a canônica (continue lendo), podemos fazer esse mapping só com as urls canônicas (www...). Como podem ver acima, alguns caminhos possuem nomes bastante diferentes dos hostnames. E pra fechar, o bloco `server {}` único e belo:

```
server {
    # todos os hostnames que nos interessam
    server_name .modelo-financeiro-startup.com.br
                .como-atrair-mais-clientes.com.br
                .como-validar-minha-ideia.com.br
                .conselhos-para-startups.com.br
                .fluxo-de-caixa-startup.com.br
                .o-que-e-o-lean-canvas.com.br
                .como-criar-um-mvp.com.br;
    # se a url usada não for a canônica, faz um 301 pra canônica
    if ($canonical_url != $http_host) {
        return 301 $scheme://$canonical_url$request_uri;
    }
}
```



```
}  
# imagens, javascripts e stylesheets estão na raiz  
location ~ ^/(images|javascripts|stylesheets)/ {  
    set $canonical_path "";  
}  
# raiz do build do middleman  
root /srv/lps/$canonical_path;  
# parte comum  
listen 80;  
index index.html;  
  
                                location ~*  
^.+\.(ogg|ogv|svg|svgz|eot|otf|woff|mp4|ttf|rss|atom|jpg|jpeg|gif|png|ico|zip|tgz|gz|rar|bz2|doc|xls|exe|ppt|tar|mid|midi|wav|bmp|rtf|css|js)$ {  
    access_log off; log_not_found off; expires max;  
}  
}
```

Conclusão

Bom, consegui chegar no meu objetivo de minimizar a duplicação usando um condicional que não vi contra indicações. Se você souber de alguma contra indicação ou uma forma mais otimizada de fazer isso aí, deixe um comentário no post ou no fórum!