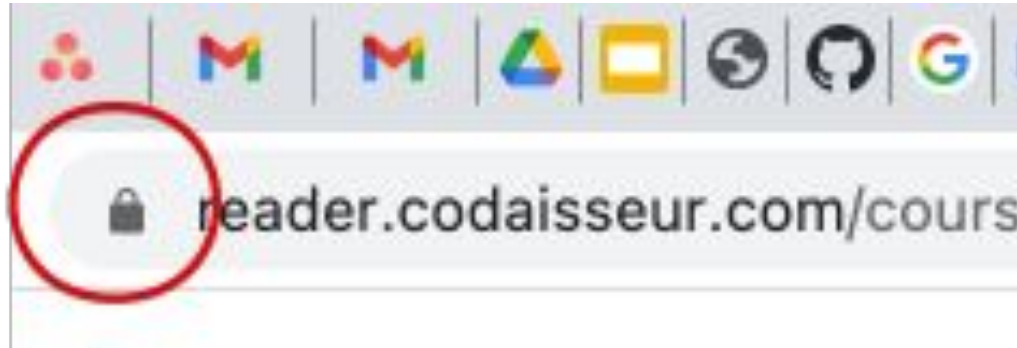


# The Secure Web

SSL, Certificates, and Hostnames



**What does this mean?**



**Why do we  
need **SSL**?**

**Encrypt data sent  
between client and  
server.**

 Why do we need  
a certificate?

**Am I connected to  
the **right** server?**

# A Certificate Says:

- Company owns the domain
- Network traffic is encrypted



**Who owns  
reader.codaisseur.com?**



# Ownership

- Has the root domain (codaisseur.com)
- Manages the DNS for that root domain
- Verified by a signing authority (CA)
- CA's are trusted by browser vendors

# Certificate

-----BEGIN CERTIFICATE-----

MIIFXjCCBEagAwIBAgIRAL4eUiD500amUEJvQyNnr38wDQYJKoZIhvcNAQELBQAw  
gZAx CzAJBgNVBAYTAkdCMRswGQYDVQQIExJHcmVhdGVyIE1hbmNoZXN0ZXIxEDAO  
BgNVBAcTB1NhbGZvc mQxGjAYBgNVBAoTEUNPTU9ETyBDQSBMaW1pdGVkMTYwNAYD  
VQQDEy1DT01PRE8gUINBIERvbWFPbiBWWYXpZGF0aW9uIFNIY3VyZSBTZXJ2ZXI g  
Q0EwHhcNMTcwNzE4MDAwMDAwWhcNMTgwNzMxMjM1OTU5WjBfMSEwHwYDVQQLEXhE  
b21haW4gQ29udHJvbCBWYXpZGF0ZWQxHTAbBgNVBA sTFFBvc2l0aXZlU1NMI Fdp

....

sDx4/Q7IsTIMzt9WQeLJcJMzK9IQrQqrOmxxUx+Q6Mx1qA==

-----END CERTIFICATE-----

## Signed Certificate (by a CA)

-----BEGIN CERTIFICATE-----

MIIFXJCCBEagAwIbAgIRAL4eUiD500amUEJVQYNnr38wDQYJKoZlHvcNAQELBQAw  
gZAxCzAJBgNVBAYTAkdCMRswGQYDVQIEXJHcmVhdGVyIE1hbmNoZXN0ZXlXEDAO  
BgNVBAcTB1NhGZvcmcXqGJAYBgNVBAoTEUNPTU9ETyBDQSBMaW1pdGVMkTYwNAYD  
VQQDEy1DT01PRE8gUINBIERwfbWpbiBwYXpZGF0aW9uIFNlY3VySBZTZXJ2ZXIg  
Q0EWHhcNMTcwNzE4MDAwMDAwWhcNMTgwNzE4MDAwMDAwMjM1MjM1OTU5WjBfMSEwHwYDVQQLExtE  
b21haW4gQ29udHJvbiBwYXpZGF0ZWQwHTAAbG9uBAFFBvc2l0aXZiU1NMFdp

■■■■■

**sDx4/Q7IsTIMzt9WQeLJcJMzK9lQrQqrOmxxUx+Q6Mx1qA==**

-----END CERTIFICATE-----

-----BEGIN CERTIFICATE-----

MIIEAjCCAx6qAwIBAgIBATANBqkghkiG9w0BAQUFADBvMQswCQYDVQQGEwJTRTEU

...

-----END CERTIFICATE-----

-----BEGIN CERTIFICATE-----

MIIFdDCCBFygAwlBAglQJ2buVutJ846r13Ci/ITeljANBgkqhkiG9w0BAQwFADBv

...

-----END CERTIFICATE-----

# Certificate "Chain"

-----BEGIN CERTIFICATE-----

MIIFXJCCBEagAwlBAGlRAL4eUiD500amUEJVQyNnr38wDQYJKoZlHvcNAQELBQAw  
gZAxCAzAJBgNVBAYTAkdCMRswGQYDVQIEXJHcmVhdGVyIE1hbmNoZXN0ZXlXEDAO  
BgNVBAcTB1NhbgZvcmcGxGqJAYBgNVBAoTEUNPTU9ETyBDQSBMaWw1pdGVMkTYwNAYD  
VQQDEy1DT01PRE8gUINBIERvfbWpbiBwYXpZGF0aW9uIFN1Y3VySBTZXJ2ZXlg  
Q0EWHhcNMTcwNzE4MDAwMDAwWhcNMTgwNzNzMjMj1OTU5WjBfMSwHwYDVQQLExtE  
b21haW4gQ29udHJvbiBwYXpZGF0ZWQwHTAaBgNVBAcTFBvc2l0aXZiU1NMFdp

◆◆◆◆

**sDx4/Q7IsTIMzt9WQeLJcJMzK9lQrQqrOmxxUx+Q6Mx1qA==**

-----END CERTIFICATE-----

-----BEGIN CERTIFICATE-----

MIIEAjCCAx6qAwIBAgIBATANBqkghkiG9w0BAQUFADBvMQswCQYDVQQGEwJTRTEU

...

-----END CERTIFICATE-----

-----BEGIN CERTIFICATE-----

MIIFdDCCBFvqAwlBAqlOJ2buVutJ846r13Ci/ITeliANBqkqhkiG9w0BAQwFADBy

...

-----END CERTIFICATE-----

## The certificate itself

## The intermediate certificate

## The root certificate

# Creating a Certificate

```
openssl req -utf8 -nodes -sha256 -newkey  
rsa:2048 -keyout www.example.com.key -out  
www.example.com.csr
```

# Creating a Certificate

- Create a Certificate Signing Request (CSR) and an encryption key
- Find a certificate vendor and provide the CSR
- Prove that you own the domain, entity exists, etc.
- Get a certificate
- Get the intermediate and root certificates from the vendor
- Provide your server with the certificates and the key file

# A Self-Signed Certificate

- Create a Certificate Signing Request (CSR) and an encryption key
- ~~● Find a certificate vendor and provide the CSR~~
- ~~● Prove that you own the domain, entity exists, etc.~~
- Get a certificate
- ~~● Get the intermediate and root certificates from the vendor~~
- Provide your server with the certificates and the key file
- **Accept that the certificate's authority is not trusted by browsers etc.**



Why would you  
use a **self-signed**  
certificate?



```
$ openssl \  
req -x509 \  
-newkey rsa:4096 \  
-keyout key.pem \  
-out cert.pem \  
-days 365 \  
-nodes
```



Why would you **NOT**  
use a **self-signed**  
certificate?

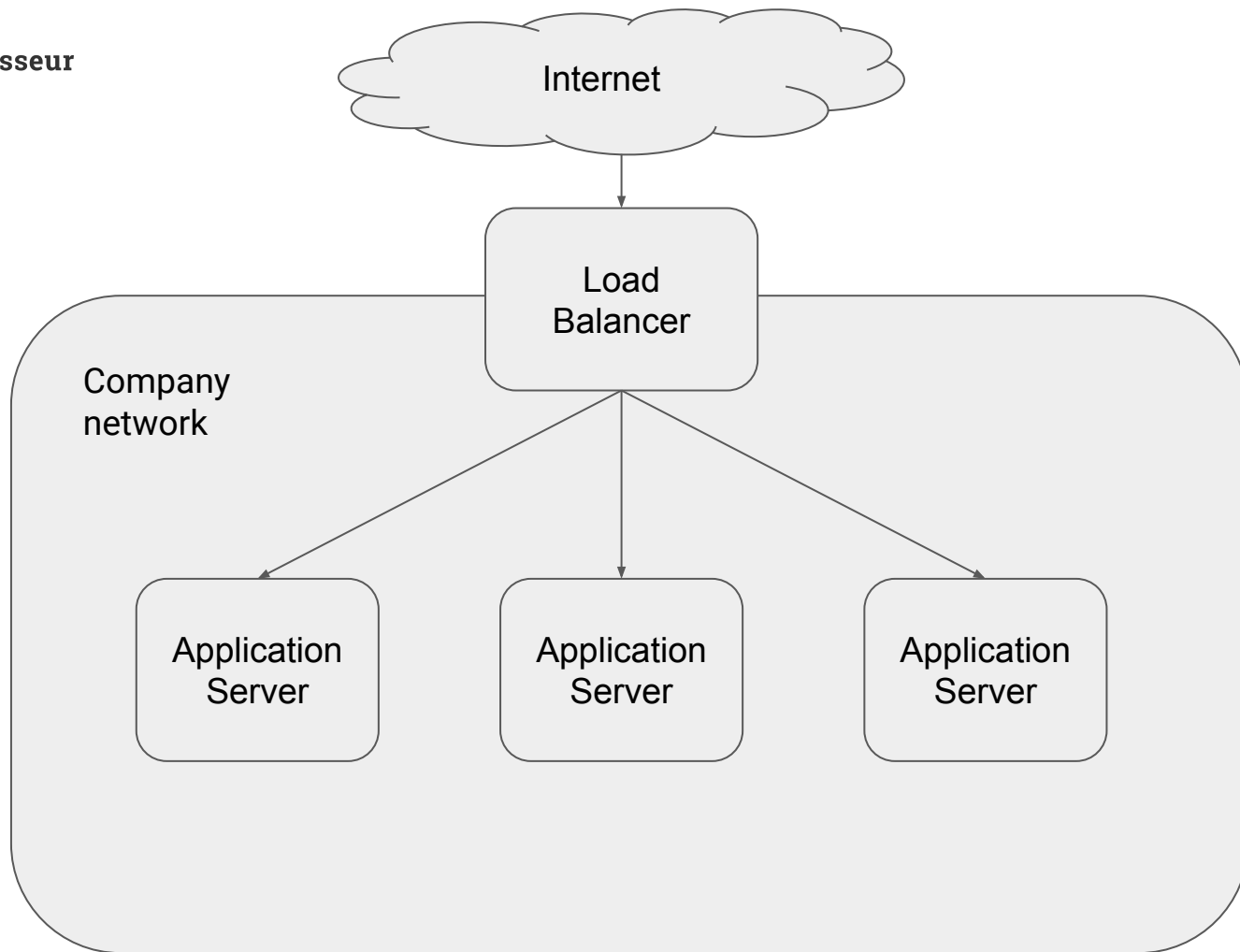


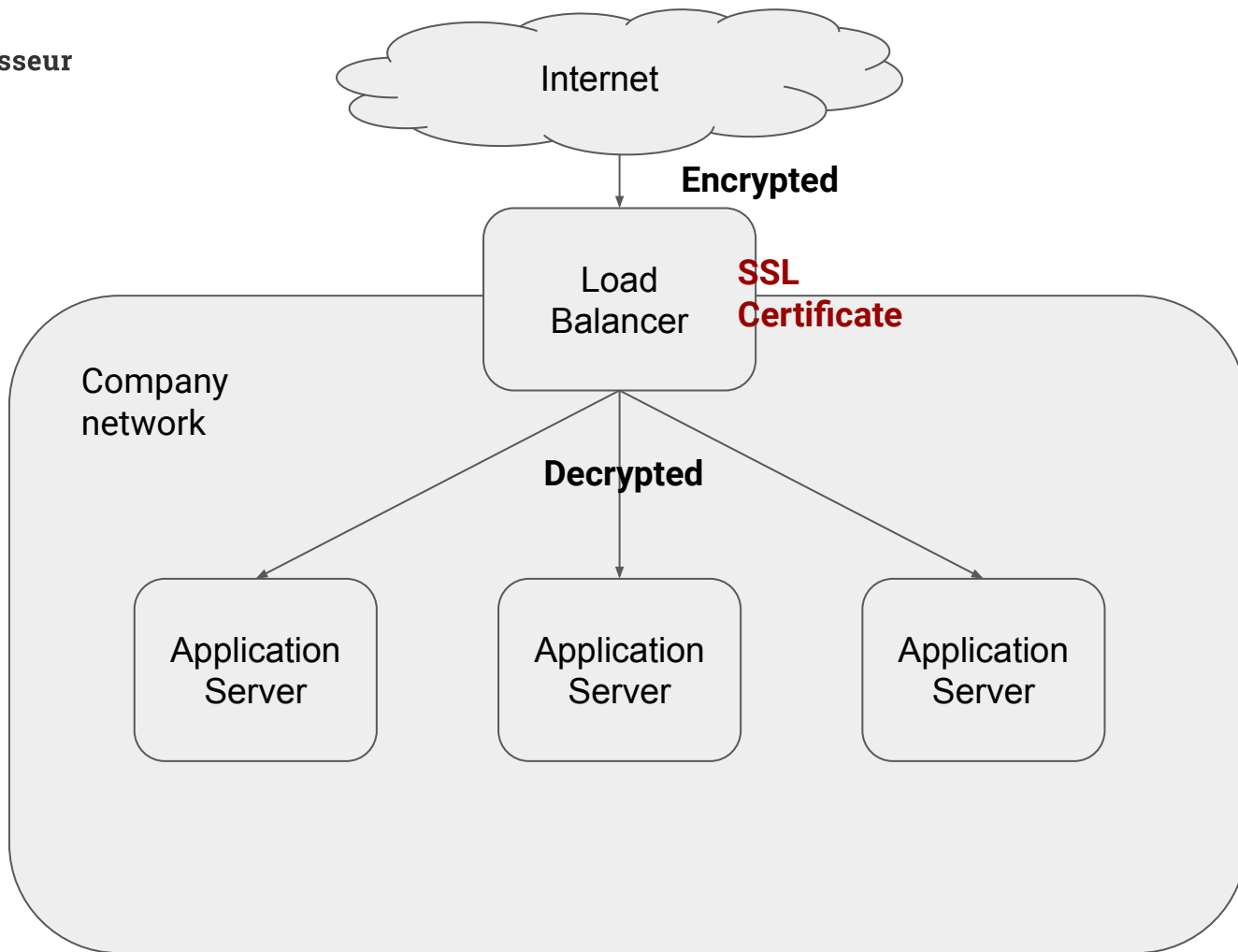
Use **LetsEncrypt** instead

# Common Practice

SSL Termination

Here is a **typical**  
setup

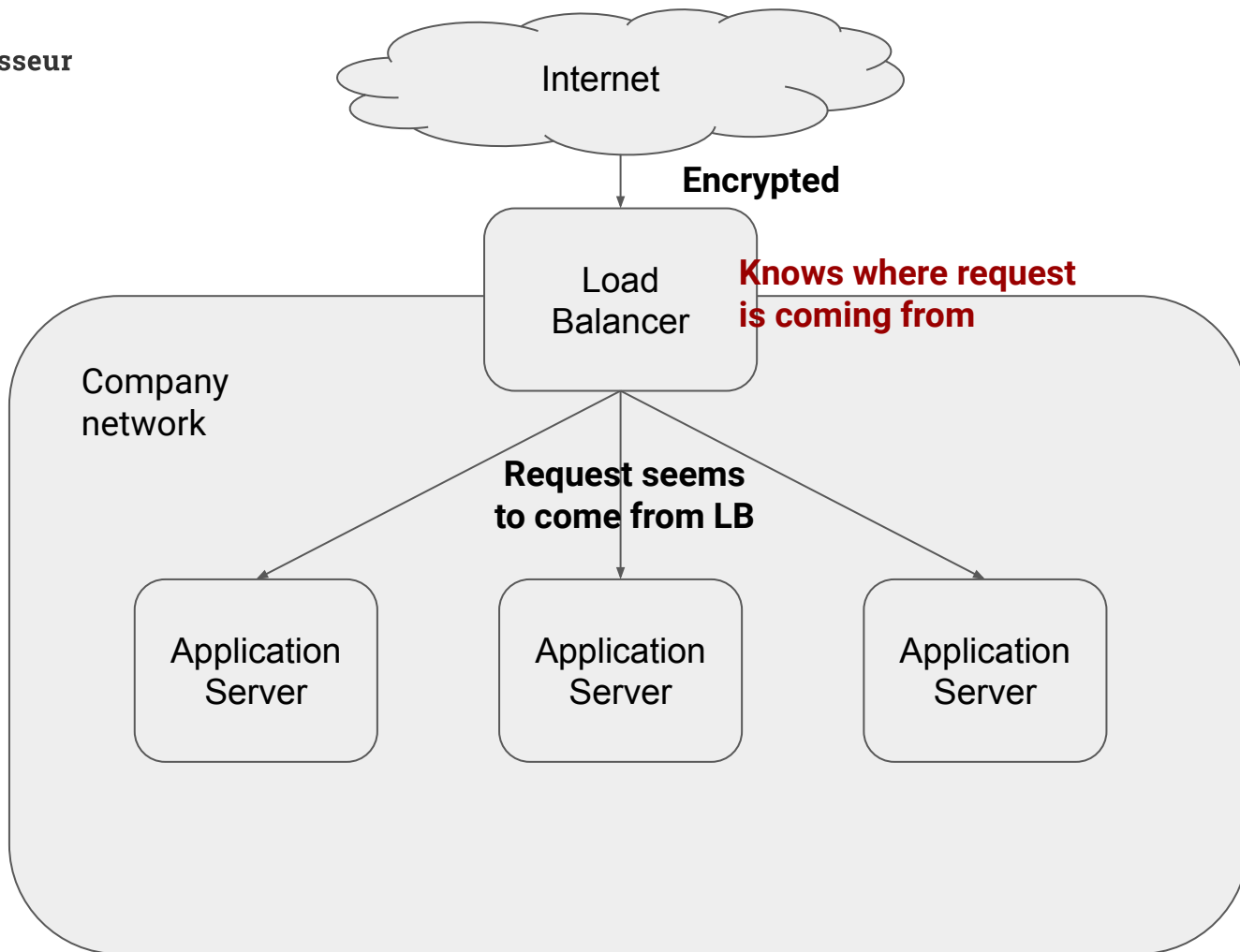


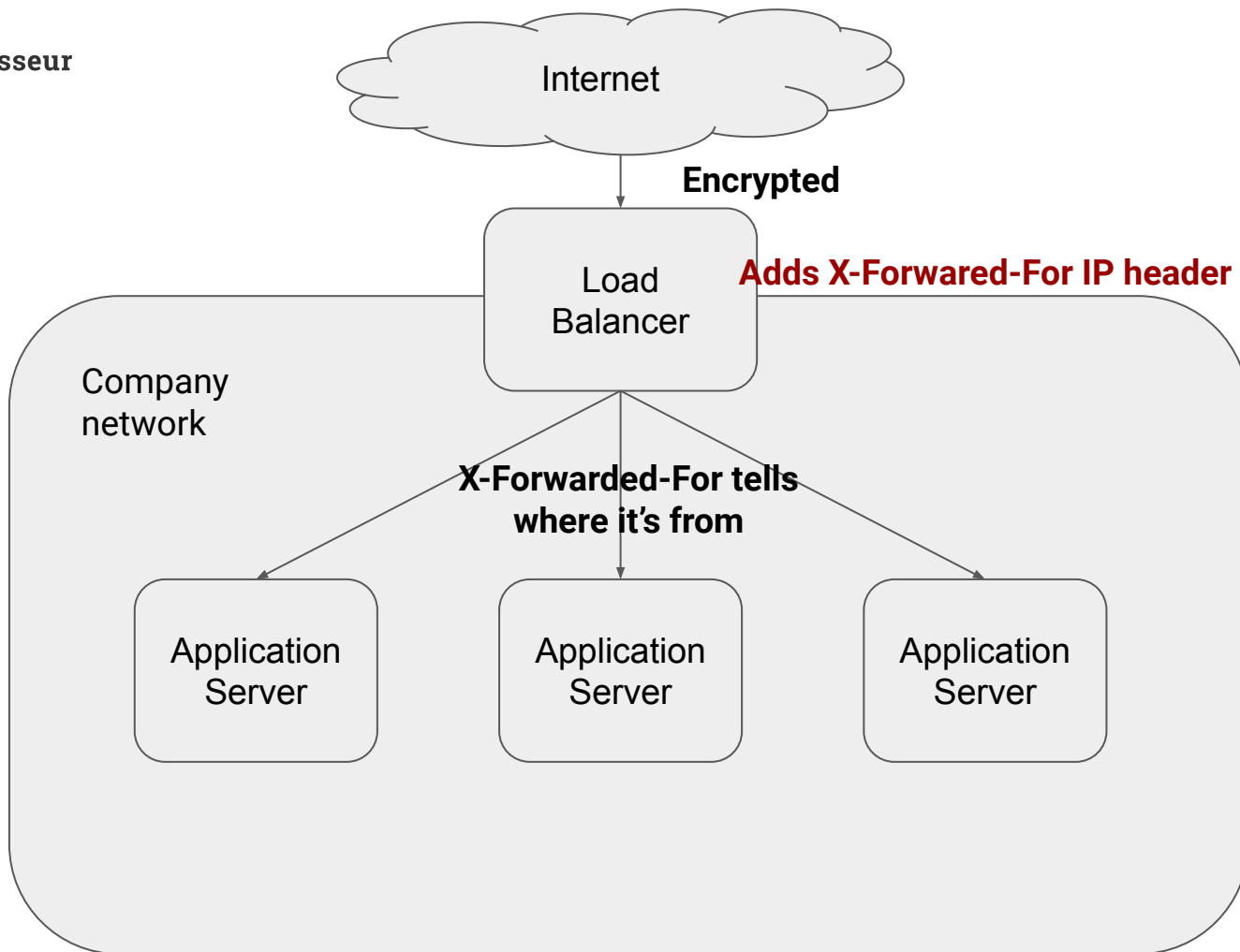


 When might this  
turn into a **security**  
issue?



 Why is this  
**common** practice?





**Also annoying:  
Wireshark is blind  
for SSL (try [this](#))**