

# Wie schütze ich mich vor dem E-Mail-Adress-Klau?

**Woher haben Spammer die E-Mail-Adressen? Die Beantwortung dieser Frage ist ein Ansatz, der das weite Problemfeld Spam an der Wurzel angeht. Es gibt mehrere Methoden, um an E-Mail-Adressen zu gelangen, jedoch ist die größte und meistgenutzte Adressquelle die Webseiten im Internet. Mittels dem sogenannten E-Mail-Adress-Harvesting (engl. „to harvest“: ernten) werden veröffentlichte E-Mail-Adressen auf den Webseiten gesammelt und für den Versand von Spam missbraucht. Wird das automatisierte Sammeln von E-Mail-Adressen durch E-Mail-Adress-Harvesting unterbunden oder zumindest erschwert, so führt dies sowohl zu weniger Spam in den Postfächern, als auch zu weniger Informationsübertragung und –verarbeitung im Internet und in den E-Mail-Infrastrukturen. Im Folgenden wird aus der Sicht eines Harvesters aufgezeigt, welche E-Mail-Adressen besonders schützenswert sind und welche Verfahren zum Schutz vor E-Mail-Adress-Harvesting besonders geeignet sind.**

Die Kommunikation über das Medium E-Mail ist für Firmen und Kunden ein attraktiver Weg, um schnell und kostengünstig miteinander in Kontakt zu treten. Hierzu werden die E-Mail-Adressen an geeigneten Stellen innerhalb einer Webpräsenz platziert und somit für potentielle Kunden/Interessenten veröffentlicht. Die Darstellung der E-Mail-Adressen auf den Webseiten kann unterschiedliche Ausprägungen haben, wie z.B. E-Mail-Adressen im Klartext, Kontaktformulare oder aber verschlüsselte bzw. verschleierte E-Mail-Adressen. Unabhängig von der Darstellung gilt, dass jede veröffentlichte E-Mail-Adresse nicht nur von menschlichen Besuchern gefunden, sondern auch stets automatisiert von Webcrawler (E-Mail-Adress-Harvester) gesammelt werden kann. Entscheidend hierbei ist nur der aufgebrauchte Aufwand seitens des Harvesters. Es ist leicht zu erkennen, dass für den Webseitenbetreiber an dieser Stelle ein Konflikt zwischen der Erreichbarkeit über E-Mail und der Höhe des Spamschutzes entsteht.

Um dieses Spannungsfeld zu beleuchten wird das Problem aus der Sicht des Harvesters dargestellt. Anhand einer Analyse des Instituts für Internet-Sicherheit werden Probleme und Chancen eines Adressensammlers definiert und diskutiert. Außerdem werden Empfehlungen für Webseitenbetreiber dargestellt.

## E-Mail-Adress-Harvesting

Wie funktioniert E-Mail-Adress-Harvesting? Mit Hilfe von Harvestern – speziellen Webcrawlern – wird das Web durchlaufen und verschiedenste Dokumente auf den Webseiten, wie z.B. HTML-, PDF-, und

Bild-Dateien, analysiert (bezüglich der Fachausdrücke siehe auch Glossar: <http://www.internet-sicherheit.de/service/glossar/>).

## Vorgehensweise eines Harvesters

Der Harvester bekommt eine Internetadresse als Einstiegspunkt übergeben und beginnt von dort aus das Sammeln der E-Mail-Adressen. Nachdem das erste Web-Dokument nach E-Mail-Adressen durchsucht wurde, wird das Sammeln auf den verlinkten Dokumenten fortgeführt. Rekursiv durchläuft der Harvester nun Dokumente auf verschiedenen Webservern. Gefundene E-Mail-Adressen werden je nach Implementierung auf dem Rechner des Harvesters gesammelt oder direkt an Spammer übermittelt.

## Verfahren zum Schutz vor E-Mail-Adress-Harvesting

Die Webseitenbetreiber befinden sich im Konflikt zwischen sehr guter Erreichbarkeit über E-Mails und der Stärke des Spamschutzes. Werden Schutzmaßnahmen angewandt, die selbst dem menschlichen Benutzer den Zugang zu E-Mail-Adressen verwehren, so ist zwar der Spamschutz maximiert, aber die Erreichbarkeit des Unternehmens nicht mehr gegeben. Wird stattdessen auf Verfahren zum Schutz vor E-Mail-Adress-Harvesting verzichtet, so sind die veröffentlichten Adressen zweifellos für den Benutzer zu erreichen, aber eben auch problemlos für Harvester. Gesucht wird also ein guter Kompromiss, der die veröffentlichten E-Mail-Adressen vor Harvestern schützt, aber gleichzeitig

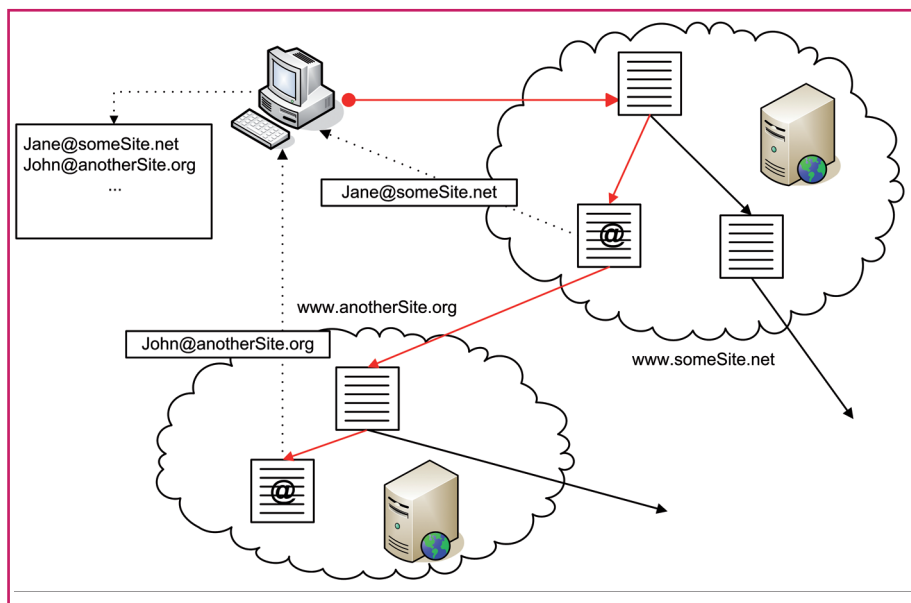


Abb. 1: Veranschaulichung des E-Mail-Adress-Harvestings

eine gute Erreichbarkeit gewährleistet. Ein Ansatz hierfür ist die Benutzung von **HTML-Formularen**. Von Vorteil ist hier, dass im Quellcode keine E-Mail-Adresse vorhanden ist, da die vom Benutzer getätigten Eingaben an ein Skript im Webserver übergeben werden, welches die E-Mail dann verschickt. Dieser hohen Sicherheit bezüglich des E-Mail-Adress-Diebstahl stehen allerdings mehrere Nachteile gegenüber. Die Benutzerfreundlichkeit ist stark eingeschränkt, da der Benutzer seinen bevorzugten E-Mail-Client nicht verwenden, keine Kopien erzeugen oder bei sensiblen Inhalten keine Signatur oder Verschlüsselung anwenden kann. Zusammenfassend können HTML-Formulare empfohlen werden, sofern die erwarteten E-Mails eher zur Kontaktaufnahme dienen, statt dem Versand von sensiblen Informationen.

Eine weitere generelle Methode zum Schutz vor E-Mail-Adress-Harvesting ist unter dem Begriff **„Munging“** vereint, was dem Akronym „MUNG – Mash until no good“ entspringt. Dies bedeutet soviel wie „bis zur Unkenntlichkeit vermischen“ und beschreibt den Fokus dieser Schutzverfahren. Hierbei soll erreicht werden, die zu schützende E-Mail-Adresse derart zu manipulieren, dass nur Menschen die Manipulation rückgängig machen können, nicht aber Harvester. Beispiele hierfür sind:

JohnDoe[at]example[punkt]com,  
JohnDoe @ example . org oder  
JohnDoe@thiscompany.org (put in correct company name),

aber auch aufwendigere Methoden wie Manipulation mittels Cascading Style Sheets (CSS). Bei CSS ist die Manipulation nur für den Harvester sichtbar und für den menschlichen Benutzer transparent, sofern der benutzte Webbrowser CSS interpretieren kann. Problematisch ist bei all diesen Verfahren das Risiko,

dass bei außergewöhnlicher Manipulation der E-Mail-Adressen ein Benutzer nicht mehr in der Lage ist, die Veränderungen rückgängig zu machen und so der Zugang zur E-Mail-Adresse verwehrt bleibt. Je nach Ausprägung der Manipulation können die E-Mail-Adressen außerdem ihre Interaktion verlieren, sie sind also nicht mehr „anklickbar“.

Diese Nachteile stehen einem hohen Spamschutz entgegen, denn einerseits gibt es für den Webseitenbetreiber beliebig viele Möglichkeiten zur Manipulation der E-Mail-Adressen, andererseits stellt die Beachtung des Kontexts (vgl. Beispiel 2 und 3) für Harvester heutzutage einen zu hohen Aufwand dar.

Mit **„Obfuscating“** sind Verfahren zum Schutz vor Harvesting gemeint, die eine zu schützende E-Mail-Adresse nicht mani-

pulieren, sondern gänzlich verstecken sollen. Hierfür gibt es zwei grundsätzliche Ansätze. Der erste Ansatz ist ähnlich

dem der Manipulation mittels CSS, nur dass jetzt die E-Mail-Adressen mit JavaScript kodiert versteckt werden. Als Beispiele können hier die ausgelagerte Entschlüsselung, sowie die Zusammensetzung der E-Mail-Adresse aufgeführt werden:

```
<a href="javascript:uncrypt('encrypted-e-mail-address')">John's address</a>
<a href="javascript:reassemble('example.org', 'JohnDoe')">John Doe</a>
```

Erneut ist die Stärke des Spamschutz sehr hoch, denn eine Interpretation wäre für einen Harvester zu teuer und zu fehleranfällig. Analog zur fehlenden Interpretation von CSS muss dies ebenfalls abgefangen werden, sodass die geschützte E-Mail-Adresse stets lesbar bleibt. Eine Möglichkeit zur Kompensation dieses Nachteils bietet sich im zweiten Ansatz des Obfuscatings. Hierbei wird die zu schützende E-Mail-Adresse **in eine externe Datei ausgelagert**, bspw. in eine Grafik, PDF- oder Flash-Datei. Da die meisten aktuellen Harvester mit Hinblick auf Zeit und Ressourcen lediglich HTML-Codes analysieren, werden die E-Mail-Adressen nicht gefunden. Die je nach Art der Auslagerung hohe Sicherheit geht allerdings mit einer geringen Benutzerfreundlichkeit einher, da unter Umständen ein externes Programm zum Betrachten der E-Mail-Adresse benötigt wird. Die Barrierefreiheit ist in den meisten Fällen nicht gegeben. In diesen Fällen ist eine Kompensation mittels CSS oder JavaScript möglich, wodurch eine gegenseitige Kompensierung der Nachteile gegeben wäre.

**Ergebnisse des Frameworks „Sherlock Harvester“**

Das Framework „Sherlock Harvester“ ist ein intelligentes Tool des Instituts für Internet-Sicherheit zur Analyse des aktuell vorherrschenden Harvestingschutzes. Mit Sherlock Harvester können Sicherheitsanalysen für einzelne Webpräsenzen von Organisationen, aber auch größere Analysen im Internet durchgeführt werden.

Mit diesem Framework wurde eine größere empirische Erhebung von mehr als 1.100 Webpräsenzen durchgeführt. Es wurden in dieser weltweiten Internet-

Dateiklasse	Trefferquote / Anteil der Dateien mit Adressen zur gesamten Dateianzahl der Klasse		Geschwindigkeit der Analyse / Volumen pro Sekunde		Ertrag / Dichte der Adressen pro 1 MB	
MS-Word-D.	23,89%	++	1,27 MB	+	3,23 (0,80)	+
PDF-Dateien (*)	17,51%	++	2,50 MB	++	0,94 (0,20)	0
HTML-Dateien	17,48%	++	0,11 MB	-	19,13 (1,44)	++
MS PowerPoint-D.	24,88%	++	2,68 MB	++	0,32 (0,12)	-
Text-Dateien	7,13%	+	0,68 MB	0	4,17 (0,96)	+
MS Excel-D.	9,41%	+	0,97 MB	0	2,30 (1,15)	+
Komp. Flash	1,88%	0	1,67 MB	+	0,12 (0,07)	-
Grafiken	0,02%	-	0,16 MB	-	0,0 (0,0)	-

(\*) Bei PDF-Dateien wurden lediglich die ersten und letzten 3 Seiten analysiert

Abb. 2: Matrix zur Bewertung von Dateiklassen nach potentieller Attraktivität gegenüber einem Harvester

## MITTELSTAND

Untersuchung etwa 3,5 Mio. Dateien analysiert, welche zusammen ein Datenvolumen von mehr als 210 GB besitzen.

Eine Quintessenz ist in Form einer Matrix (vgl. Abb. 2) dargestellt, welche die potentielle Attraktivität von Dateiklassen für Harvester beschreibt. Im Folgenden werden drei verschiedene Gesichtspunkte betrachtet: Die Trefferquote, eine Web-Datei mit mindestens einer E-Mail-Adresse zu finden, die Geschwindigkeit der Analysen in Bezug auf die jeweilige Web-Dateiklasse sowie die Dichte der E-Mail-Adressen pro Volumen (Megabyte). Werden diese drei Werte kombiniert, so wird die potentielle Attraktivität für Harvester offensichtlich.

Im Umkehrschluss bedeutet dies: Je attraktiver eine Web-Dateiklasse für einen Harvester ist, desto höher ist der Schutzbedarf der E-Mail-Adressen. Ein Harvester sollte keine Chance haben, „günstig“ an veröffentlichte E-Mail-Adressen zu kommen.

Wenn wir die **Trefferquote** betrachten, also den Anteil an Web-Dateien mit mindestens einer E-Mail-Adresse im Verhältnis zur gesamten Dateianzahl dieser Klasse, so wird klar, dass neben HTML-Dateien besonders Word-, PDF- und PowerPoint-Dateien einen hohen Schutzbedarf haben.

Aus diesem Grund sollte jedes Unternehmen eine E-Mail-Adressen Policy erstellen.

Eine solche Policy kann z.B. beinhalten, dass Word-, PDF- und PowerPoint-Dateien, die auf einer Webpräsenz verfügbar gemacht werden sollen, keine E-Mail-Adressen der Autoren enthalten sollen. Außerdem dass z.B. sämtliche veröffentlichten E-Mail-Adressen entweder maskiert (JohnDoe[-at-]example[d0t]org) oder entzerrt werden (JohnDoe @ example. org). Optimal ist eine Kombination verschiedener Verfahren, wodurch mit geringem Aufwand ein relativ hoher Spamschutz erreicht werden kann.

Ein weiteres Kriterium für die Attraktivität ist die **Geschwindigkeit**, mit der die Analysen auf die jeweiligen Dateiklassen vollzogen werden können. Herausragend ist in diesem Fall der Wert von PowerPoint-Dateien, was die Aussage zum hohen

Schutzbedarf dieser Dateiklasse sehr stark untermauert. Auch PDF-Dateien sind in dieser Hinsicht sehr attraktiv, da hier einfache Heuristiken genutzt werden können: In den meisten Fällen sind E-Mail-Adressen, wenn sie vorhanden sind, auf den ersten und letzten Seiten zu finden. Werden die Analysen auf die ersten und letzten drei Seiten beschränkt, so wird ein hoher Geschwindigkeitswert erreicht. Einen sehr niedrigen Geschwindigkeitswert in dieser Matrix haben HTML-Dateien, was daraus resultiert, dass das Framework „Sherlock Harvester“ zahlreiche, teils sehr aufwendige Analysen durchführt, was die Geschwindigkeit mindert. Dies ist aber kein Indiz dafür, dass HTML-Dateien unattraktiv für Harvester sind.

Eine Schlussfolgerung bezüglich der Geschwindigkeit ist, E-Mail-Adressen in Grafiken zu hinterlegen, da die Geschwindigkeit für Harvester sehr langsam ist. OCR-Programme verbrauchen zu viel Zeit und Ressourcen, als dass sie in naher Zukunft für Harvester interessant würden.

Das letzte betrachtete Merkmal ist die **Dichte** an E-Mail-Adressen pro Volumen, hier pro 1 MB. Wie zu erwarten war zeigt sich, dass die meisten E-Mail-Adressen in HTML-Dateien liegen. Aber erneut sind Word-Dateien sehr attraktiv, hier ist die Dichte der E-Mail-Adressen ebenfalls recht hoch. Kehrt man die Überlegungen um, so kann eine Empfehlung darin bestehen, E-Mail-Adressen in Grafiken, aber auch in Flash-Dateien auszulagern. Für Harvester gibt es keinen großen Anreiz, diese Dateien zu analysieren, allerdings müssen die bereits beschriebenen Nachteile bei einer Auslagerung durch zusätzliche Maßnahmen, wie etwa der Einsatz von CSS oder JavaScript, kompensiert werden.

### Fazit

Einen hundertprozentigen Schutz für E-Mail-Adressen gibt es nicht. Ein Harvester kann praktisch jede E-Mail-Adresse finden, wenn nur genügend großer Aufwand zum Überwinden der Schutzmaßnahmen aufgewandt wird.

Allerdings können Harvester zurzeit ihre E-Mail-Adressen sehr einfach ernten. Aktuelle Harvester gelangen mit minimalem

Aufwand an sehr viele E-Mail-Adressen. Obwohl eine Vielzahl an Schutzmaßnahmen auf unterschiedlichen Ebenen existiert, werden diese kaum benutzt. Mit Analyse-Tools wie der „Sherlock Harvester“ kann die aktuelle Situation der eigenen Webpräsenz einfach untersucht werden. Der Einsatz von einfachen Verfahren zum Schutz von E-Mail-Adressen kann die Wahrscheinlichkeit drastisch mindern, „geharvestet“ zu werden.

Wir empfehlen jedem Unternehmen, die Verwendung von E-Mail-Adressen auf den Webseiten zu überprüfen und mit Hilfe einer E-Mail-Adress Policy geeignete Maßnahmen zum Schutz der E-Mail-Adressen einzuführen.

### Literatur:

[Feld07] S. Feld: „Analyse von Verschleierungstechniken für E-Mail-Adressen zum Schutz vor E-Mail-Adress-Harvesting“, Bachelor-Thesis, Institut für Internet-Sicherheit if(is), FH-Gelsenkirchen 2007

### Autoren



#### B. Sc. Sebastian Feld

forschte im Rahmen seiner Bachelor-Thesis im Bereich „E-Mail-Adress-Harvesting“ im Institut für Internet-Sicherheit



#### Prof. Dr. Norbert Pohlmann

ist Informatikprofessor für Verteilte Systeme und Informationssicherheit sowie Leiter des Instituts für Internet-Sicherheit - if(is) an der Fachhochschule Gelsenkirchen ([www.internet-sicherheit.de](http://www.internet-sicherheit.de))