Universität Basel

Let's Encrypt @ Uni Basel

Hanspeter Spalinger, 29.03.17

Content

– SSL in a nutshell
    - Why it sucks

– Let's Encrypt
    - How to fix SSL

– Situation at Uni Basel
    - What we are doing

SSL in a nutshell

Pain points

– Manual setup

- csr/key generation
- openssl req -new -sha256 -newkey rsa:4096 -nodes -keyout mykey.pem -out myreq.pem -subj 'CN=mydomain.unibas.ch/OU=ITS/O=Universitaet Basel/L=Basel/ST=Basel-Stadt/C=CH'
- Hard to use Interfaces at CA

– Time intensive

- How do I do this?
- How did my predecessor did this 3 years ago?

– Expensive

- Not really a problem for me
- Your mileage may vary

SSL in a nutshell

Solution: Only implement SSL when it is „required“

– Organizations policy mandated

– Admins try to prevent SSL wherever possible

– Long Certificate validity

– Works for me

SSL in a nutshell

Real world moved on

– Man in the Middle got real
    - Revolutions (Egypt,Libya,...)
    - The good guys are doing it too (NSA,CIA,BND,GCHQ,...)
    - Even inside YOUR network (google: „Fuck these guys“)

– Bugs got real (heartbleed,...)
    - Turnover time is getting smaller
    - Certificates are usually replaced before 3 years

– Google increases search-rank for SSL-enabled pages

SSL in a nutshell

Real Solution

– We want ALL Traffic encrypted

– We want it SIMPLE

– We want it SECURE

– We want it FREE

– We want it AUTOMATED

Let's Encrypt

automated, free, easy-to-use CA

– Extended validation takes to much time
    - User doesn't care anyway

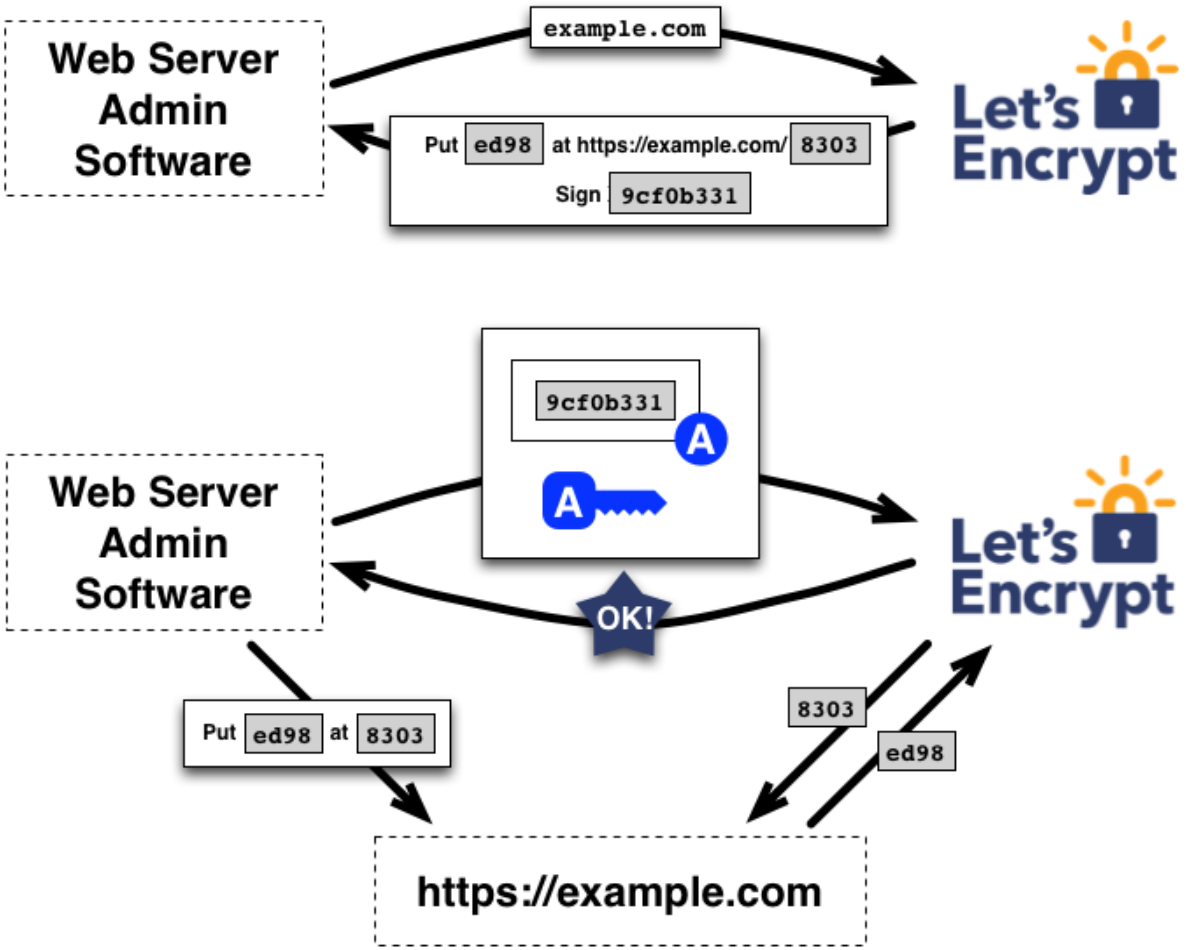
– Domain validated Certificates only

– Prove of domain ownership can be automated

– Let's encrypt is a collection of Software (Open Source) and Protocols (ACME)
    - you (or a CA) can run your own „Let's Encrypt“

Let's Encrypt

How to verify a domain ownership ... Let's use a Webserver



Let's Encrypt

So what's the problem with Webserver?

– Needs to be done on the Webserver itself

– Can not be done over SSL.

```
listen  80;
location / {
    return  301 https://$host$request_uri;
}
location /.well-known/acme-challenge {
    alias /var/www/dehydrated;
}
```

Let's Encrypt

How to verify a domain ownership ... Let's use DNS

– Put the nonce [ed98] into _acme-challenge.example.com TXT Record

– Tell Let's Encrypt to verify

– Get your Certificate

– https://letsencrypt.org/how-it-works/

Let's Encrypt

Webserver	vs	DNS verification
– Requires a public Webserver • internal services?		– Requires a public DNS
– Must/Should be done on the server		– Can be done everywhere • can be centralized
– Requires a Webserver • even without web content		– Can be done for any software • Mailserver • Custom Protocols

Situation at Uni Basel

History

– until ~2016 we used SAN Zertifikates on the webserver

- max. 50 SANs per server
- if we want www.<domain> too we actualy got 24 domains per server
- Would like to have dav.<domain> for file transfer. down to 16 domains
- See all domains hosted on the server
- „foreign domains“ could not be included
- Certificate revocation had to be managed somehow

– Now we use SNI with single Cert for every domain

- 1 Common Name + 2 SAN (www. + dav.)
- nginx to Offload SSL. Can easyily host 100+ domains on a single server
- Change of Certificate is much easier
- We can manually add „foreign domains“, but its ugly

Situation at Uni Basel

Pseudo Automatisation (ugly)

– Ansible based server managment which sends CSR to Switch
    - curl to https://www.switch.ch/pki/manage/request/
    - Fails if switch changes the website (did not happen so far)

– Approve it in the QuoVadis Web Interface (3+ clicks)

– Cron Job scans mails and deploy the script on the webserver
    - Using regular expression to parse mail
    - Using regular expression to parse switch download page

Situation at Uni Basel

End of 2016 we moved to Let's Encrypt for public webservers.

– About 350 domains

– Currently using Webserver verification

– Fully automated

- I don't care about SSL on public domains anymore.
- It just works.

– We use a shell-based Let's Encrypt script

- https://github.com/lukas2511/dehydrated

Situation at Uni Basel

Whats next?

– Move to DNS verification so we can get internal Certificates

– Provide SSL to all of our server customers

- System management provides Let's Encrypt enabled nginx
- Sends work to apache/tomcat/whatever application server

– I would love to see a ACME Protocol implementation at more/all CAs

- Single Provider for EV-,DV- and User-Certificates

That's all

Questions?