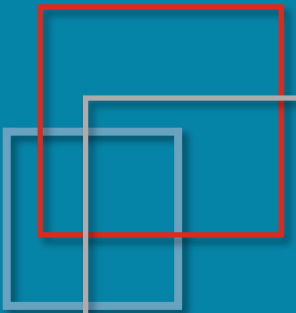


これからのメールセキュリティ

～ 暗号編 ～

JPAAWG / 株式会社クオリア

平野善隆



自己紹介

名前

平野 善隆

所属

株式会社クオリア
チーフエンジニア

資格等

Licensed Scrum Master
Certified Scrum Developer

主な活動

M³AAWG
JPAAWG
IA Japan 迷惑メール対策委員会
迷惑メール対策推進協議会
メッセージング研究所(MRI)
Audax Randonneurs Nihonbashi



メールとの関わり

1990	パソコン通信などでメールに触れる
199x	ドメインを取得して近所のISPに個人のサーバーを置かせてもらって運用開始
2000	外人さんの多い会社に転職したのでメールの漢字にふりがなを付けたりして遊ぶ (のちのhiragana.jp)
	個人のサーバーをちゃんとしたデータセンターに移動。imail.ne.jpというドメインを取って一攫千金を夢見るが挫折
2004	メールの会社に入社
以降	スパムフィルタ、誤送信防止製品の開発やサービスの立ち上げ。PPAPの礎を築く。



もくじ

- メールセキュリティ 世界と日本
- 盗聴・なりすまし受信から守る
 - STARTTLS
 - MTA-STS
 - TLS-RPT
 - DANE for SMTP
 - Require TLS
 - DANE for S/MIME
- まとめ

メールセキュリティ

世界と日本

日本は周回遅れで滅びる！



経済・企業 サイバー攻撃で滅びる日本

偽メールに騙される大企業 対策は世界に周回遅れ＝山崎文明

   2020年10月26日

<https://weekly-economist.mainichi.jp/articles/20201103/se1/00m/020/061000c>

オランダの場合

Meting Informatieveiligheidsstandaarden overheid maart 2020

(政府情報セキュリティ基準の測定2020年3月)

Implementatie-deadline	Betreffende standaarden	
遅くとも 2017年末まで	uiterlijk EIND 2017	DNSSEC SPF DKIM
遅くとも 2018年末まで	uiterlijk EIND 2018	DMARC
遅くとも 2019年末まで	uiterlijk EIND 2019	STARTTLSとDANE SPFとDMARC 厳しいポリシー

Implementatie-deadline

Betreffende standaarden

DNSSEC

SPF

DKIM

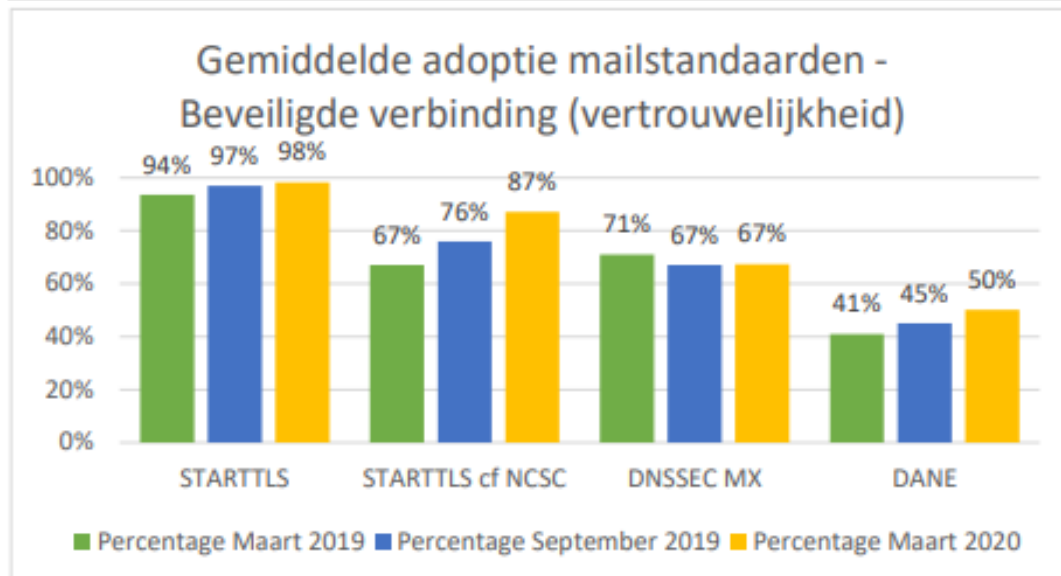
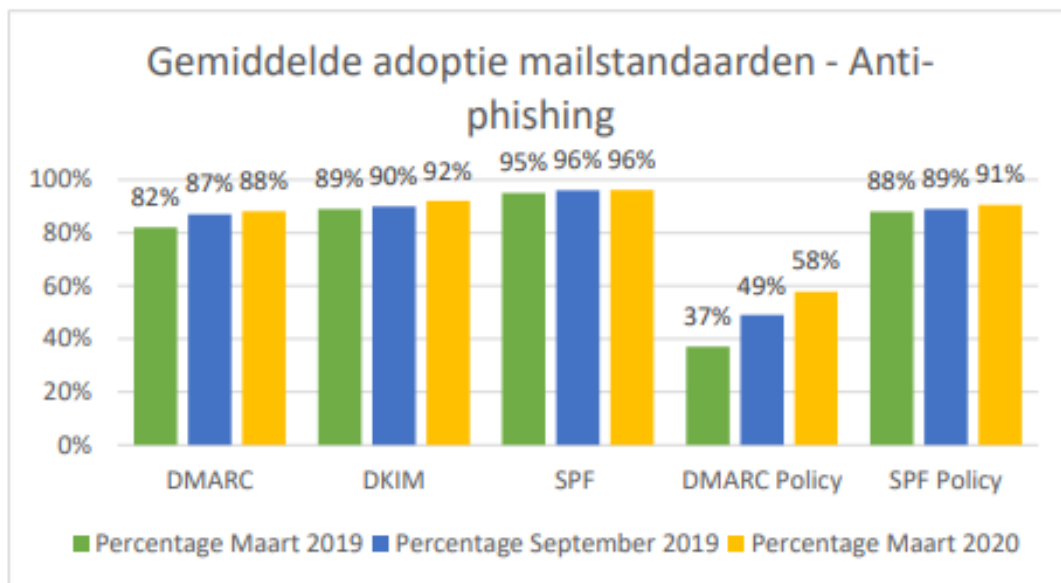
DMARC

STARTTLSとDANE

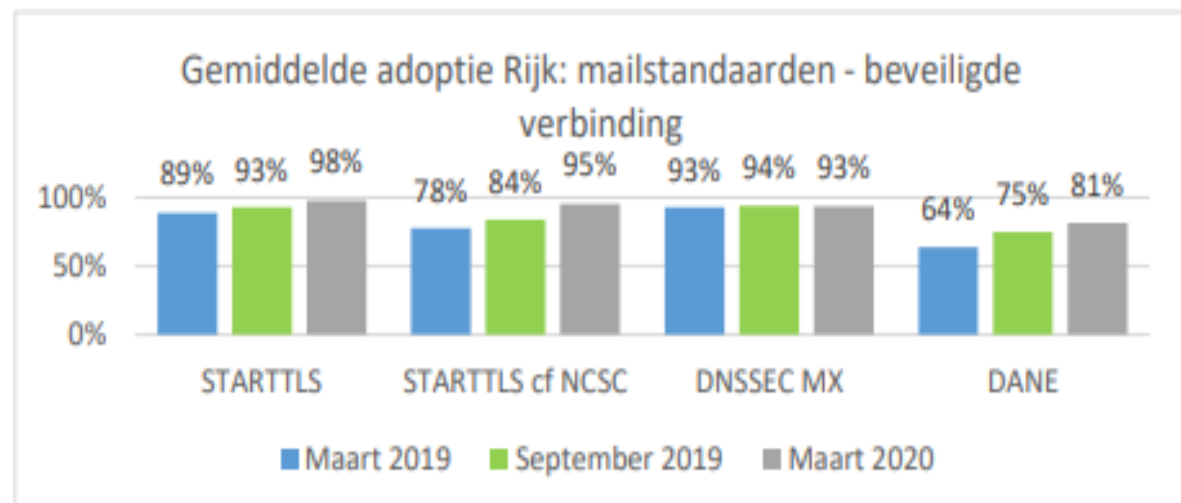
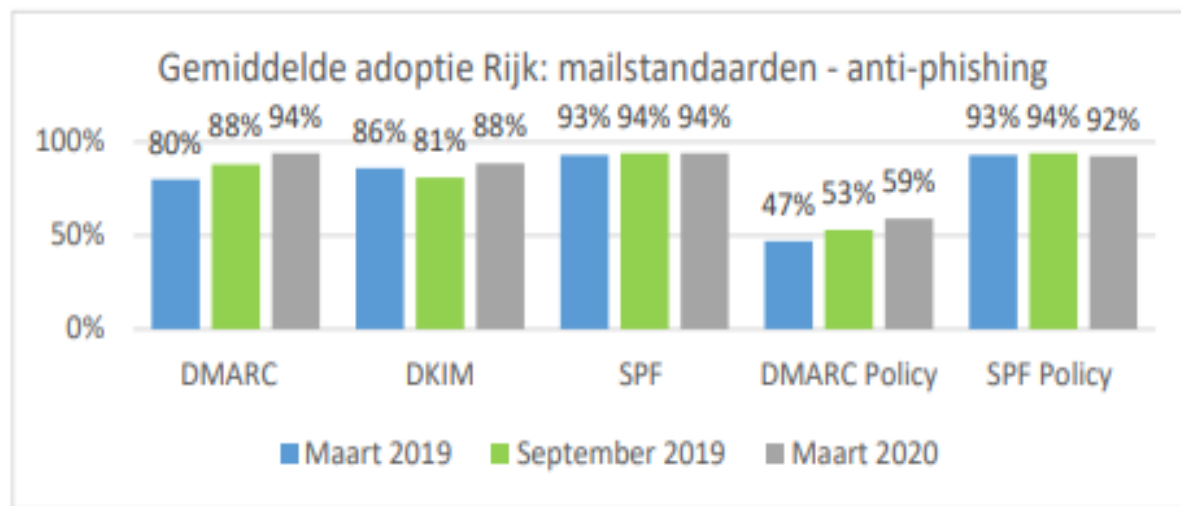
**SPFとDMARC
厳しいポリシー**

オランダでの普及率

全体

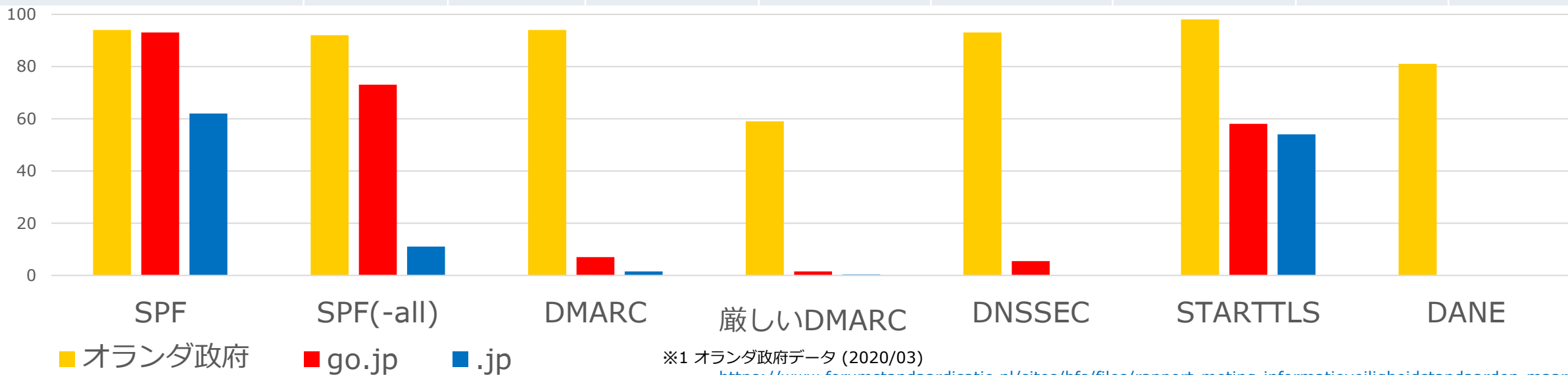


Het Rejk 国 (政府系??)



一方 日本では

	SPF	SPF -all	DMARC	厳しい DMARC	DNSSEC	START TLS	MTA- STS	DANE
オランダ政府(※1)	94%	92%	94%	59%	93%	98%	-	81%
go.jp (2020/11) (※2)	93%	73%	7.0%	1.5%	5.5%	58%	0%	0%
(2021/7) (※2)	94%	74%	7.3%	0.9%	6.0%	63%	0%	0%
.jp (2020/11) (※3)	62%	11%	1.5%	0.3%	0.04%	54%	13件	6件
(2021/1) (※4)	71%	14%	2.0%	0.3%	0.09%	62%	15件	7件
(2021/7) (※4)	71%	14%	2.3%	0.3%	0.10%	64%	18件	6件



※1 オランダ政府データ (2020/03)

<https://www.forumstandaardisatie.nl/sites/bfs/files/rapport-meting-informatieveiligheidsstandaarden-maart-2020.pdf>

※2 QUALITIA独自調べ go.jp(全てではない)のうちMXのあるドメイン(サブドメインは含まない) N=330 (2020/11) N=317(2021/7)

※3 QUALITIA独自調べ jpドメイン(全てではない)のうちMXのあるドメイン N=約32万(サブドメイン含む) ※4 N=約20万(eTLD+1のみ)

オランダ政府御用達チェックサイト



[English](#) [Nederlands](#)

[Home](#) [News](#) [Knowledge base](#) [Hall of Fame](#) [About Internet.nl](#)

Modern Internet Standards provide for more reliability and further growth of the Internet.
Are you using them?

Test your website



Modern address? Signed domain? Secure connection? Security options?

[about the test](#) >

Your website domain name:

www.example.nl

Start test

Test your email



Modern address? Signed domain? Anti-phishing? Secure connection?

[about the test](#) >

Your email address:

@ example.nl

Start test

Test your connection



Modern addresses reachable? Domain signatures validated?

[about the test](#) >

Start test

試してみた

Email test: hirano.cc

46%

- ✘ [Not reachable via modern internet address, or improvement possible](#)
- ✘ [Not all domain names signed \(DNSSEC\)](#)
- ✘ [Not all authenticity marks against email phishing \(DMARC, DKIM, etc.\)](#)
- ✘ [Mail server connection *not* or insufficiently secured \(STARTTLS and TLS\)](#)

[Explanation of test report](#)

[Permalink test result \(2020-11-10 14:14 CET\)](#)

[Seconds until retest option: 158](#)

✘ Modern address (IPv6)

Too bad! Your mail server can not be reached by senders using modern addresses ([IPv6](#)), or improvement is possible. Therefore your mail server is not part of the modern Internet yet. You should ask your email provider to fully enable IPv6.

[Show details](#)

Name servers

✘ IPv6 addresses for name servers

○ IPv6 reachability of name servers

Mail server(s)

✘ IPv6 addresses for mail server(s)

○ IPv6 reachability of mail server(s)

散々な結果

✘ Signed domain names (DNSSEC)

Too bad! Some or all of your email address and mail server domains are not valid signature ([DNSSEC](#)). Therefore senders with enabled domain signature not able to reliably query the IP address of your receiving mail server(s). An attacker can secretly manipulate the IP address and divert e-mails addressed to you to their mail server. You should ask your name server operator and/or your mail provider to activate DNSSEC.

Email address domain

✘ DNSSEC existence

○ DNSSEC validity

Mail server domain(s)

✘ DNSSEC existence

○ DNSSEC validity

✘ Authenticity marks against phishing (DMARC, DKIM and SPF)

Too bad! Your domain does *not* contain all authenticity marks against email forgery ([DMARC, DKIM and SPF](#)). Therefore receivers are *not* able to reliably separate phishing and spam emails abusing your domain in their sender address from your authentic emails. You should ask your mail provider to activate DMARC, DKIM and SPF.

[Show details](#)

DMARC

✔ DMARC existence

✘ DMARC policy

DKIM

✔ DKIM existence

SPF

✔ SPF existence

✔ SPF policy

散々な結果 (続き)

✘ Secure mail server connection (STARTTLS and DANE)

Too bad! Sending mail servers that support secure email transport ([STARTTLS and DANE](#)) can establish *no* or an *insufficiently* secure connection with your receiving mail server(s). Passive and/or active attackers will therefore be able to read emails in transit to you. You should ask your mail provider to enable STARTTLS and DANE, and configure it securely.

[Show details](#)

TLS

- ✓ STARTTLS available
- ⚠ TLS version
- ⚠ Ciphers (Algorithm selections)
- ✘ Cipher order
- ✘ Key exchange parameters
- ✓ Hash function for key exchange
- ✓ TLS compression
- ✓ Secure renegotiation
- ⚠ Client-initiated renegotiation
- ✓ 0-RTT

Certificate

- ✓ Trust chain of certificate
- ✓ Public key of certificate
- ✓ Signature of certificate
- ✓ Domain name on certificate

DANE

- ✘ DANE existence
- ⚪ DANE validity
- ⚪ DANE rollover scheme

盗聴・なりすまし受信
から守る

盗聴・改ざん・なりすまし



メール
サーバ

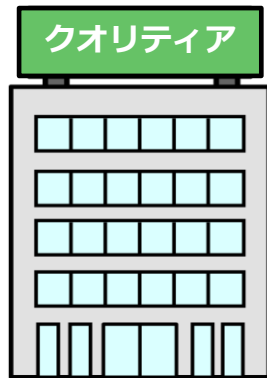


盗聴



改ざん

メール
サーバ



偽メール
サーバ



なりすまし

ZIP暗号化

ZIP暗号化



メール
サーバ

~~盗聴~~



~~改ざん~~



~~盗難~~



メール
サーバ



パスワード

STARTTLS

STARTTLS

盗聴



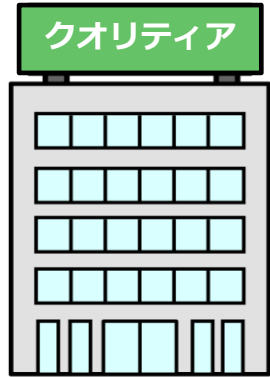
改ざん



メール
サーバ



メール
サーバ



メールサーバー間を暗号化する

やってみた

TLS	
✔ STARTTLS available	▼
⚠ TLS version	▼
⚠ Ciphers (Algorithm selections)	▼
✖ Cipher order	▼
✖ Key exchange parameters	▼
✔ Hash function for key exchange	▼
✔ TLS compression	▼
✔ Secure renegotiation	▼
⚠ Client-initiated renegotiation	▼
✔ 0-RTT	▼

あれれ。

TLSの設定をしただけでは、
まだまだ足りませんでした。



厳しい設定

Postfixの場合

```
smtpd_tls_security_level = may
smtpd_tls_key_file = /etc/letsencrypt/live/example.jp/privkey.pem
smtpd_tls_cert_file = /etc/letsencrypt/live/example.jp/fullchain.pem

smtpd_tls_ciphers = high
smtpd_tls_mandatory_ciphers = high
smtpd_tls_mandatory_protocols = !SSLv2,!SSLv3,!TLSv1,!TLSv1.1
smtpd_tls_protocols = !SSLv2,!SSLv3,!TLSv1,!TLSv1.1
tls_high_cipherlist = EECDH+AESGCM
tls_preempt_cipherlist = yes
```

もう一度挑戦！

TLS

✓ STARTTLS available

✓ TLS version

✓ Ciphers (Algorithm selections)

✓ Cipher order

✓ Key exchange parameters

✓ Hash function for key exchange

✓ TLS compression

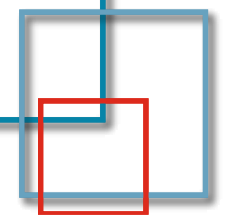
✓ Secure renegotiation

⚠ Client-initiated renegotiation

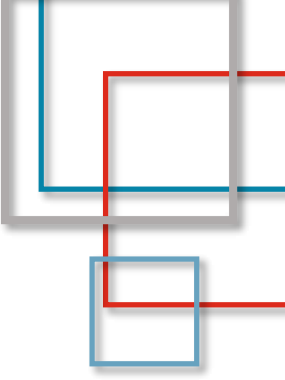
✓ 0-RTT

なかなか
いい感じになりました。

ほんとうに？



STARTTLSは



Opportunistic

=できればやる / できなければやらない

なりすましに対しては



メール
サーバ

偽メール
サーバ

TLSなんて
非対応ですよ!



なりすまし



メール
サーバ



STARTTLSに対応していても無意味

STARTTLSがあるとき

送信
サーバ

受信
サーバ

EHLO sender.example.jp

250-recv.example.jp
250-STARTTLS
250 OK

STARTTLS対応

STARTTLS

220 ready for TLS

なんやかんや やり取り

EHLO sender.example.jp 

ここから暗号化

このあたりは
平文

途中で改ざんされると

送信
サーバ

受信
サーバ



MITMさん

EHLO sender.example.jp

250-recv.example.jp
250-XXXXXXXXXX
250 OK

250-recv.example.jp
250-STARTTLS
250 OK

ふむふむ
TLS非対応
なのね

ちょこっと
書き換え

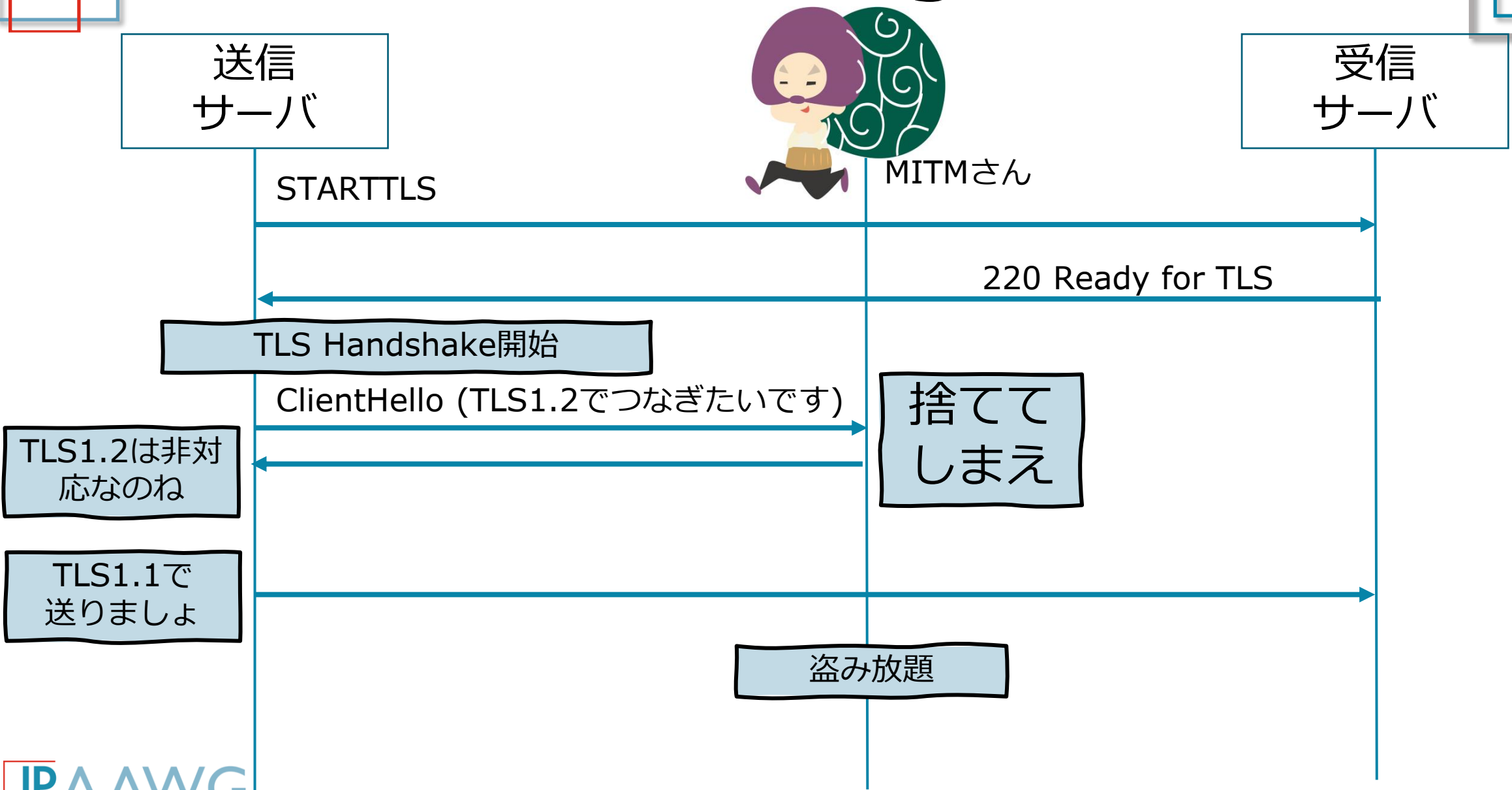
FROM: alice@sender.example.jp

暗号化せずに
送りましたよ

盗み放題

STARTTLS Downgrade Attack

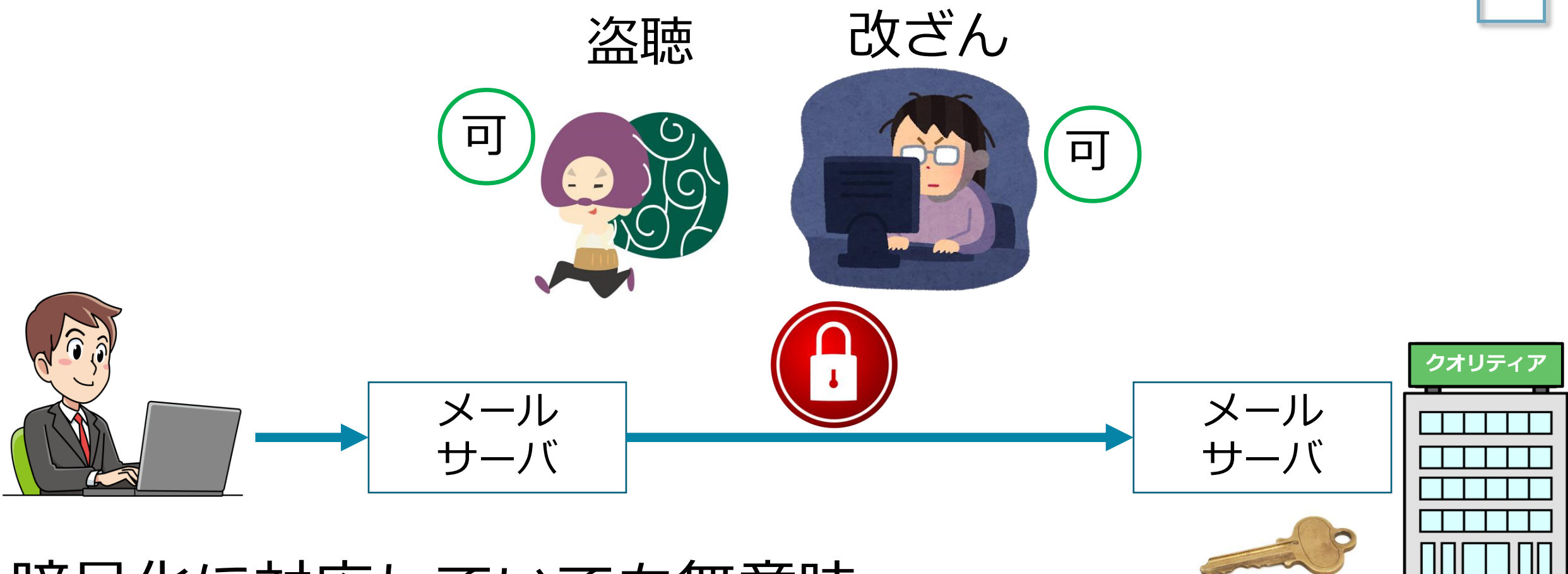
TLS Protocol Downgrade Attack



TLS1.2は非対応なのね

TLS1.1で送りましたよ

EHLO応答を改ざんされた場合





STARTTLSの問題点



- 盗聴・改ざん(MITM)に対して
 - Downgrade攻撃に対して弱い
- なりすまし受信に対して
 - 弱い

MTA-STS



MTA-STS

受信側が、送信サーバーに対して、

- STARTTLSを**必ず**使う
- TLS1.2以上を使う
- 証明書が有効でなければ配送しない

ようにしてもらおう仕組み

RFC8461 (2018/09)

MTA-STSがあるとき



メール
サーバ

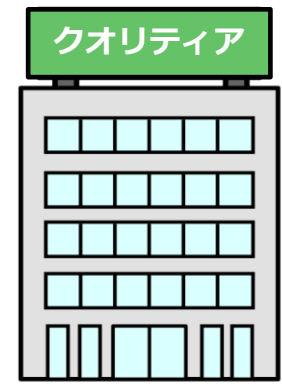
<https://mta-sts.qualitia.co.jp/.well-known/mta-sts.txt>

ポリシー

```
version: STSv1
mode: enforce
mx: mx1.qualitia.co.jp
max_age: 1296000
```



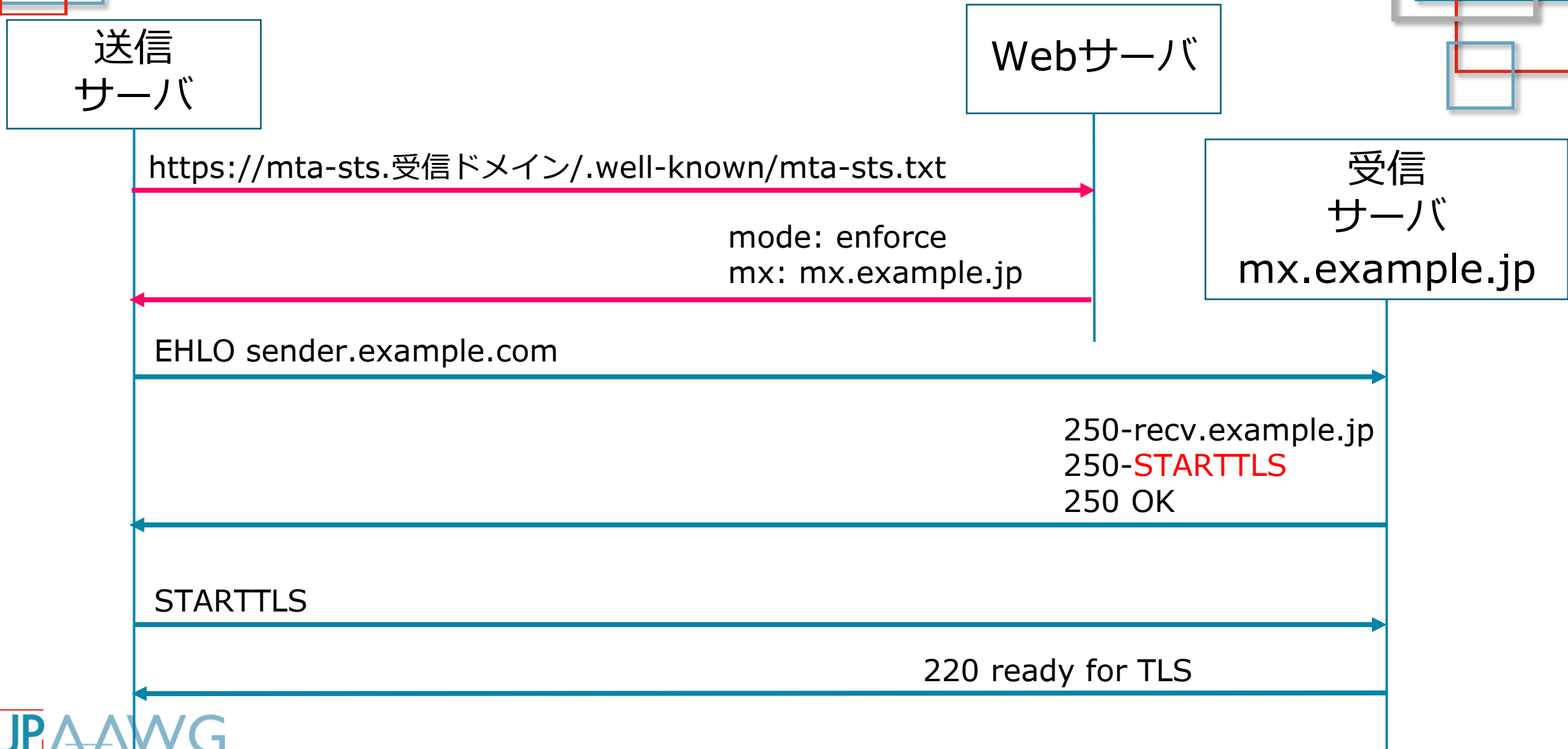
メール
サーバ



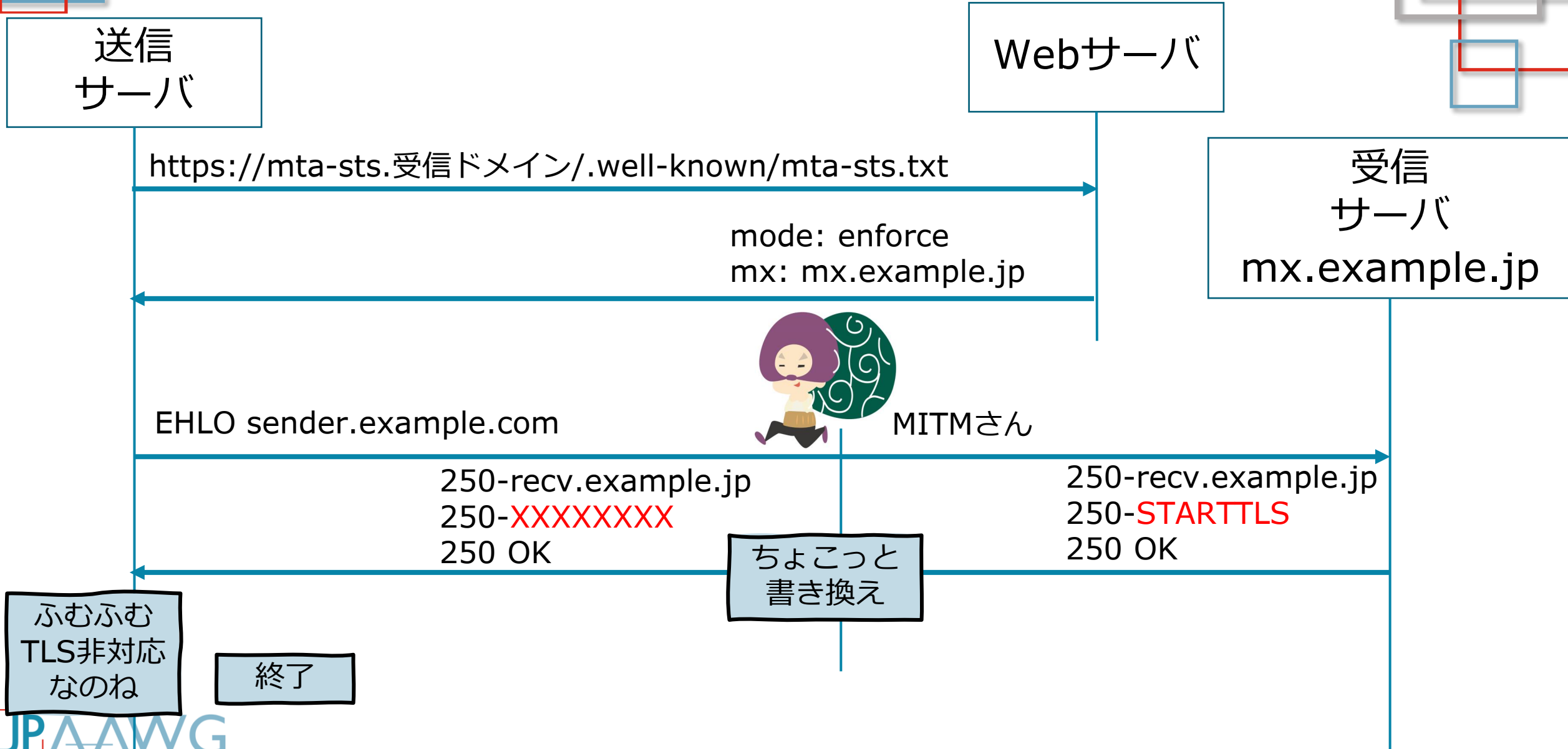
強い暗号化に対応
していなければ送らない
= 盗まれない

```
_mta-sts.qualitia.co.jp. IN TXT "v=STSv1; id=20191114123000Z;"
```

MTA-STSがあるとき



改ざんされた場合でも



MTA-STSの設定方法

受信するメールアドレス: bob@example.jp
受信メールサーバー: mx.example.jp

DNSの設定

```
_mta-sts.example.jp txt "v=STSV1; id=20201111010203"
```

Webの設定

<https://mta-sts.example.jp/.well-known/mta-sts.txt>

```
version: STSV1  
mode: enforce  
mx: mx.example.jp  
max_age: 1296000
```

none
testing
enforce

*.example.jpのようにも書けます

だがしかし

届かなかったことを知りたい

TLS-RPT



TLS-RPT



- MTA-STSやDANEの結果のレポートを受け取れます
- RFC8460 (2018/09)
[SMTP TLS Reporting]

TLS-RPTの設定方法

受信するメールアドレス: bob@example.jp
受信メールサーバー: mx.example.jp

レポートの送り先: report@example.com

```
_smtp._tls.example.jp txt  
"v=TLSRPTv1; rua=mailto:report@example.com"
```

レポートの送り先: https://example.com/report

```
_smtp._tls.example.jp txt  
"v=TLSRPTv1; rua=https://example.com/report"
```

レポートの例（問題ない場合）

```
{
  "organization-name": "Google Inc.",
  "date-range": {
    "start-datetime": "2020-09-07T00:00:00Z",
    "end-datetime": "2020-09-07T23:59:59Z"
  },
  "contact-info": "smtp-tls-reporting@google.com",
  "report-id": "2020-09-07T00:00:00Z_hirano.cc",
  "policies": [
    {
      "policy": {
        "policy-type": "sts",
        "policy-string": [
          "version: STSv1",
          "mode: testing",
          "max_age: 86400",
          "mx: *.hirano.cc"
        ],
        "policy-domain": "hirano.cc"
      },
      "summary": {
        "total-successful-session-count": 5,
        "total-failure-session-count": 0
      }
    }
  ]
}
```

成功 5通

失敗 0通

レポートの例 (問題のある場合)

```
{
  "organization-name": "Google Inc.",
  "date-range": {
    "start-datetime": "2019-10-01T00:00:00Z",
    "end-datetime": "2019-10-01T23:59:59Z"
  },
  "contact-info": "smtp-tls-reporting@google.com",
  "report-id": "2019-10-01T00:00:00Z_hirano.cc",
  "policies": [
    {
      "policy": {
        "policy-type": "sts",
        "policy-string": [
          "version: STSv1",
          "mode: testing",
          "max_age: 86400",
          "mx: *.hirano.cc"
        ],
        "policy-domain": "hirano.cc"
      },
      "summary": {
        "total-successful-session-count": 0,
        "total-failure-session-count": 55
      }
    }
  ],
}
```

```
"failure-details": [
  {
    "result-type": "validation-failure",
    "sending-mta-ip": "209.85.219.198",
    "receiving-ip": "210.158.71.76",
    "receiving-mx-hostname": "ah.hirano.cc",
    "failed-session-count": 2
  },
  {
    "result-type": "starttls-not-supported",
    "sending-mta-ip": "209.85.222.201",
    "receiving-ip": "210.158.71.76",
    "receiving-mx-hostname": "ah.hirano.cc",
    "failed-session-count": 1
  },
  .... 省略 ....
]
```

失敗 55通

MTA-STS, TLS-RPT

受信側が、送信サーバーに対して、

- STARTTLSを**必ず**使う
- TLS1.2以上を使う
- 証明書が有効でなければ配送しない

ようにしてもらおう仕組み

レポートもある

RFC8461 (2018/09)

だがしかし


なりすましの場合



メール
サーバ

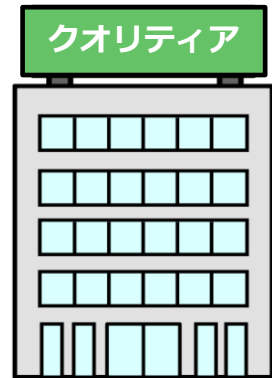
偽メール
サーバ



DNS 

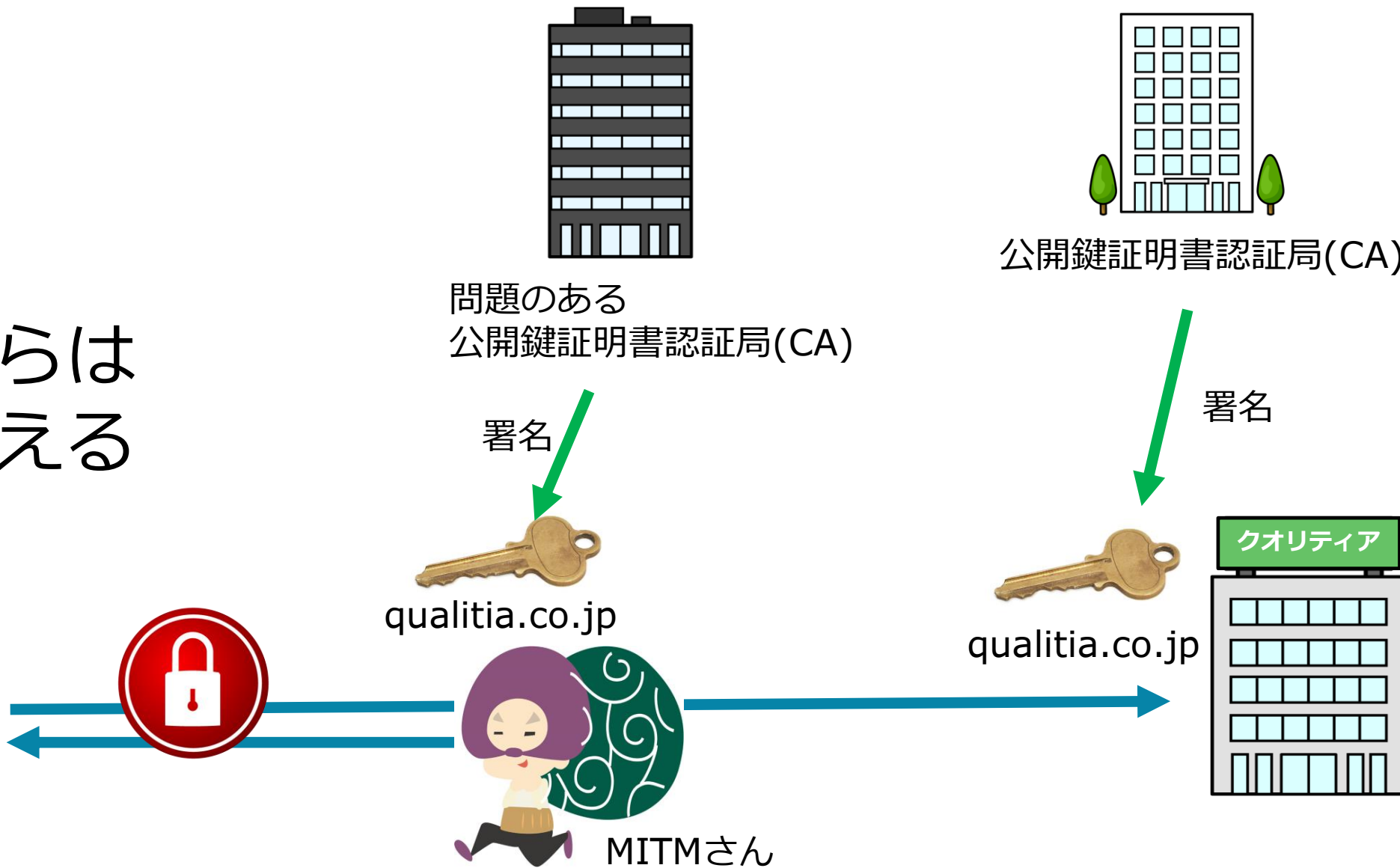
MTA-STSを無効化

メール
サーバ



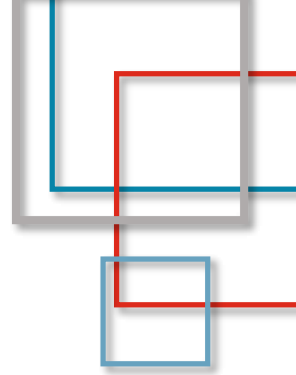
不正な認証局

送信者からは
正しく見える





MTA-STSの問題点

- DNSの毒入れなどのなりすましに弱い
 - 不正な証明書を利用したMITM攻撃に弱い
- 



DANE



DANE

- STARTTLSを必ず使う
- TLS1.0以上、できれば1.2以上を必ず使う
- DNSSECが検証できなければ配送しない
- 公開鍵が正しくない場合は配送しない

- RFC7671 (2015/10)

- RFC7672 (2015/10)



DANE

- DNSSECが必須

DNSSEC未対応の場合は、通常通り配送

- 必ずTLSを使う

TLS1.0(MUST) / TLS1.2以上(SHOULD)

- 公開鍵証明書認証局(CA)を利用しない

- 使用してもよいが検証はされない

- オレオレ証明書でもOK

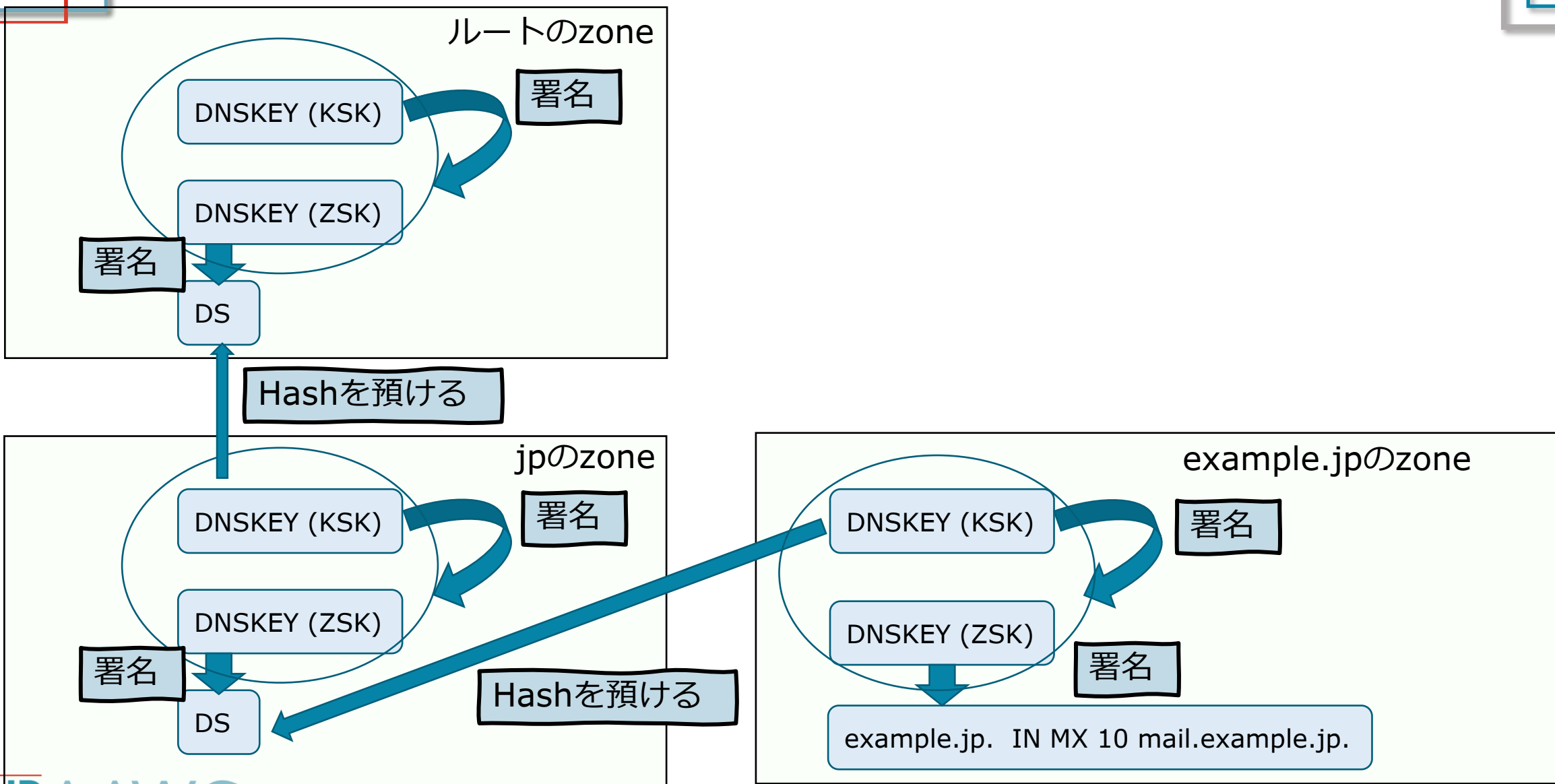


DNSSECとは？



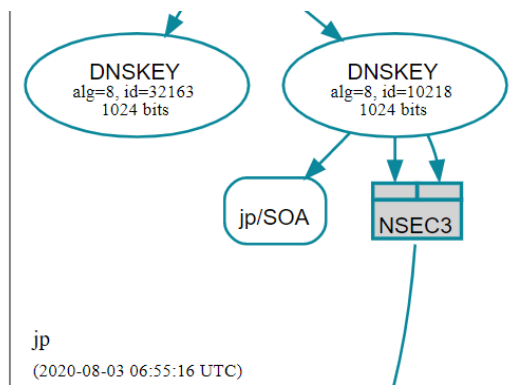
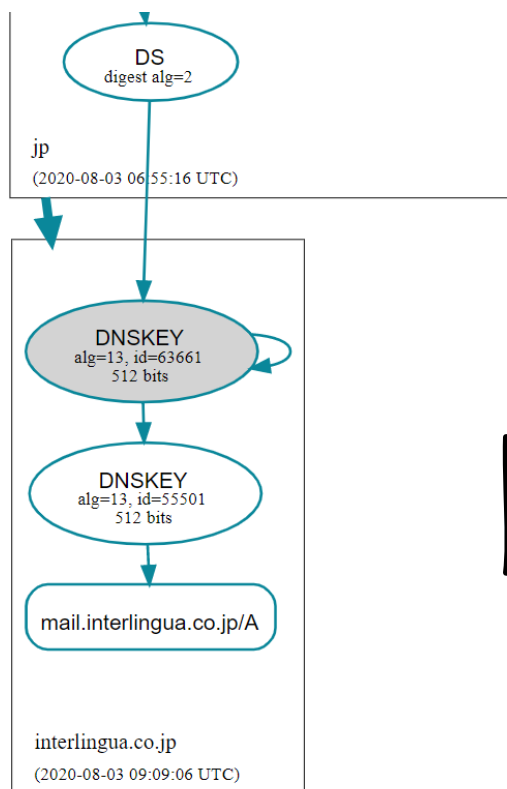
- ✗ DNSの問い合わせや応答を暗号化して守る
- DNSの応答が改ざんされていないことを保証する
- DNSの応答が正しい人からのものであることを保証する

DNSSECのトラストチェーン

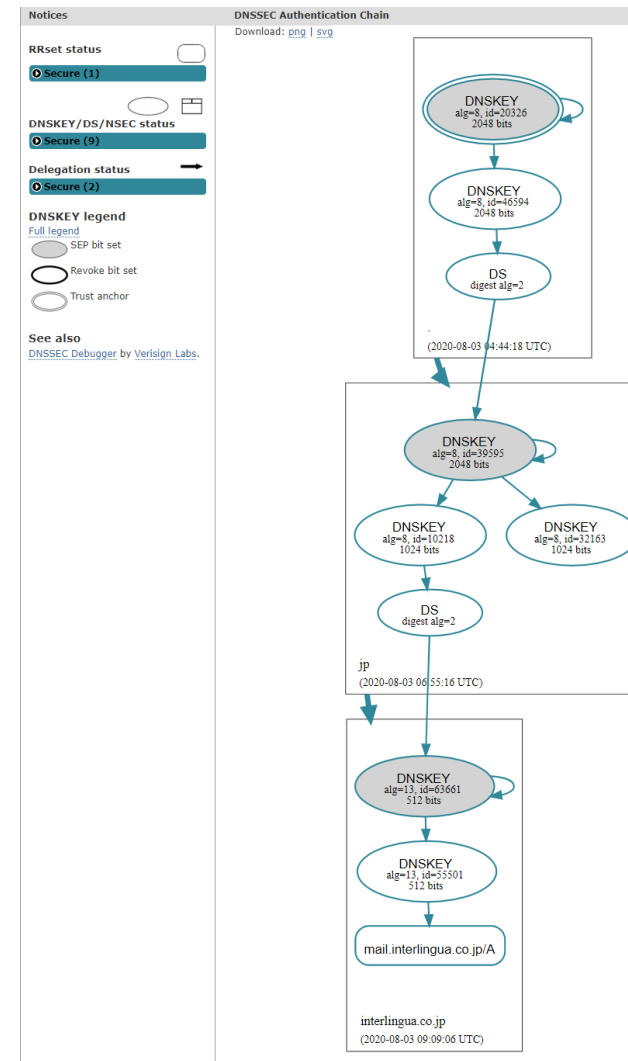


DNSSECが有効かどうかの確認

<https://dnsviz.net/>



DNSSECが失敗したところ
は黒くなります



DNSSECの応答

応答

	DNSSEC非対応ドメイン	DNSSEC対応ドメイン	なりすまされたDNSSEC対応ドメイン
DNSSEC非対応リゾルバ	回答あり	回答あり	回答あり
DNSSEC対応リゾルバ	回答あり	回答あり	SERVFAIL

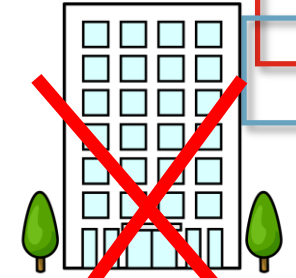
なりすまされた場合、結果が返ってこない

DANE for SMTP

CAの代わりにDNSSECを信頼



メール
サーバ



公開鍵証明書認証局(CA)

不要

信頼

ルートDNS
DNSSEC

Public KeyのHash

DNSSEC

Public Key

メール
サーバ

mx1.qualitia.co.jp



```
_25._tcp.mx1.qualitia.co.jp. IN TLSA 3 1 1 2B73BB905F..."
```




DANEの設定方法

Public KeyのHashを作成

```
openssl x509 -in cert.pem -pubkey -noout  
| openssl rsa -pubin -outform DER  
| openssl sha256
```

(stdin)= 293f3944e435835ec797acbbe52ffb1bc8e
6637879fbe62d9b6195479e01f67e

DANEの設定方法

はじめての設定なら、
サーバーから証明書を取り出すのもあり

```
openssl s_client -connect mx1.example.jp:25 -starttls smtp < /dev/null  
| openssl x509 -pubkey -noout  
| openssl rsa -pubin -outform DER  
| openssl sha256
```

(stdin)= 293f3944e435835ec797acbbe52ffb1bc8e
6637879fbe62d9b6195479e01f67e

DNSに追加

受信するメールアドレス: bob@example.jp
受信メールサーバー: mx1.example.jp

`_25._tcp.mx1.example.jp TLSA 3 1 1 293f3944e...`

メールサーバー

0: 証明書
1: 公開鍵

0: Hashなし
1: SHA256
2: SHA512

メールアドレスのドメイン部分ではありません！

※TLSのKeyを入れ替えるときにはTLSAレコードを先に書いて、DNSのキャッシュ期間が過ぎたらメールサーバーの設定を新しいKeyに変更し、古いTLSAレコードを削除します。

DANE

DNSSECに対応していて、
TLSAレコードがあれば、
STARTTLSを必須で使用し、
PublicKeyをTLSAの値で検証します。

Microsoftの対応予定

送信側の対応2020年末まで

受信側の対応2021年末まで

(by M3AAWG General Meeting (2020/06))

	TLS	DANE
Arcor	yes	no
AOL	yes	no
Bund.de	yes	yes
Comcast	yes	yes
Freenet	yes	yes
Gmail	yes	no
GMX	yes	yes
Kabel Deutschland	yes	yes
O2	yes	no
Outlook.com	yes	no
Riseup	yes	yes
T-Online	yes	no
Unitymedia	yes	yes
Vodafone	yes	yes
web.de	yes	yes
Yahoo	yes	no

実際に設定してみる

実際に設定してみる

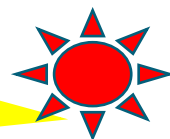
意外と高い！ DNSSECの壁！！！！

親のDSレコードに自分のZoneのKeyのHashを登録する必要がある
→example.jpから見ると、親はjpなので、
JPRSのDSレコードを変更できる必要がある
→対応しているレジストラがあまり見つからない
→DNSSECのホスティングはありそうだ
→しかし、20年以上も自分で管理してきたので、自分で管理したい

ということで

クオリティアでDNSSECホスティングサービスを作っちゃいました

Internet Weekを見た方は無料



QUALITIA DNS で検索！

DNSSEC + TLSA設定完了

Email test: interlingua.co.jp



87%

- ✗ [Not reachable via modern internet address, or improvement possible \(IPv6\)](#)
- ✓ [All domain names signed \(DNSSEC\)](#)
- ✓ [Authenticity marks against email phishing \(DMARC, DKIM and SPF\)](#)
- ✓ [Mail server connection sufficiently secured \(STARTTLS and DANE\)](#)

✓ Signed domain names (DNSSEC)

Well done! Your email address domain and your mail server domain(s) are signed with a valid signature ([DNSSEC](#)). Therefore senders with enabled domain signature validation, are able to reliably query the IP address of your receiving mail server(s).

[Show details](#)

DANE

- ✓ DANE existence
- ✓ DANE validity
- i DANE rollover scheme

Email address domain

✓ DNSSEC existence

✓ DNSSEC validity

Mail server domain(s)

✓ DNSSEC existence

✓ DNSSEC validity



最後の壁 IPv6

データセンターへ問い合わせ

IPv6の件、結論から申しますと「対応いたしません」となります。



理由は、設計計画の立案、作業に対する予算化、サービスメニュー見直し等大幅な労力を要するためです。

ご要望にお応えできず、申し訳ございませんでした。

え” ————— つ



だがしかし

送信者がTLSを強制
したい場合はどうするの？



Require TLS



Require TLS

RFC8689 (2019/11)

SMTPで

MAIL FROM: <alice@example.jp> REQUIRETLS

ヘッダに

TLS-Required: No

と書くことで、機能をOffにできます。

DANE for SMIME



DANE for S/MIME



- DNSSECが必須 DNSSEC未対応の場合は、Failure
- TCP推奨、UDP非推奨
- RDATAの書き方はDANE for TLSと同じ
- 証明書でも公開鍵でもよい
- RFC8162 (2017/5) (Experimental)

SMIMEA

ホスト名部分はメールアドレスのlocalpartのSHA256の28オクテット

```
# echo -n alice@example.jp | openssl sha256
```

```
→ 2bd806c97f0e00af1a1fc3328fa763a9269723c8db8fac4f93af71db186d6e90
```

証明書や公開鍵はDER形式にして16進数で書く

```
# openssl x509 -in smime.crt -outform DER | xxd -ps
```

```
308202cd308201b50214156aee144514e7969ff77e02936211039f9db59e  
300d06092a864886f70d01010b050030123110300e06035504030c076869  
72615f4341301e170d3231303131373134353434345a170d323230313137  
.... 以下大量
```

```
2bd806c...af71db._smimecert.example.jp SMIMEA
```

```
3 0 0 308202cd308201b50214156aee144...
```




まとめ

- DMARC 使いましょう
- STARTTLS 使いましょう
- MTA-STS 使いましょう
- DNSSEC 使いましょう
- DANE 使いましょう
- internet.nlで100点になるまでの道は険しい
- オランダすごいよ



今日の内容を少しQiitaにも書きました
<https://qiita.com/hirachan>

Thank You!

Thank you

