Informatiebeveiliging en recente IT-ontwikkelingen

Prof. Dr. J.J.A. Leenaars

THEMA

1 Introductie

Informatiebeveiliging is voor dit themanummer in de voetsporen van de 'Code voor Informatiebeveiliging' gedefinieerd in termen van Beschikbaarheid, Integriteit en Vertrouwelijkheid.[1]

Mij is gevraagd Informatiebeveiliging in het perspectief van recente IT-ontwikkelingen te willen beschouwen. Voorwaar een opgave. Wat zijn immers *de* recente IT-ontwikkelingen?

Het ontwikkelen van een houdbaar theoretisch raamwerk is in het kader van een artikel als het onderhavige niet haalbaar - zo al mogelijk. Ik zal derhalve na een korte inleidende beschouwing de probleemstelling op een vooral casuïstische manier uitwerken.

2 Inleidende beschouwingen

Beveiliging, beheersbaarheid en onderhoudbaarheid zijn geen synonieme begrippen; ze liggen inhoudelijk echter veel dichter bij elkaar dan menigeen denkt. Waarschijnlijk wordt veel onbegrip veroorzaakt door het feit, dat deze begrippen uit verschillende denk- en werkwerelden komen.

Voor bijvoorbeeld systeemontwikkelaars vormt de onderhoudbaarheid van applicaties die in een voor een 'ondernemingsomgeving' volkomen ongeschikte taal zijn geschreven een bron van toenemende zorg. Een EDP-auditor of een beveiligingsfunctionaris spreekt over 'gebrekkige beheersbaarheid' en bedoelt hiermee inhoudelijk voor een groot deel hetzelfde.

Bij de inzet van nieuwe IT-(hulp)middelen doet zich met betrekking tot deze aspecten dikwijls een 'déjà-vu'-gevoel voor. Waar bij 'ouderwetse' mainframes vele aspecten van beheersbaarheid, onderhoudbaarheid en beveiliging toereikend zijn in te vullen, is dit bij vele UNIX-omgevingen niet of veel minder het geval. PC's laat ik in dit verband maar beter onbesproken.

Samen met het empirisch vastgestelde feit, dat PC's in vele gevallen tot 'eilandautomatisering' hebben geleid, maar vrijwel nooit tot 'goedkoopte-eiland-automatisering' zal dit gebrek aan beheersbaarheid, onderhoudbaarheid en beveiliging naar mijn stellige overtuiging leiden tot centralisatie, natuurlijk gebruik makend van de (software) - technologie en beheersmechanismen van vandaag en niet van de mainframe-omgevingen, die wij uit de jaren zeventig en tachtig kennen. Deze centralisatie zal zich dan vooral uiten in standaardisatie en uniformiteit, vooral van 'interfaces'.

Gegeven immers een open communicerende wereld en de noodzaak het eigen systeem te beheersen, zal de nadruk moeten liggen op de beheersing van de 'interfaces' die toegang geven tot het eigen systeem. Men vergelijke in dit verband de deur die toegang geeft tot een pand.

In het navolgende ga ik in op een drietal recente ontwikkelingen:

- client-server computing;
- vercijfering en sleutelbeheer;
- het koppelpunt (de interface) tussen interne en externe netwerken, oftewel de zogenaamde firewalls.

Prof. Dr. J.J.A. Leenaars studeerde bedrijfseconomie aan de Hogere Economische School te Rotterdam (1973) en accountancy bij het NIVRA (1978). Hij is lid van het Beleidscomité en de Raad van Bestuur van de Robeco Groep. Hij promoveerde in 1993 en is parttime hoogleraar aan de Universiteit van Amsterdam. Zijn leerstoel is Bestuurlijke Informatieverzorging.

3 Client-server computing

Geautomatiseerde gegevensverwerking bestaat in zijn algemeenheid uit het communiceren, verwerken, opslaan en presenteren van gegevens.

Bij 'client-server computing' worden sommige van deze elementaire functies van geautomatiseerde gegevensverwerking uitgevoerd door de 'client' - meestal een Personal Computer of meer algemeen een Workstation - en andere functies door de 'server'. Deze server of 'afdelingscomputer' is meestal een 'UNIX-box', maar kan ook een PC zijn. Het komt ook voor dat een 'klassiek' mainframe als server dient.

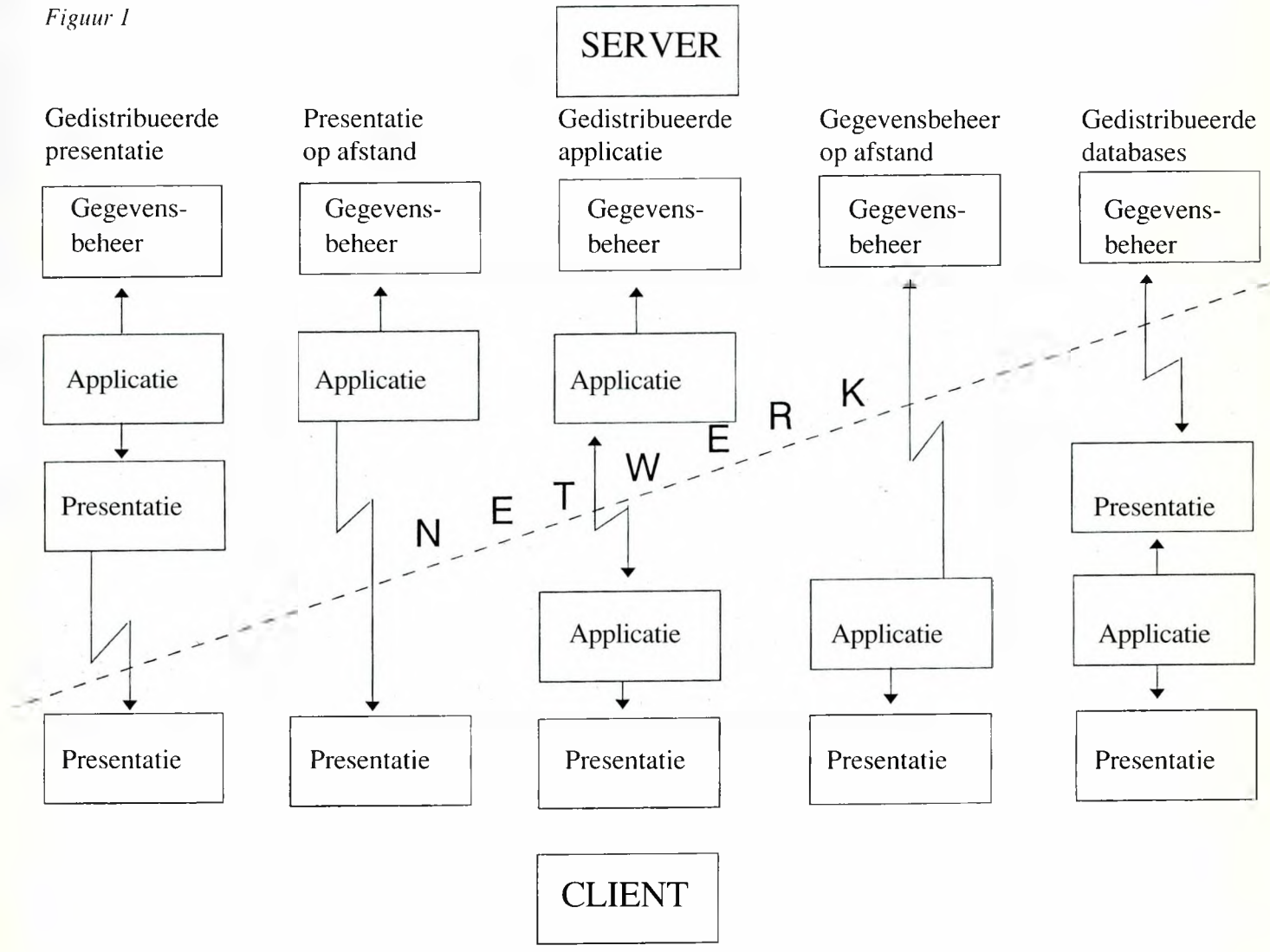
Een gebruikelijke, van de Gartner Group afkomstige indeling is weergegeven in figuur 1.

Dit schema laat zien dat deze verschillende basisfuncties van geautomatiseerde gegevensverwerking (dus: communiceren, opslaan, verwerken en presenteren) over verschillende computers, die of als 'client' of als 'server' in een netwerk optreden, kunnen worden gespreid.

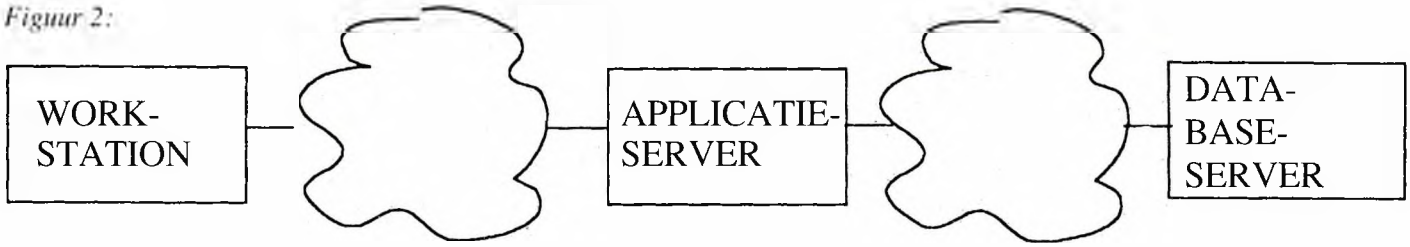
Indien en voor zover de uitvoering van applicatieprogrammatuur op de 'client' plaatsvindt, wordt de integriteit van de gegevensverwerking en van corporate gegevens bedreigd, in essentie omdat op PC-clientniveau *alle* systeemonderdelen ('resources') per definitie ten volle ter beschikking staan van de (eind)gebruiker.

Vermeldenswaard in dit verband is de aangekondigde zogenaamde netwerk- of $500-computer, die geschikt zal zijn voor Internet-verkeer 'vanuit de woonkamer', en die in beginsel alleen presentatiefuncties bevat. De applicaties draaien dan op beveiligde servers, zodat deze ontwikkeling vanuit beveiligingsoptiek een welkome is.

Figuur 1



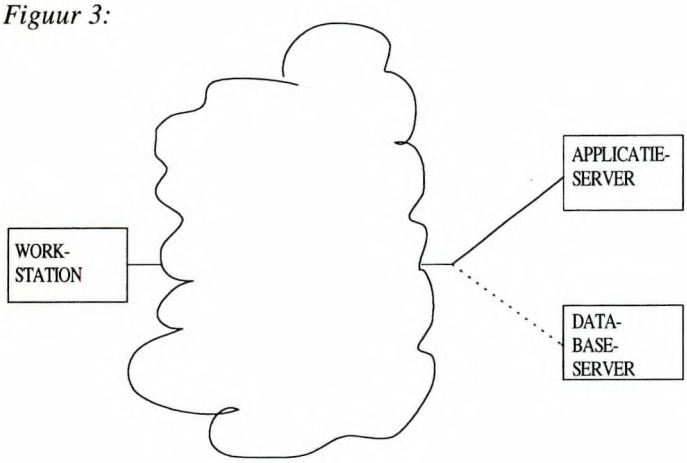
Figuur 2:



Eindgebruikers

Bedreigingen van de integriteit van informatie kunnen ook indirect van aard zijn, hetgeen het volgende voorbeeld illustreert. (zie figuur 2)

Figuur 3:



Een mij bekende grote complexe bancaire applicatie is gerealiseerd in een netwerk van RS/6000's, die als servers worden ingezet, en van PS-2 werkstations die als client dienen.

Om beveiligingsredenen emuleren deze PS-2's een 'domme terminal', in het geval dat ze aan deze applicatie zijn aangelogd.

De applicatie maakt gebruik van gescheiden applicatie- en database-servers. Als database wordt Oracle gebruikt; het 'besturingssysteem' is AIX (UNIX).

Om succesvol aan te kunnen loggen aan de Oracle-database dient men een bestaande combinatie van user-id en password in te toetsen. Men krijgt dan de beschikking over alle rechten voor gegevensgebruik, die voor de betreffende gebruiker samen met de gebruikersidentificatie en het bijbehorende password zijn vastgelegd in autorisatietabellen in de database.

De toegang kan in principe worden verkregen via verschillende applicaties of via hulpmiddelen als - bijvoorbeeld - SQL*Plus of SQL*ReportWriter of Excell, terwijl per gebruiker slechts één groep toegangsrechten (zogenaamde 'privileges') kan worden vastgelegd. Die rechten moeten voldoende zijn voor de applicatie, waarvoor de gebruiker de meeste rechten nodig heeft. Deze rechten kunnen veel te ruim zijn voor andere applicaties of 'tools', die de betrokken gebruiker mag starten.

Het gewenste concept blijkt uit figuur 2; zonder nadere maatregelen bestaat echter een situatie, die (via de gestippelde lijn) geschetst is in figuur 3.

Voor de bancaire applicatie is dit probleem als volgt opgelost:

Een gebruiker logt aan AIX aan op de applicatie-server. Vervolgens komt hij terecht in een panel, waarin een keuze kan worden gemaakt uit een aantal opties. Wordt gekozen voor de besproken applicatie dan wordt via een zelf ontwikkeld programma het aanloggen aan de Oracle-database verzorgd. Daartoe wordt de user-id veranderd en wordt een - voor gebruikers geheim - password uit een vercijferd bestand opgehaald. Hierna wordt een menu beschikbaar gesteld met de voor de betreffende gebruiker toegestane functies.

Omdat de passwords onbekend zijn kan niet op een andere manier aan de betrokken database worden aangelogd met hetzelfde user-id.

In Oracle versie 7 is dit probleem opgelost; er kan van zogenaamde 'application roles' gebruik worden gemaakt. Dit houdt in dat in de database rollen kunnen worden gedefinieerd, die met een password worden beveiligd. Aan gebruikers kunnen één of meer rollen worden gekoppeld.

Aan een rol kunnen toegangsrechten zijn toegekend terwijl voorts in de corresponderende applicatie de benodigde functies kunnen worden beveiligd. De gebruiker kan, mits een rol aan zijn user-id is gekoppeld, met de betreffende rol via de bewuste applicatie aanloggen aan de database.

In de besproken applicatie-omgeving moet aan AIX worden aangelogd (op de applicatie-server),

waarna, via een panelkeuze, de verbinding met de database (op de databaseserver) 'automatisch' wordt verzorgd.

Als voor- en nadelen van deze zelfontwikkelde beveiliging kunnen worden genoemd:
Voordelen:
- de gebruikers weten niet met welk password voor hen wordt aangelogd op database/gegevensniveau;
- de betreffende passwords worden automatisch maandelijks gewijzigd, hebben een lengte van meer dan 15 posities en worden vercijferd opgeslagen;
- in AIX kunnen password-regels en wijzigingsfrequenties worden afgedwongen.

Nadelen:
- er wordt een zelfontwikkelde module gebruikt. Deze moet worden onderhouden;
- indien men de gegevens in de database ook voor andere toepassingen wil gebruiken dan moeten daar aparte user-id's voor worden aangemaakt.

Voor de standaardoplossing die gebruik maakt van 'application roles' gelden de volgende voor- en nadelen:
Voordelen:
- een rol kan alleen via de applicatie worden geactiveerd en een gebruiker heeft alleen rechten voor die rol als deze aan zijn user-id is gekoppeld;
- application roles zijn een standaardfunctie van Oracle (vanaf release 7); softwareonderhoud berust derhalve bij de leverancier.

Nadelen:
- de gebruiker logt aan Oracle aan. Vergeleken met AIX biedt Oracle niet of nauwelijks ondersteuning bij het beheer van passwords. Een password van één positie wordt al geaccepteerd.
  Een systeemprogramma om op gecontroleerde en beveiligde wijze het password te wijzigen ontbreekt. Een beter password-beheer vergt weer eigen ontwikkeling;
- ook het onderhoud van de passwords van de roles wordt niet door Oracle verzorgd. Er is geen programma of menu voor regelmatige aanpassing en voor synchronisatie van de wijzigingen in de applicatie en de database. Dit wordt geheel aan de automatiseringsorganisatie overgelaten.

De generieke conclusie die uit het voorgaande voorbeeld volgt, is dat 'kritische' applicatieprogrammatuur niet op onveilige werkstations behoort te draaien, respectievelijk dat toegangsbeveiliging op 'servers' op gegevensniveau moet plaatsvinden.

Niet-eindgebruikers

Ook de autorisatie van niet-eindgebruikers (bijvoorbeeld automatiseringspersoneel) op servers is anno 1996 veel minder gemakkelijk implementeerbaar dan bij mainframes het geval is. Voor eenvoudige handelingen als het maken van een backup-kopie is het 'hoogste' password nodig.

Via zelfgeschreven 'scripts' kunnen evenwel vele van deze onvolkomenheden worden gerepareerd.

Hoewel een dergelijke invulling met behulp van 'scripts' aan redelijke eisen voldoet, is het toch niet de ideale implementatie van toegangsbeveiliging. Deze bestaat immers uit de volgende componenten:
- een database waarin de regels met betrekking tot de toegang tot willekeurig welke 'resource' zijn opgenomen;
- een autorisatieprogramma, dat de enige toegang vormt tot de database met toegangsregels;
- een mechanisme (in het besturingssysteem) waarmee wordt afgedwongen dat een applicatie altijd aan de database met toegangsregels refereert.

Soortgelijke problematiek kan zich ook op het niveau van de applicatie-architectuur voordoen.

Sommige krachtige ontwikkelingshulpmiddelen, die ook voor een eindgebruiker uitermate nuttig kunnen zijn, kennen geen scheiding van beheersactiviteiten, die door gebruikers respectievelijk 'automatiseerders' behoren te worden uitgevoerd; alle beheer is 'één pot nat'.

Een dergelijk hulpmiddel dat wordt uitgeprobeerd 'rondom' de eerder besproken grote, complexe, bancaire applicatie is 'Business Objects'. Dit is een hulpmiddel voor eindgebruikers waarmee ad hoc vraagstellingen op databases kunnen worden geformuleerd.

Business Objects kent alleen een lees-mogelijkheid ('read-only'): hier is met andere woorden primair het vertrouwelijkheidsaspect uit de definitie van Informatiebeveiliging aan de orde.

In Business Objects definieert men een drietal soorten gebruikers; de 'machtigste', de manager, kan en mag alles en is daarin niet te beperken.

Elke 'omgeving' (Universe) wordt door een manager gedefinieerd, inclusief onder meer autorisatie en database-objecten binnen zijn omgeving.

De autorisatie op gegevensniveau is nu niet goed te realiseren: als een gebruiker aan de database refereert, gebeurt dit niet met zijn eigen user-id, maar met een user-id/password van een Universe ('query-account'). De nadelen hiervan zijn dat aan de database-kant onbekend is wie aan het werk is en dat geen autorisatie voor een gebruiker binnen de database gebruikt kan worden (bijvoorbeeld door middel van zogenaamde views).

Het lijkt mogelijk om een Business Objects-gebruiker met zijn eigen user-id en password toegang tot de database te verlenen, maar dit levert extra beheershandelingen op, omdat voor iedere gebruiker privé-synoniemen voor te raadplegen objecten in de database moeten worden gedefinieerd. Er is bovendien geen synchronisatie van de Business Objects-passwords en de Oracle-passwords.

De oorzaak van deze problematiek is eigenlijk een simpele: beheersing en beveiliging zijn geen ontwerpcriteria geweest bij het ontwikkelen van dit hulpmiddel.

4 Vercijfering (encryptie) en sleutelbeheer

Bij het betreden van de wereld van vercijfering van elektronische berichten en sleutelbeheer stuiten wij niet alleen op een verzameling van nieuwe begrippen, zoals bijvoorbeeld Message Authentication Codes, Non-repudiation (onweerlegbaarheid) en de digitale handtekening, maar ook op oeroude beginselen uit de bestuurlijke informatieverzorging rondom het gescheiden bewaren van (delen van) sleutels.

In het navolgende behandel ik kort de toegang tot systemen met behulp van een password en de verzending van elektronische berichten (EDI) via een in beginsel openbaar netwerk.

Toegang tot systemen: passwords

Passwords zijn als een paspoort: ze legitimeren, ze authenticeren. Ze bewijzen dat je degene bent voor wie je je uitgeeft te zijn. Omdat de meeste netwerken via een 'store and forward'-beginsel werken kan een password worden afgeluisterd, indien het onbeschermd door een netwerk reist.

Er zijn twee principieel verschillende mogelijkheden om dit gevaar te beteugelen: gebruikerspasswords reizen nooit of te nimmer over een netwerk en passwords worden reeds op locatie versluierd met een vercijferingsalgoritme.

Een mooi voorbeeld van niet-reizende gebruikers-passwords kan worden gevonden in het netwerk van MIT te Boston. De 'authentication', de legitimering vindt plaats middels een mechanisme dat de veelzeggende naam Kerberos (Grieks: hellehond) draagt.

Een basispatroon dat steeds terugkomt is de vercijfering van reeds anderszins vercijferde gegevens. Alle vercijfering vindt met behulp van het DES-algoritme plaats.

Centraal in het netwerk is een authentication/autorisatie-server opgesteld: het Key Distribution Center. In dit KDC zijn alle user-id/password-tabellen voor de authenticatie alsmede tabellen met de bevoegdheden per gebruiker vastgelegd. Beveiliging van deze server is derhalve een hoofdstuk apart.

Het beveiligingsmechanisme werkt in beginsel als volgt:

Een gebruiker toetst zijn user-id en een specificatie van de verlangde netwerkservice - bijvoorbeeld toegang tot een bestand op een server - in.

Deze boodschap wordt naar het KDC geleid. Na bevoegdheidscontrole: 'Is deze gebruiker bevoegd tot de gewenste toegang?', genereert het KDC een 'ticket' - een boodschap - waarin onder meer de gebruikersnaam, de client/werkstation-identificatie en een willekeurig gegenereerde sessiesleutel[2] worden opgenomen. Dit ticket wordt vercijferd met de sleutel van de server en/of het bestand waartoe toegang wordt gezocht. De sessiesleutel wordt nogmaals toegevoegd. Het geheel - dus zowel ticket als toegevoegde sessiesleutel - wordt nogmaals vercijferd, nu met de gebruikerssleutel, waarna verzending naar het werkstation van de gebruiker plaatsvindt.

Na ontvangst van de laatste boodschap vraagt het werkstation de gebruiker om zijn password, vertaalt dit naar een DES-sleutel en 'ontcijfert' het (nog met de server/bestandssleutel vercijferde) ticket en de sessiesleutel. Het password wordt

onmiddellijk gewist uit het werkgeheugen van het werkstation.

De volgende stap, het verkrijgen van toegang tot de gespecificeerde server/bestand, gaat ten slotte als volgt.

Het werkstation creëert een 'message authenticator', waarin tijdstip van creatie, gebruikersnaam en werkstation-adres worden opgenomen. Deze message authenticator wordt vercijferd met de sessiesleutel.

Het ticket en de message authenticator worden naar de gewenste server verzonden. De server - die beschikt over haar eigen toegangssleutels - 'ontcijfert' het ticket en vindt daarin de sessiesleutel; met deze sleutel wordt de message authenticator ontcijferd. Na controle op inhoudelijke gelijkheid van identificatiegegevens in ticket en message authenticator wordt de toegang verleend.

Verzending van berichten over publieke netwerken

Berichten die over - in beginsel - publieke netwerken reizen moeten aan een aantal eisen voldoen. Kernbegrippen hierbij zijn:
- integriteit;
- exclusiviteit;
- authenticiteit (éénduidige afkomst);
- 'non-repudiation' (onweerlegbaarheid, niet-te-ontkennen).

Non-repudiation is een nieuw begrip, dat van groot belang is.[3] Het heeft twee kanten: indien A een bericht uitwisselt met B, kan A niet ontkennen het bericht te hebben verstuurd en kan B niet ontkennen het bericht te hebben ontvangen.

Vercijfering vindt plaats met behulp van encryptie-algoritmen; deze kunnen symmetrisch en a-symmetrisch[4] zijn. Symmetrische algoritmen zijn snel uitvoerbaar maar geven bij grotere aantallen gebruikers al snel sleutelbeheerproblemen. A-symmetrische algoritmen zijn qua performance veel meer belastend maar creëren een veel geringer sleutelbeheerprobleem.

In geavanceerde toepassingen vindt een gemengd gebruik van symmetrische en a-symmetrische vercijferingsalgoritmen plaats.

Een voorbeeld van dit gemengd gebruik dat in de nabije toekomst naar het zich laat aanzien zeer breed zal worden toegepast kan worden gevonden in de zogenaamde 'digitale handtekening'.

Bij dergelijke toepassingen wordt de vercijfering van het bericht zelf niet noodzakelijk geacht; wel dient in bijvoorbeeld betalingsverkeer onweerlegbaar vast te staan dat het bericht van een positief geïdentificeerde zender/klant afkomstig is. Bovendien dient het bericht niet modificeerbaar - dus integer - te zijn. Een ander bedrag op een dagafschrift als op het schermpje van de pin-automaat: 'bedrag akkoord?', 'toets ja' is immers meer dan vervelend.

Het vercijferingsproces verloopt als volgt:

Van de oorspronkelijke boodschap wordt een hash-totaal berekend met behulp van een 'one way DES-mechanisme'.[5]

De betrokken DES-sleutel wordt willekeurig gegenereerd. Het hash-totaal wordt vercijferd met de geheime (RSA = a-symmetrische) sleutel van de zender.

Dit vercijferde hash-totaal en de DES-sleutel (in klare tekst!) worden aan het bericht toegevoegd, waarna verzending plaatsvindt.

In beginsel kan iedereen die over de publieke (RSA)sleutel van de zender/klant beschikt het bericht - voorzover vercijferd - ontcijferen en de integriteit ervan nagaan, door met behulp van het one way DES-mechanisme de berekening van het hash-totaal te herhalen en gelijkheid te constateren (bij ongelijkheid kan er overigens iets aan zowel de authenticatie als aan de integriteit van de boodschap mankeren).

De doelstelling van deze vorm van vercijfering is dan ook niet de vertrouwelijkheid van het bericht zelf, maar de integriteit (i.c. het niet gemodificeerd zijn) en de onweerlegbare identificatie van de afzender van het bericht.

Zoals reeds opgemerkt geven symmetrische vercijferingsalgoritmen bij grote aantallen gebruikers al snel sleutelbeheerproblemen; zulks in tegenstelling tot a-symmetrische algoritmen.

Vooral de geheimhouding van sleutels veroorzaakt beheersbaarheidsproblematiek. Indien 4 partijen onderling communiceren met behulp van een symmetrisch vercijferingsalgoritme moeten er 6 geheime sleutelparen bestaan: ieder der partners moet 3 sleutels (en die van zichzelf) geheim houden. Men stelle zich de problematiek bij reeds enkele honderden partijen voor!

Bij a-symmetrische versleuteling heeft ieder der partners slechts de zorg één sleutel geheim te houden: de eigen geheime (RSA-)sleutel; alle andere sleutels zijn immers publieke RSA-sleutels en vergen derhalve minder zorg.

Dit betekent overigens niet dat er *geen* zorg aan deze publieke sleutels hoeft te worden besteed. Door malafide verwisseling van de openbare sleutel van een bonafide partij door een openbare sleutel van een door de malafide partij gegenereerd RSA-sleutelpaar kan de malafide partij zich voordoen als de bonafide partij. Dit laatste probleem wordt wel 'masquerading' genoemd.

Sleutelbeheer

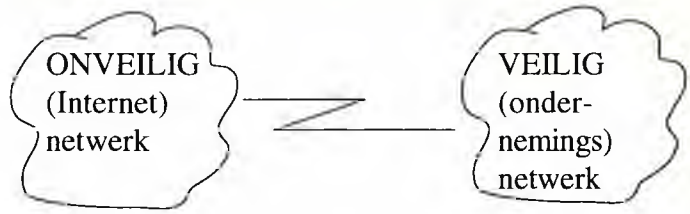
Het sleutelbeheer betreft een serie taken, waarbij traditionele functiescheiding van belang blijft. Sleutelbeheer en berichtenbeheer en -verkeer dienen aan primaire functiescheiding onderworpen te zijn.

Bij het sleutelbeheer in omgevingen met veel kritisch berichtenverkeer is onder meer het volgende van belang:
- het sleutelbeheer is dusdanig belangrijk, dat het een lijnfunctie betreft;
- het mechanisme waarmee sleutels of sleutelparen worden gegenereerd dient geheim te zijn;
- sleutelbeheer dient te impliceren dat een sleutel een eindige levensduur heeft; er dient derhalve te worden voorzien in de creatie, de distributie, de opslag, het gebruik, de vervanging en de vernietiging van sleutels en sleutelparen;
- soms bestaat er een sleutelhiërarchie: werksleutels voor berichtenverkeer, opslag- of transportsleutels om de werksleutels op te slaan en/of te distribueren, en een supersleutel om de opslag- en distributiesleutels op te slaan en/of te distribueren.

5 Firewalls

Firewalls zullen zich de komende jaren in een toenemende belangstelling mogen verheugen, vooral onder invloed van het steeds groter wordende gebruik van Internet voor zakelijke toepassingen. In beginsel ontstaat bij het gebruik van Internet de situatie die in figuur 4 wordt geschematiseerd.

Figuur 4:



In deze situatie is ook het veilige netwerk niet veilig meer.

De serieuze bedreigingen die uit Internet voortvloeien doen zich voor bij het creëren van een 'site' in WWW; World Wide Web wordt welhaast als synoniem voor Internet gebruikt, maar is in feite een van de toepassingen die gebruikmaakt van Internet.

WWW is een de facto standaard voor berichtenverkeer met als doel informatie opgeslagen in computers te transporteren over Internet en op een gebruikersvriendelijke manier te presenteren op de computer van de aanvrager van die informatie. Het WWW is de basis van de vele mogelijkheden die het Internet biedt.

Om het Internet te kunnen inzetten als communicatie- en/of distributiekanaal moet een onderneming een koppeling tot stand brengen tussen haar eigen netwerkinfrastructuur en de netwerkinfrastructuur van het Internet.

Deze koppeling impliceert echter ook een groot beveiligingsrisico voor de kritische informatie die opgeslagen is in de computers van de betrokken onderneming, omdat als er geen maatregelen worden getroffen, in principe de 'gehele wereld' toegang heeft tot die informatie.

Een van de maatregelen die een onderneming kan nemen is het inrichten van een 'firewall' tussen het in principe onveilige Internet en het veilige ondernemingsnetwerk.

Het doel van de 'firewall' is het voorkomen van ongewilde toegang van derden tot het ondernemingsnetwerk en de daarop aangesloten computers met de kritische bedrijfsinformatie. Impliciet wordt hiermee dus ook geregeld welke informatie wel door derden mag worden gebruikt. In feite is dit de informatie die de onderneming vanuit haar doelstellingen juist wil aanbieden aan de gebruikers van het Internet. Een ander doel van een 'firewall' is het reguleren van de toegang van de medewerkers van de onderneming tot het Internet.

Een 'firewall' kent verschillende implementaties; deze zijn afhankelijk van het niveau van beveiliging dat de onderneming wil implementeren en de daarbij behorende kosten.

Firewalls maken gebruik van 'routers'; dit zijn computers die als functies hebben het koppelen van netwerken onderling en het transporteren van gegevens tussen die netwerken.

Een relatief eenvoudige firewall is een firewall die boodschappen filtert.

Meestal wordt voor het transporteren van gegevens het de facto standaardprotocol TCP/IP gebruikt.

De te transporteren gegevens kunnen afkomstig zijn van verschillende soorten applicaties, bijvoorbeeld:

- terminalverkeer;
- bestandsoverdrachtverkeer;
- elektronische boodschappenverkeer;
- etc.

Bij het gebruik van TCP/IP als transportprotocol wordt een bepaald type verkeer gerelateerd aan een zogenaamde 'socket'. Terminalverkeer bijvoorbeeld heeft socketnummer 23. Op deze manier wordt het type verkeer onderscheiden en kan de ontvanger bepalen voor welke toepassing de boodschap bedoeld is.

Om te bepalen waar de gegevens naar toe moeten worden getransporteerd 'leest' de router het dataverkeer en bepaalt het adres van de geadresseerde ontvanger, het IP-adres in TCP/IP.

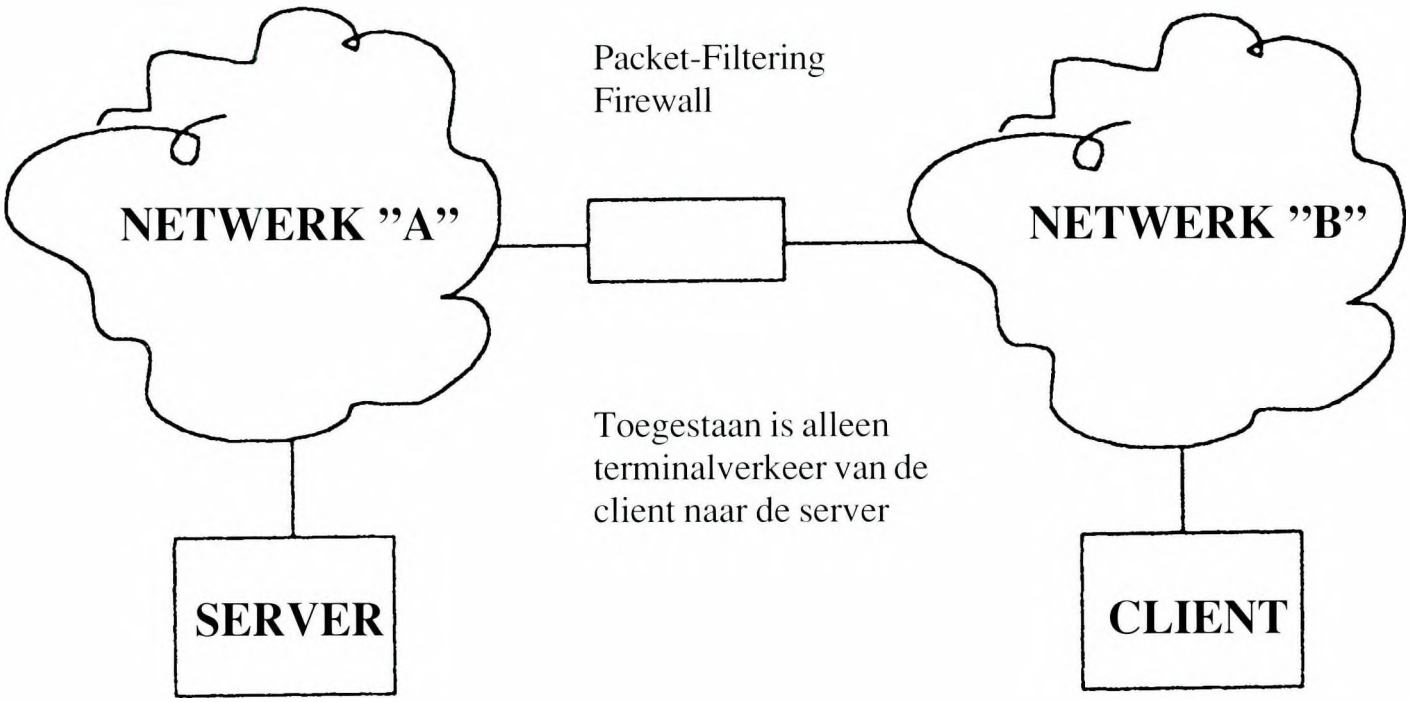
De implementatie van een 'packet-filter firewall' nu bestaat uit een router waarbij adressen in het netwerkverkeer worden getoetst aan een lijst van toegestane adressen. Verkeer bestemd voor servers en/of clients met adressen die niet zijn toegestaan wordt niet door de router getransporteerd maar wordt 'vernietigd'.

Daarbij biedt deze vorm van firewalls ook nog de mogelijkheid om selectief typen verkeer wel of niet te transporteren voor de reeds toegestane adressen.

Er wordt bijvoorbeeld alleen terminalverkeer maar geen bestandsoverdrachtverkeer toegestaan naar een bepaalde server. De router filtert dan het terminalverkeer voor deze server uit het dataverkeer, laat dit door en vernietigt al het andere verkeer dat aan deze server is geadresseerd. Zie figuur 5.

Een van de meest veilige maar ook duurdere implementaties van een 'firewall' is de zogenoemde 'screened subnet firewall', waarbij tussen het Internet en het veilige ondernemingsnetwerk een apart netwerk wordt gecreëerd dat middels routers aan de twee eerstgenoemde netwerken is gekoppeld. Het doel van deze implementatie is het voorkomen van doorgaand netwerkverkeer van het

Figuur 5:



Internet naar het ondernemingsnetwerk en vice versa. Om dit te realiseren worden in het aparte netwerk computers (bastion hosts) geplaatst waarop de informatie staat die de onderneming beschikbaar wil stellen aan de gebruikers van het Internet. Deze computers fungeren ook als intermediair tussen de medewerkers van de onderneming die toegang willen tot het Internet via WWW-applicaties zoals bijvoorbeeld Netscape, en het Internet. De routers die het aparte netwerk koppelen aan het Internet en het ondernemingsnetwerk fungeren als portier, waarbij de router aan de Internet-kant al het verkeer tegenhoudt dat afkomstig is van het ondernemingsnetwerk en de router aan de ondernemingsnetwerkkant al het verkeer tegenhoudt dat afkomstig is van het Internet (zie ook figuur 6).

6 Slotopmerkingen

De voortbrenging van producten en diensten wordt onder meer onder invloed van toenemende (internationale) concurrentie, van steeds klantspecifiekere behoeftenbevrediging en van uitbesteding steeds complexer. Hierdoor neemt de noodzaak tot coördinatie - en dus van communicatie - toe.

De Informatietechnologie van vandaag stelt ons in staat om communicatie tussen en met computers - waar ook ter wereld - tegen zeer lage kosten te realiseren.

Netwerken waartoe 'iedereen' toegang heeft, moeten als principieel onveilig worden gekwalificeerd.

In dit artikel kwam een aantal meer specifieke aspecten, die binnen netwerken spelen, aan de orde.

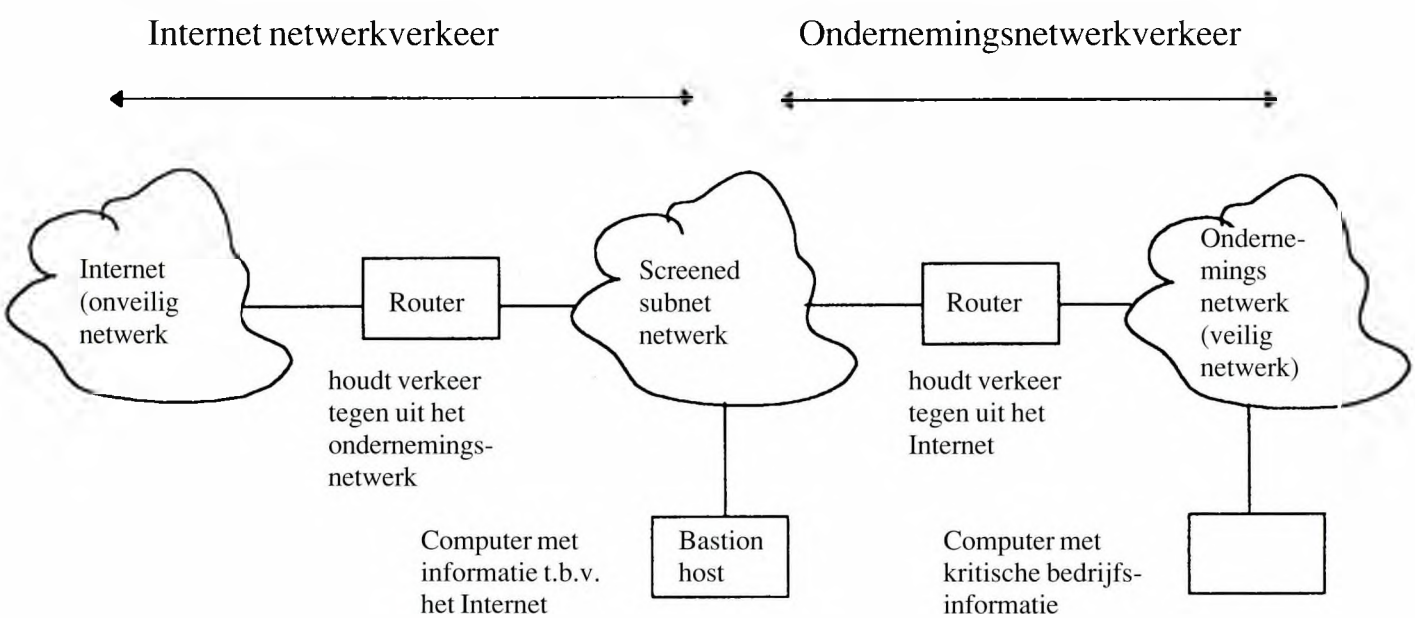
Met betrekking tot het 'client-server'-model werd geconcludeerd, dat applicatieprogrammatuur die 'kritische' betekenis heeft, niet op werkstations behoort te draaien, en dat de toegangsbeveiliging op 'servers' op gegevensniveau moet plaatsvinden.

Ter zake van vercijfering en sleutelbeheer kwam naar voren dat traditionele functiescheiding een noodzakelijke plaats inneemt: het beschikken over en het bewaren van vercijferingssleutels dient strikt te worden gescheiden van de beschikking over de inhoud van berichtenverkeer, maar ook bijvoorbeeld van netwerkbeheer. Ik acht sleutelbeheer van zodanig belang, dat deze taken niet bij bijvoorbeeld een 'security-administrator' moeten worden ondergebracht, maar dat sprake is van taken, die binnen lijnfuncties moeten worden uitgeoefend.

Rondom de koppeling van interne en externe netwerken ten slotte kwamen 'firewalls' als antwoord op dreigende inbreuken op de informatiebeveiliging aan de orde.

Deze beveiliging zal nooit 100% zijn. Indien een organisatie aan derden toegang geeft tot haar netwerk, dient er van te worden uitgegaan dat er vroeg of laat zal worden ingebroken.

Figuur 6:



Naast de opzet van een adequate netwerkbeveiliging zijn daarom de volgende attentiepunten van belang:
- het opstellen van procedures hoe te handelen indien een inbraak voorkomt c.q. wordt geconstateerd;
- het treffen van maatregelen om de beveiliging te bewaken.

Bij deze bewaking dient rekening te worden gehouden met extra organisatorische en technische (exploitatie)kosten. Deze ontstaan bijvoorbeeld door:
- het toewijzen van het netwerkbeveiligingsaspect aan een functionaris. Deze zal hieraan al snel een groot gedeelte van zijn tijd besteden;
- het treffen van (technische) voorzieningen, waarmee op continue basis kan worden vastgesteld of een (poging tot) inbraak plaatsvindt;
- het koppelen van deze technische monitoring aan menselijke acties;
- het auditen op continue basis. Een jaarlijkse beveiligingsaudit lijkt niet voldoende;
- het uitvoeren van offensieve testen van het netwerk naast uiteraard defensieve testen.

Informatiebeveiliging blijft een boeiend vakgebied.

De uitdaging van informatiebeveiliging is meestal niet gelegen in het ontstaan van nieuwe problemen; steeds nieuwe informatietechnologie vraagt echter om steeds nieuwe oplossingen.

Informatiebeveiliging is daarom een vakgebied, waarin veel creatieve activiteit moet worden ontplooid, en zeker geen vakgebied, dat stoffig, saai en niet spannend is.

LITERATUUR

Zboray en Lett (Gartner Group), (1995), *Security for the Enterprise*, september.

Schiller, (1994), Secure Distributed Computing, *Scientific American*, november.

De Groot, (1996), De toepassing van cryptografische technieken bij EDI-verkeer, *de EDP-Auditor*, jaargang 5, nummer 1.

Leenaars, (1993), *Functiescheidingen in hooggeautomatiseerde omgevingen*, Samsom BedrijfsInformatie.

NOTEN

1 Zonder op deze plaats tot formele definities te willen komen denke men bij beschikbaarheid aan de mate waarin een gebruiker 'op zijn moment' een beroep kan doen op een (geautomatiseerd) systeem, bij integriteit aan de mate waarin informatie een waarheidsgetrouwe afbeelding van de werkelijkheid vormt en bij vertrouwelijkheid aan de mate waarin informatie uitsluitend aan daartoe bevoegde functionarissen beschikbaar wordt gesteld.

2 De sessiesleutel heeft slechts beperkte geldigheid, bijvoorbeeld gedurende één uur.

3 Wat bijvoorbeeld te doen indien op basis van een EDI-bericht 100 dozen wijn besteld zijn en de afzender van het bericht later stelt het bericht niet te hebben verzonden, of slechts 10 dozen te hebben besteld?

4 Bij symmetrische algoritmen vindt vercijfering en ontcijfering met behulp van dezelfde sleutel plaats; a-symmetrische algoritmen maken bij vercijfering van een andere sleutel gebruik dan bij ontcijfering.

5 Zoals het begrip aangeeft is deze functie niet omkeerbaar, maar wel herhaalbaar!