Einlasskontrolle

Einschätzen, ob man einer Datei besser misstrauen sollte

Da liegt sie nun im Download-Ordner, die frisch heruntergeladene ausführbare Datei. Doch ist ein Doppelklick darauf womöglich gefährlich? Für eine erste Einschätzung reichen Sekunden.

Von Axel Vahldiek

Bei einer frisch heruntergeladenen oder per Mail empfangenen Datei besteht immer ein gewisses Risiko, dass sie einen Schädling enthält. Und das gilt keineswegs nur für Dateien aus dubiosen Quellen, sondern auch für Dateien von seriösen Anbietern oder Absendern, etwa weil sie selbst Opfer eines Angriffs wurden. Doch wie prüft man so eine Datei? In unseren „Analysiert“-Artikeln haben wir exemplarisch gezeigt, wie viel Aufwand Profis in gründliche Untersuchungen stecken [1, 2]. Einen ersten Eindruck bekommen Sie jedoch auch ohne Expertenwissen und in Sekundenschnelle. Mit den Tipps aus diesem Artikel prüfen Sie nicht nur die Signatur mit einem einzigen Mausklick, sondern lassen die Datei auch noch zugleich von über 60 Virenscannern untersuchen.

Um aber eines noch mal in aller Deutlichkeit zu sagen: Die nachfolgend vorgestellte Methode zur Schnellprüfung stellt keineswegs sicher, dass eine Datei

wirklich unschädlich ist. Stellen Sie sich das ähnlich wie eine Fußgängerampel vor: Wenn es dumm läuft, können Sie auch bei Grün überfahren werden. Doch wenn die Ampel rot leuchtet, wissen Sie, dass sie stehenbleiben beziehungsweise in diesem Fall eben bloß nicht doppelklicken sollten.

Das Werkzeug

Möglich macht das Ganze die Freeware Sigcheck von Sysinternals. Autor ist Mark Russinovich, der auch die bekannten Systemwerkzeuge Autoruns, Process Explorer und Process Monitor geschrieben hat und seit vielen Jahren für Microsoft arbeitet. Sie finden das wenige hundert KByte kleine Programm zusammen mit allen anderen Sysinternals-Tools unter https://live.sysinternals.com. Sigcheck.exe bietet kein GUI, ist also ein reines Kommandozeilenprogramm. Das Folgende beschreibt zuerst die Optionen und Möglichkeiten von Sigcheck und anschließend, wie Sie die Prüfung mit Sigcheck so konfigurieren, dass künftig ein simpler Mausklick reicht.

Sigcheck war ursprünglich nur zum Prüfen der Signatur von Dateien gedacht, genauer, ob das als Signatur dienende Zertifikat von einer Zertifizierungsstelle ausgestellt wurde, der Windows vertraut. Das muss nicht direkt sein, sondern kann über mehrere andere vertrauenswürdige Zertifikate hinweg erfolgen: Zertifizierungsstelle A vertraut Zertifizierungsstelle B, die wiederum Zertifizierungsstelle C und so weiter („Zertifizierungskette“). Eine gültige Signatur hinterlegt der Hersteller als Nachweis in der Datei, dass sie wirklich von ihm stammt und dass er dazu steht. Über die Fehlerfreiheit oder Ungefährlichkeit einer Anwendung sagt die Signatur damit zwar nichts aus, doch Sie wissen dann, an wen Sie sich bei Problemen wenden können. Und Hersteller von Programmen mit gültigen Signaturen bemühen sich üblicherweise schon aus Angst vor einem Image-Schaden, dass ihre Programme möglichst fehlerfrei und ungefährlich sind.

Die Signatur ist an genau diese Datei gebunden, wird also ungültig, wenn sich auch nur ein Bit davon ändert. Sofern Sigcheck also ausgibt, dass eine Datei von Microsoft signiert wurde, können Sie ziemlich sicher sein, dass das auch so ist – nur „ziemlich“ sicher, weil es leider in Einzelfällen vorkommt, dass die Signatur gefälscht oder gestohlen ist. Das ist aber sehr aufwendig, sodass momentan die meisten Angreifer das unterlassen.

Seit einiger Zeit kann Sigcheck noch etwas anderes: Virustotal.com befragen. Diese Website wird von Google betrieben. Wenn man dort eine Datei hochlädt, wird sie von über 60 Virenscannern geprüft. Das Ergebnis der Virustotal-Abfrage durch Sigcheck bekommen Sie üblicherweise bereits nach Sekunden, weil das Programm im ersten Anlauf nicht die ganze Datei, sondern bloß einen Hash hochlädt.

Einrichten

Damit Sie nicht für jeden Sigcheck-Aufruf lange Kommandozeilenbefehle eintippen müssen, finden Sie unter ct.de/y8bm die Batch-Datei sigcheck.bat, die den Job für Sie erledigt. Laden Sie diese sowie sigcheck.exe herunter und packen Sie beide gemeinsam in einen beliebigen Ordner. Als Nächstes klicken Sie im Kontextmenü der Batchdatei sigcheck.bat auf „Kopieren“. Drücken Sie nun die Tastenkombination Windows+R, es öffnet sich der „Ausführen“-Dialog. Dort tippen Sie ein:

```
shell:sendto
```

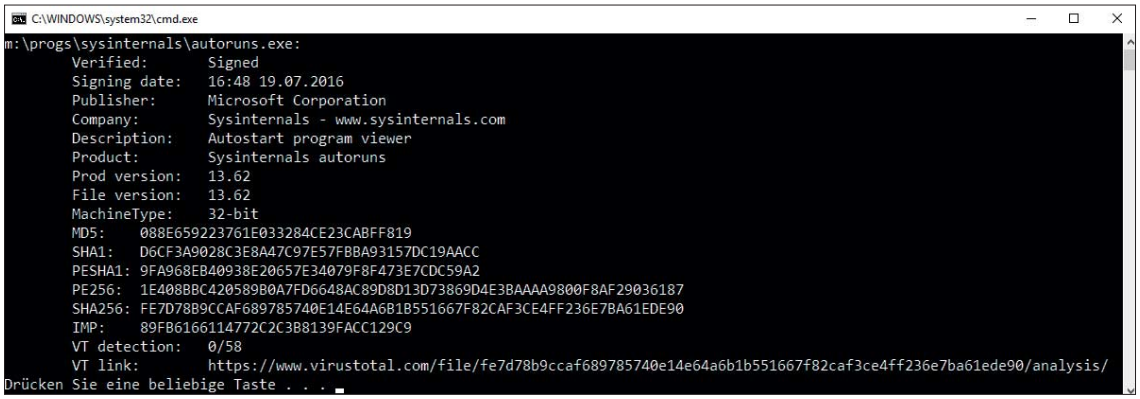
Nach dem Bestätigen mit Enter öffnet sich der Ordner, in dem die Verknüpfungen des „Senden an“-Menüs aus dem Kontextmenü von Dateien und Ordnern liegen. Dort rechtsklicken Sie in einen leeren Bereich und wählen „Verknüpfung einfügen“ – fertig. Wenn Sie mögen, können Sie die Verknüpfung nach Gusto umbenennen.

Ab sofort können Sie jede Datei per „Senden an“-Menü an Sigcheck übergeben und bekommen anschließend ein Kommandozeilenfenster mit den Prüfungsergebnissen. Beim ersten Aufruf müssen Sie einmalig die Lizenzbestimmungen von Sysinternals sowie von Virustotal.com abnicken, ab dem zweiten Aufruf geht es ohne.

Wie Sigcheck genau vorgeht, können Sie konfigurieren. Dazu öffnen Sie die Batch-Datei im Texteditor. Die einzig relevante Zeile ist die zweite, sie lautet:

```
sigcheck.exe -vr -h %1
```

Die Option `-vr` sorgt dafür, dass der Hash der zu überprüfenden Datei bei Virustotal.com hochgeladen und dass die Ergebnisseite im Standardbrowser angezeigt wird, sofern mindestens ein Virenscanner Alarm schlägt. Finden alle Scanner die Datei harmlos, erscheint nur das Prüfergebnis im Kommandozeilenfenster. Wenn Sie wollen, können Sie die Option ergänzen zu `-vrs`. Das ist für den Fall gedacht, dass Virustotal den hochgeladenen Hash-Wert nicht erkennt, dann lädt Sig-



Unser Skript bereitet seine Ausgabe zwar nicht gerade hübsch auf, informiert Sie aber in Sekundenschnelle darüber, ob eine Anwendung signiert ist und was über 60 Virenscanner darüber denken.

Eine gültige Signatur weist zwar normalerweise darauf hin, dass eine Datei vertrauenswürdig ist, doch gibt es auch Ausnahmen. In diesem Fall schlagen gleich reihenweise Virenscanner Alarm, womit trotz gültiger Signatur klar ist: bloß kein Doppelklick auf die geprüfte Datei!

check die Datei selbst automatisch zur weiteren Prüfung hoch.

Wenn Sigcheck die komplette Zertifikatskette ausgeben soll, ergänzen Sie hinter `-vr` die Option `-i`. Dadurch wird die Ausgabe allerdings erheblich länger und damit unübersichtlicher.

Die Option `-h` lässt Sigcheck zusätzlich verschiedene Hash-Werte für die untersuchte Datei ausgeben (MD5, SHA256, …). Für eine Prüfung auf Vertrauenswürdigkeit ist das eigentlich nicht erforderlich, spart aber in manch anderen Situationen ein zusätzliches Hash-Programm – falls Sie das nicht brauchen, streichen Sie die Option einfach. Das `%1` am Ende ist keine Option, sondern eine hier unverzichtbare Variable, die für die an die Batch-Datei übergebene Datei steht.

Ausgabe

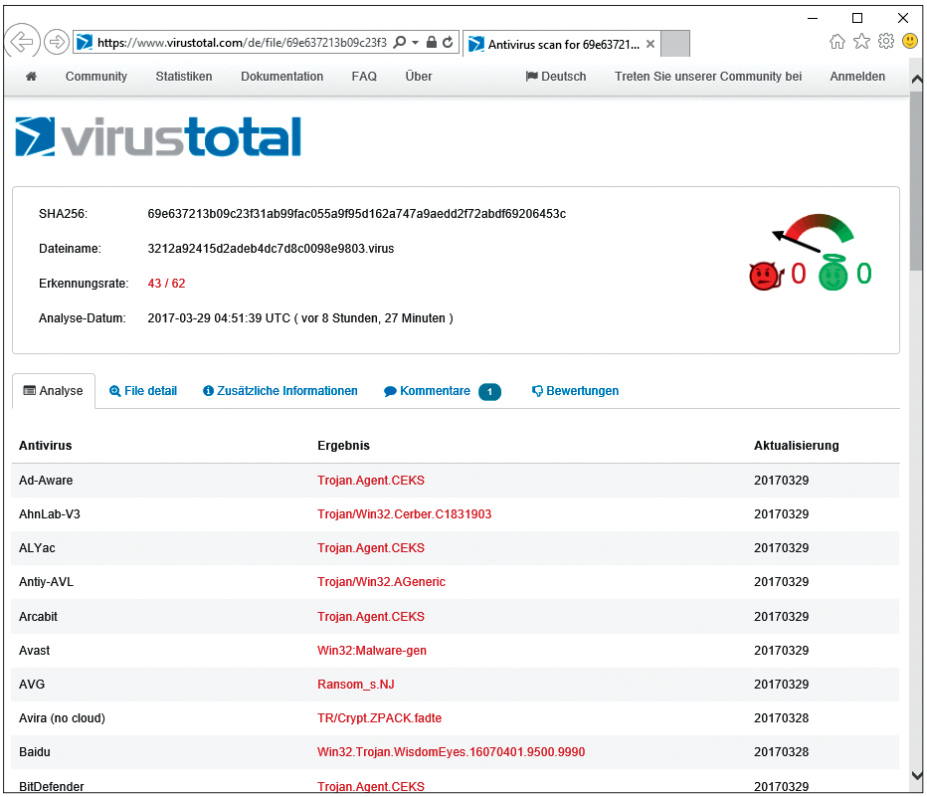
Die Ausgabe des Skripts im Kommandozeilenfenster verdient zugegebenermaßen keinen Schönheitspreis, sondern fasst einfach nur in drögen Textzeilen und zum Teil mit Abkürzungen die Ergebnisse zusammen.

Die erste Zeile beginnt mit „Verified“, dahinter steht normalerweise entweder „Signed“ oder „Unsigned“. Zumindest bei großen Firmen wie Microsoft und Google sollte die Datei grundsätzlich signiert sein, auch wenn Ausnahmen die Regel bestätigen.

In manchen Fällen ist zwar eine Signatur vorhanden, wird aber nicht als vertrauenswürdig eingestuft. Sie erkennen das an Meldungen wie „Die digitale Signatur des Objekts konnte nicht bestätigt werden“, „Ein Zertifikat wurde explizit durch den Aussteller gesperrt“ oder „Eine Zertifikatkette zu einer vertrauenswürdigen Stammzertifizierungsstelle konnte nicht aufgebaut werden“. Die Datei ist damit ebenfalls erst mal nicht vertrauenswürdig.

Die nächste Zeile nennt ein Datum, und zwar entweder das der Erstellung des Zertifikats („Signing date“) oder das der Datei („Link date“), falls das Zertifikat fehlt. Es folgen derjenige, der das Programm veröffentlicht hat („Publisher“), die Firma, die das Programm geschrieben hat („Company“), Beschreibung („Description“), Produktname und -versionsnummer sowie die Versionsnummer der Datei. Bei „MachineType“ steht, ob das Programm 32- oder 64-bittig ist.

Die Ausgabe geht weiter mit sechs verschiedenen Hash-Werten. Es folgt die „VT detection“, die Zeile nennt das Ergebnis der Virustotal-Überprüfung. Im Idealfall steht hier „0/61“, wobei die Zahl hinter dem Schrägstrich mitunter leicht variiert – manches wird nicht von jedem



Wenn auch nur ein Virenscanner Alarm schlägt, öffnet sich eine Website mit detaillierten Angaben.

Scanner geprüft. Entscheidend ist die 0 vorne, denn dann hatte kein einziger Scanner etwas zu meckern. Der in der Zeile darunter stehende Link führt zur Ergebnis-Seite der Prüfung, dort können Sie sie detailliert betrachten.

Auswerten

Grundsätzlich sollten Sie alle Programme, die keine gültige Signatur aufweisen, erst einmal mit Vorsicht behandeln und auf weitere Details achten. Wenn einzelne Zeilen der Ausgabe nicht ausgefüllt sind (es steht dann dort „n/a“), deutet das auf Pfusch beim Erstellen der Datei hin. Das kann ein Alarmsignal sein, vor allem, wenn ein großes Unternehmen als Publisher genannt ist. Auch Ihnen oder gar Google unbekannte Publisher können ein Alarmsignal sein, müssen es aber nicht. Denn vielleicht ist es auch nur das Erstlingswerk eines bislang unbekannten, aber seriösen Programmierers.

Falls mindestens einer der Virustotal-Virenscanner Alarm schlägt, öffnet sich im Browser die Website von Virustotal mit einem detaillierten Bericht. Mitunter zeigen Symbole, wie andere Nutzer das Ergebnis einschätzen. Einen weiteren Anhaltspunkt gibt, ob die Alarme nur von Scanner-Exoten oder auch von den großen Scanner-Herstellern stammen. Alarme von Exoten kann man eher ignorieren. Mitunter handelt es sich allerdings doch nicht um einen Fehlalarm: Es kommt durchaus vor, dass ein Virenscanner einen besonders frischen Schädling vor allen anderen entdeckt, und das gilt für bekannte ebenso wie für exotische Scanner. Wer ganz sichergehen will, löscht die heruntergeladene Datei, statt sie zu starten.

Falls Virustotal.com den von Sigcheck übermittelten Hash nicht kennt, kann das ebenfalls ein Alarmsignal sein, muss aber nicht. Wenn Sie beispielsweise erstmals ein selbstgeschriebenes Skript untersuchen lassen, kann Virustotal die Datei zuvor ja noch nicht geprüft haben. Anders sieht es aus, wenn Virustotal den Hash einer prominenten Software nicht kennt. Das kann bedeuten, dass bloß soeben eine ganz neue Version erschienen ist (dann wiederholen Sie das Ganze noch mal nach einigen Stunden oder Tagen), aber auch, dass die Download-Seite infiltriert wurde. Denn normalerweise werden verbreitete Dateien so oft bei Virustotal hochgeladen, dass sie regelmäßig geprüft werden und der Hash demzufolge längst bekannt ist.

Empfehlung

Wenn die Prüfergebnisse eindeutig sind, ist die Empfehlung einfach: Sofern die Datei von einem bekannten Anbieter signiert ist und kein Virenscanner etwas zu meckern hat, ist sie wahrscheinlich harmlos – obwohl, um das noch einmal zu betonen, es keine Garantie dafür gibt, dass dem wirklich so ist. Wenn hingegen die Signatur fehlt oder Seltsamkeiten aufweist und gleich mehrere Virenscanner anschlagen, können Sie es mit den Hinweisen aus dem Kasten probieren – oder Sie gehen auf Nummer sicher und löschen die Datei kurzerhand. *(axv@ct.de)* c't

Literatur

[1] Olivia von Westernhagen, Werbung statt Spielspaß, Analysiert: PS3-Emulator als Schafspelz, c't 2/17, S. 172
[2] Olivia von Westernhagen, Feind aus dem Word-Dokument, Analysiert: Das Comeback der Makro-Malware, c't 5/17, S. 142

sigcheck.bat und sigcheck.exe:
ct.de/y8bm

Grenzfälle

Von Peter Siering

Für erfahrene Anwender kann es gute Gründe geben, die Einschätzung zu ignorieren, dass eine Software als gefährlich einzustufen ist – das muss allerdings im Einzelfall geprüft und abgewogen werden. Hilfreich dabei sind Einsichten in die Art und Weise, wie Antivirus-Software vorgeht: Sie prüft auf Signaturen, die eindeutig einen Schädling identifizieren. Sie sucht Muster, die erfahrungsgemäß typisch für Schädlinge sind (Heuristik). Sie fragt bei den Cloud-Diensten der Hersteller nach, ob dort eventuell bereits Erkenntnisse vorliegen, oder sendet Code-Proben unbekannter Programme dorthin, um sie dort eingehend zu untersuchen.

Obendrein stufen die Hersteller manche Software als „potentially unwanted application“ (pua) ein: Programme, die sich als zweifelhafte Toolbar in Browsern breitmachen, die Windows-Passwörter zurücksetzen, die anderen Systemen übers Netz auf den Zahn fühlen, mit denen sich PCs fernsteuern lassen oder die Fenster verstecken. Wer so etwas auf einem PC der Schwiegermutter vorfindet, ohne dass die Herkunft klar ist, muss davon ausgehen, dass etwas faul ist. In einem Notfall-System wie unserem Notfall-Windows aus c't 26/16 machen solche Programme hingegen die Essenz aus.

Wenn man eine Datei von Virustotal analysieren lässt und nur eine Minderheit der dort eingespannten Programme Alarm schlägt, lohnt ein näherer Blick auf die Ergebnisspalte: Tauchen dort „gen“ oder „heur“ auf, dann handelt es sich eben um keinen eindeutigen Schädlingsfund, sondern nur um einen Verdacht – besonders der weniger prominente Teil der Zunft wittert schnell mal Gefahr, wo keine besteht. Stammt die Datei aus seriöser Quelle, die womöglich sogar diesen Umstand dokumentiert, muss man nicht gleich in Panik verfallen.

Eine nähere Untersuchung fällt schwer. Es gibt Dienste, an die man solche Dateien schicken kann und die sich in einer Sandbox an einer Analyse versuchen. Hundertprozentige Gewissheit liefert das nicht: Ein enthaltener Schädling könnte die Sandbox erkennen und verdächtige Funktionen erst gar nicht auslösen, etwa Netzwerkzugriffe, die ihn verrieten. Letztlich gibt es ohne detaillierte Code-Analyse keine abschließende Gewissheit – auch Software ohne Befund, obendrein aus vertrauenswürdigen Quellen, kann Überraschungen bergen.