Visualisierung von Malware-Verhalten

Philipp Trinius

24. Juli 2009

Würmer, Trojaner und Bots, kurz Malware, stellen eine in den letzten Jahren immer größer werdende Herausforderung dar. Die Anzahl neuer Malware steigt in dramatischer Weise an und macht eine manuelle Auswertung unmöglich. Obwohl es sich vielfach nur um Varianten bereits bekannter Schadsofteware handeln dürfte, müssen Analysten jedes Sample doch erst analysieren, um dies feststellen zu können.

Bei der Malwareanalyse wird grundsätzlich zwischen *statischen* und *dynamischen* Ansätzen unterschieden. Bei der statischen Analyse versucht man den Code der Malware durch *reverse engineering* aus dem Samples zu extrahieren, um diesen im Anschluss eingehend untersuchen zu können. Die dynamische Analyse hingegen beobachtet und bewertet das Verhalten der Malware während ihrer Ausführung. Die Malware wird in einer kontrollierten Umgebung – einer Sandbox – ausgeführt und die von ihr durchgeführten Operationen werden aufgezeichnet. Anschließend kann der Analyst die Malware dem aufgezeichneten Verhalten entsprechend bewerten und klassifizieren. Die dynamische Analyse lässt sich damit im Gegensatz zur statischen Analyse einfach automatisieren.

Die Ausgabe der Sandbox stellt ein detaillierter Report dar, der aus mehreren hundert Einträgen bestehen kann. Um eine Aussage über das Sample treffen zu können, muss dieser von einem Analysten gelesen werden. Das heißt, auch bei der Sandbox-Analyse hat der Analyst letztlich jedes Sample, jetzt in einer lesbaren Form, vor sich. Die Länge und die Vielzahl der darin enthaltenen Details kann einen menschlichen Analysten dabei leicht überfordern, was dazu führt, dass die wichtigen Details untergehen. Das Ziel muss es also sein, dem Analysten die Entscheidung zu erleichtern welchen Report und damit welches Sample er eingehender untersuchen soll, indem ihm die Analyseergebnisse in einer schneller fassbaren Form präsentiert werden. An dieser Stelle setzt die im Folgenden kurz vorgestellte Kombination aus Abstraktion und Visualisierung von Malwareverhalten an.

Aktuell werden die Reports der CWSandbox als *Treemaps* und als *Threadgraphen* visualisiert. Die Treemap Darstellung, siehe Abbildung 1, soll dabei einen schnellen Überblick über die in dem Report aufgezeichneten Ereignisse – die CWSandbox hooked circa 120 Windows API-Calls – und deren Frequenz geben. Auf der x-Achse sind hierzu die einzelnen Sektionen, in die sich die Ereignisse unterteilen, aufgelistet. Sektionen sind zum Beispiel die Gruppe aller Netzwerkoperationen (*network_section*) oder die der Zugriffe auf die Registry (*registry_section*). Die einzelnen Sektionen sind zusätzlich in die jeweils beobachteten Operationen unterteilt. Die in Abbildung 1 enthaltenen Label dienen hier lediglich dem Verständnis und zeigen, dass sowohl am Dateisystem (grün codiert) als auch an der Registry (blau) Veränderungen durchgeführt wurden. Welche Änderungen im Detail vorgenommen wurden, kann der Treemap nicht entnommen werden. Diese Informationen sind im gewählten Abstraktionslevel nicht mehr enthalten.

Als zweite Visualisierung liefert der Threadgraph genaue Informationen zu den während der Analyse beobachteten Threads, siehe Abbildung 2. Hier wird jeder Thread mit den von ihm ausgeführten Operationen sichtbar. Entlang der x-Achse chronologisch aufgereiht werden die einzelnen von den Threads ausgeführten Operationen auf der y-Achse aufgetragen. Ein Analyst kann damit den Ablauf und den Zusammenhang zwischen den einzelnen API-Calls auf den ersten Blick erfassen und gewinnt einen guten Überblick über die von dem Sample angestoßenen Threads. Neben den Operationen, die von den Threads ausgeführt werden, zeigt der Threadgraph auch welche Operationen nicht ausgeführt werden. Im vorliegenden Beispiel findet beispielsweise keinerlei Netzwerkkommunikation statt, was für die Beurteilung eines Samples entscheident sein kann.

Sowohl die Treemap- als auch die Threadgraph-Darstellung kann Analysten bei der Entscheidung unterstützen, ob sie einen Report und damit ein Sample genau untersuchen sollen. Sie bieten einen schnellen Überblick und sollen in kommenden Versionen – über eine Zoomfunktionalität soll zum Beispiel das Anreichern der Darstellung mit zusätzlichen Informationen umgesetzt werden – auch das detaillierte Untersuchen des Reports ermöglichen. Zusätzlich zu den beschriebenen Vorteilen der graphischen Darstellung eines einzelnen Reports bietet die Visualisierung mehrerer Reports die Möglichkeit, mit Bilderkennungs- und Clusteringalgorithmen bereits existierende Verfahren auch auf die Malwareanalyse zu portieren. Erste Versuche mit circa 2000 Samples aus 13 Familien haben gezeigt, dass die Treemaps und Threadgraphen innerhalb der Malwarefamilien sehr ähnlich sind und sich gleichzeitig gut von denen anderer Familien abgrenzen. Auch bei der Analyse infizierter Officedokumente konnten die von den Dokumenten zusätzlich angestoßenen Threads und Operationen sichtbar gemacht werden.

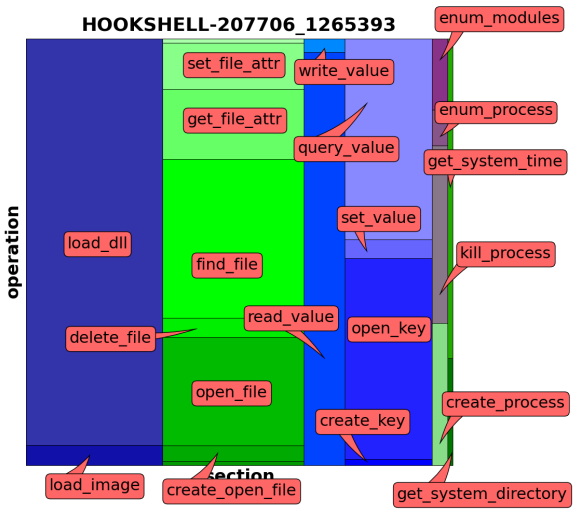

Abbildung 1: Treemap eines Hookshell Malware Sample

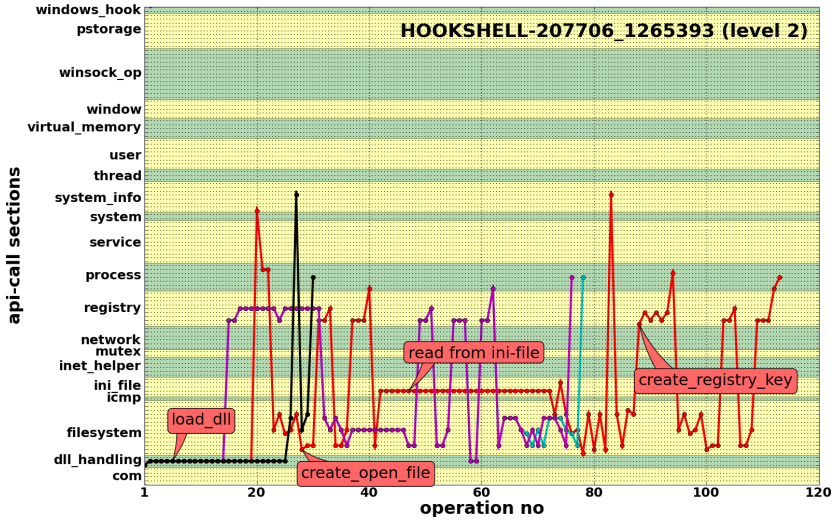

Abbildung 2: Level 2 Threadgraph eines Hookshell Malware Sample