KPMG Klynveld EDP Audit

89/1
lente 1989

COMPACT

Uit de inhoud

Het uitvoeren van een transactie-analyse, door M.C. Duym

Software escrow, door R.A. s'Jacob

Computervirussen. Worm in groot netwerk, door drs.ing. J.C. van Winkel

Beheersaspecten bij gebruik van microcomputers, door J.F.C. van Epen

The IBM AS/400. A concern to the EDP Auditor?, door H.J. Lijnes

AS/400 security, door Mw. Verena Six

Internationale gegevensstromen: abstract en moeilijk te controleren, door mr. V.A. de Pous

Computer en accountant

Inhoudsopgave

- Van de Redactie 1
- Actualiteit 5
  NIvRA. Studie Rapport 21 5
- K-Memory 6
  1988 no. 1 6
  1988 no. 2 9
- Het uitvoeren van een transactie-analyse 13
  door: M.C. Duym
- Software escrow 23
  door: R.A. s'Jacob
- Computervirussen. Worm in groot netwerk 29
  door: drs.ing. J.C. van Winkel
- Lezers reageren 34
  Hoofdstuk 6 "Beveiligen tegen computermisbruik" 34
  door: A.W. Neisingh RA en drs. J. Vossen RA
- Beheersaspecten bij gebruik van microcomputers 36
  door: J.F.C. van Epen
- The IBM AS/400. A concern to the EDP Auditor? 46
  door: H.J. Lijnes
- AS/400 security 50
  door: Mw. Verena Six
- Internationale gegevensstromen: abstract en moeilijk te controleren 53
  door: mr. V.A. de Pous

Rubrieken

- De micro in de accountantscontrole 56
- Boeken 59
  -"Practical project management (restoring quality to DP projects and systems)" 59
  -"Doorbelasting van kosten van geautomatiseerde informatievoorziening" 63
- ABC Nieuws 65
- Overzicht van de cursussen Automatisering en Controle van KPMG Klynveld EDP Audit 67
- Overzicht hoofdartikelen 1987/1988 (nummers 42 t/m 47) (Special) 68

Nummer 48
ISSN 0920-1645

Lente 1989

Van de Redactie

Na het special van de sectie Software Engineering (Compact 88/2 no. 47) is het stil geworden in de communicatie met u, geachte lezer, maar dat is slechts ogenschijnlijk zo.
Het aantal onderwerpen, waarmee KPMG Klynveld EDP Audit bezig is, geeft een beeld van de omvang van de problematiek ter zake van automatisering en controle.
Het nummer dat nu voor u ligt, weerspiegelt enigszins de huidige situatie.

Per artikel staat aangegeven, door middel van een waarderings"raampje", hoe actueel, diepgaand en/of educatief een artikel is. Het geeft geen oordeel over de schrijver. Na de hoofdartikelen vindt u de gebruikelijke rubrieken. Wij vragen uw speciale aandacht voor de rubriek Onderwijs, handelend over de nieuwe cursusbrochure 1989.

Onder de rubriek "Lezers reageren" vindt u het ontbrekende hoofdstuk 6 uit het artikel "Beveiligen tegen computermisbruik", gepubliceerd in Compact 87/3. Onze redactie is dankbaar voor de reactie uit de lezerskring.

Wij nodigen u uit om gebruik te maken van de mogelijkheid om in ons blad te publiceren en/of te reageren op geplaatste artikelen.

Wij wensen u veel leesgenot met onze eerste aflevering in 1989.

De Redactie.

Hieronder volgt een samenvatting van de inhoud van de hoofdartikelen.

Het uitvoeren van een transactie-analyse
door: M.C. Duym

laag			hoog	
			x	actueel
		x		diepgaand
			x	educatief

Transactie-analyse is een techniek om de bevoegdheden en verantwoordelijkheden ten aanzien van bestanden of programma's te analyseren ten behoeve van het verkrijgen van een uitspraak over het bestaan van een zodanig stelsel van logische toegangscontroles, dat daarmee de betrouwbaarheid van de in de betrokken bestanden opgenomen gegevens of van de in de programma's opgenomen verwerkingsfuncties, beter kan worden gewaarborgd.

Software escrow
door: R.A. s'Jacob

laag			hoog	
			x	actueel
		x		diepgaand
			x	educatief

Ter verzekering van de continuïteit van een geautomatiseerd proces kan een onderneming een aantal maatregelen treffen. De meeste van deze maatregelen zijn gericht op de beschikbaarheid van de hardware en de gegevens. Maatregelen gericht op de beschikbaarheid van software zijn nog relatief onbesproken. Eén van deze maatregelen betreft het afsluiten van een escrow-overeenkomst, inhoudende het deponeren onder bepaalde voorwaarden van de source-code van software en daaraan gerelateerde zaken.

Computervirussen. Worm in groot netwerk
door: J.C. van Winkel

laag			hoog	
			x	actueel
	x			diepgaand
			x	educatief

Enige tijd geleden heeft er in het grote ARPA-Net (Netwerk in de U.S.A.) een worm rondgewaard die duizenden computers heeft aangedaan (zie bijgaand bericht uit NYTIMES News Service, 8 november 1988). Het programma wist zich via het ARPA-net te verspreiden door een aantal "gaten" in de beveiliging van de UNIX-computers die op ARPA-net zijn aangesloten. Er zijn geen bestanden beschadigd, de enige schade die geleden is, is het verlies van werkuren.

Beheersaspecten bij gebruik microcomputer
door: J.F.C. van Epen

laag			hoog	
			x	actueel
		x		diepgaand
			x	educatief

Invoering van micro- of "personal"-computers brengt enkele organisatorische gevolgen met zich mee, die op vele punten afwijken van het gebruik van centrale automatisering. Soms ook zullen bepaalde maatregelen gelijk blijven, zij het dat deze nu tot de taken van de gebruiker van de microcomputer zijn gaan behoren in plaats van tot die van de (centrale) Automatiseringsafdeling. Het artikel gaat niet in op het besturingssysteem van de Apple-computers gezien de afwijkende filosofie achter dit besturingssysteem.

The IBM AS/400. A concern to the EDP Auditor
door: H.J. Lijnes
AS/400 Security
door: Mw. Verena Six

laag			hoog	
			x	actueel
			x	diepgaand
			x	educatief

De AS/400.
Belichting vanuit de organisatie-adviespraktijk van KPMG Klynveld Bosboom Hegener met ruime raakvlakken met EDP Audit. Daarna volgt een zeer gerichte visie op de security vanuit de EDP Audit-praktijk. Beide artikelen zijn zeer lezenswaard.

Internationale gegevensstromen: abstract en moeilijk te controleren

laag			hoog	
			x	actueel
		x		diepgaand
		x		educatief

Hoewel het voor veel mensen grotendeels abstract is, groeit de belangstelling voor Transborder Data Flows (TDF), juister gezegd, voor de regulering ervan. Hierbij gaat het om de grensoverschrijdende gegevensstromen en de feitelijke, technische en juridische beperking ervan. In mei 1987 organiseerde CELIM - een club van Europese computerrechtdeskundigen - een tweedaagse conferentie in Brussel over TDF en het recht van de Europese Gemeenschappen: "Freedom of data flows and EEC law", of voor de Franstaligen "Liberté des flux de données et droit communautaire".

door: mr. V.A. de Pous

Mr. V.A. de Pous houdt zich bezig met advies en informatieverzorging inzake juridische aspecten van de informatietechnologie en is onder meer uitgever/redacteur van de maandelijkse nieuwsbrief "News Ware".
Hij schrijft onder eigen verantwoordelijkheid, die de redactie van Compact ook niet kan overnemen. Het artikel is geheel voor verantwoordelijkheid van de auteur. Sinds 1987 heeft het artikel niet aan actualiteit ingeboet.

Compact (R) is een uitgave van

KPMG Klynveld EDP Audit

Deze informatie is in de eerste plaats bestemd voor diegenen, die in de algemene controlepraktijk werkzaam zijn van KPMG Klynveld Kraayenhof & Co. De in dit tijdschrift weergegeven meningen mogen niet altijd gezien worden als officiële zienswijze van KPMG Klynveld EDP Audit. De in rubrieken besproken tijdschriften, boeken en artikelen worden soms geheel opgenomen of verkort aangehaald, tevens als regel voorzien van commentaar.

Redactie:
A.W. Neisingh
D. Steeman
P. Veltman
H.J.M. van der Wielen (secr.)

Kopij kunt u inleveren bij de
secretaris van de redactie

Adres:
World Trade Center Amsterdam
Strawinskylaan 1257
Toren D 11e etage
1077 XX Amsterdam

Postadres:
Postbus 72001
1007 TB Amsterdam

ISSN 0920-1645

Indien u belangstelling heeft voor meerdere exemplaren kunt u deze aanvragen bij de secretaris van de redactie, evenwel zolang de voorraad strekt (telefoon 020 - 5461912).

Actualiteit

Studierapport no. 21

Op advies van het College van Beroepsvraagstukken (CBV) heeft het bestuur van het Nederlands Instituut van Registeraccountants (NIvRA) in november 1988 ingestemd met de publikatie van Studierapport no. 21 "Functiescheidingen in Software".
Dit rapport is vervaardigd onder auspiciën van de Commissie van Advies inzake Automatiserings-vraagstukken (CAV) door haar Werkgroep Functiescheidingen in Software (FUSO). Deze werkgroep is in de loop van 1985 gestart en had de volgende samenstelling:
- J.J.A. Leenaars RA (voorzitter), lid Groepsdirectie Robeco-groep;
- A.F.H. Aarts RA, hoofd afdeling Informatiebeleid Ministerie van Defensie;
- drs.S. van Duin RA (secretaris), research medewerker NIvRA;
- R.J.M. van der Horst RA, directeur Concernaccountantsdienst Algemene Bank Nederland;
- A.J. Kruseman, hoofd Planning en Systemen Delta Lloyd NV;
- H.B. Moonen RA, vennoot KPMG Klynveld EDP Audit.

Het rapport "Functiescheidingen in software" behandelt de wijze waarop functiescheidingen in een vergaand geautomatiseerde omgeving kunnen worden gerealiseerd. In plaats van respectievelijk in aanvulling op de klassieke benadering van functiescheiding (gericht op personen), introduceert dit rapport een alternatieve vorm van functiescheiding, opgenomen in applicatie-software.
Het bestuur van het NIvRA meent dat de originele beschouwingen die in dit rapport worden gewijd aan één van de elementaire bouwstenen van de accountantscontrole, van grote waarde kunnen zijn voor de benadering van de controle in een (vergaand) geautomatiseerde omgeving.
Eventuele reacties op dit studierapport ziet het bestuur gaarne tegemoet.

Tot zover het voorwoord bij dit rapport.

Inhoudsopgave

1. Inleiding

2. Functiescheiding in historisch perspectief
   - 2.1 Inleiding
   - 2.2 De betekenis van de functiescheiding voor de accountantscontrole
   - 2.3 Functiescheiding en automatisering

3. Functiescheiding in een vergaand geautomatiseerde omgeving
   - 3.1 Inleiding
   - 3.2 Automatisering: wijzigingen?

4. Functiescheiding tussen menselijke en geprogrammeerde functies
   - 4.1 Leiding
   - 4.2 Toegangscontrolemechanismen
   - 4.3 De functieverdeelstaat
   - 4.4 Transactie-impactanalyse

5. Enige specifieke aspecten van accountantscontrole

6. De invloed van systeem-software

7. Slotwoord

K-Memory

no. 1 van juni 1988

K-Memory is een door KPMG uitgegeven kwartaalschrift, bedoeld om cliënten te informeren. Per juni 1988 is K-Memory magazine in een nieuwe vorm verschenen, reden waarom we het Voorwoord van het "Editorial Office" in Compact overnemen. Mocht u het blad willen ontvangen, meldt dat dan aan de Redactie van Compact.

Dear Reader,

This is the first edition of our new K-Memory magazine. Like you, I receive and read many magazines each year. In launching this one, I am very conscious of how little time we all have for reading, and how important it is to be able to sort our information that we ourselves really need from the information that other people would like us to have!

K-Memory will be mailed to around 10,000 KPMG clients world wide; that gives us some focus in terms of common interest, but also the widest range of clients one could imagine. Well, almost!

We will be focusing clearly on computer auditing and related topics. To make it easier for you to find out exactly what is in each issue, we have included an informative index. My personal feeling is that an index should direct you not just to articles that are of obvious relevance to your business or to your work, but to those articles that will bring you a new line of thought or inspiration. For example, John Sculley's insight on the means of harnessing computer power and the short article on the problem of transmitting data across borders are both of interest to most chief executives.

There are a number of themes that K-Memory will cover, including:
- Innovative approaches to auditing and computer auditing, particularly using micro-based software. Tremendous opportunities presently exist for KPMG and its clients in this area. Although the common experience has been that computers take longer to bring benefits than we all would like, it is my view that the microcomputer revolution has now been going on long enough for us to begin to see substantial benefits accruing in practice.
- Often we don't know whether to sing the praises of our clients or our staff; both are vital ingredients of our success. By critically examining success stories of both, we can identify key areas where failure must be guarded against. A good success story should tell us all not just how right things went, but also where things could have gone seriously wrong and how that was prevented.

We'll also be covering specific countries of the world and the KPMG operations therein, as well as KPMG products and examples of their application in assisting clients.

In presenting KPMG's views about present and future trends in data processing and information technology, we will also be measuring how they stack up against the views of major manufacturers and vendors of products. I hope that we will be seeing a number of contributions to K-Memory from people around the world who are recognised as leaders in the areas with which we are concerned.

Finally, I'd like to point out that just by being a client of KPMG, you have already provided a useful influence. I hope that you will feel that you personally will benefit from reading, and on occasion contributing directly to K-Memory.

Geoff Russel Grant,
Editor.

De inhoud van het eerste herziene nummer luidt als volgt:
1. John Sculley of Apple talks about the paradox of power and ease of use;

2. other features;
   2.1 Country focus on the UK;
   2.2 System/38 an auditor's dilemma?;
   2.3 The best in audit automation;
   2.4 Crossing international barriers - transborder data flow.

Uit de inhoud het volgende:

1 The paradox of power and ease of use by Mr John Sculley

'The joint development agreement signed by Apple Computer and Digital Equipment Corporation provides a unique way to address the demand for both power and ease of use. Instead of trying to create a single system that has both characteristics, we are developing tools to connect our existing systems, which already have these strenghts.'
'The third party development community saw the possibilities of a Macintosh/VAX combination at once. With a consistent front end that was easy to learn and use, they saw that it would be possible to bring the processing power of the VAX and the world wide networking power of DECnet within easy reach of large numbers of non-specialists.'

2 Other features

2.1 Country focus on the United Kingdom

The topics of this essay are

a. All auditors need a certain degree of knowledge of computers. As new, user-friendly systems are developed they will find it easier. But a step beyond that, there is a need for specialist computer auditors. Trainees are hand-picked: all are qualified accountants but still have to sit logical aptitude tests before entry to computer audit as part of an 18-month specialist training programme. After that each specialist spends 12 to 18 months on reviews of systems and datacentres, as well as using audit software, working up to being in charge of all computer audit work for major clients and graduating to highly specialised advisory work in the control and security area.

b. We also do financial modelling for our clients. A large company might have the internal resources to do its own financial modelling, but many do not. Even if they have the capability, they have only a limited number of people, who have other things to do. If they don't develop models routinely, it takes too long thinking about it, experimenting, working out how to do it. We can set up models very quickly!
   KPMG in Birmingham also trains and works with client's internal auditors, particularly in the public sector, where they have a growing practice.

c. Another area Bill Holden identifies as growing is concern with computer security. Just as the London financial institutions are acutely aware of how much their business depends on their computers, so the senior executive in manufacturing is rapidly becoming aware of how dependent he is on his computer systems. A computer breakdown may cause a sudden dive in the sales graph, but what happens to his stock availability records, his debtor cash flow or production control?
   Contingency planning, or disaster plans are one of the first things KPMG look for in a Security Improvement Review. Deciding what to do if the boiler bursts and the computer room is flooded is more mundane than chasing viruses through an operating system, but it is just as necessary.

Met deze plastische schildering vragen de schrijvers aandacht voor de remedie tegen deze onvolkomenheden in uw organisatie. In ons blad komen bedoelde beveiligingsmaatregelen op basis van een integraal beveiligingsplan ook aan de orde. Lees het komende Compactnummer dat ijs en

weder dienende aan dit onderwerp zal zijn gewijd. KPMG Klynveld EDP Audit verricht op dit gebied, al dan niet in samenwerking met specialisten van andere disciplines, bijzondere onderzoeken.

2.2 System/38 - an auditor's dilemma?

The security policy should incorporate the following:
- the responsibility of all personnel for information security;
- the role and responsibilities of the security officer;
- the classification by sensitivity of object;
- the levels of security to be established over each object;
- the role of internal audit in monitoring compliance with security procedures

The security policy has two main functions.

Firstly, it creates an environment in which security mechanisms can function effectively and secondly it provides a framework upon which the DP department, for example, can build and configure its security systems.

Each user of the System/38 must have a user profile which describes them. The user profile contains the name of the profile, a list of objects which are under complete control of the user (owned by the user) and a list of objects which other users have granted them access to.

It is a pity that in practice so few System/38's reach their full security potential.

Many managements are now taking a new look at their security arrangements - a few of them are unfortunately doing this retrospectively, following serious violations.
KPMG computer auditors find the System/38 an interesting machine to work with and with its security potential and the powerful combination of IBM's Query language, and KPMG's System 2190 audit software available, accessing the data is also easy.

Tot zover citaten uit het artikel. Het geheel is lezenswaard.

2.3 The best in audit automation

At the leading edge 'we are taking giant steps forward in audit automation services', says Herman Roos. We have standardised our hardware and software on a global basis. The Apple Macintosh was chosen as superior in speed, memory capacity, practicality and above all user friendliness. Recent announcements, including the co-operation between Apple and DEC, underpin the wisdom of this choice.

'The user friendliness of computer hardware and software in an accounting environment is extremely important', says Casper Broeksma. In the future we are talking about there being between 40,000 - 50,000 professionals using micro's. KPMG audit staff around the world now choose from a wide variety of fast and effective audit software for every engagement. This is largely thanks to the two software development centres (one in Montvale New Yersey, the other in Amsterdam).
Just as the facility of an application varies according to the expertise of the user, so does the extent of which more sophisticated or specialised products are developed. Herman Roos sums it up. Anything is possible. It will all depend on the knowledge and skills of both the generalist and the specialist.

2.4 Crossing international barriers - transborder data flow

Recognising the need to clarify the situation, KPMG has published a survey on legal aspects transborder flow of personal data. For further information, contact your local KPMG partner.

K-Memory

no. 2 van november 1988

Dear Reader,

Welcome to the second issue of K-Memory magazine.

Feedback from the first issue shows that K-Memory addresses an audience that has never been served by such a publication before. Anyone responsible for the direction of auditing and/or control of computerised systems receives many publications on the technical aspects of various systems. However, to gain perspective, we need to stand back and take an overall view of matters, looking to the future in order to forecast the direction of the technology, and focusing on the critical longer term strategies as well as on the technical issues of the day.

We pick up on that theme in this issue, as guest contributor Bill Gates of Microsoft predicts how advances in microcomputer operating systems will enable businesses to achieve productivity benefits through automation of the office and integration of varied applications. It all sounds so very easy, yet only five years ago there were no graphical user interfaces and even the idea of transferring data between applications sounded quite impractical unless the applications came specifically designed that way from the same software house. I sometimes think that it is surprising how complex some of our straightforward office tasks are, and by contrast how simple are some of the processes which we humans find difficult. I am reminded of a quotation printed on the cover of a major multinational client's IT standards manual:
'That computers will one day learn to think like humans is less disturbing than that humans are already showing a tendency to act like computer.'

Man-machine comparisons are emotive, which leads me into the article on software viruses, which is another interesting topic. There has been much publicity about viruses, mainly microcomputer based, the damage caused and how the virus was eradicated. I have a virus detector installed on my own micro, although I must admit that I dit not volunteer my machine for testing. This seems to me to be an area where practical experience is worth more than theory, so if any readers have dealt with viruses, why not tell K-Memory? Names will not be disclosed, but send hard copy, not diskette, please!

Our series on popular computers continues with the DEC VAX, looking in depth at access security. We also take a quick look at IBM's latest mid-range offering, the AS/400, which could well have a significant installed base world-wide within a very short time period. Harry Lijnes summarises his initial reaction from an audit and control standpoint.

Combining topicality with hard fact, another regular K-Memory feature covers KPMG's operation in Australia in that continent's bicentennial year. Australia is a long way from the UK, yet I see my partner John Brown from Sydney every two or three months, keeping up a contact that has lasted over 15 years. We exchange ideas, information and staff, and many of our clients do the same.

I hope that you will find our second edition of K-Memory useful. I thank you for your interest in our publication and look forward to receiving your comments, or contributions.

Geoff Russell Grant
Editor.
De redactie van Compact koos voor u het artikel van de hand van Diana Kenning: DEC VAX-access security.

Dec VAX

Users of Digital Equipment Corporation's (DEC) VAX system have increasingly demanded improved security over access. In the following article, we discuss the questions you should ask to ensure your VAX system is well-protected.

The power, cost-effectiveness and particularly the flexibility of DEC VAX systems are the main reasons for their increasing popularity world-wide. Their versatility ranges from small office-based systems to large mainframes and vast multi-user networks. Many large commercial organisations (Shell, Reuters), financial institutions (the Bank of England, Credit Lyonnais) and government departments have installed complex VAX networks and use a VAX for specific applications. VAX is also often chosen for Computer Aided Design (CAD) and scientific purposes.

Large scale EDP systems demand parallel developments in awareness of security issues and sophistication in protective procedures. DEC appreciates that nowadays, confidentiality is much more important than previously, when its systems were mainly installed by universities, whose priorities were power and flexibility. DEC has plugged security loopholes discovered in early versions of its operating system, VMS and VMS's security features improve with each new release. However, to oppose fully the threat of security breaches, MIS and EDP managers need management's support and assistance to implement and administer security properly.

'VAX VMS system managers require specialised expertise, advice and education to meet the increasing demands for controlling their EDP environments,' a spokesman of KPMG EDP Audit said a DEC VAX team which works intensively on the technical problems of achieving maximum security. It has close working links with Digital Equipment Corporation, holding regular meetings and familiarising itself with DEC's range of products. The group has organised courses, giving both KPMG auditors and senior IT executives from client companies an in-depth understanding of VAX security features.

Any access security check-list for VAX VMS cannot be truly comprehensive without reference to specific VMS operating system commands. Nevertheless, finance executives and internal auditors, in their discussions about security and control issues with technical EDP management, will wish to consider the following fundamental topics, which form part of an access security check-list for VAX VMS environments.

Expertise of EDP staff

The key to achieving an adequate level of access security is to ensure that staff have received sufficient training and expertise to operate, monitor and review security properly. Inadequate knowledge of DEC VAX's features, especially by the person responsible for implementing and managing security, can lead to incorrect use of security options and often wasted opportunities to protect the system.

User privileges

User privileges are attributes granted to users of the VAX system which enable performance of certain functions. When user privileges are uncontrolled, low level users can easily acquire high level rights of entry into confidential files such as pay-rolls, client details etc. Effective privilege controls begin with the construction of a clear picture as to how systems users are defined and exactly which access privileges they need. On a VAX system, privileges should be closely controlled, and also restricted by 'flagging' user identities by use of the CAPTIVE and DISCTLY attributes. CAPTIVE locks users into a fixed menu and DISCTLY prevents users from breaking out of their menu/system and using VMS commands. Two powerful privileges, SETPRV and BYPASS, should be restricted to selected personnel as they allow system security to be circumvented.

Thorough reviews are required to cope with pressure from users who continually assume they need more privileges than absolutely necessary to fulfil their function. This issue raises the familiar argument between security and productivity, ie: access controls versus speed and user-friendliness. The organisation's security policy will need to address this problem.

It is advisable that users are grouped with a reasonable allocation of duties. The security manager can then control file access, using safeguards defined by the user and the group to which he or she belongs. In recent versions of VMS, use can also be made of Access Control Lists (ACLs) which add flexibility to these procedures.

Security policy and passwords

Every organisation needs to determine and maintain its own up-to-date EDP policy, including how the VAX's security features are implemented. Its priority will be to define and establish those measures needed to protect the most sensitive directories and files.

A password policy should be established for each system. It should clarify whether passwords are issued centrally or chosen by the user. In a VAX environment, this is a flexible feature controlled by the security manager. Obvious, easily guessed passwords can be subjected to successful probing attacks. Obsolete user definitions or redundant passwords may also be routes for illegal entry, particularly in networks employing many temporary operators, each with his or her own password. Safety procedures should tidy your system, set up defences, and report attempted breaches of security.

An unusual feature of the VAX's security system is the password generator, which creates random passwords of a minimum length so that easily guessed passwords can be avoided.
The value of this feature in practice is a matter of opinion and provides much debate. This is due to the risk that when users have to work with obscure passwords which are difficult to remember, they are forced to write them down and may even file them in the system.

Secure locations

All locations should be clearly accounted for and their security controls coordinated, particularly in a multimachine environment. It should be anticipated that there is a risk that when security is merely organised on a local basis, a remote user seeking access could also acquire locally designated, high-level privileges. Output should be controlled to ensure confidential printouts are restricted to appropriate locations. If dial-up facilities exist, non-eligible users should be 'disabled' from using them.

Review policies

All log-in and surveillance procedures require adequate audit trails. The installation of software, such as one of the popular relational databases on the market with its own set of privileges, could affect the security features and controls which have already been established. They may therefore need to be reviewed. It is possible to establish regular record-and-review procedures so that management is kept aware and can respond to these problems. Security alarms in VMS systems give warnings of the frequency with which entry is attempted. Appropriate responses can then be made, according to whether the system is subject to malicious intent or simply user error.

Software tools

The recommendation for management responsible for system security is: 'It is essential that they provide their technical people with the tools to do the job properly.'

User-friendly, menu-driven software packages are available which issue commands to VMS to read data from system files. They can be useful aids to system managers responsible for VAX VMS environments, releasing them from spending time composing commands. For example, one fundamental requirement of security control is knowing precisely the privileges each user has. Software tools can facilitate authorisation reviews, providing a variety of information such as lists of users with potential to control the system, or with any specified privileges etc.

An evaluation of the VMS Security Toolkit has been carried out by KPMG Klynveld EDP Audit in Amsterdam. These software Tools has been developed by Cubic Systems in Australia. We hope to give details of such software in a future issue of K-Memory.

Expertise of computer auditors

KPMG's response to the concerns of trying to achieve effective EDP audit and security controls for clients has been a comprehensive research programme on DEC VAX and other commonly used systems.

Keeping abreast of DEC VAX's security developments and issues includes participation in appropriate courses, where there is often a valuable exchange of ideas. KPMG representatives attend the DEC's annual symposium, with its vital interchange of ideas and experiences, attracting DEC VAX users from all over the world.

Through KPMG's intensive research, a unique pool of knowledge on DEC VAX and the VMS operating system's audit and security features is now available to KPMG EDP auditors.

Final analysis

The potential for implementing good security on the DEC VAX is excellent, with extensive security features applied throughout its architecture. This contrasts with machines which must utilise 'add-on' security software to bolster security. To use VAX VMS security features effectively, however, requires a comprehensive understanding of the system. Ultimately, the degree of security achieved depends on the correct implementation, operation and proper use of these features.

It is common for EDP security measures to evolve on a haphazard basis, rather than as a result of policy planning. Education, technical knowledge and an understanding of appropriate procedures are essential to make full use of DEC VAX's integrated and varied security features. In the final analysis, only actions taken as a result of a detailed EDP security review gives management the confidence of having done everything possible to protect its system.

Diana Kenning

Het uitvoeren van een transactie-analyse

door: M.C. Duym

1 Inleiding

Transactie-analyse is een techniek om de bevoegdheden en verantwoordelijkheden ten aanzien van bestanden of programma's te analyseren ten behoeve van het verkrijgen van een uitspraak over het bestaan van een zodanig stelsel van logische toegangscontroles, dat daarmee de betrouwbaarheid van de in de betrokken bestanden opgenomen gegevens of van de in de programma's opgenomen verwerkingsfuncties, beter kan worden gewaarborgd.

De accountant en/of EDP auditor zal deze techniek hanteren indien de controle-aanpak is gebaseerd op gebruikmaking van EDP-controles en deze moeten worden getest voor wat betreft het aspect logische toegang. Echter voor gebruikers die kunnen beschikken over programmatuur die met behulp van bepaalde bevoegdheden van het besturingssysteem de toegangscontrolefuncties kunnen vermijden, zullen andere testen moeten worden uitgevoerd.

Voor de volledigheid wordt opgemerkt dat deze controle-aanpak verder nog testen voorschrijft ten aanzien van:
1. fysieke toegangscontroles;
2. systeemontwikkeling en programmawijzigingen;
3. handmatige correctieprocedures.

De techniek is toepasbaar bij on-line- en bij batch-verwerking.

2 Enkele definities

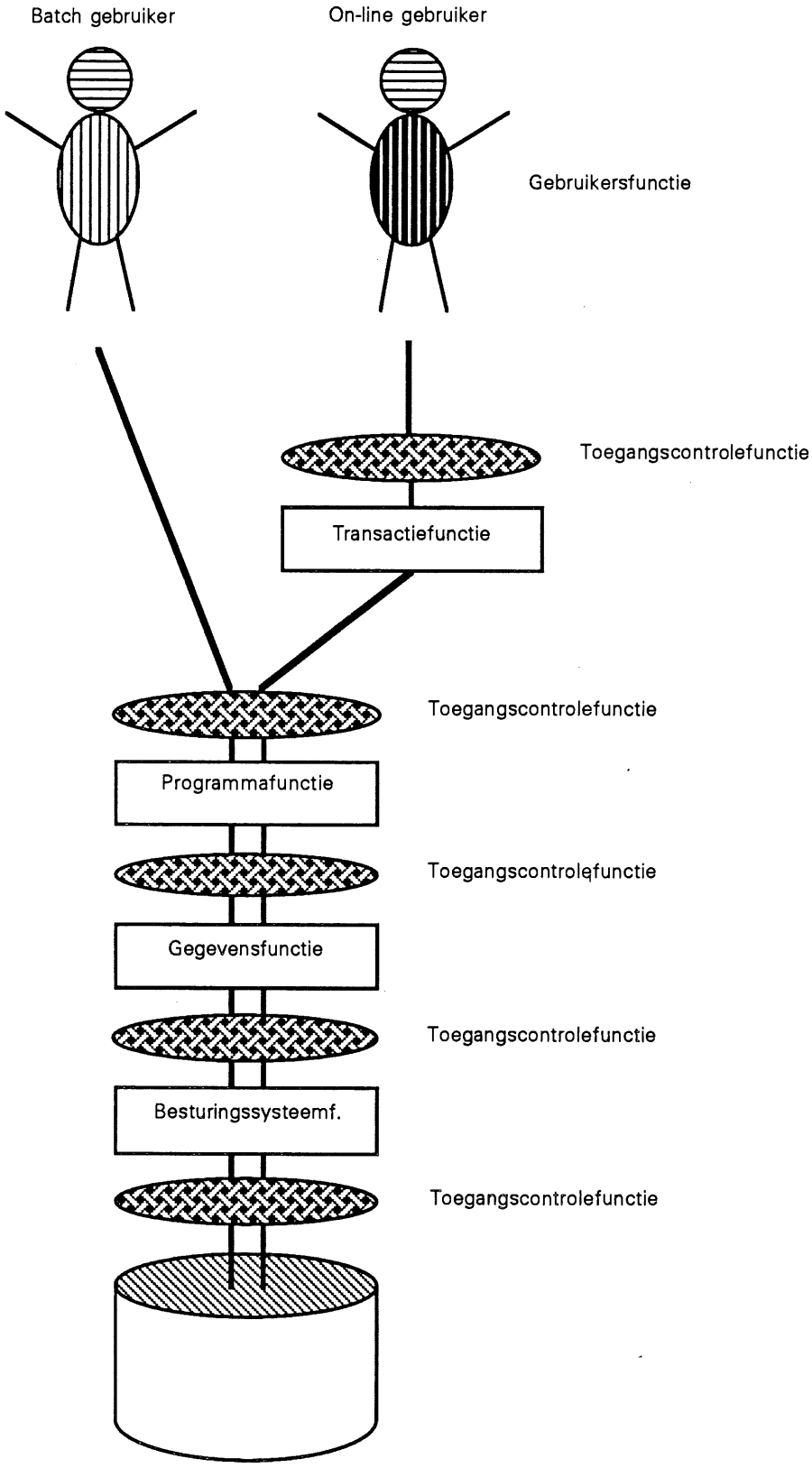
In dit artikel wordt aan de hierna genoemde begrippen de volgende betekenis toegekend:
- gebeurtenis: een door een persoon verrichte zakelijke handeling zoals inkopen, verkopen en produceren;
- gebruikerstransactie: de handelingen die de gebeurtenis weergeven in de registraties zoals opnemen crediteur en boeken factuur;
- systeemtransactie: de door het systeem aangeboden functies ten behoeve van het verwerken van de gebruikerstransacties zoals controle dubbele factuur en vastleggen factuurgegevens.

3 Functies in een transactieverwerkend systeem

3.1 Algemeen

Ten behoeve van een benadering die onafhankelijk is van de gebruikte software-produkten, wordt de volgende functionele indeling van de geautomatiseerde verwerking gehanteerd:
1. gebruikersfunctie: het initiëren van systeemtransacties door de gebruiker op basis van de gebruikerstransacties,
2. toegangscontrolefunctie: het toetsen van de door personen en programma's aangevraagde functies met de competentietabellen;
3. transactiefunctie: het vertalen van de gebruikerstransacties in één of meerdere systeemtransacties (is alleen aanwezig bij on-line-systemen);
4. programmafunctie: het verwerken;
5. gegevensopslagfunctie: het uitvoeren van de door programma's afgegeven gegevensmanipulatie-opdrachten;
6. besturingssysteemfunctie: het feitelijk uitvoeren van lees- en schrijfopdrachten.

Figuur 1. Schematisch weergave van de transactieverwerkingsfuncties.



3.2 Gebruikersfunctie

De gebruikersfunctie vormt voor de accountant het beginpunt van de transactie-analyse. Het betreft hier de laatste menselijke werkzaamheden voordat de machine het overneemt. Object voor het

accountantsonderzoek blijft natuurlijk de gebeurtenis. De vertaalslag van gebeurtenissen naar gebruikerstransacties blijft in dit artikel buiten beschouwing.

Ten behoeve van de analyse is het zinvol een onderscheid te maken tussen de functie enerzijds en de uitvoerende functionaris(sen) anderzijds. Conclusies worden dan in eerste instantie getrokken ten aanzien van de functies en vervolgens ten aanzien van de verdeling ervan over functionarissen. Dit geeft de volgende efficiency-voordelen bij het onderzoek:

1. indien in eerste instantie wordt vastgesteld dat de functies zodanig zijn geïmplementeerd dat de functiescheiding is doorbroken, is minder onderzoek verricht, omdat er meer functionarissen zijn dan functies;
2. omdat de toewijzing van functies aan functionarissen frequenter zal wijzigen dan de wijze waarop de functies in het systeem zijn geïmplementeerd, wordt de geldigheidsduur van het grootste gedeelte van het onderzoek verlengd.

Daarom zullen gegevens verzameld moeten worden over de functies en de gebruikers (functionarissen) die de functies uitoefenen. Tevens zijn gegevens over de relatie tussen beide nodig.

3.3 Toegangscontrolefunctie

Deze functie vormt de "voorkant van de machine". Zij bepaalt over welke machinefaciliteiten een gebruiker mag beschikken.

In vrijwel alle gevallen wordt aan de gebruiker gevraagd zich met een unieke identificatie te melden (user identification) en vervolgens nadere gegevens te verschaffen waarmee zijn identiteit kan worden geverifieerd (wachtwoord, kaart met magneetstrip en dergelijke).

Aan de hand van de ingebrachte gegevens wordt vastgesteld of degene die zich aanmeldt, gerechtigd is het systeem te gebruiken. Is dit het geval, dan wordt in de meeste gevallen het bij de desbetreffende gebruiker horende bevoegdhedenprofiel opgezocht, dat dient om tijdens de verwerking een bevoegdheidscontrole op een lager detailniveau te kunnen uitvoeren.

3.4 Transactiefunctie

Deze functie is voor de gebruiker veelal verborgen achter de menuschermen die worden getoond. Met behulp van deze schermen wordt de gebruiker langs de vele functies geleid die een bepaald systeem kent. Deze functies zijn de systeemtransacties.

Door het activeren van een functie wordt een programma gestart. De transactiefunctie geeft aan dit programma de door de gebruiker ingevoerde informatie door. Na afloop van het programma worden de verwerkingsresultaten door middel van deze functie weer aan de gebruiker getoond. Het door de gebruiker indirect geactiveerde programma kan zelf voor het uitvoeren van de verwerking weer andere programma's activeren. Ook deze activering loopt via de transactiefunctie.

De transactiefunctie kan zijn geïmplementeerd in/met behulp van:
- teleprocessing monitor (bijvoorbeeld CICS);
- time sharing monitor (bijvoorbeeld TSO);
- een applicatieprogramma.

3.5 Programmafunctie

Deze functie verzorgt de eigenlijke gegevensverwerking. Om de bestanden te vinden waaruit de gegevens voor de transactie-analyse kunnen worden geput, is het nodig onderscheid te maken tussen programmatypen en wijze van programmaverwerking.

De volgende typen van programma's kunnen worden onderkend:

1. interpretatieve programma's. Hierbij wordt opdracht voor opdracht omgezet in door de machine verwerkbare instructies die worden uitgevoerd. Naast het programma is dus steeds de inter-

preter werkzaam. Het komt steeds vaker voor dat voor raadplegen van opgeslagen gegevens de gebruiker de beschikking krijgt over een dergelijke taal;
2. gecompileerde programma's. Alle opdrachten worden gecompileerd in machinetaal. Vervolgens worden de dan ontstane objectmodules gelinkt om een zelfstandig uitvoerbaar programma te krijgen.

Naast dit onderscheid in typen programma's, is er een onderscheid mogelijk in de wijze waarop de programmatuur wordt uitgevoerd. Mogelijkheden zijn vanuit het oogpunt van de gebruiker bezien:
1. direct. De gebruiker maakt zelf zijn programma en geeft zelf de opdracht tot het uitvoeren daarvan;
2. indirect. Door een systeemontwikkelingsorganisatie worden door de gebruiker opgestelde (of goedgekeurde) specificaties omgezet in programma's. De uitvoering kan vervolgens plaatsvinden:
   a. door de gebruiker;
   b. door een produktie-organisatie.

Beide mogelijkheden hangen met elkaar samen. Zo zal een interpretatief programma in het algemeen met directe uitvoering gepaard gaan.

3.6 Gegevensopslagfunctie

Het afhandelen van het gegevensverkeer tussen de programma's en de opslagmedia, waarop deze zijn opgeslagen, is de hoofdtaak van deze functie.

Een nader onderscheid is te maken in:
1. gemeenschappelijk gebruikte zelfstandige sequentieel of index-sequentieel georganiseerde bestanden;
2. databases. Hieronder wordt een consistent samenhangend geheel van gegevensverzamelingen verstaan, dat met behulp van een database management-systeem wordt gemuteerd.

3.7 Besturingssysteemfunctie

Of een programma nu een rechtstreekse lees- of schrijfopdracht geeft, of door middel van Data Manipulation Language (DML) opdrachten voor het DBMS, in beide gevallen zal gebruik worden gemaakt van een toegangsmethode (access method), waarin het besturingssysteem voorziet.

Deze functie is volledigheidshalve genoemd en brengt in principe geen wijziging in de opdracht tot het benaderen van de door de gegevensopslagfunctie gewenste gegevens. In de beschrijving van de uitvoering van de transactie-analyse wordt daarom niet meer op deze functie teruggekomen.

4 Planning van de transactie-analyse

4.1 Vereiste kennis

Voor de uitvoering van een transactie-analyse is het noodzakelijk, dat de onderzoekers naast de noodzakelijke kennis van de administratieve organisatie, kennis hebben van de tabellen die worden bijgehouden door de software-pakketen die de diverse functies verrichten:
1. namen van de tabellen;
2. wijze waarop de tabellen kunnen worden geraadpleegd;
3. tabelbeschrijvingen;
4. wijze waarop de gegevens uit de tabellen door het desbetreffende pakket worden gebruikt.

4.2 Handmatige versus automatische uitvoering

Uit diverse praktijksituaties is gebleken, dat een geheel handmatige uitvoering van een transactie-analyse alleen is weggelegd voor systemen van geringe omvang. Dat wil zeggen, weinig transacties, programma's en gegevensverzamelingen.

Alhoewel het inzicht in de omvang van de bestanden pas na eigen inventarisatie kan worden verkregen, is een indicatie van het voor het systeem verantwoordelijke automatiseringspersoneel op voorhand te verkrijgen.

Ervaring leert, dat handmatige uitvoering op veel problemen stuit indien:
1. van meer dan 20 gebruikersfuncties onderzocht moet worden welke gegevens zij benaderen of;
2. het produkt van het aantal programma's en het aantal bestanden groter dan 1000 wordt.

Bovengenoemde vuistregel moet worden gehanteerd in samenhang met de volgende voor- en nadelen van een geautomatiseerde uitvoering ten opzichte van een handmatige uitvoering:
1. de initiële kosten van het bouwen van de benodigde programmatuur zijn hoog. De ontwikkelde programmatuur is erg gevoelig voor wijzigingen in de structuur van de onderzochte bestanden en het type database (hiërarchisch, netwerk, relationeel). Globaal kan worden gesteld, dat voor ieder database management-pakket een aparte versie van de programmatuur nodig is. Ontwikkeling van de programmatuur zal een ervaren programmeur al gauw 6 weken kosten;
2. de nauwkeurigheid van een geautomatiseerde verwerking, en daarmee de betrouwbaarheid van de uitkomsten, is groter;
3. doordat na de ontwikkeling van de programmatuur, het uitvoeren van de analyse weinig tijd kost, kunnen de consequenties van wijzigingen in de tabellen tijdens het onderzoek tot vlak voor het rapporteringstijdstip worden meegenomen. Dit geldt met name voor wijzigingen in de tabellen behorende bij de gebruikersfunctie, die een relatief hoge mutatiegraad hebben.

5 Uitvoering van de transactie-analyse met behulp van de computer

5.1 Inventarisatie tabellen

5.1.1 Algemeen

Om te kunnen bepalen uit welke tabellen gegevens voor de transactie-analyse kunnen worden geput, is kennis vereist van de software die is aangeschaft om de functies uit te voeren.

De verschillen tussen de beschikbare pakketten die de toegangscontrole-, transactie-, programma-, gegevensopslag- en besturingssysteemfunctie vervullen, zijn te groot om de namen te geven van de tabellen die in aanmerking komen. Wel is het mogelijk een zoekrichting aan te geven.

5.1.2 Gebruikersfunctie

Het bekende werkterrein voor de algemene accountant. Organisatie-, taak- en functiebeschrijvingen van het bedrijf moeten boven tafel komen.

De gebruikersregistratie van de toegangscontrolefunctie kan als hulp worden gebruikt. Zij geeft echter geen soelaas indien meerdere gebruikers van een zelfde toegangspad (veelal user-id-password-combinatie) gebruik maken. Door middel van interviews zal dan duidelijk moeten worden welke gebruikers tot dit toegangspad zijn geautoriseerd.

5.1.3 Toegangscontrolefunctie

Voor deze functie zijn specifieke pakketten te koop, zoals Access Control Facility 2 (ACF2) van CA en Resource Access Control Facility (RACF) van IBM voor gebruik op IBM mainframes. Veelal hebben ook pakketten die één van de andere genoemde functies verrichten een aantal controlefuncties in zich.

De specifieke beveiligingspakketten hebben de mogelijkheid om met bepaalde (veelverkochte) pakketten voor de andere functies te communiceren ten aanzien van bevoegdheden. Het zal duidelijk zijn, dat pas in een dergelijke situatie de mogelijkheden aanwezig zijn om een algehele en consistente toegangscontrole te creëren.

Ten aanzien van de transactie-analyse heeft een specifiek pakket een aantal voordelen. Deze zijn:
1. uit de bestanden van een dergelijk pakket kan veelal informatie worden verkregen over een groot deel van het gegevensverwerkend trajekt. Bijvoorbeeld:
   - de systeemgebruikers;
   - de systeemtransacties;
   - welke gebruiker welke systeemtransacties mag uitvoeren;
   - de bestanden;
   - welke gebruiker welk bestand mag benaderen en in welke hoedanigheid (lezen, wijzigen).
2. de afzonderlijke tabellen kunnen met eenzelfde set van commando's worden geraadpleegd.
3. de tabellen hebben bij een goede toepassing van het pakket een hoge betrouwbaarheid. Zie ook paragraaf 5.1.7 "Keuze en validatie van de tabellen".

Is geen specifiek pakket aanwezig, dan zal de toegangsbeveiliging in de meeste gevallen plaatsvinden door een pakket dat als hoofdfunctie de transactiefunctie heeft. Een dergelijk pakket kent tabellen met daarin:
- de systeemgebruikers;
- de systeemtransacties;
- de gebruikers en de systeemtransacties die zij mogen uitvoeren.

5.1.4 Teleprocessing-functie

Van dit gedeelte van het verwerkingstraject is informatie nodig over:
1. de systeemtransacties die er zijn en hoe deze samenhangen. Het komt namelijk vaak voor dat een transactie weer een andere transactie aanroept;
2. de relaties die de systeemtransacties hebben met de vorige en de volgende functie. Namelijk:
   a. welke gebruiker mag welke transactie benaderen;
   b. welke transactie hoort bij welk programma.

Een programma kan door meerdere systeemtransacties worden aangeroepen. Systeemtransacties verhouden zich daarmee tot programma's als n : 1.

Voor een juiste interpretatie van de tabel die de onder 2.a. genoemde gegevens bevat, moet bekend zijn of alleen de eerst aangeroepen transactie wordt vermeld, of - rekeninghoudend met het feit dat de transacties elkaar kunnen aanroepen - de uiteindelijk uit te voeren transactie(s).

5.1.5 Programmafunctie

Evenals systeemtransacties kunnen ook de programma's elkaar onderling als het ware "de bal toespelen". De als eerste aan bod komende programma's kunnen worden gehaald uit de bestanden van de transactiefunctie.

Om vast te kunnen stellen welke andere programma's worden geactiveerd staat een aantal wegen open. Namelijk het raadplegen van de:
1. systeemdocumentatie;
2. listings van het link-proces;
3. source listings met betrekking tot programma-aanroepen.

Voor het raadplegen kan een zoekprogramma worden gebruikt. Met name voor het raadplegen van de source listings is een dergelijk programma wenselijk.

5.1.6 Gegevensopslagfunctie

Deze functie beschikt in vrijwel alle systemen over een eigen tabel ten behoeve van de uitvoering van de functie. Deze tabel, data directory genoemd, bevat veelal gegevens over de wijze van opslag en toegang tot de gegevens. Soms bevat deze tabel bevoegdheidsregels die onafhankelijk van de gebruikers worden toegepast.

Is deze tabel niet aanwezig, dan moet de tabel uit de source code van de programma's worden gereconstrueerd. In paragraaf 5.2.3 zal blijken dat de transactie-analyse dan gezien de benodigde tijd en kosten economisch niet verantwoord is.

5.1.7 Keuze en validatie van de tabellen

Het zal regelmatig voorkomen, dat de voor het onderzoek benodigde gegevens uit meerdere tabellen kunnen worden geput. De keuze moet worden bepaald aan de hand van de volgende beslissingsregels:

1. neem de tabellen die tijdens de uitvoering van de transactie het door deze transactie gevolgde pad van gebruiker naar gegevens en vice versa bepalen;
2. kies hieruit de tabellen waarvoor de kwaliteit van de systeemontwikkelings- en/of produktie-organisatie bij de totstandkoming van die tabellen het meest optimaal is;
3. indien er voor een zelfde stap meerdere tabellen geschikt zijn, neem dan die tabellen die het aantal wijzen van verwerkbaar maken van de tabellen minimaliseert.

Toepassing van bovengenoemde regels in een situatie waarbij een beveiligingspakket is geïnstalleerd, zal veelal leiden tot keuze van de door dit pakket gebruikte tabellen. Wel moet vast staan, dat de installatie-opties van het desbetreffende pakket zodanig gekozen zijn, dat de daarin opgenomen toegangsregels op het moment van verwerking worden geraadpleegd en dat van deze regels niet mag worden afgeweken.

5.2 Nader onderzoek haalbaarheid

5.2.1 Algemeen

Nadat de benodigde tabellen zijn geïnventariseerd en de omvang daarvan is vastgesteld kan worden onderzocht, in hoeverre een transactie-analyse haalbaar is gezien de benodigde en beschikbare man- en computerkracht.

5.2.2 Omvang bestanden

Om te kunnen bepalen welke gebruiker, welke gegevens op welke manier mag gebruiken zullen alle tabellen moeten worden doorlopen.

Het maximum aantal paden van gebruikers naar gegevens is gelijk aan het produkt van de regels die de afzonderlijke tabellen bevatten. Dit produkt zal snel in de 9 cijfers lopen. Een beoordeling van al deze paden is onmogelijk.

Het analyseproces moet en kan dan ook worden beperkt. Dit gebeurt door een ontkoppeling te maken in de transactiegang. Op basis van de tabellen worden twee overzichten gemaakt. Eén die aangeeft welke gebruikers welke transacties mogen uitvoeren, en één die aangeeft welke transacties welke gegevens gebruiken.

Met het laatstgenoemde overzicht kan worden onderzocht wat de transacties eigenlijk behelzen en kunnen de voor het bedrijfsproces belangrijke gegevens worden onderkend. Vervolgens kan worden bepaald wat de belangrijkste systeemtransacties zijn.

Indien het privacy-aspect geen deel uitmaakt van het onderzoek, kunnen alle transacties die de gegevens alleen maar raadplegen buiten beschouwing worden gelaten. Vervolgens kunnen de minder belangrijke gegevens worden weggelaten. Het aantal combinaties wordt zodoende aanmerkelijk verkleind.

5.2.3 Toegankelijkheid bestanden

De wijze waarop de bestanden toegankelijk zijn, bepaalt in grote mate de benodigde menstijd en daarmee de financiële haalbaarheid van de transactie-analyse. Een altijd aanwezige methode om de tabellen te raadplegen is via een afdruk. De in dat geval benodigde tijd voor data entry van de

gegevens in het transactie-analyse systeem zal moeten worden afgezet tegen de voordelen van raadplegen van de bestanden met behulp van de transactie-analyseprogrammatuur.

Deze keus is niet zo makkelijk als zij misschien lijkt. Wat te denken van de situatie waarbij de programma-source moet worden geraadpleegd voor de lees- en schrijfopdrachten. Afhankelijk van de gebruikte programmeertaal kan dit een onmogelijke opgave zijn. Opdrachten in Cobol als "WRITE RECORD FROM HLP-DEB-REC" zouden met behulp van de File Section en eventueel Working Storage Section kunnen worden vertaald. Indien echter HLP-DEB-REC dan nog is gedefinieerd als REC-INHOUD PIC(80), dan wordt het zeer moeilijk een programma te schrijven dat via de diverse MOVE-opdrachten achterhaalt welke gegevens nu eigenlijk door het programma worden gemuteerd. Dit zou bijvoorbeeld alleen het telefoonnummer kunnen zijn, waardoor deze transactie als onbelangrijk zou kunnen worden bestempeld.

Nog meer inspanning zal het kosten, indien in hetzelfde systeem programma's voorkomen, geschreven in verschillende programmeertalen.

Doet bovengenoemde situatie zich voor, dan zal het in veel gevallen financieel niet haalbaar zijn om de analyse uit te voeren. Een beperking in de mate van detail in deze stap zal namelijk veelal niet wenselijk zijn, omdat zij een goede beoordeling onmogelijk maakt. Zo zou bijvoorbeeld in het gegeven voorbeeld de vereenvoudiging kunnen worden aangebracht, dat het programma met behulp van de File Section de gebruikte bestanden bepaalt en kijkt of er ook write opdrachten voor dit bestand worden gegeven. De conclusie dat alleen een telefoonnummer wordt gemuteerd en dat daarom de transactie onbelangrijk is, kan dan echter niet worden getrokken.

In bovenstaand voorbeeld is bovendien nog geabstraheerd van de mogelijkheid, dat raadpleging van de JCL-procedures noodzakelijk kan zijn om de gebruikte bestanden te achterhalen.

5.2.4 Verwerking tabellen

De verwerking van de tabellen begint op de computer waarop deze aanwezig zijn. Het kan bestaan uit (een combinatie van) afdrukken, kopiëren, converteren of construeren (denk aan het voorbeeld in de vorige paragraaf). Hierbij kan natuurlijk na kopiëren het converteren of construeren eventueel op een andere computer plaatsvinden.

Voor het koppelen van de tabellen kan het best gebruik worden gemaakt van een relationeel database-pakket. Immers het koppelen van de tabellen is niets anders dan een join-operatie op de afzonderlijke bestanden. De omvang van de bestanden is hierbij bepalend voor de benodigde processor-tijd. Een mainframe is uitermate geschikt. Met de krachtiger personal computers is echter ook goed te werken. Wel zal dan voor een complete koppeling van alle tabellen enkele uren nodig zijn.

5.3 Bouw van een transactie-analysesysteem

5.3.1 Inleiding

In dit hoofdstuk worden de componenten beschreven van een geautomatiseerd transactie-analyse systeem. Hierbij wordt geabstraheerd van het computersysteem - mainframe, micro of mainframe-micro-mix - waarop de verwerking zal plaatsvinden.

5.3.2 Verwerkbaar maken gegevens

Altijd zal moeten worden gewerkt met een kopie van de te gebruiken tabellen. De tabellen vormen immers het materiaal op basis waarvan de onderzoeksresultaten worden verkregen en moeten als een essentieel dossierstuk worden gezien.

Los van de noodzaak tot kopiëren is er de vraag, of het zinvol is de verschillende tabellen in het transactie-analysesysteem allemaal van een technisch identieke structuur te voorzien. Dit is afhankelijk van:

1. het feit of conversie van de gegevens nodig is. Wijziging van de structuur kan in dat geval meestal zonder veel extra moeite worden gerealiseerd;
2. de kostenafweging van enerzijds een consistent transactie-analysesysteem en anderzijds ontwikkeling en onderhoud van nieuwe versies van het analysesysteem.

Overwegingen bij de onder 2 genoemde kostenafweging zijn:
1. zijn er losse in- en uitvoermodules voor de tabellen, die makkelijk in het transactie-analyse systeem zijn te integreren?
2. in hoeverre vormt het onderhoudstechnisch een probleem, dat meerder toegangsmethoden in een zelfde systeem worden gebruikt? Met andere woorden, is de kennis ten aanzien van al deze toegangsmethoden voldoende verspreid?

5.3.3 Opslag gegevens uit tabellen

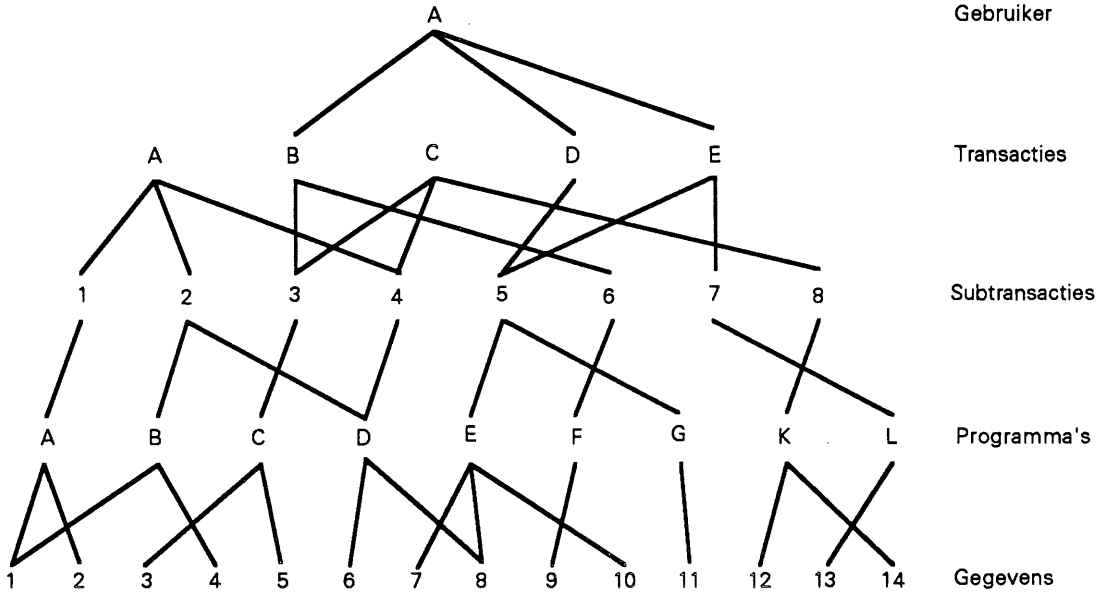
Voor elke tabel moet een transactie-analysetabel worden gemaakt. (De term transactie-analysetabel wordt in het vervolg gebruikt voor de kopie van de tabel die in het transactie-analysesysteem wordt gebruikt.) Indien is voorzien in handmatige invoer van de gegevens uit één of meer bestanden, is het noodzakelijk te voorzien in mutatiemogelijkheden op de desbetreffende transactie-analysetabellen. Het is aan te bevelen mutatiemogelijkheden op alle transactie-analysetabellen aan te brengen, zodat kleine wijzigingen in de bestanden, tijdens de transactie-ananalyse, handmatig verwerkt kunnen worden. Dit voorkomt de noodzaak tot het opnieuw uitvoeren van de conversie van het gehele bestand.

Elk record in een transactie-analysetabel moet uniek identificeerbaar zijn om problemen bij het koppelen te voorkomen. Hierop moet de programmatuur controleren.

5.3.4 Koppelen transactie-analysetabellen

Indien kan worden beschikt over een relationeel database-systeem met een join-opdracht, is het koppelen eenvoudig te verwezenlijken. Het resultaat van de join is echter niet altijd direct bruikbaar. Zoals uit figuur 2 blijkt, kunnen in elke stap "dubbele" ontstaan. In dit geval doordat zowel (systeem)transactie D als E gebruik maken van subtransactie 5. Deze dubbele zijn hooguit voor de controle op het join-proces interessant, maar niet voor de beoordeling. Zij moeten daarom worden verwijderd. Enkele database systemen bieden meerdere vormen van het join-commando. Het is dan mogelijk dat het gewenste resultaat met één van deze vormen direct kan worden verkregen.

Figuur 2. Koppelingsresultaten.



Voor gebruiker A zal de volgende resultaattabel na koppeling resulteren:

Gebruiker	Transactie	Subtransactie	Programma	Gegeven
A	B	3	C	3
A	B	3	C	5
A	B	6	F	9
A	D	5	E	7
A	D	5	E	8
A	D	5	E	10
A	E	7	L	13

Is geen relationeel database-systeem beschikbaar, dan zal de koppeling geprogrammeerd moeten worden. Hierbij kunnen tijdens het koppelingsproces dubbele worden voorkomen. Een mogelijkheid hiervoor is om voor elke gang van gebruiker naar gegeven (of andersom) een uniek nummer te genereren en dit in de geraadpleegde records achter te laten (de Klein-Duimpje-methode). Elke database beschikt daartoe over een extra veld voor dit nummer. Zodra het koppelingsalgoritme hetzelfde nummer tegenkomt, stopt het en gaat verder met de volgende transactie van de gebruiker.

Al eerder is er op gewezen dat het beoordelingsproces aanzienlijk kan worden versneld en vergemakkelijkt als de mogelijkheid aanwezig is om overzichten te verkrijgen van gebruikers met de door hen uitgevoerde transacties, en transacties met de gebruikte gegevens.

Ter beperking van het geheugengebruik en verhoging van de verwerkingssnelheid worden in het koppelingsproces alleen de unieke identificaties uit de transactie-analysetabellen gebruikt.

5.3.5 Presentatie resultaten

Bij het afdrukken van de overzichten voor de beoordeling (gebruikers - gegevens, gebruikers - transacties, transacties - gegevens) zullen, op basis van de unieke identificaties, in de resultaattabel uit de afzonderlijke transactie-analysetabellen de omschrijvingen moeten worden toegevoegd.

Identificatie van de overzichten met een datum- en tijdstempel is noodzakelijk om de meest recente versie te kunnen bepalen. Ook een einde-overzichtindicatie misstaat niet. Vooral als deze in de vorm van een teller van het aantal vermelde gebruikers, transacties of gegevens is.

6 Conclusie

Een transactie-analyse is een geschikte techniek in het onderzoek naar het bestaan van een goed stelsel van logische toegangscontroles. Toepassing van de techniek is, gezien de initiële investeringen, veelal pas efficiënt indien zij bij herhaling wordt toegepast. Als zodanig kan zij een goede plaats vinden in het toetsingsinstrumentarium van een intern controle-orgaan of een interne accountantsdienst. Uitvoering kan worden overgelaten aan een automatiseringsdeskundige die veel afweet van de wijze waarop de tabellen met gebruikers en hun bevoegdheden zijn opgeslagen. Een EDP-auditor is hiervoor een geschikt persoon. Beoordeling van de resultaten moet gebeuren door iemand die het belang van de gegevens voor de bedrijfsvoering goed kent. Dit kan ook de EDP-auditor zijn, maar veelal zal de algemene controleur de eerst aangewezene zijn.

Software escrow

door: R.A. s'Jacob

1 Wat is escrow

Ter verzekering van de continuïteit van een geautomatiseerd proces kan een onderneming een aantal maatregelen treffen. De meeste van deze maatregelen zijn gericht op de beschikbaarheid van de hardware en de gegevens. Maatregelen gericht op de beschikbaarheid van software zijn nog relatief onbesproken. Eén van deze maatregelen betreft het afsluiten van een escrow-overeenkomst.

Escrow vindt zijn basis in het Anglo-Amerikaanse recht en kan het beste als volgt worden omschreven:
"Een rechtshandeling die pas van kracht wordt na het intreden van bepaalde gebeurtenissen of verstrijken van bepaalde tijd, en die gepaard gaat met afgifte aan een derde van hetgeen waarop de wederpartij na het intreden van die gebeurtenis of verstrijken van die termijn recht heeft".

Escrow-overeenkomsten kunnen betrekking hebben op vele zaken. Een fabrikant van vrachtauto's bijvoorbeeld, die voor de produktie geheel afhankelijk is van een onderdeel dat van slechts één leverancier kan worden betrokken, kan overwegen met deze leverancier overeen te komen een kopie van de mal die voor de produktie van het betreffende onderdeel nodig is bij een onafhankelijke derde te deponeren. Indien de leverancier failliet gaat of de produktie van het onderdeel staakt, krijgt de fabrikant van de vrachtauto's de beschikking over de mal.

Het deponeren van de source code van software en de daaraan gerelateerde zaken zoals listings, onderhoudsdocumentatie, specifieke compilers en dergelijke (de "ontwikkelomgeving"), kan eveneens het object van een escrow-overeenkomst vormen. Waar in het vervolg gesproken wordt van de source code, worden hiermee tevens de overige hieraan gerelateerde in depot te nemen zaken bedoeld.

2 Waarom escrow

Organisaties die een wezenlijke of belangrijk genoeg geachte activiteit hebben geautomatiseerd, zullen de continuïteit daarvan willen garanderen. Naarmate het belang van een geautomatiseerd proces toeneemt, neemt in de meeste gevallen de afhankelijkheid van die automatisering eveneens toe. Indien een organisatie besloten heeft programmatuur niet zelf te ontwikkelen, dan kan die organisatie:
- het ontwikkelen van de programmatuur uitbesteden aan een software house;
- standaardprogrammatuur kopen.

Vaak krijgt de organisatie niet de beschikking over de broncode van de programmatuur. Hierdoor ontstaat voor wat betreft de beschikbaarheid en het onderhoud van de software een afhankelijkheid van de leverancier. Deze afhankelijkheid vergroot de gevoeligheid van de organisatie voor de volgende calamiteiten aan de zijde van de leverancier:
- faillissement;
- wanprestatie;
- overmacht;
- een besluit tot het stopzetten van een bepaalde activiteit, bijvoorbeeld het onderhoud.

Deze opsomming dekt in feite de gebeurtenissen zoals genoemd in de hiervoor vermelde definitie van escrow. Het risico dat niet kan worden beschikt over de source-code indien deze gebeurtenissen zich voordoen, kan worden beperkt met het afsluiten van een escrow-overeenkomst.

Helaas leert de praktijk dat in de meeste escrow-overeenkomsten de nadruk veelal ligt op het reguleren van de faillissementssituatie. Dit betekent dat niet of te weinig aandacht wordt besteed aan de overige mogelijke calamiteiten. Maatregelen gericht op het afweren van een curator zijn van een andere (juridische) aard dan de maatregelen gericht op het opvangen van bijvoorbeeld wanprestatie. In dit laatste geval is het belangrijk te voorzien in een snelle afgifteprocedure met betrekking tot de gedeponeerde source-code.

3 De escrow-agent

Zoals in de hiervoor weergegeven definitie is vermeld, gaat escrow gepaard met afgifte aan een derde van hetgeen waarop de wederpartij na het intreden van een gebeurtenis of verstrijken van een bepaalde termijn recht heeft. Deze derde wordt de "escrow-agent" genoemd.

De escrow-agenten kunnen in twee groepen worden onderverdeeld:
1. passieve escrow-agenten;
2. actieve escrow-agenten.

Een passieve escrow-agent treedt voornamelijk op als bewaarder. De te bewaren source-code wordt in ontvangst genomen, bewaard en afgegeven indien één van de gebeurtenissen zich voordoet zoals genoemd in de escrow-overeenkomst.

Een actieve escrow-agent verricht naast het bewaren van de source-code een aantal andere diensten. Voorbeelden van deze diensten zijn:
- het vaststellen dat de aangeboden gegevensdrager de source-code bevat zoals overeengekomen in de escrow-overeenkomst;
- het vaststellen dat aan de onderhoudsverplichtingen wordt voldaan;
- het bemiddelen bij geschillen.

Bij de keuze tussen een passieve of een actieve escrow-agent speelt de aard van de software een grote rol. Software die in geringe mate wordt onderhouden, zal in de meeste gevallen niet in escrow worden gegeven of eventueel bij een passieve escrow-agent. Software die daarentegen frequent wordt onderhouden en van cruciaal belang is voor de continuïteit van de bedrijfsvoering, zal in de meeste gevallen worden gedeponeerd bij een actieve escrow-agent.

Aan een (actieve) escrow-agent kunnen de volgende voorwaarden worden gesteld:
1. onafhankelijk van zowel de leverancier als de gebruiker;
2. betrouwbaar;
3. hoge graad van continuïteit;
4. kennis van automatisering;
5. juridische kennis;
6. internationaal.

Ook al wordt er voor een escrow-overeenkomst een passend juridisch maatwerk geconstrueerd, de effectiviteit van deze overeenkomst hangt in grote mate af van het vertrouwen dat beide partijen in de escrow-agent hebben. Alleen een escrow-agent die het vertrouwen van zowel de leverancier als de gebruiker kan dragen, is in staat een bijdrage te leveren aan het succes van een escrow-overeenkomst. Indien de escrow-agent niet aan zijn verplichtingen voldoet, kan de effectiviteit van de escrow-overeenkomst aanzienlijk dalen. Het doel van de escrow-overeenkomst, het beperken van het risico dat bij calamiteiten niet kan worden beschikt over de source-code, komt zeker in het geding indien de continuïteit van de escrow-agent in gevaar komt. In feite wordt bij het optreden van calamiteiten de afhankelijkheid van de leverancier gedeeltelijk vervangen door een afhankelijkheid van de escrow-agent. Hierdoor is het onaanvaardbaar dat de escrow-agent afhankelijk is van de leverancier of de gebruiker of dat de escrow-agent zich verschuilt achter onduidelijke constructies.

Om de leverancier en de gebruiker een passend juridisch maatwerk te kunnen leveren, dient de escrow-agent over voldoende juridische kennis te beschikken. En indien de escrow-agent een actieve rol vervult, is ook automatiseringskennis onontbeerlijk. Kennis van het produkt, de source-

code en de wijze waarop die kan worden gecontroleerd, is essentieel. Het verenigd zijn van juridische kennis en automatiseringskennis is echter een zeldzaamheid. Een praktische oplossing hiervoor is een samenwerkingsverband tussen een jurist (bijvoorbeeld een advocatenkantoor) en een onafhankelijke deskundige op het gebied van automatisering en controle.

Gezien het feit dat de automatisering ook internationaal sterk in ontwikkeling is, is het gewenst dat de organisatie van de escrow-agent eveneens internationaal georiënteerd is. Buitenlandse leveranciers van software-pakketten en buitenlandse vestigingen van in Nederland gevestigde organisaties zijn geen uitzondering.

4 Juridische aspecten van escrow

Indien een software-leverancier een bepaald produkt aan een gebruiker levert, kan er sprake zijn van een drietal overeenkomsten:

1. Licentie-overeenkomst
Dit is de eigenlijke levering van het produkt. De software wordt in licentie geleverd hetgeen inhoudt dat de leverancier (tevens auteursrechthebbende of handelend namens de auteursrechthebbende) aan de gebruiker het gebruiksrecht levert. De rechten van de gebruiker staan geregeld in de licentie-overeenkomst. Welke rechten voor de gebruiker overblijven nadat de leverancier failliet is gegaan, hangt sterk af van de inhoud van de licentie-overeenkomst. Als de licentie-overeenkomst (bij faillissement) wordt beëindigd, wordt ook het recht, verkregen van de auteursrechthebbende, om de software te gebruiken beëindigd, tenzij anders overeengekomen. Als de licentie-overeenkomst en dus het gebruiksrecht blijven bestaan na faillissement van de leverancier/auteursrechthebbende, zal de eventuele koper van de auteursrechten deze overeenkomst moeten accepteren, tenzij dit contractueel was uitgesloten, en in ieder geval het gebruiksrecht moeten erkennen.

2. Onderhoudsovereenkomst
De leverancier verplicht zich contractueel de software actueel te houden en de gebruiker te voorzien van nieuwe releases. De bij de COSSO (vereniging van software-leveranciers) aangesloten leden hebben de verplichting de onderhoudsactiviteiten van een eventueel failliet gegaan lid over te nemen.

3. Escrow-overeenkomst
De leverancier, gebruiker en escrow-agent komen overeen de source-code te deponeren. De overeenkomst op grond waarvan de source code geleverd wordt, kan worden gezien als een rechtshandeling in het kader van de uitoefening van de exploitatierechten die verbonden zijn aan het auteursrecht. De escrow-overeenkomst zal dus moeten worden aangegaan met de auteursrechthebbende of diens daartoe gemachtigde.

Escrow vindt zijn basis in het Anglo-Amerikaanse recht. In het Nederlandse recht komt deze constructie niet voor. In de plaats hiervan is een aantal juridische constructies te bedenken:

1. Bewaargevingsovereenkomst
Bewaargeving is geregeld in de artikelen 1731 en volgend van het Burgerlijk Wetboek. Bewaargeving kan slechts roerende zaken tot onderwerp hebben, zodat het auteursrecht niet, maar de source-code als roerend goed wel in bewaring kan worden gegeven. De bewaarnemer heeft de (wettelijke) plicht om zorgvuldig met de source-codes om te gaan en is op grond van de wet niet aansprakelijk als de source-code wordt beschadigd of verloren gaat wegens onvermijdelijk toeval. De bewaargevingsovereenkomst heeft voor de gebruiker de volgende nadelen:
- de bewaarnemer mag van de bewaargever/licentiegever geen bewijs vorderen dat de licentiegever eigenaar is van de source-code;
- de in bewaring gegeven goederen zijn vatbaar voor derdenbeslag, bijvoorbeeld door de curator van de failliet gegane licentiegever. Indien op de source beslag is gelegd, is de bewaarnemer niet langer verplicht de source af te geven aan de licentienemer;
- de bewaarnemer heeft een retentierecht met betrekking tot de in bewaring gegeven source-code, totdat al hetgeen de bewaargever hem ter zake van de bewaarneming verschuldigd is, is vergoed.

De licentienemer kan de bewaarnemer dus niet dwingen de source-code af te staan indien de failliete licentiegever de rekening van de bewaarneming geheel of gedeeltelijk onbetaald heeft gelaten en ook de curator niet bereid is deze alsnog te voldoen;
- zelfs als in een overeenkomst tussen de bewaargever en de bewaarnemer anders is bepaald, dient de bewaarnemer te allen tijde de gedeponeerde source-code aan de bewaargever terug te geven indien de bewaargever daarom vraagt.

Deze laatste bepaling holt de positie van de licentienemer, voor wie de source-codes in bewaring waren gegeven, uiteraard volledig uit.

2. Levering van de source-code in eigendom
Levering van de source-code in eigendom is een van de meest gehanteerde constructies om te waarborgen dat de curator geen beslag kan leggen op het gedeponeerde. Bij escrow kan beslag worden gelegd op het stoffelijk eigendom, het medium waar de source-code op staat en op het intellectuele eigendom door middel van een auteursrechtelijk beslag. Beide beslagvormen kent echter een aantal formele beperkingen en eisen alvorens geëffectueerd te kunnen worden. Bestudering hiervan is noodzakelijk om tot een goede juridische constructie te komen. Bij levering van de source-code in eigendom dienen de volgende vragen te worden beantwoord:
- Aan wie moet de eigendom worden geleverd, aan de licentienemer of aan de escrow-agent?
- Onder welke titel moet de eigendom worden geleverd?
- Welk eigendom moet worden geleverd, het stoffelijke eigendom of ook het intellectuele eigendom?

Het is de principiële vraag of levering van de source-code in eigendom aan de licentienemer niet in strijd is met de intentie van escrow; de gebruiker wil immers continuïteitsgarantie, geen eigendom van software.

Indien de source-code in eigendom van de escrow-agent wordt overgedragen, resteert er een tweetal mogelijkheden in verband met het auteursrecht:
- overdracht van het auteursrecht aan de gebruiker: dit is ongewenst voor de leverancier;
- een aanvullende licentie-overeenkomst: aan de escrow-agent wordt het recht verleend om onder opschortende voorwaarde kopieën te maken en die af te geven aan de gebruiker en aan de gebruiker het recht onder opschortende voorwaarde de source-code te gebruiken voor onderhoud.

3. Notarieel depot
Notarieel depot is het deponeren van de source-code bij een notaris door middel van het onlosmakelijk hechten van de source-code aan een authentieke akte (minuutakte). Hierdoor wordt de source-code (voor eeuwig) aan het rechtsverkeer onttrokken en is dus niet vatbaar voor derdenbeslag. De notaris heeft de volgende verplichtingen:
- de verplichting te onderzoeken of de licentiegever bevoegd is tot deponering en wie de auteursrechthebbende is met betrekking tot de source-code;
- de verplichting de source-code met alle zorgvuldigheid te bewaren;
- de verplichting de authentieke akte en de daaraan gehechte source-code aan niemand af te staan;
- de verplichting aan niemand anders dan de onmiddellijk belanghebbende personen afschriften af te geven van de authentieke akten en de daaraan gehechte source-codes.

De notaris is niet aansprakelijk voor de inhoud van de aangeboden source-code. Het aanhechten van de source-code, het maken van een afschrift ten kantore van de notaris en het verdisconteren van de kosten van het (eeuwig) bewaren in de kostprijs, levert praktische problemen op. Het onderzoek naar het feit of iemand onmiddellijk belanghebbende is kan tot vertraging leiden.
Hierbij dient te worden opgemerkt dat een aantal notarissen de aangeboden source-code als natuurlijk persoon in ontvangst neemt. Dit betekent dat het notariële depot, zoals door de wet beschreven, niet door deze notarissen wordt gebruikt.

4. Trust
De trust is een Anglo-Amerikaanse rechtsfiguur die in het Nederlands rechtssysteem niet voorkomt. Hetgeen in de trust wordt ingebracht, bijvoorbeeld de source-code, vormt een afgescheiden vermogen van zowel het vermogen van de escrow-agent als het vermogen van de licentiegever als het

vermogen van de licentienemer. De trust kan in Engeland of de Verenigde Staten worden gevestigd maar er zijn ook al uitspraken waarin de Nederlandse rechter trust-regels in Nederland toepaste. Hiervoor verwijs ik naar de literatuur op dit gebied.

Bij de civielrechtelijke constructies (ad 1. en 2.) kunnen zich problemen voordoen indien de leverancier geen auteursrechthebbende is of indien de curator de transactie nietig laat verklaren op grond van bevoordeling van bepaalde schuldeisers (Actio Pauliana).

In een (civielrechtelijk) contract dient onder andere aan de volgende zaken aandacht te worden besteed:
- het eigendomsrecht;
- het gebruiksrecht;
- de aansprakelijkheid voor de inhoud van de aangeleverde source-code;
- de mogelijkheden voor controle op de inhoud van de source-code;
- de aanlevering en organisatie van mutaties op de source-code;
- de mogelijkheid voor de contractpartners het contract op te zeggen;
- de calamiteiten op grond waarvan de source-code kan worden afgegeven;
- de procedure voor het geval er geschillen ontstaan;
- de aansluiting van het contract met de licentie-overeenkomst en de onderhoudsovereenkomst;
- de kosten van het bewaren en controleren van de source-code.

5 Voor- en nadelen van escrow

Voor de leverancier kan escrow de volgende voordelen hebben:
- het bereid zijn een escrow-overeenkomst te sluiten kan worden gebruikt als verkoopinstrument, een teken dat de gebruiker voor de beschikbaarheid van de source-code niet afhankelijk is van de continuïteit van de leverancier. Dit argument gaat vooral op voor een klein software house dat de continuïteit op geen andere wijze kan waarborgen;
- het extern bewaren van de source code is een versterking van de interne controle.

Voor de gebruiker kan escrow de volgende voordelen hebben:
- het risico van discontinuïteit van een geautomatiseerd proces is verkleind;
- de onafhankelijkheid van de leverancier is vergroot;
- een inhoudelijke toetsing van de source-code en de wijzigingen daarop is een waarborg voor kwaliteit.

Het nadeel voor zowel de leverancier als de gebruiker is het juridische maatwerk dat dient te worden gecreëerd. Escrow is een constructie die in het Nederlands recht niet voorkomt. Hierdoor dient er een, meestal ingewikkeld, alternatief te worden aangedragen. Bovendien wordt het nut van de escrow-overeenkomst aangetast indien er geschillen ontstaan over het zich al dan niet voordoen van een in de overeenkomst genoemde gebeurtenis. Het bijvoorbeeld vaststellen dat de leverancier wanprestatie heeft gepleegd, kan een langdurig proces zijn. Als gevolg hiervan kan de beschikbaarheid van de source-code in gevaar komen.

Volledigheidshalve dient hierbij nog opgemerkt te worden dat escrow slechts één van de maatregelen is die de continuïteit van een geautomatiseerd proces waarborgen. De bij de COSSO (vereniging van software-leveranciers) aangesloten leden hebben, zoals hiervoor al genoemd, de verplichting de onderhoudsactiviteiten van een eventueel failliet gegaan lid over te nemen. De noodzaak tot het sluiten van een escrow-overeenkomst wordt hierdoor verkleind. Indien de leverancier bij levering van software de source-code meelevert, is deze noodzaak zelfs in het geheel niet aanwezig. Bovendien is een adequate automatiseringsorganisatie bij de gebruiker een absolute noodzaak ter waarborging van de continuïteit van een geautomatiseerd proces.

6 Gevolgen voor de EDP auditor

Escrow is een nieuwe maatregel in het kader van het beheersen van de risico's van automatisering. In de literatuur is nog maar weinig te vinden over dit onderwerp. Naar verwachting zal de EDP auditor in toenemende mate door cliënten worden benaderd met vragen hieromtrent. Er kan dan sprake zijn van een tweetal soort opdrachten:

1. het optreden als escrow-agent;
2. het vaststellen van de overeenkomst tussen een gedeponeerde source-code en de load-versie bij de gebruiker.

Het onderkennen van de risico's die de EDP auditor bij met name het optreden als escrow-agent loopt, is essentieel. Kennis van escrow is hiervoor een vereiste. Deze kennis is te splitsen in juridische kennis en kennis op het gebied van automatisering en controle. Zoals hierboven reeds is vermeld, is het verenigd zijn van voldoende kennis op beide gebieden een zeldzaamheid. Het aangaan van een samenwerkingsverband met een organisatie die over voldoende juridische kennis beschikt, behoort tot de mogelijkheden.

Bij het vaststellen van de overeenkomst tussen een gedeponeerde source-code en de load-versie bij de gebruiker kan de vraag worden gesteld of dit ook daadwerkelijk mogelijk is, waar dit dient te geschieden en wat de kosten van zo'n onderzoek zullen zijn. Bovendien dient in de escrow-overeenkomst het gebruik van de source-code voor controledoeleinden te zijn geregeld. Het in notarieel depot gegeven van de source code kan voor controledoeleinden praktische problemen opleveren gezien de akte die er onlosmakelijk aan verbonden is en gezien de vraag of de EDP auditor als onmiddellijk belanghebbende kan worden aangemerkt.

Het is niet ondenkbaar dat bij het verkrijgen van een opdracht tot het vaststellen van de overeenkomst tussen een gedeponeerde source-code en de load-versie bij de gebruiker eveneens de opdracht wordt verkregen een oordeel te geven over de kwaliteit van de software. Hierdoor komt escrow in een geheel ander daglicht te staan. Dat het samengaan van software-escrow en software-keuringen/software-certificatie voor de EDP auditor gevolgen heeft moge duidelijk zijn. Zeker indien beide opdrachten door een gebruikersgroep van een bepaald software-pakket worden gegeven. Het is niet duidelijk ten behoeve van wie het gevraagde oordeel door de EDP auditor dan wordt afgegeven. Enerzijds kan het oordeel door de leverancier als verkoopinstrument worden gebruikt, anderzijds biedt het voor de gebruikersgroep een waarborg voor de kwaliteit van het pakket.

Het afgeven van een oordeel als boven bedoeld kan het risico van discontinuïteit van een geautomatiseerd proces bij de gebruiker verlagen. Bij de toetsing door de EDP auditor dienen de toetsingsnormen door de partijen te worden aangedragen. De EDP auditor kan hierbij een adviserende rol vervullen. De eenduidigheid van de aan de EDP auditor verstrekte opdracht naar aspect en scope is hierbij bepalend voor de waarde van het resultaat. Een EDP audit is echter tijdgebonden. Er wordt een oordeel gegeven voor een bepaald tijdsmoment. Iedere wijziging van de source-code na het moment van onderzoek kan gevolgen hebben voor het afgegeven oordeel.

Eén ding is zeker, escrow heeft toekomst. De EDP auditor dient hiermee in de pas te lopen.

Literatuurlijst:

1. Juridische aspecten rondom source code escrow
   Publikatie van het NGI, sectie computerrecht
   Onder redactie van mw. mr. A.H. Buth, mr. B.A.M. Cordemeyer, mr. T.J. Hermans, mr. A. de Leeuw, mr. N.J. Rinkel en mw. mr. J. Slager;
2. Source Code Depot
   Software-geheimen en faillissement
   mw. mr. Anne-Marie Ch. Kemna;
3. Escrow seminar
   Georganiseerd op 5 oktober 1988 door Kluwer in samenwerking met het NGI
   Congresdocumentatie, met name: Escrow in Nederland - ontwikkelingen en ervaringen, mr. J.E.H.C. Aarts.

Computervirussen. Worm in groot netwerk

door: drs.ing. J.C. van Winkel

1 Inleiding

Enige tijd geleden heeft er in het grote ARPA-Net (Netwerk in de U.S.A.) een worm rondgewaard die duizenden computers heeft aangedaan (zie bijgaand bericht uit NYTIMES News Service, 8 november 1988). Het programma wist zich via het ARPA-net te verspreiden door een aantal "gaten" in de beveiliging van de UNIX-computers die op ARPA-net zijn aangesloten. Er zijn geen bestanden beschadigd, de enige schade die geleden is, is het verlies van werkuren.

In de vakliteratuur wordt in dit geval niet gesproken van een virus, maar van een worm. Een worm is een programma dat zelf zijn weg vindt van computer naar computer en net als een bacterie ook zichzelf kan vermenigvuldigen. Een virus daarentegen is een deelprogramma dat altijd een ander programma nodig heeft om zich in te nestelen en dus ook andere programma's nodig heeft om zich te kunnen vermenigvuldigen. Dit net zoals een biologisch virus altijd een gastheercel nodig heeft voor reproduktie.

De worm die het ARPA-net parten speelde was door de zoon van een UNIX-beveiligingsexpert geschreven bij wijze van experiment. De dader had niet verwacht dat zijn produkt zich zo snel zou verspreiden.

2 Werking

De worm verkent met behulp van een in het systeem aanwezige tabel de buren die via TCP bereikbaar zijn. (TCP is een tussen UNIX-computers veel gebruikt protocol.) Er zijn nu drie wegen mogelijk:

- De worm opent een TCP-verbinding met het programma "sendmail" bij een buur. Aan sendmail wordt de opdracht gegeven in de zogenaamde "debug" mode te gaan. Deze mode is normaliter niet beschikbaar voor het programma, maar de systeembeheerder kan door hercompilatie deze optie wel beschikbaar maken.
  Via sendmail wordt de opdracht gegeven de binnenkomende data als een shell script op te vatten, zeg maar regels JCL. Dit kan alleen in de debugmode!
  Het probleem was nu dat het programma draait met systeembeheerders-privileges, en dus de binnenkomende shell script ook. Hierdoor had de worm vrij spel op het systeem.
  Vervolgens wordt een deel van de source-code van de worm opgestuurd. Dit deel is een hulpprogramma om het tweede deel (in object-code) op te halen. De source wordt gecompileerd en gelinkt, en de stap kan zich herhalen.
  Van het object-codedeel zijn twee versies: een versie voor VAX-computers onder UNIX 4.2 Bsd en 4.3 Bsd en één voor SUN 3 computers.
- De worm kan ook proberen met behulp van een "remote shell" via TCP in te loggen op het andere systeem. Hiervoor gebruikt het voor de hand liggende passwords. Deze passwords zijn in het al gecompileerde deel van de worm opgeslagen in "ge-encrypte" vorm, zodat ze niet meteen opvallen. Deze encryptie bestaat voor een deel van de passwords uit het optellen modulo 2 van een bepaalde waarde bij alle bytes van het password.
- De derde mogelijkheid verloopt via het programma "fingerd", maar door een programmeerfout in de worm kon van deze mogelijkheid geen gebruik worden gemaakt.

3 Uitstel van ontdekking

De ontdekking van de worm werd door de worm bemoeilijkt. Ten eerste was de naam van de worm zo gekozen ("sh") dat in een tabel van draaiende programma's de worm niet zou opvallen.

Ten tweede opende de worm alle eigen bestanden om ze daarna meteen te "unlinken". UNIX verwijdert dan wel de verwijzingen naar de bestanden waardoor ze in directory listings niet meer voorkomen, maar de data wordt pas van het systeem verwijderd als het bestand gesloten wordt.

De worm kon een systeem meerdere malen besmetten. Hierdoor werd de computer zo traag dat het zelfs uren kon duren om een tabel van alle draaiende processen te krijgen (iets dat normaal in enkele seconden kan). Ook kwam het voor dat er op het systeem zoveel processen kwamen te draaien dat er geen plaats meer was voor andere processen, inclusief een shell voor de systeembeheerder. De enige remedie was het stoppen van het hele systeem met behulp van de aan/uitschakelaar.

4 Vatbare computersystemen

De manier waarop de worm werkte stelt een aantal "eisen" aan de (UNIX) omgeving. Uiteraard moet een potentieel slachtoffer met behulp van TCP te bereiken zijn. Veel UNIX-systemen zijn echter niet via TCP maar met behulp van UUCP gekoppeld. Deze systemen zijn niet voor deze worm te bereiken.

Zoals boven, vermeld wordt een deel van de worm in objectvorm van computer naar computer gekopieerd. Hierdoor zal de worm alleen op zeer specifieke merken computers werken en onder specifieke versies van UNIX. De vatbare systemen zijn: DEC VAX-computers onder de Berkeley-versie van UNIX (UNIX 4.2 Bsd en 4.3 Bsd) en de SUN 3 systemen. Andere systemen kunnen niet besmet worden omdat het de worm niet lukt een executeerbare versie van zichzelf naar het andere systeem te kopiëren.

5 Remedies

Doordat de worm geen bestanden op het systeem achterlaat, is het mogelijk de worm kwijt te raken door het systeem te laten herstarten. Het is dan echter nog steeds mogelijk dat het systeem weer besmet raakt. Om dit te voorkomen moet de debug-optie van sendmail uitgezet worden door hercompilatie. Bij meerdere gekoppelde systemen moeten alle systemen tegelijkertijd uit de lucht gehaald worden.

Men heeft ontdekt dat als er in de systeembibliotheek (libc.a) een variabele PLEASEQUIT gedeclareerd wordt met een waarde ongelijk aan 0, de worm stopt met verdere verspreiding.

6 Bronnen

De hierboven genoemde informatie heb ik samengesteld uit berichten die op het UUCP netwerk over de worm verschenen. Nog steeds komen veel berichten over de worm binnen.

THE COMPUTER JAM: HOW IT CAME ABOUT

By JOHN MARKOFF

Computer scientists who have studied the rogue program that crashed through many of the nation's computer networks last week say the invader actually represents a new type of helpful software designed for computer networks.

The same class of software could be used to harness computers spread around the world and put them to work simultaneously.

It could also diagnose malfunctions in a network, execute large computations on many machines at once and act as a speedy messenger.

But it is this same capability that caused thousands of computers in universities, military installations and corporate research centres to stall and shut down the Defense Department's Arpanet system when an illicit version of the program began interacting in an unexpected way.

'It is a very powerful tool for solving problems,' said John F. Shoch, a computer expert who has studied the programs. 'Like most tools it can be misused, and I think we have an example here of someone who misused and abused the tool.'

The program, written as a 'clever hack' by Robert Tappan Morris, a 23-year-old Comell University computer science graduate student, was originally meant to be harmless. It was supposed to copy itself from computer to computer via Arpanet and merely hide itself in the computers. The purpose? Simply to prove that it could be done.

But by a quirk, the program instead reproduced itself so frequently that the computers on the network quickly became jammed.

Interviews with computer scientists who studied the network shutdown and with friends of Morris have disclosed the manner in which the events unfolded.

The program was introduced last Wednesday evening at a computer in the artificial intelligence laboratory at the Massachusetts Institute of Technology. Morris was seated at his terminal at Cornell in Ithaca, N.Y., but he signed onto the machine at MIT. Both his terminal and the MIT machine were attached to Arpanet, a computer network that connects research centres, universities and military bases.

Using a feature of Arpanet, called Sendmail, to exchange messages among computer users, he inserted his rogue program. It immediately exploited a loophole in Sendmail at several computers on Arpanet.

Typically, Sendmail is used to transfer electronic messages from machine to {x-!throughout the network, placing the messages in personal files.

However, the programmer who originally wrote Sendmail three years ago had left a secret 'backdoor' in the program to make it easier for his work. It permitted any program written in the computer language known as C to be mailed like any other message.

So instead of a program being sent only to someone's personal files, it could also be sent to a computer's internal control programs, which would start the new program. Only a small group of computer experts among them Morris knew of the backdoor.

As they dissected Morris's program later, computer experts found that it elegantly exploited the Sendmail backdoor in several ways, copying itself from computer to computer and tapping two additional security provisions to enter new computers.

The invader first began its journey as a program written in the C language. But it also included two 'object' or 'binary' files – programs that could be run directly on Sun Microsystems machines or Digital Equipment VAX computers without any additional translation, making it even easier to infect a computer.

One of these binary files had the capability of guessing the passwords of users on the newly infected computer. This permits wider dispersion of the rogue program.

To guess the password, the program first read the list of users on the target computer and then systematically tried using their names, permutations of their names or a list of commonly used passwords. When successful in guessing one, the program then signed on to the computer and used the privileges involved to gain access to additional computers in the Arpanet system.

Morris's program was also written to exploit another loophole. A program on Arpanet called Finger lets users on a remote computer know the last time that a user on another network machine had signed on. Because of a bug, or error, in Finger, Morris was able to use the program as a crowbar to further pry his way through computer security.

The defect in Finger, which was widely known, gives a user access to a computer's central control programs if an excessively long message is sent to Finger. So by sending such a message, Morris's program gained access to these control programs, thus allowing the further spread of the rogue.

The rogue program did other things as well. For example, each copy frequently signaled its location back through the network to a computer at the University of California at Berkeley. A friend of Morris said that this was intended to fool computer researchers into thinking that the rogue had originated at Berkeley.

The program contained another signaling mechanism that became its Achilles' heel and led to its discovery. It would signal a new computer to learn whether it had been invaded. If not, the program would copy itself into that computer.

But Morris reasoned that another expert could defeat his program by sending the correct answering signal back to the rogue. To parry this, Morris programmed his invader so that once every 10 times it sent the query signal it would copy itself into the new machine regardless of the answer.

The choice of 1 in 10 proved disastrous because it was far too frequent. It should have been one in 1,000 or even one in l0,000 for the invader to escape detection.

But because the speed of communications on Arpanet is so fast, Morris's illicit program echoed back and forth through the network in minutes, copying and recopying itself hundreds or thousands of times on each machine, eventually stalling the computers and then jamming the entire network.

After introducing his program Wednesday night, Morris left his terminal for an hour. When he returned, the nationwide jamming of Arpanet was well under way, and he could immediately see the chaos he had started. Within a few hours, it was clear to computer system managers that something was seriously wrong with Arpanet.

By Thursday morning, many knew what had happened, were busy ridding their systems of the invader and were warning colleagues to unhook from the network.
They were also modifying Sendmail and making other changes to their internal software to thwart another invader.

The software invader did not threaten all computers in the network. It was aimed only at the Sun and Digital Equipment computers running a version of the Unix operating system written at the

University of California at Berkeley.
Other Arpanet computers using different operating systems escaped.

These rogue programs have in the past been referred to as worms or, when they are malicious, viruses. Computer science folklore has it that the first worms written were deployed on the Arpanet in the early 1970s.

Researchers tell of a worm called 'creeper,' whose sole purpose was to copy itself from machine to machine, much the way Morris's program did last week. When it reached each new computer it would display the message: 'I'm the creeper. Catch me if you can!'

As legend has it, a second programmer wrote another worm program that was designed to crawl through the Arpanet, killing creepers.

Several years later, computer researchers at the Xerox Corp.'s Palo Alto Research Centre developed more advanced worm programs. Shoch and Jon Hupp developed 'town crier' worm programs that acted as messengers and 'diagnostic' worms that patrolled the network looking for malfunctioning computers.

They even described a 'vampire' worm program. It was designed to run very complex programs late at night while the computer's human users slept. When the humans returned in the morning, the vampire program would go to sleep, waiting to return to work the next evening.

Lezers reageren

Een zeer nauwkeurige lezer maakte ons terecht attent op het feit dat KPMG Klynveld EDP Audit zichzelf is vergeten in het artikel:
"Beveiligen tegen computermisbruik"
door: A.W. Neisingh RA en drs. J. Vossen RA.
Vindplaats: Compact 87/3, herfst 1987.

Onze redactie is dankbaar voor de opmerking en creëert ruimte om het verzuim goed te maken.

Het onderwerp van het artikel is dermate actueel, namelijk computerfraude door opzettelijke fouten dat van de eerste vijf hoofdstukken geen aanbeveling behoeft te worden teruggenomen. Het is nog steeds up-to-date en het aantal punten van attentie kan alleen maar worden uitgediept en/of aangevuld met het economisch/ethisch principe in het achterhoofd. Dit laatste om ongebreidelde groei van controle- en beveiligingsmaatregelen tegen te gaan.

Dit brengt ons tevens op het ontbrekende laatste hoofdstuk van bovengenoemd artikel:
Hoofdstuk 6 Functie van "EDP Auditing" ten aanzien van computermisbruik.

Op aanvraag levert de Redactie van Compact u geachte lezer gaarne een exemplaar van het gehele artikel. In het kader van deze voor u liggende Compact volstaan we met het publiceren van hoofdstuk 6.

6 Functie van "EDP auditing" ten aanzien van computermisbruik

EDP-audit is het door een onpartijdige deskundige onderzoeken en beoordelen van en adviseren over de kwaliteit van de logische, technische en ontwikkelinfrastructuur van een organisatie, separaat en in onderlinge samenhang, als gevolg van automatisering.

Deze definitie bevat een aantal essentiële punten die nader uitleg nodig heeft:
- onpartijdig deskundige die onderzoekt, oordeelt en adviseert. Dit geeft "audit" weer in de breedste zin des woords. Het komen tot een oordeel staat centraal en daarvoor is onderzoek nodig. Bovendien zullen uit het onderzoek en de oordeelsvorming adviezen voortvloeien om de situatie te verbeteren;
- na audit volgt kwaliteit. Dit is de invalshoek van de audit waardoor het deskundigheidsgebied van de deskundige nader wordt aangegeven. In deze definitie moet onder kwaliteitsaspecten worden verstaan:
  . betrouwbaarheid in ruime betekenis waaronder gerekend de deelaspecten als continuïteit, privacy, geheimhouding, etc.;
  . effectiviteit;
  . efficiency.
- het object van onderzoek van EDP audit is weergegeven als de logische, technische en ontwikkelinfrastructuur.

Onder de logische infrastructuur wordt verstaan het primaire bedrijfsproces met de daarin voorkomende processen (activiteiten), structuren (functies, afdelingen en personen) en informatiestromen, die voor de onderlinge koppeling zorgen.
Delen van de processen en informatiestromen worden geautomatiseerd in de ontwikkelinfrastructuur. De ontwikkelinfrastructuur is de organisatie met processen en structuren om de systeembouwprocessen te kunnen sturen en beheersen.
Naast deze infrastructuur bestaat bij geautomatiseerde gegevensverwerking nog een derde, de technische genoemd, te vergelijken met de automatiseringsorganisatie. Hierin wordt bepaald met welke mensen en middelen (hardware en besturingsprogrammatuur) de geautomatiseerde gegevensverwerking wordt verricht.

Wat kan men verwachten van een EDP-audit?
Alvorens een onderzoek in te stellen zal de deskundige (EDP auditor) in overleg met de opdrachtgever het doel van de audit vaststellen. Bij een specifiek onderzoek naar de beveiliging tegen bedreigingen van de geautomatiseerde gegevensverwerking, waarbij computermisbruik één van de bedreigingen is, zal de opzet en het bestaan van het stelsel van maatregelen en procedures gericht op de betrouwbaarheid (in de ruime betekenis) van de geautomatiseerde gegevensverwerking onderzocht worden. Echter er bestaat een wederzijdse beïnvloeding tussen de betrouwbaarheid en de effectiviteit en efficiency van de geautomatiseerde gegevensverwerking. Zo is het ondenkbaar dat onbetrouwbaarheid effectief kan zijn en onwaarschijnlijk dat 100% beveiliging efficiënt is. Dit maakt het noodzakelijk om bij een EDP audit alle aspecten mee te nemen en afhankelijk van het doel bepaalde aspecten meer te benadrukken.
Voor het onderkennen van de bedreigingen en het streven naar een evenwichtig pakket van maatregelen ter beperking van de bedreigingen wordt verwezen naar de al eerder besproken risicobeheersing.

Bij de afbakening van het onderzoeksgebied zal de onderzoeker zich te allen tijde moeten afvragen of de afbakening van het object van onderzoek tot een zinvol oordeel kan leiden en dat dit overeenkomt met de doelstelling van de opdrachtgever.
Zo zal een oordeel over een standaardpakket slechts de kwaliteit van de software en het waarheidsgehalte van de documentatie inhouden. Dit houdt niet in dat na implementatie van het pakket een betrouwbare gegevensverwerking een gevolg is en dus mede voldoende beveiliging tegen computermisbruik aanwezig is. De organisatorische maatregelen en procedures in de gebruikersorganisatie en automatiseringsorganisatie dienen dan bij het onderzoek betrokken te worden.

Voor het te onderzoeken object worden in samenwerking met de opdrachtgever eisen bepaald, die aan het stelsel van maatregelen en procedures gericht op de te onderzoeken aspecten van de geautomatiseerde gegevensverwerking gesteld worden. Deze eisen zijn de uitgangspunten/normen die bij het verder onderzoek naar de opzet en het bestaan van maatregelen en procedures gebruikt worden.

In de meeste gevallen zijn de eisen niet sec gericht op de "fraude"-bestrijding, maar worden deze opgesteld in het breder kader van de beveiligingsproblematiek tegen de bedreigingen van de geautomatiseerde gegevensverwerking.

Het resultaat van het onderzoek dient niet alleen tot een oordeel te leiden, maar een oordeel dient vergezeld te gaan met bevindingen van het onderzoek en aanbevelingen ter verbetering. In de bevindingen geeft de onderzoeker de geconstateerde tekortkomingen weer en geeft aan wat het gevaar van deze tekortkomingen is voor de opdrachtgever. Met de aanbevelingen helpt de onderzoeker zijn opdrachtgever om tot een aanvaardbaar stelsel van maatregelen te komen voor de kwaliteit van de geautomatiseerde gegevensverwerking.

Samenvattend kan gesteld worden dat ten aanzien van computermisbruik (als één van de bedreigingen van de geautomatiseerde gegevensverwerking) een EDP audit bijdraagt aan het verkrijgen van een aanvaardbaar stelsel van maatregelen en procedures gericht op de betrouwbaarheid (in de ruime betekenis) van de geautomatiseerde gegevensverwerking.
De bijdrage wordt geleverd door vaststelling van normen, onderzoek naar de opzet en het bestaan van maatregelen en procedures ter realisatie van de normen, het komen tot een oordeel en het verstrekken van aanbevelingen omtrent tekortkomingen.

Beheersaspecten bij gebruik van microcomputers

door: J.F.C. van Epen

1 Inleiding

Invoering van micro- of "personal"-computers brengt enkele organisatorische gevolgen met zich mee, die op vele punten afwijken van het gebruik van centrale automatisering. Soms ook zullen bepaalde maatregelen gelijk blijven, zij het dat deze nu tot de taken van de gebruiker van de microcomputer zijn gaan behoren in plaats van tot die van de (centrale) Automatiseringsafdeling. In het vervolg van dit artikel zullen deze aan de orde komen.

De microcomputer kan worden aangetroffen in diverse situaties:

1. **Op zichzelf staand**
   In deze situatie is voor de automatisering van bepaalde werkzaamheden één microcomputer aanwezig. Deze kan in een kleine organisatie taken vervullen waarvoor in grotere organisaties een (middel)grote computer is ingezet, zoals boekhouding, voorraadadministratie, projectenadministratie, personeelsregistratie en dergelijke, dan wel voor het automatiseren van persoonlijke routinematige of rekenkundig complexe werkzaamheden, zoals bijvoorbeeld actuariële berekeningen. Zo kunnen ook meerdere microcomputers op even zovele werkplekken zijn ingezet, elk voor verschillende taken. Het gebruik ervan is dan steeds als zuiver "personal" te kwalificeren.

2. **Aantal microcomputers met een zelfde gebruik**
   Op meerdere werkplekken zijn microcomputers ingezet, die zijn uitgerust met identieke programmatuur voor identieke werkzaamheden. Voorbeelden daarvan zijn tekstverwerking, technische tekenpakketten en dergelijke. De situatie zoals die bij KPMG Klynveld voorkomt, met vele computers ten behoeve van de accountantscontrole en elk met een verzameling soortgelijke programmatuur, is hiertoe te rekenen.
   Een mengvorm van situaties 1. en 2. is uiteraard ook mogelijk.

3. **In een netwerk geplaatste microcomputers**
   Mits in een netwerk geplaatst, kunnen microcomputers met elkaar samenwerken, dan wel aan elkaar gegevens doorgeven. Deze situatie komt thans nog op een bescheiden schaal voor, onder meer ten behoeve van kantoorautomatisering. Ontwikkelingen wijzen evenwel op een toenemend gebruik.

4. **Multi-user-systemen**
   Eveneens een soort netwerk is de opstelling van een centrale microcomputer als basiseenheid met processing-capaciteit en gegevensopslag, waaraan decentraal terminals en/of andere personal computers zijn aangesloten. Gemeenschappelijk gebruik van gegevens door meer dan één gebruiker tegelijk behoort dan tot de mogelijkheden. De centrale eenheid dient een relatief grote capaciteit te hebben.

2 Aandachtsgebieden

2.1 Algemeen

De beheersaspecten bij het gebruik van microcomputers zijn onder te brengen in een aantal deelgebieden, waaraan aandacht moet worden gegeven. Deze zijn:
a. gegevens;
b. apparatuur en standaardpakketten;
c. systeemprogrammatuur;

d. applicatieprogramma's.

Te onderkennen is dat dit in grote lijnen dezelfde aandachtsgebieden betreft als die bij het gebruik van andere, grotere computersystemen worden onderscheiden. Echter de organisatorische omgeving waarin gebruik wordt gemaakt van de microcomputer onderscheidt zich zodanig dat er sprake is van een wezenlijk andere organisatorische aanpak, die specifiek is bij het gebruik van de microcomputer.

2.2 Gegevens

Bij centrale automatisering en dus ook in de situatie zoals geschetst in hoofdstuk 1. onder 4. (multi-user-systeem), wordt het beheer van de gegevens centraal geregeld: Centrale opslag en kopiëren voor beveiligingsdoeleinden, centraal geregelde toegang tot de gegevens enz.

Decentrale verwerking, hetgeen bij de meeste microcomputertoepassingen het geval is, leidt ook tot decentrale gegevensopslag en -beveiliging. Bij elke vorm van automatisering waar gegevens van meerdere gebruikers worden beheerd, dient de toegang tot die gegevens te worden beperkt tot de geautoriseerde gebruikers. Bevoegdhedenschema's dienen te worden gehanteerd, waarin is vastgelegd "wie wat mag doen met welke gegevens".

Het gebruik van microcomputers bemoeilijkt het beheersingsproces. De classificatie van gegevens naar vertrouwelijkheidsgraad en belang voor de continuïteit van de organisatie, zal per individuele gebruiker moeten plaatsvinden. De subjectiviteit van de gebruikers kan leiden tot verschillen in benadering.

De vertrouwelijkheid van gegevens kan ook inhouden dat de toegang tot de personal computer of tot die specifieke gegevens voor onbevoegden onmogelijk moet worden gemaakt.

Per gebruiker zal de controle op de betrouwbaarheid van de gegevens, het beveiligen van de gegevens (kopiëren, veilig opbergen van de kopieën, het voorkomen van kennisneming ervan door onbevoegden en dergelijke), alsmede het aanleggen van documentatie over naam, versie en actualiteit (bijgewerkt tot en met .....) van de gegevensbestanden dienen plaats te vinden.

Ten einde te komen tot een zekere mate van uniformiteit in de wijze waarop binnen één organisatie door meerdere gebruikers een zelfde graad van betrouwbaarheid en beveiliging in acht wordt genomen, dienen afspraken ter zake te worden gemaakt en vastgelegd te worden in procedures en voorschriften, of beter nog in een procedurehandboek.

Van belang is hierbij wie voor het opstellen van procedures en voorschriften, het onderhoud daarvan en de controle op de naleving verantwoordelijkheid draagt.

2.3 Apparatuur en standaardpakketten

In organisaties waar meerdere microcomputers in gebruik zijn is het - om wildgroei te voorkomen - raadzaam een centraal aankoopbeleid te voeren. Dit beleid kan inhouden dat wordt bepaald welk type computer, dan wel andere identiek werkende computers (compatibles), binnen de organisatie zullen worden gebruikt. Dit geldt eveneens voor de standaardprogrammapakketten.

Door te streven naar uniformiteit wordt de onderlinge uitwisselbaarheid bevorderd (mogelijkheden tot interne uitwijk!), terwijl ook voordelen worden geboden ten aanzien van de aanschaf en het beheer van randapparatuur, supplies (diskettes, inktlinten, papier) en dergelijke, waarvan dan geen groot assortiment behoeft te worden aangehouden.

Een tweede zaak die heel goed op centraal niveau kan worden geregeld, is de "storingsdienst". Afhankelijk van de omvang van het aantal in gebruik zijnde microcomputers zal deze dienst reële storingsservice bieden, dan wel de contacten onderhouden met een externe onderhoudsdienst.

Vervolgens zij aanbevolen - eveneens centraal - registraties bij te houden van de in gebruik zijnde computers, randapparaten en standaardprogramma's, bij wie deze apparatuur en programmatuur in gebruik is, alsmede de per apparaat opgetreden storingen en reparatiekosten.

Mochten, om welke redenen dan ook, toch ongelijksoortige computersystemen in gebruik zijn (bijvoorbeeld voor dataprocessing en voor tekstverwerking), dan kunnen problemen optreden wanneer het noodzakelijk zou worden gegevens tussen deze verschillende systemen uit te wisselen. Meestentijds treden deze problemen op bij de diskettes die, als ze niet al een verschillende afmeting hebben, elk op een andere wijze de gegevens representeren en/of de index bijwerken. Conversie van het ene type naar het andere zal dan nodig zijn, wat met wisselend succes gerealiseerd kan worden. Bij KPMG Klynveld is hiervoor, om gegevensuitwisseling met de computers van onze cliënten mogelijk te maken, een speciaal programmapakket ontwikkeld, de File Conversion Tool, waarvoor echter ook speciale apparatuur vereist is.

Eveneens centraal te regelen is het gebruik dat van de computers mag worden gemaakt. Door of namens de leiding zal aangegeven moeten zijn welke toepassingen op welke werkplek geoorloofd zijn. Omgekeerd kan worden bepaald waarvoor de computer niet mag worden gebruikt, zoals voor spelletjes of voor privé-doeleinden.

De taken en bevoegdheden van de centrale support- en ondersteuningsgroep, dienen schriftelijk geregeld te worden in taakomschrijvingen, voorschriften en procedures, die ook aan de gebruikers bekend dienen te zijn.

Indien wordt gekozen voor het in de vorige paragraaf aanbevolen handboek, zullen de voorschriften en procedures ter zake van support en ondersteuning, de procedure voor het aanvragen van apparatuur en programmatuur, alsmede voor het doen uitvoeren van onderhoud daarin, een plaats moeten krijgen.

Ten slotte zij opgemerkt dat inherent aan het uitvaardigen van regels is de controle op de naleving van deze regels. Periodiek zal bijvoorbeeld vastgesteld kunnen worden dat uitsluitend de geautoriseerde programmatuur op de computers aanwezig is, dat de aanwezige gegevensbestanden conform de werkzaamheden worden aangehouden, dat de vereiste back-up-kopieën aanwezig zijn en van een voldoende recente datum zijn.

2.4 Systeemprogrammatuur

2.4.1 Algemeen

Zeer dicht bij de apparatuur staat de systeemprogrammatuur. Beperken we ons thans tot de belangrijkste: het besturingssysteem of operating system. In het besturingssysteem van de microcomputer zijn vier basisfuncties te onderscheiden, namelijk:
. geheugenbeheer;
. processor- en procesbeheer;
. beheer van de randapparatuur en
. data- en bestandsbeheer.

Het beheer van systeem en hulpbronnen is niet gelijk bij de te onderscheiden besturingssystemen voor microcomputers.

Enkele min of meer bekende besturingssystemen zullen kort worden besproken, gezien hun belang voor de werking van de computer en vanuit het gebruikersstandpunt het werken met de computer. In dit artikel wordt niet ingegaan op het besturingssysteem voor de Apple-computers, gezien de geheel afwijkende filosofie achter dit besturingssysteem.

2.4.2 CP/M

Dit operating system kan min of meer als standaard voor de 8-bits microcomputers worden beschouwd. Met CP/M-86 is getracht in de 16-bits markt door te dringen. CP/M-86 had aanvankelijk

een voordeel op MS-DOS, namelijk de mogelijkheid van "multitasking", hetgeen wil zeggen dat meerdere programma's gelijktijdig actief kunnen zijn.

2.4.3 MS-DOS (PC-DOS)

Een operating system dat langzamerhand standaard is geworden voor de 16-bits microcomputers. Het tot nu toe ontbreken van "multitasking" is inmiddels goedgemaakt door de opvolger van PC-DOS, OS/2.

Een opmerking is te maken voor de inmiddels verouderde, doch nog steeds aangetroffen MS-DOS versies 1.x, die ten opzichte van de versies 2.0 en hoger een afwijkende benadering van de randapparatuur kennen en daarin gelijk zijn aan CP/M. De versies 2.0 en hoger passen een ander systeem toe, dat ook bij UNIX wordt gevonden, namelijk dat randapparaten met een logische bestandsnaam kunnen worden aangeduid en gemanipuleerd. Laatstgenoemde versies kennen ook een geavanceerde structuur voor de inhoudstabel van de schijf. Mogelijkheden worden geboden om de toegang hiërarchisch te regelen en boomstructuren te ontwerpen, waardoor de toegang tot de bestanden kan worden versneld.

2.4.4 XENIX

Een beknopte versie van UNIX (zie hierna). XENIX is een oplossing voor micro's die niet genoeg intern geheugen beschikbaar hebben voor het veel grotere UNIX, terwijl in de gegeven situatie wel een operating system voor multi-user-toepassingen gewenst is.

2.4.5 UNIX

Dit operating system werd oorspronkelijk ontwikkeld voor minicomputers. Door de steeds toenemende mogelijkheden van microcomputers in de bovenste regionen wordt dit besturingssysteem steeds belangrijker. In zijn meest uitgebreide versie is dit systeem circa 2 Megabyte groot.

Een groot voordeel van dit operating system is dat het relatief goede mogelijkheden biedt de toegang tot het computersysteem en de gegevens te regelen, dus bij uitstek geschikt is voor een multi-user-omgeving. Ten einde een goede beheersing te verkrijgen bij een gewenste onderlinge uitwisselbaarheid is het raadzaam alle in gebruik zijnde microcomputers te voorzien van een zelfde operating system. Dit vereist centrale aanschaf en uitgifte, alsmede een registratie van welke systeem-software op welke machine is geïnstalleerd. De gebruiker moet in het oog houden dat programma's en diskettes meestal wel bruikbaar zijn onder hogere versies van een operating system dan waaronder zij zijn vervaardigd, maar doorgaans niet onder lagere, tenzij hiermede specifiek rekening is gehouden!

2.5 Applicatieprogramma's

Hierbij is een onderscheid te maken in standaardpakketten en "maatwerk". Voor wat betreft standaardpakketten is een centrale ondersteuning, die zorgt voor inkoop, distributie, registratie, documentatie en opleiding, een goede oplossing. "Maatwerk" betreft meestal voor één groep respectievelijk een groep van gebruikers vervaardigde specifieke programmatuur, die afhankelijk van de situatie centraal of bij de gebruiker in beheer kan worden gegeven. In het laatste geval dient echter wel centraal registratie plaats te vinden en mogelijk een kopie van het programmapakket en de daarbij behorende documentatie te worden opgeslagen.

Met betrekking tot programmatuur waarop een licentie rust, zal in de gebruikersrichtlijn het verbod tot kopiëren ten behoeve van derden, alsmede eventuele andere licentiebepalingen, expliciet moeten worden opgenomen.

Naast het hierboven genoemde onderscheid bestaat bij PC-gebruikers met enige ervaring soms de behoefte zelf toepassingen te ontwikkelen.

Belangrijk is het te onderkennen dat de beheersing hiervan slechts beperkt mogelijk is. De gebruiker verplichten de door hem ontwikkelde toepassingen op een centraal punt te laten beoordelen en registreren is nauwelijks te realiseren, omdat de gebruiker de toepassingen, die hij niet in de registratie wenst op te nemen, permanent "in de testfase" zal houden. Veel effectiever is het de programmerende gebruiker richtlijnen te geven voor het ontwikkelen van betrouwbare toepassingen.

Echter ook ingeval centraal programmatuur wordt ontwikkeld zijn richtlijnen in de vorm van ontwikkelingsstandaarden noodzakelijk. Zonder standaardisering is programmatuur, door een ander dan de auteur ervan, nauwelijks te onderhouden.

Systeemontwikkeling kan zowel plaatsvinden in de programmeertaal, een spreadsheet als met een database-pakket. In de richtlijnen dient voor elk van het voor ontwikkeling te hanteren pakket (Basic, Pascal, C, Spreadsheet, Database) de te hanteren standaarden te zijn opgenomen.

Tevens dient daarin de eis te zijn gesteld dat voldoende maatregelen van interne controle in de programmatuur zijn opgenomen, aan de hand waarvan de gebruiker de juistheid en de volledigheid van de verwerking kan vaststellen.

Aangekochte standaardpakketten gaan doorgaans vergezeld van een min of meer goede gebruikershandleiding. Met interne ontwikkelingen is dat niet steeds het geval. In een procedure zal aangegeven moeten zijn dat toepassingen voorzien moeten zijn van een adequate handleiding. Alleen op deze wijze kan consistentie in het gebruik worden bereikt.

Een gebruikershandleiding kan overigens beknopt zijn, vooral als het programmapakket voorziet in een zogenaamde "HELP"-functie, een interactieve documentatievorm, die steeds meer toepassing vindt.

Eigen ontwikkelingen zullen zodanig gedocumenteerd moeten worden dat de werking van de toepassing op een later moment daaruit is af te leiden. Voor ontwikkeling in een programmeertaal of database-pakket betekent dit een uitlijsting van het programma of de commando's, alsmede een verklaring van de gehanteerde variabelen. Voor een spreadsheet-toepassing een afdruk van het model met alle formules, aangevuld met een toelichting op het hoe en waarom. Eventueel bijzondere gebruiksvoorschriften dienen eveneens te worden vastgelegd.

Modellen en programma's, gebruikt in de accountantscontrole, zullen als voren aangegeven gedocumenteerd, een plaats dienen te krijgen in het desbetreffende cliëntendossier ten behoeve van het afleggen van verantwoording omtrent het controleproces.

Niet ongenoemd mag blijven het kostenaspect van systeemontwikkeling. Goed programmeren kost tijd en "tijd is geld". Van belang is het daarom dat voor het ontwikkelen van toepassingen, waarvan mag worden verwacht dat deze een aanzienlijk tijdsbeslag vergen, ongeacht of centraal of door de gebruiker zelf wordt ontwikkeld, vooraf toestemming wordt gevraagd. Daarbij dient tevens te worden nagegaan of intern een dergelijke toepassing niet reeds is ontwikkeld. Ook dit aspect dient in de genoemde richtlijnen te worden opgenomen.

3. Betrouwbaarheid van de gegevensverwerking

3.1 Algemeen

Aan toepassingsprogramma's die door een Systeemontwikkelingsafdeling worden vervaardigd, zal in het kader van dit artikel geen aandacht worden besteed. De eisen die daaraan gesteld moeten worden wijken niet af van die voor programmatuur voor de grotere computersystemen. De organisatie van de systeemontwikkeling dient borg te staan voor het opleveren van betrouwbare programmatuur. De gebruiker dient deze op de gebruikelijke wijze te testen en ten slotte te accepteren als de test naar wens is verlopen.

Eigen ontwikkeling door de gebruiker (personal computing) verdient met betrekking tot de beheersing van het gebruik van microcomputer bijzondere aandacht, omdat deze juist zo kenmerkend is voor de personal (micro)computer. In het algemeen zal er daarbij ook rekening mee gehouden moeten worden dat de gemiddelde gebruiker "self-made" is, hetgeen wil zeggen dat hij de theoretische achtergrond van het structureren van programmatuur, volledige beheersing van de gehanteerde programmeertaal en voldoende ervaring mist. Uitzonderingen bevestigen ook deze regel!

Indien de toepassing voor de organisatie van belang is, dat wil zeggen dat de uitkomsten ervan dienen als basis voor beslissingen, dient de toepassing te voldoen aan eisen van betrouwbaarheid. Hiertoe behoren in ieder geval een adequate test van de toepassing en een acceptatieprocedure. Wie accepteert hangt af van de eindverantwoordelijkheid voor de resultaten die met behulp van de toepassing worden verkregen.

Wordt de toepassing ontwikkeld voor (mede)gebruik door anderen dan de auteur, dan is ook een overdrachtsprocedure gewenst. Indien mogelijk dient te worden voorkomen, dat eenmaal geaccepteerde en/of overgedragen programmatuur door de gebruiker, ongeautoriseerd kan worden gewijzigd. De twee meest voorkomende ontwikkelingshulpmiddelen zijn de programmeertaal BASIC en het gebruik van spreadsheet-pakketten. Andere hulpmiddelen blijven, wegens hun veel minder frequente toepassing, hier buiten beschouwing.

3.2 Ontwikkelingen in BASIC

De programmeertaal BASIC, of meer geavanceerde versies daarvan als BASICA en GWBASIC, is een zogenaamde interpretatieve taal, die bij veel microcomputersystemen wordt meegeleverd.

Interpretatief wil zeggen dat tijdens de uitvoering van het programma opdracht voor opdracht wordt geïnterpreteerd (= omgezet in commando's voor de micro-processor) en uitgevoerd. Het programma is derhalve in de vorm zoals het werd geschreven in het werkgeheugen van de computer aanwezig.

Indien geen maatregelen daartegen worden ondernomen, kan het programma door de gebruiker dus op elk moment worden gewijzigd. Daardoor dient slechts beperkt gebruik te worden gemaakt van deze taal voor toepassingen van enige importantie, tenzij een beveiligingsmethodiek is toegepast, bijvoorbeeld compilatie in plaats van interpretatie.

BASIC is een taal die geen specifieke structuur afdwingt, waardoor een slordige wijze van programmeren toch tot een correcte verwerking kan leiden. De problemen ontstaan echter op het moment dat er wijzigingen in het programma aangebracht moeten worden of nog eerder als er toch verwerkingsfouten optreden, waarvan de oorzaak moet worden opgespoord en verholpen.

3.3 Spreadsheet-toepassingen

3.3.1 Algemeen

Naast tekstverwerking zijn spreadsheets de meest gebruikte pakketten op de microcomputer. En juist bij het gebruik van spreadsheets kunnen onopgemerkt verwerkingsfouten optreden. Vandaar dat aan de ontwikkeling van toepassingen met behulp van een spreadsheet in dit hoofdstuk extra aandacht wordt gegeven. Het ontwikkelen en gebruik van spreadsheets kan namelijk snel tot onbetrouwbare uitkomsten leiden. Een aantal redenen is hiervoor aan te voeren, zoals:
- het niet volledig begrijpen van een commando, waardoor dit anders uitwerkt dan verwacht;
- het per ongeluk geven van een foutief - soms fataal - commando;
- foutieve constructie van het model;
- foutieve rekenregels of formules;
- foutieve invoer van gegevens;
- fouten bij het bewaren.

Uit het hiernavolgende moge blijken dat deze punten zeer reëel zijn:

3.3.2 Fouten bij de hantering van commando's

Onverwachte gevolgen kunnen ontstaan bij het verwijderen van regels of kolommen, omdat buiten het geprojecteerde deel van het model gegevens of formules kunnen voorkomen die dan ongewild worden verwijderd. Fataal kan het zijn als per ongeluk kolommen worden gewist terwijl regels werden bedoeld, of omgekeerd.

Fouten kunnen ook ontstaan bij gebruik van het COPY-commando. Wordt intern, dus binnen dezelfde sheet, gekopieerd dan worden eventueel aanwezige rekenregels, zonder aanpassing van de relatieve referenties, meegekopieerd. Bij extern kopiëren, dus van de ene sheet naar de andere, worden juist geen rekenregels gekopieerd, maar uitsluitend "waarden", wat wil zeggen tekstvelden, datavelden en van formules de uitkomsten. Bovendien dient erop te worden gelet dat het resultaat van de COPY-opdracht op de juiste plaats in de sheet terecht is gekomen.

3.3.3 Foutieve constructie van het model

Als, vooral bij gecompliceerde modellen, vooraf geen analyse is gemaakt van het probleem en van de te kiezen structuur van het model, loopt men het risico na verloop van tijd opnieuw te moeten beginnen omdat het model niet werkbaar blijkt of niet tot de verwachte resultaten komt.

Belangrijk is om van te voren vast te stellen welke veranderingen in het model zijn te verwachten tijdens de gebruiksperiode ervan. Wijzigen brengt namelijk het risico van fouten door verstoring van de structuur met zich mee.

Van even groot belang is om vast te stellen hoe aan bepaalde velden wordt gerefereerd, absoluut (benoeming van het veld) of relatief (positie van het referentieveld ten opzichte van het veld waarin de formule staat).

3.3.4 Foutieve rekenregels

De fouten die bij het opstellen van de rekenregels kunnen worden gemaakt zijn dermate talrijk dat zij niet uitputtend te behandelen zijn. Het foutief bepalen van range-referenties komt veel voor, bijvoorbeeld in de sommatieformule SUM (eerste veld: laatste veld). Hier is het verstandig als eerste veld te benoemen het veld boven het feitelijke eerste veld (mits daarin geen andere waarde voorkomt) en als laatste veld het veld na het feitelijke laatste veld, waarin bijvoorbeeld de telstreep zal staan. Hiermee wordt bereikt dat bij tussenvoegen van regels de referentie correct aangepast wordt.

Een ander veel voorkomende fout is onjuist afronden. Wordt hiervoor namelijk het FORMAT-command gebruikt, dan zal in het veld correct worden afgerond, doch dat betreft slechts de presentatie van het getal. In berekeningen zal echter met de niet-afgeronde waarde worden gerekend.

Een voorbeeld moge dit verduidelijken:

Niet afgerond	FORMAT FIX 2 (2 decimalen)	FORMAT FIX 0 (geen decimalen)
3.333333	3.33	3
3.333333	3.33	3
3.333333	3.33	3
9.999999	10.00	10

Dit probleem kan worden ondervangen door te werken met de afrondingsfunctie (ROUND) of afkappingsfunctie (INT) in de rekenregels. Daarmee wordt de feitelijk gewenste (afgeronde) waarde in het desbetreffende veld geplaatst. Bij het verder rekenen met de desbetreffende waarden moet na gebruik van de ROUND-functie deze opnieuw worden gebruikt!

Een goed middel is het model te testen met behulp van eenvoudig te controleren bedragen, zodat narekenen goed mogelijk is.

3.3.5 Foutieve invoer van gegevens

Invoerfouten kunnen ontstaan door invoer van het verkeerde bedrag, door invoer in het verkeerde veld of door het verzuim een bedrag in te voeren. Het is aan te bevelen gebruik te maken van controlemiddelen en mogelijkheden, zoals voortellingen versus natellingen, voorgeprogrammeerde uitkomstcontroles, vierkantstellingen en dergelijke. Afhankelijk van de toepassing zijn er vele mogelijkheden te realiseren.

3.3.6 Fouten bij het bewaren

De spreadsheet-toepassing moet op het juiste moment zijn ge-"Saved", anders is de toepassing verloren of zijn verouderde gegevens aanwezig. Ook wordt nogal eens vergeten een "blanco" kopie van het model te bewaren (bij voorkeur in tweevoud), waardoor bij een volgende verwerking het bestaande model eerst geheel moet worden "schoon"gemaakt. Dat wil zeggen dat alle variabele informatie eruit verwijderd moet worden.

Bij sommige pakketten kunnen printfouten ontstaan, doordat het model, als de rekenfunctie "uit staat", niet wordt doorgerekend vóór het printen.

Ten slotte nog enkele attentiepunten:
- ter voorkoming van het per ongeluk wijzigen of verwijderen van formules is het mogelijk deze te beveiligen met het LOCK-command. Het verdient aanbeveling deze methode toe te passen bij operationele modellen;
- cijferbeoordeling, plausibiliteitscontroles en dergelijke vormen een goed middel om de uitkomsten te beoordelen. Accepteer niet klakkeloos de uitkomsten van de toepassing, zeker niet de eerste paar keer dat het model wordt gebruikt, of nadat het is gewijzigd;
- maak een goede documentatie van het model ten behoeve van later gebruik, gebruik door anderen en later onderhoud;
- zorg voor een identificatie van het model, bij voorkeur in de kop van de spreadsheet.

Nu wat uitvoeriger is ingegaan op de betrouwbaarheid van gebruikerstoepassingen zal de lezer in staat zijn ook voor andere talen en toepassingen, bijvoorbeeld database-pakketten of report generators zelf daarvoor betrouwbaarheidseisen te formuleren.

4. Beveiliging van de gegevensverwerking

Bij het beveiligen van de continuïteit van de gegevensverwerking zal afzonderlijk aandacht dienen te worden geschonken aan:
- apparatuur (microcomputer, printer, eventueel andere hardware). Dit zal voornamelijk betreffen de toegang tot, het in een veilige omgeving plaatsen van en het periodiek onderhouden van de apparatuur. Ook is aandacht te besteden aan diefstalpreventie;
- gegevensdragers, met name floppy disks. In verhouding tot de rest van de apparatuur en de hulpmiddelen zijn de magnetische informatiedragers zeer kwetsbaar. Floppy disks zijn relatief gevoelig voor stof, vocht en magnetische velden. Harddisks voor storen en verplaatsen. Floppy's (= diskettes) dienen steeds in hun beschermhoezen te worden bewaard. Lees de tips op de achterzijde van de beschermhoes! Zorg voor een duidelijke identificatie van de diskette (nummer, naam) en van de daarop aanwezige programmatuur en/of bestanden;
- programmatuur. Voor zover deze bedrijfseigen is, hetgeen wil zeggen dat deze bij verlies niet zonder meer opnieuw kan worden aangeschaft, zijn beveiligingsmaatregelen noodzakelijk. Zoals het kopiëren van de desbetreffende programmatuur en het veilig opslaan van de kopieën, het veilig bewaren van de bronprogramma's en blanco spreadsheet-modellen, het opnemen van extra kopieerslagen nadat een programma of model is gewijzigd enz. Is programmatuur van derden betrokken, dan dient op één of andere wijze te worden geregeld dat in geval van een calamiteit herlevering kan plaatsvinden;

- gegevens. Deze kunnen waardevolle informatie voor de organisatie bevatten en dienen derhalve goed te worden beveiligd. Een goede, op de situatie afgestemde back-up-procedure (zie elders in dit hoofdstuk) is vereist;
- documentatie. Niet verzuimd mag worden op een beveiligde plaats extra kopieën van niet zonder meer vervangbare documentatie op te bergen. Bij wijzigingen dienen deze niet vergeten te worden!

In het algemeen zal de gebruiker verantwoordelijk zijn voor een goede uitvoering van de beveiligingsprocedures. Alleen in de situatie van multi-user-toepassing zal centraal, namelijk daar waar de opslag van gegevens en programmatuur plaatsvindt, de beveiliging moeten worden geregeld.

In die situatie is ook een toegangsbeveiliging noodzakelijk, ten einde de gebruikers tegen elkaar te beschermen. Aandachtspunten zijn:
- classificatie: het classificeren van gegevens en aangeven wie wat met welke gegevens mag doen;
- identificatie van gebruikers en van de processen (programma's) en gegevens waartoe zij toegang mogen hebben;
- authenticatie: controle van de identificatie van de gebruiker. Dit geschiedt in de log-on-procedure en bestaat uit een geautomatiseerde controle op de door de gebruiker in te brengen:
  . identifier (user-id) en een
  . authenticator (password).
- autorisatie: het controleren aan de hand van een autorisatietabel of de gewenste toegang in overeenstemming is met de classificatie.

Indien gebruik wordt gemaakt van een centrale Support-afdeling kan het veilig stellen van programmatuur en documentatie aan deze afdeling worden overgelaten. In alle overige situaties is de gebruiker verantwoordelijk.

Back-up-procedures
De belangrijkste beveiligingsprocedure is de back-up-procedure.

Een inventarisatie van wat er zoal aan bestanden, programma's, documentatie en dergelijke in de geautomatiseerde omgeving wordt aangetroffen, leidt tot de volgende opsomming:
- gegevensbestanden;
- programmatuur;
- commando files;
- systeemprogrammatuur;
- operator- en eventuele overige documentatie;
- tekstbestanden;
- registraties van back-ups.

Sommige categorieën kunnen als bedrijfseigen worden aangemerkt, hetgeen wil zeggen dat deze niet buiten de organisatie opnieuw kunnen worden verkregen.

Per categorie zal afgewogen dienen te worden of, en zo ja hoe, de back-up-procedure(s) dient (dienen) te worden opgesteld: binnen elke categorie moet per object worden aangegeven hoeveel versies zijn vereist en met welke frequentie back-ups zullen worden vervaardigd. Van de uitvoering van de procedures dienen registraties te worden bijgehouden, bij voorkeur geautomatiseerd, zodat deze mede in de beveiliging zijn betrokken. Per object (bestand, database, programma, enz.) komt zo een back-up-kopie tot stand; we duiden dit aan als een objectgerichte back-up-procedure.

Daarnaast kennen we de mediumgerichte back-up. Dit houdt in het vervaardigen van een integrale back-up-kopie van een geheel medium (meestal harddisk).

Recapitulerend komen we tot twee basisprincipes voor het vervaardigen van back-up-kopieën, die elk hun eigen bezwaren kennen.

Objectgerichte back-ups houden het probleem van volledigheid in; mediumgerichte het probleem van de bewaartermijn.

Welke methode de voorkeur verdient hangt sterk af van de aard van de verwerking. Worden alle bestanden dagelijks bijgewerkt, dan is de mediumgerichte methode verreweg het gemakkelijkst. Wanneer echter op een zelfde medium ook bestanden voorkomen waarvan de up-date-frequentie veel lager ligt, dan bestaat het gevaar dat, wanneer een verwerking niet correct is verlopen, bij de volgende verwerking geen juiste versie van dat bestand meer aanwezig is. Alle goede versies zijn dan immers overschreven met de foutieve!

The IBM AS/400. A concern to the EDP Auditor?

door: H.J. Lijnes

De heer H.J. Lijnes is senior-organisatie-adviseur bij KPMG Klynveld Bosboom Hegener. Hij presenteerde dit paper op de KPMG Computer Audit Conference, Frankfurt, gehouden van 29 - 31 augustus 1988.

In June, to be precise on the 21st of June, IBM announced the successor of its mid-range 3x models, the Application System/400, AS/400.

The system, as announced, is certainly no big surprise to the System/38 user community. The System/36 user community however has to deal with a lot of differences, in particular when they enter the 'native' world of the AS/400. The new AS/400 provides the necessary capacity expansion in the areas of internal as well as external storage, and the overall performance of the larger models is providing for the necessary growth for those users at the end of the System/36 and System/38, without manpower consuming conversion activities.

I am not going to bore you with all the facts and figures of the announcement, but to set the environment, we are talking about six models, the B10, B20, B30, B40, B50 and B60.
Internal memory sizes range from 4 Mb to 96 Mb, maximum external storage capacity ranges from approximately 1 Gb to 27 Gb and a maximum number of communication lines ranging from 8 to 32.

And what may be the importance of this announcement to the EDP Audit practice?

Before answering this question, some quantities:
1. The AS/400 is not simply a replacement of the System/36 and System/38, but also a System with a well defined potential market outside the area of existing IBM users. As IBM puts it: 'The companies beyond the Fortune 500' are still good for 70% of the market:
2. As a replacement system for the S3x models we are talking about roughly 270,000 both S/34 and S/36 models and 30,000 S/38 models, give or take a few thousands;
3. At the announcement day 1250 applications were available, a number which will be grown to 5000 worldwide, at the end of this year;
4. 3700 AS/400 systems have been shipped worldwide to date, including 1709 early installs; any orders placed since announcement not included.

Based on those figures we may expect a growth up to in between 400,000 and 500,000 systems at the end of the next decade. And this estimate can be considered a rather conservative one.

To the S/36 user community and all new users, including those IBM mainframe users that will buy the AS/400 to be put in networks, the AS/400 is a completely new machine with a stunning architecture, with access control capabilities which are of proven quality and easy to handle compared to other systems in its class and beyond.

IBM's main strategy in marketing the AS/400 is:
- focus on natural markets;
- articulate explicit positioning;
- enhance IBM Business Partner relationship;
- strengthen field linkages.

The AS/400, from this marketing perspective, is an Application System, and quite a number of applications are and will be available. Some of those applications will be 'Internationally' available, however most of the applications will be 'Locally' available. The local applications are applications developed by third party software suppliers (former IBM agents) now called IBM Business Partners, who will be a non negligible factor in the selling of AS/400 Systems and the application software.

Where will this system sell? At least one third of the total sales will be in the US region, one third in Western Europe and one third in the Far East (primarily Japan, Australia and New Zealand).

Taking all this into account and looking at our own organization and its ambitions, it seems not particular bold to say that we will stumble upon at least 100,000 of those systems in our day to day practice, at sometime in the future.

So by quantity alone the system is important to the EDP audit practice.

As far as the application packages are concerned, a generally available and accessible base of knowledge and experiences on the internationally available applications seems worthwhile, from the point of view of economics and services to the clients.

End-user computing is a term which was introduced during the seventies along with some products on mainframes. At that particular time this way of end-user computing could only be used by the privileged few who had a large mainframe without performance bottlenecks. The arrival of the personal computer however caused an enormous boom in end-user computing. The only disadvantage is that in many cases the computing takes place independent of the main- and midframes, and that caused and may cause: uncontrolled growth of applications, redundant data and data inconsistencies.

The AS/400 will combine the data available on the midframe and the diversity of tools available on the personal computer by program products like OFFICE, PC SUPPORT and SQL.

SQL, which means Structured Query Language, will be available on the AS/400. SQL is a language which can be used to build, manipulate and interrogate databases. SQL experiences a growing popularity, being a language which is quite easy to learn and to use.

The language compilers of the AS/400 all support embedded SQL statements, as of November 25th, 1988, except for Pascal.

A growing number of end-users will get experienced in using SQL. EDP auditors who were using PC Support will be somewhat familiar with SQL if they transferred files from a System/38 to a personal computer.

Connecting a personal computer to the AS/400 will be quite attractive considering that the EDP auditor may formulate his own selection of data, using SQL, and transferring the data to his Personal Computer for further analysis using the available PC tools like FAT. This is no new activity, but the number of people performing transfers like this will grow substantially. Now one of the key questions is: 'Will I be able to connect to an AS/400 and transfer extracts from AS/400 databases to my Macintosh?'.

A rather small group of EDP auditors has been specializing on the System/38. As the AS/400 is quite comparable with the S/38 as to its access control capabilities, this small group will have to handle quite a large number of former S/36 users entering the world where access control need not be achieved by so called menu security alone.

The general term 'Menu security' stands for the technique of providing some kind of password protection in menu's that are used to access application environments. This way of protecting the access is difficult to manage, expensive to maintain and in many cases extremely frustrating from a performance point of view. The AS/400 offers excellent options to prevent unauthorized access, even better organized than on the System/38.

The System/38 uses a so called public authority, which by function is dependent of the type of object. Basically public means that a person can use the object in a normal way, but normal may be different for different object types. An explicitly granted authority is considered to be in addition to the public authority, which means that no person can be given less authority than public, other than by removing the public authority and granting every individual user the appropriate authorities.

The AS/400 implementation of explicitly granted authorities is instead of the public authority, which means that a person can be given less authority than public. The latter implementation simplifies the security management by object.

Another difference is the so called 'Authorization List'.
Groups of objects used by groups of users can be added to an authorization list. Users can be added to that list and be given a specific type of authorization. The user will then have the given rights to all objects belonging to that list.

When users are leaving the organization, and at the same time they are owning certain objects in the System/38, the user profile of that user can only be removed after transferring the ownership of the particular objects to another user. In the AS/400 a user profile can be removed instantaneously. The ownership of objects owned by that particular user are automatically transferred to a standard user profile which is called QDFTOWN ('default owner'). In certain situations this option may come in handy.

You may ask the question 'Looking at the AS/400, what is so different to the System/36 users? According to IBM they can use the AS/400 as if it where a System/36!'. In general that is true, although not quite. To preserve continuity, System/36 applications will be migrated to the AS/400, but new business applications will certainly use the new features, such as the database. Consequently there will be an inevitable growing usage of data in strategic applications, something that may be considered quite new for most of the users of this class of computers.

In the world of mainframes people are used to very large amounts of data stored on Direct Access Storage Devices. The saving and restoring of data is done application by application and one is used to the specific save schedules in the day to day operation of applications. In most situations data is stored on magnetic tapes, in other situations disk packs are removable. The last method is gradually disappearing.

On the AS/400, like the System/38, saving is done by library or object, resulting in copies on disk which can be transferred to magnetic tape at a later time, or resulting in copies on magnetic tape immediately. Although a magnetic tape subsystem with an instantaneous data rate of 469 Kilo bytes per second is available, try to imagine how many tape reels and how much time is needed to save 27 billion bytes (27,000,000,000). It sums up to app. 170 tape reels of 2400 feet and a density of 6,250 bytes per inch. The time needed, calculated at maximum speed and without taking time to rewind and change tapes, is at least 18 hours, using one tape unit. So, apart from more intelligent save procedures, new back-up devices are needed. IBM is developing and testing a cartridge tape streamer with a capacity of approximately 14 classic 2400 feet tape reels on one cartridge (2,2 Gb). Of course it is still unknown when the cartridge device will be available on the market.

The danger in the present situation is that the frequency of saving data may be decreased, caused by the duration of the save procedure. That seems to me something that auditors should be aware of.

Quite another problem area will be the AS/400 as a node in a network. How will the access be controlled?

System Application Architecture, in my opinion started as a surrogate for 'the lack of' standardization in operating systems and languages, contains the so called Distributed Data Management Services. The AS/400 will be the first SAA implementation of this type of services. Distributed Data Management Files are not unknown to the System/38 user community, however there seems to be a little problem in the interpretation of 'Distributed'. Up till now distributed meant that the distribution was static, you knew where to find the file. Real distribution could mean for example that a file is present at the location where it is used most frequently. When reading or updating a file the system itself searches for the right location. The consequences of this type of services are still unknown, so the specialists will have a nice topic which should be given thorough attention. Aspects like, consequences for journaling and commitment control, access control to files at different locations, etc. Verily, a true paradise for the researcher.

New developments are emerging, not with the speed of light, but fast enough to create a backlog in the necessary knowledge level of the general auditors as well as the EDP auditor. Not each and every new development should be exclusively dedicated to specialized EDP auditors. The general auditor is obliged to keep up with the level of expertise of the end-user, something that will ask for a considerable effort from his side and the side of the EDP auditors and experts as well, in providing the necessary 'Guidance notes' and training opportunities.
This should take care of moving the necessary expertise to another group, the general auditors. Isn't it true that the general auditor of today, has been the EDP auditor of twenty years ago?
And please don't worry about the experts becoming extinct.
They will find new specialisms, like good Chefs will find new recipes.

AS/400 security

door: Mw. Verena Six

1 Inleiding

Op 21 juni 1988 is Application System/400, kortweg AS/400, door IBM geannonceerd. Reeds voor deze annoncering deden al veel geruchten de ronde over deze nieuwe machine, die de opvolger zou moeten worden van de Systemen 36 en 38.

IBM noemt de AS/400 "the best of two worlds" en inderdaad zijn de invloeden van S/36 en S/38 duidelijk terug te vinden. Desalniettemin kan gesteld worden dat de architectuur voor negentig procent gebaseerd is op de oude S/38-architectuur. Die tien procent verschil moet eerder gezocht worden in toevoegingen dan in wijzigingen. De invloed van de S/36 is merkbaar in de toepassing van menu's. Standaard bevinden zich reeds meer dan 10.000 menu's op het systeem ter ondersteuning van de gebruikersvriendelijkheid.

Wijzigingen ten opzichte van de S/38 zijn onder andere de security. Aan het concept, objectgerichte beveiliging, is niets veranderd, maar er zijn wel veel uitbreidingen gekomen. Voorafgaand aan een uitwijding over de AS/400 security eerst enkele begrippen.

2 Begrippen

Object

Evenals de S/38 is de AS/400 object-georiënteerd. Dat wil zeggen dat alles een object is. Files en programma's zijn objecten. Maar ook menu's, commando's, jobqueues, outputqueues, etc. Circa 40 objecttypen bevinden zich op de AS/400. Werken op de AS/400 betekent werken met objecten en deze zijn alleen te benaderen door ze bij de naam te noemen. Het is niet mogelijk de objecten te benaderen door de fysieke adressering te gebruiken.

Library

Alle objecten bevinden zich in een library. Een library is op zichzelf ook weer een object. Objecten kunnen door het gebruik van libraries op een logische wijze gegroepeerd worden.

Menu

Nieuw ten opzichte van de S/38, bekend voor de S/36-kenners, is het gebruik van menu's. Hierdoor is de gebruikersvriendelijkheid vergroot.

Commando

De interface tussen gebruiker en machine wordt tot stand gebracht door commando's. Commando's vormen een onderdeel van de Control Language. De Control Language vormt de interface tussen de gebruiker en het operating-systeem van de AS/400. Met behulp van commando's en programma's kunnen objecten gemanipuleerd worden.

User profile

Iedere gebruiker op het systeem wordt geïdentificeerd door een user profile. Dit user profile dient niet alleen voor naam en password-verificatie, maar bevat tevens de bevoegdheden van de gebruiker.

3 Security levels

Op de AS/400 zijn drie beveiligingsniveaus, security levels, in te stellen:

Level 10 - physical security
Dit is het standaard level. Dit level houdt in dat iedereen die een naam ingeeft toegang krijgt tot de gehele machine. Voor deze gebruiker wordt automatisch een user profile gecreëerd. Er is geen enkele beveiliging van kracht.

Level 20 - sign-on security
Toegang tot de machine is alleen mogelijk met user profile en password. Daarna heeft de gebruiker alle toegang. Deze toegang is voornamelijk geregeld via menu's. Alleen de voor de gebruiker beschikbare opties op die menu's zijn in te stellen. Overige beveiliging is niet mogelijk. Dit level heeft alles te maken met menu-beveiliging en lijkt sterk op de oude S/36-beveiliging. Het gevaar blijft dat een gebruiker buiten zijn menu treedt dan wel via de menu's commando's uitvoert.

Level 30 - resource security
Dit is de eigenlijke objectbeveiliging. Iedere gebruiker kan voor ieder object al dan niet toegang gegeven worden.

4 Object security

Uitgaande van level 30-beveiliging is het mogelijk iedere gebruiker nauwgezette toegang te verschaffen tot alle objecten. Iedere gebruiker wordt bekend gemaakt aan het systeem en per gebruiker kan vervolgens worden ingegeven welk gebruik deze van welke objecten mag maken. Het gegeven welke gebruiker op welke wijze met welk object mag werken heet ook wel capability.

Deze bevoegdheden worden ingevoerd door een daartoe verantwoordelijke functionaris. In een AS/400-omgeving is deze functionaris de security officer of security administrator. Een consistent gebruik van objectbeveiliging kan tot een veilige machine leiden.

Een volledige implementatie van objectbeveiliging is tijdrovend en moeilijk onderhoudbaar. Daarvoor is ondersteuning beschikbaar van extra beveiligingsaspecten:

Library security
Logisch gegroepeerde objecten in een library kunnen in de eerste plaats afgeschermd worden door toegang tot de library te beperken tot een aantal functionarissen.

Authorisation list
Objecten kunnen toegevoegd worden aan een authorisation list. In deze lijst staat de toegang tot ieder object beschreven. Gebruikers behoeven nu niet meer voor elk object afzonderlijk geautoriseerd te worden, maar kunnen geautoriseerd worden voor de lijst, waarmee ze de toegang krijgen zoals die voor de objecten in die lijst staat gedefinieerd.

Exclude
Het is mogelijk dat vele functionarissen dezelfde toegang tot een object hebben, maar dat één of enkele daarvan uitgezonderd moeten zijn. In dat geval biedt de exclude-beveiliging uitkomst. Een functionaris met exclude voor een bepaald object heeft geen toegang tot dat object.

Special authority
Iedere gebruiker kan special authority toegewezen krijgen. De special authority staat in het user profile van de gebruiker beschreven en geldt voor alle objecten. Een voorbeeld van special authority is SAVSYS authority. Iemand met deze special authority heeft toestemming om objecten te saven en te restoren. Dit privilege geldt voor alle objecten, maar houdt eveneens in dat de gebruiker met deze authority geen gewone toegang heeft tot de objecten. Dit lost een deel van het probleem op dat operators die belast zijn met het maken van back-ups doorgaans te ruime bevoegdheden hebben.

User class
Gebruikers zijn onderverdeeld in klassen. De laagste klasse is die van eindgebruiker. Verder is er een klasse operator en een klasse programmeur. De hoogste klassen zijn die van security officer en security administrator. De security officer mag alles op het systeem. Een taak die alleen een security

officer mag uitvoeren is het aanwijzen van één of meerdere security administrators. Deze administrators hebben, evenals de security officer, het privilege om nieuwe gebruikers en toegangsbevoegdheden te definiëren. Waar een security officer echter toewijzingen kan doen met betrekking tot alle objecten, kan een security administrator gelimiteerd worden tot een beperkt aantal objecten. Hierdoor kan delegatie van de security officer taken plaatsvinden voor één subsysteem of afdeling.

Limited-capability
Gebruikers die werken met menu's hebben ook de beschikking over een commandoregel. Dat is een regel van waaruit systeemcommando's uitgevoerd kunnen worden. Deze mogelijkheid kan beperkt worden met de optie limited capability, die in het user profile staat.

5 Ten slotte

Het operating systeem van de AS/400 wordt geladen in microcode. Er is daardoor geen sprake van software waarop nog wijzigingen of speciale keuzemogelijkheden aangebracht kunnen worden zodat er in principe geen systeemprogrammeurs nodig zijn. Een voordeel, omdat dezen doorgaans ruime bevoegdheden op het systeem hebben. Bovendien is ieder operating systeem, in elke AS/400-omgeving gelijk.

De beveiliging is geïntegreerd in het operating-systeem en bevindt zich dus ook in microcode. Dankzij de object-georiënteerde architectuur van de AS/400 is het mogelijk geweest een beveiliging te implementeren die gebaseerd is op capabilities.

Het feit dat door het gebruik van capabilities een volledige toewijzing van alle objecten mogelijk is en het feit dat het operating-systeem in microcode is geladen, bieden de zekerheid dat de geïmplementeerde bevoegdheden niet ongeautoriseerd gewijzigd kunnen worden en dat niet om deze beveiliging heen kan worden gegaan.

Deze combinatie van factoren vormen een krachtig hulpmiddel waarmee zelfs in kleine organisaties met weinig mensen en middelen een effectieve beveiliging kan worden gedefinieerd.

Internationale gegevensstromen: abstract en moeilijk te controleren

Hoewel het voor veel mensen grotendeels abstract is, groeit de belangstelling voor Transborder Data Flows (TDF), juister gezegd, voor de regulering ervan. Hierbij gaat het om de grensoverschrijdende gegevensstromen en de feitelijke, technische en juridische beperking ervan. In mei 1987 organiseerde CELIM - een club van Europese computerrechtdeskundigen - een tweedaagse conferentie in Brussel over TDF en het recht van de Europese Gemeenschappen: "Freedom of data flows and EEC law", of voor de Franstaligen "Liberté des flux de données et droit communautaire".

door: mr. V.A. de Pous

Mr. V.A. de Pous houdt zich bezig met advies en informatieverzorging inzake juridische aspecten van de informatietechnologie en is onder meer uitgever/redacteur van de maandelijkse nieuwsbrief "News Ware".
Hij schrijft onder eigen verantwoordelijkheid, die de redactie van Compact ook niet kan overnemen. Het artikel is geheel voor verantwoordelijkheid van de auteur. Sinds 1987 heeft het artikel niet aan actualiteit ingeboet.

Een aantal juristen staat in de startblokken en sommige politici hebben hun issue gevonden. Internationale gegevensstromen moeten juridisch worden genormeerd, maar dat een en ander nog onduidelijk is en spraakverwarring creëert, blijkt onder meer uit de invulling van begrippen als informatie en vrijheid. Het Franse "liberté" getuigt van een meer absolute benadering van vrijheid dan het Engelse "freedom", dat reeds op zichzelf restricties inhoudt. Voor specialisten dus en die zullen wel nooit tot consensus komen. Hetzelfde geldt voor de invulling van het begrip informatie. Zware verhandelingen zijn over dit onderwerp geschreven en ieder woordenboek omschrijft informatie weer anders.

Ondanks deze problematiek presenteerden twee Franse rechtsgeleerden op de 2e CELIM-conferentie (1987) hun "Theorie juridique de l'information", waarin de introductie van nieuw type intellectueel eigendomsrecht centraal staat. Informatie als "informationeel goed". Hun theoretische opvattingen stemmen echter zo nauw met het auteursrecht overeen dat de Nederlandse jurist mr. P.B. Hugenholtz, zelf bezig met een proefschrift over de auteursrechtelijke bescherming van gegevens, weinig onderscheid constateerde.

Echter ook in Nederland doen deskundigen verwoede pogingen informatie juridisch te kadreren. Prof.mr. H. Cohen Jehoram van de Universiteit van Amsterdam spreekt in dit verband wel van informatie- en communicatierecht als "het recht met betrekking tot het produceren en overdragen van informatie in de ruimste zin van het woord". Er zijn volgens hem de drie concentrische cirkels waar te nemen: het auteursrecht als centrum met daaromheen het mediarecht en ten slotte het informatierecht dat toeziet op het genereren, verwerken, opslaan, transporteren en verspreiden van informatie. Een oeverloos vakgebied dus.

Wat mensen nog wel eens vergeten is dat het Engelse woord "information" in de Nederlandse taal vooral "inlichtingen" betekent, terwijl Cohen Jehoram het goed beschouwd over "gegevens" heeft. Dit is nogal verwarrend, mede omdat op een aantal plaatsen in het Nederlandse recht het begrip informatie in deze context wordt gehanteerd: het recht op het verkrijgen van inlichtingen tegenover de plicht deze te verstrekken. Rechten en plichten die onder meer in het contractenrecht, ondernemingsradenrecht (Wet OR), bestuursrecht (Wet openbaarheid van bestuur) en het in fiscaal recht zijn vastgelegd. Met andere woorden: informatierechten en -plichten.

Van juridische theorieën moet de internationale ondernemerspraktijk het vooralsnog niet hebben. G. Russell Pipe, deskundige op het terrein van de grensoverschrijdende gegevensstromen en uitgever van het enige tijdschrift over TDF "Transborder Data Flow", ziet het dan ook anders: "The most important issue is that information is trade. So we should be talking about information trade". Welke

juridische eigendomsconstructies op "information" of "data" worden toegepast, is daarbij van ondergeschikt belang.
Er vindt nogal wat grensoverschrijdend transport (transmissie) van gegevens in ontastbare vorm plaats, maar omdat een en ander in alle stilte geschiedt, is de conclusie dat zich blijkbaar geen problemen voordoen niet per se gerechtvaardigd. Er zijn immers twee mogelijkheden. Alles verloopt naar wens of geschillen treden niet in de openbaarheid. Een van de weinige bekende zaken naar aanleiding van internationale overdracht van gegevens betreft Data-Inspektionen versus Siemens (Zweden 1975). De multinational werd een exportvergunning voor persoonsgegevens van Siemens-werknemers in Zweden geweigerd, omdat hierdoor in Duitsland een schaduwregistratie van Zweden in het buitenland zou kunnen ontstaan.

De bescherming van de privacy is dus in eerste instantie aanleiding voor de normering van TDF. Maar er is meer.
Nationale regeringen willen grensoverschrijdende gegevensstromen normeren op grond van:
- de bescherming van privacy van hun staatsburgers;
- de bescherming van hun economie;
- de bescherming van de nationale veiligheid;
- het genereren van inkomsten door middel van invoerrechten.

Over wat nu wel en wat niet is toegestaan bestaat onduidelijkheid. De vuistregel die het bedrijfsleven toepast, luidt als volgt: Wat niet uitdrukkelijk wordt verboden, is toegestaan. Beperking van grensoverschrijdende gegevensstromen kunnen ernstige gevolgen hebben voor het internationale bedrijfsleven, bijvoorbeeld voor de luchtvaart. Recentelijk nog heeft deze bedrijfstak de noodzaak voor een vrije en ongestoorde uitwisseling van gegevens voor de Organization for Economic Cooperation and Development (OECD) uiteengezet.

De OECD is namelijk bezig met het opstellen van TDF/privacy-richtlijnen. Daarnaast heeft een OECD een werkgroep "International Computer Hacking" ingesteld, die de bestudering van het grensoverschrijdend inbreken in computersystemen tot onderwerp heeft. Een andere zaak die de aandacht heeft getrokken betreft de gedwongen verandering van de toegangscode van het centrale computersysteem van de Amerikaanse multinational Dresser, waarvan een dochteronderneming uit Frankrijk betrokken was bij de bouw van pijpleidingen in Rusland. Op 12 augustus 1982 werd Dresser France afgesloten van de Amerikaanse databank, met als gevolg dat zij binnen drie weken haar eerste orders verloor.

Grensoverschrijdende gegevensstromen kunnen feitelijk, technisch en juridisch met betrekking tot verzending worden beperkt, ook in relatie tot de inhoud, noteerde de Rotterdamse jurist mr. P.V.U. Grevenstein, sales manager bij het Mobil Oil. Zo heeft de verzender rekening te houden met toegangs- en gebruiksvoorwaarden van datanetten, voorwaarden voor aansluiting van randapparatuur op datanetwerken, toegangs- en gebruiksvoorwaarden van huurlijnen. Voor wat betreft de inhoudelijke beperkingen kan onder andere worden gewezen op de exportregels inzake strategische gegevens en privacy-voorschriften.

Tot nu toe is het zo, waarschijnlijk met Canada als enige uitzondering, dat wanneer een diskette of tape met gegevens de grens overgaat, slechts invoerrechten moet worden betaald over de waarde van de magnetische media en niet over de waarde van de daarop vastgelegde gegevens of programmatuur. Er zijn echter plannen in de maak om hierin verandering aan te brengen. Daarbij komt natuurlijk de vraag naar voren op welke wijze de belasting op het invoeren van gegevens en computerprogrammatuur, in het bijzonder in ontastbare vorm, kan worden gerealiseerd.
Daarnaast zal de controle op naleving van de invoerrechten wel voor problemen gaan zorgen.

In de Brusselse wandelgangen viel de volgende modus operandi te beluisteren. Introduceer eerst een uitvoerige registratieplicht zonder dat er van heffing sprake is.
Laat bedrijven nauwkeurig omschrijven wat zij van buiten de Europese Gemeenschappen aan data en software importeren. En dat betekent vermelding van aard van de gegevens (persoonsgegevens, financiële gegevens, strategische gegevens, etc.), omvang van de gegevens (aantal bytes), zender en ontvanger van de gegevens en wellicht eveneens een omschrijving van het gebruiksdoel van de data. Hetzelfde zou plaatsvinden ten aanzien van computerprogrammatuur. Er is overigens in toe-

nemende mate sprake van Transborder Software Flow. Zo wordt met name in het Verenigd Koninkrijk veel software-ontwikkeling aan Indiase automatiseringsbedrijven uitbesteed, die via netwerken Engeland inkomen. Het zou hier om miljoenen ponden gaan. Daarnaast wordt op dit moment software van centrale rekencentra via netwerken naar branche-offices overgebracht.

Door registratie kunnen nationale overheden dus inzicht krijgen in de aard en omvang van de grensoverschrijdende gegevensstromen. Later kunnen dan invoerrechten worden geheven. Of dit wenselijk is, is nog maar de vraag. Het internationale bedrijfsleven koestert in ieder geval een low profile om geen slapende honden wakker te maken.
Voor de douane die zich bezig moet gaan houden met invoerrechten op data software, doemen verschillenden chicanes op. Een tweetal springen eruit: het bepalen van de waarde van gegevens en computerprogrammatuur en handhaafbaarheid van maatregelen. Wat is de waarde van de gedigitaliseerde produkten? Moet bij software naar de ontwikkelingswaarde worden gekeken of zal de verkoopwaarde worden genomen?

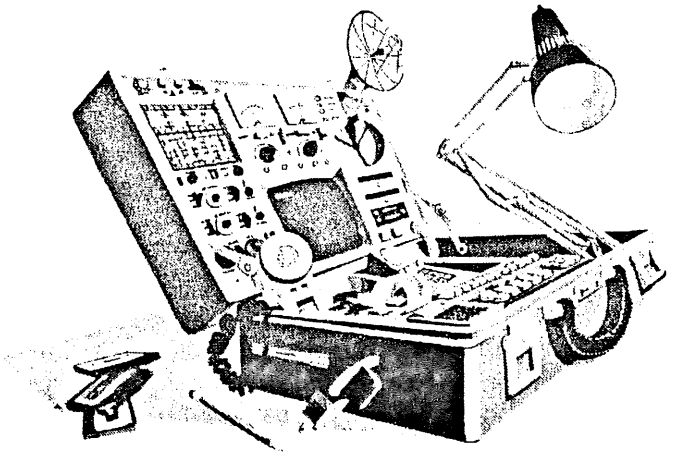
Vergelijk de situatie maar eens met de autobranche, waar de Fiat Croma vele miljoenen aan ontwikkeling heeft gekost, terwijl de belasting wordt geheven over de verkoopprijs van een auto (30 tot 40 duizend gulden). En als de keuze op de verkoopwaarde van de programmatuur valt, op welke wijze stelt de fiscus dan de waarde van de software vast, indien de ontwikkeling ervan geen handelsoogmerk heeft?

Anders gezegd: als de applicatie slechts voor eigen gebruik in een Zwitserse dochteronderneming is ontwikkeld.
Dan is er ook nog de omzetbelasting. Bij transfer van gecomputeriseerde gegevens wordt 20% B.T.W. (nu 18,5%) geheven (Viditel maar bijvoorbeeld ook de Juridische databank van Kluwer) en op gegevens in een geschrift opgeslagen meestal 6%. Zou de belasting toegevoegde waarde op eenvoudige wijze kunnen worden ontdoken door een buitenlandse gebruiker de Nederlandse databank te laten raadplegen die de gegevens via electronic mail of anderszins doorzendt naar de opdrachtgever in Nederland?
Op export van goederen en diensten is immers het O-tarief van toepassing?

Hoe is het nu in de praktijk. Er is al op gewezen dat bedrijven niet graag van hun perikelen rond grensoverschrijdende gegevensstromen verhalen. In een vorig jaar gehouden onderzoek onder dertig multinationale, Amerikaanse bedrijven naar ervaringen met TDF-restricties, is onder meer door het bedrijfsleven gezegd:
- "Korea, Taiwan and Brazil have imposed restrictions on a number of American companies in given fields with whom they will do business. It gives them the right to scrutinize the tape of data and the clientele";
- "In Latin America you cannot send data out of the countries to be processed. They will allow reports but they want all processing to stay in the country";
- "The people in these countries are curious about what information is transmitted. They can't intercept if the transmission is too fast, so they restrict and limit the speed at which data is transmitted";
- "In West Germany there is a specifical requirement for local processing. In Italy there is a red tape and government reporting regulations are cumbersome".

De ondervraagden zijn van mening dat het zowel de verantwoordelijkheid is van de betrokken regeringen als van de bedrijven om problemen op te lossen, waarbij de voorkeur uitgaat naar het opstellen en afsluiten van multilaterale overeenkomsten. Verdeeldheid heerst echter over de vraag of TDF-problemen openbaar gemaakt moeten worden, terwijl men het erover eens is dat regulering en dus beperking van TDF's de bedrijven geld kost.

Hoe het zij, dat de regulering van internationale gegevensstromen een belangrijk topic zal gaan worden lijdt geen twijfel, maar beleidsmakers en wetgevers zullen zich goed een van de belangrijkste rechtsbeginselen moeten realiseren: het recht moet namelijk relatie (blijven) houden met dat onderdeel van het maatschappelijk gebeuren dat het tracht te regelen. En dat betekent in ieder geval evenwichtige regelgeving die tevens handhaafbaar is.



De microcomputer in de accountantscontrole

door: ing. A. van der Vlist RI

In deze rubriek wordt aandacht geschonken aan de status en toekomst van "data retrieval" hulpmiddelen, dat wil zeggen hoe gaan we nu respectievelijk hoe gaan we in de toekomst, de bestanden van de cliënt te lijf. Het gaat hier niet om een uitgebreide evaluatie van bestaande produkten of het gebruik hiervan, maar meer om de vraag: "Hoe staan we ervoor en wat is de toekomst?".
Om de blik te verruimen zal dit in KPMG-verband worden beschouwd.

1 Enkele begrippen

Om verwarring te voorkomen eerst een drietal begrippen die nogal vaak gebruikt worden als het gaat over controletechnieken met behulp van de computer.

Data retrieval techniques:	Hulpmiddelen die aangewend worden voor het bereiken van controledoelstellingen. Data retrieval bestaat voornamelijk uit data-extractie, datamanipulatie en datapresentatie aan gebruikers.
Audit software:	Een verzameling computerprogramma's, gebruikt door accountants, om controledoelstellingen te bereiken. Deze programma's maken gebruik van de gegevens van de cliënt in een door een computer leesbaar formaat.
Computer Assisted Audit Techniques: (CAATS)	De audit software-applicaties zelf. CAATs zijn enerzijds de geconcretiseerde hulpmiddelen van de hierboven genoemde data retrieval techniques en anderzijds ook bijzondere vormen van "embedded routines" (stukjes controleprogrammatuur die zijn opgenomen in toepassingsprogrammatuur van de gecontroleerde), zoals extended records, SCARF, tagging en tracing. Zie "24 over EDP-auditing", artikel III, "Het gebruik van de computer in de accountantscontrole (mainframe), hoofdstuk 10. CAATs betreffen niet de algemene programmatuur ter ondersteuning van de controle als tekstverwerking en dergelijke .

In deze rubriek zal de term CAATs worden gehanteerd in de zin van data retrieval-hulpmiddelen.

2 Doel CAATs

Bij het gebruik van deze hulpmiddelen gaat het primair om een oordeel te verkrijgen over de kwaliteit van de gegevens van de cliënt in de brede zin van het woord.
Men kiest voor deze hulpmiddelen als ze effectiviteit en/of efficiency aan de controle toevoegen. Van effectiviteit kan gesproken worden als door het gebruiken van deze hulpmiddelen het controledoel wordt bereikt.

3 Status binnen KPMG

Binnen KPMG is een aantal CAATs in gebruik, zowel op de Audit Micro (Macintosh en MS-DOS) als op mainframe-computers. Deze hulpmiddelen hebben een redelijke historie, terug tot in de jaren '60! De vier belangrijkste zijn op dit moment (met een korte schets van hun werking/functionaliteit):

System 2190 — Genereert COBOL-programma's op mainframe computers aan de hand van S/2190-specificaties. Beschikbaar op een uitgebreide reeks cliëntcomputers. Front-end op Macintosh beschikbaar.

EDP Auditor — CULPRIT-bibliotheek die bestaat uit ongeveer 100 audit-functies.

GRACE (CARS) — Genereert COBOL op mainframe computers, analoog aan S/2190.

FAPL — Microcomputer (Mac, PC) georiënteerde groep van drie CAATs, te weten:

- FAT: Uitvoeren van data retrieval en audit-functies: list, sort, subtotal, extract record, calculate, select, sequence check, aging, report).
- FMT: Omgaan met twee bestanden (merge, compare, update, join).
- FCT: Conversie van vreemde formaten naar de Audit Micro.

Deze CAATs richten zich voornamelijk op zogenaamde "point-in-time" analyse op statische bestanden.

De CAATs bieden een aantal voordelen:
- groot aantal beschikbare controletaken;
- alle genoemde CAATs beschikken over de vijf basistaken list, sort, total, select en calculate;
- gebruikt door relatief veel professionals binnen KPMG.

Er kleeft echter een aantal nadelen aan deze hulpmiddelen:
- geen "Mac-look"-gebruikers-interface (dit geldt specifiek de mainframe-produkten);
- vaak is de hulp van een specialist vereist (bijvoorbeeld om de JCL bij S/2190 te schrijven);
- beperkte mogelijkheden om geavanceerde bestanden van de cliënt te benaderen, waaronder de zogenaamde databases (bijvoorbeeld benaderen met behulp van SQL DB2 of ORACLE).

De vraag is, gebaseerd op deze variëteit aan produkten, of geen nieuwe produkten gebouwd zouden moeten worden. Aan de andere kant zouden de genoemde produkten uitgebreid kunnen worden ter gelegenheid van (groot) onderhoud. Het is eigenlijk een keuze. De KPMG-microcomputergebruiker is door de Macintosh gewend geraakt aan applicaties die voor hem op maat gesneden zijn, gemakkelijk toepasbaar zijn en een minimum aan training vereisen. Vooral de mainframe-produkten vallen in de categorie "soms niet zo vergevingsgezind". Momenteel worden daarvoor aan FAPL op de Macintosh aanpassingen gedaan om de gebruikersvriendelijkheid nog verder te verhogen. Ook is er onderzoek gaande om S/2190 met behulp van geavanceerde mainframe-naar-Macintosh-programmatuur (Mac Workstation ™) gebruikersvriendelijker te maken.

4 Mogelijkheden nu

In Nederland is er een aantal alternatieven. Voor de eindgebruiker is er FAT en FMT; voor de gebruiker die het wil uitbesteden of die zijn gegevens wil laten converteren, kan worden uitgeweken naar de sectie Support & Programming (S&P) van KPMG Klynveld EDP Audit. S&P biedt specifiek hiervoor de volgende diensten:
- conversie van cliëntbestanden naar een door FAT/Excel/BFS leesbaar formaat;
- maken, uitvoeren en onderhouden van S/2190-/EDP auditor-/CULPRIT-/COBOL-toepassingen.

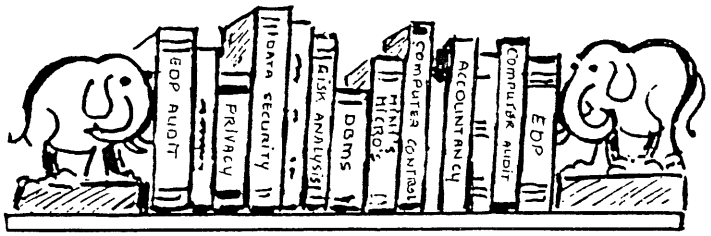
Via de microbeheerders is het mogelijk om van MS-DOS naar Macintosh te laten converteren.

Red.: BFS staat voor BASIC Financial Statements, een pakket om via journaalposten tot een financiële verantwoording te komen.

5 Toekomst

Gebaseerd op bovenstaande en als gevolg van de fusie tussen KMG en Peat Marwick, is sinds 1987 nagedacht over de ontwikkeling van nieuwe systemen en hulpmiddelen op het gebied van data retrieval. In 1988 is daarom onder de codenaam "PathFinder" - hoe vindt de auditor zijn weg naar de cliëntdata - door de sectie Software Engineering (SE) van KPMG Klynveld EDP Audit een basisontwerp voor een dergelijk nieuw systeem gemaakt. Hierop volgend heeft het Computer Advisory en Auditing Committee (CAAC) eind 1988 de Data Retrieval Task Force (DRTF) opgericht om er een internationaal KPMG gesteund project van te maken.
De zeer recent bedachte projectnaam is "WATSON", analoog aan de audit assistant van Sherlock Holmes.
Het ideaalbeeld van dit project is een koppelbare audit-micro die transparant aan de cliëntcomputer kan worden aangehaakt en een applicatie die eveneens transparant toegang geeft tot cliëntendata.

Zover zijn we echter nog niet, het project zal zich over meerdere jaren uitstrekken. Het wachten op het volmaakte systeem duurt misschien lang maar in de tussentijd is het uitstekend toeven met FAPL!



Boeken

Practical project management (restoring quality to DP projects and systems)

door: Meiler Page-Jones, Dorset House Publishing, New York ISBN: 0-932633-00-5

bespreking door: drs. P. Westdijk

1 Inhoud

De inhoudsopgave van het boek ziet er als volgt uit:

Section 1. Data Processing Serving the Corporation
1. achieving cost effective projects
2. estimating resources, costs and benefits
3. combining successful projects into successful systems
4. integrating data processing applications with business strategy

Section 2. The DP Project
5. organizing the DP department
6. managing the project
7. setting project deadlines
8. understanding project methodologies and standards
9. reporting project status and time
10. holding successful meetings
11. reviewing the project

Section 3. People: A DP Department's Greatest Resource
12. hiring and firing
13. developing your staff
14. establishing a productive working environment
15. working in a mediocracy
16. respecting reality
17. minimizing the human toll

Appendices
A. derivation of a project's CPM chart
B. problem solving
C. qualities of a good manager

2 Inleiding

De auteur beschouwt de problematiek van automatisering vanuit het gezichtspunt van de automatiseringsorganisatie; de gebruiker wordt in feite gezien als de "tegenpartij". Dit geeft, voor diegenen

die vanuit de gebruikerskant bij automatisering zijn betrokken, een verfrissende kijk op de in het boek behandelde problematiek. Hieronder zal eerst een beknopte samenvatting van het boek worden gegeven, afgesloten door een conclusie.

3 Samenvatting

Section 1: Data Processing Serving the Corporation
De automatiseringsorganisatie dient ten dienste te staan van de gehele onderneming. Te ontwikkelen systemen dienen dan ook een bepaalde verbetering binnen de gebruikersorganisatie tot gevolg te hebben (cost effective improvement to the users' business).

Het project moet een nut opleveren dat groter is dan de kosten en de kosten mogen de beschikbare resources niet te boven gaan. Het nut kan zich op vele manieren uiten, maar is altijd een verbetering van de concurrentiepositie van de onderneming.

Om tot een goede kosten/baten/beschikbare resources-analyse te komen stelt de auteur dat "het project" moet worden opgesplitst in kleine, identificeerbare delen, die elk separaat op hun merites en kosten moeten worden beoordeeld. Daarna wordt beslist:
- welke deelprojecten worden uitgevoerd;
- welke volgorde daarbij wordt gehanteerd (let op onderlinge afhankelijkheden);
- welke deelprojecten worden uitgesteld of geheel niet uitgevoerd.

Een volgende stap in dit analyseproces is het maken van schattingen. Het schatten van de beschikbare resources wordt onbetrouwbaarder bij het toenemen van de horizon in verband met personeelsverloop, uitlopen van tussentijdse projecten en dergelijke. Het schatten van kosten vergt een nauwkeurige analyse van de uit te voeren werkzaamheden en betrouwbare historische kostenoverzichten. Beide ontbreken nog te vaak in organisaties. Een aparte groep voor het verzamelen en interpreteren van kostencijfers zou hiervoor een oplossing kunnen zijn. Het schatten van opbrengsten is een taak van de gebruikers; automatiseerders hebben hierin te weinig inzicht. Verder spelen hier externe factoren en onderlinge correlaties een rol.

In de automatisering onderkent de auteur een combinatie van twee traditionele fouten: projecten worden geïsoleerd aangepakt en zijn gericht op de oplossing van een specifiek operationeel probleem. Hierdoor ontstaat het bekende probleem hoe diverse systemen op elkaar aan te sluiten om te komen tot geïntegreerde systemen zonder inconsistenties. Het gebruik van verschillende technieken bemoeilijkt de onderhoudbaarheid van systemen en heeft daardoor negatieve invloed op de betrouwbaarheid van de opgeleverde gegevens.

Door eilandautomatisering zijn er geen systemen beschikbaar voor de ondersteuning van de bedrijfsdoelen en/of ligt de nadruk op systemen die slechts een beperkte bijdrage leveren aan het succes van de organisatie.

Page-Jones staat de volgende methode voor om een gecoördineerde strategie te ontwikkelen voor de automatisering binnen organisaties:
1. het oprichten van een "Business Understanding and Strategy-group";
2. beschrijven van de bedrijfsprocessen en daaraan gerelateerde informatiebehoeften (gezichtspunten zijn hoe externe relaties de organisatie zien, welke doelen strategisch worden nagestreefd en welke operationele aspecten een belangrijke rol spelen);
3. opstellen van een technologiestrategie;
4. formuleren en uitvoeren van projecten.

(In vele seminars en cursussen over "Informatieplanning" worden op dit gebied methoden en technieken behandeld. PW.)

Section 2:. The DP project
De auteur start met enkele opmerkingen over de organisatie van projecten en het rekencentrum. Als uitgangspunt voor de verdere beschrijvingen wordt een matrix-organisatie genomen: hierbij zijn

"line managers" verantwoordelijk voor projecten en via een dwarsdoorsnede zijn de "staff managers" verantwoordelijk voor het ter beschikking stellen van kundig personeel per discipline c.q. fase van het project, het hanteren en bewaken van standaarden en dergelijke.

Project-management wordt beschreven als een cyclus van:

Plannen: gebaseerd op ervaring en de gehanteerde ontwikkelmethode wordt een planning van het project gemaakt, gespecificeerd per fase dan wel per deelproject.

Organiseren: taken worden aan de medewerkers toebedeeld.

Integreren: de individuen worden tot een gemotiveerd team gevormd.

Meten: continu dient de voortgang van de diverse deeltaken te worden bewaakt.

Bijstellen: bij afwijkingen ten opzichte van de planning dient bijsturing plaats te vinden.

Vanuit de projectplanning wordt een deadline vastgesteld ten aanzien van de oplevering van het gewenste systeem. Deadlines zijn echter niet altijd even realistisch. Dit kan als effect hebben dat de leiding de greep op het project verliest vanwege demotivatie bij de teamleden. Verder kan een te sterke nadruk op het halen van de deadline de kwaliteit van het opgeleverde produkt in de weg staan, wat nooit de bedoeling kan zijn. Het team moet tegen de druk vanwege deadlines worden beschermd: de leiding moet deze druk terugkaatsen naar de opstellers van de betreffende deadlines.

Onrealistische deadlines kunnen diverse oorzaken hebben. Overoptimisme is een eerste voorbeeld; kosten en doorlooptijd van een project worden te rooskleurig voorgesteld in het besluitvormingsproces. Verder verdient het geen aanbeveling deadlines op te stellen in een vroeg stadium van het project; het is dan nog niet duidelijk welke werkzaamheden uitgevoerd zullen moeten worden. Een regelmatig terugkerende tussentijdse bijstelling van de deadlines is dan ook noodzakelijk.

Ontwikkelmethoden geven een algemeen toepasbare projectplanning met detailbeschrijvingen per taak. Standaarden zijn (regels ten aanzien van) de praktische uitwerking van een ontwikkelmethode. De auteur gaat gedetailleerd in op de voor- en nadelen van het hanteren van methoden en standaarden.

Voordelen zijn:
- voor een nieuw project is reeds een kader beschikbaar;
- door splitsing van projecten in fasen kunnen specialisten per fase worden ingezet;
- uniformiteit bevordert uitwisselbaarheid van mensen;
- verbetering van onderhoudbaarheid van systemen;
- vereenvoudiging van opleidingsprogramma's en dergelijke.

Als nadelen worden vermeld:
- inflexibiliteit doorkruist het gezond verstand;
- de methoden/standaarden worden niet gebruikt;
- de standaarden worden intern tot doel verheven en staan daardoor in extreme gevallen de kwaliteit en efficiëntie in de weg.

Om de voortgang van een project te bewaken is een periodieke rapportage vereist inzake de status van de diverse lopende activiteiten. De nadruk dient hierbij te liggen op de afwijkingen ten opzichte van de planning. Een gedetailleerde weergave van de feiten, zodra de uitloop wordt geconstateerd, is hiervoor voldoende. Urenregistratie is hierbij een onmisbaar hulpmiddel; de gewerkte uren zijn basis voor betaling en doorbelasting van kosten en kunnen worden gebruikt om komende projecten te begroten. Aangetekend wordt dat urenregistratie een hulpmiddel is en geen doel op zichzelf.

Veel (kostbare) tijd gaat verloren met vergaderingen. De auteur geeft een aantal nuttige tips voor het verhogen van de effectiviteit van vergaderingen:

vooraf: zoek een bruikbare plek
maak vaste agenda
zorg dat iedereen inbreng heeft
tijdens: houd aan de agenda vast
notuleer het besprokene
na: noteer actiepunten en benoem personen die deze uitvoeren
verspreid de notulen.

Section 3: People; A DP Department's Greatest Resource
In deze laatste Section wordt een aantal personeelsaangelegenheden behandeld, waarop in deze samenvatting niet verder wordt ingegaan.

4 Conclusie

Samenvattend wil ik stellen dat Page-Jones met dit boek een aantal nuttige aanwijzingen geeft over het aanpakken van de problematiek rond de automatisering binnen organisaties. Hoewel sommige onderwerpen niet veel nieuws lijken te brengen, maken zijn taalgebruik en de hantering van praktijkvoorbeelden dit boek zeer lezenswaarding.

Doorbelasting van kosten van geautomatiseerde informatievoorziening

Een publikatie (uitgave van Kluwer in 1987 in de serie Informatie-management (ISBN 90.26.712375)) van het NIBIN onder redactie van: G.A.B. Janszen, G.B.M. Janzing, D. Purmer, E. de Vries en L. van der Zee

door: drs.ing. G. Swinkels

1 Inleiding

Het boek is een van de weinige Nederlandstalige publikaties over het doorbelasten van kosten van de informatievoorziening. Het is een vervolg op een rapport dat was samengesteld door een werkgroep van het NIBIN en dat in februari 1987 is verschenen onder dezelfde titel en met een vergelijkbare inhoud. In het boek wordt een relatie gelegd tussen de theorie en de praktijk van het doorbelasten van kosten van geautomatiseerde informatievoorziening. Het accent ligt op het doorbelasten van kosten van een centrale automatiseringsfunctie. De kosten die decentraal worden gemaakt komen rechtstreeks ten laste van de gebruikers.

2 Bespreking

In de eerste hoofdstukken wordt aandacht besteed aan de werkwijze van de commissie en de plaatsbepaling van het onderwerp. In het boek wordt het doel van doorbelasting als volgt omschreven: "Doorbelasting van de kosten van geautomatiseerde informatievoorziening wordt aanbevolen als instrument om de financiële gevolgen van ontwikkeling en gebruik van automatisering voor inhoud en uitvoering van bedrijfsfuncties zichtbaar en beheersbaar te maken". De relatie met andere beheersinstrumenten wordt hier niet verder uitgewerkt, waardoor de indruk zou kunnen ontstaan dat doorbelasting de enige mogelijkheid is om (gebruik van) capaciteit te beheersen.

De keuze van de doorbelastingssystematiek moet passen binnen de doelstellingen van de onderneming. Bij die keuze zijn verschillende factoren van belang, zoals consistent bedrijfsbeleid, bedrijfscultuur, afweging kosten en baten van de doorbelastingsmethodiek etc. Een voorwaarde voor doorbelasten is, dat de prestatie-eenheden die onderkend worden ook meetbaar zijn. Als de kosten niet echt worden doorbelast maar alleen kenbaar worden gemaakt, moet dezelfde systematiek worden gevolgd als bij daadwerkelijke doorberekening aan de gebruikers.

In hoofdstuk 4 worden de kosten van geautomatiseerde informatievoorziening behandeld. Voor het doorbelasten wordt aanbevolen om te werken met voorcalculatorische kosten om veelvuldige wisselingen te voorkomen en om inefficiënties te kunnen constateren. Dit is met name gebaseerd op het verbijzonderen van kosten volgens de kostenplaatsenmethode. Kostensoorten worden via (hulp)kostenplaatsen doorberekend naar de kostendragers. Omdat veel kostendragers voor de gebruiker niet zo veel zeggen (aantal CPU-seconden dat is afgenomen, aantal geprinte regels, hoeveelheid opgeslagen bytes etc.) wordt de term "werkeenheid" geïntroduceerd. Een werkeenheid is een produkt dat voor de gebruiker wel herkenbaar is. Ter vergelijking: in een restaurant staat op de menukaart niet hoeveel tijd de kok heeft besteed aan het klaarmaken, wat het bord heeft gekost en hoeveel een aardappel kost, maar een prijs per gerecht (bij de nouvelle cuisine ligt dat wat genuanceerder). Het toepassen van werkeenheden past in de ontwikkeling van rekencentrum naar service-centrum die veel automatiseringsafdelingen doormaken.

In hoofdstuk 5 wordt aangegeven hoe de theoretische uitgangspunten toegepast dienen te worden op de kosten voor ontwikkeling en onderhoud van informatiesystemen. De prestatie-eenheid die hier wordt gehanteerd zijn de uren die de mensen maken. Dit betekent het bepalen van de totale kosten en het aantal produktieve uren. Hieruit volgt een prijs per uur. Afhankelijk van het beleid kunnen uren van bepaalde functionarissen dan wel of niet worden doorbelast (bijvoorbeeld

systeembouwmedewerkers wel en de opleiders niet). De kosten van ontwikkeling kunnen ook gebruikt worden bij een afweging van "make or buy" voor een nieuw informatiesysteem.

Bij de behandeling van de doorbelasting van de kosten van de verwerking (hoofdstuk 6) blijkt dat verbijzondering van kosten bij automatisering enkele inherente problemen kent zoals gemeenschappelijke kosten, doorbelaste kosten voor (gebruik van) capaciteit, onduidelijke kostendragers en wisselende "produkt"-samenstelling. De dienstverlening van een computercentrum wordt uitgedrukt in prestatie-eenheden met rekenfactoren zoals aantal uitgevoerde instructies (verwerkingstijd), beschikbaarheid netwerk (aantal aansluitingen), gegevensopslag (ruimtebeslag (in Kb) per tijdseenheid) etc. De nadruk in dit hoofdstuk ligt met name op de vaststelling en toerekening van capaciteit en de gevolgen van nieuwe releases (besturingsprogrammatuur, firmware en hardware) voor de doorbelasting.

In het volgende hoofdstuk wordt het doorbelastingsmodel toegepast op een voorbeeld in de vorm van de organisatie "Eenvoud B.V.". De naam geeft al aan dat het hier om een eenvoudig model gaat met een beperkt aantal kostensoorten, -plaatsen en prestatie-eenheden. Er worden in dit voorbeeld geen werkeenheden genoemd. Het model dient als samenvatting voor de stof in de voorgaande hoofdstukken.

Verder zijn er nog enkele bijlagen opgenomen die op specifieke onderwerpen wat dieper ingaan zoals vaststelling capaciteit en berekenen beschikbare uren.

3 Opmerkingen

Doorbelasten van kosten is een middel om de geautomatiseerde informatievoorziening te beheersen. Door de kosten door te belasten naar de gebruiker die ze veroorzaakt zal die de afweging moeten maken of de kosten gerechtvaardigd zijn. Naast de bedoelde positieve effecten (inzicht in en beheersing van kosten) kan dat ook leiden tot negatieve effecten zoals suboptimalisatie en capaciteitsproblemen op langere termijn. Ook moeten we niet vergeten dat het systeem van doorbelasten zelf ook kosten met zich meebrengt. In het boek wordt te weinig rekening gehouden met het feit dat de methode van doorbelasten moet aansluiten op vraag en aanbod van automatiseringsdiensten (soort diensten, hoeveelheid, schommelingen in vraag en aanbod, mogelijkheden voor aanpassen vraag en aanbod), gevolgen voor de organisatie als geheel en het beleid van het management. In het boek worden wel aandachtspunten gegeven voor de keuze van de doorbelastingssystematiek, maar er wordt hoofdzakelijk gesproken over de genoemde methode van doorbelasten via de kostenplaatsenmethode. Andere (in sommige situaties betere) methoden worden af en toe genoemd maar worden niet verder uitgewerkt. Door de structuur van het boek te verbeteren zou dit al veel duidelijker worden en zouden de onderdelen beter op elkaar aansluiten. Het boek ontleent zijn waarde met name aan de vele praktische tips, aandachtspunten en praktische voorbeelden die van pas komen wanneer men een systeem voor de doorbelasting van kosten voor de geautomatiseerde informatievoorziening wil gaan opzetten.

ABC nieuws

Security modules

1 Inleiding

Volgens de Amerikaanse Treasury Directive on Electronic Funds and Securities Transfer Policy-Message Authentication (TD81-80), die vanaf 1 juni 1988 voor alle federale EFT-systemen van kracht is, dienen Electronic Funds Transfer-transacties (ook die op tape worden uitgewisseld) "properly authenticated" te zijn. Bij deze authenticatie wordt gebruik gemaakt van specifieke hardware, security-modules, die zorg dragen voor het uitvoeren van de voor authenticatie benodigde cryptografische functies. Het van kracht worden van deze richtlijnen en de daarmee samenhangende "U.S. Treasury Procedures for EFT Security Device Certification", vormen een goede aanleiding om kort aandacht te besteden aan de beveiliging van security-modules.

2 Beveiliging door cryptografie

Er zijn diverse vormen van gegevenstransport en opslag die een hoge mate van beveiliging rechtvaardigen. Hierbij kan onder andere worden gedacht aan het hierboven genoemde elektronisch betalen, PIN-verificatie, opslag van cryptografische sleutels of het geheim houden van bestanden op een PC. Maar er kan ook gedacht worden aan ander digitaal gegevenstransport, zoals fax en autotelefoon. De beveiliging betreft onder meer het vaststellen van de juiste herkomst en bestemming (authenticatie) van berichten, volledigheid en juiste berichtinhoud en bescherming tegen het afluisteren van berichten. Voor het beveiligen van het berichtenverkeer wordt in toenemende mate gebruik gemaakt van cryptografische technieken. Hierbij wordt ervan uitgegaan dat het communicatiemedium onvoldoende veilig is. Het cryptografisch algoritme is meestal algemeen bekend (bijvoorbeeld het DES-algoritme), zodat de beveiliging vooral steunt op het geheim houden van de sleutels waarmee de berichten worden vercijferd.

3 Hardware-matige oplossingen

Uit de noodzaak om de cryptografische sleutels en vercijferde berichten geheim te houden volgt dat de cryptografische sleutels nooit in klare, onvercijferde vorm beschikbaar mogen komen. Om deze hoge mate van beveiliging te realiseren is men ertoe overgegaan de cryptografische functies onder te brengen in specifiek daarvoor ontworpen hardware. Deze hardware, of de kern daarvan, wordt ook wel security-module genoemd.

De reden waarom niet wordt gekozen voor een software-matige oplossing binnen een host-computer wordt door Meyer en Matya, in hun lezenswaardige boek Cryptography, A Guide for the Design and Implementation of Secure Systems (1), als volgt verwoord: "When implemented in software, the boundaries of the cryptografic facility are not well-defined. In such cased, the physical protection achieved in hardware must now be achieved logically through programming. Ultimately, the degree of protection will depend on a processor's hardware protection features as utilized by the resident operating system (e.g., store and fetch protection, privileged operations, program execution modes, and the like). Thus the security of software implementations is no better than that of the underlying operating system".

4 Eisen waaraan een security-module moet voldoen

Het onderbrengen van cryptografische functies in een aparte security-module is feitelijk het verschuiven van het probleem van de host-computer naar een specifiek stuk hardware met de daarin

opgenomen software. Het is niet verwonderlijk dat de eisen die aan een security-module worden gesteld in de loop der tijd worden geformuleerd. Bijvoorbeeld, in een produktbeschrijving wordt verwezen naar standaarden waaraan de apparatuur voldoet, zoals de US Federal Standard 1027 "General Security Requirements for Equipment Using the Data Encryption Standard".

In deze standaard worden onder andere de volgende beveiligingseisen gesteld:
- het voorkomen van:
  . ten onrechte verzenden van onvercijferde berichten;
  . diefstal of ongeautoriseerde wijzigingen van security-modules;
  . ongeautoriseerde kennisname of wijziging van cryptografische sleutels;
  . het doorvoeren van cryptografische functies in geval van een hardware-storing: en
- de noodzaak voor:
  . het voorzien in de detectie van falende cryptografische functies.

In de "Treasury Procedures for EFT Security Device Certification" wordt door middel van een checklist aangegeven hoe getoetst kan worden of aan deze doelstellingen worden voldaan. Als voorbeeld dienen de volgende vragen:
- kan een security-module fysiek worden afgesloten met een slot;
- worden cryptografische sleutels automatisch gewist indien ongeautoriseerd toegang tot de security-module wordt verkregen;
- kan worden verzekerd dat niet-encrypte sleutels niet uitgelezen kunnen worden;
- wordt de pariteit van de ingevoerde sleutels gecontroleerd;
- blijven sleutels intact bij een stroomstoring;
- kunnen control characters worden onderscheiden van de rest van een bericht;
- wordt een signalering gegeven indien authenticatie mislukt;
- zijn beveiligingen ingebouwd en beschreven die zorg dragen voor het juist gebruik van de firmware;
- wordt voorzien in de volgende functies: standby mode, alarm reset, testmode, sleutelinvoer, geheugenwisfunctie;
- voorziet de apparatuur in de volgende status indicators: power on, DES bypass, test, battery, alarm, audible alarm, parity.

Naast vragen gericht op de fysieke beveiliging wordt in de Treasury Procedures aandacht besteed aan de voor security-modules te ontwikkelen software en documentatie daarvan in bijvoorbeeld diagrammen die de logica weergeven, timing diagrammen en de gehanteerde ontwikkelingsmethodiek.

De procedures gaan niet in op de authenticatie van de firmware die in een security box wordt geladen en evenmin op de mogelijkheid die een gebruiker heeft om vast te stellen dat met de juiste security-module wordt gecommuniceerd. Concluderend kan daarom worden opgemerkt dat er normen zijn voor het beoordelen van security-modules, maar dat deze een nadere uitwerking vereisen.

Overzicht van de cursussen Automatisering en Controle van KPMG Klynveld EDP Audit

In 1988 zijn de cursussen van KPMG Klynveld EDP Audit voor het eerst voor de open markt gegeven. Dit gebeurde in samenwerking met Kluwer Studiecentrum. Hierdoor zijn de cursussen op ruime schaal ook buiten de directe cliëntenkring van KPMG Klynveld ter beschikking gekomen. Vanwege de grote vraag naar cursussen uit het pakket, waaraan ook de cursus Controleleer van KPMG Klynveld Kraayenhof & Co. is toegevoegd, is besloten dit jaar voor het eerst ook in het voorjaar enkele cursussen te geven. Met name geldt dit voor de inleidende cursus Basiskennis Automatisering en Interne Controle (BAIC), voor de cursus Automatiseringsorganisatie (AUTO) en de cursus Systeembeoordeling (CASA). Ook de ondersteunende cursussen Administratieve Organisatie en Controleleer zullen in het voorjaar worden gegeven.
Het volledige pakket zal in het najaar 1989 opnieuw worden gegeven. Nieuw is daarbij de cursus Telecommunicatie. Deze cursus is tamelijk technisch en vormt een verdieping van de cursus Datacommunicatie. Waar de cursus Datacommunicatie vooral voor de algemeen accountant is gemaakt, daar is de cursus Telecommunicatie vooral voor de meer gespecialiseerde EDP auditor ontwikkeld. In een volgend nummer van Compact zullen we deze cursus nader introduceren.

Om een volledig beeld te krijgen van het cursuspakket inzake Automatisering en Controle, volgt hieronder het cursusschema.

Voorjaar
- Automatisering en Interne Controle (vierdaagse basiscursus)
  datum: 24 - 27 april 1989
- Automatiseringsorganisatie (inrichting en audit)
  datum: 8 - 10 mei 1989
- Systeembeoordeling (inrichting en audit van informatiesystemen)
  datum: 22 - 26 mei 1989
- Administratieve Organisatie
  datum: 13 - 17 maart 1989
- Controleleer
  datum: 29 mei - 2 juni 1989

Najaar
- Automatisering en Interne Controle (vierdaagse basiscursus)
  datum: 2 - 5 oktober 1989
- Automatiseringsorganisatie (inrichting en audit)
  datum: 9 - 11 oktober 1989
- Systeembeoordeling (inrichting en audit van informatiesystemen)
  datum: 30 oktober - 2 november 1989
  27 november - 1 december 1989 (onder voorbehoud)
- Databases (controle en beveiliging in een geïntegreerde omgeving)
  datum: 27 - 30 november 1989
- Datacommunicatie (controle en beveiliging in een netwerkomgeving)
  datum: 13 - 14 november 1989
- Telecommunicatie (techniek, controle en beveiliging)
  datum: 21 - 23 november 1989
- Administratieve Organisatie
  datum: 18 - 22 september 1989
  6 - 10 november 1989
- Controleleer
  datum: 23 - 27 oktober 1989
  20 - 24 november 1989

Voor nadere inlichtingen kunt u contact opnemen met Bureau Opleidingen van KPMG Klynveld (Pien Schepel, telefoon 020-5461733) of met Kluwer Studiecentrum (Dick Schuitema, telefoon 03465-60780), waar ook informatieve brochures verkrijgbaar zijn.

Overzicht hoofdartikelen 1987/1988

42	13e jaargang (nummer 3) zomer 1986	
	- Beveiliging: automatiserings- of organisatieprobleem	drs. H.C. Kocks
	- Conversie, Compilatie uit literatuur*)	drs. J. Kuipers
	- De microcomputer in de accountantscontrole*)	H. Veenman
43	14e jaargang (nummer 1) winter 86/lente 87	
	- Internationaal literatuuronderzoek naar computermisbruik in strafrechtelijk perspectief	mr. V.A. de Pous
	- Geïntegreerde gegevensverwerking	drs. H.C. Kocks
44	14e jaargang (nummer 2) zomer 1987	
	- Consequenties voor de beheersbaarheid ten gevolge van nieuwe technologische ontwikkelingen	A.W. Neisingh
	- Betalingsorganisatie en automatisering binnen de organisatie	J. ten Wolde
	- Electronic Banking-systemen in de praktijk	drs. D.M. Swagerman
45	14e jaargang (nummer 3) herfst 1987	
	- Escrow-depot voor computersoftware in Nederland	mr. V.A. de Pous
	- Beveiligen tegen computermisbruik	A.W. Neisingh en drs. J. Vossen
	- Geïntegreerde gegevensverwerking: Structuur van controle- en beveiligingsmaatregelen in een ADR/DATACOM DB-DC-omgeving	J.A.W. Winterink en drs. R.G.A. Fijneman
	- Belangrijke functies van een toegangsbeveiligingspakket	M.C. Duijm
46	14e jaargang (nummer 1) winter 1988	
	- SKE, Structured Knowledge Engineering	ing. A. van der Vlist
	- Beveiliging bij datatransmissie	ing. H.A.J.M. Spape
	- Electronic Funds Transfer het elektronisch uitvoeren van betalingen literatuurstudie	mw. ing. I.M. van Duin
47	15e jaargang (nummer 2) lente/zomer 1988 Special van de sectie Software Engineering	
	- De sectie Software Engineering, een inleiding	H. Veenman
	- Software Engineering	H. Veenman en ing. L.J.M.W. Gielen
	- Het testen van software	O. Kluyt
	- UNIX	ing. A. van der Vlist en ing. J.C. van Winkel RI
	- Computervirussen	ing. J.C. van Winkel RI
	- Objects	ing. L.J.M.W. Gielen
	- HyperCard	J. Schalk
	- Programmeertheorie	J. Schalk
	- Het Apple Talk netwerk, een beschouwing	J.L. Ramos Najera
	- PS/2 - OS/2	ir. J. de Graaff en drs. D.J.P. Witte
	- Elektronisch betalen, de betaalpas	ing. J. Rotteveel

*) Opgenomen in "24 over EDP-Auditing"