interweb

Generated on 22/10/2026 04:10

KNOWLEDGE BASE · PUBLIC DOCUMENT

CAA records: who may issue certificates for your domain

Security

CODE	VERSION	COUNTRY · LANGUAGE
RLX-AO-KB-726	v2026.10.21	Angola · English

Interweb Angola
RLX Angola, Lda. · Group RLX Holdings
Tax ID (NIF) 500 260 5724 · Commercial registration no. 26464-25/250724

https://www.meu.interweb.ao/knowledgebase/726

CAA records: who may issue certificates for your domain

Security Created on **September 27, 2026** Updated on **October 21, 2026** 4 min read



By default, **any** certificate authority in the world may issue a certificate for your domain, as long as it can prove you control it. A **CAA** record is the DNS line that narrows that list: it states publicly which authorities are allowed. One that is not on the list is supposed to refuse.

It is a short, useful record. It is also the record that, written badly, **leaves your site with no padlock** and nobody notices for weeks. This article covers both sides.

> **Read this before you create a CAA.**
> The automatic certificate on your account is issued by **Let's Encrypt**. Publish a CAA that does not include `letsencrypt.org` and automatic issuance stops passing: the certificate already installed stays valid until it expires, and on the day it should renew, it does not. The site starts showing a security warning. See **no padlock: the causes, in order**.

Three fields, and nothing else

In the cPanel zone editor a CAA has three fields. That is all of it.

Field	What it means
Issuer Critical Flag	A flag, `0` or `1`. With `1`, an authority that does not understand this record's tag **must refuse** to issue. The usual value, and the default, is `0`.
Tag	One of three: `issue`, `issuewild` or `iodef`. It is what gives the value its meaning.
Value	For `issue` and `issuewild`, the authority's **domain name**. For `iodef`, an address where refused attempts should be reported.

Tag	What it authorises
`issue`	Ordinary certificates, for named hosts. The common case.
`issuewild`	Wildcard certificates, the ones covering `*.asuaempresa.ao`. If you create no `issuewild` line at all, whatever is in `issue` covers wildcards too.
`iodef`	It authorises nothing: it says where to send the alert when someone tries to issue a certificate this record forbids. The value is an address, typically `mailto:security@asuaempresa.ao`.

What to write, if you decide to write anything

Your zone, as it comes, has **no CAA record at all**. That is not an oversight: with no CAA there is no restriction, and the automatic certificate is issued and renewed without anyone having to think about it. If you decide to add one, you have to think it all the way through.

1. **Start with Let's Encrypt.** One line, tag `issue`, value `letsencrypt.org`. Without it, the certificate that comes with the account stops renewing.

2. **If you bought a paid certificate**, add a second `issue` line with the name its authority tells you to use. **Do not guess that name**: each authority has its own, and it is written in their documentation. See **activating a paid certificate**.

3. **If you use wildcard certificates**, add the matching `issuewild` lines. A domain with `issue` right and `issuewild` forgotten blocks only the wildcards, which is a hard failure to spot.

4. **Add an `iodef` line** pointing at a mailbox somebody actually reads. It is the only warning you will get when something tries to issue in your name.

5. **Verify, and only then forget about it.** The proof is in the next section.

A complete example, with **Name** being the domain itself and **Flag** at `0` on all three lines:

Tag	Value	What it is for
`issue`	`letsencrypt.org`	Lets the account's automatic certificate carry on being issued.
`issuewild`	`letsencrypt.org`	The same, for certificates covering every subdomain.
`iodef`	`mailto:security@asuaempresa.ao`	Where the alert about a refused attempt arrives.

A CAA record does not break the site today.
Visitors never look at it: the authority does, **at the moment of issuing**. So a wrong CAA throws no error on the day you save it. It throws one weeks later, at renewal, when nobody remembers touching a DNS record. If you change a CAA, write the date down somewhere.

When in doubt, publish none.
A domain without CAA is not insecure: it is like the overwhelming majority of domains on the internet. A half-thought CAA is worse than no CAA.

Creating the record in cPanel

1. Open cPanel from My Interweb and go to **Zone Editor**. Click **Manage** next to the domain.
2. **Add Record**, and choose the type **CAA**.
3. Leave **Name** as the domain itself. A CAA on the domain also covers subdomains that have no CAA of their own.
4. Fill in **Issuer Critical Flag**, **Tag** and **Value**, then **Save Record**. One line at a time: three authorisations are three records.

If the domain does not use our nameservers, the record has to be born in the panel that holds them. Check where they point in **what nameservers are and which to use**, and read **switching on SSL at Cloudflare** if you serve the site through Cloudflare, because there the certificate the visitor sees may not be yours at all.

Checking it from outside

System	Command
Windows	`nslookup -type=CAA asuaempresa.ao` at the command prompt
macOS or Linux	`dig CAA asuaempresa.ao +short` in Terminal

The answer gives the flag, the tag and the value of each line. **Nothing** is also a valid answer, and the most common one: it means there is no restriction. Just created it and it is not showing? See **how long propagation takes**.

NEED A HAND?

About to publish a CAA and want us to check the list first? Send us the lines.

Open a support ticket →

SEE ALSO

- SSL certificates →
- Frequently asked questions →
- Hosting plans →

interweb

Interweb Angola

RLX Angola, Lda. · Group RLX Holdings

TAX ID (NIF)	500 260 5724
COMMERCIAL REGISTRATION NO.	26464-25/250724
REGISTERED OFFICE	Rua Direita do Patriota, Bairro Dangereux Belas, Luanda, Angola
CONTACT	info@interweb.ao · +244 942 011 191 · www.interweb.ao
OPENING HOURS	Monday to Friday - 08 AM to 04 PM Saturday & Sunday - Closed

Online version of this document

RLX-AO-KB-726 · v2026.10.21 · Generated on 22/10/2026 04:10