Multivirenscanner in virtueller Maschine

Um später zu erkennen, ob unsere modifizierte Malware noch von Virenscannern erkannt wird, war es zunächst einmal notwendig, eine virtuelle Maschine mit mehreren Virenscanner-Engines aufzusetzen.

Da die Überprüfung der zu scannenden Objekte mit allen vorhandenen Virenscannern so komfortabel wie möglich sein sollte, haben wir uns dazu entschieden, die einzelnen Virenscanner mit einem Python-Skript anzusprechen. Dazu war es nötig, dass alle verwendeten Virenscanner die Funktion bieten, Objekte über die Kommandozeile zu scannen. Dies ging jedoch nicht bei allen Herstellern aus der Produktbeschreibung sofort hervor.

Um die Nutzung mehrerer Virenscanner auf einen System zu ermöglichen, musste nach jeder Virenscanner-Installation der Echtzeitschutz in der grafischen Oberfläche deaktiviert werden und anschließend alle Dienste, die nicht zwingend für den Scanvorgang nötig sind, über msconfig.exe bzw. services.msc abgeschaltet werden.

Die ersten Versuche starteten wir mit dem ClamAV, da uns bei diesem Produkt bereits bekannt war, dass das Scannen per Kommandozeile unterstützt wird. Um ein Objekt zu scannen musste über das Python-Skript die clamscan.exe mit dem Parameter --database=[Datenbankpfad] und dem Pfad des zu scannenden Objekts aufgerufen werden. Zusätzlich haben wir am Ende noch einen weiteren Parameter (--log=[Textdatei]) verwendet, um die Ausgabe des Virenscanners in einer Textdatei zu speichern.

Der zweite verwendete Virenscanner war die freie Version von AVG AntiVirus. Hier wird der Scanvorgang über die avgscanx.exe mit dem Parameter /SCAN=[Objektpfad] gestartet. Das zugehörige Logfile erstellt hier der Parameter /REPORT=[Textdatei].

Ein weiterer Virenscanner, den wir verwendeten, war Avira Free AntiVirus. Hier wurde die Kommandozeilenfunktion nicht standardmäßig mitinstalliert, sondern musste separat heruntergeladen und in das Installationsverzeichnis kopiert werden. Ist dies erledigt, so kann der Scanvorgang über die scancl.exe mit anschließender Angabe eines Pfades des zu scannenden Objekts gestartet werden. Ein Logfile kann zusätzlich über den Parameter --log=[Textdatei] erzeugt werden.

Unser nächstes verwendetes Produkt war Microsoft Security Essentials. Um hiermit einen Konsolenscan zu starten, war es notwendig folgende Parameter anzugeben:

|  |  |
| --- | --- |
| -scan | Um den Scanvorgang zu starten |
| -DisableRemediation | Um ein automatisches Löschen nach dem Scanvorgang zu deaktivieren |
| -scantype 3 | Um eine angegebene Datei zu scannen |
| -file „Objektpfad“ | Um zu scannenden Pfad anzugeben |
| „ > Textdatei“ | Um ein Logfile zu erstellen |

Da Security Essentials das Erstellen von Logfiles nicht unterstützt muss dieses über „>“ erzeugt werden und schreibt somit ausschließlich die Anzahl der gefundenen Objekte in die Datei.

Ein weiterer kostenloser Virenscanner, den wir installierten, war avast! Free AntiVirus. Doch nach der Installation stellte sich heraus, dass hier das Scannen über die Kommandozeile nicht unterstützt wird. Deshalb mussten wir auf die kostenpflichtige Pro-Version umsteigen, die diese Funktion unterstützt. Die Überprüfung wird hier über die ashcmd.exe mit dem Parameter /p=4, der den Scan ohne weitere Benutzereingabe durchführt, sowie Funde nicht löscht bzw. in Quarantäne verschiebt und der Angabe eines zu scannenden Pfads gestartet. Das Logfile wird mittels /\_ > Textdatei erstellt.

Mit Kaspersky AntiVirus kam ein weiterer kostenpflichtiger, häufig verwendeter Virenscanner hinzu. Hier wird die Untersuchung auf Viren über avp.com und den Parametern SCAN [Objektpfad] und /i0, um das Objekt bei einem Fund unverändert zu lassen, durchgeführt. Logfiles können hier mittels /R:[Textdatei] erstellt werden.

Beim ebenfalls kostenpflichtigen Produkt McAfee, wurde sogar eine spezielle Kommandozeilenversion angeboten, was für unsere virtuelle Maschine sehr ressourcenschonend ist. Die Virensignaturen mussten jedoch separat heruntergeladen und eingebunden werden. Ausgeführt wird der Scan über die scan.exe und der Angabe eines zu scannenden Pfads. Das Logfile wird mit dem Parameter /REPORT=[Textdatei] erzeugt.

Das letzte von uns verwendete Produkt ist G Data AntiVirus. Dieses Produkt verwendet als einziges zwei verschiedene Engines zur Untersuchung von Objekten. Der Scanvorgang wird über die avkcmd.exe und dem Parameter /scan: [Objektpfad] gestartet. Die Erzeugung des Logfiles erfolgt mittels /log: [Textdatei].

Der Scanvorgang der einzelnen Virenscanner wird nun mit folgendem Python-Skript scan.py aufgerufen:

import sys, string, os

path = raw\_input("Bitte Pfad der zu pruefenden Datei angeben: ")

print "\nClamWin AntiVirus\n"

os.system("C:/Programme/ClamWin/bin/clamscan.exe --database=" + "C:/Programme/ClamWin/bin/db " + path + " --log=ClamWin.txt")

print "\nAVG AntiVirus\n"

os.system("C:/Programme/AVG/AVG2014/avgscanx.exe /SCAN=" + path + " /REPORT=AVG.txt")

print "\nAvira AntiVirus\n"

os.system('"E:/Programme/Avira/AntiVir Desktop/scancl.exe"' + " " + path + " --log=Avira.txt")

print "\nMicrosoft Security Essentials\n"

os.system('"C:/Programme/Microsoft Security Client/MpCmdRun.exe"' + " -scan -DisableRemediation -scantype 3 -file " + path + " > Microsoft.txt")

print "\navast! AntiVirus\n"

os.system('"E:/Programme/AVAST Software/Avast/ashcmd.exe"' + " /p=4 " + path + " /\_ > avast.txt")

print "\nKaspersky AntiVirus\n"

os.system('"C:/Programme/Kaspersky Lab/Kaspersky Anti-Virus 14.0.0/avp.com"' + " SCAN " + path + " /i0 /R:Kaspersky.txt")

print "\nMcAfee AntiVirus\n"

os.system('"C:/Programme/McAfee\_VirusScan/scan.exe"' + " " + path + " /REPORT=McAfee.txt")

print "\nGData AntiVirus\n"

os.system('"C:\\Programme\\G Data\\InternetSecurity\\AVK\\avkcmd.exe"' + " /scan" + ": " + path + " /log: GData.txt")

Das Skript wird über die Kommandozeile mit dem Befehl python scan.py aufgerufen.

Dann erscheint die folgende Eingabeaufforderung:

"Bitte Pfad der zu pruefenden Datei angeben: "

Hier muss eine zu scannende Datei / ein zu scannendes Verzeichnis angegeben werden.

Nach der Bestätigung mit der Eingabetaste beginnen die verschiedenen Virenscanner mit der Untersuchung und erzeugen anschließend ihr Logfile.