Projekt Luna

Red Teaming

Ergebnisbericht

MindBytes GmbH | Probststraße 15 | 70567 Stuttgart

+49 711 20709567 | hallo@mind-bytes.de

Geschäftsführung: Christian Stehle, Nina Wagner, Simon Holl
HRB 790784 | Amtsgericht Stuttgart

Version 1.0
Vertraulich
Kontakt: hallo@mind-bytes.de

Sample Company GmbH

Inhalt

1 Management Summary

Im Rahmen eines Red-Teaming-Projekts wurde ein realitätsnaher Angriff auf Sample Company durchgeführt, um Erkenntnisse über die Angriffserkennung, Angriffsabwehr und mögliche Schwachstellen zu gewinnen. Das Projekt umfasste zwei Teile:

- Von außen ins interne Firmennetz eindringen
- Von einem internen System die Kontrolle über die interne IT-Umgebung (Active Directory) erlangen

Ergebnis: Erreicht werden konnte nur eines der beiden Ziele – das Eindringen in die interne Firmenumgebung. Wir stufen das Sicherheitsniveau der IT-Umgebung und der Angriffserkennungsmaßnahmen als überdurchschnittlich gut ein.

Trotzdem wurden wichtige Erkenntnisse gewonnen, einige davon mit hohem Risiko:

- Schwachstellen ermöglichen die Übernahme einiger OT-Systeme in London, Paris und Tokio
- Weitere Findings betreffen folgende Aspekte:
  - Umgehung implementierter Schutzmechanismen
  - Lücken in der Angriffsüberwachung
  - Preisgabe von Informationen

Das zeigt, dass bestimmte Sicherheitsmechanismen effektiv waren, aber auch Verbesserungspotenzial besteht.

Handlungsbedarf: Dringend

Gesamtrisiko im Vergleich zu anderen Unternehmen1[1]: Besser

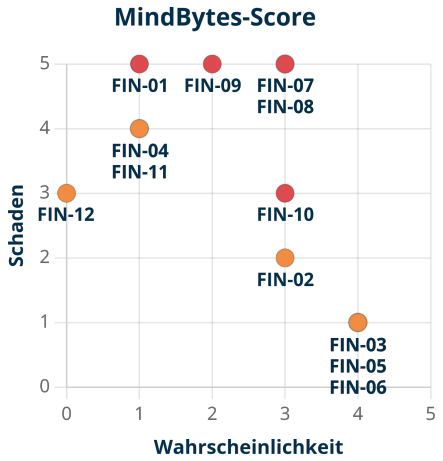


Abbildung 1 - Verteilung nach Schaden und Wahrscheinlichkeit

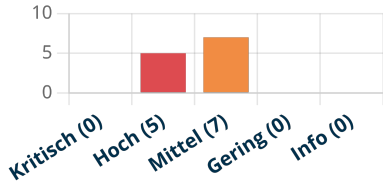


Abbildung 2 - Verteilung nach Risiko

1.1 Handlungsempfehlung

Die Einschätzung zur Behebung basiert auf unserer Erfahrung und sollte intern validiert werden. In der Regel resultieren erfolgreiche Angriffe aus der Verkettung von mehreren Schwachstellen, weshalb wir eine Behebung aller Findings empfehlen. Beim Umsetzen von Gegenmaßnahmen ist es wichtig, Schwachstellen nicht als Einzelfall zu betrachten, sondern an der Ursache zu arbeiten, um ähnlichen Schwachstellen in der Zukunft vorzubeugen.

Maßnahme	Behebung	Hinweise zur Behebung	Findings
Netzwerk segmentieren	Dringend Wochen Nein	▪ OT-Umgebung netzwerktechnisch von Office-Systemen wie der Citrix-Umgebung trennen, um Angriffen vorzubeugen	4.7 FIN-07: Intern: Erreichbarkeit der OT-Systeme aus Citrix-Umgebung
Passwort in Gruppenrichtlinie ändern	Dringend Tage Nein	▪ Gruppenrichtlinie deaktivieren und betroffene Systeme bereinigen	4.8 FIN-08: Intern: Verwendung von Passwort aus Gruppenrichtlinie
Client-Sicherheit stärken	Mittelfristig Wochen Vermutlich	▪ Behebung der Findings trägt dazu bei, die (unbemerkte) Ausführung von Schadprogrammen zu erschweren	4.9 FIN-09: Intern: Lücken in AppLocker-Konfiguration 4.11 FIN-11: Intern: Einsatz alter CrowdStrike-Version
Monitoring ausweiten	Mittelfristig Wochen Vermutlich	▪ Mehr Komponenten in das Monitoring einbeziehen ▪ Kann strukturelle Anpassungen und/oder die Einführung neuer Tools bedeuten	4.1 FIN-01: Extern: Breached Credentials 4.6 FIN-06: Extern: Blinder Fleck: Monitoring der Webanwendungen 4.12 FIN-12: Intern: Unbemerkter Aufbau eines Command-and-Control-Channels

[1] Dies ist eine relative Einschätzung und lässt keine Rückschlüsse auf die Gefährdungslage zu.

Maßnahme	Behebung	Hinweise zur Behebung	Findings
Preisgegebene Informationen verringern, Komponenten aktualisieren	⌚ Mittelfristig ⌛ Wochen 💰 Vermutlich	▪ Veraltete Software aktualisieren ▪ Informationspreisgaben regelmäßig überwachen und auf das technisch notwendige Maß reduzieren	4.2 FIN-02: Extern: Metadaten in Dokumenten 4.3 FIN-03: Extern: Mail-Adressen verifizierbar 4.4 FIN-04: Extern: Einsatz veralteter Software 4.5 FIN-05: Extern: Offenlegung von internen Hostnamen 4.10 FIN-10: Intern: Sensible Daten auf Netzwerk-Share

⌚ Priorität: dringend / mittelfristig / langfristig | ⌛ Geschätzte Behebungsdauer je Finding: Stunden / Tage / Wochen | 💰 Entstehen Kosten: nein / vermutlich (nicht) / ja

2 Technical Summary

2.1 Findings-Tabelle

Die folgende Tabelle listet die identifizierten Befunde auf. Diese sind nach internen und externen Teilen der Bewertung getrennt und basieren auf ihrem Risikoniveau sortiert.

Finding	MindBytes-Score Risiko	MindBytes-Score Schaden	MindBytes-Score Wahrscheinlichkeit
4.1 FIN-01: Extern: Breached Credentials 💡 Regelmäßige Prüfung auf neu veröffentlichte Zugangsdaten	Hoch	🔥🔥🔥🔥🔥	🎲🎲🎲🎲🎲
4.2 FIN-02: Extern: Metadaten in Dokumenten 💡 Metadaten der Dokumente vor dem Veröffentlichen bereinigen	Mittel	🔥🔥🔥🔥🔥	🎲🎲🎲🎲🎲
4.3 FIN-03: Extern: Mail-Adressen verifizierbar 💡 Härtung des Mailservers	Mittel	🔥🔥🔥🔥🔥	🎲🎲🎲🎲🎲
4.4 FIN-04: Extern: Einsatz veralteter Software 💡 Verwenden der aktuellen Version des JavaFy-Frameworks	Mittel	🔥🔥🔥🔥🔥	🎲🎲🎲🎲🎲
4.5 FIN-05: Extern: Offenlegung von internen Hostnamen 💡 Nutzung einer internen CA, Vermeidung von Fehlermeldungen	Mittel	🔥🔥🔥🔥🔥	🎲🎲🎲🎲🎲
4.6 FIN-06: Extern: Blinder Fleck: Monitoring der Webanwendungen 💡 Flächendeckende Monitoring-Einführung	Mittel	🔥🔥🔥🔥🔥	🎲🎲🎲🎲🎲
4.7 FIN-07: Intern: Erreichbarkeit der OT-Systeme aus Citrix-Umgebung 💡 Einführung und Erzwingung einer Netzwerksegmentierung	Hoch	🔥🔥🔥🔥🔥	🎲🎲🎲🎲🎲
4.8 FIN-08: Intern: Verwendung von Passwort aus Gruppenrichtlinie 💡 Deaktivieren oder Löschen der Gruppenrichtlinie	Hoch	🔥🔥🔥🔥🔥	🎲🎲🎲🎲🎲

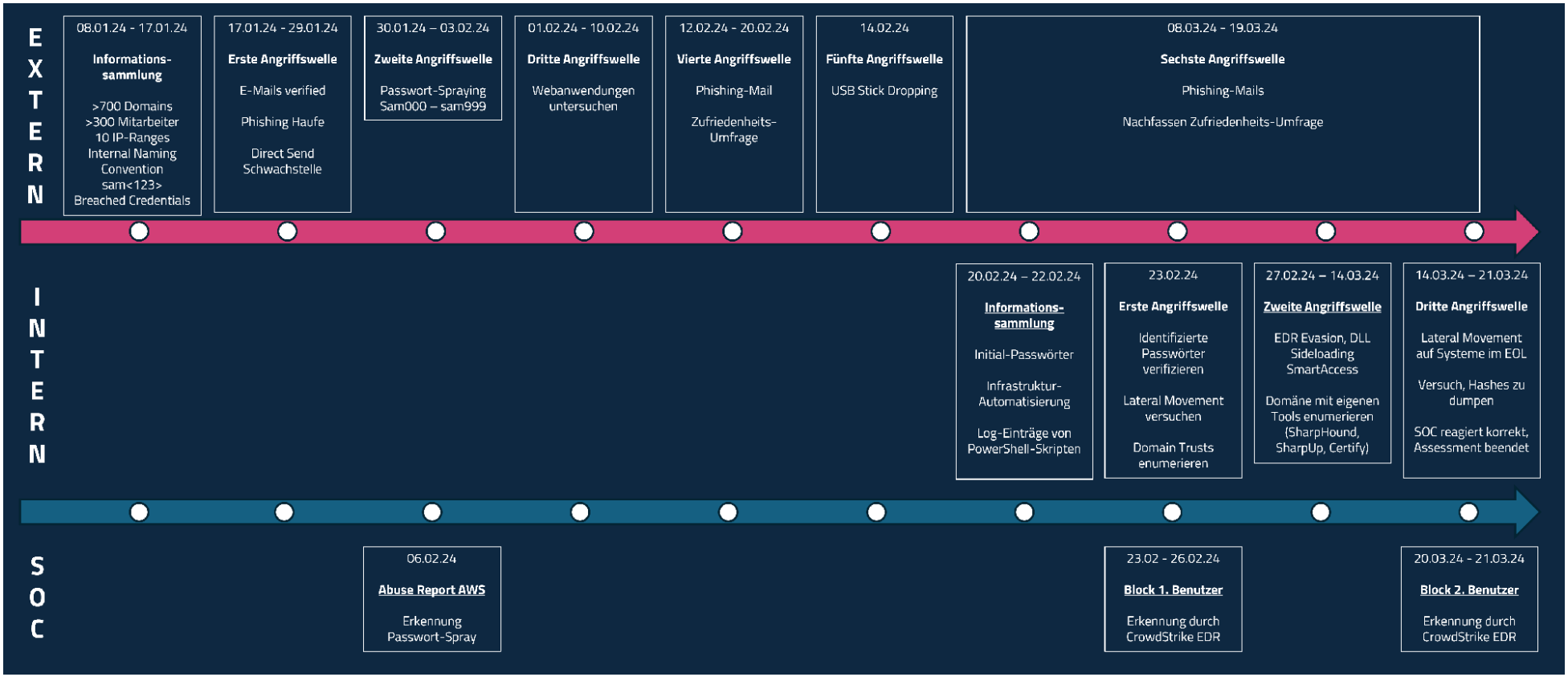
Finding	MindBytes-Score Risiko	MindBytes-Score Schaden	MindBytes-Score Wahrscheinlichkeit
4.9 FIN-09: Intern: Lücken in AppLocker-Konfiguration 💡 Bereinigen der Regeln, Einsatz von WDAC	Hoch		
4.10 FIN-10: Intern: Sensible Daten auf Netzwerk-Share 💡 Bereinigung der Shares	Hoch		
4.11 FIN-11: Intern: Einsatz alter CrowdStrike-Version 💡 Aktualisieren/Upgraden auf CrowdStrike-Agent ab 6.0	Mittel		
4.12 FIN-12: Intern: Unbemerkter Aufbau eines Command-and-Control-Channels 💡 Erkennung für gängige C2-Channel implementieren	Mittel		

Details zu den einzelnen Findings sind im Kapitel 4 Findings beschrieben. Diesem Bericht liegen folgende Dateien bei:

🔎 RedTeamDatenbasis.xlsx:
Technische Details, die während des Red Teamings gesammelt und als relevant eingestuft wurden. Nähere Informationen sind im Anhang in den Abschnitten 6.3 Gesammelte Informationen und 6.4 Red Team Activity Log zu finden.

3 Angriffsnarrativ

In diesem Kapitel werden die durchgeführten Aktionen, Erkenntnisse und Vermutungen chronologisch wiedergegeben. Die während des Assessments gefundenen Schwachstellen sind im Kapitel 4 Findings beschrieben.

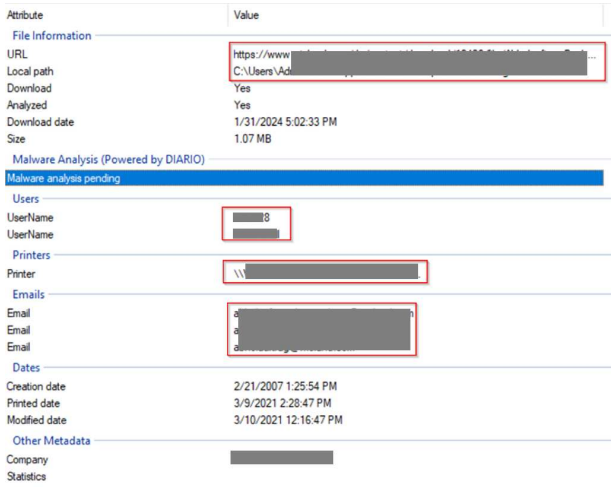


3.1 Erster Teil: Extern – Initial Access

3.1.1 Informationssammlung: Bestimmung der Angriffsoberfläche (08.01.–17.01.2024)

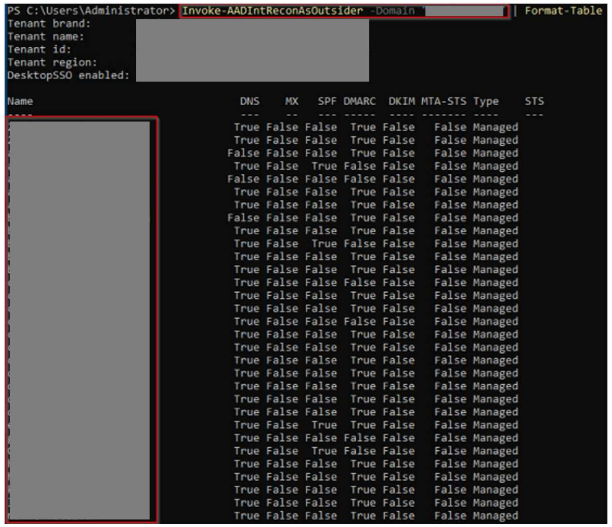
Im ersten Schritt sammelten wir so viele Informationen wie möglich, um mögliche Angriffspunkte zu finden. Hierbei starteten wir mit Open Source Intelligence (OSINT), das heißt, wir sammelten alle Informationen zum Kunden, die öffentlich einsehbar waren.

Dabei trugen wir insgesamt über 700 Domänen, 10 Netzwerk-Ranges und 336 Mitarbeiter mit Vor- und Nachnamen zusammen. Eine nähere Untersuchung der Systeme ergab, dass **M365-Logins häufig verwendet** wurden. Außerdem vermuten wir, dass die Netzwerkblöcke in **London zu On-Premise-Netzwerken gehören** und eine Verbindung ins interne Office-Netz haben. Weitere Netzwerke bei externen Hostern wurden nicht berücksichtigt, da diese aus unserer Sicht nicht zum Ziel – dem internen Netzwerk – führen würden.



Mit dem Tool FOCA analysierten wir Metadaten von veröffentlichten Dokumenten. Hierbei werden verschiedene Suchmaschinen benutzt, um Dokumente bestimmter Formate (.pdf, .docx usw.) zu finden und diese dann automatisiert herunterzuladen und zu analysieren. In den Metadaten fanden wir wertvolle Informationen: einerseits verschiedene Mitarbeiternamen sowie eine **ID im Format sam000**. Bei dieser ID vermuten wir, dass es sich um ein internes Namensschema handelt, beispielsweise eine Benutzerkennung im Active Directory.

Eine Untersuchung des M365-Tenants offenbarte viele weitere verknüpfte Domänen, die wir in unsere weitere Recherche mit aufnahmen. Diese Informationen lassen sich ohne Authentifizierung basierend auf der Domäne *samplecorp.com* auslesen.



Außerdem suchten wir nach sogenannten Breached Credentials. Das sind veröffentlichte Zugangsdaten von Webseiten, die kompromittiert wurden. Wir benutzten den Dienst *dehashed.com* und fanden **200 veröffentlichte Zugangsdaten** für die E-Mail-Domänen samplecorp.de und samplecorp.com. Auffällig war, dass mehrere Passwörter dem Schema „VornameSamplecorp“ folgten. Wir vermuten, dass es sich hierbei um ein Initialpasswort handeln könnte. Wir testeten die Zugangsdaten am öffentlich erreichbaren M365-Login-Dienst, hatten aber **keinen Erfolg**. Wie wir später herausfanden, wird für das Login nicht die E-Mail-Adresse mit dem Vor- und Nachnamen genutzt, sondern das interne Namensschema (sam<*3-stellige Nummer*>). In den Breached Credentials waren keine solchen E-Mail-Adressen (samxxx@samplecorp.com) enthalten. In den Netzblöcken prüften wir die aktiven Dienste und fanden fast ausschließlich Webanwendungen vor. Hierbei fiel auf, dass einige davon mit einem heute veralteten Framework (JavaFy 7.7.17) selbstentwickelt wurden.

Nachdem wir uns auf diese Weisen einen **Überblick** verschafft hatten, ergaben sich die **folgenden Angriffsvektoren**:

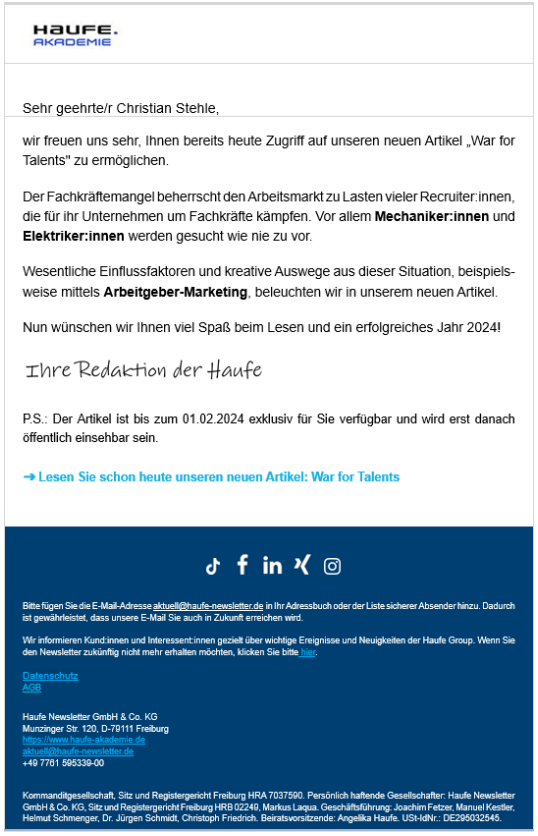
1. Phishing
2. Angriffe auf Webanwendungen
3. Passwort-Spraying
4. USB-Stick-Dropping

Relevante APT-Gruppen (Advanced Persistent Threat, Bezeichnung für eine organisierte Angreifer-Gruppe) setzen auf Phishing als effektive Methode für den initialen Zugang zu einem Unternehmen. Deshalb war dies auch unser erster Angriffsvektor. Bei der Recherche stießen wir vermehrt auf vorgeschaltete Microsoft-Logins, weshalb das Ziel unserer Phishing-Kampagnen war, an Microsoft-Zugangsdaten zu gelangen. Wir vermuteten, dass darüber verschiedene Remote-Access-Lösungen für das interne Netz, wie beispielsweise VPN- und Citrix-Endpunkte, nutzbar oder Zugriff auf sensible Informationen möglich wären.

3.1.2 Erste Angriffswelle (17.01.–29.01.2024)

Zuerst versuchten wir, gültige E-Mail-Adressen zu finden. Die bisherige Recherche ergab, dass das **E-Mail-Format *Vorname.Nachname@sample.com*** war. Nachdem wir in der Informationssammlungsphase über 300 Personen mit Vor- und Nachnamen identifiziert hatten, konnten wir auch die fehlenden E-Mail-Adressen ableiten. Da einige dieser Personen möglicherweise nicht mehr bei SampleCompany arbeiteten, versuchten wir, die **E-Mail-Adressen direkt am Mail-Server zu überprüfen**. Dies gelang, wie in 4.3 FIN-03: Extern: Mail-Adressen verifizierbar beschrieben.

Als Phishing-Szenario wählten wir einen **Newsletter-Artikel, der für Mitarbeitende aus dem Personalwesen relevant** war. Da laut Stellenanzeigen häufig Mechaniker und Elektriker gesucht wurden, erstellten wir basierend darauf eine Phishing-E-Mail. Darin **imitierten wir den Stil von Haufe**, einem bekannten Anbieter im Personalbereich, der auch einen Newsletter hat. Adressierte Mitarbeitende könnten legitime Haufe-E-Mails deshalb schon kennen und unsere Phishing-E-Mail weniger hinterfragen. Die erstellte Phishing-E-Mail ist in der Abbildung zu sehen.


HAUFE. AKADEMIE

Sehr geehrte/r Christian Stehle,

wir freuen uns sehr, Ihnen bereits heute Zugriff auf unseren neuen Artikel „War for Talents" zu ermöglichen.

Der Fachkräftemangel beherrscht den Arbeitsmarkt zu Lasten vieler Recruiter:innen, die für ihr Unternehmen um Fachkräfte kämpfen. Vor allem **Mechaniker:innen** und **Elektriker:innen** werden gesucht wie nie zu vor.

Wesentliche Einflussfaktoren und kreative Auswege aus dieser Situation, beispielsweise mittels **Arbeitgeber-Marketing**, beleuchten wir in unserem neuen Artikel.

Nun wünschen wir Ihnen viel Spaß beim Lesen und ein erfolgreiches Jahr 2024!

Ihre Redaktion der Haufe

P.S.: Der Artikel ist bis zum 01.02.2024 exklusiv für Sie verfügbar und wird erst danach öffentlich einsehbar sein.

➔ Lesen Sie schon heute unseren neuen Artikel: War for Talents

Bitte fügen Sie die E-Mail-Adresse aktuell@haufe-newsletter.de in Ihr Adressbuch oder der Liste sicherer Absender hinzu. Dadurch ist gewährleistet, dass unsere E-Mail Sie auch in Zukunft erreichen wird.

Wir informieren Kund:innen und Interessent:innen gezielt über wichtige Ereignisse und Neuigkeiten der Haufe Group. Wenn Sie den Newsletter zukünftig nicht mehr erhalten möchten, klicken Sie bitte hier.

Datenschutz
AGB

Haufe Newsletter GmbH & Co. KG
Munzinger Str. 120, D-79111 Freiburg
https://www.haufe-akademie.de
aktuell@haufe-newsletter.de
+49 7761 595339-00

Kommanditgesellschaft, Sitz und Registergericht Freiburg HRA 7037590. Persönlich haftende Gesellschafter: Haufe Newsletter GmbH & Co. KG, Sitz und Registergericht Freiburg HRB 02240, Markus Laqua. Geschäftsführung: Joachim Fetzer, Manuel Kestler, Helmut Schmenger, Dr. Jürgen Schmidt, Christoph Friedrich. Beiratsvorsitzende: Angelika Haufe. USt-IdNr.: DE295032545.

Die E-Mail lockte die Empfänger mit einem exklusiven, neu veröffentlichten Artikel. Das sollte der Grund sein, warum bei **Einsicht des Artikels zunächst eine Microsoft-Authentifizierung** stattfinden musste. Beim Klick auf den enthaltenen Link wurde der Empfänger auf eine von uns kontrollierte Seite geleitet, die den Microsoft-Login nachbildete und im Hintergrund einen echten Login bei Microsoft durchführte. Bei konfiguriertem Multi-Faktor-Login wurde auch das benötigte Token abgefragt und eine echte Microsoft-Sitzung generiert. Bei Erfolg wären wir sowohl im Besitz des Benutzernamens und Passworts als auch einer gültigen Microsoft-Sitzung, die wir vermutlich an verschiedenen von Sample-Company genutzten Diensten nutzen könnten. Um interne Einschränkungen beim Zugriff ins Internet zu umgehen, wurde die erstellte Phishing-Seite über das Azure-CDN (azureedge.net) ausgeliefert.

Beim Versenden der E-Mails stellten wir jedoch fest, dass der **Mail-Server uns zurückwies**. Die genaue Fehlermeldung ist unten aufgeführt.

Mehrere Anpassungen, wie etwa ein **gesetzter A-Record** für die Phishing-Domäne, ein **Wechsel des Mail-Providers** und das **Alter der Phishing-Domäne** von mehr als 30 Tagen, führten im weiteren Verlauf des Red Teamings dazu, dass die Phishing-E-Mails vom Mail-Server angenommen wurden. Allerdings erhielten wir von den Empfängern **keinerlei**

Reaktionen, weder Aufrufe unseres Links noch Meldungen über Phishing. Nachdem es keine Reaktion auf die Phishing-E-Mails gab, versendeten wir eine Bewerbung von einer privat wirkenden Adresse (x.y@z.de) aus, aber auch diese E-Mail wurde vom Mail-Server zurückgewiesen. Wir untersuchten das Mail-System genauer und fanden heraus, dass die Mail-Security-Lösung Cisco IronPort eingesetzt wird. Die Reputation einer E-Mail konnte auf der Website *Talos Intelligence* von Cisco überprüft werden.

Außerdem **versuchten wir, E-Mails auf einem anderen Weg zuzustellen**. Bei der Informationssammlung identifizierten wir den M365-Tenant *samplecompany* und das Exchange Online Gateway unter samplecompany-com.mail.protection.outlook.com. Oftmals kann das dazu genutzt werden, (Phishing-)E-Mails am vorgeschalteten Mail-Security-Gateway vorbei zu senden. Allerdings **schlug dies fehl**, da das Exchange Online Gateway nur von ausgewählten Systemen E-Mails akzeptierte.

3.1.3 Zweite Angriffswelle (30.01.–03.02.2024)

Als nächsten Angriff führten wir ein **Passwort-Spraying** durch. Hierbei wird ein Passwort bei mehreren Accounts ausprobiert, um Sperrungen durch mehrere Fehlversuche beim gleichen Account zu vermeiden. Zu diesem Zeitpunkt hatten wir beim internen Namensschema mehrere Ziffernblöcke identifiziert. Wir versuchten, durch Hochzählen der Ziffern gültige Accounts zu erraten. Die Login-Maske beim Microsoft-Login bestätigte außerdem, dass der Login bei M365 nicht über die E-Mail-Adresse, sondern durch die Eingabe der internen Account-ID erfolgte.

Für die Durchführung mieteten wir eine AWS-Instanz und setzten das Tool *Invoke-MSOLSpray* ein.

Wir prüften alle theoretisch möglichen Benutzernamen mit dem **Passwort „SampleCompany2023!"** aus den folgenden identifizierten Benutzer-Blöcken:

- **sam000–sam999**

Um die Angriffe nicht zu offensichtlich zu machen, wurde nach jedem Versuch eine Pause von 30 Sekunden eingehalten. Wir konnten während des ca. 3 Tage andauernden Passwort-Sprayings mit insgesamt 1000 durchprobierten Benutzernamen **keine Sperrung unserer IP-Adresse** feststellen.
Basierend auf den Rückmeldungen der Login-Versuche konnten wir ca. **500 gültige Benutzerkonten identifizieren**, jedoch hatte keins davon das von uns versuchte Passwort gesetzt.
6 Tage nach dem Start des Passwort-Sprayings erhielten wir einen **Abuse-Report von AWS**, da das Passwort-Spraying im Monitoring aufgefallen war. Dies war **die erste von uns beobachtbare Reaktion des SOCs**.

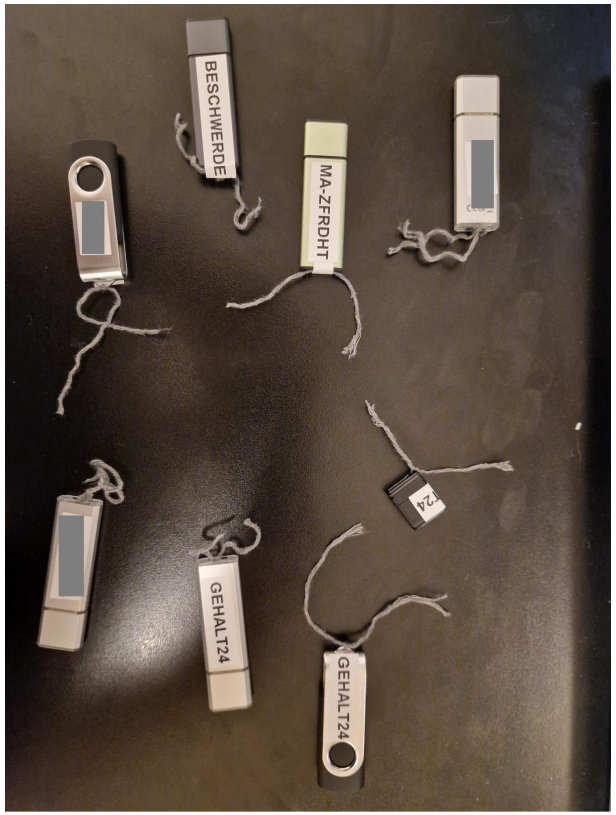
3.1.4 Dritte Angriffswelle (01.02.–10.02.2024)

Als Nächstes widmeten wir uns den **Webanwendungen in den relevanten Netzblöcken**. Zunächst untersuchten wir diese vorsichtig und unauffällig. Nachdem wir **keinerlei Einschränkungen in Form eines Schutzsystems** feststellen konnten, führten wir immer aggressivere und offensichtlichere Prüfungen durch. Wir vermuten, dass es keinerlei Schutzsystem gab, das Angriffe solcher Art erkannt und abgewehrt hätte. Bei der Untersuchung der Anwendungen gelangten wir an verschiedene **Informationen über interne Hostnamen und die interne Microsoft-Umgebung**, siehe 4.5 FIN-05: Extern: Offenlegung von internen Hostnamen und 4.4 FIN-04: Extern: Einsatz veralteter Software. Allerdings hatten wir keinen Erfolg bei der Kompromittierung eines Systems über eine Webanwendung.

3.1.5 Vierte Angriffswelle (12.02.–20.02.2024)

Da alle vorherigen Phishing-Versuche erfolglos waren, starteten wir eine **neue Phishing-Kampagne**. Diesmal fassten wir die Zielgruppe jedoch breiter und wählten eine Zufriedenheitsumfrage als Szenario. Wir imitierten eine erfundene interne Person, Barbara Rhabarber, die zur Teilnahme an der Umfrage aufrief. Unser Ziel war es weiterhin, M365-Logins zu erhalten. Relevante E-Mail-Elemente, wie beispielsweise die Signatur, erhielten wir, indem wir eine anonyme Kontaktanfrage über eine Webanwendung stellten und von dort eine Antwort erhielten. Die Mail wurde vom Absender barbara.rhabarber@samplecompany.com versendet.

3.1.6 Fünfte Angriffswelle (14.02.2024)

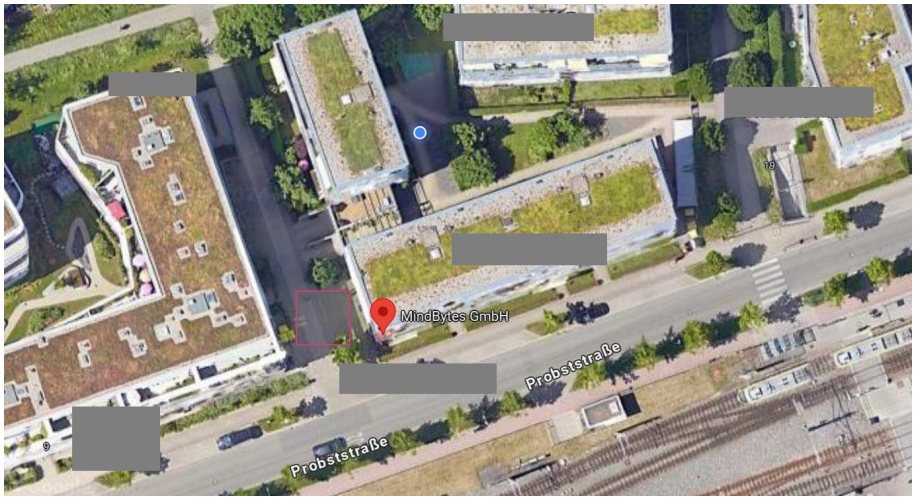


Wir entschieden uns, als nächsten Angriffsvektor **USB-Sticks** auf den Parkplätzen der Standorte zu platzieren. Ziel war es, die Neugier der Mitarbeitenden zu wecken, sodass diese die USB-Sticks in die internen Systeme einstecken und unsere Software ausführen. Wir versahen die USB-Sticks daher mit interessant wirkenden Texten, wie etwa „Gehalt" und „Beschwerde". Außerdem versahen wir die USB-Sticks mit abgerissenen Fäden, sodass es wirkte, als wären sie einer Person heruntergefallen.

Durch die Informationssammlung hatten wir einige Informationen über die internen Gegebenheiten, etwa dass die Active Directory Domäne INTERN.SAMPLE.COM lautete. Wir bereiteten den **Payload so vor, dass dieser nur ausgeführt wird, wenn er sich in dieser Domäne befindet**. Andernfalls beendet er sich sofort. Das sorgte dafür, dass sich unsere Schadsoftware nicht auf privaten Geräten oder auf Geräten von Mitarbeitenden anderer Firmen ausführen ließ. Außerdem erschwert das eine Erkennung bei der Untersuchung der Schadsoftware in einer gesonderten Umgebung.

Als Payload wurde eine Verknüpfungsdatei (Gehälter2024.lnk) erstellt, die eine ebenfalls auf dem USB-Stick befindliche und von Microsoft signierte .exe-Datei ausführt. Unsere eigentliche bösartige Payload in Form einer DLL wird beim Ausführen der .exe-Datei geladen. Das hat den Hintergrund, dass wir diverse Schutzmechanismen, wie AppLocker, vermuteten. AppLocker erlaubt standardmäßig die Ausführung von Dateien, die von Microsoft signiert sind. Der Payload baut eine Command-and-Control-Verbindung auf. Die Kommunikation würde verschlüsselt per HTTPS über das Azure-CDN laufen. Das sollte interne Einschränkungen der Kommunikation ins Internet vorbeugen. Das Microsoft-CDNs wurde gewählt, weil in der Informationssammlungsphase festgestellt wurde, dass M365 genutzt wird. Die .exe- und .dll-Dateien wurden mit dem Attribut „versteckt" versehen, sodass diese standardmäßig im Windows-Explorer nicht sichtbar waren. Beim Einstecken des Sticks erschien also nur eine Verknüpfungsdatei mit dem Symbol eines Texteditors.

Es wurden insgesamt 7 USB-Sticks an den Standorten London, Paris und Tokio auf den Mitarbeiter-Parkplätzen platziert. Die Orte der 4 USB-Sticks, die am Standort London platziert wurden, sind in der Abbildung zu sehen.



Dieser Angriffsvektor **war erfolgreich**, und zwei Command-and-Control-Verbindungen wurden hergestellt. Damit wurde der interne Zugriff erlangt, sodass dies den Erfolg der ersten Projektphase markiert.

3.1.7 Sechste Angriffswelle (08.03.–19.03.2024)

Um weitere Zugänge zum internen Netzwerk zu schaffen, **wurden weitere isolierte Phishing-Kampagnen** durchgeführt. Bestehende Kampagnen wurden wiederholt oder leicht modifiziert. Diese waren jedoch nicht erfolgreich.

3.2 Zweiter Teil: Intern – Escalate Privileges

Am 20. Februar beschlossen wir in Absprache mit unserer Kontaktperson, zum **zweiten Teil des Projekts** überzugehen. Wir nutzten unsere erfolgreich gewonnenen anfänglichen Zugänge aus (siehe3.1.5 Vierte Angriffswelle (12.02.–20.02.2024)).

3.2.1 Informationssammlung: Bestimmung der Angriffsvektoren (20.02.–22.02.2024)

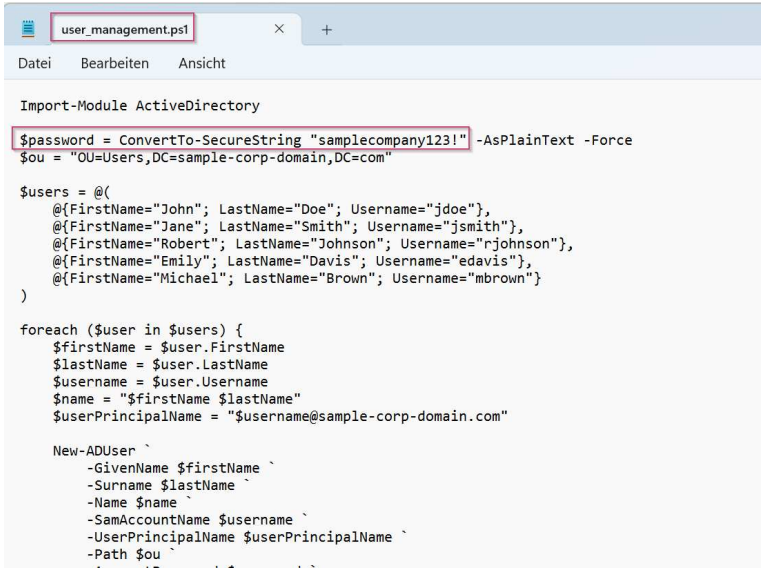
Für den zweiten Projektteil wurden Zugänge bereitgestellt, wie wir sie durch Phishing versucht hatten zu erhalten. Mit diesen Zugangsdaten konnten wir uns dann über das Citrix-Portal anmelden und so im Assume-Breach-Ansatz fortfahren.

Zunächst wurde das lokale System händisch untersucht. Hierbei stellten wir fest, dass CrowdStrike als EDR eingesetzt wird, es verschiedene Anwendungen wie SAP und Microsoft Office gibt und AppLocker vorhanden ist.

Wir versuchten, eine CrowdStrike-EDR-Instanz in unserem Labor zu installieren, um das Verhalten des EDRs zu analysieren und mögliche Umgehungen zu finden. Wir kontaktierten den Hersteller, Vertriebspartner und Partnerfirmen, jedoch war es uns nicht möglich, eine Demo-Instanz zu erhalten. Das erschwerte uns das Obfuskieren unserer Payloads deutlich, sodass es uns nicht möglich war, eine Erkennung unserer Payload durch CrowdStrike in der eigenen Labor-Umgebung zu testen.

Um **unauffällig** zu bleiben, durchsuchten wir manuell Netzwerkshares, vor allem den NETLOGON-Share. Hierbei fanden wir einige **Informationen, die für Angriffe nützlich** sein könnten, siehe 4.10 FIN-10: Intern: Sensible Daten auf Netzwerk-Share. Dazu zählten Klartextpasswörter in Skripten, die SMB-Shares einbinden, und verschlüsselte Passwörter in Automatisierungsskripten. Außerdem wurden auf dem Austauschlaufwerk X:\Austausch interessante Dateien wie Backups und private Schlüssel gefunden.

Eine große Anzahl an PowerShell-Skripten lag unter C:\ProgramFiles\SampleCompany\ und \samplesshareserver\Scripts. Wir vermuteten, dass die Administration der Domäne hauptsächlich mit PowerShell durchgeführt wird. Außerdem gaben diese PowerShell-Skripte einige Informationen über die Interna der Domäne, Benutzerverwaltung und eingesetzte Software preis. Es fanden sich an mehreren Stellen Log-Einträge zu den ausgeführten Skripten, sodass wir gut nachverfolgen konnten, wann und in welchem Kontext die Skripte ausgeführt wurden.



Ein **Passwort des Benutzers _*SampleManagement* konnten wir entschlüsseln**, wie in 4.10 FIN-10: Intern: Sensible Daten auf Netzwerk-Share beschrieben. Später stellten wir fest, dass das Passwort geändert wurde und das zugehörige Skript somit veraltet ist.

Um Details zur Domäne abzufragen, nutzten wir das auf dem System vorhandene Tool *dsa.msc*. Mit diesem Tool lassen sich die Benutzer und Computer des Active Directory auflisten. So konnten wir Details zu Benutzern und Computern, wie beispielsweise Gruppenzugehörigkeiten, das Datum der letzten Passwortänderung (pwdLastSet) oder das letzte Anmeldedatum von Computern (lastLogon), abfragen.

3.2.2 Erste Angriffswelle (23.02.2024)

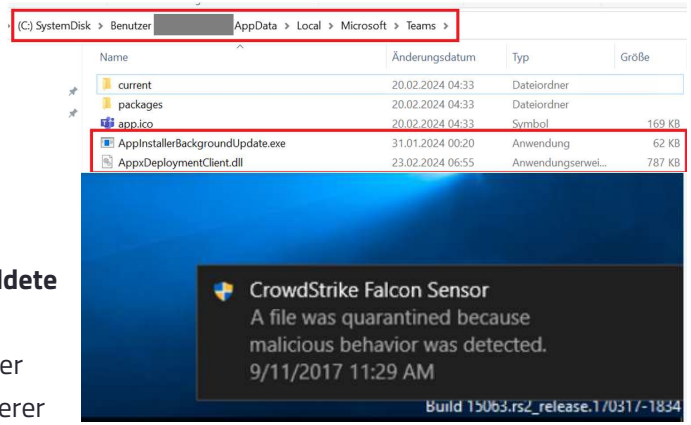
Nachdem wir feststellten, dass unser Citrix-System gut gehärtet schien, versuchten wir, auf ein anderes System zu gelangen. Das **Ziel war, auf ein System zu gelangen, das weniger gehärtet** und insbesondere nicht in der Gruppe *_sample_server_ENHANCEDSECURITY* war. Wir versuchten deshalb, uns auf einem anderen Citrix-System anzumelden. Üblicherweise werden den Citrix-Benutzern auch RDP-Rechte auf den Terminalservern zugewiesen. Das war jedoch nicht der Fall, und wir konnten uns auf diese Weise nicht lateral zu weniger überwachten Citrix-Systemen bewegen.

Anstelle von lateralen Bewegungen zwischen Systemen versuchten wir nun den Sprung zu einem anderen Benutzer. Dazu versuchten wir eine Anmeldung mit den zuvor ermittelten Zugangsdaten des Benutzers *_SampleManagement*. Um das unauffällig und losgelöst von unserem bisherigen Benutzer durchzuführen, nutzten wir den Windows-Login-Bildschirm, der uns beim Verbinden mit Citrix präsentiert wurde. Die Fehlermeldung offenbarte, dass das genutzte Passwort nicht korrekt war. Wir konnten später ermitteln, dass unsere Informationen, aus denen wir das Passwort extrahiert hatten, veraltet waren und das Passwort seitdem geändert wurde.

Danach untersuchten wir die **anderen Domänen**, die uns aufgefallen waren (**sample-subcompany1, sample-subcompany2**). Da der Kunde viele Unternehmen aufgekauft und integriert hat, vermuten wir, dass Domänen aus einer Vertrauensstellung (Trust Relation im Active Directory) weniger überwacht und kontrolliert sind. Unsere Recherche begann auch hier wieder bei den NETLOGON-Shares dieser Domänen. Der bereitgestellte Benutzer hatte die Berechtigungen, diese anzuzeigen und zu durchsuchen. Auch hier fanden sich einige interessante Informationen, aber nichts, was unsere Rechte erweitern würde.

Um mithilfe von Tools effizienter vorgehen zu können, war das nächste Ziel, das eingesetzte EDR zu umgehen. Dazu wurden zunächst mehrere Vorbedingungen geprüft. Wir konnten erfolgreich Dateien von unserem System auf das Citrix-System kopieren. Das stellte das Übertragen unserer Payload sicher. Wir konnten erfolgreich PowerShell starten. Das ermöglichte uns, die AppLocker-Regeln auszulesen. Weiterhin konnten wir **Lücken in den AppLocker-Regeln** identifizieren, die das Ausführen eigener Programme erlaubten, siehe auch 4.9 FIN-09: Intern: Lücken in AppLocker-Konfiguration. Der Aufruf einer azureedge.net-Domäne im Webbrowser war erfolgreich. Die Kommunikation zu dem von uns kontrollierten System im Internet konnte also vermutlich ebenfalls aufgebaut werden.

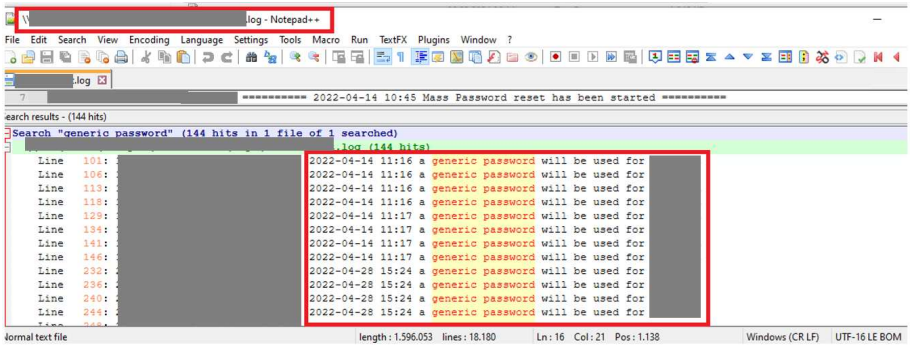
Nachdem diese Vorbedingungen zutrafen, kopierten wir unsere **vorbereitete Payload** in einer verschlüsselten Datei (sample-notes.zip) in die Citrix-Umgebung. Wir nutzten hier das gleiche Prinzip der Payload wie bei den USB-Sticks: Microsoft-signierte .exe-Datei mit DLL-Sideloading. Die übertragenen Dateien entpackten wir in den von den AppLocker-Regeln ausgeschlossenen Pfad C:\Users\johndoe\AppDate\Local\Microsoft\Teams.
Wir stellten fest, dass CrowdStrike die Datei zunächst nicht als schädlich einstufte, sondern unsere Payload weiterhin im Ordner verfügbar war. **Beim Ausführen der Payload schritt das EDR aber ein und meldete eine schädliche Aktivität**. Wie wir später von unserem Ansprechpartner erfuhren, erkannte CrowdStrike die Kombination aus einer Microsoft-signierten .exe-Datei und einer nicht von Microsoft signierten .dll-Datei. Der Schadcode in der .dll-Datei selbst wurde jedoch nicht von CrowdStrike erkannt. Dennoch konnten wir in unserer ersten Angriffsrunde CrowdStrike nicht umgehen und keinen Command-and-Control-Channel aufbauen.



Abschließend versuchten wir, **alternative Kommunikationswege** ins Internet aufzubauen. Dazu wollten wir SSH-Verbindungen zu einem vom uns kontrollierten System aufzubauen. Um das Blockieren von typischen Ports auszuschließen, verwendeten wir mehrere unterschiedliche Ports. Auf den Ports 22, 25, 80, 443 und 12345 schienen erfolgreich TCP-Verbindungen aufgebaut zu werden, jedoch wurde **nie eine SSH-Sitzung aufgebaut**. Wir vermuten, dass die Firewall den Traffic untersucht, unabhängig vom Port klassifiziert und SSH-Verbindungen ins Internet nicht erlaubt sind.

Wir vermuteten, dass der **CrowdStrike-Alarm mehrere Aktionen im SOC auslöste**. Unter anderem wurde unser Benutzeraccount gesperrt. Das geschah nicht sofort. Wir bemerkten die Sperrung beim nächsten Login-Versuch wenige Tage später am 26.03.2024. Das SOC ermittelte unseren internen Ansprechpartner, da dieser die Konten beantragt hatte, und verlangte von ihm eine Erklärung. Um das **Red Teaming weiterführen** zu können, wurde der Sachverhalt intern von unserem Ansprechpartner so geklärt, dass wir den anderen verbleibenden Benutzer weiter nutzen konnten. In diese Kommunikation waren wir nicht involviert. Das Red Teaming setzten wir mit dem zweiten bereitgestellten Benutzer fort. Echte Angreifer können ebenso Zugriff auf mehrere Benutzeraccounts haben, weshalb dieses Vorgehen legitim ist.

3.2.3 Zweite Angriffswelle (27.02.–14.03.2024)

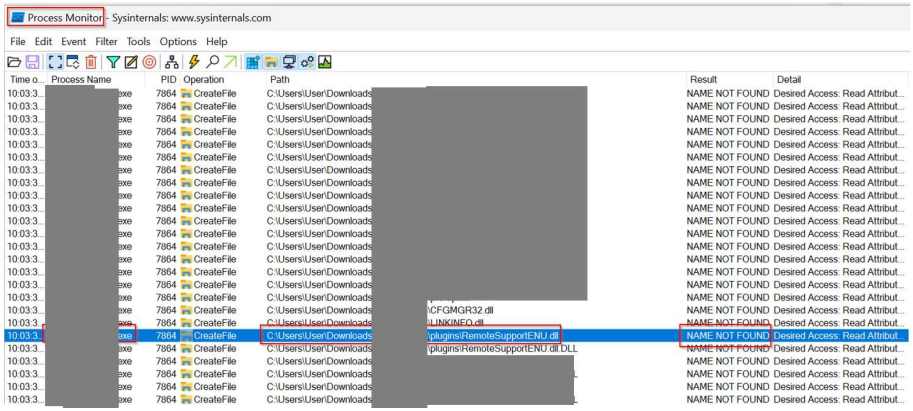


Damit sich weitere Aktionen nicht auf unser Benutzerkonto zurückführen lassen, **versuchten wir, an andere Active-Directory-Benutzerkonten zu gelangen**. Dazu nutzten wir die Erkenntnisse aus der Informationssammlungsphase bezüglich Passwörtern von Benutzern. Mit einer LDAP-Abfrage suchten wir nach Benutzeraccounts im Active Directory, die noch das von den Administratoren vergebene Passwort haben könnten. Dazu verglichen wir das Attribut *pwdLastSet* mit dem in den Log-Einträgen offenbarten Zeitpunkten. Obwohl mehrere Benutzeraccounts das exakt zu dem Zeitpunkt aus den Log-Einträgen gesetzte Passwort noch besaßen, war **kein Login erfolgreich**. Unsere Login-Versuche führten wir teilweise bei office.com, aber auch im Windows-Login-Fenster beim Verbinden mit Citrix durch. Wir vermuten, dass das generische Passwort beim Durchführen des Massen-Zurücksetzens in der grafischen Oberfläche geändert wurde und somit nicht mehr dem vordefinierten Passwort aus dem PowerShell-Skript entsprach.

Als nächsten Schritt unternahmen wir einen **nächsten Versuch, das EDR zu umgehen**. Dazu nutzten wir die zuvor gewonnenen Erkenntnisse und versuchten unsere Payload im Namen einer vorhandenen legitimen Anwendung auszuführen, die nicht von Microsoft oder einem anderen Hersteller signiert war. Unsere Payload war ein Beacon des Command-and-Control-Frameworks *Cobalt Strike*. Da unser Beacon regelmäßig HTTPS-Verbindungen zu einem Azure CDN aufbaute, suchten wir eine Anwendung, die selbst regelmäßig Netzwerkverkehr zu Microsoft herstellt. Wir untersuchten mehrere Anwendungen, und die Wahl fiel auf SmartAccess, da man sich dort mit dem M365-Konto anmeldet und das Programm somit schon einige legitime Verbindungen zu Microsoft-Diensten aufbaut.

Untersuchung SmartAccess im Labor

Wir kopierten die SmartAccess-Anwendung aufgeteilt auf mehrere verschlüsselte ZIP-Archive aus der Citrix-Umgebung in unser Labor. Die Aufteilung wurde vorgenommen, um keine Alarme durch das Kopieren von zu großen Datenmengen zu verursachen. Alle Teilarchive waren weniger als 100 MB groß. Anschließend untersuchten wir die Anwendung auf mögliche Schwachstellen, die DLL-Sideloading ermöglichen. Sollte die Anwendung versuchen, eine DLL nachzuladen, die nicht vorhanden ist, ist das ein möglicher Ansatzpunkt, eine bösartige DLL einzubringen. Für die Analyse wurde das Tool *Process Monitor* aus Microsofts Sysinternals-Sammlung verwendet.



Schließlich fanden wir eine anfällige DLL namens *RemoteAccessENU.dll* im Ordner *plugins*, die sich für unser Vorhaben eignete. In unserer Labor-Umgebung konnten wir so durch das Ausführen von SmartAccess erfolgreich ein Beacon starten.

Ausführung auf dem Citrix System

Zuerst versuchten wir, die DLL im Installationspfad des Programms abzulegen. Dort hatten wir jedoch keine Schreibrechte. Um AppLocker zu umgehen (siehe 4.9 FIN-09: Intern: Lücken in AppLocker-Konfiguration), kopierten wir den gesamten SmartAccess-Ordner nach C:/Users/sampleuser/AppData/Roaming/Citrix/SelfService/, legten unsere DLL im Ordner *plugins* ab und starteten SmartAccess.

Das Beacon baute die Verbindung auf und wir etablierten unseren Command-and-Control-Channel erfolgreich.

Über die so entstandene Verbindung konnten wir weitere Tools nachladen und ausführen.

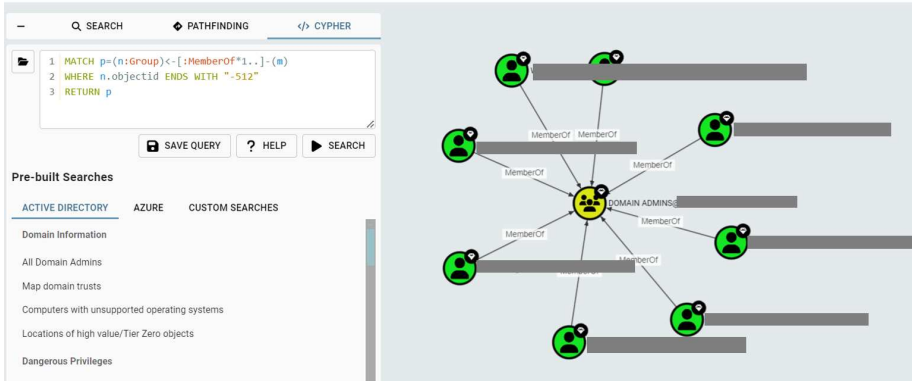
Zunächst ermittelten wir, welche Überwachungsmethoden das EDR einsetzte. Mit sogenannten Hooks können EDR-Systeme die durchgeführten Aktionen überwachen und dementsprechend reagieren. Mit dem Befehl `hooks list` des Beacons konnten wir ermitteln, dass das CrowdStrike-EDR diese Funktionalität nutzt. **Mit dem Befehl `hooks clean` konnten wir diese Überwachung unerkannt entfernen.**

```
[14.3.2024 05:23:16] (finished)[redacted]> hooks list
[!] Possible function hook found in module: ntdll.dll
    -> Function: LdrQueryImageFileExecutionOptionsEx
    -> Address:  0x77D1A1C0

[!] Possible function hook found in module: ntdll.dll
    -> Function: NtCreateFile
    -> Address:  0x77D24320

[!] Possible function hook found in module: ntdll.dll
    -> Function: NtOpenFile
    -> Address:  0x77D24100
```

Als erstes Tool zum Auskundschaften der Umgebung nutzten wir *Bloodhound*, ein Tool zur Enumeration des Active Directory. Hierzu wurde zunächst der Collector *SharpHound* auf dem Citrix-System gestartet. Dieser **sammelte umfassende Informationen aus dem Active Directory** im Kontext unseres Benutzers. Die Daten konnten anschließend heruntergeladen und offline verarbeitet und analysiert werden.



Wir **suchten nach Angriffspfaden im Active Directory**. Eine erste Analyse ergab mehrere Kerberoastable-Accounts. Unserem Benutzer selbst waren keine Berechtigungen zugewiesen, die uns weitergeholfen hätten. Der Erfolg eines Kerberoasting-Angriffs hängt von der Komplexität des Passworts des angreifbaren Kontos ab. Diesen Angriff wollten wir in der dritten Angriffswelle durchführen. Nachdem unser Benutzeraccount durch vorherige Angriffe gesperrt war, konnten wir den Angriff nicht durchführen. Durch die bisher beobachteten vielen starken Passwörter an verschiedenen Stellen schätzten wir die Erfolgschancen als eher gering ein.

Da der **ADCS-Dienst (Active Directory Certificate Services)** oftmals kritische Fehlkonfigurationen enthält, die Benutzern eine Rechteausweitung erlauben, untersuchten wir diesen Dienst als Nächstes. Hierzu wählten wir das Tool *certify*, das die eingesetzten CAs und Zertifikatsvorlagen enumerierte. Auch hier ergaben sich **keine Angriffsmöglichkeiten**. Es waren nur 3 Zertifikatsvorlagen vorhanden und diese konnten nur von Domain-Administratoren genutzt werden.

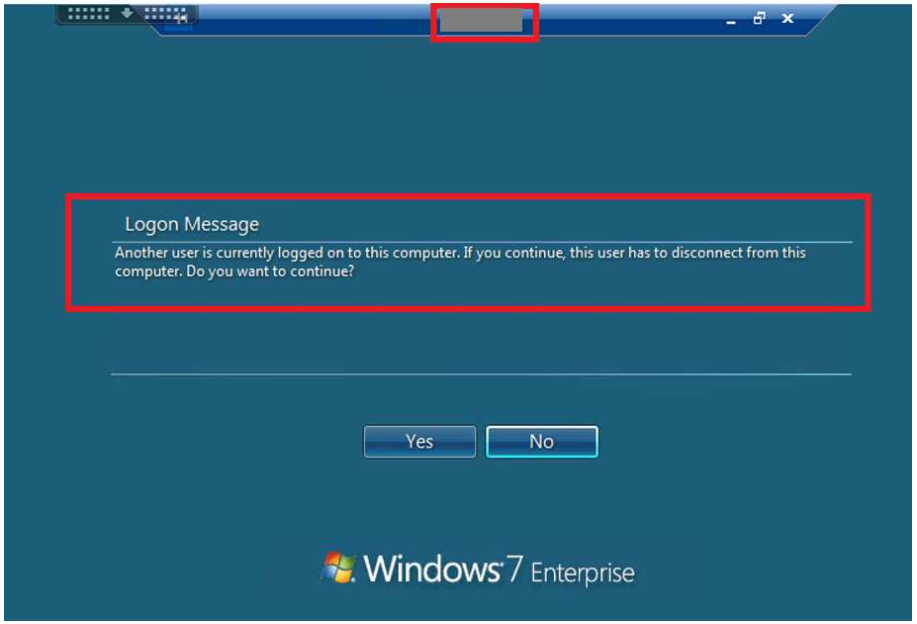
Mit dem Tool *SharpUp* wurden verschiedene Möglichkeiten **geprüft, Benutzerrechte auf dem lokalen System zu erweitern**. Hier fand sich **keine direkte Möglichkeit**. **Allerdings war eine veraltete Group-Policy mit einem gespeicherten Passwort noch aktiv**, siehe hierzu auch 4.8 FIN-08: Intern: Verwendung von Passwort aus Gruppenrichtlinie . Wir probierten das **Passwort an dem Citrix-System** aus, dieser Versuch **schlug aber fehl**. Wir sahen auch, dass das System mittels LAPS verwaltet wurde und es dementsprechend unwahrscheinlich war, dass das Passwort auf diesem System funktionierte. Eine Analyse mit den zuvor gesammelten Informationen von SharpHound offenbarte, dass die GPO anscheinend die gesamte Organization Unit *sample-hardware* betraf. Diese umfasste über 10.000 Systeme. Wir **vermuteten, dass es noch einige Systeme geben könnte, auf denen das Passwort durch diese Gruppenrichtlinie gesetzt wurde, und notierten dies als Angriffsvektor für später.**

3.2.4 Dritte Angriffswelle (14.03.–21.03.2024)

Wir fanden eine **große Anzahl an veralteten Systemen im Active Directory**, deren Betriebssystem nicht mehr unterstützt wird („End of Life“). Um abgeschaltete Systeme auszuschließen, filterten wir zunächst nach Systemen, die sich in den letzten 7 Tagen angemeldet hatten.

Veraltete Systeme sind ein attraktives Ziel, da sie nicht über die neuesten Sicherheitsfeatures verfügen. Wir vermuten, dass viele dieser Systeme zur Produktion gehören. Deswegen prüften wir zunächst die netzwerktechnische Erreichbarkeit mittels Portscans. Wir prüften, ob Port 445/TCP erreichbar war. Hier stellten wir fest, dass **fast alle Systeme erreichbar** waren, siehe auch 4.7 FIN-07: Intern: Erreichbarkeit der OT-Systeme aus Citrix-Umgebung.

Wir versuchten, uns auf den Systemen mit dem in der **Group-Policy ermittelten Passwort anzumelden**, da diese Systeme nicht von LAPS verwaltet wurden. Dazu meldeten wir uns mittels RDP mit dem Benutzer *Sample* an und hatten **bei einigen Systemen Erfolg**. Diese befanden sich laut Active-Directory-Beschreibungen in London, Paris und Tokio. Beim Anmelden stellten wir fest, dass dort **andere Benutzer aktiv** waren. Aufgrund der Standorte und des Alters der Systeme vermuteten wir, dass es sich hierbei um OT-Systeme handelt. **Da wir Störungen vermeiden wollten, hielten wir Rücksprache mit unserem Ansprechpartner, wie wir an dieser Stelle weiter vorgehen sollen.**



Wir bekamen das System SAMPLE012 genannt, welches wir weiter untersuchen durften.

Da wir auf diesem System lokale Administratorrechte hatten, versuchten wir, detaillierte Informationen zum eingesetzten EDR CrowdStrike auszulesen. Dafür luden wir Debug-Dateien herunter, die CrowdStrike anlegte. Dies ist in 4.11 FIN-11: Intern: Einsatz alter CrowdStrike-Version beschrieben.

Wir fanden einen konfigurierten Ausschluss für die Überwachung durch CrowdStrike und bereiteten entsprechend einen Payload vor. Dieser Payload versuchte, die auf dem System zwischengespeicherten Anmeldeinformationen auszulesen. Diese Anmeldeinformationen befinden sich im Speicher des Prozesses „lsass.exe". Um auf diesen Prozess-Speicher zuzugreifen, wird ein sogenannter Handle benötigt. Einen solchen Handle zu erstellen, ist sehr auffällig. Deshalb nutzten wir das Tool *nanodump* und bereiteten die Ausführung so vor, dass ein bereits bestehender Handle zum lsass-Prozess geklont wurde. Das sollte die Erkennungschancen von CrowdStrike verringern.

Die **Payload testeten wir in unserem Labor** auf einem System gleicher Art, um unerwünschte Nebeneffekte zu vermeiden. Hier ließen sich die gespeicherten Anmeldeinformationen **erfolgreich** auslesen.

Die **Ausführung auf dem System SAMPLE012 schlug jedoch fehl**, da CrowdStrike das Verhalten erkannte und blockierte. Da wir erneut ein Sperren des Benutzeraccounts vermuteten, versuchten wir, durch den Task-Manager einen Dump des lsass-Prozesses zu generieren. Das ist eine offensichtliche bösartige Methode und wird in der Regel erkannt. Dieses Mal schlug das auf dem System ebenfalls vorhandene EDR SentinelOne Alarm und blockierte die Aktion.

Wir **meldeten uns daraufhin ab und versuchten, die Spuren zu verwischen**, um wir später einen erneuten Versuch auf einem anderen System starten zu können. Wie wir am nächsten Tag feststellten, hat das **SOC den Vorfall auch hier korrekt eingestuft** und gehandelt. **Unser zweiter Benutzeraccount war nun ebenfalls gesperrt. Damit war das Assessment beendet.**

4 Findings

Dieses Kapitel beschreibt identifizierte Schwachstellen oder sicherheitsrelevante Sachverhalte. Diese sind getrennt nach dem internen und externen Teil des Assessments. Die Findings wurden nach ihrem Risiko bewertet und sortiert. Details zur Bewertungsskala sind im Kapitel „Erläuterung Bewertungsskala“ im Anhang beschrieben.

4.1 FIN-01: Extern: Breached Credentials

Betroffen: Risiko: Hoch

- Verschiedene Benutzerkonten mit E-Mail-Adresse

4.1.1 Übersicht

Bei der initialen Informationssammlung konnten einige Zugangsdaten gefunden werden, die sich Dienst- oder Benutzerkonten des Kunden zuordnen ließen. Dabei handelt es sich um Zugangsdaten, die aus sogenannten Breaches stammen, also beispielsweise von Websites, deren Benutzerdatenbank angegriffen wurde. Die identifizierten Zugangsdaten konnten nicht für eine erfolgreiche Anmeldung genutzt werden. Vermutlich wurden die Passwörter zwischenzeitlich geändert oder die Prüfung auf solche speziellen Zugangsdaten erfolgt schon regelmäßig durch den Kunden. Für den Fall, dass der Kunde bereits selbst prüft, ist dieser Sachverhalt als nichtig anzusehen.

Mögliche Folgen einer erfolgreichen Ausnutzung 🔥🔥🔥🔥🔥

- Mit gültigen Zugangsdaten kann ein Angreifer sich Zugriff auf interne Systeme verschaffen, sollte die Anmeldung nicht durch einen zweiten Faktor geschützt sein

Beispiele für Voraussetzungen für eine Ausnutzung 🎲🎲🎲🎲🎲

- Zugangsdaten können einfach bezogen werden

- Zugangsdaten müssen gültig sein
- Ggf. sind betroffene Zugangsdaten auch für andere Anwendungen gültig, wenn Benutzer Passwörter wiederverwenden.
- Keine der vorgefundenen Zugangsdaten waren gültig

4.1.2 Empfehlung

- Regelmäßig auf neu veröffentlichte Zugangsdaten prüfen, z. B. mit Diensten wie haveibeenpwned.com, sofern dies nicht schon geschieht
- Mitarbeiter für die Verwendung unterschiedlicher Passwörter sensibilisieren
- Mitarbeiter sollten sich nicht mit ihren geschäftlichen Benutzerkonten bei privat genutzten Diensten registrieren

4.1.3 Technische Details

Für die Domänen *samplecompany.de* und *samplecompany.com* wurden veröffentlichte Zugangsdaten aus verschiedenen Quellen analysiert. Insgesamt konnten 200 Benutzerkonten inklusive Passwort identifiziert werden. Allerdings war es nicht möglich, sich mit den Zugangsdaten anzumelden.

Wie wir später herausfanden, wurde für den Remote-Zugang ein anderes Namensschema verwendet. Anstelle der Mail-Adresse wurde eine interne Namenskennung genutzt. Zu solchen Konten wurden keine Zugangsdaten in Breaches gefunden. Die Zugangsdaten wurden über DeHashed abgefragt, das die Zugangsdaten aus den verschiedenen Quellen zusammenträgt, darunter auch die größten Datenbanken („Breach Compilation“).

4.2 FIN-02: Extern: Metadaten in Dokumenten

Betroffen: Risiko: Mittel

- Verschiedene Office-Dokumente unter den identifizierten Domänen

4.2.1 Übersicht

In den Metadaten öffentlich verfügbarer Dokumente werden Informationen offenbart, die einem Angreifer in seinem weiteren Vorgehen helfen können. Konkret wurden hierbei vollständige Namen von Mitarbeitern, eingesetzte Software inklusive Versionsnummer sowie ein mögliches Namensschema für interne Benutzerkonten gefunden.

Mögliche Folgen einer erfolgreichen Ausnutzung 🔥🔥🔥🔥🔥

- Preisgabe von Informationen selbst ist keine Schwachstelle
- Schaden kann entstehen, wenn Informationen nützlich für Angriffe sind
- Beispiel 1: Informationen zu eingesetzter Software und Versionen helfen, gezielt nach öffentlich bekannten Schwachstellen in der Komponente zu suchen
- Beispiel 2: Beim Vorbereiten gezielter Social-Engineering-Angriffe sind technische Informationen hilfreich, um legitim klingende Aufhänger zu konzipieren

Beispiele für Voraussetzungen für eine Ausnutzung 🎲🎲🎲🎲🎲

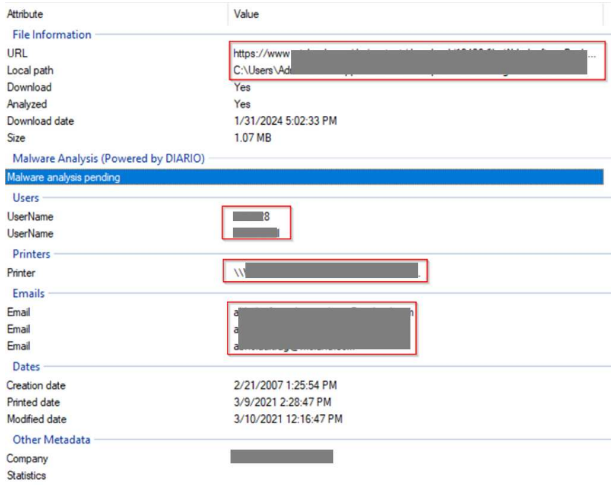
- Dokumente sind öffentlich verfügbar
- Einsehen und Sammeln der Informationen ist mit einfachen Mitteln möglich

4.2.2 Empfehlung

- Metadaten der Dokumente vor dem Veröffentlichen bereinigen

4.2.3 Technische Details

Mit dem Open-Source-Tool FOCA wurden mithilfe von Suchmaschinen wie Bing und Google öffentlich verfügbare Office-Dokumente und PDF-Dateien gesammelt, heruntergeladen und automatisiert ausgewertet. Die folgende Abbildung zeigt beispielhaft einige der gefundenen Informationen.



Außerdem vermuteten wir, dass das interne Namensschema dadurch offenbart wurde. Als Autor befanden sich Einträge mit dem Aufbau „sam123“ in Dokumenten, wobei sich nur die 3 Ziffern unterschieden. Unsere Vermutung, dass dies die internen Benutzerkonten im Active Directory sind, konnten wir im Verlauf des Red Teamings bestätigen.

4.3 FIN-03: Extern: Mail-Adressen verifizierbar

Betroffen: Risiko: Mittel

- Mailserver: mx1.samplecompany.com, mx2.samplecompany.com

4.3.1 Übersicht

Die betroffenen Mailserver erlaubten es, die Gültigkeit von Mail-Adressen zu verifizieren. Das war auch ohne das Senden von Mails möglich. Gefundene Mail-Adressen konnten so verifiziert und in weiteren Angriffen genutzt werden.

Mögliche Folgen einer erfolgreichen Ausnutzung 

- Gefundene Mail-Adressen können auf ihre Gültigkeit überprüft werden, um Hard Bounces zu vermeiden und damit die Reputation der Senderdomäne nicht zu gefährden
- Kombinationen aus beliebten Vor- und Nachnamen können ausprobiert werden, um gültige Mitarbeiter zu erraten
- Folgeangriffe wie beispielsweise Phishing mit ausschließlich gültigen Mail-Adressen sind unauffälliger

Beispiele für Voraussetzungen für eine Ausnutzung 

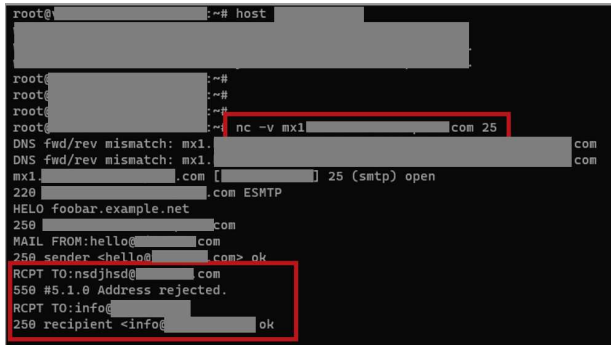
- Aufbau einer SMTP-Verbindung zum Mailserver
- Verifizieren von Mail-Adressen ließ sich leicht automatisieren
- Nach 20 Versuchen musste einige Minuten gewartet werden, bis erneut Mail-Adressen verifiziert werden konnten

4.3.2 Empfehlung

- Antwort der Mailserver sollte die Gültigkeit von Mail-Adressen nicht vor dem Absenden einer Mail offenbaren

4.3.3 Technische Details

Die beiden Mailserver *mx1.samplecompany.com* und *mx2.samplecompany.com* wurden auf gängigen Wegen untersucht, um Mail-Adressen zu verifizieren.



Dabei fiel auf, dass beide Mailserver durch den SMTP-Befehl *RCPT TO* preisgaben, ob der angegebene Empfänger existierte oder nicht. Wie in der Abbildung zu sehen, antwortete der Mailserver bei einer gültigen Mail-Adresse mit dem Status-Code „250 sender *mail*@samplecompany.com ok". Bei einer ungültigen Mail-Adresse wurde der Status-Code „550 #5.1.0 Address rejected" zurückgegeben.

Bei der Verifizierung fiel auf, dass der Mailserver stets nach ca. 20 Versuchen mit dem Status-Code „452 Too many recipients received this hour" antwortete. Das verhinderte kurzzeitig das Verifizieren weiterer Mail-Adressen. Nach wenigen Minuten war diese Blockade jedoch wieder aufgehoben.

4.4 FIN-04: Extern: Einsatz veralteter Software

Betroffen: Risiko: Mittel

- https://oldapp.samplecompany.com
- https://evenolderapp.samplecompany.com

4.4.1 Übersicht

Mehrere Webanwendungen wurden mit dem JavaFy-Framework entwickelt, das in der eingesetzten Version veraltet und im End-of-Life-Status ist. Keine der öffentlich bekannten Schwachstellen konnte ausgenutzt werden.

Mögliche Folgen einer erfolgreichen Ausnutzung 🔥🔥🔥🔥🔥

- Bekannte Schwachstellen ermöglichen Angriffe auf andere Nutzer
- Neu entdeckte Schwachstellen werden vom Hersteller nicht behoben, außer entsprechender Support wurde eingekauft

Beispiele für Voraussetzungen für eine Ausnutzung 🎲🎲🎲🎲🎲

- Ausnutzung hängt von der Schwachstelle selbst ab
- Einige der öffentlich bekannten Schwachstellen sind unauthentifiziert ausnutzbar, benötigen jedoch eine Interaktion mit einem Benutzer

4.4.2 Empfehlung

- Aktuelle Version des JavaFy-Frameworks verwenden

4.4.3 Technische Details

Bei der Untersuchung der Webanwendungen wurde festgestellt, dass mehrere Anwendungen mit dem JavaFy-Framework entwickelt wurden. Hierbei fiel auf, dass eine veraltete Version eingesetzt wird.

Die Infos wurden im Quellcode preisgegeben. Die eingesetzten Versionen enthalten mehrere Schwachstellen, deren Ausnutzung die Verfügbarkeit des Diensts beeinträchtigen können. Da unser Ziel das Eindringen ins interne Netzwerk war, unternahmen wir keine Versuche, die Schwachstellen auszunutzen.

Betroffen sind mindestens folgende Anwendungen:

- https://oldapp.samplecompany.com
- https://evenolderapp.samplecompany.com

4.5 FIN-05: Extern: Offenlegung von internen Hostnamen

Betroffen: Risiko: Mittel

- Verschiedene interne und externe Systeme, siehe Beschreibung

4.5.1 Übersicht

An verschiedenen über das Internet zugänglichen Stellen werden Informationen über interne Systeme offenbart. Die Informationen können für zielgerichtetere Angriffe nützlich sein.

Mögliche Folgen einer erfolgreichen Ausnutzung

- Die Informationen können für weitere Angriffe nützlich sein, beispielsweise bei Phishing

Beispiele für Voraussetzungen für eine Ausnutzung

- Die Informationen sind im Internet frei verfügbar

4.5.2 Empfehlung

- Für interne Systeme möglichste keine öffentlichen TLS-Zertifikate anfragen, sondern eine interne CA nutzen
- Detaillierte Fehlermeldungen vermeiden
  - Stattdessen kann ein generischer Fehlercode generiert werden, der sich serverseitig den Fehlerdetails zuordnen lässt

4.5.3 Technische Details

In den nachfolgenden Abschnitten ist erläutert, an welchen Stellen technische Informationen preisgegeben werden.

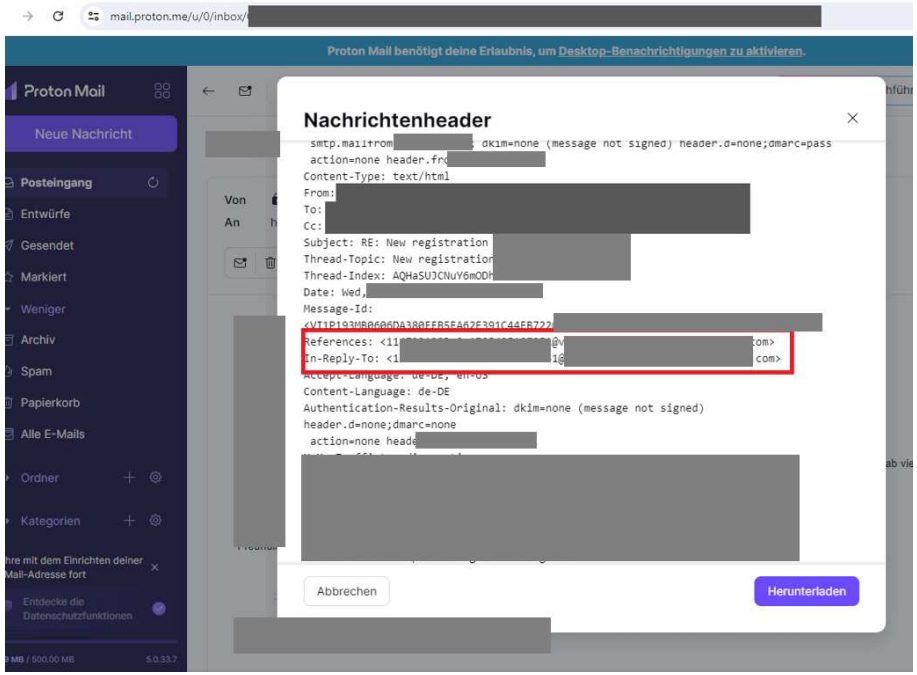
Alle aufgeführten Informationen deuten darauf hin, dass intern die Domäne *samplecompany.com* genutzt wird.

Informationspreisgabe durch TLS-Zertifikate

Bei der Suche nach Domänen konnte die Subdomäne *intern.samplecompany.com* identifiziert werden. Eine Suche nach möglichen angefragten TLS-Zertifikaten durch den Dienst *crt.sh* offenbarte viele weitere interne Domänen. Die neuesten Einträge waren wenige Tage alt. Deshalb gehen wir davon aus, dass die Namen zu existierenden internen Systemen gehören.

Informationspreisgabe durch Mail-Header

Nach einer Benutzerregistrierung bei der Webanwendung *sampleapp* wurde per Mail nach dem Grund der Registrierung gefragt. In dieser Mail wurden die Mail-Header analysiert. Diese offenbaren an mehreren Stellen interne Hostnamen.



Informationspreisgabe durch Webseite

Beim Aufruf einer nichtexistierenden URL wird in der Fehlermeldung der interne Hostname offenbart.

4.6 FIN-06: Extern: Blinder Fleck: Monitoring der Webanwendungen

Betroffen: Risiko: Mittel

- Alle öffentlich erreichbaren Webanwendungen

4.6.1 Übersicht

Auf verschiedene extern erreichbare Webanwendungen wurden Angriffe durchgeführt. Soweit von außen zu beurteilen, schienen die durchgeführten Angriffe nicht von einem Schutzsystem abgewehrt oder entdeckt worden zu sein.

Wir gehen davon aus, dass es sich hierbei um einen blinden Fleck handelt. Wir vermuten, dass der Sachverhalt im Monitoring des Blue Teams entweder nicht erfasst wurde oder nicht aufgefallen ist.

Mögliche Folgen einer erfolgreichen Ausnutzung

- Angreifer kann unbemerkt auch schwierig zu entdeckende Schwachstellen finden und ausnutzen
- Je nach Art der Schwachstelle kann ein Angreifer Zugriff in das interne Netzwerk erlangen oder wertvolle Informationen auf kompromittierten Systemen sammeln

Beispiele für Voraussetzungen für eine Ausnutzung

- Schwachstellen müssen vorhanden sein
- Webserver waren öffentlich erreichbar, keine weiteren Schritte nötig

4.6.2 Empfehlung

- Öffentlich erreichbare Webserver in das Monitoring einbinden
- Zusätzliche Schutzmaßnahmen wie eine Web Application Firewall (WAF) einführen, um Angreifer erkennen und blockieren zu können

4.6.3 Technische Details

Bei der Bestimmung der Angriffsoberfläche von außen wurden mehrere Webanwendungen identifiziert. Diese wurden zunächst unauffällig manuell untersucht. Im weiteren Verlauf wurden auch Schwachstellenscanner eingesetzt, die eine große Anzahl von Anfragen senden und die Anwendungen auf viele unterschiedliche Schwachstellen überprüfen.

Bei der Untersuchung der Anwendungen fiel auf, dass keine Gegenreaktion erfolgte und die Anwendungen auch automatisiert ungehindert untersucht werden konnten. So wurde beispielsweise ein Directory Brute Force ausgeführt, bei dem mögliche URLs geraten werden, und in verschiedene Eingabefelder wurden Payloads eingefügt, wie etwa zum Entdecken einer SQL-Injection. Dadurch erzeugten wir mehrere Anfragen pro Sekunde über mehrere Minuten. Die untersuchten Server antworteten zuverlässig und die Anfragen wurden offenbar ungehindert ausgeführt. Außerdem wurden auch nach ausführlicher Enumeration unsere IP-Adressen nicht blockiert.

Das deutet darauf hin, dass die Anwendungen kein vorgeschaltetes Schutzsystem haben und entsprechend nicht im Monitoring eingebunden waren.

Mögliche interessante Ansatzpunkte könnten die Anwendungen sein, die veraltete Software einsetzen, siehe dazu auch 4.4 FIN-04: Extern: Einsatz veralteter Software.

4.7 FIN-07: Intern: Erreichbarkeit der OT-Systeme aus Citrix-Umgebung

Betroffen: Risiko: Hoch

- Interne Netzwerke der Sample Company

4.7.1 Übersicht

Vom Citrix-Arbeitssystem konnten mutmaßliche OT-Systeme erreicht werden. Wir gehen davon aus, dass das Netzwerk unzureichend segmentiert ist bzw. keine Trennung zwischen verschiedenen Segmenten erzwungen wird.

Mögliche Folgen einer erfolgreichen Ausnutzung

- Angreifer kompromittieren OT-Systeme und stören Teile der Produktion oder erweitern ihre Rechte in der Domäne
- Angreifer können Schwachstellen leichter finden und ausnutzen, da OT-Systeme üblicherweise anfälliger sind

Beispiele für Voraussetzungen für eine Ausnutzung

- Angreifer benötigt Zugriff auf ein Citrix-System
  - Hierzu muss er beispielsweise ein gültiges Benutzerkonto kompromittieren, etwa durch Phishing

4.7.2 Empfehlung

- Netzwerksegmentierung einführen und erzwingen, um insbesondere kritische Netzwerke abzuschotten

4.7.3 Technische Details

Im Zuge unserer Schwachstellensuche kundschafteten wir zunächst die Systeme aus, auf denen eine nicht mehr unterstütze Windows-Betriebssystemversion installiert war. Diese Systeme sind oft anfälliger für Schwachstellen, da diverse Sicherheitsfeatures von neueren Versionen fehlen.

Über ein gefundenes Passwort von lokalen Administratorbenutzern, wie in 4.8 FIN-08: Intern: Verwendung von Passwort aus Gruppenrichtlinie beschrieben, ermittelten wir zunächst mittels SMB die netzwerktechnische Erreichbarkeit von Systemen. Hierbei stellten wir kaum Einschränkungen fest. Fast alle der geprüften Systeme waren vom Citrix-Arbeitssystem aus erreichbar:

- Sample123
- Sample134
- Sample133
- Sample132
- Sample136
- Sample131
- Sample130

Außerdem versuchten wir, uns mittels RDP dort anzumelden, und stellten auch hier kaum Einschränkungen fest.

Wir vermuten, dass es sich bei einigen der Zielsysteme um OT-Systeme handelt, da dort beispielsweise gemeinsam genutzte Benutzer wie *samplesshareduser* (producing_terminal) angemeldet waren.

4.8 FIN-08: Intern: Verwendung von Passwort aus Gruppenrichtlinie

Betroffen: Risiko: Hoch

- Gruppenrichtlinie „SAMPLE_GPO_LOCAL_ADMIN“ der Domäne samplecompany.local

4.8.1 Übersicht

Mit einer Gruppenrichtlinie wurde im Active Directory das Passwort für zwei lokale Administratorbenutzer auf mehreren Systemen festgelegt. Das Passwort konnte ausgelesen werden. Die Benutzer waren auf mehreren Systemen aktiv. Das betraf insbesondere OT-Systeme.

Mögliche Folgen einer erfolgreichen Ausnutzung

- Lokale Administratorrechte auf verschiedenen Systemen, insbesondere kritischen OT-Systemen
- Lateral Movement: Übernahme von anderen Systemen mit dort erhöhten Rechten
- Auslesen von zwischengespeicherten Sitzungsinformationen auf den Systemen, auf die die Adminbenutzer Zugriff haben

Beispiele für Voraussetzungen für eine Ausnutzung

- Jeder Domänenbenutzer kann das Passwort der Gruppenrichtlinie mit entsprechenden Tools auslesen

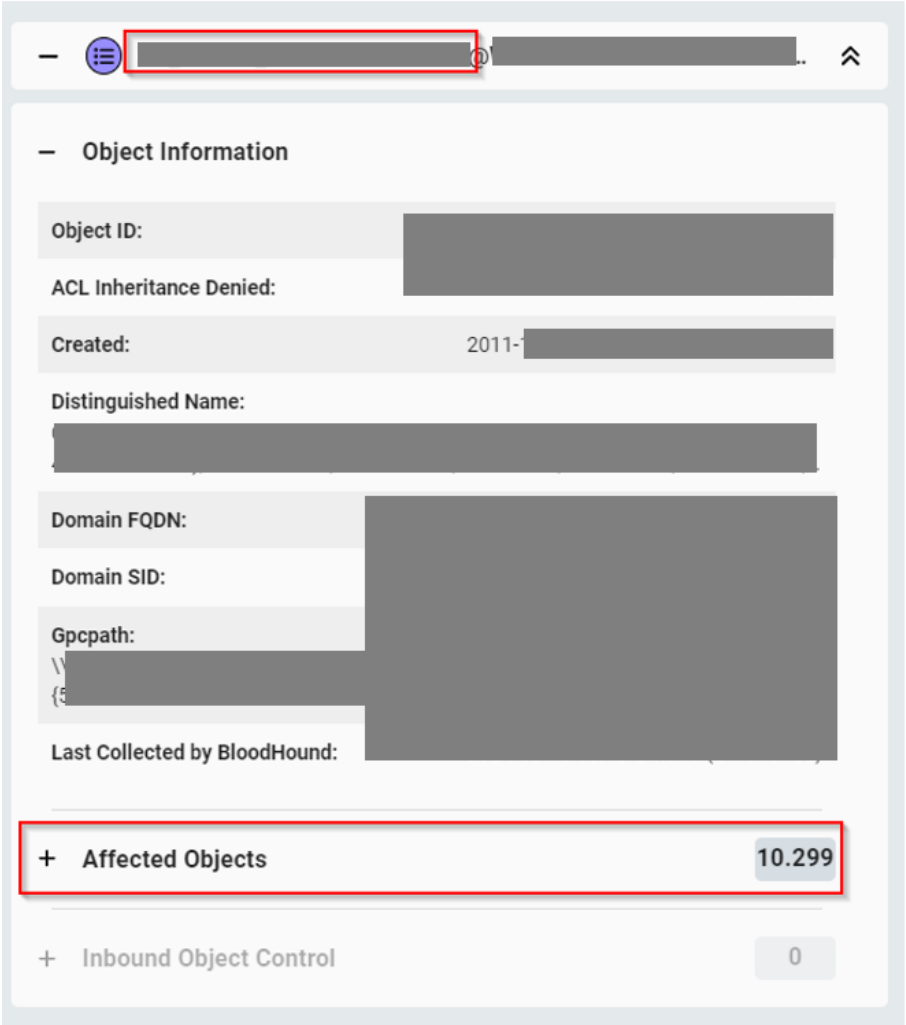
4.8.2 Empfehlung

- Gruppenrichtlinie deaktivieren oder löschen
- Prüfen, auf welchen Systemen die beiden betroffenen Benutzer noch das Passwort verwenden, und dieses ändern

4.8.3 Technische Details

Über eine Gruppenrichtlinie wurden zwei lokale Administratorbenutzer (*Administrator* und *_Client*) auf mehrere Systeme ausgerollt. Das Passwort befand sich dabei in verschlüsselter Form in der Gruppenrichtlinie selbst und konnte von jedem Benutzer eingesehen werden. Microsoft veröffentlichte den Schlüssel für diese konkrete Verschlüsselung, wodurch es einfach möglich war, an das Klartextpasswort zu gelangen.

Die Gruppenrichtlinie wurde 2011 erstellt und betrifft die ganze Organization Unit „Hardware" der Domäne *samplecompany.local*. Dies umfasst ca. 10.200 Systeme.



4.9 FIN-09: Intern: Lücken in AppLocker-Konfiguration

Betroffen: Risiko: Hoch

- Citrix Clients, z. B. SAMPLE012345

4.9.1 Übersicht

Auf den Citrix-Clients war AppLocker konfiguriert, was die Ausführung unerwünschter Programme unterbinden soll. Allerdings fanden wir mehrere Wege, das zu umgehen und so eigene Programme auszuführen.

Mögliche Folgen einer erfolgreichen Ausnutzung

- Angreifer können Schadsoftware ausführen, beispielsweise zum Aufbauen eines Command-and-Control-Channels

Beispiele für Voraussetzungen für eine Ausnutzung

- AppLocker-Regeln müssen bekannt sein
  - Können in der Regel von allen Benutzern über einen PowerShell-Befehl ausgelesen werden
- Nach Erkennen einer Lücke benötigt ein Angreifer die Möglichkeit, Dateien an den entsprechenden Orten abzulegen und auszuführen
  - Setzt üblicherweise interaktiven Remotezugriff, wie beispielsweise über Citrix, voraus

4.9.2 Empfehlung

- Windows Defender Application Control (WDAC) anstelle von AppLocker verwenden
- Falls AppLocker in Zukunft beibehalten wird, die Konfiguration in Bezug auf folgende Punkte verbessern:
  - Alle Platzhalter in erlaubten Pfaden überprüfen: Benutzer sollten nicht berechtigt sein, Dateien zu erstellen oder zu ändern, die von einer Platzhalterregel betroffen sind

- Aus verwaltungstechnischen Gründen Regeln nicht einzelnen Benutzern zuweisen und Duplikate von Regeln vermeiden
- Ausführung sogenannter LOLBAS-Dateien so weit wie möglich einschränken
- Erforderlichn Ausnahmen regelmäßig evaluieren

4.9.3 Technische Details

Die AppLocker-Konfiguration wurde auf dem Citrix-System SAMPLE012345 mit dem PowerShell-Befehl `Get-AppLockerPolicy -Effective -Xml` extrahiert und analysiert.

Dabei fiel auf, dass einige Ausschlüsse missbraucht werden können, da sie Platzhalter enthielten und unser normal berechtigter Benutzer Dateien in den betroffenen Pfaden anlegen konnte. Die AppLocker-Policy umfasst mehr als 700 Regeln (276 davon mit Wildcards) und konnte daher nicht vollständig analysiert werden. Nachfolgend wird ein Beispiel erläutert.

In der folgenden Regel werden von allen Benutzern (C:\Users*) alle Dateien und Unterordner zur Ausführung erlaubt, wenn sich diese im Benutzerordner unter AppData\Roaming\Citrix\SelfService\ befinden. Standardmäßig besitzen die Benutzer Schreibrechte in ihren eigenen Unterverzeichnissen, also auch im AppData-Ordner.

Während des Assessments wurde das zum Ausführen eigener Tools genutzt, um einen Command-and-Control-Channel aufzubauen.

Weitere betroffene Regeln:

- C:\USERS*\APPDATA\LOCAL\TEMP\TEAMVIEWER*.EXE
- C:\USERS*\APPDATA\LOCAL\TEMP\TEAMS*.EXE
- C:\SAMPLE*

4.10 FIN-10: Intern: Sensible Daten auf Netzwerk-Share

Betroffen: Risiko: Hoch

- Netzwerk-Shares der Domäne samplecompany.com, siehe technische Details

4.10.1 Übersicht

Benutzer des Active Directory (AD) können auf sensible Daten auf Netzwerk-Shares zugreifen. Diese enthalten neben personenbezogenen Daten auch Zugangsdaten für hochprivilegierte Benutzer und können für weitere Angriffe nützlich sein.

Mögliche Folgen einer erfolgreichen Ausnutzung

- Mehrere Passwörter wurden ermittelt, sowohl von normal privilegierten als auch von hoch privilegierten Benutzerkonten
- Die Log-Dateien enthielten sehr viele Informationen darüber, wie die Skripte arbeiteten, und ermöglichten gezieltere Angriffe

Beispiele für Voraussetzungen für eine Ausnutzung

- AD-Benutzer wird benötigt
- Die interessanten Informationen müssen unter der großen Anzahl an Dateien gefunden werden
- Tools zum Automatisieren des Prozesses sind vorhanden

4.10.2 Empfehlung

- Offengelegte Zugangsdaten ändern
- Evaluieren, ob gespeicherte Informationen noch benötigt werden
- Netzwerk-Shares basierend auf der angehängten Liste intern prüfen

4.10.3 Technische Details

Auf Netzwerk-Shares sind häufig Informationen zu finden, die für Angriffe nützlich sind, wie beispielsweise Passwörter in Skripten oder Dateien. Deshalb untersuchten wir die erreichbaren Netzwerk-Shares dahingehend. Im Folgenden sind die ermittelten Freigaben aufgelistet, die sensible Informationen enthalten.

\\SHARE1\

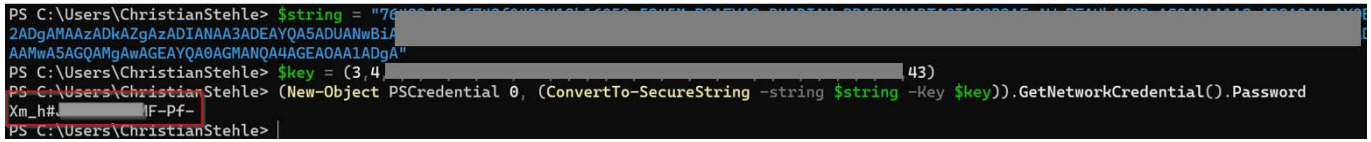
Für die bereitgestellten Benutzer war das Netzwerklaufwerk \SHARE1\ eingebunden. Diese Freigabe wurde offenbar intern genutzt, um Dateien auszutauschen. Es gab täglich neu abgelegte Dateien von anderen Domänenbenutzern. Teilweise waren das Backup-Dateien, die sensible Informationen enthalten könnten. Ein Angreifer mit mehr Zeit könnte dieses Laufwerk täglich beobachten und auf das Ablegen von sensiblen Informationen warten.

\\samplecompany.intern\netlogon

Hier wurden in mehreren Skripten Zugangsdaten im Klartext gefunden. Konkret handelte es sich um Skripte, die automatisch Netzlaufwerke einbinden. Die betroffenen Dateien waren:

- FILE1.cmd
- FILE2.cmd
- FILE3.cmd
- FILE4.cmd
- FILE5.cmd

Im Skript *FILE1.cmd* befindet sich ein Verweis auf ein anderes Skript (sample_office.ps1), dem ein Key als Parameter übergeben wurde. Das Skript *sample_office.ps1* lud einen SecureString von einer Text-Datei und entschlüsselte diesen mit dem Key. Das konnten wir nutzen, um an das Passwort des Benutzers *_sample* zu gelangen.



Wir konnten den SecureString auslesen, da dieser auf dem Dateisystem des Citrix-Systems vorhanden war. Dadurch konnten wir das Klartextpasswort des Benutzers entschlüsseln.

Danach stellten wir fest, dass das Passwort nicht (mehr) funktionierte.

4.11 FIN-11: Intern: Einsatz alter CrowdStrike-Version

Betroffen: Risiko: Mittel

- Eingesetzte CrowdStrike EDR Lösung

4.11.1 Übersicht

Die eingesetzte Version des CrowdStrike EDR erlaubte das Auslesen von Ausschlüssen sowie des Passworthashs des Supervisor-Benutzers. Das Passwort konnte im Projektzeitraum nicht ermittelt werden.

Mögliche Folgen einer erfolgreichen Ausnutzung

- Umgehung des EDRs durch unbemerkte Ausführung von Schadsoftware
- Deinstallieren des EDRs auf einem System

Beispiele für Voraussetzungen für eine Ausnutzung

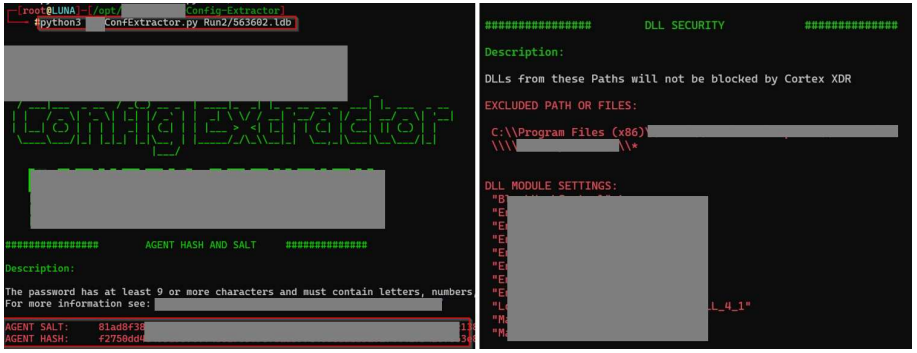
- Lokale administrative Berechtigungen sind erforderlich, um die Ausschlüsse auslesen und den Passworthash auslesen zu können

4.11.2 Empfehlung

- Auf CrowdStrike-Agent ab 6.0 aktualisieren/upgraden
- Ausschlüsse prüfen und entfernen, falls nicht mehr benötigt

4.11.3 Technische Details

Als lokaler Administrator konnten Debug-Dateien der eingesetzten CrowdStrike-EDR-Lösung ausgelesen werden. Diese Dateien enthalten einige Informationen über Interna des EDRs sowie den Passworthash des Supervisor-Benutzers. Mit dem Tool *CrowdStrike-XDR-Config-Extractor* wurden diese Debug-Dateien analysiert. Bei den Ausschlüssen sollte geprüft werden, ob sie noch relevant sind.



4.12 FIN-12: Intern: Unbemerkter Aufbau eines Command-and-Control-Channels

Betroffen: Risiko: Mittel

- Monitoring des Netzwerkverkehrs ins Internet

4.12.1 Übersicht

Der Aufbau eines Command-and-Control-Channels (C2-Channel) war unerkannt möglich. Über den Kanal wurden Befehle aus der Ferne ausgeführt und Daten wie Befehlsausgaben und Dateiinhalte an den Server des Red Teams übertragen.

Mögliche Folgen einer erfolgreichen Ausnutzung 🔥🔥🔥🔥🔥

- Angreifer kann Befehle ausführen und Daten entwenden

Beispiele für Voraussetzungen für eine Ausnutzung 🎲🎲🎲🎲🎲

- Ausführung eines Programms ist nötig, hierzu müssen Schutzmechanismen wie AppLocker und Antiviren-Software umgangen werden

4.12.2 Empfehlung

- Erkennung für gängige C2-Channel implementieren
- Implementierte Lösung auf ihre Wirksamkeit überprüfen

4.12.3 Technische Details

Auf dem System wurde ein Beacon gestartet, der eine HTTPS-Verbindung zu einem von uns kontrollieren Server aufbaute und mit ihm kommunizierte. Dabei empfing das Beacon Befehle, die wir absetzten, führt diese lokal aus und lieferte das Ergebnis zurück. Hierbei spricht man von einem Command-and-Control-Channel.

Als Domäne wurde das Azure CDN verwendet, das in vielen Proxy-Lösungen als vertrauenswürdig klassifiziert wird und somit auch in diesem Fall erreichbar war. Der Beacon sendete alle 30 Sekunden mit 30 % Jitter einen HTTPS-POST-Request an die Adresse *https://samplecompany.azureedge.net/msupdate/Setup_Install001.cab* und nutzte zur Datenübertragung einen base64-kodierten Payload.

5 Projektrahmen

5.1 Involvierte Personen

Name	Rolle	Mail-Adresse
Christian Stehle	Projektleitung & Durchführung	hallo@mind-bytes.de
Nina Wagner	Durchführung	hallo@mind-bytes.de
Simon Holl	Durchführung	hallo@mind-bytes.de
Anja Neudert	Review	hallo@mind-bytes.de
Max Mustermann	CEO	max.mustermann@samplecompany.com

5.1 Testzeitraum

01.01.24 - 21.03.24

5.2 Durchführungskonzept

Ein Red Team hat die Aufgabe, die Projektziele wie beispielsweise das Eindringen in die Infrastruktur eines Unternehmens unbemerkt zu erreichen. Dabei werden Techniken angewendet, die denen echter Angreifer entsprechen, um sowohl die technischen als auch die organisatorischen Abwehrfähigkeiten eines Unternehmens zu testen.

Insbesondere werden unter realen Bedingungen die Effektivität der Abwehrmechanismen des Blue Teams bzw. des Security Operations Centers (SOC) sowie interne Meldeketten überprüft.

Bei der Projektdurchführung arbeiten das White Team – ein Kreis eingeweihter Personen im angegriffenen Unternehmen – und das Red Team eng zusammen. Das Red Team kommuniziert den Projektfortschritt, und gemeinsam werden Entscheidungen zum weiteren Vorgehen getroffen. Das White Team teilt dem Red Team eventuelle Beobachtungen des Blue Teams mit.

5.3 Rules of Engagement

Bevor das Projekt beginnt, definieren der Auftraggeber und das Red Team in den sogenannten *Rules of Engagement* (RoE) die Ziele und Rahmenbedingungen für die Durchführung des Projekts. Die folgenden Abschnitte bieten einen Überblick über die potenziellen Vorgehensweisen in einem Red-Teaming-Projekt sowie die festgelegten Regeln für diese spezifische Durchführung.

5.3.1 Startpunkt des Red Teams

Mit welcher Ausgangsposition beginnt das Red Team seine Durchführung? Von dieser Startposition aus sollen Angriffspfade gefunden werden. Sie simuliert die des Angreifers.
Im Rahmen eines Assume-Breach-Ansatzes könnte zum Beispiel festgelegt werden, dass das Red Team mit dem Zugriff auf einen Laptop startet, wie ihn Mitarbeiter standardmäßig erhalten. Diese Position würde ein Angreifer beispielsweise nach einem erfolgreichen Phishing-Angriff haben.

In diesem Projekt: Sowohl außen als auch innen. Im ersten Teil des Projekts wurde versucht, von außen einzudringen. Im zweiten Teil des Projekts wurde von innen gestartet und versucht, die interne Umgebung zu kompromittieren.

5.3.2 In-Scope Controls

Welche Bestandteile des Unternehmens darf das Red Team angreifen?

In diesem Projekt: Externe IT-Infrastruktur, interne IT-Infrastruktur, Mitarbeitende

5.3.3 Out-of-Scope Controls

Welche Bestandteile des Unternehmens sind aus diesem Projekt explizit ausgeschlossen?

In diesem Projekt: Gebäude

5.3.4 Erlaubte Tactics, Techniques & Procedures (TTPs)

Welche Aktionen sind bei der Projektdurchführung erlaubt?

In diesem Projekt:

- Enumerieren der Angriffsoberfläche des Unternehmens über öffentlich verfügbare Informationen
- Ausnutzen technischer Schwachstellen, sofern zuvor kein Verdacht besteht, die Verfügbarkeit der Systeme zu beeinträchtigen
- Kontaktieren von Mitarbeitenden in Phishing-Kampagnen

5.3.5 Ausgeschlossene Tactics, Techniques & Procedures (TTPs)

Welche Aktionen wurden für die Projektdurchführung explizit ausgeschlossen?

In diesem Projekt:

- Kontaktieren von Mitarbeitenden in ihrem privaten Umfeld
- Absichtliches Durchführen destruktiver Aktionen

5.4 Durchführung – die Phasen eines Red Teamings

Abhängig von den festgelegten Projektregeln wählt das Red Team passende Angriffsoptionen und -techniken aus. Im Allgemeinen lässt sich ein Red-Teaming-Projekt in mehrere Phasen unterteilen, die nachfolgend erläutert sind. Aufgrund der Individualität jedes Projekts sind Abweichungen zum dargestellten Ablauf möglich.

Beispielsweise können Phasen übersprungen werden, wenn Projekte in einer fortgeschrittenen Phase unter der Annahme starten, ein Angreifer hätte diesen Ausgangspunkt erreicht. Ebenso kann flexibel festgelegt werden, bei Eintreten welcher Bedingung das Projekt beendet ist.

Recon (Informationssammlung)

In dieser Phase werden aus Quellen wie Unternehmenswebseiten, Social-Media-Profilen und Webseiten mit kompromittierten Zugangsdaten Informationen über das Unternehmen gesammelt. Dieser Ansatz wird als *Open Source Intelligence* (OSINT) bezeichnet. Zusätzlich werden durch Auswertung technischer Quellen die IP-Adressen und Domänennamen des Unternehmens sowie erreichbare Dienste ermittelt.

Initial Compromise (Zugang zum internen Netzwerk erlangen)

In dieser Phase werden die gesammelten Informationen analysiert, um Schwachstellen zu finden und auszunutzen. Oft werden Phishing-Angriffe eingesetzt, um Mitarbeitende dazu zu bringen, ihre Zugangsdaten preiszugeben oder Dateien auszuführen, mit denen das Red Team Zugriff auf das interne Netzwerk des Unternehmens erlangt.

Establish Persistence (Persistenz sicherstellen)

Nachdem sich das Red Team in der vorherigen Phase Zugang zum internen Netzwerk verschafft hat, wird in dieser Phase die Persistenz sichergestellt. Zum Beispiel wird sichergestellt, dass im Fall einer Unterbrechung der Verbindung zu einem kompromittierten Server die Verbindung ohne erneutes Ausnutzen der Schwachstelle wiederhergestellt werden kann. Das wird in der Regel mit einem Command-and-Control-Framework erreicht.

Escalate Privileges (Rechte erweitern)

In dieser Phase versucht das Red Team, administrativen Zugriff auf ein System zu erlangen. Mit diesen neu erlangten Rechten werden Informationen über die Umgebung gesammelt, um die nächsten Angriffsmöglichkeiten zu ermitteln. Im Zuge des lateralen Bewegens werden benachbarte Systeme oder andere Benutzer angegriffen, wodurch sich das Red Team schrittweise innerhalb der Infrastruktur weiter fortbewegt.

Exfiltrate and Complete Mission (Daten exfiltrieren und Projekt beenden)

Abhängig von den Projektzielen werden Daten heruntergeladen oder der Zugriff auf ein spezifisches System nachgewiesen, wie beispielsweise einen Backup-Server.

6 Anhang

6.1 Erläuterung Bewertungsskala

Die Findings werden mit einem risikobasierten Ansatz nach unserer Einschätzung bewertet. Der Fokus liegt dabei auf dem (potenziellen) Schaden und der Wahrscheinlichkeit.

- Der Schaden beschreibt, welche Folgen eine erfolgreiche Ausnutzung nach sich ziehen kann.
- Die Wahrscheinlichkeit beschreibt, wie einfach eine Schwachstelle ausnutzbar ist.

In die resultierende Risikobewertung fließen Schaden, Wahrscheinlichkeit und die Wichtigkeit der betroffenen Komponenten ein. Die Werte stellen eine intuitive Einschätzung durch uns dar.

Die folgenden Abstufungen werden verwendet:

- Info
- Gering
- Mittel
- Hoch
- Kritisch

6.2 Glossar

Begriff	Beschreibung
Beacon	Ein Programm, das das Red Team einsetzt, um in kleinen regelmäßigen Kommunikationseinheiten mit einem externen Steuerungsserver (Teamserver) zu kommunizieren. Dabei werden Anweisungen oder Daten zwischen dem infizierten System (Opfer) und dem Red Team ausgetauscht. Insgesamt ermöglicht das Beaconing eine effiziente und unauffällige Steuerung von infizierten Systemen über eine sichere und verschleierte Kommunikationsverbindung.
Blue Team	Das Verteidigungsteam des Unternehmens, das für die Erkennung und Abwehr von Cyberangriffen verantwortlich ist. Das ist üblicherweise das Security Operations Center (SOC).
Critical Functions (CF)	Die Kernfunktionen des Unternehmens, deren Schutz höchste Priorität hat. Der Begriff stammt ursprünglich aus dem TIBER-Framework. Die CF werden in der Projektvorbereitung zwischen dem Red Team und dem Auftraggeber bestimmt und sollten Zielobjekte für das Red Team darstellen.
Command-and-Control (C2)	Software, die vom Red Team zum Verwalten von Verbindungen zu kompromittierten Systemen und Ausführen von Befehlen über Beacons genutzt wird.
Foothold	Erster Zugriff, der auf eine fremde Infrastruktur oder ein fremdes System durch das Red Team erlangt wird.
In-Scope Control	Bestandteile des Unternehmens, die das Red Team in die Durchführung involvieren darf. Dazu zählen Gebäude, Mitarbeitende und IT-Infrastruktur.
Out-of-Scope Control	Bestandteile des Unternehmens, die das Red Team explizit nicht in die Durchführung involvieren darf.
Indicators of Compromise (IoC)	Dienen als forensische Nachweise für ein mögliches Eindringen in ein System oder Netzwerk. Anhand dieser Artefakte lassen sich Einbruchsversuche oder andere bösartige Aktivitäten erkennen.

Begriff	Beschreibung
Initial Compromise	Phase in der Durchführung eines Red Teamings. Hierbei erlangt das Red Team erstmalig Zugriff in die interne Unternehmensumgebung. Siehe 5.4 Durchführung – die Phasen eines Red Teamings
Initial Recon/ Information Gathering	Phase in der Durchführung eines Red Teamings, siehe 5.4 Durchführung – die Phasen eines Red Teamings
Lateral Movement	Teil einer Phase in der Durchführung eines Red Teamings, siehe 5.4 Durchführung – die Phasen eines Red Teamings
Open Source Intelligence (OSINT)	Teil einer Phase in der Durchführung eines Red Teamings, siehe 5.4 Durchführung – die Phasen eines Red Teamings
Persistence	Phase in der Durchführung eines Red Teamings, siehe 5.4 Durchführung – die Phasen eines Red Teamings
Privilege Escalation	Phase in der Durchführung eines Red Teamings, siehe 5.4 Durchführung – die Phasen eines Red Teamings
Red Team	Ein unabhängiges Team, das echte Angriffe auf das Unternehmen simuliert.
Rules of Engagement	Festgelegte Rahmenbedingungen und Regeln, die zwischen Auftraggeber und Red Team zur Durchführung des Red Teamings festgelegt werden.
Tactics, Techniques & Procedures (TTP)	Methoden, die von Angreifern genutzt werden.

Begriff	Beschreibung
White Team / White Cell	Kreis von in das Projekt eingeweihten Personen des Auftraggebers. Sie sind Ansprechpartner des Red Teams und insbesondere nicht Teil des Blue Teams.

6.3 Gesammelte Informationen

Die gesammelten Informationen des ersten Projektteils befinden sich in der separaten Datei *RedTeamDatenbasis.xlsx* in den verschiedenen Arbeitsmappen:

- Domains – die identifizierten Domänen
- IP-Adressen – die identifizierten IP-Adressen (IPv4 und IPv6), wo sie sich befinden und an welchem Standort sie sind
- Mitarbeiter – identifizierte Mitarbeiter und welche davon Ziel der Phishing-Kampagnen waren
- Websites – identifizierte Webanwendungen
- Dienste – identifizierte Dienste

6.4 Red Team Activity Log

Das Protokoll der durchgeführten Aktionen befindet sich in der separaten Datei *RedTeamDatenbasis.xlsx* in der Arbeitsmappe „RedTeamActivityLog". Dokumentiert sind alle durchgeführten Aktionen mit Zeitpunkt, Quell- und Zielsystem und ggf. verwendeten Benutzern.

In der Arbeitsmappe „IoCs" befinden sich die Indicators of Compromise für das Assessment.

7 Disclaimer

Dieses Projekt wurde durchgeführt, um die Sicherheit der im Fokus liegenden Komponenten zu bewerten und Schwachstellen aufzudecken.

1. Bei diesem Test handelt es sich um eine Momentaufnahme und keine fortlaufende Sicherheitsüberwachung. Die Sicherheitslage kann sich im Laufe der Zeit ändern, beispielsweise durch Veränderungen an den Komponenten, preisgegebenen Informationen, neue Angriffstechniken oder Schwachstellen.
2. Das Projekt wurde innerhalb eines begrenzten Zeitrahmens durchgeführt. Dies kann dazu führen, dass nicht alle potenziellen Schwachstellen und preisgegebenen Informationen identifiziert wurden.
3. Auch wenn das Projekt mit großer Sorgfalt durchgeführt wurde, sind False-Positives nicht auszuschließen.

8 Impressum

MindBytes GmbH | Probststraße 15 | 70567 Stuttgart

+49 711 20709567 | hallo@mind-bytes.de | https://mind-bytes.de

Amtsgericht Stuttgart, HRB 790784 | USt-IdNr: DE363069855

vertreten durch die **Geschäftsführung Christian Stehle, Nina Wagner, Simon Holl**