Build Week 3

Extra 1

LANDA

TRACKER SPA

Il malware **Mydoom**, apparso per la prima volta nel 2004, è uno dei worm più distruttivi della storia informatica.

Si è diffuso principalmente tramite e-mail, infettando i computer Windows e lasciando aperte le porte di rete per future intrusioni.
Il suo rapido spread ha causato gravi rallentamenti su reti aziendali e Internet globalmente.

- **Analisi Forense**: Attraverso l'analisi del codice sorgente, potete imparare come funziona un malware dal punto di vista tecnico. Questo include l'analisi delle funzioni di propagazione, le tecniche di evasione dei sistemi di sicurezza, e la comprensione di come il malware gestisce la comunicazione con i server di comando e controllo.

- **Scenario di Intelligence**: Supponiamo che la nostra intelligence abbia scoperto una nuova variante di Mydoom che sta emergendo. Dovete valutare il codice per identificare possibili modifiche o aggiornamenti rispetto alla versione originale. Questo esercizio mira a preparavi a rispondere rapidamente a nuove minacce, sviluppando capacità di analisi critica e di adattamento a scenari di sicurezza in evoluzione.

https://github.com/akir4d/MalwareSourceCode/raw/main/Win32/Win32.Mydoom.a.7z

Executive Summary (Riepilogo per il Dott.Rampino)

L'analisi del codice sorgente ha identificato la minaccia come il worm Mydoom.A, un malware auto-propagante estremamente aggressivo. Sebbene sia un malware datato, le tecniche utilizzate rimangono rilevanti e la sua architettura rappresenta un modello per molte minacce moderne.
(Fonti: **main.c** - **sco.c** - **p2p.c**)

- Il worm si diffonde principalmente tramite allegati email e reti di file-sharing (P2P). (Fonte: **massmail.c** - **xsmtp.c** - **p2p.c**)
- Una volta infettato un sistema, esegue due azioni dannose principali:
  - **Installa** una backdoor permanente, che trasforma il computer infetto in un proxy. (Fonte: **xproxy.c**). Questo permette a un aggressore di utilizzare la rete aziendale per condurre attività illecite anonime e garantisce un accesso completo e persistente al sistema per esfiltrare dati o installare altro malware (es. ransomware, spyware). (Fonte: **xproxy.c** - **client.c**)
  - **Utilizza** il computer per attacchi coordinati di tipo Distributed Denial-of-Service (DDoS) contro obiettivi esterni, consumando banda di rete e risorse di calcolo. (Fonte: **sco.c**)

Impatto Aziendale Potenziale

- **Violazione dei Dati**: Furto sistematico di rubriche di contatti e potenzialmente di informazioni sensibili contenute in documenti e email. (Fonte: **scan.c**)

- **Perdita di Reputazione**: La rete aziendale potrebbe essere utilizzata per inviare email di spam/phishing ad altre organizzazioni, danneggiando l'immagine dell'azienda. (Fonte: **massmail.c**, **xsmtp.c**)

- **Compromissione dell'Infrastruttura**: Controllo totale dei sistemi infetti da parte di un aggressore esterno. (Fonte: **xproxy.c**, **client.c**)

- **Interruzione Operativa**: Degradazione delle performance della rete a causa del traffico generato dal worm per diffondersi e per partecipare agli attacchi DDoS. (Fonte: **massmail.c**, **sco.c**)

Analisi Tecnica Dettagliata

Il malware opera in **tre** fasi distinte:

- Infezione e Persistenza
- Propagazione
- Esecuzione dei payload dannosi

Fase 1: Infezione e Persistenza

- **Vettore di Infezione**: L'ingresso primario è un utente che esegue un allegato email ingannevole. I file sono mascherati da documenti o messaggi importanti e contenuti in archivi .zip o con estensioni eseguibili come .scr, .pif, .exe. (Fonte: **msg.c**, **zipstore.c**)

- **Installazione**: Appena eseguito, il worm si copia nella directory di sistema con il nome taskmon.exe, un nome scelto per confondersi con i processi legittimi di Windows. (Fonte: **main.c**)

- **Persistenza**: Per garantire la sua esecuzione ad ogni riavvio, il malware adotta due tecniche:
  - Worm Principale: Crea una chiave nel registro di sistema in Software\Microsoft\Windows\CurrentVersion\Run, un metodo standard per l'avvio automatico. (Fonte: **main.c**)
- **Backdoor** (shimgapi.dll): Utilizza una tecnica più furtiva e avanzata chiamata COM Hijacking. Modifica una chiave di registro legata a un componente di sistema (Webcheck.dll) per forzare il processo principale di Windows (explorer.exe) a caricare la DLL della backdoor ad ogni avvio. Questo metodo è più difficile da rilevare e rimuovere. (Fonte: **xproxy.c)**

Fase 2: Propagazione

Una volta attivo, l'obiettivo primario del worm è diffondersi il più rapidamente possibile e lo fa in diversi modi come:

- **Raccolta di Indirizzi Email (Harvesting)**:
  - Il worm avvia una scansione aggressiva e ricorsiva di tutti i dischi fissi del computer. (Fonte: **scan.c**)
  - Estrae indirizzi email da una vasta gamma di file: documenti di testo, pagine web salvate (.html), archivi di posta di Outlook Express (.dbx) e, soprattutto, dalla Rubrica Contatti di Windows (.wab). (Fonte: **scan.c**)

- **Diffusione via Email (Mass-Mailing)**:
  - Utilizza un proprio motore di invio (client SMTP) per spedire le email infette. (Fonte: **xsmtp.c**)
  - Falsifica il mittente (Spoofing): Per rendere l'email più credibile, utilizza come mittente un indirizzo raccolto dalla stessa macchina della vittima. (Fonte: **msg.c**)
  - Usa oggetti e testi ingannevoli (es. "Test", "Mail Delivery Failure") per indurre l'utente ad aprire l'allegato. (Fonte: **msg.c**)
  - Tenta di inviare le email prima direttamente, poi provando nomi di server comuni e, come ultima risorsa, utilizzando il server di posta legittimo configurato nel client email dell'utente (es. il server SMTP aziendale), una tecnica efficace per bypassare i filtri di rete perimetrali. (Fonte: **xsmtp.c**)

Fase 3: Esecuzione dei Payload

- **Payload 1: Backdoor e Controllo Remoto**
  - La DLL shimgapi.dll installata dal worm apre una porta TCP nell'intervallo 3127-3199 e rimane in ascolto. (Fonte: **xproxy.c**)
  - Proxy SOCKS4: Trasforma il sistema infetto in un proxy, permettendo all'attaccante di mascherare il proprio indirizzo IP e di instradare traffico illecito attraverso la rete aziendale. (Fonte: **xproxy.c**)
  - Esecuzione di Codice Remoto (RCE): La backdoor possiede un comando speciale che permette all'attaccante di caricare ed eseguire qualsiasi file sulla macchina compromessa. Questo conferisce all'attaccante il controllo totale, consentendogli di installare spyware, ransomware o strumenti per il furto di dati. (Fonte: **xproxy.c**, **client.c**)

- **Payload 2: Attacco DDoS**
  - Questo payload era programmato per attivarsi in una data specifica (1° Febbraio 2004). (Fonte: **main.c**)
  - Una volta attivato, il worm partecipa a un attacco coordinato per sovraccaricare il sito web www.sco.com, utilizzando le risorse di rete della macchina infetta. (Fonte: **sco.c**)

Tecniche di Offuscamento e Anti-Analisi

L'autore ha utilizzato diverse tecniche per rendere il malware più difficile da rilevare e analizzare:

- **Compressione (Packing)**: L'eseguibile finale è compresso con UPX per nasconderne il codice e ridurne le dimensioni. (Fonte: **makefile**)

- **Offuscamento di Stringhe**: Testi critici come nomi di file, chiavi di registro e comandi di rete sono nascosti utilizzando il semplice ma efficace algoritmo ROT13. (Fonte: **lib.c**, **main.c**, **sco.c**, **xproxy.c**)

- **Pulizia dell'Header**: L'eseguibile finale viene "pulito" da informazioni come la data di compilazione per ostacolare le indagini forensi. (Fonte: cleanpe.cpp, **makefile**)

Raccomandazioni e Misure di Mitigazione

(Nota: Questa sezione rappresenta una serie di contromisure e best practice di sicurezza informatica applicate specificamente alle minacce e alle vulnerabilità identificate durante l'analisi del codice che LandaTrackerSPA e UnknownSquad consigliano caldamente)

- **Prevenzione:**
  - **Filtri Email Avanzati**: Implementare soluzioni di sicurezza email che analizzino il contenuto degli archivi .zip (sandboxing) e blocchino le estensioni di file pericolose.
  - **Formazione del Personale**: Condurre campagne di sensibilizzazione periodiche per educare i dipendenti a riconoscere email di phishing e a non aprire allegati sospetti.
  - **Policy di Sicurezza**: Bloccare il traffico P2P sulla rete aziendale. Applicare il principio del privilegio minimo per impedire agli utenti di installare software.

- **Rilevamento:**
  - **Monitoraggio di Rete (Egress Filtering)**: Monitorare e bloccare il traffico in uscita verso porte non standard, come l'intervallo TCP 3127-3199.
  - **Endpoint Detection and Response (EDR)**: Utilizzare soluzioni EDR per rilevare comportamenti anomali come la creazione di file eseguibili in directory di sistema, modifiche a chiavi di registro di avvio automatico (in particolare quelle relative a oggetti COM) e processi che eseguono scansioni massive del file system.

- **Risposta agli Incidenti:**
  - **Isolamento**: Qualsiasi sistema che mostri segni di infezione deve essere immediatamente isolato dalla rete per contenere la diffusione.
  - **Bonifica**: A causa della natura persistente della backdoor (COM Hijacking), la procedura di bonifica più sicura consiste nel re-imaging completo del sistema operativo a partire da un'immagine pulita.
  - **Reset delle Credenziali**: Tutte le credenziali (utente, servizi, etc.) associate alla macchina e all'utente compromessi devono essere immediatamente cambiate.

Differenze col malware originale

Il malware non presenta differenze funzionali significative rispetto al malware Mydoom.A originale, così come è stato analizzato e documentato dai ricercatori di sicurezza durante l'epidemia del 2004. Anzi, la corrispondenza è così precisa da poter affermare che si tratta proprio del codice sorgente originale da cui sono state compilate le versioni del worm diffuse "in the wild".

Ecco un confronto punto per punto tra le caratteristiche note del Mydoom.A originale e il codice che abbiamo analizzato:

Caratteristica	Mydoom.A Originale (Analisi 2004)	Codice Sorgente Inviato	Corrispondenza
Nome Eseguibile	Si installava come taskmon.exe.	Il makefile compila e il codice in main.c installa taskmon.exe.	**Sì**
Payload DDoS	Attacco DDoS contro www.sco.com a partire dal 1° Febbraio 2004.	sco.c contiene il codice per l'attacco DDoS contro www.sco.com	**Sì**
		con la stessa data di attivazione.	
Payload Backdoor	Installava la DLL shimgapi.dll che apriva una porta TCP tra 3127-3199.	xproxy.c (compilato come shimgapi.dll) apre una porta nello stesso range e agisce come backdoor.	**Sì**
Persistenza	Usava una chiave Run ("TaskMon") e il dirottamento di un oggetto COM.	main.c e xproxy.c implementano esattamente queste due tecniche di persistenza.	**Sì**
Propagazione Email	Stessi oggetti ("Test", "Hello", "Mail Delivery System"), nomi allegati e motore SMTP.	msg.c e xsmtp.c contengono la logica identica per la creazione e l'invio delle email.	**Sì**
Propagazione P2P	Si copiava nella cartella di Kazaa con nomi di file specifici (crack, etc.).	p2p.c implementa la stessa funzionalità con la stessa lista di nomi di file.	**Sì**
Data di Termine	Smaltimento della propagazione previsto per il 12 Febbraio 2004.	main.c contiene la stessa data di termine hard-coded.	**Sì**
Offuscamento	Uso di ROT13 per le stringhe e compressione con UPX.	Il codice usa ROT13 ovunque e il makefile specifica l'uso di UPX come ultimo passo.	**Sì**

Uniche differenze notabili

Le analisi del 2004 erano basate sul difficile lavoro di reverse engineering del **file binario** (.exe e .dll), ovvero il programma già compilato e offuscato.

Inoltre, il file analizzato include anche le **utility ausiliarie** (crypt1.c, cleanpe.cpp, bin2c.c) e persino il **client per la backdoor** (client.c). Questi sono strumenti che l'autore usava per creare il malware e che quasi mai vengono trovati.