Legaler Einbruch – Pentesting

Schwarze Box

Legaler Einbruch: So kann ein Pentest aussehen

Schwarze Box

Unternehmen bezahlen Pentester dafür, dass sie versuchen, in ihre Systeme einzubrechen. So dubios das für Außenstehende klingen mag – Penetrationstests sind seit Jahren ein etabliertes Mittel, die Sicherheit von Systemen und Netzwerken zu überprüfen. Im Folgenden erfahren Sie, wie ein sogenannter Bl…

Unternehmen bezahlen Pentester dafür, dass sie versuchen, in ihre Systeme einzubrechen. So dubios das für Außenstehende klingen mag – Penetrationstests sind seit Jahren ein etabliertes Mittel, die Sicherheit von Systemen und Netzwerken

zu überprüfen. Im Folgenden erfahren Sie, wie ein sogenannter Black-Box-Test abläuft.

Von Michael Wiesner

kompakt

- Pentester sind vom Betreiber eines Netzwerks beauftragte Hacker.
- Sie nutzen dieselben Werkzeuge wie echte Angreifer und decken Schwachstellen auf.
- Lediglich ausgestattet mit der Domain will Michael Wiesner in die internen Systeme seines Auftraggebers einbrechen.

Es ist April 2022, ich will eigentlich gerade den Laptop zuklappen, da flattert eine Anfrage in mein Postfach. Ein mittelständisches Maschinenbauunternehmen aus dem Norden Deutschlands will mich für einen sogenannten Pentest buchen.

Ein Pentest kann vieles sein, das Spektrum reicht von Sicherheitsanalysen einzelner Applikationen oder Systeme bis hin zur Simulation zielgerichteter Angriffe. Noch umfassender sind sogenannte „Red Team Assessments“. Dabei überprüfen Pentester, wie gut Systeme und Mitarbeiter zur Erkennung und Abwehr von Angriffsversuchen ausgerüstet sind.

Im Videotelefonat am nächsten Tag schildert der Chef der IT-Abteilung des Auftraggebers, worum es geht: Ich soll ohne Kenntnis über die IT-Infrastruktur in interne Systeme des Unternehmens einbrechen. Als einziger Anhaltspunkt dient die Domain – eine Information, die jeder Mensch mit Zugang zum Internet innerhalb von Sekunden herausfinden könnte. „Black-Box-Test“ nennt man solche Penetrationstests, bei denen der Pentester agiert wie ein typischer Angreifer. Alle weiteren benötigten Informationen muss ich dabei – in Abgrenzung zum White-Box-Test, bei dem der Pentester über Insiderwissen verfügt – selbst herausfinden. Der Einbruchsversuch soll einen zielgerichteten Angriff simulieren und möglichst verdeckt über

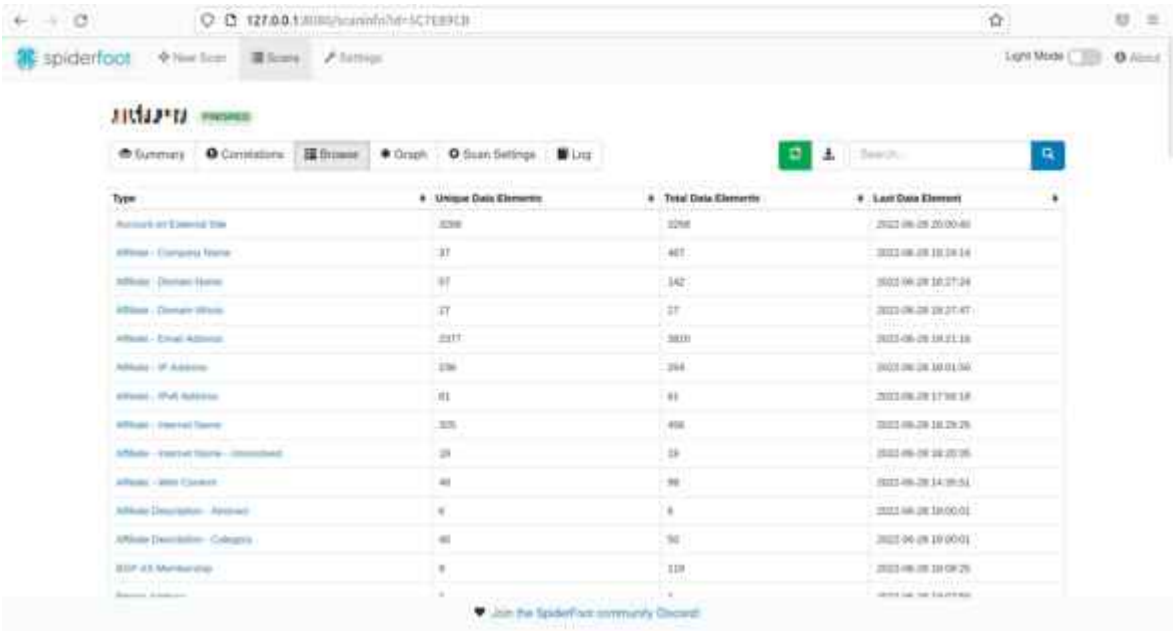
das Internet erfolgen. Entsprechend darf ich alle Mittel einsetzen, die auch ein echter Angreifer nutzen würde. Das könnte spannend werden – ich bin interessiert. Wir besprechen die Rahmenbedingungen und halten das Ganze vertraglich fest. In Angriff nehme ich den Test Anfang Juni.

Bei der Simulation eines solchen zielgerichteten Angriffs orientiere ich mich an den Phasen der MITRE ATT&CK Matrix. Darin werden die Taktiken und Techniken echter Cyberangriffe beschrieben und kategorisiert. Sie dient als Wissensdatenbank, die Verteidigern dabei hilft, Bedrohungen zu erkennen und abzuwehren, leistet mir bei einem Pentest, bei dem ich schließlich selbst in der Angreiferrolle stecke, aber ebenso gute Dienste.

Informationsbeschaffung

Ich starte mein Hacking-Vorhaben, indem ich alle frei verfügbaren Quellen nach Informationen über mein Ziel durchkämme. Dafür gibt es im Netz eine Reihe von Websites, Diensten und Datenbanken. „Open Source Intelligence“ (OSINT) nennt man diese Art der Informationsgewinnung auch (siehe c’t 16/2022, S. 138). Abfragen an WHOIS-Datenbanken und DNS-Server liefern mir erste Anhaltspunkte, mithilfe der Tools dnsrecon, spiderfoot und Shodan automatisiere ich einen Großteil der Arbeit. Das Python-Skript dnsrecon füttere ich im Bruteforce-Modus mit der Domain und einer Wortliste – es fragt Subdomains und Hostnamen ab und wertet praktischerweise anschließend gleich aus, welche IP-Adressen sich dahinter verbergen. Wie erwartet, liefert das Skript, und ich erhalte eine recht umfassende Liste der öffentlich erreichbaren Systeme meines Auftraggebers. Über die Kommandozeile rufe ich das OSINT-Tool Spiderfoot auf. Es nutzt eine größere Anzahl von Quellen für die Informationsgewinnung. Zum Beispiel ermittelt es mögliche Hostnamen auch durch die verwendeten TLS-Zertifikate. Auch liefert das Tool gültige Mailadressen, Telefonnummern, ähnliche oder verbundene IP-Adressen und Domains (siehe Bild

unten). Auch der Webdienst Shodan – sicher das prominenteste Beispiel für solche Scanner – liefert umfangreiche Informationen über die öffentlich erreichbaren Systeme meines Auftraggebers, die ich möglicherweise für den eigentlichen Angriff nutzen kann.



Spiderfoot ist ein OSINT-Tool, das dem Nutzer gleich eine ganze Palette an Informationen liefert, darunter auch Mailadressen.
Meine Zugriffe auf die öffentlichen Webseiten meines Auftraggebers werden zwar sehr sicher protokolliert, jedoch nicht blockiert, also kann ich davon ausgehen, dass sie nicht als schädlich erkannt werden. Zahlreiche Unternehmen, Organisationen und Einzelpersonen durchforsten das Internet laufend nach interessanten Systemen, Anwendungen oder Inhalten, sodass ein gewisses Grundrauschen besteht. Meine vorsichtig durchgeführten Verbindungsversuche gehen offenbar darin unter.

Vorsichtig anklopfen

Die gesammelten Daten verraten mir bereits eine ganze Menge über die Systeme und Anwendungen, die mein Auftraggeber verwendet, denn aus den DNS-Hostnamen kann ich ableiten, um welche Dienste es sich handelt: Bei „owa“ kann ich davon ausgehen, dass ein Exchange-Server betrieben wird und dieser

über „Outlook Web Access“ im Internet zur Verfügung gestellt wird. Generische Namen, wie „vpn“ oder „sslvpn“ sind selbsterklärend, „citrix“ weist auf ein Remote-Access-Gateway des gleichnamigen Herstellers hin. Teilweise sind die Hostnamen gleich, aber durchnummeriert – etwa owa2. Das könnte ein Hinweis darauf sein, dass mein Auftraggeber mehrere Versionen einer Anwendung einsetzt, oder darauf, dass ein Dienst über unterschiedliche Internetanbindungen bereitgestellt wird.

Die so gewonnenen Informationen über die Systeme und Dienste meines Auftraggebers sind noch nicht komplett. Aber um erste Angriffe zu starten, reichen sie aus. Für eine vollständigere Übersicht fehlt mir die Zeit – für den Penetrationstest sind lediglich sechs volle Arbeitstage veranschlagt – etwa zwei davon habe ich bereits für die Reconaissance-Phase, wie man die Phase der Informationsgewinnung im Fachjargon nennt – bereits aufgebracht. Für eine möglichst vollständige Übersicht muss man Ports scannen, beispielsweise mittels nmap oder einem Schwachstellenscanner wie Nessus oder OpenVAS. Die Herausforderung dabei ist, diese Scans so vorsichtig wie möglich durchzuführen, um weiterhin unentdeckt zu bleiben. Konkret bedeutet das, dass der Scan über einen langen Zeitraum verteilt werden muss, weil nur wenige gleichzeitige Verbindungen aufgebaut werden dürfen.

Da der vereinbarte Umfang und Zeithorizont des Penetrationstests kein solches Vorgehen erlaubt, wende ich die Holzhammermethode an: Über einen nur für diesen Vorgang genutzten Internetzugang starte ich ohne Rücksicht auf Verluste einen Portscan auf alle IP-Adressen und Ports. Dieser bleibt wegen des schon erwähnten Grundrauschens tatsächlich unbemerkt, liefert aber leider nicht die gewünschten Ergebnisse. Die „Portscan Protection“ der Firewall meines Auftraggebers blockiert den Scan erfolgreich. Das merke ich daran, dass mein Scan anfänglich zwar Ergebnisse liefert, die Systeme nach einer gewissen Zeit aber aufhören zu antworten.

Die Portscan Protection verlangsamt meine Verbindungsanfragen entweder stark oder blockiert sie ganz.

Für den Einstieg in die nächste Phase, die der Schwachstellenidentifikation, bleiben mir also nur die bereits gewonnen Informationen. „Schade“, denke ich – ganz so leicht scheint der norddeutsche Mittelständler es mir an dieser Stelle nicht zu machen.

Schwachstellensuche

Auch bei der Suche nach Schwachstellen greife ich auf Shodan zurück. Ich nutze den Webdienst, um auf Basis der Versionsnummern zu ermitteln, ob es bekannte Schwachstellen in den Diensten gibt. Leider ohne Treffer – laut Shodan hat nicht einer davon eine Schwachstelle, die ich für einen Angriff ausnutzen hätte können.

Ich muss wohl doch etwas genauer hinschauen: Mithilfe von Shodan, Spiderfoot und Dnsrecon habe ich knapp 70 offene Ports identifiziert. Das heißt, etwa 70 erreichbare Dienste warten darauf, genauer unter die Lupe genommen zu werden. Ich gehe systematisch vor. Zunächst filtere ich Dienste heraus, die nicht selbst betrieben werden und damit auch keinen potenziellen Zugriff auf die IT-Infrastruktur erlauben, wie etwa die Website beim Hoster. Die hebe ich mir für später auf, für den Fall, dass ich keinen direkten Weg in das Netzwerk meines Auftraggebers finden sollte. Die verbleibenden Portnummern geben Preis, um welche Art von Dienst es sich handelt. Um herauszufinden, welche Software und Version sich dahinter verbirgt, setze ich die Kommandozeilentools netcat oder alternativ telcat ein.

Zunächst surfe ich die identifizierten Webserver allerdings manuell über einen Webbrowser an, um zu erfahren, was sich hinter der URL tummelt. Dabei achte ich peinlich genau darauf, dass ich nicht nur die IP-Adresse, sondern auch den jeweilige Fully-Qualified Domain Name, kurz FQDN verwende. Dieser

vollständige Domainname einer Internetpräsenz ist eindeutig und er lässt sich den zum Nameserver gehörenden IPv4- oder IPv6-Adressen zuordnen. Das ist wichtig, weil oft mehrere unterschiedliche Webserver hinter der gleichen IP-Adresse betrieben werden – ohne die Angabe des FQDN würde ich möglicherweise nicht an die gewünschten Informationen kommen. Weil ich auf diese Weise – im Unterschied zum fehlgeschlagenen „Holzhammerscan" – nur noch einzelne Ports kontaktiere, könnte ich nun eigentlich umfangreiche Schwachstellenscans durchführen, ohne dass die Firewall wieder den Riegel vorschieben würde. Die Betonung liegt auf eigentlich – denn schon als ich die Websites manuell ansurfe, lande ich auf Login-Pages – ein deutlicher Hinweis darauf, dass mein Auftraggeber einen Reverse Proxy verwendet, um die Webseiten im Internet zu veröffentlichen. Das bedeutet, dass die Server nicht direkt angesprochen werden, sondern die Verbindungen vorab von einer Stellvertretersoftware angenommen werden. Schlimmer noch: Über die Eingabe bestimmter Parameter fingiere ich eine Directory-Traversal-Attacke und finde heraus, dass der Reverse Proxy zusätzlich über eine sogenannte Web Application Firewall verfügt. Das sind Systeme zur Erkennung und Abwehr von Angriffen, kurz WAF.

Das trübt meine Aussichten auf einen erfolgreichen Angriff über Sicherheitslücken in Web-Applikationen erheblich. Zähneknirschend verzichte ich auf einen umfangreichen Schwachstellenscan der Web-Apps – schließlich will ich die WAF nicht alarmieren. Ich verwende nikto und ein kommerzielles Tool namens Nessus, um die Websites auf Schwachstellen zu prüfen, werde jedoch nicht fündig. Komplett ertragslos verläuft meine Schwachstellensuche zum Glück trotzdem nicht. Beim manuellen Ansurfen haben Login-Pages mir verraten, dass es sich bei drei der Websites des Auftraggebers um Fernzugriffsportale handelt. Mithilfe von Nessus scanne ich sie ebenfalls auf Schwachstellen und gleiche zusätzlich die Versionsnummern mit der CVE-Datenbank https://cve.mitre.org ab. Leider fördert keine meiner Bemühungen eine

Sicherheitslücke in einem der Fernzugriffsportale zutage, aber ich nehme mir trotzdem vor, diese vorerst im Hinterkopf zu behalten.

Ein Schritt vor und zwei zurück

Die ernüchternde Zusammenfassung bis zu diesem Punkt: Keines der öffentlich erreichbaren Systeme des Auftraggebers besitzt offensichtliche Schwachstellen, die ich direkt zum Einbruch in die Systeme oder das Netzwerk hätte nutzen können. „Wäre ja auch zu leicht gewesen“, denke ich, während ich meine Optionen für das weitere Vorgehen abwäge. Ich habe nicht mehr viel Zeit, knapp zwei Drittel der maximal veranschlagten sechs Arbeitstage sind bereits verstrichen. Eine aufwendige Untersuchung der restlichen Webseiten, zum Beispiel mittels der beliebten Burp Suite fällt daher flach. Ein erneuter Blick auf die vereinbarte Leistungsbeschreibung zaubert mir dann aber doch ein Lächeln auf die Lippen. Fast hätte ich es vergessen, aber dort steht schwarz auf weiß, dass ich auch Phishingmethoden einsetzen darf. Phishing hat in der Regel das Ziel, den Empfänger dazu zu verleiten, Informationen preiszugeben oder ihn dazu zu bringen, Dateianhänge oder Links auf verseuchte Websites anzuklicken, über die dann Schadprogramme – sogenannte Remote-Access-Trojaner, kurz RAT – ausgeführt werden, die einen Zugang zum betroffenen System öffnen.

Spiderfoot hat mir während der Informationsbeschaffungsphase bereits eine Liste von circa 20 E-Mail-Adressen geliefert, denn auf der Webseite des Auftraggebers werden einige Mitarbeiter mitsamt der E-Mail-Adressen präsentiert. Das ist ein guter Start, aber ich würde die Angriffsfläche gerne vergrößern. Dabei spielen mir die beliebten Business Social Networks Xing und LinkedIn in die Hände. Anhand der Spiderfoot-Liste weiß ich ja bereits, wie die Unternehmens-Mailadressen aufgebaut sind und mit den Namen der Beschäftigten aus den Business-Netzwerken kann ich über ein

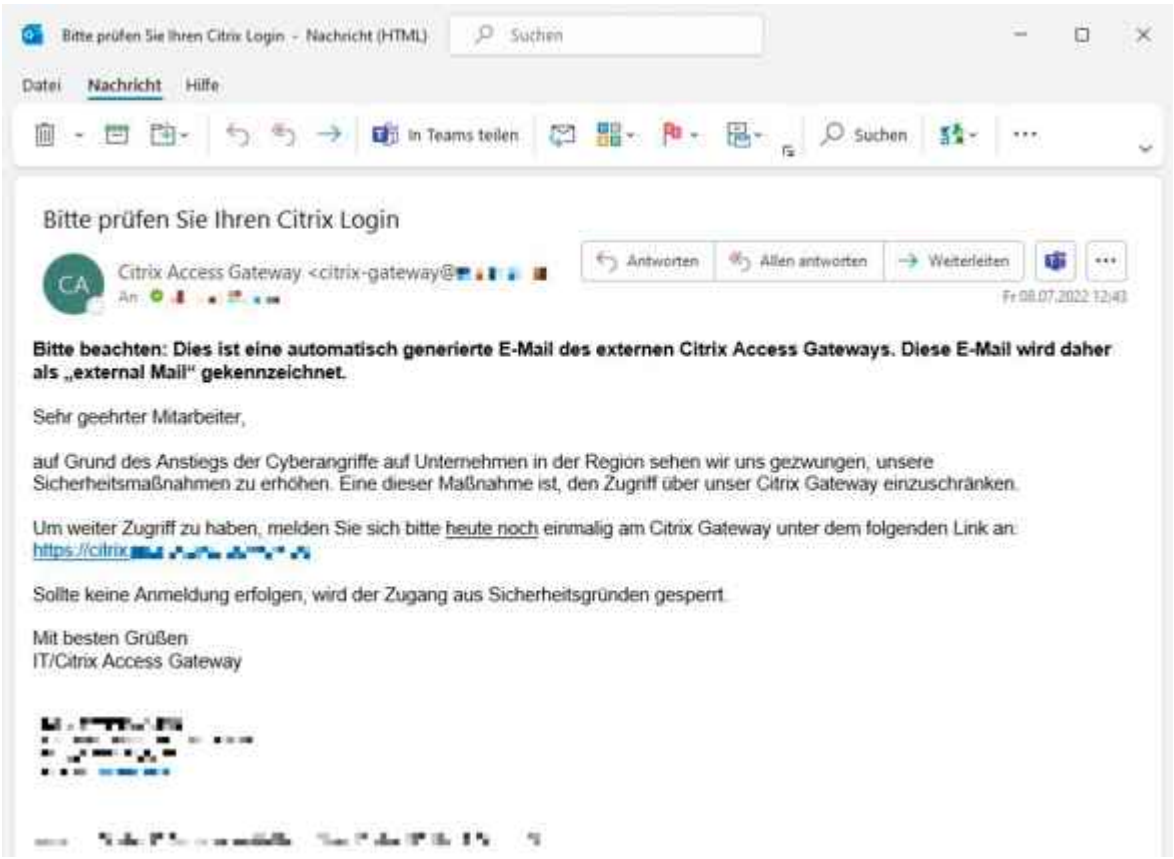
einfaches Skript leicht 50 weitere Mailadressen generieren.

Grundsätzlich sind solche Business-Portale ein Quell nützlicher Informationen. Über die eingetragenen Kenntnisse, Fähigkeiten oder die „ich biete“-Felder der Mitarbeiter lässt sich oft herausfinden, welche IT-Hersteller und Produkte eingesetzt werden. Dies ist besonders hilfreich, wenn die Hersteller von Firewalls, E-Mail-Gateways oder Endpoint-Security-Produkten genannt werden. Die Wahrscheinlichkeit ist hoch, dass diese dann auch in deren Unternehmen eingesetzt werden.

Eigene RATs zu erstellen, die nicht von der verwendeten Antivirussoftware erkannt werden, ist längst kein Hexenwerk mehr, so es sich denn um einen klassischen Virenschutz handelt. Ich versuche, den Trojaner über eine verschlüsselte Zip-Datei an der Firewall – beziehungsweise dem E-Mail-Security-Gateway – vorbeizuschmuggeln und erwarte fast, so mein Ziel zu erreichen, schließlich hat mich diese Methode in der Vergangenheit schon oft zum gewünschten Ziel geführt. Nicht jedoch bei diesem Penetrationstest. Mein Auftraggeber filtert sehr erfolgreiche alle E-Mails mit entsprechenden Dateianhängen heraus. Damit nicht genug – eine schnelle Recherche über den Webdienst mxToolbox zeigt, dass die absendende IP-Adresse bereits kurz nach den ersten Versuchen auf den bekannten Blacklisten landet. Leider führt auch das Einschleusen bösartiger Links für einen „Drive By Exploit“ nicht zum Ziel. Mir wird klar, dass ich es anders versuchen muss. Nur wie? – Die zuvor identifizierten drei Fernzugriffsportale kommen mir in den Sinn. Sie böten ideale Ansatzpunkte für eine Phishing-Kampagne, die auf das Abfischen von Zugangsdaten abzielt.

Ich mache mich an das Basteln einer weiteren Phishing-Mail. Sie soll so offiziell wie möglich aussehen – inklusive farblich passendem Layout und einer authentisch wirkenden Signatur. Von einer Wegwerf-Mailadresse frage ich höflich bei der jobs@-Mailadresse des Unternehmens an, an wen ich denn

meine Initiativbewerbung schicken könne. Wie erwartet, bekomme ich eine freundliche Antwort mit der gewünschten Information – und der Standardsignatur des Auftraggebers.



Mittels einer Phishing-Mail wird versucht, die Mitarbeiter des Auftraggebers auf eine Fake-Website zu locken.
Außerdem brauche ich eine Phishing-Webseite, in die mindestens eines meiner Opfer hoffentlich die Zugangsdaten eingeben wird. Gefälschte Webseiten bekannter Dienste, wie Microsoft Office 365, Facebook, Instagram oder Twitter lassen sich leicht über freie Tools wie zphisher erzeugen.

Zphisher erstellt Fake-Websites bekannter Dienste. Anpassen lassen sich diese aber leider nur bedingt.
In der Phishing-Mail müsste man dann nur noch auf die entsprechende Webadresse verweisen, um die eingegebenen Zugangsdaten anschließend abrufen zu können. Anpassungen, wie zum Beispiel das Logo meines Auftraggebers einzubinden, kann man daran aber leider nur bedingt vornehmen – für meine Zwecke kann ich zphisher deshalb leider nicht nutzen. Stattdessen erstelle ich manuell einen Klon von einem der verwendeten Fernzugriffsportale.

Eine geklonte Fake-Website soll die Phishing-Opfer dazu verleiten, ihre Zugangsdaten einzugeben.
Bleibt noch die Frage, auf welcher Adresse das gefakte Portal betrieben werden soll. Die Originaladresse lautet „citrix.DOMAIN.com“ – für meinen Klon verwende ich kurzerhand eine ähnlich aussehende Adresse: „citrix-DOMAIN.com“. Solche Doppelgänger-Domains sind auch bei realen Angreifern beliebt, da man auf den ersten Blick den Unterschied zwischen echten und gefälschten Adressen nicht erkennt.

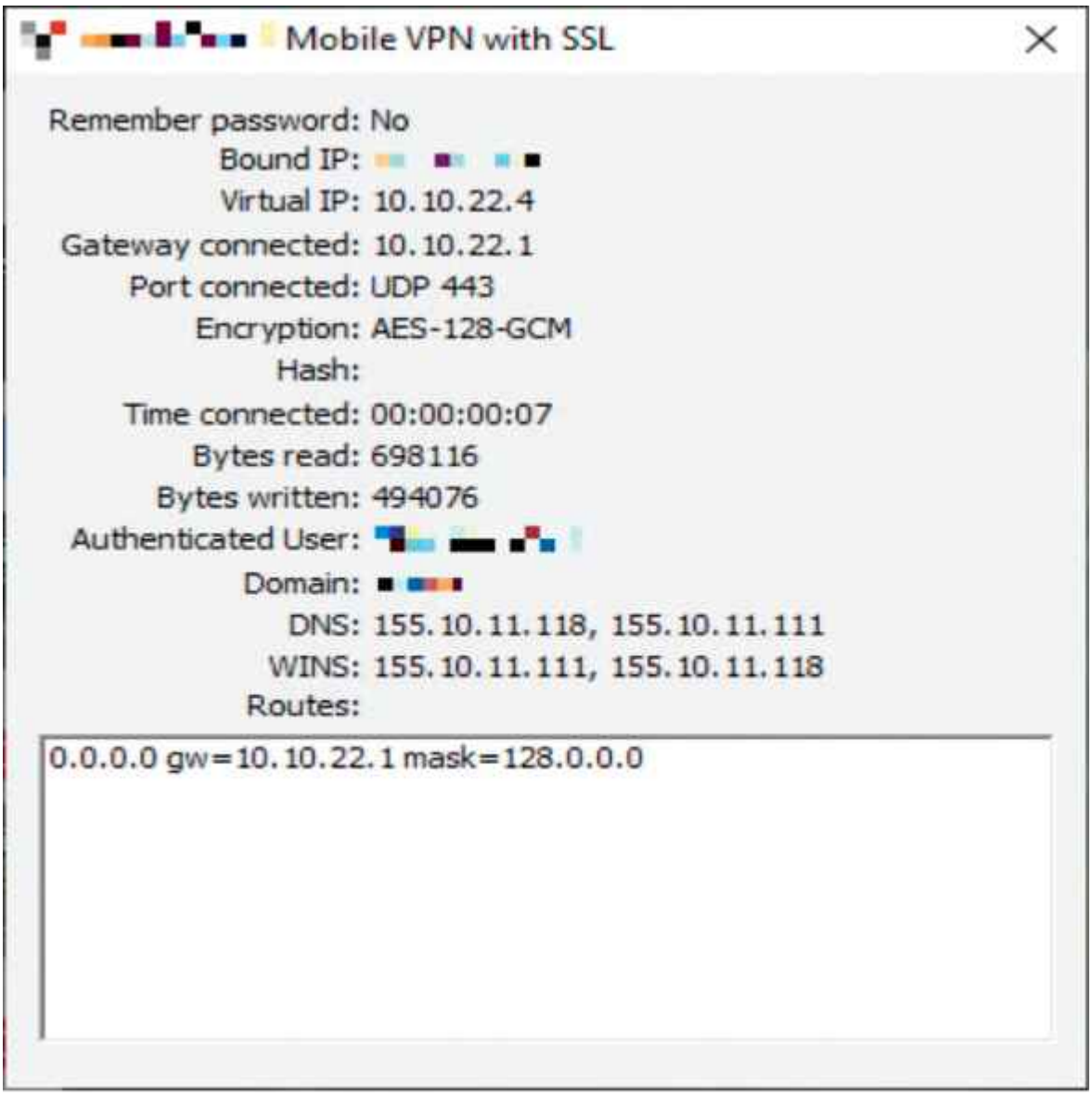
Bevor ich im großen Stil loslegen kann, gilt es vorab zu prüfen, ob die fingierte E-Mail die Empfänger überhaupt erreicht, oder ob sie – wie meine ersten beiden Versuche – durch Sicherheitssoftware blockiert wird. Ein Test mit ausgewählten Empfängern bringt die Ernüchterung: Das E-Mail-Gateway blockiert meine Phishing-Mails. Eine Überprüfung mittels mxToolbox verrät mir, dass meine IP erneut auf der Blacklist gelandet ist. Ich gerate ins Grübeln und ärgere mich zugegebenermaßen ein wenig, weil ich nicht gleich dahinter komme, warum. Die E-Mail war standardkonform, die IP-Adresse des Servers befand sich bis dato auf keiner Blacklist und auch der Text der E-Mail war nicht besonders spammy. Aber das E-Mail-Gateway ist anscheinend schlauer als gedacht: Ich vermute, es deklariert den eingebetteten Link auf die gefälschte Webseite als gefährlich, weil er Teile des Domainnamens des Auftraggebers enthält, jedoch nicht zu dessen

Servern gehört. Mir bleibt nur die Registrierung und Nutzung einer unverdächtigen Domain, die lediglich „citrix“ als Hostname aufführt und sonst möglichst offiziell aussieht. Ein Test mit dieser URL verläuft erfolgreich: Die E-Mails werden zugestellt. Anhand eintreffender Abwesenheitsnotizen kann ich zudem erkennen, dass die E-Mails nicht als Spam markiert wurden. „Feuer frei!“, denke ich grinsend.

Die eigentliche Phishing-Kampagne starte ich an einem Montag um 9 Uhr – pünktlich zum üblichen Arbeitsbeginn der Verwaltung. Und Bingo: Die ersten Zugangsdaten werden um 9:39 Uhr eingegeben. Jetzt darf ich keine Zeit verlieren. Wie klein mein Zeitfenster ist, kann ich nicht abschätzen, aber ich muss handeln, bevor es möglicherweise jemandem auffällt, dass Phishing-Mails im Umlauf sind und die Mitarbeiter – inklusive meiner Opfer – aufgefordert werden, ihre Zugangsdaten zu ändern.

Open the gates

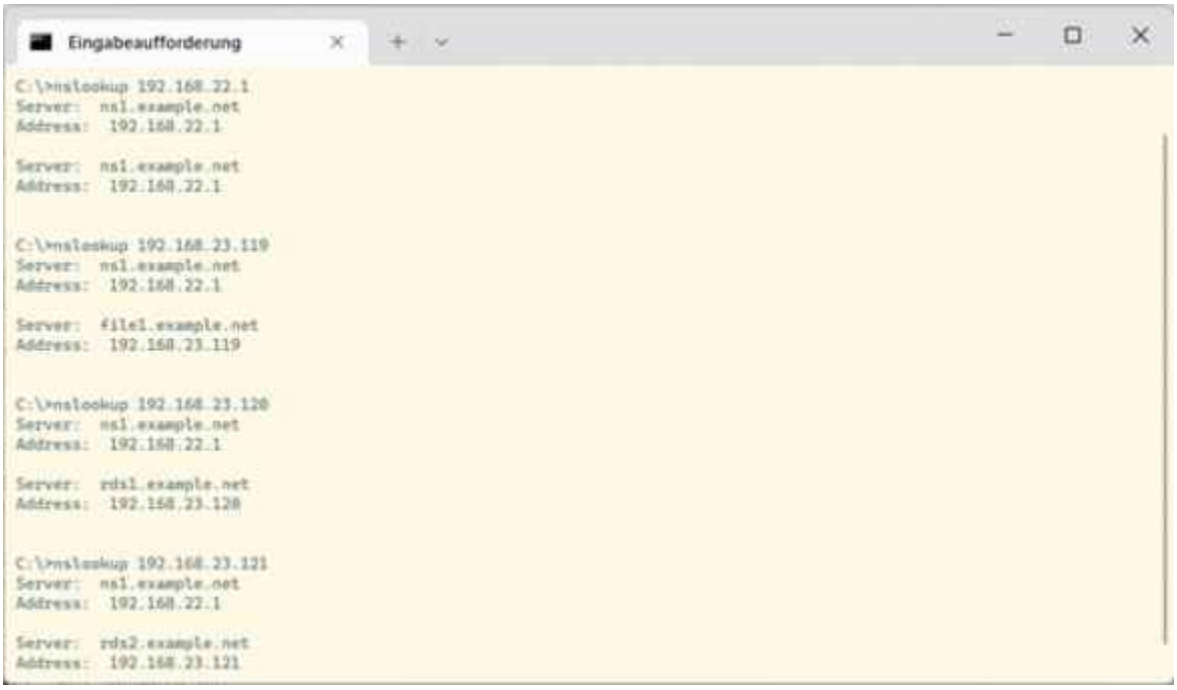
Meine Euphorie verfliegt, als die wirkliche Citrix-Anmeldeseite nach Eingabe der erbeuteten Zugangsdaten „Bitte Einmalpasswort eingeben“ meldet. Das Portal ist per Multi-Faktor-Authentifizierung abgesichert – und den zur Anmeldung benötigten zweiten Faktor besitzt nur der legitime Nutzer. Eins zeigt die Meldung jedoch: Die Zugangsdaten stimmen. Also versuche ich mein Glück beim nächsten Fernzugangsportal. Meine Hoffnung schwindet, als es ein Einmalpasswort anfordert. Aber ein Portal bleibt mir noch. Ohne große Hoffnung gebe ich die erbeuteten Zugangsdaten auch hier ein und halte die Luft an. Aber ich habe Glück: Nach dem Klick auf Enter zeigt mir das Portal einen Download-Link zum VPN-Dienst, den mein Auftraggeber verwendet. Ich installiere die Software, gebe die erbeuteten Zugangsdaten auch an dieser Stelle ein – und Bingo! – Der VPN-Client baut eine erfolgreiche Verbindung zum internen Netzwerk des Auftraggebers auf. Ich atme tief durch. Die nächsten Schritte muss ich sorgfältig planen.



Bingo! Das VPN-Gateway baut eine Verbindung zum internen Netzwerk auf.

Es gilt jetzt, das interne Netzwerk zu erkunden, um interessante Systeme und Daten zu identifizieren. Die Gefahr, erkannt zu werden, steigt dabei mit der Aggressivität des Vorgehens und den verwendeten Tools. Werden in dieser Phase Schwachstellenscanner oder Angriffswerkezuge, wie zum Beispiel Metasploit eingesetzt, ist die Gefahr groß, dass Endpoint-Security-Systeme oder vorhandene Intrusion-Detection-Systeme im Netzwerk dies erkennen, melden und anschließend die Verbindung zum internen Netzwerk getrennt wird. Unfehlbar sind solche Sicherheitsvorrichtungen allerdings nicht. Es ist möglich, sie auszutricksen, indem man sich als Angreifer verhält wie der eigentliche Anwender.

Aber ich will zuerst die wichtigste Frage klären – und dafür brauche ich sowieso noch keine Tools: Handelt es sich bei den VPN-Zugangsdaten auch um die Windows-Zugangsdaten? Klarheit verschafft mir die Anmeldung am Netlogon-Verzeichnis des Domain-Controllers. Es enthält in der Regel die Anmeldeskripte und kann daher auch von jedem Domain-Benutzer gelesen werden. Die Anmeldung funktioniert – ich habe tatsächlich die Zugangsdaten der Windows-Domain erbeutet.



Das Tool nslookup findet nach Eindringen in das Netzwerk die Namen weiterer Server heraus.

Jetzt könnte ich über PowerShell-Skripte oder Tools wie BloodHound oder PingCastle das Netzwerk und die Windows-Domain nach Schwachstellen durchforsten (siehe ct.de/yhxe). Ich entscheide mich aber lieber für die vorsichtigere Variante und sehe mich erst einmal manuell um. Im gleichen Netzbereich wie die Domain-Controller befinden sich üblicherweise auch weitere Server, deren Namen man durch einen Reverse-Lookup mittels des Standard-Tools nslookup auflösen kann. Durch den Aufbau der Hostnamen kann ich Rückschlüsse auf weitere Server ziehen. „rds1“ verweist etwa auf den ersten Terminalserver (Remote Desktop Services); „file1“ auf den ersten Fileserver (vergleiche Bild auf Seite 116). Getreu dem Motto „wer nicht wagt, der nicht gewinnt“, versuche ich mich auf den ersten

Terminalserver einzuloggen – und habe Erfolg. Der Server präsentiert die Standard-Arbeitsumgebung des unglückseligen Benutzers, dessen Zugangsdaten ich abgefischt habe – mitsamt allen Applikationen. Ich habe kompletten Zugriff im Kontext des ausgespähten Benutzers, inklusive E-Mail, Dateiablage, ERP-Software und Microsoft-365-Diensten, wie Sharepoint Online und Teams. Zudem offenbart ein kleines blaues Doppelpfeil-Symbol in der System Tray, dass der Auftraggeber das Support-Werkzeug TeamViewer einsetzt. Es könnte mir als mögliche Hintertür dienen, falls der Angriff erkannt und das VPN-Gateway abgeschaltet wird. Ein Blick in den Task-Manager des Servers zeigt die laufenden Dienste einer marktführenden „Endpoint Detection and Response"-Software, kurz EDR. Mein vorsichtiges Vorgehen war also mehr als angebracht. Der Versuch, zwischengespeicherte Authentifizierungsdaten auszulesen, zum Beispiel mit dem beliebten Tool Mimikatz, hätte höchstwahrscheinlich dazu geführt, dass ich entdeckt und aus dem System ausgesperrt worden wäre.

Bei einem klassischen „Double Extortion"-Angriff von Cyberkriminellen würden diese nun beginnen, die gefundenen Dateien zu exfiltrieren, sich im Netzwerk weitere Berechtigungen zu verschaffen, möglicherweise die Datensicherung zu manipulieren und Daten zu verschlüsseln, um anschließend Löse- beziehungsweise Schweigegeld zu fordern. Die Exfiltration von Daten bleibt meistens unerkannt. Ich simuliere den Vorgang, indem ich mehrere Gigabyte Daten in ein Zip-Archiv packe und es anschließend auf einen Webserver im Internet hochlade. Wie erwartet, bleibt der Vorgang unbemerkt. Aus Zeitgründen muss ich auf eine weitere Eskalation der Berechtigung verzichten – möglicherweise wird es dafür einen weiteren Penetrationstest geben.

Nachklang

Ich rufe meinen Auftraggeber an und informiere ihn mündlich über die gravierendsten Sicherheitslücken. Er zeigt sich

ernüchtert vom Resultat meines Penetrationstests. Weil ich keine Angriffswerkzeuge oder Schadsoftware eingesetzt habe, waren die im Unternehmen eingesetzten Sicherheitslösungen gegen meinen Angriff schlussendlich wirkungslos. Am Ende war es – wie so oft – menschliches Versagen, das mir ein Einfallstor in die geschützte Infrastruktur meines Auftraggebers eröffnete. Mehrere Mitarbeiter fielen auf meine Phishing-Attacke herein und die fehlende Multi-Faktor-Authentifizierung bei einem von drei Fernzugriffsportalen führte dazu, dass ich die erbeuteten Zugangsdaten tatsächlich nutzen konnte, um ins System einzubrechen. Eine alte Weisheit lautet „Der Angreifer muss nur einmal gewinnen, der Verteidiger immer“ – und dieser Black-Box-Test hat einmal öfter gezeigt, dass etwas Wahres dran ist.

Abgeschlossen ist die Geschichte an dieser Stelle allerdings weder für mich noch für meinen Auftraggeber. Zu meinen Aufgaben als Pentester gehört es auch, am Ende des Pentests einen aussagekräftigen Abschlussbericht zu erstellen. Notgedrungen setze ich mich wieder an den Schreibtisch und fasse die durchgeführten Schritte und die Ergebnisse des Black-Box-Tests zusammen. Ich beschreibe die entdeckten Schwachstellen genau und versuche, mich dabei möglichst verständlich auszudrücken. Das Ziel ist es schließlich, dass mein Auftraggeber die von mir entdeckten Schwachstellen versteht und sie beseitigen kann. *(kst@ct.de)*

Alle erwähnten und verwendeten Werkzeuge: ct.de/yhxe

Social-Engineering-Angriffe

auf Instagram-Accounts und wie Sie sich davor schützen

Instahack

Social-Engineering-Angriffe auf Instagram-Accounts und wie Sie sich davor schützen

Immer wieder kapern Phisher fremde Instagram-Accounts, um Profit daraus zu schlagen. So auch im Fall einer deutschen Olympiaschwimmerin, die sich Hilfe suchend an c’t wandte. Wir sind der Sache nachgegangen und stießen dabei auf weitere Fälle. Wir erklären, wie Sie Ihren Account schützen.

Von Ronald Eikenberg und Marie-Claire Koch

kompakt

- Instagram-Accounts, egal ob sehr populär oder nahezu unbekannt, sind ein lukratives Angriffsziel für Cyber-Kriminelle.
- Es ist wichtig, den Account mehrstufig abzusichern, wenn man nicht Gefahr laufen will, ihn für immer zu verlieren.
- Wer eine Mail erhält, die angeblich von Instagram stammt, sollte in der App kontrollieren, ob die Mail echt ist.

Phisher versuchen immer wieder an Zugangsdaten für Social-Media-Dienste wie Instagram zu kommen, um Accounts zu kapern und Profit daraus zu schlagen [1] – zum Beispiel durch Lösegeldforderungen oder dubiose Spam-Kampagnen. Dafür ist den

Angreifern jeder Account gut genug, doch besonders hoch im Kurs stehen Instagram-Accounts, die der begehrte blaue Haken ziert. Er zeigt, dass es sich um ein durch Instagram verifiziertes Profil einer Person öffentlichen Interesses handelt. Aber auch mit nicht verifizierten Accounts können Phisher Geld machen, mangelndes öffentliches Interesse schützt Ihren Account daher nicht.

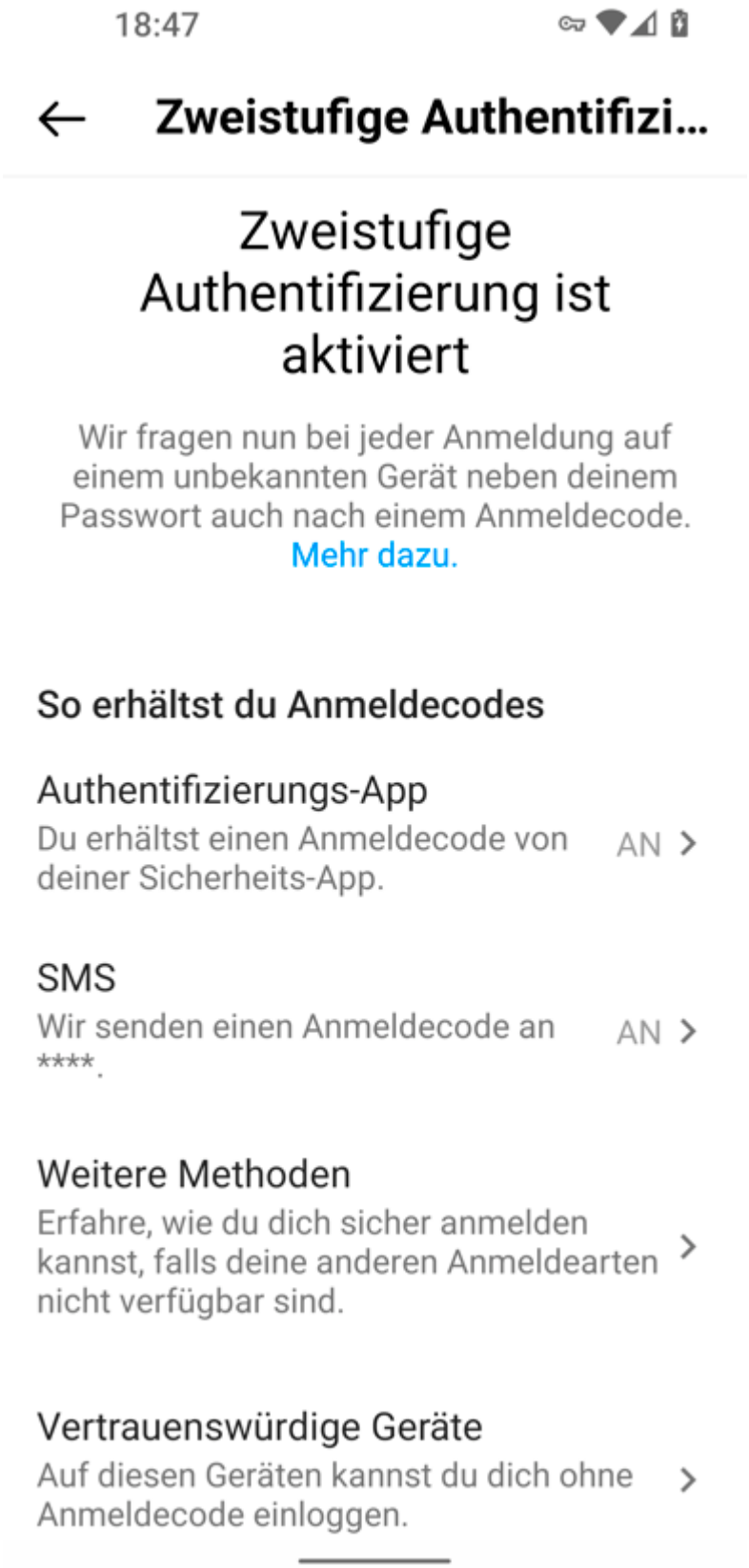
Der Instagram-Account einer Berliner Olympiaschwimmerin trägt diesen blauen Haken. Sie nutzt den Account, um mit ihren Fans in Kontakt zu bleiben und ihre Erfolge zu teilen – zum Beispiel ihre Teilnahme an den Olympischen Spielen in Tokio oder zuletzt an der Europameisterschaft in Rom. Vor einigen Monaten entdeckte auch ein Phisher die erfolgreiche Schwimmerin bei Instagram. Er kontaktierte sie über eine private Nachricht, gab sich als Instagram-Support aus, um sie in die Falle zu locken, und konnte letztlich die Kontrolle über ihren Account übernehmen.

Man spricht bei solchen Angriffen von Social Engineering, also der gezielten Manipulation des Opfers. Als die Schwimmerin bemerkte, wie ihr geschah, war das Kind bereits in den Brunnen gefallen. Der Angreifer hatte das Instagram-Konto bereits fest im Griff und die Account-Sprache auf Arabisch geändert. Die Schwimmerin wandte sich daraufhin an einen IT-Experten, der den Account jedoch auch nicht mehr retten konnte. Der Täter forderte unterdessen ein Lösegeld in Höhe von 150 Euro, zahlbar via PayPal.

Passwort: „Passwort“

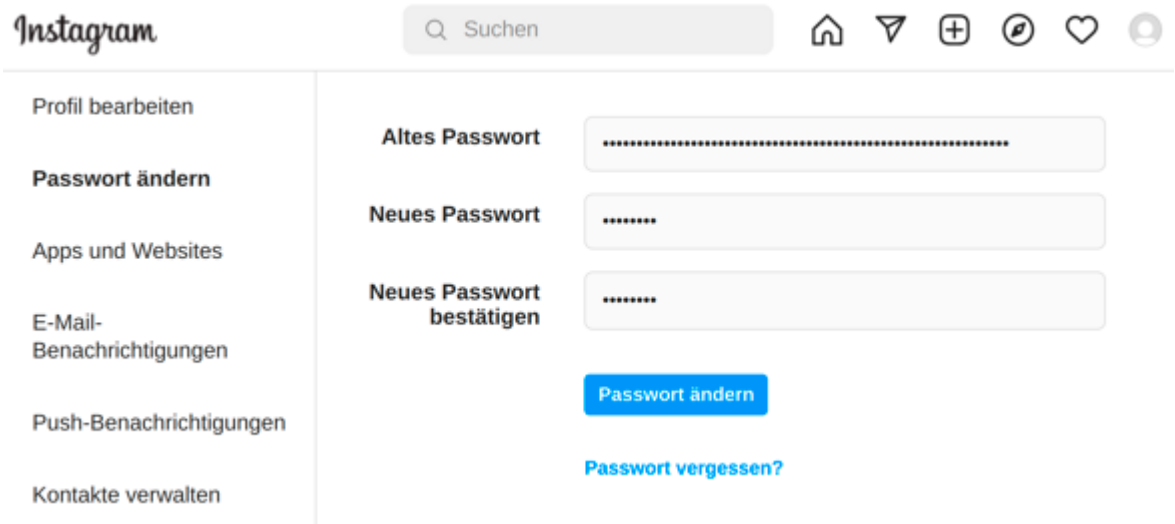
Statt der dreisten Lösegeldforderung nachzukommen, wandten sich die beiden an c’t. Im Rahmen unserer Recherche stießen wir auf drei weitere Sportlerinnen und Sportler aus dem Umfeld der Schwimmerin, deren Accounts ebenfalls gehackt waren. In zwei Fällen war ebenfalls Social Engineering im Spiel, im dritten wurde offenbar das Passwort erraten – es lautete schlicht „Passwort“. Alle betroffenen Accounts waren nicht

nach Stand der Technik abgesichert: Die sogenannte Zwei-Faktor-Authentifizierung (2FA), die Angriffe auf Online-Accounts in den meisten Fällen vereiteln kann [2], war nicht eingeschaltet.



Wer einen Instagram-Account besitzt, sollte die zweistufige Authentifizierung einschalten.
Ist die 2FA aktiv, ist zumindest beim ersten Einloggen auf einem Gerät neben dem Passwort auch noch ein zweiter Faktor nötig. Das kann zum Beispiel ein kurzzeitig gültiger Zahlencode sein, den man per SMS bekommt oder mit einer App wie dem Google Authenticator selbst generiert. Ein Hacker kommt in aller Regel nicht an SMS und erst recht nicht an das Geheimnis in der Authenticator-App. Mit einem erbeuteten Passwort kann er sich daher nicht einloggen.

Um die gehackten Instagram-Accounts der Athleten zu retten, kontaktierten wir die Pressestelle des Instagram-Betreibers Meta. Kurz darauf konnten die rechtmäßigen Account-Besitzer wieder auf ihre Konten zugreifen. Uns erreichen immer wieder ähnliche Zuschriften von Instagram-Nutzern, die Opfer von Cyber-Ganoven geworden sind. Weil wir nicht immer helfen können und damit es erst gar nicht so weit kommt, möchten wir Ihnen im Folgenden die wichtigsten Sicherheitstipps an die Hand geben, damit Sie Ihren Instagram-Account – oder die Accounts Ihrer Sprösslinge – angemessen absichern können.



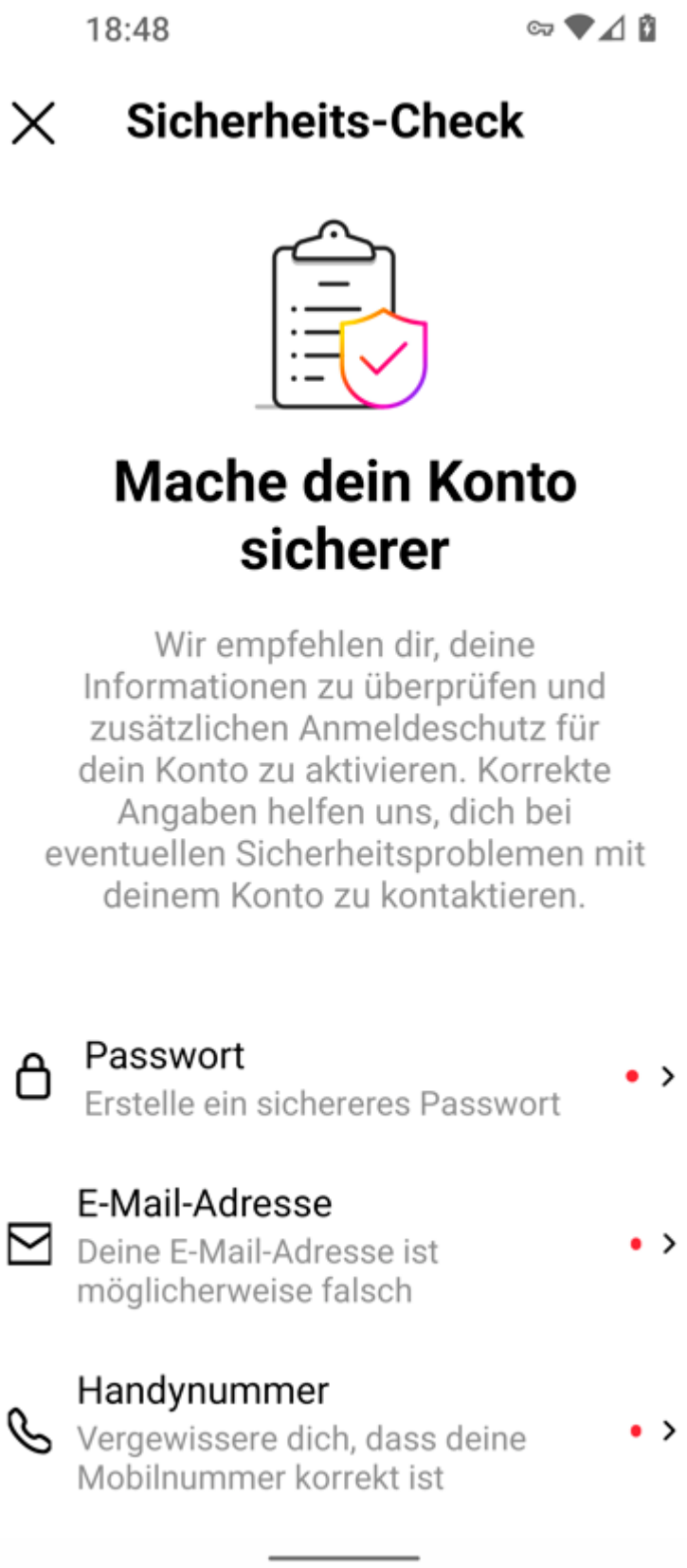
Passwort: „Passwort“ – die Passwortanforderungen von Instagram sind eher locker, damit haben Cyber-Ganoven wie in diesem Fall dann leichtes Spiel.

Instagram-Account absichern

Der beste Zeitpunkt, um sich um die Sicherheit Ihres Instagram-Accounts zu kümmern, ist genau jetzt, nicht später heute Abend oder am Wochenende. Sie müssen nur wenig Zeit investieren und ersparen sich früher oder später viel Ärger. Wenn Sie die von Instagram bereitgestellten Werkzeuge kennen und nutzen, ziehen die meisten Angreifer unverrichteter Dinge zum nächsten Account weiter, der womöglich weniger gut abgesichert ist.

Den effektivsten Schutz gegen Phishing-Angriffe bietet die bereits erwähnte Zwei-Faktor-Authentifizierung (2FA), die Instagram „Zweistufige Authentifizierung“ nennt. In der Instagram-App aktivieren Sie den Schutz über den Menüknopf oben rechts und „Einstellungen/Sicherheit/Zweistufige Authentifizierung“, auf der Website klicken Sie in den Einstellungen auf „Privatsphäre und Sicherheit“, um die zweistufige Authentifizierung zu finden. Anschließend haben Sie die Wahl, ob Sie die zum Einloggen nötigen Zahlencodes per SMS zugeschickt bekommen möchten oder lieber selbst generieren wollen, mit einer Authenticator-App auf dem Smartphone.

Die SMS-Variante ist einfacher, aber auch unsicherer, weil es Angreifern gelingen kann, die SMS-Nachrichten mit den Codes abzufangen. Dennoch ist 2FA per SMS besser als nichts. Wir empfehlen die sicherere Variante „Authentifizierungs-App“, die sie jedoch nur mit der Instagram-App aktivieren können, nicht über die Website. Anschließend empfiehlt Ihnen Instagram geeignete Authenticator-Apps wie die von Google und erklärt Ihnen, wie Sie diese mit Ihrem Instagram-Account verknüpfen. Darüber hinaus sollten Sie ein langes Passwort für Ihren Account wählen, das nicht zu erraten ist und nur bei Instagram passt. Im besten Fall nutzen Sie einen Passwortmanager, um ein langes Zufallspasswort zu generieren und zu speichern.



Mit dem Sicherheits-Check überprüfen Sie die wichtigsten Security-Einstellungen bei Instagram.

Sicherheits-Check

Hilfreich ist der „Sicherheits-Check“, den Sie ebenfalls über die Sicherheitseinstellungen in der Instagram-App starten können. Diese Funktion macht auf gängige Sicherheitsprobleme wie ein schwaches Passwort aufmerksam und empfiehlt auch das Einschalten der 2FA, sofern sie nicht bereits aktiv ist. Zudem erinnert der Sicherheits-Check daran, dass man die Aktualität der hinterlegten Mailadresse und Telefonnummer kontrollieren sollte.

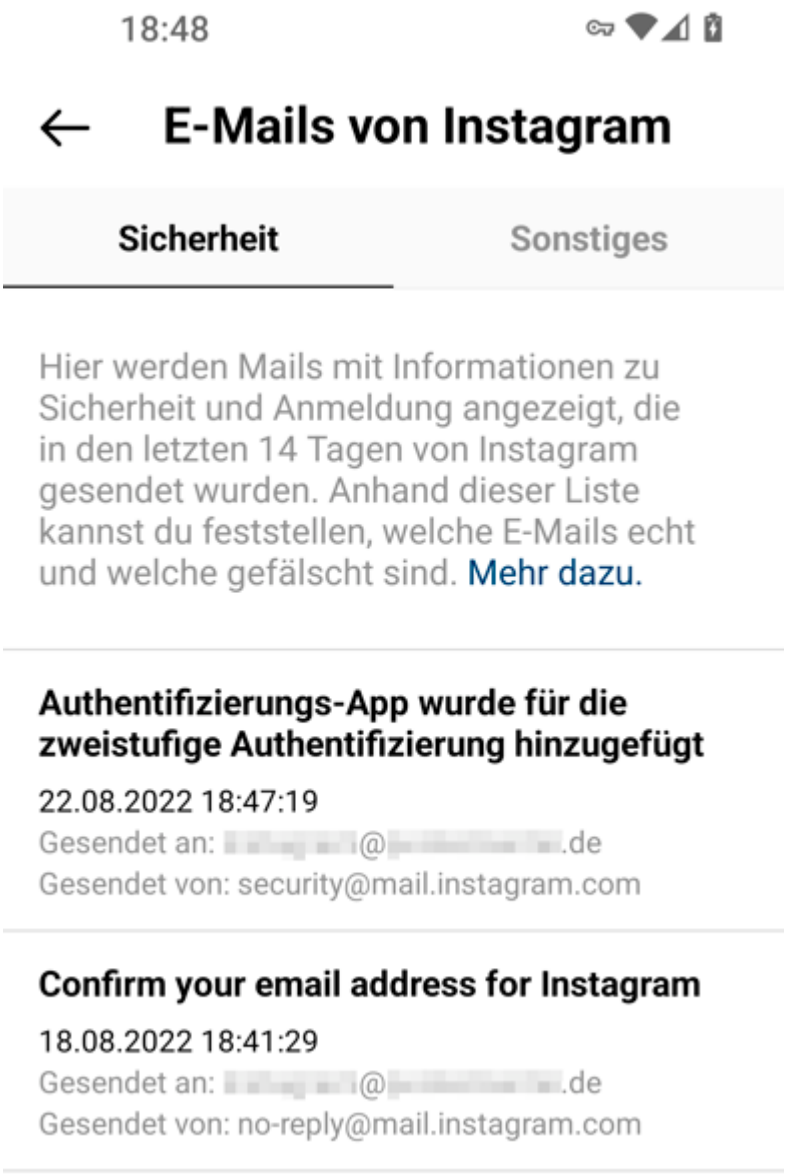
Wenn Sie Instagrams Betreiberfirma Meta diese Daten nicht anvertrauen möchten, funktionieren viele der Rettungsfunktionen von Instagram nicht, etwa weil das Unternehmen Ihnen im Fall der Fälle keinen Link zuschicken kann, über den Sie die Kontrolle über den gehackten Account zurückgewinnen können. Keine ganz leichte Abwägung, eventuell können Sie Instagram eine Zweit- oder Drittmailadresse zur Verfügung stellen – Hauptsache, Sie haben im Notfall sicher Zugriff darauf. Auch ein Profilfoto, auf dem Sie gut zu erkennen sind, kann die Rettung des Accounts erleichtern. Dazu gleich mehr.

Anti-Social-Engineering

Auch wenn Sie Ihren Account mit allen zur Verfügung stehenden Mitteln abgesichert haben: Technische Schutzmaßnahmen können Social Engineering nur erschweren, nicht verhindern. Angreifer hacken nicht Ihr Smartphone, sondern locken Sie trickreich in die Falle, etwa indem sie sich eben als Instagram-Support ausgeben und Sie mit einer plausibel klingenden Geschichte auffordern, Ihre Zugangsdaten auf einer externen Website einzugeben. Der zweite Faktor erschwert zwar einen solchen Phishing-Angriff, doch in jüngster Zeit fragen Online-Ganoven immer wieder auch nach dem temporären Einmalcode, mit dem sie den Account schließlich übernehmen können.

Allerdings können Sie sich vor dieser Form des Social

Engineering leicht schützen. Zunächst einmal sollten Sie sich darüber im Klaren sein, dass Sie Instagram niemals per Direktnachricht (Direct Message, DM) kontaktieren wird. Bei DMs ist Vorsicht geboten, auch wenn Sie den Absender kennen: Wurde ein Account gehackt, nehmen Angreifer schon mal Kontakt mit Freunden und Followern des Opfers auf, meist um die dazu zu bringen, eine gefährliche Website zu besuchen.



In der Instagram-App können Sie überprüfen, ob eine Mail, die angeblich von Instagram stammt, tatsächlich echt ist.

Mailcheck

Instagram kontaktiert Sie ausschließlich per Mail. Das wissen

allerdings auch die Cyber-Ganoven, sie verschicken täuschend echt aussehende Phishing-Mails im Instagram-Look. Wenn Sie eine Mail bekommen, die von Instagram stammen soll, sollten Sie sich also zunächst von der Echtheit überzeugen, bevor Sie die Mail ernst nehmen und auf einen Link aus der Nachricht klicken. Das ist bei Instagram erfreulich einfach: Öffnen Sie die Einstellungen in der App und tippen Sie auf „Sicherheit/E-Mails von Instagram“.

Dort listet die App alle Nachrichten auf, die Ihnen Instagram in den vergangenen 14 Tagen per Mail geschickt hat. Sie können die Nachrichten dort zwar nicht lesen, aber Sie erfahren Absender, Betreff und Sendedatum. Gleichen Sie diese Daten mit der Mail ab, um die Echtheit der Mail zu verifizieren. Der Absender sicherheitsrelevanter Instagram-Mails lautet stets security@mail.instagram.com. Wenn Sie auf Nummer sicher gehen wollen, dass der angegebene Absender nicht gefälscht ist, können Sie den Mail-Header inspizieren, wie in ct 19/2022 beschrieben [1].

Gehackten Account retten

Ist das Kind bereits in den Brunnen gefallen und Ihr Account wurde gehackt, dann müssen Sie schnell handeln. Je früher Sie aktiv werden, desto mehr Schaden können Sie abwenden. Nutzen Sie für sämtliche Rettungsversuche am besten ein Gerät, mit dem Sie bereits zuvor bei Instagram eingeloggt waren.

Beachten Sie die Mails von Instagram, um frühzeitig von einer Account-Übernahme zu erfahren. Der Dienst wird Sie über den Fremdlogin per Mail informieren und liefert Ihnen nicht nur den Zeitpunkt des Logins, Sie erfahren auch, welches Betriebssystem und welcher Browser mutmaßlich zum Einsatz kam. Zudem führt Instagram das Land an, aus dem die IP-Adresse des Nutzers stammt.

Auch wenn diese Daten nicht zu einhundert Prozent verlässlich sind: Sie eigenen sich gut, um darin Abweichungen zu Ihren

bisherigen Anmeldungen zu erkennen. Falls Ihnen bei der Kontrolle der Loginaktivität etwas komisch vorkommt, können Sie Ihren Account über den Link in der Mail („Sichere dein Konto hier“ oder „Secure your account here“) absichern. Achten Sie darauf, dass Sie auch tatsächlich auf https://www.instagram.com landen und nicht auf einer Phishing-Seite. Sie können über den Link ein neues Passwort setzen, das der Hacker nicht kennt. Überprüfen Sie von Zeit zu Zeit auch die „Login-Aktivität“ in den Sicherheitseinstellungen der App.

Informiert Sie Instagram ohne Ihr Zutun, dass Ihr Passwort oder die mit dem Account verknüpfte Mailadresse geändert wurde, sollten bei Ihnen die Alarmglocken läuten. Mit etwas Glück im Unglück können Sie aber auch in diesen Situationen die Kontrolle zurückgewinnen und die Änderung rückgängig machen, indem Sie in der Benachrichtigungsmail auf den Link „Sichere dein Konto hier“ klicken. Anschließend können Sie ein neues Passwort festlegen. Aber aufgepasst: Kontrollieren Sie auch in solch eiligen Fällen den Absender der Mail und das Ziel des Links genau, um sicherzustellen, dass es sich nicht um eine Phishing-Mail handelt. Geben Sie auf der verlinkten Seite nicht Ihr altes Instagram-Passwort ein.

Wurde der Account übernommen, kann ein Video-Selfie der letzte Ausweg sein.

Versteckter Rettungsweg

Für Härtefälle gibt es noch einen weiteren Rettungsweg über den Instagram-Support, der allerdings gut versteckt ist. Sie

erreichen ihn über die Instagram-App, indem Sie unterhalb des Login-Formulars auf „Erhalte Hilfe bei der Anmeldung“ tippen. Geben Sie oben Ihren Nutzernamen an und tippen Sie anschließend darunter auf „Du kannst dein Passwort nicht zurücksetzen?“. Die App fragt Sie daraufhin „Hast Du ein Foto von dir selbst in deinem Konto?“ – und das aus gutem Grund. Das Foto benötigt der Instagram-Support, um zu überprüfen, ob Sie der legitime Accountbesitzer sind. Falls Sie die Frage mit „Nein“ beantworten, ist Ihre Reise an dieser Stelle zu Ende und Sie landen im Hilfebereich.

Wenn Sie hingegen ein Foto in Ihrem Account haben und mit „Ja“ antworten, geht es weiter im Programm. Der genaue Ablauf variiert von Fall zu Fall. Instagram könnte Sie nach einem alten Passwort fragen und im darauffolgenden Schritt nach einem Bestätigungscode, den Sie sich an eine bei Instagram hinterlegte Mailadresse oder Handynummer schicken lassen können. Selbst wenn der Phisher die hinterlegten Daten geändert hat, stehen die Chancen gut, dass Sie hier noch Ihre wahre Rufnummer oder Mailadresse auswählen können und so an den Code kommen. Nach der Eingabe des Bestätigungscodes fragt Sie die App nach einer Mailadresse, über die Sie der Instagram-Support erreichen kann.

Video-Selfie

Haben Sie schließlich alle Hürden genommen, geht es ans Eingemachte: Die Instagram-App fordert Sie auf, Ihr Gesicht für ein sogenanntes Video-Selfie zu filmen. Im Rahmen dieses Vorgangs müssen Sie Ihren Kopf in vorgegebene Richtungen bewegen, um zu beweisen, dass Sie echt sind. Danach laden Sie das Video über den blauen „Senden“-Knopf hoch. Instagram beteuert, dass dieses Video maximal 30 Tage gespeichert wird und nicht zur Gesichtserkennung oder Speicherung biometrischer Merkmale genutzt wird. Wenn Sie das Video-Selfie hochgeladen haben, heißt es warten. Der Instagram-Support nimmt sich bis zu zwei Tage Zeit, um Ihr Anliegen zu bearbeiten.

Normalerweise geht es aber schneller. Nach der Überprüfung sendet Ihnen Instagram einen Link an die zuvor eingegebene Mailadresse, über den Sie ein neues Passwort festlegen können.

Falls Sie Ihren Facebook-Account mit Instagram verknüpft haben, gelten für diesen die gleichen Tipps: Nutzen Sie ein starkes Passwort, das nur bei Facebook passt, aktivieren Sie die Zwei-Faktor-Authentifizierung und achten Sie darauf, dass Ihre Kontaktdaten aktuell sind. Sie können zusätzlich die 2FA bei Instagram aktivieren, damit ein Angreifer, der bereits Kontrolle über Ihren Facebook-Account hat, nicht auch noch auf Ihr Instagram-Profil zugreifen kann.

Fazit

Instagram-Accounts stehen bei Cyber-Ganoven hoch im Kurs – insbesondere, aber nicht nur, wenn der begehrte blaue Haken das Profil ziert. Es ist daher wichtig, die Maschen der Angreifer zu kennen und frühzeitig geeignete Schutzmaßnahmen zu treffen. Wer sich nicht kümmert, riskiert sowohl, dass der Account gehackt wird, als auch, dass die vorhandenen Rettungsfunktionen ins Leere laufen, über die man die Kontrolle über einen gehackten Account zurückgewinnen könnte. *(rei@ct.de)*

1. Literatur
2. Ronald Eikenberg, E-Mails durchleuchtet, Phishing-Mails erkennen und abwehren, c’t 19/2022, S. 18
3. Niklas Dierking, Ronald Eikenberg, Schlosskombination, Verfahren und Geräte für sichere Online-Zugänge, c’t 9/2022, S. 18

Instagram-Hilfe zur Absicherung: ct.de/yqn6

Fehler bei Hostern gefährden die Sicherheit von DKIM

[expand title="mehr lesen…"]

Fehler bei Hostern gefährden die Sicherheit von DKIM

Wissen Konfigurationsfehler bei DKIM

Bild: Thorsten Hübner

DKIM-Fail

Fehler bei Hostern gefährden die Sicherheit von DKIM

Online-Kriminelle versenden regelmäßig E-Mails unter falschem Namen, um Nutzer zur Herausgabe von sensiblen Daten zu bewegen. Mit DKIM sind Spam-Filter in der Lage, solche gefälschten Mails zu erkennen. Doch unsere Analysen zeigen, dass einige Webhoster mit Fehlkonfigurationen Spammern und Phishern Tür und Tor öffnen. Von Leo Dessani und Jan Mahn

Eine neue E-Mail vom Chef. Laut Mailprogramm stammt sie auch von seiner Adresse. Offenbar steckt er im Ausland in Schwierigkeiten, hat seine Kreditkarte verloren und braucht schnell etwas Geld vom Firmenkonto. Was auf den ersten Blick wie eine authentische E-Mail aussieht, kann sich beim zweiten Blick als Phishing-Versuch offenbaren. Ist die gefälschte Mail gut gemacht, kann sie Filter wie SpamAssassin mit einiger Wahrscheinlichkeit umgehen und landet direkt im Posteingang des Nutzers. Aber selbst wenn der Nutzer vorsichtig ist und die E-Mail-Adresse des Absenders beim Öffnen gewissenhaft prüft, ist das keine Garantie, dass die Nachricht auch tatsächlich von dieser Adresse stammt.

Kriminelle verfolgen mit Phishing-Mails ein konkretes Ziel: das Vertrauen der Nutzer zu gewinnen und sie zu animieren, vertrauliche Daten wie Passwörter preiszugeben (Social Engineering). Senden die Täter ihre Phishing-Mails von einer echten E-Mail-Adresse einer Organisation, auf die sie selbst keinen Zugriff haben, gewinnen sie potenziell mehr Vertrauen der Nutzer, denn vielen Anwendern ist nicht bewusst, dass man Absenderadressen leicht fälschen kann. Möglich ist das durch eine konzeptionelle Schwachstelle im SMTP-Protokoll: Einen Mechanismus für die Authentifizierung der Absenderadresse gibt es im Protokoll selbst nicht.

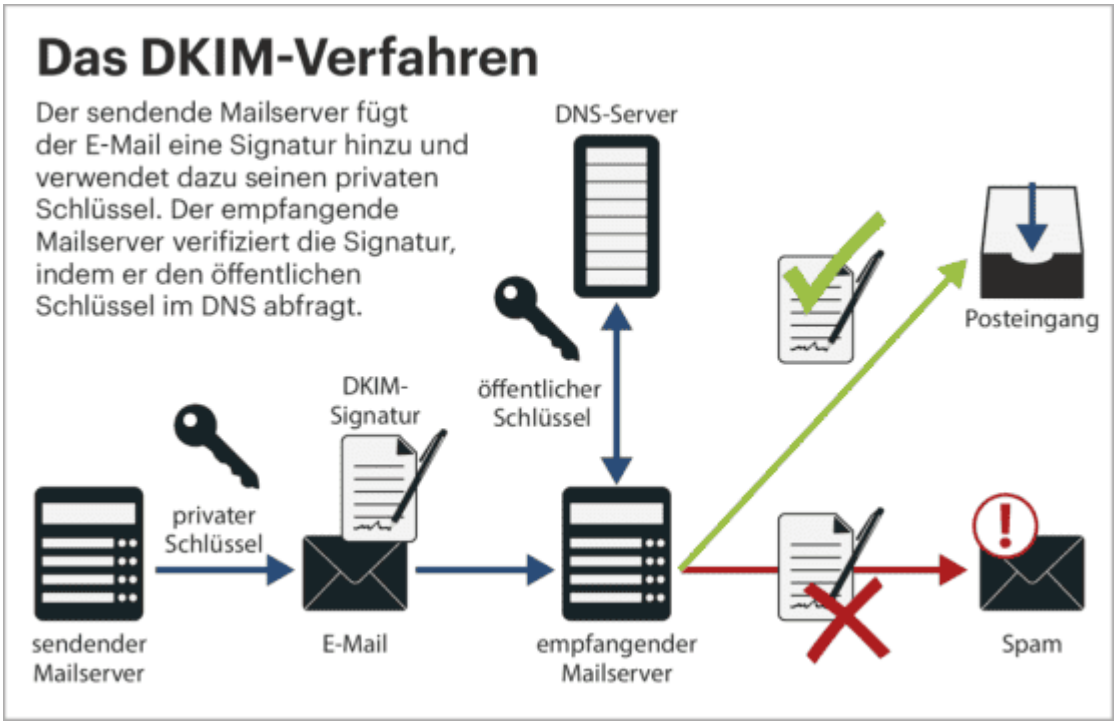
Bereits 2004 haben sich Yahoo und Cisco zusammengeschlossen

und gemeinsam einen Standard konzipiert, der das Problem lösen soll: „DomainKeys Identified Mail“ (DKIM). Seit 2011 ist DKIM als Internetstandard von der Internet Engineering Task Force (IETF) anerkannt und wird von vielen Mailserverbetreibern eingesetzt. Das Fälschen von Absenderadressen (Mail-Spoofing) soll dadurch erschwert werden, dass jeder ausgehenden E-Mail eine digitale Signatur als Mail-Header beigefügt wird. Die Signatur im Header kann vom empfangenden Mailserver validiert werden. Mails mit gefälschter Absenderadresse können so erkannt und markiert oder entsorgt werden. Wie DKIM im Detail funktioniert, erfahren Sie im Kasten rechts.

DKIM: Mit Signaturen gegen Betrüger

DKIM ist ein Standard, um die Echtheit der versendenden Domain einer E-Mail zu prüfen. Anders als zum Beispiel PGP ist für DKIM der Betreiber des Mailservers verantwortlich – als Nutzer kann man das Verfahren nicht einrichten. Ein Serverbetreiber, der DKIM-Signaturen an seine Mails anhängen möchte, generiert ein Schlüsselpaar aus öffentlichem und privatem Schlüssel. Um den öffentlichen Schlüssel bekannt zu machen, kommt DNS zum Einsatz: Den öffentlichen Schlüssel legt der Administrator als TXT-Record in der DNS-Zone seiner Domain ab. Der private Schlüssel darf den Mailserver nicht verlassen.

Beim Versenden von Mails werden zwei Prüfsummen berechnet: eine für ausgewählte Teile des Headers, eine für den Body der Mail. Die Prüfsummen werden mit dem privaten Schlüssel per RSA signiert und als Mailheader DKIM-Signature der E-Mail beigefügt, ergänzt um weitere Informationen. Zu denen zählen unter anderem die Absender-Domain, die Namen aller signierten Header-Felder sowie der sogenannte Selektor. Der Selektor entspricht dem Namen des DNS-Eintrags, in dem der öffentliche Schlüssel liegt. Die Liste der mitsignierten Header-Felder muss mindestens das Feld From: enthalten, also die Absenderadresse, die auch dem Empfänger angezeigt wird. So ist sichergestellt, dass nachträgliche Manipulationen die Signatur ungültig machen.



Empfängt ein Mailserver eine digital signierte E-Mail und ist der Server so eingerichtet, dass er DKIM prüft, fragt er aus dem DNS für die angegebene Domain den öffentlichen Schlüssel mit dem Namen des Selektors ab. Mit dem öffentlichen Schlüssel kann er die Echtheit der digitalen Signatur bestimmen. Ist die Prüfung erfolgreich, ist gewährleistet, dass die E-Mail von einem authentischen Absender stammt und nicht verändert wurde. Schlägt sie fehl, kann das ein Indiz dafür sein, dass die E-Mail gefälscht ist. Was dann passiert, kann der Betreiber des empfangenden Servers bestimmen. Oft führt das Scheitern zur sofortigen Ablehnung der E-Mail, manchmal wird sie nur als Spam-verdächtig markiert. Das Ergebnis der Prüfung fügt der empfangende Mailserver mit dem Header Authentication-Results an die Mail an. dkim=pass zeigt an, dass die Prüfung erfolgreich war, dkim=fail, dass sie fehlschlug.

Fast alle Mailserver verlassen sich nicht auf eine Methode zum Filtern allein und schalten mehrere Filter in Reihe. Mit Inhaltsfiltern reagieren sie zum Beispiel auf typische Spam-Begriffe wie „Casino“ und „Viagra“. In solchen Umgebungen vergibt jeder Filter einen Punktwert für die Einordnung der Mail – überschreitet die Summe aller Punkte einen Schwellwert, wird die Mail aussortiert oder markiert. Eine erfolgreiche

DKIM-Prüfung wirkt sich in vielen Konfigurationen positiv auf die Vertrauenswürdigkeit aus und zieht Punkte ab.

Geteilte Server

Seit einigen Jahren bieten immer mehr Webhosting-Anbieter ihren Kunden das Signieren von E-Mails mit DKIM an. Bei einigen Providern ist DKIM sogar standardmäßig für alle Domains aktiviert, bei anderen reicht ein Klick im Kundencenter, um DKIM für einzelne oder alle Domains zu aktivieren. Die Hoster machen es den Kunden leicht und übernehmen das Hantieren mit Schlüsseln und DNS-Einträgen. Ohne Zutun des Kunden erstellen sie ein Schlüsselpaar, legen den öffentlichen Schlüssel im DNS als TXT-Record ab und richten den privaten Schlüssel auf dem Mailserver ein. Fortan werden alle ausgehenden E-Mails automatisch mithilfe von DKIM signiert.

Bei Webhosting-Paketen sind sogenannte Shared Server verbreitet. Mehrere Kunden teilen sich einen Server, also dessen Ressourcen und Software. Dadurch kann der Anbieter mehr Kunden bedienen, als er tatsächlich physische Server vor Ort hat. Bei solchen Shared Servern muss gewährleistet sein, dass ein Kunde nicht auf die Daten eines anderen zugreifen kann. Für die Webseitendaten und Datenbanken funktioniert das auch sehr zuverlässig.

DMARC: Das Anti-Spam-Trio

Neben DKIM existieren zur Bekämpfung von Spam- und Phishing-Mails zwei weitere Verfahren: Sender Policy Framework (SPF) und Domain-based Message Authentication (DMARC).

SPF beruht auf der Annahme, dass alle E-Mails einer Domain von einer festen Anzahl von autorisierten Mailservern versendet werden. In einem TXT-Record veröffentlicht der Administrator die Adressen dieser Mailserver im DNS. Der Spam-Filter auf dem empfangenden Server kann bei der Entgegennahme der E-Mail

durch das Abrufen dieses DNS-Eintrages prüfen, ob der sendende Mailserver zum Verschicken berechtigt ist. Was geschieht, wenn eine E-Mail über einen nicht autorisierten Mailserver versendet wird, kann ebenfalls im DNS-Eintrag festgelegt werden.

DMARC ist keine eigene Technik, sondern kombiniert die Ergebnisse der SPF- und DKIM-Prüfungen: Mit DMARC beschreibt der Administrator, ebenfalls in Form eines DNS-Eintrages, wie der empfangende Mailserver mit einer E-Mail umgehen soll, bei der die SPF- oder DKIM-Prüfungen fehlschlagen, und wen er darüber informieren soll.

Signaturen für fremde Domains

Doch werden auch die privaten DKIM-Schlüssel verschiedener Kunden sauber getrennt? DKIM ist schließlich nur sinnvoll, wenn gewährleistet ist, dass niemand gefälschte Signaturen generieren kann. Was für den Schutz von Kundendaten aufShared Servern gilt, muss auch für Schlüsselpaare gelten: Gültige DKIM-Signaturen auf Grundlage des privaten Schlüssels dürfen ausschließlich für E-Mails generiert werden, die vom Inhaber einer Domain stammen und nicht etwa von anderen Kunden, deren Accounts zufällig auf demselben Server liegen.

Providervergleich

Um herauszufinden, ob Hosting-Anbieter die DKIM-Signaturen ihrer Kunden auf demselben Server sauber trennen, haben wir 37 deutsche Anbieter unter die Lupe genommen und angefragt, ob sie DKIM für ihre Kunden aufShared Servern bereitstellen. Die Antwort: 17 Provider bieten DKIM für ihre Kunden gar nicht an. Vier Provider stellen DKIM nur aufInstanzen bereit, die nicht mit anderen Kunden-Domains geteilt werden (zum Beispiel virtuelle Server oder Managed Server). Übrig blieben 16 Provider für unsere Tests.

DKIM-Konfigurationsfehler bei deutschen Webhostern				
Anbieter	**getestetes Paket**	**DKIM-Unterstützung**	**Ergebnis**	**Reaktion**
All-Inkl.com	Premium	automatisch aktiv	verwundbar	DKIM für die PHP-Mailfunktion deaktiviert
Contabo	Paket L	automatisch aktiv	nicht verwundbar	
creoline	WordPress Hosting S	manuell aktivierbar	verwundbar	Lücke geschlossen
Febas	Professional	manuell aktivierbar	Test nicht möglich[1]	
Hetzner	Level 4	manuell aktivierbar	verwundbar	Lücke geschlossen
hosting.de	Medium	automatisch aktiv	nicht verwundbar	
Hostinger	Premium	manuell aktivierbar	Test nicht möglich[1]	
netclusive	Easy 5.0	manuell aktivierbar	verwundbar	DKIM zunächst deaktiviert, Lücke später geschlossen
netcup	Webhosting 4000	automatisch aktiv	nicht verwundbar	
one.com	Entdecker	automatisch aktiv	nicht verwundbar	
Serverprofis	Private L 5.3	automatisch aktiv	nicht verwundbar	
Strato	Basic	automatisch aktiv	nicht verwundbar	
UD Media	Power 5.0	automatisch aktiv	verwundbar	Lücke geschlossen
webgo	SSD Profi	über den Support aktivierbar	teilweise verwundbar	
webhoster.de	Starter Tarif	manuell aktivierbar	nicht verwundbar	
WebhostOne	Basic	manuell aktivierbar	verwundbar	Lücke geschlossen
[1] keine anderen Kunden mit aktivem DKIM auf demselben Server				

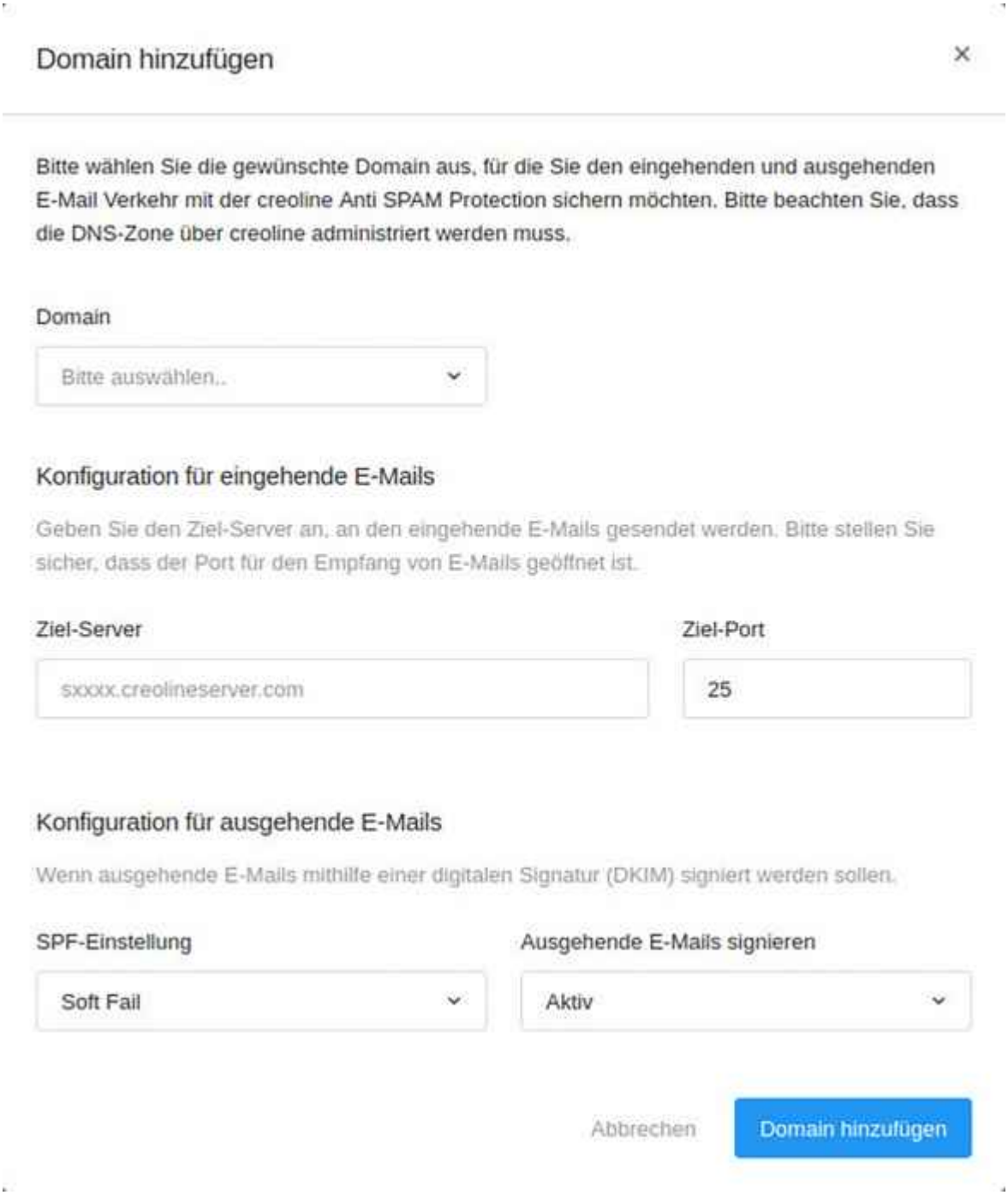
Bei All-Inkl.com, Contabo, hosting.de, netcup, one.com, Serverprofis, Strato und UD Media ist DKIM standardmäßig

aktiviert. Bei einigen Anbietern war es notwendig, DKIM im Kundeninterface einzuschalten. Für unseren Test suchten wir den DKIM-Selektor unserer Test-Domains über die DNS-Einstellungen des Kundenportals. Dann gingen wir auf die Suche nach fremden Domains von anderen Kunden, die sich mit uns einen Server teilten. Diese Recherche ist mit einer Reverse--DNS-Suchmaschine im Internet schnell erledigt, indem man nach der IP der eigenen Domain sucht. Für den Test brauchten wir eine fremde Domain, auf der ebenfalls DKIM aktiv war – ob das der Fall ist, findet man heraus, wenn man deren DNS-Einträge durchsucht. Bei den meisten Anbietern ging das schnell, da die DKIM-Selektoren für alle Domains identisch sind. All-Inkl.com, Hostinger und hosting.de vergeben individuelle DKIM-Selektoren auf Grundlage des Datums, an dem DKIM aktiviert wurde. In diesem Fall war etwas Ausdauer gefragt, da wir die fremden Domains manuell prüfen mussten. Nachdem wir fremde Domains mit aktivierter DKIM-Signatur auf „unseren“ Servern ausfindig gemacht hatten, konnte der Test beginnen.

Hoster ohne DKIM-Unterstützung	
Anbieter	DKIM-Unterstützung
1blu	nicht unterstützt
alfahosting	nicht unterstützt
centron	nicht unterstützt
checkdomain	nicht unterstützt
DM Solutions	nur für Managed Server
dogado	nicht unterstützt
DomainFactory	nicht unterstützt
ESTUGO	nicht unterstützt
goneo	nicht unterstützt
Host Europe	nicht unterstützt
Hostpress	nur für vServer
INWX	nicht unterstützt

Hoster ohne DKIM-Unterstützung	
Anbieter	DKIM-Unterstützung
IONOS 1&1	nicht unterstützt
manitu	nicht unterstützt
Mittwald	nicht unterstützt
OVH	nicht unterstützt
Packagecloud (D&T Internet)	nicht unterstützt
profihost	nicht unterstützt
Raidboxes	nur für vServer
TimmeHosting	nur für vServer
united-domains	nicht unterstützt

In allen getesteten Paketen stand uns PHP zur Verfügung – also nutzten wir die PHP-Funktion mail(), um eine E-Mail mit einer fremden Domain in der Absenderadresse, die auf demselben Server gehostet war wie unsere, an ein externes Postfach zu schicken. Eine glatte Fälschung also, die niemals hätte signiert werden dürfen.



Bei Creoline muss man DKIM im Kundencenter aktivieren. Der Anbieter war beim DKIM-Signaturdiebstahl verwundbar, konnte das Problem nach unserem Hinweis aber abstellen.
Bei sechs von sechzehn getesteten Anbietern war das Experiment erfolgreich: All-Inkl.com, creoline, Hetzner, netclusive, WebhostOne und UD Media hängten eine gültige Signatur mit dem privaten Schlüssel der fremden Domain an, obwohl wir zum Versenden nicht berechtigt waren. Unser empfangender Mailserver stufte die Mail als korrekt DKIM-signiert ein. Bei netclusive betraf dies nur Pakete auf dem Server hst1.ncsrv.de. Neue Pakete auf dem Server hst2.ncsrv.de waren

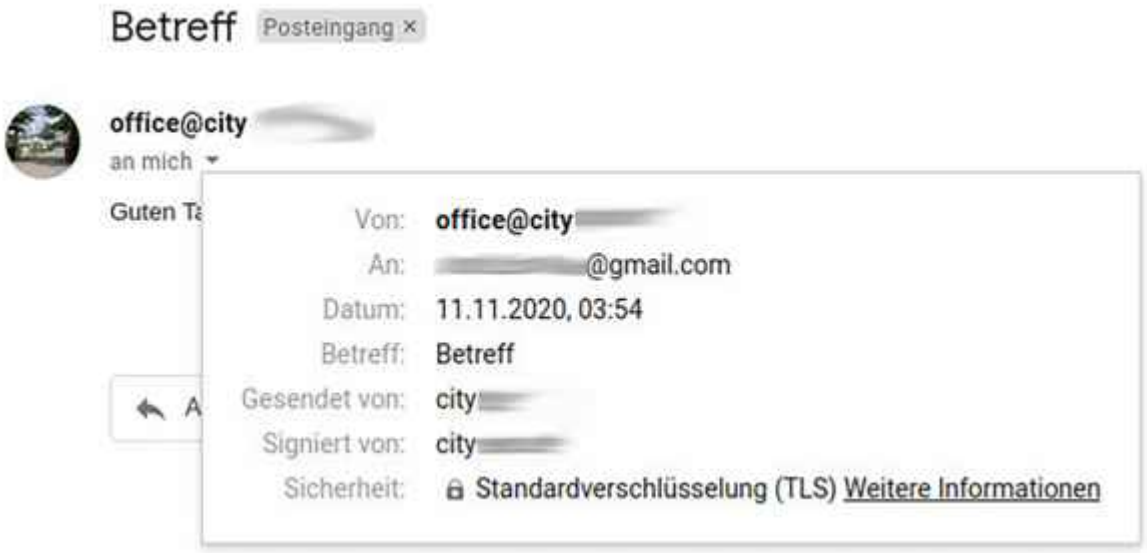
nicht betroffen.

Bei Febas, Hostinger und webgo konnten wir die Recherchen nicht abschließen, weil auf unserem Server keine anderen Domains DKIM aktiviert hatten und somit kein fremdes Schlüsselmaterial zum Testen vorhanden war.

Bei Serverprofis und Strato funktionierte der Angriffsversuch nicht. Beim Versenden aus unserem Account heraus wurde für die fremde Domain entweder eine DKIM-Signatur mit unserem privaten Schlüssel oder gar keine hinzugefügt. Zu einer unbefugt gültigen Signatur kam es nicht. Bei one.com wurden für fremde Adressen gar keine Mails verschickt, ein Angriff war also auch nicht möglich. Bei netcup und hosting.de konnten wir das Problem ebenfalls nicht reproduzieren. Dort werden Mails laut Auskunft des Supports nur dann DKIM-signiert, wenn man sie über den SMTP-Server verschickt und sich bei diesem authentifiziert. Das war hier ein wirkungsvoller Schutz gegen den Angriff.

Vertrauen verspielt

Unsere Untersuchung macht deutlich: Auch wenn das DKIM-Protokoll selbst gut konzipiert ist, haben es einige Webhoster durch fehlerhafte Konfiguration geschwächt. Bei den Anbietern, bei denen wir gefälschte E-Mails versenden konnten, haben wir die Wirksamkeit von DKIM ausgehebelt. Noch mehr: Da wir von einem autorisierten Mailserver verschickten, lieferten auch SPF und damit DMARC keine Fehler. Wir umgingen so auch vergleichsweise streng konfigurierte Spam-Filter und unsere E-Mail landete direkt im Posteingang ohne Spam-Verdacht. Auch sicherheitsbewusste Nutzer, die zum Beispiel mit dem Thunderbird-Plug-in„DKIM Verifier“ arbeiten, das bei jeder Mail das Ergebnis der Signaturprüfung prominent anzeigt, wären auf den Angriff hereingefallen.



Google Mail zeigt an, dass die Mail korrekt signiert wurde. Dabei wurde sie nicht von einem berechtigten Absender verschickt.
Für Spammer und Phisher ist dieser lockere Umgang mit den DKIM-Schlüsseln der Kunden ein großzügiges Angebot, gegen das Betreiber von Maileingangsservern und die Mailempfänger nichts tun können. Abhilfe schaffen können bei dem Problem nur die Hosting-Anbieter.

Nach unseren Experimenten kontaktierten wir die betroffenen Anbieter All-Inkl.com, creoline, Hetzner, netclusive, WebhostOne und UD Media und wiesen auf das Problem hin. Die Hoster, bei denen kein Test möglich war, wiesen wir darauf hin, dass das Problem möglicherweise auch bei ihnen besteht. Webgo bestätigte, dass die Lücke tatsächlich auf einigen Servern existiert – diese ältere Infrastruktur werde in nächster Zeit aktualisiert.

Creoline reagierte schnell mit einer Stellungnahme und wies zunächst darauf hin, dass Versuche, die Absenderadresse zu ändern, spätestens nach fünf Versuchen automatisch unterbunden wurden. Am nächsten Tag hatte man das Problem dann vollständig gelöst und die Manipulation war gar nicht mehr möglich. Netclusive antwortete einen Tag nach dem Hinweis, dass man DKIM vorübergehend ganz abgeschaltet habe, eine Woche später hatte man das Problem dann gelöst und DKIM wieder aktiviert.

Auch bei Hetzner konnte man das Problem bestätigen und stufte es als „mittelschwer“ ein – einen Tag nach der Meldung hatte man den Fehler beseitigt. Weil der Kunde DKIM selbst aktivieren muss, seien nach Angaben von Hetzner nur etwa fünf Prozent der Webhosting-Kunden betroffen gewesen. All-Inkl.com deaktivierte etwa eine Woche nach unserem Hinweis alle DKIM-Signaturen für Mails, die über die PHP-Funktion mail() verschickt wurden.

Private Schlüssel

Bei späteren Untersuchungen bemerkten wir, dass wir teilweise auch per SMTP Mails mit falscher Domain abliefern konnten, die dann signiert wurden – das Problem war bei einigen Anbietern also nicht auf die mail()-Funktion von PHP beschränkt. Die Lücke zeigte wieder ein altbekanntes Problem. DKIM basiert auf asymmetrischer Kryptografie, es gibt also einen öffentlichen und einen privaten Schlüssel. Wirklich sicher sind solche Verfahren nur, wenn der private Schlüssel auch wirklich privat bleibt. Also am besten auf einer Maschine, auf die nur der Inhaber selbst Zugriff hat. Wer DKIM bei einem Shared-Hosting-Dienst nutzt, gewinnt zwar viel Komfort, gibt aber seinen privaten Schlüssel aus der Hand und muss dem Dienstleister vertrauen. (*jam@ct.de*)

[/expand]