Theoretische und praktische Risiken der Verwendung von URL-Verkürzungsdiensten

Alexander Neumann
RedTeam Pentesting GmbH
Dennewartstr. 25-27
52068 Aachen
alexander.neumann@redteam-pentesting.de

1 Einführung

In einigen Fällen sind URLs, wie sie heute im modernen Web 2.0 verwendet werden, zu lang und damit umständlich zu übermitteln. In diesen Fällen können URL-Verkürzungsdienste verwendet werden. Diese bieten Benutzern die Möglichkeit, eine nahezu beliebig lange URL durch eine sehr kurze URL von weniger als 30 Zeichen Länge zu ersetzen. Diese kurze URL leitet beim Aufruf einfach auf die lange URL weiter. Damit scheint es für Benutzer zunächst so, als wären beide URLs, kurze und lange, äquivalent.

Seit der Einführung von Twitter, der Evolution anderer Kurznachrichtendienste, sowie sozialer Netzwerke, werden URL-Verkürzungsdienste von einer großen Menge von Benutzern tagtäglich verwendet, ohne dass jedoch die Risiken und Nebenwirkungen ausreichend beleuchtet worden sind.

In diesem Artikel werden nach einem kurzen Abriss der Entwicklungsgeschichte von Verkürzungsdiensten zunächst theoretische Risiken vorgestellt. Anschließend werden die dazu durchgeführten praktischen Untersuchungen erläutert und die Ergebnisse präsentiert. Zum Schluss wird noch ein Fazit gezogen.

1.1 Definitionen

Es wird im Folgenden zwischen den Begriffen *kurze URL* und *verkürzte URL* unterschieden. Eine *kurze URL* in diesem Artikel bezeichnet eine URL, welche aus weniger als 35 Zeichen besteht. Eine kurze URL muss also nicht notwendigerweise auf einen URL-Verkürzungsdienst verweisen. Beispielsweise besteht die URL `http://www.dfn-cert.de` aus lediglich 22 Zeichen und ist damit nach dieser Definition eine kurze URL, verweist aber nicht auf einen Verkürzungsdienst.

Eine *verkürzte URL* hingegen ist eine URL, welche auf einen URL-Verkürzungsdienst verweist. Eine solche URL muss nicht notwendigerweise eine kurze URL sein, beispielsweise enthält der Name des URL-Verkürzungsdienstes `urlshorteningservicefortwitter.com` zusammen mit der Angabe des Protokolls `http://` bereits 41 Zeichen, damit sind alle von diesem Verkürzungsdienst ausgegeben URLs nicht kurz, wohl aber verkürzt.

2 Historie

In den ersten Jahren des World Wide Web waren Benutzer häufig nicht in der Lage, einen eigenen Webserver zu betreiben. Webseiten wurden häufig auf Servern von Universitäten oder *Internet Service Providern* (ISP) betrieben. Dies führte jedoch zu langen und kryptischen URLs, die sich zudem oft änderten.

Das *Online Computer Library Center* (OCLC)[1] stellte im Jahre 1995 ein System vor[2], welches es erlaubt, Internet-Ressourcen unter einer konstanten URL (*Persistent URL*, PURL) anzubieten. Im System hinterlegt ist dabei die eigentliche URL, zu der Benutzer beim Aufruf weitergeleitet werden. Ändert sich diese URL, kann einfach die neue URL für die Ressource im System hinterlegt werden und der Zugriff über die PURL ist weiterhin möglich. Dieser Dienst ist seit Januar 1996 öffentlich verfügbar[3].

Der erste echte URL-Verkürzungsdienst `MakeAShorterLink.com` wurde im Jahre 2001 gestartet[4] und veröffentlichte Ende Dezember 2006 eine Nachricht[5], nach welcher der Dienst eingestellt werde. Im Januar 2007 gab der Dienst bekannt[6], vom URL-Verkürzungsdienst `tinyurl.com` übernommen worden zu sein. Seitdem stellt `tinyurl.com` die Weiterleitung für verkürzte URLs dieses Dienstes zur Verfügung. Den URL-Verkürzungsdienst `tinyurl.com` gibt es seit Anfang 2002[7]. In der in Abschnitt 4.1 vorgestellten Liste der Top-10 Verkürzungsdienste ist `tinyurl.com` immer noch auf dem dritten Platz zu finden.

Häufig werden Verkürzungsdienste verwendet, um URLs zu verkürzen und diese anschließend in einer Nachricht für einem Kurznachrichtendienst wie Twitter einzubetten. Seit dem Start von Twitter im Juli 2006 wurden viele neue Verkürzungsdienste veröffentlicht, aber viele dieser Dienste sind aktuell nicht mehr erreichbar oder dysfunktional. Um diesem Wildwuchs zu begegnen betreibt Twitter seit 2010 einen eigenen Verkürzungsdienst unter der Domain `t.co`.[8] Seit Mitte 2011 werden alle URLs in Twitter-Nachrichten (auch bereits kurze URLs) mit diesem Verkürzungsdienst erneut zwangsläufig verkürzt. Dies gilt für alle Nachrichten, insbesondere auch für nichtöffentliche Nachrichten zwischen Benutzern (*Direct Messages*).

---

[1] `http://www.oclc.org`

[2] `http://worldcat.org/arcviewer/1/OCC/2003/02/11/0000001697/viewer/file40.html`

[3] `http://lists.webjunction.org/wjlists/web4lib/1996-January/003126.html`

[4] `http://web.archive.org/web/20010713183709/http://makeashorterlink.com/`

[5] `http://web.archive.org/web/20061231100803/http://www.makeashorterlink.com/`

[6] `http://web.archive.org/web/20070124223840/http://makeashorterlink.com/`

[7] `http://groups.google.com/group/rec.sport.unicycling/msg/b7beb87a1f83bb75`

[8] `http://blog.twitter.com/2010/06/links-and-twitter-length-shouldnt.html`

3 Theoretische Risiken

Dieser Abschnitt stellt theoretische Risiken für Benutzer von Verkürzungsdiensten vor.

3.1 Malware

Wird ein Verkürzungsdienst verwendet, beispielsweise um eine lange URL zu kürzen und in einer E-Mail zu verwenden, muss der Empfänger immer erst den Verkürzungsdienst kontaktieren. Dieser kann dynamisch bei jeder Anfrage entscheiden, ob eine Umleitung auf die originale, lange URL oder eine andere Aktion ausgeführt wird. Diese Entscheidung kann dediziert anhand der exakten Browser-Version und das Betriebssystem des anfragenden Benutzers getroffen werden. Liefert der Dienst eine Webseite aus, welche spezielle Malware enthält, die exakt auf Browser und Betriebssystem des Benutzers zugeschnitten ist, ist eine hohe Infektionsrate zu erwarten. Des Weiteren werden entsprechend Benutzer, welche aktuelle Software einsetzen, direkt zur Ziel-URL weitergeleitet. So bemerken selbst geschulte Benutzer nicht, dass der Dienst andere Benutzer angreift.

3.2 URLs verschleiern und Spam

Benutzer können vor dem Aufruf das Ziel einer verkürzten URL nicht erkennen. Dies kann dazu genutzt werden, URLs in Spam-E-Mails zu verschleiern. Insbesondere kann die gleiche Spam-URL mehrere Male, auch mit unterschiedlichen Verkürzungsdiensten, verkürzt werden. Dies führt zu unterschiedlichen verkürzten URLs, welche alle auf die gleiche lange URL zeigen.

Spam-Erkennungssoftware kann beim Scannen einer E-Mail zunächst lediglich verkürzte URLs finden und müsste zunächst den entsprechenden URL-Verkürzungsdienst kontaktieren, um die zugehörige lange URL zu erfahren. Dies setzt eine Verbindung zum Internet voraus. Häufig bewertet solche Software die Reputation von gefundenen URLs unter anderem anhand der Domain einer URL sowie anhand der Anzahl Funde einer solchen URL. In diesem Fall kommen verkürzte URLs den Spam-Versendern auf zwei Weisen zugute: Die Reputation der Domains von Verkürzungsdiensten ist im allgemeinen gut, da diese auch für reguläre URLs verwendet werden. Zum anderen können für eine einzige Spam-Seite sehr viele unterschiedliche verkürzte URLs von verschiedenen URL-Verkürzungsdiensten erzeugt werden. Wenn nun noch für jede Spam-E-Mail eine eigene verkürzte URL erzeugt wird, ist auch anhand der Anzahl diese URL nicht als Spam-URL erkennbar.

Zusätzlich kommt hinzu, dass viele Verkürzungsdienste Zugriffsstatistiken führen und diese dem Ersteller anzeigen. Spam-Versender können somit genau feststellen, welche verkürzte URL zur Hauptseite geklickt wurde. Einige Verkürzungsdienste bieten auch weitergehende Informationen wie beispielsweise eine genaue Liste aller Zugriffe inklusive Browserversion und HTTP-Referrer. Damit können Spammer effizient Informationen über ihre Opfer sammeln, ohne jedoch eigene Infrastruktur zum Erfassen dieser Daten betreiben zu müssen.

3.3 Geheime URLs

Dienste wie Scribd[9] oder Google Documents[10] erlauben das Anlegen von Dokumenten, die von allen Benutzern eingesehen oder auch bearbeitet werden können, welche die URL zu diesen Dokumenten kennen. Andere geheime URLs könnten beispielsweise ein Fotoalbum mit privaten Fotos oder das administrative Backend einer Webseite sein. Allen diesen URLs ist gemein, dass die dort abrufbaren Informationen nur dadurch geschützt sind, dass nur berechtigte Personen die URL kennen. Auch solche URLs können jedoch zur Vereinfachung einem URL-Verkürzungsdienst übergeben werden.

Damit können geheime URLs durch zwei unterschiedliche Personengruppen gefunden werden: Zum einen können Administratoren des URL-Verkürzungsdienstes alle dort verkürzten URLs einsehen. Zum anderen sind verkürzte URLs enumerierbar, sodass unbeteiligte Dritte zufällig oder zielgerichtet diese URLs finden können.

3.4 Überwachung

Bei jedem Zugriff auf eine verkürzte URL muss zunächst der Verkürzungsdienst kontaktiert werden. Dieser kann im Browser des Benutzers ein eindeutiges Cookie mit langer Laufzeit für die Domain des Dienstes hinterlassen. Löscht der Benutzer solche Cookies nicht, kann der Dienst die einzelnen Zugriffe des Benutzers einander selbst dann zuordnen, wenn die IP-Adresse des Benutzers wechselt und so den Benutzer über die Gültigkeitsdauer des Cookies überwachen. Ist der Dienst sehr populär, kann ein immer genaueres Profile des Benutzers erstellt werden, ohne dass der Benutzer dies bemerkt, denn üblicherweise sind Verkürzungsdienste für Benutzer, die nur eine verkürzte URL abrufen, transparent.

3.5 Verfügbarkeit von Verkürzungsdiensten

Bei jedem Aufruf einer verkürzten URL muss der Benutzer zwangsläufig zunächst den Verkürzungsdienst kontaktieren. Ist dieser schlecht erreichbar, wird das Aufrufen der URL stark verzögert. Ist der Dienst nicht erreichbar oder wurde der Betrieb eingestellt, kann die verkürzte URL gar nicht mehr aufgelöst werden.

Wenn ein Verkürzungsdienst kompromittiert wurde, werden möglicherweise alle verkürzten URLs auf unerwünschte Webseiten weitergeleitet. Dies ist beispielsweise mit dem URL-Verkürzungsdienst `cli.gs` im Juni 2009 geschehen, dabei wurden die Ziel-URLs von 2,2 Millionen verkürzten URLs geändert.[11] Insgesamt waren nach Einspielen eines Backup 7 % aller verkürzten URLs unwiederbringlich verloren.[12]

[9] `http://www.scribd.com`

[10] `http://docs.google.com`

[11] `http://blog.cli.gs/news/cligs-got-hacked-restoration-from-backup-started`

[12] `http://blog.cli.gs/news/hack-update`

4 Praktische Untersuchung

4.1 Vorbereitungen

Um die im vorherigen Kapitel beschriebenen Risiken praktisch zu untersuchen, wurde zunächst eine Liste von 527 allgemeinen URL-Verkürzungsdienste erstellt. Darin enthalten waren keine speziellen URL-Verkürzungsdienste, wie beispielsweise `amzn.to` (Amazon) oder `fb.me` (Facebook). Die Liste wurde aus verschiedenen Quellen zusammengestellt, unter anderem:

- Dem Autor bereits bekannte Dienste: `bit.ly`, `tinyurl.com`, `xs.to`, `zi.ma`, `nvg8.it`, und `rdr.to`.
- Die Liste der Verkürzungsdienste aus dem Quelltext der Erweiterung *ShortenURL*[13] für den Browser Firefox.
- Die Liste der unterstützten Verkürzungsdienste[14] der Webseite `longurl.org`, diese bietet die Dienstleistung an, die Ziel-URL einer verkürzten URL darzustellen.
- Verschiedene Listen von Verkürzungsdiensten aus unterschiedlichen Blogs[15] [16] [17] [18]
- In Spam-E-Mails gefundene URL-Verkürzungsdienste (siehe Abschnitt 4.2)

Um eine Gewichtung vorzunehmen und eine realistische Auswahl treffen zu können wurde anschließend die Popularität aller Dienste in der Liste ermittelt, indem zwei Mal für jeweils 24 Stunden eine randomisierte Auswahl von bis zu 10 % aller öffentlichen Twitter-Nachrichten mitgeschnitten wurden. Der Zugang zu diesen Nachrichten wurde von Twitter zu Forschungszwecken bereitgestellt. Die genauen Zeiträume und die Anzahl der Nachrichten sind in Tabelle 1 aufgelistet. Diese Zeiträume wurden ausgewählt um Nachrichten von unterschiedlichen Benutzergruppen und zu unterschiedlichen Wochentagen (Donnerstag, Samstag auf Sonntag) abzudecken.

Beginn	Ende	Nachrichten
28. Oktober 2010, 00:23 CET	29. Oktober 2010, 00:23 CET	7,547,787
8. Januar 2011, 13:19 CET	9. Januar 2011, 13:19 CET	8,691,168

Tabelle 1: Zeiträume der Twitter-Nachrichten

Alle gesammelten Nachrichten wurden soweit möglich nach ASCII konvertiert, dabei wurden ungültige Zeichen durch Leerzeichen ersetzt. Anschließend wurde durch Anwendung des regulären Ausdrucks `https?:\/\/[^\s"\]]+` alle URLs gesucht. Dabei wurden 1.208.862 beziehungsweise 1.143.838 URLs gefunden. Die Domains dieser URLs wurden anschließend mit der Liste der Verkürzungsdienste abgeglichen, was die folgende Liste der zehn populärsten Dienste ergab:

[13] `https://addons.mozilla.org/en-US/firefox/addon/shorten-url/`

[14] `http://longurl.org/services`

[15] `http://mashable.com/2008/01/08/url-shortening-services/`

[16] `http://blog.go2.me/2009/01/exhausting-review-of-link-shorteners.html`

[17] `http://blog.go2.me/2009/08/bitly-dominates-tinyurl-and-215-other.html`

[18] `http://6uold.blogspot.com/2008/06/long-list-of-url-shorteners.html`

1. bit.ly (auch j.mp)
2. t.co
3. tinyurl.com
4. goo.gl
5. ow.ly
6. dlvr.it
7. is.gd
8. migre.me
9. dld.bz
10. lnk.ms

Die Domain j.mp ist dabei eine alternative Domain für verkürzte URLs des Dienstes bit.ly. Dieser Dienst dominiert zum Zeitpunkt der Messung alle anderen Dienste (356.553 und 264.530 URLs), gefolgt von t.co, tinyurl.com, goo.gl, ow.ly und dlvr.it (16.000 bis 44.000 URLs jeweils). Diese zehn häufig verwendeten Dienste decken dabei 97 % beziehungsweise 96 % aller verkürzten URLs in den gesammelten Twitter-Nachrichten ab. In Tabelle 2 sind die exakten Zahlen abgedruckt, Abbildung 1 stellt die Größenordnungen grafisch dar.

Host	Anzahl URLs (28.10.2010)	Anzahl URLs (08.01.2011)
bit.ly	347.565	255.413
Alias: j.mp	8.988	9.117
t.co	40.801	42.727
tinyurl.com	34.962	31.387
goo.gl	31.259	25.044
ow.ly	30.002	14.711
dlvr.it	18.368	16.917
is.gd	10.704	9.056
migre.me	5.464	3.179
dld.bz	4.252	3.899
lnk.ms	3.346	2.822

Tabelle 2: Anzahl URLs der häufigsten zehn Dienste

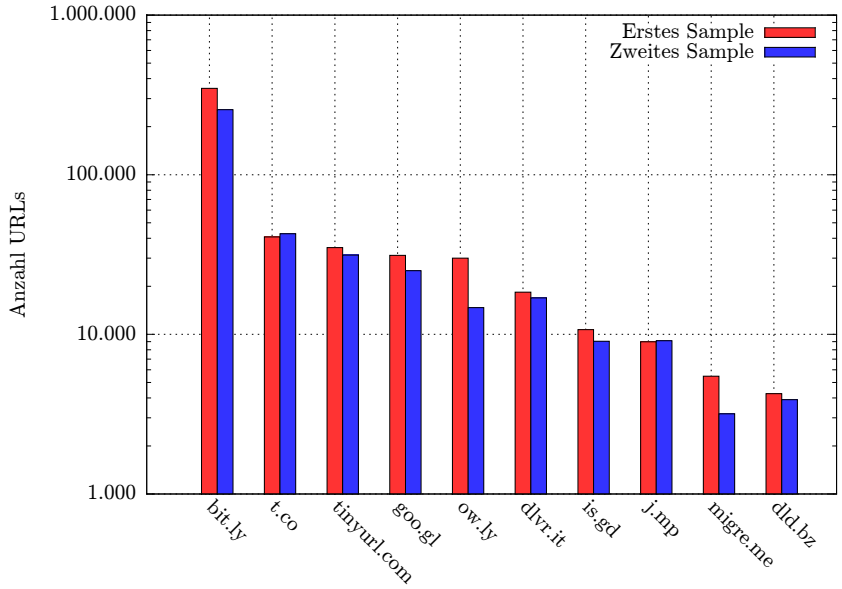


Abbildung 1: Anzahl URLs der häufigsten zehn Dienste

4.2 Verkürzte URLs in Spam

Seit Oktober 2003 sammelt das SCHNUCKI-Projekt[19] des Chaos Computer Club Cologne e.V. E-Mail-Spam, um die Intelligenz von Programmen zum Sammeln von E-Mail-Adressen auf Webseiten zu untersuchen. Die Webseite des Projekts enthält dabei eine Liste von verfremdeten E-Mail-Adressen, die für jede HTTP-Anfrage neu generiert werden. Dabei enthält jede dieser Adressen kodiert unter anderem die folgenden Daten:

- Anfragende IP-Adresse
- Datum und Zeit der Anfrage
- Verfremdungsmethode

Wird eine E-Mail an eine dieser Adressen empfangen, können diese Daten wieder extrahiert werden und es kann die Anzahl der empfangenen E-Mails für verschiedene Verfremdungsmethoden verglichen werden.

Zum Zeitpunkt der Auswertung im August 2010 lief das Projekt bereits seit sieben Jahren, in dieser Zeit wurden insgesamt 7.898.130 Spam E-Mails an 5.594 verschiedene Adressen gesammelt. Diese Daten sind Ideal für Forschungszwecke geeignet, denn da mit diesen E-Mail-Adressen nie echte Kommunikation betrieben wurde, sind alle empfangenen E-Mails Spam und können beliebig verarbeitet werden.

Aus Sicht eines Versenders von Spam haben URL-Verkürzungsdienste die folgenden hilfreichen Eigenschaften:

- E-Mail Filtersysteme wie beispielsweise SpamAsasssin[20] fragen unter anderem verschieden Blacklisten (zum Beispiels die *Spamhaus Block List*[21]) für die Domains oder auch die vollständigen URLs (beispielsweise die *SpamCop URI Blacklist*[22]) der in E-Mails enthaltenen URLs an.
  Wenn die verkürzte URL eines Spam-Versenders in einer solchen Blackliste eingetragen ist, kann einfach die originale (lange) URL erneut verkürzt werden, eventuell mit einem anderen Dienst, und es können ohne Änderungen an der Infrastruktur weiter E-Mail versandt werden. Weiterhin ist es möglich, verkürzte URLs auf unterschiedliche Arten zu verfremden, sodass die URL zwar formal eine andere ist, beim Aufruf aber immer noch auf die gleiche Webseite des Versenders weiterleitet.
- Verkürzte URLs werden auch in regulären E-Mails verwendet, deshalb ist es für eine Spam-Erkennungssoftware nicht ratsam, dies als alleiniges Kriterium einzusetzen.
- Der Spam-Versender könnte auch in jede versandte E-Mail eine neue verkürzte URL einfügen, sodass ohne eigene Infrastruktur einfach über die Statistikfunktion des URL-Verkürzungsdienstes extrahiert werden kann, welcher Benutzer URLs aus Spam-E-Mails aufruft und damit besonders interessant als Empfänger ist. Über das DONBOT-Botnetz wurde diese Technik im Jahr 2009 bereits beobachtet.[23]

[19] http://koeln.ccc.de/schnucki

[20] http://spamassassin.apache.org

[21] http://wiki.apache.org/spamassassin/Rules/URIBL_SBL

[22] http://wiki.apache.org/spamassassin/Rules/URIBL_SC_SURBL

[23] http://www.m86security.com/labs/i/Spammers-Try-URL-Shortening-Services,trace.1038~.asp

- Empfänger von solchen Spam-E-Mails können das Ziel der verkürzten URL nicht erkennen und folgend der URL vielleicht alleine schon aus Neugier. Dies erhöht den Anteil der Empfänger, welche der enthaltenen URL folgen, was gut für den Spam-Versender ist.

Alle empfangene Spam-E-Mails wurden mittels eines Ruby-Skripts gelesen, verarbeitet, alle in `text/plain` und `text/html` Teilen enthaltenen URLs ausgelesen und in einer Datenbank gespeichert. Dabei wurden Anhänge mit gängigen Kodierungen (`base64`, `quoted-printable`) vorher dekodiert. Die folgenden beiden regulären Ausdrücke wurden zur Erkennung von URLs verwendet:

```
https?:\/\/[^\s"]+
www\.[^\s"]+
```

Bei auf den zweiten Ausdruck passenden Zeichenketten wurde das URL-Schema `http` angenommen. Anschließend wurden alle URLs noch von der Ruby-Klasse `URI::HTTP` verarbeitet und so ungültige aussortiert. Alle vom SCHNUCKI-Projekt empfangenen E-Mails zu verarbeiten dauerte 52 Tage auf einem System mit einer Quad-Core CPU (Intel Xeon, 2,33GHz). Auf diese Weise wurden 12.809.937 URLs aus den Spam-E-Mails extrahiert.

Werden die URLs nach dem Hostnamen gruppiert, sind in den 20 am häufigsten gefundenen Hostnamen bereits zwei Verkürzungsdienste: `xs.to` und `bit.ly`. Dies ist in Abbildung 2 dargestellt. Insgesamt wurden 35.647 verkürzte URLs in Spam-E-Mails gefunden, dies entspricht einem Anteil von 0.3 %. Tabelle 3 listet die Top-10 in den Spam-E-Mails gefundene URL-Verkürzungsdienste mit jeweils der Anzahl unterschiedlicher URLs und der Anzahl verschiedener E-Mails, in denen eine verkürzte URL dieses Dienstes enthalten war.

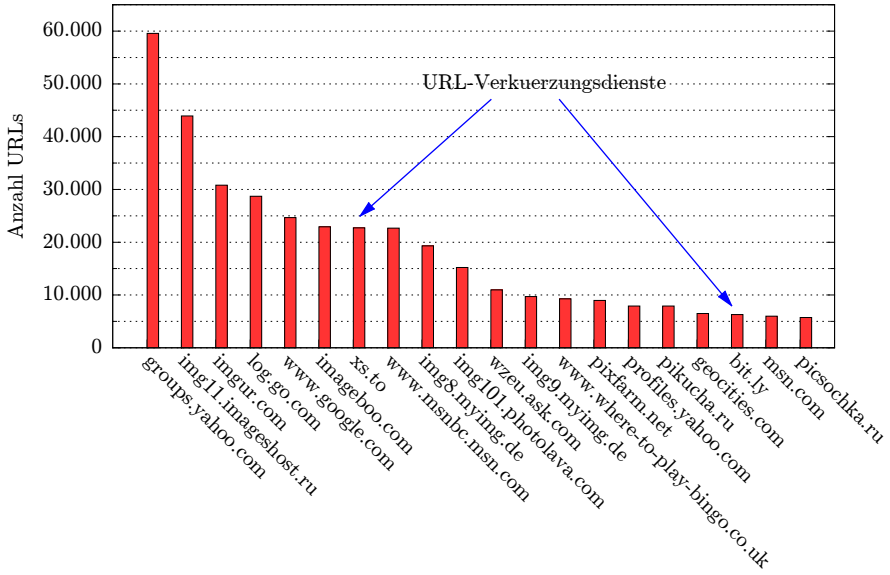


Abbildung 2: Die 20 am häufigsten gefundenen Hostnamen aus Spam-E-Mail-URLs

Alle gefundenen verkürzten URLs wurden automatisiert angefragt, und für die Verkürzungsdienste mit wenigstens 100 gefundenen URLs ausgewertet, ob die verkürzten URLs noch

Dienst	Anzahl URLs	Anzahl E-Mails
xs.to	22,711	27,382
bit.ly	6,372	9,068
tiny.cc	1,133	1,308
hurl.me	1,109	1,969
urlpass.com	683	1,012
snipurl.com	338	534
migre.me	251	568
su.pr	231	1,415
snipr.com	230	344
snurl.com	211	174

Tabelle 3: Die zehn häufigsten URL-Verkürzungsdienste aus Spam-E-Mails

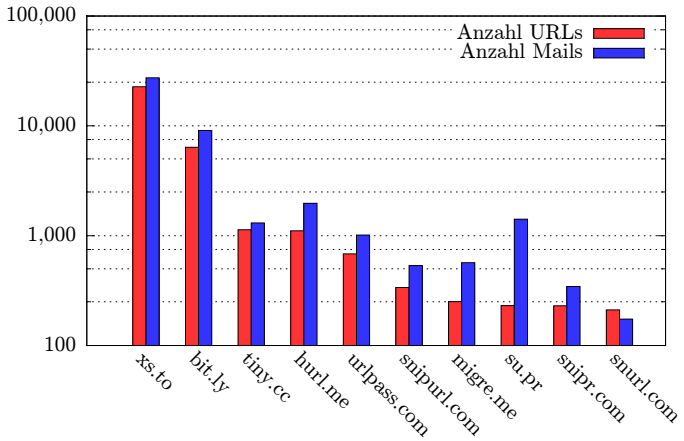


Abbildung 3: Die zehn häufigsten URL-Verkürzungsdienste aus Spam-E-Mails

funktionierten, oder als Spam erkannt wurden. Das Ergebnis ist in Tabelle 4 und Abbildung 4 dargestellt.

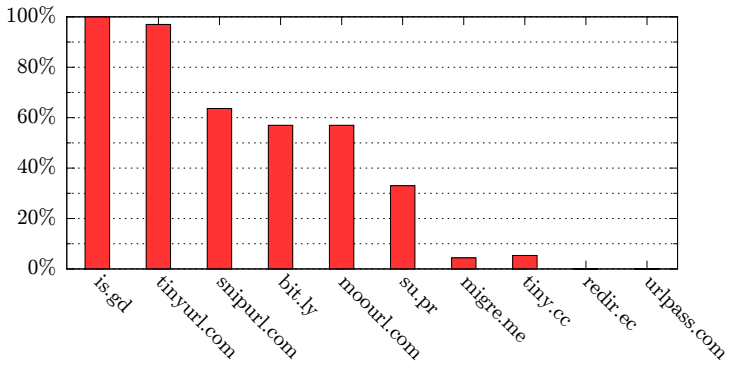


Abbildung 4: Spam-Erkennungsraten von URL-Verkürzungsdiensten

Dienst	Spam-Erkennungsrate	Hinweis
is.gd	100.00 %	
tinyurl.com	97.01 %	
snipurl.com	63.67 %	
bit.ly	57.00 %	
moourl.com	57.00 %	
su.pr	33.04 %	
migre.me	4.38 %	
tiny.cc	5.32 %	
redir.ec	0.00 %	
urlpass.com	0.00 %	
hurl.me	unbekannt	Dienst dysfunktional
xs.to	unbekannt	alle URLs gelöscht oder Dienst dysfunktional
9mp.com	unbekannt	alle URLs gelöscht oder Dienst dysfunktional
dwarfurl.com	unbekannt	Dienst dysfunktional

Tabelle 4: Spam-Erkennungsraten von URL-Verkürzungsdiensten

4.3 Malware und arglistige Verkürzungsdienste

Um herauszufinden, ob es arglistige Dienste gibt die sich wie in Abschnitt 3.1 beschrieben verhalten und Benutzer von alten oder unsicheren Browsern oder Betriebssystemen Malware ausliefern, wurde für jeden in den gesammelten Twitter-Nachrichten enthaltenen Verkürzungsdienst bis zu zwei URLs aus Twitter-Nachrichten zufällig ausgewählt. Dies ergab eine Liste von 319 verkürzten URLs von 187 verschiedenen Verkürzungsdiensten.

In jeder HTTP-Anfrage ist in der Regel die genaue Version des verwendeten Browsers sowie verschiedene Angaben über das verwendete Betriebssystem im HTTP-Header `User-Agent` enthalten. Ein andere Kombination aus Browser und Betriebssystem kann durch Setzen dieses Headers simuliert werden, beispielsweise mittels der Erweiterung *User-Agent Switcher* für den Browser Firefox. Für diese Erweiterung existieren verschiedene Konfigurationsdateien. Eine solche wurde für dieses Experiment heruntergeladen[24] und die dort enthaltenen `User-Agent` Zeichenketten für alle relevanten Betriebssysteme und Browser-Versionen extrahiert. Dies führte zu einer Liste von 83 unterschiedlichen Browsern und Betriebssystemen, zusätzlich wurde noch die leere Zeichenkette der Liste hinzugefügt.

Alle 319 verkürzten URLs wurden zwischen dem 26. Januar 2011, 19:00 Uhr CET und dem 27. Januar 2011, 03:00 Uhr CET mit jedem `User-Agent` je einmal angefragt und die folgenden Ergebnisse in einer Datenbank hinterlegt:

- HTTP-Antwortcode
- `Location`-Header (Ziel-URL)
- `Set-Cookie`-Header

Alle Dienste, die nach zehn Sekunden keine Antwort ausgelieferten, wurden ignoriert. Von den 319 URLs liefern 292 URLs immer den gleichen HTTP-Antwortcode und `Location`-Header aus. Diesen Diensten kann damit kein arglistiges Verhalten nachgewiesen werden. Für elf URLs wurden zwei verschiedene Kombinationen aus Antwortcode und `Location`-Header beobachtet. Dabei konnten fast alle Antwortcodes auf Fehler zurückgeführt werden.

[24] `http://techpatterns.com/downloads/firefox/useragentswitcher.xml`

Der Dienst `ri.ms` reagierte auf eine sehr interessante Weise. Die einzige von diesem Dienst in den Twitter-Nachrichten gefundene URL (`http://ri.ms/5wgv6`) lieferte für alle Anfragen genauso viele verschiedene URLs im `Location`-Header aus. Davon waren 82 URLs unterschiedlich, da auf eine Vorschauseite umgeleitet wurde, deren URL einen Zugriffszählerstand enthielt, welcher nach jeder Anfrage erhöht wurde. Beispielsweise wurde eine Anfrage auf die folgende URL weitergeleitet:

```
http://tinyarro.ws/preview.php
    ?page=http%3A%2F%2Fwww.cin.hdmais.com.br
        %2Fcomponent%2Fcontent%2Farticle%2F
    &count=248
```

Die Vorschauseite enthielt dabei einen in JavaScript implementierten Countdown, nach dessen Ablauf, auch mittels JavaScript, der Browser des Benutzers auf die Ziel-URL weitergeleitet wurde.

Drei Anfragen wurden jedoch mit einer direkten HTTP-Umleitung auf die Ziel-URL beantwortet: Die Anfragen mit leerem `User-Agent`-Header (leere Zeichenkette) sowie mit der Zeichenkette des Browsers *Dillo*[25] in der Version 2.0. Es ist zu vermuten, dass dieser URL-Verkürzungsdienst bei der Beantwortung einer Anfrage eine Positivliste von Browsern zurate zieht um zu entscheiden, ob der anfragende Browser JavaScript unterstützt. Ist dies der Fall, wird eine Umleitung auf die erwähnte Vorschauseite mit dem JavaScript-basierten Countdown ausgeliefert, andernfalls eine direkte Umleitung auf die Ziel-URL.

Dieses Verhalten ist nicht bösartig, aber es zeigt, dass URL-Verkürzungsdienste existieren, die abhängig vom verwendeten Browser anders reagieren.

4.4 Überwachung

Bei der Abfrage verkürzter URLs (siehe Abschnitt 4.3) mit unterschiedlichen Browsern wurden pro verkürzte URL 85 HTTP-Anfragen ausgeführt und in einer Datenbank unter anderem der vom Server zurückgelieferte HTTP-Header `Set-Cookie` in einer Datenbank gespeichert.

Von 319 URLs von 187 URL-Verkürzungsdiensten wurden insgesamt 84 Cookies gesetzt. Von diesen Cookies waren 36 Session-Cookies, welche beim Beenden des Browsers gelöscht werden. Die anderen 48 wurden von den Diensten persistent im Browser hinterlegt, diese Dienste können also potentiell Benutzer über den Gültigkeitszeitraum dieser Cookies verfolgen und überwachen.

Um entscheiden zu können, wie eindeutig ein von einem Dienst gesetztes persistentes Cookie ist, wurde für jeden Cookie-Wert ein Quotient Q berechnet, indem die Anzahl der verschiedenen Werte für das Cookie durch die Anzahl aller Anfragen geteilt wurde. Ist Q nahe bei 1 gibt es viele unterschiedliche Werte für das Cookie, ein einzelner Wert ist also sehr eindeutig einer einzigen Anfrage zuzuordnen. Ist Q nahe bei 0, kann dies auf ein generisches Konfigurations-Cookie hindeuten, beispielsweise ein boolescher Wert, ob eine Vorschauseite angezeigt werden soll (`preview=true`). Ein solcher Quotient kann aber auch dadurch verursacht werden, dass alle Anfragen von einer einzigen IP-Adresse aus durchgeführt wurden und

[25] `http://www.dillo.org/`

der Dienst diese, eventuell kodiert, in einem Cookie ablegt. In diesem Fall ist, obwohl Q nahe bei 0 liegt, die Überwachung des Benutzers möglich.

Beispielsweise wurde das Cookie namens `u` vom Verkürzungsdienst `b23.ru` in jeder Anfrage auf den Wert 1 gesetzt. Dieses Cookie kann nicht zum Verfolgen von Benutzern verwendet werden, der Wert von Q ist 0,059. Auf der anderen Seite wurden für das Cookie namens `fwsid` des Dienstes immer unterschiedliche Zeichenketten von 32 Zeichen empfangen. Der Wert von Q für dieses Cookie ist 1, es kann also zum Verfolgen von Benutzern verwendet werden.

Gültigkeitsdauer und Q-Wert aller von Verkürzungsdiensten gesetzten persistenten Cookies sind in Abbildung 5 dargestellt. Daraus ist zu erkennen, dass die einzigen zwei Dienste aus der Liste der Top-10 Dienste `bit.ly` und `dlvr.it` sind, dabei setzt `bit.ly` ein für ein halbes Jahr gültiges Cookie, welches zusätzlich einen Q-Wert von 1 hat und damit gut zum Verfolgen von Benutzern geeignet ist.

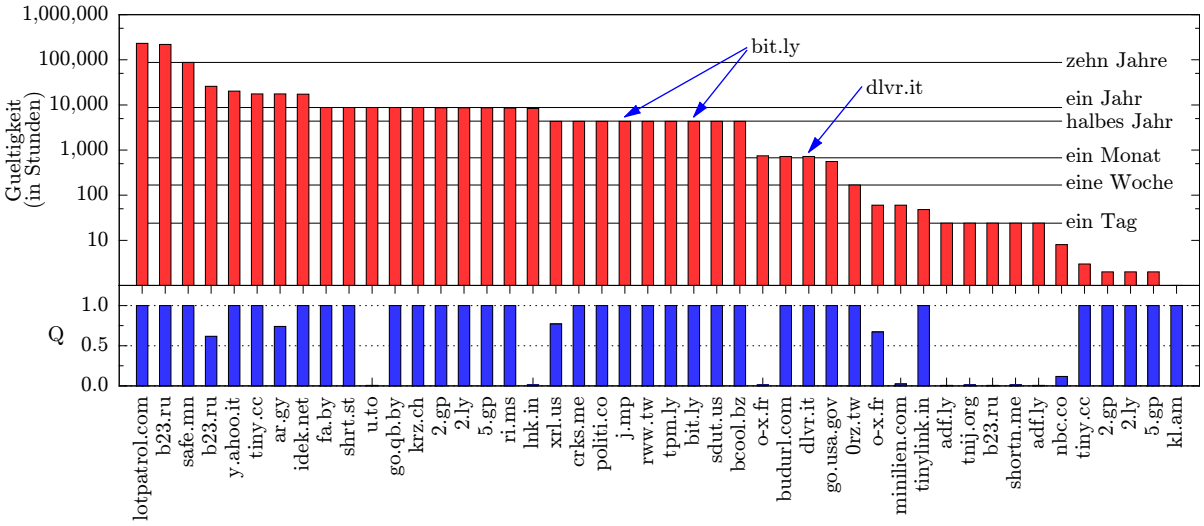


Abbildung 5: Gültigkeit persistenter Cookies von URL-Verkürzungsdiensten

4.5 Geheime URLs

Wenn URL-Verkürzungsdienste verwendet werden, erhält eine dritte Partei nicht nur Kenntnis von allen Zugriffen, sondern auch von den zu verkürzenden URLs. Insbesondere werden alle in über Twitter versandten privaten Nachrichten (*Direct Messages*) enthaltenen URLs mittels des Twitter-eigenen URL-Verkürzungsdienstes zwangsverkürzt.

4.5.1 URL-Verkürzungsdienste enumerieren

Verkürzte URLs sind prinzipiell enumerierbar, denn aus Platzgründen ist innerhalb der URLs eines Dienstes nur ein kleiner Teil der URL variabel. Beispielsweise verwendet der Dienst `bit.ly` pro verkürzter URL einen eindeutigen Pfad, dies ist in der URL `http://bit.ly/BlE9j` die der aus sechs Zeichen bestehende Text `BlE9j`. Dies ist die am häufigsten verwendete

Methode, viele individuelle URLs auszugeben, die möglichst wenig Zeichen verbrauchen. Alle in Abschnitt 4.1 bestimmten Top-10 URL-Verkürzungsdienste verwenden diese Methode.

Wie bereits in Abschnitt 3.3 beschrieben gibt es Dienste, die so verwendet werden können, dass der Zugriff auf von Benutzern hochgeladene Daten allen Besuchern gestattet wird, welche die vom Dienst ausgegebene URL zu diesen Daten kennen. Im Normalfall enthält die URL dabei einen hinreichend nicht-vorhersagbaren Teil, beispielsweise eine zufällig generierte Zeichenkette ausreichenden Länge, sodass die URL nicht vorhergesagt oder geraten werden kann. Wenn diese URLs jedoch verkürzt werden, können sie über die verkürzte URL relativ leicht enumeriert werden.

Für einen Angreifer, der an kürzlich verkürzten URLs interessiert ist, ist es interessant vor dem Enumerieren von verkürzten URLs zunächst zu analysieren, wie die jeweiligen Dienste die variablen Teile von neu verkürzten URLs vergeben. Im Folgenden wird die Struktur dieser variablen Teile der URLs der Top-10 URL-Verkürzungsdienste untersucht und anschließend ein relevant großer Teilbereich angefragt.

4.5.2 Struktur der häufig verwenden Dienste

Um die Struktur der variablen Teile von verkürzten URLs der Dienste zu untersuchen, wurden alle in den in Abschnitt 4.1 beschrieben Twitter-Nachrichtensamples gefundenen verkürzten URLs eines Dienstes verwendet. Dabei wurden für beide Zeiträume jeweils einzeln alle variablen Teile aller verkürzten URLs des Dienstes extrahiert. Anschließend wurde die relative Häufigkeit verwendeten Zeichen an jeder Position untersucht. Um dies auszuwerten und grafisch anschaulich darzustellen, wurden Heatmaps[26] verwendet. Dabei werden in einem Graphen die verwendeten Zeichen auf der X-Achse und die Position auf der Y-Achse aufgetragen. An den Schnittpunkten im Graphen wird ein Quadrat gezeichnet, wobei die Farbe der relativen Häufigkeit des Zeichens an dieser Position entspricht. Bei der Untersuchung konnte festgestellt werden, dass bei Verwendung einer logarithmischen Skala für die Farbkodierung das anschaulichste Ergebnis erzielt werden konnte.

Die untersuchten Dienste lassen sich anhand dieser Untersuchung grob in drei Gruppen unterteilen:

- Sequentiell/teilsequentiell: `bit.ly`, `ow.ly`, `is.gd`, `migre.me`
- Sequentiell mit Auslassen: `tinyurl.com`, `dlvr.it`, `dld.bz`, `lnk.ms`
- Zufällig: `goo.gl`, `t.co`

Im Folgenden wird für je einen Dienst aus Gruppe der variable Teil der verkürzten URLs dieses Dienstes untersucht. Die Heatmaps der verbleibenden Dienste finden sich in Anhang 6.

Für die erste Gruppe der sequentiell/teilsequentiell vergebenen variablen Teile der verkürzten URLs wird der Dienst `bit.ly` untersucht. Die Heatmaps für beide Nachrichtensamples sind in Abbildung 6 dargestellt. Der variable Teil in URLs dieses Dienstes besteht aus sechs Zeichen, Groß- und Kleinbuchstaben sowie Zahlen. Dieser Dienst vergibt das erste Zeichen sequentiell, im ersten Sample haben die Zeichen `9` und `a` bis `d` eine viel höhere Häufigkeit von 13 bis

[26] `http://de.wikipedia.org/wiki/Heatmap`

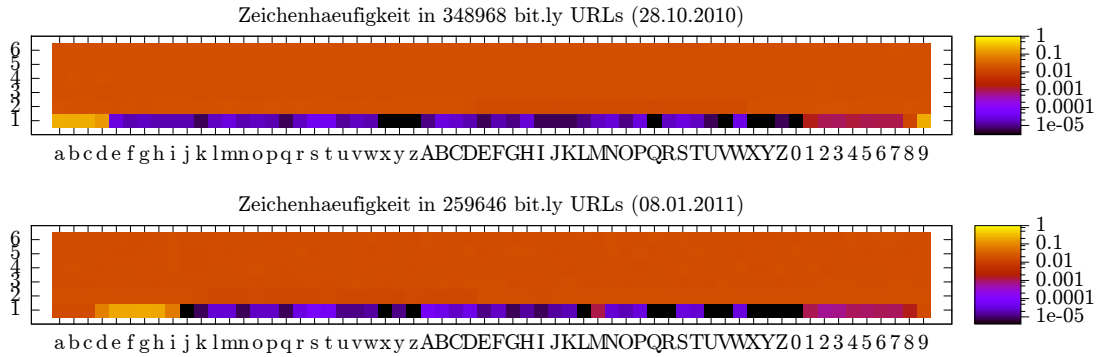


Abbildung 6: Zeichenhäufigkeiten in `bit.ly`-URLs

20 % als die umgebenden Zeichen (unter 1 %). Im zweiten Nachrichtensample, welches 72 Tage später gesammelt wurde, sind die Zeichen `d` bis `i` die häufigsten Zeichen an der ersten Position. Dabei haben die Zeichen `d` und `i` eine Häufigkeit von 8 %, die Zeichen `e` bis `h` jeweils 20 %. Für die Zeichen an den Positionen zwei bis sechs ist keine Struktur erkennbar, in beiden Nachrichtensamples liegen die relativen Häufigkeiten aller Zeichen bei 1,6 %, was einer Gleichverteilung entspricht. Insgesamt ist zu vermuten, dass zumindest das `bit.ly` zumindest das erste Zeichen des variablen Teils in verkürzten URLs sequentiell vergibt.

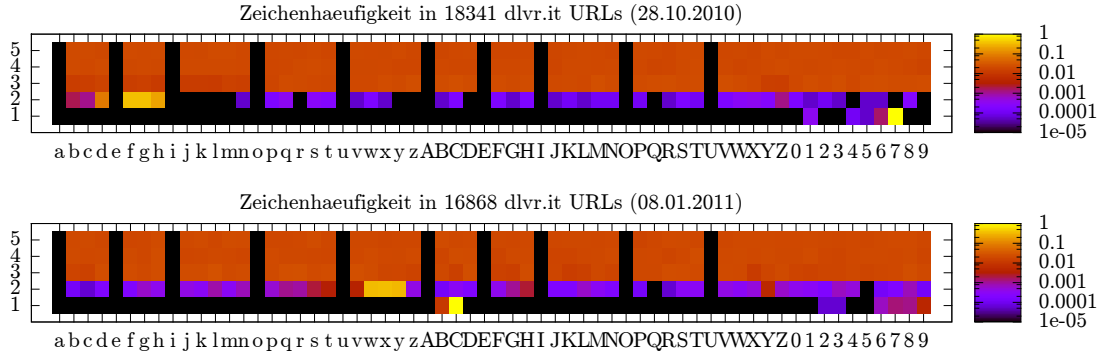


Abbildung 7: Zeichenhäufigkeiten in `dlvr.it`-URLs

Aus der zweiten Gruppe wird die Analyse für den Dienst `dlvr.it` vorgestellt, die Heatmaps dazu sind in Abbildung 7 dargestellt. Dieser Dienst verwendet für den eindeutigen Teil verkürzter URLs fünf Zeichen Groß- und Kleinbuchstaben sowie Zahlen. Es ist zu erkennen, dass im ersten Nachrichtensample die variablen Teile der URLs bevorzugt mit den Zahlen `6` und `7`, im zweiten Sample jedoch mit `8` und `9` sowie `A` und `B` beginnen. Das zweite Zeichen ist im ersten Sample mit hoher Wahrscheinlichkeit `d` oder `f` bis `h`, im zweiten Sample `w` bis `y`. Es ist daher zu vermuten, dass dieser Dienst verkürzte URLs sequentiell vergibt. Dabei werden konsequent alle Zeichen ausgelassen, die als andere Zeichen gelesen werden können, also `i`, `l`, `o`, `I`, `L`, `O` sowie die Zahlen `1` und `0`.

Für die dritte Gruppe wird der Twitter-eigene URL-Verkürzungsdienst `t.co` analysiert. Dessen verkürzte URLs bestehen aus sieben Zeichen Groß- und Kleinbuchstaben sowie Zahlen. Die zugehörige Heatmap ist in Abbildung 8 dargestellt. Es ist keine Struktur erkennbar, alle Zeichen kommen mit der bei einer Gleichverteilung zu erwartenden Häufigkeit von 1,6 % vor.

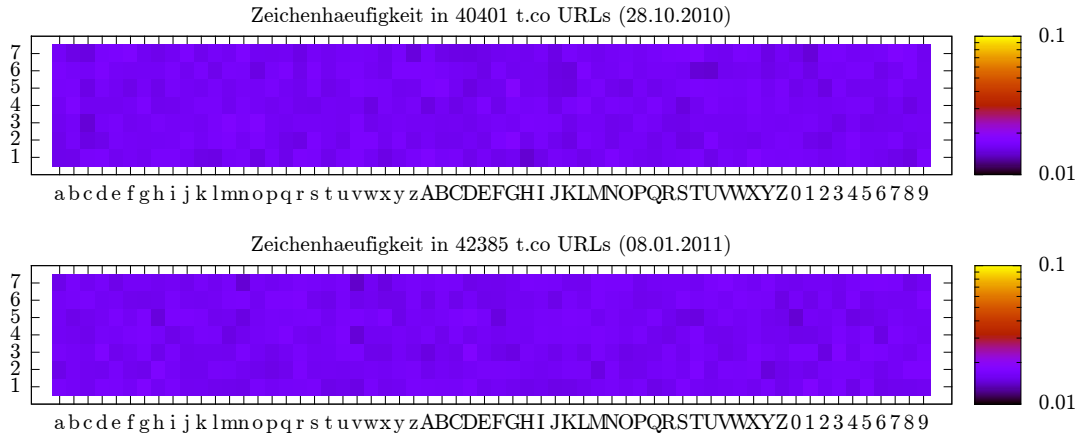


Abbildung 8: Zeichenhäufigkeiten in `t.co`-URLs

4.5.3 Dienste enumerieren

URL Vorlage	Anzahl URLs	Zeichen
`http://bit.ly/b22***`	238.328	{a...z, A...Z, 0...9}
`http://tinyurl.com/23jX***` mit `X` aus {k,l,m,n,o}	233.280	{a...z, 0...9}
`http://t.co/iDAL***`	238.328	{a...z, A...Z, 0...9}
`http://goo.gl/U***`	238.328	{a...z, A...Z, 0...9}
`http://ow.ly/30***`	238.328	{a...z, A...Z, 0...9}
`http://dlvr.it/7f***`	238.328	{a...z, A...Z, 0...9}
`http://is.gd/go***`	238.328	{a...z, A...Z, 0...9}
`http://lnk.ms/Dy***`	238.328	{a...z, A...Z, 0...9}
`http://dld.bz/3***`	238.328	{a...z, A...Z, 0...9}
`http://migre.me/1P***`	238.328	{a...z, A...Z, 0...9}

Tabelle 5: Bereiche der zu enumerierenden verkürzten URLs

Basierend auf der in Abschnitt 4.5.2 vorgestellten Analyse wurde für jeden Dienst ein Bereich von 233,000–238,000 verkürzten URLs ausgewählt (siehe Tabelle 5) und alle URLs in diesem Bereich angefragt. Da der Dienst `tinyurl.com` Groß- und Kleinbuchstaben gleich behandelt, wurden nur Kleinbuchstaben angefragt.

Lediglich der Dienst `goo.gl` ließ nicht beliebig viele Anfragen an verkürzte URLs zu und lieferte bei zu vielen Anfragen in kurzer Zeit keine Weiterleitung, sondern lediglich eine Fehlermeldung aus. Durch Begrenzung der Anfragen und Einführen einer Wartezeit von 200 ms zwischen zwei Anfragen konnte dies umgangen werden. Insgesamt dauerte es 27 Stunden, die verkürzten URLs für `goo.gl` abzufragen.

Für alle gültigen verkürzten URLs wurde zusätzlich die entsprechende lange URL selbst, sowie die für diese URL zuständige Datei `robots.txt`[27] abgefragt. Anhand des Inhalts konnte herausgefunden werden, welche der langen URLs im Index der Suchmaschine Google enthalten sein und damit über Google gefunden werden könnten. Ist dies nicht der Fall, wird diese lange URLs als geheim eingestuft. Das Ergebnis dieser Untersuchung ist in Tabelle 6 dargestellt.

[27] `http://de.wikipedia.org/wiki/Robots.txt`

Dienst	Gültige URLs	Geheime URLs	Anteil
bit.ly	191.528	11.919	6,2 %
tinyurl.com	169.254	6.105	3,6 %
t.co	31	0	0 %
goo.gl	190.685	8.546	4,5 %
ow.ly	236.853	5.173	2,2 %
dlvr.it	235.870	11.162	4,7 %
is.gd	235.564	5.810	2,5 %
lnk.ms	237.653	51	0,02 %
dld.bz	156.604	6.675	4,3 %
migre.me	233.202	351	0,2 %

Tabelle 6: Anteil geheimer URLs in angefragten verkürzten URLs

Im Anschluss wurde die Liste aller gefundenen langen URLs manuell durchsucht. Durch die Suche nach dem Pfad `/admin/` sowie dem HTTP-Statuscode 200 konnten 153 administrative Webseiten gefunden werden. Die Suche nach dem Hostnamen `docs.google.com`, dem Query-Bestandteil `authkey=` sowie dem HTTP-Statuscode 200 wurden 71 öffentlich zugängliche Dokumente gefunden. Darunter waren verschiedene Archive eindeutig privater Fotos, eine Seminararbeit, die Liste der Einnahmen und Ausgaben einer Firma, die Lebensläufe zweier Personen sowie die Liste der Namen, Adressen und Telefonnummern einer Kindergartengruppe aus Lindlar aus der Nähe von Köln.

4.5.4 Geheime URLs verkürzen

In Abschnitt 3.3 wurde bereits das Risiko erwähnt, dass Administratoren sowie durch Enumerieren auch unbeteiligte Dritte Kenntnis von verkürzten, geheimen URLs erhalten können. Um dies zu prüfen wurde ein Webserver aufgesetzt und für die Domains `fd0.me` und `www.fd0.me` konfiguriert. Wenn eine der Domains ohne weiteren Pfad aufgerufen wird, liefert der Webserver eine Seite mit Informationen zu diesem Experiment sowie Kontaktdaten aus. Enthält der Pfad jedoch drei oder mehr Schrägstriche (Beispielsweise `/test1/test2/test3`), wird eine Webseite ähnlich der in Abbildung 9 dargestellten ausgeliefert. Diese Webseite enthält die aktuelle Uhrzeit, Informationen zur Anfrage, beispielsweise den verwendeten Browser und einen möglicherweise übermittelten Referrer. Weiterhin ist ein Hinweis auf die Kontaktinformationen sowie ein für dieses Experiment generiertes Passwort (`knycsUpjaj8`) enthalten. Zu Beginn des Experiments wurde das Passwort von Google nicht gefunden.

Ziel dieses Experiments war es, herauszufinden, welche Zugriffe auf eine URL auf Verkürzungsdienste zurückzuführen sind, wenn die verkürzte URL nie veröffentlicht wird. Dabei können diese Zugriffe von Administratoren oder auch unbeteiligten Dritten stammen. Zusätzlich soll untersucht werden, ob durch Verkürzen einer URL Suchmaschinen diese URL anschließend finden. Dies kann beispielsweise geschehen, wenn der Verkürzungsdienst beispielsweise die Liste der 100 zuletzt verkürzten URLs veröffentlicht.

Im Folgenden wurde für alle 527 allgemeinen URL-Verkürzungsdienste der in Abschnitt 4.1 zusammengestellten Liste jeweils zwei URLs nach dem folgenden Schema erzeugt:

- `http://fd0.me/secret/SERVICE/TIMESTAMP`
- `http://www.fd0.me/blog/archive/2011/01/14/index.php?article=SERVICE#TIMESTAMP`

Abbildung 9: Ausgelieferte Informationsseite bei Aufruf einer geheimen URL

Die Zeichenketten `SERVICE` und `TIMESTAMP` wurden dabei jeweils durch einen Text von acht Zeichen Länge ersetzt, welcher eindeutig den Dienst und die aktuelle Zeit darstellt. Die verwendeten Texte wurden erzeugt, indem die ersten vier Byte des SHA1-Hashes des Dienstnamens beziehungsweise der aktueller Uhrzeit und des Datums in hexadezimal dargestellt wurden. Die soll dazu dienen, bei einem Zugriff exakt den Dienst sowie den Zeitpunkt der Verkürzung der angeben zu können, ohne dass jedoch diese Daten offensichtlich in der URL enthalten sind. Nach dem ersten Schema erzeugte URLs (im Folgenden: *geheime* URLs) sollen die Neugier von Administratoren erwecken, das zweite Schema soll möglichst unverdächtig aussehende URLs (im Folgenden: *Blog*-URLs) erzeugen. Beispielhaft sind im folgenden zwei URLs aufgeführt:

- `http://fd0.me/secret/a0df29ac/bb42ce8b`
- `http://www.fd0.me/blog/archive/2011/01/14/index.php?article=69e325eb#a5a6c61c`

Für jeden der 527 allgemeinen URL-Verkürzungsdienste wurde zwischen dem 11. und 13. Januar 2011 manuell versucht, eine geheime URL zu verkürzen. Dysfunktionale Verkürzungsdienste sowie Dienste, die nur nach Anmeldung zugänglich waren, wurden dabei aussortiert. Wenn möglich, wurde ein Dienst mit mehreren unterschiedlichen Namen nur einmal verwendet (beispielsweise gehören `bit.ly` und `j.mp` zum gleichen Dienst). Die Blog-URLs wurden am 14. Januar 2011 verkürzt. Insgesamt wurden 326 geheime und 208 Blog-URLs von 225 Diensten verkürzt.

Am 2. Februar 2011, etwa zwei Wochen nach der Verkürzung, wurde die Logdatei des Webservers ausgewertet, welche Zugriffe auf diese verkürzten URLs stattgefunden haben. Anfragen bis zu 60 Sekunden nach Senden an den Verkürzungsdienst wurden als automatische Anfragen klassifiziert und sind in der folgenden Analyse nicht enthalten.

Insgesamt wurden 497 Zugriffe auf 49 geheime und 31 Blog-URLs registriert. Die Einträge in der Logdatei wurden manuell inspiziert, insbesondere der HTTP-Referrer ist dabei interessant. Es wurden neun Zugriffe registriert, bei denen im Referrer die URL der nichtöffentlichen

Suchmaschine	Geheime geheime URLs	Über Dienst	Anzahl Blog-URLs	Über Dienst	Summe URLs
Google	7	ikr.me o-x.fr p.ly re.p.ly siteo.us yep.it	8	go2.me ikr.me o-x.fr p.ly qru.ru re.p.ly siteo.us yep.it	15
Yahoo	5	0rz.tw j2j.de o-x.fr siteo.us yep.it	7	j2j.de p.ly qik.li quud.com re.p.ly siteo.us yep.it	13
Baidu	1	orz.se	1	orz.se	2

Tabelle 7: Anzahl von Suchmaschinen besuchte URLs

Administrationsschnittstelle des jeweiligen Verkürzungsdienstes enthalten war. Dies bedeutet, dass neun Administratoren von URL-Verkürzungsdiensten die URLs manuell aufgerufen haben.

Der Autor wurde während des Experiments von vier Administratoren anderer Verkürzungsdienste angeschrieben, die jeweils Interesse an den Ergebnissen des Experiments interessiert waren. Auf die Frage, warum sie mit ihrem Dienst verkürzte URLs aufrufen, war in jedem Fall die Antwort, dass ihr jeweiliger Dienst klein sei und regelmäßig von Spam-Versendern missbraucht werden würde, sodass sie „verdächtige“ URLs manuell prüfen würden. Insgesamt haben Administratoren von 13 unterschiedlichen URL-Verkürzungsdiensten URLs des Experiments aufgerufen.

In der Logdatei des Webservers fanden sich auch Zugriffe von den automatischen Crawlern der Suchmaschinen Google, Yahoo und Baidu. Tabelle 7 stellt dar, über welche Dienste die abgerufenen URLs zu den Suchmaschinen gelangten. Eine Suche nach dem in der Webseite eingebetteten Passwort ergab, dass die Seiten tatsächlich im Index der Suchmaschinen enthalten waren.

4.6 Latenz und Verfügbarkeit

Bei jedem Zugriff auf eine verkürzte URL wird immer zunächst eine Verbindung zum Verkürzungsdienst aufgebaut, eine HTTP-Anfrage gesendet und auf die Antwort gewartet. Um die dabei auftretende Verzögerung des Aufrufs einer verkürzen URL und gleichzeitig die Verfügbarkeit des Dienstes zu messen, wurde für die Top-10 URL-Verkürzungsdienste über einen Zeitraum von zwei Wochen vom 12. bis 26. Januar 2011.

Um lokale Probleme zu umgehen, wurden zwei vollständig getrennte Server für die Messung verwendet. Der erste, im folgenden *Server 1* genannt, war über die RWTH Aachen über Gigabit Ethernet an das Internet angeschlossen. Der zweite, *Server 2* stand in einem von der

Hetzner AG betriebenen Rechenzentrum in Falkenstein bei Chemnitz. Server 1 hatte dabei einen Ausfall der Internetverbindung am 17. Januar 2011, von 13:05 bis 14:10 Uhr CEST.

Gemessen wurden zum einen die Netzwerk-Latenz mittels ICMP-PING. Dabei wurden Pakete als verloren klassifiziert, wenn nach 20 Sekunden keine Antwort empfangen wurde. Zum anderen wurde die HTTP-Latenz gemessen, indem eine HTTP-Anfrage nach einer echten, gültigen Verkürzten URL des Dienstes durchgeführt wurde und die Messung erst beendet wurde, nachdem die Antwort auf diese Anfrage korrekt empfangen wurde. In dieser Messung wird auch die Zeit des Verbindungsaufbaus inklusive DNS-Auflösung des Namens berücksichtigt. Die Messungen wurden mit der Software *SmokePing*[28] durchgeführt. Die durchschnittliche ICMP- und HTTP-Latenz sowie die maximal beobachteten Werte sind in Abbildung 10 dargestellt. In Tabelle 8 im Anhang sind die vollständigen Ergebnisse aufgelistet.

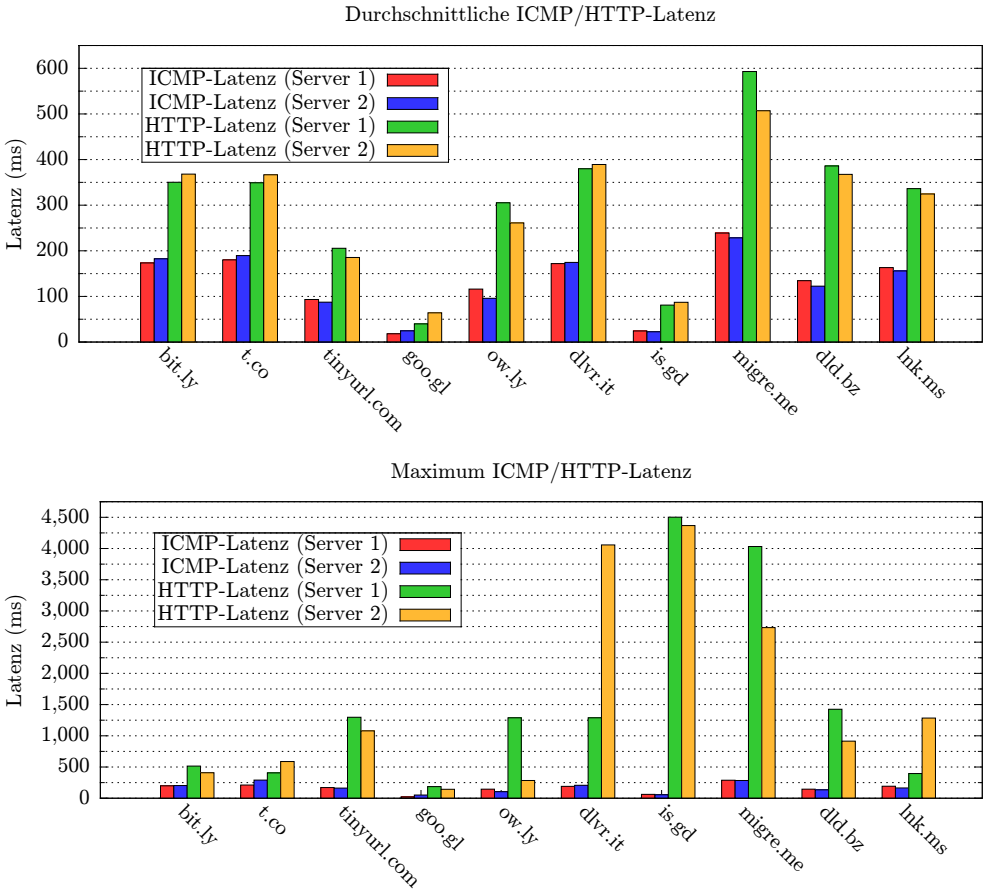


Abbildung 10: Average and maximum ICMP/HTTP latency of all tested services

[28] http://oss.oetiker.ch/smokeping/

5 Fazit

In diesem Artikel wurden theoretische Risiken der Verwendung von URL-Verkürzungsdiensten beschrieben sowie praktisch durchgeführte Experimente zu diesen Risiken und deren Ergebnisse vorgestellt. Die wichtigsten Ergebnisse werden im Folgenden zusammengefasst.

Es konnte kein Hinweis gefunden werden, dass es URL-Verkürzungsdienste gibt, die abhängig von der verwendeten Browser-Version Malware ausliefern. Es wurde jedoch gezeigt, dass es Dienste gibt, die abhängig vom verwendeten Browser unterschiedliche Antworten ausliefern.

URL-Verkürzungsdienste werden in E-Mail-Spam verwendet, dabei ist die Spam-Erkennungsrate vieler Dienste nicht optimal. Geheime URLs konnten durch Enumerieren von Verkürzungsdiensten gefunden werden. Es wurde gezeigt, dass Verkürzen von geheimen URLs unter Umständen dazu führt, dass Administratoren und Suchmaschinen Kenntnis von diesen URLs erhalten.

Die Überwachungsmöglichkeiten mittels persistenten HTTP-Cookies wurde untersucht. Dabei wurde festgestellt, dass viele Dienste eindeutige Cookies mit langer Laufzeit in den Browsern von Besuchern hinterlegen, sodass diese überwacht und einzelne Anfragen einander zugeordnet werden könnten.

Schlussendlich wurde die Verfügbarkeit und Latenz der Top-10 Dienste für einen Zeitraum von zwei Wochen gemessen. Dabei wurden teilweise deutliche Latenzen gemessen.

Insgesamt ist die Verwendung von Verkürzungsdiensten mit Risiken verbunden. Einige dieser Risiken wurden hier näher untersucht und es konnte gezeigt werden, dass diese eine reale Gefahr für Benutzer darstellen. Insbesondere konnte gezeigt werden, dass es nicht ratsam ist, geheime URLs zu verkürzen.

6 Anhang: Heatmaps URL-Verkürzungsdienste

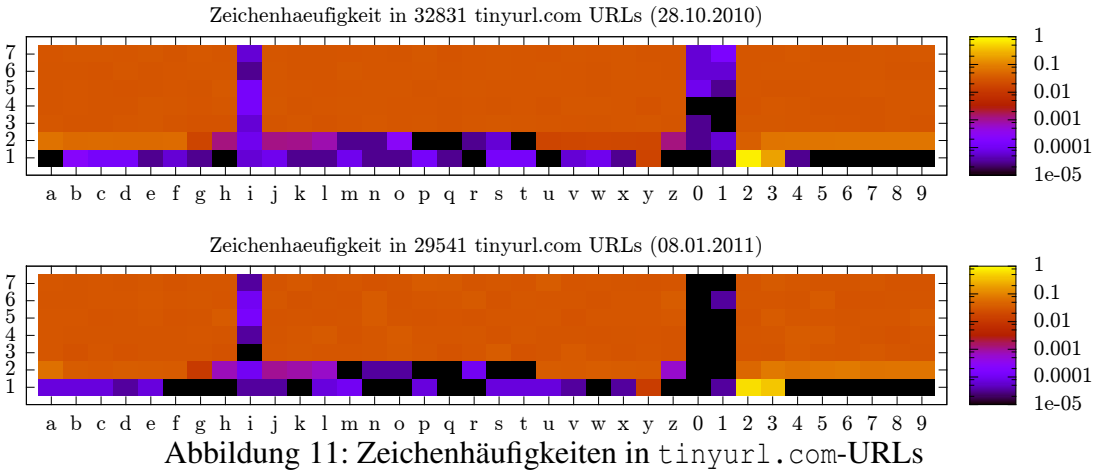


Abbildung 11: Zeichenhäufigkeiten in `tinyurl.com`-URLs

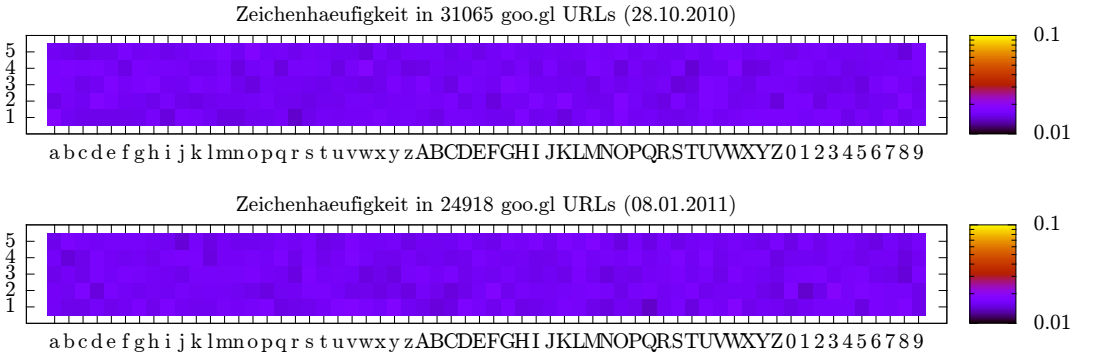


Abbildung 12: Zeichenhäufigkeiten in `goo.gl`-URLs

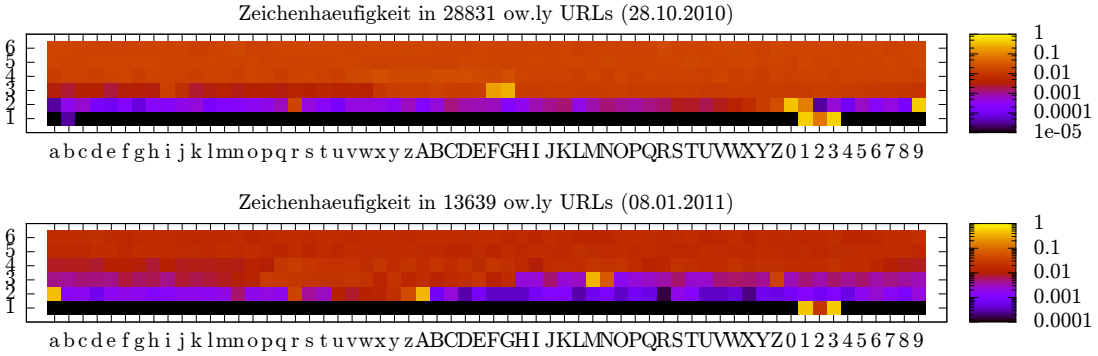
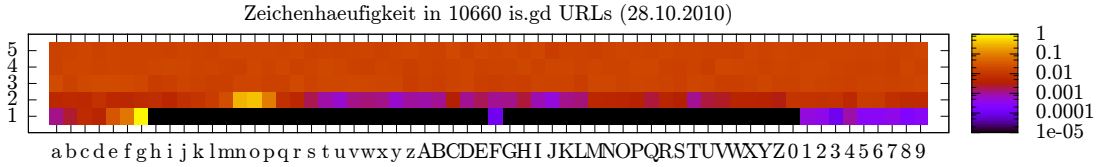


Abbildung 13: Zeichenhäufigkeiten in `ow.ly`-URLs



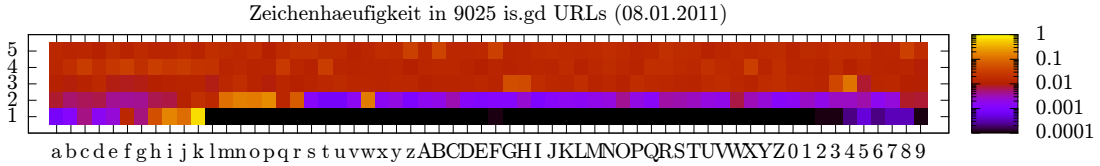
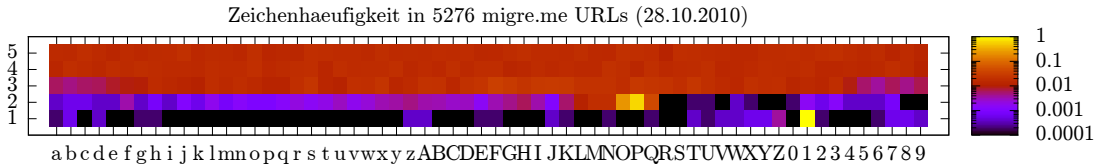


Abbildung 14: Zeichenhäufigkeiten in `is.gd`-URLs



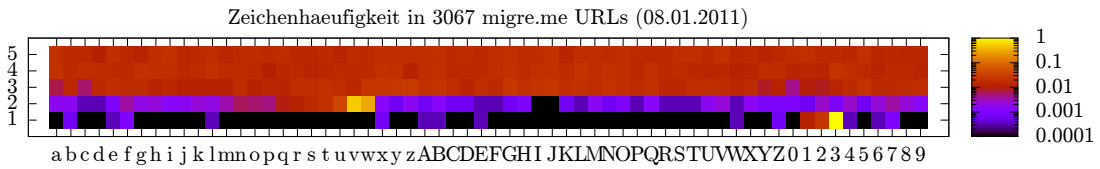
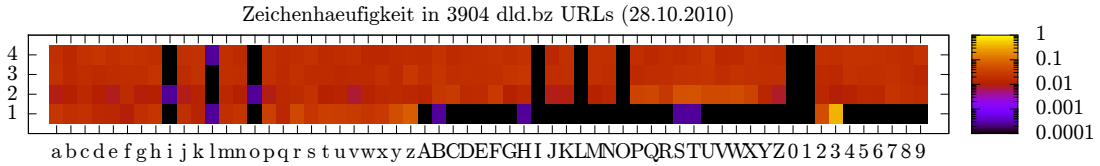


Abbildung 15: Zeichenhäufigkeiten in `migre.me`-URLs



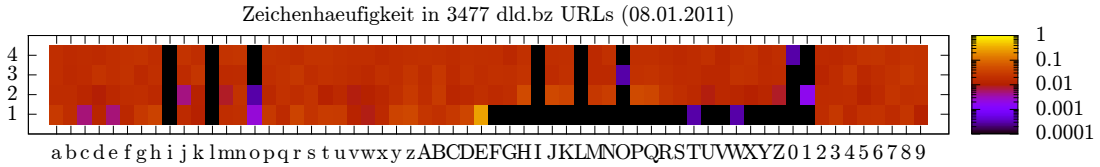
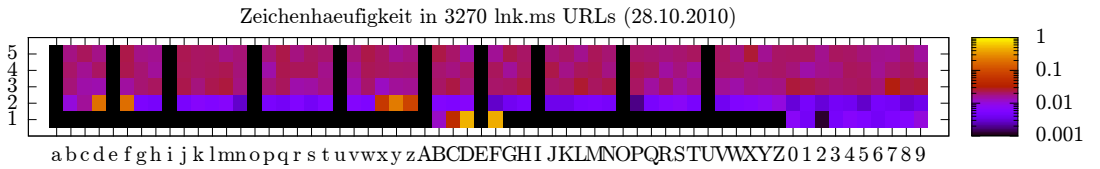


Abbildung 16: Zeichenhäufigkeiten in `dld.bz`-URLs



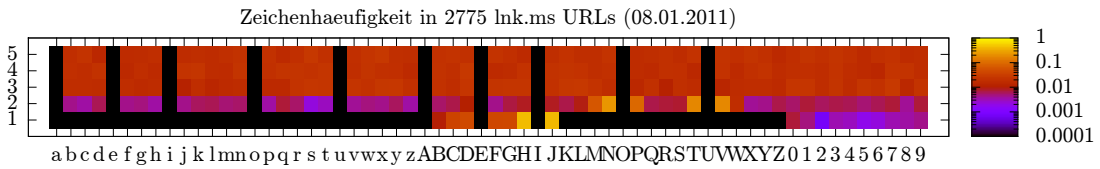


Abbildung 17: Zeichenhäufigkeiten in `lnk.ms`-URLs

7 Anhang: Latenz und Verfügbarkeit von URL-Verkürzungsdiensten

Dienst	ICMP Latenz von Server 1				ICMP Latenz auf Server 2			
	Min (ms)	Average (ms)	Max (ms)	Average Loss (%)	Min (ms)	Average (ms)	Max (ms)	Average Loss (%)
bit,ly	173,44	166,58	199,36	0,47	182,55	176,60	202,46	0,35
t,co	180,03	168,46	210,77	0,48	189,61	176,73	290,08	0,34
tinyurl,com	93,29	12,09	171,90	0,47	87,47	13,04	162,03	0,16
goo,gl	18,24	5,17	20,64	0,35	24,56	16,47	50,42	0,02
ow,ly	116,01	105,45	144,54	0,46	95,97	93,24	104,44	0,00
dlvr,it	171,93	168,59	190,85	0,65	174,55	173,02	206,33	0,57
is,gd	24,20	23,53	61,19	0,40	22,47	20,64	54,81	0,45
migre,me	238,93	210,73	288,40	2,22	228,48	215,20	283,13	0,29
dld,bz	134,43	131,33	144,72	0,41	122,43	119,48	134,54	0,10
lnk,ms	163,24	152,61	192,05	0,45	155,79	153,02	164,21	0,00

Dienst	HTTP Latenz von Server 1				HTTP Latenz auf Server 2			
	Min (ms)	Average (ms)	Max (ms)	Average Loss (%)	Min (ms)	Average (ms)	Max (ms)	Average Loss (%)
bit,ly	350,28	337,38	515,59	0,00	368,07	356,46	407,41	0,00
t,co	349,26	334,50	406,47	0,00	366,73	354,41	587,28	0,00
tinyurl,com	205,45	26,42	1297,39	0,00	185,32	28,41	1079,59	0,00
goo,gl	39,86	12,83	185,07	0,00	64,13	35,90	143,95	0,00
ow,ly	305,35	269,46	1289,35	0,00	261,05	248,56	285,03	0,00
dlvr,it	379,94	276,03	1289,00	0,00	389,18	351,77	4058,36	0,08
is,gd	80,94	41,16	4501,48	0,06	87,15	44,33	4367,34	0,03
migre,me	592,97	432,18	4032,20	0,06	507,07	437,57	2733,17	0,06
dld,bz	385,86	332,07	1425,57	0,00	367,68	309,29	914,96	0,00
lnk,ms	336,13	309,55	394,28	0,00	324,46	312,81	1283,64	0,00

Tabelle 8: ICMP and HTTP Latenzen für alle Verkürzungsdienste