Active Directory: Wie Angreifer mit Deception in die Falle gelockt werden

Überlistet und ausgebremst

Frank Ully

Neben passiven Schutzmaßnahmen, die ein Eindringen von Angreifern in Active-Directory-Umgebungen verhindern sollen, gibt es auch offensivere Methoden: Eine aktive Verteidigung soll Eindringlinge in die Irre führen, damit sie schneller entdeckt werden und weniger Schaden anrichten.

Der letzte Teil der in den vergangenen Monaten erschienenen Artikelreihe widmet sich einem noch jungen Ansatz zur Absicherung des Active Directory (AD): der aktiven Verteidigung. Ging es bei den vorherigen Beiträgen darum, welche Einfallstore das Active Directory bietet, wie Systemverwalter Fehlkonfigurationen vermeiden, ihre Umgebung härten und mögliche Angriffe durch Logs und Monitoring entdecken können, so zielt die neue Methode darauf, Angriffsversuche schnell zu erkennen und Eindringlinge in die Irre zu führen. Vermeintlich fehlkonfigurierte Konten, die aussehen, als seien sie echt, bei Missbrauch aber Alarm auslösen, sollen unerwünschte Gäste im Unternehmensnetzwerk in die Falle locken. Unter Sicherheitsexperten ist unumstritten: Nach dem „Assume Breach"-Ansatz bereitet es Angreifern keine Schwierigkeiten, ein beliebiges System in einem Unternehmen zu kompromittieren, beispielsweise durch Phishing oder das Ausnutzen einer Schwachstelle, und von dort Befehle auszuführen. Dass eine Organisation angegriffen wird, ist also keine Frage des Ob, sondern nur eine Frage des Wann. Spätestens mit den immer verheerenderen Ransomware-Attacken sollte dies deutlich geworden sein.

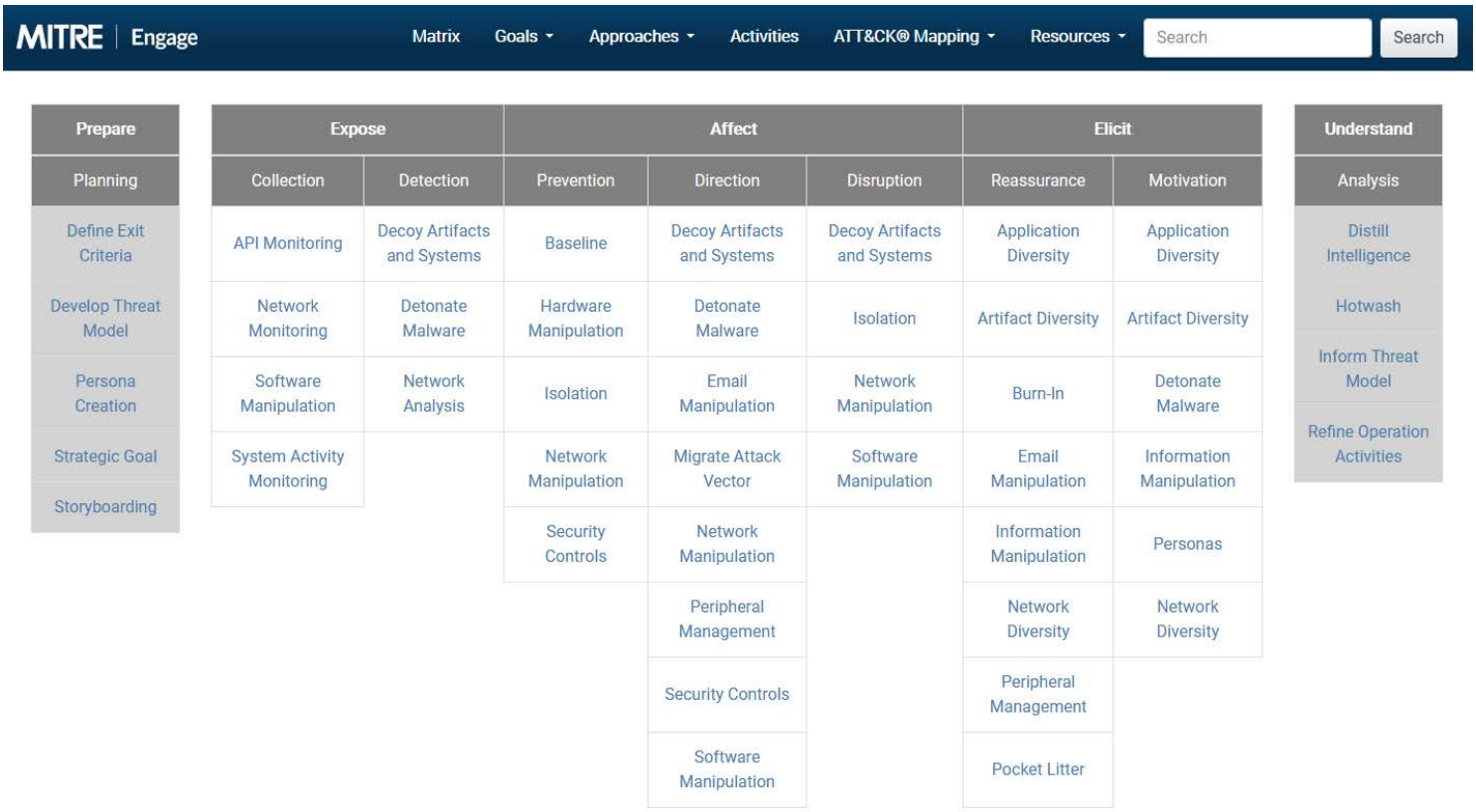
Spät erkannte Angriffe

Noch immer investieren Unternehmen meist nur in das Verhindern und kaum in das Erkennen möglicher Sicherheitsvorfälle. Zwar drängen mit Endpoint Detection and Response (EDR) und Extended Detection and Response (XDR) neuere Techniken auf den Massenmarkt, die kompromittierte Systeme erkennen sollen (siehe [1] und Artikel „Auf dem Radar" Seite 52). Allerdings haben sich fortgeschrittene Angreifer daran angepasst und können diese und etablierte Sicherheitsmaßnahmen wie Malwarescanner mit etwas Aufwand umgehen (siehe ix.de/zbmf). Selbst wenn ein Monitoring eingerichtet ist, auf das ein internes oder externes Security Operations Center (SOC) [2] ein wachsames Auge hat, sind die Meldungen größtenteils falsche Warnungen, was zu einer gewissen Alarmmüdigkeit der Verteidiger führt. All dies bedeutet, dass es in Europa – je nach Quelle, jüngst erschien beispielsweise der FireEye M-Trends Report 2021 (siehe ix.de/zbmf) – im Mittel immer noch zwei Monate dauert, bis ein Angriff entdeckt wird.

Ein Lichtblick ist, dass bei typischen Attacken mit dem ersten Glied in der Angriffskette noch nicht alles vorbei ist: Das Starten von Schadsoftware aus einer Phishing-E-Mail bedeutet in der Regel nicht, dass die Verteidiger sofort unrettbar verloren haben. Erst wenn geistiges Eigentum abfließt oder alle Daten im Netzwerk verschlüsselt werden, die Angreifer also ihr eigentliches Ziel erreicht haben, hat das Opfer einen Schaden erlitten. Auch wenn sich die Zyklen bei Ransomware-Angriffen verkürzen und wie bei der BazaCall-Kampagne nur 48 Stunden vom initialen Zugriff bis zum Starten der Verschlüsselung vergehen (siehe ix.de/zbmf), benötigen die Angreifer dennoch Zeit, um ihr Ziel zu erreichen – Zeit, die den Verteidigern Gelegenheit gibt, sie zu erkennen und aus dem eigenen Netzwerk zu vertreiben.

Angreifersicht und Paradigmenwechsel

Deswegen ist es als Verteidiger wichtig, die Sicht der Angreifer einzunehmen. Angreifer wiegen sich gegenüber Verteidigern im Gefühl einer womöglich trügerischen Überlegenheit. Sie blicken ein bisschen verächtlich auf die Admins und den desolaten Zustand ihrer unsicheren Netzwerke herab und gehen davon aus, dass sie viel Zeit haben, sich langsam in der Umgebung auszubreiten. Verteidiger müssen alle Schwachstellen stopfen, Angreifern reicht für jeden Schritt ihrer Attacke eine einzige Lücke, die sie erfolgreich ausnutzen. Dabei verlassen sie sich auf die Informationen, die sie außerhalb und innerhalb des Unternehmensnetzwerks gesammelt haben, und stürzen sich auf tief hängende



Die MITRE-Engage-Matrix ist das Gegenstück zur ATT&CK-Angriffssammlung. Die dargestellten Abwehrtechniken sind auf die jeweiligen Angriffe zugeschnitten (Abb. 1).

Früchte, suchen zum Beispiel den schnellsten Weg zum Domänenadministrator.

Deception, also Täuschung, stellt dieses Paradigma auf den Kopf und wendet die bei Angreifern beliebte Taktik, etwa sich bei einer Spear-Phishing-E-Mail als jemand anders auszugeben, gegen sie selbst, um die Verteidigung zu verbessern. Täuschung ist die Idee, einen Eindringling zu entdecken, indem man ihn mit falschen Fährten wie scheinbar echten Benutzern, Geräten, Diensten, Dokumenten oder gar gesamten Infrastrukturen in die Falle lockt. Da er gefälschte nicht von echten Daten unterscheiden kann, wird er versuchen, sie zu verwenden, und damit einen Alarm auslösen. Täuschung ersetzt nicht andere Sicherheitsmechanismen; sie ergänzt und unterstützt die anderen Maßnahmen.

Mit Täuschung müssen die Angreifer jeden aufgespannten Stolperdraht umgehen, um nicht entdeckt zu werden, während die Verteidiger nur eine der vielen Alarmglocken hören müssen. Das führt zu einer veränderten Perspektive der Verteidiger, die nicht mehr nach dem Bösen in ihren Netzwerken suchen, sondern nach dem Unnormalen. Kein legitimer Benutzer sollte auf ein Honigtopfsystem zugreifen oder die als Falle ausgelegten Zugangsdaten verwenden – daher ist jede Interaktion damit mindestens verdächtig und im schlimmsten Fall bösartig. Auch wissen die Verteidiger nun nicht nur, dass ein Angriff stattfindet, sondern können bereits kompromittierte Ressourcen identifizieren. Und selbst wenn die Angreifer einige Täuschungsversuche entdecken, ist die Deception damit nicht nutzlos geworden: weil die Angreifer nun den gesammelten Informationen nicht mehr vorbehaltlos vertrauen können und sich bei jedem Schritt fragen müssen, ob sie womöglich erneut in eine Falle der Verteidiger tappen.

Schon Organisationen mit niedrigem Sicherheitsniveau können von täuschungsbasierten Schutzmechanismen profitieren, weil sie kaum Fehlalarme auslösen. Besser aufgestellten Unternehmen dient sie zur Ergänzung anderer Sicherheitsmaßnahmen, die wie Malwarescanner oder EDR auf Signaturen oder Heuristiken angewiesen sind. Die Verweildauer von Angreifern in einem Netzwerk und die Reaktionszeit der Verteidiger sinken.

Gartner stuft in seinem „Hype Cycle for Security Operations“ die Wirksamkeit von Deception-Techniken als hoch ein, sieht den Weg in den Mainstream aber erst in ein paar Jahren (siehe ix.de/zbmf). Neben zahlreichen mehr oder weniger gepflegten Open-Source-Projekten,

iX-TRACT

- Täuschung zielt im Sinne einer aktiven Verteidigung auf die Denkweise von Angreifern. Sie kann Eindringlinge durch Fehlinformationen bremsen und dazu bringen, ihr weiteres Vorgehen und ihre Taktiken zu verraten. Das Rahmenwerk MITRE Engage bietet dabei Orientierung.
- Ohne viele Fehlalarme auszulösen, können verschiedene Arten von Honigtöpfen – wie Honigsysteme, -dienste und -token – in Active-Directory-Umgebungen Angreifer entdecken, die versuchen, nach einem initialen Zugriff ihr eigentliches Ziel zu erreichen.
- Admins können dank Windows-Bordmitteln und Open-Source-Werkzeugen selbst Stolperfallen aufstellen. Alle anderen finden auf dem Markt zahlreiche kommerzielle Deception-Produkte.

die teilweise im Folgenden vorgestellt werden, tummeln sich auf dem Markt kommerzielle Anbieter, darunter Aittvo, CounterCraft, Fidelis, Illusive, Javelin, TrapX und Smokescreen. Auch Schwergewichte wie Microsoft haben in Produkten wie Defender for Identity (ehemals Azure ATP) Funktionen zum Verwalten von Deception eingebaut.

Täuschung ist die beste Verteidigung

Bei Täuschungstechniken herrscht ein Begriffswirrwarr. Honeypots [3] oder Honigtöpfe sind der bekannte Oberbegriff: Ressourcen, die keinen für die Kernfunktion des Unternehmens relevanten Wert darstellen und explizit dazu gedacht sind, dass Angreifer mit ihnen interagieren, aber niemals legitime Benutzer.

Im engeren Sinn sind klassische Honigtöpfe nach außen gerichtet, um als Research-Honeypots Erkenntnisse über verschiedene Malware-Familien und das typische Vorgehen von Angreifern zu erforschen, wenn sie beispielsweise über SSH-Passwort-Brute-Forcing Zugriff auf einen Linux-Server erlangen. Honigtöpfe für Deception richten sich hingegen nach innen, um dort Angreifer auf falsche Fährten zu locken, damit einen Angriff zu erkennen und Erkenntnisse über die spezifischen Täter zu sammeln.

Wenn man nach Art des Honigtopfes unterscheidet, lassen sich sehr komplexe Taxonomien bilden – einfach und praktikabel ist folgende Aufteilung: Ein Honigsystem ist ein echter oder simulierter Rechner im Netzwerk. Ein Honigdienst verhält sich wie eine bestimmte Serveranwendung oder ein Protokoll; Beispiele dafür sind ein SSH-Server oder eine SMB-Verzeichnisfreigabe. Ein Honigtoken imitiert legitime Daten; Beispiele sind ein speziell gestaltetes Word-Dokument in einer geschützten Dateifreigabe oder ein Active-Directory-Benutzerkonto mit absichtlich schlechtem Passwort.

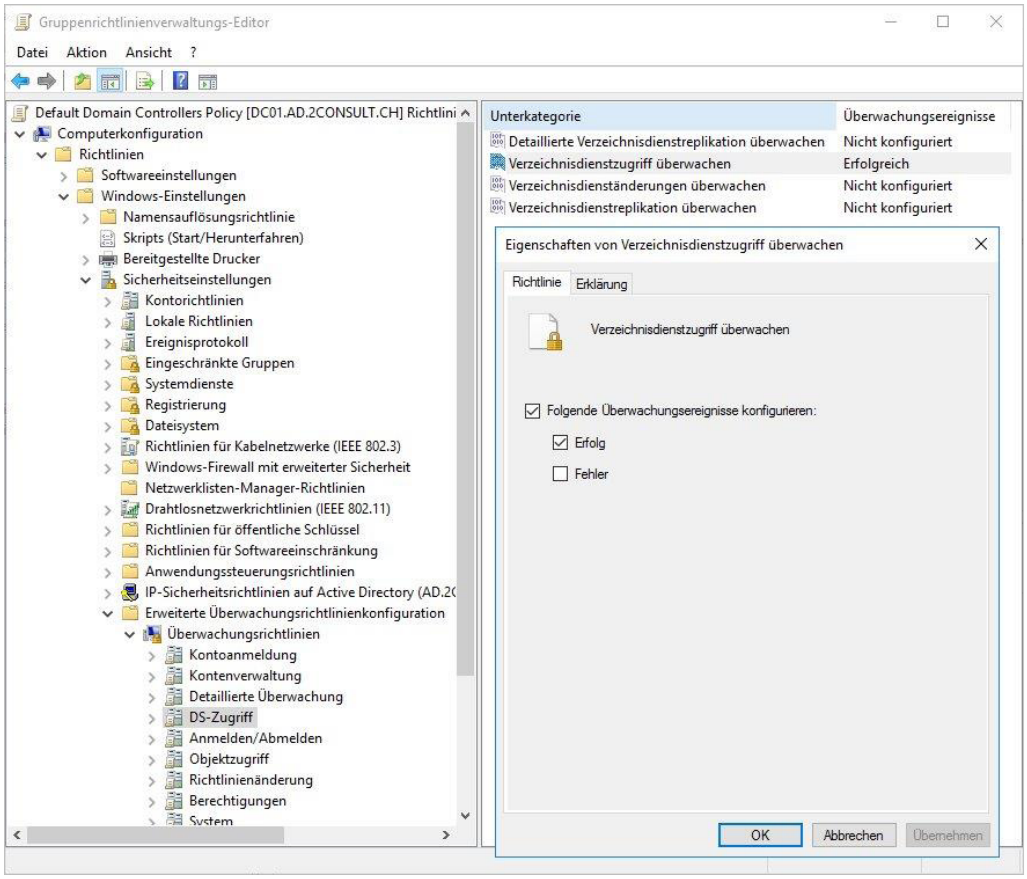
Gemeinsam haben in dieser Klassifizierung alle Arten von Honigtöpfen, dass eine Interaktion mit ihnen einen Alarm auslöst. Sie können auch miteinander kombiniert werden: Ein Honigsystem kann verschiedene Honigdienste anbieten, die wiederum Honigtoken ausliefern. Um Angreifer gezielt zu einem Honigtopf zu locken, werden Breadcrumbs (Brotkrumen) ausgelegt, beispielsweise eine Verknüpfung auf dem Desktop zu einem Honigdienst oder eine fingierte E-Mail, deren Text auf ein Honigtoken verweist.

Unter den Herstellern kommerzieller Täuschungsplattformen hat sich zudem eine komplexe und teils widersprüchlich verwendete Terminologie entwickelt, die man nicht übernehmen muss, aber kennen sollte: Decoys (Lockvögel) ist eine synonyme Bezeichnung für Honigsysteme oder -dienste. Lures (Lockmittel) ist ein anderer Name für Brotkrumen. Sie initiieren für sich genommen keinen Alarm. Baits (Köder) oder Canaries (Kanarienvögel) hingegen sind Honigtoken, also eigenständige Objekte wie Dokumente, die beispielsweise beim Öffnen Alarm auslösen.

Die Angreifer beschäftigen

Aktive Verteidigung nutzt im Gegensatz zur klassischen passiven Verteidigung auch offensive Maßnahmen, um einen Gegner zu überlisten und einen Angriff zu erschweren. Das Verlangsamen des Angreifers, sodass er nicht vorankommt oder den Angriff nicht zu Ende führen kann, erhöht die Wahrscheinlichkeit, dass er einen Fehler macht und entdeckt wird. Während passive Verteidigung die Wahrscheinlichkeit eines Angriffs reduzieren soll, ist aktive Verteidigung in gewisser Weise eine Reaktion auf einen Angreifer und ermöglicht es, wenn sie weitergetrieben wird, den Angreifer unter kontrollierten Bedingungen zu beobachten.

Die MITRE Corporation ist eine gemeinnützige Organisation, die Beiträge zur IT-Sicherheit leistet, etwa mit der ATT&CK-Matrix, die Techniken, Taktiken und Methoden aus der Perspektive von Angreifern abbildet [4]. Als Gegenstück stellt die im August 2021 veröffentlichte Engage-Matrix (beide Matrizes siehe ix.de/zbmf) Maßnahmen zur aktiven Verteidigung dar, mit einem Fokus unter anderem auf Täuschung. In der obersten Zeile der Matrix (siehe Abbildung 1) stehen die Ziele. Darunter besteht die Matrix aus neun Spalten zu verschiedenen Ansätzen – beispielsweise



Wenn die Standardgruppenrichtlinie entsprechend konfiguriert wird, zeichnet sie die Enumerationsversuche bei Active-Directory-Objekten auf (Abb. 2).

Listing 1: Erstellen eines Brotkrumenpfads über Zugriffskontrolllisten zu Honig-Zugangsdaten

```
PS > Import-Module C:\Deploy-Deception\Deploy-Deception.psd1
PS > Add-Type -AssemblyName System.Web
PS > Create-DecoyUser -UserFirstName Erster -UserLastName Benutzer -Password ([System.Web.Security.Membership]::GeneratePassword(128,2)) | Deploy-UserDeception ⤸
                                                                                -GUID d07da11f-8a3d-42b6-b0aa-76c962be719a -Verbose
PS > Create-DecoyUser -UserFirstName Zweiter -UserLastName Benutzer -Password ([System.Web.Security.Membership]::GeneratePassword(128,2)) | Deploy-UserDeception ⤸
                                                                                -GUID d07da11f-8a3d-42b6-b0aa-76c962be719a -Verbose
PS > Deploy-SlaveDeception -SlaveSamAccountName ZweiterBenutzer -DecoySamAccountName ErsterBenutzer  -Verbose
PS > Deploy-UserDeception -DecoySamAccountName ZweiterBenutzer -Principal ErsterBenutzer -Right WriteDacl -Verbose
```

Planung, Sammlung und Erkennung –, die Verteidiger zur Abwehr von Eindringlingen nutzen können. Verweise in den einzelnen Zellen führen zur Beschreibung von Aktivitäten, beispielsweise „Decoy Artifacts and Systems“. MITRE verwendet statt des Wortbestandteils „Honey“ bevorzugt „Decoy“.

Darüber hinaus ist die Matrix in zwei Kategorien unterteilt: Strategische Ziele, Ansätze und Aktivitäten (grau hinterlegt) bilden Beginn und Abschluss und stellen sicher, dass Planung und Analyse nicht vergessen werden. Einsatzziele, -ansätze und -aktivitäten beschreiben konkrete Maßnahmen, bei denen etwa Deception-Tools helfen können, die strategischen Ziele zu erreichen. MITRE hat das im ATT&CK-Framework abgebildete Angreiferverhalten den entsprechenden Aktivitäten der Verteidiger in Engage zugeordnet, das derzeit in einer Betaphase ist (Version 0.9) und laufend weiterentwickelt wird. Das Engage-Framework ersetzt mittelfristig den Vorgänger MITRE Shield (siehe ix.de/zbmf); eine Version 1.0 ist noch für 2021 geplant.

In Bezug auf das ATT&CK-Framework kann Täuschung bei einem Angreifer vor allem in den Phasen „Privilege Escalation“ bis hin zu „Collection“ zur Entdeckung führen. Andere Sicherheitsansätze wie EDR legen den Fokus auf die früheren ATT&CK-Phasen, Data Leakage Prevention (DLP) [5] auf die späteren.

Allerdings kann Deception prinzipiell schon viel früher eingesetzt werden, um die ersten Glieder einer Angriffskette zu erkennen: Bereits in der Phase des externen Sammelns öffentlich zugänglicher Informationen („Reconaissance“ bei ATT&CK), etwa über Mitarbeiter und eingesetzte Techniken, können die Verteidiger beispielsweise auf Geschäftsnetzwerken wie LinkedIn gefälschte Daten streuen.

Schutz schon bei der Enumeration

Beim Einstieg in Deception ist es ratsam, erst bei späteren Angriffsphasen anzusetzen und sie schrittweise auszubauen. Da das Active Directory bei den meisten Organisationen der zentrale Dienst für die Ressourcenverwaltung ist, bietet sich der Einsatz von Täuschung und Honigtöpfen hier besonders an.

Dort kann Täuschung bereits bei der Enumeration ansetzen, der Beschaffung von Informationen über Benutzer- und Computerkonten und Gruppen und einer der frühesten Angriffsphasen, nachdem Eindringlinge initial Zugriff auf eine AD-Umgebung erlangt haben. Zu ihrem Aufbau können mitgelieferte Werkzeuge von Windows wie PowerShell, Gruppenrichtlinien und Ereignisprotokolle genutzt werden.

Zunächst muss in den erweiterten Protokollierungsrichtlinien die Einstellung „Verzeichnisdienstzugriff überwachen“ („Audit Directory Service Access“) auf „Erfolg“ gesetzt werden. Das lässt sich beispielsweise über die Standardgruppenrichtlinie für Domänencontroller konfigurieren (siehe Abbildung 2). Diese Konfiguration protokolliert auf dem Domänencontroller (DC) ein Sicherheitsereignis mit der Ereignis-ID 4662, wenn auf ein AD-Objekt zugegriffen wird. Die genaue Protokollierung von Enumerierungsversuchen muss danach auf Objektebene konfiguriert werden.

Honig für Benutzer, Computer und Gruppen

Nikhil Mittal stellt mit dem PowerShell-Modul Deploy-Deception (siehe ix.de/zbmf) eine kostenfreie und einfache Möglichkeit bereit, Honigtoken in einer Domäne anzulegen.

Beispielsweise sind Domänenbenutzer für einen Angreifer interessant, wenn ihr Passwort nie abläuft, sie Mitglieder privilegierter Gruppen mit hohen Rechten sind oder sie über Access Control Lists (ACLs) Veränderungen an anderen AD-Objekten vornehmen können.

Die Befehle in Listing 1, in einer administrativen PowerShell-Sitzung mit Domänenadmin-Rechten ausgeführt, legen zwei neue Benutzer mit jeweils sehr starken Zufallspasswörtern an und geben dem ersten Benutzer die `WriteDacl`-Berechtigung über den zweiten.

Mit dem ersten und zweiten `Create-DecoyUser`-Befehl werden zwei Benutzer angelegt und Logging für ihre Enumeration aktiviert, wenn jeweils ihr Attribut x500uniqueIdentifier (mit der GUID d07da11f-8a3d-42b6-b0aa-76c962be719a) gelesen wird. Dieses Attribut wird von mitgelieferten Admin-Werkzeugen wie net user nicht abgefragt, aber von typischen Angriffswerkzeugen wie PowerView, die alle Objekteigenschaften auflisten. Das vermeidet Fehlalarme.

Mit dem vorletzten Befehl in Listing 1 erhält der erste Benutzer `WriteDacl`-Rechte über den zweiten. Listet nun ein Angreifer Benutzer in der Domäne auf, ist diese Enumeration auf dem DC in den Sicherheitsprotokollen zu finden (siehe Abbildung 3), wenn dort nach der Ereignis-ID 4662 gefiltert wird. Die Protokolldetails zeigen, dass ein Aufzählungsskript von `alice.musterfrau` gestartet wurde. Zusätzlich würde durch den letzten Befehl ein Protokolleintrag erzeugt, wenn ein Angreifer tatsächlich die Berechtigungen am zweiten Nutzer ändern würde.

Auch Computerobjekte im AD sind für einen Angreifer interessant, beispielsweise dann, wenn sie alte Betriebssysteme verwenden, als Mitglieder privilegierter Gruppen oder für Delegierung konfiguriert sind. Folgender Befehl verwendet ein vorhandenes Computerobjekt und aktiviert daran die uneingeschränkte Delegierung. Vorsicht: Das macht diesen Computer zu einem lohnenswerten Ziel, er muss besonders abgesichert werden. Protokolliert wird die Auflistung des Computers immer dann, wenn seine ACL oder alle Attribute gelesen werden.

```
PS > Deploy-ComputerDeception ⤸
            -DecoyComputerName dateiserver01 ⤸
          -PropertyFlag TrustedForDelegation ⤸
                  -Right ReadControl -Verbose
```

Ebenso lassen sich Gruppen zur Täuschung nutzen. Verlockend für Angreifer sind beispielsweise Benutzergruppen mit „Admin“ im Namen sowie Mitglieder hochprivilegierter Gruppen wie Domänen- oder Organisationsadmins. Folgender Befehl legt eine neue Gruppe `Gesamt-Admins` an und macht sie zu einer Untergruppe der eingebauten Gruppe `DNSAdmins`. Deren Mitglieder sind nur wenige Tastenanschläge davon entfernt, Domänenadministrator zu werden. Schließlich fügt der Befehl den zuvor angelegten Benutzer zur `Gesamt-`

Admin-Gruppe hinzu. Ein Ereignis mit der ID 4662 wird dann aufgezeichnet, wenn jemand die Mitglieder dieser Gruppe auflistet:

```
PS > Create-DecoyGroup -GroupName ⤸
          "Gesamt-Admins" -Verbose | Deploy-⤸
   GroupDeception -AddMembers ZweiterBenutzer ⤸
                  -AddToGroup DNSAdmins -GUID ⤸
  bc0ac240-79a9-11d0-9020-00c04fc2d4cf -Verbose
```

Kerberoasting und Angreiferbewegungen

Die eben beschriebenen Täuschungsmanöver entdecken Angreifer, sobald sie potenziell lohnenswerte Objekte enumerieren. Einen Schritt weiter kann Deception gehen, wenn Eindringlinge dabei beobachtet werden, wie sie aktive Angriffe durchführen, etwa Kerberoasting. Bei solchen Angriffen werden für Dienstkonten mit einem Service Principal Name (SPN) Tickets angefragt, deren Passwörter dann offline geknackt werden können, falls sie schwach sind.

Folgender Befehl legt einen Benutzer an und setzt einen SPN – zusätzlich wird wie oben beschrieben eine Auditregel aktiviert, die schon beim Anzeigen der ACL einen Logeintrag erzeugt:

```
PS > Create-DecoyUser -UserFirstName svc ⤸
     -UserLastName manager -Password ([System.⤸
    Web.Security.Membership]::GeneratePassword⤸
         (128,2)) | Deploy-UserDeception -SPN ⤸
       'MSSQLSvc/dateiserver01.ad.2consult.ch:⤸
              1433' -Right ReadControl -Verbose
```

Der Blogartikel „Honeyroasting: How to detect Kerberoast breaches with honeypots“ (siehe ix.de/zbmf) beschreibt ausführlich, welche erweiterte Überwachungsrichtlinie für Kerberos zusätzlich aktiviert werden muss und wie XPath-Protokollfilter Falschmeldungen vermeiden. Wenn nun ein Angreifer versucht, alle Konten in der Domäne zu rösten, oder ihm das Kerberoasting des Honigkontos beispielsweise durch dessen Mitgliedschaft in einer privilegierten Gruppe schmackhaft gemacht wurde, entsteht ein Protokolleintrag mit der ID 4769.

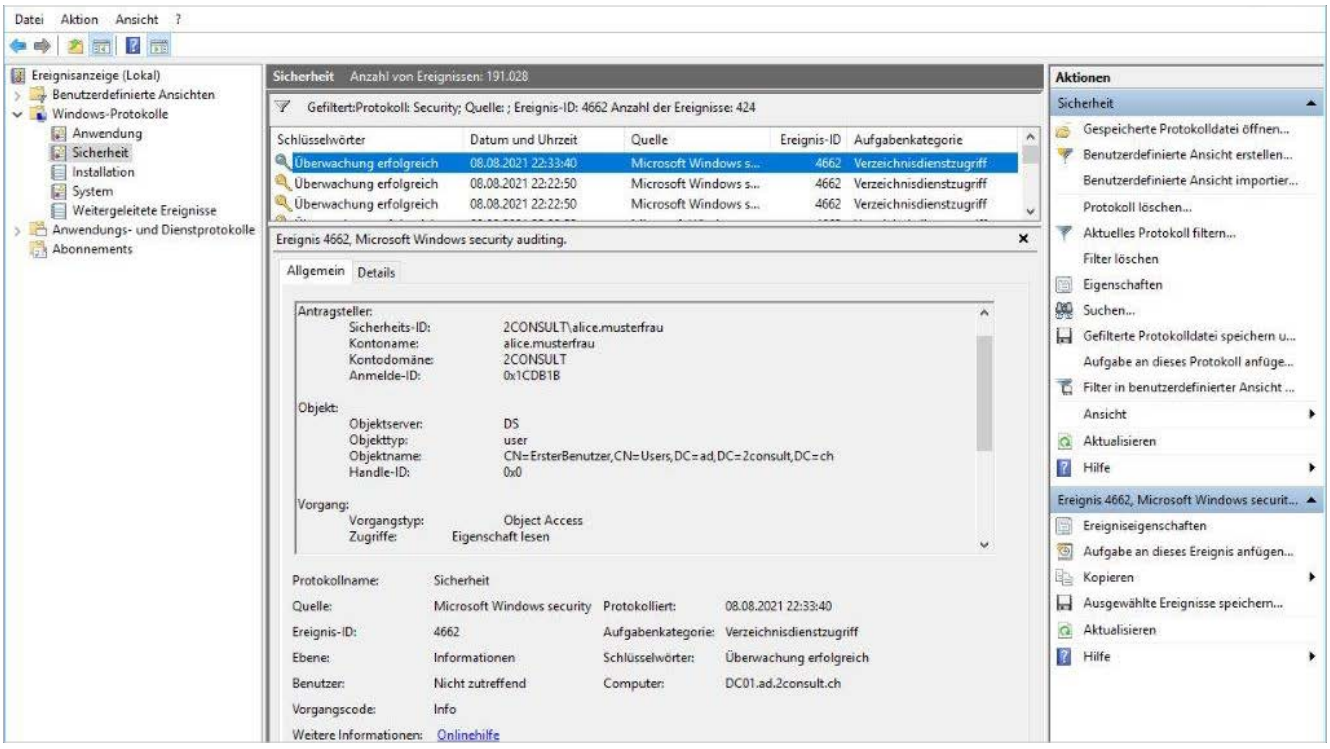
Ein erkennbarer Angriff ist auch das Password Spraying gegen alle oder einige AD-Benutzer oder der Versuch, sich mit einem Kennwort anzumelden, das Systemverwalter unvorsichtigerweise im allgemein lesbaren Beschreibungsfeld eines Benutzerkontos hinterlassen haben. Als Täuschungsmaßnahme können Administratoren mit eingebauten grafischen oder Kommandozeilenwerkzeugen von Microsoft entsprechende Benutzer mit einem scheinbar gültigen Passwort in der Beschreibung anlegen, das aber vom eigentlichen Kennwort abweicht.

Auf Domänencontrollern erscheinen gescheiterte Anmeldeversuche mit ungültigen Honig-Zugangsdaten in den Protokollen mit den Ereignis-IDs 4625 (Authentifizierungsversuch über Net-NTLM) und 4771 (Kerberos-Authentifizierung), mehr dazu im Artikel zu Logging [6]. Wird wie im Honeyroasting-Blogartikel beschrieben auch für diese Ereignisse ein XPath-Filter erstellt, bleibt das Protokoll übersichtlich. Mit eigens angelegten Honigkonten sollte sich kein legitimer Benutzer je anmelden, dadurch ist die Zahl der Fehlalarme gering.

Lokal ausgelegte Honig-Zugangsdaten

Beim Platzieren von Brotkrumen zu Honig-Zugangsdaten sind der Fantasie der Verteidiger keine Grenzen gesetzt, sofern sie dabei das übliche Vorgehen von Eindringlingen im Hinterkopf behalten: Honigtoken können in typischen Dateien platziert werden, die in schlecht administrierten Umgebungen hart codierte Passwörter im Klartext enthalten und nach denen Angreifer oft suchen. Das sind beispielsweise Skripte in der SYSVOL-Standardfreigabe auf Domänencontrollern oder Dateien mit klingenden Namen wie Passwörter.txt auf öffentlich zugänglichen Netzwerkfreigaben.

Ebenso beliebt sind gespeicherte Zugangsdaten in den Group Policy Preferences (GPP) im SYSVOL-Ordner. Neue Passwortdaten können zwar seit Langem nicht mehr in Gruppenrichtlinien-Einstellungen hinterlegt werden, aber in vielen Organisationen sind dort vor Jahren gespeicherte, immer noch gültige lokale Admin- und sogar Domänenadministrator-Anmeldeinformationen zu finden. In



Enumerierungsversuche erzeugen einen Protokolleintrag und entlarven so die Aktivitäten eines Angreifers (Abb. 3).

aktuellen Umgebungen hilft Verteidigern das PowerShell-Skript GPPDecoy (siehe ix.de/zbmf) dabei, Fährten zu Honig-Zugangsdaten in den GPP auszulegen.

Auch auf lokalen Systemen wie Servern oder Clients können absichtlich ungültige Zugangsdaten platziert werden. Eine Möglichkeit, nach der Eindringlinge Ausschau halten, die etwa über Phishing den ersten Client in einer Domäne kompromittiert und darauf lokale Adminrechte erlangt haben, sind Windows-Dienste, die mit Domänenkonten gestartet werden. Wenn dafür ein Konto mit hohen Privilegien in der Domäne verwendet wird, haben Angreifer normalerweise leichtes Spiel – Verteidiger können ihnen aber ein falsches Passwort für einen echten, hoch privilegierten Account unterjubeln, indem sie einen Windows-Dienst erstellen (siehe Listing 2).

Auch im Speicherbereich des Local Security Authority Subsystem Service, kurz LSASS, jagen Angreifer nach Zugangsdaten. Dieser Prozess speichert Anmeldeinformationen von Benutzern mit aktiven Windows-Sitzungen. Angreifer können daraus Passwort-Hashes, Kerberos-AES-Schlüssel und unter gewissen Umständen auch auf neueren Windows-Systemen Klartextpasswörter auslesen.

Das auf GitHub bereitstehende PowerShell-Skript New-HoneyHash.ps1 (siehe ix.de/zbmf) bringt Honig-Zugangsdaten in den Arbeitsspeicher ein, wenn im heruntergeladenen HoneyHash-Skript eine neue Zeile am Ende eingefügt wird:

```
New-HoneyHash -Domain 2consult -Username 7
          susanne.server -Password "Ungültig, 7
                                  aber plausibel"
```

Das lokal unter anderem Namen gespeicherte PowerShell-Skript startet wie folgt als geplante Aufgabe oder in einem Login-Skript bei jeder Anmeldung eines Benutzers:

```
powershell -window hidden -noexit -file 7
   C:\Windows\System32\2consult_SystemCheck.ps1
```

Anmeldeversuche mit diesen gefälschten Zugangsdaten lassen sich durch die Ereignis-IDs 4625 und 4771 finden.

Während zumeist bei Deception-Honigtöpfen die Erkennung nicht auf dem initial kompromittierten System, sondern auf Netzwerkebene oder anderen Servern wie einem Domänencontroller stattfindet, beschreibt ein Blogbeitrag von Outflank (siehe ix.de/zbmf), wie unter anderem mit Honigtoken schon die lokale Enumeration auf einem Rechner festgestellt werden kann.

Namensfälschungen auf der Spur

Net-NTLM-Relaying, das eine technische Grundlage der gerade kritisch diskutierten Schwachstelle „PetitPotam“ bildet [7], ist schon lange ein bekannter Angriffspfad in Active-Directory-Umgebungen. Eine Möglichkeit, Net-NTLM-Relaying auszulösen, ist Name Spoofing über DNS-Fallback-Mechanismen wie NetBIOS-Nameservice (NBT-NS) oder Link Local Multicast Name Resolution (LLMNR) [8].

Das Deception-Werkzeug HoneyCreds, das auf Python basiert und am besten unter Linux läuft, und das PowerShell-Skript Invoke-HoneyCreds für Windows (beide siehe ix.de/zbmf) können missbräuchliche Namensauflösungen durch Angriffstools erkennen und einen Log-

eintrag erstellen – und mit Honig-Zugangsdaten antworten, worin die Deception-Komponente besteht. Wer versucht, diese Zugangsdaten für einen Angriff zu nutzen, hat sich endgültig enttarnt. Diese Tools sind also für eine Erkennung mit wenigen False Positives nützlich.

Der Schlüssel für eine erfolgreiche Deception besteht darin, zu verstehen, wie Angreifer vorgehen, und Fallen zu stellen, die ein ihrer Erwartung entsprechendes Trugbild zeichnen. Dabei können herkömmliche präventive Sicherheitsmaßnahmen dazu dienen, ihre Bewegungsfreiheit einzuschränken und sie in die Falle zu locken. Wenn sich die Angreifer hingegen recht frei im Netzwerk bewegen können, müssen die Fußangeln weiträumig aufgestellt werden. Allerdings können zu viele und vor allem plumpe Täuschungsversuche auch dazu führen, dass Angreifer sie erkennen.

Damit Verteidiger glaubhafte Trugbilder zeichnen können, müssen sie verstehen, wie ihre Umgebung normalerweise von Angreifern wahrgenommen wird und auf welche Schwachstellen und Fehlkonfigurationen sie typischerweise in Netzwerken treffen. Schließlich müssen sie sich bewusst sein, was die Kronjuwelen ihrer eigenen Organisation sind und wie vertrauliche Daten aussehen, hinter denen die Angreifer her sind. Die Täuschung bringt die Angreifer dann dazu, das zu tun, was die Verteidiger wollen.

Die richtige Balance

Beim Entwickeln von Täuschung müssen Verteidiger die richtige Balance finden. Einerseits sollen ihre Maßnahmen auffallen, beispielsweise ein Benutzer mit Passwort in der Beschreibung. Andererseits sollen sie aber auch in der bestehenden Umgebung „normal“ wirken, etwa Servicebenutzer mit einem glaubhaften Namen, die als Ziel für Kerberoasting dienen können.

Ein guter Honigtopf kann von einem Angreifer leicht gefunden werden und scheint für ihn zu wertvoll zu sein, um ignoriert zu werden – das Prinzip der niedrig hängenden Früchte. Verteidiger sollten den Honigtopf einfach überwachen können. Er sollte aber bei normalen administrativen Tätigkeiten wie dem Auflisten von Domänenbenutzern keinen Alarm auslösen; das ist bei den zuvor vorgestellten Techniken der Fall.

Wenn die Scheinkonten dann noch im AD gut verteilt werden, etwa in unterschiedlichen Organisationseinheiten (OUs), bieten sie eine gute Abdeckung und werden von unerwünschten Gästen leicht gefunden.

Plausible Daten kreieren

Notwendig ist vor allem, dass ein Honigtopf glaubhaft erscheint und nicht auf den ersten Blick als Fälschung auffliegt. Ein deaktiviertes AD-Benutzerkonto oder ein Konto ohne Benutzerprofil, mit dem sich demzufolge nie jemand angemeldet hat, wird von etwas pfiffigeren Angreifern gemieden. Deswegen sollte sich das Täuschungsmanöver in die Umgebung einfügen: Die Namen von Benutzern, Computern und Gruppen zur Deception sollten der Konvention bei echten AD-Objekten folgen. Felder wie Beschreibung, Vor- und Nachname, Betriebssystemversion oder bei Gruppen auch die restlichen Mitglieder werden realistisch befüllt.

Auch sollte der Honigtopf nicht aussehen, als bestünde er erst seit gestern. Im Kontext von Active Directory bedeutet das etwa, dass Objektattribute wie `whencreated`, `pwdlastset`, `badPwdCount` oder `lastLogon` plausible Werte haben. Allerdings sollten von einer Täuschungsplattform auch nicht alle möglichen Attribute gefüllt werden – ein Angreifer kann das im Vergleich mit den Attributen eines echten Objekts leicht feststellen.

Der Blogartikel „The Art of the Honeypot Account: Making the Unusual Look Normal“ (siehe ix.de/zbmf) geht darauf ein, wie Benutzerkonten als täuschend echt drapiert werden. Eine Möglichkeit, realistische Accounts zu kreieren, die aber wohl überlegt und mit einem Betriebsrat abgesprochen sein sollte: Wenn ein Mitarbeiter das Unternehmen verlässt, wird sein AD-Konto nicht gelöscht, sondern mit geändertem Passwort als Täuschungsobjekt eingesetzt. Bei Honig-AD-Computern empfiehlt sich meist, eine echte Maschine zu verwenden, weil sie schwieriger als ein neu angelegtes Deception-Computerkonto zu erkennen ist.

Sowohl beim Zusammenstellen von Open-Source-Werkzeugen als auch vor dem Kauf kommerzieller Produkte sollte die Glaubhaftigkeit der Deception geprüft werden, beispielsweise in einem Audit durch einen IT-Sicherheitsdienstleister oder zumindest mit öffentlich verfügbaren Tools zum Aufspüren von Honigtöpfen wie Honeypot Buster (siehe ix.de/zbmf).

Nicht zu früh aussperren

Der Einsatz von Honigtöpfen darf die Sicherheit der Organisation nicht schwächen. Bei Deception-Benutzerkonten, mit denen sich ein Angreifer nicht anmelden können soll, sollten sehr starke Passwörter gesetzt werden; als zusätzlicher Schutz können die Anmeldestunden so begrenzt werden, dass nie eine Anmeldung möglich ist.

Ist ein Angreifer in eine Falle getappt, ist oft der erste Impuls, ihn von dem System, auf dem der Alarm ausgelöst wurde, und damit vermeintlich aus dem gesamten Netzwerk auszusperren.

Besonders wenn man sich als Organisation gut gerüstet sieht, kann es aber wertvoller sein, den Eindringling weiter zu beobachten. Dabei lässt sich spezifische Threat Intelligence [9] über den konkreten Akteur gewinnen – damit können die Verteidiger weitere kompromittierte Systeme aufspüren, deren Infektion gar nicht bekannt war, sie können die eigentlichen Ziele der Attacke einschätzen und vielleicht zukünftige Aktionen und weitere Zielsysteme vorhersagen.

Werden zur Täuschung Benutzerkonten mit absichtlich schlechtem Passwort oder Passwort in der Beschreibung eingesetzt, mit denen die Anmeldung erfolgreich sein soll, darf das schwache Passwort („Firmenname2021!“) nicht außerdem von echten Benutzern in der Domäne verwendet werden, sonst wird der Angreifer auch ihre Konten kompromittieren. Überdies sollten diese Konten nicht Mitglied von Gruppen wie `VPN-Benutzer` sein, die eine Ferneinwahl ermöglichen, sonst kann die Organisation darüber von außen angegriffen werden. Und schließlich sollten Benutzerkonten, die ein Angreifer aktiv nutzen soll, stark mit Logging überwacht werden, sonst kann der Angreifer darüber seine Spuren verwischen.

Unzählige Möglichkeiten

Mit der Täuschung von Angreifern, die Enumeration und Bewegung im Active Directory erkennen lässt, sind noch lange nicht alle Möglichkeiten ausgeschöpft.

In frühen Angriffsphasen nutzen Angreifer oft Portscans. Scannen sie ein System, auf dem ein Honigdienst oder dessen vereinfachte Variante, ein Honigport, lauscht, löst dies eine Warnung aus. Ein eigens angelegter Servicebenutzer, dessen SPN auf diesen Honigtopf zeigt, kann als Brotkrumenpfad dorthin dienen.

Angreifer suchen gerne nach eingebundenen Laufwerken, die auf Netzfreigaben verweisen: Freigaben auf Dateiservern enthalten oft interessante Dateien mit vertraulichen Daten oder auch Passwörtern in Excel-Tabellen, an denen Angreifer natürlich interessiert sind. Verteil-

Listing 2: Erstellen eines Brotkrumenpfads zu Honig-Zugangsdaten über einen Windows-Dienst

```
PS > $passwd = ConvertTo-SecureString "Falsches, aber plausibles Kennwort, beispielsweise Sommer2021!" -AsPlainText -Force
PS > $creds = New-Object System.Management.Automation.PSCredential ("2CONSULT\domadm_backuptool", $passwd)
PS > $binaryPath = "c:\BackupTool\BackupTool.exe"
PS > New-Service -name "BackupTool" -binaryPathName $binaryPath -displayName "Backup Tool" -Description "ACME Backup Software" -startupType Manual ⤸
                                                                                                                              -credential $creds
```

diger können dies ausnutzen, indem sie beispielsweise zur echten Freigabe „FORSCHUNG auf DATEISERVER01“ auf Laufwerk F: einen Honigdienst „FORSCHUNG2 auf DATEISERVER02“ einrichten und zu diesem mit dem Laufwerk G: von allen Clientsystemen Brotkrumen legen. Auf dieser Freigabe können gefälschte Excel-Tabellen liegen, die als Honigtoken dienen; werden sie geöffnet, geht ein Alarm an die Verteidiger. Auch können Ordner so präpariert werden, dass schon das Öffnen eine Meldung auslöst. Selbst ausführbare Dateien und API-Schlüssel für Cloud-Dienste sind als Honigtoken nutzbar. Anwendungsfälle der unterschiedlichen Honigtöpfe überschneiden sich, aber sie können gezielt eingesetzt werden, um verschiedene Angriffe zu erkennen.

Eine Software, die beim Erstellen und Verwalten von Honigtoken hilft, ist Canarytokens, die als kostenfreier Webdienst vom Hersteller Thinkst angeboten wird, aber auch unter einer freien Lizenz auf GitHub verfügbar ist (beide siehe ix.de/zbmf) und damit von Organisationen etwa in einem Docker-Container selbst gehostet werden kann. Ein Blick in ihre Dokumentation (siehe ix.de/zbmf) lohnt auf jeden Fall, um Ideen für den Einsatz von Honigtoken zu bekommen.

Tiefergehende Informationen

Deception ist ein umfassendes Gebiet und trotz früher Anfänge erst seit relativ Kurzem eine Disziplin, für die es reife Werkzeuge für den Einsatz innerhalb von Organisationen gibt. Der vom SANS-Institut bereitgestellte „Implementer’s Guide to Deception Technologies“ hilft dabei, sich zwischen Open-Source-Werkzeugen und kommerziellen Tools sowie zwischen kostenpflichtigen Täuschungslösungen zu entscheiden (siehe ix.de/zbmf).

Wer basierend auf Open-Source-Software selbst Täuschung in seinem Netzwerk implementieren möchte, findet konkrete Handreichungen im Ende 2020 veröffentlichten, sehr empfehlenswerten Buch „Intrusion Detection Honeypots: Detection through Deception“ von Chris Sanders; darüber hinaus liefert es konzeptionelle Hinweise zum Entwurf und zur Platzierung von Honigtöpfen. Eine Linux-Distribution mit gesammelten kostenfreien Werkzeugen zum Thema aktive Verteidigung – ähnlich wie es Kali Linux mit Angriffswerkzeugen für Sicherheitsprüfungen ist – steht mit der Active Defense Harbinger Distribution (ADHD; siehe ix.de/zbmf) zur Verfügung. Sie enthält beispielsweise die Canarytokens-Software und bringt eine umfangreiche Dokumentation zu allen enthaltenen Tools mit. Zusammenstellungen kostenfreier Quellen zu Honigtöpfen und aktiver Verteidigung liefern die beiden GitHub-Projekte „Awesome Honeypots“ und „Awesome Active Defense“ (beide sind über ix.de/zbmf zu finden).

Eine gute Übersicht über Täuschung und zu kommerziellen Anbietern von Deception-Lösungen bietet das Whitepaper eines IT-Sicherheitsdienstleisters (siehe ix.de/zbmf). Mehr theoretische Hintergründe und spannende Ideen zu täuschungsbasierter Verteidigung liefern die Links auf „Awesome Deception“ sowie das Buch „Offensive Countermeasures: The Art of Active Defense“ von John Strand, auch wenn dessen zweite und aktuelle Auflage 2017 erschienen ist und manche aktuellen Entwicklungen fehlen.

Fazit

Deception kann in einer AD-Umgebung wesentliche Phasen eines Angriffs aufdecken und neben Honigsystemen und -diensten besonders durch Honigtoken wertvolle Erkenntnisse über Angriff und Angreifer liefern.

Deception-Technologie ist jedoch wie andere Sicherheitsmaßnahmen kein Allheilmittel, das ohne Planung und Schulung der Admins alle Probleme löst – aber sie kann, nach anfänglichem Aufwand für die Einrichtung, relativ wartungsarm zu schnellem Erfolg führen: einem frühen Erkennen von Angriffen und dem Gewinnen spezifischer Threat Intelligence. Das gilt nicht nur für Vorreiterorganisationen mit großem Budget und mehrköpfigem Sicherheitsteam, sondern ebenfalls für mittelständische Unternehmen, die sich in einer immer feindlicheren digitalen Umwelt bewegen.

Vorbeugen ist ideal, Erkennen von Angriffen ein Muss, aber ohne eine angemessene Reaktion von geringem Wert. Wichtig ist das Entwickeln und Üben eines Prozesses für die Behandlung von Vorfällen, wenn ein Incident-Response-Prozess nicht bereits vorhanden ist [10].

Dies ist der letzte reguläre Artikel der in *iX* 10/2020 begonnenen Reihe zu Angriffen auf und Verteidigung von Active Directory. Wegen des großen Interesses werden in kommenden *iX*-Ausgaben weitere AD-bezogene Beiträge erscheinen – zu Passwortaudits, forensischen Untersuchungen, zur Absicherung nach IT-Grundschutz und schließlich zu Azure Active Directory, Microsofts cloudbasiertem Verzeichnisdienst. (ur@ix.de)

Quellen

[1] Stefan Strobel; Erkennen und reagieren; Neue Verteidigungsansätze: EDR und XDR; *iX* 5/2021, S. 122

[2] Hans-Wilhelm Dünn; Beste Reaktion; Security Operations Center intern oder als Managed Service; *iX* 10/2020, S. 118

[3] Markus Manzke; Klebefallen; Botnetzangriffe mit Honeypots analysieren; *iX* 12/2015, S. 98

[4] Marko Klaus; Modellierte Kriegsführung; Realistische Vorhersage von Cyberattacken; *iX* 3/2020, S. 120

[5] Reimar Karlsburger, Mark Sobol; Datenschutz mit Programm; Data-Discovery- und Data-Leakage-Prevention-Tools; *iX* 9/2018, S. 38

[6] Fabian Murer; Protokollschätze; Incident Response und Forensik – Angreifer durch Logs enttarnen; *iX* 8/2021, S. 94

[7] Hans-Joachim Knobloch; Und ewig grüßt das Nilpferd; PetitPotam und weitere Wege, die Kontrolle über das AD zu übernehmen; *iX* 9/2021 S. 91

[8] Hans-Martin Münch; Mein Name ist Hase; Kompromittierung von Windows durch LLMNR Spoofing und NTLM Relaying; *iX* 10/2016, S. 106

[9] Stefan Strobel; Schlaue Hilfe; Threat Intelligence – neuer Hype oder wirksam gegen Cyberrisiken?; *iX* 8/2015, S. 92

[10] Martin Wundram, Alexander Sigel; Aus Fehlern lernen; Organisatorische und technische Maßnahmen zum IT-Selbstschutz; *iX* 2/2021, S. 48

[11] Die im Artikel angesprochenen Blogeinträge, Werkzeuge und weiteren Materialien sind über ix.de/zbmf zu finden.

Frank Ully

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