Progetti

A.1 Mobile Sniffing Tool (Marco Querini).

Molte delle applicazioni mobili attuali, in uso nel 2016, sollevano diverse problematiche di privacy e security dei dati in transito e/o memorizzati nel dispositivo. Ad esempio per le applicazioni di messagistica diverse minacce alla security e alla privacy coinvolgono le applicazioni più note come Telegram [1], Whatsapp [2,3,4], Snapchat [5], etc..[6]. In questi mesi alcune applicazioni stanno passando per la prima volta a supportare la end-to-end-encryption dei dati in transito [7], ma molte applicazioni risultano insicure.

1) Si realizzi un software per dispositivi mobili (e.g., Android) in grado di sfruttare questi security flaws. Se i dispositivi si trovano nella stessa rete wifi il software realizzato deve poter rilevare i dati in transito:
   - Se i dati non sono cifrati (Plain-text):
     - Sniffing e visualizzazione dei dati in transito.
   - Se i dati in transito sono cifrati (e.g., end-to-end encryption)
     - Sniffing e visualizzazione della parte di dati non cifrata, ad esempio, time stamp, mittente e destinatario potrebbero essere catturati [8].
2) Si analizzi un campione di almeno 10 applicazioni reali di messagistica presenti sugli Store, per ciascuna applicazione analizzata descrivere il grado di sicurezza percepito a seguito dei tentativi di sniffing. Individuare casi di security flaws costituisce un plus.

A.2. Build Your Own Botnet v. 1(Giovanni Bottazzi).

Si sviluppi un software per workstation (no mobile) [9].

Obiettivo principale.

Il software deve essere in grado di contattare (http GET) una o più URL INTERNET. Per ciascuna URL bisogna prevedere una serie di variabili:

- dettaglio delle URL da contattare (minimo 2 diverse);
- periodicità di contatto, che può essere fissa (tempo impostato in secondi) o random in un intervallo min-max fisso (tempo sempre impostato in secondi);
- numero massimo di contatti per ciascuna URL;
- impostazione di uno "sleep mode", da intendere come insieme di condizioni per non effettuare alcuna azione (es. giorni dispari della settimana, orario AM o PM, ecc.);
- personalizzazione del campo user-agent (es.: "BYOB v. 1").

I valori delle variabili dovranno essere impostati attraverso un file (txt) di configurazione.

I contatti ed i parametri di configurazione dovranno esser loggati in un file di testo contenente, oltre alle informazioni di configurazione, anche il timestamp di contatto delle URL ed il dettaglio delle URL contattate.

Estensioni.

Le variabili precedenti dovranno essere impostate dall'utente tramite GUI. Aggiungere inoltre tra le variabili impostabili anche l'indirizzo di un proxy pubblico (URL o IP) da usare per i contatti.

Raccogliere e scrivere su un file TXT le informazioni relative al Sistema Operativo e al/i browser presenti sulla postazione su cui il software è installato.

A.3. Build Your Own Botnet v. 2(Giovanni Bottazzi).

Si sviluppi un web service con due diverse home page [9].

Obiettivo principale.

Il web server deve distinguere se è stato contattato canonicamente (es. index.php tramite browser), presentando una prima home page generica, oppure se è stato contattato con una "codifica particolare" (client sviluppato ad hoc), presentando una seconda home page generica.

Non viene fornito alcun vincolo sulla modalità di "codifica particolare", ma solo alcune possibili indicazioni:

- user-agent custom (es. "BYOB v. 2");
- metodo POST in luogo di GET;
- variabili a seguito della home page (es. "/index.php?var1=1&var2=2")

Estensioni.

In risposta al contatto con "codifica particolare", il web server può fornire, in luogo della generica home page, un file di testo con un elenco di coppie (ID, VALORE) da memorizzare in locale sul file system del client che contatta la home page (bisogna sviluppare anche il client).

Riferimenti:

[1] Telegram. http://thehackernews.com/2015/11/telegram-security-privacy.html (2015)
[2] Whatsapp. http://www.deccanchronicle.com/150626/technology-mobiles-and-tabs/article/shocking-whatsapp-not-secure-it-could-land-you-big (2015)
[3] Whatsapp sniffer. http://www.whatsappsnifferdownload.com/
[4] Whatsapp. https://www.wordsmart.it/spiare-le-conversazioni-whatsapp-nel-2016/. (2016)
[5] Snapchat. http://news.filehippo.com/2016/03/snapchat-breach-and-the-biggest-security-flaw-ever/. (2016)
[6] 11 Unsecure Messangers. https://blog.kaspersky.com/11_unsecure_messengers/6806/
[7] End to end encryption. http://www.recode.net/2016/4/19/11586234/viber-end-to-end-encryption-security-update
[8] http://www.livemint.com/Consumer/Xs9trQc9cfPjE3Q6NhAJYJ/What-WhatsApp-is-not-encrypting.html
[9] SANS – Institute. "BYOB: Build Your Own Botnet". https://www.sans.org/reading-room/whitepapers/threats/byob-build-botnet-33729