MASTERARBEIT

HÄRTEN VON WINDOWS BETRIEBSSYSTEMEN MIT BOARDMITTELN

ausgeführt am

Studiengang

Informationstechnologien und Wirtschaftsinformatik

Von: Philipp Kropik

Personenkennzeichen: 1510319023

Graz, am 28. Juni 2022 .....

Unterschrift

EHRENWÖRTLICHE ERKLÄRUNG

Ich erkläre ehrenwörtlich, dass ich die vorliegende Arbeit selbstständig und ohne fremde Hilfe verfasst, andere als die angegebenen Quellen nicht benützt und die benutzten Quellen wörtlich zitiert sowie inhaltlich entnommene Stellen als solche kenntlich gemacht habe.

.....

Unterschrift

DANKSAGUNG

An dieser Stelle möchte ich mich herzlich bei allen bedanken, die mich in den letzten Jahren auf dieser Reise bis hin zum Schreiben dieser Arbeit begleitet haben.

Im Speziellen möchte ich mich bei den Studienkollegen aus der ersten Reihe bedanken, die mir, auch wenn sie vor mir fertig geworden sind, jetzt noch die Kraft gegeben haben einen letzten Endspurt zu setzen und diese Arbeit nach langer Zeit doch zu beenden. Danke Denise, David und Dominik.

Natürlich möchte ich mich auch bei meiner Familie bedanken, die mich in der gesamten Phase des Studiums und allen damit verbundenen Entscheidungen unterstützt haben. Ein herzlicher Dank geht auch an Alexander und seine Familie, die trotz der immer kürzer gewordenen Besuche durch das Studium immer für mich da waren. Ein Großer Dank geht auch an Johanna, Kathi und Andrea, drei meiner langjährigsten Freunde, die so groß die Distanz und so eng auch die Zeit in den vergangenen Jahren war, immer in der Lage waren mich zum Lachen zu bringen.

DANKE!

KURZFASSUNG

Das Thema IT-Sicherheit ist eines der sich am schnellsten wandelnden Gebiete in der Informationstechnologie.Mit Werten von bis zu einer Million neuen Schadprogrammen pro Tag konfrontiert, sieht man auch, wo dieser Wandel seinen Ursprung hat.

In einem so volatilen Umfeld gibt es folglich viele Lösungen, die Sicherheit gegen monetären Einsatz versprechen. Die Kosten einiger High-End Lösungen sind aber leicht in der Lage, das IT-Budget einer Firma zu sprengen. In diesem Bereich setzt die vorliegende Masterarbeit an.

Diese Arbeit beschäftigt sich mit der Frage, ob es möglich ist, ein aktuelles Microsoft Betriebssystem nur mit Hilfe von Bordmitteln und dem LAPS Toolkit so weit abzusichern, dass es aktuellen Bedrohungen standhalten kann. Unabhängig vom Ausgang soll ein Nebenprodukt eine möglichst sichere Konfigurationsempfehlung für das Microsoft Windows Betriebssystem bzw. dessen Schutzkomponenten sein.

Hierzu befasst sich die Arbeit zuerst mit der theoretischen Situation, wie Angriffsvektoren, der Kategorisierung von Gefahren und der Taxonomie von unerwünschten und schädlichen Programmen. Nach der theoretischen Betrachtung folgt ein Blick auf die aktuelle Situation und die Betrachtung von exemplarischen Angriffen, zum einen von Malware zum anderen auf Basis der Befragung von Red-Team Sicherheitsexperten.

Im Anschluss werden die Verteidigungstechniken vorgestellt, deren optimale Einstellungen vorgestellt und argumentiert sowie der Einfluss auf Angriffe besprochen.

Aus der Kombination der Informationen über die aktuelle Sicherheitssituation sowie den gewonnenen Daten aus der Betrachtung der Wirksamkeit der Schutzmaßnahmen, erfolgt anschließend die Bewertung des Schutzgrades, der rein mit Bordmitteln erreicht werden kann.

ABSTRACT

The field of IT- Security is one of the fastest moving areas in the realm of information technology. Looking at numbers like up to one million new malware variants being created per day, or damages in the trillion Euro range this fast pace is understandable.

In such a large volatile field it's only natural to see a lot of players offering security in exchange for money, but the cost for some of the offered high-end solutions aren't feasible for the IT budget of smaller or medium scale firms, this was the inspiration for this thesis.
The subject of this thesis is to explore the possibility to secure a Windows computer system with a current patch level and the free LAPS Toolkit in a way that it could be considered safe in a real-world threat context.
Regardless of the outcome part of this work is a configuration recommendation to increase the security of a Windows computer.

To facilitate this the thesis firstly establishes the current threat situation by exploring threat vectors and categorising risks and describing the taxonomy of malware and other potentially unwanted software. Following this is a look at the momentary real-world situation, a look at two past compromises by large malware strains, and the result of the interview of multiple Red-Team security experts about their attack techniques used in security audits of Windows OS clients and servers.
After defining the threats this thesis looks at the defensive capability and tools that can be used to secure a Windows Client, as well as the discussing the optimal settings and their impact on the described attacks.

Resulting from the combination of the current state of cybersecurity as well as the described attacks and defensive capabilities this work tries to evaluate the effectiveness and judge the level of protection that can be given to a system by just using internally integrated tools.

INHALTSVERZEICHNIS

1 EINLEITUNG

"If you spend more on coffee than on IT security, you will be hacked. What's more, you deserve to be hacked" – Richard Clarke

Wenn man die Nachrichten der letzten Jahre verfolgt, ist das Thema Cybersicherheit und seine Folgen immer stärker auch im Mainstream angekommen. Abgesehen von den positiven Aspekten was das Bewusstsein der User betrifft, kann es aber auch dafür sorgen, dass Sicherheitsverantwortliche in einem permanenten Alarmzustand sind und nur darauf warten, dass etwas passiert. Wenn man die Historie betrachtet, hat sich auf Seite der Sicherheitsbedenken nicht viel verändert, wenn man bedenkt, dass bereit 1989 Eugene Howard Spafford einer der führenden Sicherheitsforscher folgenden Grundsatz zur Sicherheit von IT-Systemen verfasst hat. „Das einzig wirklich sichere System ist ausgeschalten, in einen Betonblock eingegossen und von bewaffneten Wächtern bewacht in einem Blei ausgekleideten Raum gelagert. Und selbst dann hätte ich meine Zweifel"

Hier hat die Geschichte der vergangenen Jahre, durchaus gezeigt, dass er Recht behalten sollte.

Von staatlich geschaffenen Viren, mit denen es gelungen ist, SCADA Systeme zu beschädigen, die sich in Air-Gaped Netzwerken befunden haben, bis hin zu Ransomware-as-a-Service war im letzten Jahrzehnt viel zu hören.

Die Verbreitung des Stuxnet Wurms außerhalb der gewollten Umgebung hat hier auch schön gezeigt, wie schnell sich eine solche Malware verbreiten kann.

Abseits der Doktrin, dass „der Verteidiger dem Angreifer in diesem Umfeld meist mindestens einen Schritt hinterher ist" hat die Entwicklung der vergangenen Zeit auch gezeigt, dass Angreifer immer professioneller werden und sich Angriffe auf Computersysteme zu einem Milliardengeschäft entwickelt haben. Hier gilt es, es dem Angreifer zumindest so schwer wie möglich zu machen, um sich und seine Anwender so gut wie möglich zu schützen.

1.1 Forschungsfrage

In diesem Spannungsfeld geht es nun um die Frage:

Inwieweit lassen sich aktuelle Top Bedrohungen der IT-Sicherheit mit Hilfe von Bordmitteln abwehren oder im Schadensausmaß begrenzen?

Dabei wird grundsätzlich eine der folgenden Antworten erwartet:

H0: Es ist nicht möglich ein System mit Bordmitteln ausreichend abzusichern.

H1: Es ist möglich ein System mit Bordmitteln ausreichend abzusichern.

1.2 Ziele

Ziel ist es zu Beginn die theoretischen Gefahren in modernen, windowsbasierten Firmennetzwerken, nach Kategorien auszuarbeiten und vorzustellen. In Folge sollen diese Risiken den potenziellen Schutzmaßnahmen des Betriebssystems gegenübergestellt werden. Dabei soll zum einen die optimale Konfiguration der betriebssystemeigenen Schutzmechanismen gezeigt werden und weiters soll dies die Beantwortung der Forschungsfrage ermöglichen.

Somit soll am Ende der Gegenüberstellung zum einen die Einschätzung stehen, ob es möglich ist, durch das Härten des Betriebssystems mit systemeigenen Mitteln die Angriffsfläche so weit zu verringern, dass gängige Angriffe abgewehrt werden können, ohne dass von einer Firma, oder einer versierten Privatperson, Geld in zusätzliche Sicherheitslösungen investiert werden muss. Zum anderen soll auch ein Set an Einstellungen mit denen dies, sofern möglich, gelingen sollte ein Ergebnis sein.

1.3 Methoden

Der Aufbau der Arbeit setzt sich aus zwei Teilen zusammen. Im ersten Teil betrachtet die Arbeit auf Basis aktueller Daten von Virenschutzherstellern und öffentlich zugänglichen Daten von Organisationen wie CERT o.Ä. die häufigsten Angriffe und beleuchtet den technischen Hintergrund zu diesen.

Auf die Verwendung einer Befragungstechnik zur Ermittlung der häufigsten Angriffsmethoden oder deren Erfolgsquote wird bewusst verzichtet. Eine Vorabklärung im eigenen Firmenumfeld sowie Gespräche mit IT-Leitern aus anderen Unternehmen hat gezeigt, dass auch wenn im persönlichen Gespräch über erfolgte und erfolgreiche Angriffe gesprochen wird, es unter dieser Gruppe keinen gab, der sich im Zuge einer wissenschaftlichen Arbeit als Quelle zitieren ließe, oder mit dem Firmennamen genannt werden möchte. Da es in diesem Bereich, aber gesicherte Informationen in Form der jährlichen „Gefahrenreports“ von diversen Antiviren Herstellern sowie öffentlichen Stellen gibt, scheinen diese als besser geeignete Quelle für diese Arbeit zu dienen. Bei diesen gibt es kein Interesse daran, Gefahren oder Vorfälle ungenannt zu lassen, um das eigene Gesicht zu wahren bzw. den Firmennamen zu schützen.

Im Bereich der Technik soll zum einen darauf eingegangen werden, welche Angriffsarten es gibt und welche Vektoren für diese Angriffe genutzt werden. Weiters sollen die erforderlichen Rahmenbedingungen, welche vorherrschen müssen, beleuchtet werden, und abschließend, wie der technische Ablauf eines Angriffes aussieht. Hier fand eine Befragung von mehreren Security Professional statt bei der erhoben wurde, welche Punkte erfüllt sein sollten, um eine Härtung des Systems zu erreichen und häufige Angriffe zu mitigeren.

Im zweiten Teil werden die im ersten Teil der Arbeit betrachteten und ausgearbeiteten Angriffe dahingehend analysiert, ob und wenn ja, inwieweit sich diese durch verfügbare Bordmittel des Windows Betriebssystems mitigeren oder abschwächen lasse. Aternativ, wenn diese

Möglichkeiten nicht bestehen, soll erarbeitet werden wie sie es Administratorinnen und Administratoren zumindest erleichtern können, auf einen Sicherheitsvorfall aufmerksam zu werden, um schneller reagieren zu können.

Weiters wurde zusammen mit mehreren Sicherheitsspezialisten eine Liste an empfohlenen Absicherungen von Windows Systemen im Zuge einer Befragung erarbeitet. Die genannten Abwehrmethoden bzw. einfache OS Settings, welche keine gesonderte, genauere Erklärung benötigen, wie beispielsweise Zugriffsrechte, wurden diesen gegenübergestellt, um bewerten zu können, ob eine ausreichende Absicherung ohne weitere kostenpflichtige Hilfsmittel möglich ist.

1.4 Motivation

Bei dem Thema Informationssicherheit handelt es sich im Feld der Informationstechnologie sicher um eines der umfangreichsten. Ähnlich sieht es daher mit der Palette der Empfehlungen und der auf dieses Segment zugeschnittenen Produktpalette und deren Preisen aus. Dennoch existiert dieses Thema in vielen Nicht-Technologieunternehmen, außerhalb des Fokus der Budgets und somit dem Spannungsfeld der Kosten-Nutzen-Rechnung. Die internen oder zugekauften IT-Dienstleister wissen meist, welche Gefahren es gibt und mit welchem, meist mit hohen Lizenzkosten verbundenem Produkt, diese zumindest eingedämmt werden können. In den Bereichen der Geschäftsführung hingegen muss abgewogen werden, welches Risiko abgedeckt werden muss und welchen möglichen Schaden man, mehr oder weniger willentlich, in Kauf nimmt. Das liegt daran, dass bei einem Nichteintreten der Gefahr die Lizenzkosten höher wären, als die potenziellen Kosten im Falle eines Sicherheitsvorfalles.

Trotz dieser Überlegungen ist IT-Security in den letzten Jahren zu einem florierenden Geschäft geworden, und immer mehr Produkte versuchen abseits der, in zumindest den meisten Firmen installierten, Virenscannern und Firewalls den Markt für sich zu behaupten.

In diesem Spannungsfeld zwischen Sicherheit und Kosten, für eben diese, versucht diese Arbeit anzusetzen. Sie soll beleuchten, welche Möglichkeiten es gibt, die häufigsten Angriffsszenarien ohne Kosten für zusätzliche Soft- oder Hardware zu stoppen, deren Auswirkungen so gut es geht zu minimieren oder sofern beides nicht möglich ist, zumindest darauf aufmerksam zu machen, dass etwas geschieht, das in dieser Form nicht passieren sollte.

Während man als Administrator oder als Administratorin, zwar meist nach einem Vorfall entdeckt, dass auf der einen oder anderen Seite eines Sicherheitsforschers einzelne Tipps zu finden sind, wie sich hier ein Schaden hätte einschränken lassen können, gibt es aktuell keine für derzeitige Betriebssysteme geschriebene, allgemein gültige Konfigurationsempfehlung. Hier soll diese Arbeit ansetzen.

Ziel der Arbeit ist es am Ende sowohl einen verständlichen Überblick über die aktuelle Bedrohungslage und die technische Funktion dieser Bedrohungen zu schaffen, sowie Administratoren eine einfache Kompilation an Konfigurationsvorschlägen zu liefern, die dazu geeignet sind, ohne einen erheblichen Mehraufwand die Sicherheit in einem internen IT-System merklich zu erhöhen.

2 ABGRENZUNG

In einem so weitläufig gestreuten Forschungsfeld wie der Cybersecurity und Systemverteidigung ist es natürlich leicht den Fokus zu verlieren. Es ist daher wie in jedem Forschungsgebiet notwendig, Abgrenzungen zu finden, die es ermöglich effektiv zu arbeiten.

Im Umfeld dieser Arbeit sind dies die Einschränkungen im Bereich der betrachteten Betriebssysteme, der Angriffsszenarien, sowie der Umfang der nutzbaren Abwehrmöglichkeiten.

2.1 Betriebssysteme

Als Betriebssystemumgebung wird im Zuge der Arbeit das Windows Betriebssystem Umfeld betrachtet. Im genaueren Windows 10 in den beiden aktuellsten Build Stufen 2020H1 und 2020H2 sowie die derzeit am weitesten verbreiteten Serversysteme „Windows Server 2012 (R2)“ und „Windows Server 2016“. (Statcounter, 2019)

2.2 Angriffsszenarien

Im Zuge dieser Arbeit werden Angriffe auf die Softwareebene eines Computersystems betrachtet, die darauf abzielen Computersysteme zu kompromittieren, stillzulegen oder nachhaltig zu schädigen.

Physikalische Angriffe auf Computersysteme wie beispielsweise das Umgehen von Festplattenverschlüsselung durch kryologische Manipulation von Speichermodulen, oder Sidechannel Angriffe, die auf unterschiedliche CPU Timings je nach Eingabe abzielen, um korrekte Eingaben erkennen zu können, sind nicht Teil dieser Arbeit.

Angriffe über potenziell unsichere Applikationsprogramme, die innerhalb des Betriebssystems ausgeführt werden, werden im Zuge der Angriffsvektoren betrachtet, jedoch nur in einer allgemeinen Übersicht und nicht auf ein spezielles Programm fokussiert.

Angriffe auf besondere Teilsysteme, die nicht generell auf Systemen verfügbar sind, wie beispielsweise Angriffe über den integrierten Webserver (Internet Information Service IIS) von MS Betriebssystemen befinden sich außerhalb des Scopes für diese Arbeit, da die Absicherung dieses Bereichs zu stark in den Bereich des „Open Web Application Security Project“ (OWASP) fällt.

Supply Chain Angriffe sind Angriffe, die darauf abzielen, Endbenutzersysteme zu kompromittieren, indem vorgelagerte Softwarelieferanten angegriffen werden, um beispielsweise Updatekomponenten zu übernehmen, um Schadsoftware über digital signierte Pakete und den Update Prozess der Anwendersoftware direkt zu installieren (Solaflare Angriff 2021), oder auch nur die Schadsoftware mittels gültiger digitaler Signatur, schwerer erkennbar zu machen (KSL

2017). Diese Angriffe werden zwar von einer im Bereich der Gegenmaßnahmen erwähnten Technologie erfasst, bzw. erschwert, stellen aber nicht den zentralen Fokus der Arbeit dar, da allein die Betrachtung des Ablaufs eines solchen Angriffs, den Umfang einer eigenen Arbeit aufweisen würde. Einfachere Supply Chain Angriffe, wie das Übernehmen einzelner Postfächer bei einem Lieferanten, werden in Rahmen der Arbeit gestreift, da es sich dabei um einen beliebten Initialvektor handelt.

2.3 Zusätzliche Tools

Grundsätzlich befasst sich die Arbeit nur mit Bordmitteln der Windows Betriebssysteme. Eine Ausnahme stellt in diesem Fall das Tool LAPS dar:

Dieses Tool ist zwar nicht in der Grundinstallation des Betriebssystems enthalten, wird aber im Zuge der Arbeit dennoch im Bereich der Abwehrmaßnahmen betrachtet.

Grundlage dafür ist, dass „LAPS – Local Admin Password Solution“, mittlerweile generell als „Best Practice“ gesehen wird und einen massiven Einfluss auf die Resilienz eines Netzwerks hat, sollte ein Client durch Schadsoftware kompromittiert werden.

3 ANGRIFFSARTEN

Im folgenden Kapitel soll betrachtet werden, welche Arten von Schadsoftware es gibt, auf welche Art und Weise der Zugang zu den Systemen erfolgt, und welche technischen Infektionsvektoren im Kontext dieser Arbeit betrachtet werden. Ziel ist es den Leserinen und Lesern einen groben Überblick über die Thematik zu geben, um die in späteren Kapiteln genannten Informationen leichter in einen Kontext setzen zu können.

3.1 Malware

Der Begriff Malware oder Schadprogramme stellt einen Sammelbegriff für, je nach Sammlungsschema, bis zu 30 verschiedene Gruppen von Programmen dar.
Die wichtigsten Kategorien sollen hier kurz vorgestellt werden. Als Basis dieser Aufzählung dienen die Einschätzungen und Klassifizierungen von Cisco Talon, Barracuda und Lastline.
Bei den im Folgenden aufgeführten Klassifizierungen handelt es sich um grobe Kategorien. Die meisten aktuell aktiven Schadprogramme sind eine Kombination aus mehreren der hier genannten Klassen und der darin beschriebenen Verhaltensarten. So verbreitet sich zum Beispiel die Ransomware Funktion des Schadprogramms WannaCry/WannaCrypt mit Hilfe der Ausnutzung von Schwachstellen auf Betriebssystemebene, hat somit auch Verhaltensweisen wie ein Wurm. Zusätzlich wurde das DoublePulsar Backdoor im Zuge der Infektion nachgeladen. Die Verhaltensweisen wären hier also Ransomware / Wurm und Trojaner. (EMSI, 2012) (Kaspersky Labs)

3.1.1 Virus

Viren zeichnen sich dadurch aus, dass sie sich Dateien anhängen, die bis zur Infektion sauber sind und sich auf diesem Weg verbreiten. Wird eine mit dem Virus infizierte Datei ausgeführt, kann der Lauf des Programms beginnen. Die Auswirkungen eines Virus können relativ harmlos verlaufen, wie dies beim „Cookie Monster“ Virus der Fall war. Dieser gab nach dreißig Ausführungen einer infizierten Datei den Text „YOU KNOW WHAT? I WANT A COOKIE!“ aus und fror bis zur Eingabe des Wortes „COOKIE“ den Screen auf der Meldung ein. Im Gegensatz dazu waren Viren wie der Win9x.CIH/“Chernobyl“ in der Lage auf einigen Mainboards das Bios zu überschreiben und somit die betroffenen Geräte bis zum Tausch des Bios Chips unbrauchbar zu machen. (EMSI, 2012; Kaspersky Labs)

3.1.2 Würmer

Anders als Viren warten Würmer nicht in einem passiven Zustand darauf, dass sie durch Ausführen einer infizierten Datei auf ein Zielsystem gelangen, sondern versuchen sich durch aktive Propagationstechniken auf neue verwundbare Systeme zu verbreiten.
Die drei bekanntesten Vertreter dieser Kategorie sind in der Öffentlichkeit wahrscheinlich der

erste großflächig erfolgreiche Wurm „I Love You / Loveletter“ der sich in den Frühen 2000ern per Mail verteilte. Ein weiterer ist der W32. Conficker/Downadup der sich im Jahr 2008 verbreitete und mit welchem auch den Autor dieser Arbeit, einige Stunden Nachtarbeit verbinden. Der einzige Schädling in dieser Aufzählung der Hardwareschäden hinterlassen konnte: W32.Stuxnet. Dieser wurde abgesehen von der Infektion von mehreren Millionen an PC-Systemen dazu genutzt gezielt Siemens Industriesteueranlagen zu infizieren und in diesen dann dafür zu sorgen, dass angeschlossene Frequenzumrichter und die damit verbundenen Uran Zentrifugen durch permanenten Geschwindigkeitswechsel unbrauchbar gemacht wurden. (EMSI, 2012) (Kaspersky Labs)

3.1.3 Trojaner

Trojaner leiten ihren Namen vom Trojanischen Pferd der Legenden ab und stellen also eine Klasse an Schadprogrammen dar, die sich meist zusammen mit einem nützlichen Programm verbreiten und dieses als Infektionsquelle nutzen. Einmal installiert kann ein Trojaner genutzt werden, um andere Kategorien an Schadsoftware nachzuladen. (EMSI, 2012) (Kaspersky Labs)

3.1.4 Spyware

Spyware hat zum Ziel das Benutzerverhalten und Daten der Benutzer aufzuzeichnen und diese an einen Command and Control Server zu übermitteln, wo sie dem bösartigen Akteur zur Verfügung stehen. Der Aktionsraum dieser Programme reicht von Zugangsdaten, die via Keylogger gesammelt werden und so zur Kompromittierung von Accounts sorgen, bis hin zur Erpressung mit in Dateien abgelegten Informationen. Im Firmen- und Industriekontext können mit dieser Kategorie an Malware auch Betriebsgeheimnisse gestohlen werden. (Kaspersky Labs) (EMSI, 2012)

3.1.5 Ransomware

Ransomware, wie der Name übersetzt schon andeutet, hat einzig das Ziel die Benutzer zu erpressen und so die Betroffenen zu einer Zahlung zu nötigen. Diese Schadsoftware Variante hat vor allem zu Zeiten des Kryptowährungs-Booms 2016 und 2017 eine massive Häufung gesehen. Die Funktion dieser Schadsoftware kann zum einen das Blockieren des Zugangs zum System sein, meist unter dem Einblenden einer bedrohlich wirkenden „Blockseite“, die unter dem Logo einer Strafverfolgungsbehörde zur Zahlung einer Strafe auffordert. Zum anderen die moderneren Varianten, die den Zugang zu den auf dem Computer oder Server abgelegten Dateien durch das Verschlüsseln eben dieser verhindert. Je nach Variante der Infektion, hilft ab diesem Zeitpunkt nur noch das Wiederherstellen der Daten aus einem Backup, oder sofern kein von AV-Herstellern bereitgestelltes Entschlüsselungstool existiert, das Zahlen der Forderung. In einigen Fällen der letzten Jahre wurde jedoch die Kryptographie durch die Schadsoftwareautoren so fehlerhaft umgesetzt, dass eine Entschlüsselung der betroffenen Daten auch nach der Zahlung des Lösegeldes nicht möglich war. (EMSI, 2012)

3.1.6 Adware

Adware dient nach der Installation, die meist im Zuge der Installation eines Freeware Programms erfolgt dazu, der Anwenderin oder, dem Anwender Werbung zu präsentieren. Dies kann über Pop-Ups, manipulierte Webseiten oder das Umleiten auf bestimmte Webseiten passieren. Zusätzlich zu diesem für die Userin oder den User erkennbaren Verhalten, kann die Schadsoftware ebenfalls im Hintergrund das Userverhalten analysieren und die Daten an einen Server weiterleiten. Anders als bei Spyware ist hier jedoch nicht das Ausspähen von Zugangsdaten das Ziel, sondern das Anlegen eines möglichst genauen Werbeprofils. (EMSI, 2012; Kaspersky Labs)

3.1.7 Backdoors / Botnets

Backdoors werden meist im Zuge einer Infektion mit einem Trojaner nachträglich in das System eingebracht und dienen dem bösartigen Akteur dazu, Zugang zu einem System oder Netzwerk zu erlangen. Sie stellen in Firmennetzwerken oft den Brückenkopf für die Übernahme des Netzwerkes oder die Exfiltration von Betriebsgeheimnissen dar. Je nach Funktion des Backdoors kann es auch sein, dass das befallene System Teil eines Botnetzes wird. In diesem Fall kann das System unter anderem als Proxyserver zur Verschleierung von Spuren, als Teilnehmer in einem DDOS Angriff, oder als nicht rückverfolgbarer Ablageort für illegale Daten genutzt werden. Das Ergebnis einer Infektion ist sehr plakativ gesehen immer dasselbe, man ist nicht mehr Herr über seinen eigenen Rechner. (EMSI, 2012)

3.2 Kompromittierung

“We discovered in our research that insider threats are not viewed as seriously as external threats, like a cyberattack. But when companies had an insider threat, in general, they were much more costly than external incidents. This was largely because the insider that is smart has the skills to hide the crime, for months, for years, sometimes forever.” — Dr. Larry Ponemon

Während man im Groben auch bei einer Infektion mit Malware von einer Kompromittierung des Systems sprechen kann, wollen wir hier eher den Fall des Eindringens in ein System oder Netzwerk mit traditionellen Mitteln betrachten. Die Gefahren einer Systemkompromittierung sind je nach betroffenem System weit gestreut, und können vom Abfangen und Manipulieren von Daten bis hin zu Industriespionage reichen. Das kann mitunter weitreichende Folgen für ein Unternehmen haben, wenn diese Daten, unabhängig davon ob echt oder manipuliert, zum falschen Zeitpunkt verteilt werden. Ein solches Ereignis kann, wie man 2016 bei der US-Wahl gesehen hat, die Demokratie einer Supermacht ins Wanken bringen.
In Zeiten des dezentralisierten Cloud Computings kann das Kompromittieren eines einzelnen

Zugangs zu einem Unternehmensaccount bei nicht Einhaltung der Bestpractices, wie beispielsweise Multifaktor Authentifikation, leicht weitreichende Folgen haben.

Unter diesen Überbegriff fallen, für den gewählten Rahmen dieser Arbeit, vor allem die im Folgenden definierten Fälle:

3.2.1 Ausnutzen von Softwareschwachstellen zur Übernahme von Systemen

Ein Angreifer schafft es mit Hilfe einer ihm bekannten Schwachstelle sich Zugang zu einem System zu verschaffen, indem er beispielsweise ein manipuliertes Datenpaket an den Server sendet und den Server dazu bringt, beliebigen Code in dieser Übertragung auszuführen. Auf diesem Weg ist es beispielsweise möglich eine Remoteshell zu installieren oder sich einen privilegierten Useraccount anzulegen. Dieser kann dann genutzt werden, um das System nach brauchbaren Informationen zu durchforsten, oder den übernommenen Rechner als Brückenkopf für einen späteren Angriff zu nutzen.

3.2.2 Ausnutzen von organisatorischen Schwächen zur Übernahme von Systemen

Hier gelingt es einem Angreifer über Schwächen in der Organisation, Zugriff zu Daten und Informationen zu erhalten auf die er diesen nicht haben sollte. Ein klassisches Beispiel wären hier schlecht berechtigte Fileserver oder auch nicht korrekt eingeschränkte lokale Rechte.

3.2.3 Laterale Verbreitung

Nach der Übernahme eines erstens Knotens welcher als Brückenkopf genutzt werden kann, beginnt die Verbreitung des Angriffs im weiteren Netzwerk. Besonders erleichtert wird dieser Schritt beispielsweise durch ein auf mehreren Systemen identes Administratorenkennwort. Dies ermöglicht es einem internen so wie externen „Angreifer“ mit Zugriff auf das Administratorenkonto eines Systems andere Systeme zu übernehmen.

3.3 Denial-of-Service ((D)DOS)

Bei Denial-of-Service Attacken handelt es sich, wie der Name schon suggeriert, um Angriffe, die darauf abzielen, die Systeme des Angegriffenen funktionsunfähig zu machen und damit deren Kundinnen und Kunden bzw. Nuterinnen und Nutzern und Nutzern den Zugriff auf die dahinterliegenden Services zu verweigern. Auch hier gibt es unterschiedliche Kategorien.

3.3.1 Denial of Service Schwachstellen

Diese Angriffe können zum einen auf Schwachstellen in einem System basieren und somit meist mit relativ wenig Aufwand für den Angreifer durchgeführt werden. Beispiele hierfür sind der „Ping-

of-Death“, mit dem es gelang den Netzwerkstack verwundbarer Betriebssysteme mit einem einzigen manipulierten ICMP Paket zum Absturz zu bringen.
Etwas neuere Vertreter währen hier einige der im letzten Jahr bekannt gewordenen Schwachstellen in der weit verbreiteten Bibliothek OpenSSL, welche auch gezeigt haben, dass das Patchen solcher verwundbaren Programmstellen ein Problem darstellen kann, wenn es sich um eine einbettete Systembibliothek handelt.

3.3.2 Distributed Denial of Service Angriffe

Eine andere Angriffsmethode, die vor allem im Umfeld von „Cyberaktivismus“ und Erpressung weit verbreitet ist, sind die so genannten Distributed Denial of Service Angriffe. Hier werden eine Vielzahl an Quellsystemen genutzt, um ein System oder ein Service unter der Last von unzähligen Angriffen zum Absturz zu bringen. Beispielsweise, um das Ziel des Angriffs zum Schweigen zu bringen. Ein Beispiel hierfür wäre unter anderem der im September 2016 erfolgte Angriff auf den Sicherheitsforscher Brian Krebs. Hierbei wurde ein Botnetz aus übernommenen IoT Geräten genutzt, um die Seite mit einer Datenlast von 665 Gigabit pro Sekunde zum Zusammenbruch zu bringen.

Zusätzlich zu Botnetzen und gezielten Angriffen durch eine Vielzahl an bewusst teilnehmenden offensiv genutzten IT Systemen, können auch in dieser Kategorie Schwachstellen ausgenutzt werden, um sogenannte „Reflektions–" oder „Verstärkungsattacken“ durchzuführen. Bei diesen werden Fehler in legitimen Systemen genutzt, um mit wenig eingehenden Daten eine große Menge an, auf das Ziel gerichtete, ausgehende Daten zu erzeugen. (Cloudflare, 2019; Krebs, 2016)

4 ANGRIFFSVEKTOREN

“Ransomware is more about manipulating vulnerabilities in human psychology than the adversary's technological sophistication”

— James Scott, Sr. Fellow, Institute for Critical Infrastructure Technology

Nachdem wir im letzten Kapitel die potenziellen Bedrohungen betrachtet haben, soll in diesem Kapitel ein Blick darauf geworfen werden, wie diese Bedrohungen ihren Weg in ein Netzwerk finden und sich dort verbreiten können. Grundsätzlich ist immer davon auszugehen, dass kein System perfekt ist und es für jedes System einen passenden Angriffsvektor gibt. Dies beweist unter anderem der erfolgreiche Angriff auf das „Air gapped“, also das nicht aus dem globalen Netz erreichbare System der Uranzentrifugen des Iranischen Atomprogramms. Anders als in diesem Fall erfordert ein Angriff auf ein normales Unternehmen in den meisten Fällen nur eine unwissentliche Mittäterin oder einen unwissentlichen Mittäter, welcher entweder zur falschen Zeit am falschen digitalen Ort ist, oder einfach nur von der Neugierde überrascht wird. Die moderne IT-Landschaft trägt dieser Erkenntnis, dass alle Systeme angreifbar sind, mit den beiden Grundsätzen „Zero Trust Computing“ und „Assume Compromise“ Rechnung, welche langsam die früher geltende Doktrin des sicheren internen Netzes und des „bösen“ externen Netzes ablösen.

4.1 Human Factor /Failure

Wie schon durch das Eingangszitat versucht wird zu vermitteln, ist in vielen Fällen in denen Systeme erfolgreich angegriffen werden, die wirkliche Schwachstelle der Mensch. Wenn es Angreifern gelingt, zwar einen potenziellen Virenscanner eines Mailsystems zu überlisten und die Dateien zur initialen Infektion eines Systems an einen gewünschten User zu senden, muss der Inhalt der Mail zumindest so gewählt sein, dass der User, der die Mail motiviert wird, den Anhang auch zu öffnen und keinen Verdacht zu schöpfen. Alternativ kann es bei einem ungeschulten und unvorsichtigen User auch ausreichen, die Datei so zu bezeichnen, dass er sich verpflichtet fühlt diese zu öffnen. Mahnungen eignen sich für das Auslösen dieses Verhaltens in den meisten Fällen besonders gut. Aber auch das generelle Wecken von Neugier oder Gier im Allgemeinen kann sich für einen Angreifer schnell bezahlt machen.

Andere Beispiele für den Human Faktor abseits der Infektion von Systemen sind auch Social Engineering Angriffe die die Angegriffenen entweder dazu bringen Geheimnisse auszuplaudern, vom eigenen Passwort bis hin zum Staatsgeheimnis oder auch der CEO-Fraud, bei dem es den Angreifern gelingt, die Betroffenen unter dem Vorwand der Geheimhaltung dazu zu bringen hohe Summen ins Ausland zu überweisen, ohne die internen Prozesse und Monitoring Systeme anzustoßen. (Hadlington, 2018; Kaspersky Labs, 2018)

4.2 DLL Sideloading

DLL Sideloading macht sich die Möglichkeit zu Nutze, dass in Windows Systemen Applikationen die Möglichkeit haben, Dynamic Link Librarys (DLLs) zur Laufzeit dynamisch nachzuladen. Die ersten beobachteten Angriffe mit dieser Technik fanden bereits in den frühen 200ern statt und nutzten die Lücke CVE-2000-0854, welche sich das Verhalten von Word 2000 zu Nutze machte, das DLLs zuerst im Verzeichnis des zu öffnenden Dokuments gesucht wurden. (MITRE, 2001; Steward, 2014)

4.2.1 Was sind DLLs

„Eine DLL ist eine Bibliothek, die Code und Daten enthält. Sie kann von mehreren Programmen gleichzeitig verwendet werden. In Windows-Betriebssystemen führt die DLL „Comdlg32“ beispielsweise häufig verwendete dialogfeldbezogene Funktionen aus. Daher kann jedes Programm die in dieser DLL enthaltene Funktionalität zur Implementierung eines Öffnen-Dialogfelds verwenden. Dies trägt zur Wiederverwendung des Codes sowie zu einer effizienten Speichernutzung bei. Mithilfe von DLLs lässt sich ein Programm in separate Komponenten (so genannte Module) aufgliedern. […] Außerdem können Aktualisierungen einfacher auf die einzelnen Module angewendet werden, ohne dass sich dies auf andere Teile des Programms auswirkt. Beispiel: Sie besitzen ein Lohn- und Gehaltsprogramm, und die Steuersätze ändern sich jährlich. Wenn diese Änderungen in einer DLL isoliert werden, können Sie ein Update einbringen, ohne das Programm vollständig neu erstellen oder installieren zu müssen.“ (Microsoft, 2019a)

4.2.2 Wie funktioniert der Angriff

Wie in vielen anderen Betriebssystemen auch üblich, erlaubt es Windows DLL Dateien nicht nur beim Start einer Applikation, sondern auch zur Laufzeit des Programms neue DLLs nachzuladen. Die Pfadangaben zu diesen DLL Dateien können mit dem absoluten Pfad angegeben sein, als relativer Pfad, als Manifest Datei oder einfach nur per Namen.
Der Angriff zielt darauf ab, eine bösartig modifizierte Version einer DLL Datei im Filesystem des Opfers zu positionieren und damit den Angriff im Kontext der Benutzer durch deren Verwendung einer Software durchzuführen. Am leichtesten gelingt dies, wenn nur der Name einer DLL ohne Pfadangabe gewählt wird, dann können sich die Angreifer das Suchverhalten an definierten Pfad Variablen zu Nutze machen, um die manipulierte Version so zu positionieren, dass sie vor der legitimen gefunden wird. (Steward, 2014)

4.3 Software Schwachstellen

Fehlerfreie Software stellt einen Mythos dar, der sich kaum erreichen lässt. In der Fachliteratur wird die Zahl der Fehler, die durchschnittlich in einem Programmierprojekt mit in etwa 15-50 Fehlern pro tausend Codezeilen angegeben. Oder 0,5 „Defects“ pro den besagten tausend Zeilen an Code. (McConnell, 2016)

Wenn man von diesen Werten ausgehend nun beispielsweise das Firefox Projekt betrachtet, das zum aktuellen Zeitpunkt nach eigenen Angaben aus 20.548.088 Zeilen reinem Code ohne Leerzeilen und Kommentaren besteht, würde das bedeuten, dass sich im Code zwischen 308 221 und 1 027 404 Fehler beziehungsweise 10 274 Defects verstecken. (BlackDuck | OpenHub, 2019)

Während die meisten dieser potenziellen Schwachstellen wahrscheinlich keine Auswirkung auf die Systemsicherheit haben, gibt es immer wieder Schwachstellen die erfolgreich ausgenutzt werden können, um ein System zu übernehmen. Vor allem Schwachstellen in Browsern, Office Anwendungen und dem Betriebssystem selbst, eignen sich zur Systemübernahme besonders gut. Dass es vor allem im Bereich der Browser ein großes Potential dafür gibt, zeigt beispielsweise der seit 2007 auf der Kanadischen Sicherheitskonferenz „CanSecWest“ abgehaltene Hacking Wettbewerb „Pwn2Own“, bei dem Angreiferinnen und Angreifer das übernommene Gerät behalten dürfen.

4.4 Code Injektion / Remote Code Execution

Code Injektion, also das Einschleusen und Ausführen von Code, welcher nicht Bestandteil des gewollten Ablaufs eines Programms ist, kann grundsätzlich jedes Programm treffen, welches Eingaben annimmt und verarbeitet. Die bekanntesten Fälle sind hier mit Sicherheit Webapplikationen, wo Codeinjektionen regelmäßig unter den Top 10 der Angriffsarten sind. Die Folgen können von Datenpreisgabe, über Datenverlust, bis hin zur Kompromittierung des Systems führen, abhängig vom Ziel des Angriffs und der Konfiguration des angegriffenen Systems.
Ziel ist es den eigenen Code im Kontext des verwundbaren Programms laufen zu lassen und auf diese Weise das System zu einem nicht geplanten Verhalten zu bringen.

Wie Eingangs bereits erwähnt, kann es auf diese Weise möglich sein, sich Zugang zu Informationen zu verschaffen, oder diese für den Besitzer unbrauchbar machen. Gelingt es, ein Programm auf diese Weise zu kompromittieren das Zugang zu Systeminformationen hat, kann ein kompletter Verlust der Kontrolle über das System und alle mit Systemmitteln gesicherten Informationen darin die Folge sein. Ein Beispiel dafür stellt der „Atom Bombing“ Angriffe auf die AtomTables, systemeigene gemeinsam genutzte Speicherbereiche für Strings und deren Identifyer, dar, mit dem es theoretisch gelingt, vollständig gepatchte Systeme zu übernehmen. (Microsoft; SentinelOne)
Das besondere an diesen Angriffen ist, dass unter bestimmten Bedingungen sogar die Programme für einen Angriff genutzt werden können, die dazu dienen das System im Normalfall

zu schützen. So zum Beispiel die von Thomas Dullien und Google Project Zero entdeckte Schwachstelle CVE-2018-0986. Bei dieser Schwachstelle kann ein Fehler in der Scanengine des Windows Defenders ausgenutzt werden, um in gepackten Dateien enthaltenen Schadcode auszuführen, statt diesen zu erkennen. Ähnliche Schwachstellen gab es auch schon in andere Antiviren Produkten, beispielsweise dem CVE-2007-4560 im freien AV Produkt ClamAV. (Mitre.org, 2007; Thomas Dullien, 2018)

4.5 Privilege Eskalation

Ziel dieser Angriffstechnik ist es, egal mit welchen Rechten der Angriff gestartet wurde, diese so weit zu erhöhen, dass das Ziel des Angriffs erfüllt werden kann. Diese Rechte sind meistens Admin, oder System Rechte.

4.5.1 Wie sind Userrechte aufgebaut

Unter Windows gibt es in aktuellen Systemen zwei große Berechtigungsgruppen:

- **Benutzer**:
  Diese haben das Recht, ein System zu verwenden, also eine interaktive lokale Anmeldung am System durchzuführen. Sie dürfen Einstellungen am System treffen, die ihren Benutzeraccount betreffen, nicht aber das ganze System. Sie können von ihnen erstellte Dateien über das Recht als Ersteller/Besitzer, erstellen, lesen, bearbeiten und löschen. Sie sollten aber im Generellen nicht in der Lage sein, Applikationen zu installieren oder Veränderungen an Dateibereichen wie „c:\Programme“ durchzuführen. Ausnahmen bei der Installation von Programmen stellen Programme dar, die sich einzig in das Appdata Verzeichnis des Users installieren. Beispiele dafür wären MS-Teams, Spotify, oder die Dropperprogramme diverser Malwareprodukte.
- **Administratoren:**
  Mitglieder der Gruppe Administratoren, haben in einer unveränderten Windowsinstallation grundsätzlich alle Rechte auf das System. Einzig einige Systemverzeichnisse, sowie Benutzerverzeichnisse anderer Benutzer sind nicht von Haus aus für sie zugänglich. Auf diese Bereiche kann sich ein Administrator die Zugriffe jedoch selbst erteilen. Während in früheren Versionen von Windows, vor Vista, Benutzer der Rechtegruppe Administratoren, alle Programme mit vollen administrativen Rechten ausgeführt wurden, muss dies seit Vista durch Bestätigen eines UAC - User Account Control Dialogs bestätigt werden. Dies erfolgt zur zusätzlichen Absicherung auf einem Savescreen, der nur schwer von anderen Prozessen manipuliert werden kann (Remote Schutz).
  Ein Sonderfall unter den administrativen Benutzern ist der bei der Installation erstellte Administratoren Account, der als einer der „Well known Security Principals“ gilt und dessen SID auf -500 endet. Dieser Account kann, wenn er nicht von einem anderen Administrator deaktiviert wird, beispielsweise nicht durch eine Accountsperrrichtlinie bei zu vielen falschen Passworteingaben gesperrt werden.

Ein weiterer Sonderfall ist der Benutzer System, dieser ist nicht für die interaktive Anmeldung durch eine Anwenderin oder einen Anwender gedacht, sondern, dient dem Betriebssystem und Diensten, die im System gestartet werden zur Autorisation und zum Durchgriff auf die Festplatte. Ein im Kontext dieses Accounts gestarteter Prozess, läuft mit umfassenden Rechten auf das System, analog zu einem Prozess der als Administrator mit UAC Bestätigung gestartet wurde.

4.5.2 Möglichkeiten für eine Privilege Eskalation.

Die Möglichkeiten zur Privilege Eskalation ergeben sich zu einigen Teil aus den weiter oben beschriebenen Angriffsmöglichkeiten.

So bieten beispielsweise veraltete Installationen von Windows ein größeres Angriffsfenster, indem Angriffswege nicht korrekt geschlossen wurden und Services und Programme Schwachstellen aufweisen, die es ermöglichen durch Einschleusen von Code oder einfachen Befehlen, neue Prozesse in deren Kontext aufzurufen.
Ein unsauber gesetzter „Ausführen in“ Eintrag eines Service könnte ebenfalls missbraucht werden. Wenn dieser nicht sauber mit Anführungszeichen abgesichert ist, erfolgt der erste Versuch des Ladens von Programmteilen unter C:\Program statt unter „C:\Program Files“ und da C:\Program nicht per Default existiert, kann eine Nutzerin oder ein Nutzer, bzw. ein Programm im Nutzerkontext, diesen Ordner erstellen und eine bösartige Payload dort positionieren. Sobald das System das nächste Mal gestartet wird und versucht den Dienst zu starten, wird das Schadprogramm im Kontext des User System ausgeführt und die Erhöhung der Privilegien war erfolgreich.

Abgesehen von technischen Maßnahmen zur Erhöhung von Rechten, darf auf Systemen auf denen Benutzer selbst Administratorenrechte besitzen die Option des Social Engineering auch nicht außer Acht gelassen werden. So werden zum Beispiel „Rechnungen“ im Word Format per Mail verschickt, die den Empfängern erklären, dass sie zur Anzeige der Rechnung bestimmte Schritte durchführen müssen - meist das Aktivieren von Makros und das Zulassen von Administratorenrechten. Wenn ein User dann dieser Anleitung folgt, kann auch auf diesem Weg der Schadprozess direkt mit Administratorenrechten gestartet werden, gänzlich ohne dass Schutzmechanismen auf technische Weise überwunden werden müssen.

Sollte der Angriff nur darauf abzielen, der Anwenderin oder dem Anwender selbst und nicht einem Programm Administratorenrechte zu verschaffen, bieten sich lokale Angriffsmöglichkeiten auf nicht verschlüsselte Benutzerdaten in Konfigurationsdateien, oder Überreste des Imageprozesses in denen administrative Zugangsdaten ggf. sogar auf Domainebene, hinterlegt sind an. Sollte es sich um ein unverschlüsseltes System handeln, bleibt zusätzlich die Möglichkeit, das Kennwort des Administratorenaccounts mit Hilfe einer Boot-CD/eines USB-Sticks zurückzusetzen, sofern bei der Absicherung des Systems der Start von externen Medien erlaubt ist. (MITRE, 2019)

5 ANGRIFFSRANKING DER LETZEN JAHRE

„Mögest du in interessanten Zeiten Leben“
-Chinesischer Fluch / Robert F. Kennedy

Die letzten Jahre haben gezeigt, dass wir im Bereich der IT-Sicherheit in mehr als interessanten Zeiten leben. Die Gefahren Szenarien haben sich mit der immer weiter voranschreitenden Vernetzung immer weiter gewandelt. Von der Gefahr mit einem physischen Datenträger über Autoplay versehentlich einen Rechner zu infizieren, über die Frühzeiten des Internets wo Gefahren in den grauen oder dunklen Bereichen des Netzes auf unvorsichtige Opfer gewartet haben, zur aktuellen Situation, wo sich legitime Seiten mit gekaperter Werbung, realistisch wirkenden Phishing Wellen und IoT Angriffe, zu den bereits bestehenden Angriffsarten hinzugesellen.

Was sich hier zeigt ist, dass vielleicht mit Ausnahme des Bereichs der Premium Dialer, auf Grund des Aussterbens von Dial-Up Verbindungen, keine der Bedrohungen, die es seit der Frühzeit gab, ausgestorben ist. Es scheinen nur neue hinzuzukommen und bestehende Bedrohungen leicht adaptiert zu werden.

Am Beginn der 2010er Jahre waren es noch Erpressungsviren, die den Desktop des Opfers gesperrt haben und durchschnittlich Beträge in den niedrigen hundert Euro in Geschenkkarten von Onlinehändlern gefordert haben, um eine fiktive Strafverfolgung einzustellen und den Desktop wieder freizugeben. Heute werden nach der Verschlüsselung ganzer Firmennetzwerke meist mehrere Millionen Euro gefordert, um eine Entschlüsselung der Daten zu ermöglichen und eine potenzielle Veröffentlichung der Daten zu verhindern.

Auch wenn diese Angriffe wie in der Abgrenzung bereits definiert kein genauer betrachteter Teil der Arbeit sind, zeigen die letzten Monate mit mehreren, sehr medienpräsenten Angriffen, dass durch die Verwendung von Managed Services zusätzliche Gefahrenstellen geschaffen werden, wenn vertrauenswürdige Stellen übernommen werden und dazu verwendet werden, um Schadsoftware, in den meisten Fällen Verschlüsselungstrojaner, unter den Kundinnen und Kunden zu verteilen.

Im Folgenden wollen wir die Statistik der letzen Jahre etwas näher betrachten und versuchen auf Basis der unterschiedlichen Quellen eine gemeinsame Übersicht über die aktuelle Bedrohungslage zu bilden.

5.1 Zahl der Bedrohungen

Im Folgenden wollen wir einen statistischen Blick auf die aktuelle Bedrohungslage im Gesamten und im Speziellen auch auf die letzten Monate werfen.

5.1.1 Malware

Wenn man die Zahlen betrachtet, kann man relativ gut sehen, dass das relative Sicherheitsgefühl vieler Administratorinnen und Administratoren sie nicht trügt. Zum Zeitpunkt der Erstellung dieser Arbeit waren laut AV-Test GmbH, 1.353.430.000 unterschiedliche Varianten an Malware bekannt. (AV-TEST GmbH, 2022)

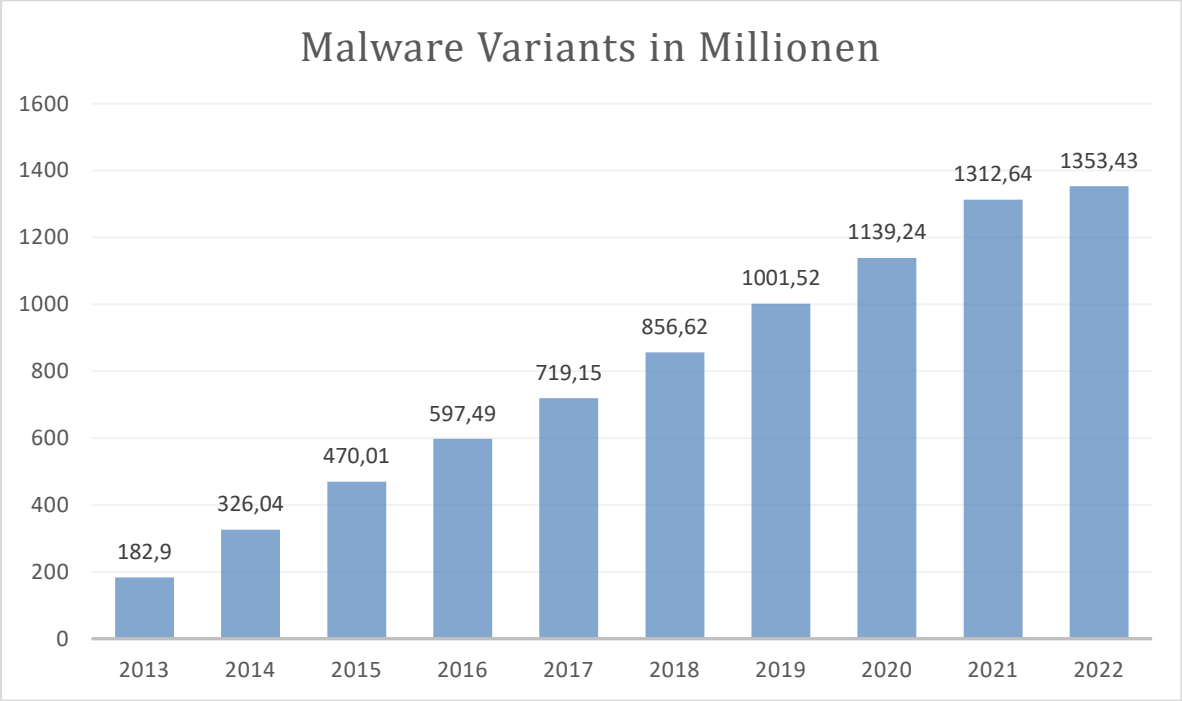

Abbildung 5-1 (AV-TEST GmbH, 2022)

Dabei zeigt sich auch gut, dass in den letzten Jahren, jedes Jahr mehr als 130 Millionen neue Varianten dazu gekommen sind. Diese Zahl der neuen Varianten ist der Entwicklung hin zu Baukastensystemen geschuldet. Diese ermöglichen es, Erpressungssoftware von bekannten Gruppen, auf das gewünschte Ziel angepasst für einen Anteil an der Erpressungssumme, zu mieten. So ist es auch Personen deren technisches Verständnis nicht über das Benutzen eines TOR-Browsers hinaus reicht möglich, eine Ransomware Kampagne gegen die von ihnen gewünschten Ziele zu starten.

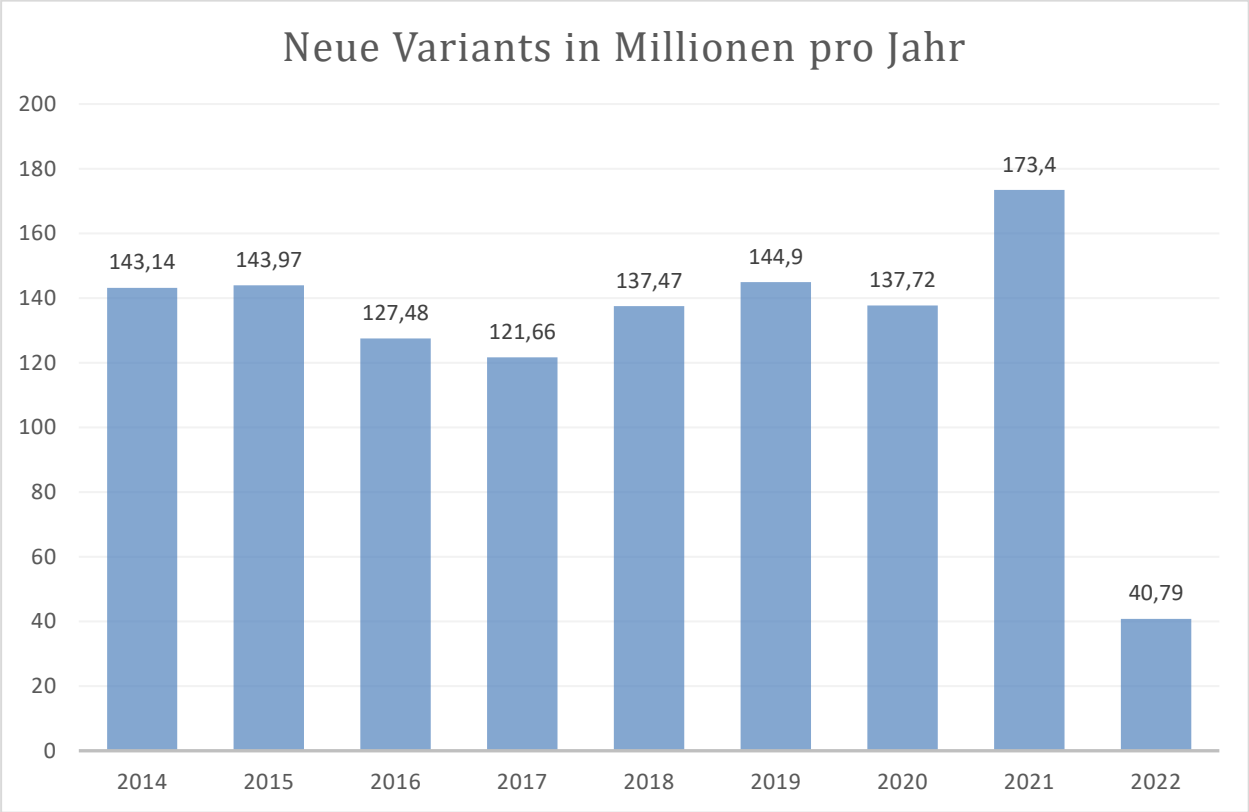

Abbildung 5-2 (AV-TEST GmbH, 2022)

Die Statistik zeigt hier schon, dass allein in diesem Jahr schon 38 Millionen neue Malware Samples erkannt wurden.

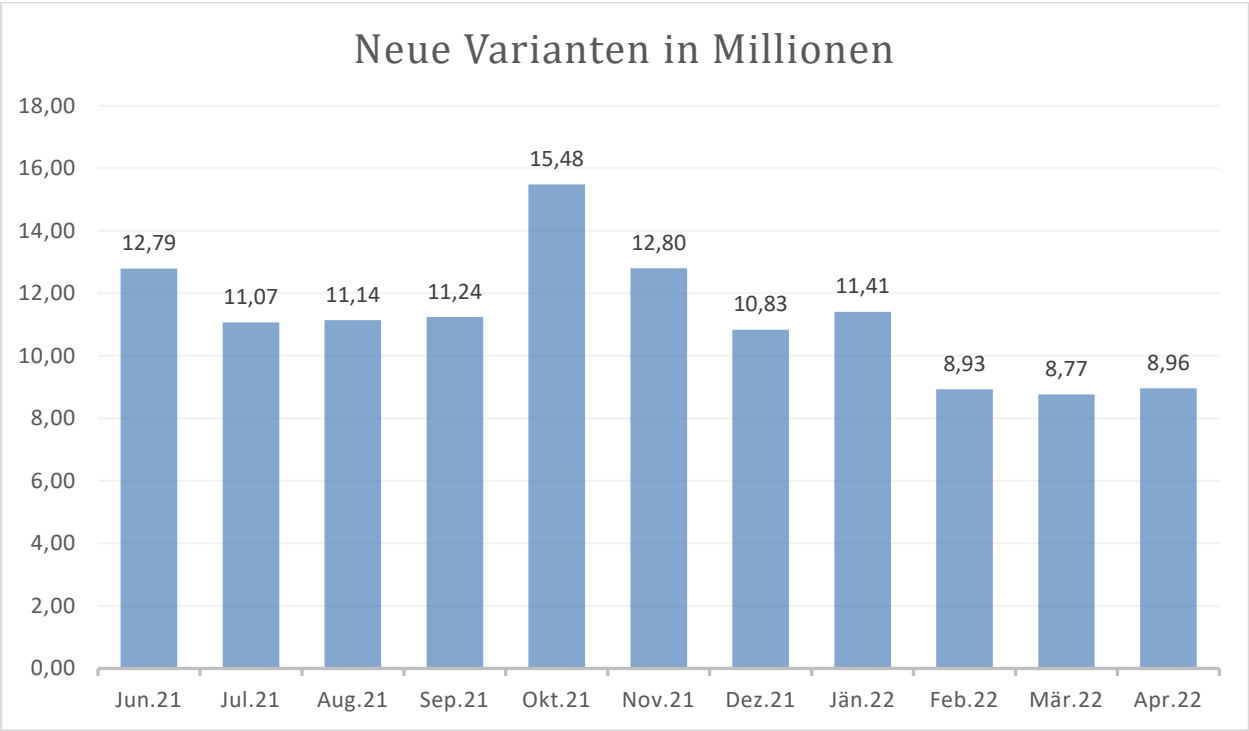

Abbildung 5-3 (AV-TEST GmbH, 2022)

Und bei diesen Zahlen zeigt sich zum einen, bis zu einem gewissen Punkt, das Problem von klassischen signaturbasierten Virenscannern. Zum anderen zeigt es auch gut, wie viel Energie Angreifer aufwenden, um doch einen erfolgreichen Angriff zu ermöglichen und Verteidigungstechniken zu überwinden versuchen.

Den Zahlen einer Industrieumfrage eines namhaften Sicherheitssoftware Herstellers nach, wurden im Jahr 2021, bei knapp 57% der befragten Unternehmen, zumindest Angriffe mit Ransomware festgestellt. (Sophos, 2021, S. 5)

Bei durchschnittlich 54 % dieser Angriffe ist es den Angreifern auch gelungen, Daten zu verschlüsseln, bei 7 % wurden zusätzlich Daten abgezogen und mit der Veröffentlichung gedroht. Hier wäre eines der aktuellen Beispiele der Angriff auf die IT-Systeme des Landes Kärnten.

5.1.2 Schwachstellen

Im Folgenden betrachten wir die am leichtesten auszunützende Angriffsoberfläche unter Windows. Bekannte Schwachstellen, die bereits gemeldet wurden und mit einer CVE Nummer versehen wurden.

Dabei fällt auf, dass 2020 scheinbar auch einige White Hat Hacker oder Sicherheitsforscher viel Zeit hatten, da in diesem Jahr, fast doppelt so viele Schwachstellen gemeldet wurden, wie im Jahr davor oder auch dem Folgejahr. (CVE-Details, 2022b)

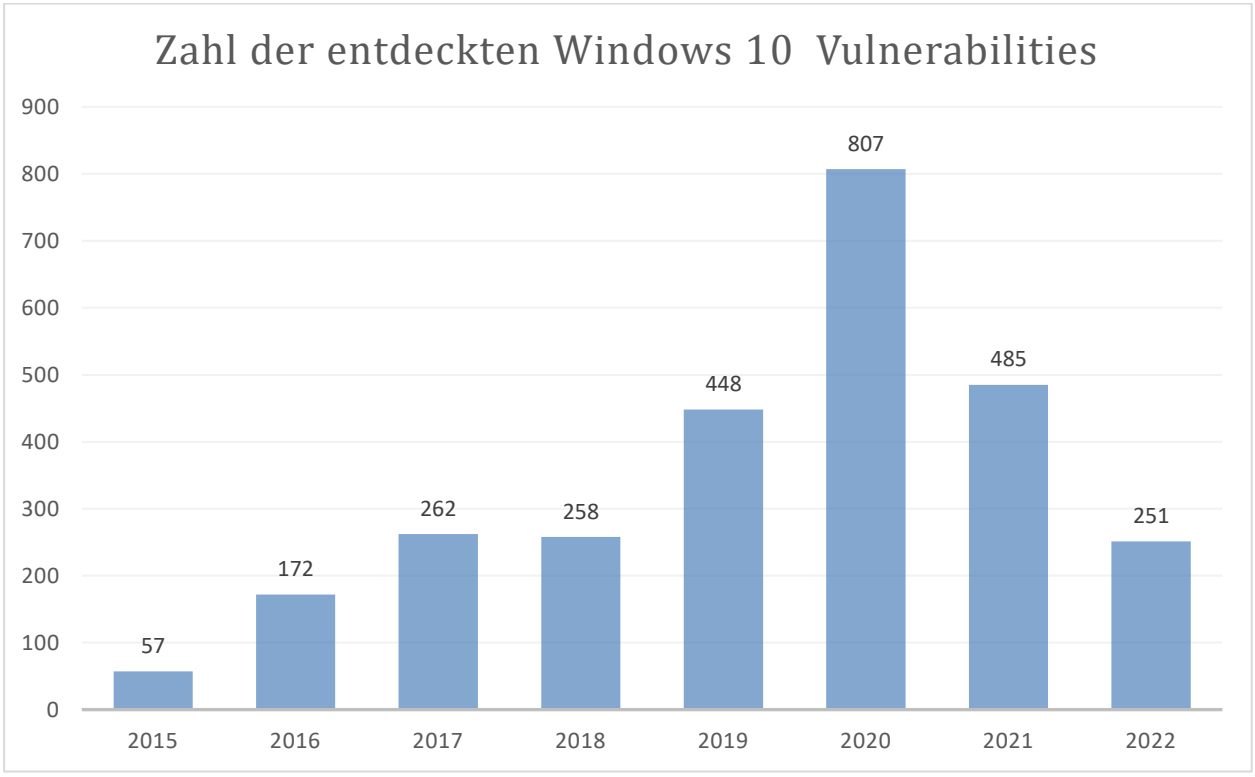

Abbildung 5-4 (CVE-Details, 2022b)

Ein ähnliches Bild zeigt sich auch im Bereich des betrachteten Serverbetriebssystems „Windows Server 2016“. Auch hier zeigt das Jahr 2020 eine überdurchschnittlich hohe Anzahl an neu gemeldeten Schwachstellen. (CVE-Details, 2022c)

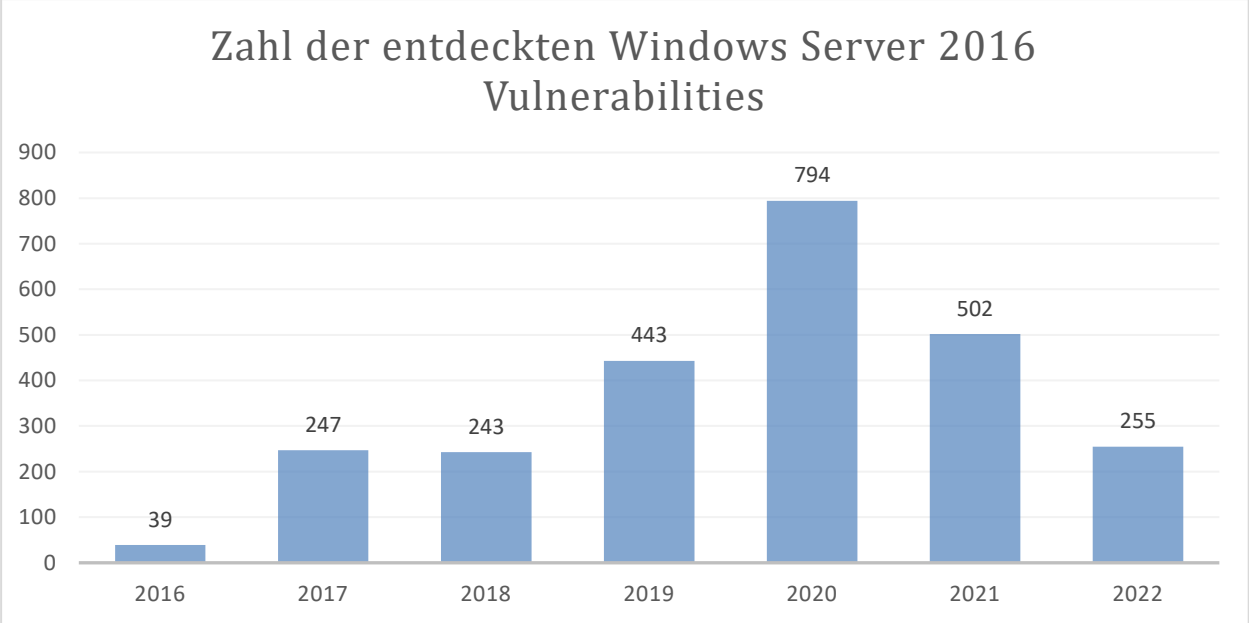

Abbildung 5-5 (CVE-Details, 2022c)

Ein Blick auf die Verteilung der Schwachstellen zeigt auch, dass die Kategorien, „Code Execution“, „Gain Privilege“ und „Bypass Something“ zusammen, mehr als 50% aller gemeldeten Schwachstellen ausmachen. Dies garantiert, in Anbetracht der Tatsache, dass die erfolgreiche Kombination dieser drei Kategorien in einem Angriff, quasi den Erfolg .

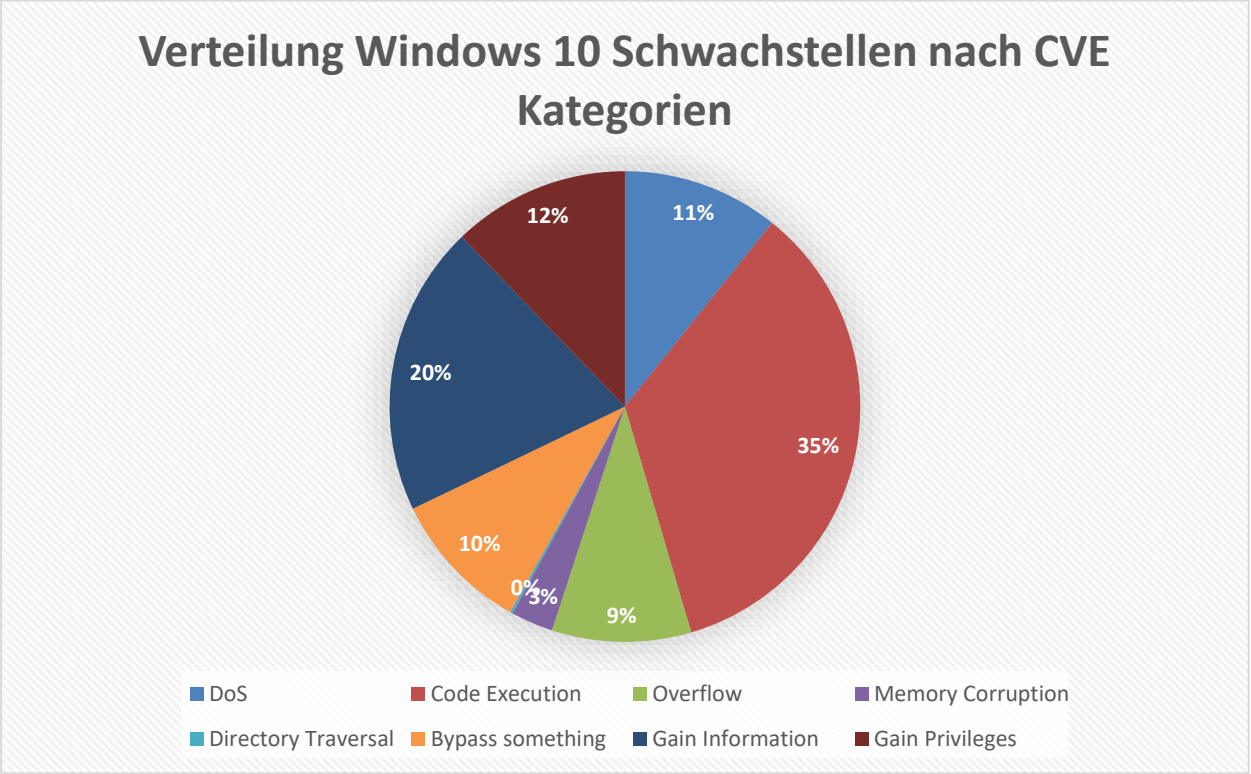

Abbildung 5-6 (CVE-Details, 2022b)

Ein ähnliches Bild zeigt sich auch im Serverumfeld. Wobei hier die Verteilung minimal anders aussieht. Während auch hier die kritischen drei Kategorien, über 50 Prozent ausmachen, betreffen im Server Umfeld etwas mehr Schwachstellen die Code Ausführung und weniger Schwachstellen das Erhöhen von Privilegien. Dies zeigt, dass trotz des eigentlich identen Kernels, Unterschiede in entscheidenden Sicherheitsbereichen existieren.

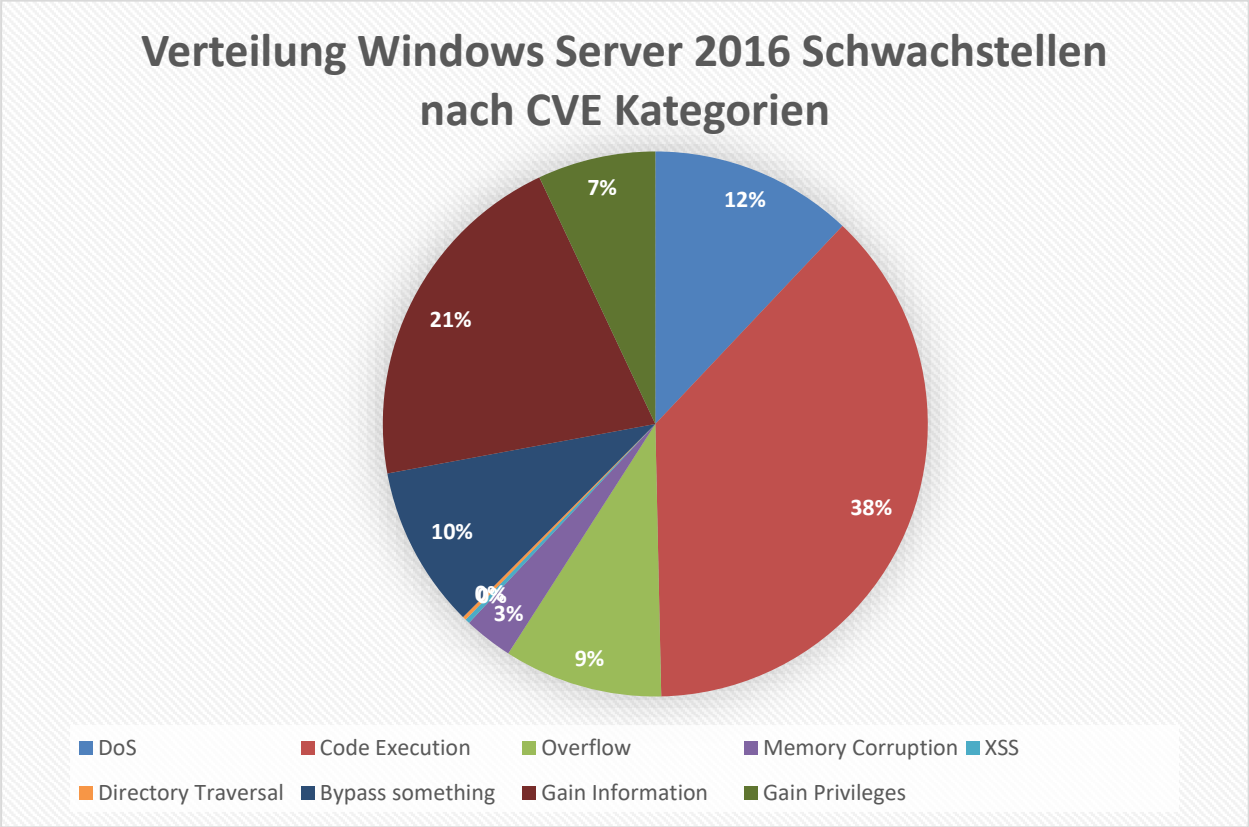

Abbildung 5-7 (CVE-Details, 2022c)

Zum Abschluss des Themenkomplexes Schwachstellen und um die oben gezeigten Zahlen der Schwachstellen im Windows Umfeld in den Kontext des Gesamten CVE Systems zu setzen, folgt hier noch der kurze Blick auf die Gesamtzahlen der gemeldeten Schwachstellen über den für Windows 10 und Windows Server 2016 relevanten Vergleichszeitraum. (CVE-Details, 2022a)

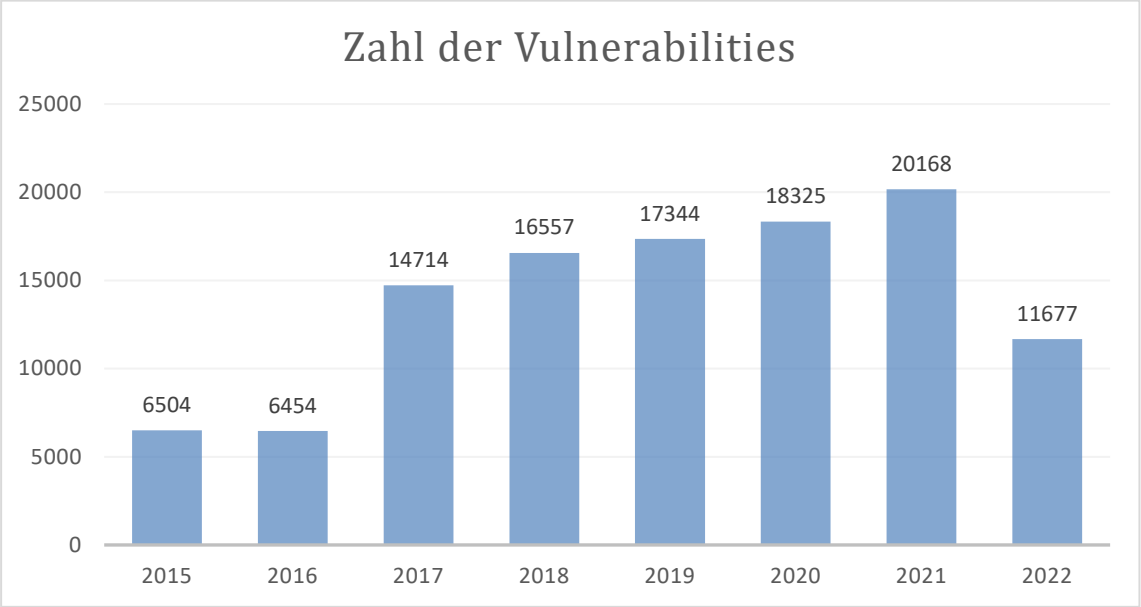

Abbildung 5-8 (CVE-Details, 2022a)

5.2 Angriffsstatistiken & Prognosen

Nachdem im vorangegangenen Kapitel die Verfügbarkeit sowie die möglichen Vektoren kurz beleuchtet wurden, soll hier darauf eingegangen werden, wie diese genutzt werden und damit die aktuelle Sicherheitslage, der sich Administratoren und Administratorinnen stellen müssen.

Bei der Suche nach konkreten Zahlen und Statistiken in diesem Bereich zeigt sich gut das Problem der Ehrlichkeit, was die Bekanntgabe von Vorfällen betrifft. So schwanken die zu diesem Thema gefundenen Zahlen teilweise massiv. Wenn sich Unternehmen und Sicherheitsverantwortliche nur der Zahl der Angriffe gegenübergestellt sehen würden, die laut Kriminalstatistik des Bundeskriminalamts (BKA) laut der jährlichen Berichte zur Anzeige gebracht werden, würde es wahrscheinlich weniger Posten für Sicherheitsverantwortliche geben. Die Betrachtung dieser Statistik zeigt, dass es im Jahr 2021 in Österreich nur 354 Anzeigen wegen „Datenbeschädigung“ §126a StGB, somit sieben Anzeigen weniger als im Jahr 2020, gab. Ähnlich sieht das Bild bei Anzeigen nach §126b StGB „Störung der Funktionsfähigkeit eines Computersystems“ aus. Hier gab es zwar mit 95 Anzeigen einen Anstieg gegenüber der 71 Anzeigen die noch 2020 gestellt wurden, aber auch dies deckt sich nicht mit Statistiken, die von Sicherheitsfirmen veröffentlicht werden. Unabhängig von der Anzahl der Straftaten ist auf jeden Fall die Aufklärungsquote über die vergangenen Jahre sehr stabil. Bei der Aufklärung von §126a StGB lag der Wert in den vergangenen beiden Jahren bei 21 % im Jahr 2020 und 18 % im Jahr 2021. Noch schlechter sieht es bei der Aufklärung von §126b StGB aus. Während es im Jahr 2020 noch 14 % der Fälle waren, die geklärt wurden, ist die Aufklärungsquote im Jahr 2021 sogar noch um zwei Prozent auf 12 % gefallen. (Bundesministerium für Inneres, 2021, 2022)

Diesen Zahlen kann man noch gegenüberstellen, dass bei einer Befragung des Handelsverbands Österreich unter Onlineshop Kunden, mehr als ein Drittel der Befragten angegeben hat bereits

Opfer einer Infektion mit Schadsoftware geworden zu sein und zusätzliche sechs Prozent angegeben haben bereits Opfer „Digitaler Erpressung“ (Ransomware) gewesen zu sein. (Handelsverband, 2022)

Ein Fakt, der in den meisten publizierten Datensätzen zu Angriffen in den vergangenen Jahren unumstritten ist, auch wenn die genauen Zahlen je nach Bericht um einige Prozentpunkte schwanken, ist, dass der derzeit für den initialen Angriff von Unternehmen sowie Endbenutzer am meisten genutzte Vektor, wie schon in den Jahren davor, weiterhin E-Mail ist. Hier hat sich vor allem in den letzten beiden Jahren der Coronakriese gezeigt, dass diese E-Mails immer besser werden. Waren es früher in vielen Fällen noch schlecht geschriebene und damit leicht zu erkennende Nachrichten, handelt es sich nun zumindest bei Angriffen auf Firmen meist um korrekt geschriebene Nachrichten, die in ihrem Kontext Sinn ergeben. In diesem Bereich haben sich in den letzten Jahren die Gefahren der extern, ohne Einschränkung erreichbaren Mailboxen gezeigt. So kann ein einfaches, mit einem wiederverwendeten Kennwort gesichertes Exchange Online Konto für einen Angreifer ausreichen, um über eine gewisse Zeit den E-Mail-Verkehr mitzulesen und im Anschluss einen Angriff in einer für dieses Konto glaubwürdigen Form auf das Unternehmen selbst, oder Geschäftspartner, die mit dem Unternehmen in Kontakt stehen starten. Sofern die darin verwendete Schadsoftware unbekannt ist, oder Maßnahmen zum Umgehen von Schutzmechanismen getroffen werden, erhält das Opfer eine Mail, von einem vertrauenswürdigen Absender, der alle Prüfungen zur Authentizität bestanden hat. Wenn es sich dann noch um einen geschickten Angreifer handelt, der den Text der Nachricht korrekt formuliert, ist es vergleichsweise leicht den Empfänger der Nachricht zum Ausführen des Anhangs, oder Aufrufen eines Links zu bringen. Es besteht also kein Vergleich zu den in der Vergangenheit häufig aufgetretenen Mails, im Namen einer Bank, die in gebrochener Sprache dazu aufgefordert haben, eine Sicherheitssoftware der Bank von einer vollkommen unzusammenhängenden URL herunterzuladen. (Harris Insight & Analytics LLC, 2021; Morgan, 2020; Stouffer, 2021)

Um die Schäden und die finanzielle Tragweite der aktuellen Situation und der prognostizierten Entwicklung in einen Kontext zu setzten, kann man die monetären Schäden im Vergleich zur weltweiten Wirtschaftsleistung betrachten. Während die durch Angriffe verursachten Schäden im Jahr 2015 mit drei Billionen Euro noch knapp hinter dem Bruttoinlandsprodukt Deutschlands, der größten Europäischen Volkswirtschaft, lagen, zeigen die Daten für das vergangene Jahr 2021 bereits einen Schaden von sechs Billionen Euro. Damit hatte das Finanzvolumen des Schadens durch Cyberkriminalität im vergangenen Jahr Platz drei in der Liste der größten Volkswirtschaften inne. Und die Prognosen gehen weiter von einem jährlichen Wachstum aus, sodass erwartet wird, dass im Jahr 2025 die Grenze von zehn Billionen Euro überschritten wird. Angemerkt sollte jedoch werden, dass in diesen Prognosen und generellen Kostenaufstellungen auch Schäden abseits von Arbeitsausfällen durch beschädigte Systeme und Lösegeld fällt. Wenn man Erpressungssoftware allein betrachtet, sind die finanziellen Schäden von zirka 325 000 Dollar im Jahr 2015 auf zwanzig Milliarden Dollar im letzten Jahr angestiegen. Damit haben sich die Schäden innerhalb von sechs Jahren fast versechszigfacht. (Morgan, 2020) (Stouffer, 2021)

Auch in den Medien ist die angespannte Bedrohungslage in den letzten Jahren immer präsenter. So beispielsweise die Berichterstattungen über den Ransomware Befall der Kärntner

Landesregierung im Mai 2022 oder die erste, mittlerweile fallengelassenen Mordanklage im Zuge eines Angriffes mit Verschlüsselungssoftware auf ein Deutsches Krankenhaus. (Born, 2020)

Trotz aller medialen Aufmerksamkeit und dem, zumindest in Umfragen widergespiegelten Unsicherheitsgefühl, gaben 58 % der Befragten in einer Umfrage von Symantec an sich „mehr denn je“ Sorgen zu machen von Cyberkriminalität betroffen zu sein. Im Jahr 2021 haben dennoch 47 % der befragten Unternehmen mindestens einen Vorfall gehabt, in dem mindestens eine Mitarbeiterin oder ein Mitarbeiter versucht hat, Schadsoftware herunterzuladen. (Check Point Reserch, 2021; Stouffer, 2021)

5.3 Angriffsbetrachtung

Im Folgenden werden mehrere Angriffsketten aufgezeigt, die zu einem Schadsoftware Ausbruch in mehreren Unternehmen geführt haben. Es wurde dabei versucht die Beispiele so zu wählen, dass eine möglichst breite Palette an Angriffsszenarien abgebildet wird. Während die Absicherung von externen Produkten nicht Teil dieser Arbeit ist, wird bei der Angriffsbetrachtung eine Ausnahme gemacht, solange es sich rein um einen initialen Vektor handelt, in Folge aber nicht mehr für den Angriff genutzt wird. Dabei wurde darauf geachtet, dass bei den hier betrachteten Beispielen keine der lokalen Schutzmaßnahmen durch die initiale Kompromittierung deaktiviert wurden.

Der erste betrachtete Angriff wurde von Trend Micro analysiert, welche auch ein „Post Incident Writeup“ zu diesem Fall veröffentlicht haben. Das finale Ziel des Angriffs war, das Verteilen und Ausführen der Black Cat Malware, welche vor allem in Österreich, seit dem Angriff auf die Kärntner Landesregierung im Frühjahr 2022 zusätzliche Bekanntheit erlangt hat. Dabei handelt es sich um ein, in der Programmiersprache Rust geschriebenes, Schadprogramm welches wie schon beschrieben als „Ransomware as a Service“ angeboten wird und dessen Ziel eine zweifache Erpressung des Opfers ist. Um zu verhindern, dass sich Opfer mit einem vollständigen und vor Löschung durch die Schadsoftware gesichertem Backup, weigern für die Entschlüsselung den geforderten Betrag zahlen, werden vor der Datenverschlüsselung auch relevant wirkenden Daten abgezogen und mit der Veröffentlichung gedroht. Je nach Brisanz der Daten, kann dies eine höhere Erfolgsquote als der Verkauf des Entschlüsselungscodes haben.

In dem von Trend Micro betrachteten Fall wurde als initialer Vektor eine Schwachstelle in einem Microsoft Exchange Server genutzt (CVE-2021-31207), welche es einem unautorisierten Angreifer erlaubt mit Hilfe einer PowerShell Verbindung, Daten aus einer Mailbox im Kontext des Benutzers, in welchem der Exchange Dienst gestartet ist, ins Filesystem zu kopieren. Diese Möglichkeit wurde dazu genutzt, um eine Remote Shell in einem Unterordner des lokalen Webservers abzulegen, welche von der lokal laufenden Defender Installation nicht als schädlich erkannt wurde. Nachdem dieser initiale Angriff geglückt war, wurde die Web Shell dazu verwendet Code aus der Ferne auf dem System auszuführen. Im hier betrachteten Fall wurde in Folge eine PowerShell Instanz dazu genutzt, mit Hilfe der Fähigkeit Webabfragen aus der PowerShell heraus zu verwenden, eine DLL von einem Remote Host herunterzuladen. Bei der auf diesem

Weg nachgeladenen DLL, handelte es sich in diesem Fall um eine manipulierte Form einer OpenSLL DLL, welche mit einem nicht mehr gültigen Signaturzertifikat der Firma Zoom signiert war. In Folge wurde über die Applikation Rundll32.exe die heruntergeladene DLL zur Ausführung geladen. Ziel dieses Vorgangs ist es, den in der DLL positionierten Shell Code des Cobalt Strike Toolkits auszuführen welcher teilweise unverschlüsselt und teilweise verschlüsselt neben einer Menge an unnötigem Code in dieser Datei vorhanden war. Nach der Entschlüsselung des Stager Codes und der folgenden Ausführung, findet anhand von vordefinierten Parametern ein weiterer Webaufruf statt, welcher weiteren auszuführenden Code nachlädt. An keinem der bis jetzt gesehenen Punkte, hat der Windows Defender auf dem System angeschlagen und einen Prozess gestoppt oder eine verdächtige Datei isoliert. Der nächste Schritt dieses Angriffs war das Laden des WerFault.exe Prozesses, um diesen als vertrauenswürdigen Husk für eine Code Injection zu nutzen. Mit diesem auf diese Weise manipulierten Prozess wurden in Folge weitere Programme zur Untersuchung des Netzwerks nachgeladen sowie unter Verwendung legitimer lokalen Microsoft Dienstprogramme weitere Erkundungen durchgeführt. So wurden mit Hilfe der net.exe lokale und domänenbasierte Adminaccounts gelistet. Bei den Nachgeladenen Programmen handelte es sich mit NetScan und Bloodhound um klassische Tools zum Durchsuchen des Netzwerks und Auflisten von verfügbaren Shares im Fall von Net Scan und zum Auslesen der Active Directory Infrastruktur inklusive der Visualisierung des kürzesten Angriffspfads von einem derzeit kontrollierten Accounts zu den gewollten Domänen Administrationsrechten. Als weitere Angriffstools wurde CrackMapExec und die PowerShell Version von Inveigh heruntergeladen. Mithilfe der Version von Inveigh wurde zum Mitschneiden von mDNS und NTLM Traffic verwendet. Als letzter Schritt, vor der Ausführung der wirklichen Schadfunktion, wurden Kopien der manipulierten DLL Datei mittels SMB Verbindungen an weitere Computer im Netzwerk verteilt. Im Anschluss wurde die Schadfunktion mit Hilfe zweier nicht mehr rekonstruierbarer Batch Dateien gestartet. Mit dem Start der Schadfunktionen wurde eine Kette von Systemprogrammen aufgerufen, um die Ziele des Angreifers zu erreichen. Für die Erstellung des Schlüssels und der TOR Adresse, sowie zur eindeutigen Identifikation des Computersystems wurde der System Universally Unique Identifier (UUID) unter Verwendung der WMI Command-Line (WMIC) abgerufen. Mit Hilfe der vssadmin.exe, der Applikation zur Steuerung des Volume Shadow Copy Services (VSS) wurden alle bestehenden Shadow-Copys gelöscht. Dieser Vorgang verhindert ein schnelles Wiederherstellen einer verschlüsselten Datei, indem man diese auf einen Snapshot der Datei vor der Verschlüsselung zurücksetzt. Zusätzlich wurde das Verhalten des Systems beim Folgen von symbolischen Links auf extern liegende Dateien über einen Parameter der Anwendung Fsutil angepasst, sodass die Schadsoftware in der Lage ist, auch diesen Links zu folgen und dort die Verschlüsslung fortzusetzen. Weiters wurde über die BCDedit, die Option gelöscht das System in den Wiederherstellungsmodus zu versetzen. Dabei handelt es sich normalerweise um ein kleines Windows PE Image, welches in der Lage ist, ein Basisset an Befehlen auszuführen, welches zur Bekämpfung der Infektion genutzt werden kann. Mit Hilfe eines einfachen Kommandozeilenbefehls wurde die maximale Anzahl an Netzwerkverbindungen, die die lokale LAN-Manager Service erlaubt, massiv erhöht. Dies soll dafür sorgen, dass Verschlüsselungen auf Remote Verzeichnissen so schnell und parallel wie möglich erfolgen. Nachdem es, sofern das Remote System nicht kompromittiert wurde, bei einer langsamen Verschlüsslung auffallen könnte, wenn plötzlich verschlüsselte Dateien in einem

Remote Filesystem auftauchen und damit den Angriff durch Aussperren des verschlüsselnden Users zum Scheitern bringen würden. Sofern der ausführende Benutzer die nötigen Rechte hat, wird versucht mit Hilfe einer integrierten Version des Tools PsExec, einem Microsoft Tool zum Ausführen von Prozessen auf entfernten Systemen, die Malware, welche auch im Sysvol Netzwerkverzeichnis der Domäne abgelegt wurde, zu starten. Abschließend wurde noch der lokale Webserver über das Tool IISReset gestoppt und mit Hilfe des Programms wevtutil.exe alle Eventlogs auf dem System gelöscht. Bei der Verschlüsselung wendet die Ransomware eine interne Liste an Ausnahmen an, um zu verhindern in Folge der Verschlüsselung Systemdateien zu verschlüsseln, um das System unbrauchbar zu machen. Das würde in Folge die Chance reduzieren, dass das verlangte Lösegeld gezahlt wird. Nachdem mit Ausnahme einiger weniger Varianten, das Ziel der Infektion das Lukrieren des Lösegelds ist.

Um den Angriff auch hier in den Kontext des MITRE ATT&CK Frameworks zu setzen, die Im Zuge des Angriffs genutzten Techniken sind:

- T1027.002 – Obfuscated Files or Information: Software Packing
- T1027 – Obfuscated Files or Information
- T1007 – System Service Discovery
- T1059 – Command and Scripting Interpreter
- TA0010 – Exfiltration
- T1082 – System Information Discovery
- T1490 – Inhibit System Recovery
- T1485 – Data Destruction
- T1078 – Valid Accounts
- T1486 – Data Encrypted For Impact
- T1140 – Encode/Decode Files or Information
- T1202 – Indirect Command Execution
- T1543.003 – Create or Modify System Process: Windows Service
- T1550.002 – Use Alternate Authentication Material: Pass the Hash

(Global Research & Analysis Team, Kaspersky Lab, 2022; Silva & Froes, 2022)

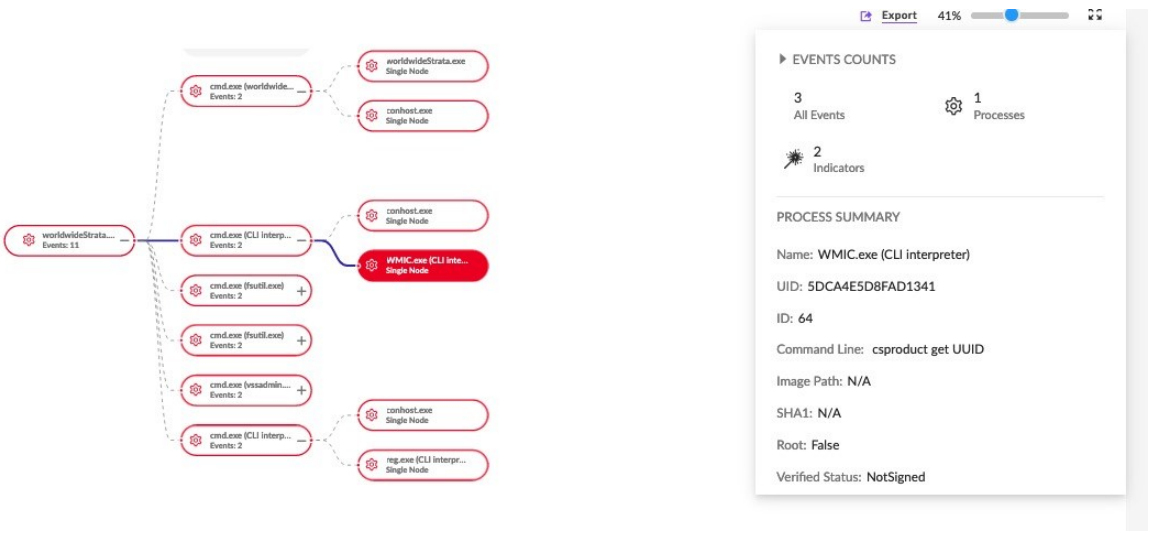

Abbildung 5-9- Blackcat KillChain - SentinelLabs

Eine weitere Angriffsserie der vergangenen Jahre war die Emotet Malware, welche ihren Anfang 2014 als Banking Trojaner genommen hat und sich über die Jahre hin zu einem modularen und gefährlichen Werkzeug entwickelt hat. Während zwischen Jänner 2021 und Februar 2022 die Angriffe aus diesem Botnetz nach Aktionen von Europol und lokalen Behörden in der Ukraine stark nachgelassen haben, beobachtet man aktuell wieder einen starken Anstieg der Aktivität aus diesem Botnet. (Knop, 2022; Schirmacher, 2022; Westerhagen, 2021)

Spannend ist, dass sich bei dieser Schadsoftwarevariante über die gesamte Laufzeit ihrer Existenz, der Weg wie Systeme infiziert werden, nie nennenswert geändert hat. Einzig der Ansatz wie die Dateien übermittelt werden, wurde an die Fähigkeiten normaler Virenscanner und Abwehrfunktionen in Office Produkten angepasst. Die Datei, die die Infektion auslöst wird per Mail, aus dem Emoted Spam Botnet, versendet. Das passiert im besten Fall von einem kompromittierten Rechner der bereits Kontakt mit dem Opfer hatte. Während am Anfang direkt das Word oder Excel Sheet mit dem automatisch startenden Makro verschickt wurden, sind es mittlerweile verschlüsselte Zip Dateien, mit dem Ziel die Analyse durch einen Virenscanner in der Mailübertragung zu verhindern. Nachdem das Kennwort aber im Klartext in der Mail hinterlegt ist, kann der empfangende Anwender die Datei problemlos öffnen. In seltenen Fällen wird statt der direkt angehängten Datei auch ein Link zu einer infizierten Datei auf einem übernommenen Webserver genutzt. Öffnet ein Benutzer nun diese Datei, wird er von einem beschädigt aussehenden Dokument oder Excel Sheet begrüßt, mit einer Nachricht, die ihn dazu auffordert, die aus Sicherheitsgründen in modernen Office Versionen automatisch deaktivierten Makros zu aktivieren um die „Wichtige Rechnung“ oder Ähnliches anzuzeigen. Wenn der Anwender infolgedessen wirklich die Ausführung der Makros erlaubt, setzt sich die Infektionskette in Gang. Zu erwähnen ist, das sämtliche Befehle, die hier und in Folge genannt sind, in den Dateien nicht im Klartext abgelegt sind, sondern unter Zuhilfenahme diverser Verschleierungstechniken versuchen, sich vor klassischen Erkennungsmaßnahmen zu schützen. Je nach verwendeter Technik, ist der „Klartext“ meist nur zur Laufzeit im Speicher erkennbar, was auch den forensischen Aufwand im Nachgang stark erhöht. Sobald die Makro Ausführung beginnt, wird

eine Kette an Kommandozeilen aufgerufen, welche zum einen den Benutzer ablenken sollen, indem sie mit Hilfe der msg.exe eine Windows Benachrichtigung ausgeben, dass es ein Problem mit der Datei gibt und dieses geschlossen wird. Im besten Fall wird der Anwender damit beruhigt und ignoriert den Fehler, oder er versucht vom Absender eine korrigierte Version dieser Datei zu erhalten. Im Hintergrund wird aus einem der verbleibenden CMD Prozessen ein übergebener PowerShell Aufruf gestartet, der unter Verwendung des EX Bypass Flags eine durch das VBA Makro erstellte und im Benutzerprofil abgelegte .ps1 Datei ausführt. In dieser wird unter Laden der .NET Klasse WebClient ein Download der ersten Payloads gestartet. Dafür werden die Dateien aus einer Liste an verfügbaren URLs heruntergeladen, diese werden im selben Verzeichnis abgelegt wie schon die Skriptdatei, die für den Download der Dateien verantwortlich ist. Nach dem Download werden die lokalen Dateien validiert, um sicher zu stellen, dass diese die korrekte erwartete Größe haben. Dies soll verhindern, dass Daten von einem manipulierten Command und Control Server geladen und verarbeitet werden. In Folge wird, wie auch beim BlackCat Angriff, eine heruntergeladene DLL Datei mit Hilfe der Rundll32.exe geladen und eine Funktion daraus aufgerufen. In Folge werden über diesen Rundll32 Prozess weitere DLL Dateien nachgeladen. Als letzte Stufe des Download Prozesses, wird das Infektionsfile aus dem Userprofil in einen zufällig benannten Ordner unter c:\Winows\System32\ unter einem zufällig generierten Namen, mit einer ebenso zufallsgenerierten Erweiterung gespeichert. Nachdem dieser Schritt erfolgt ist, wird um eine Persistenz auf dem System zu erreichen mit Hilfe der Systemfunktion „CreateServiceW()“, welche direkt aus der geladenen DLL gestartet werden kann, ein automatisches Service generiert. Dieses Service hat zwar einen relativ kryptischen Namen, nämlich den zufällig generierten Namen der abgelegten DLL, verfügt aber in den letzten Versionen über eine von fünf sinnvollen Beschreibungen, die zumindest bei Betrachtung durch einen Leien standhalten. Gestartet wird wiederum die abgelegte Datei mit Hilfe der rundll32.exe. Die so gestartete Malware ist nun dazu bereit mit der Command und Control Infrastruktur zu kommunizieren und weitere Befehle entgegenzunehmen oder Komponenten nachzuladen. Diese können von reinen Schadfunktionen über die Informationssammlung bis hin zur Verschlüsselung und folgenden Erpressung reichen.

Um den Angriff auch hier in den Kontext des MITRE ATT&CK Frameworks zu setzen, die im Zuge des Angriffs genutzten Techniken sind:

- T1087.003 Account Discovery: Email Account
- T1560 Archive Collected Data
- T1547.001 Boot or Logon Autostart Execution: Registry Run Keys / Startup Folder
- T1110 Brute Force: Password Guessing
- T1059.001 Command and Scripting Interpreter: PowerShell
- T1059.003 Command and Scripting Interpreter: Windows Command Shell
- T1059.005 Command and Scripting Interpreter: Visual Basic
- T1543.003 Create or Modify System Process: Windows Service
- T1555.003 Credentials from Password Stores: Credentials from Web Browsers

- T1114.001 Email Collection: Local Email Collection
- T1573.002 Encrypted Channel: Asymmetric Cryptography
- T1041 Exfiltration Over C2 Channel
- T1210 Exploitation of Remote Services
- T1040 Network Sniffing
- T1571 Non-Standard Port
- T1027 Obfuscated Files or Information
- T1027.002 Software Packing
- T1003.001 OS Credential Dumping: LSASS Memory
- T1566.001 Phishing: Spearphishing Attachment
- T1566.002 Phishing: Spearphishing Link
- T1057 Process Discovery
- T1055.001 Process Injection: Dynamic-link Library Injection
- T1021.002 Remote Services: SMB/Windows Admin Shares
- T1053.005 Scheduled Task/Job: Scheduled Task
- T1552.001 Unsecured Credentials: Credentials In Files
- T1204.001 User Execution: Malicious Link
- T1204.002 User Execution: Malicious File
- T1078.003 Valid Accounts: Local Accounts
- T1047 Windows Management Instrumentation

Die hier gezeigte massive Anzahl an genutzten Angriffs- sowie Selbstschutztechniken, zeigt gut warum diese Malware Familie über viele Jahre erfolgreich aktiv war.

(Hornet Security; Knop, 2022; Navarette, Jia, Tennis & Shao, 2021; Schirmacher, 2022)

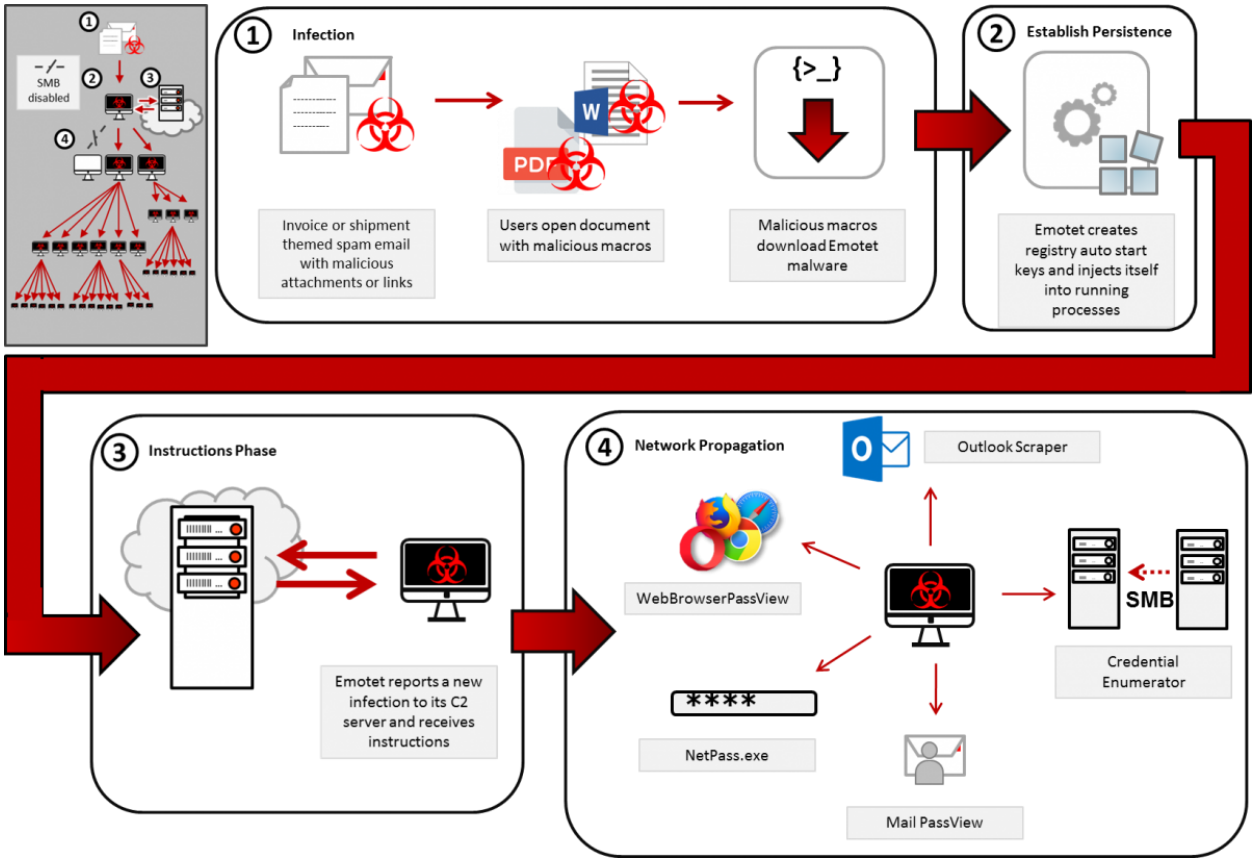

Abbildung 5-10- Emotet Infektionskette - US Cert

5.4 Relevante Verteidigungsgrundlagen.

Im Gespräch mit mehreren Sicherheitsexperten von mehreren namhaften Sicherheitsfirmen wurden die folgenden Angriffsmöglichkeiten und Schwachstellen identifiziert, die bei Audits als relevante Schwachstellen angesehen werden. Genannt werden hier die Kategorien, die in mehr als der Hälfte der Firmen Teil eines OS Audits sind.

Was leicht zu erkennen ist ist, dass sich auch diese Audit Szenarien mehr oder weniger stark an das MITRE ATTK&CK anlehnen, darum wird im Folgenden auch bei den betreffenden Checks, ein Verweis auf die entsprechenden Techniken innerhalb der MITRE ATTK&CK Matrix gesetzt, um die Punkte noch besser in einen Kontext setzen zu können.

5.4.1 Berechtigungsschwachstellen

Im Bereich dieser Schwachstellen geht es meist hauptsächlich darum sicher zu stellen, dass nicht privilegierte User den Zugriff bekommen, den sie zur Durchführung ihrer Arbeit und dem Ausführen der dafür nötigen Programme erhalten, aber nicht mehr. Also die strikte Einhaltung eines „Least Privilege“ Konzeptes.

Hier können vor allem Legacy Applikationen, die sich nicht an die Verwendung aktueller Ordner Strukturen halten und für einfache User Schreibzugriff in den „Programme“ Ordner vergeben, für einfache Fehkonfigurationen sorgen, in dem zu weiträumige Berechtigungen vergeben werden.

Die Folge einer solchen zu weitreichenden Berechtigung kann auf der einen Seite sein, dass Anwender an Informationen gelangen, die Sie nicht erhalten sollen, wenn beispielsweise die Verzeichnisse anderer User auf demselben System nicht korrekt geschützt sind. Dies wäre besonders auf Multiuser Systemen, wie Remote Desktop Servern ein Problem.

Auf der anderen Seite stehen die Möglichkeiten, die sich einem Angreifer bieten, um Malware einzuschleusen, indem System- oder Programmdateien ausgetauscht werden können, wenn auf den betreffenden Verzeichnissen die Berechtigungen so gesetzt sind, dass das Austauschen von bestehenden Dateien ermöglicht wird. Dies kann dazu führen, dass ein legitimer Nutzer eines Systems beim Start seiner gewohnten Applikationen auf einem System Malware exekutiert, ohne selbst etwas falsch gemacht zu haben. Wenn es dabei noch gelingt eine Applikation zu kompromittieren, die normalerweise nur von lokalen Administratoren, oder noch besser Domänen-Administratoren ausgeführt wird, erfolgt die Ausführung der Schadsoftware so sogar im höchstmöglichen System Kontext.

Von bösartigen Angriffen abgesehen, besteht bei falsch gesetzten Berechtigungen natürlich auch das Risiko von unbeabsichtigten Benutzerfehlern, die zu einem erhöhten Supportaufwand führen können, wenn Applikationen, oder das gesamte System auf Grund von verschobenen oder gelöschten Dateien nicht mehr gestartet werden kann oder Fehlermeldungen bringen.

Im Kontext des MITTRE Frameworks sind die hier beschriebenen Tests im Bereich der Technik T1574 (Hijack Executuon Flow) welche vor allem die Bereiche Persistence und Privilege Escalation betreffen.

5.4.2 Extraktion von Sicherheitsinformation

Einer der großen Punkte bei Audits, der von allen befragten Sicherheitsspezialisten erwähnt wurde, ist das Auslesen von Sicherheitsinformationen. Diese Angriffskategorie ermöglicht es auf einem System, auf dem eine initiale Kompromittierung stattgefunden hat, Daten für die weitere Verbreitung im Netzwerk oder Userdaten mit höheren Rechten auf dem System zu erhalten. Wenn es im Zuge eines solchen Angriffs gelingt, die Anmeldedaten eines Domänen Administrators zu erlangen, kann beispielsweise das gesamte Netzwerk als kompromittiert angesehen werden. Denn bei diesen Accounts handelt es sich quasi um die „Schlüssel zum Königreich“.

Die erste und wahrscheinlich schnellste Prüfung ist der Versuch die Sicherheitsdatenbank aus der Registry abzuziehen. Sofern der Angreifer im Kontext eines Users operiert der über lokale Administratorenrechte verfügt, gelingt es leicht die Daten aus dem Security Accounts Manager (SAM) Registry Schlüssel HKLM(HKEY_LOCAL_MACHINE)\SAM auszulesen und in ein File zu dumpen. In diesem Schlüssel sind die Sicherheitsinformationen der lokalen Accounts enthalten. Es werden somit also der Status des lokalen Users, dessen Name, die SID und der Hash des Passworts ausgelesen und können für spätere Angriffe mit einer Vielzahl von Werkzeugen auf den HASH genutzt werden.

Zusätzlich zu den Angriffen über die Registry kann ein Angriff über die SAM auch über das Dateisystem erfolgen, nachdem eine Version dieser Informationen auch unter C:\Windows\System32\config\SAM abgelegt ist. Wenn hier Dateiberechtigungen falsch gesetzt sind, wie es bei allen Windows 10 Systemen der Fall ist, wenn diese von einer früheren Build-Version auf 1809 und später gehoben wurden (Born, 2021b), kann die Datei leicht mit Hilfe des lokalen VSS Prozesses kopiert werden. Man kann also davon ausgehen, dass eine große Anzahl von derzeit aktiven Windows Systemen, hier eine leicht zu nutzende Schwachstelle hat.

Der nächste Angriff im Bereich eines Audits erfolgt auf das Local Security Authority Subsystem Service (LSASS), nachdem sich im Speicher dieses Prozesses, nicht nur die Daten der lokalen Benutzer befinden, sondern auch zwischengespeicherte Benutzernamen und Kennwörter von Domänen Benutzern zu finden sind. Für Angriffe auf diesen Prozess sind bereits erhöhte Rechte erforderlich, der angreifende Benutzer muss über das Recht „SEDebugPrivilege“ verfügen, um ein Speicherabbild (Dump) des Prozesses zu erstellen. Sollte es aber durch eine Schwachstelle oder durch lokale Administratorenberechtigungen gelingen diese Datei zu erzeugen, kann dieser, wie schon bei den SAM Daten exfiltriert werden und danach ein Offline-Angriff auf die darin enthaltenen Kennwort Hashes erfolgen.

Ein weiterer Test im Zuge eines Audits ist die Prüfung der für den Start der Services verwendeten Benutzer. Sollten auf dem System Services definiert sein, die im Kontext eines im Domänen Umfeldes hoch privilegierten Users gestartet werden, ist es für einen lokalen Angreifer, der über Administrationsrecht verfügt möglich, die im Service gespeicherten Kennwörter mithilfe der PowerShell oder einfachen Tools auszulesen und damit weitere Zugänge im Netzwerk zu erhalten. (Goude, 2012)

Eine Prüfung, die vor allem in Umgebungen stattfindet, bei denen bekannt ist, dass diese von frühen Versionen des Windows Betriebssystems immer wieder gehoben wurden, ist die Prüfung, ob Systeme zum Speichern von Kennwörtern noch LAN-Manager-Hash (LM-Hash) als Verfahren zum Speichern der Kennwort Hashes verwenden. Sollte dies der Fall sein, ist es auf Grund bekannter Schwachstellen im Hashingmechanismus trivial die gespeicherten Kennwörter auszulösen.

Ein letzter Punkt bei der große Einigkeit bestand, ist das Prüfen, ob sich im System noch Reste eines automatischen Deployments finden lassen. Vor allem in größeren Firmennetzen werden Systeme meist automatisch installiert, was die Möglichkeit mit sich bringt, die Werte einer unbeaufsichtigten Installation auszulesen, falls das Konfigurationsdokument nicht nach Abschluss der Installation aus dem System gelöscht wurde. Im schlechtesten Fall befinden sich in dieser Datei, neben dem Hash des lokalen Administratorenkennworts, auch noch der Hash und Benutzername eines privilegierten Domänen Benutzers, der dazu berechtigt ist, Systeme in die Domäne aufzunehmen.

Dieser Block an Tests zeigt zwar, dass es für eine Vielzahl der Angriffe bereits erforderlich ist einen erhöhten Userkontext zu haben, man darf aber nicht vergessen dass es zusätzlich zu den Wegen, auf denen ein Angriff im Normalfall erfolgen kann, immer auch die Möglichkeit gibt, dass eine nicht behobene Schwachstelle ein Umgehen dieser Erfordernisse ermöglicht. So hat es

beispielsweise fast 3 Jahre gedauert, bis das Problem der falschen Berechtigungen auf die SAM Database im Zuge eines Windows Build Updates aufgefallen ist.

Im Kontext des MITTRE Frameworks sind die hier beschriebenen Tests im Bereich des „Credential Acces“ verortet und bilden die Techniken T1555 (Credentials from Password Store) und T1003 (OS Credential Dumping).

5.4.3 Endpoint Protection Schwachstellen

Die Zahl der potenziellen Schwachstellen in der Konfiguration oder Funktion der End Point Protection, zeigt gut auf, wie sehr das erhöhte Gefühl der Sicherheit, durch die Installation einer solchen Lösung trügen kann. Von den seit Jahren in diversen Kreisen tobenden Diskussionen über die Nützlichkeit oder den Schlangen Öl Vorwürfen, wird bewusst Abstand genommen.

Eindeutig ist aber, dass keine derzeit am Markt existierende Lösung unfehlbar ist, und sich abgesehen von Hackergruppen, auch Schreiber von IT-Büchern mit dem Thema der Umgehung von Malware Erkennung befassen.

So kann es beispielsweise ausreichen, wenn die Schutzlösung so konfiguriert ist, dass beim Anstecken eines neuen USB-Gerätes, der Zugriff zu den darin enthaltenen Daten bereits ermöglicht wird, bevor die vorhandene Schutzkomponente einen Scan des Laufwerks vorgenommen hat. Wenn dazu noch die Möglichkeit gegeben ist, dass ein Autostart erfolgt, kann es passieren, dass bis ein Scan anschlagen kann, der Schaden bereits entstanden ist.

Im Bereich der Endpoint Protection ist sicher zu stellen, dass die Lösung über einen robusten Selbstschutz verfügt. Wenn es beispielsweise durch eine schlecht konfigurierte Berechtigung möglich ist, das Scanning Service im Rechtekontext des Nutzers zu beenden, besteht die Möglichkeit noch bevor ein sonst legitim wirkendes Skript damit beginnt erkennbare Schadsoftware auf das System zu laden, die Schutzlösung zu deaktivieren. Damit wäre das System ungeschützt und der nachgeladene Code kann nicht mehr geprüft werden. Sollte der Benutzer zu diesem Zeitpunkt nicht aufmerksam sein und auf die im besten Fall auftretende Warnung, dass der „Viren Schutz deaktiviert wurde“ reagieren, bleibt ohne zentrale Überwachungsstruktur diese Kompromittierung bis zum endgültigen Ziel der Schadsoftware unerkannt.

Ähnliches gilt bei gezielteren Angriffen, bei denen der Angreifer Informationen zur verwendeten Schutzsoftware besitzt, sofern der Angegriffene beispielsweise zum Schutz der Schutzkomponente, das Standardkennwort des Anbieters, statt einem eigenen Komplexen Kennwort verwendet. Auch in diesem Fall ist es für einen Angreifer, relativ leicht den Schutz des Zielsystems auszuhebeln und unbemerkt Schadsoftware zu hinterlassen.

Sofern es dem Angreifer gelingt, aus der Konfiguration der Schutzkomponente, die Liste der Ausnahmen zu extrahieren, ist es möglich, „Potenziell unerwünschtes Programm“ (PUP), wie übliche Hackingtools oder Malware in Verzeichnissen zu positionieren, die von Echtzeitscans ausgenommen sind. Grundlage hierfür ist, abgesehen vom Wissen über die bestehenden Ausnahmen, aber auch eine Schwachstelle wie sie im Punkt der Berechtigungsschwachstellen

beschrieben wurde. Sollte das Zusammenspiel mehrere Schwachstellen gelingen ist auch hier die Folge ein kompromittierter Client oder Server.

Im Bereich der Fehlkonfigurationen, die es Angreifern ermöglichen, auch bereits bekannte Malware einzusetzen, liegt es, wenn der User das Recht hat, erkannte Dateien oder Applikationen aus der Quarantäne zu entfernen und für die Ausführung freizugeben. Dies kann zum einen von Anwendern genutzt werden, die bewusst die schadhafte Funktion ausführen wollen, sei dies durch ein übernommenes Konto, oder den wirklichen Willen dem System Schaden zuzufügen. Zum anderen kann dieses Recht von Angreifern genutzt werden, um Benutzer ohne böse Absicht, davon zu überzeugen, dass es sich um einen Fehlalarm handelt und die Datei also bedenkenlos ausgeführt werden kann. Stellenweise kann es sich bei einem unbemerkt modifizierten Download sogar um den legitimen Hersteller der Software handeln. Das hat der Fall eines kompromittierten Treiberdownloads für einen Chipkartenleser gezeigt.

Auf Grund der sich immer weiterentwickelnden Bedrohungssituationen, sollte sichergestellt sein, dass die verwendete Sicherheitslösung in der Lage ist, auch Angriffe zu erkennen, die ohne das Ablegen von lokalen Dateien erfolgen, also rein im Arbeitsspeicher passieren.

Es sollte sichergestellt werden, dass sowohl die verwendete Endpoint Lösung, als auch die auf dem System verwendeten Applikationen das Windows Antimalware Scan Interface kurz AMSI nutzen, so dass auch über Programme exekutierte Skriptkomponenten und Makros, laufend durch den Virenscanner überwacht werden können. (Carius, 2022)

Als letzter Punkt im Bereich der möglichen Schwachstellen ist noch die Resilienz bei der Erkennung von Malware zu nennen, die mit Hilfe von statischen oder dynamischen Mittel verschlüsselt wurde um so den eigenen Code zu „Obfuscaten“ und eine Erkennung über statische Definitionen zu verhindern. Hier muss die verwendete Sicherheitslösung ebenfalls wieder in der Lage sein, den Speicher so effektiv zu überwachen, dass die Ausführung des Schadcodes erkannt und gestoppt werden kann, sobald die Entschlüsselung abgeschlossen wurde. Im besten Fall wird aber bereits die verschleierte Datei selbst auf Grund der Struktur als potenziell bösartig eingestuft und der Zugriff, beziehungsweise die Ausführung dieser, blockiert. (Jeyashankar, 2022)

Die Anzahl der zu beachtenden Punkte, die hier betrachtet wurden, zeigt schon, dass es auch im Bereich der Endpoint Protection zu wenig ist, sich nach der Installation darauf zu verlassen, dass diese ohne weiteres Zutun einen optimalen Schutz bietet. Einige Punkte zeigen auch, warum aktuell die Entwicklung im Bereich der Sicherheitssoftware, immer mehr weg von der statischen Erkennung von Schadsoftware über Definitionsdateien, hin zu dynamischer Verhaltensanalyse geht.

Im Kontext des MITTRE Frameworks sind die hier beschriebenen Tests im Bereich der „Defense Evation“ zu sehen und bilden vor allem die Techniken T1027 (Obfuscated Files or Information) und T1211 (Exploitation for defense Evation).

5.4.4 Erreichen höherer Berechtigungen

Wie im Kapitel 4.5 schon beschrieben, stellen Methoden zum Ausweiten der Benutzerberechtigungen einen Teil vieler Angriffe dar. Das Ziel in den meisten Fällen ist, die angreifende Software im Kontext des Systems oder eines administrativen Nutzers auszuführen. Im Folgenden werden einige Techniken betrachtet, die es ermöglichen, Software oder zumindest Module in einem Berechtigungskontext auszuführen, der höher ist als der des aktuell angemeldeten Benutzers. Einige der möglichen Schwachstellen erfordern es, dass die am Anfang dieses Kapitels beschriebenen Prinzipien des „Least Priviledge“ bei Berechtigungen nicht eingehalten werden.

So kann etwa, wenn eine Applikation in der Entwicklung unsauber konfigurierte Pfade verwendet, die nicht unter Hochkomma gesetzt sind, dazu gebracht werden, Daten und vor allem DLLs aus Ordnern zu laden, die nur einem Teil des Pfades entsprechen, sofern in einem nicht mit Hochkomma gesicherten Pfad ein Blank oder Leerzeichen vorkommt. Durch diese Methode kann es gelingen, einem automatisch in einem höheren Kontext gestarteten Programm schadhaften Code unterzujubeln und diesen automatisch starten zu lassen.

Ein weiterer Bereich sind im Windows Umfeld Services, vor allem wenn diese im Kontext der Benutzer „SYSTEM" oder „NETZWERKDIENST“ laufen. Nachdem beide Dienstidentifikationen quasi über vollwertige Administratorenberechtigungen auf dem System verfügen, ist die Manipulation eines Service in diesem Kontext besonders lukrativ für einen Angreifer.

Hier gibt es zwei Ansätze, die bei allen befragen in den jeweiligen Audits überprüft werden. Zum einen, ist es möglich durch falsch gesetzte Berechtigungen im Filesystem die ausführbare Datei des Service auszutauschen und durch eine manipulierte Applikation oder Malware zu ersetzen, so, dass diese automatisch gestartet wird. Zum anderen, die Möglichkeit durch falsch gesetzte Berechtigungen auf Ebene des Service selbst den Pfad zur ausführbaren Datei des Service zu manipulieren. Dies ermöglicht es, die gewünschte bösartige Datei, in einem beliebigen Ordner im Dateisystem abzulegen, in dem der Angreifer Schreibrechte hat. Es erfordert also keine Fehlkonfiguration der Dateisystemberechtigungen.

Was für Services gilt, gilt im Windows Umfeld ebenfalls für Programme im Autostart.

Auch hier bieten sich wieder zwei sehr ähnliche Methoden zur Aushebelung der Systemsicherheit. Zum einen, das Austauschen der ausführbaren Datei die in der Registy entweder unter dem User Pfad HKCU(HKEY_CURRENT_USER)\SOFTWARE\Microsoft\Windows\CurrentVersion\Run oder dem systemweiten Pfad HKLM(HKEY_LOCAL_MACHINE)\SOFTWARE\Microsoft\Windows\CurrentVersion\Run referenziert wird. Hierfür werden wieder Rechte auf dem Zielordner benötigt.

Wenn jedoch die Berechtigungen auf die Schlüssel in der Registry falsch gesetzt sind, kann es gelingen nicht nur einen Autostart Eintrag für den gerade angemeldeten User, sondern auf Ebene des Systemkontexts für alle User zu setzen. Nachdem es grundsätzlich für Benutzer erlaubt ist,

sich selbst Autostart Einträge zu erstellen, gilt es hier beim Erstellen der Sicherheitsrichtlinien zu entscheiden, ob dies ein gewolltes oder benötigtes Verhalten ist, oder ob hier Einschränkungen zu treffen sind.

Eine Anfälligkeit die sich vor allem auf Systemen finden lässt, die Legacy 32-Bit Applikationen verwenden, betrifft die Settings des „Windows Installers“. Dieser kann durch das Setzen des Eintrags „AlwaysInstallElevated“ in der Registry unter den Schlüsseln HKEY_CURRENT_USER\Software\Policies\Microsoft\Windows\Installer
oder
HKEY_LOCAL_MACHINE\Software\Policies\Microsoft\Windows\Installer
dazu gebracht werden, Software auch unter administrativen Rechten zu installieren, auch wenn der betreffende User, der die Installation ausführt, diese Rechte eigentlich nicht besitzt.

Abgesehen von Legacy Installationen, die vergessen wurden, kann natürlich auch dieser Schlüssel bösartig gesetzt werden, um Schadsoftware auf diesem Weg einzuschleusen. (Bachelor, Sharkey & Satran, 2021)

Die folgenden Punkte zeigen nur Relevanz, sofern die im späteren Verlauf noch betrachteten „Software Restriktionen“ zum Einsatz kommen. Nachdem diese aber als Schutzkomponente zum Härten eines Windowssystems betrachtet werden, werden hier auch deren potentielle Schwachstellen betrachtet.

Bei den gemeinsam genannten Punkten handelt es sich in den meisten Fällen um Konfigurationsfehler dieser Komponente.

Abhängig von der Einstellung kommt dem Whitelisting eine entscheidende Rolle zu, nachdem dieses, abhängig vom Detailgrad, entscheidet, ob das Ausführen von Applikationen auf Basis des Pfads des Herausgebers, der Applikation, der Version oder sogar dem Datei Hash erlaubt wird. Ein zu freizügiges Whitelisting, eröffnet hier Angreifern wieder die Möglichkeit unerwünschte Software einzuschleusen und auszuführen. Ein klassisches Beispiel wäre hier das Erlauben der Ausführung von Programmen aus dem Appdata Verzeichnis des Users, da dieser mehrere legitime Applikationen aus diesem Verzeichnis startet. Nachdem ein User aber Vollzugriff auf diesen Ordner hat, wäre dies eine eindeutige Sicherheitslücke.

Ein, den Aussagen nach, am häufigsten gesehener Fehler ist die Kombination aus korrekt gesetzten Regeln, aber dem inkorrekten Setzen des Security Levels. Wird dieses auf Default vergessen, können alle anderen Einstellungen korrekt sein, das Schutzsystem bleibt aber inaktiv und hat so keinen Effekt für die Systemsicherheit.

Wie auch schon bei allen zuvor genannten Kategorien ist auch hier eine der potenziellen Schwachstellen, die es zu prüfen gilt, ob der „Least Priviledge“ Grundsatz bei allen Ordnern in der Whitelist eingehalten wird. Nachdem auch hier wieder gilt, wenn es einem Angreifer möglich ist, Software in einem Ordner zu platzieren, der es erlaubt, dass diese in Folge trotz Software Restrictions ausgeführt wird, ist der Schutz erfolgreich ausgehebelt.

Die letzten beiden häufig genannten Checks im Zuge eines Audits, sind im Kontext dieser Arbeit etwas als Outlier zu betrachten, nachdem es ohne weiterentwickelten Schutz im Bereich Endpoint Detection und Response (EDR) nur schwer möglich ist, auf solche Angriffe zu reagieren. Hier wäre zum einen, das Umgehen von Software Restrictions durch das Laden und Aushöhlen eines legitimen Prozesses aus einem erlaubten Pfad. Hierbei wird der Ladeprozess der Applikation durch eine Suspend Aufruf gestoppt, der noch zu ladende Programmcode aus dem Arbeitsspeicher entladen, und durch den zu ladenden manipulierten Code ersetzt., Danach wird der Suspend Zustand aufgehoben und die Applikation lädt den manipulierten Code und führt diesen aus. (Chai, 2020)

Zum anderen gibt es die Möglichkeit, legitime, standardmäßig mit Windows installierte und damit mit hohem Vertrauenskontext versehene Applikationen zu nutzen, um Malware zu installieren, oder diese nach einer erfolgreichen Kompromittierung für Persistenz und laterale Verbreitung zu nutzen. Angriffe dieser Art werden als “Living of the Land” Binarys (LOLBins) bezeichnet. (Yiftach, 2020)

Diesem ganzen Angriffskomplex widmet das MITRE ATT&CK ein eigenes Kapitel unter dem Themenkomplex „Priviledge Escalation“. Die hier beschriebenen Checks fallen besonders in die Angriffskategorien T1068 „Exploitation for Priviledge Escalation“, T1574 „Hijack execution Flow“ und T1055 „Process Injection“.

5.4.5 Mangelnde System Härtung

Die im Folgenden betrachteten Punkte aus den Audit Plänen befassen sich hauptsächlich mit der Prüfung der Missbrauchsmöglichkeiten der im vorhergegangenen Abschnitt erwähnten Tools für einen „Living of The Land“ Angriff. Nachdem Angriffe, die mit Hilfe dieser Komponenten durchgeführt wurden, vor allem für klassische Sicherheitslösungen nur sehr schwer bis gar nicht erkennbar sind, hat dieser Bereich ein besonderes Gewicht bei der Absicherung von Systemen ohne Mehrkosten zu verursachen.

Die ersten Prüfungen befassen sich mit einem der umfangreichsten Tools im Arsenal des Windows Betriebssystems, der PowerShell. Diese eignet sich in Folge des enormen Funktionsumfangs besonders gut für den Einsatz als Angriffswerkzeug insbesondere dann, wenn es möglich ist, entweder durch Downgrade auf Version 2 oder Manipulieren der Umgebungsvariablen die Schutzoption „Eingeschränkter Sprachmodus“ zu umgehen.

Ebenso kritisch ist es, wenn es gelingt die Ausführungsrichtlinie dahingehend anzupassen, dass unsignierter oder Remote Code aus einem Skriptfile angepasst werden kann.

Bevor bei beiden Funktionen jedoch geprüft wird, ob es technisch möglich ist die Methoden zu umgehen, wird vorab geprüft, ob diese Einstellungen überhaupt gesetzt wurden.

Eine weitere Prüfung ist, ob die Möglichkeit besteht, Zugriff zu Laufwerken zu erlangen, auf die der Zugriff mit Hilfe von Gruppenrichtlinien oder anderen Methoden gesperrt wurde. Hier kann es zum Teil schon genügen, wenn ein Angreifer die Möglichkeit hat, eine Kommandozeile zu öffnen.

Eine weitere Prüfung, die auf jeden Fall bei allen befragten Experten mit im Audit Portfolio ist, ist die Überprüfung, welche der Tools, die für einen „Living of the Land“ Angriff genutzt werden können, für den Kontext des aktuellen Users genutzt werden können.

Sofern lokal oder remote eine interaktive Anmeldung auf dem System erfolgen kann, fällt auch die Prüfung des Zugriffes auf diverse Kontextmenüs in den Scope der meisten Prüfungen, ebenso wie, sofern es Sperren gibt, ob sich diese umgehen lassen.

Ein letzter Check, der in die Klasse der lokalen Systemchecks fällt, ist die Prüfung, die die Einschränkungen für die Verwendung von USB Anschlüssen betrifft. Hier kann es sich zum einen um die Prüfung handeln, ob ein Massenspeicher eingebunden und zugänglich gemacht werden kannZum anderen gibt es auch noch die Möglichkeit das System mittels BadUSB anzugreifen, sofern das System es erlaubt, dass sich ein Gerät als Human Interface Device (HID) melden darf, was das automatisierte Ausführen von Code über eine emulierte Tastatur erlaubt.

Im Kontext des MITTRE Frameworks sind die hier beschriebenen Tests im Bereich der „Defense Evation“ zu sehen und bilden vor allem die Technik T1562 „Impair Defences“. Zusätzlich wird der Bereich Execution mit T1059 „Command and Script Interpreter“ genutzt.

5.4.6 MangeIndes Patchlevel

Ein Punkt, den so gut wie jeder der ein Windows System nutzt kennt, unabhängig davon, ob Anwender oder Administrator, ist die Thematik der Windows Updates.

Aktuelle Systeme zwingen Anwender und Administratoren mittlerweile zwar grundsätzlich schon dazu, die monatlich erscheinenden Updates zeitnah einzuspielen, um so bekanntgewordene Lücken möglichst zügig zu schließen.

In der Vergangenheit haben einige schwere Schwachstellen bereits gut gezeigt, wie leicht verwundbar Systeme, mit nicht aktuellem Patchlevel sind. Namen wie “Eternal Blue” oder “Hafnium” haben es in den vergangenen Jahren geschafft, medial große Aufmerksamkeit zu erhalten.

Leider hat es Microsoft in diesem Bereich aber auch schon öfter geschafft zu beweisen, dass die Qualität der herausgegebenen Updates nicht immer den Erwartungen der Enduser und Administratoren entsprechen. Was dazu führt, dass vor allem im Firmenumfeld eine Verteilung in Ringen passieren sollte, um nicht mit einem fehlerhaften Update über Nacht alle Systeme zu beeinträchtigen. Dennoch sollte dafür gesorgt werden, das Updates zeitnahe installiert werden und sofern nicht spezifische Gründe für das Auslassen eines Updates bestehen, nie mehr als eine Generation an Updates auf den Systemen offen ist. Bei besonders schweren Schwachstellen, insbesonders wenn diese bereits aktiv ausgenützt werden, sollte der Schutz der Infrastruktur gegenüber potenziellen Problemen in Folge des Updates priorisiert werden. Sofern es durch das Update nicht ebenfalls zu einem Komplettausfall kommt.

Abgesehen von Betriebssystemupdates dürfen auch Updates anderer Hersteller nicht ganz außer Acht gelassen werden. Wenn man die Zahl der jährlich gefundenen Schwachstellen, alleine aus der CVE Datenbank betrachtet, wird relativ schnell klar, dass das Patchen von IT Systemen,

wenn diese aktuell gehalten werden sollen, ein massives Unterfangen ist, sofern mehr als nur das Betriebssystem auf einem Computer installiert ist. Eine Lücke, die dies im vergangenen Jahr besonders schön gezeigt hat, war „Log4J“, eine Schwachstelle in einer häufig genutzten Java Bibliothek, wodurch gleichzeitig alle Softwareprodukte, die diese Bibliothek nutzen, die selbe Sicherheitslücke aufweisen und mit passenden Updates versehen werden müssen.

Im Kontext des MITTRE Frameworks sind die hier beschriebenen Tests im Bereich des „Initial Access“ und hier hauptsächlich die Technik T1189 „Drive by Compromise“ und bis zu einem gewissen Punkt den Bereich Execution mit der Technik T1204 „User Execution“.

6 SYSTEMEIGENE SCHUTZMECHANISMEN

Im Folgenden werden nun die Verteidigungsmechanismen betrachtet, die es mit Ausnahme einer Komponente, Built-in in den betrachteten Systemen gibt.

Auch, wenn diese Mechanismen von Haus aus in den Systemen vorhanden sind, sind die Default Settings in den meisten Fällen darauf ausgelegt, es zwar den häufigsten Bedrohungen zu erschweren ein System zu kompromittieren, aber dabei den User so wenig wie möglich zu irritieren. Nachdem Anwender dazu tendieren, sofern Sie dazu in der Lage sind, Sicherheitsmechanismen unabhängig von der Sinnhaftigkeit, als störend zu empfinden, und sie zu deaktivieren, sofern sie sie bei der normalen Arbeit behindern, oder diese auch nur gefühlt verlangsamen. Ein gutes Beispiel hierfür war die Einführung der UAC in Windows Vista, wo es kurz nach Einführung bereits unzählige Artikel gab, wie der Dialog unterdrückt werden kann. Etwas, das mit den gelernten Lektionen und Verbesserungen am System der UAC, bei späteren Versionen nicht mehr nötig war.

Dieses Kapitel nimmt damit auch die gefühlte Unbequemlichkeit, also Kriterium für die „optimalen Einstellungen“ auf. Nachdem die bestmöglichen Sicherheitseinstellungen nicht helfen, wenn das System dadurch für den Anwender nicht mehr nutzbar wird.

6.1 UAC – User Account Control

Bei der User Account Control handelt es sich um eine Sicherheitsfunktion, die mit Windows Vista und Server 2008 in der Microsoft Umgebung eingeführt wurde. Mit der Einführung wurde das Verhalten der Microsoft Betriebssysteme in Hinsicht auf die Verwendung von Benutzerrechten so verändert, dass sich dieses mehr an Linux und Unix Systemen orientiert. Sie soll damit zusammen mit dem automatischen Deaktivieren, des lokalen „Well Known“ Users „Administrator“ während der Installation einen gewissen Schutz gegen die Ausnutzung von privilegierten Benutzeraccounts bieten.

6.1.1 Funktion

Die Implementation der UAC ermöglicht es unter Windows Betriebssystemen, einen Benutzeraccount zu verwenden. Das Mitglied der Gruppe Administratoren ist zu verwenden und Applikationen, dennoch bei einem „normalen Start“ nur mit dem Rechtetoken eines normalen Benutzers zu starten. Sollte es aber notwendig werden, dass ein Programm mit administrativen Rechten gestartet werden muss oder soll, kann der User dies selbst entscheiden, oder das Programm kann die Erhöhung der Rechte direkt anfordern.

Um die unterschiedlichen Rechte zu veranschaulichen, sollen hier zwei Screenshots der Ausgabe des Befehles „whoami“ dienen. Auf der folgenden Abbildung einmal ohne den Start als Administrator.

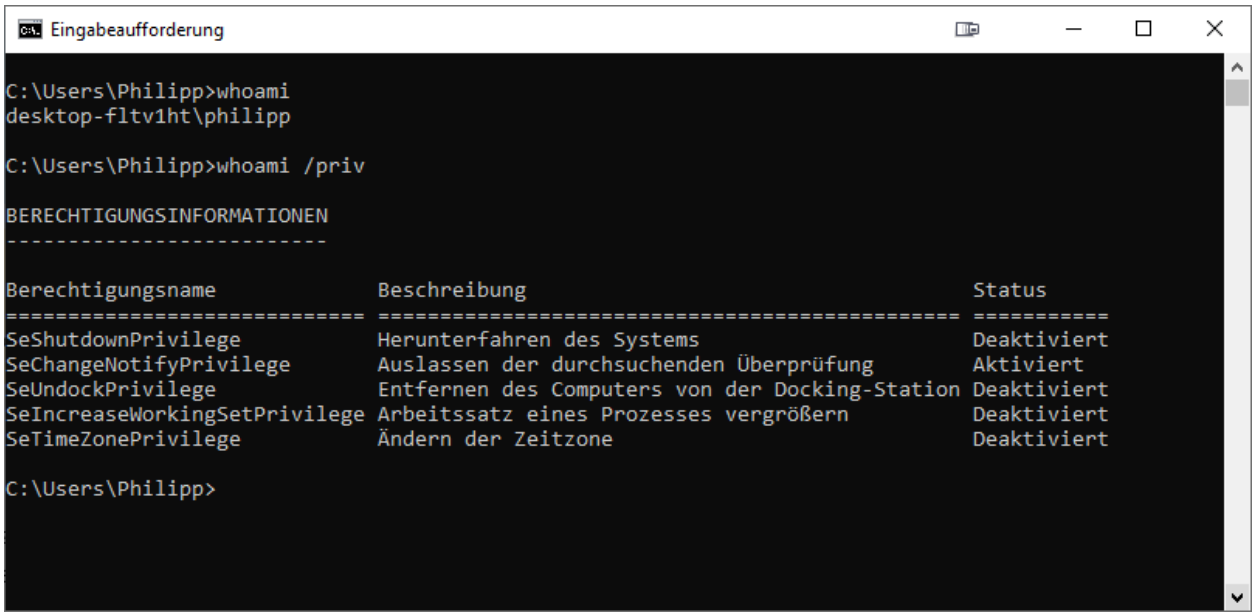

Abbildung 6-1

Und auf der folgenden Abbildung einmal, nachdem die Eingabeaufforderung „als Administrator“ gestartet wurde.

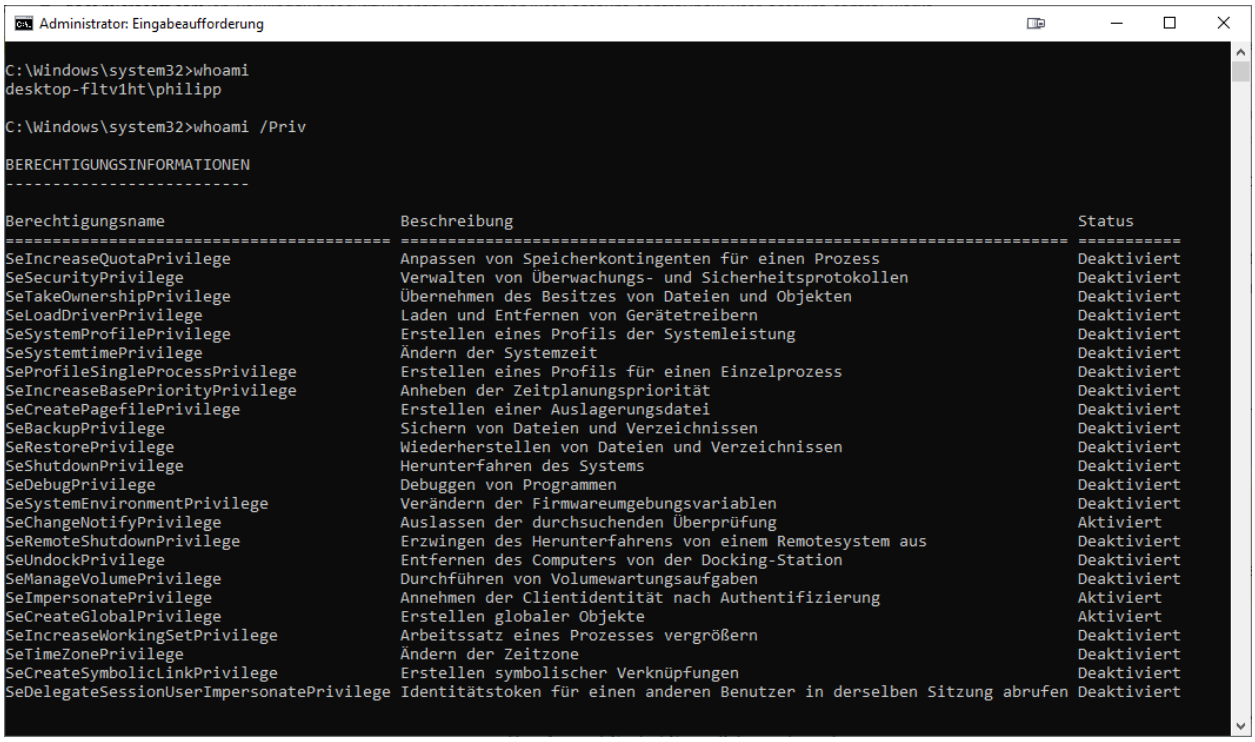

Abbildung 6-2

Es zeigt sich hier eindeutig, dass derselbe Benutzer, nur auf Grund des Startkontextes, wesentlich mehr Rechte auf dem System hat.

Diese Trennung des Benutzerkontexts soll verhindern, dass ein versehentliches Ausführen einer bösartigen Applikation nur im Kontext eines Standard-Users erfolgt und nicht im Kontext eines Administrators und somit keine weitreichenden Änderungen am System durchgeführt werden.

6.1.2 Optimale Setting

Die Art und Weise wie diese Funktion implementiert und verwendet wird, ist essenziell für die Userakzeptanz und die Nützlichkeit. Bei der Einführung unter Windows Vista wurde die Funktion zwar von Securtity Professionals positiv wahrgenommen, von normalen Anwendern, wurde die Funktion aber auf Grund der Häufigkeit der Dialogabfragen und auf Grund von Problemen mit alter Software, eher als negativ und störend empfunden. Somit fanden sich vor allem im privaten Bereich schnell Anleitungen und Tools im Internet, die das Verhalten der UAC deaktivierten und somit den Schutz ausgehebelt haben. In den späteren Versionen hat Microsoft hier deutlich dazu gelernt und hat das Verhalten stark verbessert., Auch der Wegfall der Kompatibilität mit Programmen, die nicht dem neuen Schema der Ordnerverwendung entsprechen hat hier sicher geholfen.
Daraus ergeben sich für aktuelle Umgebungen die folgenden dmpfohlenen Settings.

- Normale User Umgebung
    - Administrativer User:
        - Bestätigungsdialog auf nicht sicherem Desktop.
    - Nicht administrativer User
        - Abfrage zur Anmeldung als administrativer User
- Bereich mit erhöhten Sicherheitsanforderungen
    - Administrativer User
        - Sollte grundsätzlich nicht zum Arbeiten verwendet werden
        - Abfrage von Benutzer und Kennwort auf einem Secure Screen
    - Nicht administrativer User
        - Abfrage von Benutzer und Kennwort auf einem Secure Screen

Ziel dieser Settings ist es, möglichst das beste Gleichgewicht aus der Sicherheit der Funktion zu ziehen als auch den Benutzer so wenig wie möglich in seiner Arbeit zu behindern. Es sei zur Vollständigkeit erwähnt, dass diese Einstellungen unter der Annahme empfohlen werden, dass es sich bei Personen mit Administratorenberechtigungen um fachkundige Anwender handelt.

6.1.3 Einfluss auf die beschriebenen Angriffe

Während die technische Schutzfunktion dieses Schutzmechanismus grundsätzlich als sicher anzusehen ist, fällt hier besonders der menschliche Faktor ins Gewicht.
Ein Benutzer der administrative Rechte besitzt und nicht darauf achtet oder nicht das Verständnis besitzt, um zu beurteilen, ob die Situation, in der die Abfrage zur Bestätigung von erhöhten Rechten angefordert wird, diese auch rechtfertigt, kann hier durch einen Klick die Sicherheit umgehen und das System nachhaltig schädigen. Sollte es also notwendig sein, Benutzern auf lokalen Systemen administrative Rechte zu geben, beispielsweise um regelmäßige Updates von Benutzerapplikationen durchzuführen, darf die Schulung der Benutzer nicht außer Acht gelassen werden.

Generell kann die UAC als erste Verteidigungsstufe gegen das direkte Ausführen von Schadsoftware gesehen werden. Sofern Techniken eingesetzt werden, die Rechte anderer Prozesse nutzen oder schlechte Programmierung genutzt wird, um DLLs von regulär mit erhöhten Rechten ausgeführten Applikationen auszutauschen, hat die UAC wenig Möglichkeiten Schaden abzuwenden.

6.2 DEP – Data Execution Prevention

Data Execution Prevention ist eines der “älteren“ internen Schutzsysteme, in Microsoft Betriebssystemen, welches mit Windows XP Service Pack 2 bzw. Server 2003 Service Pack 1 eingeführt wurde. Dieser Schutzmechanismus soll verhindern, dass Code der in nicht für die Ausführung vorgesehene Bereiche geladen wurde, ausgeführt werden kann.

6.2.1 Funktion

Sofern DEP für einen Prozess aktiviert ist, wird der Stack eines Programms, bzw. die dahinter zu Grunde liegenden Speicherseiten von der für sie zuständigen Applikation, beziehungsweise dem Betriebssystem, mit einem NX-Bit (no execute) versehen. Wenn nun diese mit NX-Bit markierten Speicherseiten auf Grund eines Programmfehlers, oder durch gezielte Manipulation durch ein bösartiges Programm zur Ausführung an die CPU übergeben werden, erkennt die CPU diesen Umstand, und löst statt der Ausführung des hinterlegten Codes in diesen Seiten, ein Interrupt aus, welcher in Folge auf Betriebssystemebene dazu führt, dass der aufrufende Prozess beendet wird. Dies kann, sofern es sich um einen Systemprozess handelt, bis hin zum Anhalten des gesamten Systems führen und einen STOP Error, unter Windows auch als Bluescreen bekannt, auslösen. (Andreas Kroschel, 2011)

6.2.2 Optimales Setting

Im Windows Umfeld stehen vier Setting Varianten zur Verfügung, „Always on“, „Always off“, „Opt In“ und „Opt Out“.

Hier ist die optimale Einstellung stark von den verwendeten Anwendungen abhängig, nachdem einige alte Applikationen. vor allem wenn sie Verschlüsselungen verwenden, die Ausführung im RAM benötigen und somit nur in den Modis „Always Off“ und „Opt In“ lauffähig sind. Unter Umständen kann eine solche inkompatible Applikation auch im Modus „Opt Out“ funktionieren, wenn dieses „Opt Out“ per „Application Compatibility Toolkit“ per Shim ausgenommen wird.

Je nach Anzahl der betroffenen Applikationen ist der Modus „Opt Out“ der empfohlene, dieser ist bei aktuellen Serverbetriebssystemen von Microsoft auch der System Default.

6.2.3 Einfluss auf die beschriebenen Angriffe

Bei „Data Execution Prevention“ handelt es sich, wie schon erwähnt, um eine der älteren und vor allem grundlegenden Schutztechniken im einem Microsoft Betriebssystem.

Der Schutz erstreckt sich hier also auf das Verhindern, dass Speicherseiten von Exploits oder Schadsoftware Speicherseiten übernehmen können und diese dazu nutzen um Schadcode im Kontext der übernommenen Applikation oder im schlimmsten Fall im Systemkontext ausführen können. Je nach SystemuUmgebung und gewählter Einstellung handelt es sich dabei um ein gutes aber in manchen Bereichen löchriges Schutzmodul. (Tudor, 2021)

6.3 ASLR – Address space layout randomization

„Adress space Layout Randomization“ ist ein Sicherheitsfeature, das Code Injections über Speicherseiten verhindern soll, indem es das Layout des Arbeitsspeichers randomisiert. Die betroffene Applikation muss das Feature jedoch unterstützen, da es ohne die passende Unterstützung zu Problemen kommen kann, wenn explizite Adressierungen verwendet werden, welche durch die zufällige Zuweisung der Speicherseiten nicht mehr zutreffend sind. (Microsoft, 2022e)

6.3.1 Funktion

Die Schutzfunktion von ASLR besteht darin, dass die Daten aller teilnehmenden Prozesse im Arbeitsspeicher bei jedem Start anders positioniert werden. So ist es nicht möglich die Speicherbereiche von beispielsweise Systemapplikationen vorherzusehen.

Hierdurch wird es beispielsweise erschwert, mittels Heap Spraying Angriff, in Memory Daten zu überschreiben und so den Prozess zur Verarbeitung zu bringen.

Unter Windows 10 kann die Verwendung eines Trusted Platform Modules (TPM) die Entropie noch weiter erhöhen.

Die folgende Abbildung von der Microsoft Produktseite veranschaulicht die Funktionsweise am übersichtlichsten.

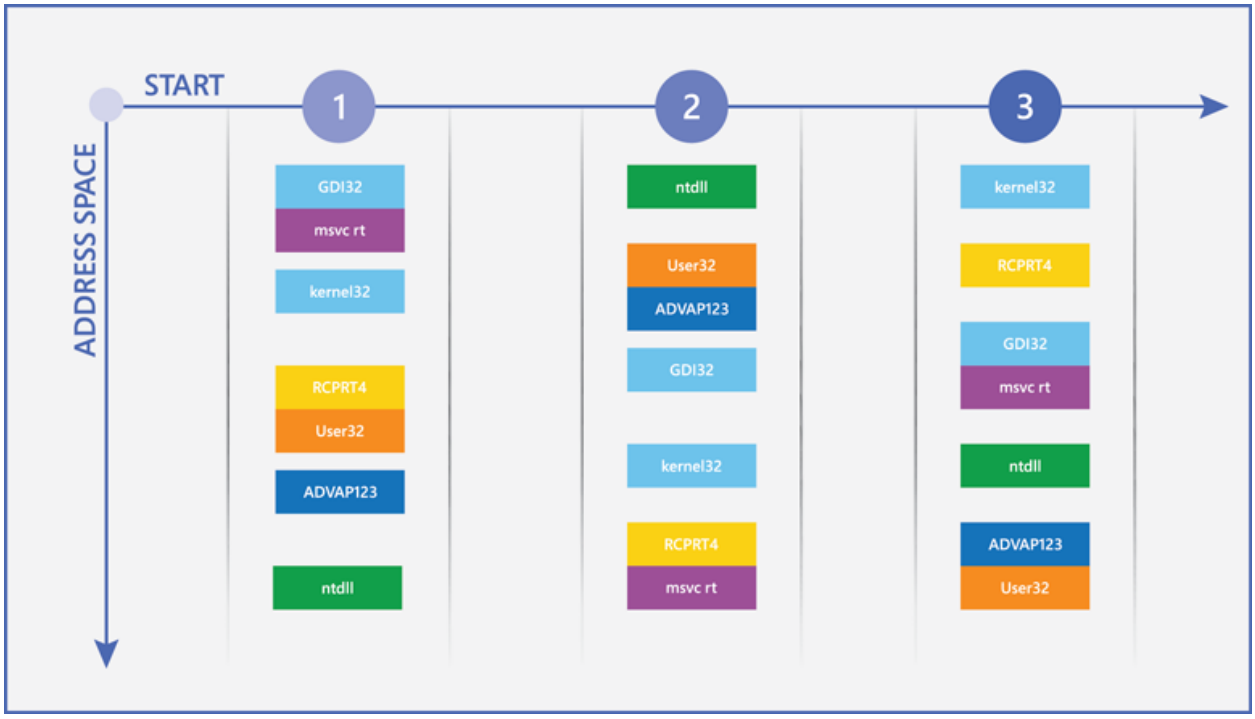

Abbildung 6-3 (Microsoft, 2022d)

6.3.2 Optimales Setting

Sofern alle auf dem System installierten Anwendungen, mit der Funktion umgehen können, empfiehlt es sich die Option, „zufällige Anordnung für Images erzwingen“ von ihrem Standardwert „Deaktiviert“ auf „Aktiviert“ umzustellen. Da sonst nur Applikationen, die diese Funktion bewusst aufrufen im Schutz integriert sind.

Die beiden weiteren Einstellungsmöglichkeiten „Speicherbelegung zufällig anordnen“ und „ASLR mit hoher Entropie“ sind bereits im Default Zustand aktiviert und sollten auch auf diesen Settings belassen werden.

6.3.3 Einfluss auf die beschriebenen Angriffe

Der größte Einfluss hier zeigt sich bei der Abwehr von Exploits, die auf die vorhersehbare Adressierung von Speicherseiten eines Programmes abzielen. Ein solcher Angriff schlägt auf einem System mit aktivierter ASLR fehl und kann im schlimmsten Fall noch dafür sorgen, dass das System mit einem Bluescreen abstürzt, weil versucht wird einen Speicherwert zu ändern, der geschützt ist.

6.4 Bitlocker

Bei Bitlocker handelt es sich um die seit Windows Vista in Microsoft integrierte Verschlüsselungstechnologie. Diese kann sowohl lokale Datenträger, so wie als Bitlocker-to-Go auch mobile Datenträger verschlüsseln.

6.4.1 Funktion

Bei Bitlocker handelt es sich um eine Verschlüsselungstechnik die AES mit einer Schlüssellänge von 128-256 Bit mit CBC arbeitet, seit Windows 10 Version 1511 wird statt CBC XTS verwendet. Bei Bitlocker liegt die Verschlüsselung der Daten nur in Ruhe vor, im laufenden Betrieb sind die Datenträger entschlüsselt und damit offen lesbar.

Für die Entschlüsselung im Startprozess liegen drei normale Optionen sowie eine Notfall Option vor. (Microsoft, 2022b)

Transparent Mode: Die Daten sind verschlüsselt, aber beim Starten des Systems ist keine Eingabe von PIN oder Pass Code erforderlich, da direkt auf die Entschlüsselungskeys zugegriffen wird, die im TPM gespeichert sind.

Pin-Mode: Die Daten auf der Platte sind verschlüsselt, zum Starten des Geräts und zum Zugreifen auf die Daten ist die Freigabe mittels PIN oder Kennwort erforderlich.

Token Mode: Die Daten sind verschlüsselt, zum Starten des Geräts und dem Entschlüsseln der Daten ist das Anstecken eines bestimmten Datenträgers erforderlich, auf dem die Entschlüsselungskeys gespeichert sind.

Notfall Modus: Bei der Verschlüsselung des Laufwerks wird ein Recoveryschlüssel erstellt, welcher bei korrekter Einstellung bei Geräten mit Active Directory Anbindung im Computerkonto gespeichert wird. Zusätzlich muss er als Datei, oder als physischer Ausdruck angelegt werden, um das Setup abschließen zu können. Mithilfe dieses Schlüssels kann ein Laufwerk freigeschaltet werden, wenn keine der oben genannten Standardmethoden möglich ist.

6.4.2 Optimales Setting

Bei den Einstellungen gibt es eine klare Abwägung zwischen Komfort und Sicherheit bei der Verwendung. Um sich gegen Szenarien zu verteidigen bei denen Datenträger ohne die dazugehörige PC-Hardware entwendet werden, ist es ausreichend Bitlocker im Transparent Mode zu verwenden, da in diesem Fall der zum Entschlüsseln notwendige TPM Chip nicht mehr vorhanden ist. Um sicher zu stellen, dass auch im Fall eines Diebstahls des gesamten Systems, die Daten nicht in falsche Hände geraten, stellt sich die Variante den Zugriff mit einem PIN oder Pass Code zu schützen als die beste heraus, da eine Entschlüsselung via Token Device im Fall, dass der Token zum Moment des Diebstahls angesteckt ist, ebenfalls keinen zusätzlichen Schutz

bietet. Sofern das Gerät in einer Windows Domäne betrieben wird, ist es auf jeden Fall zu empfehlen den Recovery Schlüssel im Active Directory automatisch ablegen zu lassen. Zusätzlich sollte das Recovery Dokument getrennt von der Hardware an einem sicheren Ort aufbewahrt werden. Digital würde sich hier ein Netzlaufwerk mit starken Zugriffsschutz anbieten. In physikalischer Form empfiehlt sich das Ablegen, weitab des Geräts, oder zentral in der IT-Abteilung. (Microsoft, 2022b, 2022c)

6.4.3 Einfluss auf die beschriebenen Angriffe

Während der Einfluss auf die meisten Angriffe zu vernachlässigen ist, verhindert ein korrekt verschlüsseltes Laufwerk im Fall des Verlusts oder Diebstahls des Gerätes in den meisten Fällen, dass das Gerät für Folgeangriffe oder zur Informationsgewinnung genutzt werden kann. So wird beispielsweise dadurch verhindert, dass eine DSGVO Meldung gemacht werden muss, sollten sich Kundendaten auf dem Gerät befinden. Zum anderen ist es durchaus möglich, dass aus einem unverschlüsselten Gerät mit Hilfe von Tools wie Mimikatz gespeicherte Zugangsdaten ausgelesen werden, die je nach Reaktionszeit, bei der Erkennung des Verlustes für den Zugang zu Clouddiensten oder dem Unternehmensnetzwerk genutzt werden können.

6.5 Software Restrictions

Eine der wahrscheinlich restriktivsten Optionen, die im Betriebssystem standardmäßig zur Verfügung stehen, um das System vor der Ausführung ungewollter Programme zu unterbinden.

Beim Umfang der zur Verfügung stehenden Möglichkeiten muss man zwischen Home/Pro und Enterprise Versionen des Betriebssystems unterscheiden. Für den vollen Funktionsumfang ist es also erforderlich die Windows Client Versionen unter Software Assurance zu haben. Ohne diese zusätzliche Lizenz ist der Einstellungsumfang massiv reduziert. Nachdem die Enterpriese Lizenz mittlerweile auch über ein Microsoft Online E5 Abonnement erhältlich ist, und keine 500 Lizenzen mehr erforderlich sind, um ein Enterprise Agreement mit Microsoft abzuschließen, werden in der Folge beide Optionen betrachtet. (Gerend, Ross, Parente & Poggemeyer, 2021)

6.5.1 Funktion

Softwarerestriktion verhindern oder erlauben es, ausführbare Dateien aus bestimmten Ordnern zu starten.

Die Funktion ermöglicht es ebenfalls zu definieren, in welche Ordner Applikationen, welche in einem bestimmten Userkontext laufen, Dateien ablegen können.

Generell gibt es für dieses Service nur zwei Arbeitsmodi, den aktiven Modus und den Logging Modus, in welchem nur Applikationslog Events erstellt werden, wenn eine Applikation aus einem nicht erlaubten Ordner gestartet wird.

6.5.2 Optimales Setting

Die optimalen Einstellungen in diesem Bereich lassen sich nicht auf generelle Settings zusammenfassen, sondern nur auf Grundsätze für die Konfiguration, da diese massiv von der eingesetzten Software und dem Verhalten dieser abhängig sind.

So kann man generell aber festhalten, dass auch hier ein Konzept der geringsten Berechtigungen angesetzt werden sollte. Es empfiehlt sich also die Ausführung von Applikationen abseits der bewussten definierten Installationspfade zu blockieren. Zusammen mit einer korrekten Konfiguration der Filesystemberechtigungen, sowie der Einschränkung in welchen Ordnern Applikationen im Kontext eines Users Daten verändern dürfen, sollte ein solider Schutz gegeben sein.

Problematisch werden diese Restriktionen, wenn Applikationen ins Spiel kommen, deren ausführbare Dateien per Default in Benutzerverzeichnissen (Appdata) gelegt werden, da sich hier der Schutz vor Manipulation, wesentlich schwerer umsetzen lässt, da Benutzer in ihrem Profil grundsätzlich Vollzugriff haben.

Solche Situationen lassen sich dann nur mit Hilfe, der kostenpflichtigen Komponente AppLocker lösen, diese erfordert aber den Besitz einer Enterprise Lizenz.

6.5.3 Einfluss auf die beschriebenen Angriffe

Richtig konfiguriert ist diese Funktion sicher eine der besten Möglichkeiten Angriffe auf ein System abzuwehren.

Das System ist unter Verwendung dieser Sicherheitsfunktion grundsätzlich in der Lage, das Starten von jeder Form von Malware oder anderer unerwünschten Software zu unterbinden, solange es einem Angreifer nicht gelingt diese in einem Ordner abzulegen, der in der Whitelist definiert ist. In den meisten Fällen wird eine so genaue Steuerung der ausführbaren Applikationen aber nur in einer sehr statischen Umgebung umsetzbar zu sein, da kleine Veränderungen an den Pfaden schon ausreichen, um einen erneuten Eingriff in die Steuerung der Restriktionen notwendig zu machen. Für Bereiche mit hohem Sicherheitsbedürfnis, stellt die Funktion aber dennoch einen guten Sicherheitsansatz dar.

6.6 Windows Defender

Mit dem Windows Defender hat Microsoft seit Windows 10 eine eigene Antiviren Engine an Bord, die standardmäßig aktiviert ist, sofern keine andere Software zur Bekämpfung von Schadsoftware auf dem System installiert wurde. Seit seiner Einführung ist das Produkt durchaus nicht unumstritten, von Monopol Vorwürfen, über das mögliche Einschleusen von Schadsoftware über einen Fehler in der Scanengine, bis hin zu anfangs schlechten Noten bei diversen AV Tests.

Mittlerweile zeigen die meisten unabhängigen Tests durchwegs gute Ergebnisse, unter anderem gelang Microsoft sogar ein Testsieg beim AV-Test 2019, zusammen mit einem Mitbewerber.

Auch bei diesem Produkt gibt es unterschiedliche Versionen, gegen eine zusätzliche Lizenz kann eine im Umfang stärkere Version genutzt werden. Diese ist aber nicht Teil der Betrachtung, nachdem eine zusätzliche Lizenz erforderlich ist, um diesen zu nutzen. (Microsoft, 2022e)

6.6.1 Funktion

Es handelt sich beim Windows Defender um einen „klassischen“ definitionsbasierten Virenscanner mit Cloud Unterstützung der somit für viel seiner Erkennungsleistung auf Basis seiner Virendatenbank generiert. Dies stellt grundsätzlich ein Problem bei der Erkennung von 0-Day Angriffen dar, hier sollte dann der Cloud Schutz greifen, der es möglich macht, global auftretende Malware schnell zu erkennen und zu blockieren, auch wenn die lokale Datenbank noch keine konkrete Signatur gegen die angreifende Malware hat. Sollte keine Internetverbindung bestehen, oder die Richtlinien des Arbeitsplatzes die Nutzung des Cloud-Dienstes nicht zulassen, steht bei der Basis Version des Defenders nur die eigene Signaturdatenbank zur Verfügung.

Die kostenpflichtige Version des Produkts welche sich wie ein Endpoint Detection and Response (EDR) Client verhält, stellt weitere Erkennungsmethoden auf Basis von heuristischer Analyse und Verhaltenserkennung zur Verfügung.

6.6.2 Optimales Setting

Hier sind die von Haus aus gesetzten Einstellungen grundsätzlich schon die korrekten.

Defender hat sich in den Default Einstellungen über Windows Updates selbst aktuell, der Echtzeitschutz, sowie der Cloud unterstützte Erkennungsdienst sind von bereits standardmäßig aktiviert.

Das Feature zum Selbststutz des Defenders, vor Status oder Settings Änderungen, durch andere Applikationen ist ebenfalls bereits per Default aktiv.

Das Feature „kontrollierter Ordnerzugriff“ schützt in seiner Standardeinstellung bereits alle Benutzerverzeichnisse gegen Schreibzugriff von nicht vertrauenswürdigen Programmen und dient somit als Schutz vor Veränderungen dieser durch Ransom Ware.

6.6.3 Einfluss auf die beschriebenen Angriffe

Durch die beschränkte Heuristik und fehlende Verhaltenserkennung, ist die Auswirkung der Schutzkomponente im Vergleich zu einem verhaltensbasierten Anti-Malware Produkt oder einem vollständigen EDR Client natürlich reduziert.

Die letzten AV-Test Ergebnisse zeigen aber, dass der Windows Defender Antivirus, zumindest im Bereich der bekannten Bedrohungen eine fast 100%ige Erkennungsquote vorweist.

Schwachstellen zeigen sich, wie bei allen definitionsbasierten Schutzsystemen bei gezielten Angriffen oder noch wenig verbreiteten 0-Day, da bei diesen meist weder die Speichersignaturen im Arbeitsspeicher noch die Hashwerte der gedroppten Dateien in den Definitionen enthalten sind. Je nach Angriffsmuster, kann in dieser Phase auch die Umgehung des Selbstschutzes stattfinden. Danach liegt das System dem Angreifer offen.

Sofern es dem Angriff nicht gelingt den Defender zu deaktivieren, verhindert die Funktion „kontrollierter Ordnerzugriff“, dass eine Verschlüsselung von Dateien in den darin definierten Ordnern stattfindet, da der Schreibzugriff verhindert wird. Eine mögliche Exfiltration zur späteren Erpressung wird durch diese Funktion aber nicht verhindert.

Unabhängig von den Schwächen, die der „Defender Antivirus“ gegen über der kostenpflichtigen Variante dem „Defender for Endpoint“ hat, ist die Verwendung, wenn keine kostenpflichtige Lösung zur Verfügung steht auf jeden Fall zu empfehlen. Zumindest sofern nicht wieder eine Schwachstelle wie CVE-2021-42298 vorliegt, dann wird der Defender plötzlich Teil der Angriffskette.

6.7 Windows Defender Firewall

Bei der „Windows Defender Firewall mit erweiterter Sicherheit“ handelt es sich um eine erweiterte lokale End Point Firewall, die es ermöglicht, für das lokale System äußerst granulare Kommunikationsregeln zu setzen. Welche zum einen das Erstellen von Regeln klassisch auf Basis von IP und Port Definitionen und zum anderen auf Basis von Applikationssettings ermöglicht. (Microsoft, 2022e)

6.7.1 Funktion

Als Endpoint Firewall dient die „Windows Defender Firewall mit erweiterter Sicherheit“ als Netzwerkfilter für den eingehenden so wie ausgehenden Netzwerkdatenverkehr auf dem Client System.

Anhand des bestehenden Regelsatzes arbeitet die Firewall Netzwerkpakete ab und erlaubt diese entweder, sofern es eine bestehende Regel gibt, die den Datenverkehr erlaubt, oder verwirft

diese, falls es keine passende Regel dafür gibt, oder es eine Übereinstimmung mit einer expliziten Deny Regel gibt.

Aufgrund der Applikationsawareness der „Windows Defender Firewall mit erweiterter Sicherheit“ ist es auch möglich, in diesem Umfeld Regeln nicht nur wie bei klassischen Firewalls üblich auf Basis von Quell- oder Zieladresse, Port und Protokoll zu treffen, sondern auch dynamische Regeln auf Basis von Applikationsregeln zu setzen.

Nachdem es sich bei der „Windows Defender Firewall mit erweiterter Sicherheit“ um eine Stateful Inspection Firewall handelt, ist es nicht notwendig für Programme, die nach dem initialen Verbindungsaufbau einen Portwechsel durchführen, Ranges der erlaubten Ports zu definieren, diese werden solang sie im Kontext einer aufgebauten Verbindung erfolgen, automatisch zugelassen.

6.7.2 Optimales Setting

Die optimalen Settings, richten sich hier stark nach den Erfordernissen der Umgebung.

Als generelle Faustregel gilt: „So viel wie nötig, so wenig wie möglich“

Auf einem eher statischen System mit klarem Einsatzgebiet macht es durchaus Sinn, für alle ausgehenden Programme explizite „Allow Regeln“ auf Basis der benötigten Zieladressen zu setzen, sowie am Ende der Regelkette eine implizite „Deny All Regel“ zu setzen, die jede Form, der nicht erlaubten Kommunikation unterbindet. Gleiches gilt auf diesen Systemen auch für die Regeln betreffend eingehende Verbindungen.

Auf Systemen, die weniger klar definierte Aufgaben haben, oder von Usern als Clientsysteme genutzt werden empfiehlt sich eher das bewusste Sperren von unerwünschten Zugriffsarten mit expliziten „Deny Regeln“, gefolgt von einem impliziten „Allow Any“ Regel, da sonst der Verwaltungsaufwand enorm steigen kann falls beispielsweise Services mit nicht Standard-Ports genutzt werden.

Was auf jeden Fall auf allen Systemen anzuraten ist, ist das Unterbinden von nicht benötigten SMB Verbindungen zwischen Systemen, die nicht explizit Dateidienste zur Verfügung stellen. Hier sollte aber nicht vergessen werden, falls Administratoren die Funktionalität benötigen, die Jump Hosts der Administratoren in die Ausnahme der Regel aufzunehmen.

6.7.3 Einfluss auf die beschriebenen Angriffe

Der größte Einflusspunkt auf die Angriffskette ist hier sicher, wenn ein Minimalberechtigungskonzept gewählt wird, ist zum einen das Erschweren von lateraler Ausbreitung einer Bedrohung im selben Netzwerksegment. Zum anderen das stellenweise Verhindern des Nachladens von Malware Komponenten, nach der initialen Kompromittierung,

wenn beispielsweise für Scripting Host oder den PowerShell Interpreter mit Hilfe einer „Deny Regel“ der Zugriff auf Ressourcen außerhalb des eigenen Netzwerks verboten werden.

Sofern die Anforderungen der Umgebung es erlauben, können Angriffe mit Hilfe der Firewall also entweder eingedämmt, oder sogar daran gehindert werden nach einer initialen Kompromittierung, Persistenz, durch das Nachladen von anderen Programmteilen, zu erlangen.

6.8 Gruppenrichtlinien

Gruppenrichtlinien bilden eines der zentralen Instrumente, um Windows Systeme in einem Firmennetzwerk zu kontrollieren und einheitliche Einstellungen zu verteilen.

Abgesehen von den grundlegenden von Microsoft, mit dem Betriebssystem ausgelieferten Richtliniendefinitionen, können diese auch um weitere Richtlinien, für andere Programme erweitert werden.

So ist es möglich, dass auch Softwarekomponenten wie „Google Chrome“ oder „Microsoft Office“ und viele andere, auch zentral gesteuert werden können.

Viele der in diesem Kapitel bereits beschriebenen Komponenten lassen sich ebenfalls über dieses Instrumentarium kontrollieren. Im Folgenden liegt der Fokus aber auf den wichtigsten Punkten, um zum Angriffsflächen zu schließen und Punkte aus der erarbeiteten Liste zu erfüllen.

6.8.1 Funktion

Die Funktion der Gruppenrichtlinien ist die zentrale oder lokale Steuerung der meisten Einstellungen von Windows. Sie bietet damit den Kern der Steuerungsfunktionen, die auch ohne die zusätzlichen Kosten, eines MDM Systems, für alle Windows Anwender zur Verfügung stehen.

Für die Verarbeitungsreihenfolge der Richtlinien gilt das Prinzip „LSDOU“ also Lokal, Site, Domain, Organisationseinheit. Die Relevanz hierbei ist, dass die an der letzten verarbeiteten Richtlinie im Konfliktfall zwischen zwei Einstellungen zieht. (Aigner et al., 2020)

6.8.2 Optimales Setting

Nachdem Microsoft regelmäßig neue Richtlinien für neue Systemversionen herausgibt, ist es wichtig in regelmäßigen Abständen die aktuellen Definitionen und Übersetzungsdateien einzuspielen, nachdem diese nicht in den normalen Updates für Domaincontrollern enthalten sind. Bei lokalen Systemen werden die Richtlinien bei den betreffenden Updates automatisch eingespielt.

Die relevantesten Richtlinien, um die Sicherheit eines Systems zu erhöhen sind:

Computerkonfiguration\Richtlinien\Windows-Einstellungen\Sicherheitseinstellungen\
Kontorichtlinie\

- Kennwort muss Komplexitätsvoraussetzungen entsprechen – Aktiviert

  Diese Option erzwingt die Verwendung von mindestens drei von vier möglichen Zeichentypen innerhalb eines Kennworts. Zusätzlich verhindert dieses Setting, dass mehr als 3 Buchstaben des Namens, des Benutzers im Kennwort verwendet werden. Diese Option verhindert die Verwendung von zu leichten Kennwörtern, die sich durch Wörterbuchattacken leicht knacken lassen.

- Kennwortchronik erzwingen \n gespeicherte Kennwörter – 5 oder mehr

  Diese Option verhindert, dass User eines ihrer letzten x Kennwörter wieder als neue Kennwort verwenden. Diese Option verhindert somit, dass bekanntgewordene Kennwörter nach kurzer Zeit bereits wieder gültig werden.

- Minimale Kennwortlänge – 12 Zeichen

  Diese Option erzwingt die minimale Kennwortlänge von x Zeichen. Durch die größere Länge des Kennworts werden Bruteforce Angriffe auf das Kennwort erschwert.

- Minimales Kennwortalter – 1 Tag

  Diese Option verhindert, dass ein Kennwort direkt mehrfach am selben Tag gewechselt wird, um die Chronik der gespeicherten Kennworte zu umgehen und direkt wieder das alte Kennwort zu setzen.

- Maximales Kennwortalter Kein Ablauf

  Das Thema der Kennwortgültigkeitsdauer ist in den vergangenen Jahren immer mehr in Verruf geraten, so rät Microsoft in seiner Cloud Umgebung Azure Active Directory (AAD) dazu, keinen Ablaufzeit zu definieren. Ebenso sehen die Empfehlungen von NIST, SANS, und dem UK Cyber Security Center aus. All diese Empfehlungen basieren jedoch auf der Annahme, dass der Systembetreiber in der Lage ist für seine Benutzer eine Multifaktor Authentifizierung anzubieten. (National Cyber Security Centre, 2018; Spitzner, 2019)

Computerkonfiguration\Richtlinien\Windows-Einstellungen\Sicherheitseinstellungen\
Kontosperrungsrichtlinie\

- Kontosperrungsschwellwert – 6 Versuche

  Diese Richtlinie verhindert Online Bruteforce, oder Credential Stuffung Angriffe auf Konten, indem es die Anmeldung für den betreffenden User nach der Eingabe von fünf falschen Anmeldungen sperrt. Der Wert fünf Versuche ist so gewählt, dass eine versehentliche Sperre verhindert wird, aber auch das Erraten, wenn Teile des Passworts bekannt werden, dennoch unwahrscheinlich ist. (Aigner et al., 2020, S. 497; PCI Security Standards Council, 2022, S. 173)

- Kontosperrdauer – 30 Minuten bis Unendlich

Nachdem ein Konto gesperrt wurde, beginnt automatisch ein Timer zu laufen ,der die Zeit bis zur erneuten Freigabe des Kontos zählt. Je nach Situation empfiehlt es sich aber, generell Useraccounts, die die Lockout Policy ausgelöst haben, bis zur Interaktion mit einem Helpdesk Mitarbeiter oder einem Administrator oder einer Administratorin gesperrt zu lassen. Da im Zuge der Interaktion mit dem Support Personal die Möglichkeit besteht die Identität des gesperrten Users zu validieren und im Falle, dass der User oder die Userin, nicht weiß, warum der Account gesperrt ist, Untersuchungen zur Ursache der Kontosperre zu beginnen. Wodurch Angriffe auf bestimmte Useraccounts wesentlich schneller entdeckt werden können. (PCI Security Standards Council, 2022, S. 173)

Computerkonfiguration\Richtlinien\Windows-Einstellungen\Sicherheitseinstellungen\Lokale-Richtlinien\Sicherheitsoptionen

- Netzwerksicherheit: Keine LAN-Manager-Hashwerte für nächste Kennwortänderung Speichern – Aktiviert

  Diese Richtlinie ist bei Legacy Systemen relevant, falls bereits Hashes in diesem Format bestanden haben. Sie erzwingt, dass aktuelle Hashformate zur Speicherung der Benutzerkennwörter genutzt werden.

- Netzwerksicherheit: LAN-Manager Authentifizierungsebene – NTLMv2 antworten senden, LM&NTLM verweigern

  Diese Richtlinie stellt sicher, dass nur die aktuellste Kommunikationstechnologie für die Authentifizierung verwendet wird. (Kann in Umgebungen mit Legacy Systemen zu Problemen früheren)

- Interaktive Anmeldung: Anzahl der zwischenzuspeichernden Anmeldungen […] – 0

  Dieser Wert verhindert, dass Anmeldungen von Domänen Benutzeraccounts auf dem Computer zwischengespeichert werden. Was zum einen die Sicherheit zwar massiv erhöht, nachdem die zwischengespeicherten Benutzerdaten, User-ID und Passworthash, nach der Anmeldung wieder direkt gelöscht werden. Auf der anderen Seite erfordert dieses Setting aber eine permanente Verbindung zu mindestens einem Domaincontroller, da sonst keine Anmeldung durchgeführt werden kann. Sie ist somit nur für stationäre Systeme mit einem besonders hohen Grad der Bedrohung geeignet. Keinesfalls sollte diese Richtlinie auf mobilen Endgeräten eingesetzt werden, wenn nicht sichergestellt werden kann, dass eine VPN Verbindung bereits vor der ersten Anmeldung verfügbar ist und das Anmelden ohne bestehende Verbindung wirklich ein unerwünschtes Verhalten ist.

Computerkonfiguration\Richtlinien\Windows-Einstellungen\Sicherheitseinstellungen\Lokale-Richtlinien\Zuweisen von Benutzerrechten

- Debuggen von Programmen – Aktiviert

  Keine eingetragenen User

Diese Einstellung verhindert, sogar bei Administratoren, dass der User das Recht SeDebugPrivilege erhält. Dies macht es dem User unmöglich ein Speicherabbild eines laufenden Prozesses zu erzeugen. Damit laufen Angriffe, die auf das Auslesen von Benutzerinformationen aus dem LSASS Service abzielen ins Leere.

Computerkonfiguration\Administrative Templates\Windows-Komponenten\Remotedektopdienst\Remotedesktopsitzungs-Host\

- Benutzerauthentifizierung mit Authentifizierung Netzwerkebene ist für die Remoteverbindung erforderlich – Aktiviert

  Diese Einstellung erzwingt, dass der Benutzer eine gültige Anmeldung übermittelt, um eine Remotedesktop Verbindung herstellen zu dürfen. Damit wird verhindert, dass Angreifer ohne gültige Anmeldedaten eine RDP Session herstellen können und diese zur Informationsgewinnung nutzen können. Der Nachteil ist, dass ein anstehender Passwortwechsel bei der Anmeldung mit dieser Technik nicht durchgeführt werden kann.

Computerkonfiguration\Administrative Templates\Windows-Komponenten\Powershell

- Protokollierung von Powershell-Skriptblöcken – Aktiviert

  Dieses Setting aktiviert und protokolliert die Ausführung von PowerShell Skripts. Während diese Einstellung keinen direkten Einfluss auf die momentane Sicherheit des Systems hat, ermöglicht sie in der weiteren Folge die leichtere forensische Aufarbeitung eines Vorfalles, sofern die PowerShell als Vektor genutzt wurde.

- Skriptausführung aktivieren – Nur Signierte

  Diese Option verhindert das Ausführen von PowerShell Skripten, sofern diese nicht mit einem gültigen Codesignatur Zertifikat signiert wurden. Diese Einstellung eignet sich nur, sofern die sie nutzende Organisation über die entsprechenden Signaturzertifikate verfügt. In diesem Fall erhöht sich dadurch die Sicherheit aber massiv, da die Ausführung von Skripten oder der Remote Aufruf von Befehlen ohne Signatur verweigert wird. Damit sinkt der Wert der PowerShell als Angriffswerkzeug massiv.

Computerkonfiguration\Administrative Templates\Netzwerk\Netzwerkanbieter\

- Gehärtete UNC-Pfade – Aktiviert

  *\Sysvol & *\Netlogon jeweils mit den Einstellungen: RequireMutualAuthentication=1,RequireIntegrity=1,RequirePrivacy=0

  Diese Settings sorgen dafür, dass die Verarbeitung der Gruppenrichtlinien und login sowie Logoff Skripte, nur von einem Server erfolgen kann, dem auch vertraut wird, und der sich auch gegenüber dem Client korrekt authentifiziert. Es verhindert somit, dass ein „Man in the Middle“ Angriff durch einen gefälschten Server erfolgen kann.

6.8.3 Einfluss auf die beschriebenen Angriffe

Aufgrund der Vielzahl der Optionen, die in diesem Abschnitt unter den optimalen Einstellungen genannt wurden, befinden sich die Informationen zur Auswirkung der Einstellungen direkt unter der jeweiligen Option, bzw. dem empfohlenen Wert.

6.9 Treiber Signierung

Beginnend mit dem Release der 64 Bit Version von Windows XP hat Microsoft begonnen für die Installation und das anschließende Starten von Treibern eine valide digitale Signatur zu erwarten. Dabei handelt es sich um eine Technik zum Schutz der Endbenutzer vor manipulierten oder nicht den Richtlinien der „Windows Hardware Quality Labs“ entsprechenden Treibern.

6.9.1 Funktion

Bei der Erzwingung der Treibersignatur handelt es sich um eine Lösung, die sicherstellen soll, dass nur Treiber installiert und geladen werden können, deren Hersteller den Microsoft Vertrauenskriterien entsprichen und deren Treiber den Aufbaugrundlagen des WHQL entsprechen. Besonders für Windows 10 und 11 Kernel Mode Treiber wurde im Jahr 2021 das Set der Kriterien für die erfolgreiche Signierung nochmals angepasst. Hier reicht das reine Signieren mit einem gültigen Zertifikat nicht mehr aus, der Treiber muss auch mit Microsoft registriert sein. (Born, 2021a)

6.9.2 Optimales Setting

Nachdem es sich um eine von Microsoft als besonders wichtig angesehene Schutzfunktion handelt, ist der Default Zustand hier bereits das optimale Setting in einer normalen Umgebung, die es erlaubt.

Es kann jedoch, in Bereichen wo mit ungewöhnlicher Hardware gearbeitet wird, notwendig sein, diesen Schutz zu deaktivieren, wenn der Hersteller der Spezialhardware keine aktuell signierten Treiber anbietet, oder generell unsignierte Treiber zur Verfügung stellt. Dies kann vor allem im Produktionsumfeld vorkommen. (Gibb, 2017)

6.9.3 Einfluss auf die beschriebenen Angriffe

Wie bei vielen anderen Sicherheitsmechanismen ist auch dieser nicht unfehlbar was die Verteidigung gegen Angriffe mit Treibern betrifft. Angriffe mit manipulierten oder mit rein bösartigem Zweck geschriebene Treiber, werden durch diese Methode an der Installation gehindert, was die Sicherheit des Systems erhöht, indem es die Schwelle für den Erfolg eines Angriffs höher setzt. Es bleiben dennoch zwei Angriffsszenarien bestehen. Zum einen können durch Sicherheitsprobleme bei Hardwareentwicklern valide Signaturzertifikate entwendet

werden, was es für die Zeit, die ein Angriff unentdeckt bleibt, potenziell ermöglicht einen bösartigen Treiber zu signieren und zur Validierung zu übermitteln. In dieser Phase würde er sich auf Systemen, ohne dass er als illegitim markiert wird, installiert. Ein weiteres Szenario ist weiterhin die Installation von validen Herstellertreibern, die selbst schwere Sicherheitsschwachstellen beinhalten. Sogenannte „Bring Your Own Vulnerable Driver“ (BYOVD) Angriffe nutzen meist weitverbreitete Treiber, die Schwachstellen beinhalten, die ihnen die nötigen Zugriffsrechte verschaffen. Der Vorteil des Nutzens dieser Treiber ist es, dass weitverbreitete Treiber wie beispielsweise ein Dell Treiber, der bei normaler Verwendung Nutzern das Updaten der Gerätefirmware erlaubt, von Angreifern dazu genutzt werden kann Kernel Zugriff zu erhalten. Trotz der Gefahr, die er mit sich bringt, auf Grund der massiven Nebenwirkungen für legitime User nicht auf der Blacklist von Virenherstellern oder Microsoft landen können. (Baines, 2021)

6.10 Windows PowerShell

Während es sich bei der Microsoft PowerShell nicht um ein klassisches Tool zum Schutz des Systems handelt, handelt es sich doch um eines der mächtigsten und vor allem vielseitigsten Werkzeuge, die in neuen Windows Systemen integriert sind.

Diese Vielseitigkeit bringt die Tatsache mit sich, dass diese in vielen Angriffen genutzt wird, damit macht es Sinn die Settings im Kontext der Systemverteidigung zu betrachten.

6.10.1 Funktion

Bei der Windows handelt es sich um das Schweizer Taschenmesser der Windows Systemumgebungen. Sie ist in der Lage in so gut wie alle Bereiche des Windows Betriebssystems einzugreifen. In der Standardinstallation reichen diese bereits von der Integration des „.Net Frameworks“, Zugriff auf das „Component Object Model“ (COM), „Windows Management Instrumentation“ (WMI) bis hin zum „Windows Application Programming Interface“ (WinAPI). Abgesehen von diesen Schnittstellen, kann die PowerShell von Haus aus auf die Registry sowie, sofern die Nutzerrechte es erlauben, auf das Active Directory zugreifen, sowie natürlich auf das lokale Filesystem.

Abgesehen von diesen von Haus aus vorhandenen Funktionen, lässt sich die Windows PowerShell durch das Hinzufügen weiterer Module aus lokalen sowie Onlinequellen zusätzlich um Funktionen für andere Services erweitern.

All diese Funktionen lassen sich zusätzlich, sofern auf dem Remotesystem der WinRM Dienst läuft und die Firewall den Zugriff zulässt, mithilfe einer gültigen Authentifizierung auch auf einem oder beliebig vielen Hosts im Netzwerk entfernt steuern.

6.10.2 Optimales Setting

Wie die Liste der Optionen die einem die PowerShell bietet schon zeigt, ist es besonders hier wichtig korrekte Einstellungen zu finden, um die Sicherheit zu erhöhen.

Der erste Punkt ist die Ausführungsrichtline. Auch wenn es sich dabei nur um ein „Safety und kein Security“ Feature handelt, das bei einem gezielten Angriff mithilfe von Bypass Flags umgangen werden kann, kann es zumindest das versehentliche Ausführen von Skripten blockieren. Die optimale Einstellung ist hier stark von der Umgebung abhängig. Die Default Einstellung eines Windows 10 Systems ist „Restricted“, diese Richtlinie erlaubt nur das Ausführen von einzelnen Befehlen in der PowerShell, nicht aber den Aufruf von Skriptdateien. Für Systeme, auf denen es keine Notwendigkeit gibt, Skripts auszuführen muss die Konfiguration nicht verändert werden.

Auf Systemen, die eine Notwendigkeit haben, Skripts auszuführen, empfiehlt sich, sofern die Möglichkeit zur Signierung von Skripten im Netzwerk existiert, das Richtlinien Level auf „AllSigned“ zu setzen. In diesem Fall können nur Skripte automatisch ausgeführt werden, die eine vertrauenswürdige digitale Signatur aufweisen. Sofern die Signatur noch nicht als vertraut gilt, wird zur Bestätigung des Vertrauens aufgefordert.

Bei Systemen wo die Möglichkeit der Signatur nicht besteht und zusätzlich nur lokale selbstgeschriebene Skripte ausgeführt werden sollen, wäre die optimale Einstellung “RemoteSigned“. Auf dieser Einstellung erlaubt die Ausführungsrichtlinie das Ausführen aller lokal vom aktiven User geschriebenen Skripte, auch ohne digitale Signatur,erfordert diese aber bei allen Skripten und Konfigurationsdateien, die von externen Quellen bezogen wurden. (Wheeler & Lombardi, 2022a)

Nachdem sich diese Richtlinien aber mit einem einfachen Aufruf der Powershell.exe mit dem Parameter -ex „Bypass“ umgehen lassen, sollte der Zugang und die Verwendung der PowerShell, sofern sie nicht aktiv genutzt wird, blockiert werden. (Sutherland, 2014)

Dies kann über das Setzen der Gruppenrichtline Benutzerkonfiguration\Administrative Templates\System „Angegebene Windows-Anwendungen nicht ausführen“ auf aktiv mit dem Eintrag für die Applikationen powershell.exe und powershell_ise.exe erfolgen.

Zusätzlich zur Ausführungsrichtlinien gibt es auch noch die Möglichkeit den Sprachmodus einzuschränken. Von Haus aus läuft die Windows PowerShell im „Full Language“ Modus, welcher alles zulässt.

Hier empfiehlt es sich den Modus auf den „Constrained Language Mode“ zu setzen. In diesem Modus stehen der PowerShell nur definierte .Net Klassen und nicht das volle Spektrum der oben beschriebenen Komponenten zur Verfügung. (Wheeler & Lombardi, 2022b) Dies kann am verlässlichsten über die Automatik der PowerShell passieren, indem mit Hilfe einer Software-Restriktion der Zugriff der PowerShell auf das lokale Temp Verzeichnis blockiert wird. Zur Erkennung im $env:temp definierten temporären Verzeichnis versucht eine Datei zu erstellen. Sollte die PowerShell Instanz nicht in der Lage sein, diese Datei zu erstellen, schaltet diese automatisch in den eingeschränkten Sprachmodus. (Sommergut, 2019)

Es sollte dafür gesorgt werden, dass nur aktuelle Versionen der PowerShell zugänglich sind, was durch die Deinstallation von alten PowerShell Versionen (2.0) erreicht werden kann.

Sofern sich nicht für die Variante der Deaktivierung der PowerShell entschieden wird, empfiehlt es sich auf jeden Fall, wie auch schon im Kapitel über die Gruppenrichtlinien erwähnt, dass Logging der ausgeführten Skripte aktiviert werden. Auf Systemen an denen zusätzlich zu Skripten die PowerShell auch interaktiv genutzt wird, sollte über die Verwendung der Transkriptionsfunktion nachgedacht werden, um sicherzustellen, dass alle abgesetzten Befehle auf dem System dokumentiert werden. Zusätzlich sollte diese Datei dann auch vor der Zerstörung geschützt und nach Möglichkeit ausgelagert werden. Um diese Option zu nutzen, muss im Vorfeld die PowerShell noch mittels von MS mitgeliefertem PowerShell Skript als Eventprovider registriert werden.

Sollte der Remotezugriff via PowerShell Sessions auf anderen Computersystemen gewünscht sein, sollten die folgenden Einstellungen beachtet werden.

Für den Verbindungsaufbau sollte auf jeden Fall die verschlüsselte SSL Verbindung ausgewählt werden. Besonders wenn die Authentifikation nicht über Kerberos oder NTLM stattfindet, da sonst in Folge sowohl die Authentifizierung als auch der Nachfolgende Datenverkehr im Klartext gelesen werden kann.

Wie bei allen andern Management Instrumenten sollte auch für PowerShell Remoting darauf geachtet werden, dass auf der lokalen Firewall nur Zugriffe von definierten Hosts oder Netzen erlaubt sind, um die Angriffsoberfläche weiter zu minimieren. (Wheeler, Chase & Schonning, 2021)

6.10.3 Einfluss auf die beschriebenen Angriffe

Während die Ausführungsrichtlinie für einen zielgerichteten Angriff kein wirkliches Hindernis darstellt, da sie sich zu leicht auf zu vielen Wegen ausheben lässt, kann das Setzen dennoch zufällige Opportunitätsangriffe unterbinden, da eine automatische Ausführung unterdrückt wird, oder zumindest dem User im Fall einer nicht Signatur mit einem nicht automatisch vertrauten Zertifikat, ein Dialog eingeblendet wird, der wenn er unerwartet kommt, den User zum Nachdenken anregen sollte.

Schwerer zu umgehen und damit besser zur Sicherung geeignet ist die Begrenzung des Sprachmodus, vor allem wenn diese nicht nur über die Umgebungsvariable erfolgt, sondern mit Hilfe von beschränkten Rechten über die Software-Restriktionen. Während dieses Setting zwar wenig gegen die Ausführung von Skripten unternehmen kann, in der nur normale PowerShell Befehle genutzt werden, verhindert es doch den Einsatz komplexer Skripte, die über Systemschnittstellen Manipulationen vornehmen wollen.

Das Verbieten des Aufrufs der PowerShell ist zwar mit Abstand die eindeutigste Lösung, um Angriffe zu blockieren die PowerShell Funktionen als Teil der Angriffskette sehen,hat aber auch die radikalsten Auswirkungen auf die Nutzbarkeit des Systems. So rät beispielsweise das

Australische Cybersecurity Center davon ab und empfiehlt Härtungsschritte sowie Loging. (Australien Goverment, 2016)

Sollte ein Angriff über den PowerShell Vektor erfolgen, kann es für die forensischen Untersuchungen in Folge der Kompromittierung von enormem Vorteil sein, wenn sowohl die Informationen aus dem Skript Block Logging, als auch die Transkripte vorhanden sind. Diese ermöglichen es schnell zu erkennen was geschehen ist, und erleichtern das Entdecken weiterer betroffener Systeme, auf denen möglicherweise noch kein Schaden entstanden ist.

Im Bereich des Remotings gilt klar, dass alle Verbindungen, die bereits im Vorfeld blockiert werden, Verbindungen sind um dessen Kontext man sich im Nachhinein keine Sorgen machen muss. Wenn ein Angreifer, also auf Grund einer korrekt gesetzten Firewall Regel das System nicht erreicht, können beispielsweise weder Schwachstellen in der Authentifizierung ausgenutzt werden, noch in Folge Funktionen der Remote PowerShell genutzt werden.

6.11 Secure Boot

Bei Secure Boot handelt es sich um eine seit dem Erscheinen von Windows 8 zum Einsatz kommenden Schutztechnologie, die auf Basis des „Unified Extensible Firmware Interface“ (UEFI), den Nachfolger des klassischen „Basic Input Output System“ (BIOS). Es soll sicherstellen, dass beim Start des Systems, nur signierte Komponenten geladen werden. (Möhring, 2021)

6.11.1 Funktion

Secure Boot bietet als eine der grundlegenden Sicherheitsfunktionen des Mainboards einen Schutz gegen die Manipulation des Startprozesses.

Es wird bei Verwendung von Secure Boot sichergestellt, dass sowohl der Betriebssystem Bootloader als auch alle Firmwarekomponenten und Treiber, mit einer korrekten, und vertrauenswürdigen digitalen Signatur versehen sind. Damit sollte in der Theorie sichergestellt sein, dass während des Systemstarts, bis zur vollständigen Übernahme des Systembetriebs durch das Betriebssystem nichts geladen werden kann, das nicht über ein gültiges Zertifikat verfügt. Ab diesem Punkt sollte unter Windows dann die verpflichtende Treibersignatur diese Aufgabe übernehmen.

6.11.2 Optimales Setting

Es handelt sich hierbei um ein binäres Setting, solange die Verwendung nicht durch Probleme mit der Firmware oder Treibern der verwendeten Hardware, oder bewusste Manipulation am Bootloader unmöglich gemacht wird, sollte das Setting auf dem Default Wert der meisten ab dem Jahr 2020 ausgelieferten Mainboards verbleiben. Dieser ist „Aktiviert“.

6.11.3 Einfluss auf die beschriebenen Angriffe

Während das System auf Grund von einigen bekannten Schwachstellen gezielt angreifbar ist, erhöht es doch die Sicherheit gegen die unbewusste Nutzung eines Systems mit tiefliegender Schadsoftware wie beispielsweise einem Rootkit.

Ein grundsätzliches Problem dieser Lösung ist der Fokus auf die digitale Signatur durch Microsoft.So kann es durchaus notwendig sein, auf Systemen mit Treibern von kleineren Herstellern, die nicht den MS Signaturprozess durchlaufen, ohne Secure Boot zu betreiben. Eine von Microsoft bereitgestellte Lösung für die Verwendung von alternativen Bootloadern, untergräbt hier die eigene Sicherheitsvision ebenfalls. Vorteil der Erfordernis der Signatur durch eine von Microsoft kontrollierte Zertifizierungsstelle ist jedoch, dass der Verlust eines Signaturzertifikats durch einen Hardware-Hersteller, nicht zu einer kompletten Aushebelung des Systems führt.

Somit handelt es sich leider, wie bei vielen Punkten der Sicherheitsarchitektur nicht um die Panazee um sicherzustellen, dass ein System ohne Manipulation gestartet werden kann, aber es ist ein weiterer Baustein in der Sicherheitskette.

6.12 Local Admin Password Solution – LAPS

Bei LAPS handelt es sich um eine in Netzwerken dringend empfohlene und von Microsoft kostenlos zur Verfügung gestellten Lösung, deren Zweck darin liegt, Administratorenkennwörter pro Gerät zufällig zu generieren.

6.12.1 Funktion

Anhand von Einstellungen, die in den Gruppenrichtlinien für Benutzerkennworte gesetzt wurden, erstellt das Programm in den über die Kennwortgültigkeitsdauer gesteuerten Abständen neue Kennworte für den lokalen Administratorenaccount. Das Kennwort wird nach der Erstellung in ein Objektattribut des Computeraccounts im Active Directory geschrieben und erst nach dem erfolgreichen Schreiben dieses Attributs wird das Kennwort für den Account wirklich gesetzt.

6.12.2 Optimales Setting

Während das LAPS selbst nicht wirklich viele Einstellungsmöglichkeiten mitbringt, muss für den Einsatz vorab eine Schema Erweiterung für das MS-Active Directory eingespielt werden, welche die Liste der Computerobjektattribute um die Attribute ms-Mcs-AdmPwd und ms-Mcs-AdmPwdExpirationTime erweitert.

Für diese Attribute müssen in Folge noch mit Hilfe der mitgebrachten PowerShell Module die Rechte an die Computerkonten vergeben werden, dieses Attribut für sich selbst zu beschreiben, sodass die installierte LAPS Instanz das neue Passwort sowie die Ablaufzeit ins AD Übertragen kann.

Zusätzlich müssen ebenfalls die Leserechte für jene Benutzergruppen vergeben werden, die in der Folge dazu berechtigt sein sollen, das Kennwort wieder auszulesen. Hier empfiehlt es sich wieder besonders bei sicherheitsrelevanten Clients den Kreis der berechtigten Personen besonders gering zu halten. Für generelle Systeme empfiehlt es sich die Rechte all jenen Personen zu geben, die im Zuge der Wartung oder dem Support eines Clients Zugriff auf den lokalen Administratoren Account benötigen. (Aigner et al., 2020, S. 500–504; Dishan, 2021)

6.12.3 Einfluss auf die beschriebenen Angriffe

Der größte Einfluss auf die bisher vorgestellten Angriffe ist hier, dass sobald diese Lösung installiert ist und die Arbeit aufgenommen ist alle Systeme mit unterschiedlichen lokalen Administratorenkennwörtern ausgestattet sind. Welche sich zusätzlich auch noch in regelmäßigen Abständen selbstständig aktualisieren. Dies hat einen massiven Einfluss auf die Möglichkeiten eines Schadprogramms oder auch Angreifers sich im Netzwerk auszubreiten, nachdem ein Host kompromittiert wurde. Da auf Grund dieser Lösung, selbst wenn es gelingt, den Hash des Administratoren Accounts erfolgreich auszulesen und in Folge offline zu knacken, nur um das Kennwort zu einem Account handelt und nicht wie früher in vielen Netzwerken gängig, um das Kennwort für alle lokalen Administratorenaccounts.

6.13 Windows Sandbox

Mit der Windows Sandbox bietet Microsoft Usern von Windows 10 ab der OS Build Version 18305 oder auch dem Feature Update Namen 19H1 eine Möglichkeit, Software oder verdächtig wirkende Dateien in einer isolierten und flüchtigen Umgebung zu testen, ohne die Sicherheit der lokalen Installation zu gefährden. (Microsoft, 2019b)

6.13.1 Funktion

Bei der Windows 10 Sandbox handelt es sich um ein klassisches nicht persistentes Sandbox System, das mit Hilfe der Hyper-V Grundlage die Sicherheitsfunktionen dieser Umgebung ausnützt. So wird die virtuelle Umgebung automatisch mit vollständiger Kernel Isolation betrieben, die ein Ausbrechen aus der VM verhindern soll.

Eine Besonderheit der Sandbox ist, dass es sich zwar um eine vollwertige Installation von Windows handelt, diese aber wesentlich weniger Storage Impact auf dem ausführenden System hat, als eine vollwertige zweite Windows Installation. Dies wird dadurch erreicht, dass nur jene Dateien der Windows Installation für die Sandbox neu angelegt werden, die sich im Betrieb des Systems ändern können. Dateien die unveränderlich sind, werden weiterhin direkt vom Host genutzt, und kommen damit mit nur knapp 500 MB an zusätzlichen Plattenplatzverbrauch aus.

Auch bei der Belegung des Arbeitsspeichers verhält sich das Sandbox System eher wie eine Container Virtualisierung und weniger wie eine klassische virtuelle Maschine.

Mit der verwendeten Lösung des integrierten Kernel Scheduler wird laut Microsoft sicher gestellt, dass auf dem Host System laufende Prozesse mit hoher Priorität weiterhin die benötigten CPU-Ressourcen erhalten, die sie benötigen.

Mit dieser Umgebung ist es sogar möglich, das Interface der Grafikkarte direkt zu exponieren, um in der Sandbox laufenden Applikationen direkten Zugriff auf die Grafikkarte zu geben. Diese Funktion hat in der Vergangenheit, aber bei anderen Virtualisierungslösungen bereits erfolgreich zur Umgehung der VM Isolierung geführt.

Eine Einschränkung die Windows 11 User nicht mehr betrifft ist, dass Installationen, die einen Neustart erfordern, nicht möglich sind nachdem bei einem Neustart das Image wieder auf den Grundzustand zurückversetzt wird. Die Daten der durchgeführten Installation gehen damit verloren. (Microsoft, 2022f)

6.13.2 Optimales Setting

Die Einstellungen der Sandbox lassen sich über ein Konfigurationsfile erzielen.

vGPU: Deaktiviert, sofern es der Test, der in der Sandbox durchgeführt werden soll, nicht unbedingt erfordert.

Netzwerk: Deaktiviert, nachdem die Betrachtung der Sandbox vor allem im Sicherheitskontext erfolgt, macht es definitiv Sinn der Instanz den Zugriff zum lokalen Netzwerk zu verbieten, um zu verhindern, dass ein potentiell unerwünschtes Programm, das in dieser Umgebung getestet wird, Zugang zum Netzwerk und damit potentiell anderen nicht isolierten Systemen erhält.

Mapped Folders: Der Pfad des Host Ordners an dem sich die zu testenden Dateien befinden. Wichtig ist hier die „Read Only“ Kennzeichen zu setzen.

Audio Input: Deaktiviert, solange es der Test nicht notwendig macht.

Video Input: Deaktiviert, solange es der Test nicht notwendig macht.

Protected Client: Aktiviert, dieses Setting sorgt dafür, dass das Windows Image mit erweiterten „Sicherheitsfeatures“ gestartet wird.

(Microsoft, 2022g)

6.13.3 Einfluss auf die beschriebenen Angriffe

Das Sandbox Feature an sich hat für normale Anwender nur einen begrenzte Auswirkung auf die Sicherheit eines Systems, kann aber von Administratoren genutzt werden, um schnelle Test an verdächtigen Dateien oder Programmen durchzuführen. Hierbei muss aber bedacht werden, dass es sich nicht um eine ausgeklügelte Laborumgebung handelt, die die Tatsache, dass es sich um eine virtuelle Maschine handelt, versucht zu verschleiern. So kann es sein, dass sich eine potentiell schädliche Applikation auf diesem System normal verhält, um einem Analyseversuch zu entgehen. Aber auch eine Applikation oder Datei ohne Funktion liefert einem aufmerksamen Administrator die Antwort, die er benötigt.

6.14 Windows Ereignisanzeige

Im besten Fall passiert auf einem Computersystem nichts, ohne dass ein Log Ereignis davon erstellt wird, unabhängig davon, ob die Aktion erfolgreich war oder auch nicht. Dies vereinfacht die Fehlerdiagnose, oder auch die Jagd auf einen Eindringling im System. Der Ort an dem unter Windows diese Ereignisse zusammengetragen werden, ist die Windows Ereignisanzeige. Sie Stellt damit für Administratorinnen und Administratoren ein zentrales Interface zur Verfügung welches zur Betrachtung von Logeinträgen genutzt werden kann.

6.14.1 Funktion

Die Basisfunktion der Ereignisanzeige ist bereits im Namen der Anwendung, bzw. des Microsoft Management Console (MMC) Plugins enthalten, die Anzeige und Verwaltung von Ereignisprotokollen auf Windows Betriebssystemen.

Abgesehen davon, verfügt die Ereignisanzeige aber noch über weitere Funktionen, die es ermöglichen Ereignisse an andere Systeme weiterzureichen, oder auch selbst bei anderen Systemen über Abonnements Ereignisse abzuholen. Wodurch eine Art Logverwaltung Light entstehen kann. Dies hat zum Vorteil, dass Logs auf anderen Systemen gesichert werden können und entscheidende definierte Logeinträge auch nach einem totalen Verlust eines Systems noch zugänglich sind.

Des Weiteren können Ereignisse auch als Auslöser für definierte Tasks genutzt werden.

6.14.2 Optimales Setting

Die optimalen Settings sind in diesem Bereich schwer generell zu definieren, da sie je nach System stark variieren. Generell sollten die Einstellungen aber so gewählt sein, dass Sicherheitsereignisse mindestens 30 Tage in die Vergangenheit abrufbar sein sollten. Aufgrund der Art wie Microsoft seine Logs speichert, bedeutet dies auf jedem System einen anderen Größenwert, falls Logüberschreibung aktiviert ist. So kann es auf einem Client PC ausreichen ein 300 MB Logfile zu definieren, um 30 Tage an Logs im Sicherheitslog aufzubewahren. Auf einem Domaincontroller im Gegenzug werden für dieselbe Zeit um die 15-20 Logfiles zu je 3 GB benötigt. Wichtig ist hierbei zu beachten, dass ab 3 GB Log Größe die Nutzbarkeit der Dateien massiv abnimmt.

Im Bereich der Berechtigungen haben von Haus aus nur Administratoren Zugriff auf das Sicherheitslog. Anwendungslog, Installationslog und Systemlog lassen sich jedoch auch von normalen Anwendern öffnen, nicht aber bereinigen. Während diese Einstellungen im Normalfall vollkommen ausreichend sind, sollte auf Systemen, an welchen normale Anwender auf Grund von Applikationsanforderungen, Administratorenrechte benötigten, diesen Benutzern das Recht zum Löschen des Sicherheitslogs entzogen werden.

Die Folgenden besonders sicherheitsrelevante Logs sollten für die Logweiterleitung an einen besonders geschützten Logging Host definiert werden.:

- Log ID 4624 Successfull Logon – Besonders auf Systemen, auf denen es im Normalfall wenige oder keine interaktiven Anmeldungen abseits von Wartungen gibt, kann dieses Event relevant sein. Zusätzlich kann damit auch die Aktivität besonders interessanter Accounts beobachtet werden. (Administrative Accounts, o.ä)
- Log ID 4625 Failed Logon – Ähnlich wie bei den erfolgreichen Anmeldungen, gibt dieses Ereignis Aufschluss über Anmeldeversuche, und kann damit einen Indikator für laufende Angriffe auf ein Konto liefern, oder dem Helpdesk dabei helfen, dem Anwender zu beweisen, dass er sein Kennwort gerade auf seiner Station x-mal falsch eingegeben hat und sein Account jetzt gesperrt ist.
- Log ID 4740 User Account locked out – Das Folgeereignis, wenn zu viele Logs der ID 4625 für einen User generiert wurden. Dieses Event zeigt an, dass ein User das Kennwort ausreichend oft falsch eingegeben hat, ohne sich dazwischen einmal korrekt anzumelden, um den in der Richtlinie Kontensperrungsschwellwert gesetzten Wert zu überschreiten. Vereinzelt kann die Sperre eines Benutzerkontos darauf hinweisen, dass ein Nutzer sein Passwort vergessen hat. Wenn dieses Event für einen User häufig auftritt, kann wenn es sich nicht um einen besonders vergesslichen Benutzer handelt, davon ausgegangen werden, dass es sich um einen Angriff auf diesen Benutzeraccount handelt. Wenn die Vermutung besteht, dass es ungewollte Aktivität im Bereich der Login Versuche, gibt, sollte zusätzlich zur Eventweiterleitung an den Logmanagement Host, ein Alarm per Mail verschickt werden.
- Log ID 1102 Log cleared – Dieses Ereignis ist, wenn es unerwartet und ohne Interaktion eines Administrators mit dem System ein Indikator, dass auf dem aufzeichnenden System gerade etwas erfolgt ist, was in den seltensten Fällen erwünscht ist. Dieses Event, ist das erste, das nach dem Leeren eines Logs wieder in das Log geschrieben wird, um darauf hinzuweisen, dass kein normaler Rollover, sondern eine händische Leerung des Logs stattgefunden hat. Zusätzlich zur Eventweiterleitung an den Logmanagement Host, sollte bei einem Auftreten dieses Events zusätzlich ein Alarm per Mail verschickt werden.
- Log ID 4663 Attempt made to access object - Dieses Event deutet darauf hin, dass ein nicht autorisierter User versucht hat auf eine Datei oder einen Ordner zuzugreifen, auf welchen er keine Berechtigung hat. Die Zahl der Ereignisse kann auf Kritische eingeschränkt werden, sofern das Dateisystem „Accesbased Ennumeration“ aktiv hat und dem User im Normalfall nur jene Dateien angezeigt werden, für welche er auch zumindest Leserechte hat. Auf Systemen mit diesem Feature deutet dieses Event darauf hin, dass versucht wurde, gezielt eine gesperrte Datei oder einen gesperrten Ordner zuzugreifen.
- Log ID 4728 Member added to securityenabled global group – Dieses Event zeigt an, dass ein Useraccount einer globalen Sicherheitsgruppe hinzugefügt wurde. Das kann im Fall, dass ein User neu angelegt wird und seine initialen Gruppenmitgliedschaften eingerichtet bekommt oder im Zuge einer Positionsänderung zusätzliche Berechtigungen erhält eine vollkommen banale Meldung sein. Es kann aber auch, wenn es sich um ein unerwartet stattfindendes Event handelt, Grund zur Sorge und schnellstmöglichen Reaktion sein,

wenn es sich zum Beispiel um das Hinzufügen eines unerwarteten Users zur Gruppe der Domänen- Administratoren handelt.

- Log ID 7932 Member added to securityenabled local group – Dieses Event zeigt an, dass auf einem Computersystem ein Benutzeraccount einer lokalen Sicherheitsgruppe hinzugefügt wurde. Dies kann sein, wenn das Ereignis in Folge ein bewussten Administratoren Eingriffs gemeldet wird. Sollte das Ereignis ohne Zutun eines Administrators und unerwartet stattfinden, kann es sich um die ersten Anzeichen eines Angriffs mit dem ersten Erfolg in der Ausweitung der Rechte des Angreifers handeln.
- Log ID 4756 Member added to securityenabled universal group - Dieses Event zeigt an, dass ein Benutzerkonto einer universellen Sicherheitsgruppe hinzugefügt wurde. Dies kann im Fall, dass ein User neu angelegt wird und seine initialen Gruppenmitgliedschaften eingerichtet bekommt oder im Zuge einer Positionsänderung zusätzliche Berechtigungen erhält eine vollkommen banale Meldung sein. Es kann aber auch, wenn es sich um ein unerwartet stattfindendes Event handelt Grund zur Sorge und schnellstmöglichen Reaktion sein, wenn es sich zum Beispiel um das Hinzufügen eines unerwarteten Users zur Gruppe der Domänen- Administratoren handelt.
- Log ID 4964 Special groups have been assigned to a new logon – Sofern die Überwachung für Logon Events von speziellen Benutzergruppen aktiviert ist, zeigt dieses Event an, dass sich ein Mittglied dieser Benutzergruppe auf diesem System neu angemeldet hat. Dies kann zum Beispiel genutzt werden, um das Login Verhalten von unter Kompromittierungsverdacht stehenden Benutzeraccounts zu verfolgen
- Log ID 4719 System audit policy was changed. – Dieses Event zeigt an, dass eine gesetzte Auditing Richtlinie geändert oder deaktiviert wurde und wird immer ausgelöst, wenn ein solches Event auftritt, unabhängig von der Einstellung der Auditierung. Das unerwartete Auftreten dieses Events sollte Anlass zur Sorge geben, dass bestehende Überwachungsrichtlinien überschrieben wurden, um möglicherweise auf Dateien zuzugreifen ohne Spuren zu hinterlassen. Abgesehen von der Event Weiterleitung an einen definierten Log Management Host, sollte hier auf Systemen mit besonderem Stellenwert über das Auslösen einer E-Mail-Benachrichtigung nachgedacht werden, wenn ein Event dieses Typs erkannt wird.
- Log ID 4765 SID History was added to an account – Bei diesem Event handelt es sich, wenn es nicht im Zuge einer Domänen Migration auftritt, in den meisten Fällen um ein absolutes Alarmsignal. Dieses Event zeigt an, dass einem Useraccount die SID eines zusätzlichen Accounts eingetragen wurde. In einem legitimen Fall handelt es sich dabei um die SID aus einer weiteren Domäne, oder in einem Fall wo ein Account nach einer Löschung neu angelegt statt wiederhergestellt wurde um die SID des alten Benutzerkontos, um weiterhin Zugriff auf einzelberechtigte Ressourcen zu erhalten. In den meisten Fällen und vor allem wenn dieses Event unerwartet auftritt, handelt es sich um einen Versuch die Privilegien eines Benutzerkontos zu erweitern und sollte dringend verfolgt werden. Aufgrund der Kritikalität dieses Events sollte zusätzlich zur Weiterleitung

an einen Log Management Host, auch eine Benachrichtigung an die Administratoren der Domäne Versand werden.

- Log ID 4766 An attempt to add SID History to an account failed. – Dieses Event zeigt an, dass ein Versuch, der im vorangegangenen Event beschriebenen Rechteausweitung fehlgeschlagen ist. Dies kann an fehlenden Rechten oder einer ungültigen Formatierung der SID gelegen haben. Dieses Ereignis kann als Indikator eines beginnenden Angriffs gewertet werden und sollte demnach zum einen weitergeleitet werden und zum anderen eine Alarmierung über einen automatisch getriggerten Mail Task mit sich bringen.
- Log ID 4104 Execute a remote Command – Diese Event zeigt an, dass ein Befehl in der Powershell abgesetzt wurde und speichert zusätzlich den abgesetzten Befehl im Event Log ab. Dies kann auf Systemen, die häufig zum Ausführen, Erstellen, oder Testen von Skripten zum Archivieren der Skript Vergangenheit genutzt werden und somit einen Audit Trail darstellen. Auf diesen Systemen muss entschieden werden, ob die Events nur lokal gespeichert werden, oder ob diese ein einen Management Host weiter gereicht werden. Auf Systemen, auf denen das Aufkommen von PowerShell Skripts eher gering oder generell unerwartet ist, sofern nicht eine aktive Interaktion mit einem Administrator stattfinden, kann es wiederum Sinn machen die Events weiterzuleiten und auch mit einem gesteuerten E-Mail Event zu versehen. So erhalten Administratoren eine Warnung, wenn auf einem System unerwartet ein PowerShell Skript ausgeführt wird. Um diese Option zu nutzen, muss im Vorfeld die PowerShell noch mittels von MS mitgeliefertem PowerShell Skript als Eventprovider registriert werden.
- Log ID 4688 A new process has been created – Während dieser Logeintrag allein noch keine Aussagekraft hat, ist er in Verbindung mit Aufrufen der PowerShell insoweit interessant, dass sich hier Aufrufe unter Umgehung der Ausführungsrichtlinie erkennen lassen. Außerdem würden sich mit diesem Event Prozessketten verfolgen lassen, die atypisch sind. Beispielsweise wie im Emotet Angriff beschrieben das Starten der PowerShell durch einen Subprozess der WinWord.exe. Aufgrund des Volumens ist eine Weiterleitung dieser Events nur sehr begrenzt ratsam. Sie können aber im Fall, dass ein anderes verdächtiges Event, einen Alarm auslöst zur Nachforschung verwendet werden, sofern der Alarm nicht durch das Löschen eines Eventlogs ausgelöst wurde.

  (Aigner et al., 2020; ManageEngine, 2021; Metcalf, 2015; Microsoft, 2022a)

Zusätzlich zum Abruf der Ereignisse über das MMC Plugin, ist es auch möglich, Daten aus der „Eventvwr.exe“ per Kommandozeile abzurufen, wobei sowohl lokale als auch Remote Abfragen möglich sind, sowie komplexe Filtereinstellungen. (Aigner et al., 2020)

Gleiches gilt auch für den Abruf der Daten mit Hilfe der PowerShell mit Hilfe des Commandlets „Get-EventLog“ auch hier stehen komplexe Filtermöglichkeiten zur Verfügung, die per Skript genutzt werden können, um auf Systemen gezielt nach Events zu suchen die von administrativem Interesse sind. (Wheeler, Chase & Vasin, 2022)

6.14.3 Einfluss auf die beschriebenen Angriffe

Anders als bei klassischen Defensivtechniken handelt es sich bei dem Themenkomplex Eventlogging weniger um Einstellungen und Werkzeuge, die einen Angriff verhindern sollen oder können. Es geht viel mehr um ein Werkzeug, das es Administratoren und Sicherheitsexperten im Anschluss an ein Ereignis ermöglichen soll, herauszufinden dass etwas passiert ist, was passiert ist und welchen Weg ein Angriff durch das Netzwerk genommen hat. Aber auch hier ist anzumerken, dass die klassischen Bordmittel von Microsoft Windows nur eine beschränkte Leistungsfähigkeit haben. Wenn es sich um Systeme mit hohem Log aufkommen handelt, befasst man sich schnell mit einer Vielzahl von archivierten Log Dateien, um den korrekten Zeitraum zu finden, oder man arbeitet mit Logfiles deren Größe es so gut wie unmöglich macht, sie effizient zu durchsuchen, ohne das System auf dem gearbeitet wird in die Knie zu zwingen. Ähnliches gilt auch für den Host, der zur Weiterleitung von Events genutzt wird, sofern das Volumen der weitergeleiteten Ereignisse groß genug ist.

Mit Abstand die nützlichste Technik, die auch, sofern die Events dadurch schnell genug gesehen werden, dazu dienen kann einen beginnenden oder bereits laufenden Angriff zu erkennen ist das Auslösen von E-Mail Events sofern eine bestimmte Event-ID Erkannt wurde. Wenn in einem solchen Fall schnell genug von einem Administrator reagiert wird, kann ein Angriff dadurch gestoppt werden noch bevor, oder zumindest bevor weiterer Schaden entsteht. Zusätzlich werden dadurch Systeme identifiziert, die in Folge isoliert und forensisch untersucht werden sollten, um sicher zu stellen, dass nichts zurückgeblieben ist. Auch ist das reine Protokollieren von Events zu wenig, wenn diese Daten nicht 24x7 oder zumindest täglich kontrolliert werden. Was bei der Flut an Informationen, ohne ein ausgedehntes Security Operation Center (SOC) Team so gut wie unmöglich ist. Hier zeigt sich gut der Vorteil moderner Logmanagement Systeme, welche bewusst in den Daten nach Auffälligkeiten suchen und die Anwender grafisch gezielt auf solche Anomalien hinweisen.

7 ZUSAMMENFASSUNG

Ziel der Arbeit war es zum einen die aktuelle Bedrohungslage der sich Anwenderinnen und Anwender so wie Administratorinnen und Administratoren aktuell stellen müssen zu beleuchten. Dies erfolgte durch die Betrachtung der aktuellen und historischen Daten zur Zahl von Malware Samples, die sich im Umlauf befinden sowie den Zahlen von bekannten Schwachstellen aus der CVE Datenbank. Des Weiteren wurden zusammen mit Spezialisten aus mehreren Sicherheitsfirmen eine Liste der häufigsten Audits im Bereich von Windowsbetriebssystemen erstellt und mit der Begründung sowie den relevanten Teilen des MITRE ATT&CK Frameworks verbunden. Weiters wurden exemplarisch die Infektionsketten von zwei der am weitesten verbreiteten Malware Familien betrachtet, um aufzuzeigen, wo Betriebssystemkomponenten genutzt werden, um einen Angriff erfolgreich durchzuführen. In der Folge wurden dann die systemeigenen Schutzmechanismen sowie für Angriffe genutzte lokale Komponenten wie die PowerShell betrachtet und versucht, Einstellungen aufzuzeigen, welche eine bösartige Ausnutzung verhindern. Bei der Gegenüberstellung der aktuellen Bedrohungssituation und der gewählten Schutzmechanismen kam es zu dem folgenden Ergebnis.

7.1 Beantwortung der Forschungsfrage

Bei der Betrachtung der gestellten Frage:

Inwieweit lassen sich aktuelle Top Bedrohungen der IT-Sicherheit mit Hilfe von Bordmitteln abwehren oder im Schadensausmaß begrenzen?

Sowie der Betrachtung der beiden daraus abgeleiteten Hypothesen

H0: Es ist nicht möglich ein System mit Bordmitteln ausreichend abzusichern.
H1: Es ist möglich ein System mit Bordmitteln ausreichend abzusichern.

Ergibt sich aus der Sicht dieser Arbeit die folgende Beantwortung:

Microsoft liefert mit seinen aktuellen Betriebssystemen eine große Reihe von defensiv nutzbarer Anwendung und Optionen. Diese sind aber technisch nicht in der Lage, Bedrohungen in Umgebungen vollständig abzuwehren, die es nicht zulassen Funktionen radikal zu beschränken. Auf Systemen, wo dies nicht möglich ist, kann zumindest durch Verwendung von korrekt gesetzten Reaktionen auf gewisse Log Ereignisse ein potenzielles sicherheitsrelevantes Ereignis schneller erkannt werden und durch Isolierung des Systems weiterer Schaden verhindert werden. Ähnliches gilt auch für die Unterbindung von lateraler Verbreitung von Schadsoftware durch korrekt gesetzte Share Berechtigungen, oder das generelle Unterbinden von Verbindungen zwischen Client Systemen auf Ebene der Firewall. Somit hat sich aus Sicht des Autors, gestützt von Aussagen von Sicherheitsexperten die Hypothese H0 bestätigt und die Hypothese H1 wird verworfen.

7.2 Gewonnene Erkenntnisse

Im Zuge der Erstellung dieser Arbeit hat sich gezeigt, dass die Zeiten, in denen man als Betreuer oder Betreuerin eines Systems ruhig schlafen konnte, sobald ein System, im Bereich der Einstellungen einer hohen Security Baseline entsprach, vorbei sind. Zumindest wenn die Systeme sich in einem Zustand befinden sollen, der für einen Normalen Anwender noch nutzbar ist.
Während es durchaus so ist, das Microsoft Administratoren mit Optionen wie den Software-Restriktionen ein mächtiges Werkzeug ohne Zusatzkosten in die Hand gibt, handelt es sich dabei um eine vollkommen statische und pfadbasierende Lösung, welche es leicht macht, versehentlich zu viel auf eine Allow List zu setzen, um nicht bei jeder kleinen Änderung die Einträge anpassen zu müssen. Die modernere Lösung „Applocker“, die ebenfalls im Betriebssystem integriert ist, erfordert leider eine wesentlich höherpreisige Enterprise Lizenz, die nicht jedem zugänglich ist und damit nicht in die Betrachtung gefallen ist.
Auch zeigt die Zahl der Angriffe, die Windowskomponenten nutzen, um Angriffe zu starten oder eigene Komponenten unbemerkt nachzuladen gut, dass Microsoft eine Palette unheimlich leistungsstarker Tools bietet, die Absicherung dieser aber in vielen Fällen vernachlässigt hat. Das wahrscheinlich beste Beispiel dafür ist das Herzstück der Automatisierung unter Windows, die PowerShell. So lassen sich hier zwar Richtlinien setzen, die Ausführungen von Scripts und höheren Funktionen verhindern sollten, diese lassen sich aber mit einfachen Aufrufoptionen umgehen. Ähnliches gilt auch für Programme wie Vssadmin und fsutil. Hier bleibt für Sicherheitsverantwortliche nur noch die Option, den Zugriff auf diese Werkzeuge für User komplett zu sperren, was jedoch wieder massive Auswirkungen auf den Arbeitsablauf und die Funktionalität mit sich bringt. Wodurch diese Option nicht in jeder Umgebung zur Verfügung steht.
Der für die Sicherheit wahrscheinlich am besten ausgebaute Bereich unter Windows betrifft die Defender Komponenten. Auch wenn es sich beim Windows Defender Antivirus nur um einen klassischen definitionsbasierten Virenscanner handelt, der dadurch mittlerweile im Vergleich zu KI-Gestützten Systemen und EDR Lösungen das Nachsehen hat, ist er den letzten AV Tests nach, dennoch in der Lage alle bekannten getesteten Malwaresamples zu blockieren. Im Bereich von Exploits und höher entwickelter Malware scheitert die Schutzkomponente allerdings. Hier bietet Microsoft ebenfalls für zahlende Kunden eine verbesserte Version welche EDR Komponenten und eine erweiterte Exploitprävention mit sich bringt. Mit der „Windows Defender Firewall mit erweiterter Sicherheit“ liefert Microsoft ohne zusätzliche Kosten sicherlich das beste Produkt, um die Sicherheit des Betriebssystems zu erhöhen. Zwar gibt es auch hier Probleme mit der Absicherung gegen Manipulation von außen und die klassischen Tradeoffs zwischen Bequemlichkeit und gesteigerter Sicherheit, aber es handelt sich dennoch um eine komplette Lösung, die es mit kommerziellen Produkten leicht aufnehmen kann.
Was sich auch hier wieder gezeigt hat ist, dass ein weiteres wichtiges Instrument, sofern es ausreichend Personal gibt, um dieses zu überwachen, die Ereignisanzeige und damit die Systemlogs sind. Zwar können diese allein einen Angriff nicht verhindern, aber können einen Administrator oder eine Administratorin darauf hinweisen, dass auf den Systemen etwas passiert bei dem es sich um unerwartetes Verhalten handelt. Wenn man hier betrachtet, dass Angreifer, meist bevor sie Aktionen setzen, die eine solche Warnung auslösen könnten, das Unternehmen

im Vorfeld so weit erkunden, dass sie die Arbeits-/Logon-Zeiten von administrativen Benutzern kennen und einen Angriff möglichst so timen, dass er lange Unbemerkt bleibt. Das vergangene Jahr hat beispielsweise eine Häufung von Angriffen an Feiertagen oder langen Wochenenden gezeigt. Stellt auch eine perfekt eingestellte Überwachung relevanter Ereignisse keine Sicherheit für die rechtzeitige Erkennung da, wenn die Systembetreiber nicht über ein 24x7 SOC Team oder einen Externen SOC Betreiber verfügen, der diese Überwachung und rechtzeitige Alarmierung durchführt.

Abgesehen von den technischen Angriffen und Methoden darf aber auch die Gefahr durch die menschliche Komponente in der Betrachtung nicht übersehen werde. Neugier oder Vertrauen sind zwei Mittel, mit denen selbst eigentlich sicherheitsbewusste Mitarbeiter dazu gebracht werden können, eine Datei oder einen Link anzuklicken, der es in ihre, oder die Mailbox eines Kollegen, den, sie in Vertretung betreuen geschafft haben. Gleiches gilt für Daten auf USB-Sticks, die ihnen übergeben wurden oder die sie gefunden haben. Somit scheint es unumgänglich hier bereits Schutzschichten zu haben die, zuschlagen bevor ein Anwender oder eine Anwenderin die Chance haben, die Datei oder den Link anzuklicken. Beides ist mit Windows Bordmitteln ohne komplette Funktionalitäten nicht möglich zu deaktivieren. Abgesehen von der technischen Abwehr, darf aber auch nicht auf die Schulung der Anwender vergessen werden, hier ist besonders wichtig den Anwendern klarzumachen gegen den Impuls des Versteckens zu arbeiten. Sollten sie doch einmal das Gefühl haben etwas falsch gemacht zu haben und sich direkt in der IT melden, sodass sich fachkundige Personen das betroffene System mit äußerster Vorsicht ansehen können.

Generell kann man in der aktuellen Situation davon ausgehen, dass es keine absolute Sicherheit in IT-Systemen gibt. Dies ist zwar keine neue Erkenntnis, aber sie ist aktuell wahrer denn je. Wenn man bedenk, dass organisierte Cybercrime Banden in den vergangenen Jahren Milliarden mit Ransomware verdient haben, ist nicht davon auszugehen, dass die Zahlen und die Raffinesse der Angriffe in Zukunft abnehmen werden. Während LAPS beispielsweise lange ein wirklich vielversprechender Schutz war, bringen nun die ersten Angriffstools bereits die Möglichkeit mit, sofern ein berechtigter User kompromittiert wurde, die Verbindungsdaten direkt aus dem Active Directory auszulesen und sich so passend mit jedem Computer zu verbinden.

7.3 Ausblick

Mit diesem düsteren Ausblick ergeben sich mehrere potenzielle weitere Forschungsansätze.

Zum einen könnte man dieselbe Betrachtung um die Funktionen der Enterpriese Lizenz sowie der von Microsoft selbst angebotenen O365 Enterprise Mobility Suite erweitern und ein so geschütztes System echten Bedrohungen in einer isolierten Umgebung gegenüberstellen. Dies sollte die Effektivität der erweiterten Schutzkomponenten in den Bereichen zeigen, die vor allem im Bereich des besseren Exploit Schutzes sowie des verbesserten Ausführungsschutzes, den die Applocker Komponente mit sich bringt.

Zum anderen lässt sich die Betrachtung auch weg von reinen Microsoft Komponenten hin zu den diversen Herstellern am Markt der Sicherheitslösungen ausdehnen. Auf welchem es auf Grund

der vielfältigen Ansätze zur Lösung von bestehenden Sicherheitsproblemen einige Möglichkeiten zu vergleichenden Tests gibt.

ABKÜRZUNGSVERZEICHNIS

CVE - Common Vulnerabilities and Exposures
PUP – Potentially unwanted Program
MS – Microsoft
DDOS – Distributed Denial of Service
DOS – Denial of Service
IoT – Internet of Things
CERT – Cyber Emergency response Team
ASLR – Adress Space Layout Randomisation
DEP – Data Execution Prevention
UAC – User Account Control
EMET – Enhanced Mitigation Experience Toolkit
LAPS – Local Administrator Password Solution
EDR – Enterprise Detection and Response
LOLBINS – Living of the Land binary
WinRM – Windows Remote Management
COM – Component Object Model
WinAPI - Windows Application Programming Interface
BYOVD - Bring Your Own Vulnerable Driver
AAD – Azure Active Directory

ABBILDUNGSVERZEICHNIS

LITERATURVERZEICHNIS

Literaturverzeichnis

Aigner, R., Gebeshuber, K., Hackner, T., Kania, S., Kloep, P., Kofler, M. et al. (2020). *Hacking & Security. Das umfassende Handbuch* (1. Auflage, 4., korrigierter Nachdruck). Bonn: Rheinwerk Verlag.

Andreas Kroschel (WindowsPro, Hrsg.). (2011). *Datenausführungsverhinderung (DEP) konfigurieren oder abschalten | WindowsPro*. Zugriff am 20.09.2021.

Signals Directorate (Australien Goverment, Hrsg.). (2016). *Securing PowerShell in the Enterprise,* Signals Directorate. Zugriff am 24.06.2022. Verfügbar unter: https://www.cyber.gov.au/acsc/view-all-content/publications/securing-powershell-enterprise

AV-TEST GmbH (AV-TEST GmbH, Hrsg.). (2022). *Malware Statistic*. Zugriff am 10.06.2022. Verfügbar unter: https://www.av-test.org/de/statistiken/malware/

Bachelor, D., Sharkey, K. & Satran, M. (Microsoft, Hrsg.). (2021). *AlwaysInstallElevated*. Zugriff am 25.06.2022. Verfügbar unter: https://docs.microsoft.com/en-us/windows/win32/msi/alwaysinstallelevated

Baines, J. (Rapid 7, Hrsg.). (2021). *Driver-Based Attacks: Past and Present*. Zugriff am 24.06.2022. Verfügbar unter: https://www.rapid7.com/blog/post/2021/12/13/driver-based-attacks-past-and-present/

BlackDuck | OpenHub. (2019). *Mozilla Firefox Code Analysis*. Zugriff am 16.10.2019. Verfügbar unter: https://www.openhub.net/p/firefox/analyses/latest/languages_summary

Born, G. (2020). *Ransomware-Befall in Uniklinik Düsseldorf nicht für Todesfall verantwortlich*. Zugriff am 10.06.2022. Verfügbar unter: https://www.borncity.com/blog/2020/11/18/ransomware-befall-in-uniklinik-dsseldorf-nicht-fr-todesfall-verantwortlich/

Born, G. (2021a). *Windows 10: Achtung, Treibersignierung ändert sich 2021, Alt-Treiber nicht mehr nutzbar*. Zugriff am 10.06.2022. Verfügbar unter: https://www.borncity.com/blog/2021/01/06/windows-10-achtung-treibersignierung-ndert-sich-2021-alt-treiber-nicht-mehr-nutzbar/

Born, G. (Günter Born, Hrsg.). (2021b). *Windows 10: SAM-Zugriffsrechte ab 1809 nach Upgrade kaputt, Benutzerzugriff möglich*. Zugriff am 20.06.2022. Verfügbar unter: https://www.borncity.com/blog/2021/07/20/windows-10-sam-zugriffsrechte-ab-1809-nach-upgrade-kaputt-benutzerzugriff-mglich/

Bundesministerium für Inneres, B. (Hrsg.). (2021). *Cybercrime Report 2020. Lagebericht über die Entwicklung von Cybercrime*. Zugriff am 20.06.2022. Verfügbar unter: https://bundeskriminalamt.at/306/files/Cybecrime_2020_web.pdf

Bundesministerium für Inneres, B. (Hrsg.). (2022). *Cybercrime Report 2021. Lagebericht über die Entwicklung von Cybercrime*. Zugriff am 20.06.2022. Verfügbar unter: https://bundeskriminalamt.at/306/files/2022-222_Cybercrime_Report_2021_-_V20220621_1030_webBF.pdf

Carius, F. (MSXFAQ.de, Hrsg.). (2022). *AMSI - AntiMalware Scan Interface*. Zugriff am 10.06.2022. Verfügbar unter: https://www.msxfaq.de/windows/endpointsecurity/amsi_antimalware_scan_interface.htm

Chai, W. (techtarget.com, Hrsg.). (2020). *DEFINITION process hollowing*. Zugriff am 15.06.2022. Verfügbar unter: https://www.techtarget.com/whatis/definition/process-hollowing

Check Point Reserch (Hrsg.). (2021). *CYBER SECURITY REPORT 2021*. Zugriff am 15.06.2022. Verfügbar unter: https://www.checkpoint.com/downloads/resources/cyber-security-report-2021.pdf

Cloudflare. (2019). *What is a DDoS Attack? DDoS attacks are a primary concern in Internet security today. Explore details about how DDoS attacks function, and how they can be stopped.* Zugriff am 10.10.2019. Verfügbar unter: https://www.cloudflare.com/learning/ddos/what-is-a-ddos-attack/

CVE-Details (CVE-Details, Hrsg.). (2022a). *Vulnerabilities By Date*. Verfügbar unter: https://www.cvedetails.com/browse-by-date.php

CVE-Details (CVE-Details, Hrsg.). (2022b). *Windows 10 - Vulnarability Statistics*. Zugriff am 10.06.2022. Verfügbar unter: https://www.cvedetails.com/product/32238/Microsoft-Windows-10.html?vendor_id=26

CVE-Details (CVE-Details, Hrsg.). (2022c). *Windows 2016 Server - Vulnarability Statistics*. Zugriff am 10.06.2022. Verfügbar unter: https://www.cvedetails.com/product/34965/Microsoft-Windows-Server-2016.html?vendor_id=26

Dishan, F. (Microsoft, Hrsg.). (2021). *Step-by-Step Guide: How to Configure Microsoft Local Administrator Password Solution (LAPS)*. Zugriff am 23.06.2022. Verfügbar unter: https://techcommunity.microsoft.com/t5/itops-talk-blog/step-by-step-guide-how-to-configure-microsoft-local/ba-p/2806185

EMSI. (2012, 8. März). *Malware und Viren – was ist eigentlich der Unterschied?* Zugriff am 04.10.2019. Verfügbar unter: https://blog.emsisoft.com/de/3175/tec120308de/

Gerend, J., Ross, E., Parente, J. & Poggemeyer, L. (Microsoft, Hrsg.). (2021). *Work with Software Restriction Policies Rules*. Zugriff am 25.06.2022. Verfügbar unter: https://docs.microsoft.com/en-us/windows-server/identity/software-restriction-policies/work-with-software-restriction-policies-rules

Gibb, T. (How-To Geek, Hrsg.). (2017). *How to Disable Driver Signature Verification on 64-Bit Windows 8 or 10 (So That You Can Install Unsigned Drivers)*. Zugriff am 20.06.2022. Verfügbar unter: https://www.howtogeek.com/167723/how-to-disable-driver-signature-verification-on-64-bit-windows-8.1-so-that-you-can-install-unsigned-drivers/

Global Research & Analysis Team, Kaspersky Lab (Hrsg.). (2022). *A Bad Luck BlackCat*. Zugriff am 25.06.2022. Verfügbar unter: https://securelist.com/a-bad-luck-blackcat/106254/

Goude, N. (Microsoft, Hrsg.). (2012). *Use PowerShell to Decrypt LSA Secrets from the Registry*. Zugriff am 10.06.2022. Verfügbar unter: https://devblogs.microsoft.com/scripting/use-powershell-to-decrypt-lsa-secrets-from-the-registry/

Hadlington, L. (2018). *The “Human Factor” in Cybersecurity:. Exploring the Accidental Insider*. De Montfort University, UK. Zugriff am 15.10.2019. Verfügbar unter: https://pdfs.semanticscholar.org/cc2a/0622bc2c44f8e117c76dad9a68a12e70572d.pdf

Handelsverband (Hrsg.). (2022). *SICHERHEITS STUDIE 2021. BETRUG IM ONLINEHANDEL*. Zugriff am 20.06.2022. Verfügbar unter: https://www.handelsverband.at/fileadmin/content/images_publikationen/Studien/Sicherheitsstudie_2021/Sicherheitsstudie_2021_Web_Final.pdf

Harris Insight & Analytics LLC (Symantec, Hrsg.). (2021). *2021 NORTON CYBER SAFETY INSIGHTS REPORTGLOBAL RESULTS*. Zugriff am 20.06.2022. Verfügbar unter: https://now.symassets.com/content/dam/norton/campaign/NortonReport/2021/2021_NortonLifeLock_Cyber_Safety_Insights_Report_Global_Results.pdf

Hornet Security (Hrsg.).. *Was ist Emotet? Und wie kann ich mich schützen? Maßnahmen zum Schutz vor der gefährlichsten Schadsoftware der Welt*. Zugriff am 25.06.2022. Verfügbar unter: https://www.hornetsecurity.com/de/wissensdatenbank/emotet/?_adin=02021864894

Jeyashankar, A. (Socinvestigation.com, Hrsg.). (2022). *Most Common Antivirus Evasion and Bypass Techniques*. Zugriff am 10.06.2022. Verfügbar unter: https://www.socinvestigation.com/most-common-antivirus-evasion-and-bypass-techniques/

Kaspersky Labs (Kaspersky Labs, Hrsg.).. *Malware-Klassifizierungen*. Zugriff am 04.10.2019. Verfügbar unter: https://www.kaspersky.de/resource-center/threats/types-of-malware

Kaspersky Labs (Kaspersky Labs, Hrsg.).. *Was ist ein Adware?* Zugriff am 04.10.2019. Verfügbar unter: https://www.kaspersky.de/resource-center/threats/trojans

Kaspersky Labs (Kaspersky Labs, Hrsg.).. *Was ist ein Computervirus bzw. ein Computerwurm?* Zugriff am 04.10.2019. Verfügbar unter: https://www.kaspersky.de/resource-center/threats/viruses-worms

Kaspersky Labs (Kaspersky Labs, Hrsg.).. *Was ist ein Trojaner?* Zugriff am 04.10.2019. Verfügbar unter: https://www.kaspersky.de/resource-center/threats/trojans

Kaspersky Labs. (2018). *The Human Factor in IT Security: How Employees are Making Businesses Vulnerable from Within*. Zugriff am 15.10.2019. Verfügbar unter: https://www.kaspersky.com/blog/the-human-factor-in-it-security/

Knop, D. (Heise.de, Hrsg.). (2022). *Emotet-Botnet verstärkt Aktivitäten*. Zugriff am 25.06.2022. Verfügbar unter: https://www.heise.de/news/Emotet-Botnet-verstaerkt-Aktivitaeten-6340267.html

Krebs, B. (2016). *KrebsOnSecurity Hit With Record DDoS*. Zugriff am 10.10.2019. Verfügbar unter: https://krebsonsecurity.com/2016/09/krebsonsecurity-hit-with-record-ddos/

ManageEngine (Hrsg.). (2021). *The 8 most critical Windows security event IDs*. Verfügbar unter: https://download.manageengine.com/products/active-directory-audit/kb/the-eight-most-critical-windows-event-ids.pdf

McConnell, S. (2016). *Code complete* (2nd ed., 27th printing). Redmond, Wash.: Microsoft Press.

Metcalf, S. (ADSecurity.org, Hrsg.). (2015). *Sneaky Active Directory Persistence #14: SID History*. Zugriff am 10.06.2022. Verfügbar unter: https://adsecurity.org/?p=1772#:~:text=SID%20History%20is%20an%20attribute,effectively%20be%20cloned%20to%20another.

Microsoft.. *About Atom Tables*. Verfügbar unter: https://docs.microsoft.com/en-us/windows/win32/dataxchg/about-atom-tables?redirectedfrom=MSDN

Microsoft. (2019a). *What is a DLL?,* Microsoft. Zugriff am 12.10.2019. Verfügbar unter: https://support.microsoft.com/en-us/help/815065/what-is-a-dll

Microsoft (Hrsg.). (2019b). *Windows Sandbox*. Zugriff am 25.06.2022. Verfügbar unter: https://docs.microsoft.com/en-us/windows/security/threat-protection/windows-sandbox/windows-sandbox-overview

Microsoft (Hrsg.). (2022a). *Appendix L: Events to Monitor*. Zugriff am 25.06.2022. Verfügbar unter: https://docs.microsoft.com/en-us/windows-server/identity/ad-ds/plan/appendix-l--events-to-monitor

Microsoft (Hrsg.). (2022b). *BitLocker*. Zugriff am 10.06.2022. Verfügbar unter: https://docs.microsoft.com/en-us/windows/security/information-protection/bitlocker/bitlocker-overview

Microsoft (Hrsg.). (2022c). *BitLocker Countermeasures*. Zugriff am 24.06.2022. Verfügbar unter: https://docs.microsoft.com/en-us/windows/security/information-protection/bitlocker/bitlocker-countermeasures

Microsoft, Daniel Simpson (Mitarbeiter) (Microsoft, Hrsg.). (2022d). *Minimieren von Risiken mithilfe der Sicherheitsfeatures von Windows10*. Zugriff am 20.6.22. Verfügbar unter: https://docs.microsoft.com/de-de/windows/security/threat-protection/overview-of-threat-mitigations-in-windows-10#address-space-layout-randomization

Microsoft (Hrsg.). (2022e). *Mitigate threats by using Windows 10 security features*. Zugriff am 25.06.2022. Verfügbar unter: https://docs.microsoft.com/en-us/windows/security/threat-protection/overview-of-threat-mitigations-in-windows-10

Microsoft (Hrsg.). (2022, 2. Junif). *Windows Sandbox architecture*. Zugriff am 25:6:2022. Verfügbar unter: https://docs.microsoft.com/en-us/windows/security/threat-protection/windows-sandbox/windows-sandbox-architecture

Microsoft (Hrsg.). (2022, 2. Junig). *Windows Sandbox configuration*. Zugriff am 25.06.2022. Verfügbar unter: https://docs.microsoft.com/en-us/windows/security/threat-protection/windows-sandbox/windows-sandbox-configure-using-wsb-file

MITRE. (2001). *CVE-2000-0854*. Verfügbar unter: https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2000-0854

MITRE. (2019). *Privilege Escalation*. Zugriff am 10.10.2019. Verfügbar unter: https://attack.mitre.org/tactics/TA0004/

Mitre.org (Hrsg.). (2007). *CVE-2007-4560*. Verfügbar unter: https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2007-4560

Möhring, C. (Heise.de, Hrsg.). (2021). *Windows: Secure Boot - was ist das und wie kann ich es aktivieren?* Verfügbar unter: https://www.heise.de/tipps-tricks/Windows-Secure-Boot-was-ist-das-und-wie-kann-ich-es-aktivieren-6207260.html

Morgan, S. (Cybercrime Magazine, Hrsg.). (2020). *Cybercrime To Cost The World $10.5 Trillion Annually By 2025*. Zugriff am 20.06.2022. Verfügbar unter: https://cybersecurityventures.com/cybercrime-damages-6-trillion-by-2021/

National Cyber Security Centre (Hrsg.). (2018, 19. November). *Password administration for system owners. Password strategies that can help your organisation remain secure.* Zugriff am 15.06.2022. Verfügbar unter: https://www.ncsc.gov.uk/collection/passwords/updating-your-approach

Navarette, C., Jia, Y., Tennis, M. & Shao, R. (Palo Alto, Hrsg.). (2021, 8. März). *Attack Chain Overview: Emotet in December 2020 and January 2021*. Zugriff am 25.06.2022. Verfügbar unter: https://unit42.paloaltonetworks.com/attack-chain-overview-emotet-in-december-2020-and-january-2021/

PCI Security Standards Council (Hrsg.). (2022, März). *Payment Card Industry Data Security Standard. Requirements and Testing Procedures.* 4. Verfügbar unter: https://www.pcisecuritystandards.org/documents/PCI-DSS-v4_0.pdf

Schirmacher, D. (Heise.de, Hrsg.). (2022). *Emotet war kaputt, infiziert jetzt aber wieder vermehrt Windows-Computer*. Zugriff am 25.06.2022. Verfügbar unter: https://www.heise.de/news/Emotet-war-kaputt-infiziert-jetzt-aber-wieder-vermehrt-Windows-Computer-7064903.html

SentinelOne.. *AtomBombing Code Injection: Real Threat Or Just A Lot Of Hype?* Zugriff am 05.10.2020. Verfügbar unter: https://www.sentinelone.com/blog/atombombing-code-injection-threat-hype/#:~:text=What%20Is%20AtomBombing%20Code%20Injection,that%20store%20strings%20and%20identifiers).

Silva, L. & Froes, L. (Trend Micro, Hrsg.). (2022). *An Investigation of the BlackCat Ransomware via Trend Micro Vision One*. Zugriff am 24.06.2022. Verfügbar unter: https://www.trendmicro.com/en_us/research/22/d/an-investigation-of-the-blackcat-ransomware.html

Sommergut, W. (WindowsPro, Hrsg.). (2019). *Constrained Language Mode: PowerShell-Risiken entschärfen*. Zugriff am 24.06.2022. Verfügbar unter: https://www.windowspro.de/wolfgang-sommergut/constrained-language-mode-powershell-risiken-entschaerfen.

Sophos (Sophos, Hrsg.). (2021). *Ransomware-Report 2021*. Verfügbar unter: https://assets.sophos.com/X24WTUEQ/at/cpcppqng7r4fzb9s8qkws37/sophos-state-of-ransomware-2021-wpde.pdf

Spitzner, L. (SANS, Hrsg.). (2019). *Time for Password Expiration to Die. Password expiration is a dying concept. Essentially, it's when an organization requires their workforce to change their passwords every 60, 90 or…* Verfügbar unter: https://www.sans.org/blog/time-for-password-expiration-to-die/

Statcounter (Statcounter, Hrsg.). (2019). *Desktop Windows Version Market Share Worldwide. September 2019,* Statcounter. Zugriff am 04.10.2019. Verfügbar unter: https://gs.statcounter.com/windows-version-market-share/desktop/worldwide/

Steward, A. (2014). *DLL SIDE-LOADING:. A Thorn in the Side of the Anti-Virus Industry,* FireEye. Zugriff am 11.10.2019. Verfügbar unter: https://www.fireeye.com/content/dam/fireeye-www/global/en/current-threats/pdfs/rpt-dll-sideloading.pdf

Stouffer, C. (Symantec, Hrsg.). (2021). *115 cybersecurity statistics and trends you need to know in 2021*. Zugriff am 20.06.2022. Verfügbar unter: https://us.norton.com/internetsecurity-emerging-threats-cyberthreat-trends-cybersecurity-threat-review.html

Sutherland, S. (NetSPI, Hrsg.). (2014). *15 Ways to Bypass the PowerShell Execution Policy*. Zugriff am 24.06.2022. Verfügbar unter: https://www.netspi.com/blog/technical/network-penetration-testing/15-ways-to-bypass-the-powershell-execution-policy/

Thomas Dullien (CVE, Hrsg.). (2018). *CVE-2018-0986*. Zugriff am 05.10.2020. Verfügbar unter: https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2018-0986

Tudor, D. (2021, 3. September). What Is Data Execution Prevention (DEP)? *Heimdal Security*. Zugriff am 20.09.2021. Verfügbar unter: https://heimdalsecurity.com/blog/dep-data-execution-prevention-windows/

Westerhagen, O. von (Heise.de, Hrsg.). (2021, 27. Januar). *Emotet: Strafverfolger zerschlagen Malware-Infrastruktur*. Zugriff am 25.06.2022. Verfügbar unter: https://www.heise.de/news/Emotet-Strafverfolger-zerschlagen-Malware-Infrastruktur-5038233.html

Wheeler, S., Chase, W. & Schonning, N. (Microsoft, Hrsg.). (2021). *Security Considerations for PowerShell Remoting using WinRM*. Zugriff am 23.06.2022. Verfügbar unter: https://docs.microsoft.com/en-us/powershell/scripting/learn/remoting/winrmsecurity?view=powershell-7.2

Wheeler, S., Chase, W. & Vasin, S. (Microsoft, Hrsg.). (2022). *Get-EventLog*. Zugriff am 25.05.2022. Verfügbar unter: https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.management/get-eventlog?view=powershell-5.1

Wheeler, S. & Lombardi, M. (Microsoft, Hrsg.). (2022a). *about_Execution_Policies*. Verfügbar unter: https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.core/about/about_execution_policies?view=powershell-7.2

Wheeler, S. & Lombardi, M. (Microsoft, Hrsg.). (2022b). *about_Language_Modes*. Zugriff am 23.06.2022. Verfügbar unter: https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.core/about/about_language_modes?view=powershell-7.2

Yiftach, K. (Cynet, Hrsg.). (2020). *What Are LOLBins and How Do Attackers Use Them in Fileless Attacks?* Zugriff am 16.06.2022. Verfügbar unter: https://www.cynet.com/attack-techniques-hands-on/what-are-lolbins-and-how-do-attackers-use-them-in-fileless-attacks/