KAPITEL 4: RECONNAISSANCE – ZIELNETZWERK AUSWERTEN

Es gibt doch nichts, auf dem soviel Verführung und so viel Fluch liegt wie auf einem Geheimnis.

— SØREN KIERKEGAARD

Im ersten Schritt einer Auswertung müssen möglichst viele Informationen zum Angriffsziel zusammengetragen werden. Eine beachtliche Anzahl Hintergrundinformationen kann durch verschiedene Footprinting-Methoden in Erfahrung gebracht werden. Die klassischen Ansätze wurden in Kapitel 3 illustriert. Dieses Wissen bildet die Grundlage für das weitere strukturierte Vorgehen, mit dem Schwachstellen aufgedeckt und Sicherheitslücken ausgenutzt werden sollen. Es ist wichtig, um Zugriffe zu planen und möglichst effizient ausführen zu können.

Der nächste Schritt sind erste Auswertungen der Zielumgebung, wobei Eigenschaften im direkten Kontakt ermittelt werden sollen. Dabei geht es zuerst darum, das Netzwerk auszukundschaften und eine netzwerkbasierte Reconnaissance (dt. Erkennung) durchzuführen. Da die Verbindung zu den vernetzten Zielsystemen über dieses Medium läuft, müssen so viele Details wie möglich darüber in Erfahrung gebracht werden. Grundsätzlich werden dabei vier Ziele verfolgt:

- Herausfinden von IP-Adressen (Kapitel 4.1): Die Verbindung zu Systemen wird in TCP/IP-Netzen anhand der eindeutigen IP-Adresse aufgebaut. Man muß sie für direkte Auswertungen und zielgerichtete Attacken kennen. Anhand verschiedener Techniken werden die IP-Adressen einer Zielumgebung im ersten Teil dieses Kapitels ermittelt.
- Ermitteln der Broadcast-Adressen (Kapitel 4.2): Die sogenannten Broadcast-Adressen spielen eine wichtige Rolle. Sie lassen sowohl eine Eingrenzung des IP-

Adreßbereichs als auch die Vorbereitung spezieller Zugriffstechniken zu. Der zweite Teil widmet sich diesem Thema.

– Ermitteln der Routen (Kapitel 4.3): Die Kommunikation innerhalb eines Netzwerks wird durch das Routing vorgegeben. Es definiert die Kommunikationswege und die Möglichkeiten des Datenaustauschs. Es werden verschiedene Methoden gezeigt, anhand derer die Routen identifiziert und ein Zugriffspfaddiagramm erstellt werden kann. Dabei werden sogar erste Auswertungen von Firewall-Systemen durchgeführt.
– Herausfinden des Mediums (Kapitel 4.4): Zum Schluß wird durch verschiedene Techniken das eingesetzte Medium ermittelt. Dabei werden natürliche Unterschiede, wie sie zum Beispiel zwischen Kabel- und Satelliten-Verbindungen gegeben sind, gesucht. Diese Kenntnisse helfen dabei, zielgerichtete Netzwerkangriffe unter Berücksichtigung der individuellen Eigenschaften vorzubereiten.

Dieses Kapitel relativ umfangreich. Zum einen hat das mit der umfassenden und detaillierten Besprechung der jeweiligen Techniken zu tun. So werden schon an dieser frühe Stelle raffinierte Methoden vorgestellt, mit denen sich Firewall-System zweifelsfrei entdecken und effektiv umgehen lassen.

Zum anderen ist die tiefgehende Diskussion erforderlich, um die Betrachtungen überhaupt fachgerecht umsetzen zu können. Hierbei werden schon sehr früh einige zentrale Konzepte von TCP/IP-Netzwerken behandelt. So werden elementare Netzwerktechnologien wie Namensauflösungen, Maximum Transmission Unit (MTU), IP-Fragmentierung und TTL-Werte vorgestellt. Sie spielen bei den detaillierten Betrachtungen der Netzwerkkommunikationen sowie spezifischer Angriffstechniken über das Netzwerk auch im späteren Verlauf des Buchs eine zentrale Rolle. Die besagten Mechanismen werden immer zum Verständnis der besprochenen Techniken herangezogen und können sie durch ihren geschickten Einsatz in wichtigen Punkten optimieren.

4.1 IP-Adressen herausfinden

Die IP-Adresse ist die weltweit eindeutige Identifikationsnummer eines jeden innerhalb von TCP/IP vernetzten Systems. Obwohl dies spätestens seit der Einführung von privaten Adreßbereichen mit RFC 1918 und der Erfindung von NAT (Network Address Translation) nicht mehr ganz stimmt. Trotzdem ist deren Kenntnis eines der wichtigsten Zwischenziele beim langen Weg eines erfolgreichen Angriffs über ein Netzwerkmedium. Denn nur wenn die IP-Adresse eines Zielsystems bekannt ist, kann mit diesem direkt in Kontakt getreten oder können effizient indirekte Attacken umgesetzt werden.

Mit verschiedenen Footprinting-Techniken, wie sie vor allem Kapitel 3 zeigt, kann man theoretisch Hostnamen, IP-Adressen oder IP-Adreßbereiche der Zielumgebung herausfinden. Doch was soll man tun, wenn sämtliche Vorarbeit keinen Erfolg bringen? Noch mühsamer ist es, wenn sich die IP-Adresse des Zielsystems ständig ändert, zum Beispiel dann, wenn bei jedem Neustart oder jeder Einwahl eine neue IP-Adresse anhand DHCP, BOOTP oder RARP vergeben wird.

Nachfolgend werden verschiedene Methoden gezeigt, wie sich IP-Adressen lokalisieren und damit Ziele identifizieren lassen. Dabei wird zum Beispiel die Technik der Namensauflösung mittels DNS (Domain Name System), die im Verlaufe dieses Buches immer wieder ein Thema sein wird, eine wichtige Rolle spielen. Durch das Miteinbeziehen von erweiterten whois-Abfragen können zudem IP-Adreßdetails in Erfahrung gebracht werden, wie dies auch schon in Kapitel 3.3 besprochen wurde.

4.1.1 Namensauflösung

Das Domain Name System (DNS), es wurde von Paul Mockapetris im Jahr 1983 erfunden, gilt als eines der wichtigsten Dienste im Internet [Mockapetris 1983]. Die Hauptaufgabe des verteilten Systems ist es, eine Auflösung von Hostnamen zu IP-Adressen und umgekehrt zu ermöglichen. Die Idee von DNS ist darauf zurückzuführen, daß sich Menschen viel besser Hostnamen in der Form von www.computec.ch merken können, anstatt mit komplexen IP-Adressen in der Form von 192.168.0.10 zu jonglieren. Im Gegenzug ist die elektronische Datenverarbeitung durch Computer anhand von IP-Adressen (und MAC-Adressen) einfacher. Entsprechend mußte ein komfortabler und möglichst transparenter Auflösungsmechanismus her.
Innerhalb von DNS gilt eine Baumstruktur. Die nationale TLD der Schweiz lautet .ch und wird durch die Switch vergeben und verwaltet. Im Gegenzug ist in Deutschland Denic für die TLD .de zuständig. Ein gültiger Domainname ist zum Beispiel *computec.ch*. Die jeweiligen Systeme können www.computec.ch (Webserver) und mail.computec.ch (Mailserver) lauten. Ebenso sind auch weitere Subdomains wie test.www.computec.ch denkbar. Die Blätter und Knoten des Baumes werden als Labels bezeichnet.
DNS arbeitet im üblichen Client/Server-Prinzip. Den Server-Teil bilden sogenannte Nameserver beziehungsweise DNS-Server, wobei zwischen autoritativen und nichtautoritativen Nameservern unterschieden wird. Ein autoritativer Nameserver ist das für eine Zone zuständige System, dessen Antworten als gesichert deklariert werden. Jede Zone muß einen solchen Primary Nameserver zur Verfügung stellen. Dieser wird durch den SOA Resource Record in einer Zonendatei geführt.
Zur Gewährleistung der Erreichbarkeit und der Lastenverteilung werden autoritative Nameserver meistens als Verbund in einem Cluster betrieben. Dabei übernehmen einer oder mehrere secondary Nameserver im Falle eines Problems das Backup. Die Definition der sekundären Nameserver erfolgt als NS Resource Record. Die DNS-Einträge werden grundsätzlich auf dem primary Nameserver generiert. Bei der Beantwortung der Anfragen sind jedoch sämtliche Nameserver (sowohl primary als auch secondary) gleichwertig und gelten als autoritativ. Die Synchronisation zwischen dem primären und den sekundären Nameservern wird durch einen Zonentransfer realisiert. Dabei findet ein Datenaustausch zwischen Master und Slave über den TCP-Port 53 (dns) statt. Im Notify-Verfahren informiert der Master sämtliche Slaves über Änderungen in der Zonendatei. Der Slave hat dann die entsprechenden Informationen anzufragen und in der eigenen Datenbank zu aktualisieren. Dies geschieht entweder als kompletter Zonentransfer oder zur Schonung der Ressoursen als inkrementeller Transfer [Ohta 1996].
Die alternative Methode zur Synchronisation der Datensätze zwischen Master und Slave ist das Slave-Hol-Verfahren. Der Slave versucht in zyklischen Abständen (die Refresh

Time; in der Regel eine Stunde), den SOA Resource Record des Masters zu beziehen. Daraufhin wird die Seriennummer des SOA-Record verglichen. Ist diese beim Master größer als beim Slave, wurden Anpassungen an den Einträgen vorgenommen, weshalb eine Re-Synchronisation erforderlich ist. Diese findet als absoluter oder inkrementeller Transfer ebenfalls über TCP-Port 53 (dns) statt. Die Notify-Methode gilt gemeinhin als effizienter, da sie ressourcenschonender ist sowie eine weitaus zeitnahere Abgleichung erlaubt.
Im Gegenzug arbeiten nicht-autoritative Server als Mittelsmänner. Die von ihnen angebotenen Informationen gelten nicht als gesichert, da sie die Daten aus zweiter oder dritter Hand beziehen. Ihre Hauptaufgabe ist, einmal durch den Resolver getätigte Auflösungen im Cache zwischenzuspeichern und damit die Performance entsprechender Abfragen zu optimieren. Jeder Eintrag enthält dabei eine TTL (Time to Live; dt. Lebenszeit), nach deren Ablauf der Datensatz wieder gelöscht und bei der nächsten Anfrage erneut eingeholt und zwischengespeichert werden muß. Da die Informationen im Domain Name System nur wenigen Änderungen unterworfen sind, gibt es nur in den seltensten Fällen Probleme wegen der dadurch eingeführten Latenz. Für den alltäglichen Gebrauch hat also diese Mechanik entscheidende Vorteile. Damit ein nicht-autoritativer Server seiner Aufgabe der Auflösungen von IP-Adressen und Hostnamen nachkommen kann, bedient er sich drei unterschiedlicher Strategien:

- Delegierung (engl. delegation): Manchmal werden Teile von Domains an Subdomains ausgelagert. Sie können dann als verteiltes System mit separaten Nameservern betrieben werden. Der Nameserver der Zone kennt die jeweiligen Nameserver der Subdomains und kann die Anfragen an sie abgeben. Eine Lastenverteilung dieser Art findet oft besonders bei großen Namensräumen statt.
- Weiterleitung (engl. forwarding): Wird an einen Nameserver eine Anfrage geschickt, die außerhalb seines Zuständigkeitsbereichs liegt, wird sie an einen fest definierten Nameserver weitergeleitet.
- Auflösung über Root-Server: Falls kein Weiterleitungs-Server definiert wurde oder er im Rahmen der Anfrage nicht antwortet, findet die Auflösung über einen der Root-Server statt. Gegenwärtig gibt es dreizehn DNS-Root-Server, deren Bezeichnungen von A bis M reichen. Diese zentralen Server beantworten ausschließlich iterative Anfragen, da sie sonst hoffnungslos mit der Anzahl der Zugriffe überlastet wären. Ihre Anzahl ist durch den auf 512 Byte begrenzten Datenteil des Transportprotokolls UDP beschränkt.

In größeren und oft frequentierten Umgebungen wird entsprechend oft mit Hostnamen gearbeitet. Sie erleichtern die Arbeit enorm, da man sich als Benutzer nicht die IP-Adressen der einzelnen Systeme merken muß. Dabei befolgen Administratoren bestimmte Namenskonventionen, die das Auswerten einer Umgebung mit einer entsprechenden Systematik vereinfachen. Globale Namenskonventionen sind durch die typischen öffentlichen Dienste gegeben:

- Webserver: So wird der Hostname *www* gerne für den öffentlichen Webserver verwendet (beispielsweise www.computec.ch als Webserver der Domain computec.ch). Alternative Hostnamen, wenn mehrere Webserver zum Einsatz kommen, sind *www2* und *web*. Für sichere Webserver, die primär oder gar ausschließlich über HTTPS erreichbar sind, wird auch gerne *wwws* (World Wide Web Secure) oder *secwww* als Hostname genutzt.

- Mailserver: Grundsätzlich gleich verhält es sich bei Mailserver-Systemen. Der generische Name hierfür lautet *mail* (beispielsweise mail.computec.ch). Kommen verschiedene Systeme für das Angebot des Mailversands über SMTP und die Mailabholung über POP3 zum Einsatz, lauten die Hostnamen auch oft smtp.computec.ch und pop.computec.ch. Eine alternative Darstellungsform für letztere wäre pop3.computec.ch.
- Firewall-Systeme: Eine andere gängige Konvention gilt für Firewalls. Dort sind Hostnamen wie *firewall* oder *fw* naheliegend. Vor allem Perimeter-Firewalls werden aber auch als *gateway* oder *gw* bezeichnet. Liegen mehrere Firewalls vor oder besitzen die jeweiligen Systeme multiple Schnittstellen, können sie auch durchnumeriert werden (beispielsweise {fw1, fw2, …, fw5}). Zu weiteren Details zu Firewall-Elementen sei auf Kapitel 11 verwiesen.
- Netzwerkelemente: In ähnlicher Form wie Firewalls werden manchmal auch Router benannt, oft als *router, route* oder *rtr*. Eine Durchnumerierung der einzelnen Elemente ist besonders bei Providern sehr beliebt, die eine Vielzahl entsprechender Netzwerkgeräte betreuen müssen. Dabei verwenden viele Provider den geographischen Namen (beispielsweise router-zuerich.provider.example) oder den Netzbereich (beispielsweise router-192-0-0-0.provider.example) als Indikator. In vielen Umgebungen werden auch die VLANs im Hostnamen deklariert (beispielsweise vlan23.provider.example).

Viele Netzwerkadministratoren pflegen ebenfalls eine interne Namensstruktur für Systeme. Beispielsweise können Planetennamen oder Namen von Charakteren aus Filmen als Hostnamen herhalten. Lautet der Name eines Firewall-Systems *gandalf*, könnte sich hinter dem Host mit dem Namen *gimli* ein Mailserver verbergen. Das sind Figuren aus dem Fantasy-Trilogie »The Lord of the Rings« von J.R.R. Tolkien. Wird eine solche Namenskonvention durchschaut, lassen sich relativ einfach weitere potentielle Ziele ermitteln. In diesem Fall kämen wahrscheinlich Hostnamen wie *aragorn*, *frodo* oder *arwen* in Frage.

Grundsätzlich ist es naheliegend, durch eine Adreßauflösung die IP-Adresse eines Hostnamens zu ermitteln. Die verschiedenen Betriebssysteme stellen unterschiedliche Utilities für diese Aufgabe zur Verfügung. Bei älteren Linux-Systemen und unter Microsoft Windows kommt das zeilenbasierte Tool *nslookup* zum Tragen. Ihm wird beim Programmaufruf der aufzulösende Hostname als erstes Argument übergeben. Dann werden die in der Netzwerkkonfiguration des Systems definierten Nameserver abgefragt und das Ergebnis der Auflösung dargestellt.

Auf jüngeren Linux-Systemen gibt es das *host*-Kommando, dessen Verhaltensweise der von nslookup ähnelt. Auch hier werden bei einem Programmaufruf der aufzulösende Hostname des Systems angegeben (Zeile 01) und die Resultate des Zugriffs ausgegeben (Zeile 02). Im nachfolgenden Beispiel findet eine Namensauflösung für den Host mit dem Hostnamen www.computec.ch statt. In der Ausgabe wird als dessen IP-Adresse 80.74.129.35 ermittelt. Je nachdem, wie die Zielumgebung eingerichtet ist, könnten umliegende IP-Adressen im Zielbereich ebenfalls von der Zielorganisation zur Verfügung gestellt werden. Eine intensive whois-Abfrage für diese IP-Adresse, so wie sie in Kapitel 3.3.2 besprochen wurde, bietet sich an.

```
01  mruef@debian:~$ host www.computec.ch
02  www.computec.ch         A       80.74.129.35
```

Das nächste Beispiel demonstriert die gleiche Form der Namensauflösung, dieses Mal aber für den Hostnamen www.google.com. Es ist zu sehen, daß für diesen zwei unterschiedliche IP-Adressen ausgewiesen werden (Zeilen 03-04). Dies deutet auf die Nutzung eines Server-Clusters hin, der durch den redundanten Einsatz gleichwertiger Websysteme eine Lastverteilung ermöglicht. Anhand dieser Information kann also schon nur von einem Hostnamen auf zwei potentielle Ziele geschlossen werden.

```
01  mruef@debian:~$ host www.google.com
02  www.google.com          CNAME   www.l.google.com
03  www.l.google.com        A       66.249.93.104
04  www.l.google.com        A       66.249.93.99
```

Außerdem können nun Namensauflösungen für die zuvor besprochenen Namenskonventionen durchgeführt werden. Es ist naheliegend, daß ebenfalls mail.computec.ch, smtp.computec.ch sowie pop.computec.ch einer Namensauflösung unterzogen werden. Ist sie erfolgreich, ist ein potentielles Ziel lokalisiert und damit werden weitere Eckdaten zur Zielumgebung eingeholt. Schnell lassen sich so weitere Adreßbereiche ausmachen und damit die Angriffsfläche einer Zielumgebung erweitern.

4.1.2 Erweiterte DNS Zonetransfers

Eine klassische Schwachstelle von Nameservern ist, daß diese komplette Zonetransfers, wie sie zur Synchronisation von Master und Slave stattfinden (Kapitel 4.1.1), für jedermann zulassen (CVE-1999-0532). Dieses Problem besteht bei etwa einem Prozent aller professionell betriebenen Umgebungen. Durch solche Zonentransfers ist es möglich, in den Besitz sämtlicher IP-Adressen und Hostnamen einer Zone zu kommen. Dies entspricht schon fast einer detaillierten Karte des Netzwerks, durch die sich das weitere Vorgehen minutiös vorbereiten läßt. Diese generische Nameserver-Schwachstelle wird von den meisten automatisierten Vulnerability Scannern erkannt. Nessus faßt das Problem im Plugin 10595 (DNS AXFR) wie folgt zusammen:

The remote name server allows DNS zone transfers to be performed. A zone transfer will allow the remote attacker to instantly populate a list of potential targets. In addition, companies often use a naming convention which can give hints as to a servers primary application (for instance, proxy.company.com, payroll.company.com, b2b.company.com, etc.).
As such, this information is of great use to an attacker who may use it to gain information about the topology of your network and spot new targets.

Das Nessus-NASL-Plugin ist sehr komplex und mindestens so effizient. So werden die Datenpakete, wie sie an den Nameserver auf TCP-Port 53 (dns) geschickt werden sollen, manuell zusammengestellt. Ein derartiger DNS Zonentransfer ohne zusätzliche Scanning-Tools kann mit einer Software wie *nslookup* realisiert werden. Das nun vorgestellte Vorgehen ist eine klassische Möglichkeit der Auswertung einer Zielumgebung:

```
01  mruef@debian:~$ nslookup
02  Default Server: ns1.computec.ch
03  Address: 192.168.0.13
```

```
04  server 192.168.0.14
05  Default Server: [192.168.0.14]
06  Address: 192.168.0.14
07  set type=any
08  ls -d computec.ch. >/tmp/zone.txt
```

Ein-/Ausgabe	Zeile	Erläuterung
Aufruf von nslookup	01	*nslookup* ist das traditionelle zeilenbasierte Utility zur Umsetzung von Hostname-Abfragen. Es wurde auf Linux-Systemen vorwiegend von den Befehlen *dig* und *host* abgelöst (Kapitel 4.1.1), bei Microsoft Windows hingegen wird nach wie vor die hauseigene Implementierung von nslookup genutzt.
Angabe des genutzten Nameservers	02-03	Direkt nach dem nslookup-Aufruf zeigt die Software an, welcher Nameserver genutzt wird. Das ist der als erster in der Netzwerkkonfiguration des lokalen Systems eingetragene DNS-Server. In diesem Beispiel ist dies der Nameserver mit dem Hostnamen ns1.computec.ch (192.168.0.13).
Ändern des abgefragten Nameservers	04-06	Wird nslookup ohne Parameter aufgerufen, startet es im interaktiven Modus. Dort wird eine autonome Kommandozeile zur Verfügung gestellt, die dedizierte Eingaben entgegennimmt. Mit dem *server*-Kommando kann ein alternativer Nameserver definiert werden. Für den Zonetransfer muß natürlich ein Nameserver genutzt werden, der in der Zielumgebung als autoritativ gilt, entsprechend die interessanten Informationen verwaltet und auch abgefragt werden kann. In diesem Beispiel wird zur Referenzierung des zweiten Nameservers die IP-Adresse 192.168.0.14 angegeben (Zeile 04). Nach erfolgreicher Annahme dieser Eingabe gibt nslookup die neu referenzierten Daten zurück (Zeilen 05-06).
Ändern der abzufragenden Typen	07	Mit *set type* werden die abzufragenden DNS-Ressourceneintragstypen angegeben. Der Wert A läßt zum Beispiel Host-IP-Adressen auflösen, mit CNAME wird ein kanonischer Name für ein Alias ausgegeben und mit MX wird der Mail-Exchanger, voraussichtlich mail.computec.ch, identifiziert. Durch das Attribut *any* werden sämtliche Typen ausgelesen. Die erfolgreiche Eingabe dieses Kommandos wird in nslookup nicht bestätigt. Wird jedoch ein nicht-existenter Typ angegeben, wird mit einer einfachen Fehlermeldung abgebrochen (»unknown query type«).
Durchführen des Zonetransfers für computec.ch.	08	*ls* fordert ein Listing (dt. Auflistung), das entspricht der gleichen Eingabe in Linux-Shells, an. Die Option *-d* gibt sämtliche Einträge der abgefragten Domain aus. Gleichen Effekt hätte die Option *-t ANY*. Die Ausgabe dieses Zugriffs wird durch die spitze Klammer direkt in die Datei */tmp/zone.txt* umgeleitet. Mit einem Texteditor oder unter Unix/Linux anhand der Befehle *less* beziehungsweise *more* kann später eine ausführliche Analyse der damit zusammengetragenen Informationen vorgenommen werden.

Tabelle 4.1: Aufruf von nslookup

In einem solchen Zonetransfer können ganz unterschiedliche Daten festgehalten sein. Wie die nachfolgende Zusammenfassung zeigt, werden Hostnamen für die verschiedensten Systeme ausgegeben. Zum Beispiel können mehrere Mailserver (MX) mit unterschiedlichen Prioritäten betrieben werden. Fällt das System mit der höheren Priorität (10) aus, kann der Dienst sofort durch das Failover-System (20) übernommen werden. Außerdem befinden sich darin optional als HINFO-Einträge zusätzliche Details zu den jeweiligen Systemen. Von dieser Option machen aber wegen des erhöhten Aufwands nur die wenigsten Administratoren Gebrauch. Ist man im Besitz eines vollständigen Zonetransfers, bietet sich damit eine ausgezeichnete Ausgangslage für weitere Zugriffe auf die Zielumgebung.

```
01  [localhost]
02  $ORIGIN linux.test.
03  @                       1D IN SOA       ns hostmaster (
04                                          199802151       ; Seriennummer
05                                          8H              ; refresh
06                                          2H              ; retry
07                                          1W              ; expiry
08                                          1D )            ; minimum
09
10                          1D IN NS        ns
11                          1D IN NS        ns2.computec.ch.
12                          1D IN TXT       "Nameserver 2"
13                          1D IN MX        10 mail
14                          1D IN MX        20 mail2.computec.ch.
15  rtr                     1D IN A         192.168.0.254
16                          1D IN HINFO     "Cisco" "IOS"
17                          1D IN TXT       "Der Router"
18  mail                    1D IN A         192.168.0.20
19                          1D IN MX        10 mail
20                          1D IN MX        20 mail.computec.ch.
21                          1D IN HINFO     "AMD64" "Debian GNU/Linux"
22  localhost               1D IN A         127.0.0.1
23  www                     1D IN CNAME     ns
24  ns                      1D IN A         192.168.0.2
25                          1D IN MX        10 mail
26                          1D IN MX        20 mail.computec.ch.
27                          1D IN HINFO     "x86" "Debian GNU/Linux"
```

4.1.3 whois-Abfragen

Durch whois-Abfragen können zwar nicht direkt IP-Adressen in umfangreicher Weise ermittelt werden, es lassen sich mit diesem Dienst jedoch Adreßbereiche ausfindig machen und damit Rückschlüsse auf die eingesetzten Systeme ziehen. Als Voraussetzung und Grundlage dieses Vorgehens gilt der in Kapitel 3.3 besprochene Umgang mit dem whois-Client unter Linux. In diesem Kapitel werden die Hintergründe des whois-Dienstes sowie die grundlegenden Möglichkeiten der Nutzung besprochen. Hier geht es nun ausdrücklich um das Miteinbeziehen von whois zur Ermittlung von Netzwerkbereichen und potentiellen Zielen.

In der Regel ist während eines netzwerkorientierten Tests mindestens ein Zielobjekt identifizierbar. Meistens handelt es sich dabei um ein exponiertes System, dessen Hostname bekannt ist, zum Beispiel ein öffentlich im Internet erreichbarer Webserver oder eine Perimeter-Firewall. Durch eine IP-Adreßauflösung mittels DNS (Kapitel 4.1.3) kann die entsprechende IP-Adresse in Erfahrung gebracht werden. Damit ist das Ziel eindeutig identifiziert und kann in zukünftigen Interaktionen direkt adressiert werden.
Meistens befindet sich in der Zielumgebung jedoch nicht nur ein einzelnes System, vielmehr sind gerade in Unternehmen verschiedene Hosts und Netzwerkelemente im Einsatz. Entsprechend ist es vorteilhaft, weitere möglichen Ziele in der Zielumgebung zu kennen. Durch eine whois-Abfrage läßt sich der von einer Organisation registrierten IP-Adreßbereich identifizieren und damit das Umfeld abstecken. Eine solche Abfrage kann ganz einfach als eine übliche Netzwerkabfrage, Details dazu siehe Kapitel 3.3.2, umgesetzt werden. Ein solcher Zugriff und die damit gewonnenen Ergebnisse wird nachfolgend am Beispiel von Google vorgeführt.

```
01  mruef@debian:~$ whois 66.249.85.99
02
03  OrgName:    Google Inc.
04  OrgID:      GOGL
05  Address:    1600 Amphitheatre Parkway
06  City:       Mountain View
07  StateProv:  CA
08  PostalCode: 94043
09  Country:    US
10
11  NetRange:   66.249.64.0 - 66.249.95.255
12  CIDR:       66.249.64.0/19
13  NetName:    GOOGLE
14  NetHandle:  NET-66-249-64-0-1
15  Parent:     NET-66-0-0-0-0
16  NetType:    Direct Allocation
17  NameServer: NS1.GOOGLE.COM
18  NameServer: NS2.GOOGLE.COM
19  Comment:
20  RegDate:    2004-03-05
21  Updated:    2004-11-10
22
23  OrgTechHandle: ZG39-ARIN
24  OrgTechName:   Google Inc.
25  OrgTechPhone:  +1-650-318-0200
26  OrgTechE-Mail:  arin-contact@google.com
27
28  # ARIN WHOIS database, last updated 2006-07-13 19:10
29  # Enter ? for additional hints on searching ARIN's WHOIS database.
```

Ein-/Ausgabe	Zeile	Erläuterung
Whois-Abfrage für 66.249.85.99	01	Eine Netzwerkabfrage mit whois ist einfach. Beim zeilenbasierten Client unter Linux wird als Argument des Programmaufrufs die IP-Adresse angegeben, für deren Adreßbereich die Daten eingeholt werden sollen. In diesem Beispiel ist dies die IP-Adresse 66.249.85.99, die in einer Vorabklärung (Kapitel 4.1.1) einem öffentlichen Webserver von www.google.com zugewiesen werden konnte.
Registrierter IP-Adreßbereich	11	In den Zeilen 11 bis 18 werden die Kerninformationen der Abfrage dargestellt. Dort befindet sich mit der NetRange der Hinweis darauf, welcher IP-Adreßbereich (engl. IP range) registriert wurde. In diesem Fall ist das 66.249.64.0 bis 66.249.95.255.
CIDR-Darstellung des Adreßbereichs	12	Darauf wird die CIDR-Darstellung in der Form 66.249.64.0/19 ausgegeben. Das ist eigentlich nur eine andere Formatierung des mit der Zeile 11 angezeigten IP-Adreßbereichs. Der Netzname wird ebenso angegeben (Zeile 14) wie das hierarchisch darüberliegende Parent Network (Zeile 15). Zudem finden sich Hinweise auf die autoritativen Nameserver dieser Zone (Zeilen 17 bis 18).
Netzname	13	Der Netzname, in diesem Fall GOOGLE, wird angezeigt. Das ist der eindeutige Name, mit dem der IP-Adreßbereich in whois referenziert wird. Der gewählte Name kann Rückschlüsse auf den Bereich zulassen. In diesem Beispiel handelt es sich wohl um das für öffentliche Dienste vorgesehene Netzwerk von Google.
Network Handle	14	Der Network Handle ermöglicht eine weitere eindeutige Identifikation des Netzwerks. Die Darstellungsform ist dabei an die dotted-decimal Schreibweise von IP-Adressen angelehnt.
Parent Network	15	Das Parent Network ist der übergeordnete Netzwerk-Adreßbereich.
Die autoritativen Nameserver	17-18	In diesem Beispiel werden die beiden autoritativen Nameserver der Zone ausgegeben. Diese Information läßt sich für weitere Auswertungen nutzen, weil mit der Nennung dieser Hosts zwei weitere Angriffsziele des Zielnetzwerks vorgeschlagen wurden.

Tabelle 4.2: whois-Abfrage eines IP-Adreßbereichs

Man sollte die umliegenden Netzwerkbereiche der gleichen whois-Abfrage zu unterziehen. Durch das Einholen der Daten für die IP-Adressen 66.249.63.255 sowie 66.249.96.1 kann ermittelt werden, ob sich der IP-Adreßbereich von Google noch über weitere Teile erstreckt und hier quasi eine Segmentierung stattfand. Damit lassen sich weitere Angriffsziele ausmachen.

4.1.4 Suchmaschinen abfragen

Eine Methode zum Auffinden von IP-Adressen, die immer wieder vergessen oder unterschätzt wird, ist das Abfragen von Suchmaschinen. So könnte vorab (beispielsweise Social Engineering oder Auswertung eines SMTP-Headers) eine IP-Adresse der Zielumgebung in Erfahrung gebracht werden. In diesem Beispiel ist sie Teil des privaten IP-Adreßbereichs für Klasse-A-Netzwerke, wie sie in RFC 1918 definiert sind. Damit läßt sich eine Suchmaschine freier Wahl benutzen, um dort nach der IP-Adresse zu suchen. Wie Abbildung 4.1 zeigt, können für die gesuchte IP-Adresse 10.10.1.117 verschiedene Treffer erzielt werden.

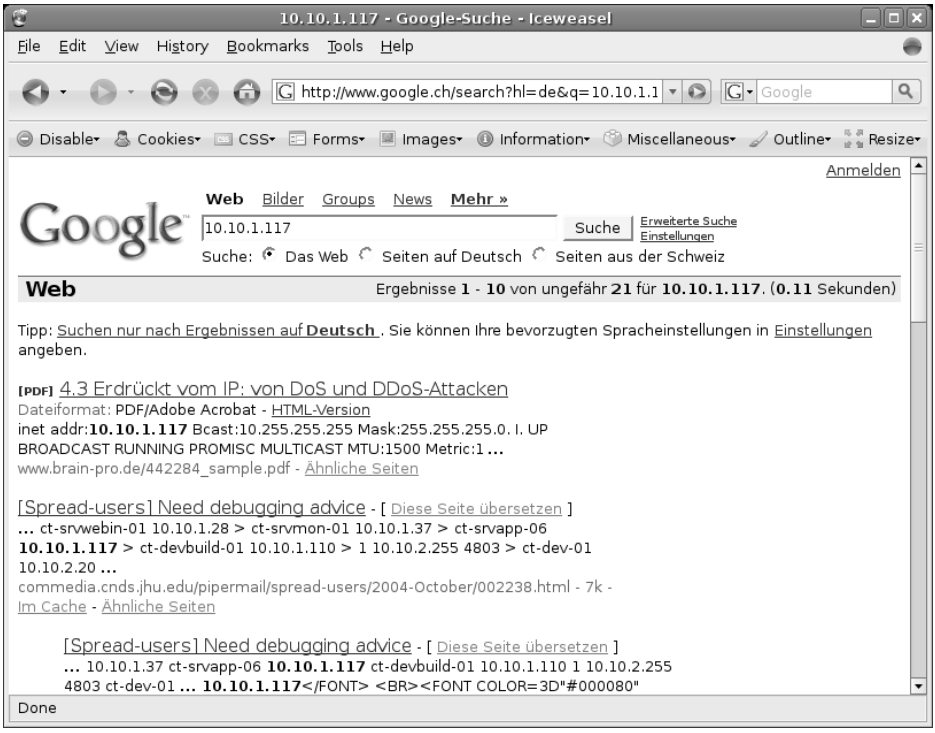

Abbildung 4.1: Suchabfrage mit Google für die IP-Adresse 10.10.1.117

Bei vielen Suchmaschinen wird nicht ohne weiteres das gewünschte Resultat erzielt. Wird bei Google beispielsweise die IP-Adresse eingegeben, ohne sie in Anführungszeichen einzuschließen, wird die Anfrage zu Auswertungszwecken falsch verarbeitet. Das Suchmaschinen-System weist darauf hin, daß zur URL 10.10.1.117 keine Informationen gefunden werden konnten. Google geht also davon aus, daß die IP-Adresse die URL einer Webseite und nicht ein Suchbegriff in einem Webdokument ist. Das ist natürlich in diesem Zusammenhang falsch.

Entweder wird eine angepaßte Anfrage –, die IP-Adresse muß in doppelte Anführungszeichen geschrieben werden – durchgeführt oder auf dem ersten Suchresultat von Google wird auf den Link »Webseiten durchsuchen, die den Begriff 10.10.1.117 enthalten« geklickt. Dann wird automatisch die vorige Eingabe umformatiert.

Dieser Anfrage zielt darauf ab, Informationen zur angegebenen IP-Adresse ausfindig machen zu können. Man hat also die Hoffnung, daß dieses System in einem Dokument erwähnt wird und auf diese Weise zusätzliche Daten zum Zielnetzwerk zusammengetragen werden können. Vor allem größere Organisationen oder Hochschulen dokumentieren ihr Netzwerk umfassend, um den Benutzern per World Wide Web eine komfortable Übersicht zu geben. Dabei werden manchmal Konfigurationshinweise oder Informationen zu den verfügbaren Diensten angeboten. Zudem indizieren eine Vielzahl an Suchmaschinen und Diensten die Postings im Usenet. Da mit der Suchabfrage über eine umfassende Suchmaschine eben auch diese Daten durchforstet werden, werden unter Umständen Hinweise zu

Situationen geliefert, in denen das gesuchte System eine Rolle spielte. Dies ist am ehesten bei Mailserver-Systemen, Antiviren-Gateways, Firewall-Lösungen oder Routing-Systemen der Fall. Also bei eher zentralen, populären und exponierten Mechanismen.
Ein Erfolg stellt sich jedoch sehr selten ein, genauer gesagt mit einer Wahrscheinlichkeit von unter einem Prozent. Eine Abfrage für die sehr beliebte private IP-Adresse 192.168.0.1 bringt bei Google ungefähr 1.740.000 Resultate hervor, bei der Adresse 192.168.0.2 sind es dann nur noch 856.000 (Stand 28. April 2007). Obwohl das alles vielversprechende Treffer sind, werden sich nur die wenigsten wirklich in Verbindung mit dem eigentlichen Zielnetzwerk bringen lassen. Da es sich hierbei um für den privaten Gebrauch reservierte Adreßbereiche handelt, sind diese eben weltweit in einer Vielzahl von LANs gebräuchlich. Das Filtern der Resultate und das Herausfinden von Zusammenhängen ist mit enormen Aufwand verbunden.

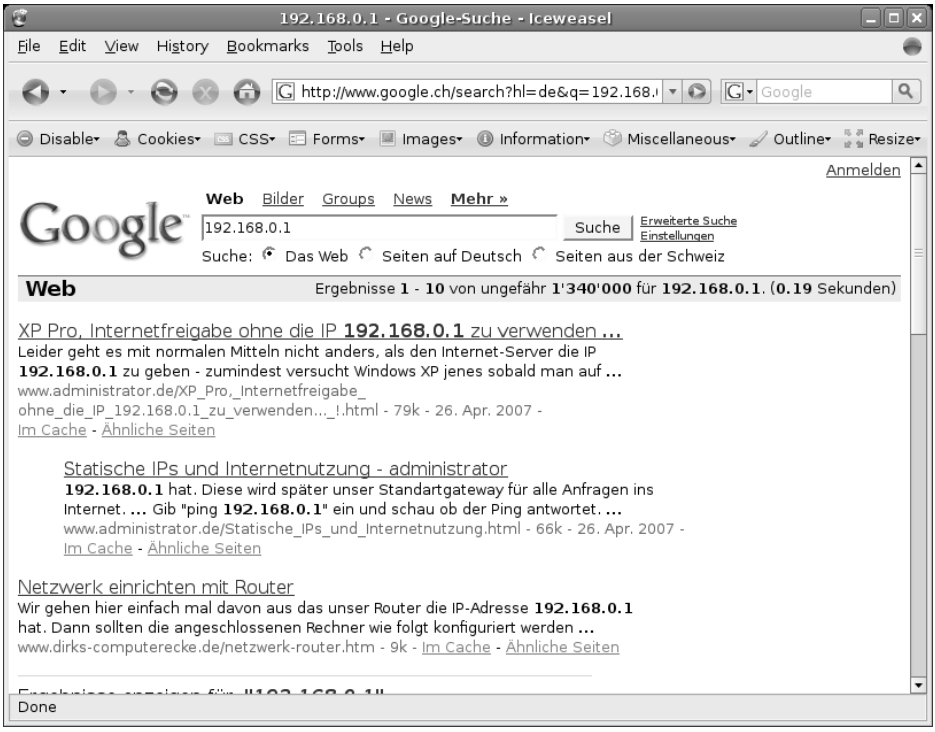

Abbildung 4.2: Suchabfrage mit Google für das populäre 192.168.0.1

Die einzige Möglichkeit zur Optimierung einer derartigen Suche sind zusätzliche Begriffe. Diese Begriffe sollten in möglichst eindeutigem Zusammenhang mit der Zielumgebung stehen. Soll zum Beispiel das System mit der IP-Adresse 192.168.0.1 gefunden werden, das mit der Person »Marc Ruef« in Verbindung gebracht werden kann, ist folgende Suchabfrage bei Google einzugeben:

```
+"192.168.0.1" +"Marc Ruef"
```

Die Pluszeichen (+) geben an, daß sämtliche Begriffe, denen ein solches vorangestellt wurde, im gesuchten Dokument enthalten sein müssen. Die Reihenfolge der Treffer ist dabei jedoch irrelevant. In diesem Fall sind es beide Begriffe, die IP-Adresse und der Name. Damit ergeben sich die beiden folgenden Kombinationen für Suchtreffer:

– IP-Adresse UND Name
– Name UND IP-Adresse

Diese beiden Begriffe werden jeweils in doppelte Anführungszeichen eingeschlossen. Bei der IP-Adresse muß das sein, weil sonst eine URL-Abfrage durchgeführt werden würde. Vor- und Nachname müssen in Anführungszeichen stehen, damit genau diese Wortkombination gesucht wird. Der Name wird damit als einheitliche Zeichenkette (String) verstanden. Ansonsten würden sämtliche Dokumente gefunden, in denen das Wort »Marc« unabhängig vom Wort »Ruef« vorkommt.

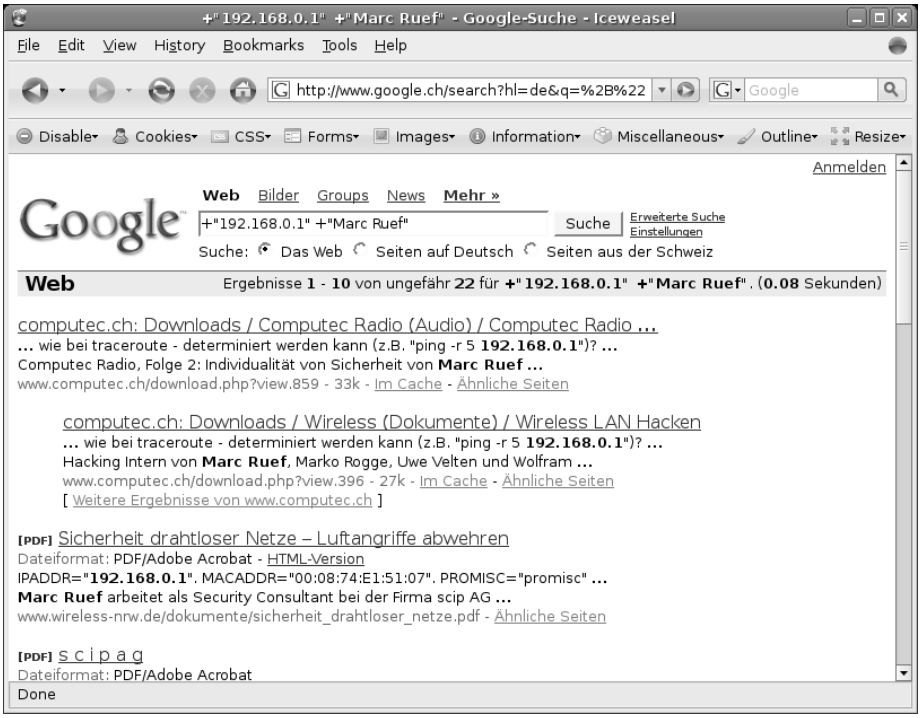

Abbildung 4.3: Suchabfrage für 192.168.0.1 im Zusammenhang mit Marc Ruef

Obwohl das spezifische Suchen von absoluten IP-Adressen denkbar ist, läßt sich die Suche indirekt auch auf absolute Adreßbereiche erweitern. Dabei muß lediglich das letzte Viertel der dotted-decimal Schreibweise weggelassen werden. Soll also nach sämtlichen Hosts gesucht werden, die sich im Adreßbereich zwischen 192.168.0.0 und 192.168.0.255 befinden, muß als Suchbegriff eingegeben werden:

```
"192.168.0."
```

Diese unvollständige Eingabe führt dazu, daß alle möglichen 254 Systeme mit eben dieser IP-Adresse in den Suchresultaten aufgelistet werden. Damit ist zwar das exakte Eingrenzen von IP-Adreßbereichen nicht möglich (beispielsweise nur von 192.168.0.10 bis 192.168.0.20). Jedoch läßt sich dadurch in gewisser Art eine ungenaue Suche, eine sogenannte Fuzzy Search, durchführen. Dies kann aufgrund der fehlenden Einengung mehr Treffer und damit mögliche Ziele ergeben.

4.1.5 E-Mail und Postings untersuchen

Eine sehr klassische, passive Methode des Herausfindens von IP-Adressen einer Zielumgebung ist das Analysieren von Mailheadern und Headern von Usenet-Postings. Die dabei angewandten Protokolle SMTP und NNTP bringen beim Versand von Nachrichten erweiterte IP-Adreßinformationen im Header der Mitteilung unter. Durch das Auswerten dieser Daten können Rückschlüsse auf den Pfad der Nachricht sowie den Absender gezogen werden. Diese Technik ist passiv, weil in den meisten Fällen keine erstmalige oder zusätzliche Interaktion mit dem Opfer nötig ist. Mit ein bißchen Glück kann man verschickte Mitteilungen einsehen. Es kann trotzdem Situationen geben, in denen man nicht auf die gewünschten Daten zurückgreifen kann. Dann wandelt sich die ansonsten passive Methode zu einer aktiven Technik: Es gilt, in einem ersten Schritt eine Antwort des Gegenübers zu provozieren. Das kann mit der Hilfe einer neckischen Aufforderung, die sich am besten per Mail zustellen läßt, geschehen. Das Schreiben wird dabei in der Hoffnung verfaßt, daß sich der Empfänger zu einer Antwort hinreißen läßt. Dieses Ziel läßt sich durch die Mittel des Social Engineerings erreichen.

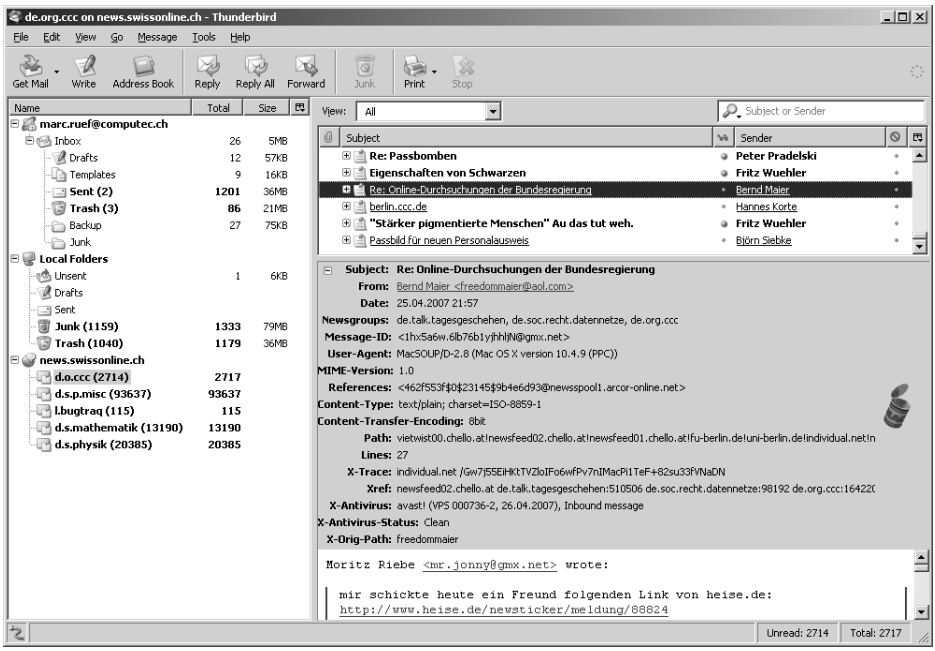

Abbildung 4.4: Nach ähnlichem Prinzip funktionieren Usenet-Postings

Ist dieses Zwischenziel erreicht, wird im zweiten Schritt wieder passiv analysiert. Das Einsehen der Mailheader erlauben praktisch sämtliche Mailclients. Meistens lassen sich diese erweiterten Informationen entweder permanent einblenden (beispielsweise Ansicht/ Kopfzeilen/Alle bei Mozilla Thunderbird) oder temporär auslesen (beispielsweise Rechtsklick auf die gewünschte Nachricht, Auswählen von Optionen bei Microsoft Outlook). Webdienste mit der Möglichkeit des Zugriffs auf E-Mails oder Usenet-Postings verhalten sich in etwa ähnlich. Dort kann entweder im Profil eine permanente Einblendung erzwungen oder die Informationen über eine Online-Funktion temporär ausgelesen werden.
Nachfolgend eine Analyse eines typischen Mailheaders. Das Augenmerk liegt dabei nur auf Informationen, die Rückschlüsse auf die IP-Adresse des Gegenübers erlauben. Alle anderen Zeilen des SMTP-Headers, die zum Beispiel für ein Application Fingerprinting von Interesse wären (Kapitel 7), werden an dieser Stelle nicht besprochen.

```
01  From marc.ruef@computec.ch Mon Dec 30 13:03:52 2002
02  Return-Path: <marc.ruef@computec.ch>
03  Received: from mail1.exigo.ch (mail.exigo.ch [195.65.88.10])
04       by mx.hispeed.ch (8.12.6/8.12.6/tornado-1.0) with ESMTP id gBUC6S6h008717
05       for <m.ruef@swissonline.ch>; Mon, 30 Dec 2002 13:06:28 +0100
06  X-Sending-Host: mail.computec.ch
07  X-Sending-IP: 195.65.88.10
08  Received: (qmail 12349 invoked by uid 510); 30 Dec 2002 12:06:28 -0000
09  Delivered-To: marc.ruef@computec.ch
10  Received: (qmail 12347 invoked from network); 30 Dec 2002 12:06:28 -0000
11  Received: from isp247n.hispeed.ch (HELO smtp.hispeed.ch) ([62.2.95.247]) (envelope-
    sender <marc.ruef@computec.ch>)
12            by mail.computec.ch (qmail-ldap-1.03) with DES-CBC3-SHA encrypted SMTP
13            for <marc.ruef@computec.ch>; 30 Dec 2002 12:06:28 -0000
14  Received: from there (dclient217-162-157-195.hispeed.ch [217.162.157.195])
15       by smtp.hispeed.ch (8.12.6/8.12.6/tornado-1.0) with SMTP id gBUC60d6016398
16       for <marc.ruef@computec.ch>; Mon, 30 Dec 2002 13:06:26 +0100
```

Ein-/Ausgabe	Zeile	Erläuterung
Hinweise auf den Urheber der E-Mail	01-02	Die Zeilen *From* und *Return-Path* lassen Rückschlüsse zu, von wem das E-Mail verfaßt wurde (marc.ruef@computec.ch) und wie dieser Benutzer per E-Mail zu erreichen ist (ebenfalls marc.ruef@computec.ch). Der Domänenname dieser Mailadresse ist wertvoll, denn damit lassen sich Namensauflösungen (Kapitel 4.1.1) und whois-Abfragen (Kapitel 3.3.2) einleiten. Damit kann zumindest der IP-Adreßbereich der Zielumgebung eingegrenzt werden. Es ist jedoch nicht gesagt, daß sich der Benutzer im gleichen IP-Adreßbereich befindet, wie der Domänenname registriert ist. Sehr wenige Benutzer gönnen sich den Luxus einer eigenen Domäne mit persönlichem Adreßbereich. Dies gilt zum Beispiel für die meisten Anwender, die auf Angebote mit dynamischen IP-Adressen (beispielsweise Dial-up oder ADSL) zurückgreifen. Nur Anwender oder Unternehmen mit Standleitungen und festen IP-Adressen kommen als Besitzer für IP-Adreßbereiche in Frage.

Tabelle 4.3: SMTP-Mailheader (Teil 1 von 2)

Ein-/Ausgabe	Zeile	Erläuterung
Angabe des ersten SMTP-Relays	14-16	Die erste SMTP-Zwischenstation des E-Mails. Zeile 14 ist dabei die eigentliche Goldgrube des Mail-Headers: Das erste SMTP-Relay hat hier ebenso freundlicherweise die IP-Adresse des Mail-Clients dokumentiert (217.162.157.195). Damit sind zwei wichtige Informationen zur Zielumgebung eingeholt. Nun sind weitere Zugriffe auf den Mailclient sowie den Mailserver denkbar.

Tabelle 4.3: SMTP-Mailheader (Teil 2 von 2)

Nicht alle Mailheader sind für die Auswertung von IP-Adressen gleich geeignet. Die Angabe des ersten SMTP-Relays ist jedoch Pflicht, damit der Versand von E-Mails über das Internet überhaupt funktioniert. Ein zusätzlicher Bonus ist, wenn der SMTP-Header der Nachricht auch die IP-Adresse des Senders beziehungsweise seines Mailclients enthält. Einige populäre Freemailer vermerken diese Information ebenfalls im Mailheader, um den Versand von anonymen Nachrichten sowie Spam zu erschweren. Gerade wenn solche Angebote von einer Zielperson genutzt werden, kann die Auswertung besonders lukrativ sein. In Tabelle 4.4 werden einige populäre Dienste und Provider mit ihrem Verhalten bezüglich des automatischen IP-Adreßvermerks aufgelistet (Stand Dezember 2006).

Anbieter	Automatischer IP-Adressen-Vermerk
Arcor.de Webinterface	In separater X-Webmailclient-IP-Zeile
Hotmail.com Webinterface	In separater X-Originating-IP-Zeile
GMX SMTP-Server	Erstes SMTP-Relay
GMX Webinterface	In separater X-Authenticated-IP-Zeile
Google Gmail	Keiner
Yahoo.com Grußkarten	Erstes SMTP-Relay
Yahoo.de Webinterface	Erstes SMTP-Relay

Tabelle 4.4: Mail-Anbietern vermerken IP-Adressen

4.1.6 Social Engineering: Direktverbindungen

Eine klassische und in der Internet-Literatur oft diskutierte Form, die IP-Adresse eines Zielobjekts in Erfahrung zu bringen, ist die Provokation von Direktverbindungen. Hierbei wird die Zielperson dazu verleitet, eine IP-Verbindung zu einem unter der Kontrolle des Angreifers stehenden Objekt aufzubauen. Während der Verbindung können beide Seiten durch systeminterne Befehle (beispielsweise *netstat*) oder Software von Drittherstellern (beispielsweise *iptraf, tcpview* oder *tcpdump*) die bestehenden Verbindungen betrachten und Details ausgeben lassen. Damit lassen sich die IP-Adresse und der genutzte Port des Gegensystems identifizieren. In folgendem Beispiel dient hierfür das Kommando *netstat* (Windows und Unix/Linux):

```
01  C:\Dokumente und Einstellungen\mruef>netstat -np TCP
02
03  Aktive Verbindungen
04
05    Proto  Lokale Adresse          Remoteadresse          Status
06    TCP    192.168.0.11:80        192.168.0.12:2183      HERGESTELLT
07    TCP    192.168.0.11:2259      192.168.0.1:53         SCHLIESSEN_WARTEN
```

Ein-/Ausgabe	Zeile	Erläuterung
Ausführen der netstat-Analyse	01	*-n* veranlaßt eine numerische Darstellung, auf Namensauflösung wird verzichtet. Mit *p* wird die Ausgabe auf bestimmte Protokolle beschränkt, das Argument *TCP* weist nur die TCPv4-Verbindungen, die in diesem Fall relevant sind, aus.
Anzeige der eingehenden Webserver-Verbindung	06	Erste bestehende Verbindung. Die erste Spalte (Proto) zeigt das genutzte Protokoll, aufgrund des zuvor mit *-p* definierten Protokollfilters ist sowieso nur mit TCP zu rechnen, an. In der zweiten Spalte (lokale Adresse) steht die Adresse der lokalen Schnittstelle, in diesem Beispiel die der ersten Ethernet-Schnittstelle zugewiesene IP-Adresse 192.168.0.11. Als Port wird der well-known Port 80 ausgewiesen, der Standard für Webserver (HTTP). In der dritten Spalte (Remoteadresse) ist der Socket-Endpunkt des Remote-Systems, an diesem ist man bei einer solchen Auswertung interessiert, angegeben, hier die IP-Adresse 192.168.0.12 und der kurzlebige Port 2183. In der vierten Spalte wird der Status der Verbindung angezeigt. Dies sind die typischen Stati im Rahmen von TCP (Kapitel 5.3.1). Der Status HERGESTELLT (engl. ESTABLISHED) zeigt an, daß eine Verbindung zwischen den beiden Punkten im Rahmen von TCP etabliert wurde und nun ein Datentransfer stattfinden kann. Das ist also das Zielsystem, das zu einer Verbindungsaufnahme verleitet wurde. Dessen IP-Adresse ist somit identifiziert.
Anzeige einer anderen ausgehenden Verbindung	07	Eine ausgehende Verbindung. Hier wird wieder als Protokoll TCP angegeben. Die lokale IP-Adresse 192.168.0.11 ist auch hier die erste Ethernet-Schnittstelle. Dabei kommt im Quell-Socket ein kurzlebiger Port (2259) zum Einsatz, was auf eine ausgehende Verbindung hinweist. Als Zielsystem wird die IP-Adresse 192.168.0.1 und als Zielport 53 (DNS) angezeigt. Dies deutet auf einen DNS-Zonetransfer oder eine umfassende DNS-Rückantwort hin, die vom lokalen Host beim Zielsystem initiiert wurde. Als Status wird jedoch SCHLIESSEN_WARTEN (engl. FIN_WAIT) angezeigt. Dieser gilt dann, wenn eine Seite die Verbindung mit einem FIN-Segment schließen will, das Gegenüber dies aber noch nicht mit einem ACK-Segment bestätigt hat und ebenfalls ein FIN-Segment zum kompletten Beenden der Beziehung sandte. Dies wird als halbgeschlossene Verbindung (engl. half-closed) bezeichnet und ist näher in Kapitel 5.3.1 illustriert.

Tabelle 4.5: Offene Verbindungen unter Windows

Die klassische Provokation einer Direktverbindung wurde im Zusammenhang mit der Chat-Software ICQ angewandt. Ebenso einfach und bisweilen gar noch komfortabler ist

das Provozieren einer Direktverbindung durch einen lokal installierten Webserver. Dieser wird aufgesetzt, damit er für das Zielsystem erreichbar ist. Eine Installation im Internet mit einer offiziellen IP-Adresse ist die beste Lösung, da der Zugriff unkompliziert und direkt erfolgen kann. Grundsätzlich ist es dabei von Vorteil, wenn sämtliche Verbindungen zu diesem System protokolliert werden. Die meisten modernen Webserver-Implementierungen bieten standardmäßig ein solches Feature in Form von Logdateien an. Notfalls kann man auch einen Protokoll-Analyzer (beispielsweise *tcpdump*) installieren, der den Datenverkehr für die nachträgliche Analyse aufzeichnet. Das ist natürlich, sofern keine restriktiven Filter zum Einsatz kommen, mit einem sehr hohen zu speichernden Datenvolumen verbunden.

Die Idee ist nun, daß dieses Webserver-System einen Honeypot anbietet. Das Wort *Honeypot* stammt eigentlich aus dem Intrusion-Detection-Umfeld. Dort werden Systeme installiert, die dem Angreifer besonders lukrativ erscheinen und deshalb oft penetriert werden. Die daraus resultierenden Attacken und Einbrüche werden aufgezeichnet und analysiert. Damit sollen Methoden, Vorgehensweisen und Strategien von Angreifern untersucht werden, um neue Attacken und Trends zu entdecken. Der in diesem Kapitel vorgestellte Webserver wird jedoch nicht in erster Linie als klassischer IDS-Honeypot genutzt, vielmehr soll ein Angebot geschaffen werden, das vom Benutzer des Zielsystems automatisch oder manuell aufgerufen wird. Der damit provozierte Zugriff, er ist hier legitimer Natur, soll die Auswertung der IP-Adressen ermöglichen.

Eine sehr einfache Standardlösung läßt sich mit einem Apache-Webserver auf unserem Debian GNU/Linux realisieren. Durch die Eingabe von *apt-get install apache2* läßt sich die Version 2.x unkompliziert einrichten. Der Dienst wird dabei standardmäßig an den well-known Port tcp/80 (HTTP) gebunden. Das System kann sich dabei im lokalen LAN befinden. Durch Port-Forwarding auf der Perimeter-Firewall kann es später auch über das Internet angesprochen werden. Für diese Zwecke eignet sich auch jede andere Server-Software, die die Protokollierung von Verbindungen unterstützt. Webserver sind jedoch aus verschiedenen Gründen bestens für das Vorhaben geeignet:

- Einfachere Provokation: Eine Zielperson vom Besuch eines Webangebots zu überzeugen, ist viel einfacher, als eine Verbindung zu einem Mailserver oder einem komplett unbekannten Dienst zu provozieren. Die Zielperson muß unter Umständen lediglich einem einfachen Link folgen und muß sich nicht dafür anstrengen, sich unbewußt zu exponieren. Das in den letzten Jahren stark popularisierte Phishing arbeitet auf diese Weise.
- Populäre Ports: In den meisten Umgebungen mit Firewalling sind Verbindungen zum Zielport tcp/80 erlaubt. Würde der Webserver an einen anderen untypischen Port gebunden (beispielsweise tcp/82) oder eine andere Server-Applikation wählen (beispielsweise SMTP-Mailserver auf TCP-Port 25), sind die Chancen hoch, daß der Zielrechner (der Client des Opfers) den angebotenen Dienst nicht erfolgreich ansprechen kann. Alternativ kann auch auf den Port tcp/443, der für HTTPS vorgesehen ist, zurückgegriffen werden.
- Einfaches Einrichten: Die Installation und Wartung eines funktionierenden Webserver-Angebots ist mit relativ wenig Aufwand verbunden. Die meisten versierten Benutzer oder Administratoren sind mit der Funktionalität von Lösungen wie Apache bestens vertraut.

– Automatismen: Das im World Wide Web genutzte HTTP-Protokoll ist verhältnismäßig non-interaktiv: Auf eine Anfrage (engl. request) wird mit einer Antwort (engl. response) reagiert. Würde auf SMTP (Simple Mail Transfer Protocol) gesetzt werden, müßte die Umgebung mit einem Mehr an Interaktivität umgehen können. Das soll nicht das Ziel sein und viele Situationen während einer bestehenden Verbindung mit dem Zielobjekt erlauben dies auch gar nicht. Es ist viel einfacher, eine simple URL aufzurufen, anstatt komplexe Kommandoeingaben in einem Makro unterzubringen (beispielsweise HELO, MAIL FROM, RCPT TO, DATA, QUIT).

Grundsätzlich ist es am einfachsten, wenn der Benutzer auf eine vermeintlich interessante oder wichtige Webseite gelockt wird. Technisch gesehen bedeutet das, daß die Zielperson mit ihrem Computer eine direkte Verbindung zum Rechner herstellen muß, deren Protokolldateien vom Auditor eingesehen werden können. Dieser Zugriff geschieht normalerweise mit einem Webbrowser oder einer Software mit ähnlicher Funktionalität (beispielsweise erweiterter Mailclient).

Es ist nun eine Frage des Charakters, welche Webseite für die Zielperson die größte Anziehungskraft auszuüben in der Lage ist. Psychologisch gesehen, werden durch »niedere Instinkte«, wie zum Beispiel dem besonders beim männlichen Homo sapiens sehr ausgeprägten Sexualtrieb, am meisten Energie entwickelt. Es liegt also nahe, daß man dem Honeypot-Angebot eine erotische und unterhaltsame Färbung gibt. Am einfachsten ist es, wenn man eine anziehende Erotikseite als leicht modifizierte Kopie auf seinem Webserver zur Verfügung stellt.

Abbildung 4.5: Eine Erotikseite als Kopie

In professionellen Überprüfungen in Firmen wird jedoch vorwiegend von einer sexuellen Komponente abgesehen, sondern es wird anhand von Identitätsdiebstahl eine E-Mail mit falschem Absender verschickt. Darin werden die Kollegen auf ein spezielles Angebot (beispielsweise Vergünstigungen zur Weihnachtszeit) oder anstehende Aufgaben (beispielsweise Zurücksetzen der Benutzerdatenbank) hingewiesen. Die lieben Kollegen versuchen dann, der Bitte des vermeintlich authentischen Kollegen nachzukommen.
Die Analyse der durch das manipulierte Opfer aufgebauten, bestehenden und abzubauenden Verbindungen mit einem Netzwerkdiagnoseutility wie tcpdump ist relativ einfach und schnell. Problematisch dabei ist, daß der Analyse-Zugriff genau dann erfolgen muß, nachdem das ersten SYN-Segment von der Zielperson oder bevor das letzte FIN- beziehungsweise RST-Segment verschickt wurde. Das Treffen dieses Zeitpunkts ist relativ schwer, wenn nicht unmöglich. Eine Übersicht dieses komplexen Mechanismus des Transportprotokolls befindet sich in Kapitel 5.3.1.
Aus diesem Grund lohnt sich als Alternative die Durchsicht der Protokolldateien des Serversystems. Jede professionelle Lösung erlaubt die Protokollierung bestehender Verbindungen (oder zumindest die Anfragen). Dies geschieht meistens in einer dedizierten Logdatei oder auf einem vordefinierten Log-Repository (beispielsweise syslog-Server im Netzwerk). Nachdem die Datensammlung während der Zeitdauer der potentiellen Zugriffe stattfand, können dieser Daten später analysiert werden. Die Ermittlung des Zielobjekts ist damit nur noch eine Frage der Zeit. Bei einer Apache2-Installation können die erfolgreichen Zugriffe auf dem Webserver in Echtzeit mit dem *tail*-Kommando auf der Konsole beobachtet werden:

```
01  debian:~# tail -f /var/log/apache2/access.log
02  192.168.0.12 - - [28/Nov/2006:22:40:08 +0100] "GET /fake.php HTTP/1.1" 200 471 "-"
    "Mozilla/5.0 (Windows; U; Windows NT 5.1; de; rv:1.8.1) Gecko/20061010 Firefox/2.0"
```

Wird als Honeypot ein System verwendet, das einer Vielzahl an Zugriffen ausgesetzt ist, kann die Ermittlung des eigentlichen Zielsystems relativ schwierig sein. Es ist dann gut zu wissen, von welchem IP-Bereich in etwa der Zugriff stattfinden wird. Bei größeren Organisationen als Zielobjekt kann eine vorgängige whois-Abfrage den potentiellen Angriffsbereich abstecken (Kapitel 3.3.2) oder es wird eine geographische Eingrenzung der Anfragebereiche vorgenommen. Durch erweiterte Filter läßt sich die Suche optimieren.
Sollen beispielsweise absichtlich mehrere Zielobjekte zur Kontaktaufnahme mit einem Honeypot bewegt werden, lohnt sich ein Token-System. Dabei wird versucht, daß sich jeder Zugriff durch die unterschiedlichen Quellsysteme voneinander unterscheidet. Das Token fungiert dabei quasi als Pre-shared Secret, dessen wahre Funktion nur dem Angreifer bekannt ist. Wird zum Beispiel zum Aufruf einer Webseite verleitet, kann je nach Zielsystem eine andere URL mitgegeben werden. Durch das Nutzen von eigentlich nichtigen Parametern in der URL kann jeder Anfrage eine eigene ID zugeordnet werden (beispielsweise http://www.computec.ch/fake.php?herr_meier für Herrn Meier und http://www.computec.ch/fake.php?herr_mueller für Herrn Müller). Um die Auffälligkeit des persönlichen Angriffs zu entschärfen, kann auch mit IDs gearbeitet werden

(beispielsweise http://www.computec.ch/fake.php?23 für Herrn Meier). Dies erfordert jedoch ein automatisches Mapping zwischen ID und Benutzername beziehungsweise Mailempfänger. Das wird in einer PHP-Umgebung am besten mit einer SQL-Datenbank realisiert.
Ein solches Tokensystem ist nicht bei jedem Dienst ohne weiteres umsetzbar. Voraussetzung ist, daß das Token transparent mitgegeben wird und sich im Vorfeld durch den Angreifer definieren läßt. Webserver-Lösungen sind dafür prädestiniert, da sich diese Anforderungen mit einfachen URLs und GET-Variablen ($_SERVER[,QUERY_STRING']) erfüllen lassen. Bei SMTP-Lösungen und anderen komplexen Diensten ist es nicht so einfach, dort müßte mit manuell definierten Kommandos, die erst nach dem Verbindungsaufbau eingegeben werden können, gearbeitet werden. Die meisten Mailclients unterstützen ein solches Skripting nicht.
Wichtig ist, daß bei einer Auswertung dieser Art über das Internet immer die öffentlichen IP-Adressen des Endpunkts vorliegen. Sollte ein Opfer auf das World Wide Web über ein Gateway zugreifen (beispielsweise eine Perimeter-Firewall oder Proxy-Element), wird in den Logs des zugegriffenen Webservers die IP-Adresse der externen Schnittstelle auftauchen. Dafür verantwortlich ist entweder NAT (Network Address Translation) oder die Entkoppelung der beiden Netze. Mit obiger Technik ist es nicht ohne weiteres möglich, an die internen IP-Adressen zu kommen, die vom eigentlichen Mailclient verwendet werden. Eine solche Auswertung zielt also in erster Linie nur auf Perimeter-Informationen ab, die bei einer Überprüfung über das Internet unabdingbar sind.

4.1.7 HTML-Mail mit Internet-Referenz

Das Medium E-Mail wurde ursprünglich von Jonathan Postel im Jahr 1982 konzipiert, um einfache Textdaten über das Netzwerk zu schicken [Postel 1982]. Dabei konzentrierte man sich auf jene Zeichen, die typischerweise in einem Brief ihre Anwendung finden. Dieser Zeichensatz ist aus dem 7-bittigen ASCII-Standard entstanden, der nur die Alltagszeichen im englischen Sprachgebrauch vorsieht. Exotische und regionale Sonderzeichen wie Umlaute (beispielsweise ä, ö, ü) oder das Scharf-s (ß) sind nicht darin enthalten.
Aufgrund des technischen Fortschritts mußte und konnte diese Einschränkung überwunden werden. Der Internetboom Ende der neunziger Jahre verlangte, daß auch internationale Gegebenheiten umfassend berücksichtigt wurden. Durch MIME (Multipurpose Internet Mail Extensions), das in RFC 2045 im November 1996 spezifiziert wurde, kann in einer Nachricht angegeben werden, auf welche Struktur und welchen Aufbau zurückgegriffen wird [Borenstein und Freed 1996]. Damit läßt sich auch ein anderer Zeichensatz verwenden, was auch Sonderzeichen einschließt.
Damit ist der Versand von HTML- beziehungsweise RTF-gestützten E-Mails möglich. Dabei wird im Inhalt einer Nachricht nicht mehr nur auf ASCII-Text gesetzt, sondern er wird um vielerlei Möglichkeiten erweitert. So ist die Darstellung verschiedener Schriften, Farben und Tabellen möglich und eine E-Mail kann wie eine zu verschickende Webseite behandelt werden. Der Autor der HTML-Mail schreibt seinen HTML-Code, den er in Form eines E-Mails an den Empfänger schickt. Letzterer läßt durch seinen MIME- und HTML-kompatiblen Mailclient die Nachricht interpretieren und kommt damit in den Genuß der »persönlichen mobilen Webseite«.