Relazione 1: Analisi Forense di Win32.Mydoom.a (Variante 2004)

Data: 15 Aprile 2025 **Analista:** Gemini AI **Oggetto:** Analisi tecnica del codice sorgente del worm Win32.Mydoom.a

1. Introduzione Questa relazione dettaglia l'analisi forense condotta sul codice sorgente fornito, identificato come Win32.Mydoom.a, un worm apparso per la prima volta nel Gennaio 2004. L'analisi si basa sull'esame statico dei file sorgente in linguaggio C (.c, .h) e dei file di supporto (makefile, _readme.txt, etc.) per determinare le funzionalità, i meccanismi di propagazione, le tecniche di evasione e i payload del malware.

2. Inizializzazione e Persistenza (main.c)

- **Esecuzione Iniziale:** Il malware verifica se è la prima esecuzione controllando la presenza di una specifica chiave di registro (Software\Microsoft\Windows\CurrentVersion\Explorer\ComDlg32\Version - nome offuscato con ROT13).
- **Mutex:** Crea un mutex (SwebSipcSmtxS0 - nome offuscato con ROT13) per garantire che solo un'istanza sia attiva. Se il mutex esiste già (non è la prima istanza), termina l'esecuzione.
- **Installazione:** Copia il proprio file eseguibile nella directory di Sistema o Temporanea di Windows con il nome taskmon.exe (offuscato con ROT13).
- **Persistenza Avvio:** Si registra per l'avvio automatico creando un valore (TaskMon - offuscato) nella chiave di registro Run (HKLM o HKCU\Software\Microsoft\Windows\CurrentVersion\Run) che punta al file taskmon.exe installato.
- **Drop Backdoor:** Decifra (decrypt1_to_file) e scrive su disco il componente backdoor (xproxy_data incluso via #include) salvandolo come shimgapi.dll (offuscato con ROT13) nella directory di Sistema o Temp.
- **Persistenza Backdoor:** La DLL backdoor (shimgapi.dll) viene caricata (LoadLibrary) e assicura la propria persistenza tramite COM Hijacking, modificando la chiave di registro del CLSID di Webcheck.dll ({E6FB5E20-DE35-11CF-9C87-00AA005127ED} - offuscato) per essere caricata da Explorer (xproxy.c: shellsvc_attach). Su Win9x, tenta di registrarsi come processo di servizio (xproxy.c: regsvc9x).
- **Distrazione Visiva:** Alla prima esecuzione, lancia un thread (sync_visual_th) che crea un file temporaneo con dati casuali e lo apre con Notepad.

- **Date di Attivazione/Termine:** Controlla la data corrente. Il payload DDoS (sco.c) si attiva dopo il 1 Febbraio 2004. L'intero worm cessa le attività principali dopo il 12 Febbraio 2004 (sync_checktime).

3. Meccanismi di Propagazione

- **Email (Mass Mailing):**
    - *Raccolta Indirizzi (scan.c):* Scansiona file locali (.txt, .htm, .html, .wab, .dbx, etc.) e Temporary Internet Files per estrarre indirizzi email.
    - *Invio (massmail.c, xsmtp.c, xdns.c):* Utilizza un proprio motore SMTP per inviare email. Risolve i record MX dei domini destinatari tramite API o query DNS dirette. Può tentare l'invio anche tramite l'SMTP predefinito dell'utente. Gestisce una coda di invio e thread multipli per l'invio.
    - *Contenuto Email (msg.c):* Genera dinamicamente email con mittente, oggetto e corpo variabili (spesso offuscati con ROT13). Allega una copia di sé, spesso compressa in un file ZIP (zipstore.c), usando nomi file e estensioni comuni o tecniche di inganno (spazi multipli, doppia estensione). L'allegato è codificato in Base64.
- **Peer-to-Peer (p2p.c):**
    - Copia se stesso nella cartella di condivisione della rete P2P KaZaA (percorso letto dal registro), usando nomi file casuali scelti da una lista (es. winamp5, icq2004-final, activation_crack - offuscati con ROT13).

4. Payload

- **Backdoor (xproxy.c):** Il componente shimgapi.dll apre una backdoor TCP sulle porte 3127-3198. Agisce come un server SOCKS4, permettendo all'attaccante di usare la macchina infetta come proxy. Include una funzionalità per ricevere ed eseguire file eseguibili inviati dall'attaccante (socks4_exec).
- **DDoS Attack (sco.c):** Tra il 1 e il 12 Febbraio 2004, lancia un attacco DoS (Denial of Service) inviando un flood di richieste HTTP parziali verso www.sco.com (offuscato con ROT13) sulla porta 80, utilizzando numerosi thread.

5. Tecniche di Evasione e Offuscamento

- **Offuscamento Stringhe:** Uso estensivo di ROT13 (lib.c) per nascondere stringhe (nomi file, chiavi registro, host, comandi).
- **Packing:** Il makefile indica l'uso di UPX per comprimere l'eseguibile finale.

- **Cifratura Payload:** La DLL backdoor è memorizzata nel corpo del worm cifrata con XOR.
- **Modifica PE Header:** Utilizzo di cleanpe.exe per azzerare il timestamp del file eseguibile.
- **Nomi File Legittimi:** Utilizzo di taskmon.exe e shimgapi.dll.
- **Persistenza Nascosta:** Uso di chiavi di registro Run e COM Hijacking.
- **Filtri Email:** Evita l'invio a domini di sicurezza; uso di allegati ZIP e nomi file ingannevoli.

6. Conclusione Win32.Mydoom.a è un worm complesso per l'epoca (2004), che combina propagazione via email e P2P con un payload DDoS mirato e una backdoor SOCKS4 funzionale. Utilizza diverse tecniche di offuscamento (ROT13, packing, cifratura XOR) e persistenza per massimizzare la sua diffusione e il suo impatto. L'analisi del codice sorgente conferma le funzionalità riportate e fornisce dettagli tecnici sulla sua implementazione.

Relazione 2: Confronto tra MyDoom (2004) e una Variante Ipotetica (2025)

Data: 15 Aprile 2025 **Analista:** Gemini AI **Oggetto:** Evoluzione delle minacce: Confronto tra Win32.Mydoom.a (2004) e una potenziale variante del 2025

1. Introduzione Il worm MyDoom (Win32.Mydoom.a), apparso nel 2004, è stato uno dei malware a più rapida diffusione della storia. Questa relazione confronta le capacità tecniche di quella variante originale, analizzata tramite il suo codice sorgente, con le capacità che una ipotetica variante moderna, aggiornata al 2025, potrebbe possedere, riflettendo l'evoluzione delle minacce informatiche negli ultimi 21 anni.

2. Meccanismi di Propagazione

- **MyDoom 2004:** Si basava principalmente sull'invio massivo di email con allegati eseguibili (diretti o zippati) e sulla condivisione di file sulla rete P2P KaZaA. L'ingegneria sociale era rudimentale (oggetti e nomi file generici).
- **MyDoom 2025 (Ipotetico):** Sfrutterebbe vettori molto più diversificati e sofisticati:
    - *Email:* Campagne di spear-phishing mirate, email con link malevoli invece di allegati diretti, sfruttamento di vulnerabilità nei client di posta.
    - *Web:* Exploit kit che colpiscono vulnerabilità dei browser e dei plugin.

- *Altri:* Diffusione tramite drive USB, reti social, app di messaggistica, compromissione di aggiornamenti software (supply chain attack), vulnerabilità in dispositivi IoT o servizi cloud. La componente P2P KaZaA sarebbe obsoleta e probabilmente sostituita o rimossa.

3. Payload e Obiettivi

- **MyDoom 2004:** I payload principali erano una backdoor SOCKS4 (per controllo remoto e proxy) e un attacco DDoS mirato e limitato nel tempo contro www.sco.com. L'obiettivo sembrava essere la disruption e la creazione di una base per inviare spam o altri attacchi tramite la backdoor.
- **MyDoom 2025 (Ipotetico):** Gli obiettivi sarebbero probabilmente più focalizzati sul profitto o sullo spionaggio:
  - *Payload:* Ransomware per estorcere denaro, data stealer per rubare credenziali bancarie, dati personali, proprietà intellettuale, cryptominer per sfruttare le risorse della vittima, moduli di spionaggio avanzati.
  - *Backdoor:* Sostituita da un RAT (Remote Access Trojan) completo con funzionalità estese (keylogging, cattura schermo, accesso webcam/microfono, gestione file system).
  - *Botnet:* Integrazione in botnet per eseguire attacchi DDoS più potenti e complessi (attacchi volumetrici, applicativi, multi-vettore) contro una gamma più ampia di target, o per vendere l'accesso alle macchine infette.

4. Comando e Controllo (C&C)

- **MyDoom 2004:** La comunicazione C&C avveniva tramite connessioni dirette alla backdoor sulle porte TCP 3127-3198. Il traffico non era cifrato.
- **MyDoom 2025 (Ipotetico):** I meccanismi C&C sarebbero progettati per la massima resilienza e discrezione:
  - *Comunicazione:* Traffico C&C cifrato (SSL/TLS o protocolli custom).
  - *Infrastruttura:* Utilizzo di DGA (Domain Generation Algorithms) per rendere difficile il blocco dei domini C&C, uso della rete Tor o di proxy P2P moderni per anonimizzare il traffico, utilizzo di servizi legittimi (piattaforme cloud, social network, pastebin) per ospitare o scambiare comandi.

5. Tecniche di Evasione e Offuscamento

- **MyDoom 2004:** Usava tecniche basilari come ROT13, packing UPX, azzeramento del timestamp PE e nomi file comuni.
- **MyDoom 2025 (Ipotetico):** Impiegherebbe un arsenale molto più avanzato per eludere le moderne soluzioni di sicurezza (Antivirus, EDR, XDR, Sandbox):
    - *Offuscamento:* Packers/crypter multilivello, polimorfismo/metamorfismo per cambiare il codice ad ogni infezione.
    - *Anti-Analisi:* Tecniche anti-VM, anti-debug, anti-sandbox (rilevamento ambiente, ritardi nell'esecuzione).
    - *Esecuzione:* Tecniche "fileless" (esecuzione in memoria senza scrivere file su disco), code injection in processi legittimi, living-off-the-land (abuso di strumenti di sistema).
    - *Persistenza:* Metodi più stealth (es. WMI event subscription, scheduled tasks, modifica bootloader/UEFI).
    - *Rootkit:* Possibile inclusione di componenti rootkit (user-mode o kernel-mode) per nascondere processi, file e connessioni di rete.

6. Targeting e Piattaforme

- **MyDoom 2004:** Mirava principalmente a sistemi Windows (XP/2000) con un approccio opportunistico e di massa.
- **MyDoom 2025 (Ipotetico):** Potrebbe essere multi-piattaforma (Windows, macOS, Linux, forse anche mobile/IoT). Potrebbe essere usato sia per campagne di massa che per attacchi mirati (APT-like) contro specifiche organizzazioni o settori industriali. Sfrutterebbe vulnerabilità più recenti nei sistemi operativi e nelle applicazioni.

7. Conclusione Il confronto tra MyDoom 2004 e una sua ipotetica evoluzione al 2025 evidenzia la drastica sofisticazione raggiunta dal malware. Se MyDoom 2004 era una minaccia significativa basata su diffusione rapida e disruption, una variante moderna sarebbe probabilmente focalizzata su obiettivi finanziari o di spionaggio, utilizzando tecniche di propagazione, C&C ed evasione estremamente più avanzate e difficili da contrastare. Questo sottolinea la continua necessità di evolvere le strategie e le tecnologie di difesa informatica per affrontare minacce sempre più complesse e insidiose.