





VERISIGN

# Outline

- What is DANE?
- The TLSA Record
- TLSA Browser Plugin
- Generating the TLSA Record
- Other uses for DANE



VERISIGN®

# DNS-Based Authentication of Named Entities (DANE)





VERISIGN

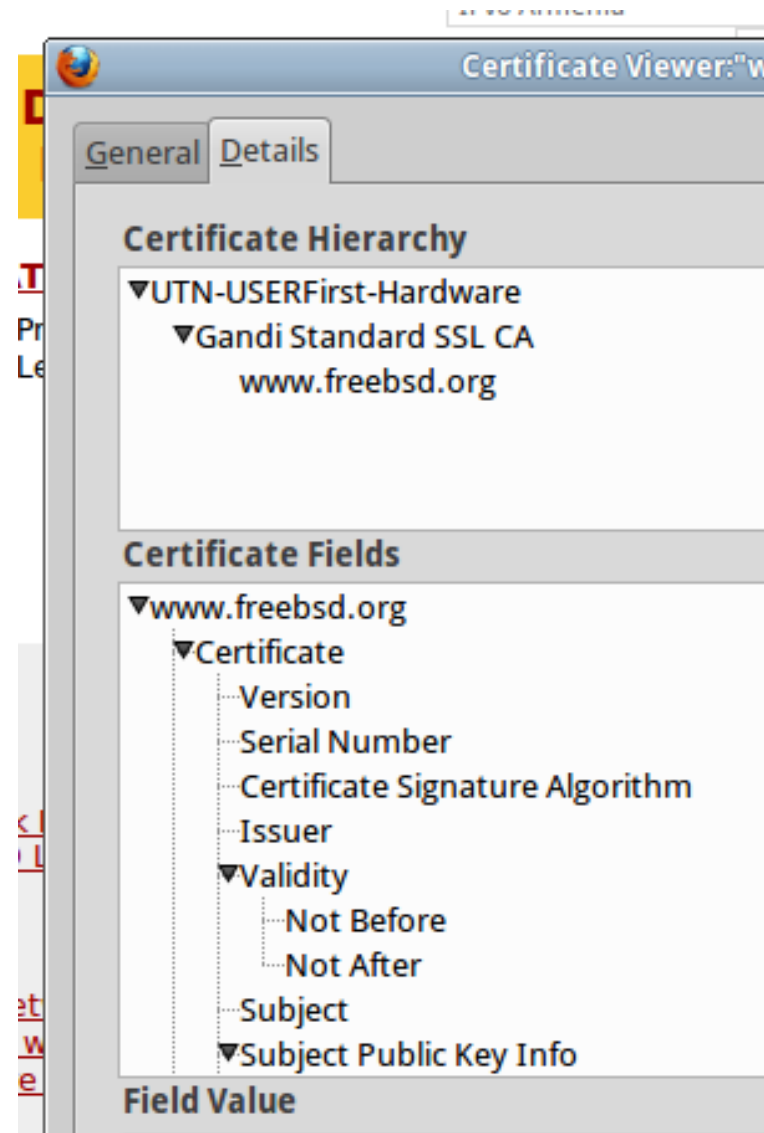
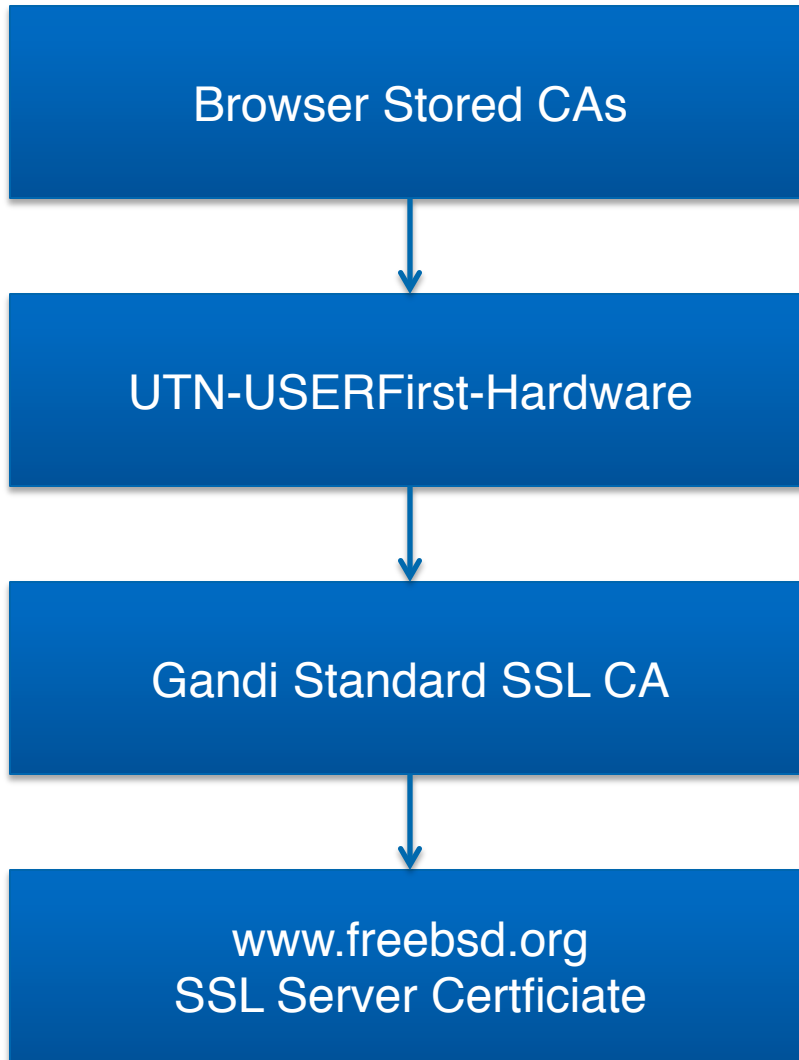
# What is DANE?

- A suite of RFCs describing how to represent and authenticate “named entities” in DNS and DNSSEC.
- Named Entities:
  - Web sites, and other servers
  - Email addresses
  - Jabber/Chat IDs
- RFC 6394 “Use Cases and Requirements for DNS-Based Authentication of Named Entities (DANE)”
- RFC 6698 “The DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS) Protocol: TLSA”
- Internet-Drafts: S/MIME, SMTP, IPSEC, PGP, OTR



VERISIGN

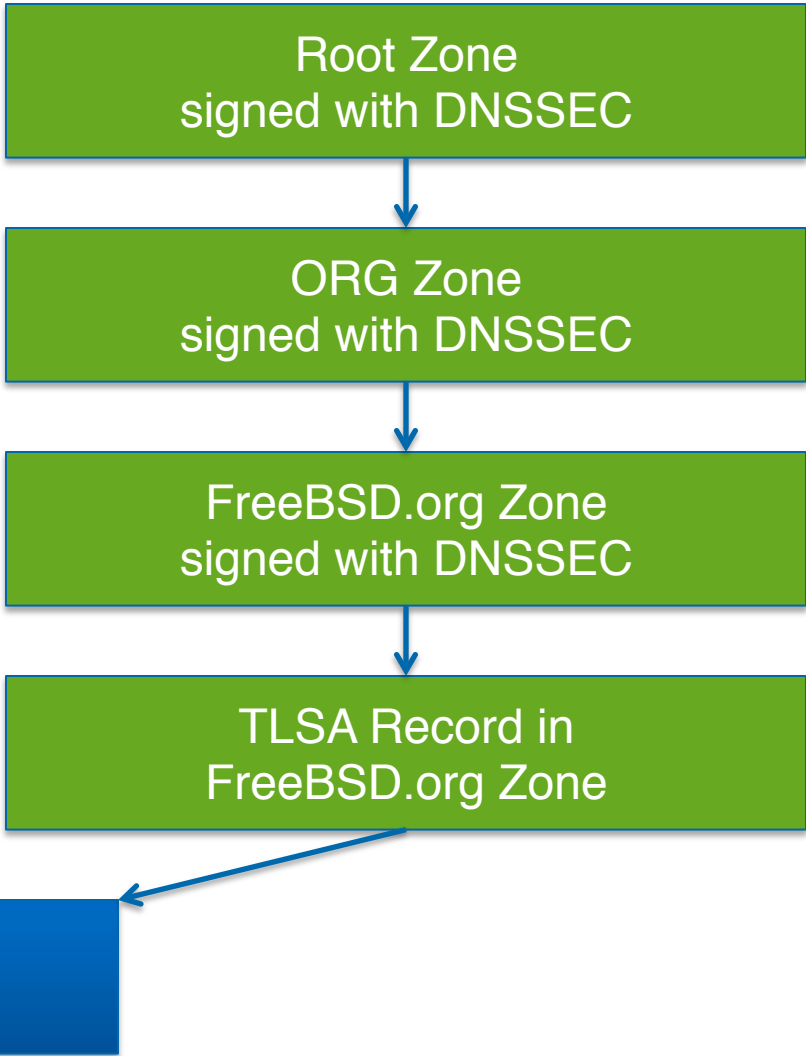
# The CA Way





VERISIGN

# The DANE/TLSA Way





VERISIGN

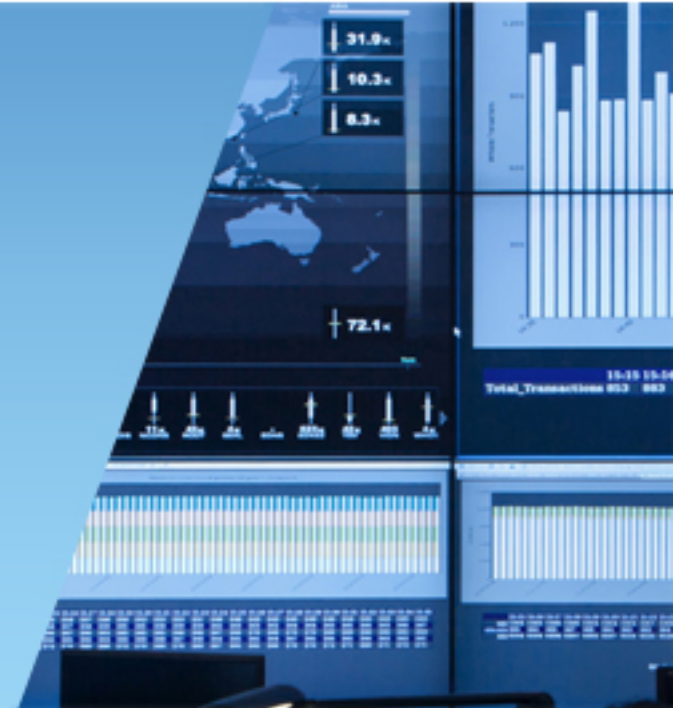
# The DANE/TLSA Way

- TLSA can be used in other ways as we'll see next
  - Match only the public key part of a cert
  - Match a CA, rather than server cert
  - Specify new CA trust anchor



VERISIGN®

# The TLSA Record







VERISIGN

# TLSA Record Names

- Similar to SRV records
- Prefixed with port and protocol labels
- For example, the TLSA record for `https://www.freebsd.org` is at:

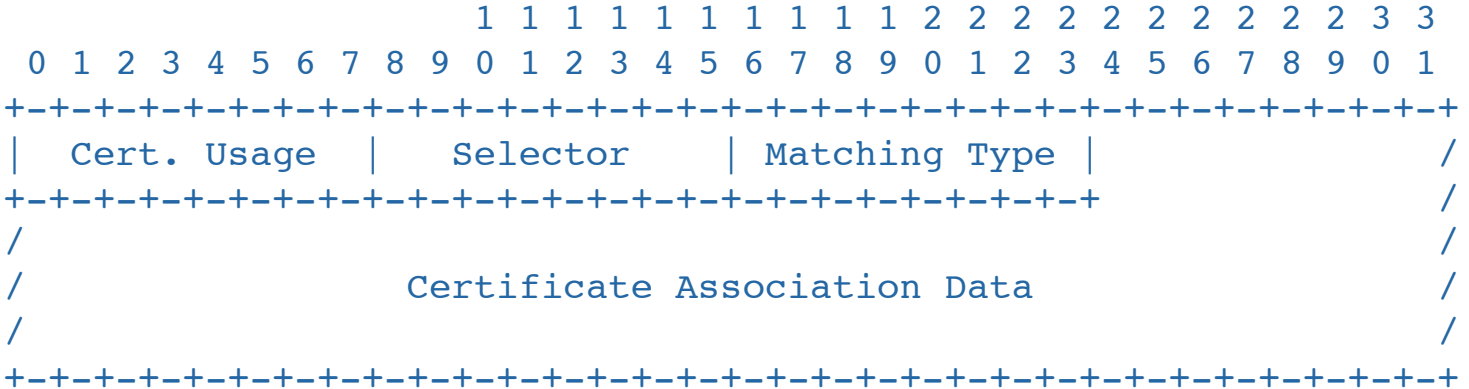
```
_443._tcp.www.freebsd.org.
```

- For SMTP submission, perhaps:

```
_587._tcp.mx1.freebsd.org.
```



# TLSA Wire Data Format





# TLSA Record Fields

- **Certificate Usage**
  - 0 = “CA constraint”
  - 1 = “service certificate constraint”
  - 2 = “trust anchor assertion”
  - 3 = “domain-issued certificate”
- **Selector**
  - 0 = full certificate
  - 1 = public key only
- **Matching Type**
  - 0 = exact match
  - 1 = match SHA-256 hash
  - 2 = match SHA-512 hash
- **Certificate Association Data**
  - full certificate, or public key data, or hash value



## Certificate Usage 0 – “CA constraint”

- The server’s cert must be validated by the CA cert referenced in the TLSA record.
- The CA cert must be part of the client’s normal set of trusted authorities.
- Any valid cert from the CA will be accepted.
  - But not other CAs known to the application.
  - Can get a new cert from same CA without updating TLSA record.



VERISIGN

## Certificate Usage 1 – “service certificate constraint”

- The server’s cert must match the cert referenced in the TLSA record.
- The cert must validate via the application’s standard PKIX mechanisms.
- Must update TLSA record each time server’s cert changes.



VERISIGN

## Certificate Usage 2 – “trust anchor assertion”

- Specifies a new trust anchor for validating the server’s cert.
- Similar to usage 1, except the trust anchor need not be previously known to the application.



VERISIGN

## Certificate Usage 3 – “domain-issued certificate”

- Server’s cert must match the TLSA record.
- No other CA/PKIX validation necessary.



VERISIGN

# Size of Certificate Association Data

- SHA256 – 32 octets
- SHA512 – 64 octets
- Public (e.g. RSA) key – varies
  - 1024 bits – 128 octets
  - 2048 bits – 256 octets
  - 4096 bits – 512 octets
- Full certificate – varies
  - 4096 bit RSA key – approx 1400 octets





VERISIGN

## For Today...

- Certificate Usage = 3 (domain-issued)
- Selector = 0 (full certificate)
- Matching Type = 1 (SHA256 hash)



VERISIGN®

# TLSA Browser Plugin





VERISIGN

# CZ.NIC's DNSSEC/TLSA Validator add-on for Web Browsers

- <https://www.dnssec-validator.cz/>
- Indicates DNSSEC status
- Indicates TLSA status
- Works with:
  - Internet Explorer
  - Mozilla Firefox
  - Google Chrome
  - Opera
  - Apple Safari



VERISIGN

# Add-on Setup Screen

The screenshot shows the Firefox Add-ons Manager interface. On the left, there is a sidebar with navigation options: Get Add-ons, Extensions, Appearance, Plugins, and Services. The main content area displays the details for the 'DNSSEC/TLSA Validator 2.1.1' add-on by CZ.NIC Labs. The add-on's icon is a green key. A preview image shows a Firefox browser window with a notification from the add-on indicating that the domain 'www.nic.cz' is secured by DNSSEC. The description explains that the add-on checks DNSSEC security and TLSA records. Below the description, there are settings for 'Automatic Updates' (set to Default), 'Last Updated' (March 20, 2014), 'Homepage' (http://www.dnssec-validator.cz/), and 'Rating' (4.5 stars from 18 reviews). At the bottom, there are buttons for 'Preferences', 'Disable', and 'Remove'.

## DNSSEC/TLSA Validator 2.1.1

By CZ.NIC Labs

Check DNSSEC security of domain names and check TLSA records if exist.

DNSSEC Validator gets DNS records for a domain name used in page address and compares them to IP addresses Firefox used to download the page. If the records contain DNSSEC signatures which can be validated, the user is protected by DNSSEC. Otherwise the user could have been a victim of DNS spoofing. The result of the comparison is displayed as green/orange/red key right in the address bar.

DNSSEC Validator uses external library to resolve and validate DNSSEC signatures.

Automatic Updates  Default  On  Off

Last Updated March 20, 2014

Homepage <http://www.dnssec-validator.cz/>

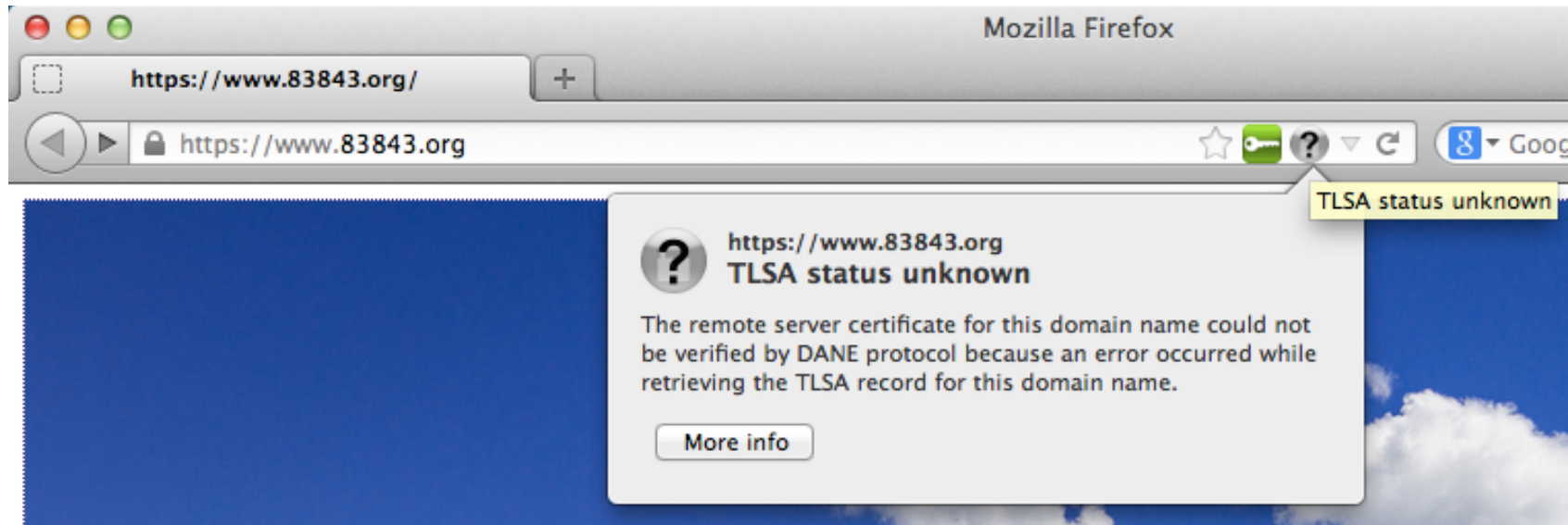
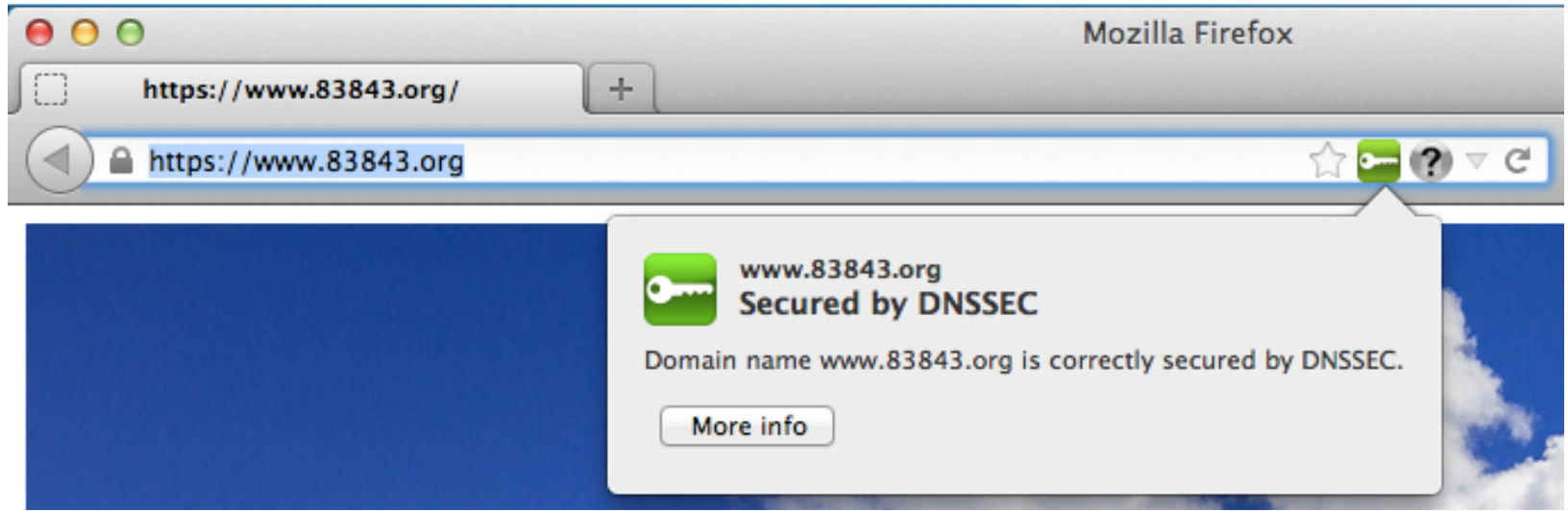
Rating 18 reviews

Preferences Disable Remove



VERISIGN

# Zone is signed, but no TLSA record yet





VERISIGN®

# Generating the TLSA Record







# First we need the certificate...

```
$ cat /etc/apache2/ssl/www.83843.org.crt
-----BEGIN CERTIFICATE-----
MIIEYDCCAkgCARcwDQYJKoZIhvcNAQEFBQAwgZgx CzAJBgNVBAYTA lVTMQswCQYD
VQOIEwJJRDEPMA0GA1UEBxMGTW9zY293MRcwFQYDVQQKEw5QYWNRZXQgUHVzaGVy
czELMAkGA1UECxmCQ0ExGjAYBgNVBAMTEVBhY2tldCBQdXNoZXJzIENBMSkwJwYJ
KoZIhvcNAQkBFhp3ZXNzZWxzQHhY2tldC1wdXNoZXJzLmNvbTAeFw0xNDAzMjEx
NjQ4MzBaFw0yMzEyMTkxNjQ4MzBaMFmx CzAJBgNVBAYTA lVTMQswCQYDVQOIEwJ
RDEPMA0GA1UEBxMGTW9zY293MQ4wDAYDVQQKEwU4Mzg0MzEWMBQGA1UEAxMNd3d3
LjgZODQzLm9yZzCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBBAJ7GES+C
ROT0cbylBSpp+29iBunyD286/B6REv4azvMq/l4alvMwTi+OxZsjkc1IO0ghgWgr
V1gIME65LtPo5qlqUp+MF/MdOn0BYRYKcDfYWhHh8+Ng5YZJ0GnZcZQdZB370O5L
7tJhF9NB38wKlpm/z7zVst1Eil2GE+0DfY2ME5wzkg1K7a7dzSdMICyKqkCZF5z
KkE6MZRQKrKRUn8UfQ8gUizyZK2tkrcd4Fjfg7z38vRnStc7OWvDMET1XzBWN1n/
u0y5zmOaGsyke4klobDYK910XvJn8gJ2kAUPtx14ktpRie0nJcJ5nxD3kFPKI6q8
wqC+Cu5rP6PZrMMCAwEAATANBgkqhkiG9w0BAQUFAAOCAgEAAh3/5zkdwjb0+rr
Sb+b+IOZxoxTBNE49wwCYg++DZpKJOvrz0JWE0cJVpiqyluwmMcntlGekf9pF2Yl
shOoFBo00lVQ9JGyT0q1uju809p5qpw+wM5gytemnSSg01/Acy+AezTz8VqduHlx
ne+uzVQNeDa60ezQENjKSzJdTOGFYy10a8e/xznlGXVdRUbHDHXHERIcYQgd8aDv
HkJvGhkjv25vyZhMwo9NmCuyqmWEKcUePTtTBBQwURLFg/DZ+/WHMSVnLUZ1tiCF
MDs7TrJBAvznes7AM6zkFnZJf2shCVfit5pQ1Or9ZR/ONUyD/2SYoy5604yamBuj
HuyAKObnNYoZVzzfqWGL+77AtQIdpJwoXMeIdB7mEcdND6jBTtTDDN0V3sz+ds2m
CryCawl8t1x/4jfnWTW7CVDRn2TPH/TwyG32ag1XTKHw8nTq/umLgxwfCufs9ltJ
+tWDPQgStD2926V+lZRQl323GXTgu4UaNHvdNDW9e43rYChZfX48YsuDqMf6yQO1
Lgn6Cuev0s7LEN5iY6uVkrX0U+Y/HOKRiWrsJSD+e8FfL8EZ7JiIj235ZG02GKFP
aN26PPyLWK190Km5B9A9yUFSDdJGZZ40cLlz8Jt0KQwwomfBji7Iu/ujMB8M23Ng
SRIG3sP95b9Hq0Ce6Ok1BKdbt5U=
-----END CERTIFICATE-----
```



VERISIGN

# Tools to Generate TLSA Records

- <http://imgtfy.com/?q=generate+tlsa>
- [https://www.huque.com/bin/gen\\_tlsa](https://www.huque.com/bin/gen_tlsa)
- <http://people.redhat.com/pwouters/hash-slinger/>
- [https://github.com/shuque/tlsa\\_rdata](https://github.com/shuque/tlsa_rdata)





# Input to Shumon's TLSA Generator

Generate TLSA Record

Generate DNS TLSA resource record from a certificate and given parameters.

**Usage Field:**

- 0 - CA Constraint
- 1 - Service Certificate Constraint
- 2 - Trust Anchor Assertion
- 3 - Domain Issued Certificate

**Selector Field:**

- 0 - Use full certificate
- 1 - Use subject public key

**Matching-Type Field:**

- 0 - Full Contents
- 1 - SHA-256 hash
- 2 - SHA-512 hash

**Enter/paste PEM format X.509 certificate here:**

```
HkJvGhkjv25vyZhMwo9NmCuyqmWEKcUePtTBBQwURLFg/DZ+/WHMSVnLUZ1tiCF
MDs7TrJBAvznes7AM6ZkFnZJf2shCVfit5pQ1Or9ZR/ONUyD/2SYoy5604yamBuJ
HuyAKObnNYoZVzzfgWGL+77AtQIdpJwoXMeIdB7mEcdND6jBTTDDN0V3sz+ds2m
CryCaw18t1x/4jfnWTW7CVDRn2TPH/TwyG32ag1XTKHw8nTq/umLgxwfcuFs9ltJ
+tWDPQgStD2926V+1ZRQ1323GXTgu4UaNhvdNDW9e43rYChZfX48YsuDgMf6yQ01
Lgn6Cuev0s7LEN5iY6uVkrX0U+Y/HOKRiWrsJSD+e8FfLBEZ7JiIj235ZG02GKFP
aN26PPyLWK190Km5B9A9yUFSdJGZ240cLlZ8Jt0KQwwomfBjI7Iu/ujMB8M23Ng
SRIG3sP95b9Hq0Ce6Ok1BKdbt5U=
-----END CERTIFICATE-----
```

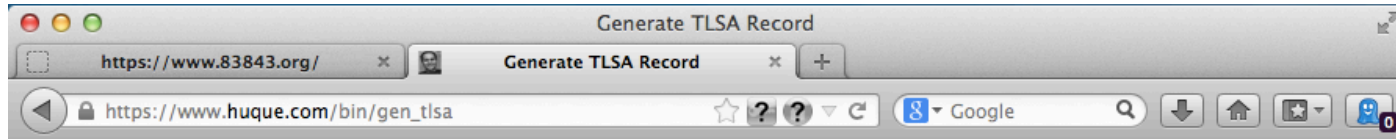
Port Number:  (eg. 443)

Transport Protocol:  (eg. tcp, udp, sctp, dccp)

Domain Name:



# Output from Shumon's TLSA Generator



## Generate TLSA Record

Generate DNS TLSA resource record from a certificate and given parameters.

### Certificate Information:

Serial : 17  
Issuer : C=US, ST=ID, L=Moscow, O=Packet Pushers, OU=CA, CN=Packet Pushers  
CA/emailAddress=wessels@packet-pushers.com  
Subject: C=US, ST=ID, L=Moscow, O=83843, CN=www.83843.org

### TLSA Parameters:

Usage: 3 - Domain Issued Certificate  
Selector: 0 - Full Certificate  
Matching Type: 1 - SHA-256 Hash

### Service Parameters:

Port: 443  
Transport: tcp  
Domain name: www.83843.org.

### Generated DNS TLSA Record:

```
443._tcp.www.83843.org. IN TLSA 3 0 1  
12f4e90c509bf11748a9def05daad4a6435ed915addcc2e7b25e2fc713743fab
```

[Generate another TLSA record?](#)



# Add TLSA Record to Zone

- Edit zone file, etc
- Wait an appropriate amount of time for cache expiry
- Check it with dig

```
; <<>> DiG 9.9.5 <<>> +norec @173.230.152.222 _443._tcp.www.83843.org tlsa
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 11434
;; flags: qr aa; QUERY: 1, ANSWER: 1, AUTHORITY: 2, ADDITIONAL: 4

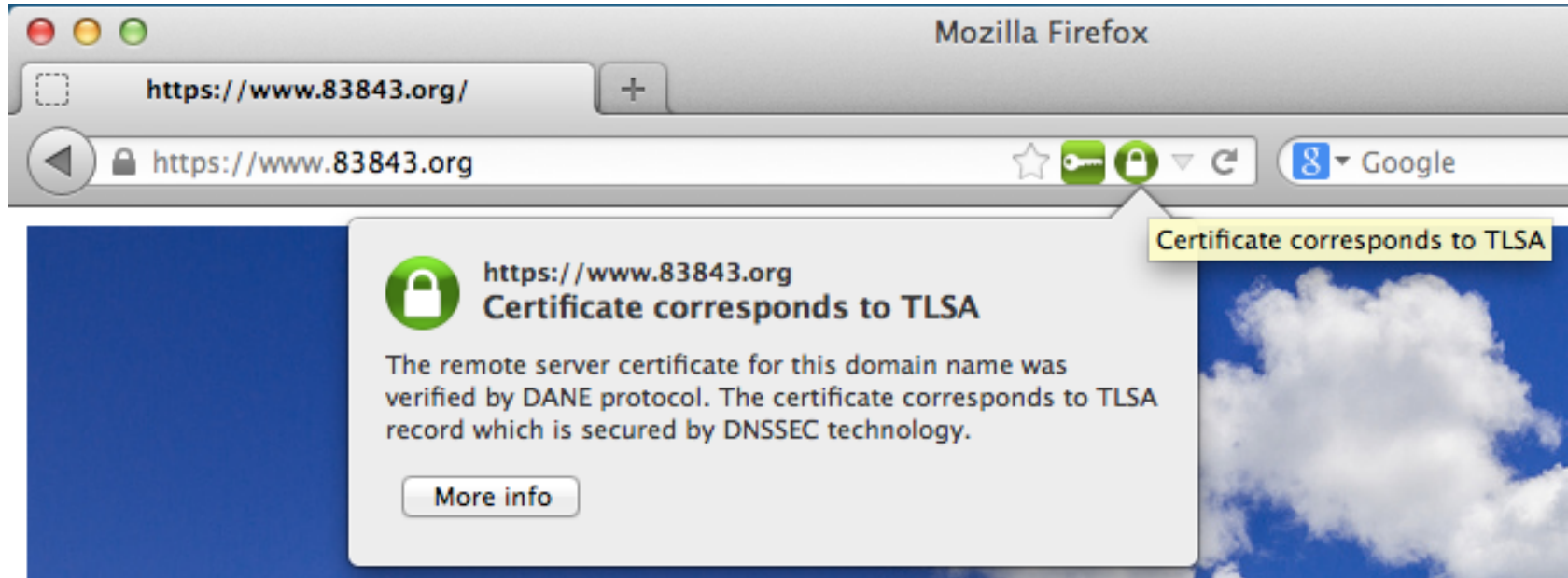
;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;_443._tcp.www.83843.org.      IN          TLSA

;; ANSWER SECTION:
_443._tcp.www.83843.org. 15      IN          TLSA      3 0 1
12F4E90C509BF11748A9DEF05DAAD4A6435ED915ADDCC2E7B25E2FC7 13743FAB
```



VERISIGN

# Check Browser and Add-on





VERISIGN®

# Other Uses for DANE/TLSA





VERISIGN

## Postfix (SMTP)

- Postfix now looks for TLSA records
- [http://www.postfix.org/TLS\\_README.html#client\\_tls\\_dane](http://www.postfix.org/TLS_README.html#client_tls_dane)



VERISIGN

# S/MIME

- draft-ietf-dane-smime
- New SMIMEA record type
- Records are named with hash of user part of email address:

```
3f51f4663b2b798560c5b9e16d6069a28727f62518c3a1b33f7f5214._smimecert.example.com
```



VERISIGN

# OpenPGP Keys

- draft-wouters-dane-openpgp-02
- New OPENPGPKEY record type
- openpgpkey-milter
  - “attempt to automatically PGP encrypt plaintext emails received by the MTA/MUA before relaying the message further towards the recipient(s).”





VERISIGN

# Off-the-Record Messaging Protocol

- **draft-wouters-dane-otrfp-01**
- **New OTRFP record type**



# Thank You

© 2013 VeriSign, Inc. All rights reserved. VERISIGN and other trademarks, service marks, and designs are registered or unregistered trademarks of VeriSign, Inc. and its subsidiaries in the United States and in foreign countries. All other trademarks are property of their respective owners.



**VERISIGN®**