Aufgabe 1

1.1

26^14 = 6,451 * 10^19 Passwörter

6,451 * 10^19 / 200 pw/sec = 3,225 * 10^17 sec = 8,960 * 10^13 h

1.2

Wenn der Angreifer die Angaben aus der Aufgabe hat könnte er für einen Acc die 1000 Passwörter druchprobieren und wenn es keinen Treffer gibt direkt zum nächsten Account gehen.

Für 1 Acc bräuchte er 1000 pw / 200 pw/s = 5 sec.

Wenn 90% eines dieser Passwörter verwenden ist er in 9 von 10 Fällen nach spätestens 5 sec erfolgreich und muss dementsprechend noch 10 weitere Accounts probieren, also insgesamt 110 Versuche * 5s = 550s

1.3

Die Wahrscheinlichkeit, dass der Angreifer ein „Sicheres“ Passwort knackt (eines welches nicht auf der Liste ist) ist im Vergleich zum anderen Fall verschwindend gering. Daher betrachte ich nur den Worst Case(Für den „Verteidiger“), d.h. dass der Angreifer Versucht einen Acc zu knacken der eine schwaches Passwort verwendet. Um zu 99% erfolglos zu bleiben darf man ihm Maximal 10 Versuche gewähren.

1.4

Passwörter: 6,451 * 10^19

50% -> 3,225 * 10^19

2 Jahre = 63 244 800 Sekunden

3,225 * 10^19 / 63 244 800 Sekunden = 5,099 * 10^11 pw/s

Wenn ein Salt für alle Benutzer verwendet wird und dieses dem Angreifer bekannt ist sollte das keine Auswirkung auf die Geschwindigkeit haben (Außer das Hashen mit Salt ist an sich schon aufwändiger). Wenn es für jeden Nutzer ein eigenes Salt verwendet wird und der Angreifer nicht weiß welches Salt zu welchem Nutzer gehört müsste er alles Passwörter mit alles Salts hashen, was die Anzahl der hash Operationen von 6,451 * 10^19 auf (6,451 * 10^19)^2 erhöhen würde.

Aufgabe 2

Ein Python Skript hat den Output von Accesschk auf das Verzeichnis und die Rechte des Admins in diesem Verzeichnis reduziert (Falls keine Rechte angegeben waren wird „Keine Berechtigung“ eingefügt)

Ein weiteres Python Skript hat anschließend ausgewertet auf wie viele Verzeichnisse (Unterhalb der Verzeichnisse auf der ersten Ebene) der Admin nur RW/R oder gar keinen Zugriff hat.

Diesen Vorgang habe ich anschließend für die Verzeichnisse unter C:\Windows wiederholt

Ich konnte das Skript bisher leider nicht so erweitern, dass es selbstständig „tiefer“ geht, daher konnte ich nicht alle Verzeichnisse vollautomatisch auswerten.

Des Weiteren bin ich mir nicht sicher, wie Zuverlässig Accesschk ist. Einige Verzeichnisse die ich mir genauer angeschaut habe wiesen im File Explorer andere Rechte auf. In einigen konnte ich schreiben, obwohl ich angeblich(laut accesschk) nur Leserechte haben sollte. Außerdem bin ich mir nicht sicher, ob der Admin Nutzer die richtige Wahl war. Allerdings haben für meinen eigenen Benutzer regelmäßig die Einträge komplett gefehlt, weshalb ich damit keine Sinnvolle Auswertung vornehmen konnte.

2.2

Nur Lesezugriff auf:

C:\Programm Files\Windows Media Player\Icons

C:\Users\Default\AppData\Local\Microsoft\WindowsApps

Auf ca 50% Aller Directories im Windows Verzeichnis

Darin kaum Zugriff auf Verzeichnisse unter

C:\Windows\WinSxS (lediglich 82 von 70k)

C:\Windows\System32(Bei ca 200 gar kein Zugriff, 80 nur Lesend)

C:\Windows\rescache

Kein (Schreib)zugriff auf einzelne Verzeichnisse unter C:\Windows\

- Globalisation
- IME
- INF
- PrintDialouge
- Servicing
- Speech

2.3 Zu den Schreibbaren Verzeichnissen mit Ausführbaren Dateien gehören unter anderem System32,

WinSxS/Package_for_RollupFix~31bf3856ad364e35~amd64~~19041.2251.1.11\wow64_microsoft-windows-security-spp-clientext_31bf3856ad364e35_10.0.19041.1682_none_4bb88fb09acf494f

servicing/LCU

2.4

Ausführbare Dateien in Verzeichnissen mit nur Leserecht:

C:\Windows\WinSxS\

x86_netfx35linq-system.data.services_31bf3856ad364e35_10.0.19041.1_none_492ca01caacb396b

Assembly

Microsoft.NET

Verschiedene amd64 Verzeichnisse

2.5

Da die Standard Windows Verzeichnisse ein regelrechtes Labyrinth darstellen bin ich der Meinung, dass es nicht sinnvoll ist manuell Verzeichnisse zu whitelisten. Statdessen wäre es sinnvoller eine standard WDAC Policy zu verwenden, die alle nativen Windows Apps zulässt und den Rest blockiert. Anschließend kann man ein seperates Verzeichnis für manuell installierte Anwendungen erstellen und dieses Whitelisten.

Hier ein Ausschnitt aus der Auswertung:

(Ober)Verzeichnis, RW, R, Keine, Gesamt

['$WINDOWS.~BT', 3, 0, 0, 3]

['apache-ant-1.10.9', 101, 0, 0, 101]

['AStyle', 43, 0, 0, 43]

['BIOS', 3, 0, 0, 3]

['Checkstyle', 1, 0, 0, 1]

['Drivers', 3, 0, 0, 3]

['ESD', 1, 0, 0, 1]

['Gaomon Tablet', 40, 0, 0, 40]

['jdk-15', 87, 0, 0, 87]

['jerm', 5, 0, 0, 5]

['karel', 25, 0, 0, 25]

['Notepad++', 16, 0, 0, 16]

['PerfLogs', 1, 0, 0, 1]

['pipe', 242, 0, 0, 242]

['prog1-uebungen', 42, 0, 0, 42]

['Program Files', 9235, 3193, 1, 12429] C:\Program Files\ModifiableWindowsApps

['Program Files (x86)', 2025, 1, 0, 2026] C:\Program Files\Windows Media Player\Icons

['spotbugs-4.1.3', 9, 0, 0, 9]

['Users', 62705, 1, 0, 62706]
C:\Users\Default\AppData\Local\Microsoft\WindowsApps

['Windows', 125122, 70216, 241, 195579]

Angeblich nur R User Ordner:

C:\Users\Default\AppData\Local\Microsoft\WindowsApps

RW NT SERVICE\TrustedInstaller

R VORDEFINIERT\Administratoren

R NT-AUTORITÄT\SYSTEM

R VORDEFINIERT\Benutzer

R ZERTIFIZIERUNGSSTELLE FÜR ANWENDUNGSPAKETE\ALLE ANWENDUNGSPAKETE

R ZERTIFIZIERUNGSSTELLE FÜR ANWENDUNGSPAKETE\ALLE EINGESCHRÄNKTEN ANWENDUNGSPAKETE

Angeblich nur R Zugriff auf folgende Programm Files:

C:\Program Files\Windows Media Player\Icons

RW NT SERVICE\TrustedInstaller

R NT-AUTORITÄT\SYSTEM

R VORDEFINIERT\Administratoren

R VORDEFINIERT\Benutzer

Windows:

['addins', 1, 0, 0, 1]

['appcompat', 8, 0, 0, 8]

['apppatch', 7, 0, 0, 7]

['AppReadiness', 1, 0, 0, 1]

['assembly', 1555, 0, 0, 1555]

['bcastdvr', 1, 0, 0, 1]

['Boot', 0, 86, 0, 86]

['Branding', 4, 0, 0, 4]

['CbsTemp', 1, 0, 0, 1]

['Containers', 2, 0, 0, 2]

['Cursors', 1, 0, 0, 1]

['de-DE', 1, 0, 0, 1]

['debug', 2, 0, 0, 2]

['diagnostics', 0, 46, 0, 46]

['DiagTrack', 3, 0, 0, 3]

['DigitalLocker', 2, 0, 0, 2]

['Downloaded Program Files', 1, 0, 0, 1]

['en-US', 1, 0, 0, 1]

['Firmware', 2, 0, 0, 2]

['Fonts', 1, 0, 0, 1]

['GameBarPresenceWriter', 1, 0, 0, 1]

['Globalization', 6, 4, 0, 10]----------

['Help', 15, 0, 0, 15]

['IdentityCRL', 3, 0, 0, 3]

['IME', 13, 0, 0, 13]

['ImmersiveControlPanel', 9, 1, 0, 10]----------

['INF', 95, 1, 0, 96]----------

['InputMethod', 4, 0, 0, 4]

['L2Schemas', 1, 0, 0, 1]

['Lenovo', 10, 0, 0, 10]

['LiveKernelReports', 3, 0, 0, 3]

['Logs', 11, 0, 1, 12]

['Media', 14, 0, 0, 14]

['Microsoft.NET', 1342, 0, 0, 1342]

['Migration', 2, 0, 0, 2]

['Minidump', 1, 0, 0, 1]

['ModemLogs', 1, 0, 0, 1]

['OCR', 0, 2, 0, 2]

['Offline Web Pages', 1, 0, 0, 1]

['Panther', 6, 0, 0, 6]

['PCHEALTH', 4, 0, 0, 4]

['Performance', 3, 0, 0, 3]

['PLA', 7, 0, 0, 7]

['PolicyDefinitions', 3, 0, 0, 3]

['Prefetch', 2, 0, 0, 2]

['PrintDialog', 5, 1, 0, 6]----------

['Provisioning', 7, 0, 0, 7]

['Registration', 2, 0, 0, 2]

['rescache', 9, 159, 0, 168]----------

['Resources', 10, 0, 0, 10]

['SchCache', 1, 0, 0, 1]

['schemas', 3, 4, 0, 7]----------

['security', 8, 0, 0, 8]

['ServiceProfiles', 184, 0, 12, 196]

['ServiceState', 4, 8, 0, 12]----------

['servicing', 116069, 10, 0, 116079]----------

['Setup', 2, 0, 0, 2]

['ShellComponents', 1, 0, 0, 1]

['ShellExperiences', 1, 0, 0, 1]

['SHELLNEW', 1, 0, 0, 1]

['SKB', 2, 0, 0, 2]

['SoftwareDistribution', 29, 0, 0, 29]

['Speech', 6, 4, 0, 10]----------

['Speech_OneCore', 10, 0, 0, 10]

['System', 1, 1, 0, 2]

['System32', 4580, 29, 224, 4833]----------

['SystemApps', 428, 0, 0, 428]

['SystemResources', 72, 13, 0, 85]

['SystemTemp', 1, 0, 0, 1]

['SysWOW64', 364, 19, 4, 387]

['TAPI', 1, 0, 0, 1]

['Tasks', 1, 0, 0, 1]

['Temp', 5, 0, 0, 5]

['TempInst', 7, 0, 0, 7]

['tracing', 1, 0, 0, 1]

['twain_32', 5, 0, 0, 5]

['Vss', 4, 0, 0, 4]

['WaaS', 0, 3, 0, 3]

['Web', 9, 0, 0, 9]

['WinSxS', 82, 69824, 0, 69906]----------