

Boas práticas na gestão de bloqueios de DNS determinado por autoridades



Ayub T.

Sistemas Autônomos de ISPs andam sobre o fio da navalha

- **Quem manda bloquear?**
 - Anatel
 - Ancine
 - Juiz do Poder Judiciário
 - Ministério da Fazenda
 - ANPD

Sistemas Autônomos de ISPs andam sobre o fio da navalha

- **Quem manda bloquear?**

- Anatel
- Ancine
- Juiz do Poder Judiciário
- Ministério da Fazenda
- ANPD

- Quem manda **desbloquear**?

- Raramente **judiciário** determina desbloqueio ou ordens que expiram.
- **Anatel** e **Ancine** têm operações com hora inicial e hora final para grandes AS.
- **Ministério da Fazenda** também apresenta data de expiração no pedido inicial.
- **ANPD** ainda não determinou o primeiro bloqueio.

E se não desbloquear a tempo?

- O Sistema Autônomo estará violando:
 - A neutralidade da rede (Art. 9º) do **Marco Civil da Internet**.
 - Falha na prestação do serviço (Art. 14º) do **Código de Defesa do Consumidor**.
 - Dano Material e Lucros Cessantes (Art. 186 e 927) do **Código Civil**.

**O débito técnico do desbloqueio é também
um *débito* jurídico.**

**Boa prática: uma gestão dos bloqueios
e desbloqueios de FQDN nos servidores
de DNS recursivo.**

Como usualmente é feito?

```
# /etc/unbound/blocks.conf  
local-zone: "pirataria-velha.com." always_nxdomain  
local-zone: "bet-irregular.net." always_nxdomain  
local-zone: "streaming-valido.xyz." always_nxdomain
```

Como usualmente é feito?

```
# /etc/unbound/blocks.conf  
local-zone: "pirataria-velha.com." always_nxdomain  
local-zone: "bet-irregular.net." always_nxdomain  
local-zone: "streaming-valido.xyz." always_nxdomain
```

O quê?
Como?
Quando?
Por quê?
Quem?

Como usualmente é feito?

```
# /etc/unbound/blocks.conf  
local-zone: "pirataria-velha.com." always_nxdomain  
local-zone: "bet-irregular.net." always_nxdomain  
local-zone: "streaming-valido.xyz." always_nxdomain
```

O quê está bloqueado e o que não está?

Como foi realizado esse bloqueio?

Quando ele foi iniciado e quando foi encerrado?

Por que ele foi bloqueado (a ordem)?

Quem realizou o bloqueio?

Como deveria ser feito?

```
# /etc/unbound/blocks.conf  
local-zone: "pirataria-velha.com." always_nxdomain # OFICIO-111 | EXPIRA: 2026-04-15  
local-zone: "bet-irregular.net." always_nxdomain # OFICIO-222 | EXPIRA: 2026-06-01  
local-zone: "streaming-valido.xyz." always_nxdomain # OFICIO-333 | EXPIRA: 2026-12-31
```

Como deveria ser feito?

```
# /etc/unbound/blocks.conf  
local-zone: "pirataria-velha.com." always_nxdomain # OFICIO-111 | EXPIRA: 2026-04-15  
local-zone: "bet-irregular.net." always_nxdomain # OFICIO-222 | EXPIRA: 2026-06-01  
local-zone: "streaming-valido.xyz." always_nxdomain # OFICIO-333 | EXPIRA: 2026-12-31
```

O quê?

Como?

Quando?

Por quê?

Quem?

Como deveria ser feito?

```
# /etc/unbound/blocks.conf  
local-zone: "pirataria-velha.com." always_nxdomain # OFICIO-111 | EXPIRA: 2026-04-15  
local-zone: "bet-irregular.net." always_nxdomain # OFICIO-222 | EXPIRA: 2026-06-01  
local-zone: "streaming-valido.xyz." always_nxdomain # OFICIO-333 | EXPIRA: 2026-12-31
```

O quê?

Como?

Quando?

Por quê?

Quem?

Como deveria ser feito?

```
# /etc/unbound/blocks.conf  
local-zone: "pirataria-velha.com." always_nxdomain # OFICIO-111 | EXPIRA: 2026-04-15  
local-zone: "bet-irregular.net." always_nxdomain # OFICIO-222 | EXPIRA: 2026-06-01  
local-zone: "streaming-valido.xyz." always_nxdomain # OFICIO-333 | EXPIRA: 2026-12-31
```

O quê?

Como?

Quando?

Por quê?

Quem?

Como deveria ser feito?

```
# /etc/unbound/blocks.conf  
local-zone: "pirataria-velha.com." always_nxdomain # OFICIO-111 | EXPIRA: 2026-04-15  
local-zone: "bet-irregular.net." always_nxdomain # OFICIO-222 | EXPIRA: 2026-06-01  
local-zone: "streaming-valido.xyz." always_nxdomain # OFICIO-333 | EXPIRA: 2026-12-31
```

O quê?

Como?

Quando?

Por quê?

Quem?

Git para versionamento de .conf

src/core/instance/render-helpers/resolve-scoped-slots.js

```
2
3 export function resolveScopedSlots (
4   fns: ScopedSlotsData, // see flow/vnode
5
6   hasDynamicKeys: boolean,
7   contentHashKey: number,
8   res?: Object
9 ): { [key: string]: Function, $stable: boolean } {
10   res = res || { $stable: !hasDynamicKeys }
11   for (let i = 0; i < fns.length; i++) {
12     const slot = fns[i]
13     if (Array.isArray(slot)) {
14       resolveScopedSlots(slot, hasDynamicKeys, null, res)
15     } else if (slot) {
16       // marker for reverse proxying v-slot without scope on this.$slots
17       if (slot.proxy) {
```

```
export function resolveScopedSlots (
```

```
20   }
21   }
22   if (contentHashKey) {
23     res.$key = contentHashKey
24   }
25   return res
26 }
```

```
2
3 export function resolveScopedSlots (
4   fns: ScopedSlotsData, // see flow/vnode
5   res?: Object,
6   // the following are added in 2.6
7   hasDynamicKeys?: boolean,
8   contentHashKey?: number
9 ): { [key: string]: Function, $stable: boolean } {
10   res = res || { $stable: !hasDynamicKeys }
11   for (let i = 0; i < fns.length; i++) {
12     const slot = fns[i]
13     if (Array.isArray(slot)) {
14       resolveScopedSlots(slot, res, hasDynamicKeys)
15     } else if (slot) {
16       // marker for reverse proxying v-slot without scope on this.$slots
17       if (slot.proxy) {
```

```
21   }
22   }
23   if (contentHashKey) {
24     (res: any).$key = contentHashKey
25   }
26   return res
27 }
```

Git para versionamento de .conf

- Registre no **git** todas as alterações do arquivo de bloqueio.
- Cada técnico ou ferramenta de automação tem que ter **login individual** (quem?).
- Faça o envio (**push**) das mudanças para o repositório central.
- Receba as mudanças (**pull**) no servidor em produção.
- **Automatize** os dois últimos passos.

Automatize a detecção de expiração!

```
grep "EXPIRA:" blocks.conf | awk -F"EXPIRA: " -v t=$(date +%Y-%m-%d) 'substr($2,1,10)  
<= t'
```


Automatize a detecção de expiração!

```
grep "EXPIRA:" blocks.conf | awk -F"EXPIRA: " -v t=$(date +%Y-%m-%d) 'substr($2,1,10)  
<= t'
```

Automatize a detecção de expiração!

```
grep "EXPIRA:" blocks.conf | awk -F"EXPIRA: " -v t=$(date +%Y-%m-%d) 'substr($2,1,10)  
<= t'
```

Automatize a detecção de expiração

```
grep "EXPIRA:" blocks.conf | awk -F"EXPIRA: " -v t=$(date +%Y-%m-%d) 'substr($2,1,10)  
<= t'
```

Tempere seu script à gosto!

A ideia central é

- O que está em vigor está no arquivo de conf.
- O que esteve em vigor está registrado no git:
 - Em qual data cada linha **entrou**.
 - Em qual data cada linha **saiu**.
 - Quem **adicionou**.
 - Quem a **removeu**.

O quê?
Como?
Quando?
Por quê?
Quem?

```
# git log -p -S "streamingpirata.xyz"
commit affc13a6cbf3b3ea9364f7f13d7113524a6d441f (HEAD -> master)
Author: Ayub T. <github@ayub.net.br>
Date: Tue Jul 7 22:35:12 2026 -0300
```

Expirando bloqueios do OFICIO-03

```
diff --git a/blocks.conf b/blocks.conf
index b6d4d29..4896f28 100644
--- a/blocks.conf
+++ b/blocks.conf
@@ -1,3 +1,2 @@
 local-zone: "roleta-burlada.net." always_nxdomain # OFICIO-01 | EXPIRA: 2026-07-06
 local-zone: "streaming-futebol.xyz." always_nxdomain # OFICIO-02 | EXPIRA: 2026-07-07
-local-zone: "streamingpirata.xyz" always_nxdomain # OFICIO-03 | EXPIRA: 2026-07-08
```

```
commit 5f9a4dac0aa4dcb496f9581d5b32afe890cdaac5
Author: Ayub T. <github@ayub.net.br>
Date: Tue Jul 7 22:34:40 2026 -0300
```

Bloqueando endereços do OFICIO-03

```
diff --git a/blocks.conf b/blocks.conf
index 4896f28..b6d4d29 100644
--- a/blocks.conf
+++ b/blocks.conf
@@ -1,2 +1,3 @@
 local-zone: "roleta-burlada.net." always_nxdomain # OFICIO-01 | EXPIRA: 2026-07-06
 local-zone: "streaming-futebol.xyz." always_nxdomain # OFICIO-02 | EXPIRA: 2026-07-07
+local-zone: "streamingpirata.xyz" always_nxdomain # OFICIO-03 | EXPIRA: 2026-07-08
```


A ideia central é

- O que está em vigor está no arquivo de conf.
- O que esteve em vigor está registrado no git:
 - Em qual data cada linha **entrou**.
 - Em qual data cada linha **saiu**.
 - Quem **adicionou**.
 - Quem a **removeu**.

O quê?
Como?
Quando?
Por quê?
Quem?



THIAGO.AYUB@SAGENETWORKS.COM.BR

 WWW.SAGENETWORKS.COM.BR



[sage_networks](https://www.instagram.com/sage_networks)



[Sage Networks](https://www.linkedin.com/company/Sage%20Networks)



[+55 \(19\) 3500-6269](https://api.whatsapp.com/send?phone=551935006269)