Lithnet Password Protection for Active Directory

Schlüsseldienst

von Dr. Christian Knermann

Die Sicherheit einer Active-Directory-Infrastruktur steht und fällt mit den Passwörtern der Benutzerkonten. Das freie Lithnet Password Protection for Active Directory sorgt für flexiblere Regeln, als sie mit Gruppenrichtlinien allein möglich wären, und beugt der Verwendung bereits kompromittierter Passwörter vor. Wir zeigen Inbetriebnahme und Einsatz.

Die Multifaktor-Authentifizierung (MFA) gilt als Stand der Technik zur Absicherung von Benutzerkonten. Diese Erkenntnis ist längst auch bei weniger IT-affinen Nutzern angekommen, nachdem zahlreiche Onlinedienste MFA-Verfahren anbieten oder sogar erzwingen. Ein Dienst, dem viele Anwender tagtäglich gegenüberstehen, unterstützt indes auch mit dem Windows Server 2022 ohne Weiteres nur die althergebrachte Methode mittels Benutzername und Passwort. Gemeint ist die Anmeldung an Clientcomputern, die einer Active-Directory-Domäne angehören. Umso wichtiger sind in diesem Fall sichere Passwörter.

Länge versus Komplexität

Was ein sicheres Passwort ausmacht und wie oft es gewechselt werden sollte, ist weltweit Gegenstand eifriger Diskussionen unter IT-Sicherheitsexperten. Einigkeit herrscht darüber, dass Komplexität und Länge die entscheidenden Faktoren sind. Das Bundesamt für Sicherheit in der Informationstechnik (BSI) setzt in seinen Handreichungen beide Faktoren in Bezug zueinander [1].

So empfiehlt das BSI bei kurzen Passwörtern mit lediglich acht bis zwölf Zeichen Länge eine hohe Komplexität. Die besteht in der Verwendung von typischerweise vier Zeichenarten, wie sie vielen Anwendern vertraut sein dürften, einer Mischung aus großen und kleinen Buchstaben, Zahlen sowie Sonderzeichen. Die Empfehlungen belohnen weiterhin Länge mit geringeren Anforderungen an die Komplexität. Wer sich für ein deutlich längeres Passwort mit 20 bis 25 Zeichen entscheidet, soll etwa nur noch zwei der vier genannten Anforderungen an die Komplexität erfüllen müssen.

Dienste wie die Webseite "Wie sicher ist mein Passwort?" demonstrieren spielerisch den mathematischen Hintergrund [2]. Mit der Länge steigt der Rechenaufwand, um ein Passwort zu knacken, exponentiell und ist ab 15 bis 20 Zeichen zumindest mit heutiger Technik nicht mehr in endlicher Zeit zu bewältigen. Die Länge allein hilft allerdings nicht, falls ein Passwort für Wörterbuchattacken anfällig ist oder sich auf einer Liste bereits kompromittierter Passwörter wiederfindet. Ein gewisser Grad an Komplexität ist also auch bei langen Phrasen angeraten.

Gruppenrichtlinien unflexibel

Die Bordmittel, die Microsoft dem AD in den Gruppenrichtlinien mitgibt, sind nicht vollends geeignet, die vorangegangenen Überlegungen umzusetzen. Sie finden die Voreinstellungen zur Passwortsicherheit im GPO der "Default Domain Policy" und dort unter "Computerkonfiguration / Richtlinien / Windows-Einstellungen / Sicherheitseinstellungen / Kennwortrichtlinien". Ein frisch installierter Domaincontroller (DC) unter Windows Server 2022 setzt das maximale Kennwortalter auf 42 Tage. Das BSI hatte sich dagegen bereits in der letztjährigen Ausgabe seines Grundschutz-Kompendiums davon verabschiedet, einen regelmäßigen Wechsel des Passworts zu empfehlen. Sie können entsprechend abwägen, diese Frist zu verlängern oder ganz abzuschaffen.

Sollten Sie einen periodischen Wechsel bevorzugen, definiert die Default Domain Policy weiterhin ein minimales Kenn-wortalter von einem Tag. Diese Einstellung soll verhindern, dass findige Anwender ihr Passwort mehrfach kurz hintereinander wechseln, um wieder den ursprünglichen Wert zu erhalten. Dem beugt das System weiterhin vor, indem es sich pro Benutzer die Chronik der letzten 24 verwendeten Passwörter merkt.

Eher unflexibel erweisen sich die Einstellungen der Gruppenrichtlinien im Hinblick auf Komplexität und Länge. Letztere setzt Microsoft unter Windows Server 2022 ab Werk auf mindestens sieben Zeichen fest und aktiviert zusätzlich die Ein-

stellung "Kennwort muss Komplexitätsvoraussetzungen entsprechen". Dies sorgt dafür, dass Passwörter drei von vier Zeichenarten enthalten müssen – Großbuchstaben, Kleinbuchstaben, Zahlen und Sonderzeichen. Ein variables System, das Nutzer besonders langer Passwörter mit weniger Anforderungen an die Komplexität belohnt, können die Gruppenrichtlinien von Haus aus nicht abbilden. Doch mit den "Password Filters" bietet Microsoft Drittanbietern eine Schnittstelle, um Funktionen nachzurüsten [3].

Sicherer mit Lithnet

An diese Schnittstelle dockt die Lithnet Password Protection for Active Directory (LPP) an, ein Modul zur Installation auf Domaincontrollern, das der Hersteller quelloffen und kostenfrei veröffentlicht hat [4]. LPP bringt nicht nur Einstellungen mit, die die Komplexität in Bezug zur Länge eines Passworts setzen, sondern vergleicht auf Wunsch Passwörter auch mit einer vom Admin gepflegten Liste gesperrter Wörter sowie der Datenbank des Dienstes "Have I been pwned?" (HIBP) [5]. Das Wort "pwned" steht im englischen Sprachraum umgangssprachlich für "owned". Frei übersetzt meint HIBP also "Wurde ich erwischt?". Der Dienst beantwortet entsprechend die Frage, ob ein bestimmtes Passwort sich bereits auf im Internet kursierenden Listen gestohlener Zugangsdaten wiederfindet.

Sie konfigurieren LPP mittels Gruppenrichtlinien. Die Passwortdatenbank liegt lokal, sodass das Ganze ohne Onlinezugang funktioniert. Weder Passwörter noch Hash-Werte verlassen Ihr internes Netz. Voraussetzung für die Prüfung von Passwörtern gegen die Liste von HIBP ist, dass Sie zunächst diese im Format "NTLM / ordered by hash" herunterladen und extrahieren. Das dauert eine Weile, denn die Liste belegt gepackt 8,5 GByte und ausgepackt sogar 20 GByte. Diesen Platz benötigen Sie aber nur temporär. Sobald Sie die Liste in das Datei-basierte Datenbankformat von LPP einlesen, schrumpft der Platzbedarf wieder auf 6 GByte.

HIBP-Datenbank bereitstellen

Setzen Sie mehrere DCs ein, was im produktiven Betrieb grundsätzlich zu empfehlen ist, dann müssen Sie entscheiden, wo Sie die Datenbank mit den kompromittierten Passwörtern speichern wollen. Lithnet stellt eine von drei Strategien zur Wahl: eine gemeinsam genutzte Ablage in Form einer Dateifreigabe, die lokale Ablage auf jedem einzelnen DC mit manueller Replikation per Robocopy oder XCopy und zu guter Letzt die lokale Ablage mit automatischer Replikation über Microsofts Distributed File System (DFS).

Da sich die Liste von HIBP nur selten ändert, ist für kleinere Umgebungen die manuelle Replikation ein gangbarer Weg. Der Hersteller empfiehlt gerade für geografisch verteilte Infrastrukturen DFS, verweist aber explizit darauf, dass Sie nicht die bereits vorhandene Replikation der SYSVOL-Freigabe zweckentfremden, sondern eine dedizierte DFS-Replikationsgruppe erstellen müssen.

Haben Sie entschieden, wo die Datenbank liegen soll, laden Sie das LPP-Installationspaket herunter und installieren die Software im Kontext eines Domänen-Administrators mit allen Bestandteilen auf jedem DC. Im zweiten Dialogschritt konfigurieren Sie den Pfad zur Datenbank. Die Installation erfordert einen Neustart, damit der Passwortfilter aktiv werden kann. Anschließend nutzen Sie die PowerShell, um auf einem DC die HIBP-Liste in eine LPP-Datenbank zu konvertieren. Lithnet liefert dazu ein PowerShell-Modul mit, das uns auch später noch nützlich sein wird:

```
Import-Module
  LithnetPasswordProtection

Open-Store 'C:\Program Files\
  Lithnet\Active Directory Password
  Protection\Store'

Import-CompromisedPasswordHashes
  -Filename C:\temp\pwned-passwords-
  ntlm-ordered-by-hash-v8.txt
```

Gruppenrichtlinien vorbereiten

Während dieser Prozess läuft, nutzen Sie die Zeit, um weitere Vorbereitungen zu treffen. Verwenden Sie in Ihrer Domäne eine zentrale Ablage für Gruppenrichtlinienvorlagen, wie es bei mehreren DCs die beste Wahl ist, dann kopieren Sie auf einem DC mit installierter LPP die beiden ADMX-Vorlagen "lithnet.activedirectory.passwordfilter.admx" und "lithnet.admx" aus dem lokalen Pfad "C:\Windows\PolicyDefinitions" an den zentralen Speicherort unter "\\<Domain-Name> \ SYSVOL \ <Domain-Name> \ Policies \ PolicyDefinitions". Die zugehörigen ADML-Sprachdateien befördern Sie in die passenden Unterordner, mindestens in den Ordner "en-US".

Öffnen Sie nun die Gruppenrichtlinienverwaltung und schalten Sie in der "Default Domain Policy" Microsofts Vorgaben zu Länge und Komplexität der Passwörter ab, denn darum wird sich gleich LPP kümmern. Nun erstellen Sie ein neues Gruppenrichtlinienobjekt (Group Policy Object, GPO), das Sie auf oberster Ebene der Domäne oder mit der OU der Domaincontroller verknüpfen.

Öffnen Sie das neue GPO daraufhin im Gruppenrichtlinieneditor und navigieren Sie zum Pfad "Computerkonfiguration / Richtlinien / Administrative Vorlagen… / Lithnet / Password Protection for Ac-tive Directory / Default Policy". In diesem Ordner finden Sie sämtliche Einstellungen zur Konfiguration der LPP (Bild 1). Aktivieren Sie hier im ersten Schritt die Einstellung "Reject passwords found in the compromised password store" und setzen Sie bei den Optionen beide Haken bei "Enable for password set operations" sowie "Enable for password change operations".

Wenig aussagekräftige Meldungen bleiben erhalten

Warten Sie die Aktualisierung der Gruppenrichtlinien auf allen DCs ab oder erzwingen Sie diese per gpupdate /force. Im nächsten Schritt versuchen Sie, im Kontext eines beliebigen Benutzers an einem Clientcomputer ein Passwort zu setzen, das sich auf der schwarzen Liste von HIBP befindet, wie etwa "P4ssw0rd!".

Daraufhin sollten Sie zweierlei feststellen: Zum einen, dass LPP grundsätzlich funktioniert und die Änderung verweigert, zum anderen aber auch, dass Windows dies mit der üblichen und wenig aussagekräftigen Standardmeldung quittiert.

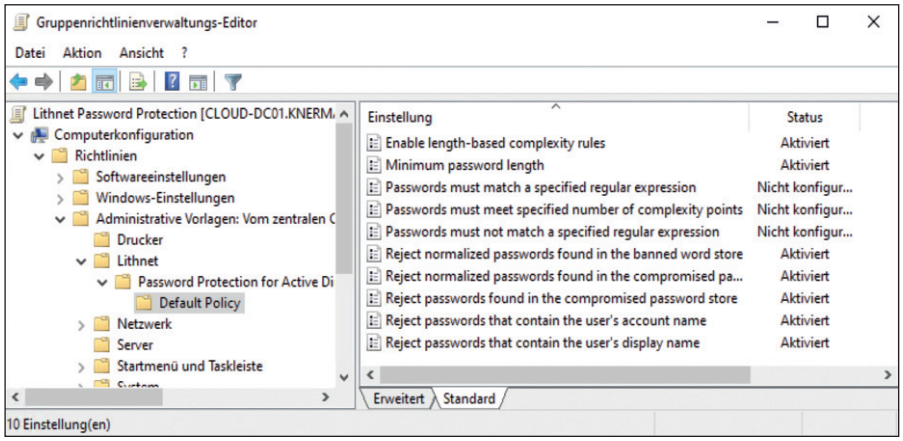

Bild 1: Die LPP-Gruppenrichtlinien erlauben deutlich feinere Einstellungen als Microsofts Bordmittel.

Während LPP also unter der Haube detaillierte Anforderungen an Passwörter ermöglicht, kann es leider nicht das Verhalten der Clients beeinflussen. Endanwender erfahren also nicht, woran genau die Änderung ihres Passworts gescheitert ist. Den Grund finden nur Administratoren, und zwar im Ereignisprotokoll "Anwendung" des DC, der den Versuch der Passwortänderung verarbeitet hat. Hier protokolliert LPP im Detail, welche Einstellung der Gruppenrichtlinien gegriffen hat. Alle Ereignis-IDs, die Ihnen hier begegnen können, hat der Hersteller dokumentiert [6].

Die beiden Einstellungen "Reject passwords that contain the user's account / display name" verhindern unabhängig von Länge und Komplexität, dass Anwender ihren eigenen Namen ins Passwort einbauen. Freunde regulärer Ausdrücke dürfen sich zudem über zwei Einstellungen freuen, die Passwörter anhand frei definierbarer Regeln zulassen oder ablehnen.

Die Option "Passwords must meet specified number of complexity points" bietet Ihnen ein individuell konfigurierbares Punktesystem für die Komplexität. So legen Sie eine minimale Punktezahl fest, die ein Passwort erreichen muss, sowie eine Anzahl von Punkten pro Großbuchstaben, Kleinbuchstaben, Zahl und Sonderzeichen. Ein Anwender sammelt dann etwa einen Punkt pro Großbuchstaben, zwei für jede Zahl und drei für ein Sonderzeichen. LPP akzeptiert das Passwort nur, wenn es die minimale Gesamtzahl an Punkten erreicht.

Individuelle Wörter sperren

Noch schärfer wird der Filter von LPP, wenn Sie die Einstellungen "Reject normalized passwords found in the compromised password store" und "Reject normalized passwords found in the banned word store" aktivieren. Damit letztere Einstellungen greifen, müssen Sie noch die Datenbank zu blockierender Wörter mit naheliegenden Begriffen füttern. Dazu nutzen Sie:

```
Import-Module
  LithnetPasswordProtection

Add-BannedWord <Knermann>
```

Möchten Sie eine größere Menge an Begriffen ausschließen, müssen Sie diese nicht mühsam von Hand eingeben. Mit dem Import-BannedWords-Cmdlet lesen Sie alternativ ganze Wörterbücher ein. Das Cmdlet erwartet hierbei eine Textdatei, die jeweils ein zu sperrendes Wort pro Zeile enthält.

Normalisierung erhöht den Schwierigkeitsgrad

Gibt ein Benutzer nun ein neues Passwort ein, normalisiert LPP dieses vor dem Abgleich mit den schwarzen Listen. Normalisierung bedeutet, dass der Filter ein Passwort zunächst komplett in Kleinbuchstaben umwandelt, außerdem Leerzeichen sowie Zahlen und Sonderzeichen am Anfang und Ende entfernt. Weiterhin versteht sich LPP auch auf die im Internet beliebte "Leetspeak", also das möglichst kreative Austauschen von Buchstaben durch ähnlich aussehende Zahlen und Sonderzeichen. Den String "Kn3rm@nn!" normalisiert der Filter folglich zu "knermann" und lehnt die Passwortänderung ab.

Von der Wirkungsweise der Filterregeln können Sie sich nicht nur durch Versuch und Irrtum an einem Clientcomputer überzeugen, sondern auch per PowerShell. Mithilfe des Get-PasswordFilterResult-Cmdlets testen Sie Passwörter gegen Ihr Regelwerk und erhalten sofort eine aussagekräftige Rückmeldung, ob und warum LPP ein bestimmtes Passwort ablehnt. Ein von uns getestetes Passwort aus dem Titel dieses Artikels hält in unserer Beispielkonfiguration der Prüfung stand (Bild 2).

BSI-Empfehlungen umsetzen

Aktivieren Sie die Einstellung "Minimum password length" und setzen Sie eine Mindestlänge von acht Zeichen fest. Einen direkten Bezug zwischen Länge und Komplexität stellen Sie nun mit der Option "Enable length-based complexity rules" her. Hier dürfen Sie bis zu drei Schwellenwerte für die Länge mit jeweils unterschiedlichen Anforderungen an die Komplexität definieren. Sie können so die Empfehlung des BSI umsetzen und Benutzer für besonders lange Passwörter mit weniger strikten Vorgaben im Hinblick auf deren Komplexität belohnen.

Legen Sie etwa den ersten Schwellenwert auf 13 Zeichen fest. Im Feld darunter definieren Sie, wie viele der vier Anforderungen – Großbuchstaben, Kleinbuchstaben, Zahlen, Sonderzeichen – ein Kennwort mit weniger Zeichen erfüllen muss. Exemplarisch wählen wir hier mit vier von vier den höchsten Wert. Alternativ können Sie mit Checkboxen genau festlegen, welchen der Anforderungen das Kennwort genügen muss.

Den zweiten Schwellenwert setzen Sie auf 20 Zeichen und legen fest, dass Passwörter mit weniger Zeichen noch drei der vier Anforderungen abdecken müssen. Der unterste Bereich der Konfiguration gilt für alle Passwörter, die länger als der zweite Schwellwert sind. Verlangen Sie in diesem Fall noch zwei unserer vier Anforderungen und wenden Sie die Richtlinie an, setzt LPP den Kennwortfilter passend zu den Empfehlungen des BSI in Kraft.

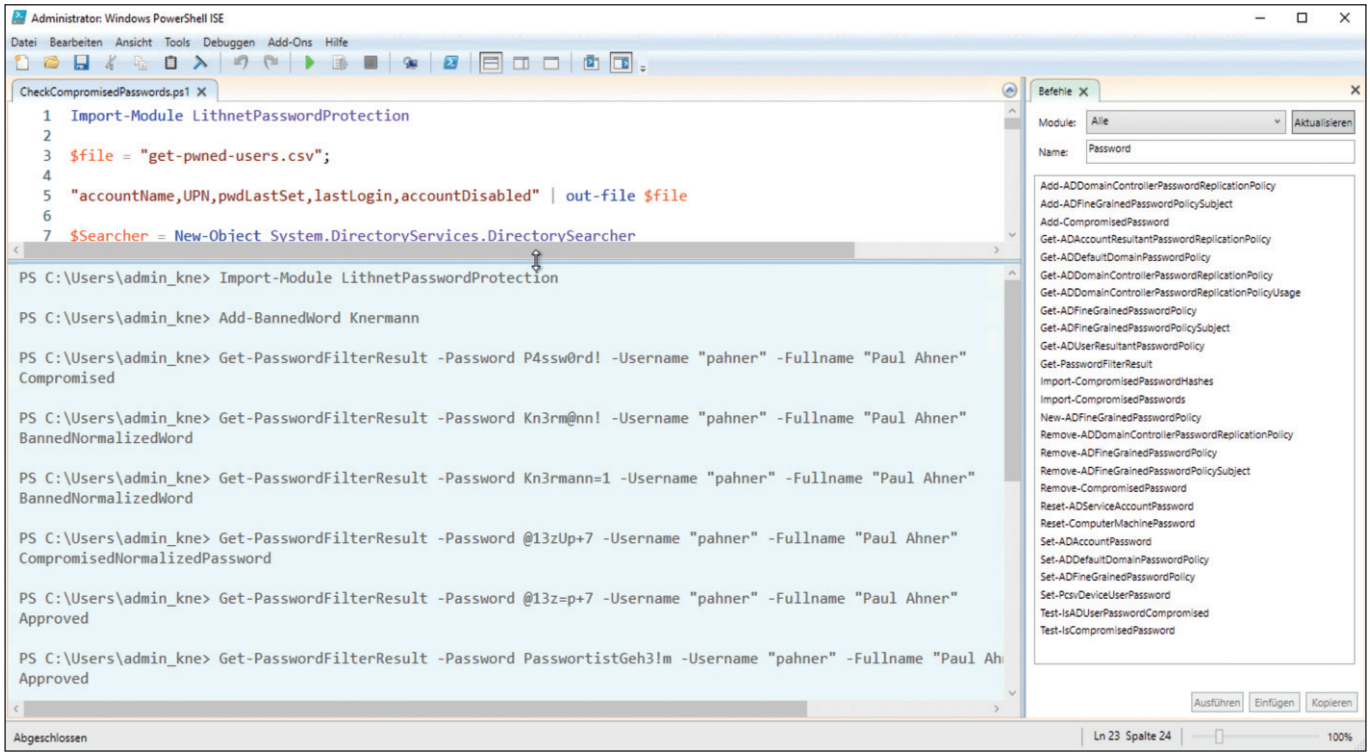

Bild 2: Die Cmdlets des Lithnet-PowerShell-Moduls helfen beim Überprüfen der Kennwortrichtlinien.

Bereits vorhandene Passwörter testen

Die per GPO definierten Spielregeln wirken sofort auf alle Versuche der Anwender, ihr Passwort zu ändern. Auch ein Admin, der in der Konsole "Active Directory-Benutzer und -Computer" ein neues Kennwort für ein Benutzerkonto setzt, kann sich nicht darüber hinwegsetzen. Doch wie steht es um Passwörter, die bereits existierten, bevor Sie das Regelwerk für LPP etabliert haben? Hier hilft Ihnen das Cmdlet "Test-IsADUserPasswordCompromised". Für ein einzelnes Benutzerkonto prüfen Sie damit, ob es ein als kompromittiert bekanntes Passwort verwendet:

```
Test-IsADUserPasswordCompromised
  -UPN <paul.ahner@knermann.local>
```

Das Cmdlet gibt als Ergebnis "True" oder "False" zurück. Alle Benutzerkonten Ihrer Umgebung in einem Rutsch testen Sie mittels eines von Lithnet bereitgestellten PowerShell-Skripts [7]. Dies schreibt alle Benutzerkonten, deren Passwort leer oder auf der HIBP-Liste zu finden ist, in eine CSV-Datei, sodass Sie die betroffenen Anwender informieren und zum Passwortwechsel auffordern können.

Zu beachten ist dabei, dass das Cmdlet dazu den Hash-Wert des Passworts aus der AD-Datenbank ausliest, nicht das Passwort selbst. Das Cmdlet kann folglich nur prüfen, ob das Passwort bei HIBP bekannt ist. Ob das Passwort Ihre sonstigen Anforderungen an die Komplexität erfüllt, stellt das Werkzeug nicht fest. Weiterhin setzt auch das Auslesen eines Hash-Wertes sehr weitreichende Berechtigungen voraus. Das Konto, in dessen Kontext Sie das Cmdlet und das Skript verwenden, muss Mitglied der Gruppe der Domänen-Admins sein oder mindestens die Berechtigung "Alle Verzeichnisänderungen replizieren (DS-Replication-Get-Changes- All)" besitzen. Letztere ist unter Nutzern des Tools Mimikatz beliebt, da sie sich für einen DCSync-Angriff missbrauchen lässt [8]. Tool und Skript sollten Sie entsprechend nur auf vertrauenswürdigen Geräten, am besten direkt auf einem DC verwenden.

Fazit

Lithnet Password Protection for Active Directory zeigt sich im Vergleich zu Microsofts Bordmitteln deutlich flexibler. Erst der Einsatz von LPP erlaubt variable Regeln, die die Länge eines Kennworts abhängig von seiner Komplexität prüfen. Als nachteilig erweist sich lediglich, dass LPP die Mitteilsamkeit eines Windows-Clients nicht verbessert. Sollte ein Passwort vor Ihrem Regelwerk nicht bestehen, liefert Windows dem Endanwender nur die wenig aussagekräftige Standardmeldung zurück und keinen Hinweis darauf, welche Bedingung genau das Ändern des Passworts verhindert hat. Die Einführung von LPP sollten Sie durch eine Kampagne flankieren, mit der Sie Ihre Anwender für die Notwendigkeit sicherer Kennwörter sensibilisieren und Ihre Spielregeln erläutern. *(dr)* IT

Link-Codes

[1] BSI: Sichere Passwörter erstellen
l0z01

[2] Wie sicher ist mein Passwort?
l0z02

[3] Passwortfilter
l0z03

[4] Lithnet Password Protection
l0z04

[5] Veröffentlichte Kennwörter
l0z05

[6] Lithnet Event-Logging und -Reporting
l0z06

[7] Audit vorhandener Passwörter
l0z07

[8] DCSync-Attacken gegen das Active Directory
l0z08