Alternative Whitelist

Wie gut schützen Positivlisten und wie viel Aufwand bedeuten sie?

Application-Whitelisting verspricht ein höheres Schutzniveau als Virenscanner, denn nur explizit erlaubte Software wird ja ausgeführt – Zeroday-Exploit adé! Viele scheuen allerdings den Aufwand, eine solche Liste zu pflegen – doch wie hoch ist der wirklich?

Von Simon Albersmeier und Torsten Valentin, Werl

Eigentlich ist alles ganz einfach: Eine Application-Whitelisting-Lösung muss lediglich wissen, welche Software im Unternehmensnetzwerk ausgeführt werden soll, alles andere ist unerwünscht und wird blockiert. Spiele, „Schattensoftware", die am Arbeitsplatz nicht vorgesehen ist, oder Malware: Alle werden an der Ausführung gehindert, ohne dass diese vorher in irgendeiner Form bekannt sein müssten – einfach aus dem Grund, weil sie eben nicht als bekannt und erwünscht definiert waren.

Doch aus diesem Ansatz ergeben sich ganz automatisch auch Fragen: Bedeutet Application-Whitelisting nicht einen erhöhten Aufwand? Wie erstellt man eine solche Whitelist mit all ihren – vielleicht tausenden – Einträgen? Wie kann man Änderungen wie Updates und neue Software erfassen? Was ist mit dynamischen Anpassungen im Betrieb? Wie lange dauert es, bis eine Änderung in der Policy aktiv wird? Nachrangig ergeben sich je nach eingesetztem Produkt auch Detailfragen in Bezug auf das Verhalten auf mobilen Systemen, zu Reporting, Ausnahmen und individuellen Policies. Welches Schutzniveau wird tatsächlich erreicht? Wie ist die Performance?

Was ist Weiß?

Zur Definition des Erlaubten und Erwünschten gibt es verschiedene Kritereien, die je nach Anbieter und Lösung unterschiedlich genutzt werden:

_____ Dateipfad, Dateiname, Dateigröße: Diese allgemeinen Attribute sind naturgemäß sehr schwach, da es leicht möglich ist, bösartige Dateien in das gleiche Verzeichnis zu legen, den gleichen Namen zu geben oder die gleiche Größe zu erzeugen. ## Für eine Securitylösung eignen sie sich also nicht.

_____ Dateibesitzer: Dieses Attribut ermöglicht eine relativ einfache Umsetzung einer Application-Whitelist, indem sichergestellt wird, dass nur Dateien, die im Dateisystem einem bestimmten Benutzer (z. B. dem Administrator) gehören, ausgeführt werden können.

_____ Kryptografischer Hash: Ein kryptografischer Hash bietet einen zuverlässigen, einzigartigen Wert für eine Software. Hashes sind exakt, funktionieren unabhängig vom Speicherort und Namen einer Datei. Allerdings ändert sich der Hashwert mit jeder noch so minimalen Modifikation einer Datei (z. B. bei Patches/Updates), sodass der Hash der aktualisierten Version ebenfalls erst freigegeben werden muss.

_____ Digitale Signatur oder Herausgeber: Die digitale Signatur einer Software liefert im Prinzip ein zuverlässiges Prüfkriterium um den Hersteller zu identifizieren und sicherzustellen, dass die Datei legitim ist und nicht geändert wurde. Leider jedoch ist auch heute die meiste Software noch *nicht* von ihren Herausgebern signiert – nicht einmal alle zu Windows gehörenden internen Programme tragen eine digitale Signatur. Somit ist es nur mit erheblichem Aufwand möglich, dieses Attribut zu verwenden, indem man alle Programme ohne Signatur nachträglich damit ausstattet, also selbst signiert. Einige Application-Whitelisting-Lösungen können sich auf die Überprüfung der Herausgeber-Identität beschränken, anstatt einzelne digitale Signaturen zu betrachten: Dies basiert auf der Annahme, dass alle Anwendungen von vertrauenswürdigen Herausgebern auch selbst vertrauenswürdig sein sollten. Diese Annahme kann allerdings unzureichend sein, wenn ein Softwarehersteller etwa mehrere Anwendungen liefert, eine Organisation aber beschränken möchte, welche dieser Anwendungen ausgeführt werden können. ## Und nicht zuletzt sind in jüngster Vergangenheit auch Fälle von missbräuchlich signierter Software bekannt geworden.

Auch bei der technischen Umsetzung beschreiten Anbieter von Application-Whitelisting-Lösungen durchaus sehr unterschiedliche Wege: Hersteller wie Ivanti und Microsoft (AppLocker) setzen auf windowsinterne APIs, um auf Dateisystem-Ebene eines oder mehrere der oben genannten Attribute zu prüfen – SecuLution verwendet ein eigenes Ring-Zero-Kernel-Modul, das die Zuweisung von RAM-Speicher für nicht erlaubten Code unterbindet.

Hersteller wie EgoSecure oder Lumension dokumentieren das von ihnen verwendete Verfahren nicht weiter.

Usability und Wartbarkeit

Die Frage des Aufwands für Implementation und Betrieb ist in der Regel entscheidend für den Erfolg einer Application-Whitelisting-Implementation. Aufgrund der unterschiedlichen genutzten Konzepte stellen die Lösungsanbieter auch hier sehr verschiedene Ansätze zur Verfügung – hier ergeben sich womöglich die größten Unterschiede zwischen den verfügbaren Produkten.

Lösungen mit dezentraler Policy: Lösungen, die auf Basis von Dateisystem-Attributen arbeiten, benötigen keine zentrale Datenbank, da das Dateisystem ja selbst schon die Policies abbildet. Eine der empfohlenen Vorgehensweisen für eine derartige Umsetzung besteht darin, Musterrechner (sog. Golden Images) zu erstellen und diese für Produktivsysteme zu klonen. Als Ergebnis hat der Administrator ein maximal homogenes Netzwerk, was sich auch positiv auf den Wartungsaufwand im Betrieb auswirken kann. Dieser Ansatz bringt jedoch auch den Nachteil mit sich, dass kurzfristige, dynamische Änderungen von Policies im laufenden Betrieb sehr aufwändig bis unmöglich sind und für jede individuelle Softwareausstattung ein eigenes Golden Image gepflegt werden muss. Darüber hinaus muss der Administrator die gesamte IT-Infrastruktur und den Softwarestand auf das Application-Whitelisting anpassen – Updates erfordern dann immer auch ein Update des jeweiligen Referenz-Images.

– Lösungen mit zentraler Policy-Datenbank: Ein andere Lösung ist es, die Whitelist zentral im Netzwerk bereitzustellen. Agents auf den Endpoints errechnen dann etwa beim Start(versuch) einer Software den Hash der Anwendung und fragen ihre Berechtigung in Echtzeit bei der zentralen Instanz an. Durch Lernmodi und Regeln können Hashes für die zentrale Datenbank eventuell auch automatisch erfasst, klassifiziert und verwaltet werden, ohne dass der Administrator seine Infrastruktur selbst anpassen müsste. Einige Hersteller versprechen, so auch große heterogene Netzwerk zügig abzubilden und ohne Mehraufwand in der Administration eine Referenzdatenbank zu erzeugen – allerdings sind solche Lösungen nur in entsprechend großen Netzwerken kosteneffizient einsetzbar.

Kosten

Alle Lösungen von Drittanbietern kosten deutlich mehr als Virenscanner – nur Microsofts AppLocker ist bereits Bestandteil moderner Windows-Betriebssysteme, wodurch bei diesem Produkt keine zusätzlichen Lizenzkosten anfallen. Kosten durch entstehenden zusätzlichen Wartungsaufwand sind allerdings erfahrungsgemäß umgekehrt proportional zu den Investitionskosten anderer kommerzieller Lösungen.

Der Leistungsumfang der Lösungen variiert erheblich: Fast alle Application-Whitelists sind als Modul einer anderen Produktlinie zu erwerben – es gibt allerdings auch Anbieter, die Application-Whitelisting als Hauptprodukt vermarkten und dann wiederum weitere Features (z. B. USB-Whitelists) als Modul anbieten. ## Generell sollte man bei Preisvergleichen immer den „Total Cost of Ownership“ hinterfragen – das Produkt mit dem geringsten Wartungsaufwand ist nicht selten das teuerste in der Anschaffung.

Restrisiko

Auch Whitelisting kann naturgemäß nicht hundertprozentig vor allem schützen. Application-Whitelisting-Produkte, die auf internen Berechtigungen von Windows basieren, sind etwa dem Risiko ausgesetzt, dass Code mit Administrator- oder Systemrechten zur Ausführung kommen und damit ihre Sicherheit unterwandern kann. Prominentes Beispiel ist die kürzlich aufgedeckte Sicherheitslücke „Eternalblue“, die auch von der Ransomware WannaCry genutzt wurde: Sie nutzt einen Fehler im SMB-Dienst von Windows. Ein Angriff über die Ausnutzung dieser Sicherheitslücke hat zur Konsequenz, dass der Angreifer Code mit den gleichen Rechten ausführen kann, die der angegriffene Prozess hatte, in diesem Falle also System-Rechte.

Eine Application-Whitelisting-Lösung, die sich ausschließlich auf das Windows-interne Berechtigungsmodell stützt, kann damit an dieser Stelle keinen Schutz bieten, da der Code des Angreifers mit höchsten Rechten ausgeführt wird – und damit auch seine Payload speichern und starten kann. Folglich verwundert es nicht, dass inzwischen bekannt wurde, dass auch mit AppLocker geschützte Netzwerke von WannaCry erfolgreich angegriffen wurden. Unabhängige Kontrollansätze können in solchen Szenarien mehr Schutz bieten und auch Code blockieren, der unter den Berechtigungen des Administrator- oder Systembenutzers ausgeführt werden soll.

Allen Whitelisting-Produkten gemein ist der Umstand, dass ein Interpreter wie WScript in der Regel zu den erlaubten Anwendungen zählt, ein vom Interpreter ausgeführtes Skript aber damit nicht mehr der Application-Whitelist-Kontrolle unterliegt. In der Praxis bedeutet dies jedoch nicht zwangsläufig eine Einschränkung der Sicherheit, da etwa Angriffe über Office-Makros meist zweistufig ablaufen: Das Makro selbst ist dann nur der „Dropper“, der die eigentliche Schadsoftware auf dem angegriffenen System ablegt und versucht, diese zu starten. Dieser Versuch unterliegt dann jedoch wieder der Kontrolle des Whitelisting und wird demzufolge unter-

bunden, da die eingeschleuste Malwre nicht als erlaubte Anwendung eingestuft ist.

Ein Sonderfall ist die PowerShell: Hiermit hat Microsoft einen Interpreter bereitgestellt, der per Default von jedem Benutzer verwendet werden kann und derart mächtig ist, dass damit sogar komplette Ransomware implementiert werden konnte. Um hier für verstärkten Schutz zu sorgen, muss man die Ausführung der PowerShell auf bestimmte Benutzer beschränken oder diese gänzlich unterbinden – dabei kann Application-Whitelisting allerdings wirksam unterstützen.

Fazit

Verschiedene Application-Whitelisting-Produkte sind derart unterschiedlich, dass sich Erfahrungen mit einzelnen Lösungen keineswegs auf andere übertragen lassen: Verfahren, Implementatierung, Betrieb, Wartung et cetera unterscheiden sich bei den gängigen Produkten so erheblich voneinander, dass letztlich leider nur ein individueller Test im eigenen Umfeld aussagekräftig sein kann – schließlich hat jeder Administrator seine eigenen Prioritäten und jedes Netzwerk stellt andere Anforderungen an die eingesetzte Lösung. Glücklicherweise bieten einige Hersteller Teststellungen, teils auch kostenfrei. ■

Simon Albersmeier arbeitet im Marketing, Torsten Valentin ist Geschäftsführer der SecuLution GmbH.

Literatur

[1] Adam Sedgewick, Murugiah Souppaya, Karen Scarfone, Guide to Application Whitelisting, NIST Special Publication 800-167, Oktober 2015, http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-167.pdf