Uptimes

Mitgliederzeitschrift der
German Unix User Group

Snowden und Ethik

Rätselhafte SSH-Keys

Soziale Kompetenzen

guug 2013-3

Inhaltsverzeichnis

Gruß vom Vorstand
Liebe Mitglieder!

von **Wolfgang Stief, Vorstandsvorsitzender**

So kann es nun nicht weitergehn!
Das, was besteht, muss bleiben.
Wenn wir uns wieder wiedersehn,
Muss irgendetwas geschehn.

Joachim Ringelnatz, Liebesbrief

Das zu Ende gehende Jahr war für die GUUG ein sehr zähes. Das Budget konnte erst im Mai von der Mitgliederversammlung genehmigt werden, der Vorstand schrumpfte im Zuge der Neuwahl von neun auf sechs Mitglieder, es gab teilweise hitzige Diskussionen über den Zustand und die Zukunft der GUUG, und eine Reihe der geplanten Vereinsaktivitäten fanden am Ende dann doch nicht statt. Letzteres liegt an einer Mischung aus fehlender Zeit und fehlendem Enthusiasmus, und ich muss mir da sicher auch selbst an die Nase fassen.

Umso mehr freut es mich, dass die Begeisterung innerhalb der Uptimes-Redaktion nicht abreißt und das kleine Team rechtzeitig zu den Weihnachtsferien eine weitere interessante Ausgabe zusammengestellt hat. Danke den Aktiven!

Ich wünsche mir für das nächste Jahr, dass diese Begeisterung um sich greift und der Verein wieder aktiver und sichtbarer wird. Der eine oder andere wird jetzt einwerfen, warum denn dann noch nichts zum Frühjahrsfachgespräch 2014 bekannt ist. Auch so eine Sache, die in diesem Jahr zäh lief: Die ursprüngliche Planung sah vor, dass wir nach Berlin gehen. Dort konnten wir leider weder an der TU noch an der FU Räume bekommen – die einen sanieren das Gebäude, an der anderen konnte unser Ansprechpartner keine Zeit in die Organisation investieren. Wir sind dann umgeschwenkt auf das Konferenzzentrum der Uni Bochum.

Während wir dort um ausreichend Räumlichkeiten an einem Termin Anfang Mai gefeilscht hatten, kam der LinuxTag e. V. und hat kurzerhand und sehr öffentlich seinen LinuxTag in Berlin von Ende Juni auf Anfang Mai vorverlegt. Nägel mit Köpfen sagt man dazu wohl. Wir wollen uns aber nun nicht gegenseitig um die Teilnehmer prügeln und Euch auch genügend Verschnaufpause zwischen der einen und der anderen Veranstaltung lassen, sodass wir das FFG 2014 wohl auf die Zeit zwischen Mitte Mai und Ende Juni legen, damit das FFG auf jeden Fall für alle noch vor den Sommerferien liegt. Ich hoffe, bald einen Termin bekannt geben zu können. Den traditionellen GUUG-Empfang am LinuxTag wird es natürlich trotz Terminverschiebung geben. Zumal die GUUG im kommenden Jahr 30 Jahre alt wird.

Bereits jetzt absehbare Aktivitäten für das Jahr 2014 sind zum einen die Beteiligung an den *Chemnitzer Linux-Tagen* 2014 am 15. und 16. März 2014. Neben Sponsoring haben wir dort auch wieder einen eigenen Stand und stellen interessierten Messebesuchern die GUUG vor. Für Mitglieder ist das auch immer eine gute Gelegenheit, sich mit Vorstandsmitgliedern persönlich zu unterhalten und auszutauschen. Zum anderen unterstützen wir das *Libre Graphics Meeting* 2014 in Leipzig vom 2. bis 5. April 2014 finanziell.

Ich wünsche Euch und Euren Familien noch eine ruhige Vorweihnachtszeit, ein frohes Weihnachtsfest und ein erfolgreiches 2014!

Über Wolfgang

Wolfgang Stief hat Linux während des E-Technik-Studiums kennen und schätzen gelernt (Kernel 0.99). Seit 1998 verdient er sein Geld überwiegend mit Solaris, Storage-Systemen und verteilten Dateisystemen. Wenn er gerade mal nicht beruflich oder für die GUUG unterwegs ist, züchtet er Gemüse, treibt sich in sozialen Netzwerken herum oder macht Musik.

Vereinsleben
Aus Verein und Vorstand Mitte bis Ende 2013

Dieser Artikel berichtet regelmäßig über die mehr oder weniger sichtbaren Vorgänge innerhalb der GUUG – ein lebendiger Verein mit diversen Mailinglisten, jährlicher Konferenz, Mitgliederversammlung und mehreren Vorstandssitzungen pro Jahr.

von **Anika Kehrer**

Reden ist Silber,
Handeln ist Gold,
Gemeinsam handeln ist Dynamit.

Frei nach dem bekannten Sprichwort unbekannter Herkunft.

neben alltäglichen Themen wie Budgetverteilung oder formalen Beschlussfassungen, ging es in den zwei Vorstandssitzungen des zweiten Halbjahres 2013 wegen der Öffnung der Vorstandsarbeit für nicht-Vorständler auch um deren operative Umsetzung. Am 29. Juni in Bochum war zum Beispiel erstmals ein Gast anwesend. In der Diskussion darüber, wie Teilnahmeanfragen gehandelt werden sollen, kam die Idee auf, aktiv auf bestimmte Leute zuzugehen, etwa die Orga-Ansprechpartner von Lokalgruppen oder die Leiter von Arbeitskreisen. Denn Voraussetzung sollte sein, dass die Teilnehmer tatsächlich Interesse an der Vereins- und Vorstandsarbeit mitbringen. Bis auf weiteres gilt das Verfahren, dass der Vorstand jeweils pro Einzelfall abstimmt, und dass die letztgenannten Personengruppen bevorzugt würden.

Der Lauf der Dinge

Auch die neuen öffentlichen Sitzungsprotokolle bedürfen eines definierten Prozesses. So einigte sich der Vorstand, dass das Protokoll jeweils im Vorstandswiki erstellt und in der darauffolgenden Vorstandssitzung genehmigt wird, um dann ins Mitglieder-Wiki umzuziehen. Ein anderer Prozess ist bereits festgelegt, nämlich der, was passiert, wenn ein Vorstandsmitglied ausscheidet. Nachdem Kerstin Mende-Stief erst auf der diesjährigen MV als stellvertretende Vorsitzende Vorstandsmitglied geworden war, ist sie nun aus der GUUG ausgetreten. Für diejenigen, die die Hintergründe interessieren, hat Kerstin auf Vorschlag der Redaktion in dieser Uptimes-Ausgabe ihren offenen Brief dazu veröffentlicht. Jedenfalls muss der Vorstand nun nach §10, Absatz 5 der Vereinssatzung zum 1. Januar 2014 ein persönliches Mitglied als neuen stellvertretenden Vorsitzenden berufen – denn laut §10, Absatz 1 sind zwei gleichberechtigte stellvertretende Vorstandsvorsitzende nötig, um die Satzung zu erfüllen.

Noch nicht viel Neues gibt es von der GUUG-Zukunftswerkstatt: Der Status-quo hängt an der Task, ein Team zu finden, das Konzept, Rahmen und Kosten erarbeitet. Als Idee erwägt der Vorstand außerdem eine vorbereitende Umfrage, die anonym erfolgen soll. Im Zentrum steht die Frage an jedes Mitglied, warum es in der GUUG ist. Damit nicht nur, aber zum Beispiel auch in Fällen wie diesen tatsächlich jedes Mitglied auch erreichbar ist, hat sich der Vorstand entschlossen, dass die Mailingliste *mitglieder@guug.de* [1] keine Unsubscribe-Möglichkeit besitzt: Diese Mailingliste sei für die Kommunikation des Vorstandes an die Mitglieder gedacht, die nicht gekündigt werden soll.

Wanted: Fachautor oder GUUG-Reporter

Wie Ihr im Gruß vom Vorstand gelesen habt, findet das Frühjahrsfachgespräch (FFG) der GUUG im Jahr 2014 etwas später statt als bisher, nämlich zu Sommeranfang. Da sich so die Sommer-Uptimes mit den FFG-Proceedings überschneiden würden, deklarieren wir die Sommer-Uptimes 2014-2 kurzerhand zu einer Frühlings-Uptimes 2014-1 um! Sie erscheint Ende April. Der Redaktionsschluss ist Sonntag, 30. März 2014. Und jetzt: Keine falsche Bescheidenheit. Lustige Workarounds gesichtet? Schlaue Skripte geschrieben? Nützliche Tools entdeckt, oder auch fiese Fehler? Mit einem Fachartikel in der Uptimes kannst Du zeigen, womit Du Dich beschäftigst. Oder ein Thema setzen, das Dich interessiert und zu dem Du schon recherchiert hast. Der erste Schritt ist ein Blick in die Autorenrichtlinien hinten im Heft.

Auch, welche Diskussionen die Mitglieder und Artverwandte auf den SAGE- oder Member-Mailinglisten führen, darüber besitzt die Chefredaktion keine Informationen, wenn kein Autor das Thema in der Rubrik *Vereinsleben* aufgreift. Zumal es sehr sinnvoll wäre, wenn beispielsweise in dieser regelmäßigen Mauerschau *Aus Verein und Vorstand* mehr stehen könnte, als sich an Ergebnissen aus Protokollen der MV und den Vorstandssitzungen zusammentragen lässt. Bedenkt: Ich habe keinen Zugriff auf vereinsinterne Mailinglisten, und das ist auch gut so.

Dafür sind stattdessen Reporter aus dem Kreis der GUUG-Mitglieder notwendig. Dies muss kein regelmäßiger Job sein, der einem am Bein hängt. Sondern ab und zu wären zum Beispiel Zusammenfassungen von Diskussionen viel wert – egal, ob als Hinweis oder als eigener Beitrag in der Uptimes. Auch einzeln handhabbare und inhaltlich interessante Maßnahmen wären auch kleine, selbstinitiierte Umfragen. Wenn beispielsweise Thema X auf den Listen präsent ist, lassen sich mit relativ wenig Aufwand eine kleine Zahl von Fragen an ein paar Beteiligte verschicken. Der Hinweis, dass hier mit Blick auf Uptimes gefragt wird, wäre dann natürlich fair.

Die Redaktion macht gleich mal den Anfang. Allerdings schickt sie keine Fragen herum, sondern die befinden sich bereits in dieser Ausgabe: Ihr findet in dem Artikel von Andreas Lohrum einige Fragen, inwiefern der Fall Edward Snowden für die Praxis von Sysadmins Konsequenzen hat. Der Autor hat sich diese Fragen selbst gestellt. Das ist eine super Idee. Und es wäre interessant zu erfahren, wie Ihr Eurerseits diese Fragen beantwortet. Und welche Fragen Ihr Euch vielleicht noch zusätzlich stellt. Wer sich selbst oder ein paar Bekannte nach dem Beispiel Lohrum interviewt, schickt die Antworten bitte an <redaktion@uptimes.de>. In der nächsten Ausgabe berichten wir dann.

Links

[1] Zu den verschiedenen Mailinglisten der GUUG siehe „Aus MV und Vorstand“, Uptimes 2013-2, S. 6: http://www.guug.de/uptimes/2013-2/index.html

Über Anika

Anika Kehrer ist freie IT-Journalistin und seit der Wiederauflage der Uptimes im Sommer 2012 als Chefredakteurin an Bord. In dieser Rolle hält sie die Fäden in der Redaktion zusammen und akquiriert und betreut die Artikel der redaktionellen Sommer- und Winterausgaben. Leitidee ist, die Uptimes als Fachmagazin und als Vereinszeitschrift der GUUG anspruchsvoll und transparent zu gestalten, in Zusammenarbeit mit einem Redaktionsteam aus GUUG-Mitgliedern.

An die Mitglieder der GUUG
Offener Brief zu meinem Vereinsaustritt

Auf Vorschlag der Chefredaktion veröffentlicht die Autorin diesen offenen Brief, der zuerst auf der Mailingliste *guug-members* erschienen war, in der Uptimes-Rubrik *Vereinsleben*. Er zeigt, was Einzelne von aktiver Vereinsarbeit abhält – was aus Vereinssicht kein wünschenswerter Zustand ist.

von **Kerstin Mende-Stief**

If you choose not to decide,
you still have made a choice.

©RUSH, Free Will

An die Mitglieder der GUUG,

am 20. September habe ich die Kündigung meiner Vereinsmitgliedschaft zum Jahresende an meine Vorstandskollegen übermittelt. Kurz darauf habe ich das in einem offenen Brief auch den Mitgliedern, die auf *guug-members* mitlesen, mitgeteilt. Die Veröffentlichung meines offenen Briefes in der Uptimes soll Spekulationen und Gerüchten vorbeugen. Ich möchte, dass jeder von mir selbst erfährt, was mich zu meinem Schritt bewogen hat.

Mit Erlöschen meiner Mitgliedschaft lege ich alle Ämter nieder, die ich derzeit als Vorstandsmitglied und stellvertretender Vorstandsvorsitzender innehabe. Ich bin niemand, der sich normalerweise mit einem Status quo zufrieden gibt, und so war ich angetreten, dem Verein in der Öffentlichkeit mehr Gewicht zu verleihen. Aufgrund der aktuellen und bereits länger andauernden Situation des Vereins bin ich nicht mehr überzeugt, dass der Verein seinem Zweck gerecht wird, und ich kann auch nicht erkennen, dass sich mittelfristig die Situation ändern wird.

Es hat etwas mit Erwartungshaltung zu tun. Meine war eine andere. Ich habe mich zum Beispiel immer gefragt, warum von über 600 Mitgliedern nur rund 10 Prozent an den Mitgliederversammlungen teilnehmen. Wenn die Mitgliederversammlung nicht am Rande des FFG stattfinden würde, kämen wahrscheinlich nicht mal 1 Prozent.

Wie ich in der Zwischenzeit erkannt habe, genügt es den Mitgliedern offenbar, durch Vereinszugehörigkeit ihre Affinität zu unixoiden Betriebssystemen zu statuieren und zum vergünstigten Satz am FFG teilzunehmen. Das ist mir persönlich zu wenig.

Allerdings verstehe ich jetzt auch, warum trotz kontinuierlicher Aufrufe des Vorstandes zur Mitarbeit von den Vereinsmitgliedern kaum aktive Beiträge oder Vorschläge zur zukünftigen Gestaltung des Vereins kommen: Vom Vorstand eingebrachte Vorschläge werden fast durchgehend negativ bewertet und abgelehnt. Dass der Vorstand für seine Bemühungen von den Mitgliedern dann auch noch regelmäßig abgewatscht wird, dafür bringe ich absolut kein Verständnis auf.

Und damit kommen wir zum Auslöser meines Aus- und Rücktritts. Schon seit langem fiel mir auf, dass der Umgangston im Verein vorwurfsvoll und von Misstrauen geprägt ist; ja, teilweise sogar beleidigende Züge trägt. Das bin ich nicht mehr bereit zu akzeptieren. Ich habe sehr selten in einer auch nur annähernd destruktiven Umgebung gearbeitet. Und ich sehe partout nicht ein, in ein solches Umfeld meine Lebensqualität und Freizeit zu investieren.

Ich wünsche Euch weiterhin alles Gute. Vor allem jedoch wünsche ich dem Verein, dass die Mitglieder, die jetzt immer nur meckern und nach Fehlern bewusst zu suchen scheinen, entweder selbst einmal etwas Konstruktives beitragen, oder einfach ihre Klappe halten.

Eure Kerstin.

Über Kerstin

Kerstin hat 2012 das Hamsterrad hinter sich gelassen. So abwechslungsreich wie ihre Vergangenheit (sie war Elektriker, Event Manager, Marktforscher, Verkäufer, Barkeeper, Journalist, Webmaster, Promoter, hat in Biergärten bedient u.v.m.) ist ihre Gegenwart als Visionär, Konnektor, Berater, Unternehmer und Intrapreneur. Die freiberufliche Unternehmensberaterin sitzt im Aufsichtsrat einer Aktiengesellschaft (an der sie auch beteiligt ist) und engagiert sich in der *Deutschen Wolke*, einer Workinggroup der Open Source Business Alliance (OSBA), sowie in der Arbeitsgruppe *Cloud Computing* im TeleTrusT.

Exkurs: Anmerkung der Redaktion

Die Uptimes ist ein öffentliches Medium, und es ist eine Policy-Frage, ob auch nicht so schöne Vereinsinterna wie diese darin ihren Platz finden sollen. Folgende Überlegungen haben die Chefredaktion veranlasst, den offenen Brief aktiv in die Uptimes zu holen:

- Die Uptimes ist nicht als Jubelautomat gedacht, auch nicht im Vereinsteil – denn dies würde die Tatsache verzerren, dass Vereinsarbeit zu gleichen Teilen von Freude, Zielorientierung und Sozialkompetenz lebt, was nicht so einfach unter einen Hut passt.
- Der angesprochene Sachverhalt betrifft nicht nur die GUUG, vielmehr kennt mutmaßlich jeder Verein und jedes freie Projekt die angesprochene Problematik. Damit betrifft dieser offene Brief die GUUG nur exemplarisch und steht für ein weiteres thematisches Umfeld, als nur GUUG-Interna.

Ich freue mich über Leser-Perspektiven an <redaktion@uptimes.de>:

- zu dem Vorgehen offener Briefe in der Uptimes;
- zum Sachverhalt dieses Artikels.

Übrigens: Dass sich ausgerechnet in dieser Ausgabe ein Fachartikel über soziale Kompetenzen befindet, ist Zufall. Ehrenwort. Betroffener Artikel von Stefan Schumacher liegt der Redaktion schon länger vor und ist diesmal endlich zum Zuge gekommen (wobei zu hoffen ist, in Zukunft noch mehr Artikel von Stefan zu sehen, was ich ihm auch schon gesagt habe :-)). Das stand aber fest, lange bevor der Vereinsaustritt von Kerstin Mende-Stief der Redaktion überhaupt bekannt war.

Winterkrimi 2013
Die rätselhaften SSH-Hostkeys

Auf Spurensuche am Honeypot-Tatort verfolgen die Ermittler Voss & Bunten verschiedene Theorien. Sie besitzen plausible Lösungen, ganz schließen können sie die Akte jedoch nicht. Hinweise aus der Bevölkerung, Pardon, der Leserschaft sind erbeten.

von **Andreas Bunten und Torsten Voss**

Tod im Eise, Tod im Eise
Wirst erfrieren, wirst erfrieren
Wirst träumerisch leicht krepieren
Leichter Schnee fiel, bedeckte
zudeckte den Hostkey

Frei nach Kyra Sellnow, Wintermärchen II

Es trug sich zu im Hamburg und Frankfurt des Jahres 2012, dass Andreas Bunten und Torsten Voss mehr über SSH-Angreifer herausfinden wollten. Sie bauten verschiedene Arten von verwundbaren Linux-Systemen, damit böswillige Angreifer sich an diesen vergreifen konnten und beobachteten das Treiben. Die Angreifer versuchten Passwörter zu erraten, und so wurden den Systemen Root-Passwörter gegeben, die leicht zu erraten waren. Ein Bericht darüber enthält die Uptimes vom Sommer 2012 ([1], Exkurs 1: Hintergrund).

Dem aktuellen Fall liegt der Vorgang zu Grunde, dass jedes der angreifenden Systeme mit einem Portscan bedacht wurde. Es wurden keine wahllosen Scans durchgeführt, sondern lediglich die Systeme überprüft, von denen unerwünschte Verbindungsversuche ausgingen. Die Prüfvorgänge ermittelten auch den SSH-Hostkey des angreifenden Systems, falls dort ein SSH-Server Verbindungen annahm.

Das Hostkey-Rätsel

Bei der Auswertung der so ermittelten Hostkeys fiel schnell auf, dass da etwas nicht stimmen kann. Insgesamt fanden sich 19.609 verschiedene Hostkeys, und jeder tauchte im Schnitt 2,5 mal auf. Aber es gab genau 3 Hostkeys, die wir mehrere tausend Male sahen! SSH-Hostkeys müssen laut RFC nicht einzigartig sein (siehe Exkurs 2: SSH-Hostkeys), dennoch ist das die allgemeine Praxis. Wieso also sehen wir diese drei Keys so oft (Tabelle 1)?

Ermittlertheorien

Die ersten Vermutungen zur Ursache dieses Befundes sahen so aus:

- Unsere erste Theorie drehte sich um einen **vagabundierenden IT-Berater**, der von Kunde zu Kunde tingelt. Sein verseuchter Laptop greift aus dem jeweiligen Netzwerk immer wieder unsere Lockvogel-Systeme an.
- Die zweite Theorie erkennt dies als Angriff des amerikanischen Geheimdienstes! Denn die **NSA** steckt überall, und die Angriffe stammen tatsächlich immer von den gleichen Systemen.
- Im dritten Anlauf tippten wir auf **Netzwerk-Appliances**, die mit statischem Hostkey ihre Besitzer erreichen. Da dies unsichere Praxis ist, verwundert es nicht, wenn einige solcher Appliances kompromittiert und nun für Angriffe missbraucht werden.

Um der Sache auf den Grund zu gehen, blieb uns nichts anderes übrig, als weiter nachzuforschen.

Name	Länge	Fingerprint	Verfahren
Key-1	1024	dc:cd:da:72:fe:6e:db:70:ff:11:e5:cc:b4:27:80:80	RSA
Key-2	1024	20:e4:a9:50:e3:40:f4:54:cc:d4:47:02:bc:99:7b:f3	DSA
Key-3	1040	1b:7e:77:e2:9e:2d:9d:4c:38:43:83:e6:37:2d:4b:ed	RSA

Tabelle 1: Diese drei SSH-Hostkeys kamen tausendfach auf den angreifenden Systemen vor.

Indiz 1: IP-Adressen und Länder

Die einzelnen Hostkeys tauchten häufig auf, außerdem von vielen unterschiedlichen IP-Adressen. Bei den beiden Hostkeys Key-2 und Key-3 etwa kam praktisch jeder Verbindungsversuch von einer neuen IP-Adresse. Bei Key-1 war dies lediglich jedes zweite mal der Fall (Tabelle /reftab:vorkommen).

Befund	Key-1	Key-2	Key-3
Anzahl Angriffe	1111	1566	888
Anzahl unterschiedlicher Quell-IPs	493	1558	884

Tabelle 2: Auftreten der einzelnen Hostkeys im Vergleich

Die IP-Adressen ordneten wir anhand von Whois-Informationen Ländern zu. Die Auflösung ist nicht vollständig korrekt, aber liefert einen ersten Anhaltspunkt (Abbildung 1).

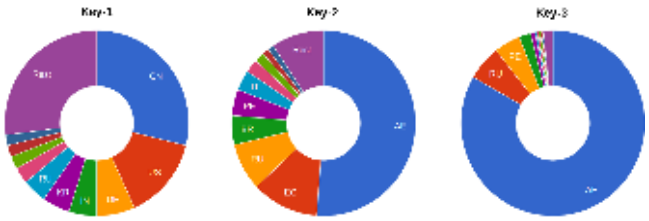


Abbildung 1: Verteilung der Hostkeys nach Ländern: AE - Vereinigte Arabische Emirate; BR - Brasilien; CN - China; DE - Deutschland; EG - Ägypten; IN - Indien; IT - Italien; KR - Südkorea; PE - Peru; RU - Russland; US - USA

Zunächst waren es viel mehr Länder, als wir angenommen hatten. Dabei schienen Key-2 und Key-3 vor allem in den Vereinigten Arabischen Emiraten vorzukommen – Key-1 war gleichmäßiger verteilt. Auf dem zweiten Blick gleicht die Verteilung von Key-1 auch der allgemeinen Verteilung der Länder, aus denen wir Angriffe von kompromittierten Systemen sehen.

Das ist das Ende der Theorie des **vagabundierenden IT-Beraters**: So viel um die Welt muss hoffentlich niemand reisen. Wir haben diese Theorie bald aufgegeben.

Indiz 2: Das angreifende System

Ein genauerer Blick auf die angreifenden Systeme gibt vielleicht einen Fingerzeig zur Lösung. Zunächst sahen wir uns also den SSH-Dienst des angreifenden Systems an. Wir entdeckten ihn nicht nur auf dem Standard-Port 22/TCP, sondern auch auf anderen Netzwerkports. Bei unseren drei rätselhaften Hostkeys ergab sich ein gemischtes Bild.

Keys 2 und 3 wurden nur auf Port 22/TCP festgestellt, und es handelte sich immer um das *OpenSSH 3.6p1 protocol 2.0*. Key-1 lag auf vielen unterschiedlichen Ports, die normalerweise nicht so verwendet werden, und fast nie auf Port 22 (Abbildung /refports). Die Client-ID lautete aber fast immer *SCS sshd 2.0.13 protocol 1.5*. Selten kam auch *SCS sshd 1.2.25 protocol 1.5* vor.

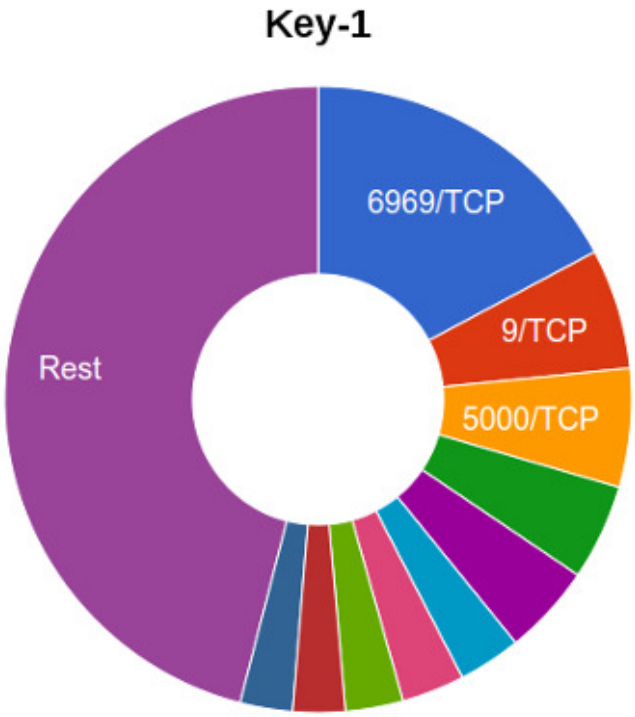


Abbildung 2: Verteilung der Ports, auf denen Key-1 anzutreffen war.

In der Regel bieten SSH-Server unterschiedliche Verschlüsselungsverfahren an, und sie halten einen Hostkey für jedes Verfahren vor. Wurden also im vorliegenden Fall die Hostkeys jeweils in der gleichen Kombination mit anderen Hostkeys angeboten, oder gab es Variationen? Antwort: Lediglich Key-2 trat mit vielen verschiedenen RSA-Schlüsseln auf. Die anderen traten allein oder maximal in zwei Kombinationen auf. Da fiel uns ein, dass zwischen 2006 und 2008 aufgesetzte Debian-Systeme eine fehlerhafte OpenSSL-Bibliothek besaßen. Damit erzeugte Hostkeys waren leider vorhersagbar.

Eine entsprechende Überprüfung förderte zu Tage, dass es sich bei 56 der 19.609 festgestellten Hostkeys tatsächlich um solche *Debian Weak Keys* handelte. Unsere drei speziellen Schlüssel waren aber nicht dabei.

Wir hatten immer vollständige Portscans gegen die angreifenden Systeme durchgeführt. Können nun vielleicht die anderen Dienste einen Hinweis

darauf geben, um was für Systeme es sich handelt? Und treten immer die gleichen Kombinationen von Diensten auf?

Bei den Hostkeys 2 und 3 traten einige Kombinationen auf, die sich aber ähnelten. Konkret wurden vor allem FTP, SSH, HTTPS und Port 1050/TCP angeboten. Die Systeme mit Key-1 hingegen boten extrem unterschiedliche Dienste an.

Da sich die IP-Adressen stark änderten und auch die Distanz im Netzwerk variierte – gemessen anhand der TTL in den Antwort-Paketen –, vermuteten wir, dass es sich wirklich um unterschiedliche Systeme handelt, obwohl die angebotenen Dienste und deren Versionen so ähnlich aussahen.

Schließlich versuchten wir, Hostnamen zu den IP-Adressen zu ermitteln. Insgesamt existierte zu den meisten IP-Adressen kein Reverse Lookup, so dass sich der Hostname nicht bestimmen ließ. Die Namen, die sich auflösen ließen, deuten bei Key-2 und Key-3 vor allem auf Einwahlprovider hin. Am meisten tauchte der Dienst *Al Schamil* des Unternehmens *Etisalat* auf, dem Staatsprovider der Vereinigten Arabischen Emirate, auf dessen Homepage nachzulesen ist: „Al Shamil is a broadband High Speed Internet service targeted at home users." Die auflösbaren Hostnamen für Key-1 ließen zum Teil zwar ebenfalls auf Einwahlprovider schließen. Aber diese waren weiter über die Welt verstreut. Weiterhin tauchten auch eine Reihe von Namen auf, die auf Server-Systeme hindeuteten.

Soviel zur Theorie **NSA**: Es ist mittlerweile klar, dass es sich um viele teils sehr verschiedene Systeme handeln muss. Die NSA kann getrost ausgeschlossen werden.

Indiz 3: Die Verbindung

Da Key-1 fast nie auf Port 22 auftaucht, könnte dies ein zusätzlicher SSH-Server sein, den der Angreifer nach Kompromittierung des Systems nachinstalliert. Eine SSH-Backdoor also. Der immer gleiche Hostkey könnte darauf zurückzuführen sein, dass ein fertig konfiguriertes Paket inklusive Hostkey installiert wird.

Verband sich ein Angreifer zu einem unserer Honeypots, so wurde nicht nur seine IP-Adresse protokolliert, sondern auch alle relevanten SSH-Verbindungs-Parameter. Die Parameter verraten neben dem Client-Namen beispielsweise die unterstützten Verschlüsselungs-Algorithmen. Die Verbindungen von Key-1 Systemen verwendeten verschiedene Parameter-Kombinationen. Anscheinend wurden unterschiedliche Arten von Angriffs-Software verwendet.

Auf der Suche nach den Verbindungsparametern für die Key-2 und Key-3 mussten wir feststellen, dass keine vorliegen! Dies führt unmittelbar zur Anzahl der Anmeldeversuche seitens der Aggressor-Systeme. Für Systeme mit Key-1 lagen 444.896 Anmeldeversuche vor. Für die anderen beiden Hostkeys kein einziger! Das heißt, dass diese Systeme lediglich Connect-Scans durchgeführt haben.

Ein Blick auf die zeitliche Verteilung der Verbindungen lässt erneut stutzen. Angriffe von Systemen mit Key-dc treten kontinuierlich auf. Die anderen beiden Hostkeys hingegen trafen wir nur im September und Oktober 2012 sowie im Mai 2013 an. Eine Auswertung der Tageszeit, zu der die Angriffe stattfanden, ergab wiederum keine markanten Aktivitätsperioden.

Die Polizei bittet um Mithilfe

Die Hostkeys sind augenscheinlich auf unterschiedliche Art jeweils etwas Besonderes. Dies sind unsere aktuellen Vermutungen:

Exkurs 1: Hintergrund

Torsten und Andreas untersuchen seit zwei Jahren SSH-Angriffe mit Honeypots [1]. Angreifer verfahren nach ihren Beobachtungen in der Regel in zwei Stufen: Zunächst erraten sie das Passwort durch Ausprobieren. Im Schnitt benötigt das 18 Anmeldeversuche. Das häufigste Ziel ist der Root-Account, und es werden Passwörter wie `123456`, `password` oder `1q2w3e` durchprobiert. Die meisten Angriffe versuchen das Passwort, das dem Kontonamen entspricht (also das Konto *webmaster* mit dem Passwort `webmaster`). Dieser Teil des Angriffs erfolgt fast immer von bereits kompromittierten Systemen und selten von den eigenen Rechnern der Angreifer. Stunden bis wenige Tage später folgt die zweite Stufe, bei der sich die Angreifer manuell am System anmelden. Sie machen den Opfer-Rechner zum Teil eines Botnets oder installieren Angriffs-Software lokal, um nach weiteren verwundbaren Systemen zu suchen.

Torsten und Andreas stellten bei der Analyse der Daten unter anderem fest, dass SSH-Hostkeys sehr gut geeignet sind, um angreifende Systeme wiederzuerkennen [3]. Auf der Homepage des Projekts [4] ist ein Beispiel-Video der Shell-Session eines Angreifers zu sehen, der gerade Zugriff auf einen High-Interaction-Honeypot erlangt hat [5].

- Key-1 ist wahrscheinlich Teil eines fertig konfigurierten SSH-Servers, den Angreifer in der Regel auf einem hohen Port als Backdoor auf kompromittierten Systemen installieren. Wir vermuten, dass es sich um ein Standard-Paket handelt, dass von verschiedenen Gruppen genutzt wird, da wir unterschiedliche Angriffs-Software feststellten.
- Wir denken, dass die Theorie **Netzwerk-Appliance** bei Key-2 fast zutrifft: Es handelt sich wahrscheinlich um DSL- oder Kabelmodems. Diese werden vor allem in den Vereinigten Arabischen Emiraten verwendet. Die immer ähnliche Kombination der angebotenen Dienste lässt uns darauf schließen, dass es sich nicht um die Rechner der Benutzer sondern das Modem davor handelt. Ein Angreifer könnte die Modems auf irgendeine Weise kompromittiert haben und mit diesen entweder nach SSH-Servern gesucht oder das IPv4-Internet einem Scan unterzogen haben, wie es von Unbekannten im sogenannten *Internet Census 2012* geschehen ist [1].
- Key-3 scheint fast das gleiche wie Key-2 zu sein. Wir tippen auch hier auf DSL- oder Kabelmodems in den Vereinigten Arabischen Emiraten. Allerdings ist die Länderverteilung etwas anders. Wenn es sich um den gleichen Angreifer wie bei Key-2 handeln sollte, stellt sich die Frage, warum überhaupt unterschiedliche Hostkeys verwendet wurden.

Doch bleiben speziell bei Key-2 und Key-3 Fragen offen. Wurde der Hostkey vom Angreifer oder vom Hersteller installiert? Warum tritt der eine Hostkey in vielen Kombinationen auf? Gibt es vielleicht eine deutlich bessere Erklärung für die vielfach auftretenden Hostkeys?

Wir würden uns freuen, wenn jemand unter dem Weihnachtsbaum an weiteren möglichen Lösungen knobelt. Theorien bitte an <krimi@bunten.de>. Ein frohes Fest und einen guten Rutsch!

Exkurs 2: SSH-Hostkeys

Bei einem SSH-Hostkey handelt es sich um einen kryptographischen Schlüssel des SSH-Servers. Der öffentliche Teil des Schlüssels wird beim Verbindungsaufbau an den Client übertragen, damit dieser die Identität des Servers prüfen kann. Der geheime Teil des Hostkeys verbleibt auf dem Server. Hostkeys werden meistens durch einen 16 Byte langen sogenannten Fingerprint in hexadezimaler Form dargestellt. RFC 4251 besagt, dass Hostkeys nicht eindeutig sein müssen:

> 4.1 Host Keys - Each server host SHOULD have a host key. Hosts MAY have multiple host keys using multiple different algorithms. Multiple hosts MAY share the same host key.

In der Praxis hat sich durchgesetzt, dass jedes System einen eigenen Hostkey besitzt. Netzwerk-Appliances generieren diesen meistens beim ersten Start dynamisch. Hat ein Angreifer Zugriff auf den Hostkey eines Servers, so ist er in der Lage, dessen Identität vorzutäuschen und Man-in-the-Middle-Angriffe gegen die Benutzer des Servers durchzuführen. Daher sollten Hostkeys nach Kompromittierung eines Servers unbedingt neu erzeugt werden.

Literatur und Links

[1] Heise-News am 19. März 2013: Botnetz scannt das Internet mit Hilfe von gehackten Endgeräten:
http://heise.de/-1825634

[2] Andreas Bunten und Torsten Voss: Stille Beobachter – SSH-Angreifern mit Honeypots nachstellen. In: Uptimes, Mitgliederzeitschrift der German Unix User Group, 2012-2, S. 13ff:
http://www.guug.de/uptimes/2012-2/index.html

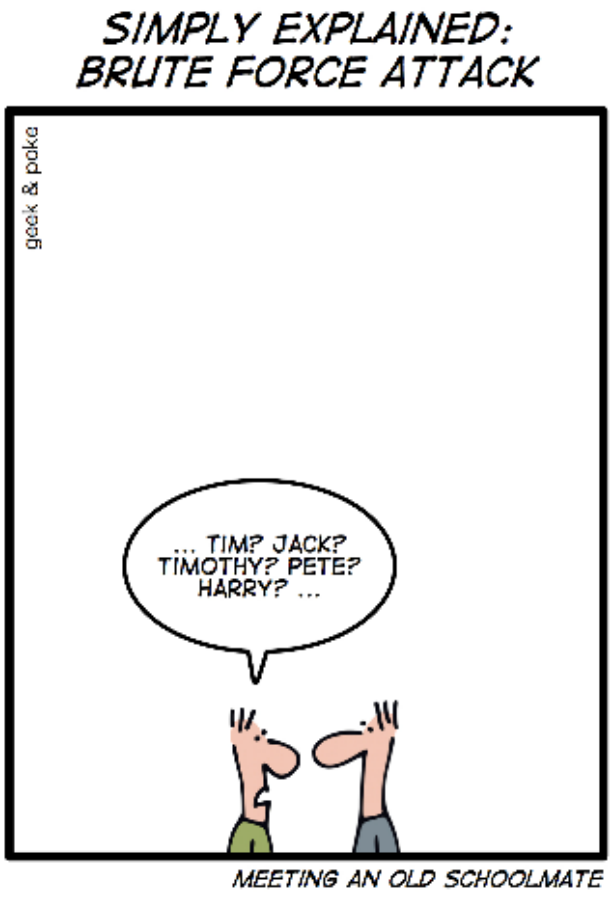
[3] Andreas Bunten und Torsten Voss: SSH-Honeypots und neue Schutzmaßnahmen gegen Brute-Force-Angriffe. In: Konferenzband des 20. DFN-Workshops, Februar 2013

[4] Homepage von Torstens und Andreas' SSH-Honeypot-Projekt:
http://bunten.de/ssh.html

[5] Video der Shell-Session eines Angreifers auf einem High-Interaction-Honeypot:
http://vimeo.com/39218661

Über Andreas und Torsten

Andreas Bunten (Bild) ist seit 1996 im Bereich Unix-Administration und Security tätig. Er war acht Jahre Mitglied des Emergency-Response-Teams des Deutschen Forschungsnetzes im *DFN-CERT* und hat dabei eine Vielzahl kompromittierter Systemen untersucht. Seit drei Jahren berät und unterstützt er die Kunden der *Controlware GmbH* im Bereich IT-Sicherheit.
Torsten Voss (ohne Bild) studierte technische Informatik und arbeitet seit 2006 beim *DFN-CERT* im Emergency-Response-Team des Deutschen Forschungsnetzes. Sein Schwerpunkt ist die Vorfallsbearbeitung und die Aufklärung von Kompromittierungen.



Kommentar: Snowden und Ethik
Betreff Snowden – Auswirkungen auf Sysadmin-Praxis

Anlässlich des Verhaltens seines Berufskollegen Edward Snowden stellt sich der Autor einige Fragen.

von **Andreas Lohrum**

In der Sittenlehre
ist nicht Unterricht,
sondern Übung
die Hauptsache.

Johann Bernhard Basedow

Überwachung durch Dienste und Industrie – dieses Thema bewegt den interessierten Sysadmin schon eine ganze Weile. So war es kein Wunder, dass wir es beim diesjährigen Guru-Grill in München im Licht der aktuellen Ereignisse diskutierten: Snowden und die Veröffentlichungen im Guardian [1]. Meinungsstark fiel ich Anika auf, sodass sie mich für einen Artikel gewinnen wollte. Wer kann einer freundlichen Bitte widerstehen? Ich gebe hier also ein paar Antworten wieder.

Warum sollte es uns als europäische Systemadministratoren interessieren, wenn im fernen Amerika ein externer Administrator seinen Auftraggeber verlässt und Dienstgeheimnisse nicht nur mitnimmt, sondern auch der Weltöffentlichkeit zur Verfügung stellt?

Im Fall Manning, der nur Zugriffsrechte als normaler Anwender hatte, wurde klar, daß die internen Security-Richtlinien der Amerikaner nicht ausreichend sind. Zu viele Informationen sind vorhanden, der Zugriff darauf nur schwach reglementiert, und die Zugriffe wurden nicht protokolliert, zumindest nicht ausreichend.

Die Veröffentlichung durch einen Systemadministrator mit deutlich weitreichenderen Zugangsrechten hat die amerikanischen Dienste erschüttert. Als Reaktion auf diese Verletzung der Vertraulichkeit will beispielsweise die NSA den Zugang zu Informationen stärker regulieren: Nach einem Medienbericht [2] will sie 90 Prozent ihrer Systemadministratoren durch automatisierte Abläufe ersetzen. Im Usenet-Forum führte das zu dem Thread [3] „I will replace you with a very small shell script".

In der Angst, weitere Risiken durch gegebenenfalls unzuverlässige Menschen einzugehen, werden weitere Verantwortliche den Versuch machen, deutlich mehr Automatisierung bei reduzierten Zugangsrechten einzelner durchzusetzen. Die Arbeit in so einer Umgebung wird kleinteiliger und weniger erfüllend sein. Das Vieraugenprinzip wird vorherrschen, Mistrauen die erste Regung sein.

Jeder Arbeitgeber wurde nochmals deutlich mit der Nase auf das Problem der Vertraulichkeit gestoßen. Daher sollten sich alle Sysadmins mit diesem Thema beschäftigen und sich eine reflektierte Meinung bilden. Damit sind sie im nächsten Einstellungsgespräch gewappnet.

Ist Snowdens Handeln mit den Ethik-Guidelines der SAGE vereinbar?

Die Ethik-Richtlinien, die viel zuwenig bekannt sind, sind weit vor den aktuellen Ereignissen erstellt worden, sozusagen in einer besseren oder auch naiveren Welt (siehe Kasten *Ethik-Guidelines der SAGE*. Trotzdem ist die soziale und ethische Verantwortung bereits prominent herausgestellt. Vor allem folgender Absatz liest sich für mich wie die Aufforderung, eine öffentliche Diskussion zu erzwingen:

> I will do my best to make decisions consistent with the safety, privacy, and well-being of my community and the public, and to disclose promptly factors that might pose unexamined risks or dangers.

Wäre die Eskalation an Vorgesetzte oder an den zuständigen Kongressausschuss nicht die weniger schlimme, aber trotzdem wirksame Maßnahme gewesen?

Es gab außer dem Fall Manning noch weitere Fälle, in denen sich Whistleblower mit ihrem Anliegen an die vorgesehenen Stellen gewendet hatten. Das resultierte in Haftstrafen, in der Sache

hat sich wenig bis nichts bewegt. Die Bilanz von Aufwand/Risiko/Konsequenzen und erwünschter Wirkung dieses Handelns ist also negativ.

Ist Snowden, wie es die deutsche Presse derzeit kolportiert, ein Held und anderen Systemadministratoren ein Rollenvorbild?

Ich habe großen Respekt davor, dass Edward Snowden einen gut dotierten Job, seine Beziehungen zu Familie, Freunden und Kollegen, die Heimat in den USA, seine persönliche Freiheit und möglicherweise sein Leben riskiert, um auf die Missstände aufmerksam zu machen. Er hat sich entschieden, die Sache vor seine persönliche Annehmlichkeit zu stellen. In dieser Hinsicht verdient er das Etikett „Held“. Momentan sieht es so aus, dass die Veröffentlichungen auch in Teilen der Politik auf Resonanz stoßen, sodass es Anpassungen geben wird.

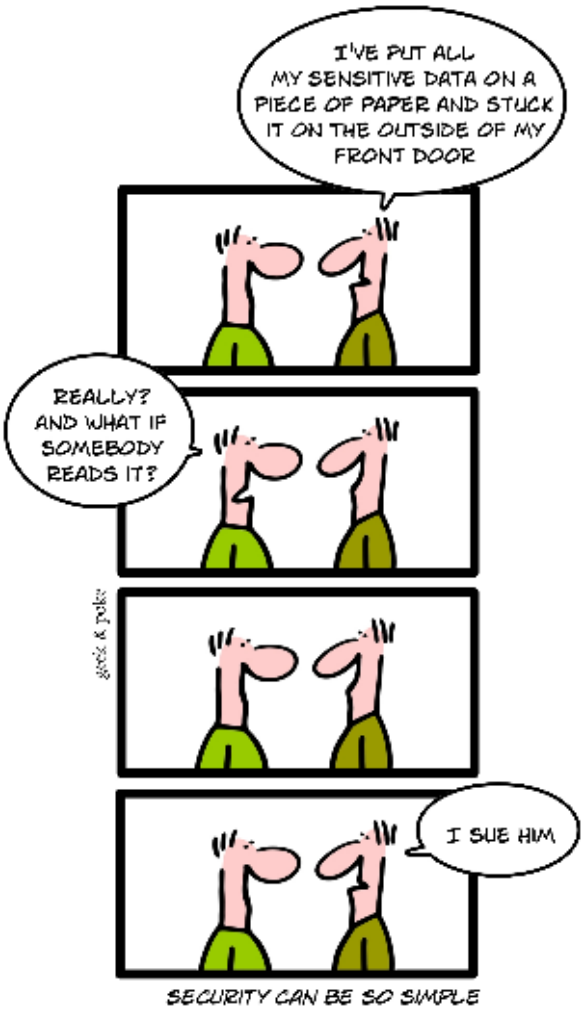
Ich bin aber auch der Meinung, dass man diesen Einsatz nicht von jedem Systemadministrator einfordern kann, in dessen Bereich berichtenswerte Gefahren für die eigene Firma oder gar die gesamte Gesellschaft auftreten. Damit wir nicht persönlich in diese Bredouille kommen, müssen wir im Vorfeld dafür einstehen, dass effektive interne und externe Möglichkeiten bestehen, um über Missstände zu berichten – ohne dass dies gleich lebensverändernde Konsequenzen nach sich zieht.

Was habe ich gelernt?

Ich kenne den alten Spruch:

> Für jemanden, der mit Security befasst ist, lautet die Frage nicht: Bin ich paranoid? Die Frage lautet: Bin ich paranoid genug?

Irgendwie habe ich nicht genug Phantasie, mir in aller Vollständigkeit vorzustellen, was mittlerweile alles zur Realität geworden ist. Ich möchte jedoch nicht, dass meine Tochter in einer *1984*-Umgebung aufwächst. Ob das noch abzuwenden ist? Ich werde dafür arbeiten.



Exkurs: Ethik-Guidelines der SAGE

Der *System Administrators' Code of Ethics* der SAGE wurden im Jahre 2003 nach Diskussionen zwischen Rob Kolstad [4], Evi Nemeth [5] und Alan Paller [6] im Vorfeld der SAGE-Gründung erarbeitet. Dabei wurden Leitgedanken zur professionellen Arbeit, dem Umgang mit Vertraulichkeit und der sozialen Verantwortlichkeit aufgestellt. Im Jahre 2006 wurden die Richtlinien auch von der Usenix, LISA und der LOPSA unterzeichnet. Die Ethik-Guidelines lauten [7]:

„We as professional System Administrators do hereby commit ourselves to the highest standards of ethical and professional conduct, and agree to be guided by this code of ethics, and encourage every System Administrator to do the same.

Professionalism

- I will maintain professional conduct in the workplace and will not allow personal feelings or beliefs to cause me to treat people unfairly or unprofessionally.

Personal integrity

- I will be honest in my professional dealings and forthcoming about my competence and the impact of my mistakes. I will seek assistance from others when required.
- I will avoid conflicts of interest and biases whenever possible. When my advice is sought, if I have a conflict of interest or bias, I will declare it if appropriate, and recuse myself if necessary.

Privacy

- I will access private information on computer systems only when it is necessary in the course of my technical duties. I will maintain and protect the confidentiality of any information to which I may have access, regardless of the method by which I came into knowledge of it.

Laws and policies

- I will educate myself and others on relevant laws, regulations, and policies regarding the performance of my duties.

Communication

- I will communicate with management, users, and colleagues about computer matters of mutual interest. I will strive to listen to and understand the needs of all parties.

System integrity

- I will strive to ensure the necessary integrity, reliability, and availability of the systems for which I am responsible.
- I will design and maintain each system in a manner to support the purpose of the system to the organization.

Education

- I will continue to update and enhance my technical knowledge and other work-related skills. I will share my knowledge and experience with others.

Responsibility to computing community

- I will cooperate with the larger computing community to maintain the integrity of network and computing resources.

Social responsibility

- As an informed professional, I will encourage the writing and adoption of relevant policies and laws consistent with these ethical principles.

Ethical responsibility

- I will strive to build and maintain a safe, healthy, and productive workplace.
- I will do my best to make decisions consistent with the safety, privacy, and well-being of my community and the public, and to disclose promptly factors that might pose unexamined risks or dangers.
- I will accept and offer honest criticism of technical work as appropriate and will credit properly the contributions of others.
- I will lead by example, maintaining a high ethical standard and degree of professionalism in the performance of all my duties. I will support colleagues and co-workers in following this code of ethics.“

Links

[1] *The Guardian* zu Edward Snowden: http://www.theguardian.com/world/edward-snowden
Lesenswert – Security-Blog von Bruce Schneier, fachmännische Kommentator beim Guardian: https://www.schneier.com/
[2] Bericht über Admin-Einsparungen bei der NSA: http://orf.at/stories/2194108/2194109/
[3] Usenet-Diskussion dazu:
https://groups.google.com/forum/m/#!topic/de.alt.sysadmin.recovery/Wf-DY0DqQs0
[4] Rob Kolstad: http://en.wikipedia.org/wiki/Berkeley_Software_Design
[5] Evi Nemeth: http://en.wikipedia.org/wiki/Evi_Nemeth
[6] Alan Paller: http://www.sans.org/press/photos_bios.php
[7] System Administrators' Code of Ethics: https://www.usenix.org/lisa/system-administrators-code-ethics

Über Andreas

Andreas Lohrum ist Inhaber der *consulting4projects GmbH* aus München und hauptsächlich als Projektleiter im RZ-Umfeld beschäftigt. Er hat seine ersten Unix-Kenntnisse 1988 mit Microport Unix und später SCO Unix erworben, 1994 erfolgte der Umstieg auf NetBSD. Nach Stationen als Unix-Systemadministrator, Call-Bearbeiter an einer Unix-Hotline (SunOS und Solaris), Unix-Consultant und Trainer für Systemadministration (Storage, HA Cluster, Hardware) hat Andreas seinen beruflichen Schwerpunkt ins Projektmanagement verlagert. Seit 1995 ist er Mitglied in der GUUG.

Soziale Kompetenzen für Informatiker

Fehler > Kritik | Kritik > Feedback | Feedback > Ziel

Trotz breiten Aufgabenspektrums gilt die Informatik als technozentrisch: Informatiker werden traditionell als Ingenieur oder Naturwissenschaftler ausgebildet. Soziale Kompetenzen kommen in der Ausbildung kaum vor. Dieser Artikel stellt Grundlagen der sozialen Kompetenzen vor und zeigt, wie sie trainiert werden.

von **Stefan Schumacher**

"Finden Sie,
das Ganze hier ist es wert,
seine Freizeit massiv
dafür einzuschränken?"
"Wir melden uns."

Tweet von @Goganzeli am 5. Oktober 2013

Informatiker, Software-Entwickler und Systemadministratoren arbeiten mit Kunden sowie projektbezogen in Gruppen. Sie benötigen daher Kenntnisse des Projektmanagements – aber nicht nur auf der technischen Ebene. Ebenso müssen Sie in der Lage sein, sich und Projektgruppen zu organisieren. Außerdem arbeiten Informatiker in der Regel nicht nur mit anderen Informatikern oder Ingenieuren, sondern interdisziplinär mit anderen Berufs- oder Forschergruppen zusammen. Daher ist es notwendig, das sich Informatiker in kurzer Zeit in bisher fremde Fachgebiete einarbeiten. Außerdem müssen Sie nebenbei die Sprache des Fachgebiets erlernen, um mit ihren Kollegen kommunizieren zu können.

Auf solche sozialen Kompetenzen legen immer mehr Stellenanzeigen und Personalmanager Wert, ohne dass auch nur der Hauch eine einheitliche Definition oder ein allgemeingültiges Verständnis existieren. In einem kontinuierlichen Studienprojekt untersucht zum Beispiel das Bundesinstitut für Berufsbildung (BIBB), welche beruflichen Qualifikationen Zukunft haben. Das Projekt Nr. 2.0.501 [1] *Früherkennung von Qualifikationsentwicklungen* untersucht unter anderem die Datenbank KURS der Bundesagentur für Arbeit, die über Kurse zur beruflichen Aus- und Weiterbildung auflistet – mit über 600.000 Angeboten laut Projekt-Unterseite "die größte Weiterbildungsdatenbank, die es in Deutschland gibt". Die Abteilung gibt zur Auskunft, dass gleich nach EDV-Kursen Angebote zu so genannten *Soft Skills* folgen: Schulungen in Gesprächs- und Verhandlungsführung, Moderations- und Präsentationstechniken seien besonders gefragt [2].

Der Begriff *Kompetenz*

Was ist nun unter Kompetenzen zu verstehen? Staudt, Kailer und Kottmann ([3], S.440) beschreiben Kompetenz als

> Schlüssel zur Innovation: Sie ist Voraussetzung, neue Sach- und Dienstleistungen, Materialien und Verfahren zu entwickeln und in wirtschaftliche Erfolge umzusetzen. Die Kompetenzen der Führungs- und Fachkräfte bestimmen die Innovationsfähigkeit von Unternehmen in zwei Richtungen: Als limitierender Faktor begrenzen sie die Unternehmensentwicklung, wenn Kompetenzdefizite die Diffusion neuer Techniken oder Erschließung neuer Märkte behindern. Als initiierender Faktor erschließen sie neue Möglichkeiten auch außerhalb der traditionellen Unternehmensentwicklung.

Weiter führen sie aus, das es schwierig sei, die diffusen Kategorien eindeutig voneinander abzugrenzen und zu operationalisieren. Konsens bestehe lediglich darin, „dass Kompetenz inhaltlich nur handlungsorientiert zu bestimmen ist. Kompetent ist jemand für konkrete Handlungen in einer konkreten Handlungssituation“.

Nach Gessler ([4], S. 26) definiert der Deutsche Bildungsrat Kompetenz wie folgt:

> Kompetenz befähigt einen Menschen zu selbstverantwortlichem Handeln und bezeichnet den tatsächlich erreichten Lernerfolg. Qualifikation ermöglicht die Verwertung von Kenntnissen, Fertigkeiten und Fähigkeiten.

North/Reinhardt ([5], S.29) definieren Kompetenz als die

> Fähigkeit, situationsadäquat zu handeln. Kompetenz beschreibt die Relation zwischen den an eine Person oder Gruppe herangetragenen oder selbst gestalteten Anforderungen und ihren Fähigkeiten bzw. Potenzialen, diesen Anforderungen gerecht zu werden. [...] Kompetenzen konkretisieren sich immer erst im Moment der praktischen Wissensanwendung in einem konkreten Handlungsbezug und werden am erzielten Ergebnis der Handlungen messbar.

Nach Kanning ([6], S. 14) versetzen Kompetenzen einen Menschen in die Lage, eine bestimmte Aufgabe lösen zu können. Dazu komme es jedoch erst dann, wenn die Kompetenzen in Verhalten umgesetzt werden. Er betont den Unterschied zwischen Kompetenz und Verhalten (Abbildung 1):

> Allein aus dem Fehlen eines entsprechenden Verhaltens kann nicht zweifelsfrei auf entsprechende Defizite in den Kompetenzen geschlossen werden. Umgekehrt bieten Kompetenzen keine Gewähr für kompetentes Verhalten. Kompetenzen erhöhen lediglich die Wahrscheinlichkeit für das Auftreten kompetenten Verhaltens. Ob und inwieweit Kompetenzen tatsächlich in Verhalten umgesetzt werden, hängt von vielen Faktoren ab, die in der Umwelt und der handelnden Person selbst liegen können.

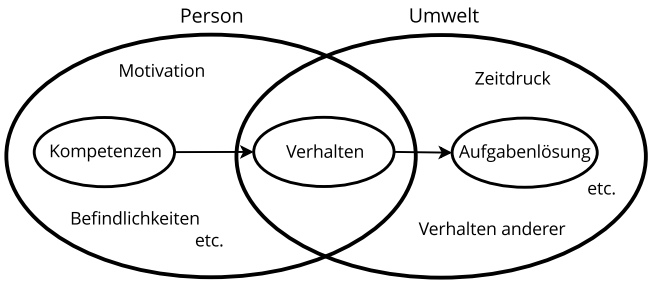


Abbildung 1: Zusammenhang zwischen Kompetenz und Verhalten nach Uwe Peter Kanning, Soziale Kompetenzen in der Personalentwicklung [6], S. 15.)

Ebenso uneinheitlich wie der Kompetenzbegriff werden soziale Kompetenzen verstanden. So schreibt Kanning weiter, dass sozial kompetentes Verhalten situationsabhängig sei. Ein Polizist etwa könne sich gegenüber einem Randalierer anders verhalten als gegenüber einem fragenden Bürger. Darüber hinaus sei eine konkrete, branchenübergreifende, verbindliche Definition sinnlos. Jedes Unternehmen sei darauf angewiesen, spezifische Anforderungen an das Verhalten ihrer Mitarbeiter zu stellen und diese bei der Personalauswahl und -entwicklung zu berücksichtigen.

Doch es bleibt dabei: Gerade Informatiker arbeiten in der Regel mit IT-fremden Fachabteilungen zusammen und müssen daher vielfältige soziale Fähigkeiten mitbringen (siehe zum Beispiel [7], [8]). VigenschowSchneider bezeichnen die Kommunikationsfähigkeit gar als „zentralen Erfolgsfaktor“ ([9], S.V) und nennen ihr Buch *Soft Skills für Softwareentwickler* scherzhaft auch: *Warum Entwickler nicht zuhören und Fachbereiche nicht entwickeln können*.

Einige neuere Trends in der Entwicklung, wie Agiles Programmieren oder Extreme Programming, legen besonderen Wert auf Teamarbeit. Einige Firmen setzen beim Extreme Programming auch internationale Entwicklergruppen ein, die zeitversetzt arbeiten. So gibt es je ein Entwicklerteam in Deutschland, Japan und den USA. Durch die versetzten Zeitzonen übergeben die Japaner ihr Projekt vor dem Feierabend an die deutsche Gruppe. Diese arbeitet dann weiter am Projekt und übergibt es an die Amerikaner. Diese übergeben es dann wieder an die Japaner und so weiter. Jede dieser Übergaben erfordert ausgeprägte kommunikative und interkulturelle Fähigkeiten.

Entstehung sozial kompetenten Verhaltens

Sollen nun diese so genannten sozialen Kompetenzen (weiter-)entwickelt werden, ist es notwendig zu verstehen, wie kompetentes oder auch inkompetentes Verhalten überhaupt entsteht. Basierend auf den Prinzipien der Handlungstheorie stellt Kanning ([6], S. 20) dafür das *Modell der elaborierten Steuerung des Sozialverhalten* vor. Es beschreibt einen rational gesteuerten Prozess, der zu sozial kompetentem Verhalten führt. Das Modell eignet sich besonders für ungewohnte oder wichtige Situationen, da es eine bewusste Verhaltenssteuerung voraussetzt. In alltäglichen Situationen greift das Modell hingegen nicht, da solche gewohnten Situationen vereinfacht gesteuert werden. Dies hängt mit der Einschätzung des Kontextes und seiner Bedeutung zusammen, denn un-

wichtige Situationen werden sozusagen auf Autopilot durchflogen (vgl. [10]).

Die elaborierte Steuerung des Sozialverhaltens nach Kanning erfolgt in vier Schritten:

- *Analyse der Situation:* Das Individuum legt seine eigenen Ziele fest, die kurz- oder langfristig, neben- oder nacheinander existieren. Daneben sind aber auch die Anforderungen der sozialen Umwelt zu beachteten. Dies führt zu einer komplexen Situation, denn oft sind die eigenen Ziele nicht deckungsgleich mit den Erwartungen der Umwelt. Zudem sind nicht einmal die Erwartungen anderer Beteiligter deckungsgleich. Daher, so Kanning, ist die Abwägung und Priorisierung der Ziele und Erwartungen notwendig.
- *Analyse der Verhaltensoptionen:* Aufgrund der so festgelegten und priorisierten eigenen Ziele und Anforderungen der Umwelt müssen die möglichen Verhaltensoptionen durchdacht werden.
- *Umsetzung des ausgewählten Verhaltens:* In dieser Phase werden die vorher durchdachten Optionen in Handlungen umgesetzt. Hier kommt es darauf an, Kompetenzen in Handlung umzuwandeln. Die Ausführung der Handlung kann an mangelnder Kompetenz scheitern.
- *Analyse der Konsequenzen:* Nach der Handlung gilt es, den Soll-Zustand (Erreichung der Ziele) mit dem Ist-Zustand zu vergleichen. Diese Phase kann je nach Zielart (kurzfristig vs. langfristig) entsprechend dauern. Sind bestimmte Ziele nicht erreicht worden, werden diese mit Schritt 1 beginnend neu untersucht und nachgesteuert.

Nicht alle Verhaltensweisen werden elaboriert gesteuert. Einfache (oder gar stumpfsinnige) Prozesse, wie das Bezahlen an einer Supermarktkasse, laufen unbewusst und automatisch ab. Der Autor dieser Zeilen hat zum Beispiel während seiner Schulzeit eineinhalb Jahre an einer Supermarktkasse gejobbt und spricht daher aus Erfahrung.

Im *Modell der automatisierten Steuerung des Sozialverhaltens* gibt es daher nur zwei Phasen. In der vereinfachten Situationsanalyse werde ein sozialer Hinweisreiz wahrgenommen (wie: ein Kunde legt eine Ware aufs Band) und blitzartig in die Verhaltensphase (Waren scannen und abkassieren) gewechselt. Die meisten sozialen Interaktionen laufen nach diesem vereinfachten Steuerungsmodell ab.

Nun ist die Frage, wie ein Individuum die eigenen Kompetenzen einschätzt. Oftmals ist es so, das inadäquates Verhalten den Handelnden nicht unbedingt auffällt, jemand also nicht merkt, das er oder sie die Erwartungen der sozialen Umwelt nicht erfüllt. Migge ([11], S. 105) beschreibt die Phasen der Kompetenzwahrnehmung wie folgt:

- *Unbewusste Inkompetenz:* Die Möglichkeiten, Fähigkeiten, Lernfelder und Einsichten sind unbekannt. Im Hinblick auf Unbekanntes oder Ausgeblendetes kann man sich nicht bewusst inkompetent fühlen.
- *Bewusste Inkompetenz:* Ein Lernfeld, Problem oder Kontext wird erkannt. Erforderliche und vorhandene Kompetenz werden verglichen und eine erhebliche Differenz wird als bewusste Inkompetenz erlebt. Beispiel: Fehlende Fahrkenntnisse beim Besuch der Fahrschule.
- *Bewusste Kompetenz:* Die Differenz zwischen erforderlichen und vorhandenen Fähigkeiten ist überwunden. Das Lernfeld oder der Zusammenhang sind bekannt und werden beherrscht, wenn auch noch mit bewusster Konzentration. Beispiel: Fahrkenntnisse nach gerade abgelegter Führerscheinprüfung.
- *Subjektives Gefühl der Kompetenz* kann zu inadäquatem Verhalten führen. Wer von sich selbst fälschlich annimmt, etwas zu können, besteht gelegentlich die Prüfung in der Praxis nicht. Dies wird gern verleugnet: „Ich bin eine kompetente Führungspersönlichkeit. Anderen passieren Fehler, mir aber nie!“

Verfahren zur Entwicklung sozialer Kompetenzen

Kanning unterscheidet vier Verfahren, um soziale Kompetenzen zu erwerben. *Wissensbasierte Verfahren* „fokussieren diejenigen Wissensbestandteile, die für die erfolgreiche Auswahl von Verhaltensoptionen und ihre Umsetzung in tatsächliches Handeln notwendig sind“ ([6], S.37f.). Er beschreibt das sozial relevante Wissen als wichtigen Faktor für sozial kompetentes Handeln, das in der Regel wenig bewusst abgearbeitet wird. So sei es für jeden Opernbesucher klar, keine Chips zu essen, während dies im Kino akzeptables Verhalten wäre. Oftmals fällt daher das explizite (oder auch implizite) Wissen um soziale Begebenheiten erst auf, wenn man auf unbekannte Subkulturen trifft oder im Ausland tätig wird.

Die Personalentwicklung konzentriert sich daher im Allgemeinen auf *sozial geteiltes Wissen*, also gesellschaftlich verbreitete Verhaltensstandards,

wie Begrüßungen oder Höflichkeitsrituale. Ebenso kann es sich dabei aber auch um Verhaltensrichtlinien innerhalb einer Organisation handeln, beispielsweise Park- oder Grüßordnungen. Neben dem sozial geteilten Wissen kann in Workshops oder Seminaren auch *individuell entwickeltes Wissen* diskutiert werden. Dabei handelt es sich um Verhaltensstrategien, die einzelne Mitarbeiter im Umgang mit bestimmten Problemen (beispielsweise schwierigen Kunden) erprobt haben.

Wissensbasierte Verfahren vermitteln also vorrangig das Wissen um bestimmte Normen, Regeln und Werte in spezifischen Organisationen oder Kulturen. Dieses Wissen lässt sich relativ leicht vermitteln, man kann beispielsweise Broschüren verteilen, die die Goldenen Regeln für den Umgang mit Partnern aus dem Ausland erklären. Oder Videos auf einer Webseite zeigen den korrekten Umgang eines Verkäufers mit Kunden. Vorteilhaft für den Mitarbeiter ist hier die freie Zeiteinteilung, so kann er beispielsweise nach Büroschluss die Broschüren lesen oder in weniger stressigen Zeiten die Videos anschauen. Problematisch ist allerdings, dass es kein Training der erwünschten Verhaltensweisen in natürlichen Situationen gibt. Lediglich das Wissen um bestimmte Verhaltensregeln bedeutet noch nicht entsprechendes Verhalten. Schließlich lernt auch jeder Fahrschüler, dass Alkohol am Steuer tabu ist. Trotzdem verunfallen jedes Jahr tausende betrunkene Autofahrer.

Verhaltensorientierte Verfahren (vgl. ebd. S. 101f.) trainieren dagegen Sozialverhalten praktisch. Hier gilt Wissensvermittlung nur als Mittel zum Zweck, denn das Ziel ist die direkte Verhaltensänderung. Dazu spielen Trainings spezifische Situationen durch. Damit die Mitarbeiter diese Beispiele auf allgemeine Situationen generalisieren können, ist es wichtig, eine Reflexionsphase einzubauen. Schließlich sollen die Mitarbeiter nicht Theater spielen lernen, sondern die Voraussetzungen für kompetentes Verhalten in (unbekannten) Alltagssituationen erwerben. Prototypisch verläuft ein solches Training nach einem bestimmten Schema ab:

Erläutern des Problemgebiets, um Problembewusstsein herzustellen. Anschließend Erarbeitung von Lernzielen zusammen mit den Teilnehmern. In der eigentlichen Trainingsphase helfen Videosequenzen oder Rollenspiele zum Erkennen richtigen und falschen Verhaltens. In der anschließenden Reflexionsphase erarbeitet die Gruppe die Unterschiede zwischen der guten und schlechten Lösung und generalisiert diese. Um das eigene Verhalten der Teilnehmer zu verändern, ist es notwendig, das diese selbst aktiv in Rollenspielen teilnehmen: Denn durch Diskussion und Feedback kann das eigene Verhalten analysiert und verbessert werden. bei dieser Feedbackphase hilft Videotechnik, Schlüsselszenen zu analysieren und dem Protagonisten seine eigene Wirkung vorzuführen. Die Rollenspiel-und-Feedback-Phase ist gegebenenfalls mehrfach zu iterieren, um ihre Wirkung schrittweise zu entfalten.

Beratungsorientierte Verfahren (vgl. ebd. S. 263f.) befassen sich mittels Einzelfallanalyse mit dem Individuum. Dieser Ansatz setzt voraus, dass sich Trainer und Trainee über einen längeren Zeitraum mehrfach treffen und sich mit der individuellen Lebenswirklichkeit des Mitarbeiters auseinandersetzen. Es handelt sich also um eine Beratung, die man landläufig als *Coaching* bezeichnet. Migge ([11], S. 22]) definiert Coaching als

> gleichberechtigte, partnerschaftliche Zusammenarbeit eines Prozessberaters mit einem Klienten. Der Klient beauftragt den Berater, ihm behilflich zu sein. [...] Die Klienten sollen durch die gemeinsame Arbeit an Klarheit, Handlungs- und Bewältigungskompetenz gewinnen. Coaching ist keine Wissenschaft, sondern eine handlungsorientierte hilfreiche Interaktion.

Selbsterfahrungsorientierte Verfahren (vgl. [6], S.299f.) gelten zwar als spektakulär, setzen aber selten ein konkretes Lernziel voraus. Ihre Wurzeln liegen zum großen Teil in der humanistischen Psychologie, nach der in jedem Individuum eine eigene Natur auf Entfaltung dränge. Heutzutage spielen derartige Angebote nur mehr eine untergeordnete Rolle, insbesondere da viele Manager kaum zwei bis drei Tage im Wald verbringen wollen, ohne konkrete Lernziele vor Augen zu haben. (Allerdings weiß der Autor aus seiner Erfahrung während des Wehrdienstes, dass eine ordentliche Durchschlageübung eine Jäger-Gruppe zusammenschweißt. Oder vollends zerbrechen lässt.)

Zusammenfassend lässt sich sagen, das die ersten drei Verfahren immer Hand in Hand gehen. Es gibt keine rein wissensbasierten Verfahren oder rein verhaltensorientierte. Für die Personalentwicklung ist es daher notwendig, verschiedene Maßnahmen durchzuführen oder diese zu kombinieren, um so die Erfolgsquote zu erhöhen.

Maßnahmen, um Kommunikationsfähigkeit zu fördern

Egal, wie technisch die Ausbildungsberufe laut BIBB-Beschreibung auch ausgerichtet sein mögen, ein Punkt kommt immer vor: Die Beratung und Schulung von Nutzern. Das heißt, das ein Informatiker in der Lage sein muss, mit anderen Personen zu kommunizieren – in der Regel mit Nicht-Informatikern. Er muss den IT-Fachjargon in die Sprache des Benutzers übersetzen sowie Probleme auf deren Lösungswege herunterbrechen und verständlich machen. Oftmals sind die Benutzer IT-fern ausgebildet, verfügen also über kein oder nur geringes Verständnis für technische Belange. Hier ist der Informatiker gefordert, sich dem Benutzer anzunähern und ihm gegebenenfalls Grundlagen zu erläutern. Außerdem ist es hilfreich, Jargons zu definieren oder abzustimmen: Ein Lemma ist in der Informatik etwas anderes als in der Linguistik, ein Psychologe versteht unter Reliabilität etwas anderes als ein Datenbankadministrator.

Aber auch die Sprache des Benutzers oder Kunden muss der Informatiker in kurzer Zeit erlernen und verstehen. Nicht jeder weiß von Haus aus, was die Preiselastiziät der Nachfrage kennzeichnet oder der reziproke Kurzschlussstromübertragunsfaktor ist. Hierbei ist es hilfreich, in Projektgruppen so genannte Bindestrich-Informatiker einzusetzen, also beispielsweise Ingenieur-Informatiker, die im Nebenfach Maschinenbau oder Elektrotechnik studiert haben, oder Wirtschafts-Informatiker, die sich mit BWL und VWL auskennen.

Arbeitet ein Informatiker als Anwendungsentwickler, muss er im *Software Engineering* das so genannte Requirements Engineering durchführen. Denn Software Engineering bezeichnet in der Informatik die Meta-Ebene der Software-Entwicklung, also Projektmanagement und Ähnliches. Im Schritt des Requirement Engineerings führt er die Anforderungsanalyse durch. In der Theorie erhält der Informatiker vom Kunden ein Pflichtenheft, in dem alle zu lösenden Aufgaben detailliert aufgeführt sind. In der Realität trifft der Informatiker auf Kunden, die nicht wissen was sie wollen, was sie können oder welche Probleme man überhaupt mit der Informatik lösen kann. Daraus folgen dann vollständig überzogene oder schlichtweg falsche Erwartungen des Kunden an die Informatik. Technische Maßnahmen können eben keine sozialen Probleme lösen, auch wenn dies gebetsmühlenartig wiederholt wird. In solchen Fällen ist es notwendig, *Analyse- und Fragetechniken* zu beherrschen, mit denen der Informatiker den eigentlichen Problemen auf den Grund gehen kann.

Vigenschow/Schneider ([9], S.51) stellen zum Beispiel die 6-Stufen-Fragetechnik für die Software-Analyse vor. Sie basiert auf der natürlichsprachlichen Analyse des *Neuro-Linguistischen Programmierens* und besteht aus den folgenden sechs Schritten:

- *Prozesswörter überprüfen:* Verben und subsubstantivierte Verben identifizieren Prozesse. Überprüfen Sie sie mit den W-Fragen: Wer? Was? Wann? Wie? Wo?
- *Bezugssysteme bestimmen:* Worauf bezieht sich Vergleiche oder Steigerungen? Beispiel: „Das neue System muss schneller werden!“ – „Schneller als was? Wie erkennen Sie den Geschwindigkeitsgewinn?“ - „An der gesteigerten Produktivität meiner Arbeitsgruppe.“ – „Wie macht sich die gesteigerte Produktivität bemerkbar, vielleicht durch mehr Aufträge pro Mannstunde?“ – „Ja.“ - „Wieviele Aufträge bearbeiten Sie denn momentan pro Mannstunde, und wieviel möchten Sie gern schaffen?“
- *Quantoren überprüfen:* Allquantoren wie *alle*, *keiner*, *nie*, *immer* gehen oft mit Ausnahmen einher. Es gilt dann, diese Ausnahmen in Erfahrung zu bringen.
- *Bedingungen überprüfen:* Sind alle Verzweigungen und Ausnahmen bekannt? Sind die jeweiligen Determinanten festgelegt (wie Einstiegs- und Abbruchsbedingungen)?
- *Variablen und Konstanten klarifizieren:* Identifizieren Sie alle Variablen und Konstanten und geben Sie ihnen sprechende Namen. `$Mindest_Lagerbestand_Motoren == 1000;` sagt mehr aus als `$mlbm==1000;`.
- *Abkürzungen und Fachbegriffe glossieren:* Pflegen Sie ein Glossar, in dem alle Abkürzungen und Fachbegriffe erklärt werden. Achten Sie darauf, das unterschiedliche Kontexte unterschiedliche Bedeutungen implizieren: Eine Signatur ist in der Kryptografie etwas anderes als in der Kunstgeschichte. Weisen Sie auf solche Bedeutungsunterschiede hin. Stellen Sie das Glossar allen Beteiligten zur Verfügung, etwa durch Wissensmanagement-Software wie Wikis oder Blogs. Das stellt sicher, das alle Beteiligten über dasselbe reden können.

Maßnahmen, um Konfliktmanagement zu fördern

Konflikte treten auf, sobald zwei Menschen aufeinandertreffen (vgl. zum Beispiel [12], [13], [14]). Daher ist es unausweichlich, als Informatiker, Entwickler oder Systemadministrator den Umgang mit Konflikten zu lernen. Gerade in der Gruppenarbeit von Entwicklern, dem Prozess des Requirement Engineerings mit Kunden oder als Administrator, der Benutzer einweisen muss, lauern vielfältige Konflikte: Entwickler kommunizieren oftmals nicht persönlich, sondern greifen auf E-Mails und Logfiles zurück. So fällt der Teil einer Nachricht weg, die sonst via Körpersprache und Stimme übertragen würde. Korrelierend mit diesem Entropieverlust steigt die Wahrscheinlichkeit von Konflikten.

Latente Konflikte werden laut Werpers [15] häufig ignoriert, bis sie eine Toleranzschwelle erreichen und sich manifestieren. Dabei steigen jedoch Dauer und Intensität des Konfliktes, was wiederum durch direkte Kosten für die Lösung eines Konfliktes die Gesamtkosten in die Höhe treibt, beispielsweise Honorare für Anwälte oder Mediatoren, Krankheitskosten oder Prozesskosten. Weiterhin entstehen vielfältige Opportunitätskosten, weil zum Beispiel Reibungsverluste in der Kommunikation und Zusammenarbeit auftreten, die sich auf aktuelle und zukünftige Projekte auswirken. Konflikte binden Ressourcen und verursachen Schäden, die vermieden oder zumindest vermindert werden können. Andererseits, so Werpers, erzeugen Konflikte neue Ideen und damit auch neue Produkte. Dazu sei allerdings notwendig, zielgerichtetes Konfliktmanagement zu betreiben, damit der Konflikt nicht eskaliert. Ein *Konflikttraining* zahlt sich also in mehrfacher Hinsicht aus, es reduziert die entstehenden Kosten, verbessert das Betriebsklima und erhöht das Leistungspotenzial.

Lutz [16] weist nach, das Kommunikation mit die Hauptaufgabe einer Führungskraft ist: Der Konflikt als Sonderfall der Kommunikation erfordere besondere Kompetenzen. Oftmals gelte jedoch die Devise, das sich eine Führungskraft durchsetzen müsse. Machtworte gelten also als adäquate Lösungen für Konflikte. Dies führt aber gerade bei sehr guten Entwicklern und Admins zu Verstimmungen, den klassischen Nerds. Der Konflikt wird verschleppt und in späteren Konflikten aus Rache erneut aufs Tapet gebracht, um offene Rechnungen zu begleichen. Derartige Vorgehensweisen sprengen oftmals eine Gruppe und führen zum Kollaps. Ein Beispiel ist der Ausschluss von Theo de Raadt aus dem NetBSD-Projekt [17] und dessen anschließender Fork zu OpenBSD.

Für den Begriff *Konflikt* existieren viele Definitionen. Pruitt/Rubin [18] definieren ihn folgendermaßen:

> For us, conflict means perceived divergence of interest, or a belief that the parties' current aspirations cannot be achieved simultaneousley.

Zu betonen ist die „Wahrnehmung unvereinbarer Interessen". Dies ist wichtig, da es nach Watzlawik [10] keine einzig wahre Realität gibt, sondern jedes Individuum seine eigene, inhärente, Biografie-abhängige Realität konstruiert (worauf sich insbesondere der *Radikale Konstruktivismus* stütze). Dies ist einer der Gründe, warum es zu Konflikten kommt, die unterschiedlich wahrgenommen werden.

Verhalten in Konflikten lässt sich in verschiedenen Modellen beschreiben. Rahim (vgl. [19]) unterscheidet fünf Konfliktverhaltensstrategien, wobei er die Interessen beider Parteien in zwei Dimensionen beachtet:

- *Dominieren* bedeutet, das die betreffende Partei ihre Interessen durchsetzt, ohne auf die Interessen der anderen Partei zu achten.
- *Entgegenkommen* heißt, das eine Partei ihre Interessen hintenanstellt und den Interessen der anderen Partei Vorrang gewährt.
- *Vermeiden* ist ein Verhalten, in dem beide Parteien darauf verzichten, ihre Interessen durchzusetzen.
- *Kompromisse* sind ein Weg, bei dem beide Seiten Zugeständnisse an die andere Seite machen. Beide Parteien gewinnen und verlieren also gleichermaßen.
- *Integration/Konsens* ist die einzige wirkliche Konflikt*bewältigungs*strategie. Hierbei werden die Interessen aller Parteien im größtmöglichen Ausmaß zu wahren versucht. Dazu ist es allerdings notwendig, den Konflikt nicht eskalieren zu lassen sowie die zu Grunde liegenden Ursachen beziehungsweise Interessen für den Konflikt zu identifizieren.

Exkurs: Konfliktmanagement anhand des Orangen-Beispiels

Alice und Bob streiten sich um eine Orange. Alice könnte Bob dominieren, um ihm die Orange beispielsweise unter Gewaltandrohung abzunehmen. Sie könnte Bob auch entgegenkommen und ihm die Orange überlassen. In ersten Falle geht Bob, im zweiten Alice leer aus. Möchten Alice und Bob dem Konflikt aus dem Weg gehen, verteilen sie die Orange überhaupt nicht. Beide können dann ihr Bedürfnis nicht befriedigen. Die beliebteste Konfliktvermeidungsstrategie ist der Kompromiss. Alice und Bob teilen die Orange und befriedigen ihr Interesse jeweils teilweise.

Alle bisherigen Strategien ignorieren geflissentlich die Gründe für das Interesse an der Orange. Keiner der beiden weiß, warum der andere überhaupt die Orange haben will. Dies ist jedoch der Ansatz für die Integration und den Konsens. Zuerst wird geklärt, warum Alice und Bob die Orange haben möchten. Nun stellt sich heraus, das Alice den Orangensaft für ihren Whisky wünscht. Bob hingegen möchte einen Kuchen bereiten und wünscht sich die Orangenschalen für den Teig.

Betrachtet man nun Alice' und Bobs Interessen, erscheinen sie auf einmal recht einfach vereinbar: Erst presst Alice den Orangensaft in ihr Whisky-Glas, anschließend kann Bob die Schale raspeln und in seinen Teig rühren. Beide Bedürfnisse werden also voll erfüllt. Beim Kompromiss hingegen – jeder eine halbe Orange – hätte Alice den halben Saft vergeben und eine halbe Schale bekommen, die sie jedoch nicht braucht. Das Inverse gilt für Bob, der unnötigerweise den halben Saft bekommen, doch die halbe Schale vermisst hätte.

Das Orangenbeispiel illustriert die fünf Strategien (siehe *Exkurs: Konfliktmanagement anhand des Orangen-Beispiels*). Es gilt zu beachten, dass jede Konfliktstrategie ihre Berechtigung hat und je nach Situation unterschiedlich sinnvoll sein kann. Niemand möchte nach einem Autounfall schwerverletzt auf Rettung hoffen und erleben, wie ein Rettungsassistent und ein Notarzt über Behandlungsmethoden Kompromisse aushandeln. Zentrales Element eines Konflikttrainings sollte dennoch sein, die eigene Einstellung gegenüber Konflikten zu reflektieren und den Konsens als neuen Königsweg der Konfliktlösung zu etablieren. Es ist daher notwendig, das Weltbild *Der Kompromiss ist die beste Konfliktlösung* zu erschüttern. Darüber hinaus sollten Grundlagen der menschlichen Kommunikation behandelt werden (die beispielsweise [20], [21], [22] und [23] beschreiben).

Maßnahmen, um Feedback zu fördern

Feedback ist ein Mechanismus in informationsverarbeitenden Systemen, bei dem ein Teil der Ausgangsgröße direkt oder in modifizierter Form auf den Eingang des Systems zurückgeführt wird. Ein einfaches Beispiel dafür ist ein Heizungsthermostat. Am Thermostat wird eine Soll-Temperatur eingestellt. Mit einem Messfühler überwacht das Thermostat die Ist-Temperatur. Das Ergebnis des Vergleichs – es ist zu warm oder zu kalt – wird an die Heizung zurückgemeldet, die daraufhin stärker oder schwächer heizt.

Während des zweiten Weltkrieges befassten sich einige Wissenschaftler und Ingenieure mit der Steuerungs- und Regelungstechnik auf dem Gebiet des Maschinenbaus und der Elektrotechnik. Vorrangiges Ziel war, neue Waffensysteme zu entwickeln, die immer komplexer wurden, wie beispielsweise Marschflugkörper. 1948 prägte der Mathematiker Norbert Wiener für diese Wissenschaft den Begriff Kybernetik, vom altgriechischen Wort für Steuermann. Ende der 1940-er, Anfang der 1950-er Jahre befassten sich dieselben Wissenschaftler unter anderem damit, kybernetische Mechanismen wie Rückkopplung und Zirkulär-Kausalität in sozialen Systemen zu untersuchen. Mit Hilfe der Kybernetik und der Systemtheorie lassen sich beliebige Lernvorgänge allgemein beschreiben. Dazu betrachtet man das lernende System und seine Umwelt sowie deren strukturelle Kopplung:

- Das System muss seine Umwelt wahrnehmen (Abbildung 2). Beim Menschen erfolgt dies über seine fünf Sinne, beispielsweise indem er ein Buch liest oder eine Fernsehsendung sieht und hört.

- Die wahrgenommene Umwelt muss verarbeitet werden, der Mensch muss beispielsweise in der Lage sein, die Schrift im Buch zu dekodieren und zu verarbeiten, also Sinnzusammenhänge herzustellen. Die neu aufgenommene Repräsentation der Umwelt – die Wörter im Buch – wird verarbeitet und in das eigene Sinnsystem aufgenommen (Abbildung 3).

- Idealerweise führt das veränderte Sinnsystem (Abbildung 4) dazu, das eigene Verhalten zu ändern.

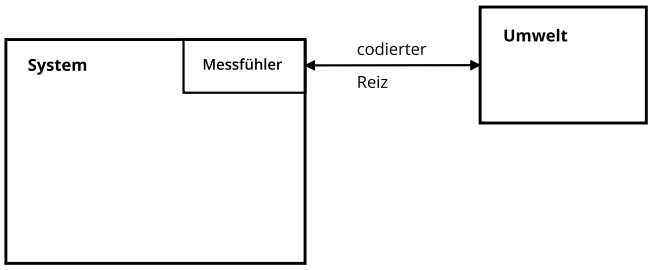


Abbildung 2: Das System nimmt über einen Sensor Reize aus der Umwelt auf.

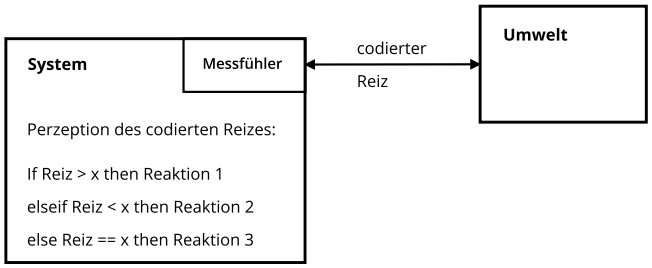


Abbildung 3: Das System verarbeitet (perzipiert) die aufgenommenen Reize und verändert sich dabei.

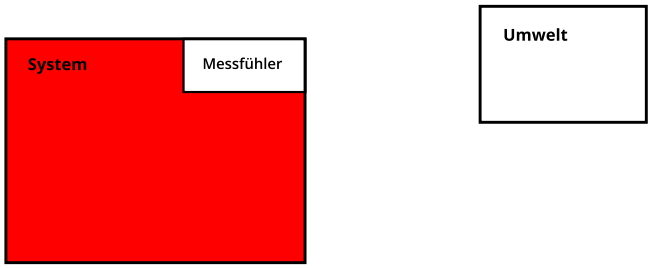


Abbildung 4: Das System hat die aufgenommenen Reize verarbeitet und ist nun ein anderes. Es entspricht jetzt nicht mehr dem System aus Abbildung 2.

Allerdings darf man kein mechanistisches Verhalten zu Grunde legen. Menschen sind keine Maschinen und erst recht keine Computer. Nach einigen fehlgeschlagenen Sozialexperimenten erkannten dies auch die Kybernetiker. Insbesondere reagieren Menschen äußerst sensibel auf äußere Steuerungsversuche – und unterlaufen diese gerne, beispielsweise mit der so genannten Reaktanzreaktion: „Wenn Menschen das Gefühl haben, dass ihre Freiheit, so zu handeln oder so zu denken, wie sie wollen, bedroht oder eingeschränkt ist,“, sagen Aronson/Wilson/Akert ([14], S.252), „wird ein unangenehmer Zustand von Reaktanz hervorgerufen. Diese Reaktanz kann dadurch gemindert werden, indem die bedrohte Handlung ausgeführt wird.“

Daher ist beim Thema Feedback ein gewisses Fingerspitzengefühl notwendig. Der Kommunikationspsychologe Schulz von Thun ([13], S.69] veranschaulicht, dass „die innere Reaktion auf eine Nachricht sich als eine Wechselwirkung zwischen der Saat (gesendeter Nachricht) und dem psychischen Boden [erweist], auf den diese Saat beim Empfänger fällt“. Das heißt, dass der Empfänger eine Botschaft anders wahrnimmt als der Sender, und es daher darauf ankommt, wie der Empfänger die Nachricht wahrnimmt. Er entwickelte das so genannte Kommunikationsquadrat, um menschliche Kommunikation näher zu beschreiben ([24], Abbildung 5). Hierbei ordnet er dem Sender vier Schnäbel auf verschiedenen Ebenen und dem Empfänger passende Ohren zu. Jede Botschaft besteht immer aus diesen vier Ebenen, die je nach Sender und Empfänger unterschiedlich gewichtet sind.

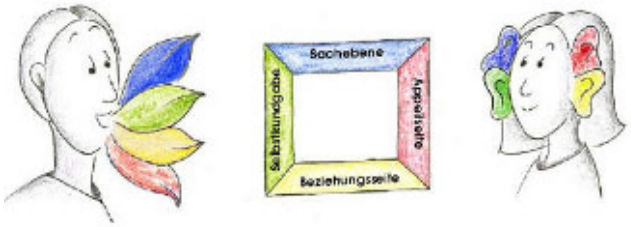


Abbildung 5: Das *Kommunikationsquadrat* von Friedemann Schulz von Thun [24].

Die vier Ebenen sind Sachinhalt, Selbstkundgabe, Beziehung und Appell. Der *Sachinhalt* besteht aus überprüfbaren Daten und Fakten, die wahr oder unwahr, relevant oder irrelevant, hinlänglich bekannt oder unbekannt sein können. Der Sender muss sich auf dieser Ebene klar und deutlich ausdrücken und überprüfen, ob die von ihm kommunizierten Daten wahr, relevant und bekannt sind. Die *Selbstkundgabe* enthält einen Teil, in dem der Sender etwas explizit oder implizit über sich selbst sagt. Beispiele sind der gegenwärtige Gefühlszustand oder das eigene Rollenverständnis. Der Empfänger fragt sich auf dieser Ebene, in welcher Stimmung der Sender ist. Die *Beziehungsebene* zeigt, wie der Sender zum Empfänger steht. Der Empfänger fühlt sich auf dieser Ebene durch den Sender behandelt – etwa ungerecht, zuvorkommend oder herablassend. Der *Appell* schließlich sendet die (versuchte) Einflussnahme des Senders, also Wünsche, Anregungen oder gar Befehle. Der Empfänger fragt sich, was er tun (oder lassen) soll.

Da Menschen dazu neigen, sich zu ihrem bisherigen Verhalten konsistent zu zeigen, fühlen sie sich durch Kritik oftmals angegriffen. Daher ist es notwendig, Feedbackregeln aufzustellen und einzuhalten. Urs Bärtschi gibt in seinem Artikel *Eine tragfähige Feedbackkultur - Teil I* [25] mehrere wichtige Hinweise zu konstruktivem Feedback. Wichtig

sei es demnach, Gelungenes wahrzunehmen und auszusprechen, statt Fehler zu suchen und damit den Empfänger anzugreifen. Feedback zeige außerdem Differenzen zwischen Selbst- und Fremdwahrnehmung auf und kann so Verbesserungen ermöglichen. Es ermöglicht somit Lernprozesse, die zu Verhaltensänderungen führen können. Weiterhin gibt Bärtschi den Tipp, dass „das Sprechen in Ich-Botschaften sich als große Hilfe in allen konfliktträchtigen Situationen [erweist]". Alle Aussagen sollten also möglichst nur ein Sprechen über sich selbst sein, zum Beispiel:

- „Mir geht es ..., wenn ich das erlebe."
- „Ich denke ..., wenn Sie das sagen."
- „Ich fühle mich ..., wenn das geschieht."
- „Mein Wunsch wäre ..."

Leider neigen wir stattdessen in Konfliktsituationen dazu, Vorschläge, Forderungen oder gar Drohungen zu äußern. Solche oder ähnliche Redewendungen führen direkt in die Eskalation:

- „Sie müssen endlich damit aufhören!"
- „Sie sagen immer ... !"
- „Sie hören mir nie zu!"
- „Weil Sie das getan haben, werde ich jetzt ... !"

Feedback ist unverzichtbar. Fraglich ist nur, ob beiden Parteien der Sinn und Zweck des Feedbacks bekannt ist. Im Zweifelsfall sind daher Schulungen zur menschlichen Kommunikation erforderlich und eine externe Moderation und Mediation hilfreich, wie sie beispielsweise Werpers [15] und Brandenburg/Faber [26] beschreiben.

Maßnahmen, um Fehlermanagement zu fördern

Jeder Mensch neigt dazu, Fehler zu begehen. Insbesondere in der Anwendungsentwicklung ist es zwar theoretisch machbar, aber praktisch unmöglich, Software zu verifizieren. Stattdessen wird sie nur validiert, also praktisch auf die Anwesenheit von Fehlern hin überprüft. Insbesondere komplexe Projekte mit mehreren hunderttausend Code-Zeilen und hunderten Entwicklern benötigen eine ausgefeilte Organisationsstruktur, um Fehler zu vermeiden. Dies wird umso wichtiger, je komplexer ein Projekt und je sensibler der Einsatzbereich ist. Schließlich steuern Computer heutzutage Bremssysteme in Autos, Kernspintomographen, Atomsprengköpfe oder die Ariane 5. Auch in der Systemadministration werden komplexe Systeme aufgebaut, die nicht verifizierbar sind. Außerdem muss der Systemadministrator oftmals zusammen mit Benutzern Systeme nach Fehlern untersuchen und diese beheben. Es ist daher erforderlich, ein Fehlermanagement zu etablieren und die beteiligten Mitarbeiter in Trainingseinheiten zu schulen.

Verschiedene Studien belegen, dass der Großteil aller Unfälle durch menschliches Versagen herbeigeführt werden. Leider existieren für den Bereich Software-Entwicklung und IT-Sicherheit keine Zahlen, da es keine Veröffentlichungspflicht gibt. Allerdings werden die Fehlerursachen nicht merklich divergieren, hat doch ein simpler Rundungsfehler in einer Steuerungsroutine aus der ersten Ariane 5G ein 320.000.000 Euro teures Feuerwerk gemacht [27]: Die Steuerroutinen der Ariane 4 wurden unangepasst für die Ariane 5 verwendet. Dabei musste die Software eine 64-Bit-Gleitkommazahl in eine vorzeichenbehaftete 16-Bit-Ganzzahl umwandeln, was zu einem Überlauf führte. Die Steuerung stürzte daraufhin ab und die Rakete explodierte. Bemerkenswert ist hierbei, das die Laufzeitumgebung der Programmiersprache ADA den Variablen-Überlauf hätte feststellen und behandeln können. Diese Funktion wurde aber von den Entwicklern als Ballast angesehen und deaktiviert. Menschen haben also bewusst entschieden, Fehlerbehandlungsroutinen nicht zu benutzen.

Fehlermanagement versucht nicht, den Schuldigen ausfindig zu machen, da menschliches Denken und Handeln zu komplex ist, um einfache Kausalitätsmuster zu verfolgen. Die Kernaussagen für ein Fehlermanagement lauten stattdessen nach Brandenburg/Faber ([26], S.216f.):

- Es gibt keine Null-Fehler-Systeme.
- Jeder Mensch ist ein permanenter potenzieller Gefahrenherd für seine Umwelt.
- Es gibt kein menschliches Versagen. Denn Fehler entstehen nicht urplötzlich, sondern aus einer Kette von Ereignissen und Faktoren.

Es ist daher erforderlich, Fehler, Fehlerfolgen und Fehlerkonsequenzen zu analysieren, aus diesen Fehlern zu lernen und entsprechende Präventionsmaßnahmen zu trainieren. Dazu ist es notwendig die Denkstrukturen von Individuen und Teamstrukturen zu verstehen. Individuen sind keine Turing-Maschinen und handeln daher nicht streng rational, sondern kontextabhängig. Entscheidungen werden durch Motive und Motivationen, Absichten und Emotionen beeinflusst. So erhöht Schlafentzug beispielsweise die Fehleranfälligkeit (vgl. [28] und [29]).

Neben Aspekten der Persönlichkeitspsychologie kommt in Gruppen auch die Sozialpsychologie zum Tragen. So lassen sich Individuen durch tatsächliche oder vorgespielte Autorität beeinflussen, wie das Milgram-Experiment (vgl. [8]) und das Stanford-Prison-Experiment [30] zu Autorität eindrucksvoll belegen. Neben den Ebenen Individuum und Team muss auch die Organisation untersucht und als Fehlerquelle in Betracht gezogen werden. Es ist zu einfach, lediglich ein Team oder ein Individuum als Fehlerquelle identifizieren zu wollen und dabei zu ignorieren, das handelnde Menschen oft nur am Ende einer Kette von vorangegangenen Fehlentscheidungen auf Organisationsebene stehen.

Literatur und Links

[1] Webseite des BIBB-Projekts *Früherkennung von Qualifikationsentwicklungen*: http://www.bibb.de/de/wlk8205.htm
[2] Artikel über das BIBB-Projekt in der ZEIT Nr. 15, 2008: http://www.zeit.de/2008/15/C-Berufsberatung
[3] Erich Staudt, Norbert Kailer, Marcus Kottmann: Kompetenzentwicklung und Innovation. Münster, Waxmann 2002.
[4] Michael Gessler: Das Kompetenzmodell. In: Reiner Bröckermann, Michael Müller-Vorbrüggen (Hgg.), Handbuch Personalentwicklung; Die Praxis der Personalbildung, Personalförderung und Arbeitsstrukturierung. Stuttgart, Schäffer-Poeschel 2006, SS. 23 - 42.
[5] Klaus North, Kai Reinhardt: Kompetenzmanagement in der Praxis; Mitarbeiterkompetenzen systematisch identifizieren, nutzen und entwickeln. Wiesbaden, Gabler 2005.
[6] Uwe Peter Kanning (Hg.): Förderung sozialer Kompetenzen in der Personalentwicklung, Göttingen, Hogrefe 2007.
[7] Nicole Britz: Betriebs(system)blind. In: Uptimes, Mitgliederzeitschrift der German Unix User Group, 2007-4, S. 30-35.
[8] Felix Pfefferkorn: Admins ausbilden. In: Uptimes 2008 (FFG-Proceedings), S. 11-20.
[9] Uwe Vigenschow, Björn Schneider: Soft Skills für Softwareentwickler, Heidelberg, dpunkt 2007.
[10] Robert B. Cialdini: Die Psychologie des Überzeugens. Bern, Hans Huber 2007.
[11] Björn Migge: Handbuch Coaching und Beratung. Weinheim/Basel, Beltz 2005.
[12] Paul Watzlawick: Wie wirklich ist die Wirklichkeit; Wahn, Täuschung, Verstehen. München, Piper 2007.
[13] Friedemann Schulz von Thun: Miteinander reden 1. Reinbek, Rowohlt 2007.
[14] Elliot Aronson, Timothy D. Wilson, Robin M. Akert: Sozialpsychologie. München, Pearson 2003.
[15] Katja Werpers: Konfliktmanagement in Organisationen. In: [4], S. 198-213.
[16] Lutz von Rosenstiel: Führung. In: Heinz Schuler (Hg.), Lehrbuch der Personalpsychologie. Göttingen, Hogrefe 2006, S. 353-384.
[17] Mail von NetBSD-Mitgründer Adam Glass, die über Theo de Raaths Disqualifizierung informiert: http://mail-index.netbsd.org/netbsd-users/1994/12/23/0000.html
[18] D. G. Pruitt, J. Z. Rubin: Social conflict; Escalation, stalemate and settlement. New York, Random House 1986. Zitiert nach: [12], S. 200.
[19] M. A. Rahim: Measurement of organizational conflict. In: Journal of General Psychology 109/1983, S. 79-86.
[20] Claude E. Shannon, Warren Weaver: The Mathematical Theory of Communication. Urbana, University of Ilinois Press 1963.
[21] Vera F. Birkenbihl: Kommunikationstraining. Heidelberg, Mvg 2006.
[22] Paul Watzlawick: Wenn du mich wirklich liebtest, würdest du mehr Knoblauch essen. München, Piper 2008.
[23] Paul Watzlawick: Anleitung zum Unglücklichsein. München, Piper 2008.
[24] Kommunikationsquadrat nach Friedemann Schulz von Thun: http://www.schulz-von-thun.de/index.php?article_id=71
[25] Urs Bärtschi, Eine tragfähige Feedbackkultur - Teil I: http://www.perspektive-mittelstand.de/Personalfuehrung-Feedback-als-Fuehrungsinstrument-Leitfaden-Teil-I/management-wissen/401.html
[26] Torsten Brandenburg, Thomas Faber: Fehlermanagement-Training – Entwicklung sozialer Kompetenzen und der Umgang mit Fehlern in Risiko-Arbeitsbereichen. In: [4], S.216-237.

[27] Jacques-Louis Lions: Flight 501 Failure. Report by the Inquiry Board, European Space Agency, 1996: http://esamultimedia.esa.int/docs/esa-x-1819eng.pdf
[28] Thomas A. Langens and Kurt Sokolowski and Heinz-Dieter Schmalt: Das Multi-Motiv-Gitter (MMG). In: John Erpenbeck, Lutz von Rosenstiel (Hgg.), Handbuch Kompetenzmessung, Stuttgart, Schäffer-Poeschel 2003, S.71-79.
[29] Howard S. Friedman, Miriam W. Schustack: Persönlichkeitspsychologie und Differentielle Psychologie. München, Pearson 2004.
[30] Stanford-Prison-Experiment: http://www.prisonexp.org/deutsch/

Über Stefan

Stefan Schumacher ist geschäftsführender Direktor des Magdeburger Instituts für Sicherheitsforschung und gibt zusammen mit Jörg Sambleben das Magdeburger Journal zur Sicherheitsforschung heraus. Er befasst sich seit knapp 20 Jahren als Hacker mit Fragen der Informations- und Unternehmenssicherheit und erforscht Sicherheitsfragen aus pädagogisch-psychologischer Sicht. Seine Forschungsergebnisse stellt er auf Fachkongressen und in Publikationen vor. Seine Schwerpunkte liegen auf Social Engineering, Security Awareness, Organisationssicherheit, internationale Cyber-Security und Mensch-Maschine-Interaktion.

Geschichtsstunde IV
Die Evolution des Internet

Unix und Internet – ein ideales Paar: Ohne Unix hätte sich das Internet, wie wir es kennen, möglicherweise nicht so schnell entwickelt, oder vielleicht wäre etwas viel Bizarreres dabei herausgekommen. Auch die Russen (genauer: die UdSSR) waren nicht ganz unschuldig.

von **Jürgen Plate**

Auf lange Sicht mag der Aspekt,
die zwischenmenschliche
Kommunikation zu fördern,
wichtiger werden
als technische Ziele.

Andrew S. Tanenbaum in seinem Buch *Computer-Netzwerke*

Das Internet wurde vor etwa 20 Jahren aus einem Forschungsprojekt namens *ARPA-Net* geboren, das primär vom US-amerikanischen Verteidigungsministerium finanziert wurde. Das Ziel dieses experimentellen Projektes war, ein Netzsystem zu entwickeln, das auch partielle Ausfälle verkraften konnte. Kommunikation sollte nur zwischen einem Sender und einem Empfänger stattfinden. Das Netz dazwischen galt als unsicher. Die beiden Endpunkte der Kommunikation, *Sender* und *Empfänger*, trugen jegliche Verantwortung für die richtige Datenübertragung. Jeder Rechner im Netz sollte mit jedem anderen kommunizieren können.

Die *Advanced Research Projects Agency* (ARPA) entstand 1957 als Reaktion auf den Start des Sputniks durch die UdSSR (Abbildung 1). Insofern sind eigentlich die Sowjets schuld am Internet. Aus ähnlichen Gründen wurde etwas später auch die *National Space and Aeronautics Agency* (NASA) ins Leben gerufen. Die ARPA hatte die Aufgabe, Technologien zu entwickeln, die auch für das Militär von Nutzen sein könnten. Die ARPA forschte nicht selbst, sondern vergab und kontrollierte Aufträge an Universitäten und Forschungsinstitute. Später wurde die ARPA in *Defense Advanced Research Projects Agency* (DARPA) umbenannt, da ihre Interessen primär militärischen Zwecken dienten.

Um die geforderte Zuverlässigkeit eines nichthierarchischen Netzes zu erreichen, sollte das Netz als ein paketvermitteltes Netz (*packet-switched network*) gestaltet werden. Bei der Paketvermittlung sind zwei Partner während der Kommunikation nur virtuell miteinander verbunden. Der Sender zerlegt die zu übertragenden Daten in Stücke und überträgt sie über die virtuelle Verbindung. Der Empfänger setzt die Stücke nach dem Eintreffen wieder zusammen. Im Gegensatz dazu sind in einem leitungsvermittelten Netz (*circuit-switched network*) für die Dauer der Datenübertragung die Kommunikationspartner fest miteinander verbunden.

Abbildung 1: Der Sowjetische Sputnik führte zur Gründung von ARPA und NASA. (Foto: NASA)

Geburt: 1969

An einem Netzwerk zwischen Computern wurde schon seit dem Jahr 1960 geknobelt [1]. Die Geschichte nahm aber damit Fahrt auf, dass sich Bob Taylor von der ARPA darüber ärgerte, dass er drei verschiedene Terminals brauchte, um mit drei Universitäten zu kommunizieren. Denn alle drei betrieben für die ARPA militärische Grundlagenforschungen. Dieser Wunsch nach einem – wie man heute sagen würde – einheitlichen Kommunikationsprotokoll wurde von J.C.R. Licklider und Bob Taylor 1968 in dem bahnbrechende Papier *The Computer as Communications Device* aufgegriffen [2].

Die Geburt des Internet dürfte auf dem 2. September 1969 liegen. An diesem Tag schloss Leonard Kleinrock in seinem Labor an der Universität von Kalifornien in Los Angeles den ersten Computer an einen *Interface Message Processor* (IMP) an (Abbildung 2). „Wir hielten das nicht gerade für einen historischen Moment", erinnerte sich Kleinrock 1999 gegenüber einem AP-Reporter [3]: „Wir hatten nicht einmal eine Kamera dabei. Jeder war bereit, um mit dem Finger auf den anderen zu zeigen, wenn es nicht funktionieren sollte."Der IMP war ein Klotz von einem Spezialrechner, der nach militärischen Normen von der Firma *Bolt, Beranek & Newman* (BBN) gebaut worden war. Seine einzige Aufgabe bestand darin, Daten zu senden und zu empfangen, den Empfang zu überprüfen und das Senden zu wiederholen, wenn etwas nicht klappte. Ein IMP sollte einem Computer vorgeschaltet sein und rund um die Uhr laufen – eine beträchtliche Anforderung zu einer Zeit, in der Rechner jede Woche für mehrere Stunden gewartet werden mussten.

Abbildung 2: Folie aus Leonard Kleinrock Präsentation *Eine kurze Geschichte des Internet* vom 29.10.2004 auf der Veranstaltung *The 35th Anniversary of the Internet* an der UCLA. (Quelle: Leonard Kleinrock, [4])

Frank Heart, Ben Barker, Bernie Cosell, Will Crowther, Robert Kahn, Severo Ornstein und Dave Walden bildeten das Software-Team für die Programmierung der IMPs und später die *Network Working Group* am IETF. Am 7. April 1969 schickte Crocker ein Memo mit dem Titel *Host-Software* als *Request for Comments* herum [5] – das erste Dokument der heute als RFC bekannten Internet-Standards, von denen es über 7000 gibt. Die Implementierung der Netzwerk-Protokolle, eine Reihe von Programmen, erblickte 1970 als *Network Control Protocol* (NCP) das Licht der Welt. An der UCLA arbeiteten Vint Cerf, Steve Crocker und Jon Postel zusammen mit Kleinrock an den Protokollen.

Als der IMP am 2. September 1969 in Los Angeles mit einem Computer in Kleinrocks Büro Daten austauschte, war die Geburt des Internet noch nicht ganz zu Ende. BBN musste drei weitere IMP liefern, die peu à peu in Stanford, Santa Barbara und Salt Lake City aufgestellt wurden. Zwischen dem Büro von Kleinrock und dem Stanford Research Institute ging anschließend das erste `ping` durch die Leitung. Somit lief zwischen Stanford und Los Angeles lief das erste funktionsfähige *Wide-Area-Network* (WAN): Das Internet war geboren.

Ende 1969 wurde dann von der University of California Los Angeles, der University of California Santa Barbara, dem Stanford Research Institute und der University of Utah ein experimentelles Netz, das ARPA-Net (Abbildung 3), mit vier Knoten in Betrieb genommen. Diese vier Universitäten wurden von der ARPA gewählt, da sie bereits eine große Anzahl von ARPA-Verträgen hatten. Das ARPA-Netz wuchs rasant und überspannte bald ein großes Gebiet der Vereinigten Staaten.

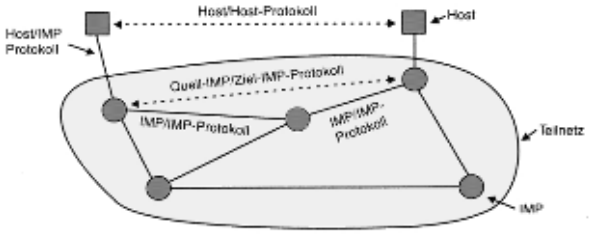


Abbildung 3: Prinzipielle Architektur des frühen ARPA-Netzes. Von TCP/IP war noch nicht die Rede.

Selbst heute, fast 40 Jahre später, ist die Bedeutung der kulturtechnischen Leistung des Internet erst in Umrissen zu erahnen. Der weitere Ausbau war zunächst langsam und gemächlich verlaufen: Nach mehr als 10 Jahren hatten gerade einmal rund 200 Systeme (Hosts) im ARPA-Net zusammengearbeitet. Doch schon zu diesem Zeitpunkt war das ARPA-Net kein Netzwerk wie jedes andere auch, sondern definierte eine Kommunikationsstruktur. Jeder Host im ARPA-Net konnte ein Zentralcomputer in einem lokalen Netzwerk sein, so daß das ARPA-Net ein Netzwerk aus Netzwerken bildete – eben ein *Internet*. Es wucherte daher unaufhaltsam weiter. Allmählich beschleunigte sich das Wachstum und nahm einen exponentiellen Verlauf. Im Oktober 1984 zählte man rund 1000 Hosts. 1987 waren es etwa 10.000. Und 1989, nur zwei Jahre später, über 100.000 (Abbildung 4).

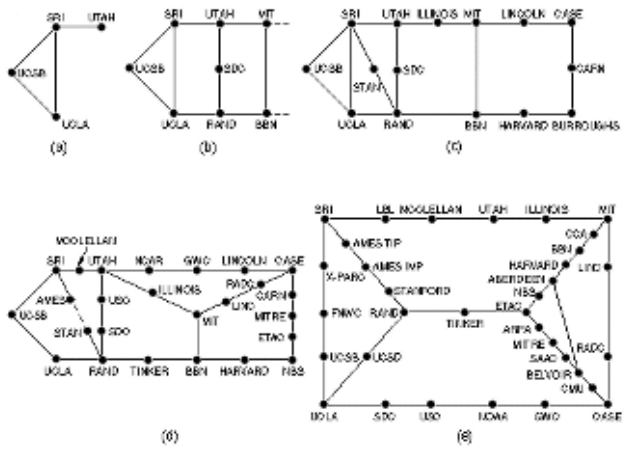


Abbildung 4: Wachstum des ARPA-Net nach Andrew S. Tanenbaum [6]: a = Dezember 1969, b = Juli 1970, c = März 1971, d =April 1971, e = September 1972.

Mit der Zeit und angesichts des sich immer weiter ausbreitenden ARPA-Nets wurde klar, dass die bis dahin gewählten IMP-Protokolle nicht mehr für den Betrieb eines größeren Netzes geeignet war, das auch mehrere (Teil-)Netze miteinander verband. Anfang der 1970er Jahre kam die Idee auf, die IMPs von denjenigen Computern abzulösen, die keine Spezialrechner waren. Im Jahre 1972 beschäftigte sich der Xerox-Informatiker Bob Metcalfe damit, das hausinterne Netzwerk MAXC an das ARPA-Net zu hängen. Dabei erfand er eine Übertragungstechnik mittels Koaxialkabel, die er *Ethernet* nannte. Die Erfindung erregte das Interesse von Bob Kahn und Vint Cerf, die 1974 den ersten Vorschlag für ein einheitliches Rechnerprotokoll machten. Dieses Protokoll wurde *Transmission Control Protocol/Internet Protocol* (TCP/IP) genannt.

Generalisierung: TCP/IP

TCP/IP wurde mit der Zielsetzung entwickelt, mehrere verschiedenartige Netze zur Datenübertragung miteinander zu verbinden. Die wohl wichtigste Eigenschaft des Designs war die Aufteilung in einzelne logische Ebenen (oder Schichten, Protokoll-Stacks), wobei jeweils eine Schicht nur mit ihren Nachbarn kommuniziert. So lassen sich einzelne Ebenen austauschen, ohne das ganze Protokollgebäude zum Einsturz zu bringen. Die IP-Ebene sorgte für die Adressierung und die Weiterleitung der Datenpakete. Auf dieser Protokollebene können die Datenpakete unterschiedliche Wege zwischen Absender und Empfänger nehmen und es dürfen auch mal Pakete verloren gehen. Die darüber liegende Ebene TCP sorgt dann beispielsweise dafür, dass die eingehenden Pakete vollzählig und in der richtigen Reihenfolge auf den Betriebssystemschnittstellen landen. Sie erledigt auch die Fluss-Steuerung sowie den Auf- und Abbau von Verbindungen. Unterhalb von IP hingegen können die verschiedensten Übertragungsverfahren zum Einsatz kommen, also etwa DSL, Glasfaserkabel, Funkstrecken, UMTS oder LTE. TCP/IP erhielt am 1. Januar 1983 den Rang eines offiziellen Standards. Viele Netzwerker halten daher auch dieses Datum für das offizielle Geburtsdatum des Internet.

Da etwa zur gleichen Zeit an der University of California ein neues Betriebssystem mit Namen *Unix* entwickelt wurde, beauftragte die DARPA die Firma BBN und die University of California at Berkeley mit der Integration von TCP/IP in UNIX [7]. Dies bildete auch den Grundstein des Erfolges von TCP/IP in der Unix-Welt. Ein weiterer Meilenstein beim Aufbau des Internet war die Gründung des NSF-Net der *National Science Foundation* (NSF) Ende der 1980er Jahre, die den amerikanischen Hochschulen damit fünf neue Super Computer Center zugänglich machte. Dies war ein wichtiger Schritt, da bis zu diesem Zeitpunkt Super Computer nur der militärischen Forschung und einigen wenigen Anwendern sehr großer Firmen zur Verfügung standen.

Parallel zu den Entwicklungen im ARPA-Net und NSF-Net arbeitete die *International Standards Organization* (ISO) seit den achtziger Jahren an der Standardisierung der Rechner-Kommunikation. Die Arbeiten mündeten in die Definition des ISO/OSI-Referenzmodells. Die Entwicklung entsprechender OSI-Protokolle und -Anwendungen gestaltete sich aber äußerst zäh und ist bis heute nicht abgeschlossen. Hersteller und Anwender konnten darauf natürlich nicht warten, und so wurde die Internet Protokoll-Familie TCP/IP im Lauf der Zeit in immer mehr Betriebssystemen implementiert. TCP/IP entwickelte sich so unabhängig von den offiziellen Standardisierungsbestrebungen zum Quasi-Standard.

Im Jahr 1983 wurde das ARPA-Net schließlich von der *Defense Communications Agency* (DCA) aufgeteilt, welche seine Verwaltung von der DARPA übernahm. Der militärische Teil des ARPA-Net wurde in ein separates Teilnetz abgetrennt, das MIL-Net, das durch streng kontrollierte Gateways vom Rest des ARPA-Net separiert wurde, das der Forschungsteil blieb. Nachdem TCP/IP das einzige offizielle Protokoll des ARPA-Net wurde, nahm die Zahl der angeschlossenen Netze und Hosts rapide zu. Das ARPA-Net wurde von Entwicklungen überrannt, die es selber hervorgebracht hatte. Das ARPA-Net in seiner ursprünglichen Form

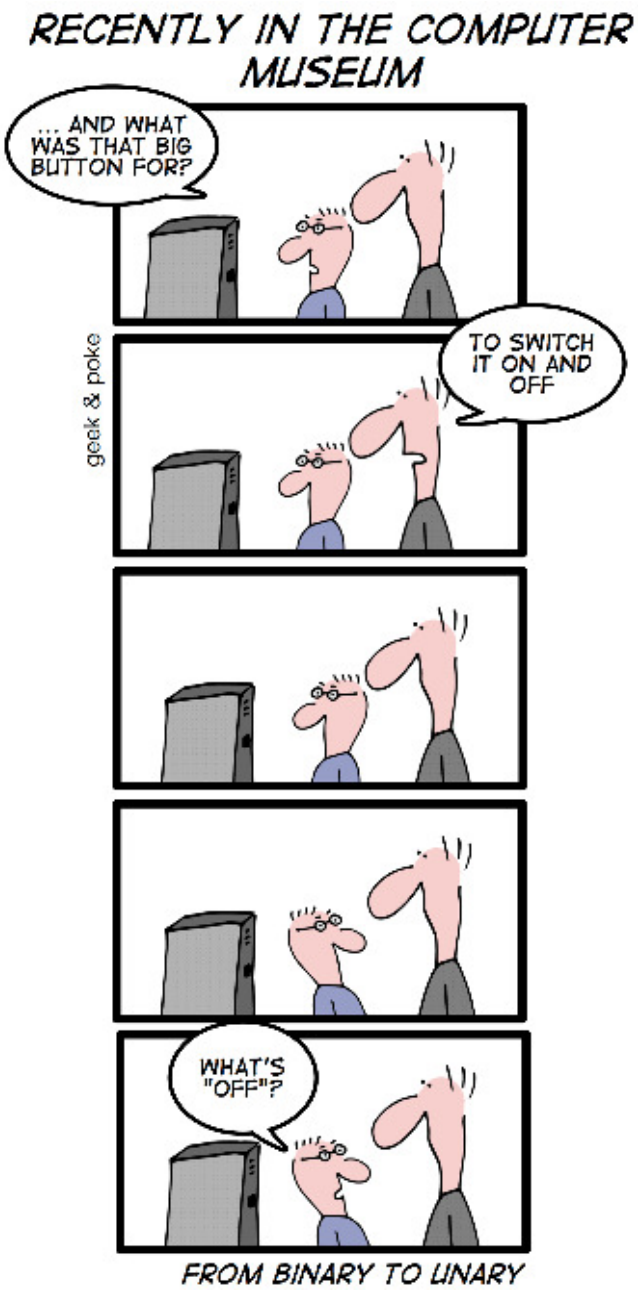
existiert heute nicht mehr.Das MIL-Net ist aber noch in Betrieb.

Wendejahr: 1989

Das Jahr 1989 markiert einen Wendepunkt. Zum einen wurde zum 20. Geburtstag des ARPA-Net dessen Auflösung beschlossen – es ging in das 1986 gegründete Netzwerk der National Science Foundation (NSF) über. Zum anderen schrieb Tim Berners-Lee am Genfer Kernforschungszentrum CERN ein Diskussionspapier mit dem Titel *Information Management: A Proposal* [8], mit dem er den Kommunikationsprozess am CERN verbessern wollte. Aus diesem Vorschlag entwickelte sich in den nächsten Monaten das *World Wide Web* (WWW).

Das System leistete erheblich mehr als geplant. Es entpuppte sich als das einfachste, effizienteste und flexibelste Verfahren, um beliebige Informationen im Internet zu publizieren. Die Einführung des WWW sorgte für den bis dato kräftigsten Wachstumsschub des Internet. Dauerte es fast 20 Jahre (1969 bis 1989), bis mehr als 100.000 Hosts zusammengeschlossen waren, so brauchet es nur zwei Jahre, bis die Zahl von 300.000 auf über eine Million wuchs (1990 bis 1992). Der Durchbruch setzte dann 1993 ein, als Marc Andreessen sein grafisches Programm *Mosaic* herausbrachte, mit dem auch der ungeschulte Computerlaie auf die vormals kryptischen Kommandos und erhebliches Spezialwissen verzichten konnte: Nun genügte ein Mausklick. Aus Mosaic wurde ein Jahr später *Netscape*, und irgendwann bemerkte auch Microsoft das Internet. Wie wir alle wissen, war damit die Entwicklung nicht zu Ende, denn das sogenannte Web 2.0 brachte uns die *Social Networks* mit Facebook oder Twitter, wobei man über manche Entwicklungen durchaus geteilter Meinung sein darf.

Laien benutzen die Begriffe *World Wide Web* und *Internet* heute vielfach synonym, die Größe des Internet verdoppelt sich alle paar Jahre. Schätzungen von *Internetsworldstats* aus Mitte 2012 gehen von rund 2,5 Milliarden Internet-Nutzern weltweit aus [9]. Ende Oktober 2013 gab es etwa 190 Millionen Top Level Domains [10]. Allerdings sind derartige Zahlen und Erhebungen nur mit großer Vorsicht zu genießen. Schon die technische Messung der Host-Zahlen ist nicht trivial und interpretationsbedürftig. Nur eines ist wirklich sicher: Das Internet und das WWW breiten sich seit Jahren mit schwindelerregender Geschwindigkeit aus. Ende 2013 existierten ungefähr 16 Millionen de-Domains. Rechnerisch besitzt damit, neben Firmen und Behörden, fast jeder sechste Bundesbürger eine eigene de-Domain. Das begann übrigens vor 25 Jahren: Am 5. November 1986 erfolgte der Eintrag der Top-Level-Domain *.de* in der IANA-Datenbank. Damit war auch in Deutschland der Startschuss für das Internet gefallen.



Alte Kumpel: Mailboxen und BTX

Bevor alle Welt im World Wide Web surfte und in E-Mails der Spam ein Vielfaches der Nutzinformation einnahm, gab es natürlich auch schon Möglichkeiten der Kommunikation: Online-Dienste wie *CompuServe*, *T-Online* oder *AOL* und so genannte Mailboxsysteme, kurz *Mailbox*. Dies ist ein Computer, den man per Telefon und Modem oder über ISDN erreichen kann (ja, damals waren 19

KBit/s schon schnell und 64 KBit/s superschnell). Er bot dann bestimmte Dienstleistungen an, unter anderem den Tausch von Dateien – Sie sehen, alles alter Käse. Neben dem Datenaustausch mit anderen Benutzern, also Nachrichten und Dateien, gab es so genannte Foren oder Schwarze Bretter, die den Newsgruppen entsprechen und gewissermaßen die Vorläufer der Wikis und Blogs vorstellen. Außerdem gab es Programmbibliotheken, aus denen die Benutzer Programme abrufen und im eigenen Computer speichern können, Unterhaltung mit anderen Benutzern (Chat) und Spiele, die man mit anderen Benutzer der Mailbox spielt – darunter der immer-noch-Klassiker *Nethack*. Die Leistungen der Mailbox waren aber zunächst nur auf wenige Rechner beschränkt, sodass eine relativ kleine Gruppe von Personen davon profitierte.

Aus dieser Isolation heraus entstanden dann einerseits Online-Dienste wie CompuServe, die mittels Einwählpunkten oder *Points of Presence* weltweit Rechner unterhielten (Abbildung 5). Die Daten liefen in einer Zentrale zusammen. Ein anderer Weg, die lokale Begrenzung zu überwinden, boten Mailboxnetze wie *Fidonetz*, *Zerberusnetz* oder das *Mausnetz* (von Spöttern „Tiernetze“ tituliert). Hier tauschten die Mailboxrechner nachts oder sogar stündlich die neuen Mitteilungen automatisch untereinander aus. Die Mailboxnetze unterschieden sich zum Beispiel in der Form der Datenweitergabe von Einträgen in Schwarzen Brettern und E-Mails. Das Fidonetz transportierte die Daten grundsätzlich nach dem gleichen Verfahren, unterschied aber in *personal mail* (entspricht E-Mail) und *netmail* (entspricht News). Fast alle größeren Mailboxen boten später neben lokalen Informationen auch einen Zugang zum Internet in Form von News und E-Mail. Außerdem gibt es Übergänge, also Gateways, zwischen den Netzen.

Abbildung 5: Ich gestehe – auch ich war CompuServe-Kunde.

Auch das *User Network* (Usenet) von Unix gehört von seiner Arbeitsweise in diese Gruppe. Allerdings tauschten die Nutzer des Usenet und der legendären Programme der `uucp`-Familie (Unix-to-Unix-Copy) nicht nur Mail, News und Dateien aus, sondern führten auch Kommandos auf einen fernen Computer aus. Das Ganze war ins Betriebssystem eingebunden und für den Normalbenutzer transparent.

Die verschiedenen Online-Dienste zeichneten sich durch unterschiedliche Angebote aus. Die beiden größten in Deutschland waren AOL und T-Online. Sie boten ein großes Angebot an Tagesinformationen (etwa Agenturnachrichten oder Börsenwerte), Computerinformationen und elektronischen Treffpunkten (Diskussionsforen und Konferenzen). Erst sehr spät trat Microsoft mit seinem *Microsoft Network* (MSN) hinzu, das beim Launch von Windows 95 heftig beworben wurde. Heute ist MSN nur noch ein Portal, AOL eher ein Filmproduzent, T-Online wieder in die Telekom eingegliedert und CompuServe gänzlich verschwunden.

Im weitesten Sinn gehörte auch das deutsche *Bildschirmtext*-System (BTX) zu den Mailboxen – es wurde aber vom Hoheitsträger betrieben. Die Deutsche Bundespost startete damit einen interaktiven Online-Dienst, der anfangs ein spezielles Gerät erforderte. 1993 wurde BTX Bestandteil des neuen Dienstes *Datex-J*. BTX verlor zunehmend seine Bedeutung aufgrund der Konkurrenz durch das offene Internet. In Deutschland wurde BTX im Mai 2007 endgültig eingestellt – 30 Jahre nach seiner Vorstellung bei der Internationalen Funkausstellung (IFA) in Berlin.

Damals hatte der Anbieter aus einem Feldversuch mit je etwa 2.000 Teilnehmern in Düsseldorf und Berlin erwartbare Nutzerzahlen hochrechnen wollen. Sie wurden nach dem offiziellen Start 1983 allerdings nie erreicht: 1986 sollten es rund eine Million sein, tatsächlich waren es aber nur um die 60.000. Die Million wurde erst zehn Jahre später erreicht, nachdem BTX ab 1995 mit dem neuen T-Online-Angebot inklusive E-Mail und Internet-Zugang gekoppelt war. Am 31. Dezember 2001 wurde der ursprüngliche BTX-Dienst offiziell abgeschaltet und nur noch eine abgespeckte Variante für Online-Banking bis zum 10. Mai 2007 betrieben.

Das Letzte

Menschen nutzen das Internet heute für die gleichen Dinge, für die sie früher Briefe, Telefax, Te-

lefon, oder such den Wochenmarkt genutzt haben. Für einige ist es das WWW sogar zum Ersatz für Kneipe und Verein geworden. Ich halte jedoch nichts von Menschen, die das Gespenst der *Digitalen Demenz* an die Wand malen, nur weil die eigenen Kids auf dem Schlautelefon rumdaddeln, statt draußen auf Bäume zu kletten. Letzteres ist für Stadtkinder sowieso unmöglich.

Es ist auch falsch, dass das das Internet ein *rechtsfreier Raum* sei. Quatsch: Gerade zu diesem Medium werden zahlreiche Urteile verkündet. So klagte die Stadt Heidelberg gegen den Druckmaschinenhersteller Heidelberg, um sich die Domain *www.heidelberg.de* zu sichern. Zahlreiche Anwaltskanzleien verdienen sich mit dem Geschäftsmodell *Internet-User-Abmahnung* eine goldene Nase – auch wenn ab und an eine Oma ins Räderwerk gerät, die weder einen DSL-Anschluss noch einen Computer besitzt.

Heute werden große Teile der weltweiten Kommunikation abgehört. Amazon und Facebook wissen genauer über das Verhalten ihrer Benutzer und Kunden Bescheid als diese selbst. Die Visionen aus dem Film *Minority Report* rücken näher. Hier kann unsere Demokratie enden und in Schattendiktaturen münden, deren Befehlshaber im Dunkel bleiben.

Ich will aber gar nicht mit ganz so düsteren Ansichten schließen – auch wenn die *Geschichtsstunde* hier endet. Nicht enden soll das Thema *Internet*. Kommende Artikeln beschäftigen sich mit den Internet-Protokollen, den Diensten und der Programmierung eigener Internet-Anwendungen. Insofern schließe ich heute mit dem bekannten Zitat des kürzlich verstorbenen Marcel Reich-Ranicki:

> Die Zeit ist um, man schweigt betroffen, Vorhang zu und alle Fragen offen.

Links und Literatur

[1] Lawrence Roberts, Internet chronology 1960 - 2001: http://www.packet.cc/internet.html

[2] Aufsatz *The Computer as Communications Device*:
http://www.cc.utexas.edu/ogs/alumni/events/taylor/licklider-taylor.pdf

[3] Bericht des AP-Reporters Matthew Fordahl, wiedergegeben Jürgen Plate: Geschichtsstunde. IT-Frühzeit von 1950 bis 1975. Uptimes 2012-2, S. 22 - 26:
http://old.chronicle.augusta.com/stories/1999/09/01/met_269151.shtml

[4] Slide aus Leonard Kleinrock Präsentation *Eine kurze Geschichte des Internet* vom 29.10.2004 auf der Veranstaltung *The 35th Anniversary of the Interne* an der UCLA:
http://internetanniversary.cs.ucla.edu/slides/internet35/kleinrock_a_brief_history_of_the_internet.pdf

[5] Der erste RFC von 1969 handelt von der Host-Software des IMP: http://tools.ietf.org/html/rfc1

[6] Andrew S. Tanenbaum, Computernetzwerke. Hallbergmoos: Pearson 2003, 4. aktual. Aufl.

[7] Jürgen Plate: Geschichtsstunde II. 1975 bis 1990: Unix erobert die Welt. Uptimes 2012-3, S. 30 - 36:
http://www.guug.de/uptimes/2012-3/index.html

[8] Tim Berners Lee, *Information Management: A Proposal*:
http://www.w3.org/History/1989/proposal.html

[9] Internetworldstats: http://www.internetworldstats.com/stats.htm

[10] Domain-Statistiken: http://www.denic.de/hintergrund/statistiken/internationale-domainstatistik.html

Internet Society, History of the Internet:
http://www.internetsociety.org/internet/what-internet/history-internet

Musch, Jochen: Die Geschichte des Netzes – ein historischer Abriss. In: B. Batinic (Hg), Internet für Psychologen. Göttingen, Hogrefe 1997.

Hauben, Michael: *Behind the Net – The Untold History of the ARPANET and Computer Science*. Öffentlicher Entwurf des Kapitel 7 von *Hauben, Michael and Ronda: Netizens – On the History and Impact of Usenet and the Internet*:
http://www.columbia.edu/~rh120/ch106.x07

Hauben, Ronda: The Birth and Development of the ARPA-Net. Öffentlicher Entwurf des Kapitel 8 von *Hauben, Michael and Ronda: Netizens – On the History and Impact of Usenet and the Internet*:
http://www.columbia.edu/~rh120/ch106.x08

Musch, Jochen: Die Geschichte des Netzes – ein historischer Abriss. In: B. Batinic (Hg), Internet für Psychologen. Göttingen, Hogrefe 1997.

Plate, Jürgen: Internet glasklar – Einführung für Studenten. München, Oldenbourg 1996.

Plate, Jürgen: Internet kompakt. München, Pflaum 1997.
Wie schwer es für frühere Generationen war, ins Internet zu kommen, dokumentieren die folgenden Videos aus der Sendereihe *Computer Chronicles*, die bei Youtobe zu finden sind:
http://www.youtube.com/user/ComputerChroniclesYT

Über Jürgen

Jürgen Plate ist Professor für Elektro- und Informationstechnik an der Hochschule München. Er beschäftigt sich seit 1980 mit Datenfernübertragung und war, bevor der Internetanschluss für Privatpersonen möglich wurde, in der Mailboxszene aktiv. Unter anderem hat er eine der ersten öffentlichen Mailboxen – TEDAS der mc-Redaktion – programmiert und 1984 in Betrieb genommen.

Shellskripte mit Aha-Effekt II
Umleitung oder nicht?

Wie Spielen mit Shellkommandos den Bildschirm verschönert.

von **Jürgen Plate**

Läuten Feierabendglocken
Hacken Admins unerschrocken
Auf die Tastatur geschwind,
Schelmisch sinnend wie ein Kind.

Frei nach Joseph von Eichendorff, Der Schalk

Manche Kommandos wie `ls` registrieren, ob ihre Ausgabe auf den Bildschirm geht (Standardausgabe) oder umgeleitet wurde, und zeigen ein entsprechendes Verhalten. Bei `ls` zum Beispiel werden die Datenamen auf dem Terminal mehrspaltig ausgegeben, in eine Datei umgeleitet hingegen einspaltig.

Es stellt sich die Frage, ob man ein solches Verhalten in eigenen Shellskripten auch hinbekommen kann. Dank des Unix-Grundsatzes „Alles ist Datei“ ist auch das Terminal einer Gerätedatei zugeordnet. Es spielt keine Rolle, ob es sich um ein Konsolenterminal, ein Xterm oder vielleicht sogar eine serielle Konsole handelt. Wie die Gerätedatei heißt, verrät das Kommando `tty`:

```
plate@netzmafia:~$ tty
/dev/pts/0
```

Außerdem gibt es noch das Gerät `/dev/tty`, das immer der aktiven Konsole zugeordnet ist. Leicht überprüfen lässt sich das mit dem `echo`-Kommando:

```
plate@netzmafia:~$ echo "Rembremerdeng" >/dev/tty
Rembremerdeng
```

Dieses Wissen lässt sich für Shellskripte verwenden, die eine Eingabeumleitung erkennen. Wer in einem Shellskript mit dem Kommando `tty` die Standardeingabe abfragt, erhält eines von zwei Ergebnissen: Geht die Ausgabe auf den Bildschirm (die Konsole), liefert `tty` das Terminaldevice wie im Beispiel oben, und den Rückgabewert *0*. Im anderen Fall erhält man eine Fehlermeldung (bei Linux: „kein Ausgabegerät“), und der Rückgabewert ist *1*.

Über den Rückgabewert des Kommandos `tty` erhalten wir also Auskunft darüber, ob die Ausgabe auf den Bildschirm geht oder in eine Datei respektive Pipe. Das folgende Beispiel zeigt, wie das geht:

```
#!/bin/sh

# Output von tty zeigen (nur zur Demonstration)
tty <&1 >&2 ; echo "return: $?" >&2

# Hier kommt die eigentliche Abfrage
if tty -s <&1 ; then
  echo "Output ist ein tty" >&2
else
  echo "Output ist KEIN tty" >&2
fi
```

Der erste Aufruf von `tty` dient nur der Demonstration und gehört später entfernt. Damit man auch bei der Ausgabeumleitung etwas sieht, sind alle Ausgaben auf die Standardfehlerausgabe umgeleitet: Die etwas kryptische Zeichenkombination `>&2` leitet die Ausgabe auf die Datei mit der Handle-Nummer 2 – und das ist immer die Standardfehlerausgabe. Weil eventuell die Eingabe des Shellskripts umgeleitet sein könnte, weisen wir das `tty`-Kommando an, von der Standardeingabe mit der Handle-Nummer 1 zu lesen (`<&1`). Der Rückgabewert des Kommandos steht anschließend in der Shell-Variablen `$?`, die mittels `echo` ausgegeben wird.

Nach dem informativen Vorspiel kommt die eigentliche Abfrage mit dem `tty`-Kommando. Da uns diesmal nur der Rückgabewert und sonst nichts interessiert, bekommt das Kommando den Parameter `-s` mit (für *silent*). Und schon kann mit einer bedingten Anweisung entschieden werden, ob der Output des Shellskripts umgeleitet wurde oder nicht. Lässt man das Skript laufen, erhält man folgende Ausgaben:

```
plate@netzmafia:~$ sh tt.sh
/dev/pts/0
return: 0
Output ist ein tty

plate@netzmafia:~$ sh tt.sh > /dev/null
kein Ausgabegerät
return: 1
Output ist KEIN tty
```

Eine einfache Anwendung dafür wäre, die Ausgabe des Programms durch `more` zu leiten, wenn es sich um den Bildschirm handelt, und bei einer Ausgabeumleitung nicht. Zusammen mit `tput` (für *terminal put*) lässt sich als anderes Beispiel auch der Textbildschirm aufmöbeln.

`tput` sendet Steuerbefehle an das Terminal, wobei es die eigentliche Steuerinfo der *terminfo*-Datenbasis entnimmt. Als Parameter dient der symbolische Name der entsprechenden Steuerfunktion. Der große Vorteil von `tput` liegt darin, dass es mit jedem Terminaltyp funktioniert, der in der Terminfo-Datenbank aufgeführt ist. Derzeit akzeptiert `tput` allerdings nur einen Parameter, mehrere Steuerbefehle müssen also nacheinander folgen. Die häufigsten Parameter verzeichnet Tabelle 1.

Paramater	Bedeutung
`blink`	Text auf Blinken schalten
`bold`	Text heller darstellen (hohe Intensität)
`clear`	Bildschirm löschen, Cursor nach links oben
`cup`	Cursor positionieren, zum Beispiel für Zeile 10, Spalte 40: `tput cup 10 40`
`home`	Cursor in linke obere Ecke
`rev`	Text auf Inversdarstellung schalten
`rmso`	Wieder auf normale Darstellung schalten
`setaf`	Schriftfarbe setzen, zum Beispiel: `tput setaf 1`
`sgr0`	Attribute zurücksetzen

Tabelle 1: Die häufigsten Parameter für `tput`

Die folgende Demonstration zeigt das Prinzip der Kombination unseres Detektors für Bildschirmausgabe und `tput`. Am Anfang definiert das Skript leere Variablen: Sollte die Ausgabe auf den Bildschirm gehen, füllt das Shellskript sie mit den entsprechenden Steuerzeichenkombinationen für den Bildschirm, die es per Subshell-Aufruf von `tput`-Befehlen ermittelt. Ist die Ausgabe des Shellskripts dagegen umgeleitet, bleiben die Variablen leer.

```
#!/bin/bash

fg_rot="" ;
fg_gruen="" ;
fg_gelb="" ;
fg_blau="" fg_magenta="" ;
fg_cyan="" ;
fg_weiss=""  fett="" ;
invers=""  ;
attr_end=""

if tty -s <&1 ; then
  fg_rot=$(tput setaf 1)
  fg_gruen=$(tput setaf 2)
  fg_gelb=$(tput setaf 3)
  fg_blau=$(tput setaf 4)
  fg_magenta=$(tput setaf 5)
  fg_cyan=$(tput setaf 6)
  fg_weiss=$(tput setaf 7)
  fett=$(tput bold)
  invers=$(tput rev)
  untersr=$(tput smul)
  attr_end=$(tput sgr0)
fi

echo "Dies ist ${fg_rot}rot${attr_end}"
echo "Dies ist ${fg_gruen}gruen${attr_end}"
echo "Dies ist ${fg_gelb}gelb${attr_end}"
echo "Dies ist ${fg_blau}blau${attr_end}"
echo "Dies ist ${fg_magenta}magenta${attr_end}"
echo "Dies ist ${fg_cyan}cyan${attr_end}"
echo "Dies ist ${fg_weiss}(fast)weiss${attr_end}"
echo "Dies ist ${fett}fett${attr_end}"
echo "Dies ist ${invers}reverse${attr_end}"
echo "Dies ist ${untersr}unterstrichen${attr_end}"
```

Weitere Informationen bieten die Manualpages von `tput` und `termcap`.

X-Hacks
Weihnachtsbasteln für Große

Ästhetisch aufgepeppte, mitunter auch gefährlich manipulierte Lichterketten kennt jeder. Es gibt aber noch eine Reihe anderer Weihnachts-Hacks – zum Lustigfinden, Inspirierenlassen, teilweise zum Aufessen und in jedem Fall zum Nachmachen.

von **Anika Kehrer**

It's the most hackable
tiiiiiiiime of the yearrrrrrr...

Parenthacks.com, Blogpost vom 6.12.2013

Abbildung 1: Die *Candy Cookie Cones* sind hiermit offiziell zum Vertreter des so genannten Food Hackings ernannt: Das Hackermaul – fünf Euro in die Wortspielkasse – wälzt schokoladig gefüllte Eiswaffeln (Bild rechts oben) in Zuckerguss und verziert sie mit allerlei, zum Beispiel mit Pulverschnee alias Kokosraspeln (Bild links). Das Ergebnis (Bild links unten) darf sich wahrlich „Zauberwald" nennen. Die Anleitung gibt es hier [1], weitere weihnachtliche Food-Hacks hier [2]. (Bilder: Mit freundlicher Genehmigung von ©Nancy, Couponclippingcook.com).

Abbildung 2: Aus Draht und Pfeifenreiniger entsteht *Charly Brown's christmas tree* (links oben), wenn jemand dem Weihnachtsfest ironisch zu Leibe rücken, aber doch nicht ganz darauf verzichten möchte (Anleitung unter [3], Bild: CC-BY-NC-SA, AZNdude bei Instructables.com). Alternativ, viel kleiner und auf elektronisch: LED-Baum (rechts oben und mittig; Anleitung unter [4], Bilder: Mit freundlicher Genehmigung von ©Chris, PyroElectro.com). Ähnlich handlich ist der 3D-gedruckte Weihnachtsbaum mit modularem Schmuck (links unten), wie gezeigt von Stefan Birghan und Jörg Weindl auf der *x.make Munich* Anfang Dezember 2013 (Bild: Anika Kehrer). Mehr Weihnachtliches gibt es bei *Thingiverse* [5], und wer keinen eigenen 3D-Drucker besitzt, kann bei den *3D Hubs* ausdrucken [6]. Doch jenen, die trotz Nadel-Ärger Wert auf echte Tanne legen, gibt ein Praxis-Hack die Möglichkeit, die Wässerung des Baumes praktisch unsichtbar zu machen. Oder wer findet im großen Bild ganz unten die Bewässerungsanlage? Na also. – Wer angesichts dieses Rätsels nicht mehr schlafen kann: Die goldene Geschenkschachtel hinter der roten enthält einen Eimer, der Wasserschlauch ist im Deko- und Tannengesumms verborgen. (Anleitung unter [7], Bild: CC-BY-NC-SA, Ricky Spears bei Instructables.com).

Abbildung 3: Mit dem Desktophintergrund *Don't hack me this christmas* ist Ruhe garantiert. Nun ja, vielleicht nicht, aber nett ist dieser Einfall der Free-BSD-Distribution *HeX, Version 1.0.2 – The Christmas Release* [8] trotzdem. (Bezugsquelle unter [9], Bild: ©HeX)

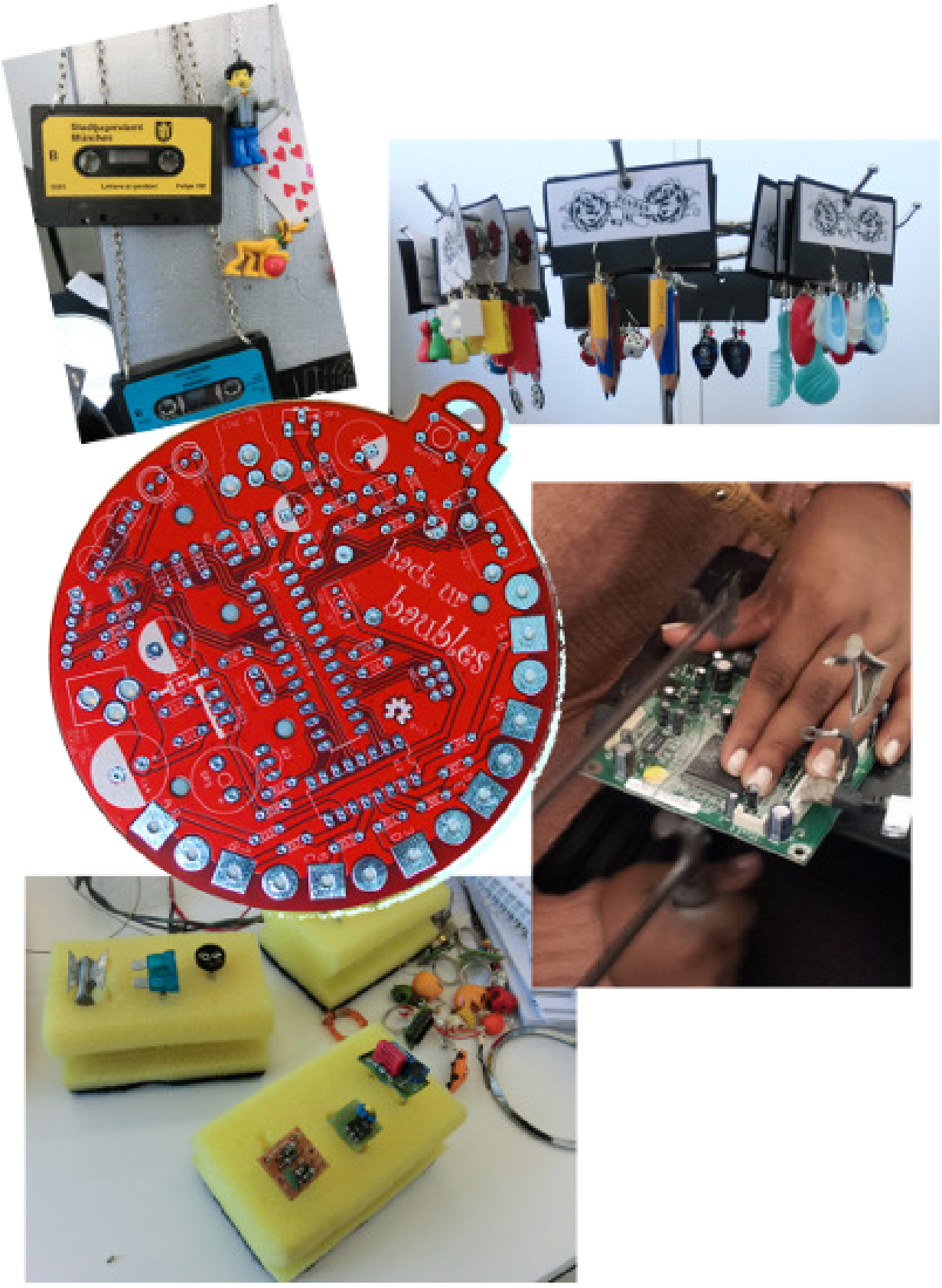


Abbildung 4: Jetzt wird es handwerklich. Blinkender, klingender Baumschmuck mit Arduino: Diese christbaumkugelförmige Platine alias *hack ur baubles* (links mittig) ist ein Controller für einzelne LEDs oder ganze Lichterketten, inklusive Audio-Ausgabe und Funkschnittstelle für Fernbedienungen (Dokumentation und Kaufmöglichkeit für rund 15 US-Dollar unter [10], Bild: Mit freundlicher Genehmigung von ©Michael Grant, Krazatchu.ca, und Megan Smith, Megansmith.ca). Wer schließlich sich selbst oder andere behängen respektive beschenken möchte, bastelt sich Ketten (links oben) oder Ohrringe (rechts oben) aus altem Kram, oder sägt Ringe aus Platinen (rechts mittig und unten). Die Beispiele wurden hergestellt und gezeigt von *Rokrokinc.com* auf der *x.make Munich* (Bilder: Anika Kehrer).

Abbildung 5: Praktisch, die Box für SD-Karten, oder (links oben)? Lässt sich 3D-ausdrucken, nicht nur zu Weihnachten. Gezeigt von Stefan Birghan und Jörg Weindl auf der *x.make Munich* Anfang Dezember 2013. 3D-Druck bringt übrigens eine Menge frischen Wind in den Modellbau: Entweder Fans von Computerspielen drucken sich ihre Lieblingswelt zum Anfassen aus (rechts; [11]), ebenfalls gezeigt von Stefan und Jörg. Oder haptisch Interessierte verbinden 3D-ausgedruckte Teile mit traditioneller Modellbaukunst, wie die Macher des Tabletop-Spiels *Lands of Ruin* (unten; [12]). (Bilder: Anika Kehrer).

Abbildung 6: Ihr wart bestimmt schon traurig. Doch auch diesmal gibt es ein Panoramabild. Ist das nicht schön? Zugegeben: Im Winter will man etwas anderes sehen, Schnee zum Beispiel. Aber wenn es schon nicht schneit, dann ist Sonnenschein die zweite Wahl – wie hier auf einer Dachterrasse der Technischen Universität München. Am besten ist natürlich beides: Schnee und Sonne. Mit diesem frommen Wunsch auf in die letzte Vorweihnachtswoche, viel Spaß beim Basteln, und danach ein paar ruhige, kuschelige Tage zwischen den Jahren! (Bild: Anika Kehrer)

Links

[1] Anleitung *Candy Cookie Cones*:
http://www.couponclippingcook.com/candy-cookie-cones/
[2] Weihnachtliche Zuckerkreationen mit Schmunzelfaktor:
http://www.nceasyfood.org/top-awesome-20-christmas-easy-food-hacks
[3] Anleitung *Charly Brown's christmas tree*:
http://www.instructables.com/id/Charlie-Browns-Christmas-Tree/?ALLSTEPS
[4] Anleitung LED-Baum:
http://www.pyroelectro.com/projects/christmas_tree_digital_hardware/
[5] Weihnachtlicher 3D-Druck und Konsorten bei *Thingiverse*:
http://www.thingiverse.com/search/page:1?q=christmas
[6] Besitzer von 3D-Druckern in der Nachbarschaft finden: http://www.3dhubs.com
[7] Anleitung für die versteckte Weihnachtsbaumbewässerung:
http://www.instructables.com/id/Make-a-Hidden-Christmas-Tree-Watering-System/?ALLSTEPS
[8] *HeX*, Version 1.0.2 – *The Christmas Release*:
http://geek00l.blogspot.de/2007/12/hex-102-christmas-release.html
[9] Bezugsquelle für HeX-Christmas-Wallpaper:
http://wallpaperstock.net/dont-hack-this-christmas_wallpapers_16407_1280x1024_1.html
[10] Controller-Platine *hack ur baubles*:
http://krazatchu.ca/2013/11/25/hack-ur-baubles-assembly-instructions/
[11] Stefan Birghans Blog zu 3D-Drucken aus der Welt von *Dawn of War*:
http://warforge.de
[12] Tabletop-Spiel *Lands of Ruin*: http://www.landsofruin.com

Hilfreiches für alle Beteiligten
Autorenrichtlinien

Selbst etwas für die Uptimes schreiben? Gern! Als Thema ist willkommen, was ein GUUG-Mitglied interessiert und im Themenbereich der GUUG liegt. Was sonst noch zu beachten ist, steht in diesen Autorenrichtlinien.

Der Schriftsteller ragt zu den Sternen empor,
Mit ausgefranstem T-Shirt.
Er raunt seiner Zeit ihre Wonnen ins Ohr,
Mit ausgefranstem T-Shirt.

Frei nach Frank Wedekind, Die Schriftstellerhymne

Wir sind an Beiträgen interessiert. Wir, das ist diejenige Gruppe innerhalb der GUUG, die dafür sorgt, dass die Uptimes entsteht. Dieser Prozess steht jedem GUUG-Mitglied offen. Der Ort dafür ist die Mailingliste <redaktion@uptimes.de>.

Welche Themen und Beitragsarten kann ich einsenden?

Die Uptimes richtet sich als Vereinszeitschrift der GUUG an Leser, die sich meistens beruflich mit Computernetzwerken, IT-Sicherheit, Unix-Systemadministration und artverwandten Themen auseinandersetzen. Technologische Diskussionen, Methodenbeschreibungen und Einführungen in neue Themen sind für dieses Zielpublikum interessant, Basiswissen im Stil von *Einführung in die Bourne Shell* hingegen eher nicht. Wer sich nicht sicher ist, ob sein Thema für die Uptimes von Interesse ist, kann uns gern eine E-Mail an <redaktion@uptimes.de> schicken.

Neben Fachbeiträgen sind Berichte aus dem Vereinsleben, Buchrezensionen, Konferenzberichte, humoristische Formen und natürlich Leserbriefe interessant. Wer nicht gleich mehrseitige Artikel schreiben möchte, beginnt also mit einem kleineren Beitrag.

Fachbeiträge sind sachbezogen, verwenden fachsprachliches Vokabular und anspruchsvolle Erläuterungen, besitzen technische Tiefe und ggf. auch Exkurse. Berichte aus dem Vereinsleben greifen aktuelle Themen auf oder legen Gedankengänge rund um die GUUG und ihre Community dar. Konferenzberichte zeigen, welche Veranstaltungen jemand besucht hat, was er/sie dort erfahren hat und ob die Veranstaltung nach Meinung des Autoren beachtenswert oder verzichtbar war. Unterhaltsame Formen können ein Essay oder eine Glosse sein, aber auch Mischformen mit Fachartikeln (Beispiel: der "Winter-Krimiïn Ausgabe 2013-3). Auch unterhaltsame Formen besitzen jedoch inhaltlichen Anspruch. Denn die Uptimes ist und bleibt die Mitgliederzeitschrift eines Fachvereines.

In der Uptimes legen wir daher auch Wert auf professionelle publizistische Gepflogenheiten und einheitliche Schreibweisen. Dafür sorgt zum Beispiel ein einheitliches Layout der Artikel, oder etwa die grundsätzliche Vermeidung von Worten in Großbuchstaben (entspricht typografisches Schreien) oder von Worten in Anführungsstrichen zum Zeichen der Uneigentlichkeit (entspricht Distanzierung von den eigenen Worten). Wichtig sind außerdem beispielsweise Quellenangaben bei Zitaten, Kenntlichmachung fremder Gedanken, Nachvollziehbarkeit der Argumentation sowie Informationen zum Autor nach dem Artikel.

In welchem Format soll ich meinen Artikel einsenden?

ASCII: Am liebsten blanke UTF8-Texte.

LaTeX: Wir setzen die Uptimes mit LaTeX. Weil wir – wie es sich beim Publizieren gehört – mehrspaltig setzen und ein homogenes Erscheinungsbild anstreben, verwenden wir für die Uptimes bestimmte Formatierungen. Es ist nicht erwünscht, eigene Layoutanweisungen einzusenden. Wir behalten uns vor, Texte für die Veröffentlichung in der Uptimes umzuformatieren. Eine Vorlage mit den von uns verwendeten Auszeichnungen für Tabellen, Kästen und Abbildungen gibt es unter <redaktion@uptimes.de>.

Listings: Der mehrspaltige Druck erlaubt maximal 45 Zeichen Breite für Code-Beispiele, inklusive 1 Leerzeichen und einem Zeichen für den Zeilenumbruch innerhalb einer Code-Zeile (Backslash). Breitere Listings formatieren wir um, verkleinern

die Schriftgröße oder setzen sie als separate Abbildung.

Bilder: Wir verarbeiten gängige Bildformate, soweit ImageMagick sie verdaut und sie hochauflösend sind. Am besten eignen sich PNG- oder PDF-Bilddateien. Plant bei längeren Artikeln mit 1 Abbildung pro 3000 Zeichen. Das müssen nicht Bilder sein, sondern auch Tabellen, Listings oder ein Exkurskasten sind möglich. Verseht Eure Bilder nicht mit Rahmen oder Verzierungen, weil die Redaktion diese im Uptimes-Stil selbst vornimmt.

Wie lang kann mein Artikel sein?

Ein einseitiger Artikel hat mit zwei Zwischentiteln um die 2.700 Anschläge. Mit etwa 15.000 Anschlägen – inklusive 3 Abbildungen – landet man auf rund vier Seiten. Wir nehmen gern auch achtseitige Artikel, achten dabei aber darauf, dass der Zusammenhang erhalten bleibt und dass es genug Bilder gibt, damit keine Textwüsten enstehen.

Wer Interesse hat, für die Uptimes zu schreiben, macht sich am besten um die Zeichenzahl nicht so viele Gedanken – auch für kurze oder lange Formate finden wir einen Platz. Die Redaktion ist bei der konkreten Ideenentwicklung gern behilflich. Für eine Artikelidee an <redaktion@uptimes.de> reicht es, wenn Ihr ein bestimmtes Thema behandeln wollt.

Wohin mit meinem Manuskript?

Am einfachsten per E-Mail an <redaktion@uptimes.de> schicken. Das ist jederzeit möglich, spätestens jedoch vier Wochen vor dem Erscheinen der nächsten Uptimes. Zum Manuskript ist ein kleiner Infotext zum Autor wichtig, ein Bild wünschenswert.

Nützlich ist, wenn der Text vor Einsendung durch eine Rechtschreibkorrektur gelaufen ist. `aspell`, `ispell` oder `flyspell` für Textdateien sowie die von LibreOffice bieten sich an. Wenn Ihr Euren Text an die Redaktion schickt, solltet Ihr also weitestmöglich bereits auf die Rechtschreibung geachtet haben: Nach der Einschickung ist Rechtschreibung und Typo-Korrektur Aufgabe der Redaktion. Die Texte in der Uptimes folgen der neuen deutschen Rechtschreibung.

Wie verlaufen Redaktion und Satz?

Wir behalten uns vor, Texte für die Veröffentlichung in der Uptimes zu kürzen und zu redigieren. Das bedeutet, dafür zu sorgen, dass der Artikel nicht ausufert, versehentliche Leeraussagen wegfallen, Syntax und Satzanschlüsse geglättet werden, dass Passiva und Substantivierungen verringert und Unklarheiten beseitigt werden (die zum Beispiel Fragen offen lassen oder aus Passivkonstruktionen resultieren, ohne dass der Schreibende das merkt). Manchmal ist dieser Prozess mit Nachfragen an den Autoren verbunden.

Die endgültige Textversion geht jedem Autoren am Ende zur Kontrolle zu. Dabei geht es um die inhaltliche Kontrolle, ob sich durch den Redaktionsprozess Missverständnisse oder Falschaussagen entstanden sind. Danach setzt die Redaktion die Artikel. Wenn der Satz weitgehend gediehen ist – also ein *Release Candidate* als PDF vorliegt – erhalten die Autoren als erste diesen RC. Danach wird die Uptimes dann veröffentlicht.

Gibt es Rechtliches zu beachten?

Die Inhalte der Uptimes stehen ab Veröffentlichung unter der CC-BY-SA-Lizenz, damit jeder Leser die Artikel und Bilder bei Nennung der Quelle weiterverbreiten und auch weiterverarbeiten darf. Bei allen eingereichten Manuskripten gehen wir davon aus, dass der Autor sie selbst geschrieben hat und der Uptimes ein nichtexklusives, aber zeitlich und räumlich unbegrenztes Nutzungs- und Bearbeitungsrecht unter der CC-BY-SA einräumt.

Bei Fotos oder Abbildungen Dritter ist es rechtlich unabdingbar, dass der Autor sich bei dem Urheber die Erlaubnis zu dieser Nutzung einholt, und fragt, wie die Quelle genannt zu werden wünscht. Die Frage nach der CC-BY-SA ist hierbei besonders wichtig.

An Exklusivrechten, wie sie bei kommerziellen Fachzeitschriften üblich sind, hat die Uptimes kein Interesse. Es ist den Autoren freigestellt, ihre Artikel noch anderweitig nach Belieben zu veröffentlichen.

Bekomme ich ein Autorenhonorar?

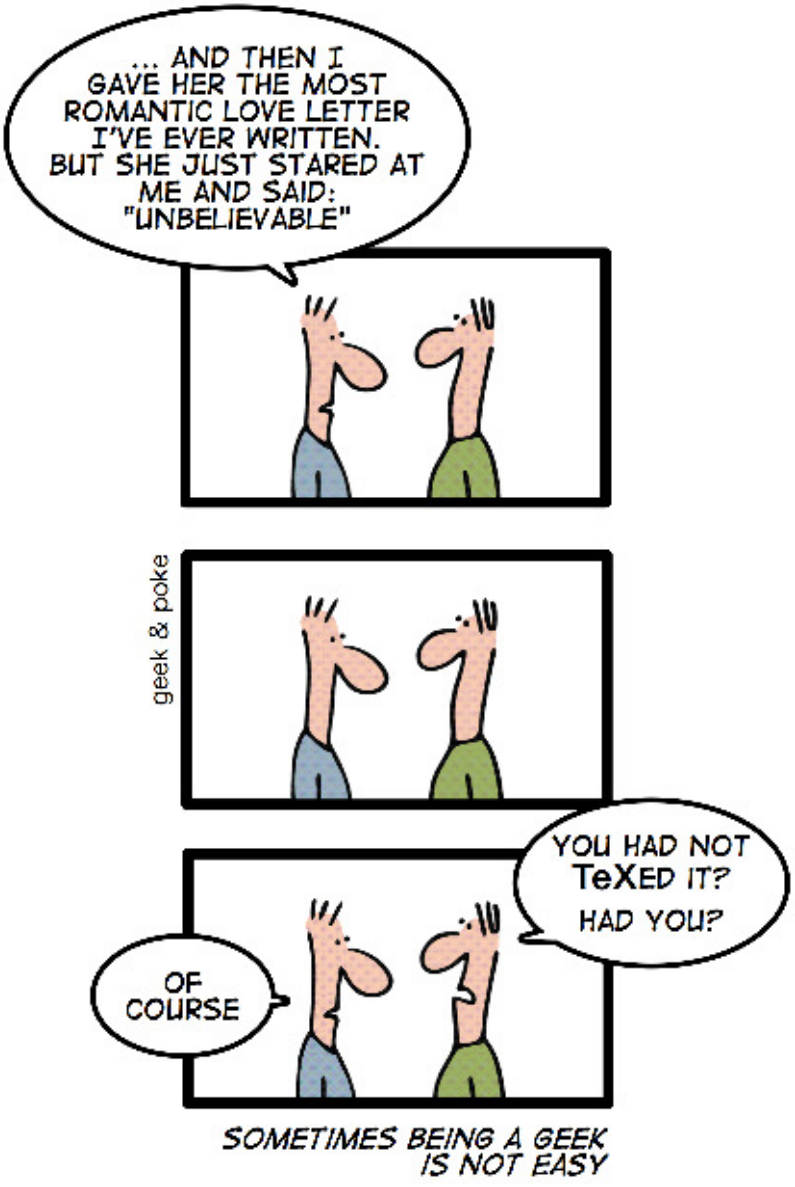
Für Fach- und literarische Beiträge zahlt die GUUG dem Autor nach Aufforderung durch die Redaktion und Rechnungstellung durch den Autor pro Seite 50 € zuzüglich eventuell anfallender USt. Beiträge für die Rubrik „Vereinsleben“, Buchrezensionen und Artikel bezahlter Redakteure sind davon ausgenommen. Gleiches gilt für Paper, wenn die Uptimes die Proceedings der Konferenz enthält.

Nächste Ausgabe (digital): Uptimes 2014-1, Frühlings-Ausgabe

- Redaktionsschluss: Sonntag, 23. März 2014.
- Erscheinung: Sonntag, 27. April 2014.
- Gesuchte Inhalte: Fachbeiträge über Unix und verwandte Themen, Veranstaltungsberichte, Rezensionen, Beiträge aus dem Vereinsleben.
- Fragen, Artikelideen und Manuskripte an: <kehrer@guug.de> oder an die Mailingliste <redaktion@uptimes.de>

Nächste Ausgabe (analog): Uptimes 2014-2, Proceedings-Ausgabe

- Redaktionsschluss: TBD.
- Erscheinung: TBD.
- Gesuchte Inhalte: Paper zum Frühjahrsfachgespräch 2014.
- Manuskripte an: Mailingliste <redaktion@uptimes.de>



Über die GUUG
German Unix User Group e.V.

Vereinigung deutscher Unix-Benutzer

Die Vereinigung Deutscher Unix-Benutzer hat gegenwärtig rund 700 Mitglieder, davon etwa 90 Firmen und Institutionen.

Im Mittelpunkt der Aktivitäten der GUUG stehen Konferenzen. Ein großes viertägiges Event der GUUG hat eine besondere Tradition und fachliche Bedeutung: In der ersten Jahreshälfte treffen sich diejenigen, die ihren beruflichen Schwerpunkt im Bereich der IT-Sicherheit, der System- oder Netzwerkadministration haben, beim *GUUG-Frühjahrsfachgespräch* (FFG).

Seit Oktober 2002 erscheint mit der *Uptimes* – die Sie gerade lesen – eine Vereinszeitung. Seit 2012 erscheint die Uptimes einerseits zu jedem FFG in Form einer gedruckten Proceedings-Ausgabe (ISBN), und andererseits im Rest des Jahres als digitale Redaktionsausgabe (ISSN). Daneben erhalten GUUG-Mitglieder zur Zeit die Zeitschrift *LANline* aus dem Konradin-Verlag kostenlos im Rahmen ihrer Mitgliedschaft.

Schließlich gibt es noch eine Reihe regionaler Treffen (http://www.guug.de/lokal): im Rhein-Ruhr- und im Rhein-Main-Gebiet sowie in Berlin, Hamburg, Karlsruhe und München.

Warum GUUG-Mitglied werden?

Die GUUG setzt sich für eine lebendige und professionelle Weiterentwicklung im Open Source-Bereich und für alle Belange der System-, Netzwerkadministration und IT-Sicherheit ein. Wir freuen uns besonders über diejenigen, die bereit sind, sich aktiv in der GUUG zu engagieren. Da die Mitgliedschaft mit jährlichen Kosten

Fördermitglied	350 €
persönliches Mitglied	90 €
in der Ausbildung	30 €

verbunden ist, stellt sich die Frage, welche Vorteile damit verbunden sind?

Neben der Unterstützung der erwähnten Ziele der GUUG profitieren Mitglieder auch finanziell davon, insbesondere durch die ermäßigten Gebühren bei den Konferenzen der GUUG und denen anderer europäischer UUGs. Mitglieder bekommen außerdem *c't* und *iX* zum reduzierten Abopreis.

Wie GUUG-Mitglied werden?

Füllen Sie einfach das umseitige Anmeldeformular aus und schicken Sie es per Fax oder Post an die unten auf dem Formular angegebene Adresse. Falls Sie die Seite nicht herausreißen wollen: Sie können den Mitgliedsantrag als PDF herunterladen, siehe URL auf dem Mitgliedsantrag.

Impressum

Uptimes – Mitgliederzeitschrift der
German Unix User Group (GUUG) e.V.
Herausgeber: GUUG e.V.
Bruno-Walter-Ring 30
D-81927 München
Tel.: +49-(89)-380 125 95 0
Fax: +49-(89)-380 125 95 9
E-Mail: <redaktion@uptimes.de>
Internet: http://www.guug.de/uptimes/

Autoren dieser Ausgabe: Andreas Lohrum, Jürgen Plate, Andreas Bunten, Torsten Voss, Wolfgang Stief, Anika Kehrer, Stefan Schumacher.
V.i.S.d.P: Wolfgang Stief, Vorstandsvorsitzender, Anschrift siehe Herausgeber
Chefredaktion: Anika Kehrer
LaTeX-Layout (PDF): Robin Schröder
XHTML-Layout (ePub): Mathias Weidner
Titelbild: Beltsville Agricultural Research Center, Brian0918 (PD)
http://commons.wikimedia.org/wiki/File:Snow_crystals_2b.png
Titelgestaltung: Hella Breitkopf
Bildnachweis: Comicreihe *geek & poke* CC-by-sa, mit freundlicher Genehmigung von Oliver Widder. Andere Quellennachweise am jeweiligen Bild.
Verlag: Lehmanns Media GmbH, Hardenbergstraße 5, 10623 Berlin
ISSN: 2195-0016

Wenn Sie Interesse an Anzeigen in der Uptimes haben,
wenden Sie sich bitte an <werbung@guug.de>.