

Harvesting

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

Prof. Dr. Norbert Pohlmann
pohlmann (at) internet-sicherheit (dot) de

B. Sc. Sebastian Feld
feld (at) internet-sicherheit (dot) de

Institut für Internet-Sicherheit – if(is)
Fachhochschule Gelsenkirchen
<http://www.internet-sicherheit.de>

- Motivation der Spammer
- Definition eines Harvesters
- Grundsätzliche Verfahren zur Gewinnung von E-Mail-Adressen
- Verfahren zum Schutz vor E-Mail-Adress-Harvesting
- Framework „Sherlock Harvester“
- Empirische Erhebung und Ergebnisse
- Fazit / Ausblick / Empfehlungen

- **Motivation der Spammer**
- Definition eines Harvesters
- Grundsätzliche Verfahren zur Gewinnung von E-Mail-Adressen
- Verfahren zum Schutz vor E-Mail-Adress-Harvesting
- Framework „Sherlock Harvester“
- Empirische Erhebung und Ergebnisse
- Fazit / Ausblick / Empfehlungen

Motivation der Spammer

- Was braucht ein Spammer?
 - (Gekaperte) Rechner zum Versenden von Spam-Mails
 - E-Mail-Adressen der potentiellen „Kunden“

- Woher haben Spammer die Adressen?
 - Mehrere Methoden, um an Adressen zu gelangen
 - Größte und meistgenutzte Adressquelle: Internetseiten
→ **E-Mail-Adress-Harvesting** (engl. „to harvest“: ernten)

- Das Unterbinden oder zumindest Erschweren des automatisierten Sammelns von Adressen führt...
 - Zu weniger Spam in den Postfächern
 - Zu weniger Informationsübertragung und -verarbeitung

- **Konflikt:** Erreichbarkeit vs. Stärke des Spamschutz

- Motivation der Spammer
- **Definition eines Harvesters**
- Grundsätzliche Verfahren zur Gewinnung von E-Mail-Adressen
- Verfahren zum Schutz vor E-Mail-Adress-Harvesting
- Framework „Sherlock Harvester“
- Empirische Erhebung und Ergebnisse
- Fazit / Ausblick / Empfehlungen

Definition

→ Techniken eines Harvesters

■ Bot

- autonom
- Mensch wäre mengen- und/oder zeitmäßig überfordert
- meist simpel, aber effizient

■ Webcrawler

- spezieller Bot
- durchläuft Webseiten
- analysiert Dokumente (z.B. HTML, PDF, TXT, JPG, ...)

■ Harvester

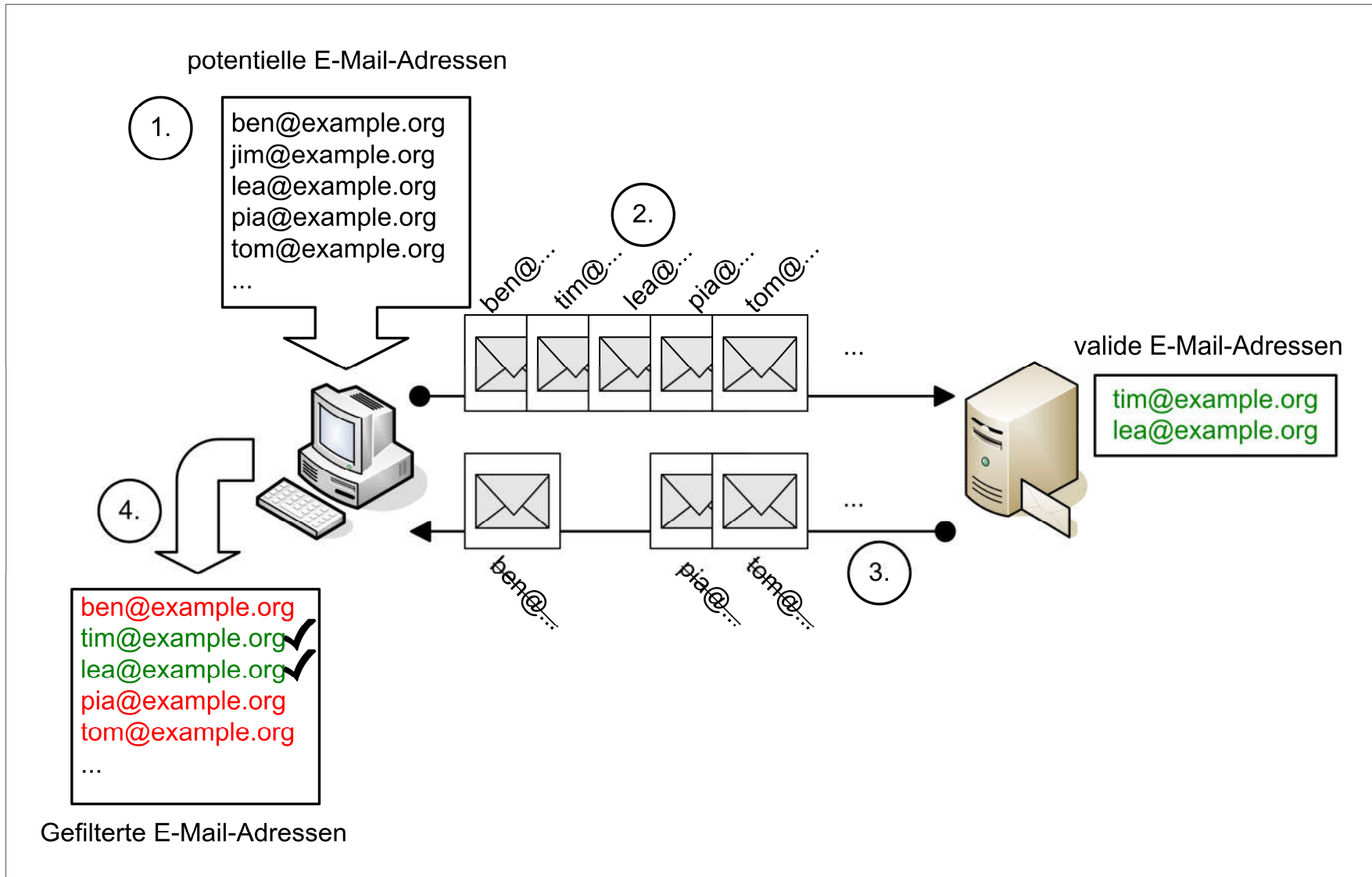
- spezieller Webcrawler
- „erntet“ Informationen (z.B. E-Mail-Adressen, Telefonnummern, ...)

- Motivation der Spammer
- Definition eines Harvesters
- **Grundsätzliche Verfahren zur Gewinnung von E-Mail-Adressen**
- Verfahren zum Schutz vor E-Mail-Adress-Harvesting
- Framework „Sherlock Harvester“
- Empirische Erhebung und Ergebnisse
- Fazit / Ausblick / Empfehlungen

Grundsätzliche Verfahren zur Gewinnung von E-Mail-Adressen

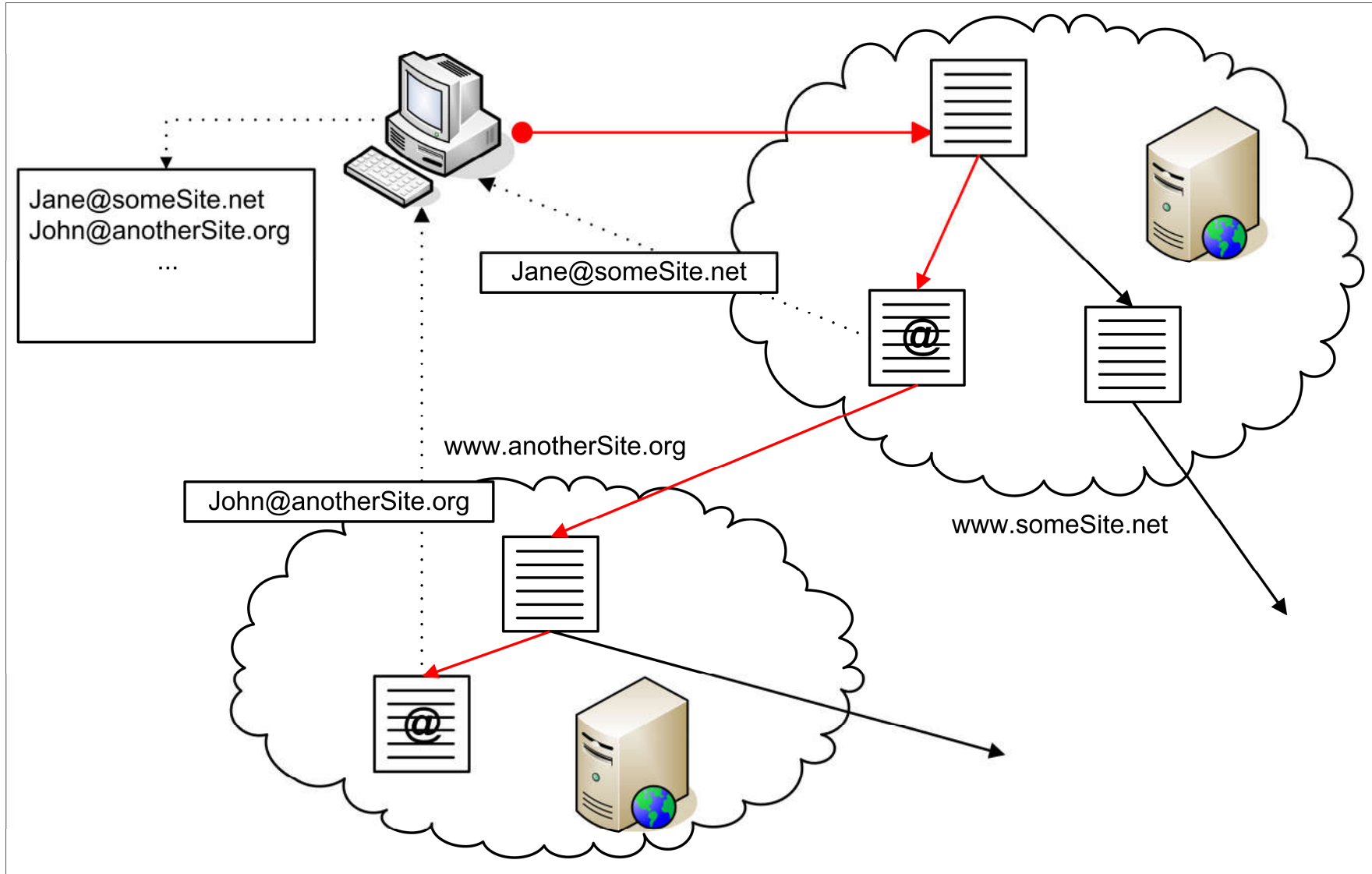
- **Directory-Harvest-Angriffe (DHA)**
 - Verbreitete Vor- und Nachnamen
 - Alle Kombinationen (Brute-Force)
 - Angriffsziel: **E-Mail-Gateway**
- **E-Mail-Adress-Harvesting**
 - Webcrawler für E-Mail-Adressen
 - Angriffsziel: **Webserver**
- ...

Directory-Harvest-Attacke (DHA)



Angriffsziel: **E-Mail-Gateway**

E-Mail-Adress-Harvesting



Angriffsziel: **Webserver**

- Motivation der Spammer
- Definition eines Harvesters
- Grundsätzliche Verfahren zur Gewinnung von E-Mail-Adressen
- **Verfahren zum Schutz vor E-Mail-Adress-Harvesting**
- Framework „Sherlock Harvester“
- Empirische Erhebung und Ergebnisse
- Fazit / Ausblick / Empfehlungen

Verfahren zum Schutz vor E-Mail- Address-Harvesting

- E-Mail-Adresse als Hyperlink
- HTML-Formular
- Munging (Manipulieren)
- Obfuscating (Verstecken)
- Weitere Schutztechniken
 - IP-Blacklist
 - Referer-String
 - ...

E-Mail-Adresse als Hyperlink

- Beispiel:
 - `JohnDoe@example.org`
 - `John Doe`
- Benutzerfreundlichkeit / Erreichbarkeit:
 - Maximiert
 - Aktivierung des Verweises öffnet E-Mail-Client
- Sicherheit / Stärke des Spamschutz:
 - Nicht vorhanden, da „mailto:“ und „@“ gleich zwei Schlüsselworte darstellen, die ohne Aufwand vom Harvester erkannt werden können
- Fazit / Empfehlung:
 - „Fahrlässige“ Preisgebung der E-Mail-Adresse. Vermeiden!

HTML-Formular

- Beispielformular mit mailto-Schema:
 - `<form action="/cgi-bin/mail.pl" method=get> ... </form>`
- Benutzerfreundlichkeit / Erreichbarkeit:
 - Vorteil: Benutzer braucht keinen E-Mail-Client
 - Nachteil: Benutzer kann seinen bevorzugten E-Mail-Client nicht nutzen, keine Signatur anwenden oder Kopien erzeugen
- Sicherheit / Stärke des Spamschutz:
 - Hoch, da keine E-Mail-Adresse im Quellcode vorhanden ist
 - Angriffe wie E-Mail-Header-Injection oder XSS müssen ausgeschlossen werden
- Fazit / Empfehlung:
 - Problematisch bei sensiblen Inhalten von E-Mails, sonst bei sachkundiger Implementierung zu empfehlen.

Munging (1/5)

- Beispiel: Hinzufügen von adress-fremden Teilen
 - JohnDoe@REMOVETHISexample.org
 - JohnDoe<!-- I don't like Spam -->@example.org
- Benutzerfreundlichkeit / Erreichbarkeit:
 - Benutzer muss Manipulation erkennen und händisch rückgängig machen
 - Geschieht ein Fehler, so wird dies nicht erkannt und die E-Mail wird trotzdem verschickt
- Sicherheit / Stärke des Spamschutz:
 - Begrenzt, da mittels regulärer Ausdrücke Begriffe wie „NOSPAM“ oder „REMOVETHIS“ leicht erkannt und entfernt werden können
- Fazit / Empfehlung:
 - Moderate Sicherheit und Fehleranfälligkeit auf Seiten des Benutzers schrecken ab.

- Beispiel: Maskierung
 - JohnDoe-at-example-org
 - JohnDoe[ät]example[punkt]org
- Benutzerfreundlichkeit / Erreichbarkeit:
 - Erneut: Benutzer muss Manipulation erkennen und manuell rückgängig machen
 - Zu wilde Variationen der Schlüsselwörter können eine Entschlüsselung durch den Benutzer unmöglich machen, wodurch keine E-Mail verschickt werden kann
- Sicherheit / Stärke des Spamschutz:
 - Vorteil für den Webseitenbetreiber, da beliebig viele Möglichkeiten der Maskierung existieren
- Fazit / Empfehlung:
 - Guter Schutz, aber u.U. auf Kosten der Nerven des Besuchers.

Munging (3/5)

- Beispiel: Unicode-Kodierung
 - JohnDoe@example.org
- Benutzerfreundlichkeit / Erreichbarkeit:
 - Maximiert
 - Webbrowser interpretiert eine so kodierte E-Mail-Adresse und stellt sie als normalen Klartext dar
- Sicherheit / Stärke des Spamschutz:
 - Nicht vorhanden, da die beiden Schlüsselwörter „@“ und „.“ lediglich linear transformiert werden
- Fazit / Empfehlung:
 - Schutzpotential nicht mehr vorhanden. Meiden!

Munging (4/5)

- Beispiel: Entzerren der Adresse und „Rätsel“
 - JohnDoe @ example . org
 - JohnDoe@thiscompany.org (put in correct company name)
- Benutzerfreundlichkeit / Erreichbarkeit:
 - Eingeschränkt, da der Benutzer durch den Kontext die Begrenzungen der E-Mail-Adresse / die Domain erkennen muss
 - E-Mail-Adresse besitzt kein Verhalten mehr
- Sicherheit / Stärke des Spamschutz:
 - Relativ hoch, da die Beachtung des Kontexts für Bots ein großes Problem darstellt (jedenfalls heutzutage)
- Fazit / Empfehlung:
 - Guter Schutz, aber je nach Einsatzgebiet nicht möglich.

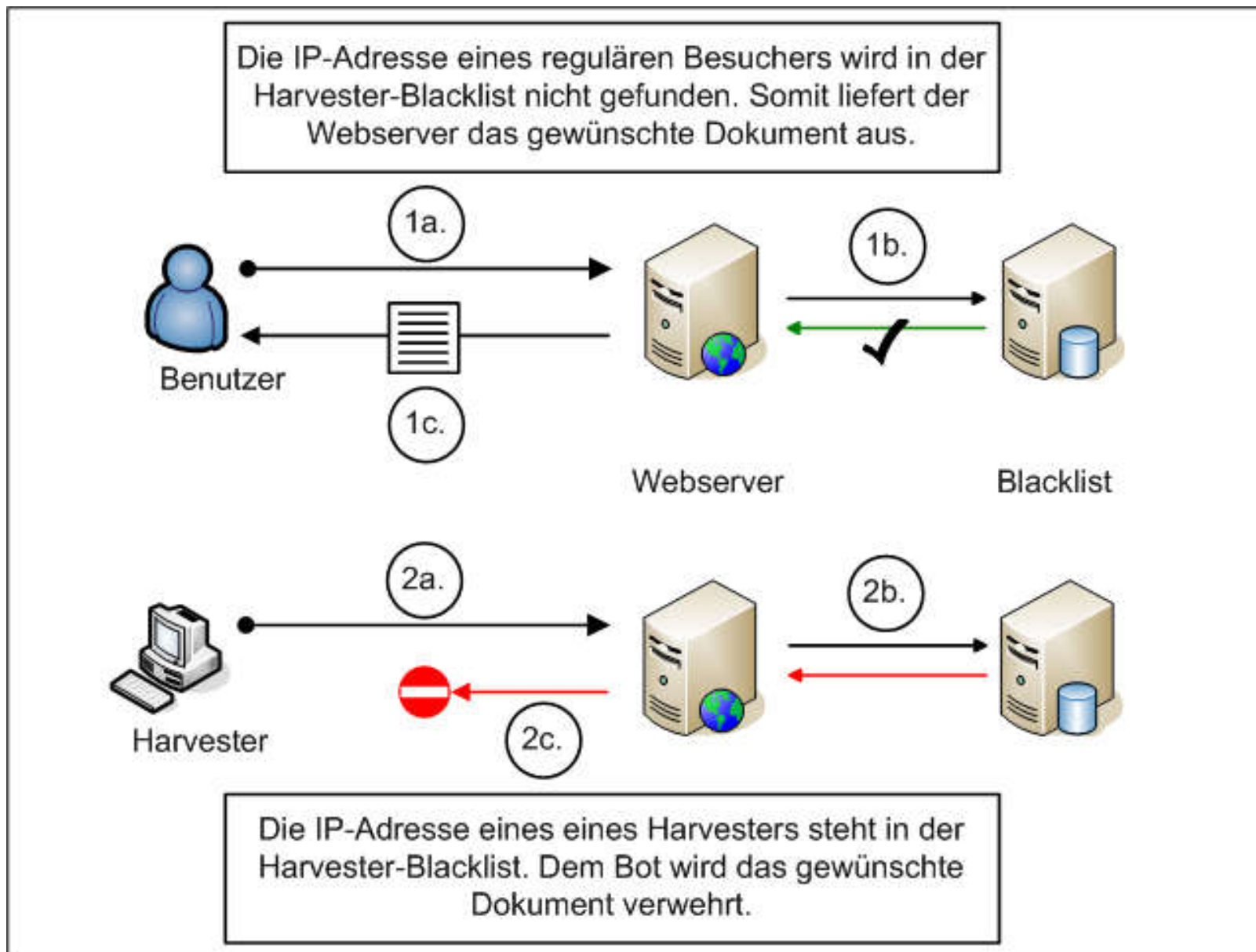
Munging (5/5)

- Beispiel: Manipulation mittels Cascading Style Sheets (CSS)
 - `gro.elpmaxe@eoDnhoJ`
 - `JohnDoenospam@example.org`
- Benutzerfreundlichkeit / Erreichbarkeit:
 - Maximiert, wenn der Webbrowser den Code interpretiert und normale Klartext-Adressen ausgibt (transparent)
 - Sehr begrenzt, wenn der Webbrowser keine CSS-Befehle interpretieren kann und die Adressen somit unkenntlich bleiben können
- Sicherheit / Stärke des Spamschutz:
 - Mit erhöhtem Aufwand bietet sich hier eine starke Verschleierungsmethode
- Fazit / Empfehlung:
 - Sehr zu empfehlen. Fehlende Interpretation von CSS muss allerdings abgefangen werden.

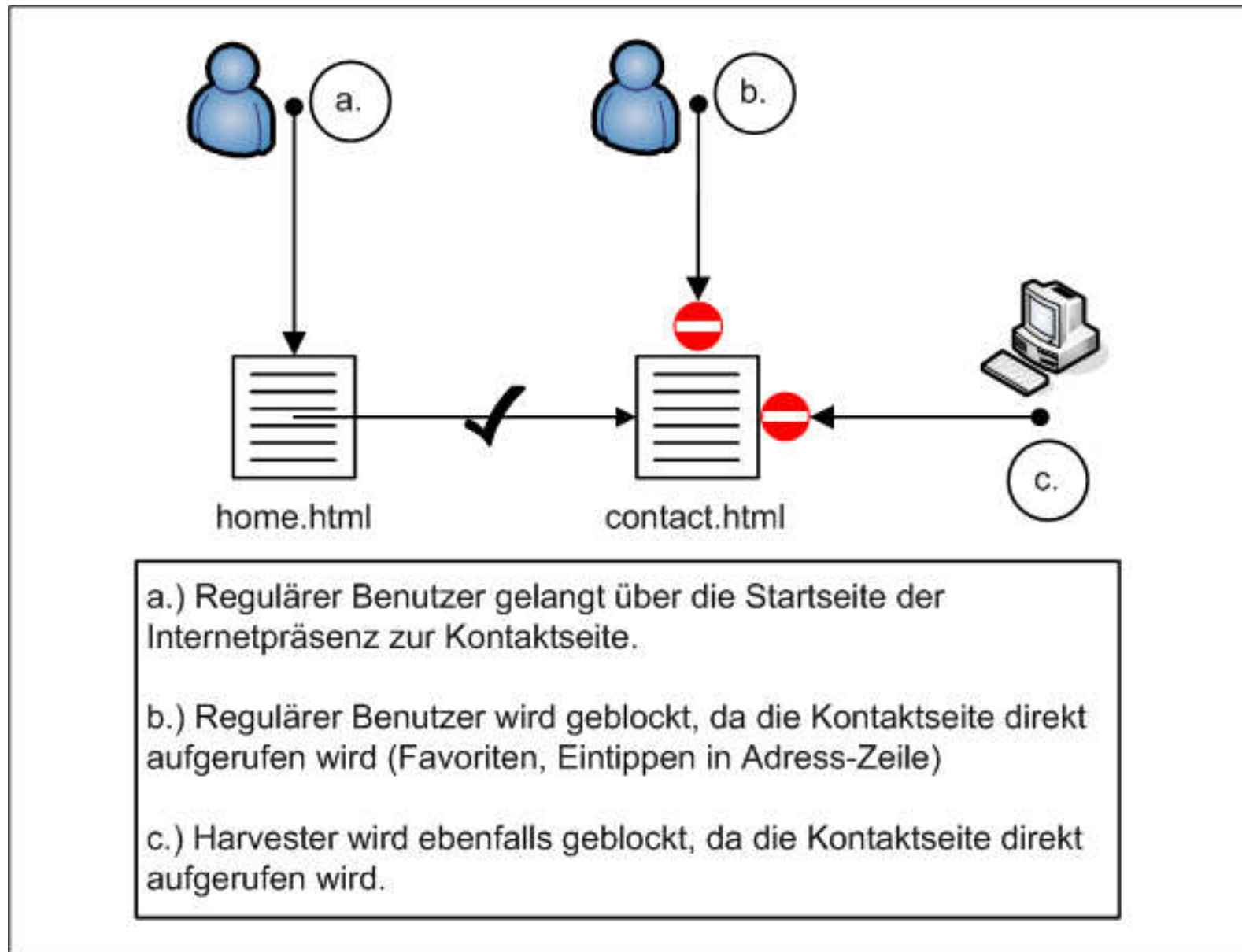
- Beispiel: Auslagerung in externer Datei
 - E-Mail-Adresse wird in externe Datei wie Bild, PDF-Datei, Flash-Datei o.ä. ausgelagert
 - Harvester, der nur HTML-Code analysiert, kann diese nicht finden
- Benutzerfreundlichkeit / Erreichbarkeit:
 - Gering, da der Benutzer u.U. ein externes Programm benötigt
 - Barrierefreiheit nicht gegeben
- Sicherheit / Stärke des Spamschutz:
 - Je nach Art der Auslagerung sehr starke Sicherheit möglich
- Fazit / Empfehlung:
 - Sehr zu empfehlen, wenn Nachteile durch weiteren Mechanismus wie CSS oder JavaScript kompensiert werden.

- Beispiel: JavaScript-Verschleierung
 - `John's address`
 - `John Doe`
- Benutzerfreundlichkeit / Erreichbarkeit:
 - Maximiert, wenn der Webbrowser JavaScript interpretiert und eine normale Klartext-Adresse zurückliefert
 - Nicht vorhanden, wenn der Webbrowser JavaScript nicht interpretiert und somit dem Benutzer die E-Mail-Adresse gänzlich verwehrt
- Sicherheit / Stärke des Spamschutz:
 - Sehr stark, da eine Interpretation für einen Bot zu teuer und zu gefährlich ist.
- Fazit / Empfehlung:
 - Sehr zu empfehlen. Fehlende Interpretation muss wie bei CSS abgefangen werden.

Identifizierung und Abwehr des Harvesters mittels IP-Blacklist



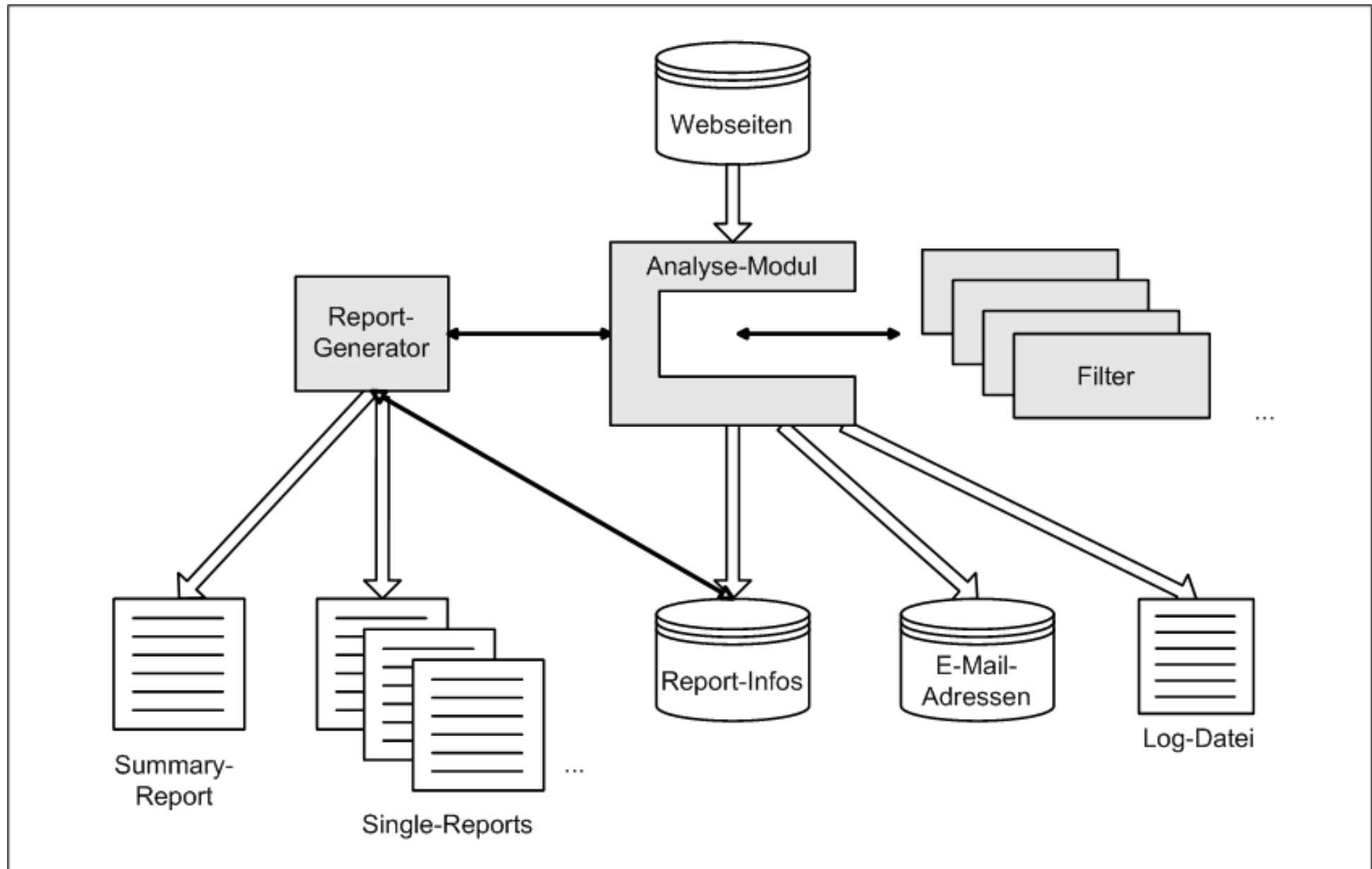
Prüfung des Referer-Strings zur Beschränkung des Zugriffs



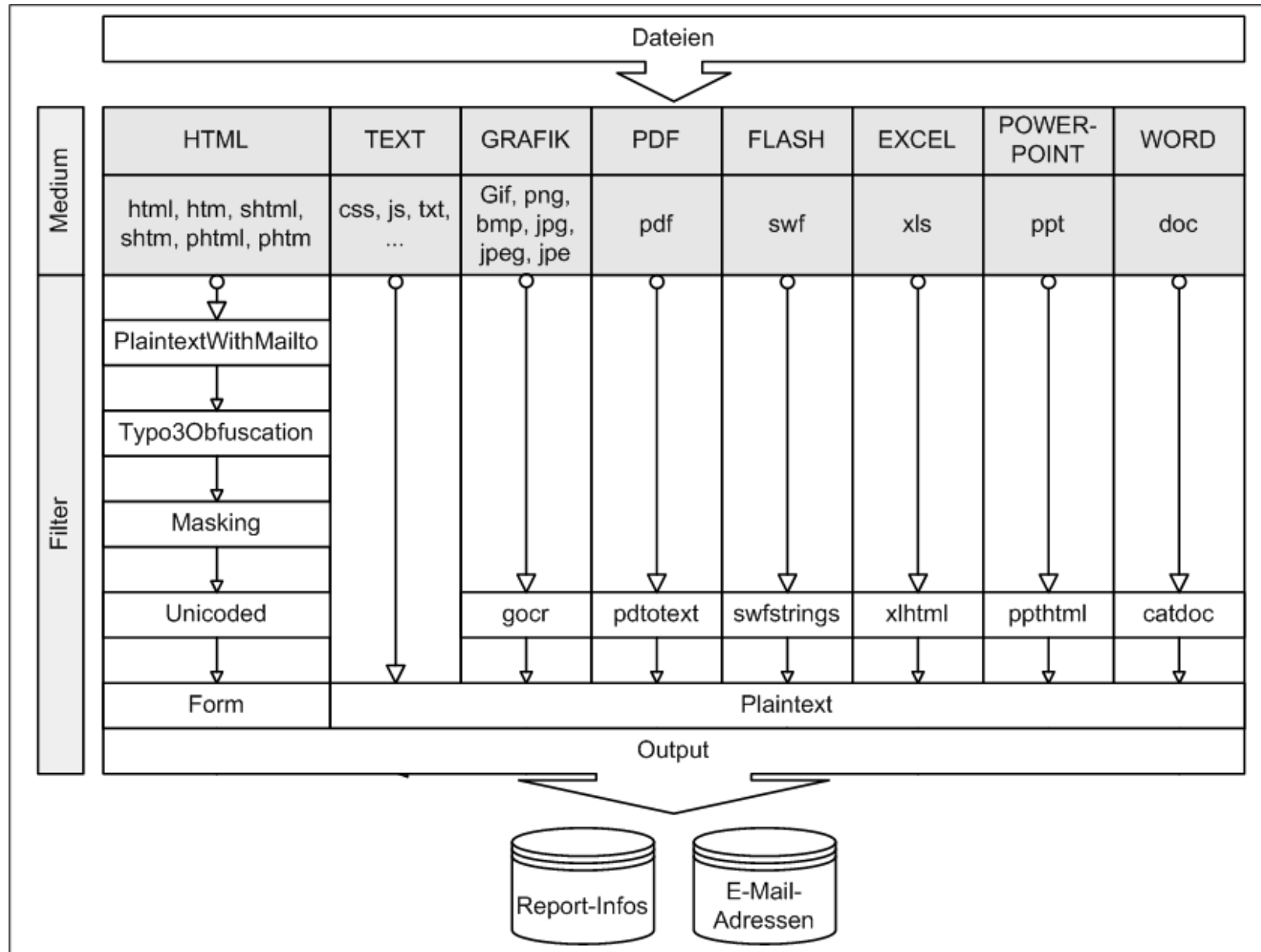
- Motivation der Spammer
- Definition eines Harvesters
- Grundsätzliche Verfahren zur Gewinnung von E-Mail-Adressen
- Verfahren zum Schutz vor E-Mail-Adress-Harvesting
- **Framework „Sherlock Harvester“**
- Empirische Erhebung und Ergebnisse
- Fazit / Ausblick / Empfehlungen

- Motivation zur Erstellung des Frameworks
 - Ermittlung der aktuellen Lage in Bezug auf Harvestingschutz
 - Beleuchtung des Problems aus Sicht des Harvesters bietet neue Erkenntnisse
 - Durchführung von Sicherheitsanalysen für Internetpräsenzen
- Grobe Vorgehensweise
 - Lokales Sichern der Webseiten
 - Konvertierung vieler Dateiformate in HTML
 - Analyse der HTML-Dateien
 - Report-Generierung

Architektur des Frameworks „Sherlock Harvester“



Schematische Darstellung des Analyse-Moduls



Report-Generator

- Single-Report
- Summary-Report

- Motivation der Spammer
- Definition eines Harvesters
- Grundsätzliche Verfahren zur Gewinnung von E-Mail-Adressen
- Verfahren zum Schutz vor E-Mail-Adress-Harvesting
- Framework „Sherlock Harvester“
- **Empirische Erhebung und Ergebnisse**
- Fazit / Ausblick / Empfehlungen

- Quintessenz des Frameworks „Sherlock Harvester“
- Liste „Forbes Global 2000“
- Generell
 - 1.187 Internetpräsenzen
 - 219 GB
 - 3,47 Mio. Dateien
 - 1.029.626 (distinct: 90.518) E-Mail-Adressen
 - 416.453 Dateien mit mindestens einer E-Mail-Adresse
 - Pro 1 MB 4,69 (distinct: 0,41) E-Mail-Adressen
 - 30.089 verschiedene Domains der E-Mail-Adressen

Die „typische“ Internetpräsenz (Median)

Dateiendung	Volumen	Anzahl der Dateien	Anzahl der E-Mail-Adressen
pdf	29,22 MB	56	3 (distinct: 2)
html	2,77 MB	177	8 (distinct: 3)
jpg	1,45 MB	75	0
gif	360 KB	102	0
swf	46 KB	1	0
js	15 KB	3	0
CSS	11 KB	2	0

Erfolg der implementierten Filter

Filter	Anzahl der Domains mit Erfolg
form	1.000
plaintextWithMailto	911
pdftotext	748
plaintext	276
catdoc	177
xlhtml	111
swfstrings	85
ppthtml	66
gocr	11
unicoded	8
masking	2
typo3Obfuscation	0

Trefferquote für die Dateiklassen

Dateiklasse	Anzahl der Dateien mit E-Mail-Adresse zur Gesamtanzahl der Dateien dieser Klasse
MS PowerPoint-Dateien	24,88%
MS Word-Dateien	23,89%
PDF-Dateien	17,51%
HTML-Dateien	17,48%
MS Excel-Dateien	9,41%
Text-Dateien	7,13%
Kompilierte Flash-Dateien	1,88%
Grafiken	0,02%

Zeitlicher Aufwand der Dateiklassen

Dateiklasse	Größe des analysierten Volumens pro Sekunde
MS PowerPoint-Dateien	2,68 MB
PDF-Dateien (*)	2,50 MB
Kompilierte Flash-Dateien	1,67 MB
MS Word-Dateien	1,27 MB
MS Excel-Dateien	0,97 MB
Text-Dateien	0,68 MB
Grafiken	0,16 MB
HTML-Dateien	0,11 MB

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

Ertrag der Dateiklassen / Adress-Dichte

Dateiklasse	Anzahl der E-Mail-Adressen	Anzahl der E-Mail-Adressen pro 1 MB
HTML-Dateien	885.013 (distinct: 66.513)	19,13 (distinct: 1,44)
PDF-Dateien (*)	127.129 (26.829)	0,94 (0,20)
Text-Dateien	7.387 (1.702)	4,17 (0,96)
MS Word-Dateien	5.972 (1.486)	3,23 (0,80)
MS Excel-Dateien	2.715 (1.354)	2,30 (1,15)
MS PowerPoint-Dateien	665 (250)	0,32 (0,12)
Kompilierte Flash-Dateien	317 (191)	0,12 (0,07)
Grafiken	23 (22)	0,0 (0,0)

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

Potentielle Attraktivität gegenüber Harvester

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)	o
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	o	4,17 (0,96)	+
MS Excel-D.	9,41%	+	0,97 MB	o	2,30 (1,15)	+
Komp. Flash	1,88%	o	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

- Motivation der Spammer
- Definition eines Harvesters
- Grundsätzliche Verfahren zur Gewinnung von E-Mail-Adressen
- Verfahren zum Schutz vor E-Mail-Adress-Harvesting
- Framework „Sherlock Harvester“
- Empirische Erhebung und Ergebnisse
- **Fazit / Ausblick / Empfehlungen**

- Ein Harvester kann jede E-Mail-Adresse finden. Mit genügend großem Aufwand kann jede Schutzmaßnahme überwunden werden.
- Aktuell: Mit minimalem Aufwand kommen Harvester an sehr viele E-Mail-Adressen.
- Vielfältige Schutzmaßnahmen auf unterschiedlicher Ebene
- Verschleierungstechniken werden kaum benutzt, obwohl sie die Wahrscheinlichkeit „geharvestet“ zu werden minimieren.

Ausblick

→ „Sherlock Harvester“

- Implementierung weiterer Filter und Konverter
- Wiederholtes Analysieren der Domains
- Sammlung von weiteren Informationen
- Sortierung der Domains und Unterscheidung der Analysen nach Unternehmen, Organisationen, Universitäten, etc.
- Validierung der gefundenen E-Mail-Adressen
- Webserver zur Analyse der „weiteren Schutztechniken“ aufsetzen

Empfehlungen

- Nur E-Mail-Adressen veröffentlichen, die Sinn machen
- Erstellung einer Policy für E-Mail-Adressen (Chats, Foren, Gewinnspiele, Golfclub, ...)
- Bei Benutzung von Verschleierungsmethoden kann bereits mit geringem Aufwand eine Vielzahl von Harvestern umgangen werden
- Aufwendiger, aber effektiver: Kombinierte Verwendung von Verschleierungsmethode. Eventuelle Nachteile werden kompensiert.
- Durchführung einer Sicherheitsanalyse auf den Webseiten
 - Erfahrung von Parametern der Internetpräsenz
 - Ermittlung der Sicherheit der E-Mail-Adressen
 - Auch: „Aufräumen“ von großen Internetpräsenzen bzgl. E-Mail-Adressen

Harvesting

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

**Vielen Dank für Ihre Aufmerksamkeit
Fragen ?**

**Prof. Dr. Norbert Pohlmann
pohlmann (at) internet-sicherheit (dot) de**

**B. Sc. Sebastian Feld
feld (at) internet-sicherheit (dot) de**

Institut für Internet-Sicherheit – if(is)
Fachhochschule Gelsenkirchen
<http://www.internet-sicherheit.de>