Automatisierte Erkennung von Daten-Exfiltration

Nils Rogmann

Hochschule Darmstadt | Controlware GmbH
nils.rogmann@controlware.de

Zusammenfassung

Weltweit werden trotz des Einsatzes von aktueller IT-Sicherheitsinfrastruktur und moderner Schutzmaßnahmen immer häufiger Sicherheitsvorfälle mit dem Ziel eines Datendiebstahls beobachtet. Für die Exfiltration von sensiblen Informationen kommen zunehmend fortgeschrittene Exfiltrationstechniken zum Einsatz, die von etablierten Sicherheitslösungen bisher nicht zuverlässig detektiert werden können. Eine stetig wachsende Herausforderung stellt insbesondere die Verwendung steganographischer Exfiltrationstechniken dar. So exfiltrieren verschiedene Malware-Varianten bereits heute Daten mittels maliziöser DNS- oder ICMP-Kommunikation. Mit dieser wissenschaftlichen Arbeit wird ein neuartiges Verfahren zur zuverlässigen Erkennung von netzwerk-steganographischen Daten-Exfiltrationen vorgestellt. Dieses basiert im Wesentlichen auf statistischer Analyse und Methoden des überwachten maschinellen Lernens. Unter Verwendung des entwickelten Verfahrens kann zunächst in einer Lernphase mithilfe von bekannten Mustern bestehend aus extrahierten Merkmalen von legitimer Kommunikation ein Modell erzeugt werden. Dieses erlaubt dem speziell für diese Arbeit angepassten einklassigen naiven Bayes-Klassifikator innerhalb einer Arbeitsphase die Erkennung von unbekannten Mustern einer DNS- oder ICMP-basierten Daten-Exfiltration.

1 Einleitung

Weltweit kommt es trotz des Betriebs von aktueller IT-Sicherheitsinfrastruktur und der Etablierung verschiedener Schutzmaßnahmen immer häufiger zu Sicherheitsvorfällen wie beispielsweise Systemeinbrüchen [Bund16], [PwC15]. Ziel dieser Einbrüche ist es mitunter, die Systeme innerhalb eines Netzwerks mit Schadsoftware zu infizieren sowie kritische Unternehmensdaten auf den Systemen zu suchen und über das Internet zu stehlen. Bereits während der ersten sechs Monate des Jahres 2016 wurden basierend auf den Hochrechnungen der IT-Sicherheitsfirma *Risk Based Security* als Folge von weltweit ungefähr 1.800 öffentlich gewordenen Sicherheitsvorfällen über 1,1 Milliarden Datensätze unbemerkt von mehreren Organisationen entwendet. Hiervon sind alleine 957 Millionen auf Systemeinbrüche und den Einsatz von Schadsoftware zurückzuführen [Risk16]. Bei den gestohlenen Informationen handelte es sich überwiegend um E-Mail-Adressen und Passwörter sowie Namen, Anschriften und Kreditkartendaten.

Eine stetig wachsende Herausforderung stellt insbesondere die Verwendung steganographischer Exfiltrationstechniken dar. So exfiltrieren verschiedene Malware-Varianten bereits heute unbemerkt Daten mittels maliziöser DNS- oder ICMP-Kommunikation. Ein einschlägiges Beispiel für die bisher unzureichenden Möglichkeiten einer zuverlässigen Detektion bietet die Mal-

P. Schartner · N. Pohlmann (Hrsg.) · D•A•CH Security 2018 · syssec (2018) 117-130.

ware *FrameworkPOS*. Diese hat im Jahr 2014 in einem Zeitraum von mehreren Monaten unbemerkt 56 Millionen Kreditkartendaten des amerikanischen Unternehmens *The Home Depot* über das DNS-Protokoll exfiltriert [GDAT14]. Neben den auf Datendiebstahl spezialisierten Malware-Familien existieren außerdem frei verfügbare sowie ohne besonderes Expertenwissen einsetzbare Software-Werkzeuge. Diese bieten auch der Allgemeinheit verschiedene Möglichkeiten zur verdeckten, bisher nicht zuverlässig detektierbaren Daten-Exfiltration.

Ein vielversprechender Lösungsvorschlag, der im Zuge einer vorangegangenen Masterarbeit im Detail herausgearbeitet wurde und in dieser Ausarbeitung zusammenfassend dargestellt wird, besteht in der Kombination von *statistischer Analyse* und *maschinellem Lernen* [Rogm17]. Diese Kombination soll es ermöglichen, die verhaltensbasierten Muster einer legitimen Kommunikation innerhalb von dedizierten Netzwerken durch die Extraktion statistischer Merkmale automatisiert zu erlernen. Hierdurch wird im Wesentlichen die Realisierung einer automatisierten Unterscheidung des legitimen Netzwerkverkehrs von potentiell unbekannten Mustern einer Daten-Exfiltration angestrebt.

Nachfolgend wird einleitend eine Übersicht der zurzeit weltweit beobachteten Techniken zur Daten-Exfiltration gegeben. Im Zuge dessen erfolgt eine Einschätzung hinsichtlich der aktuellen Möglichkeiten und Grenzen zur Detektion bzw. Prävention dieser Exfiltrationstechniken. Daran anknüpfend werden am Beispiel des DNS-Protokolls insbesondere die aktuellen Herausforderungen zur zuverlässigen Erkennung von Netzwerk-Steganographie herausgearbeitet sowie vor dem aktuellen Stand der Technik und Wissenschaft diskutiert. Im nächsten Schritt gilt es, das grundlegende Konzept und die Realisierung des neuartigen Erkennungsverfahrens basierend auf statistischer Analyse und maschinellem Lernen vorzustellen. Abgeschlossen wird diese wissenschaftliche Ausarbeitung durch ein Fazit sowie einen Ausblick hinsichtlich möglicher Folgearbeiten.

2 Daten-Exfiltration

Bei einer Daten-Exfiltration handelt es sich im Allgemeinen um einen nicht autorisierten und von einem Angreifer durchgeführten Datentransfer zwischen zwei oder mehreren Systemen. In der Fachliteratur erfolgt in diesem Kontext typischerweise eine Klassifizierung zwischen offenen und verdeckten Kanälen, welche einem Angreifer unter Anwendung verschiedener Techniken einen Datendiebstahl ermöglichen [Secu14]. Bezugnehmend auf eine Illustration innerhalb einer Forschungsarbeit der Lancaster Universität – jedoch zur Vereinheitlichung abstrahiert und neu strukturiert – sind in Abbildung 1 zur Übersicht verschiedene häufig von Angreifern zur Exfiltration verwendete Kanäle dargestellt [Secu14]:

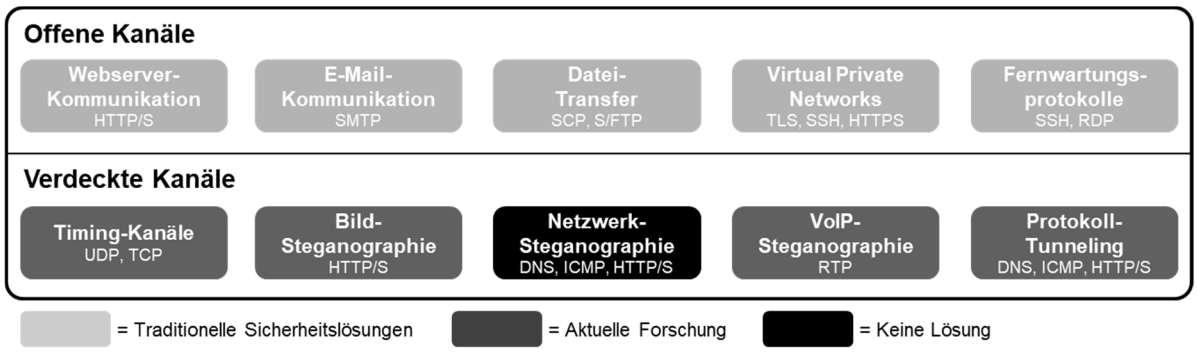

Abb. 1: Aktueller Stand der Überwachung von offenen und verdeckten Kanälen, [Rogm17]

Bei offenen Kanälen (engl. *overt channels*) handelt es sich – auch im Hinblick auf vorhandene Sicherheitsrichtlinien – um legitime Kommunikationswege und Protokolle, die innerhalb eines Systems oder Netzwerks von einer Applikation verwendet werden [EC-C09]. Diese ermöglichen einem Angreifer im Allgemeinen eine Exfiltration mit hoher Bandbreite, lassen sich jedoch in der Regel mit traditionellen Sicherheitslösungen überwachen oder durch präventive Maßnahmen einschränken.

Verdeckte Kanäle (engl. *covert channels*) werden im Allgemeinen von Angreifern zur unbemerkten – und innerhalb von legitimer Kommunikation versteckten – Datenübertragung eingesetzt. Während der Grad der Verborgenheit einer Exfiltration durch Mitbenutzung oder Zweckentfremdung eines zulässigen Netzwerkprotokolls im Wesentlichen erhöht wird, erfolgt hierdurch im Gegensatz zur Verwendung von offenen Kommunikationswegen eine Einschränkung der zur Verfügung stehenden Bandbreite [Secu14]. Dies ist damit zu begründen, dass die im Zuge einer verdeckten Exfiltration herangezogenen Protokolle typischerweise nur kleine Felder für Nutzdaten (Payload) haben und sich nicht für den Transfer großer Datenmengen eignen.

Die in der Abbildung dargestellten *offenen* Kanäle lassen sich im Allgemeinen bereits durch traditionelle Sicherheitslösungen überwachen und grundlegend einschränken. Diese Lösungen umfassen unter anderem Antivirus-Software, Firewalls, Proxy-Server sowie Systeme zur Intrusion- oder Breach-Detection. Weiterhin existieren speziell zur Erkennung und Vermeidung von Datendiebstählen entwickelte Data Leakage Prevention-Systeme (DLP).

Zur Erkennung oder Unterbindung einer fortgeschrittenen Daten-Exfiltration über *verdeckte* Kanäle sind traditionelle Sicherheitslösungen hingegen in der Regel nicht ausreichend. Beispielhaft kann mittels einer Firewall im Wesentlichen keine Exfiltration über DNS-Pakete oder HTTP-Verbindungen unterbunden werden, da eine Sperrung dieser Protokolle im Allgemeinen zu einer massiven Beeinträchtigung der täglichen Arbeit führen würde. Werden die Daten vor dem Transport verschlüsselt oder enkodiert, ist ferner eine Erkennung durch eine auf Regeln basierende DLP-Lösung meist technisch nicht mehr realisierbar [Mogu09]. Weiterhin kann eine auf Basis von Mustern und statischen Regeln funktionierende Intrusion Detection durch unbekannte oder modifizierte Kommunikationsmuster – zum Beispiel durch eine Variation von Paketgrößen oder der Reduzierung von Sendeintervallen zwischen Paketen – umgangen und eine Exfiltration somit unbemerkt durchgeführt werden [Extr16]. Auch eine Detektion von Timing-Kanälen kann aktuell typischerweise nicht von IDS- oder BDS-Lösungen geleistet werden [GiBC06].

Vielversprechender sind hingegen einige Erkennungsansätze aus der aktuellen Forschung. Als ein neuartiger Ansatz zur Detektion von Timing-Kanälen wurde im Oktober 2014 auf der Usenix-Konferenz die *Time-Deterministic Replay*-Technik (TDR) vorgestellt [CMX+14]. Weiterhin existieren Forschungsarbeiten, die der Identifizierung von TCP/IP-basierter Exfiltration mittels Timing-Kanälen gewidmet sind [LuCC08]. Auch die aktuellen Techniken zur Erkennung von Bild-Steganographie wurden im Zuge einer wissenschaftlichen Arbeit vorgestellt und in einer neuartigen Software kombiniert [Boeh14].

Ebenso erfolgte in einer wissenschaftlichen Untersuchung aus dem Jahr 2013 eine Klassifizierung etablierter Verfahren zur Daten-Exfiltration mittels VoIP-Steganographie [Mazu13]. Nicht zuletzt sind auch hinsichtlich verschiedener Techniken des Protokoll-Tunnelings einschlägige Veröffentlichungen zu finden. Unter anderem hat das Sysadmin, Networking and Security-Institut (SANS) eine Zusammenfassung verschiedener statistischer Metriken, welche

zur Detektion von DNS-Tunneling nutzbar sind, veröffentlicht [FaAt13]. Darüber hinaus existiert ein vielversprechender Ansatz, der durch ein *statistisches Fingerprinting* typischer Protokolle getunnelte Verbindungen identifizieren soll [DCG+09].

Im Bereich der Netzwerk-Steganographie sind in der Vergangenheit mehrere wissenschaftliche Arbeiten mit Fokus auf der Erkennung von versteckten Informationen innerhalb von verschiedenen Protokoll-Headern veröffentlicht worden. In diesem Kontext wurde unter anderem der Einsatz von *Active Wardens*, welche im Wesentlichen eine Deep Packet Inspection (DPI) zur Identifizierung von Header-Anomalien durchführen sollen, diskutiert [MuLe05]. Weiterhin wurde im Jahr 2012 eine wissenschaftliche Ausarbeitung veröffentlicht, die sich der Erkennung von steganographischen Verfahren zur Daten-Exfiltration mittels Internet Control Message Protocol (ICMP) widmet [Dinc12]. Als Möglichkeit zur Detektion erfolgte hierbei jedoch ausschließlich der Verweis auf den Einsatz von IDS-Lösungen, welche in diesem Kontext nur eingeschränkt zur Erkennung herangezogen werden können. Somit ist die Frage nach Möglichkeiten einer zuverlässigen Identifizierung von ICMP-Steganographie offen gelassen worden. Auch hinsichtlich des verdeckten Datendiebstahls mittels DNS-Steganographie findet lediglich eine Diskussion über Exfiltrationstechniken statt, während im Wesentlichen keine Möglichkeiten zur Detektion aufgezeigt werden [ANO+11], [DrSU16]. In letzterer der beiden genannten Quellen wird ferner explizit darauf hingewiesen, dass aktuell ohne dedizierte Filter oder der Entwicklung einer speziellen Erkennungssoftware eine zuverlässige Identifizierung dieser fortgeschrittenen Techniken nahezu unmöglich ist [DrSU16].

Daher wurde mit Fokus auf DNS und ICMP nach einem neuartigen, auf statistischer Analyse und maschinellem Lernen basierenden Verfahren zur Detektion dieser fortgeschrittenen Datendiebstähle unter Einsatz von netzwerk-steganographischen Verfahren geforscht. Die Kombination dieser beiden wissenschaftlichen Themengebiete soll dazu dienen, die verhaltensbasierten Muster einer legitimen Kommunikation innerhalb eines dedizierten Netzwerks automatisiert zu erlernen. Auf diese Weise soll eine zuverlässige Unterscheidung des legitimen Netzwerkverkehrs von einschlägigen Mustern einer Daten-Exfiltration mittels Netzwerk-Steganographie ermöglicht werden.

3 Netzwerk-Steganographie

Innerhalb dieses Kapitels erfolgt die exemplarische Betrachtung eines generischen Ablaufs und der technischen Umsetzung einer in der realen Welt beobachteten Exfiltrationstechnik unter Verwendung von DNS-Steganographie. Weitere einschlägige Beispiele für fortgeschrittene netzwerk-steganographische Daten-Exfiltrationen auf Basis von DNS und ICMP sind in [Rogm14] einzusehen.

Die Internet Assigned Number Authority (IANA) gibt vor, dass für jede Webseite mindestens zwei autoritative Nameserver zur Auflösung von Domain-Anfragen in IP-Adressen vorhanden sein müssen. Diese autoritativen Systeme dienen im Wesentlichen dazu, die von nicht-autoritativen, rekursiven Nameservern zwecks Auflösung weitergeleiteten DNS-Anfragen verbindlich zu beantworten. Die Weiterleitung der Anfragen erfolgt grundsätzlich, wenn die aufzulösenden Domains den nicht-autoritativen Servern entweder unbekannt sind oder die Gültigkeit der Antworten bereits abgelaufen ist. Dieser im DNS-Protokoll etablierte Mechanismus kann zum verdeckten Datendiebstahl missbraucht werden. Hierzu ist lediglich die Kontrolle über die für eine Domain verantwortlichen autoritativen Nameserver notwendig: Enkodiert ein Angreifer die Daten in eine für den zuständigen rekursiven Nameserver unbekannte DNS-Anfrage,

erreicht diese Nachricht zur verbindlichen Auflösung aufgrund der Beschaffenheit des Protokolls automatisch einen zuständigen autoritativen Nameserver. Wird dieser Server von dem Angreifer kontrolliert, ist an dieser Stelle die Dekodierung der in der Anfrage übertragenen Daten möglich.

In Abbildung 2 wird am Beispiel der Domain *malicious.com* der generische Ablauf eines verdeckten Datendiebstahls mittels DNS, der bereits unter Verwendung von frei verfügbarer Software wie *dnscat2* oder Malware wie *FrameworkPOS* zu beobachten war [GDAT14], dargestellt und nachfolgend im Detail beleuchtet:

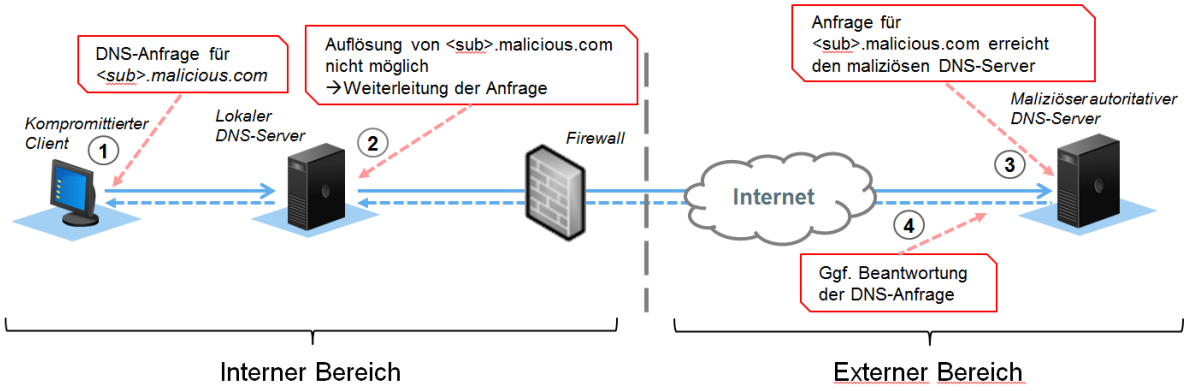

Abb. 2: Daten-Exfiltration mittels eines autoritativen DNS-Servers

Den Ausgangspunkt für die Daten-Exfiltration bildet beispielsweise ein mithilfe von Malware kompromittierter Client innerhalb eines Netzwerks (**1**). Unter Verwendung dieses Clients erfolgt mit dem Ziel eines unbemerkten Diebstahls von sensiblen Informationen eine Anfrage zur Auflösung der Domain *<sub>.malicious.com*. Hierzu wird ein DNS-Paket, welches die enkodierten Daten enthält, zu einem Zielsystem unter Kontrolle des Angreifers gesendet. Bei *<sub>* handelt es sich idealerweise um eine einzigartige Subdomain, die bisher noch keinem rekursiven Nameserver bekannt ist. Hierdurch soll sichergestellt werden, dass das DNS-Paket zwecks Beantwortung der vermeintlich legitimen Anfrage bis zum Nameserver des Angreifers weitergeleitet wird.

Die Konfiguration des skizzierten Netzwerks lässt keine Auflösung über externe Nameserver zu. Stattdessen werden alle Anfragen ausschließlich über den lokalen nicht-autoritativen, rekursiven DNS-Server beantwortet (**2**). Wie der Abbildung zu entnehmen ist, verfügt dieser über keine Informationen zu *<sub>.malicious.com*. Daher erfolgt eine Weiterleitung der Anfrage an einen Nameserver im Internet. An dieser Stelle haben die sensiblen Daten trotz des Einsatzes einer Firewall oder anderer Sicherheitssysteme aufgrund der häufig nicht vorhandenen Einschränkung des DNS-Protokolls bereits das lokale Netzwerk verlassen.

Über das Internet wird eine Weiterleitung der Anfrage für *<sub>.malicious.com* an mehrere rekursive Nameserver, die in der global aufgestellten DNS-Infrastruktur etabliert sind, durchgeführt. Wenn hierbei keine Antwort geliefert werden kann, erreicht das DNS-Paket zuletzt einen der beiden autoritativen Nameserver, welche für die Domain *malicious.com* verantwortlich sind (**3**). Diese befinden sich unter der Kontrolle des Angreifers und ermöglichen somit, die innerhalb des Pakets enkodierten sensiblen Informationen zu dekodieren. Zuletzt wird eine Antwort mit einer oder mehreren beliebigen IP-Adressen generiert und rekursiv über alle kontaktierten DNS-Instanzen zurück an den kompromittierten Client gesendet (**4**). Dieses Vorgehen ist zwar nicht zwingend zur erfolgreichen Exfiltration erforderlich, soll jedoch allgemein

die maliziöse DNS-Kommunikation durch Wahrung der Protokoll-Konformität bestmöglich tarnen.

Der beschriebene generische Ablauf erlaubt grundsätzlich eine Daten-Exfiltration mittels verschiedener netzwerk-steganographischer Verfahren. Eine aus Perspektive des Angreifers vielversprechende Technik besteht in der Verwendung des in [ANO+11] beschriebenen Ansatzes. Hierbei werden die zu transferierenden Daten in das zwei Bytes große *Identification*-Feld des DNS-Headers enkodiert. Dieses wird normalerweise zufällig erzeugt und dient in Kombination mit dem Quell-Port des Absenders einer DNS-Anfrage zur eindeutigen Identifizierung der zugehörigen Antwort, welche einen identischen Wert enthalten muss. Der wesentliche Nachteil dieses Verfahrens besteht in der Größe des Header-Felds. Bei lediglich zwei Bytes, die zur Exfiltration zur Verfügung stehen, würde beispielsweise die Übertragung einer ca. 1.000 Bytes großen */etc/shadow*-Datei, welche im Allgemeinen auf jedem Linux-System verfügbar ist und die Hashwerte aller Benutzer-Passwörter enthält, die Erzeugung von wenigstens 500 DNS-Anfragen an Domains der Form *<sub>.malicious.com* erfordern. Aus diesem Grund wird in der Fachwelt – wie auch im Zuge des thematisierten Sicherheitsvorfalls bei *The Home Depot* – häufig ein anderer Ansatz, der eine verdeckte Übertragung von sensiblen Daten innerhalb von Subdomains erlaubt, beobachtet [GDAT14].

Eine auf Subdomains basierende Exfiltration kann auf einem kompromittierten System im Normalfall bereits ohne speziell zu diesem Zweck benötigte Software mithilfe von System-Werkzeugen durchgeführt werden. Handelt es sich beispielsweise um ein Linux-System, erlauben die *bash*-Shell mit *root*-Berechtigung und der Einsatz des *nslookup*-Befehls bereits einen enkodierten Datentransfer mittels DNS-Anfragen (siehe Abbildung 3):

```
# i=1; for hex in `cat /etc/shadow | xxd -c 16 -p`; \
do nslookup $hex.$i.malicious.com; \
   i=$(($i+1)); \
done;
```

Abb. 3: Netzwerk-steganographische Daten-Exfiltration mit System-Werkzeugen

Die Abbildung zeigt eine `for`-Schleife, in der die */etc/shadow*-Datei iterativ in 16 Bytes lange hexadezimale Zeichenketten `hex` zerlegt und jeweils mittels `nslookup` eine DNS-Anfrage der Form *<hex_value>.<counter>.malicious.com* durchführt wird. Die autoritativen Nameserver kontrollierend, kann beispielsweise mithilfe eines *Python*-Skripts das Verhalten eines DNS-Servers nachgebildet und parallel die in den Subdomains enkodierten Daten zu einer Datei zusammengesetzt werden. Die korrekte Sortierung der einzelnen Zeichenketten erfolgt hierbei unter Verwendung des innerhalb der jeweiligen Anfrage enkodierten Zählers.

Diese langen, dynamisch erzeugten Subdomains sind nicht intuitiv lesbar und werden daher üblicherweise nicht manuell vergeben. Allerdings verwenden einige Content Delivery Networks (CDNs) wie Amazon Web Services oder Antivirus-Hersteller wie Sophos ebenfalls diese Art der Subdomains für die dynamische Verteilung von Anfragen. Das Aussehen der eingesetzten Subdomains unterscheidet sich hierbei kaum von dem einer Daten-Exfiltration und erschwert infolgedessen unter anderem die Erkennung mittels entropiebasierter Ansätze [FaAt13].

4 Erkennungsverfahren

Mit Fokus auf die Protokolle DNS und ICMP wurde nach einem neuartigen, auf statistischer Analyse und (überwachtem) maschinellem Lernen basierenden Verfahren zur Detektion dieser fortgeschrittenen Datendiebstähle unter Einsatz von netzwerk-steganographischen Verfahren geforscht. Die Kombination dieser beiden wissenschaftlichen Themengebiete soll dazu dienen, in einer initialen *Lernphase* die verhaltensbasierten Muster einer legitimen Kommunikation innerhalb eines dedizierten Netzwerks zunächst automatisiert zu erlernen. Auf diese Weise soll in einer darauf folgenden *Arbeitsphase* eine zuverlässige Unterscheidung des legitimen Netzwerkverkehrs von einschlägigen Mustern einer Daten-Exfiltration mittels Netzwerk-Steganographie ermöglicht werden (vgl.Abbildung 4).

Die statistische Analyse dient hierbei sowohl der Identifizierung als auch der Auswertung von Mustern innerhalb des Netzwerkverkehrs, die zur Klassifizierung von legitimer Kommunikation und Daten-Exfiltration herangezogen werden sollen. Das Muster (engl. *pattern*) wird hierbei als Menge von Merkmalen zur Beschreibung eines zu klassifizierenden Netzwerkobjekts, zum Beispiel eines Pakets oder Netzwerk-Streams, definiert und in Form eines Vektors dargestellt [Rogm17]. Bei einem Merkmal (engl. *feature*) handelt es sich entweder um eine kategorische oder numerische Variable. Während beispielsweise eine IP-Adresse kategorisch ist, stellt die Paketgröße eine numerische Variable dar. Erfolgt eine konkrete Wertzuweisung zu einem Merkmal, wird diese auch als Merkmalsausprägung bezeichnet.

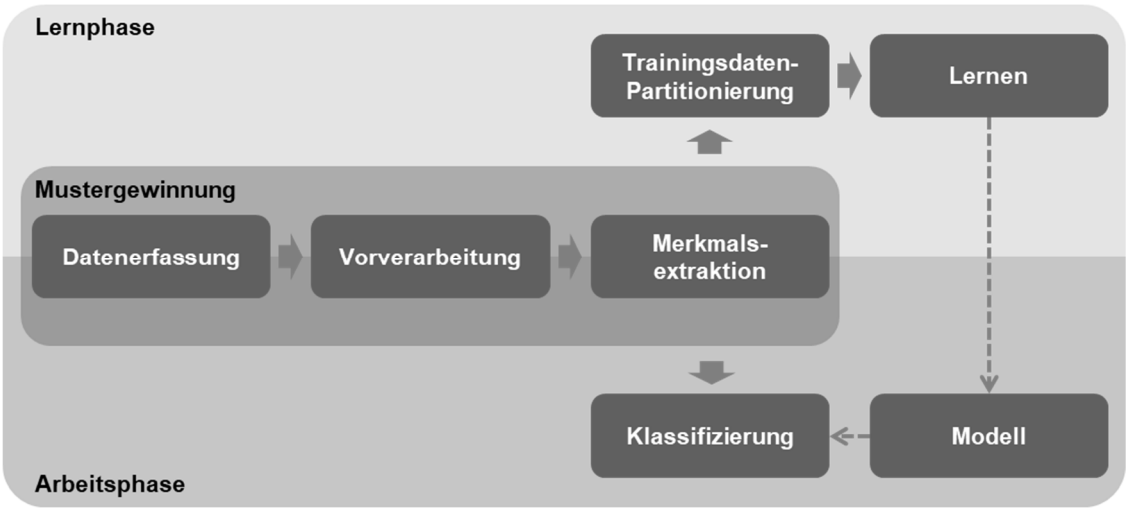

Abb. 4: Prozess der Mustererkennung [Rogm17]

In der Abbildung wird der für das Erkennungsverfahren entwickelte Prozess der Mustererkennung dargestellt. Die Schritte „Datenerfassung“, „Vorverarbeitung“ sowie „Merkmalsextration“ werden als „Mustergewinnung“ zusammengefasst und sind ein wesentlicher Bestandteil von Lern- *und* Arbeitsphase.

4.1 Mustergewinnung

Zur Gewährleistung einer möglichst umfassenden *Datenerfassung*, die eine Extraktion von Merkmalen unter Verwendung von allen zu Grunde liegenden Protokollebenen erlaubt, fungiert ein Paket-Sniffer als Sensor. Dieser dient der Sammlung des Datenverkehrs in Form von Netzwerkpaketen, die im nächsten Schritt einer Vorverarbeitung unterzogen werden können.

Damit ein verdeckter Datendiebstahl auf Basis von verhaltensbasierten Mustern detektiert werden kann, besteht die grundlegende Idee der *Vorverarbeitung* darin, diese Pakete in Form von Netzwerk-Streams, vergleichbar zu TCP-Streams, zusammenzufassen. Das angestrebte Ziel ist es hierbei – in Anlehnung an wissenschaftliche Veröffentlichungen zur Untersuchung von TCP-Verbindungen [AuMG07], [DCG+09], [Rogm16] – eine statistische Analyse der miteinander in Beziehung stehenden DNS-Pakete zu realisieren. Dies soll eine Erfassung von statistischen Merkmalen wie den Zeitabständen von aufeinander folgenden Paketen an dasselbe Ziel oder der Anzahl der innerhalb eines Streams übertragenen Bytes ermöglichen. Die Menge an verschiedenen Merkmalen dient hierbei wiederum als Muster, welches schließlich eine Klassifizierung der jeweiligen Streams erlaubt.

Während einzelne Pakete einer TCP-Verbindung anhand eines Tupels bestehend aus Quell- und Ziel-IP sowie Quell- und Ziel-Port identifizierbar sind, ist dies bei dem typischerweise auf UDP-Nachrichten basierenden DNS-Protokoll nicht vorgesehen. DNS-Anfragen enthalten hingegen im Header einen zufällig erzeugten *Identifier*, der für eine Zuordnung ebenfalls in dem entsprechenden Antwort-Paket vorhanden sein muss. Dieser ist allerdings nicht zur Identifizierung von miteinander in Beziehung stehenden Paketen geeignet, da für jede Anfrage ein anderer *Identifier* generiert wird. Stattdessen haben sich die Quell- und Ziel-IP in Kombination mit dem Domain-Namen für die Zusammenfassung einzelner Pakte zu einem DNS-Stream als zielführend herausgestellt. Ein vergleichbares Vorgehen hat sich unter anderem im Zuge einer wissenschaftlichen Arbeit zur Tunnel-Erkennung als vielversprechend erwiesen [Jaro09] und wird nachfolgend an einem Beispiel verdeutlicht:

Unter Verwendung eines aktuellen Browsers ohne Werbeblocker wie *NoScript* oder *AdBlock Plus* erfolgen bei einem Aufruf der Webseite *spiegel.de* ungefähr 130 verschiedene DNS-Anfragen. Neben den Adressauflösungen von Werbenetzwerken werden wiederholt Subdomains wie unter anderem *www.spiegel.de*, *cdn3.spiegel.de*, *cdn4.spiegel.de* sowie *magazin.spiegel.de* angefragt (vgl.Abbildung 5).

No.	Protocol	Length	Info
25	DNS	74	Standard query 0x0b0a A www.spiegel.de
27	DNS	90	Standard query response 0x0b0a A www.spiegel.de A 62.138.116.25
95	DNS	74	Standard query 0xa1cb A www.spiegel.de
96	DNS	90	Standard query response 0xa1cb A www.spiegel.de A 62.138.116.25
104	DNS	74	Standard query 0x03f9 A www.spiegel.de
106	DNS	90	Standard query response 0x03f9 A www.spiegel.de A 62.138.116.25
108	DNS	74	Standard query 0x8ee8 A www.spiegel.de
109	DNS	90	Standard query response 0x8ee8 A www.spiegel.de A 62.138.116.25
110	DNS	74	Standard query 0xd40f A www.spiegel.de
112	DNS	90	Standard query response 0xd40f A www.spiegel.de A 62.138.116.25
113	DNS	74	Standard query 0x79a6 A www.spiegel.de
115	DNS	90	Standard query response 0x79a6 A www.spiegel.de A 62.138.116.25
145	DNS	75	Standard query 0x4f97 A cdn3.spiegel.de
147	DNS	74	Standard query 0x413d A www.spiegel.de
149	DNS	90	Standard query response 0x413d A www.spiegel.de A 62.138.116.25
150	DNS	78	Standard query 0xd3fd A magazin.spiegel.de
152	DNS	78	Standard query 0x3ac9 A magazin.spiegel.de
160	DNS	75	Standard query 0xdf99 A cdn4.spiegel.de
161	DNS	75	Standard query 0xe260 A cdn4.spiegel.de

Abb. 5: Auszug der DNS-Anfragen beim Aufruf der Domain *spiegel.de*

Im Zuge der Vorverarbeitung werden alle Anfragen und Antworten der Form *<sub>.spiegel.de* eines Clients einem einzigen DNS-Stream, der alle Pakete bezüglich der Domain *spiegel.de* beinhaltet, zugeordnet.

Die *Merkmalsextraktion* bildet den letzten Schritt der Mustergewinnung und dient im Wesentlichen der Bestimmung von Merkmalen auf Basis der erzeugten Netzwerk-Streams. Pro Stream wird hierbei die Menge der jeweiligen extrahierten kategorischen oder numerischen Variablen als ein Muster zusammengefasst, welches diesen DNS-Stream beschreiben und hierdurch eine Klassifizierung des Netzwerkobjekts erlauben soll.

Prinzipiell kann eine Vielzahl von Merkmalen aus den erzeugten Netzwerk-Streams extrahiert werden. Daher erfolgte eine umfassende wissenschaftliche Untersuchung verschiedener Merkmale hinsichtlich des Potentials zur Klassifizierung zwischen legitimer Kommunikation und DNS- bzw. ICMP-Steganographie (siehe Tabelle 1).

Tab. 1: Übersicht der untersuchten Merkmale zur Klassifizierung

Merkmal	Beschreibung	Protokoll	Typ
byte_count	Byte-Anzahl eines Streams	*	Numerisch
duration	Dauer eines Streams	*	Numerisch
dest_ip	Ziel-Adresse der DNS-Anfragen	DNS	Kategorisch
mean_psize_out	Durchschnittliche Paketgröße (ausgehend)	*	Numerisch
mean_sublen	Durchschnittliche Subdomain-Länge	DNS	Numerisch
mean_pit_out	Durchschnittliche PIT (ausgehend)	*	Numerisch
packet_count	Paketanzahl eines Streams	*	Numerisch
unique_subcount	Anzahl einzigartiger Subdomains	DNS	Numerisch
unique_datacount	Anzahl einzigartiger data-Felder	ICMP	Numerisch

Einige numerische Variablen wie der *byte_count* oder die *mean_pit_out*, die in der Tabelle durch das Zeichen * gekennzeichnet sind, dienen bereits in verschiedenen Veröffentlichungen zur Klassifizierung von Netzwerkverkehr [AuMG07], [DCG+09], [FaAt13], [Rogm16]. Andere Merkmale wie der *unique_subcount* oder die *dest_ip* sind hingegen im Zuge einiger initialer Tests entwickelt worden. Neben generischen Variablen wie dem *packet_count*, der grundsätzlich zur Analyse von verschiedensten Protokollen dienen kann, erfolgte weiterhin die Suche nach messbaren Protokoll-spezifischen Besonderheiten wie der *mean_sublen* oder dem *unique_subcount*.[1]

Als ein Beispiel für ein effektives Merkmal zur Erkennung von Daten-Exfiltration ist der *byte_count*, also die Gesamtzahl der übertragenen Bytes innerhalb eines Netzwerk-Streams, anzuführen. Dieser ist unabhängig von einem spezifischen Protokoll messbar und wächst naturgemäß mit der Menge der zu exfiltrierenden Daten. Die Bedeutsamkeit zur Erkennung von verdeckten Datendiebstählen wird besonders bei einer beispielhaften graphischen Gegenüberstellung von legitimer Kommunikation und DNS-Steganographie deutlich (vgl.Abbildung 6).

Die Abbildung zeigt auf einer logarithmischen Skala jeweils die Anzahl der Bytes, die aus jedem der innerhalb eines produktiven Netzwerks aufgezeichneten DNS-Streams extrahiert

1 Ein wesentlicher Teil der wissenschaftlichen Untersuchung innerhalb der vorangegangenen Masterarbeit bestand darin, das Potential der einzelnen Merkmale hinsichtlich der Erkennung von netzwerk-steganographischer Daten-Exfiltration aufzuzeigen und zu diskutieren. Zur Reduzierung des Umfangs kann an dieser Stelle jedoch nur exemplarisch darauf eingegangen werden.

wurde. Der *byte_count* eines legitimen Streams ist hierbei in Form eines Kreises dargestellt, während die gemessenen Werte der Daten-Exfiltrationen als Rauten abgebildet sind.

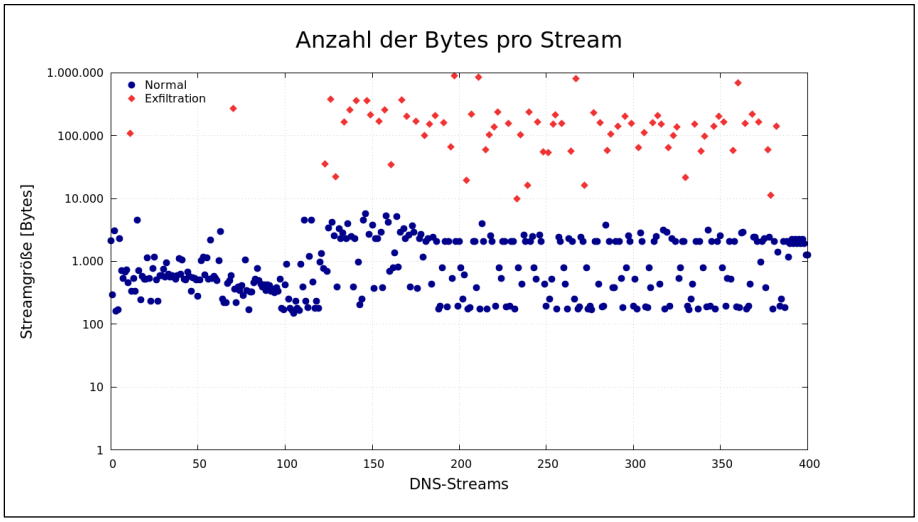

Abb. 6: Vergleich des *byte_counts* bei legitimer und maliziöser Kommunikation

Anhand des Diagramms wird ersichtlich, dass die Betrachtung der Byte-Anzahl bereits eine erste Differenzierung zwischen normalen und maliziösen Netzwerk-Streams ermöglicht. Diese erste Beobachtung ist intuitiv nachvollziehbar, da in Abhängigkeit der Größe einer zu übertragenden Datei die Gesamtzahl der Bytes eines Streams ebenfalls ansteigt.

Analog zum *byte_count*, wurde im Zuge einer wissenschaftlichen Untersuchung die Effektivität jedes der in **Fehler! Verweisquelle konnte nicht gefunden werden.** aufgelisteten statistischen Merkmale evaluiert sowie die Vor- und Nachteile des jeweiligen Merkmals zur Erkennung von Netzwerk-Steganographie im Detail herausgearbeitet und vergleichbar zu **Fehler! Verweisquelle konnte nicht gefunden werden.** visualisiert. Das konkrete Vorgehen und die Ergebnisse dieser Untersuchung können in [Rogm17] im Detail eingesehen werden.

4.2 Klassifikator

Zur Klassifizierung der extrahierten Muster des Netzwerkverkehrs kommt der naive Bayes-Klassifikator zum Einsatz. Dieser erlaubt unter anderem bereits eine Detektion von Protokoll-Tunneln [DCG+09] sowie eine allgemeine Klassifizierung von Netzwerkverkehr [ErMA06] und soll auch eine Erkennung von netzwerk-steganographischer Daten-Exfiltration ermöglichen. Ein wesentlicher Vorteil dieses Verfahrens besteht darin, dass bereits mit einer kleinen Menge von bekannten Mustern ein effizientes Modell zur zuverlässigen Erkennung trainiert werden kann. Das Lernen und die Klassifizierung sind hierbei nahezu in Echtzeit realisierbar und lassen sich ferner wegen der zu Grunde liegenden Mathematik im Detail nachvollziehen.

Der naive Bayes-Klassifikator soll dazu dienen, auf Basis einer berechneten Wahrscheinlichkeit jedes extrahierte Muster $\vec{x}$ einer der Klassen ω_1 = „Legitime Kommunikation“ oder ω_2 = „Daten-Exfiltration“ zuzuordnen [Rogm17]. Für eine bestmögliche Zuordnung ist es im Zuge der Lernphase erforderlich, eine ausreichende Anzahl an Trainingsdaten von beiden Klassen zur Erzeugung eines bestmöglichen Modells heranzuziehen. Dieses Vorgehen wird in der Fachliteratur als *mehrklassige Klassifizierung* (engl. *multi-class classification*) bezeichnet und setzt für eine möglichst hohe Genauigkeit der korrekten Zuordnungen in der Lernphase eine ungefähre Balance zwischen der Anzahl sowie der Qualität von bekannten Mustern *aller* relevanten

Klassen voraus. Vor allem bei fortgeschrittenen Angriffstechniken wie einer netzwerk-steganographischen Daten-Exfiltration stehen jedoch zur Extraktion geeigneter Muster oftmals nur begrenzt Mitschnitte von Netzwerkobjekten der maliziösen Klasse zur Verfügung. Infolgedessen wird in dieser Arbeit ein weiterer in der Fachwelt etablierter Ansatz – die *einklassige Klassifizierung* (engl. *one-class classification*) – herangezogen [Tax01]. Diese soll es ermöglichen, ein Modell ausschließlich anhand der Klasse ω_1, also der legitimen Kommunikation, welche im Allgemeinen vielfach existiert oder mit einfachen Mitteln aufzuzeichnen ist, zu trainieren.[2]

4.3 Lern- und Arbeitsphase

Orientiert an dem einleitend dargestellten Prozess der Mustererkennung, untergliedert sich die *Lernphase* in eine „Trainingsdaten-Partitionierung“ und das eigentliche „Lernen“. Im Zuge der Trainingsdaten-Partitionierung werden die zu Grunde liegenden Trainingsdaten, also die bekannten Muster einer legitimen Kommunikation, unter Auswahl einer geeigneten Partitionierungsstrategie zunächst in eine Trainings-, Validierungs-, und Testmenge aufgeteilt. Während des Lernens wird die Trainingsmenge zur Erzeugung mehrerer Modelle herangezogen. Diese unterscheiden sich im Allgemeinen hinsichtlich einiger modifizierbarer Parameter des zu Grunde liegenden naiven Bayes-Klassifikators, welche die Qualität der Klassifizierung beeinflussen können. Unter Verwendung der Validierungsmenge werden die erzeugten Modelle ggf. durch eine erneute Anpassung der jeweiligen Parameter optimiert und schließlich ein bestmögliches Modell mit der höchsten Qualität ausgewählt. Zuletzt soll auf Basis der Testmenge die Abschätzung des ausgewählten Modells hinsichtlich der Zuverlässigkeit einer korrekten Klassifizierung der unbekannten Muster von bisher nicht betrachteten Netzwerk-Streams erfolgen.

Nach Abschluss der Lernphase folgt die *Arbeitsphase*, in der die Klassifizierung von unbekannten Mustern angestrebt wird. Das während des Lernens erzeugte Modell soll es dem Bayes-Klassifikator hierbei ermöglichen, die Klassenzugehörigkeit von unbekannten Mustern eines Netzwerk-Streams zu bestimmen und hierdurch eine Detektion von fortgeschrittener Netzwerk-Steganographie zu ermöglichen.

5 Realisierung

Das entwickelte Verfahren wird in Form einer in Python programmierten Erkennungssoftware bereitgestellt. Diese ermöglicht eine automatisierte Detektion von netzwerk-steganographischer Daten-Exfiltration und wird der Allgemeinheit unter der Lizenz GNU GPLv3 auf der Plattform *Github*[3] quelloffen zur Verfügung gestellt. Bei der Software handelt es sich um ein verteiltes System. Hierdurch soll es grundsätzlich ermöglicht werden, die Sensoren an verschiedenen Positionen innerhalb eines zu überwachenden Netzwerks zu platzieren. Während ein Sensor beispielsweise innerhalb des Client-Segments positioniert werden kann, besteht die Möglichkeit, eine oder mehrere weitere Instanzen in den DMZ-Bereichen zu betreiben. Die plattformunabhängige Programmiersprache *Python* erlaubt es hierbei, die Komponenten sowohl auf Windows- als auch auf Unix-Systemen einzusetzen.

Neben der Durchführung einer umfassenden wissenschaftlichen Messreihe, deren Aufbau, Durchführung und Ergebnis im Detail in [Rogm17] einzusehen sind, konnte in Rücksprache

[2] Auf eine umfassende Herleitung der einzelnen mathematischen Formeln wird an dieser Stelle aufgrund des Umfangs verzichtet. Diese kann jedoch bei Bedarf in [Rogm17] nachgelesen werden.

[3] Die Erkennungssoftware ist auf https://github.com/0x71/PyExfilDetect zu finden.

mit der Sicherheitsforscherin Mila Parkour eine Untersuchung der aufgezeichneten Kommunikation echter Schadsoftware von vergangenen Sicherheitsvorfällen, die nicht unmittelbar mit einer Daten-Exfiltration in Verbindung gebracht wurden, durchgeführt werden. Parkour stellte hierzu eine Vielzahl verschiedener Netzwerk-Mitschnitte von Malware, unter anderem die Exploit Kits *Angler* und *Sweet Orange* sowie dem Banking-Trojaner *Dridex*, bereit. Insgesamt hat unter Verwendung der entwickelten Erkennungssoftware eine automatisierte Untersuchung von 534 Aufzeichnungen unterschiedlicher Malware-Kommunikation, die überwiegend öffentlich auf dem von Parkour betriebenen Blog *Contagio Dump* verfügbar ist, stattgefunden [Park17]. In sechs der analysierten Mitschnitte konnte jeweils unter Verwendung der entwickelten Erkennungssoftware mithilfe des einklassigen naiven Bayes automatisiert eine DNS-Steganographie identifiziert werden.

Neben vier steganographischen Daten-Exfiltrationen unter Verwendung autoritativer Nameserver konnten ferner zwei verdeckte Datendiebstähle mithilfe einer direkten DNS-Kommunikation detektiert werden. Bei letzteren kam eine gefälschte Domain, *google.com*, zum Einsatz. Bei der zu Grunde liegenden Malware, welche die maliziösen Streams erzeugt hat, handelte es sich um die Exploit Kits *Angler* und *Sweet Orange*.

Die vollständigen Analyse-Ergebnisse aller untersuchten Malware-Kommunikationen sind im Internet auf einer eigens im Zuge dieser Arbeit entwickelten Plattform *ExfilDetect.com* verfügbar [Exfi17]. Hierbei handelt es sich um eine Plattform, die ohne ein spezifisches technisches Vorwissen hinsichtlich des entwickelten Verfahrens eine automatisierte Analyse von aufgezeichneten Netzwerk-Mittschnitten ermöglicht. Folglich werden mit dieser Arbeit zwei Möglichkeiten für den praktischen Einsatz des implementierten Verfahrens zur Erkennung von bisher nicht zuverlässig detektierbarer netzwerk-steganographischer Daten-Exfiltration in Form der Erkennungssoftware sowie der Plattform veröffentlicht.

6 Fazit & Ausblick

Die Erkennungssoftware kann zukünftig in beliebigen Netzwerken eingesetzt werden und durch eine frühzeitige Detektion von verdeckten Datendiebstählen zur Erhöhung der Sicherheit beitragen. Weiterhin erlaubt die bereitgestellte Webseite *ExfilDetect.com* eine kurzfristige Analyse von Netzwerk-Mitschnitten, die typischerweise zur zeitkritischen Untersuchung von Sicherheitsvorfällen durch ein Incident Response-Team benötigt wird.

Eine angepasste Version der implementierten Erkennungssoftware soll darüber hinaus in eine quelloffene Malware Analyse-Umgebung, der *Cuckoo Sandbox*, integriert werden. Hierzu ist bereits ein *Pull Request* auf der Plattform *Github* erstellt worden. In Rücksprache mit den Entwicklern erfolgt aktuell eine Diskussion hinsichtlich der Optionen für eine sinnvolle Kombination der beiden Projekte. Durch die Zusammenarbeit soll es den Nutzern dieser Sandbox zukünftig erstmalig ermöglicht werden, eine dynamisch untersuchte Malware zusätzlich automatisiert hinsichtlich einer Daten-Exfiltration zu prüfen.

Das Erkennungsverfahren ist mit dem Fokus auf der Detektion von Exfiltrationen mittels DNS- und ICMP-Steganographie entwickelt worden. In der Konsequenz erfolgt aktuell ausschließlich eine Untersuchung von Netzwerk-Kommunikation basierend auf diesen beiden Protokollen. Auf Basis der Ergebnisse dieser Arbeit könnte das veröffentliche Verfahren zukünftig um eine Überwachung weiterer Protokolle, zum Beispiel HTTP oder HTTPS, ergänzt werden. Durch

diese Art der Erweiterungen wäre langfristig eine ganzheitliche Überwachung des Netzwerkverkehrs unter Verwendung des neuartigen Verfahrens basierend auf statistischer Analyse und maschinellem Lernen denkbar.

Literatur

[AuMG07] T. Auld, A. Moore, S. Gull: Bayesian Neural Networks for Internet Traffic Classification. In: IEEE Transactions on Neural Networks, Vol. 18, No. 1, IEEE, (2007) 223-239.

[ANO+11] A. Altalhi; M. Ngadi; S. Omar, et al.: DNS ID Covert Channel based on Lower Bound Steganography for Normal DNS ID Distribution. In: IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 6, No 3, IJCSI (2011).

[Boeh14] B: Boehm: arxiv.org: StegExpose - A Tool for Detecting LSB Steganography. https://arxiv.org/abs/1410.6656 (2014).

[Bund16] Bundesministerium für Sicherheit in der Informationstechnik: bund.de: Die Lage der IT-Sicherheit in Deutschland 2016. https://www.bsi.bund.de/ DE/ Publikationen/Lageberichte/lageberichte_node.html (2016) 22.

[CMX+14] A. Chen, B. Moore, H. Xiao, et al.: Detecting Covert Timing Channels with Time-Deterministic Replay. In: OSDI'14 Proceedings of the 11th USENIX conference on Operating Systems Design and Implementation (2014) 541-554.

[Dinc12] L. Dinca: Secret message in a ping: creation and prevention. In: SITE'12 Proceedings of the 11th international conference on Telecommunications and Informatics, (WSEAS), World, Wisconsin, USA (2012) 32-37.

[DCG+09] M. Dusi, M. Crotti, F. Gringoli, et al.: Tunnel Hunter: Detecting application-layer tunnels with statistical fingerprinting. In: Computer Networks Vol. 53, No. 1 (2009) 81-97.

[DrSU16] M. Drzymała, K. Szczypiorski, M. Urbański: Network Steganography in the DNS Protocol. In: International Journal of Electronics and Telecommunications, Vol. 62, No 4, IJET (2016) 343-346.

[EC-C09] EC-Council: Ethical Hacking and Countermeasures: Threats and Defense Mechanisms: Delmar Cengage Learning (2009) 3.

[ErMA06] J. Erman, A. Mahanti, M. Arlitt: Internet Traffic Identification using Machine Learning. In: IEEE Globecom, IEEE (2006).

[Exfi17] ExfilDetect: exfildetect.com: ExfilDetect - Free Online Data Exfiltration Detection Service. https://exfildetect.com/ (2017).

[Extr16] ExtraHop: extrahop.com: Detecting Data Exfiltration. https:// www. extrahop. com/solutions/data-exfiltration-detection/ (2016).

[FaAt13] G. Farnham, A. Atlasis: sans.org: Detecting DNS Tunneling. https://www. sans. org/reading-room/whitepapers/dns/detecting-dns-tunneling-34152 (2013).

[GiBC06] A. Giani, V. Berk, G. Cybenko: Proceedings of SPIE - The International Society for Optical Engineering: Data Exfiltration and Covert Channels. http://www. ists.dartmouth.edu/library/293.pdf (2006) 7.

[GDAT14] GDATA: gdatasoftware.com: New FrameworkPOS variant exfiltrates data via DNS requests. https:// blog. gdatasoftware. com/ 2014/ 10/ 23942-new-frameworkpos- variant-exfiltrates-data-via-dns-requests (2014).

[Jaro09] H. Jarod: defcon.org: Catching DNS tunnels with A.I. https://www.defcon. org/ images/defcon-17/dc-17-presentations/defcon-17-jhind-dns_tunnels_with_ai.pdf (2009).

[LuCC08] X. Luo, E. Chan, K. C. Chang: TCP covert timing channels: Design and detection. In: 2008 IEEE International Conference on Dependable Systems and Networks with FTCS and DCC (DSN), IEEE (2008).

[Mazu13] W. Mazurczy: arxiv.org: VoIP steganography and its Detection - A survey. https://arxiv.org/ftp/arxiv/papers/1203/1203.4374.pdf (2013)

[MuLe05] S. Murdoch, S. Lewis: Embedding Covert Channels into TCP/IP. In: 7th International Workshop, IH 2005, Barcelona, Spain, June 6-8 (2005). Revised Selected Papers, Heidelberg, Springer, 2005 247-261.

[Mogu09] R. Mogull: techtarget.com: How to use data loss prevention tools to stop data exfiltration. http://searchfinancialsecurity.techtarget.com/tip/How-to-use-data-loss-prevention-tools-to-stop-data-exfiltration (2009).

[Park17] M. Parkour: http://contagiodump.blogspot.de/: Contagio Malware Dump. contagiodump.blogspot.de (2017).

[PwC15] PwC: pwc.co.uk: 2015 Information Security Breaches Survey. http://www. pwc.co.uk/assets/pdf/2015-isbs-technical-report-blue-digital.pdf (2015).

[Risk16] Risk Based Security: riskbasedsecurity.com: Data Breach QuickView. https:// pages. riskbasedsecurity. com/ 2016-midyear-data-breach-year-in-review, (2016) 1-3.

[Secu14] Security Lancaster: lancaster.ac.uk: Data Exfiltration Report. http://www. lancaster.ac.uk/media/lancaster-university/content-assets/images/security- lancaster/ seculanc_data_exfil_report.pdf (2014) 12-16.

[Rogm16] N. Rogmann: nilsrogmann.de: Automatiserte Erkennung von Infection-Proxys mithilfe von statistischer Analyse. https://www.fbi.h-da.de/fileadmin/Inhalt/ dokumente/ KOSI/ Auszeichnung-Absolventen/ Nils_Rogmann/Automatisierte_ Erkennung_ von_ Infection-Proxys_mithilfe_von_statistischer_Analyse_-_Nils_ Rogmann.pdf (2016).

[Rogm17] N. Rogmann: Automatisierte Erkennung von Daten-Exfiltration mithilfe von statistischer Analyse und maschinellem Lernen. https://www.fbi.h-da.de/fileadmin/ Inhalt/ dokumente/KOSI/Auszeichnung-Absolventen/ Nils_Rogmann/ rogmann_ nils_MA_1490104662.pdf (2017).

[Tax01] D. Tax, David: tudelft.nl: Concept-learning in the absence of counter-examples. http://homepage.tudelft.nl/n9d04/thesis.pdf (2001).