« GOTO FAIL »

ENORME faille de sécurité d'Apple
au travers SSL/TLS

Introduction

- Référencée : 8 janvier 2014
- Faille active : environ un an et demi
- Rappel sur le SSL
- CVE-2014-1266

Firefox certificate warning

Description:

Secure Transport failed to validate the authenticity of the connection. This issue was addressed by restoring missing validation steps.

Impact:

An attacker with a privileged network position [like a MitM] may capture or modify data in sessions protected by SSL/TLS

Cas basique d'un navigateur consultant un site via HTTPS

Client

Serveur

Chiffrements supportés

Chiffrements supportés
Clé publique

- Vérification de l'authenticité de la clé publique
-Choix d'un chiffrement en commun
- Chiffrement d'une clé de session secrète avec la clé publique

Clé de session secrète chiffrée

fin de la transmission en clair

Transmission chiffrée par clé secrète

Fin de la transmission en clair

Transmission chiffrée par clé secrète

- Vérification de la clé

WHAT WAS WRONG WITH APPLE'S SSL CODE?

SSLProcessHandshakeRecord()

-> SSLProcessHandshakeMessage()

-> SSLProcessClientHello()

-> SSLProcessServerHello()

-> SSLProcessCertificate()

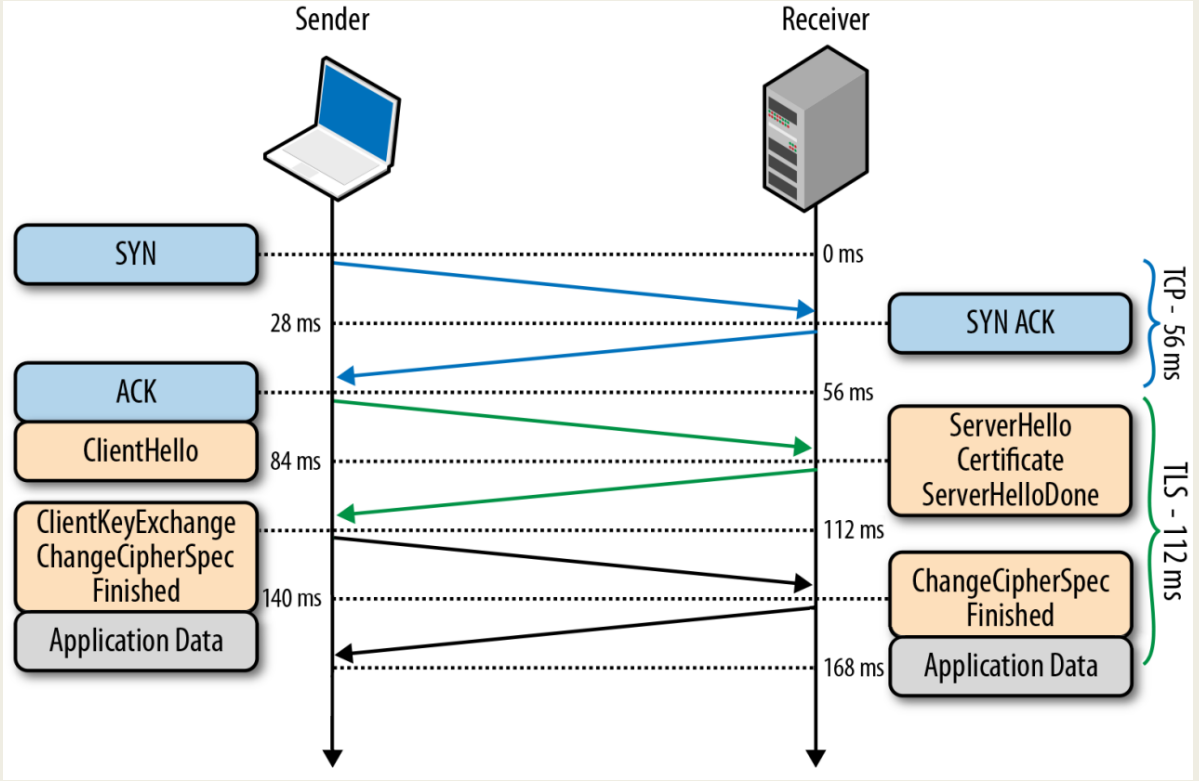
-> SSLProcessServerKeyExchange()

-> SSLDecodeSignedServerKeyExchange()

-> SSLDecodeXXKeyParams()

IF TLS 1.2 -> SSLVerifySignedServerKeyExchangeTls12()

OTHERWISE -> SSLVerifySignedServerKeyExchange()



Erreur de duplication ou oubli d'accolades…

```
    hashOut.data = hashes + SSL_MD5_DIGEST_LEN;
hashOut.length = SSL_SHA1_DIGEST_LEN;
if ((err = SSLFreeBuffer(&hashCtx)) != 0)
    goto fail;

if ((err = ReadyHash(&SSLHashSHA1, &hashCtx)) != 0)
    goto fail;
if ((err = SSLHashSHA1.update(&hashCtx, &clientRandom)) != 0)
    goto fail;
if ((err = SSLHashSHA1.update(&hashCtx, &serverRandom)) != 0)
    goto fail;
if ((err = SSLHashSHA1.update(&hashCtx, &signedParams)) != 0)
    goto fail;
    goto fail;
if ((err = SSLHashSHA1.final(&hashCtx, &hashOut)) != 0)
    goto fail;

    err = sslRawVerify(ctx,
                       ctx->peerPubKey,
                       dataToSign,                    /* plaintext */
                       dataToSignLen,               /* plaintext length */
```

/* MISTAKE! THIS LINE SHOULD NOT BE HERE */

Analyse du code vulnérable

```
err = ReadyHash(&SSLHashSHA1, &hashCtx)
err = SSLHashSHA1.update(&hashCtx, &clientRandom)
err = SSLHashSHA1.update(&hashCtx, &serverRandom)
err = SSLHashSHA1.update(&hashCtx, &signedParams))
err = SSLHashSHA1.final(&hashCtx, &hashOut)
err = sslRawVerify(ctx, ctx->peerPubKey,
dataToSign,dataToSignLen,signature,signatureLen);
Fail: …
      return err
```

Variable *err* non affectée

Variable *err* affectée

→ return 0 !!!
= Verification OK

Une répétition lourde de conséquences…
puisque les connexions SSL sont ainsi **validées** que les certificats soient **conformes ou non**.

Attaque éventuelle

An attacker now has a way to trick users of OS X 10.9 into accepting SSL/TLS certificates that ought to be rejected, though admittedly there are several steps, and he needs to:

- Trick you into visting an imposter HTTPS site, e.g. by using a **poisoned public Wi-Fi access point**.
- Force your browser (or other software) into using **forward secrecy**
  → possible because the server decides what encryption algorithms it will support.
- Force your browser (or other software) into using **TLS 1.1**
  → possible because the server decides what TLS versions it will allow.
- Supply a **legitimate-looking TLS certificate** with a mismatched private key.

Réflexion sur cette vulnérabilité

- **Cette vulnérabilité aurait-elle pu être découverte plus tôt ?**
  Une analyse statique de ce code aurait rapidement permis de détecter la faille de sécurité…
- **Un laxisme intrigant…**
  - *Manque de relecture et de tests automatiques*
  - *Correctif officiel tardif (8 Jan – 26 Fev)*
  - *Pression gouvernementale ? NSA ? Backdoor volontaire ?*
- **Un impact significatif…**
  Le navigateur Safari ainsi que de nombreuses applications (Mail, iCloud, Twitter…) s'appuyant sur cette librairie sont donc impactées par cette vulnérabilité.
  Systèmes affectés : iOS 6.x, iOS 7.x, MacOS X 10.9.x, Apple TV

Sources

- http://www.numerama.com/magazine/28547-apple-securite-goto-fail-ios-macosx-securetransport-tls-ssl.html
- http://www.zdnet.fr/actualites/goto-fail-la-vulnerabilite-tres-etonnante-d-apple-39798118.htm
- https://nakedsecurity.sophos.com/2014/02/24/anatomy-of-a-goto-fail-apples-ssl-bug-explained-plus-an-unofficial-patch/
- http://www.clubic.com/mac-os/mac-os-x/actualite-685642-os-x-ssl-goto-fail-faille-theorie-du-complot-nsa.html
- https://www.dwheeler.com/essays/apple-goto-fail.html