TLSAssistant: uno strumento per mitigare i problemi di sicurezza di TLS

Introduzione

Gli ultimi anni hanno visto una crescita esponenziale dell'utilizzo di TLS (Transport Layer Security) per la messa in sicurezza di servizi ed applicazioni on-line. Nonostante questa popolarità, molti amministratori di sistema non riescono a gestire correttamente i problemi di sicurezza che derivano da configurazioni errate (ad esempio, l'utilizzo di cifrari deboli). Per questo motivo, sono stati sviluppati molti strumenti automatici per l'identificazione di vulnerabilità nelle implementazioni di TLS. Nonostante la semplicità di utilizzo, tali strumenti non assistono gli amministratori nel difficile compito di applicare le adeguate misure per mitigare i problemi riscontrati. Di conseguenza, agli amministratori viene richiesto di investire una grande quantità di tempo, che di solito non hanno a disposizione, per trovare, comprendere e tradurre in azioni concrete che portano alle opportune modifiche della configurazione di TLS. Per ovviare a questi problemi, descriviamo TLSAssistant, uno strumento capace di individuare non solo i potenziali problemi di sicurezza in una configurazione di TLS ma anche di generare un rapporto che indichi quali sono le mitigazioni da adottare.

Strumenti disponibili

La maggior parte degli strumenti disponibili per analizzare la sicurezza di TLS, una volta invocati, inviano ripetutamente messaggi di tipo ClientHello (il primo messaggio di un handshake TLS) e analizzano il contenuto del ServerHello (la risposta) da cui è possibile risalire alla configurazione e,

di conseguenza, ricercare indicatori di vulnerabilità.

Alcuni di questi strumenti sono liberamente scaricabili mentre altri possono essere utilizzati direttamente da browser. Nell'ambito webapp troviamo aziende come Qualys che offrono sia una piattaforma a pagamento che un servizio di scansione gratuito in grado di rilevare vulnerabilità. Lo strumento di Qualys esegue un gran numero di analisi e valuta sistematicamente il livello di sicurezza assegnando un voto (da A+ a F) al sito analizzato. Il report finale riassume inoltre la totalità dei test eseguiti ed evidenzia le anomalie rilevate utilizzando una scala di colori per caratterizzarne la gravità.

Un altro strumento online accessibile gratuitamente è Discovery. Il servizio viene fornito da Cryptosense, una startup attiva nell'ambito della sicurezza in ambito crittografico, e sembra essere lo strumento più simile a TLSAssistant. Oltre ad un sistema di valutazione simile al precedente, Cryptosense Discovery offre inoltre una descrizione su come migliorare la sicurezza del sistema analizzato, ad esempio suggerendo di aumentare la lunghezza delle chiavi crittografiche in casi si usino alcuni cifrari.

A titolo di esempio descriviamo brevemente tre strumenti che richiedono l'installazione in locale:

- ssl-enum-ciphers: script sviluppato per il noto strumento di *port scanning* nmap. Lo script è in grado di elencare tutti i cifrari supportati da un dato server, l'eventuale supporto alla compressione dei pacchetti ed una piccola serie di errori di configurazione noti. Il report da esso generato mostrerà inoltre un livello di sicurezza calcolato utilizzando lo stesso metodo offerto dal servizio online di Qualys (già menzionato);
- TLS-Attacker: framework sviluppato allo scopo di analizzare le librerie TLS. È completamente personalizzabile e permette, tra le altre

caratteristiche, di simulare un qualsiasi tipo di messaggio TLS per testare il comportamento dei server in analisi. Il progetto è corredato di una serie di flussi predefiniti in grado di simulare alcuni degli attacchi a TLS;

- testssl.sh: è forse il più potente tool open-source (per analisi TLS) disponibile sul web. Il tool è infatti in grado di estrarre informazioni utili a partire dell'*header* di risposta del server, di verificare i certificati offerti (e la relativa catena di fiducia), di simulare una moltitudine di *handshake* e, soprattutto, di segnalare esplicitamente le vulnerabilità delle quali il server potrebbe essere affetto. Nonostante la presenza di un report molto verboso, lo strumento rende espliciti i problemi riscontrati permettendo quindi una rapida valutazione del server analizzato.

La quasi totalità degli strumenti disponibili sono sì in grado di rilevare possibili falle ma non offrono suggerimenti sul come risolvere i problemi riscontrati.

TLSAssistant

TLSAssistant combina diversi strumenti di analisi allo stato dell'arte con un sistema di report che suggerisce le mitigazioni appropriate. TLSAssistant è *open source*, rilasciato sotto licenza Apache-2.0 e disponibile online al seguente *repository*. Lo strumento è stato sviluppato all'interno di Digimat, un laboratorio congiunto tra FBK e Poligrafico e Zecca dello Stato.

TLSAssistant è scritto in Bash e può quindi essere richiamato dalla riga di comando. Lo strumento prende come input il target da valutare (ad es. l'indirizzo IP di un server) e genera un file di report, il cui contenuto dipende dalle vulnerabilità rilevate e dal livello di verbosità scelto dall'utente. Figura 1 mostra i due componenti principali: *Analyzer* e *Evaluator*.



Figura 1. Architettura semplificata di TLSAssistant

L'*Analyzer* integra i migliori strumenti disponibili per l'individuazione dei problemi di sicurezza di TLS tra i quali testssl.sh (descritto in precedenza).

L'*Evaluator* è il contributo principale di TLSAssistant e si occupa della generazione del report che contiene le vulnerabilità rilevate dall'*Analyzer* e relative misure di mitigazione a vari livelli di astrazione:

1. *Descrizione testuale* (in linguaggio naturale) della vulnerabilità TLS, individuazione del rispettivo CVE, CVSS e relative mitigazioni.Ad esempio, se viene rilevata la vulnerabilità **CRIME**, questo è l'output testuale:

> **C**ompression **R**atio **I**nfo-leak **M**ass **E**xploitation
> **Description**: By exploiting the information leakage provided by DEFLATE (compression algorithm), an attacker is able to retrieve the session cookie. In particular, the attacker guesses parts of the cookie, injects them in a valid client packet and analyzes the server's response. Thanks to the properties of a DEFLATE output, if the server's response is bigger than an untouched packet, then the guess is wrong.
> **Mitigation**: Disable the TLS compression mechanism.
> **CVE**: 2012-4929
> **CVSS3**: 2.6 (Low)

2. *Code snippet*: frammento di codice che può essere copiato e incollato all'interno della configurazione del server web per correggere la vulnerabilità; il report dettaglia come individuare il file / la riga corretti da modificare. TLSAssistant è in grado di rilevare qualsiasi server web ma al momento è in grado di fornire istruzioni dettagliate solo per server HTTP *Apache* e

nginx.Considerando la vulnerabilità **CRIME** e un server *Apache,* il report conterrà il seguente *code snippet*:

1. open your Apache configuration file (default: */etc/apache2/sites-available/default-ssl.conf*);
2. search for the line starting with: **SSLCompression**
   - if found, change the value to **off**;
   - if not, add the line SSLCompression off within the file.

N.B. restart the server by typing: sudo service apache2 restart.

A breve, TLSAssistant sarà in grado di esportare il risultato dell'analisi in linguaggio STIX (*Structured Threat Information eXpression*), che supporta attività di cyber threat intelligence e facilita l'integrazione con altri strumenti come ad esempio quelli della piattaforma sviluppata nel progetto europeo FINSEC (attività attualmente in corso).

In sintesi, TLSAssistant è attualmente in grado di rilevare e fornire mitigazioni per una gran numero di vulnerabilità che vanno da 3SHAKE a Lucky13 passando per i problemi di gestione dei certificati e di gestione di TLS su Android.

Esempio di utilizzo

Per valutarne l'efficacia in ambito aziendale, TLSAssistant è stato utilizzato per analizzare il *deployment* di un servizio cloud di tipo SaaS (*Software as a Service*) per la gestione delle segnalazioni di **irregolarità** nel contesto lavorativo in conformità alla normativa sul *whistleblowing*. Grazie all'analisi prodotta da TLSAssistant abbiamo riscontrato un problema di configurazione che può dare la possibilità di eseguire un attacco *Man in the Middle* noto come *SSL stripping*. Questo attacco ideato nel 2009 permette di forzare la vittima ad utilizzare una connessione non cifrata (HTTP) per la trasmissione delle informazioni, riuscendo quindi a violare la

proprietà di confidenzialità normalmente garantita da TLS (e fondamentale nel caso di un servizio di *whistleblowing).*

Uno dei prerequisiti per realizzare l'attacco è la non adozione del meccanismo HSTS (HTTP Strict Transport Security) da parte del sito web. Anche se HSTS è presente, l'attacco può essere montato nell'intervallo di tempo tra il primo avvio di un browser e la ricezione della direttiva HSTS: questo caso è particolarmente rilevante per i servizi SaaS di *whistleblowing* che vengono generalmente utilizzati *una tantum.* Per ovviare a questo, la mitigazione suggerita da TLSAssistant è l'aggiunta del dominio del servizio SaaS all'elenco di "precaricamento". Il servizio, offerto da Google, consiste nella redazione di un elenco di siti che indicano la propria volontà di attuare la politica HSTS già dal primo avvio del browser.

Conclusioni

TLSAssistant integra rilevazione delle vulnerabilità e indicazioni operative precise per implementare le opportune mitigazioni e definire una configurazione sicura del protocollo TLS. Lo strumento è stato utilizzato sia in contesti aziendali che come ausilio didattico per un'esperienza di laboratorio nel contesto di un corso universitario introduttivo alla sicurezza informatica. Maggiori informazioni riguardanti lo strumento – nonché il codice – sono reperibili al seguente indirizzo: https://stfbk.github.io/tools/TLSAssistant.html.

Articolo a cura di **Salvatore Manfredi**, **Silvio Ranise** e **Giada Sciarretta**

https://www.ictsecuritymagazine.com/articoli/tlsassistant-uno-strumento-per-mitigare-i-problemi-di-sicurezza-di-tls/